

MASTER'S THESIS

DEVOPS - SHARED GOAL AND SHARED UNDERSTANDING

Collaborative teams

Nadorp, T (Timo)

Award date:

2020

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us at:

pure-support@ou.nl

providing details and we will investigate your claim.

Downloaded from <https://research.ou.nl/> on date: 09. Sep. 2021

Open Universiteit
www.ou.nl



DEVOPS - SHARED GOAL AND SHARED UNDERSTANDING

Collaborative teams

Degree programme: Open University of the Netherlands, Faculty of Management, Science & Technology

Business Process Management & IT master's programme

Degree programme: Open University of the Netherlands, Faculty of Management, Science & Technology

Business Process Management & IT master's programme

Course: IM0602 BPMIT Graduation Assignment Preparation
IM9806 Business Process Management and IT Graduation Assignment

Student: Timo Nadorp

Identification number:

Date: 21-06-2020

Thesis supervisor Prof. dr. ir. J.J.M. Trienekens

Second reader MSc Michiel van Belzen (PhD candidate)

Version number: AF Version 4.9

Status: Definitive

Abstract

An organization that adopts DevOps creates a different context. A transition demands a different approach. DevOps has several dimensions, one of them is collaboration. This research focuses on DevOps teams. An important element of collaboration in a DevOps team is the multifunctional nature of the team. This research investigates the influence of a shared goal and shared understanding on collaboration in a DevOps context. Collaboration, and its main aspects, shared goal and shared understanding are redefined for a DevOps context and the relevant literature is discussed. The conceptual model provides sub aspects for every main aspect. Furthermore, the conceptual model is operationalized by defining a set of indicators. The impact of a shared goal and shared understanding on collaboration in a DevOps context is investigated via a case study. The conclusion is that these main aspects have a positive influence, resulting in more collaborative teams. The conceptual model portrays which sub aspects influence that process and the data corroborates and amends the relationships and indicators. Organizations should not underestimate the impact of shared goal and shared understanding on the collaboration of a DevOps team.

Key terms

Shared understanding, Shared Goal, Collaboration and DevOps.

Summary

DevOps has several dimensions, among them: collaboration. There has been a lot of research regarding collaboration in organizations. However, DevOps introduces a new mindset. That creates a new context, which could influence how collaboration can be optimized. This research is scoped to collaboration on the team level, in an attempt to contribute to the knowledge of how teams can be more effective.

Chapter 1 introduces the subject. The definitions and discussion of collaboration, shared goal and shared understanding in a DevOps context are presented in Chapter 2, which also incorporates a conceptual model. That model is operationalized and investigated in practice; the research method is described in Chapter 3. The model can be reused by other researchers and organizations. Chapter 4 depicts the results of the data analysis and Chapter 5 presents the conclusion and discussion.

This research strives to answer what the influence of shared goal and shared understanding is on collaboration in a DevOps context. The conceptual model is built by answering what collaboration, shared goal and shared understanding in a DevOps context entails. Collaboration is defined as cross-functional team members that execute activities in an attempt to fulfill a goal. The literature suggests that there are two main-aspects, which influence the way teams collaborate: shared goal and shared understanding.

Furthermore, every main-aspect is influenced by three sub-aspects. For shared goal these are IT Leadership, organizational structure and shared vision and cadre. For shared understanding these are construction, co-construction and constructive conflict. Chapter two puts these terms in perspective, which results in a conceptual model.

The conceptual model is tested via a case-study. Therefore, it is a qualitative research, with an exploratory focus. The research design is based on a mixed-methods approach, consisting of a semi-structured interview and content analysis. In order to operationalize the conceptual model, every sub-aspect has several indicators. These indicators and the conceptual model are used to create a semi-structured interview protocol. During the interview phase, seven respondents share their perspective. The raw data will be, via coding, transformed to information that can be compared against the conceptual model.

That comparison shows that the conceptual model is partly true. The direction seems to be accurate. The impact of shared goal and shared understanding on collaboration in a DevOps context is corroborated. However, not every relation in the conceptual model is fully accurate. For instance, IT leadership seems to have a more powerful impact than assumed. While respondents do not agree regarding the extent of the influence of constructive conflict. Lastly, the term; organizational structure, seems to conflict with some of the values that team members have.

In conclusion, an organization that adopts DevOps creates a different context. That demands a new approach. Organizations that are considering that change should consider what that means for the teams and how they can enable that they have a shared goal and a shared understanding. Based on this research these main-aspects have a positive influence on collaboration in a DevOps context. Additional research will be required to deepen the knowledge and strengthen the scientific proof.

Contents

Abstract	2
Key terms	2
Summary	3
Chapter 1: Introduction	7
1.1 Exploration of the topic	7
1.2 Motivation & Relevance.....	8
1.3. Problem statement	9
1.4. Report structure.....	9
Chapter 2: Literature Review	11
2.1 Literature review approach	11
2.2 Implementation	23
2.2.1 What is collaboration?	23
2.2.2 Shared goal	25
2.2.3 Shared understanding.....	28
2.2.4 Relationships.....	32
2.3 Conceptual model.....	33
Chapter 3: Research design	35
3.1 Research approach.....	35
3.2 Case study	36
3.3. Data collection	37
3.4 Data analysis	41
3.5 Validity, Reliability & Triangulation.....	44
Chapter 4: Results	47
4.1 Research implementation.....	47
4.2 Research subject – Contextual information.....	47
4.2.1 Organization.....	47
4.2.2 Business Units	48
4.2.3 Teams	48
4.2.4 Respondents	49
4.3 Research results - Interviews	50
4.2.1 Indicators	51
4.2.2 Sub-aspects	55
4.2.3 Main-aspects.....	59
4.4 Research results – content analysis	63

4.5 Theory and Data.....	64
4.6 Sub research question 4.....	65
4.7 Sub research question 5.....	67
Chapter 5: Conclusion, discussion, recommendations and reflection	69
5.1 Conclusions	69
5.2 Discussion.....	69
5.3 Practice recommendations	71
5.4 Recommendations for further research	72
5.5 Reflection	74
References	76
Appendix	83
Appendix 1: Literature study	83
1.1 Criteria enforcement – Per search engine	83
1.2 Search queries – First set	85
1.3 Search queries – Second set	92
1.4 Valuable sources	97
1.5 Selected sources – Building blocks method.....	104
1.6 Selected sources – Result per search engine.....	106
1.7 ABDC & Scimago	107
1.8 Building blocks – Per research question	108
1.9 Snowballing methods.....	110
1.10 Selected literature – Building blocks & snowballing method	116
Appendix 2: Method	120
2.1 Unstructured interview.....	120
2.2 Interview tables	121
2.3 Interview question – Reasoning.....	128
2.4 Interview protocol – Shared goal.....	136
2.5 Interview protocol – Shared understanding	142
2.6 Letter of consent.....	148
2.7 Introduction Letter.....	150
Appendix 3: Coding	151
3.1 Coding – Indicators	151
3.2 Coding – Sub-aspects	165
Appendix 4: Interviews & content analysis.....	170
4.1 Results - Indicators.....	170
4.2 Results – Sub-aspects.....	199

4.3 Results – Main-aspects	208
4.4 Results – Content analysis	231
4.5 Results – Theory and Data	243

Chapter 1: Introduction

Information Technology is an ever-changing sector. The complexity of the products is growing, while the demand for new applications, updates and improvements are skyrocketing (Lee & Xia, 2010). This means that an IT organization has to produce better, faster and cheaper software. Which was impossible with the best practices of the '90s and the '00s. Aligning all the stakeholders and creating an outcome that suites everyone is easy to set as a goal and extremely difficult to achieve.

Lately, there have been several studies (Denning, 2015; Morgan, Richey & Autry, 2016) that indicate the importance to adapt quickly to changes in the market. Most IT organizations deliver IT services to customers. These services often are perpetual and the customers might change their demands over time. Another important insight is that a customer will expect quality from a recurring service (Erich, Amrit & Daneva, 2017). IS Development and IT Operations (Iden, Tessem, & Paivarinta, 2012) need to be more collaborative if they want to meet the demands of the customer. The main focus will be on investigating the main-aspects of collaboration within a DevOps context.

1.1 Exploration of the topic

During the last decades, a new way of working, called Agile, obtained attention and spread its beliefs. As a result, many organizations have (tried) to reform into a more adaptable organization (Lee & Xia, 2010). Agile is often categorized as a mind-set. Currently there is a new set of best practices emerging from the shadows, that seems to builds on the Agile mind-set (Lwakatare, Kuvaja, & Oivo, 2015; Claps, Svensson, & Aurum, 2015), which is called DevOps. Agile is aiming for iterative development of releasable (software) components, emphasizing the need for feedback loops, transparency, flexibility and interactive inspection (Lee & Xia, 2010). DevOps is known for the phrase: 'if you make it you own it'. This way of thinking forces the development team to look further than the release date (Claps et al., 2015). Therefore, it is a concept that includes the entire lifecycle of an IT-product (Ghezzi, 2017; Erich et al., 2017) and helps the business to adapt to the ever-changing business environment (Rong, Zhang & Shao, 2016).

The relationship between Agile and DevOps is intriguing and ambiguous. They can, or maybe should (Davis & Daniels, 2015; Hoda & Murugesan, 2016), co-exist together. The subject of Agile and DevOps is to complex and broad to handle in one research paper. In an attempt to understand the distinction of the terms it is useful to discuss the concept of DevOps. According to Humble & Molesky (2011) the concept of DevOps can be described as a practice meant to align the incentives of developers, testers and operations regarding software delivery. The focus lies on delivering high quality software as fast as possible (Penners & Dyck, 2015), which is achieved by including the collaboration dimension (Ghezzi, 2017; Lwakatare et al., 2015). According to Lwakatare, Kuvaja & Ovio (2016) practitioners have two sets of definitions for DevOps. One group stresses the importance of reliable software products and services that can be delivered faster by allowing constant interactions (Lwakatare et al., 2016), transparency and cross-functional collaboration (Lwakatare et al., 2015; 2016). The other group focuses more on the means, by utilizing advanced automation and reassessing the roles and responsibilities that are required for delivering software (Lwakatare et al., 2016).

Penners & Dyck (2015) defined DevOps as a mindset that encourages cross-functional collaboration, especially between development and IT operations (Iden et al., 2012), in order to create resilient and adaptable systems. Lwakatare et al. (2016) stated that DevOps is not just a mindset, but rather a pattern of practices described by practitioners. One of the observations is that organizations need to

reexamine the way they think about roles and responsibilities. These roles should be focused on enabling the performance of the entire system or service, instead of the performance of a specific department or software component. In this research we define DevOps context as a department or organization, where a collaborative mindset is implemented and DevOps practices are advocated in an attempt to deliver and maintain a high quality of IT solutions.

There are several conceptual models who aimed to describe the goal and dimensions of DevOps. Among them are Lwakatare et al. (2016), Giudice & Condo (2017), Lwakatare et al. (2015) and Humble and Molesky (2011). All these researchers agree that DevOps is a multidimensional concept. Intensification of collaboration in the daily work of development and operations is necessary to increase the impact of implementing DevOps (Lwakatara et al. 2016). A statement that is strengthened by Lwakatare et al. (2015), Ghezzi (2017) and Humble and Molesky (2011) who mention collaboration as one of the dimensions of DevOps. Lwakatare et al. (2016, pp. 95) describe the dimension of DevOps as “the process of rethinking and reorientation of roles and teams in development and operations activities”. Others, like Humble and Molesky (2011) and Ghezzi (2017) give similar definitions, that stress the importance of aligning the way of working.

Main-aspects of the dimension; collaboration, are a shared goal (Jassawalla & Sashittal, 1998; Vangen & Huxham, 2012; Eldor, 2019; Kwak & Anbari, 2009; Polat, Lynn, Akgün & Emre, 2018) and shared understanding of cross-functional responsibilities (Bittner & Leimeister, 2014; Van den Bossche, Gijsselaers, Segers, Woltjer & Kirschner, 2011; Iden & Bygstad, 2018; Berggren & Johansson & Baroutsi, 2017). In this research the dimension, collaboration, will be operationalized in order to investigate it thoroughly. The research is scoped to the effects of a shared goal and a shared understanding of cross-functional responsibilities (in short: shared understanding) on collaboration in a DevOps context.

1.2 Motivation & Relevance

Organizations that operate in an IT Service market are often building long lasting relationships (Sau & Narisawa, 2016). Therefore, the main focus is on adapting to the needs of the customer, in order to reduce the chance of attrition. If an organization wants to survive, it will need to keep a substantial portion of the clientele that it has.

Thongpapanl, De Clercq & Dimov (2012) stress the importance for future research regarding (cross-functional) collaboration and its relationship with adaptability, which is seen as one of the main reasons to introduce DevOps (Lwakatare et al. 2015; Ghezzi, 2017). By focusing on collaboration, and the main-aspects that influence it, this research strives to do two things. Broaden the theoretical fundament of DevOps and shed light on an apparent practical problem. Little is known about the dimensions of DevOps. Lwakatare et al. (2016) and Humble & Molesky (2011) stress the importance for further empirical evidence of DevOps practices, in organizations that have implemented the concept. According to them, the dimensions of DevOps lack scientific corroboration. Collaboration in a DevOps context presents a way to connect IS development and IT operations, but it lacks a shared definition and scientific agreement regarding the main-aspects that influence it (Bass, Weber & Zhu, 2015).

The suggestion explored in this research is that teams that operate in a DevOps context will perform better when there is attention to the benefits of improving collaboration. This study will help organizations to decide whether it is worth to invest more in collaboration within a DevOps context. The goal is to deliver more insight in the inner mechanics of collaboration and to broaden the

fundament of theoretical knowledge. The relation between DevOps and collaboration has not been researched thoroughly yet, while practitioners often assume that this relationship exists (Ghezzi, 2017, Lwakatare et al., 2016). Tessem & Iden (2008) stress the importance of further research to understand the way Development and Operations (should) collaborate. They state that several investigations will increase the pallet of choices, which will help an IT organization in defining an approach that best fits their specific situation.

Bittner & Leimeister (2014) advice future researchers to further examine shared understanding, especially in heterogenous groups or teams. They see room for improvement regarding the implementation and execution of shared understanding of cross-functional responsibilities. Jassawalla & Sashittal (1998) mention the relevance of reassessing their work, regarding a shared goal, in the context of a cross-functional organization. While Aronson, Shenhar & Patanakul (2013) emphasize concepts like leadership and vision. One of the core concepts of a DevOps context is cross-functional collaboration, therefore this research connects to the recommendations of Bittner & Leimeister (2014) and Jassawalla & Sashittal (1998).

Recent scientific literature suggests that collaboration is a key dimension of DevOps (Humble & Molesky, 2011; Lwakatare et al., 2016; Ghezzi, 2017). This study attempts to create more insight by assessing the impact of a shared goal and shared understanding on, one of the commonly accepted dimensions of DevOps, collaboration.

1.3. Problem statement

As stated before, the IT sector is an ever-changing sector. The software delivery patterns are changing rapidly nowadays, in an attempt to be more successful. Organizations need to be adaptable and customer focused (Denning, 2015; Morgan et al., 2016; Rong et al., 2016). The research goal is to contribute to the development of adaptable teams that deliver software at the pace that customers require. Collaboration seems to be an important enabler for this situation.

The relationship between the main-aspects of collaboration, shared goal and shared understanding, and the dimension of DevOps, collaboration, has not been established by substantial scientific research. This research aims to explore that dimension, by investigating the influence of the aforementioned main-aspects on collaboration in a DevOps context. It strives to answer the following main research question: *What is the influence of a shared goal and a shared understanding of cross functional responsibilities on collaboration in the context of DevOps?*

The following sub research questions have been formulated to answer the main research question:

1. What is collaboration in the context of DevOps?
2. What is a shared goal in the context of DevOps?
3. What is a shared understanding of cross functional responsibilities in the context of DevOps?
4. How does a shared goal influence collaboration in a DevOps context?
5. How does a shared understanding of cross functional responsibilities influence collaboration in a DevOps context?

1.4. Report structure

This report strives to answer the research question that has been presented in this chapter. Chapter 2 encapsulates the literature review, focusing on theories regarding (the main-aspects of) collaboration: shared goal and shared understanding. The end result is a conceptual model that is presented at the end of chapter 2. The first three sub research questions are the foundation of the

model. Chapter 3 focuses on the methodology that is used for this research. It will explain which research methods are used to enable the scientific research and to increase reliability and viability. The data collection and data analysis process is explained there as well. Chapter 4 focuses on the results of the data analysis and compares the conceptual model against the collected data. Chapter 5 will provide the conclusion and discussion, answering the research question and reflecting on the scientific relevance of this study. The limitations of this research and the recommendations for other research will also be presented here.

Chapter 2: Literature Review

The previous chapter incorporates the provocative, motivation and goal of this study. The main research question is presented and the subject is briefly introduced. The main focus is to put collaboration in perspective for an organization that delivers software in a DevOps context. To comprehend the mechanics involved a conceptual model has been created based on existing scientific knowledge.

In this section the priority will lie on analyzing the existing literature. This chapter incorporates three parts. Section 2.1 introduces the approach that was adopted to perform the literature review. Section 2.2 describes the relevant literature that has been selected in Section 2.1. Section 2.3 describes the conceptual model that will be tested during this research.

2.1 Literature review approach

This section describes and justifies the research approach that has been selected to perform the selection and usage of scientific literature. It is founded upon the theories and guidelines of Saunders, Lewis and Thornhill (2016) and Wolfswinkel, Furtmueller & Wilderom (2013).

Wolfswinkel et al. (2013) have developed a grounded theory with four sequential phases. These phases have been used as a mainstay to select suitable literature. The foundation of the research is strengthened by following this structure, because all the references are inspected in a similar fashion. To ensure this structure the first phase focuses on defining criteria to scope the research. Wolfswinkel et al. (2013) encourage to do this in four steps:

1. Inclusion and exclusion criteria;
2. Fields of research;
3. Selection of sources;
4. Search terms.

After the prerequisites have been formulated the second phase can be executed, which is the search process to gather a list of potential sources with relevance for the study. The third phase is the selection process, which aims to select the sources that fulfill the prerequisites of phase one. The last phase strives to analyze the worthiness of the selected sources. In the following paragraphs these phases are used to explain the research approach of this study.

Phase 1 - Definition

By applying the right filters, the quality of the assembled literature can be approved. Table 2.1 shows the search criteria and fields of research. In this case the fields of research are part of the inclusion criteria. The inclusion criteria are important for finding the most relevant literature.

Table 2.1: Search criteria

Criteria	Norm	Reason for using criteria
Limited to	Peer-reviewed	The literature has been reviewed on quality and applicability, making the source more reliable. The bias is reduced by this action.

Criteria	Norm	Reason for using criteria
Accessible	Source fully available online	The source should be, in full, available online, in order to evaluate the relevance.
Content type	Academic journal (e)Book Conference papers	The academic journal is the preferred option. Good alternatives are (e)books and conference papers. All should be scientific by nature and peer reviewed.
Research area	Business Computer science Management	This research focuses on business and IT concepts.
Date of publication	2001-present	DevOps started to gain attention around 2009. Before that other concepts, like agile, were gaining attention. Therefore, this research looks a little bit further back. Starting from the beginning of this century. Exceptions can be made when a source is older, but has a high relevance and scientific foundation. The focus, of course, will be on recent literature.
Language	English	The common language for scientific literature is English. It is the most accessible language for researchers, including this researcher.

Literature sources (search engines)

Several search engines have been used to create a complete set of literature. It reduces the effect that one library can have on the outcome of this research. The library of the Open University has only been used to scan for articles. If an article could only be found in this library, then it was excluded for the research. Table 2.2 presents the search engines that have been used to search for viable literature:

Table 2.2 search engines

Search engine	Informatica / Computer science	Business / Management
Academic Search Elite (EBSCO host)	Yes	Yes
Business Source Premier (EBSCO host)	No	Yes
JSTOR	No	Yes
Web of Science	Yes	Yes

The first column shows the name of the search engine. The other two columns tell which kind of information can be found by utilizing the engine. For instance, Web of Science has literature in the research area; informatica / computer science, but also in business / management. The search in Academic Search Elite and Business Source Premier was done simultaneously by using EBSCO Host. The ACM digital library has several articles that focus on the automation dimension of collaboration; therefore, this library is excluded as a search engine. Appendix 1.1 depicts how the criteria were enforced.

Search strategy and terms

The first step is to search based on keywords using the building blocks method. The first sub research question is focused on DevOps and collaboration, the second on shared goal, the third on shared understanding. The other two questions combine key words that have been used for the first

three sub research questions. Based on this, the following keywords have been used per sub research question:

1. DevOps, Collaboration
2. Shared Goal, DevOps,
3. Shared Understanding, DevOps
4. DevOps, Collaboration, Shared Goal
5. DevOps, Collaboration, Shared Understanding, Cross-functional

For all search queries the inclusion criteria, depicted in table 2.1, are used as a search requirement. Appendix 1.2 presents four extensive tables that visualize the search queries and used criteria per search engine.

Sadly, these keywords did not result in a substantial set of research papers for sub research question 2, 3, 4 and 5. The lack of scientific references for sub research question 4 and 5 have been accepted. These main-aspects of the main research question will be answered by the result section of this research. The main reason for the low quantity of references for sub research question 2 and 3 seemed to be the limited amount of research regarding DevOps. Combining keywords with the DevOps building block results in a too narrow set of scientific literature. Therefore, the keyword DevOps has been omitted for sub research question 2 and 3, and the keyword collaboration has been added, broadening the search towards collaboration in general.

In an attempt to search in the right direction, the keyword; cross-functional has been added for sub research questions 2 and 3. DevOps teams are cross-functional and by adding this element the papers regarding collaboration are steered towards the direction that resembles the DevOps context. This resulted in the following set of keywords:

1. DevOps, Collaboration
2. Shared Goal, Collaboration, Cross-functional
3. Shared Understanding, Collaboration, Cross-functional

Table 2.3 depicts the four search-queries that have been applied at first for the three sub research questions. For set 1 the first step, row 1 of table 4, is to search in the title and/or abstract for DevOps and in the full text for collaboration. The second step for set 1 (row 2) is to do this the other way around. The third step (row 3) is added to find the most interesting papers and search for both terms in the abstract and/or title. This will not give new sources, but gives insight regarding the resemblance to the research subject. The fourth step (row 4) is to search for both keywords in the full text (no specifications like title or abstract). After these steps, the other key words are used as full-text add-ons when there is the need for extra specification or diversification. The complete set of search queries is attached in appendix 1.3.

Table 2.3 - most relevant search queries

SRQ 1				SRQ 2				SRQ 3				
		Title	Abstract	Full-text		Title	Abstract	Full-text		Title	Abstract	Full-text
1	DevOps	X	X		Shared goal	X	X		Shared understanding	X	X	
	Collaboration			X	Collaboration			X	Collaboration			X
2	DevOps			X	Shared goal			X	Shared understanding			X

	<i>Collaboration</i>	x	x		<i>Collaboration</i>	x	x		<i>Collaboration</i>	x	x
3	<i>DevOps</i>	x	x		<i>Shared goal</i>	x	x		<i>Shared understanding</i>	x	x
	<i>Collaboration</i>	x	x		<i>Collaboration</i>	x	x		<i>Collaboration</i>	x	x
4	<i>DevOps</i>			x	<i>Shared goal</i>			x	<i>Shared understanding</i>		x
	<i>Collaboration</i>			x	<i>Collaboration</i>			x	<i>Collaboration</i>		x

For set 2 and 3 the approach is similar. The first four search queries are build up with shared goal and collaboration and shared understanding and collaboration. After that, the other keywords are used to specify or diversify, see sub research question 2 and 3 in appendix 1.3. The keywords that are distilled from the sub research question form the base here. Therefore the AND operator is preferred, when combining the keywords. This research prefers to cite articles that had one or more of the key concepts of this research as their research subject. That chance increases when the first few searches are in title and abstract instead of full text.

A second method is utilized when it is hard to retrieve a substantial set of data based on the database searches. This method is the snowball method. There are two forms, backward snowballing and forward snowballing. In this research backward snowballing is utilized, because the research subject, DevOps, is a young research subject. Therefore, it was deemed important to utilize the backward snowballing method, to find relevant literature that pioneers used to support their claims. For research question 2 forward snowballing is utilized, due to an older very relevant source. Table 2.7, 2.8 and 2.9 show the results of these methods.

The references that do not meet the criteria depicted in table 2.1 are excluded immediately. If the references, found by applying the building blocks method, meet the criteria they were analyzed and deemed relevant or irrelevant. When the sources on a subject are limited the backward snowballing method is applied, which will be explained later on.

Phase 2 - Search

By using the aforementioned key words a list of the valuable sources was constructed. The literature was found by using the keywords and inclusion criteria that have been introduced in phase one. Therefore, every search was conducted by using combinations of the keywords. The complete set of literature has been attached in appendix 1.4.

Phase 3 - Selection

To distillate the most valuable sources all 87 references were assessed by analyzing the abstract. The outcome was binominal, a first set was rejected for further analyzation and a second set was deemed interesting enough to investigate. The rejection was based on the fact that the abstract did not show an overlap with the research. At this point there were already 49 references rejected. For the remaining 38 sources the theoretical framework and the conclusion were interpreted. Most of them were deemed valuable for the research. One was removed as a source. The removed source had similar keywords, but the theoretical model and conclusion showed no relevance to this research. The source had a theoretical model that focused on financial elements, which had no chance of contributing to this research.

The execution of the search took place in December 2018. By applying the criteria, formalized in the previous section, a selection was made by using the building blocks method. The criteria of table 2.1 were used as filters in the search engines, further explained in appendix 1.1. Therefore, all the 37 sources that were selected using the building block method, depicted in appendix 1.5, meet these requirements. Appendix 1.5 shows the authors and titles, the numbers of the first column correspond to the ID's of Appendix 1.6. Appendix 1.6 presents where and in how many databases the sources were found.**A**

The literature of appendix 1.5 is the first set literature, which will form the basis of this research. Table 2.4 gives an insight in the scientific influence of these papers. It helps to assess the quality of the used literature.

Table 2.4: The influence of journals

Nr	Journal	ABDC	Publisher	Scimago Journal - Q1-Q4 (SJR, 2017)
1	British Journal of Management	A	John Wiley & Sons, Inc.	-
2	Academy of Management Review	A*	Academy of Management	Q1 (sjr: 7.88)
3	Information Systems Journal	A*	John Wiley & Sons, Inc.	Q1 (sjr: 1.75)
4	Creativity and Innovation Management	C	John Wiley & Sons, Inc.	Q2 (sjr: 0.75)
5	Theoretical Issues in Ergonomics Science	-	Taylor & Francis Online	Q2 (sjr: 0.42)
6	Journal of Management Information Systems	A*	Taylor & Francis Online	Q1 (sjr: 2.49)
7	International Journal of Project Management	A	Elsevier	Q1 (sjr: 1.46)
8	Supply Chain Management: An International Journal	A	Emerald Group Publishing	Q3 (sjr: 0.24)
9	Group Organization Management: an international journal	A	Sage	Q1 (sjr: 1.25)
10	Journal of High Technology Management Research	C	Elsevier	Q2 (sjr: 0.46)
11	Information and Software Technology	A	Elsevier	Q2 (sjr: 0.58)
12	Management Decision	B	Emerald Group Publishing	Q1 (sjr: 0.54)
13	Journal of Management Studies	A*	John Wiley & Sons, Inc.	Q1 (sjr: 3.80)
14	Journal of Software: Evolution and Process*	-	John Wiley & Sons, Inc.	Q3 (sjr: 0.23)
15	Ecology and Society	-	Resilience Alliance	Q1 (sjr: 1.73)
16	Journal of Software: Evolution and Process*	-	John Wiley & Sons, Inc.	Q3 (sjr: 0.23)
17	Information Systems Journal	A*	John Wiley & Sons, Inc.	Q1 (sjr: 1.75)
18	Information and Software Technology	A	Elsevier	Q2 (sjr: 0.58)
19	Academy of Management Review	A*	Academy of Management	Q1 (sjr: 7.88)

20	Journal of Applied Psychology	A*	American Psychological Association	Q1 (sjr: 4.69)
21	International Journal of Project Management	A	Elsevier	Q1 (sjr: 1.46)
22	Information Systems Research	A*	The Institute for Operations Research and the Management Sciences (INFORMS)	Q1 (sjr: 3.16)
23	International Journal of Project Management	A	Elsevier	Q1 (sjr: 1.46)
24	MIS Quarterly	A*	Association for Information Systems (AIS)	-
25	Journal of Organizational Behavior	A*	John Wiley & Sons, Inc.	Q1 (sjr: 3.15)
26	International Journal of Project Management	A	Elsevier	Q1 (sjr: 1.46)
27	International Journal of Physical Distribution & Logistics Management	A	Emerald Group Publishing	Q1 (sjr: 1.82)
28	Academy of Management Learning & Education	A*	Academy of Management	Q1 (sjr: 1.48)
29	International Journal of Innovation**	-	-	-
30	Academy of Management Proceedings***	-	Academy of Management	-
31	Human Service Organizations: Management, Leadership & Governance	-	Taylor & Francis Online	Q2 (sjr: 0.37)
32	Communications of the ACM	A	Association for Computing Machinery	Q1 (sjr: 0.71)
33	Supply Chain Management: An International Journal	A	Emerald Group Publishing	Q1 (sjr: 1.99)
34	Industrial Marketing Management	A*	Elsevier	Q1 (sjr: 1.66)
35	Journal of Public Administration Research and Theory	A	Oxford University Press	Q1 (sjr: 5.41)
36	Ecology and society	-	Resilience Alliance	Q1 (sjr: 1.73)
37	Journal of labor economics	A*	The University of Chicago Press	Q1 (sjr: 9.11)

* This is the only journal that is not listed as the top 50% of the respected scientific field.

** This is the only journal that has no entry in the ABDC list and the Scimago database, without an obvious reason to explain its absence or assume its influence.

*** The Academy of proceedings is a scientific journal, published yearly by the academy of management. It contains the most influential papers and conference papers of that year. Although it has no scores in the ABDC and Scimago databases it is safe to assume that the quality of the papers is high, since all other Academy of Management journals have an enormous influence.

The information found in Table 2.4 shows that the combined scientific influence of these papers is strong. The assumption in this research is that one or two papers that are of sound quality, but have

not been published by an influential journal and/or publisher, does not harm the fortitude of this literature study. That combined with the limited amount of DevOps studies led to the conclusion to keep the few research papers that were qualified as C-level (ABDC, 2019) or Q3-level (Scimago, 2019). More information regarding these quality levels can be found in appendix 1.7.

More information regarding the key words that led up to the selected sources can be found in appendix 1.8. The ID's in appendix 1.8 correspond to the ID's of appendix 1.5 and 1.6. In appendix 1.8 is presented which source was used for which research question and in which database(s) the source was found. The full text is analyzed for every selected source. Table 2.5 represents the information that will be extracted from the selected sources.

Table 2.5: Information that will be extracted from the selected sources

	SRQ 1	SRQ 2	SRQ 3
	Collaboration	Shared Goal	Shared Understanding
1. Definition	x	x	x
2. Discussion of the concept	x	x	x
3. Aspects	x	x	x

Some of the sources contain very relevant material and therefore will be the backbone of the study. In table 2.6 the sources of table appendix 1.5 are depicted again, the last column shows which information will be extracted.

Table 2.6: Information that will be extracted from the building block sources

Nr	Author(s)	Year	Retrieved Information
1	Akbar, Baruch & Tzokas	2018	Discussion of the concept
2	Alexander & Van Knippenberg	2014	Definition of Shared Goal Discussion of the concept
3	Aubé, Rousseau & Tremblay	2015	Definition of Shared Understanding
4	Basadur & Gelade	2006	Discussion of the concept
5	Berggren, Johansson & Baroutsi	2017	Sub-aspects of Shared Understanding
6	Bittner & Leimeister	2014	Definition of Shared Understanding Discussion of the concept Sub-aspects of Shared Understanding
7	Bygballe, Swärd & Vaagaasar	2016	Discussion of the concept
8	Cetindamar, Çatay & Basmaci	2005	Definition of Shared Goal Discussion of the concept
9	Cha, Kim, Lee & Bachrach	2015	Discussion of the concept Sub-aspects of Shared Goal
10	Charoensuk, Wongsurawat & Khang	2014	Discussion of the concept
11	Claps, Svensson & Aurum	2015	Definition of Collaboration Discussion of the concept

12	Detzten, Verbeeten, Gamm & Möller	2018	Definition of Shared Goal Discussion of the concept
13	Eldor	2019	Sub-aspects of Shared Goal
14	Erich, Amrit & Daneva	2017	Definition of Collaboration Discussion of the concept Main aspects of Collaboration
15	Folke, Carpenter, Walker, Scheffer, Chapin III & Rockström	2010	Discussion of the concept
16	Ghezzi	2017	Definition of Collaboration Discussion of the concept
17	Ghobadi & Mathiassen	2016	Discussion of the concept
18	Gupta, Kapur & Kumar	2017	Definition of Collaboration Discussion of the concept Main aspects of Collaboration
19	Hackman & Wageman	2005	Sub-aspects of Shared Understanding
20	Hu & Liden	2011	Sub-aspects of Shared Goal
21	Iden & Bygstad	2018	Definition of Shared Understanding Discussion of the concept Sub-aspects of Shared Understanding
22	Joshi, Chi, Datta & Han	2010	Discussion of the concept
23	Kwak & Anbari	2009	Definition of Shared Goal
24	Lee & Xia	2010	Discussion of the concept
25	Mohammed & Dumville	2001	Definition of Shared Understanding Discussion of the concept Sub-aspects of Shared Understanding
26	Mok, Shen & Yang	2015	Discussion of the concept
27	Morgan, Richey Jr & Autry	2016	Discussion of the concept
28	Ohland, Loughry, Woehr, Bullard, Felder, Finelli, Layton, Pomeranz & Schmucker	2012	Sub-aspects of Shared Understanding
29	Polat, Lynn, Akgün & Emre	2018	Definition of Shared Goal Discussion of the concept
30	Powers, Morgeson & Lyons	2014	Discussion of the concept
31	Prentice & Brudney	2016	Discussion of the concept
32	Roche	2013	Definition of Collaboration Discussion of the concept
33	Soosay, Hyland & Ferrer	2008	Discussion of the concept
34	Tsai & Hsu	2014	Discussion of the concept
35	Vangen & Huxham	2012	Discussion of the concept Sub-aspects of Shared Goal
36	Walker, Holling, Carpenter & Kinzig	2004	Discussion of the concept

37	Zábojník	2002	Sub-aspects of Shared Goal
----	----------	------	----------------------------

Snowball method

Several of the other sources that are utilized in this research are found by the backward and forward snowballing methods. Some of them are added at a later stage due to a new insight. The potential of sources is enormous, but, of course, these sources also have to meet the criteria depicted in table 2.1.

The references found for the first three sub research questions, by using the building block method, are substantial and depicted as a total set in appendix 1.5. Backward snowballing is applied for sub research question 1 and 3 to strengthen the scientific foundation even further. For sub research question 2 forward snowballing is utilized. The literature that will be used for research question 4 and 5 is based on the literature that is gathered to answer the first three questions. The relationships between collaboration and its main-aspects can be described based on the literature depicted in table 2.6 and the literature that is found by applying the snowball methods.

In an attempt to broaden the scientific foundation, for research question 1, a starting paper was selected for backward snowballing. Erich, Amrit & Daneva (2017) used Lwakatare et al. (2016) as a reference and this paper was referenced by other literature and fitted the research question very good. This paper was selected as a paper for backward snowballing.

Table 2.7 shows the sources that were selected using the backward snowballing method. There is more information regarding the backward snowballing method in appendix 1.9. The tables depicted there will also show the sources that have not been selected, but met the criteria. For backward snowballing the inclusion criteria were based on the criteria depicted in table 2.1. However, one criterion has been added in an attempt to select the most relevant materials; only journals articles can be selected. If a source meets all the criteria it is deemed relevant (column 3). It will be added as a reference (column 4) if the full text has a resemblance to the research. The last column of table 2.7 shows which information was retrieved from the journals.

Table 2.7: sources & extracted data - backward snowballing – sub research question 1

Starting paper	ID	References (used by Lwakatare, Kuvaja & Oivo) that met criteria	Relevant?	Added as a reference	Retrieved information
Lwakatare, Kuvaja & Oivo (2016)	3	Penners R. & Dyck, A. (2015) Release Engineering vs. DevOps An Approach to Define Both Terms Full-scale Software Engineering	Yes	Yes	Definition collaboration, Discussion of the concept
	4	Lwakatare, L.E., Kuvaja, P. & Oivo, M. (2015) Dimensions of DevOps Paper presented at the International Conference on Agile Software Development.	Yes	Yes	Definition collaboration, Discussion of the concept
	5	Humble, J. & Molesky, J. (2011) Why enterprises must adopt DevOps to	Yes	Yes	Definition collaboration, Discussion of

		enable continuous delivery Cutter IT Journal. Vol. 24-8. pp. 6-12			the concept
	6	Iden, J., Tessem, B. & Paiväranta, T. (2011) Problems in the interplay of development and IT operations in system development projects: A Delphi study of Norwegian IT experts Information and Software Technology. Fol. 53-4. pp. 394–406.	Yes	Yes	Discussion of the concept Main aspects collaboration
	8	Dyck, A., Penners, R. & Lichter, H. (2015) Towards definitions for release engineering and DevOps	Yes	Yes	Discussion of the concept

The backward snowballing method was applied for sub research question 1 since the DevOps concept is a quite new. It was interesting to find out which sources Lwakatare et al. (2016) used to substantiate the claims they made regarding the dimensions of DevOps. They deemed collaboration to be one of these dimensions.

To broaden the scientific foundation, for research question 2, a starting paper was selected for forward snowballing. Jassawalla & Sashittal (1998) were found as a credible source, that did not meet the criteria of > 2001. This paper was selected as a paper for forward snowballing, to utilize and modernize the statements that Jassawalla & Sashittal (1998) made.

Table 2.8 shows the sources that were selected using the forward snowballing method. There is more information regarding the forward snowballing method in appendix 1.9. The tables depicted there will also show the sources that have not been selected, but met the criteria. For forward snowballing the inclusion criteria were based on the criteria depicted in table 2.1. However, one criterion has been added in an attempt to select the most relevant materials, namely; only articles that had collaboration, shared goal, vision, leadership or organizational structure in the title could be selected. This was done to distillate the most relevant set of articles that had been building on the theories of Jassawalla & Sashittal (1998). If a source meets all the criteria it is deemed relevant (column 3). It will be added as a reference (column 4) if the full text has a resemblance to the research. The last column of table 2.8 shows which information was retrieved from the journals.

Table 2.8: sources & extracted data - forward snowballing – sub research question 2

Starting paper	ID	References (that used Jassawalla & Sashittal) that met criteria	Relevant?	Added as a reference	Retrieved information
Jassawalla & Sashittal (1998)	3	Stock, R.M., Totzauer, F. & Zacharias, N.A. (2013) A Closer Look at Cross-functional R&D Cooperation for Innovativeness: Innovation-oriented Leadership and Human Resource Practices as Driving Forces Journal of Product Innovation Management. Vol. 31-5. pp. 924-938.	Yes	Yes	Sub-aspects of Shared Goal

	5	Aronson, Z. H., Shenhar, A. J., & Patanakul, P. (2013). Managing the Intangible Aspects of a Project: The Affect of Vision, Artifacts, and Leader Values on Project Spirit and Success in Technology-Driven Projects. Project Management Journal, vol. 44-1. pp. 35–58.	Yes	Yes	Discussion of the concept Sub-aspects Shared Goal
	7	Thamhain, H.J. (2009) Leadership lessons from managing technology-intensive teams. International Journal of Innovation and Technology Management. Vol. 06-2	Yes	Yes	Discussion of the concept Sub-aspects Shared Goal
	8	Lynn G.S. & Akgün, A.E. (2003) Project visioning: Its components and impact on new product success Journal of Product Innovation Management. Vol. 18-6. pp. 374-387.	Yes	Yes	Discussion of the concept Sub-aspects Shared Goal

The forward snowballing method was applied for sub research question 2 based on the fact that Jassawalla & Sashittal (1998) made some interesting claims, but did not meet all the criteria. It was interesting to find out which sources quoted Jassawalla & Sashittal (1998) to substantiate their claims.

To broaden the scientific foundation, for research question 3, a starting paper was selected for backward snowballing. Bittner & Leimeister (2014) was selected as a paper for backward snowballing. Their theory has an interesting model that has a lot of information that can contribute to research question.

Table 2.9 shows the sources that were selected using the backward snowballing method. There is more information regarding the backward snowballing method in appendix 1.9. The tables depicted there will also show the sources that have not been selected, but met the criteria. For backward snowballing the inclusion criteria were based on the criteria depicted in table 2.1. However, one criterion has been added in an attempt to select the most relevant materials; only journals articles can be selected. If a source meets all the criteria it is deemed relevant (column 3). It will be added as a reference (column 4) if the full text has a resemblance to the research. The last column of table 2.9 shows which information was retrieved from the journals.

Table 2.9: sources & extracted data - backward snowballing – sub research question 3

Starting paper	ID	References (used by Bittner & Leimeister) that met criteria	Relevant?	Added as a reference	Retrieved information
Bittner & Leimeister (2014)	1	<u>Akkerman, S.; Van den Bossche, P.; Admiraal, W.; Gijsselaers, W., Segers; M., Simons, R.-J.; and Kirschner, P. (2007) Reconsidering group cognition: From conceptual confusion to a boundary area between cognitive and socio-cultural perspectives?</u>	Yes	Yes	Definition Shared Understanding, Discussion of the concept

		<u>Educational Research Review, 2, 1. pp. 39–63.</u>			
	10	<u>Kleinsmann, M., and Valkenburg, R. (2008) Barriers and enablers for creating shared understanding in co-design projects. Design Studies, 29, 4. pp. 369–386.</u>	Yes	Yes	Discussion of the concept
	14	<u>Mohammed, S., and Dumville, B.C. (2001) Team mental models in a team knowledge framework: Expanding theory and measurement across disciplinary boundaries. Journal of Organizational Behavior, 22, 2. pp. 89–106.</u>	Yes	Already a Source	Definition Shared Understanding, Discussion of the concept Sub-aspects Shared Understanding
	17	<u>Van den Bossche, P.; Gijssels, W.; Segers, M.; Woltjer, G.; and Kirschner, P. (2011) Team learning: Building shared mental models. Instructional Science, 39, 3. pp. 283–301.</u>	Yes	Yes	Definition Shared Understanding, Discussion of the concept Sub-aspects Shared Understanding

The backward snowballing method was applied for sub research question 3 since Bittner & Leimeister (2014) made some interesting claims and based their work on an extensive set of scientifically strong literature.

Phases 4 - Analysis

The selection process has resulted in a substantiated set of sources that will be used to design the conceptual model. By utilizing these phases there is a structure that can be reproduced by other researchers. A visualization of the 50 sources that have been selected is presented in appendix 1.10.

By following the method advised by Wolfswinkel et al. (2013) the distilled set of literature gained scientific tenacity. The literature will be used to answer the first three sub research questions. This will require sufficient data extraction, the subjects that are important during this process are depicted in table 2.5. The literature will be used to formulate definitions for collaboration, shared goal and shared understanding. Tables 2.6-2.9 present which information is extracted, from which source (in both tables the last column) and appendix 1.10 shows an overview of the total literature set. To address the concepts, collaboration, shared goal and shared understanding in full, it is important to discuss the concept and reflect upon the concept by utilizing several sources. Furthermore, it will clarify the main-aspects for collaboration, shared goal and shared understanding. Based on the information in appendix 1.10 every reference is analyzed. The classification of sources, definition, discussion of concept and respective aspects, helps with the analysis of every reference.

Definitions and aspects are used to draft a substantiated conceptual model, which will help to answer the research question. More information is required to answer sub research question 4 and 5, this will be done by utilizing qualitative research.

It is imperative that definitions and aspects are substantiated. So, the goal is to find an overlay in the literature. The combined knowledge will be used to build up the theoretical framework (2.2 - implementation) and the conceptual model (2.3). The implementation will bundle the relevant literature that will be extracted from the sources that were introduced in this section. Sub research questions 1, 2 and 3 will be answered by the theoretical analysis of the scientific literature. Sub research question 4 and 5 will be answered by a case study based on the conceptual model.

2.2 Implementation

The goal of this section is to present an overview of the relevant literature regarding collaboration (paragraph 2.2.1) and the main-aspects of collaboration (paragraph 2.2.2 and paragraph 2.2.3). The relationships between collaboration and its main-aspects are discussed in paragraph 2.2.4. The results have been visualized, figure 2.6 in section 2.3, in the conceptual model.

2.2.1 What is collaboration?

Lwakatare et al. (2015) and Humble and Molesky (2011) formulated four dimensions of DevOps. Both studies mention collaboration, automation and measurement as a characteristic, but the fourth characteristic differs. Humble and Molesky (2011) identified sharing as a characteristic, while Lwakatare et al. (2015) identified monitoring. In a later study of Lwakatare et al. (2016) culture is defined as a fifth characteristic. Lwakatare et al. (2015) based their research partly on research done by Bang, Chung, Choh & Dupuis (2013), they stated that a collaboration culture is essential for DevOps. Giudice & Condo (2017) state that DevOps requires leaders to change the way they organize their teams and staff, how they collaborate and how they use automation. An important assumption of this research is that leaders believe in the empowerment of teams by giving them responsibility and expecting ownership. The focus will be on the collaboration within a team and which main- and sub-aspects influence it.

On a strategic level the alignment between Business and IT is often researched (Folke, Carpenter, Walker, Scheffer, Chapin III & Rockström, 2010; Soosay, Hyland & Ferrer, 2008). However, the alignment of departments or roles within the Business-context or IT-context are often neglected (Charoensuk, Wongsurawat & Khang, 2014). Iden et al. (2012) stress the importance of aligning IS Development and IT Operations. The importance of aligning important stakeholders is well established (Bygballe, Swärd & Vaagaasar, 2016; Iden et al. 2012; Mok, Shen & Yang, 2015), but it is not yet fully understood how this process unfolds (Iden & Bygstad, 2018; Folke et al., 2010; Soosay, Hyland & Ferrer, 2008; Prentice & Brudney, 2016). In the past, the departments for development and operations worked separately to achieve a common goal: delivering software to the customers. The division of responsibilities was clear. IS Development is developing the product, while IT Operations is focused on ensuring the defined level of service (Iden & Bygstad, 2018). By applying this structure an employee of IS Development will probably not feel responsible for a high priority issue in production. Iden & Bygstad (2018) concluded, based on practitioners in the field, that there is a need for committed collaboration.

DevOps strives to reduce the perceived gap (Wettinger, Breitenbücher & Leymann, 2014) between IS Development and IT Operations. Part of this reduction could be achieved by improving the collaboration. Davis & Daniels (2015) argue that DevOps is a cultural movement which improves the software development processes and the lives of the professionals. Patrick Debois (2011) emphasizes

that the concept of DevOps should be implemented for the entire organization and not only to the relationship between IS Development and IT Operations.

Collaboration in a DevOps context includes cross-functional disciplines, who are all contributing to the artifacts produced by the delivery process. Roche (2013) argues that organizations that were able to adopt DevOps needed to build teams that were more collaborative. Gupta, Kapur & Kumar (2017) state that DevOps emphasizes the collaboration of Development and Operation teams over the entire software cycle. They introduce the terms; OpsinDev and DevinOps. OpsinDev supports the principle that operational personnel provides information regarding the behavior of software and infrastructure to the developers. DevinOps means that developers share their knowledge about the requirements, design and software components. The purpose of the knowledge transfer is to achieve a shorter time to market and improve quality (Gupta et al., 2017).

A DevOps Team has a shared responsibility towards developing and operating features and functionalities in a stable manner (Erich et al., 2017). The people working in these teams, need to be open for cross-functional collaboration. Collaboration is a concept that focuses on social interaction (Iden & Bygstad, 2018; Roche, 2013). Lwakatare et al. (2015) argue that the collaboration dimension impacts the team structure and the required capabilities.

Although the form, scope and structure of relationships vary, it is well established that collaboration is worthwhile and to be encouraged (Prentice & Brudney, 2016). The scarcity of concrete activities that organizations should undertake to achieve more collaborative team puts practitioners in an awkward position. They often do not know what effective collaboration entails and how or why they should foster it. Organizations practicing DevOps strive to remove the cultural barrier between IS Development and IT Operations (Iden et al., 2012). Dyck, Penners & Lichter (2015) argue that it aims to create empathy and cross-functional perspective.

Lwakatare et al. (2015) state that collaboration, the dimension of DevOps, addresses the problem of poor communication and results in shared responsibilities. They stress the importance of cross-functional collaboration between software development and operations (Iden & Bygstad, 2018). According to Bang et al. (2013) and Lwakatare et al. (2016) it also encapsulates cross-functional collaboration with other departments, for instance quality assurance. Collaboration in a DevOps context influences the team structure, required skillset of team members and responsibilities that teams have. It is designed to create transparency regarding the prerequisites of a successful team.

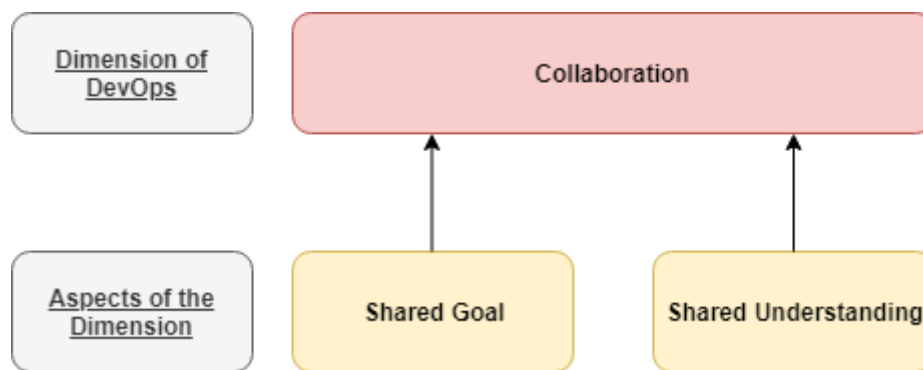
Iden & Bygstad (2018) broke collaboration down in partnership and shared knowledge. They define partnership as the cooperative interaction between actors. A definition that emphasizes the mutual activity and therefore shows the collaborators have a goal. The degree to which team members are informed of each other's knowledge is described as shared knowledge. A definition that shows that Iden & Bygstad (2018) saw the importance of an understanding of each other activities. Lwakatare et al. (2015) state that collaboration can be achieved by boosting shared responsibility.

By analyzing the literature regarding collaboration (in a DevOps context) three elements stand out, namely; cross-functional perspective (Bang et al., 2013; Lwakatare et al., 2015; Iden & Bygstad, 2018, Gupta et al., 2017), executing activities based on a shared understanding (Bittner & Leimeister, 2014; Tessem & Iden, 2008; Iden & Bygstad; 2018; Charoensuk et al., 2014; Bygballe et al., 2016) and having a shared goal (Kwak & Anbari, 2009; Polat et al., 2018; Kozlowski & Bell, 2003; LePine, 2005; Alexander & Van Knippenberg, 2014) In a DevOps context it is common to have cross-functional

teams (Lwakatare et al., 2015; Iden & Bygstad, 2018), therefore it is a given that cross-functional team members have to cope with the challenges this can introduce. The other two elements, namely shared understanding and shared goal can help them to become more effective (Polat et al., 2018; Bittner & Leimeister, 2014; Iden & Bygstad, 2018). These main-aspects will be used to build up the definition of collaboration in a DevOps context and will be examined in 2.2,2 and 2.2,3, see figure 2.1 for the main-aspects of collaboration.

In this study collaboration in a DevOps context is defined as cross-functional (Bang et al., 2013; Lwakatare et al., 2015; Iden & Bygstad, 2018, Gupta et al., 2017) team members that execute activities (Bittner & Leimeister, 2014; Tessem & Iden, 2008; Iden & Bygstad; 2018; Charoensuk et al., 2014; Bygballe et al., 2016) in an attempt to fulfill a goal (Kwak & Anbari, 2009; Polat et al., 2018; Kozlowski & Bell, 2003; LePine, 2005; Alexander & Van Knippenberg, 2014).

Figure 2.1: the main-aspects of the collaboration



2.2.2 Shared goal

The two main-aspects of collaboration have been selected, based on the definition of collaboration. These main-aspects are a shared goal and shared understanding (of cross-functional responsibilities). Multidisciplinary teams require alignment and clarity. According to several researchers (Kwak & Anbari, 2009; Polat et al., 2018; Kozlowski & Bell, 2003; LePine, 2005; Alexander & Van Knippenberg, 2014), enabling teams to have a shared goal is an important instrument to achieve this. This paragraph introduces shared goal, as a main aspect of collaboration, in an attempt to find appropriate insights regarding sub research question 2.

Teams increase effectiveness when they are able to create a collective mind (Polat et al., 2018). Cetindamar, Çatay & Basmacı (2005), Alexander & Van Knippenberg (2014) and Kwak & Anbari (2009) argue that in order to collaborate effectively it is important to have a shared goal. This main-aspect corresponds with our definition of collaboration. A shared goal can help to create an environment in which the actions of all employees are aligned (Kozlowski & Bell, 2003; LePine, 2005), because there is clarity and focus. Detzten, Verbeeten, Gamm & Möller (2018) state that target rigidity can help teams to increase their performance. They address the importance for teams to steer their combined efforts towards a clear goal. This can be achieved by three relevant sub-aspects; leadership (Vangen & Huxham, 2012; Alexander & Van Knippenberg, 2014; Aronson et al., 2013; Stock, Totzauer & Zacharias, 2013), organizational structure (Jassawalla & Sashittal, 1998; Thamhain, 2009) and a shared vision and cadre (Eldor, 2019; Aronson et al., 2013; Lynn & Akgün, 2003).

In this study a shared goal is defined as an organizational instrument to create focus, by setting an objective or responsibility which is applicable for all team members (Detzten et al., 2018; Kozlowski

& Bell, 2003; Alexander & Van Knippenberg, 2014). It enables teams to perform (Polat et al., 2018; Cetindamar et al., 2005; Verbeeten et al., 2018) and collaborate (Alexander & Van Knippenberg, 2014; Kwak & Anbari, 2009; Polat et al., 2018) more effectively, which is influenced by the introduced sub-aspects.

An organization should facilitate several things, in order to have a team that is willing to have a shared goal. Jassawalla & Sashittal (1998) investigated the elements that are required for cross-functional collaboration, regarding new product development. Organizations should create a situation where team members are collectively working on a shared goal (Vangen & Huxham, 2012). There should be enough convergence to focus on a shared goal, but there should also be enough friction to have the appropriate discussions along the way (Vangen & Huxham, 2012). This is a complex state, which can only be maintained with sufficient leadership (Vangen & Huxham, 2012; Jassawalla & Sashittal, 1998). The organizational structure can, if structured appropriately, help the teams to excel (Jassawalla & Sashittal, 1998). Another important element is that the vision of the organization is shared and communicated (Eldor, 2019). It is defined as the awareness of employees regarding the objectives and future aspirations of the organization. The following sections will further explain the sub-aspects of a shared goal. It will introduce indicators for these sub-aspects, to be able to assess when these sub-aspects are present and flourishing.

Leadership

An important sub-aspect of having a shared goal is leadership. When an organization requires teams to have a shared goal it is important that there are leaders in the organization that provide clarity (Stock et al., 2013). One of the things leaders should foster is the belief that working together (Jassawalla & Sashittal, 1998), as a team, will come with benefits (Cetindamar et al., 2005), economically or otherwise. Every individual should see the necessity (Stock et al., 2013), or at least the benefit (Aronson et al., 2013) of actively working together. One of the arguments could be that teams who have cross-functional capacities, can reduce the risks by aligning forces. The organization should be clear about the reasoning behind the team structure (Kwak & Anbari, 2009).

Jassawalla & Sashittal (1998) state that it is important how and why leaders are selected. Leaders that have a strong affiliation to the environment (Thamhain, 2009) and have the support of senior management are able to create teams that deliver more than the sum of their talents. One of the distinctive behaviors of these leaders is that they show cross-functional individuals what the interdependencies are between their actions and the business goal (Jassawalla & Sashittal, 1998; Cha, Kim, Lee, & Bachrach, 2015). Two indicators seem to be relevant: 1) benefit of working together is clear and communicated (Jassawalla & Sashittal, 1998; Cetindamar et al., 2005; Stock et al., 2013; Aronson et al., 2013) and 2) leaders know the environment and senior management (Jassawalla & Sashittal, 1998; Thamhain, 2009; Cha et al., 2015).

Organizational structure

Another sub-aspect that can foster a shared goal is the organizational structure (Jassawalla & Sashittal, 1998; Kwak & Anbari, 2009). Organizations need to find their balance between centralized or decentralized decision making. "Centralization is preferred to decentralization if and only if the manager's signal is better than the worker's signal" (Zábojník, 2002, pp. 8). Decentralized decision-making means that the decisive power of day to day decisions are made by the team. It is based on

the assumption that the people who execute the work should also make most of the decisions regarding day to day operations. It creates mature teams (Gupta et al., 2017) that can talk about their goal within the team and with management (Jassawalla & Sashittal, 1998). They will show more willingness to be accountable in comparison to teams for whom all the decisions are made by the management.

Another way organizational structures can help is in the way resources are allocated. Individual team members should have a primary focus on the team. It works counterproductive when team members are assigned to several teams (Tessem & Iden, 2008; Giudice & Condo, 2017). Then there is a limited focus on the goals of the team (Thamhain, 2009) and therefore the chance of having a shared goal is reduced. Two indicators seem to be relevant: 3) decentralized day-to-day decisions (Zábojník, 2002; Jassawalla & Sashittal, 1998; Gupta et al., 2017) and 4) individual team members have focus on the team (Tessem & Iden, 2008; Thamhain, 2009; Giudice & Condo, 2017).

Shared vision and cadre

A team can only define a relevant shared goal when they know the strategy of their organization (Eldor, 2019). Therefore, they need to know the mission statement of the organization (Eldor, 2019; Aronson et al., 2013; Lynn & Akgün, 2003). Furthermore, they should have an insights in when they are successful as a team (Gutiérrez, Lloréns-Mones & Bustinza Sánchez, 2009), in order to determine and execute the required tasks (Paris, Salas & Cannon-Bowers, 2000). Organizations should allocate time to ensure that the business goals are understood by team members (Jassawalla & Sashittal, 1998; Gutiérrez et al., 2009).

It is important for a team to know what they can decide for themselves and when they should inform, consult or ask the management (Eldor, 2019). When an organization can provide clear guidelines, a team can operate more effectively (Hu & Liden, 2011; Gutiérrez et al., 2009). Two indicators seem to be relevant: 5) strategy and expectations of the organization are known (Eldor, 2019; Gutiérrez et al., 2009; Aronson et al., 2013; Lynn & Akgün, 2003; Paris et al., 2000; Jassawalla & Sashittal, 1998) and 6) opportunities and limitations are known (Eldor, 2019; Hu & Liden, 2011; Gutiérrez et al., 2009).

The indicators that are presented in table 2.10 have an influence on having a shared goal, not on achieving a shared goal. The focus will be on this type of indicators, because they have an impact on clarifying the purpose of a team.

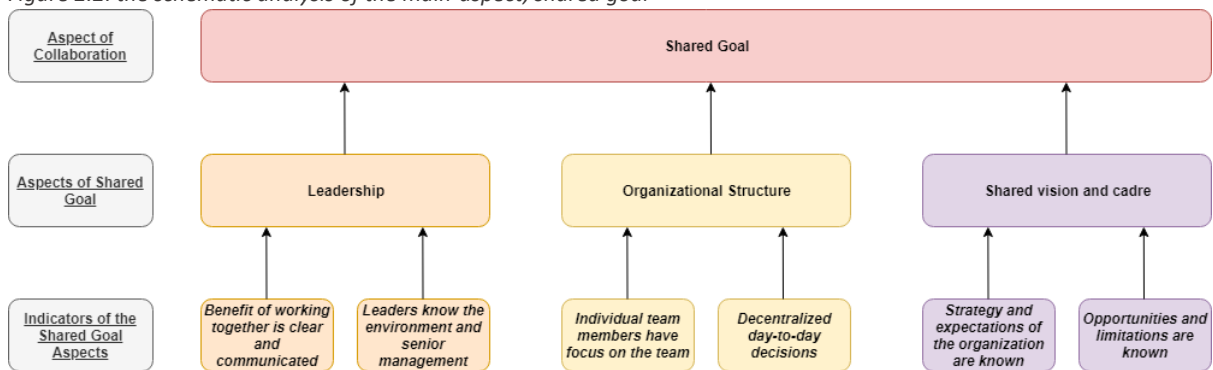
Table 2.10: indicators of having a shared goal

Indicators of Shared Goal	Related to the Sub-aspect of Shared Goal	Sources
<i>Benefit of working together is clear and communicated</i>	<i>Leadership</i>	<i>Jassawalla & Sashittal, 1998; Cetindamar et al., 2005; Stock et al., 2013; Aronson et al., 2013</i>
<i>Leaders know the environment and senior management</i>	<i>Leadership</i>	<i>Jassawalla & Sashittal, 1998; Thamhain, 2009; Cha et al., 2015</i>

<i>Decentralized day-to-day Decisions</i>	<i>Organizational Structure</i>	<i>Zábojník, 2002; Jassawalla & Sashittal, 1998; Gupta et al., 2017</i>
<i>Individual team members have focus on the team</i>	<i>Organizational Structure</i>	<i>Tessem & Iden, 2008; Thamhain, 2009; Giudice & Condo, 2017</i>
<i>Strategy and expectations of the organization are known</i>	<i>Shared vision and cadre</i>	<i>Eldor, 2019; Gutiérrez et al., 2009; Aronson et al., 2013; Lynn & Akgün, 2003; Paris et al., 2000; Jassawalla & Sashittal, 1998</i>
<i>Opportunities and limitations are known</i>	<i>Shared vision and cadre</i>	<i>Eldor, 2019; Hu & Liden, 2011; Gutiérrez et al., 2009</i>

When an organization is able to enable the sub-aspects that are depicted in figure 2.2 a team will be more likely to have or come to a shared goal. There are three sub-aspects that seem to influence shared goal: leadership, organizational structure and shared vision and cadre. Every sub-aspect of the shared goal has several indicators, which are depicted in figure 2.2 and in table 2.10. These sub-aspects are elements that the team has only partly influence on. The sub-aspects should mostly be facilitated by the organization (Eldor, 2019; Gutiérrez et al., 2009; Tessem & Iden, 2008; Zábojník, 2002; Vangen & Huxham, 2012).

Figure 2.2: the schematic analysis of the main-aspect; shared goal



2.2.3 Shared understanding

In this paragraph shared understanding will be introduced as a main aspect of collaboration, in order to answer sub research question 3. Shared understanding becomes increasingly important when individuals, with different perspectives, have to work together (Bittner & Leimeister, 2014; Kleinsmann & Valkenburg, 2008; Akkerman, Van den Bossche, Admiraal, Gijsselaers, Segers, Simons & Kirschner, 2007; Charoensuk et al., 2014). Due to the recent developments in the IT landscape, namely Agile (Lee & Xia, 2010) and DevOps (Iden & Bygstad, 2018), teams are more cross-functional (Ghobadi & Mathiassen, 2016; Joshi, Chi, Datta & Han, 2010). Therefore it becomes increasingly important to understand other perspectives (Akbar, Baruch & Tzokas, 2018; Basadur & Gelade, 2006; Walker, Holling, Carpenter & Kinzig, 2004; Morgan, Richey Jr & Autry, 2016), in order to reach the goals that are set by the organization (Tsai & Hsu, 2014). A shared understanding helps to

organize the work (Van den Bossche et al., 2010) and enables teams to focus on the right activities (Akkerman et al., 2007; Morgan, Richey Jr. & Autry, 2016). Aubé & Rousseau & Tremblay (2015) state that it works as a motivator for teams.

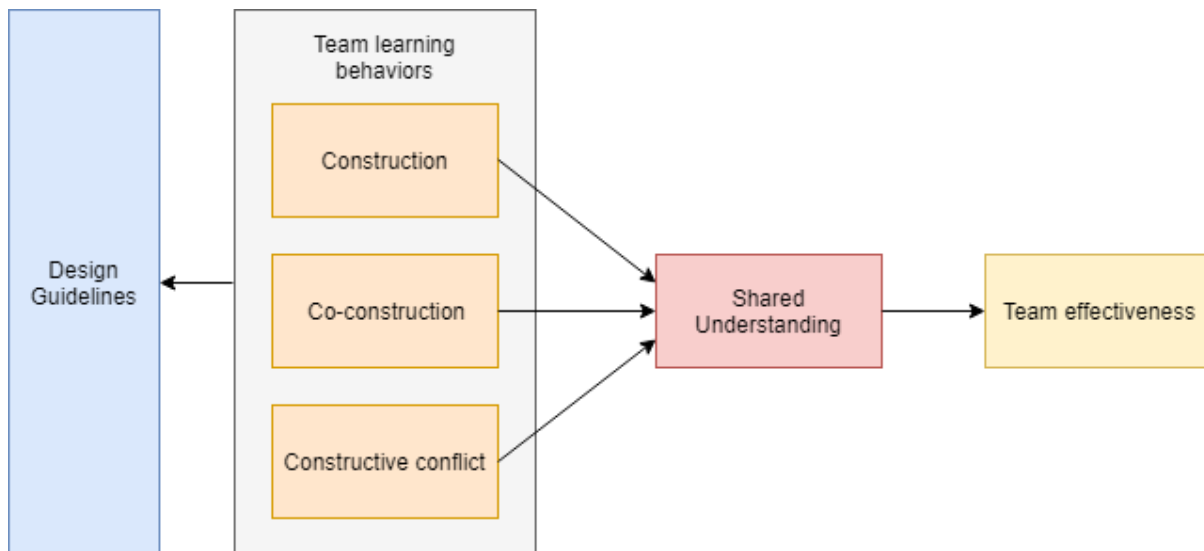
Mulder, Swaak & Kessels (2002) stress the importance of mutual knowledge, beliefs and assumptions, while Smart, Mott, Sycara, Braines, Struc & Shadbol (2009) include the ability of a team to coordinate their behavior in order to reach their objective (in other words: shared goal). Bittner & Leimeister (2014) stress the importance of first defining understanding; “understanding is an ability to exploit bodies of causal knowledge for the purpose of accomplishing cognitive and behavioral goals” (Smart et al., 2009, pp. 2). An important element here is causal knowledge, this stresses that the activities of the team are directed towards a (shared) goal (Bittner & Leimeister, 2014).

The concept of understanding is a dynamic state, which can change over time due to experimentation and learning (Bittner & Leimeister, 2014; Akkerman et al., 2007). According to Bittner & Leimeister (2014) the definition of shared understanding is: “the degree to which people concur on the value of properties, the interpretation of concepts, and the mental models of cause and effect with respect to an object of understanding”. Mohammed & Dumville (2002) and Stout, Cannon-Bowers, Salas & Milanovich (1999) state that team effectiveness will improve if team members have a shared understanding. They concretize this by stating it is important to align the concepts of task, team, equipment and situation (Mohammed & Dumville. 2002)

In this study a shared understanding is defined as a concept influenced by the team that creates clarity and alignment regarding the required activities, responsibilities and situation (Bittner & Leimeister; 2014; Mohammed & Dumville, 2001; Stout et al., 1999) with respect to an object of understanding. It enables a team to perform and collaborate more effectively (Aubé et al., 2015; Mohammed & Dumville, 2002; Stout et al., 1999; Kleinsmann & Valkenburg, 2008), which is influenced by (Bittner & Leimeister, 2014; Van den Bossche et al., 2011) construction, co-construction and constructive conflict.

The model of Bittner & Leimeister (2014), depicted in figure 2.3, has been selected as the backbone for explaining shared understanding of cross-functional responsibilities. This decision has been made since Bittner & Leimeister (2014) focus on heterogenous teams, which corresponds with the cross-functional nature of teams in a DevOps context. Another argument for selecting this model is the focus on the individual team member, the relationship between team members and the way to create a constructive discussion. Furthermore, they talk about creating effective teams, which comes back in this research as well (Kwak & Anbari, 2009; Polat et al., 2018) and corresponds with the research goal.

Figure 2.3 The model from Bittner & Leimeister, regarding shared understanding of heterogenous teams



Adapted from Bittner & Leimeister (2014), which based their work on Van den Bossche et al. (2011)

The model is very similar to the model of Van den Bossche et al. (2011), although they define the concept as a shared mental model and Bittner & Leimeister (2014) as a shared understanding. Furthermore, Bittner & Leimeister (2014) introduced design guidelines to substantiate what can help to create a shared understanding. According to Bittner & Leimeister (2014), their definition of shared understanding can be helpful by determining the value of properties, the interpretation of concepts and the mental models of cause and effect with respect to an object. The practical relevance for a DevOps team is enormous. A DevOps team is cross-functional (Iden & Bygstad, 2018) and needs to find common ground while skill sets, theoretical backgrounds and experience differ (Gupta et al., 2017).

Construction, co-construction and constructive conflict (Van den Bossche et al., 2011) are learning behavior concepts that influence a shared understanding. Bittner & Leimeister (2014) introduced several guidelines for designing this process. The following sections will further explain the sub-aspects of a shared understanding. It will introduce indicators for these sub-aspects, to be able to assess when these sub-aspects are present and flourishing.

Construction

Teams should be able to create a construction of the individual. Self-aware individuals are able to portrait themselves realistically and understandable (Van den Bossche et al., 2011). Team members need to be aware of their own mental model (Bittner & Leimeister, 2014). Being self-aware is the capability to have a conscious knowledge of one's own character and individuality. This helps in aligning behavior and understanding how others perceive you. It is important that fellow team members are actively listening and trying to grasp everyone's perspective. Using each individual understanding helps to give meaning to the situation at hand (Bittner & Leimeister, 2014; Van den Bossche, 2011). If something is unclear, they should ask questions to complete the image (Bittner & Leimeister, 2014; Van den Bossche, 2011). By doing so, they strengthen the clarity of each individual perspective. Three indicators seem to be relevant: 1) self-aware team members (Berggren et al., 2017) and 2) environment for asking questions (Bittner & Leimeister, 2014; Van den Bossche, 2011) and 3) team members listen to each other (Bittner & Leimeister, 2014; Mohammed & Dumville, 2001).

Co-construction

Teams should be able to create a co-construction of the images of two or more team members. A team can learn from reflection and evaluation patterns (Mohammed & Dumville, 2001). Reflection is taking the time to consider your experience in order to learn from it. Evaluation builds on that by assessing the situation, to improve. They should be able to evaluate the outcome and compare the individual understanding of team members (Van den Bossche et al., 2011). The team could benefit from having an insight of what their responsibilities are and what kind of skills they need (Hackman & Wageman, 2005) to fulfill them. The team needs to have a similar idea and should be able to adapt when the context changes. It is important to talk about the similarities and differences (Bittner & Leimeister, 2014), that individuals perceive and contribute. Since a self-assessed situation may be biased and reflect only a perceived development, it is important that a team strengthens itself by sharing their perspectives. This is most valuable when team members complement each other, by having different personalities and perspectives. Indicators that can be drawn from this part is that a team has: 4) moments of reflection and evaluation (Bittner & Leimeister, 2014; Mohammed & Dumville, 2001) and 5) team members complement each other (Bittner & Leimeister, 2014; Mohammed & Dumville, 2001; Ohland et al., 2012) and have some level of experience. Tessem & Iden (2008) and Hackman & Wageman (2005) add that 6) all responsibilities and required skills should be clear and described.

Constructive Conflict

Teams should be able to create a constructive conflict in an attempt to understand each other (Van den Bossche et al., 2011). The differences should be discussed. All elements of the “conflict” should be addressed and there should be room for questions and conflict negotiation. There should be an environment that accepts disagreement. Psychological safety creates the trust for the team to share innovative ideas, without the need for being careful of what is shared. It should be encouraged that team members share divergent views (Bittner & Leimeister, 2014). Opinions and ideas are verified by asking each other critical questions. By implementing this the team creates a situation where an important decision is not taken without appropriate consideration (Bittner & Leimeister, 2014; Kleinsmann & Valkenburg, 2008). Team members share relevant information proactively, to ensure that the team has all the input that is required to make an informed decision. The indicators that could be drawn from this are: 7) psychological safety to share opinions (Bittner & Leimeister, 2014; Kolfschoten, Briggs & Vreede, 2009), 8) a culture of asking questions (Bittner & Leimeister, 2014; Kleinsmann & Valkenburg, 2008) and 9) team members that share relevant information (Bittner & Leimeister, 2014; Kolfschoten, Briggs & Vreede, 2009; Mohammed & Dumville, 2001).

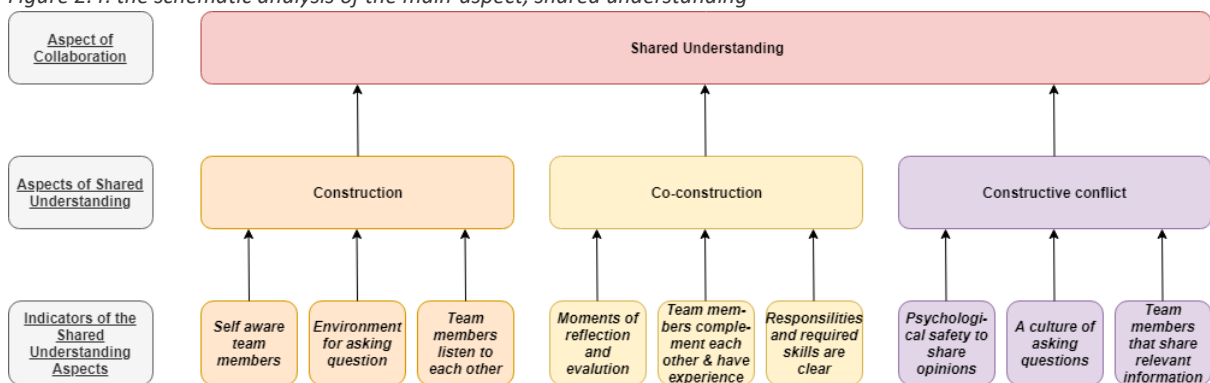
Table 2.11: indicators of having a shared understanding of cross-functional responsibilities

Indicators of having a Shared Understanding	Related to the Sub-aspect of Shared Understanding	Sources
<i>Self-aware team members</i>	<i>Construction</i>	<i>Berggren et al., 2017; Bittner & Leimeister, 2014; Van den Bossche et al., 2011</i>
<i>Environment for asking questions</i>	<i>Construction</i>	<i>Bittner & Leimeister, 2014; Van den Bossche et al., 2011</i>
<i>Team members listen to each other</i>	<i>Construction</i>	<i>Bittner & Leimeister, 2014; Mohammed & Dumville, 2001</i>

<i>Moments of reflection and evaluation</i>	<i>Co-construction</i>	<i>Bittner & Leimeister, 2014; Mohammed & Dumville, 2001</i>
<i>Team members complement each other and have some level of experience</i>	<i>Co-construction</i>	<i>Bittner & Leimeister, 2014; Mohammed & Dumville, 2001; Ohland et al., 2012</i>
<i>Responsibilities and skills should be clear</i>	<i>Co-construction</i>	<i>Bittner & Leimeister, 2014; Tessem & Iden, 2008; Hackman & Wageman, 2005</i>
<i>Psychological safety to share opinions</i>	<i>Constructive conflict</i>	<i>Bittner & Leimeister, 2014; Kolfshoten et al., 2009;</i>
<i>A culture of asking questions</i>	<i>Constructive conflict</i>	<i>Bittner & Leimeister, 2014; Kleinsmann & Valkenburg, 2008</i>
<i>Team members that share relevant information</i>	<i>Constructive conflict</i>	<i>Bittner & Leimeister, 2014; Kolfshoten et al., 2009; Mohammed & Dumville, 2001</i>

Table 2.11 depicts the indicators for a shared understanding. The sub-aspects, with all the indicators, of a shared understanding are depicted in figure 2.4. A team that is able to create an environment where many of these sub-aspects thrive, will be more likely to foster a shared understanding (Bittner & Leimeister, 2014; Kolfshoten, Briggs & Vreede, 2009; Mohammed & Dumville, 2001; Ohland et al., 2012; Hackman & Wageman, 2005). In comparison to the shared goal sub-aspects, the team will have more impact on enabling these sub-aspects themselves. Van den Bossche (2011) stated that collaborative groups need to express, share and listen to their individual understanding (construction) and discuss them to reach a shared understanding (co-construction). In several situations it is necessary to negotiate an agreement through constructive conflicts. Construction, co-construction and constructive conflict have a positive influence on collaboration (Bittner & Leimeister, 2014; Van den Bossche, 2011), within a DevOps context. All aforementioned indicators can be adopted and executed by team members (Van den Bossche et al., 2011). Although team members seem to have the most influence on shared understanding, also leaders around the team can help in providing the required climate (Bittner & Leimeister, 2014).

Figure 2.4: the schematic analysis of the main-aspect; shared understanding



2.2.4 Relationships

Based on the literature that has been discussed in paragraph 2.2.1, 2.2.2 and 2.2.3 it can be argued that having a shared goal and a shared understanding of cross-functional responsibilities has an influence on collaboration in a DevOps context. However, there is no common understanding about what these relationships entail and how an organization can approach these subjects.

Collaboration

A DevOps context is posed as a concept that could be used by an organization that aims to increase the adaptability of teams by reducing the gap between IS Development and IT Operations (Wettinger et al., 2014; Iden et al., 2012). One of the, commonly recognized, dimensions of DevOps is collaboration. Gupta et al. (2017) and Iden & Bygstad (2018) tie a shared goal directly towards DevOps. This study investigates the main-aspects of collaboration to determine the influence they have on collaboration in a DevOps context. Organizations that enable teams to have a shared goal (Kwak & Anbari, 2009; Cetindamar et al., 2005; Detzten et al., 2005) and a shared understanding (Bittner & Leimeister, 2014; Tsai & Hsu, 2014; Powers et al., 2014) will have more collaborative teams.

Selected main-aspects of collaboration in a DevOps context

According to the existing literature, it seems that an organization that has adopted DevOps will automatically need to focus on collaboration (Roche, 2013; Gupta et al., 2017). Meaning that they should try to create a culture that incorporates a shared goal (Kwak & Anbari, 2009; Cetindamar et al., 2005) and a shared understanding (Bittner & Leimeister, 2014; Tsai & Hsu, 2014). Both main-aspects are a dynamic state, that rely on the cohesion and finetuning of the team (Powers, Morgeson & Lyons, 2014; Van den Bossche et al., 2011) and its dependencies.

A shared goal and a shared understanding are important drivers of successful collaboration. As described above, a shared goal ensures focus. It corresponds with purpose. A shared understanding has nothing to do with a vision or a target. It corresponds with activities. Cross-functional teams that are able to understand their roles and responsibilities, are able to create a way of working in which they can plan, do, check and act in a logical fashion. When teams have a shared goal and a shared understanding, it increases the chance that the performed set of activities will fulfill the preset purpose.

Understanding the effect of the main-aspects on the ability of teams to collaborate is paramount for this study. The main-aspects are influenced by the sub-aspects. The theory suggests that the sub-aspects have a positive influence and this research has incorporated indicators in order to assess whether the sub-aspect are present.

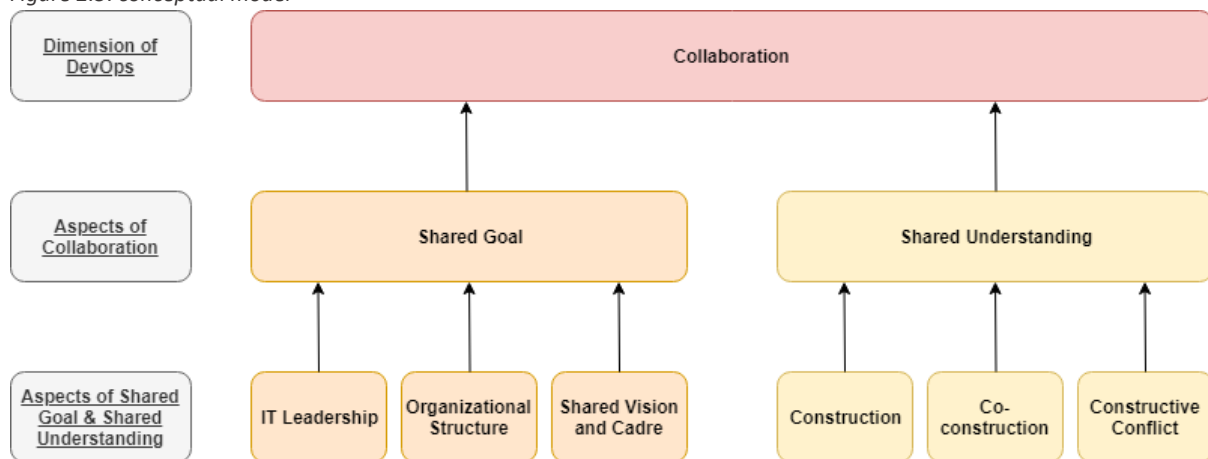
2.3 Conceptual model

Due to the ever-changing demands of the customer it becomes more important for IT organizations to have teams that collaborate effectively. This study strives to investigate the relationship between the main-aspects of collaboration, shared goal and shared understanding, and collaboration within a DevOps context. Figure 2.5 shows the conceptual model of this study. The figure consists of three layers. The first layer shows the dimension of DevOps; collaboration. This dimension has been defined as cross-functional (Bang et al., 2013; Lwakatere et al., 2015; Iden & Bygstad, 2018, Gupta et al., 2017) team members that execute activities (Bittner & Leimeister, 2014; Tessem & Iden, 2008; Iden & Bygstad, 2018; Charoensuk et al., 2014; Bygballe et al., 2016) in an attempt to fulfill a goal (Kwak & Anbari, 2009; Polat et al., 2018; Kozlowski & Bell, 2003; LePine, 2005; Alexander & Van Knippenberg, 2014).

The second layer of figure 2.5 is focused on the main-aspects of collaboration, namely shared goal and shared understanding. The two main-aspects can help teams to become more effective (Polat et al., 2018; Bittner & Leimeister, 2014; Iden & Bygstad, 2018). A shared goal is defined in paragraph 2.2.2 (Detzten et al., 2018; Kozlowski & Bell, 2003; Alexander & Van Knippenberg, 2014). It enables teams to perform (Polat et al., 2018; Cetindamar et al., 2005; Verbeeten et al., 2018) and collaborate (Alexander & Van Knippenberg, 2014; Kwak & Anbari, 2009; Polat et al., 2018) more effectively, which is influenced by the sub-aspects. A shared understanding is defined in paragraph 2.2.3 (Bittner & Leimeister; 2014; Mohammed & Dumville, 2001; Stout et al., 1999). It enables a team to perform and collaborate more effectively (Aubé et al., 2015; Mohammed & Dumville, 2002; Stout et al., 1999), which is influenced by the sub-aspects.

The third layer of figure 2.5 shows the sub-aspects of shared goal and shared understanding. These concepts are elements that influence the main-aspects of collaboration. For shared goal, the sub-aspects are: leadership (Vangen & Huxham, 2012; Thamhain, 2009), organizational structure (Jassawalla & Sashittal, 1998; Thamhain, 2009) and shared vision and cadre (Eldor, 2019). For shared understanding, the sub-aspects are construction (Bittner & Leimeister, 2014; Van den Bossche et al., 2011), co-construction (Bittner & Leimeister, 2014; Mohammed & Dumville, 2001) and constructive conflict (Bittner & Leimeister, 2014; Van den Bossche et al., 2011). There are several indicators, depicted in figure 2.2 and figure 2.3, that relate to these sub-aspects.

Figure 2.5: conceptual model



The theoretical foundation is depicted in a conceptual model, shown in figure 2.5. This model aims to address the concepts that have been discussed in paragraph 2.2.1, 2.2.2 and 2.2.3. The relationships have been described in paragraphs 2.2.4. The information presented in section 2.2 answers sub research question 1, 2 and 3. Therefore the focus for the remainder of this research will lie on sub research question 4 and 5. Chapter 3 will describe the research design, explaining which research methods have been applied and which data is gathered, interpreted and analyzed.

Chapter 3: Research design

After the analysis of literature and the creation of the conceptual model it is apparent that there are several authors that discovered relationships between DevOps and collaboration (Lwakatare et al., 2015; Gupta et al., 2017; Iden & Bygstad, 2018). While this is promising, the research regarding DevOps still seems to be in an explorative phase. Therefore, the foundation is slim and the validation of the revealed relationships remains in need of attestation. This study investigates DevOps and collaboration for organizations that develop software products.

Chapter 3 construes the research design which aims to corroborate the aforementioned relationships, that have been depicted in the conceptual model of the previous chapter. Section 3.1 explains the research approach and the arguments for applying mixed methods, while section 3.2 presents the selection process of the organizations that will be participating in this study. Section 3.3 introduces the data collection process, explaining the different steps involved. Thereafter, the data analysis methods are introduced, in section 3.4. The challenges regarding validity, reliability and triangulation are discussed in section 3.5.

3.1 Research approach

One of the most fundamental classifications of research approaches is the distinction between qualitative and quantitative oriented studies (Saunders et al., 2016). Qualitative studies attempt to understand how and why phenomena originate or develop, by assessing and including the environmental, contextual and human influence on the subject. A quantitative study follows the opposite approach, using constructed models, comprehensive theories and mathematical analysis to test a hypothesis, that often focuses on a what-question. Statistics are often used to decide whether the hypothesis has been significantly justified.

This study attempts to investigate how DevOps and collaboration are connected and how a shared goal and shared understanding contribute to the attainment of more collaborate teams. Therefore, a qualitative research approach is the most appropriate methodology to adopt for this research. It fits the maturity level of the scientific subject, because explorative studies often are qualitative studies (Saunders et al., 2016). An explorative study aims to gather more information, often triggered by the absence of sufficient knowledge. Considering the maturity of the research subject it is easy to explain why most studies regarding this subject focus on exploring, rather than determining or falsifying a hypothesis.

There still is no common understanding of what collaboration in a DevOps context entails and which main and sub-aspects are influential. Therefore, this study will employ an embedded case study (Yin, 2014), using mixed methods (Saunders et al., 2016). An embedded case study focuses on more than one sub-unit of the organization. According to Yin (2014) it is best to select a case study when it is critical to test a specific theory and reveal a situation. Yin (2003; 2014) argues that a multiple case study enables the possibility to analyze several situations and allows the researchers to investigate the similarities and differences. While this would make sense in a later stadium, the decision has been made to utilize the benefit of the case study that Yin (2014) posited. Gustafson (2017) argues that a single case study is often the better choice when a subject would benefit from a deeper insight. Which suits the explorative nature of this study.

Yin (2014) describes the attempt to find similar results via multiple case studies as literal replication. Vannoni (2014;2015) states that this can help the study to make relevant contributions to the existing literature. This could be an interesting second step, which goes beyond the scope of this study. Future researchers could strive for theoretical replication, explaining the differences found within an asymmetric set of case studies (Yin, 2014).

Three phases have been described, to validate the model of Chapter 2 in an efficient, viable and reliable fashion. For every phase there will be a set of questions in the semi-structured interviews. The acquired data will be analyzed to validate the results from the phases. This approach aims to gather qualitative data to substantiate the conceptual model.

Although this research should not be typified as an Action Research, there certainly are elements of the theorem that inspired this research design. The first two cycles that Saunders et al. (2016) typify as “teasing out the issues” and “understanding the customer and project” closely link to the semi-structured interview and the qualitative approach. Applying mixed methods ultimately results into triangulation, which is increasing the validity of the research. By utilizing mixed methods and investigating several departments the internal validity of the research is established. However, that it is hard to have a high validity in a case study. The next section introduces the case study selection process and the criteria that were set for the participants.

3.2 Case study

As mentioned in 3.1 this research will utilize a case study, to acquire practical insight regarding the conceptual model. The research design explicitly states that this is an embedded case study research. It is imperative that the research subject is actively introducing or applying DevOps practices.

The name of the organization that will participate will not be mentioned in order to protect their identity and the objectivity of this research. The respondents will be likely to answer more honestly. The research subject has his roots in the Netherlands. The pseudonym for the research subject will be DevOps1. Most of their employees are developing, delivering or testing software. There are also several account managers, customer care employees and managers. The employees who are required to deliver software, work in small cross-functional teams. Appendix 2.1 contains a brief summary of an unstructured interview with the Development Manager, the informant of this research. In table 3.1 the most relevant information is depicted. The goal was to determine whether the organization, DevOps1, was suited for this research. Based on the available respondents and the definition they hold for DevOps they were deemed suitable.

Table 3.1: research subject

CASE STUDY SUBJECT	EMPLOYEES	INTRODUCED DEVOPS	AVAILABLE RESPONDENTS
DEVOPS1	~200	Yes (2016-17)	As much as required

Their business model often results in long term relationships with the customers, that pay an annual price to retain the license for the software.

3.3. Data collection

As previously stated, the information will be gathered using two methods for data collection, specifically: semi-structured interviews and content analysis. By applying mixed methods, the conclusions that will follow from the data interpretation will be more viable. The three phases of data collection will be explained in the following paragraphs.

Semi-structured interview

The semi-structured interview is based upon the conceptual model. The questions that have been formulated are presented in appendix 2.2. This appendix shows the questions per subject (for instance: organizational structure or shared understanding). In appendix 2.3 is explained why the questions are selected for the interview. There will be two sets of questions, which are presented in appendix 2.4 and 2.5. These appendixes show the information that will be sent to the respondents. Appendix 2.4 is for the managers and appendix 2.5 is for the team members. One interview will focus more on shared goal and the other will focus more on shared understanding. The first interview will be called; Interview SG (shared goal) and the second interview; interview SU (shared understanding). Interview SG will focus on respondents 1-3 and Interview SU on respondents 4-7, see table 3.2.

The reason for this approach is that the theoretical framework showed that leaders have more impact on the main-aspect; shared goal and team members have more impact on the main-aspect; shared understanding. It remains important to cross-validate and therefore, both interviews will have questions regarding all the main-aspects and sub-aspects.

To form a representative group, it is crucial to carefully select the respondents. Although it is impossible to create an optimal objective environment, a group of respondents which incorporates the diversity of roles and views helps to reduce the respondents bias (Saunders et al., 2016). Three of the respondents will have a managerial position and four of them will be a team member in a DevOps context. Two of the team members will have development as his or her main responsibility and the last two will be working in a DevOps team with operations as his or hers main responsibility. The respondents are presented in table 3.2.

Table 3.2: Respondents

CASE STUDY RESPONDENTS	CATEGORY	ROLE(S)/TASK(S)	COMPANY
RESPONDENT-1	Managerial	Technical manager	DevOps1
RESPONDENT-2	Managerial	Lead architect, team coach	DevOps1
RESPONDENT-3	Managerial	R&D manager, team coach	DevOps1
RESPONDENT-4	Team member	Architect, operations, developing	DevOps1
RESPONDENT-5	Team member	Developing, design	DevOps1
RESPONDENT-6	Team member	Developing, quality assurance, operations	DevOps1
RESPONDENT-7	Team member	Operations, developing	DevOps1

As depicted in table 3.3, the respondents have to be a medior or senior, furthermore they need to work longer than two years for DevOps1. The informant explained that there was a transition in 2016-2017, it is beneficial if the respondents were a part for that process. This will be a criterion for the managerial respondents, because this will shed light on the actions that have been performed since then. Another managerial criterion is that the leaders are not working in the teams, but do carry a responsibility regarding the output of the team. By selecting this subset of managers, the focus will more likely be on the reason why the team exists and the elements that are important for shared goal (leadership, organizational structure, vision). Since it is an embedded case study it is important to speak to managers of several departments. This will give an insight in what are the common practices of DevOps for DevOps1. For the team members of a DevOps team it is important that they are aware of the term since they have to be able to contribute to the research question. Furthermore, the respondents will be picked from several teams, so that different team cultures are discussed. This helps to distill the most important practices.

Table 3.3: Criteria of respondents

General Criteria	Managerial criteria	Team member criteria
Medior / Senior	Part of the transition process	Aware of DevOps
Longer than 2 years employed by DevOps1	One layer or more above the team	Several teams
	Several units	

All the respondents work in one of the three business units of DevOps1. One of the business units is called; core, they build the platform of the organization. The other two business units are perceived as a customer of the core business unit. These two units are specialized in specific markets, where DevOps1 has a lot of customers. All the business units develop software and have contact with the real customer, but the core unit delivers the backbone that the other departments use to make a difference for their customers. In 2016 there was a transition in the core unit, moving more towards DevOps. There was an enormous need to improve the time to market. In the years that followed the other business units also moved towards DevOps. The business units started to focus on making products that will help all the customers. They have dedicated teams to specific parts of the application, to become more effective. All the teams that have been created in the last few years have a cross-functional character and are customer focused. The respondents, selected based on the criteria depicted in table 3.3, work in one of the three business units. The managerial respondents are only outranked by a managing director. The team members have a product owner who prioritizes epics, a team lead that helps them in their personal career and an architect who acts as a technical coach.

The semi-structured interview is a very applicable data collection method, especially for qualitative studies (Saunders et al., 2016) due to the flexibility that is built in to adapt on new insights. This method seems to fit the research, considering the sub research questions that have a broad narrative.

The interviews will take place at the office of the participating organization. All respondents will be informed about the main subject and structure before the interview starts. They will also be asked to

accept a recorded session and sign a letter of consent, which is attached in appendix 2.6. The letter of consent will be accompanied by an introduction letter, which is attached in appendix 2.7, and attachments (a copy of the conceptual model (paragraph 2.3) and the interview protocol (appendix 2.4 and 2.5)). The interview will be transcribed and respondents will be asked whether they accept the paraphrases. The estimate is that the interview will last about 60-75 minutes. A copy of the transcription will be sent to all respondents, so that they can control the gathered material. If one the respondents request to alter data, then this will be executed without hesitation.

The interview will be conducted in Dutch. The transcripts are also in Dutch, because that is the native tongue of the respondents. Later on the results that needed for quotation or explanation are translated. Therefore a quote from a respondent is always a translated quote.

Semi-structured Interview: phase 1

During the first phase the questions will concentrate on the role of DevOps within the organization. The goal is to establish a first impression of DevOps and collaboration, which will establish whether the findings regarding sub research question 1 are corroborated by the practices in DevOps1. Furthermore, it will help to have some background information before the research focusses on the main-aspects and sub-aspects. Since sub research question 1 has been answered already in chapter 2, this part of the interview will be brief.

Semi-structured Interview: phase 2

During the second phase of the interview the questions will concentrate on the main part of the interview. For the managers this will be shared goal and for the developers and operations personnel this will be shared understanding. The goal is to assess the influence of a shared goal or shared understanding on collaboration in a DevOps context. The analysis will start with a brief introduction of the term. Subsequently, open questions will be posed in an attempt to establish whether the concept is present at DevOps1 and what the impact is on collaboration.

Table 3.4: Phase 2

Phase 2		
Topic	<i>Respondent</i>	<i>Answers</i>
Shared Goal	Manager	Sub Research Question 4
Shared Understanding	Developer / Operations	Sub Research Question 5

The goal is to establish an insight of the influence of the main-aspects, shared goal and shared understanding, on collaboration. Table 3.4 shows which topic is handled and which sub research question is answered based on the role of the respondent.

Semi-structured Interview: phase 3

During the third phase of the interview the questions will concentrate on the corroborative part of the interview. For the managers this will be shared understanding and for the developers and operations personnel this will be shared goal. The goal is to cross check the answers that the other

group gave regarding the influence of a shared goal or shared understanding on collaboration in a DevOps context. The analysis will start with a brief introduction of the term. Subsequently, open questions will be posed in an attempt to establish whether the concept is present at DevOps1 and what the impact is on collaboration. This phase of the interview will be considerably shorter than phase 2.

Table 3.5: Phase 3

Phase 3		
Topic	Respondent	Answers
Shared Goal	Manager	Sub Research Question 5
Shared Understanding	Developer / Operations	Sub Research Question 4

The goal is to establish an insight of the influence of the main-aspects, shared goal and shared understanding, on collaboration. Table 3.5 shows which topic is handled and which sub research question is answered based on the role of the respondent.

Due to the semi-structured nature of the interview there is room for in depth questions when the situation requires it. After the interview, the interviewee will get information regarding the next steps. The transcription will be sent to enable them to rephrase if necessary.

Content analysis

The semi-structured interviews will be strengthened by employing content analysis (Elo & Kyngäs, 2007), enabling an enhanced analysis. The documents which are required to do a content analysis will be retrieved before the interviews. It is possible that extra documents are retrieved after the interview, depending on the output of the interviews. Strategic and policy-oriented documents regarding DevOps and the implementation of DevOps will be used to substantiate the results from the semi-structured interviews. Table 3.6 depicts the relevant sources based on the sub-aspects of shared goal and shared understanding.

Table 3.6: Content analysis - documents

Subject		Relevant Documents	Reason
Shared Goal			
	Leadership	Goals for management Strategy regarding team management	Gives an insight in the indicators of leadership
	Organizational Structure	Organogram and related documents like how to organize teams and autonomy	Gives an insight in the indicators of organizational structure
	Shared Vision & Cadre	Strategy documents Responsibilities per team/individual	Gives an insight in the indicators of shared vision and cadre
Shared Understanding			

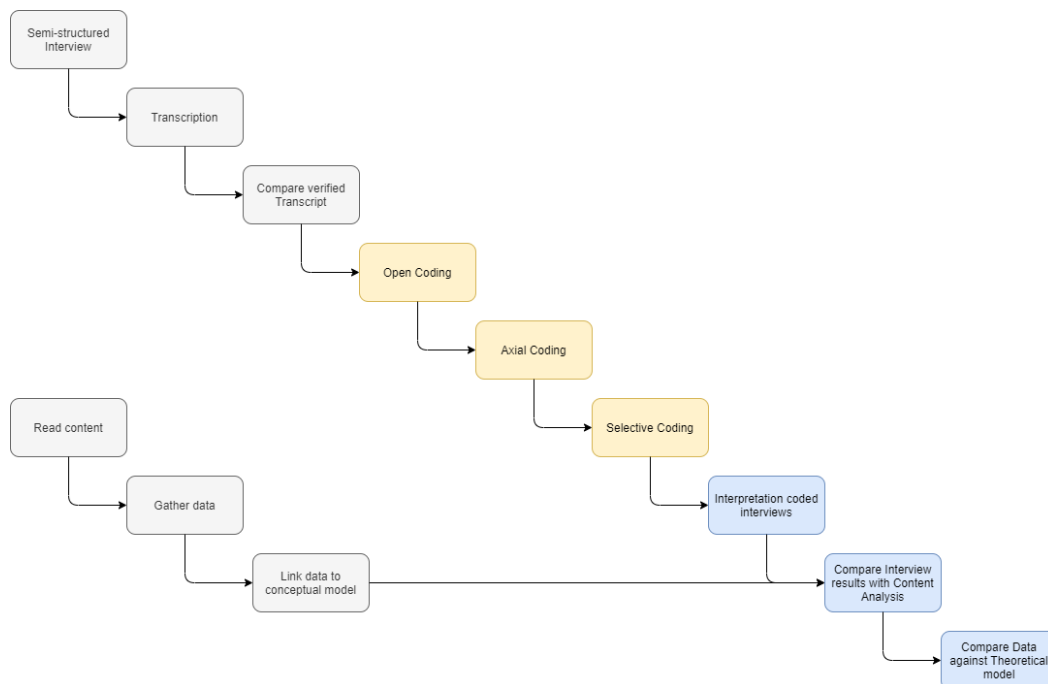
	Construction	Documents that describe personal growth elements (like self-awareness) or exercises to understand each other	Gives an insight in the indicators of construction
	Co-construction	Documents that describe team building activities, way to select team members based on personality of competence or team growth elements	Gives an insight in the indicators of co-construction
	Constructive Conflict	Documents that provide information regarding discussions, evaluations et cetera.	Gives an insight in the indicators of constructive conflict

3.4 Data analysis

The data analysis process is based on the interviews, the content analysis and the theory of Chapter 2. Data will be analyzed after the semi-structured interviews have been conducted. Figure 3.1 shows the steps in the process from conducting the semi-structured interviews till an overarching comparison between the data and theoretical model.

When the interviews are transcribed, they can be compared and the coding process can begin. This part of the process aims to prepare and structure the data for a coherent analysis. After these steps, the interview results can be interpreted. When this is done the gathered data from the content analysis is used to substantiate or falsify the results from the interviews. At this point all the gathered data has been considered and analyzed. The last step is to compare the findings against the theoretical model.

Figure 3.1: Data process



From “Semi-structured interview” to “Selective coding”

The steps, that relate to the abovementioned title, are depicted in figure 3.1. When the respondents agree with the results, the transcriptions will be compared. Thereafter, the interviews will be coded based on three coding stages of Strauss and Corbin (1998). This ensures a structured approach; starting with a simple analysis regarding the categories that exist, moving towards relations between the categories and ending with main categories and its relationships. This approach fits the construction of the theory of Chapter 2, because there are three layers; indicators, sub-aspect and main-aspects. Stage 1, open coding, reorganizes data into categories. For this research that will be the indicators. This step is executed by reading the entire transcribed interview and attaching codes, the indicators, to text fragments. The attached code is an indication of the subject of that fragment.

Stage 2, axial coding, aims to find similarities between the aforementioned categories. The codes of the “open coding”-stage are compared to each other and related codes are merged in an overarching category. Every code can belong to multiple overarching categories. In this research the sub-aspects will be the foundation for defining the overarching category of axial coding.

While stage 3, selective coding, integrates and identifies essential categories, enabling an abstract interpretation of the retrieved data. The main categories found in the “Axial coding” -stage are used as the input for the selective coding. The connections between the main categories are used to describe relationships. In this research the main-aspects will be the foundation for defining the essential categories.

An important insight for this study is that the semi-structured interview contains layers, meaning that some questions focus more on the indicators and some questions focus more on the sub-aspects. Therefore, it depends on the layer whether all the three stages of coding are executed.

These three stages form an abductive method (Suddaby, 2006), which fits the earlier mention of a mixed methods approach. Table 3.7 depicts the details for open, axial and selective coding. The build-up of the coding strategy reflects the conceptual model. The main-aspects of the theoretical model will be the umbrella for the sub-aspects and the indicators. Therefore, open coding will utilize the indicators, axial coding the sub-aspects and selective coding the main-aspects in order to structure the data. By using this approach, it becomes possible to compare the theory against the data.

Table 3.7: Coding

Sub research question	Type of Coding	Main-aspect	Sub-aspect	Indicators per aspects
4	Selective	Shared goal		
	Axial		Leadership	
	Open			Benefit of working together is clear and communicated
	Open			Leaders know the environment and senior management
	Axial		Organizational Structure	
	Open			Decentralized day-to-day Decisions
	Open			Individual team members have focus on the team

	<i>Axial</i>		<i>Shared vision and cadre</i>	
	<i>Open</i>			<i>Strategy and expectations of the organization are known</i>
	<i>Open</i>			<i>Opportunities and limitations are known</i>
5	<i>Selective</i>	<i>Shared understanding</i>		
	<i>Axial</i>		<i>Construction</i>	
	<i>Open</i>			<i>Self-aware team members</i>
	<i>Open</i>			<i>Environment for asking questions</i>
	<i>Open</i>			<i>Team members listen to each other</i>
	<i>Axial</i>		<i>Co-construction</i>	
	<i>Open</i>			<i>Moments of reflection and evaluation</i>
	<i>Open</i>			<i>Team members complement each other and have some level of experience</i>
	<i>Open</i>			<i>Responsibilities and skills should be clear</i>
	<i>Axial</i>		<i>Constructive conflict</i>	
	<i>Open</i>			<i>Psychological safety to share opinions</i>
	<i>Open</i>			<i>A culture of asking questions</i>
	<i>Open</i>			<i>Team members that share relevant information</i>

“Content analysis” to “Compare Interview results with content analysis”

As depicted in Figure 3.1 there are two data-streams. One is the semi-structured interviews and the other is the content analysis. The content analysis is not very extensive, mainly due to the limited number of documents that were deemed relevant. However, it helps to falsify or substantiate the interview results. The first step is reading the content. Secondly the relevant data will be gathered and structured based on the conceptual model. By ordering the data in this fashion, it is easier to compare the results.

After the interpretation of the coded interviews the data will be compared with the data from the content analysis.

Interpretation coded interviews – Compare Data against Theoretical model

The result of the coding steps is a coherent set of data, which can be seen as the output from the semi-structured interviews. The most important conclusions will be described. Now that the data analyzation process of the semi-structured interviews has been performed it is time to analyze the required set of data. The results of the coded transcripts will be compared against the results of the content analysis.

At this point the retrieved data is fully analyzed. The combined set of data shall be compared against the conceptual model, especially regarding sub research question 4 and 5. The conceptual model will be used to assess the relationships, resulting in an answer on the sub questions of this research. The conceptual model, with all the indicators, sub-aspects and main-aspects is based on definitions and assumptions. It will be interesting to compare the theory against the retrieved data.

3.5 Validity, Reliability & Triangulation

This section will focus on the risks and countermeasures of this research. Furthermore, the subjects of validity, reliability and triangulation are described. The focus of the enumeration, listed underneath, will be on the ethical aspects that could play a role during a research. For every aspect is explained what the countermeasures have been.

1. Explicit permission of usage from respondents
 - *Risk*: The researcher could expect that the approval of conducting an interview means that the data can be used.
 - *Countermeasure*: The respondent signs a letter of consent and will be told that after that he or she still has the possibility to revise the story (appendix 2.1). Furthermore, the respondent will be briefed of his rights during and after the interview (appendix 2.4 and 2.5, methods 3.3). One of the things that will be asked explicitly, is whether the respondent accepts anonymous citations (appendix 2.4, 2.5 and 2.6).
2. Reliability of data and the anonymity of respondents
 - *Risk*: Anonymity of respondents could be a harmful for the respondent and the research. By securing the anonymity the reliability of the research can be increased.
 - *Countermeasure*: There are three groups of interviewees; developers, operations personnel and managers, and a total of seven respondents (methods 3.3, appendix 3). None of the interviewees will have access or insight into the data of the other participants (appendix 2.6, methods 3.2). Furthermore, all the transcripts will be handed to the respondent before publishing anything (appendix 2.6, methods 3.3). If needed quotations will be paraphrased to remain anonymity (appendix 2.6).
3. Privacy of the respondents
 - *Risk*: Respondents could expect or even have negative consequences by attending and answering the questions.
 - *Countermeasure*: The interviews will be one on one (methods 3.3, appendix 2.6). All the transcribed information will be anonymized and drafts will be given to the respondents for a respondent approval (methods 3.3, appendix 2.6). Personal data will be handled with care (appendix 2.6).
4. Voluntary participation and the right of withdraw
 - *Risk*: The respondent could feel coerced to participate in the research and the interview, due to a lack of alternatives.
 - *Countermeasure*: The respondent is informed, before and after the interview, that he or she can choose to withdraw at any moment during the research (appendix 2.6).
5. Avoidance of detriment
 - *Risk*: The researcher misunderstands the respondent, which could lead to a potentially harmful statement for the business and or the respondent.
 - *Countermeasure*: The respondent will always be asked to approve the corresponding transcript before anything is published (methods 3.3, appendix 2.6).

Validity

Saunders et al. (2016) state that case studies are an effective instrument to research a subject in his natural context. As Yin (2002) states, a case study gives a researcher the ability to compare situations within their specific context.

A qualitative study is harder to validate internally, therefore it is important to reduce bias (Saunders et al., 2016). This has been attempted by using several respondents of every type of respondent, therefore reducing the effect of the bias of one respondent. It is also important to reduce the researchers bias, therefore transcripts and data analysis are shared with respondents. Therefore, giving them the opportunity to intervene if deemed necessary.

Next to internal validity, there is also external validity. This has been described by Saunders et al. (2016) as the ability to generalize the results of the study, meaning that it is arguably applicable to other contexts than the one that has been researched. It is hard to externally validate a case study, therefore it is crucial to clearly describe the utilized research methods. By doing this the external validity is maximized, but it remains hard to generalize the results of a case study.

This is caused by the specific context of a case study, one could argue that an embedded case study strives to reduce this effect (Saunders et al., 2016). To ensure a higher internal and external validity the embedded case study method has been opted for this research. Another way to ensure the validity is by not overselling the generalization power of the case study (Flyvbjerg, 2006), this is what this research has done by explicitly stating the limitations of this research design. The application of a case study is appropriate due to the goal to assess a conceptual model in the real world and to explore its concepts.

Reliability

The respondents play an important role in the reliability of this research. The transcripts have been anonymized, the participating organizations have been pseudonymized and the interviews are conducted in a closed session. It is important that respondents feel free to talk and give their honest opinion, therefore it is crucial that a safe environment is created (Saunders et al., 2016). As mentioned before the answer will be transcribed, but it will not be used before asking for confirmation from the respondents. To reduce the chance of different interpretations. Every respondent will get a copy of the transcript so that they can also challenge the data at any given moment when they feel like it has been documented incorrectly.

Triangulation

Triangulation is used to increase the internal validity of this research. The mixed methods approach helps to achieve this. Different data collection and analysis methods have been used, to decrease the chance of bias or mistakes. Participants with different roles and responsibilities, have been selected to reduce the subjectivity of the respondents. Furthermore, the participants come from several departments within DevOps1 to strengthen this effect.

By employing a case study to investigate the influence of a shared goal and shared understanding on collaboration the research design is suitable. Semi-structured interviews and the three coding stages (Strauss and Corbin, 1998) are used to boost the validity and reliability.

Chapter 4: Results

This section contains the results of the research. Section 4.1 explains how the research was implemented. The contextual information is presented in section 4.2. The results of the semi-structured interview are depicted in section 4.3, while the content analysis is discussed in section 4.4. Subsequently, the data is compared against the theory in section 4.5. The last sections, 4,6 and 4,7, formulate an answer on sub research question 4 and 5.

4.1 Research implementation

Chapter 1 posed five sub research questions. The last two of them still need to be answered:

- How does a shared goal influence collaboration in a DevOps context?
- How does a shared understanding of cross functional responsibilities influence collaboration in a DevOps context?

Chapter 2 answers the first three sub research questions and provides guidance for the remainder of the research. The results of the interviews supplements the conceptual model. In the first sections the results are depicted and later on the last sub research questions are answered. Appendix 4 incorporates more detailed results and analysis.

As described in chapter 3, this research is an embedded case, which uses interviews. The interviews were conducted in the summer of 2019. Every interview took between the 60-90 minutes. There always is a learning curve and lessons and takeaways will be discussed in chapter 5.2: discussion. It is important to state that this is an explorative research, due to the qualitative nature of this research and the small pool of respondents.

There are seven respondents, 3 managers and 4 team members. Respondent4 and Respondent6 work in the same team and Respondent5 and Respondent7 work together in another team. All respondents filled in the consent form and were willing to answer all questions. That resulted in detailed transcripts, which were coded. The results of the coding process can be found in Appendix 3.

4.2 Research subject – Contextual information

The information presented in this section is gathered during the preparation with the informant and the semi-structured interviews. Paragraph 4.2.1 will introduce the organization and will briefly state the status quo of the organization. Paragraph 4.2.2 will focus on the business unit and paragraph 4.2.3 on the teams, with a focus on teams that are mentioned during the interviews. Paragraph 4.2.4 will present the respondents.

4.2.1 Organization

The research subject is an organization that develops software that enables the developers of their customers to build better software. They are specialized in making generic solutions that can be used by customer-based developers to make tailor-made products.

The organization has been split up in three business units, each with their own focus area. The three units are: Core, Finance & Public. The Finance and Public unit focus on specific markets, wherein the

organization has a large customer base. The Core unit is responsible for developing the product that all three units sell to their customers. Core is the unit that has the lowest number of direct customers, because most of the employees are working on developing and maintaining the product. If Finance or Public wants a feature that is not supported by Core, there are two options. The first option is that Finance or Public convinces Core to build it and the second option is that the relatively small development capacity of Finance or Public is used to build this feature. This situation creates a challenge regarding vision. The three business units can have different interests, but depend on the same product. In practice, Core is often the unit that defines the product strategy.

In 2016 the product was not stable and new releases resulted in several problems. That was the main trigger to make the transition to a DevOps organization. It is a decision that has had a big impact on the internal organization. An important goal, which was shared as a dream, was to always be able to release a new high-quality version of software. That dream impacts the way they need to automate and collaborate. It started at Core, but the other business units also entered the transformation. Before 2016, most of the employees of Core were developers that worked on the client side. Nowadays, most of the developers are working on a product, which they can sell to customers. The research subject wants to be the owner of the software they produce.

4.2.2 Business Units

The business unit public is a self-sufficient business unit, which offers services to the central government. These services are provided by 60 employees and the focus has been chosen based on the strong market position. By focusing more on this market there should be potential to grow.

The business unit core is a self-sufficient business unit, which develops a product that is used by internal and external customers. This business units exist of 40 employees and most employees focus on ensuring they have a qualitative and innovative product. The building blocks that their application offers need to be user friendly. Furthermore, it is important to find the required level of abstraction, so the building blocks are fit for use for several businesses. On the other hand, they should be unique and well scoped, so that the building blocks are maintainable. Besides the technical discussions, there is also the political discussion. Three business units need to be aligned regarding the product strategy.

The business unit finance is similar to the unit finance. They are depending on the product that the unit core develops and maintains. They have customers in a special vertical and need to define a product vision in collaboration with the other two business units. In this research the business unit finance did not partake in the interviews.

The business units are separate entities. However, they also depending on each other. That is an interesting situation, which can make it hard to align a product strategy. There are professionals that have a responsibility to manage that. Some of them are part of a holding company, which owns the research subject.

4.2.3 Teams

The teams have cross-functional developers, who work together in sprints. Each individual team member has a team lead, who is responsible for their development. However, this team lead can differ per individual team member.

The business unit core has three teams. Two of them are interviewed; Rubic and 69. There is also a team in Romania, named; UTB. This team focuses on the non-functionals. The teams decided the names for themselves. The goal of the organization is that the teams are self-organizing and autonomous.

Team Rubic seems to be the most mature team. It is a team with many seniors and the team composition has been stable. They have a clear responsibility, which is to replace the current studio. Team 69 consists of more junior employees and the team composition has not been very stable. They focus on the process engine and runtime. The leadership team has given both teams the freedom to develop their own culture. Team 69 is known for their focus on quality, while team Rubic is known for their focus on productivity.

The business unit public contains one team. Which has been formed since a year ago. Prior to this formation everybody was working at the client site. Respondent1 is the manager of the team and they have followed some of the best practices from the teams of core. The teams of core started to transform to DevOps teams in 2016.

4.2.4 Respondents

Respondent 1 is the technology lead of the business unit; public. He is part of the MT and leadership team and his main responsibility is managing R&D, maintenance and support. So, ensuring that the technology is present to be successful in the market. In this role he often has contact with the architects, manager of the team, developers, customer success managers and the maintenance & support department. Furthermore, he often has contact with the other business units. Mostly, with the business unit; Core.

Respondent 2 is the lead architect of the business unit: core. He works nine years for DevOps1, first as a developer and later as an architect. Since 2016 he is the product architect and since 2017, he is also a team lead. The PO's, who are managed by Respondent3, and the architects do most of the translation to the teams. As a lead architect Respondent2 is responsible for managing the other architects (product and projects with customers), ensuring that standards are met and that the product fits to the strategy. 'We have smart people, it's important to challenge and help them to think and decide for themselves'.

Respondent 3 is the Research & Development manager of the business unit: core. He started his career as an IT consultant, with focus on quality, project-management and management. DevOps1 is his employer since 2016. His main focus is to ensure a qualitative product that fits the needs of customers and the future. He focusses on reducing waste. Secondly, he has a HR responsibility, for some of the employees, as a team lead. Facilitating a culture of growth and autonomy is important for this respondent. "We need to be scalable and I believe the best way to ensure that is increasing the autonomy of teams". Respondent3 forms a Leadership Team with Respondent2. Together they are responsible for the product. Respondent2 coaches and manages the Product Owners.

Respondent 4 is one of the architects that reports to respondent2, therefore he also works in the business unit: core. Furthermore, he works as a team lead, so he also has a HR responsibility. Due to his architectural responsibilities and background in development he often works in one of the teams. His team input is committed to the team; 69. The informant described him as someone who can discuss the shared understanding within the team, because he is actively part of one team. He has been working for almost 10 years at DevOps1. His goal is to make the teams decide for themselves,

based on good architecture principles. Aside his main activities, he is also responsible for third-party license dependencies.

Respondent 5 is one of the software engineers of team Rubic. He has been working for DevOps1 for more than three years and started immediately after a study informatica. Although his limited experience, his function title is Senior Software Engineer. The first year he worked on projects at the client side. Since 2017 he has been working on product development. Team Rubic has the goal to replace a legacy application. His main activities are thinking about the steps and techniques that are necessary to ensure a future proof design and implementing these steps together with the other software engineers.

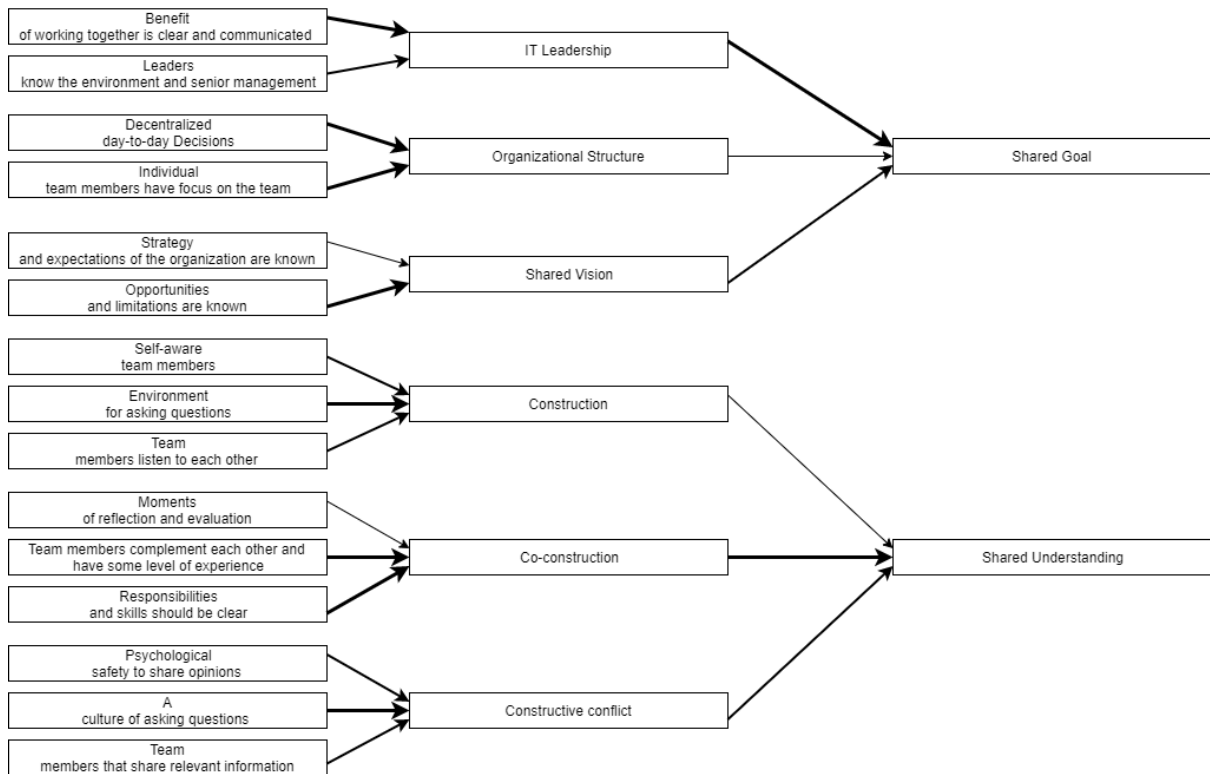
Respondent 6 is one of the software engineers of team 69. He has been working for DevOps1 for more than five years. First, he was responsible for operations and maintenance. The last three years he has been working on product development. Prior to this employment he worked for one of the customer of DevOps1. Team 69 focusses mostly on runtime. As a Senior Software Engineer, he coaches other team members, remains oversight and focusses on quality. He is a developer that takes the operational and quality related tasks very serious. Therefore, he also focusses on tasks like; monitoring builds, monitoring CI servers and ensuring test automation is implemented according standards.

Respondent 7 is one of the software engineers of team Rubic. He has been working for DevOps1 for more than three years. DevOps fits with his mindset. He started his career in IT maintenance and developed himself towards a role as software engineer. This package of skills drives his activities as a team member. He maintains builds, CI servers and documentation. Therefore, he is well aware of what needs to be done to be successful in the long run. Next to his operational activities, he also contributes by engineering new functionalities.

4.3 Research results - Interviews

As depicted in table 3.7 the data of the interviews is bundled in an orderly fashion. The open coding focused on the information that was retrieved regarding the indicators. Based on that bundled information the axial coding was performed, in an attempt to establish a link towards the sub-aspects. The selective coding focused on the link towards the main-aspects. The main goal was to investigate the conceptual model and its indicators in practice. Furthermore, an attempt was made to visualize the perceived relative relevance. Figure 4.1 shows an outcome from that analysis. This figure is based on the information that the respondents gave regarding the indicators and their impact on the sub-aspects and main-aspects. The thickness of the arrow shows how relatively strong the relationship is. The following sections will show the reasoning behind this figure.

Figure 4.1: research result – relationships



4.2.1 Indicators

The results of the data analysis regarding indicators are discussed in this section. 3 leaders/managers and 4 team members were interviewed. Due to the time limit, the interviews with the managers had a focus on the indicators of a shared goal. Similarly, the interviews with the team members focused on the indicators of a shared understanding. The most relevant information per indicator can be found in Appendix 4.1.1.

A system was introduced to quantify the results. This helps to visualize the data set. However, this research is a qualitative study and therefore the argumentation will be emphasized over the system. The data, regarding the various indicators, of the respondents would be categorized as yes (y), no (n) or ambiguous (a). Most of the data for this overview was retrieved with interview questions 6, 7, 9, 10, 12, 13, 19, 20, 21, 23, 24, 25, 27, 28 and 29. A yes would lead to 1 point, an ambiguous to 0.5 point and a no to 0 points. The argumentation for the scores is summarized in Appendix 4.1.2 and further explained in Appendix 4.1.3. The total sum of the points would be divided by the number of respondents. So, for shared goal;3 and for shared understanding; 4. The maximum number of points for shared understanding would be 4, this would indicate that the indicator is very much present in the organization. The minimum number of points; 0, would indicate the opposite.

In an attempt to categorize the results, there are two benchmarks. For shared goal, with three respondents, a score equal or above 2.5 is deemed a high score (green). Furthermore, a score lower than 1.5 is deemed a low score (red). There is also an average score (orange) which contains out of the scores in between the two limit values. For shared understanding, with four respondents, a score above 3 is deemed a high score (green). Furthermore, a score lower than 2 is deemed a low score (red). The average score (orange) contains out of the scores in between the two limit values.

Table 4.1: research result – score of indicators

Sub-aspect	Indicator	Score				
			R1	R2	R3	
IT Leadership	Benefit of working together is clear and communicated	y	y	y	y	
	Leaders know the environment and senior management	y	y	a	n	
Organizational Structure	Decentralized day-to-day Decisions	y	y	y	y	
	Individual team members have focus on the team	a	a	y	y	
Shared vision & cadre	Strategy and expectations of the organization are known	a	a	n	n	
	Opportunities and limitations are known	y	y	y	y	
			R4	R5	R6	R7
Construction	Self-aware team members	y	y	a	n	y
	Environment for asking questions	y	y	y	y	y
	Team members listen to each other	n	n	a	y	a
Co-construction	Moments of reflection and evaluation	n	n	a	n	y
	Team members complement each other	y	y	y	y	y
	Responsibilities and skills should be clear	y	y	y	y	y
Constructive conflict	Psychological safety to share opinions	y	y	a	a	y
	A culture of asking questions	y	y	y	y	y
	Team members that share relevant information	a	a	y	a	y

The scores of green, orange or red resemble with the thickness of the arrows in figure 4.1, that link the indicators to the sub-aspects. A red arrow results in a narrow arrow, while a green score results in the thickest arrow. It is important to stress that this is a model, based on a limited number of respondents. The figure is for visualization purposes.

4.2.1.1 Indicators – Interesting remarks or relations

Appendix 4.1.1 portrays the relevant information per indicator and Appendix 4.5.1 compares the data against the theory. This section introduces the most relevant information, that is retrieved, regarding indicators.

Team members have similar answers

Respondent4 and Respondent6 are team members and Respondent5 and Respondent7 are team members. Table 4.2 shows a comparison of the data that was retrieved from the team members.

Table 4.2: research – team members compared

		69		Rubic	
		R4	R6	R7	R5
Construction	Self-aware team members	y	n	y	a
	Environment for asking questions	y	y	y	y
	Team members listen to each other	n	y	a	a
Co-construction	Moments of reflection and evaluation	n	n	y	a
	Team members complement each other	y	y	y	y
	Responsibilities and skills should be clear	y	y	y	y
Constructive conflict	Psychological safety to share opinions	y	a	y	a
	A culture of asking questions	y	y	y	y
	Team members that share relevant information	a	a	y	y

In both cases there are three instances where the team members disagree. For team Rubic the score is never farther apart than 0.5. If we look at non-team members there is a different picture. See table 4.3.

Table 4.3: research – two sets of non-team members

		69		Rubic	
Team name		R4	R5	R7	R6
Construction	Self-aware team members	y	a	y	n
	Environment for asking questions	y	y	y	y
	Team members listen to each other	n	a	a	y
Co-construction	Moments of reflection and evaluation	n	a	y	n
	Team members complement each other	y	y	y	y
	Responsibilities and skills should be clear	y	y	y	y
Constructive conflict	Psychological safety to share opinions	y	a	y	a
	A culture of asking questions	y	y	y	y
	Team members that share relevant information	a	y	y	a

This time there are five different answers in both comparisons, instead of the three of table 4.2. This is an indication of the differences in team culture and maturity that might be present. Building on that the case could be made that this suggests the presence of some level of shared understanding within the teams. In this study shared understanding is defined, in short, as a concept that creates clarity and alignment with respect to an object of understanding. If there are two teams, where this process takes place, it makes sense that the differences of individual perceptions are bigger between non-team members than team members. Table 4.2 and Table 4.3 corroborate that suggestion.

Indicators of organizational structure

It is quite interesting that the indicators of organizational structure both have a high score. It is the only sub-aspects whereof the indicators all are green. It could mean that the management have thought well about the things that are necessary for teams to have enough structure. The interesting thing is that the team members, Respondents 4-7, did not see organizational structure as an important element. They did talk a lot about autonomy, which corresponds with the indicator: decentralized day-to-day decisions.

Strategy and expectations of the organization are known

This indicator has a low score of 0.5/3. The only manager that was ambiguous about it was Respondent1. All three respondents describe the problem of setting a strategy that fits the three business units and the platform vision. Some of the respondents think that it makes more sense to let the owners of the platform set the strategy. It is obvious that this indicator reflects a problem for the organization and it also a concern that is shared by respondents 4, 5, 6 and 7. They also see that there is not much clarity. Luckily, the cadre is pretty clear. Therefore, they do not have the guidance of a clear vision, but they do have the handles of a clear cadre.

It is interesting that all respondents see that they should improve here and that it would help the teams. The red score and that notion combined give a clear signal that this is a big point for improvement. So, although the score is red, they do think there is a real relationship between the indicator and the sub-aspect. It is just not embedded effectively in the organization yet.

Moments of reflection and evaluation

The same logic holds for the indicator: moments of reflection and evaluation. Respondent7 is the only respondent who seems to think they perform well regarding this indicator. The other team respondents are less enthusiastic. The other group, with Respondents 1-3, is ambiguous. They do describe a culture of giving feedback, but they also state that retrospectives could be sharper and that not every team is able to improve this without help. Although they are critical about the way they are performing, they do see the relation between the indicator and sub-aspect.

It is interesting to mention that Rubic scores all the points here. They seem to think they are quite capable of giving feedback. Respondent5 explains that it is not something they only do during retrospectives. Team 69 scores very low on this indicator. Respondent4 and Respondent6 both state that the change of team members, 3 in the last year, might have had an influence. They are not giving feedback outside retrospectives and during the meeting it is not as critical and transparent as it should be.

Team members that share relevant information

This indicator again shows a difference between team Rubic and team 69. Team Rubic scores high here with 2/2, while 69 scores average with 1/2. The biggest difference is the best practice that Rubic has to have knowledge sessions, in which they actively share knowledge regarding certain areas of the application. Both respondents of team Rubic are very enthusiastic about that and explain that it helps them to reduce the single point of failures.

Leadership know the environment and senior management

The respondents answer very different here. Respondent1 thinks this is quite good and that senior management is approachable. Respondent2 states he could be more approachable, but that they found a way to make it work. While Respondent3 argues that the organization should be flatter in order to create that availability. Now he is often too occupied with the unit level discussions, which should not be there. It is clearly an indicator which causes partition. The indicator could be not specific enough. Another option is that the organization needs to talk about this indicator, because they do not seem to have a common answer. Perhaps this is something that the managerial layer of the organization should redefine.

Organization culture

It is very interesting that there are seven indicators that gather a full score. These are: benefit of working together is clear and communicated, decentralized day-to-day decisions, opportunities and limitations are known, environment for asking questions, team members complement each other, responsibilities and skills should be clear and a culture of asking questions. It is quite interesting to see that everybody seems to think the same about these topics. Even the respondents that did not answer all the questions about these indicators said nothing to contradict these high scores for the indicators. It seems like these indicators are truly part of the organizational culture.

For shared goal it is one indicator per sub-aspect. It seems that they thought hard about why they will start working differently and felt like that the teams should really know why. They also wanted the team to feel autonomy and make decisions for themselves. Both things show that they take the team very serious. They also gave the cadre to team, to ensure that they know what they can and cannot do. They score a bit lower on the managerial and content-oriented topics, like availability of management and known strategy and expectations.

The opposite is true for the team. They score high on the more content-oriented indicators. It is truly an environment for asking questions. Every respondent said: mainly content questions. They also think that team members complement each other and that skills and responsibilities are clear. Which is a tangible thing. Furthermore, they state that the culture of asking questions is really there. While the lowest scores, and therefore the most doubt, is in the more intangible indicators; self-aware team members, team members listen to each other and moments of reflection and evaluation.

Conclusion

The indicators are mostly corroborated by the data. The indicator; Leaders know the environment and senior management, created the most confusion. According to the data, the other indicators seem to be relevant and influence the sub-aspect.

4.2.2 Sub-aspects

The results of the data analysis regarding sub aspects are discussed in this section. All respondents answered questions regarding the sub-aspects. The most relevant information per indicator can be found in Appendix 4.2.1. Most of the gathered data is obtained by interview questions: 8, 11, 15, 22, 26 and 30. The information that was retrieved is summarized in Appendix 4.2.2 and 4.2.3.

Table 4.4 shows the scores of the sub-aspects. They are based on the answers of the respondents on questions 16 and 31 the three aspects of shared goal and shared understanding got a score of 1, 2 or 3. In some cases the respondents only chose the most important one and in that case the other two sub-aspects got the score: 2. A 1 is given to the most important sub-aspect, while a 3 is given to the least important one. The last two rows show the cumulative score and the average score. So, the sub-aspect with the lowest average score is the sub-aspect which respondents see as the most important. For the main-aspect, shared goal, the most important sub-aspect is IT leadership. For the main-aspect, shared understanding, the most important sub-aspect is Co-construction. IT Leadership has a score of 1.43 and Co-construction has a score of 1.29. For shared goal, the order is: IT

Leadership, Shared Vision and cadre and Organizational Structure. For shared understanding, the order is: Co-construction, Constructive conflict and Construction.

Table 4.4: Scores of sub-aspects – all respondents

	Shared Goal			Shared Understanding		
	IT Leadership	Organizational Structure	Shared Vision & cadre	Construction	Co-construction	Constructive Conflict
Respondent 1	2	3	1	2	2	1
Respondent 2	1	2	2	2	1	2
Respondent 3	1	2	3	2	2	1
Respondent 4	2	3	1	2	1	2
Respondent 5	1	3	2	2	1	3
Respondent 6	1	2	2	3	1	2
Respondent 7	2	3	1	2	1	3
	10	18	12	15	9	14
	1,43	2,57	1,71	2,14	1,29	2,00

Table 4.5 shows an interesting effect. It only portrays the scores of the sub-aspects when the respondents answered all the questions regarding its indicators. So, a division between managers and team members. The interesting thing is that the scores of IT Leadership and Co-construction, the two high scoring sub-aspects, are even lower. Two out of three managers think IT leadership is the most important sub-aspect and the only manager who thinks otherwise scores it as number 2. While 4/4 team members score co-construction as the most important sub-aspect.

Table 4.5: Scores of sub-aspects – respondents who also discussed all indicators of the respective sub-aspect

	Shared Goal			Shared Understanding		
	IT Leadership	Organizational Structure	Shared Vision	Construction	Co-construction	Constructive Conflict
Respondent 1	2	3	1			
Respondent 2	1	2	2			
Respondent 3	1	2	3			
Respondent 4				2	1	2
Respondent 5				2	1	3

Respondent 6				3	1	2
Respondent 7				2	1	3
	4	7	6	9	4	10
	1,33	2,33	2,00	2,25	1,00	2,50

The results differ when the other group is isolated. It is quite interesting to see that the team members see no difference between IT Leadership and Shared vision and cadre. At least, not in term of importance. They do all state that they are very pleased with the leadership and not that pleased with the vision. This is due to the lack of a clear and shared vision and cadre. However, quite a different outcome than the managers. Similarly, the managers have a different opinion than the team members. The team members score co-construction as the most important sub-aspect, but the managers disagree. They state that the indicator; constructive conflict is more important. The sub-aspect that team members scored the lowest. Two out of three managers state that constructive conflict is the most important sub-aspect for a shared understanding. The other manager thinks it is the second most important one. The data set is small, and the results are not significant. However, it would be interesting to focus more on that difference between team members and management.

Table 4.6: Scores of sub-aspects – respondents who did not discuss all indicators of the respective sub-aspect

	Shared Goal			Shared Understanding		
	IT Leadership	Organizational Structure	Shared Vision	Construction	Co-construction	Constructive Conflict
Respondent 1				2	2	1
Respondent 2				2	1	2
Respondent 3				2	2	1
Respondent 4	2	3	1			
Respondent 5	1	3	2			
Respondent 6	1	2	2			
Respondent 7	2	3	1			
	6	11	6	6	5	4
	1,5	2,75	1,5	2,00	1,67	1,33

4.3.2.1 Sub-aspects – Interesting remarks or relations

Appendix 4.2.1 portrays the relevant information per indicator and Appendix 4.5.2 compares the data against the theory. This section introduces the most relevant information, that is retrieved, regarding the sub-aspects.

Organizational Structure

The sub-aspect with the poorest score; 2,57/3, where a 1 is the best score. The result is quite interesting, because the indicators of organizational structure scored very high. So, it brings up the question what the impact of these indicators is if organizational structure is deemed as unimportant. Furthermore, you see a difference between the managers and the team members. The team members score organizational structure very low, with comments as; it did not really have an impact, it is supportive at best or it can't have been that important since it hasn't always been clear. The only team respondent that did think it was important was Respondent6. However, managers give another insight. Respondent1 scored it as the lowest sub-aspect, but did state that it is important and can be used to achieve a shared goal. Respondent2 and 3 score it with a 2. Respondent3 states that organizational structure and IT leadership are almost evenly important. He thinks that the structure was a big enabler, especially in the beginning of the transition to DevOps.

There are two other possible reasons for the low score. The first is based on a statement of Respondent3. He explained that the organization had a workshop in which they were trying to find the motivation of the employees. What does everyone need? The employees scored very low on process, stating that they had enough structure and that they knew what was expected from them. If that is the case, then the need for structure is lower. Which might cause it to score a lot lower. Another argument could be the sequential way of thinking that some people have. Respondent1 describes that he thinks you should first have a vision, then a leadership team that guides teams to that vision and then you need to start thinking about the structure of your organization.

IT Leadership

A very high score. Everybody, including team members, are quite satisfied with the leadership. Team members acknowledge the effect that leadership has on the autonomy of the team. Based on the literature, IT Leadership has a big impact on shared goal. The data suggests that it also has quite a big impact on several of the indicators and sub-aspects of shared understanding. They really set an example for team members and team members praise them for being able to set a culture, make a change and give them the required space to grow. This suggests that IT Leadership has a bigger impact than portrayed in the conceptual model. Respondent2 explains that it is important to show your vulnerabilities, which could impact shared understanding. If you want that in an organization, the leadership has to start with it.

Vision

This is an interesting sub-aspect. None of the respondents is very enthusiastic about the vision and the way it is shared. However, they do all seem to think it is an important sub-aspect. Everybody states that it is important to have a vision, that is aligned across all levels of the organization. Preferably, a vision which is broadly shared and can be used as guidance. Although the organization fails in creating such a situation, they do seem to crave for it.

Construction, Co-Construction and Constructive Conflict

There is only one respondent who did not say at first: 'well you need all three' and that was Respondent5. He said that constructive conflict was not that important for his team. All other respondents were quite clear about the importance of these sub-aspects and that the sub-aspects strengthen each other.

Co-construction versus constructive conflict

As described earlier it is quite interesting that the team members clearly chose co-construction, while the managers preferred constructive conflict. Especially the difference in how constructive conflict is perceived by the two groups is interesting. 1.33/3 versus 2.5/3. The argument that managers use for their perception is almost exclusively that it is important to remove the sharp edges and that you can do that by having a constructive conflict. They talk about showing passion and having a firm debate, especially about feelings. Team members, however, seem to think that it is more about opening up and communication. You need to know each other, what people can and cannot do and what you need as a team from every individual. Reflection is also important and that is, as also described in paragraph 4.1, clearly something that can be improved.

Conclusion

The sub-aspects are mostly corroborated by the data. The sub-aspect; organizational structure, was deemed the least important. Whereas IT Leadership is seen as very important. According to the data, the sub-aspects seem to have a positive influence on the main-aspects and are recognized by the respondents.

4.2.3 Main-aspects

Every respondent seems to think that this are two very important aspects for collaboration in a DevOps context. When they were asked what the most important main-aspect is, the results were ambiguous. Four out of seven respondents chose for shared goal and the other three chose shared understanding. Two managers and two team members stated the most important main-aspect is shared goal. So, the answers are spread widely. Appendix 4.3 presents more information regarding the main-aspects. The respondents were asked several questions, like question 2, 3, 5, 18 and follow-up questions, which gave an insight in their perspective towards the main-aspects and collaboration. Appendix 4.3.1 portrays a summary for those questions and Appendix 4.3.2 incorporates the data that was used to create that summary.

According to Respondent1 successful collaboration starts with a clear organizational purpose. If that is clear the team can create a shared goal and then they can move towards shared understanding. He describes it as a sequential process, wherein shared goal is the most important factor. Respondent4 agrees by suggesting that; 'you know what needs to be done when a goal is clear. If a goal changes everything will follow'. Respondent4 and Respondent6 state that shared understanding takes time, you can work on that as a team.

Respondent2 has a different notion; 'shared understanding has the biggest impact. If you know each other and the benefit of working together well, then the goal is secondary. Respondent2: 'shared understanding is harder to achieve and it's more fragile. You can always create a new goal'. Respondent5 adds that the goal has not always been clear for their team. That they created that

bottom-up and defined a long-term goal based on the clarity that came from shared understanding. Respondent7 refers to a statement of one of his colleagues; 'I could do a horrible job with my current colleagues'.

Respondent3 states that a shared goal and a shared understanding have to intertwine. To achieve that you need facilitation, leadership and ceremonies with a clear purpose. Respondent3; 'shared goal versus shared understanding is a management-team balance. For me, the shared goal is slightly more important, because it is the starting point. It has an immediate effect, while shared understanding is a process'.

There is another thing that stands out. Which, again, comes down to the difference between teams. Respondent4 and Respondent6 state that shared goal is more important. They are both in team 69. Respondent5 and Respondent7 state that shared understanding is more important. They are both in team Rubic.

4.3.3.1 Main-aspects – Interesting remarks or relations

Appendix 4.3.1 portrays the relevant information per main-aspect focused question. This section introduces the most relevant information, that is retrieved, regarding the main-aspects.

Team of the respondent(s)

As stated earlier there is a division between Respondent4 and Respondent6 on one side and Respondent5 and Respondent7 on the other side. This might have something to do with the team they work in.

Team Rubic is an experienced team, with 5 seniors and 1 mediator. They have been working together for a long time, there has only been one personnel change in two years. Team 69 is a more junior team, lately there have been three changes. Rubic is described as a goal-oriented team that focuses on speed, while 69 is a quality-oriented team, that focuses on process and quality. Respondent3 explains that the teams were formed based on the preferences of the employees.

Respondent2 and Respondent3 mention that Rubic is the most autonomous team of the organization. They are often free to decide themselves. It might not be strange that these team members state that shared understanding is more important. Respondent5 says it nicely; *Based on our shared understanding we created an image of the future and based on that we created a shared goal*".

Respondent6 states the following: *"For me, and I think also for our team, shared goal is more important. Without a goal you do not need shared understanding. What are you doing without a goal?"* Which might not be so weird for a team that it is partially new and has more junior members. Therefore, it might be possible that the maturity of the team has a big impact on which of the two main-aspects are more important for creating collaboration in a DevOps context. That would mean that both teams should be led in a different way. Which resembles to remarks from Respondent2 and Respondent3.

Collaboration & DevOps

The respondents are very positive about the collaboration within the teams. It is interesting to stress that teams seem to have different maturity levels. Respondent1 is talking about the positive influence of having a team and how hard it is to develop a team, which makes sense if you take into account that the team exists for only one year. Within the business unit core, there are two teams that were discussed during the interviews.

They had the dream to be able to release at any moment. This created a whole new way of thinking and created many questions, regarding: the server park, build servers, team autonomy, CICD, non-functionals, monitoring, provisioning, containerization et cetera. DevOps seems to demand a certain mindset. That was not present at the business unit; core, in 2016. Many of the respondents started to work for the department during that year or quickly after the start of the transition. All the respondents that work in the development teams state that the aforementioned dream had a big impact. It was an enormous stretch for the organization, but created the discussion on how to achieve it.

Respondent1 explains how they had to set up a new way of working. One wherein the customer plays an important role and feedback loops are implemented. Respondent2 sees the dream to release whenever it is necessary as an important driver for success. Teams needed to think up front about branching strategy, testing, prioritization of functionalities. It also was the starting point for a different mindset, one based on passion. 'Be proud of your work'. Respondent3 describes two important roles in that process. One he called the police, "for instance Respondent4", they will enforce the principles. The other group challenges the police, when the rules do not apply, "like Respondent5". There are several visible benefits since the transition, like customers satisfaction (respondent1, respondent2 and respondent5), employer happiness (respondent1, respondent3) and knowledge sharing (respondent2 and respondent7). By removing traditional barriers between roles, you create friction and discussion upfront, which is good according to Respondent2; 'they will start understanding each other'.

Most respondents see that the effect of DevOps is teams with more ownership. Respondent4; 'they feel more motivation to start doing something, to create. That starts with responsibility'. Another element that several respondents appreciate is the interpersonal relations outside the office. They became close and people tend to stay quite long. The team feels the responsibility to ensure that they deliver a good product, that will perform as required. Respondent5 adds that they; 'automated some of the pain away and introduced processes where it was absolutely necessary'. That gave them the possibility to attain more autonomy. Further information about the collaboration and the transition can be found in Appendix 4.3.2.

Shared goal of the teams

Team Rubic has a very clear and demarcated shared goal. They are responsible for replacing the old studio. Respondent2: 'it creates energy and ownership when you give a team that level of autonomy'. They decide, also regarding architectural principles. Respondent5 and Respondent7 are very pleased with the level of autonomy. Respondent5: 'since Q2 of last year our shared goal is very clear. We are, since then, building a new platform. Before that it is was quite scattered. We had no sense of ownership or influence. Now we feel the autonomy and it helps to stand for something and to really go for it'. Respondent7: we have a few cadres and within that we exercise our autonomy. We do use stakeholders for their knowledge and experience and ask for feedback when we have a concrete piece of the puzzle'.

Team 69 has focus on Runtime, there are several components that require attention and per quartile they focus on one of them. Respondent3 describes the year goal for them as; 'work on the Runtime and improve the overall performance and the process engine. According to Respondent6 that makes it harder to concretize the shared goal. He describes it as; 'deliver generic, robust building blocks that are multi-usable and deliver value for our customers and colleagues'. Respondent4 also sees a difference with Team Rubic. He focusses mainly on the department goal, which is: deliver a high-quality and well-functioning platform. He adds that the development mix is also an indicator of team goals. Roughly 50% of their work is focused on functionality, which means that other things like support, technical debt, non-functionals and innovation are also very important within the teams.

Shared understanding of the teams

According to Respondent1 the shared understanding of the Public DevOps team is quite high. If he would have the pinpoint one improvement point it would be naivety. The team members are too eager to help the customer, but they should try to find a shared understanding on how to handle a new request. Respondent2 likes the private contact. 'For me that's one of the benefits of working here'. He is corroborated by Respondent4 and Respondent5. Respondent2: 'It is not only Instagram life. That is an important element of our culture. Respect for each other problems is a cornerstone. If somebody would disrespect that, he or she would have a hard time here. Lately, one of our HR employees told me that she is always overwhelmed when she arrives at the office. Her mailbox is full, people at her desk et cetera. So, what can we do? Now she comes at the office at 10:30 and everybody knows do not call her between 09:00 and 10:30. Power to you. We try to create a situation where that is possible. That requires transparency and a culture of acceptance'.

A point that could be improved is the level of feedback. Especially Respondent3, Respondent4 and Respondent6 see that necessity. Team 69 has had some personnel changes, so it makes sense that they need to reinstate that practice. Most respondents like the transparency in the organization. Respondent5; 'if something is going on in your private life, we will discuss that. We are very transparent'. Furthermore, they know what everybody likes and what they want to achieve. Respondent7: 'some of the team members inspire others to grow'. He claims that sharing is a very important factor, in order to be able to replace each other. Respondent3 likes how the teams investigate whether they need things to keep their shared understanding on the required level. He recalls an example where one of the teams asked him for a junior, to attain a certain balance. 'Their own idea based on a need. Sometimes I feel like they are more equipped to look into the future than us'.

Other aspects mentioned by the respondents

There were two respondents who gave alternatives regarding the main-aspects. Respondent6 stated that inter-team collaboration is also an important factor. He thinks the teams could help each other more by sharing knowledge and ideas. He stated it as: 'more *we*, instead of *us* and *them*'. For this research it was out of scope, but it seems to be a valid and interesting point. Respondent3 also mentioned an alternative main-aspect. He stated that it is very important to show the results of the collaboration to stakeholders. The results are better: happier employees, more satisfied customers, better ideas, higher quality etc. He emphasizes it is very important to deliver this message to the stakeholders, so that teams can keep the autonomy they have nowadays. It is also a version of sharing and could add knowledge to the dimension of DevOps: collaboration.

Conclusion

The main-aspects are corroborated by the data. According to the data, they have a positive influence on collaboration in a DevOps context.

4.4 Research results – content analysis

In Appendix 4.4 two anonymized documents are attached. One focuses on the way employees would like to improve their organization. How they want to be. They do this by showing several questions and asking employees to come up with two situations: 1) the as is situation and 2) the to be situation. The other document is a management tool which shows the things that the management uses to measure and manage. There were not a lot of documents that fitted the requirements. Most of the data is captured in tooling and some of the requirements are not documented at all. Therefore the content analysis can substantiate some suggestions, but cannot substantiate the full semi-structured interview.

Document – As is versus To be

This document contains 8 questions, and most answers to the questions link in some way to the conceptual model. Table 4.7 depicts the questions and the perceived link to the conceptual model. They participated in the workshop to create more synergy within their DevOps organization.

Table 4.7: Document has a link to the conceptual model

Questions	Link to conceptual model?	Link
What is your shared image of the future?	Yes	Shared Vision, Decentralized Decisions
What are your biggest chances and possibilities?	Yes	Shared Vision, Responsibilities and skills should be clear
What interests do you serve?	Yes	Shared Goal
What would you describe as a nice way to collaborate?	Yes	Team members complement each other
What can you do easily?	No	
What would be the right steps for you?	Yes	Shared Vision
What is your story?	Yes	Shared Vision, Self-aware
What results in the most action in your organization?	Yes	Moments of reflection and evaluation

More information, regarding the questions, can be found in Appendix 4.4.1. One of the things that stands out is that in four out of eight questions a necessity for a (shared) vision is mentioned. This strengthens the statement that the shared vision is an important aspect of shared goal. Especially because the questions are focused on a common goal or something that they would like to have. In many of the cases they state that a shared vision is needed to move forward. The results from the interviews were similar. It is also interesting that they said something that resembles to the

indicator: moments of reflection and evaluation. An indicator that was scored very low, but they also mention here as a point for improvement.

Document – Management tool

The document can be found in Appendix 4.4.2. All the measurement and equipment in this document could be seen as guidelines for the team. It helps them to know whether they are doing well as a business unit and the teams are improving on every level. Furthermore, it is a way to create a clear cadre for the teams. This fits the information retrieved by the interviews. The management team is very good at giving the team a cadre, with known opportunities and limitations. They could improve on having a shared vision for the product and the organization. Also, in this document the emphasis is on what is a good product, how should we test or produce software and how can we measure and improve that.

Conclusion

The two documents were a relevant addition for the research. They show the need for more shared vision and the existence of a clear cadre. Furthermore, they do not falsify indicators or sub-aspects. The argument could be made they also strengthen some of the other indicators: like moments of reflection and evaluation.

4.5 Theory and Data

This section compares the theory against the gathered data. In Appendix 4.5 extra information is depicted. The theory-data comparison for the indicators (paragraph 4.5.1) and the sub-aspects (paragraph 4.5.2) can be found there. The comparison for the main-aspects will be presented in this paragraph.

Shared goal

As described earlier, shared goal is defined as an organizational instrument to create focus, by setting an objective or responsibility which is applicable for all team members (Detzten et al., 2018; Kozlowski & Bell, 2003; Alexander & Van Knippenberg, 2014). It enables teams to perform (Polat et al., 2018; Cetindamar et al., 2005; Verbeeten et al., 2018) and collaborate (Alexander & Van Knippenberg, 2014; Kwak & Anbari, 2009; Polat et al., 2018) more effectively, which is influenced by the introduced sub-aspects.

The respondents all see an effect of shared goal on collaboration. They all state that. It is described as something that could be a kickstart and creates guidance. The impact of organizational structure seems to be the lowest if we consider the data. It seems to have some merit, but the other two sub-aspects are deemed more relevant. Several respondents share that it helps when the vision is clear, and you know the cadre. Furthermore, it is a process that should be guided by good IT leadership. It is preferable if the team has the feeling that they had an impact in the goal and they really share it. They should feel responsible for the goal. The respondents of Team Rubic describe that they went through such a process and that they are really proud on their product and shared goal.

The theory and data are strongly related. The data suggest that the level of maturity and autonomy is a factor in how the shared goal is created. An experienced and mature team will get more autonomy and will more likely create the goal based on that level of freedom. While, a less experienced team will get more guidance top-down, via management. It is a matching principle between management and the teams. It depends on how much the team can decide for themselves how the shared goal is formulated.

Another interesting point was the difference in clarity regarding the shared goal. Team Rubic had a very clear goal, according to all respondents. For Team 69, it was harder to formulate what their goal is. Although it is hard to draw conclusions from that fact alone, they might collaborate better if their shared goal would be clarified.

Shared understanding

In this study a shared understanding is defined as a concept influenced by the team that creates clarity and alignment regarding the required activities, responsibilities and situation (Bittner & Leimeister; 2014; Mohammed & Dumville, 2001; Stout et al., 1999) with respect to an object of understanding. It enables a team to perform and collaborate more effectively (Aubé et al., 2015; Mohammed & Dumville, 2002; Stout et al., 1999; Kleinsmann & Valkenburg, 2008), which is influenced by the introduced sub-aspects.

The respondents all see an effect of shared understanding on collaboration. They all state that. The main-aspect is seen as a fragile process, which can empower teams in the long run. Several respondents share that it helps when they really understanding each other and know how to achieve something together. Almost all respondents state that the three sub-aspects have to interconnect; it is a process of knowing yourself, knowing others via effective communication and handling potential conflicts professionally. The relative value of enablers for growing shared understanding is a point for discussion. Management argues that the most important sub-aspect is constructive conflict, while team members state that it is co-construction.

Furthermore, it is process that should be guided by good leadership. That is something that is not mentioned in the conceptual model. However, management is very focused on guiding that process. This is also something that the team members notice and appreciate.

The theory and data are strongly related. The data suggest that the teams are very good in making sure they have the required skills and information to perform well. However, the more personal elements tend to get less attention. They do know it is important and have clearly given it some attention when required, still all the team members state they could do more regarding those indicators.

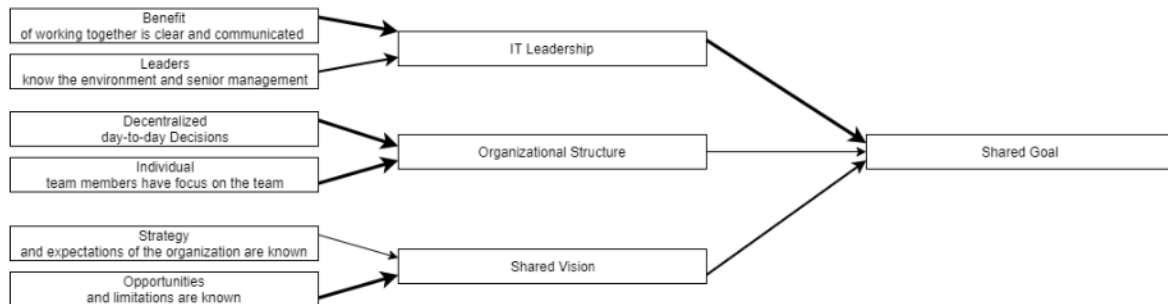
Another interesting revelation was that the team members are very aware of the fact that shared understanding is an important thing. They appreciate the culture of the department and their team culture and want to help each other on the personal and work level.

4.6 Sub research question 4

This section formulates an answer on sub research question 4; *how does a shared goal influence collaboration in a DevOps context?*

Most of the sub-aspects were deemed important for the main-aspect: shared goal. All respondents stated that IT Leadership and Shared vision are important sub-aspects. There was some debate regarding organizational structure. Managers deemed it important, while team members did not see it as an important sub-aspect. They spoke more about autonomy. Managers did that too, but saw organizational structure as a way to enable that.

Figure 4.2: Relations indicators of the sub-aspects and sub-aspects of shared goal



As depicted in figure 4.2, the shared goal segment of figure 4.1, there are different effects. Based on what the respondents state, the impact of IT Leadership is the highest. It might even be so high that it also influences shared understanding. Shared vision and cadre is also deemed very important, although the organization has not been able to fulfill the indicators. The respondents all think that vision could be clearer and better shared. However, they do see that they would benefit from it if that changed.

It is also interesting that the indicators of organizational structure score very high, while the impact of organizational structure on shared goal is deemed the lowest. All respondents said that shared goal has an influence on collaboration within a DevOps context. Four out of the seven respondent stated that it is the most important main-aspect. They all think that a goal unifies and that a team needs this to be effective. The results suggest that the maturity of a team decides whether there is more need for a shared goal or for a shared understanding.

A shared vision can give the team guidance. The respondents state that the inter-business unit discussions, regarding vision, sometimes prevent them from having a clear shared goal. The team is not always able to understand and prepare for the future, which makes it harder to make the right choices. Therefore, team members and management both state that more clarity could have a huge effect. The organizational structure is the structure that formalizes the organization in a way that the shared vision can be reached. Not all respondents (respondent4, respondent5 and respondent7) saw a huge effect here and some disclosed that the structure was not always clear. However, others (respondent2 and respondent3) state that the structure at the beginning of the transition defined the cadres for teams. Within these cadres the team could be autonomous. When they became more mature, the cadres became blurrier. Point for discussion is whether the new situation is still an organizational structure. The team members seem to interpret organizational structure with procedures and not with freedom.

According to the respondents leadership, should focus its energy on guarding the culture (respondent2 and respondent3), autonomy (all respondents), maturity (respondent2, respondent3 and respondent4), cadres (respondent1, respondent3 and respondent7), vision (all respondents), stakeholder management (respondent2, respondent3, respondent5 and respondent6) and work-life balance (respondent2 and respondent3). By creating the right environment for teams, they can flourish. That gives them the tools to attain a shared goal. Respondent2 described that they try to

address the pain, if others will not. Respondent3 adds that they focus on whether people are healthy. The leadership can be an example, by showing the right mentality, and has the opportunity to give teams what they need. Respondent3 calls that servant management. All respondents seem to agree that you will not meet the required vision if the leadership does not fulfill its role.

The sub-aspect IT Leadership is deemed important and well equipped within the organization. The sub-aspect Shared vision is deemed important and not well equipped within the organization. The content analysis verifies this statement. The sub-aspect Organizational structure is deemed less important and well equipped within the organization. Most respondents agree that these sub-aspects have an impact on the main-aspect: shared goal. An improvement on the sub-aspect; shared vision seems to be a fitting goal for this organization.

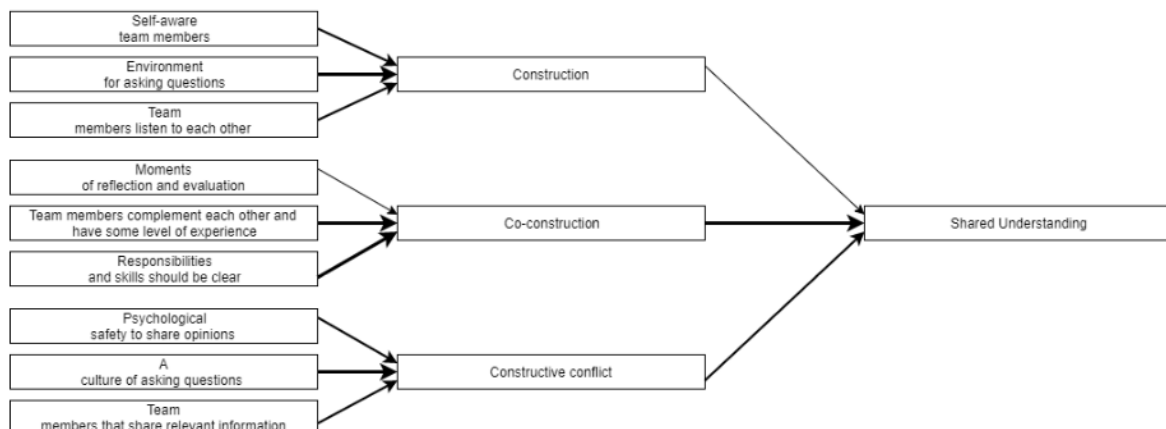
Based on the theory and the results of the interviews it can be stated that a shared goal has a positive impact on collaboration in a DevOps context. The data suggests that a shared goal has a quick effect, making it very effective during the first phases of team formation. Teams could need more guidance in that phase, especially if unexperienced. Due to the explorative nature of this research other researchers should strive to falsify that statement.

4.7 Sub research question 5

This section formulates an answer on sub research question 5; *how does a shared understanding of cross functional responsibilities influence collaboration in a DevOps context?*

All the sub-aspects were deemed important for the main-aspect: shared understanding. All respondents, but one, stated this quite clear. The team members thought that co-construction has the biggest impact, while managers think that constructive conflict is more important.

Figure 4.3: Relations indicators of the sub-aspects and sub-aspects of shared understanding



As depicted in figure 4.3, the shared understanding segment of figure 4.1, there are different effects. Based on what the respondents disclosed the impact of co-construction is the highest. Construction and constructive conflict are also deemed important. Managers even think that constructive conflict is the most important sub-aspect. Furthermore, the team members gave low scores on the indicators that focus less on content and more on relations. It seems like the organization could really improve there. Especially on the indicator: moments of reflection and evaluation, this is an indicator of co-construction. They gave this indicator the lowest score, while they stated that co-construction has the highest impact. During the interviews, the respondents showed that they think feedback, evaluation and reflection is very important.

All respondents said that shared understanding has an influence on collaboration within a DevOps context. Three out of the seven respondent stated that it is the most important main-aspect. They all think that a shared understanding is hard to achieve and that it is process that takes time. The results suggest that the maturity of a team decides whether there is more need for a shared goal or for a shared understanding.

Construction helps team members to understand their own way of thinking and understanding. The respondents stated that they have focused on this sub-aspect. The leadership has created session where team members thought about what they like and what gives them the energy to come to work. Team members also spoke about these things in length. They feel it is necessary to have a certain level of construction in order to be effective in co-construction. Which is the sub-aspect that team members all see as the most important indicator. They perceive it as transparency and explain that bad things can and will happen in life. They have a culture of sharing, so team members can help each other. The work-life balance is mentioned a lot. Furthermore, the indicator regarding feedback and reflection is important. Team members feel that they could improve and be more honest to each other. The constructive conflict is an interesting indicator. The management states that this is a way to enable growth. Things should clash occasionally. The team members allocate more worth to co-construction and construction. Some of the team members recognize that there are not a lot of conflicts. Respondent6 said that they are not having many conflicts and might need to think about whether everybody feel the required psychological safety. Interestingly, all respondents state that these three sub-aspects interconnect.

The sub-aspect co-construction is deemed important and well equipped within the organization, with the exception of the indicator: moments of reflection and evaluation. The sub-aspects construction and constructive conflict are deemed important too and there are several indicators with room for improvement. Almost every respondent states that these three sub-aspects have an impact on the main-aspect; shared understanding. An improvement of the indicator; moments of reflection and evaluation, of the sub-aspect; co-construction, seems to be a fitting goal for this organization.

Based on the theory and the results of the interviews it can be stated that a shared understanding has a positive impact on collaboration in a DevOps context. Creating a shared understanding takes time. The data suggests that the effect of shared understanding gets higher when team members trust each other and work together for a while. Due to the explorative nature of this research other researchers should strive to falsify that statement.

Chapter 5: Conclusion, discussion, recommendations and reflection

This chapter contains the conclusion of the research. The main research question is answered and discussed. Paragraph 5.1 will contain the conclusion. Secondly, paragraph 5.2 presents the discussion, which places the conclusion in the scientific context. Furthermore, paragraph 5.3 portrays the practical relevance for organizations. Especially the things that could be applied right away. The recommendations for future research are presented in paragraph 5.4. Lastly, paragraph 5.5 reflects on the research itself.

5.1 Conclusions

In this section the main research question is answered. This research has explored one of the dimensions of DevOps; collaboration, by investigating the influence of the main-aspects of collaboration in a DevOps context. The main research question of this research is: *What is the influence of a shared goal and a shared understanding of cross functional responsibilities on collaboration in the context of DevOps?*

Collaboration in a DevOps context is defined as cross-functional team members that execute activities in an attempt to fulfill a goal. Teams can have more focus when they have a shared goal, which is applicable for all team members. They are similarly influenced by shared understanding, which creates clarity and alignment.

Based on the theory and data there is enough ground to state that both main-aspects contribute positively to the collaboration dimension of DevOps. Figure 2.5 shows the conceptual model, which incorporates the answers of sub research questions 1-3. The last two sub research questions were 'how' questions, that needed investigation. The data indicates that the main-aspects have a positive influence on collaboration in a DevOps context. Shared goal seems to have a quicker effect, making it effective in the first phases of the team formation. Creating a shared understanding takes time, therefore the effect of this main-aspect, normally, gets higher when team members trust each other and work longer together.

Organizations that want to improve their collaboration dimension in a DevOps context should strive to create teams with a shared goal and shared understanding.

5.2 Discussion

In this section the research is discussed. The focus will be on what the research revealed and which lessons can be drawn from the research.

Discussion of conclusion / Impact of conclusion

Teams in a DevOps context can improve their collaboration by having a shared goal and a shared understanding. Organizations should strive to facilitate that by utilizing the elements in the conceptual model.

If there is one thing that this interview showed it is that there are still many things unclear regarding collaboration in a DevOps context. However, there are a few suggestions in this research that are

relevant to pose. The assembled theory and acquired data indicate that teams need clarity. They can achieve that clarity by having a clear shared goal or by having a shared understanding. Its most efficient when a team can really exercise their ownership on a product. They will fully embrace their shared goal and exercise responsibility. Shared understanding sheds light on how the teams approaches the work and acts as a team. When these team values are clear the team can grow.

Both main-aspects have an impact. It depends on the context which is more important. For instance, if a team is young, non-autonomous, unexperienced and unaware of the goals of the organization, it might be more efficient to boost their shared goal. So, help them to achieve that. This can be done by utilizing the effect of IT leadership, organizational structure and shared vision and cadre. Perhaps the term, organizational structure, should be refined: organizational chaos, supportive structure or the balance between autonomy and maturity. The team members seem to have a bias towards organizational structure: an organization that focuses on this has a process mindset. In this research IT leadership and shared vision and cadre showed the highest impact within a DevOps organization. The suggestion was made by one respondent that the organizational structure might have had a higher impact for creating a shared goal in a "Waterfall" context. That environment requires processes and control, while DevOps requires autonomy and trust.

A team can also be senior, autonomous, experienced and aware of the goals of the organization. In that case it might have more impact to boost their shared understanding. They will probably have all the capabilities to refine the shared goal themselves. The need for guidance is less high, but the need for understanding each other might be even higher. With experience, comes divergence between individuals. The shared understanding can be improved by utilizing the effect of construction, co-construction and constructive conflict. In this research team members stated that co-construction has the highest effect, while managers deemed constructive conflict more important. Every respondent stated that these sub-aspects needs to interconnect, in order to maximize the results.

Interestingly, most respondents agreed that the IT Leadership has a prominent role in deciding the right approach towards individuals and teams. Therefore, the role of IT Leadership could have been different in the conceptual model. It also seems to impact shared understanding. Examples are that the leadership has created situation wherein team members were challenged to think about their goals and personality. The leadership also challenged people to start sharing, also the more personal things. According to the respondents, the impact of that change in mindset has been high.

Another interesting insight is that a shared goal can be defined quite fast. It might take some time to really share it together, but it easier to create than a shared understanding. That is described as a delicate process. It takes a long time to build the right amount of trust and communication in a team. Individuals need to learn how to disagree in an effective manner and why it is not a problem to disagree. Every team is unique, so this dynamical process will restart when teams are reorganized. The research suggests that it could be done faster if the organizational culture is strong.

In traditional organizational models, the individual is the entity that executes the tasks. This works differently in DevOps. Managers do not appoint tasks to individuals, but to teams. That shared responsibility could even be one of the reasons why a shared goal and shared understanding are so important to achieve effective collaboration. However, traditionally the problem was; individuals do not always talk with the colleagues they need to get the job done. It is quite interesting that several respondents saw the same problem between teams nowadays. Teams tend to focus on their own

responsibility and forget to share information or lessons learned. That indicates that the learning curve of the organization could be even steeper. There are several respondents who would like to improve that and it could be very interesting to think how that could be achieved.

Integration with and extension on the existing literature

DevOps is a relatively new research subject, where collaboration is significantly older. The theories of Lwakatare et al. (2016), Giudice & Condo (2017), Lwakatare et al. (2015) and Humble and Molesky (2011) creates the foundation for the DevOps context and its dimensions. This research has gathered theory regarding collaboration that could fit the new context of DevOps. Several older collaboration studies still hold merit in the DevOps context, like Jassawalla & Sashittal (1998) and Mohammed & Dumville (2001). The main aspects that influence collaboration in a DevOps context are the main contribution of this research.

This research has posed definitions for collaboration, shared goal and shared understanding in a DevOps context, based on the nature of DevOps and the available literature. Other researchers and organizations could use that to clarify these concepts. Furthermore, this research has added sub-aspects that influence the main-aspects and indicators that indicate whether a sub-aspect is present. That could help organizations to assess their own situation and take steps to improve their status quo.

The indicators of the sub-aspects add an operationalizable method for repeating this study in other contexts. The lessons learned, described in section 5.5. could help to further improve the conceptual model. Most respondents recognized the indicators and were able to describe their impact on the corresponding sub-aspect. The conceptual model and the indicators are a worthy addition to the existing pool of scientific knowledge.

However, this research has not significantly proven the relationships that are depicted in the conceptual model. It has established indications for them. There are several suggestions for future research, which are described in section 5.4. The explorative nature of this study establishes an ideal situation to guide future research.

5.3 Practice recommendations

This section introduces the practical recommendations for organizations. Organizations that are implementing or have implemented DevOps could benefit from analyzing the conceptual model. It would help to understand the concepts that are described there. The recommended question is: on which sub-aspects and main-aspects do we perform well (and why) and where should we improve (and how)? This research can help to envision how cross-functional teams collaborate in a DevOps context and what influences that. For many organizations it is still new and challenging to create teams where developers, testers and operational personnel coexist. As described in 1.2; motivation & relevance, this research explores whether it is worthwhile to invest in the main-aspects that influence collaboration in a DevOps context. Based on the assembled theory the impact of collaboration in a DevOps context is deemed relevant.

Furthermore, the acquired dataset shows that the main-aspects have a clear impact on collaboration. Organizations that doubt whether they should invest more on the collaboration side

should see this investigation as a reason to do so. Every context is unique, so it is advisable that organizations consider several studies and assess which elements are relevant for their organization. Sometimes a long-term investment delivers more quality and value. Several respondents spoke about ways to allocate time and resources. Teams did not focus all their work on new functionalities, but also on limiting the level of technical debt. Similarly, organizations should find that balance and help teams to find that balance.

Organizations should not underestimate the impact of IT leadership, a sub-aspect of shared goal. It became obvious that DevOps needs another way of leadership. Therefore, organizations that envision a DevOps mindset should think whether their current leadership can help to create that. The organization that was researched needed new personnel; new leadership and new leaders in the teams, to make the transition successful. The leadership team started with clear cadres and a dream, but had the long-term strategy to create more autonomy. That means that the team needs to take accountability and needs to be coached to do so. An organization that wants to make the transition or struggles during the transition should reflect on the balance between leadership and teams. The conceptual model can help to make that analysis. The first lesson should be that shared goal and shared understanding matter. So, think about how you can create that.

If you want to create more autonomous teams it might start with clarifying in which regard, they can exercise their autonomy. It helps to have a clear vision and cadre. If that is in place the organization will need to balance the level of autonomy with the maturity of the team. It was quite interesting that the more mature team stated that shared understanding was important, while the less mature stated that shared goal was more important. The mature team even created their own shared goal based on that shared understanding. Organizations should consider that statement and try to assess for each of their teams what is the right approach. Should they get more freedom or should we help the team with more guidance. Of course, it is about finding the right balance and not about making a choice between one of the two.

The team members were really focused on sharing knowledge. They felt that it was one of the enablers of their work. Especially the mature team had several elements in place to foster that. They are aware of the risks of single point of failures. The management team stated that the teams were quite good in feedback and reflection, although they still could improve in the level of honesty. Team members are more skeptical and state that they are not sharing their inner thoughts and could give more feedback on the spot. Furthermore, management and team members have different thoughts about the impact of a constructive conflict. It might be worthwhile to talk about both things in a meeting with the management and the team.

Lastly, it was quite apparent that it is important to balance your teams. Elements that should be considered are experience, work preference (development or operations), type of personality and skills. Several respondents stated the impact of that and the research suggests that organizations should take that into account.

5.4 Recommendations for further research

This section introduces the recommendations for future research. However, all of the above is established, there is a need for more research. Firstly, because this was an explorative research. It would be interesting to distillate the results of this research and test them in a quantitative research.

Furthermore, it could be interesting to perform a multiple case study. This could create substance, by attaining literal replication. So, there could be several ways to substantiate the results of this research.

One of the things that are addressed in section 5.2 is the relation between how mature a team is and which main-aspect has more impact. That could be intriguing to investigate. Based on the data in this research a shared goal has a bigger impact on a new and less mature team, while shared understanding has a bigger impact on a mature team. One of the respondents even stated that their level of shared understanding created the situation to really describe their shared goal. Building on that, autonomy seems to be a key word in a DevOps environment. It would be interesting to further analyze that. Which personal values fit a DevOps culture and how do you create the right climate for teams to operate. Many respondents mention the important role for leadership. They can guard the balance between maturity and autonomy. Researchers could still clarify a lot in this regard, which could help organizations to tailor their leadership towards the nature of the team.

The second thing that is discussed in 5.2 is the sub-aspect; constructive conflict. The two groups of respondents had a different idea regarding its impact. Management saw it as the most important sub-aspect of shared understanding, while team members strongly disagreed. It would be interesting to investigate why that difference exists. It could have something to do with the nature of the work. Leadership has to build teams, while teams have to execute the work. So, they are the ones who have to have the conflict. Perhaps, they do not feel comfortable when they have conflicts. However, it could also be the case that management overvalues the debate and should talk more about convergence. Clarifying that could help management, team coaches and teams in where they focus their energy.

The comparison between theory and data resulted in suggestions. It could be interesting to investigate them. Two respondents had recommendations that also influence collaboration in a DevOps context. One of the respondents spoke about the inter-team collaboration. It could be interesting to assess whether the structure of the conceptual model is applicable for that. The other respondent mentioned that it takes a lot of effort to create the required time and climate for collaboration in a DevOps context. It is not easy to convince the financiers of the benefit, because it is not about short team value. This can be hard to create, especially in the beginning or when things go wrong. It could be interesting to investigate how senior management and financiers can be persuaded. What drives them to invest? Which measurements can show the value that it brings? The respondent suggested things like; employee happiness, customer satisfaction, level of technical debt, level of autonomy et cetera. It could help leadership to create the required climate for teams.

It was interesting to see that the management was more relationship oriented, while the team members were more task focused. In this research the indicators were not labeled as relationship or task oriented, but if they were it would have been quite visible. The data suggests that management is more equipped to handle the relation-focused indicators that boost the sub-aspects, while it is vice versa for the teams. If we take the responsibilities into account it makes sense, but they should be able to influence all the indicators. In the end it is about the right balance. It could be interesting to investigate whether this imbalance exists. And if so, why that is the case and how can it be improved.

The research subject DevOps is still quite new. Therefore, there are many things to strengthen and deepen. Researchers should take that opportunity. The impact of DevOps will probably have a long-lasting impact on many organizations.

5.5 Reflection

This section reflects on the way the research was conducted and what could have been improved. Every research has its flaws and it is important to pinpoint them. Other researchers, who will investigate a similar topic, can take this into account.

Conducting a semi-structured interview requires preparation, which enables to have the control during the interviews and coding. There are two things that could have helped in attaining that control. The first factor zooms in on the limited timeframe, the habit of respondents to digress and the need for comparable data. The interviewer should take the lead and needs to intervene when required. Knowing that upfront is not the same as knowing how to bring that into practice, so there was a learning curve. The most important lessons where; deep dive by asking open questions if something is relevant, interrupt the respondent and further explain when required, give extra guidance per subject and then ask the questions, listen and summarize, be as objective as possible and don't share your experience. These things are easy to think of, but quite hard to put effectively into practice. The main lesson is that it helps to practice and to explain these things upfront to the interviewee. Explaining it helps him or her to understand why you do certain things and keeps the respondent in an open mind. That is required, because you want the respondent to share.

The second factor is about the coding process. There should be a loosely coupled relation with the coding principles, the conceptual model and the structure of the semi-structured interview. This will help to map the gathered data according the conceptual model. Again, a lot can be learned during the execution. My advice to future researchers would be to have a trial run when the interview protocol and the coding principles are clear. For instance, with the informant or maybe with someone that does not even work at the research subject. Furthermore, the researcher should try to code this trial run. There would be three benefits for doing this; 1) the researcher can learn how the semi-structured interview is perceived by a respondent and whether the required set of data is retrieved, 2) the researcher can test the coding principles and has a change to readjust the balance between the conceptual model, the coding principles and the interview protocol and 3) the researcher has information to better predict the course of future interviews, the change to alter question accordingly and is able to have more meaningful interviews. When the test run is complete, it could be reviewed by a scientific partner. That person should be unaware of the content and aware of general scientific principles. This will also reduce the researchers bias even more.

Another element that is hard to control is the bias. This can come in many forms. One that is interesting to mention is the bias of a respondent. Language is a tricky thing and someone can get in a certain state of mind by using a term in a certain context. Therefore, it is important to use objective words in the questions. However, even then, respondents will respond based on their own interpretations and experiences. That could create a situation where a word or a sentence triggers the respondent into a direction that is not fitting for the research. For instance, two respondents often spoke of a sequential process. They think in processes. First, we need A, then we need B. When they were asked to share what they found most important, they started with A. Their sequential way of thinking let them to believe that they had to create an order, a sequence. While

the question was more focused on the weight or the impact. It is hard to correct the respondent's bias, without showing your researchers bias. There are two solutions; share clear definitions and use semi-prepared open questions to further analyze the answer. For instance, the designation and definition of the indicator; leaders know the environment and senior management, could have been clearer. It resulted in more shattered answers, because every respondent could put his own bias in the question.

Lastly, it would have been preferable to have more respondents. This would have created a stronger foundation. The most interesting addition would be to have respondents from the team of respondent1, the business unit; public. That would have made the research stronger. Due to time restrictions and availability that has not been put into practice. However, it is important to state it and to be aware of the fact that it would have increased the validity.

References

A list of all references used, in accordance with the APA format.

ABDC. (2019, April 8). *Master Journal List*. Retrieved from: <http://iimbg.ac.in/sites/default/files/ABDC-Journal-pdf.pdf>

Akbar, H., Baruch, Y. & Tzokas, N. (2018). *Feedback Loops as Dynamic Processes of Organizational Knowledge Creation in the Context of the Innovations' Front-end*. British Academy of Management. Vol. 29-3. pp. 445-463.

Akkerman, S., Van den Bossche, P., Admiraal, W., Gijssels, W., Segers, M. Simons, R.-J. & Kirschner, P. (2007). *Reconsidering group cognition: From conceptual confusion to a boundary area between cognitive and socio-cultural perspectives?* Educational Research Review. Vol. 2-1. pp. 39-63.

Alexander, L., & Van Knippenberg, D. (2014). *Teams in Pursuit of Radical Innovation: A Goal Orientation Perspective*. Academy of Management Review. Vol. 39-4. pp. 423–438.

Anthony, E.L., Green, S.G. & McComb, S.A. (2013). *Crossing functions above the cross-functional project team: the value of lateral coordination among functional department heads*. Journal of Engineering and Technology Management. Vol. 31. pp. 141-158.

Aronson, Z. H., Shenhar, A. J., & Patanakul, P. (2013). *Managing the Intangible Aspects of a Project: The Affect of Vision, Artifacts, and Leader Values on Project Spirit and Success in Technology-Driven Projects*. Project Management Journal. Vol. 44-1. pp. 35–58. <https://doi.org/10.1002/pmj.21322>

Aubé, C., Rousseau, V., & Tremblay, S. (2015). *Perceived shared understanding in teams: The motivational effect of being “on the same page.”* British Journal of Psychology. Vol. 106-3. pp. 468–486.

Balkema, A., & Molleman, E. (1999). *Barriers to the development of self-organizing teams*. Journal of Managerial Psychology. Vol. 14-2. pp. 134-150

Bang, S., Chung, S., Choh, Y. & Dupuis, M. (2013). *A Grounded Theory Analysis of Modern Web Applications: Knowledge, Skills, and Abilities for DevOps*. In: 2nd Annual Conference on Research in Information Technology. ACM, New York. pp. 61–62.

Barile, S., Saviano M., Simone C. (2015). *Service economy, knowledge, and the need for T-shaped innovators*. World Wide Web. Vol. 18- 4. pp. 1177-1197.

Basadur, M. & Gelade, G. (2006). *The Role of Knowledge Management in the Innovation Process*. Creativity and Innovation Management. Vol. 15. pp. 45-62.

Bass, L., Weber, I. & Zhu, L. (2015). *DevOps: A Software Architect's Perspective*. Addison-Wesley Professional.

Berggren, P., Johansson, B.J.E. & Baroutsi, N. (2017). *Assessing the quality of Shared Priorities in teams using content analysis in a microworld experiment*. Theoretical Issues in Ergonomics Science. Vol. 18-2. pp. 128-146.

Bittner, E.A.C. & Leimeister, J.M. (2014). *Creating Shared Understanding in Heterogeneous Work Groups: Why It Matters and How to Achieve It*. Journal of Management Information Systems. Vol. 31-1. pp. 111–144.

- Bygballe, L.E., Swärd, A.R. & Vaagaasar, A.L. (2016). *Coordinating in construction projects and the emergence of synchronized readiness*. Int. J. Proj. Manag. Vol. 34-8. pp. 1479–1492.
- Cetindamar, D., Çatay, B.O. & Basmaci, S. (2005). *Competition through collaboration: insights from an initiative in the Turkish textile supply chain*. Supply Chain Management: An International Journal. Vol. 10-4. pp. 238-240.
- Cha, J., Kim, Y., Lee, J., & Bachrach, D., G. (2015). Transformational leadership and inter-team collaboration. Group & Organization Management. Vol. 40-6. pp. 715-743.
- Charoensuk, S., Wongsurawat, W. & Khang, D.B. (2014). *Business-IT Alignment: A practical research approach*. Journal of High Technology Management Research. Vol. 25-2. pp. 132-147.
- Claps, G.G., Svensson, B.R., Aurum, A. (2015). *On the Journey to Continuous Deployment: Technical and Social Challenges Along the Way*. Information and Software Technology. Vol. 57. pp. 21–31.
- Davis, J. & Daniels, K. (2015). *Effective DevOps Building a Culture of Collaboration, Affinity, and Tooling at Scale*. Published by O'Reilly Media, Inc. , 1005.
- Debois, P. (2011). *Devops: A software revolution in the making? The Journal of Information Technology Management*. Vol. 24-8. pp. 3–5.
- Detzen, N., Verbeeten, F. H.M., Gamm, N. & Möller, K. (2018). *Formal controls and team adaptability in new product development projects*, Management Decision, Vol. 56 Issue: 7, pp.1541-1558, <https://doi.org/10.1108/MD-07-2017-0692>
- Dyck, A., Penners R., & Lichter, H. (2015). *Towards definitions for release engineering and DevOps*, Proceedings of the Third International Workshop on Release Engineering, May 16-24, Florence, Italy.
- Eldor, L. (2019). *How Collective Engagement Creates Competitive Advantage for Organizations: A Business-Level Model of Shared Vision, Competitive Intensity, and Service Performance*. Journal of Management Studies.
- Elo, S. & Kyngäs, H. (2007). *The qualitative content analysis process*. J Adv Nurs. Vol. 62. pp. 107–115.
- Erich, Amrit, C., & Daneva, M. (2017). *A qualitative study of DevOps usage in practice*. Journal of Software: Evolution and Process. Vol. 29-6.
- Flyvbjerg, B. (2006). *Five misunderstandings about case-study research*. Qualitative Inquiry. Vol. 12 pp. 219–245.
- Folke, C., Carpenter, S.R., Walker, B.H., Scheffer, M., Chapin III, F.S. & Rockström, J. (2010). *Resilience thinking: Integrating resilience, adaptability and transformability*. Ecology and Society. Vol. 15-20.
- Fowler, M., Highsmith, J., (2001). *The agile manifesto*. Softw. Dev. Vol 9-8. pp. 28–35.
- Gandomani, T. J., Zulzalil, H., Ghani, A. A. A. & Sultan, A. B. M. (2013). *A systematic literature review on relationship between agile methods and open source software development methodology*. CoRR.
- Ghezzi, C. (2017). *Of software and change*. J. Softw. Evol. Process. Vol. 29. pp. 1–14.
- Ghobadi, S. & Mathiassen, L., (2016). *Perceived barriers to effective knowledge sharing in agile software teams*. Inf. Syst. J. Vol. 26-2. pp. 95–125.

- Gupta, V., Kapur, P.K., & Kumar D. (2017). *Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling*. Information and Software Technology.
- Gustafsson, J. (2017). *Single Case Studies vs. Multiple Case Studies: A Comparative Study*. Retrieved from: <http://www.diva-portal.org/smash/get/diva2:1064378/FULLTEXT01.pdf>.
- Gutiérrez, L.J.G., Lloréns-Mones, F.J. & Bustinza Sánchez, O.F. (2009). *Six sigma: from a goal-theoretic perspective to shared-vision development*. International Journal of Operations & Production Management. Vol. 29-2. pp.151-169.
- Hackman, J. R., & Wageman, R. (2005). *A theory of team coaching*. Academy of Management Review. Vol. 30-2. pp. 269-287.
- Hoda, R., & Murugesan L.K. (2016). *Multi-level agile project management challenges*. Journal of Systems and Software. Vol. 117, pp. 245-257.
- Hoda, R., Noble, J. & Marshall, S. (2004). *Organizing Self-Organizing Teams*. Artificial Life. Vol. 10-4, pp. 379-395.
- Hoda, R., Noble, J. & Marshall, S. (2010). *Organizing self-organizing teams*. 32nd ACM/IEEE International Conference on Software Engineering Proceedings, ACM. Vol. 1, pp. 285-294.
- Hu, J. & Liden R.C. (2011). *Antecedents of Team Potency and Team Effectiveness: An Examination of Goal and Process Clarity and Servant Leadership*. Journal of Applied Psychology. Vol. 96-4. pp. 851–862.
- Humble, J., & Molesky, J. (2011). *Why enterprises must adopt devops to enable continuous delivery*. Cutter IT Journal. Vol. 24-8. pp. 6.
- Iden, J., & Bygstad, B. (2018). *The social interaction of developers and IT operations staff in software development projects*. International Journal of Project Management. Vol. 36. pp. 485–497.
- Iden, J., Tessem, B., & Paivarinta, T. (2012). *IS development/IT operations alignment in system development projects: a multi-method research*. International Journal of Business Information Systems. Vol. 11-3. pp. 343-359.
- Inayat, I., Salwah, S., Marczak, S., Daneva, M., & Shamshirband, S. (2015). *Computers in human behavior a systematic literature review on agile requirements engineering practices and challenges*. Computers in Human Behavior. Vol. 51. pp. 915–929.
- Jassawalla, A.R. & Sashittal H.C. (1998). *An examination of Collaboration in High-Technology New Product Development Processes*. The Journal of Product Innovation Management. Vol 15-3. pp. 237-254
- Joshi, K.D., Chi, L., Datta, A. & Han, S. (2010). *Changing the Competitive Landscape: Continuous Innovation Through IT-Enabled Knowledge Capabilities*. Information Systems Research. Vol. 21-3. pp. 472–495.
- Kleinsmann, M. & Valkenburg, R. (2008). *Barriers and enablers for creating shared understanding in co-design projects*. Design Studies. Vol. 29-4. pp. 369-386.
- Kolfschoten, G., Briggs, R.O. & De Vreede, G.J. (2009). *A diagnostic to identify and resolve different sources of disagreement in collaborative requirements engineering*. In K.D. Kilgour and Q. Wang

- (eds.), *International Conference on Group Decision and Negotiation*. Toronto: Wilfried Laurier University. pp. 1–12.
- Kozlowski, S.W.J. & Bell, B.S. (2003). *Work groups and teams in organizations*. Handbook of psychology. Vol. 12. pp. 333-375
- Krancher, O., Luther, P. & Jost, M. (2018). *Key Affordances of Platform-as-a-Service: Self-Organization and Continuous Feedback*. Journal of Management Information Systems. Vol. 35-3, pp. 776–812.
- Kwak, Y.H. & Anbari, F.T. (2009). *Analyzing project management research: perspectives from top management journals*. Int. J. Proj. Manag. Vol. 27-5. pp. 435–446.
- Lee, G. & Xia, W. (2010). *Toward agile: an integrated analysis of quantitative and qualitative field data*. MIS Q. Vol. 34-1. pp. 87–114.
- LePine, J. A. (2005). *Adaptation of teams in response to unforeseen change: Effects of goal difficulty and team composition in terms of cognitive ability and goal orientation*. Journal of Applied Psychology, 90, 1153–1167.
- Liu, Y., Li, C. & Liu, W. (2014). *Integrated Solution for Timely Delivery of Customer Change Requests: A Case Study of Using DevOps Approach*. International Journal of U- & E-Service, Science & Technology Vol. 7. pp. 41–50.
- Lo Giudice, D. & Condo, C. (2017). *Master DevOps For Faster Delivery Of Software Innovation*. Forrester research (nasdaq: Forr).
- Lwakatare, L. E., Kuvaja, P., & Oivo, M. (2015). *Dimensions of devops*. Paper presented at the International Conference on Agile Software Development.
- Lwakatare, L. E., Kuvaja, P., & Oivo, M. (2016). *An Exploratory Study of DevOps Extending the Dimensions of DevOps with Practices*. ICSEA 2016, 104.
- Lynn G.S. & Akgün, A.E. (2003). *Project visioning: Its components and impact on new product success*. Journal of Product Innovation Management. Vol. 18-6. pp. 374-387.
- Moe, N.B., Dingsøyr, T., Dyba, T., 2008, March. *Understanding self-organizing teams in agile software development*. In: 19th Australian Conference on Software Engineering, 2008. ASWEC 2008. IEEE. pp. 76–85.
- Moe, N.B., Dingsøyr, T. & Dybå, T (2010). *A teamwork model for understanding an agile team: A case study of a Scrum project*. Information and Software Technology. Vol 52. pp. 480–491.
- Mohammed, S. & Dumville, B.C. (2001). *Team mental models in a team knowledge framework: Expanding theory and measurement across disciplinary boundaries*. Journal of Organizational Behavior. Vol. 22-2. pp. 89–106.
- Mok, K.Y., Shen G.Q. & Yang, J. (2015). *Stakeholder management studies in mega construction projects: a review and future directions*. Int. J. Proj. Manag. Vol. 33-2. pp. 446–457.
- Morgan, G. (1986). *Images of Organization*. Sage Publications, Beverly Hills.
- Morgan, T.R., Richey Jr, R.G. & Autry, C.W. (2016). *Developing a reverse logistics competency: The influence of collaboration and information technology*. International Journal of Physical Distribution & Logistics Management. Vol. 46-3. pp. 293-315.

- Mulder, I., Swaak, J. & Kessels, J. (2002). *Assessing group learning and shared understanding in technology-mediated interaction*. Journal of Educational Technology & Society. Vol. 5-1. pp. 35-47.
- Ohland, M. W., Loughry, M. L., Woehr, D. J., Bullard, L. G., Felder, R. M., Finelli, C. J., ... & Schmucker, D. G. (2012). *The comprehensive assessment of team member effectiveness: Development of a behaviorally anchored rating scale for self and peer evaluation*. Academy of Management Learning & Education. Vol. 11-4. pp. 609-630.
- Patten, Karen; Whitworth, Brian; Fjermestad, Jerry; and Mahindra, Edward, "Leading IT Flexibility: Anticipation, Agility and Adaptability" (2005). AMCIS 2005 Proceedings. 361.
<https://aisel.aisnet.org/amcis2005/361>
- Penners, R. & Dyck, A. (2015). *Release Engineering vs. DevOps - An Approach to Define Both Terms*. Full-scale Software Engineering, 2015.
- Polat, V., Lynn, G., Akgün, A., & Emre, O. (2018). *Formal and Informal Communication in New Product Development Teams: The Mediation Effect of Team Trust*. International Journal of Innovation. Vol. 6-2. pp. 95- 109.
- Powers, C.L., Morgeson, F.P. & Lyons, B.L. (2014). *Dynamic Shared Leadership Theory: Understanding the Structures and Processes of Shared Leadership*. Academy of Management Proceedings. Vol 1.
- Prentice, C.R. & Brudney, J.L. (2016), *Definitions do make a difference: county managers and their conceptions of collaboration*. Human Service Organizations: Management, Leadership & Governance, Vol. 40-3. pp. 193-207.
- Roche, J. (2013). *Adopting DevOps Practices in Quality Assurance*. Communications of the ACM 56. pp. 38–43.
- Rong, G., Zhang, H., & Shao, D. (2016). *CMMI guided process improvement for DevOps projects: an exploratory case study*. Proceedings of the International Workshop on Software and Systems Process, Austin. pp. 76-85.
- Saunders, M. N. K., Lewis, P. & Thornhill, A. (2016). *Research methods for business students* (Seventh edition. ed.). New York: Pearson Education.
- Scimagojr (2019, April 9). *Journal Impact*. Retrieved from:
<https://www.scimagojr.com/journalrank.php>
- Smart, P. R., Mott, D., Sycara, K., Braines, D., Strub, M. & Shadbolt, N. R. (2009). *Shared Understanding within Military Coalitions: A Definition and Review of Research Challenges*. Knowledge Systems for Coalition Operations. Conference paper: United Kingdom.
- Soosay, C.A., Hyland, P.W. & Ferrer, M. (2008). *Supply chain collaboration: capabilities for continuous innovation*. Supply Chain Management: An International Journal. Vol. 13. pp. 160–169.
- Stipp, D.M., Pimenta, M.L, Jugend, D. (2018). *Innovation and crossfunctional teams: Analysis of innovative initiatives in a Brazilian public organization*. Team Performance Management: An International Journal. Vol. 24-2. pp. 84-105.
- Stock, R.M., Totzauer, F. & Zacharias, N.A. (2013). *A Closer Look at Cross-functional R&D Cooperation for Innovativeness: Innovation-oriented Leadership and Human Resource Practices as Driving Forces*. Journal of Product Innovation Management. Vol. 31-5. pp. 924-938.

- Stout, R.J., Cannon-Bowers, J.A., Salas E. & Milanovich, D.M. (1999). *Planning, shared mental models, and coordinated performance: An empirical link is established*. Human Factors Vol. 41. pp. 61-71
- Straus, A. & Corbin, J. (1998). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. California: Sage Publications.
- Suddaby, R. (2006). *What grounded theory is not*. Academy of Management Journal. Vol. 49: pp. 643–642.
- Takeuchi, H., Nonaka, I., (1986). *The new product development game*. Harvard Bus. Rev. Vol. 64-1. pp. 137–146.
- Tessem, B., & Iden, J. (2008). *Cooperation between developers and operations in software engineering projects*. Paper presented at the Proceedings of the 2008 international workshop on Cooperative and human aspects of software engineering.
- Thamhain, H.J. (2009). *Leadership Lessons From Managing Technology-Intensive Teams*. International Journal of Innovation and Technology Management. Vol. 06-2. pp. 117-133.
- Thongpapanl, N.T, De Clercq, D. & Dimov, D. (2012). *An investigation of the performance consequences of alignment and adaptability: contingency effects of decision autonomy and shared responsibility*. R&D Management. Vol 42-1. pp. 14-30.
- Tsai, K.-H., & Hsu, T. T. (2014). *Cross-Functional collaboration, competitive intensity, knowledge integration mechanisms, and new product performance: A mediated moderation model*. Industrial Marketing Management. Vol. 43-2. pp. 293–303.
- Tu, Y.-M., Wang, W.-T & Tseng, Y.-T (2009). *Transformation in a Self-organizing Team*. System Dynamics Review. Vol. 25-2. pp. 135-159.
- Van de Ven, A., Polley, D.E., Garud, R. & Venkataraman, S. (2008). *The Innovation Journey*. New York, NY: Oxford University Press.
- Van den Bossche, P., Gijssels, W., Segers, M., Woltjer, G. & Kirschner, P. (2011). *Team learning: Building shared mental models*. Instructional Science. Vol. 39-3. pp. 283–301.
- Vangen, S, Huxham, C. (2012). *The Tangled Web: Unraveling the Principle of Common Goals in Collaborations*. Journal of Public Administration Research and Theory. Vol. 22-4. pp. 731–760.
- Vannoni, M. (2014). *What are case studies good for? Nesting comparative case study research into the lakatosian research program*. Cross Cultural Research. Vol. 49-4. pp. 331-357.
- Walker, B.H., Holling, C.S., Carpenter, S.R. & Kinzig, A. (2004). *Resilience, adaptability and transformability in social–ecological systems*. Ecology and Society 9(2):5.
- Wettinger, J., Breitenbücher, U. & Leymann, F. (2014). *DevOpSlang – Bridging the Gap between Development and Operations*. In: Villari, M., Zimmermann, W., Lau, K.-K. (eds.) ESOCC 2014. LNCS, vol. 8745, pp. 108–122.
- Wolfswinkel, J. F., Furtmueller, E. & Wilderom, C. P. M. (2013). *Using grounded theory as a method for rigorously reviewing literature*. European Journal of Information Systems, 22(1), pp. 45-55.
- Yin, R. K. (2003). *Case study research: Design and methods*. Thousand Oaks, CA: Sage.
- Yin, R. (2014). *Case study research: Design and methods*. In: Sage publications.

Zábojník, J. (2002). *Centralized and Decentralized Decision Making in Organizations*. Journal of Labor Economics. Vol. 20-1. pp. 1-22

Appendix

This chapter contains several sections with additional information. This background information strengthens the statement that have been made in the other chapters. Appendix 1 presents the information related to the literature study. Appendix 2 incorporates the background information regarding the method. Appendix 3 portrays the details regarding coding. Appendix 4 combines the relevant information regarding the semi-structured interviews and content analysis.

Appendix 1: Literature study

This appendix presents background information regarding the literature study. Section 1.1 presents how the criteria per search engine were enforced. Section 1.2 and 1.3 presents the search queries for the two steps that were conducted regarding the key words. Section 1.4 portrays the valuable sources. Section 1.5 presents the selected sources based on the building blocks method, while section 1.6 portrays the selected sources per search engine. Section 1.7 incorporates extra information regarding ABDC & Scimago. Section 1.8 presents the building blocks per research question and section 1.9 incorporates more information regarding the snowballing method. Finally, section 1.10 presents the selected literature.

1.1 Criteria enforcement – Per search engine

This section, by presenting Table 1.1.app, shows how the criteria were enforced per search engine. The second column shows which criteria is described and column three to six explain per search engine how the criteria was enforced. Most criteria are selectable filters, but in some cases a manual action was required.

Table 1.1.app criteria enforcement per search engine

ID	Criteria for the references	Academic Search Elite (EBSCO Host)	Business Source Premier (EBSCO Host)	JSTOR	Web of Science
1	<u>Limited to:</u> Peer-reviewed	Adjusted with a filter	Adjusted with a filter	This was verified manually (in this study it was enough to verify this criterion in one of the other search engines)	This was verified manually (in this study it was enough to verify this criterion in one of the other search engines)
2	<u>Accessible:</u> Source fully available online	Adjusted with a filter	Adjusted with a filter	Adjusted with a filter	Adjusted with a filter
3	<u>Content type:</u> Academic journal, (e)book or conference paper	Adjusted with a filter (not possible for conference paper)	Adjusted with a filter (not possible for conference paper)	Adjusted with a filter (not possible for conference paper)	Adjusted with a filter
4	<u>Research area:</u> business, computer	Semi-Adjustable (used the "(Thesaurus) research subjects"-	Semi-Adjustable (used the "(Thesaurus) research subjects"-	Adjusted with a filter	Adjusted with a filter (used the research areas)

	<i>science or management</i>	<i>filter, which are narrower)</i>	<i>filter, which are narrower)</i>		
5	<i><u>Date of publication:</u> 2001 - present</i>	<i>Adjusted with a filter</i>	<i>Adjusted with a filter</i>	<i>Adjusted with a filter</i>	<i>Adjusted with a filter (starting from 2009)</i>
6	<i><u>Language:</u> English</i>	<i>Adjusted with a filter</i>	<i>Adjusted with a filter</i>	<i>Adjusted with a filter</i>	<i>Adjusted with a filter</i>

1.2 Search queries – First set

This section of appendix 1 presents the first set of search queries that has been applied during the first iteration of the building blocks method:

Academic Search Elite:

Table 1.2.app Search queries – step 1 – Academic Search Elite

SRQ	Academic Search Elite (EBSCO Host)						
	Criteria	Peer-reviewed	Full Text		Publication type:	Thesaurus research subjects	Publication Date
		Checkbox: Y	Checkbox: Y		Academic journal, (e)book	technological innovations, business enterprises, management, computer software, leadership, computer software development, information technology	2001 - 2020
1	Search Queries	<u>Search Word 1</u>	<u>Selected field (search scope)</u>	<u>Operator(s)</u>	<u>Search Word 2</u>	<u>Selected field (search scope)</u>	<u>Search Word 3</u>
	Query 1.1	DevOps	Abstract	AND	Collaboration	All Text	
	Query 1.2	DevOps	All Text	AND	Collaboration	Abstract	
	Query 1.3	DevOps	Abstract	OR	Collaboration	Abstract	
	Query 1.4	DevOps	All Text	AND	Collaboration	All Text	
2	Query 2.1	Shared Goal	Abstract	AND	DevOps	All Text	
	Query 2.2	Shared Goal	All Text	AND	DevOps	Abstract	
	Query 2.3	Shared Goal	Abstract	OR	DevOps	Abstract	
	Query 2.4	Shared Goal	All Text	AND	DevOps	All Text	
3	Query 3.1	Shared Understanding	Abstract	AND	DevOps	All Text	
	Query 3.2	Shared Understanding	All Text	AND	DevOps	Abstract	
	Query 3.3	Shared Understanding	Abstract	OR	DevOps	Abstract	
	Query 3.4	Shared Understanding	All Text	AND	DevOps	All Text	

	Query 3.5	Shared Understanding	Abstract	AND + OR	DevOps	Abstract	Cross-functional
	Query 3.6	Shared Understanding	All Text	OR + AND	DevOps	All Text	Cross-functional
	Query 3.7	Shared Understanding	Abstract	AND	Cross-functional	All Text	
	Query 3.8	Shared Understanding	All Text	AND	Cross-functional	Abstract	
	Query 3.9	Shared Understanding	Abstract	OR	Cross-functional	Abstract	
	Query 3.10	Shared Understanding	All Text	AND	Cross-functional	All Text	
4	Query 4.1	DevOps	Abstract	AND + AND	Collaboration	All Text	Shared Goal
	Query 4.2	DevOps	All Text	AND + AND	Collaboration	All Text	Shared Goal
	Query 4.3	DevOps	All Text	OR + AND	Collaboration	Abstract	Shared Goal
	Query 4.4	DevOps	All Text	AND + OR	Collaboration	All Text	Shared Goal
5	Query 4.1	DevOps	Abstract	AND + AND	Collaboration	All Text	Shared Understanding
	Query 4.2	DevOps	All Text	AND + AND	Collaboration	All Text	Shared Understanding
	Query 4.3	DevOps	All Text	OR + AND	Collaboration	Abstract	Shared Understanding
	Query 4.4	DevOps	All Text	AND + OR	Collaboration	All Text	Shared Understanding

Business Source Premier:

Table 1.3.app Search queries – step 1 – Business Source Premier

SRQ	Business Source Premier (EBSCO Host)						
	Criteria	Peer-reviewed	Full Text		Publication type:	Thesaurus research subjects	Publication Date
		Checkbox: Y	Checkbox: Y		Academic journal, (e)book	technological innovations, business enterprises, management, computer software, leadership, computer software development, information technology	2001 - 2020

1	Search Queries	Search Word 1	Selected field (search scope)	Operator(s)	Search Word 2	Selected field (search scope)	Search Word 3
	Query 1.1	DevOps	Abstract	AND	Collaboration	All Text	
	Query 1.2	DevOps	All Text	AND	Collaboration	Abstract	
	Query 1.3	DevOps	Abstract	OR	Collaboration	Abstract	
	Query 1.4	DevOps	All Text	AND	Collaboration	All Text	
2	Query 2.1	Shared Goal	Abstract	AND	DevOps	All Text	
	Query 2.2	Shared Goal	All Text	AND	DevOps	Abstract	
	Query 2.3	Shared Goal	Abstract	OR	DevOps	Abstract	
	Query 2.4	Shared Goal	All Text	AND	DevOps	All Text	
3	Query 3.1	Shared Understanding	Abstract	AND	DevOps	All Text	
	Query 3.2	Shared Understanding	All Text	AND	DevOps	Abstract	
	Query 3.3	Shared Understanding	Abstract	OR	DevOps	Abstract	
	Query 3.4	Shared Understanding	All Text	AND	DevOps	All Text	
	Query 3.5	Shared Understanding	Abstract	AND + OR	DevOps	Abstract	Cross-functional
	Query 3.6	Shared Understanding	All Text	OR + AND	DevOps	All Text	Cross-functional
	Query 3.7	Shared Understanding	Abstract	AND	Cross-functional	All Text	
	Query 3.8	Shared Understanding	All Text	AND	Cross-functional	Abstract	
	Query 3.9	Shared Understanding	Abstract	OR	Cross-functional	Abstract	
	Query 3.10	Shared Understanding	All Text	AND	Cross-functional	All Text	
4	Query 4.1	DevOps	Abstract	AND + AND	Collaboration	All Text	Shared Goal
	Query 4.2	DevOps	All Text	AND + AND	Collaboration	All Text	Shared Goal
	Query 4.3	DevOps	All Text	OR + AND	Collaboration	Abstract	Shared Goal

	Query 4.4	DevOps	All Text	AND + OR	Collaboration	All Text	Shared Goal
5	Query 4.1	DevOps	Abstract	AND + AND	Collaboration	All Text	Shared Understanding
	Query 4.2	DevOps	All Text	AND + AND	Collaboration	All Text	Shared Understanding
	Query 4.3	DevOps	All Text	OR + AND	Collaboration	Abstract	Shared Understanding
	Query 4.4	DevOps	All Text	AND + OR	Collaboration	All Text	Shared Understanding

JSTOR:

Table 1.4.app Search queries – step 1 – JSTOR

SRQ	JSTOR						
	Criteria	Peer-reviewed	Full Text		Publication type:	Research area	Publication Date
		Manual check	Checkbox: Y		Academic journal, (e)book	Business, Computer Science or Management	2001/01 - 2019/06
1	Search Queries	<u>Search Word 1</u>	<u>Selected field (search scope)</u>	<u>Operator(s)</u>	<u>Search Word 2</u>	<u>Selected field (search scope)</u>	<u>Search Word 3</u>
	Query 1.1	DevOps	Abstract	AND	Collaboration	All fields	
	Query 1.2	DevOps	All fields	AND	Collaboration	Abstract	
	Query 1.3	DevOps	Abstract	OR	Collaboration	Abstract	
	Query 1.4	DevOps	All fields	AND	Collaboration	All fields	
2	Query 2.1	Shared Goal	Abstract	AND	DevOps	All fields	
	Query 2.2	Shared Goal	All fields	AND	DevOps	Abstract	
	Query 2.3	Shared Goal	Abstract	OR	DevOps	All fields	
	Query 2.4	Shared Goal	All fields	AND	DevOps	All fields	
3	Query 3.1	Shared Understanding	Abstract	AND	DevOps	All fields	
	Query 3.2	Shared Understanding	All fields	AND	DevOps	Abstract	

	Query 3.3	Shared Understanding	Abstract	OR	DevOps	Abstract	
	Query 3.4	Shared Understanding	All fields	AND	DevOps	All fields	
	Query 3.5	Shared Understanding	Abstract	AND + OR	DevOps	Abstract	Cross-functional
	Query 3.6	Shared Understanding	All fields	OR + AND	DevOps	All fields	Cross-functional
	Query 3.7	Shared Understanding	Abstract	AND	Cross-functional	All fields	
	Query 3.8	Shared Understanding	All fields	AND	Cross-functional	Abstract	
	Query 3.9	Shared Understanding	Abstract	OR	Cross-functional	Abstract	
	Query 3.10	Shared Understanding	All fields	AND	Cross-functional	All fields	
4	Query 4.1	DevOps	Abstract	AND + AND	Collaboration	All fields	Shared Goal
	Query 4.2	DevOps	All fields	AND + AND	Collaboration	All fields	Shared Goal
	Query 4.3	DevOps	All fields	OR + AND	Collaboration	Abstract	Shared Goal
	Query 4.4	DevOps	All fields	AND + OR	Collaboration	All fields	Shared Goal
5	Query 4.1	DevOps	Abstract	AND + AND	Collaboration	All fields	Shared Understanding
	Query 4.2	DevOps	All fields	AND + AND	Collaboration	All fields	Shared Understanding
	Query 4.3	DevOps	All fields	OR + AND	Collaboration	Abstract	Shared Understanding
	Query 4.4	DevOps	All fields	AND + OR	Collaboration	All fields	Shared Understanding

Web of Science:

Table 1.5.app Search queries – step 1 – Web of Science

SRQ	Web of Science						
	Criteria	Peer-reviewed	Full Text		Publication type:	Research area	Publication Date
		Manual check	Checkbox: Y		Academic journal, (e)book Proceedings (conference) paper	Business Economics, Computer Science	2009-2019
1	Search Queries	<u>Search Word 1</u>	<u>Selected field</u>	<u>Operator(s)</u>	<u>Search Word 2</u>	<u>Selected field (search scope)</u>	<u>Search Word 3</u>

			(search scope)				
	Query 1.1	DevOps	Abstract	AND	Collaboration	All fields	
	Query 1.2	DevOps	All fields	AND	Collaboration	Abstract	
	Query 1.3	DevOps	Abstract	OR	Collaboration	Abstract	
	Query 1.4	DevOps	All fields	AND	Collaboration	All fields	
2	Query 2.1	Shared Goal	Abstract	AND	DevOps	All fields	
	Query 2.2	Shared Goal	All fields	AND	DevOps	Abstract	
	Query 2.3	Shared Goal	Abstract	OR	DevOps	All fields	
	Query 2.4	Shared Goal	All fields	AND	DevOps	All fields	
3	Query 3.1	Shared Understanding	Abstract	AND	DevOps	All fields	
	Query 3.2	Shared Understanding	All fields	AND	DevOps	Abstract	
	Query 3.3	Shared Understanding	Abstract	OR	DevOps	Abstract	
	Query 3.4	Shared Understanding	All fields	AND	DevOps	All fields	
	Query 3.5	Shared Understanding	Abstract	AND + OR	DevOps	Abstract	Cross-functional
	Query 3.6	Shared Understanding	All fields	OR + AND	DevOps	All fields	Cross-functional
	Query 3.7	Shared Understanding	Abstract	AND	Cross-functional	All fields	
	Query 3.8	Shared Understanding	All fields	AND	Cross-functional	Abstract	
	Query 3.9	Shared Understanding	Abstract	OR	Cross-functional	Abstract	
	Query 3.10	Shared Understanding	All fields	AND	Cross-functional	All fields	
4	Query 4.1	DevOps	Abstract	AND + AND	Collaboration	All fields	Shared Goal
	Query 4.2	DevOps	All fields	AND + AND	Collaboration	All fields	Shared Goal
	Query 4.3	DevOps	All fields	OR + AND	Collaboration	Abstract	Shared Goal
	Query 4.4	DevOps	All fields	AND + OR	Collaboration	All fields	Shared Goal

5	Query 4.1	DevOps	Abstract	AND + AND	Collaboration	All fields	Shared Understanding
	Query 4.2	DevOps	All fields	AND + AND	Collaboration	All fields	Shared Understanding
	Query 4.3	DevOps	All fields	OR + AND	Collaboration	Abstract	Shared Understanding
	Query 4.4	DevOps	All fields	AND + OR	Collaboration	All fields	Shared Understanding

1.3 Search queries – Second set

This section of appendix 1 presents the second set of search queries that has been applied during the second iteration of the building blocks method:

Academic Search Elite:

Table 1.6.app Search queries – step 2 – Academic Search Elite

SRQ	Academic Search Elite						
	Criteria	Peer-reviewed Checkbox: Y	Full Text Checkbox: Y		Publication type:	Thesaurus research subjects	Publication Date
					Academic journal, (e)book	technological innovations, business enterprises, management, computer software, leadership, computer software development, information technology	2001 - 2020
1	Search Queries	Search Word 1	Selected field (search scope)	Operator(s)	Search Word 2	Selected field (search scope)	Search Word 3
	Query 1.1	DevOps	Abstract	AND	Collaboration	All text	
	Query 1.2	DevOps	All text	AND	Collaboration	Abstract	
	Query 1.3	DevOps	Abstract	OR	Collaboration	Abstract	
	Query 1.4	DevOps	All text	AND	Collaboration	All text	
2	Query 2.1	Shared Goal	Abstract	AND	Collaboration	All text	
	Query 2.2	Shared Goal	All text	AND	Collaboration	Abstract	
	Query 2.3	Shared Goal	Abstract	OR	Collaboration	All text	
	Query 2.4	Shared Goal	All text	AND	Collaboration	All fields	
	Query 2.5	Shared Goal	Abstract	AND + OR	Collaboration	Abstract	Cross- functional
	Query 2.6	Shared Goal	All text	OR + AND	Collaboration	All text	Cross- functional
3	Query 3.1	Shared Understanding	Abstract	AND	Collaboration	All text	
	Query 3.2	Shared Understanding	All text	AND	Collaboration	Abstract	
	Query 3.3	Shared Understanding	Abstract	OR	Collaboration	Abstract	

	Query 3.4	Shared Understanding	All text	AND	Collaboration	All text	
	Query 3.5	Shared Understanding	Abstract	AND + OR	Collaboration	Abstract	Cross-functional
	Query 3.6	Shared Understanding	All text	OR + AND	Collaboration	All text	Cross-functional
	Query 3.7	Shared Understanding	Abstract	AND	Cross-functional	All text	
	Query 3.8	Shared Understanding	All text	AND	Cross-functional	Abstract	
	Query 3.9	Shared Understanding	Abstract	OR	Cross-functional	Abstract	
	Query 3.10	Shared Understanding	All text	AND	Cross-functional	All text	

Business Source Premier:

Table 1.7.app Search queries – step 2 – Business Source Premier

SRQ	Business Source Premier						
	Criteria	Peer-reviewed	Full Text		Publication type:	Thesaurus research subjects	Publication Date
		Checkbox: Y	Checkbox: Y		Academic journal, (e)book	technological innovations, business enterprises, management, computer software, leadership, computer software development, information technology	2001 - 2020
1	Search Queries	<u>Search Word 1</u>	<u>Selected field (search scope)</u>	<u>Operator(s)</u>	<u>Search Word 2</u>	<u>Selected field (search scope)</u>	<u>Search Word 3</u>
	Query 1.1	DevOps	Abstract	AND	Collaboration	All text	
	Query 1.2	DevOps	All text	AND	Collaboration	Abstract	
	Query 1.3	DevOps	Abstract	OR	Collaboration	Abstract	
	Query 1.4	DevOps	All text	AND	Collaboration	All text	
2	Query 2.1	Shared Goal	Abstract	AND	Collaboration	All text	
	Query 2.2	Shared Goal	All text	AND	Collaboration	Abstract	

	Query 2.3	Shared Goal	Abstract	OR	Collaboration	All text	
	Query 2.4	Shared Goal	All text	AND	Collaboration	All fields	
	Query 2.5	Shared Goal	Abstract	AND + OR	Collaboration	Abstract	Cross-functional
	Query 2.6	Shared Goal	All text	OR + AND	Collaboration	All text	Cross-functional
3	Query 3.1	Shared Understanding	Abstract	AND	Collaboration	All text	
	Query 3.2	Shared Understanding	All text	AND	Collaboration	Abstract	
	Query 3.3	Shared Understanding	Abstract	OR	Collaboration	Abstract	
	Query 3.4	Shared Understanding	All text	AND	Collaboration	All text	
	Query 3.5	Shared Understanding	Abstract	AND + OR	Collaboration	Abstract	Cross-functional
	Query 3.6	Shared Understanding	All text	OR + AND	Collaboration	All text	Cross-functional
	Query 3.7	Shared Understanding	Abstract	AND	Cross-functional	All text	
	Query 3.8	Shared Understanding	All text	AND	Cross-functional	Abstract	
	Query 3.9	Shared Understanding	Abstract	OR	Cross-functional	Abstract	
	Query 3.10	Shared Understanding	All text	AND	Cross-functional	All text	

JSTOR:

Table 1.8.app Search queries – step 2 – JSTOR

SRQ	JSTOR						
	Criteria	Peer-reviewed	Full Text		Publication type:	Research area	Publication Date
		Manual check	Checkbox: Y		Academic journal, (e)book	Business, Computer Science or Management	2001/01 - 2019/06
1	Search Queries	<u>Search Word 1</u>	<u>Selected field (search scope)</u>	<u>Operator(s)</u>	<u>Search Word 2</u>	<u>Selected field (search scope)</u>	<u>Search Word 3</u>
	Query 1.1	DevOps	Abstract	AND	Collaboration	All fields	
	Query 1.2	DevOps	All fields	AND	Collaboration	Abstract	
	Query 1.3	DevOps	Abstract	OR	Collaboration	Abstract	
	Query 1.4	DevOps	All fields	AND	Collaboration	All fields	

2	Query 2.1	Shared Goal	Abstract	AND	Collaboration	All fields	
	Query 2.2	Shared Goal	All fields	AND	Collaboration	Abstract	
	Query 2.3	Shared Goal	Abstract	OR	Collaboration	All fields	
	Query 2.4	Shared Goal	All fields	AND	Collaboration	All fields	
	Query 2.5	Shared Goal	Abstract	AND + OR	Collaboration	Abstract	Cross-functional
	Query 2.6	Shared Goal	All fields	OR + AND	Collaboration	All fields	Cross-functional
3	Query 3.1	Shared Understanding	Abstract	AND	Collaboration	All fields	
	Query 3.2	Shared Understanding	All fields	AND	Collaboration	Abstract	
	Query 3.3	Shared Understanding	Abstract	OR	Collaboration	Abstract	
	Query 3.4	Shared Understanding	All fields	AND	Collaboration	All fields	
	Query 3.5	Shared Understanding	Abstract	AND + OR	Collaboration	Abstract	Cross-functional
	Query 3.6	Shared Understanding	All fields	OR + AND	Collaboration	All fields	Cross-functional
	Query 3.7	Shared Understanding	Abstract	AND	Cross-functional	All fields	
	Query 3.8	Shared Understanding	All fields	AND	Cross-functional	Abstract	
	Query 3.9	Shared Understanding	Abstract	OR	Cross-functional	Abstract	
	Query 3.10	Shared Understanding	All fields	AND	Cross-functional	All fields	

Web of Science:

Table 1.9.app Search queries – step 2 – Web of Science

SRQ	Web of Science						
	Criteria	Peer-reviewed	Full Text		Publication type:	Research area	Publication Date
		Manual check	Checkbox: Y		Academic journal, (e)book Proceedings (conference) paper	Business Economics, Computer Science	2009-2019
1	Search Queries	<u>Search Word 1</u>	<u>Selected field (search scope)</u>	<u>Operator(s)</u>	<u>Search Word 2</u>	<u>Selected field (search scope)</u>	<u>Search Word 3</u>
	Query 1.1	DevOps	Abstract	AND	Collaboration	All fields	
	Query 1.2	DevOps	All fields	AND	Collaboration	Abstract	

	Query 1.3	DevOps	Abstract	OR	Collaboration	Abstract		
	Query 1.4	DevOps	All fields	AND	Collaboration	All fields		
2	Query 2.1	Shared Goal	Abstract	AND	Collaboration	All fields		
	Query 2.2	Shared Goal	All fields	AND	Collaboration	Abstract		
	Query 2.3	Shared Goal	Abstract	OR	Collaboration	All fields		
	Query 2.4	Shared Goal	All fields	AND	Collaboration	All fields		
	Query 2.5	Shared Goal	Abstract	AND + OR	Collaboration	Abstract	Cross-functional	A
	Query 2.6	Shared Goal	All fields	OR + AND	Collaboration	All fields	Cross-functional	A
3	Query 3.1	Shared Understanding	Abstract	AND	Collaboration	All fields		
	Query 3.2	Shared Understanding	All fields	AND	Collaboration	Abstract		
	Query 3.3	Shared Understanding	Abstract	OR	Collaboration	Abstract		
	Query 3.4	Shared Understanding	All fields	AND	Collaboration	All fields		
	Query 3.5	Shared Understanding	Abstract	AND + OR	Collaboration	Abstract	Cross-functional	A
	Query 3.6	Shared Understanding	All fields	OR + AND	Collaboration	All fields	Cross-functional	A
	Query 3.7	Shared Understanding	Abstract	AND	Cross-functional	All fields		
	Query 3.8	Shared Understanding	All fields	AND	Cross-functional	Abstract		
	Query 3.9	Shared Understanding	Abstract	OR	Cross-functional	Abstract		
	Query 3.10	Shared Understanding	All fields	AND	Cross-functional	All fields		

1.4 Valuable sources

This section presents all the sources that were found during the selection of valuable sources. There are four tables, each corresponding to one of the sub research questions. In the column relevant is depicted whether the source has relevant for tis research. Most of these sources have been selected for use. This can be found in the column 'decision'.

Table 1.10.app Valuable sources – SRQ1

Nr	Title	Author(s)	Year	Relevance	Decision	Source
1	The Effects of Process Orientations on Collaboration Technology Use and Outcomes in Product Development	Bala, Massey & Montoya	2017	No	No	Business source premier
2	Virtual team collaboration: building shared meaning, resolving breakdowns and creating translucence	Bjorn & Ngwenyama	2009	No	No	Web of Science
3	Advanced multi-phase trust evaluation model for collaboration between co-workers in dynamic virtual project teams	Chen & Chen	2009	No	No	Academic Search Elite
4	On the Journey to Continuous Deployment: Technical and Social Challenges Along the Way	Claps, Svensson & Aurum	2015	Yes	Yes	Business source premier & Web of Science
5	Collaboration technology in teams and organizations: Introduction to the special issue	De Vreede, Antunes, Vassileva, Gerosa & Wu	2016	No	No	Web of Science
6	The Impact of Information Technology on Academic Scientists' Productivity and Collaboration Patterns	Ding, Levin, Stephan & Winkler	2010	No	No	JSTOR
7	Using Free and Open Source Tools to Manage Software Quality	Dowling & McGrath	2015	No	No	Business source premier
8	A qualitative study of DevOps usage in practice	Erich, Amrit & Daneva	2017	Yes	Yes	Academic Search Elite
9	Challenges of interorganizational collaboration for information technology adoption: Insights from a governmental financial decision-making process in Egypt.	Ezz, Papazafeiropoulou	2009	No	No	Academic Search Elite
10	Resilience thinking: Integrating resilience, adaptability and transformability	Folke, Carpenter, Walker, Scheffer, Chapin III & Rockström	2010	Yes	Yes	Academic Search Elite, Business source premier & JSTOR
11	Of software and change	Ghezzi	2017	Yes	Yes	Academic Search Elite

12	Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling	Gupta, Kapur & Kumar	2017	Yes	Yes	Business source premier & Web of Science
13	Convenience matters: A qualitative study on the impact of use of social media and collaboration technologies on learning experience and performance in higher education	Jang	2015	No	No	Web of Science
14	Enhancing the Motivational Affordance of Information Systems: The Effects of Real-Time Performance Feedback and Goal Setting in Group Collaboration Environments	Jung, Schneider & Valacich	2010	No	No	JSTOR
15	Containers Will Not Fix Your Broken Culture (and Other Hard Truths)	Kromhout	2018	No	No	Business source premier
16	Toward agile: an integrated analysis of quantitative and qualitative field data	Lee & Xia	2010	Yes	Yes	Business source premier & Web of Science
17	Innovating or doing as Told? Status Differences and Overlapping Boundaries in Offshore Collaboration	Levina & Vaast	2008	No	No	JSTOR
18	Documentation Is Automation	Limocelli	2018	No	No	Business source premier
19	Stakeholder management studies in mega construction projects: a review and future directions.	Mok, Shen & Yang	2015	Yes	Yes	Business source premier & Web of Science
20	Definitions do make a difference: county managers and their conceptions of collaboration	Prentice & Brudney	2016	Yes	Yes	Business source premier & Web of Science
21	Definitions Do Make a Difference: County Managers and Their Conceptions of Collaboration.	Prentice & Brudney	2016	Yes	Yes	Academic Search Elite
22	Membership Turnover and Collaboration Success in Online Communities: Explaining Rises and Falls from Grace in Wikipedia	Ransbotham & Kane	2011	No	No	JSTOR
23	Adopting DevOps Practices in Quality Assurance.	Roche	2013	Yes	Yes	Business source premier & Web of Science

24	Modeling the relationship between firm IT capability, collaboration and performance	Sanders & Premus	2005	No	No	Business source premier
25	Supply chain collaboration: capabilities for continuous innovation, Supply Chain Management	Soosay, Hyland & Ferrer	2008	Yes	Yes	Business source premier
26	Not All International Collaboration is Beneficial: The Mendeley Readership and Citation Impact of Biochemical Research Collaboration	Sud & Thelwall	2016	No	No	Web of Science
27	Accelerating Application Delivery in a Hybrid World	Weir, Richard & Ueda	2018	No	No	Web of Science
28	Collaborative gathering and continuous delivery of DevOps solutions through repositories	Wettinger, Breitenbücher, Falkenthal & Leymann	2017	No	No	Web of Science
29	Architectural refactoring for the cloud: a decision-centric view on cloud migration	Zimmermann	2017	No	No	Business source premier
30	Defacto and Deeded Intellectual Property: Knowledge-Driven Co-Evolution of Firm Collaboration Boundaries and IPR Strategy	Zucker & Darby	2014	No	No	JSTOR

Table 1.11.app Valuable sources – SRQ2

ID	Title	Author(s)	Year	Relevance	Decision	Source
1	Teams in Pursuit of Radical Innovation	Alexander & Van Knippenberg	2014	Yes	Yes	Business source premier
2	Coordinating in construction projects and the emergence of synchronized readiness.	Bygballe, Swärd & Vaagaasar	2016	Yes	Yes	Business source premier
3	Competition through collaboration: insights from an initiative in the Turkish textile supply chain.	Cetindamar, Çatay & Basmacı	2005	Yes	Yes	Business source premier
4	Transformational leadership and inter-team collaboration	Cha, Kim, Lee & Bachrach	2015	Yes	Yes	Business source premier
5	A Model to Update Accounting Curricula for Emerging Technologies	Coyne, Coyne & Walker	2016	No	No	Business source premier
6	StarEast targets software test automation and quality	Dang, Klain & Gauli	2016	No	No	Academic Search Elite
7	Formal controls and team adaptability in new product development projects	Detzten, Verbeeten, Gamm & Möller	2018	Yes	Yes	Business source premier
8	How Collective Engagement Creates Competitive Advantage for Organizations	Eldor	2019	Yes	Yes	Business source premier
9	Organizing work to support relational coordination	Gittell	2000	No	No	Business source premier
10	Relational bureaucracy: structuring reciprocal relationships into roles	Gittell & Douglas	2012	No	No	Business source premier
11	Antecedents of Team Potency and Team Effectiveness	Hu & Liden	2011	Yes	Yes	Business source premier
12	Continuous Delivery Sounds Great, but Will It Work Here?	Humble	2018	No	No	Business source premier

13	Key Affordances of Platform-as-a-Service: Self-Organization and Continuous Feedback	Krancher, Luther & Jost	2018	No	No	Business source premier
14	Analyzing project management research: perspectives from top management journals	Kwak & Anbari	2009	Yes	Yes	Business source premier & Web of Science
15	How leading companies practice software development and delivery to achieve a competitive edge	Lesser & Ban	2016	No	No	Business source premier
16	10 Optimizations on Linear Search	Limocelli	2016	No	No	Business source premier
17	Project Goals, Team Performance, and Shared Understanding.	McComb & Green	1999	No	No	Business source premier
18	HPC Cloud for Scientific and Business Applications: Taxonomy, Vision, and Research Challenges	Netto, Calheiros, Rodrigues, Cunha & Buyya	2018	No	No	Business source premier
19	Enhancing workplace digital learning by use of the science of learning	Okano, Kacmarzyk & Gabrieli	2018	No	No	Academic Search Elite
20	Reducing the Software Value Gap	Pass & Ronen	2014	No	No	Business source premier
21	Polat Lynn Akgün Emre Formal and Informal Communication in New Product Development Teams	Polat, Lynn, Akgün & Emre	2018	Yes	Yes	Academic Search Elite
22	The Tangled Web: Unraveling the Principle of Common Goals in Collaborations	Vangen & Huxham	2012	Yes	Yes	Business source premier
23	A Personal Perspective on a Conceptual Foundation for Information Systems	Watson	2014	No	No	Business source premier
24	Streamlining DevOps automation for Cloud applications using TOSCA as standardized metamodel	Wettinger, Breitenbücher, Kopp & Leymann	2016	No	No	Business source premier & Web of Science
25	Cross-Functional Team Organizational Citizenship Behavior in China	Wong, Tjosvold & Liu	2009	No	No	Academic Search Elite
26	Centralized and Decentralized Decision Making in Organizations	Zábojník	2002	Yes	Yes	Business source premier

Table 1.12.app Valuable sources – SRQ3

ID	Title	Author(s)	Year	Relevance	Decision	Source
1	High Touch Through High Tech: The Impact of Salesperson Technology Usage on Sales Performance via Mediating Mechanisms	Ahearne, Johnes, Rapp & Mathieu	2008	No	No	Business source premier
2	Feedback Loops as Dynamic Processes of Organizational Knowledge Creation in the Context of the Innovations' Front-end	Akbar, Baruch & Tzokas	2018	Yes	Yes	Business source premier & Web of Science
3	Perceived shared understanding in teams	Aubé, Rousseau & Tremblay	2015	Yes	Yes	Academic Search Elite
4	The Role of Knowledge Management in the Innovation Process.	Basadur & Gelade	2006	Yes	Yes	Business source premier

5	Contract design choices and the balance of ex ante and ex post transaction costs in software development outsourcing	Benaroch, Liechtenstein & Fink	2016	No	No	Business source premier
6	Assessing the quality of Shared Priorities in teams using content analysis in a microworld experiment	Berggren, Johansson & Baroutsi	2017	Yes	Yes	Web of Science
7	Creating Shared Understanding in Heterogeneous Work Groups: Why It Matters and How to Achieve It.	Bittner & Leimeister	2014	Yes	Yes	Business source premier & Web of Science
8	Business-IT Alignment: A practical research approach.	Charoensuk, Wongsurawat & Khang	2014	Yes	Yes	Business source premier
9	Data, information, and knowledge-driven manipulation between strategical planning and technical implementation for wireless sensor network construction	Duan, Shao, Yang, Sun, Zhou & Yu	2017	No	No	Web of Science
10	Multi-Disease Data Management System Platform for Vector-Borne Diseases	Eisen, Coleman, Lozano-Fuentes, McEachen, Orlans & Coleman	2011	No	No	Academic Search Elite
11	Withdrawal of Team Autonomy During Concurrent Engineering	Gerwin & Moffat	1997	No	No	Business source premier
12	Perceived barriers to effective knowledge sharing in agile software teams	Ghobadi & Mathiasen	2016	Yes	Yes	Business source premier & Web of Science
13	Open source electronic health records and chronic disease management	Goldwater, Kwon, Nathanson, Muckle, Brown & Cornejo	2014	No	No	Web of Science
14	A theory of team coaching	Hackman & Wageman	2005	Yes	Yes	Business source premier
15	The social interaction of developers and IT operations staff in software development projects	Iden & Bygstad	2018	Yes	Yes	Business source premier & Web of Science
16	Cutthroat Cooperation: Asymmetrical Adaptation to Changes in Team Reward Structures	Johnson, Hollenbeck, Humphrey,	2006	No	No	JSTOR

		Ilgen, Jundt & Meyer				
17	Changing the Competitive Landscape: Continuous Innovation Through IT-Enabled Knowledge Capabilities	Joshi, Chi, Datta & Han	2010	Yes	Yes	Business source premier & Web of Science
18	Theoretical Framework and Literature Review in Graduate Records Management Research	Kemoni	2008	No	No	Academic Search Elite
19	The Downside of Self-Management: A Longitudinal Study of the Effects of Conflict on Trust, Autonomy, and Task Interdependence in Self-Managing Teams	Langfred	2007	No	No	JSTOR
20	Web Service Composition: A Survey of Techniques and Tools	Lemos, Daniel & Benatallah	2016	No	No	Business source premier
21	Bridging knowledge boundaries: the use of boundary objects in virtual innovation communities.	Marheineke, Habicht & Möslein	2016	No	No	Business source premier
22	Virtual organization: specifics of creation of personnel management system	Merkevicius, Davidaviciene, Raudeliuniene & Buleca	2015	No	No	Web of Science
23	Team mental models in a team knowledge framework	Mohammed & Dumville	2001	Yes	Yes	Business source premier
24	Developing a reverse logistics competency: The influence of collaboration and information technology	Morgan, Richey Jr & Autry	2016	Yes	Yes	Business source premier & Web of Science
25	The comprehensive assessment of team member effectiveness: Development of a behaviorally anchored rating scale for self and peer evaluation	Ohland, Loughry, Woehr, Bullard, Felder, Finelli, Layton, Pomeranz & Schmucker	2012	Yes	Yes	Business source premier
26	Stroke patients' utilisation of extrinsic feedback from computer-based technology in the home: a multiple case study realistic evaluation	Parker, Mawson, Mountain, Nasr & Huri	2014	No	No	Academic Search Elite
27	Dynamic Shared Leadership Theory: Understanding the Structures and Processes of Shared Leadership	Powers, Morgeson & Lyons	2014	Yes	Yes	Business source premier

28	The determinants of ICT competencies among employees	Tijdens & Stein	2005	No	No	Business source premier
29	Knowledge sharing using IT service management tools: conflicting discourses and incompatible practices	Trusson, Doherty & Hislop	2014	No	No	Business source premier
30	Cross-Functional collaboration, competitive intensity, knowledge integration mechanisms, and new product performance: A mediated moderation model.	Tsai & Hsu	2014	Yes	Yes	Business source premier
31	Resilience, adaptability and transformability in social-ecological systems	Walker, Holling, Carpenter & Kinzig	2004	Yes	Yes	Academic Search Elite & JSTOR
32	Drivers of management accounting adaptability: the agility lens	Yigitbasioglu	2017	Yes	No	Web of Science

1.5 Selected sources – Building blocks method

This section of appendix 1 presents all the selected sources using the building blocks method, see table 1.13.app for more information.

Table 1.13.app: sources - authors and titles

Nr	Title	Author(s)	Year
1	Feedback Loops as Dynamic Processes of Organizational Knowledge Creation in the Context of the Innovations' Front-end	Akbar, Baruch & Tzokas	2018
2	Teams in Pursuit of Radical Innovation	Alexander & Van Knippenberg	2014
3	Perceived shared understanding in teams	Aubé, Rousseau & Tremblay	2015
4	The Role of Knowledge Management in the Innovation Process.	Basadur & Gelade	2006
5	Assessing the quality of Shared Priorities in teams using content analysis in a microworld experiment	Berggren, Johansson & Baroutsi	2017
6	Creating Shared Understanding in Heterogeneous Work Groups: Why It Matters and How to Achieve It.	Bittner & Leimeister	2014
7	Coordinating in construction projects and the emergence of synchronized readiness.	Bygballe, Swärd & Vaagaasar	2016
8	Competition through collaboration: insights from an initiative in the Turkish textile supply chain.	Cetindamar, Çatay & Basmacı	2005
9	Transformational leadership and inter-team collaboration	Cha, Kim, Lee & Bachrach	2015
10	Business-IT Alignment: A practical research approach.	Charoensuk, Wongsurawat & Khang	2014
11	On the Journey to Continuous Deployment: Technical and Social Challenges Along the Way	Claps, Svensson & Aurum	2015
12	Formal controls and team adaptability in new product development projects	Detzten, Verbeeten, Gamm & Möller	2018
13	How Collective Engagement Creates Competitive Advantage for Organizations	Eldor	2019
14	A qualitative study of DevOps usage in practice	Erich, Amrit & Daneva	2017
15	Resilience thinking: Integrating resilience, adaptability and transformability	Folke, Carpenter, Walker, Scheffer, Chapin III & Rockström	2010
16	Of software and change	Ghezzi	2017

17	Perceived barriers to effective knowledge sharing in agile software teams	Ghobadi & Mathiassen	2016
18	Modeling and measuring attributes influencing DevOps implementation in an enterprise using structural equation modeling	Gupta, Kapur & Kumar	2017
19	A theory of team coaching	Hackman & Wageman	2005
20	Antecedents of Team Potency and Team Effectiveness	Hu & Liden	2011
21	The social interaction of developers and IT operations staff in software development projects	Iden & Bygstad	2018
22	Changing the Competitive Landscape: Continuous Innovation Through IT-Enabled Knowledge Capabilities	Joshi, Chi, Datta & Han	2010
23	Analyzing project management research: perspectives from top management journals	Kwak & Anbari	2009
24	Toward agile: an integrated analysis of quantitative and qualitative field data	Lee & Xia	2010
25	Team mental models in a team knowledge framework	Mohammed & Dumville	2001
26	Stakeholder management studies in mega construction projects: a review and future directions.	Mok, Shen & Yang	2015
27	Developing a reverse logistics competency: The influence of collaboration and information technology	Morgan, Richey Jr & Autry	2016
28	The comprehensive assessment of team member effectiveness: Development of a behaviorally anchored rating scale for self and peer evaluation	Ohland, Loughry, Woehr, Bullard, Felder, Finelli, Layton, Pomeranz & Schmucker	2012
29	Polat Lynn Akgün Emre Formal and Informal Communication in New Product Development Teams	Polat, Lynn, Akgün & Emre	2018
30	Dynamic Shared Leadership Theory: Understanding the Structures and Processes of Shared Leadership	Powers, Morgeson & Lyons	2014
31	Definitions do make a difference: county managers and their conceptions of collaboration	Prentice & Brudney	2016
32	Adopting DevOps Practices in Quality Assurance.	Roche	2013
33	Supply chain collaboration: capabilities for continuous innovation, Supply Chain Management	Soosay, Hyland & Ferrer	2008
34	Cross-Functional collaboration, competitive intensity, knowledge integration mechanisms, and new product performance: A mediated moderation model.	Tsai & Hsu	2014
35	The Tangled Web: Unraveling the Principle of Common Goals in Collaborations	Vangen & Huxham	2012
36	Resilience, adaptability and transformability in social–ecological systems	Walker, Holling, Carpenter & Kinzig	2004
37	Centralized and Decentralized Decision Making in Organizations	Zábojník	2002

1.6 Selected sources – Result per search engine

The first column of table 1.5.app shows the identifying number for the literature, column 2-5 show in which database the source was found and the last column shows whether it was peer reviewed. A color coding is used to show in how many databases the source was found. Id 9 and 16 are found in three databases, which shows that it a well spread paper.

Table 1.14.app: results per search engine

Id	Academic Search Elite	Business Source Premier	Web of Science	Jstor	Peer reviewed
1		x	x		Yes
2		x			
3	x				
4		x			Yes
5			x		
6		x	x		Yes
7		x	x		Yes
8		x			Yes
9		x			
10		x			Yes
11		x	x		Yes
12		x			Yes
13		x			Yes
14	x		x		Yes
15	x		x	x	Yes
16	x				Yes
17		x	x		Yes
18		x	x		Yes
19		x			Yes
20		x			Yes
21		x	x		Yes
22		x	x		Yes
23		x	x		Yes
24		x	x	x	Yes
25		x			Yes
26		x	x		Yes
27		x	x		Yes
28		x			Yes
29	x				Yes
30		x			Yes
31		x	x		Yes
32		x	x		Yes
33		x			Yes
34		x			Yes

35		x			Yes
36	x			x	Yes
37		x			
	<u>7</u>	<u>30</u>	<u>16</u>	<u>3</u>	<u>37</u>

1.7 ABDC & Scimago

The influence of the research paper was also taken into account. It has not been included as an inclusion criterion, because there are not many DevOps articles. The decision has been made, however, to give an insight of the literature that has been used, because a large portion consists of A/A* rated journals (ABDC, 2019) and have a high h-index (Scimagojr, 2019). The ABDC list is a renowned source for assessing the quality of a scientific journal. A* journals are the journals with a high level of influence, while D journals are not very influential. Eight papers are published in a journal that has no entry in the ABDC journal. From the 29 that are, 26 have a rating of A or A*. The results are depicted in table 2 (third column).

Scimago is one the well-known assessors of journals. One of the metrics they use is the Q1-Q4 rating. It is a quartile division of journals regarding research subjects (Scimagojr, 2019). The most influential 25% is Q1 and the least influential 25% is Q4. Out of the 37 research papers, 4 had no entry in the Scimago database. From the other 33, none was of the Q4 level. An astonishing 24 are of level Q1, 7 of level Q2 and 3 of level Q3. Scimago also has a SJR rating (year 2017), which is similar to the impact factor. It accounts for the number of citations received by a journal and the importance of the journals where the citations come from. Both metrics are found in table 2 (last column).

1.8 Building blocks – Per research question

The following three tables will show the keywords that were used in the search. In EBSCO host the option ‘also search for related words’ was selected, to get a big result set. The results have been found by applying combinations of the keywords that have been selected for the specific sub research question. The searches were conducted in the title and the abstract.

Table 1.15.app shows the sources that were found, using the building block method, for sub research question 1. The building block method was quite effective for this sub research question.

Table 1.15.app: sources related to sub research question 1

Id	Relevant concepts	Sub research question	Academic Search Elite	Business Source Premier	Web of Science	Jstor
11	Collaboration, DevOps	1		x	x	
14	DevOps, Collaboration	1		x	x	
15	Collaboration	1		x	x	
16	DevOps, Collaboration	1		x	x	
18	DevOps, Collaboration	1		x	x	
24	Collaboration, Agile software development	1		x	x	
26	Collaboration	1		x	x	
31	Collaboration	1		x	x	
32	DevOps	1		x		
33	Collaboration	1		x		

Table 1.16.app shows the sources that were found, using the building block method, for sub research question 3. The building block method was quite effective for this sub research question.

Table 1.16.app: sources related to sub research question 2

Id	Relevant concepts	Sub research question	Academic Search Elite	Business Source Premier	Web of Science	Jstor
2	Shared goal, Cross-functional, Collaboration	2		x		
7	Shared goal	2	x			
8	Shared goal, Cross-functional, Collaboration	2		x	x	
9	Cross-functional collaboration	2		x		
12	Cross-functional, Collaboration	2		x		
13	Shared goal, Collaboration	2		x		
20	Shared goal, Collaboration	2		x		
23	Shared goal	2		x	x	
29	Shared goal, Cross-functional, Collaboration	2	x			
35	Shared goal, Cross-functional, Collaboration	2		x		

37	Shared goal	2	x		
----	-------------	---	---	--	--

Table 1.17.app shows the sources that were found, using the building block method, for sub research question 3. The building block method was quite effective for this sub research question.

Table 1.17.app: sources related to sub research question 3

Id	Relevant concepts	Sub research question	Academic Search Elite	Business Source Premier	Web of Science	Jstor
1	Collaboration, Shared understanding	3		x		
3	Shared understanding	3	x			
4	Cross-functional, Collaboration	3		x	x	
5	Shared understanding, collaboration	3			x	
6	Shared understanding, Cross-functional	3	x		x	
10	Collaboration, Shared understanding	3		x	x	x
17	Shared understanding	3		x	x	
19	Shared understanding, Cross-functional, Collaboration	3		x		
21	Shared understanding, Cross-functional, Collaboration	3		x		
22	Cross-functional, Collaboration	3	x		x	x
25	Shared understanding	3		x		
27	Cross-functional, Collaboration	3	x			x
28	Shared understanding	3		x		
30	Shared understanding	3		x		
34	Cross-functional, Collaboration	3		x	x	
36	Collaboration, Cross-functional	3		x		

1.9 Snowballing methods

In this section of appendix 1 presents more information regarding the performed snowballing methods for research question 1, 2 and 3. In table 1.18.app-1.20.app are the methods depicted. The first column shows the starting paper. The fourth column shows whether a source is relevant. Some sources are relevant, but not added as a source. For instance, since it is already a source. The last column shows the retrieved information, based on the concept in table 2.5.

Table 1.18.app: collaboration – SRQ1 – backward snowballing

Starting paper	ID	References (used by Lwakatare, Kuvaja & Oivo) that met criteria	Relevant?	Added as a reference	Retrieved information
Lwakatare, Kuvaja & Oivo (2016)	1	Leppanen, M., Makinen, S., Pagels, M., Eloranta, V.-P., Itkonen, J., Mantyla, M. V. & Mannisto, T. (2015) The Highways and Country Roads to Continuous Deployment IEEE Software. vol. 32-22. pp. 64–72.	No	No	-
	2	Rodríguez, P., Haghhighatkah, A., Lwakatare, L.E., Teppola, S., Suomalainen, T., Eskeli, J., Karvonen, T., Kuvaja, P., Verner, J.M. & Oivo, M. (2016) Continuous Deployment of Software Intensive Products and Services: A Systematic Mapping Study Journal of Systems and Software	No	No	-
	3	Penners R. & Dyck, A. (2015) Release Engineering vs. DevOps An Approach to Define Both Terms Full-scale Software Engineering	Yes	Yes	Definition collaboration, Discussion of the concept
	4	Lwakatare, L.E., Kuvaja, P. & Oivo, M. (2015) Dimensions of DevOps Paper presented at the International Conference on Agile Software Development.	Yes	Yes	Definition collaboration, Discussion of the concept
	5	Humble, J. & Molesky, J. (2011) Why enterprises must adopt DevOps to enable continuous delivery Cutter IT Journal. Vol. 24-8. pp. 6-12	Yes	Yes	Definition collaboration, Discussion of the concept
	6	Iden, J., Tessem, B. & Paivärinta, T. (2011) Problems in the interplay of development and IT operations in system development projects: A Delphi study of Norwegian IT experts	Yes	Yes	Discussion of the concept Main aspects collaboration

		Information and Software Technology. Fol. 53-4. pp. 394–406.			
	7	Edith, T., Aurum, A. & Vidgen, R. (2013) An exploration of technical debt Journal of Systems and Software. Vol. 86,-6. pp. 1498–1516	No	No	-
	8	Dyck, A., Penners, R. & Lichter, H. (2015) Towards definitions for release engineering and DevOps	Yes	Yes	Discussion of the concept
	9	Roche, J. (2013) Adopting DevOps practices in quality assurance Communications of the ACM. pp. 1–8.	Already a Source	No	-
	10	Ebert, C., Gallardo, G., Hernantes, J. & Serrano, N. (2016) DevOps IEEE Software. Vol. 33-3. pp. 94–100.	No	No	-

Table 1.19.app: collaboration – SRQ2 – forward snowballing

Starting paper	ID	References (that used Jassawalla & Sashittal) that met criteria	Relevant?	Added as a reference	Retrieved information
Jassawalla & Sashittal (1998)	1	Kyriazis, E. Massey, G., Couchman P. & Johnson, L. (2015) Friend or foe? The effects of managerial politics on NPD team communication, collaboration and project success R&D Management. Vol. 47-1. pp. 61-74.	No		
	2	David A. Griffith and Hannah S. Lee (2016) Cross–National Collaboration of Marketing Personnel within a Multinational: Leveraging Customer Participation for New Product Advantage Journal of International Marketing. 10.1509/jim.16.0028. Vol. 24-4. pp. 1-19.	No		
	3	Stock, R.M., Totzauer, F. & Zacharias, N.A. (2013) A Closer Look at Cross-functional R&D Cooperation for Innovativeness: Innovation-oriented Leadership and Human Resource Practices as Driving Forces	Yes	Yes	Sub-aspects of Shared Goal

		Journal of Product Innovation Management. Vol. 31-5. pp. 924-938.			
	4	Yan T. & Dooley, K. (2014) Buyer–Supplier Collaboration Quality in New Product Development Projects Journal of Supply Chain Management. Vol. 50-2. pp.59-83.	No	No	
	5	Aronson, Z. H., Shenhar, A. J., & Patanakul, P. (2013). Managing the Intangible Aspects of a Project: The Affect of Vision, Artifacts, and Leader Values on Project Spirit and Success in Technology-Driven Projects. Project Management Journal, vol. 44-1. pp. 35–58.	Yes	Yes	Discussion of the concept Sub-aspects Shared Goal
	6	Un, C. A., Cuervo-Cazurra, A. & Asakawa, K. (2010) R&D Collaborations and Product Innovation Journal of Product Innovation Management. Vol. 27-5. pp. 673-689.	No	No	
	7	Thamhain, H.J. (2009) Leadership lessons from managing technology-intensive teams. International Journal of Innovation and Technology Management. Vol. 06-2	Yes	Yes	Discussion of the concept Sub-aspects Shared Goal
	8	Lynn G.S. & Akgün, A.E. (2003) Project visioning: Its components and impact on new product success Journal of Product Innovation Management. Vol. 18-6. pp. 374-387.	Yes	Yes	Discussion of the concept Sub-aspects Shared Goal

Table 1.20.app: collaboration – SRQ3 – backward snowballing

Starting paper	ID	References (used by Bittner & Leimeister) that met criteria	Relevant?	Added as a reference	Retrieved information
Bittner & Leimeister (2014)	1	<u>Akkerman, S.; Van den Bossche, P.; Admiraal, W.; Gijsselaers, W., Segers; M., Simons, R.-J.; and Kirschner, P. (2007) Reconsidering group cognition: From conceptual confusion to a boundary area between cognitive and socio-cultural perspectives?</u> <u>Educational Research Review, 2, 1. pp. 39–63.</u>	Yes		Definition Shared Understanding, Discussion of the concept

	2	<i>Briggs, R.O. (2006) On theory-driven design and deployment of collaboration systems. International Journal of Human-Computer Studies, 64, 7. pp. 573–582.</i>	No	No	-
	3	<i>Briggs, R.O., and Reinig, B.A. (2010) Bounded ideation theory. Journal of Management Information Systems, 27, 1, 127–149.</i>	No	No	-
	4	<i>Briggs, R.O.; Kolfshoten, G.L.; Vreede, G.J. de; Lukosch, S.; and Albrecht, C.C. (2013) Facilitator-in-a-box: Process support applications to help practitioners realize the potential of collaboration technology. Journal of Management Information Systems, 29, 4. pp. 159–193.</i>	No	No	-
	5	<i>Fischer, F., and Mandl, H. (2005) Knowledge convergence in computer-supported collaborative learning: The role of external representation tools. Journal of the Learning Sciences, 14, 3. pp. 405–441.</i>	No	No	-
	6	<u>Garfield, M.J., and Dennis, A.R. (2012) Toward an integrated model of group development: Disruption of routines by technology-induced change. Journal of Management Information Systems, 29, 3. pp. 43–86.</u>	Yes	No	-
	7	<i>Hevner, A.R.; March, S.T.; Park, J.; and Ram, S. (2004) Design science in information systems research. MIS Quarterly, 28, 1. pp. 75–105.</i>	No	No	-
	8	<u>Jeong, H., and Chi, M. (2007) Knowledge convergence and collaborative learning. Instructional Science, 35, 4. pp. 287–315.</u>	No	No	-
	9	<i>Kleinsmann, M.; Buijs, J.; and Valkenburg, R. (2010) Understanding the complexity of knowledge integration in collaborative new product development teams: A case study. Journal of Engineering and Technology Management, 27, 1–2. pp. 20–32.</i>	No	No	-
	10	<u>Kleinsmann, M., and Valkenburg, R. (2008) Barriers and enablers for creating shared understanding in co-design projects. Design Studies, 29, 4. pp. 369–386.</u>	Yes	Yes	Discussion of the concept

11	<i>Kolfschoten, G.; Briggs, R.O.; Vreede, G.J. de; Jacobs, P.H.M.; and Appelman, J.H. (2006)</i> <i>A conceptual foundation of the thinkLet concept for collaboration engineering. International Journal of Human-Computer Studies, 64, 7. pp. 611–621.</i>	No	No	-
12	<i>Leimeister, J.M.; Huber, M.; Bretschneider, U.; and Krcmar, H. (2009)</i> <i>Leveraging crowdsourcing: Activation-supporting components for IT-based ideas competition. Journal of Management Information Systems, 26, 1. pp. 197–224.</i>	No	No	-
13	<i>McKay, J., and Marshall, P. (2001)</i> <i>The dual imperatives of action research. Information Technology & People, 14, 1. pp. 46–59.</i>	No	No	-
14	<u>Mohammed, S., and Dumville, B.C. (2001)</u> <u>Team mental models in a team knowledge framework: Expanding theory and measurement across disciplinary boundaries.</u> <u>Journal of Organizational Behavior, 22, 2. pp. 89–106.</u>	Yes	Already a Source	Definition Shared Understanding, Discussion of the concept Sub-aspects Shared Understanding
15	<u>Mohammed, S.; Ferzandi, L.; and Hamilton, K. (2010)</u> <u>Metaphor no more: A 15-year review of the team mental model construct.</u> <u>Journal of Management, 36, 4. pp. 876–910.</u>	Yes	No	-
16	<i>Reinig, B.A.; Briggs, R.O.; and Nunamaker, J.F., Jr. (2007)</i> <i>On the measurement of ideation quality. Journal of Management Information Systems, 23, 4. pp. 143–161.</i>	No	No	-
17	<u>Van den Bossche, P.; Gijssels, W.; Segers, M.; Woltjer, G.; and Kirschner, P. (2011)</u> <u>Team learning: Building shared mental models.</u> <u>Instructional Science, 39, 3. pp. 283–301.</u>	Yes	Yes	Definition Shared Understanding, Discussion of the concept Sub-aspects Shared Understanding
18	<i>Wegge, J.; Roth, C.; Neubach, B.; Schmidt, K.-H.; and Kanfer, R. (2008)</i> <i>Age and gender diversity as determinants of performance and health in a public organization: The role of task complexity and group size.</i>	No	No	-

		<i>Journal of Applied Psychology, 93, 6 (2008), 1301–1313.</i>			
--	--	--	--	--	--

1.10 Selected literature – Building blocks & snowballing method

This section of appendix 1 presents all the selected sources. The information is depicted in table 1.21.app.

Table 1.21.app: all selected sources – building blocks and snowballing method

Nr	Author(s)	Year	Retrieved Information
1	Akbar, Baruch & Tzokas	2018	Discussion of the concept
2	Alexander & Van Knippenberg	2014	Definition of Shared Goal
			Discussion of the concept
3	Aubé, Rousseau & Tremblay	2015	Definition of Shared Understanding
4	Basadur & Gelade	2006	Discussion of the concept
5	Berggren, Johansson & Baroutsi	2017	Sub-aspects of Shared Understanding
6	Bittner & Leimeister	2014	Definition of Shared Understanding
			Discussion of the concept
			Sub-aspects of Shared Understanding
7	Bygballe, Swärd & Vaagaasar	2016	Discussion of the concept
8	Cetindamar, Çatay & Basmaci	2005	Definition of Shared Goal
			Discussion of the concept
9	Cha, Kim, Lee & Bachrach	2015	Discussion of the concept

			Sub-aspects of Shared Goal
10	Charoensuk, Wongsurawat & Khang	2014	Discussion of the concept
11	Claps, Svensson & Aurum	2015	Definition of Collaboration
			Discussion of the concept
12	Detzten, Verbeeten, Gamm & Möller	2018	Definition of Shared Goal
			Discussion of the concept
13	Eldor	2019	Sub-aspects of Shared Goal
14	Erich, Amrit & Daneva	2017	Definition of Collaboration
			Discussion of the concept
			Sub-aspects of Collaboration
15	Folke, Carpenter, Walker, Scheffer, Chapin III & Rockström	2010	Discussion of the concept
16	Ghezzi	2017	Definition of Collaboration
			Discussion of the concept
17	Ghobadi & Mathiassen	2016	Discussion of the concept
18	Gupta, Kapur & Kumar	2017	Definition of Collaboration

			Discussion of the concept
			Sub-aspects of Collaboration
19	Hackman & Wageman	2005	Sub-aspects of Shared Understanding
20	Hu & Liden	2011	Sub-aspects of Shared Goal
21	Iden & Bygstad	2018	Definition of Shared Understanding
			Discussion of the concept
			Sub-aspects of Shared Understanding
22	Joshi, Chi, Datta & Han	2010	Discussion of the concept
23	Kwak & Anbari	2009	Definition of Shared Goal
24	Lee & Xia	2010	Discussion of the concept
25	Mohammed & Dumville	2001	Definition of Shared Understanding
			Discussion of the concept
			Sub-aspects of Shared Understanding
26	Mok, Shen & Yang	2015	Discussion of the concept
27	Morgan, Richey Jr & Autry	2016	Discussion of the concept
28	Ohland, Loughry, Woehr, Bullard, Felder,	2012	Sub-aspects of Shared Understanding
	Finelli, Layton, Pomeranz & Schmucker		
29	Polat, Lynn, Akgün & Emre	2018	Definition of Shared Goal
			Discussion of the concept
30	Powers, Morgeson & Lyons	2014	Discussion of the concept

31	Prentice & Brudney	2016	Discussion of the concept
32	Roche	2013	Definition of Collaboration
			Discussion of the concept
33	Soosay, Hyland & Ferrer	2008	Discussion of the concept
34	Tsai & Hsu	2014	Discussion of the concept
35	Vangen & Huxham	2012	Discussion of the concept
			Sub-aspects of Shared Goal
36	Walker, Holling, Carpenter & Kinzig	2004	Discussion of the concept
37	Zábojník	2002	Sub-aspects of Shared Goal
38	Penners & Dyck	2015	Definition collaboration,
			Discussion of the concept
39	Lwakatare, Kuvaja & Oivo	2015	Definition collaboration,
			Discussion of the concept
40	Humble & Molesky	2011	Definition collaboration,
			Discussion of the concept
41	Iden, Tessem & Paivärinta	2011	Discussion of the concept
			Sub-aspects collaboration
42	Dyck, Penners & Lichter	2015	Discussion of the concept
43	Stock, Totzauer & Zacharias	2013	Sub-aspects of Shared Goal
44	Aronson, Shenhar & Patanakul	2013	Discussion of the concept
			Sub-aspects Shared Goal
45	Thamhain	2009	Discussion of the concept
			Sub-aspects Shared Goal
46	Lynn & Akgün	2003	Discussion of the concept
			Sub-aspects Shared Goal
47	Akkerman, Van den Bossche, Admiraal, Gijsselaers, Segers, Simons, & Kirschner	2007	Definition Shared Understanding,
			Discussion of the concept
48	Kleinsmann & Valkenburg	2008	Discussion of the concept
50	Van den Bossche, Gijsselaers, Segers, Woltjer & Kirschner	2011	Definition Shared Understanding,
			Discussion of the concept
			Sub-aspects Shared Understanding

Appendix 2: Method

This section of the appendix contains all the relevant information regarding the research method. Paragraph 2.1 portrays the unstructured interview that is conducted early in the process. Paragraph 2.2 to 2.7 are related to the semi-structured interview; 2.2 presents the interview tables that are used during the interview, 2.3 shows the argumentation per research question, 2.4 and 2.5 present the interview protocols and 2.6 and 2.7 portrait the letters that have been given to the respondents.

2.1 Unstructured interview

This section contains the unstructured interview that has been conducted with respondent3. He is the informant of this research. The goal of the unstructured was to assess whether DevOps1 is suitable as a research subject.

STRUCTURED INTERVIEW

JOB TITLE:	Development Manager
SURROUNDINGS:	Face-to-face meeting, followed up by a brief phone call
INTERVIEWER	Timo Nadorp
DEVOPS:	<p>The organization is focusing on making team collaborate more and become autonomous. There are multidisciplinary teams, with specialisms like testing, developing and operations.</p> <p>The definition of DevOps that is DevOps1 uses is: <i>“continuously improving the way teams operate and collaborate in an attempt to meet the demands of the customers”</i>. They have started growing towards a DevOps organization in the last two years.</p>
ROLES:	All the roles that are needed for the interviews are present in this organization. They are also willing and able to help
RESEARCH:	<p>Several people around and in the teams are available for an interview.</p> <p>DevOps1 seems to be a suitable organization to investigate. The elements that are required to answer the research question are present and available. Results from the interviews are expected to be relevant for the results of this research.</p>

2.2 Interview tables

In this paragraph of appendix 2 are the interview tables, which contain all the interview questions in an orderly fashion. There is a total of ten tables. Starting with an introduction and concluding remarks table. Then there is a table (2) with question regarding collaboration, followed up by tables for shared goal (3) and its sub-aspects (4-6) and shared understanding (7) and its sub-aspects (8-10).

Table 2.1.App: Introduction and concluding remarks

Introduction & Concluding statement				
Input			Output	
Question nr.	Question	Sources	Examples	Relevant data (output)
1	What is your role within the organisation?	-		
1a	Welke verantwoordelijkheden horen hierbij?	-		
1b	Hoe lang werk je al voor DevOps1?	-		
1c	Hoeveel ervaring heb je met DevOps?			
32	Any last remarks or things you would like to add?	-		

Table 2.2.App: Collaboration

Dimension of DevOps: Collaboration				
Input			Output	
Question nr.	Question	Sources	Examples	Relevant data (output)
2	What is your observation regarding the collaboration of cross-functional team members in the DevOps team(s)?	Lwakatare et al. (2015); Humble and Molesky (2011); Bang et al., 2013;		
3	What is the impact of the introduction of DevOps on the way team members collaborate?	Iden & Bygstad, 2018 Charoensuk et al., 2014; Bygballe et al., 2016 Polat et al., 2018; Kozłowski & Bell, 2003; LePine, 2005		
3a	What do you think about the definition of collaboration?			

Table 2.3.App: Shared Goal

Main-aspects of Collaboration: Shared Goal				
Input			Output	
Question nr.	Question	Sources	Examples	Relevant data (output)

4	What do you think of the proposed definition?	Jassawalla & Sashittal, 1998; Kwak & Anbari, 2009; Detzten et al., 2018; Kozlowski & Bell, 2003; Alexander & Van Knippenberg, 2014 Verbeeten et al., 2018		
5	What is the shared goal of the DevOps team?	Jassawalla & Sashittal, 1998; Kwak & Anbari, 2009; Detzten et al., 2018; Kozlowski & Bell, 2003; Alexander & Van Knippenberg, 2014 Verbeeten et al., 2018		
5a	Which elements substantiate this?			
5b	How was this shared goal defined/created?			
5c	Who has influence on it?			
5d	Do you have examples?			
16	Which of the three sub-aspects (leadership, organizational structure and shared vision and cadre) has the biggest impact on the creation of a shared goal for your team?			
16a	Which sub-aspect contributed the least?			

Table 2.4.App: Leadership

Sub-aspects of Shared Goal: Leadership						
		<i>Input</i>			<i>Output</i>	
<i>Question nr.</i>	<i>Question</i>	<i>Relevant indicator</i>	<i>Sources</i>	<i>Examples</i>	<i>Relevant data (output)</i>	
6	What is, according to IT leadership/management the necessity of collaboration?	Benefit of working together is clear and communicated	Jassawalla & Sashittal, 1998; Cetindamar et al., 2005; Stock et al., 2013; Aronson et al., 2013			
6a	Which elements substantiate this?					
6b	How is the necessity communicated?					
7	What is the access of IT leadership/management to higher management?	Leaders know the environment and senior management	Thamhain, 2009; Cha et al., 2015 Jassawalla & Sashittal, 1998			
7a	Which elements substantiate this?					
8	How is the leadership in and around DevOps teams?					
8a	Who are the leaders and what do they do?					

8b	What is the impact of IT leadership on the shared goal of the team?		Jassawalla & Sashittal, 1998; Vangen & Huxham, 2012; Aronson et al., 2013; Alexander & Van Knippenberg, 2014; Stock et al., 2013		
----	---	--	--	--	--

Table 2.5.App: Organizational Structure

Sub-aspects of Shared Goal: Organizational Structure					
Input				Output	
Question nr.	Question	Relevant indicator	Sources	Examples	Relevant data (output)
9	How strong is the focus of team members on the team?	Individual team members have focus on the team	Tessem & Iden, 2008; Giudice & Condo, 2017; Thamhain, 2009		
9a	How many other tasks do they have?				
9b	How would you describe the impact of other activities?				
10	What is the influence of the team on day-to-day (operational) decisions?	Decentralized day-to-day Decisions	Zábojník, 2002; Gupta et al., 2017; Jassawalla & Sashittal, 1998		
10a	How often can the team decide autonomous?				
10b	How often is the management involved?				
10c	What is the impact of deciding autonomous?				
11	How is the organizational structure around DevOps teams?				
11a	What is the impact of this structure on the shared goal of the team?		Jassawalla & Sashittal, 1998; Thamhain, 2009		
11b	What are examples that have an influence?				

Table 2.6.App: Shared vision and cadre

Sub-aspects of Shared Goal: Shared vision and cadre					
Input				Output	
Question nr.	Question	Relevant indicators	Sources	Examples	Relevant data (output)
12	How are the strategy and the expectations from higher management communicated?	Strategy and expectations of the organization are known	Eldor, 2019; Gutiérrez et al., 2009; Aronson et al., 2013; Lynn & Akgün, 2003; Paris et		

			al., 2000; Jassawalla & Sashittal, 1998		
12a	Who knows the strategy?				
13	How clear are the limitations for the teams?	Opportunities and limitations are known	Eldor, 2019; Hu & Liden, 2011; Gutiérrez et al., 2009		
13a	How is information received?				
14	What does the organization do to help the DevOps team to remain focused on their responsibility?		Eldor, 2019; Aronson et al., 2013;		
14a	How is this focus supported?				
14b	What is the effect of focus on having a shared goal?				
15	What is the vision of the organization?				
15a	How is the vision shared with the DevOps team?		Eldor, 2019; Aronson et al., 2013; Lynn & Akgün, 2003		
15b	What is the impact of knowing the vision on the shared goal of the team?				

Table 2.7.App: Shared understanding

Main aspects of Collaboration: Shared Understanding				
Input			Output	
Question nr.	Question	Sources	Examples	Relevant data (output)
17	What do you think of the proposed definition?	Bittner & Leimeister; 2014; Mohammed & Dumville, 2001; Stout et al., 1999; Aubé et al., 2015; Van den Bossche et al., 2011		
18	How is the degree of shared understanding of the DevOps team?	Bittner & Leimeister; 2014; Mohammed & Dumville, 2001; Stout et al., 1999; Aubé et al., 2015; Van den Bossche et al., 2011		
18a	Which elements substantiate this?			
18b	How was this state of shared understanding created?			
18c	Who has influence on it?			
18d	Do you have examples?			

31	Which of the three sub-aspects (construction, co-construction and constructive conflict) has the biggest impact on the shared understanding of the team?			
31a	Which sub-aspect contributed the least?			

Table 2.8.App: Construction

		Sub-aspects of Shared Understanding: Construction			
		Input		Output	
Question nr.	Question	Relevant indicator	Sources	Examples	Relevant data (output)
19	How self-aware are team members?	Self-aware team members	Berggren et al., 2017		
19a	What substantiates this degree of self-awareness?				
19b	What is a good example to substantiate your claim?				
20	How easy is it to ask questions to each other?	Environment for asking questions	Bittner & Leimeister, 2014; Van den Bossche, 2011		
20a	Why is that?				
20b	What is a good example to substantiate your claim?				
21	How developed is the team's ability to listen to each other?	Team members listen to each other	Bittner & Leimeister, 2014; Mohammed & Dumville, 2001		
21a	What is a good example to substantiate your claim?				
22	How is the "construction" of the DevOps team?		Bittner & Leimeister, 2014; Van den Bossche, 2011		
22a	What is the impact of the construction on the shared understanding?				

Table 2.9.App: Co-construction

		Sub-aspects of Shared Understanding: Co-construction			
		Input		Output	
Question nr.	Question	Relevant indicator	Sources	Examples	Relevant data (output)
23	How does the team reflect and evaluate?	Moments of reflection and evaluation	Bittner & Leimeister, 2014; Mohammed & Dumville, 2001		
23a	What is the impact?				
23b	How often is enough?				
23c	What are examples to substantiate your claim?				

24	How complementary are the team members?	Team members complement each other and have some level of experience	Bittner & Leimeister, 2014; Mohammed & Dumville, 2001; Ohland et al., 2012		
24a	What is the effect on the team?				
25	What are the responsibilities and required competencies for the team?	Responsibilities and skills should be clear	Tessem & Iden, 2008; Hackman & Wageman, 2005		
25a	How is this decided?				
25b	How clear is it for the team?				
26	How is the co-construction of the team?				
26a	What is the impact of the co-construction on the shared understanding?		Bittner & Leimeister, 2014; Van den Bossche, 2011; Mohammed & Dumville, 2001;		

Table 2.10.App: Constructive conflict

Sub-aspects of Shared Understanding: Constructive conflict					
Input			Output		
Question nr.	Question	Relevant indicator	Sources	Examples	Relevant data (output)
27	How safe is it to honest and share opinions?	Psychological safety to share opinions	Bittner & Leimeister, 2014; Kolfshoten et al., 2009		
27a	What does this say about the psychological safety?				
27b	What are examples to substantiate your claim?				
28	How safe is it to ask critical questions and give constructive feedback?	A culture of asking questions	Bittner & Leimeister, 2014; Kleinsmann & Valkenburg, 2008		
28a	What are examples to substantiate your claim?				
29	When do you share information?	Team members that share relevant information	Bittner & Leimeister, 2014; Kolfshoten et al., 2009; Mohammed & Dumville, 2001		
29a	What is the trigger?				
29b	How often does it happen (and is that enough)?				
30	How is the constructive conflict of the team?				

30a	What is the impact of the constructive conflict on the shared understanding?		Van den Bossche et al., 2011; Bittner & Leimeister, 2014; Kolfshoten et al., 2009;		
-----	--	--	--	--	--

2.3 Interview question – Reasoning

This section contains the argumentation and side notes per research question. It helps the researcher to create the structure that is required during the interview. Furthermore, it describes why the questions are relevant.

Introduction & Concluding statement			
Question nr.	Question	Reason to ask the question	Sidenote
1	What is your role within the organisation?	Show the resemblance to the research and guarantee that respondents have the appropriate knowledge and role to contribute to the research	Introduction and corroborating some of the respondent criteria
1a	Welke verantwoordelijkheden horen hierbij?	Building on the previous question and indicating whether their role is similar to the markets interpretation. It also a conversation starter.	
1b	Hoe lang werk je al voor DevOps1?	This is to corroborate the data of the informant. It is important that they work there for a few years. Especially for the managers (see table 3.3.)	
1c	Hoeveel ervaring heb je met DevOps?	This is to corroborate the data of the informant. It is important that they work there for a few years. Especially for the managers (see table 3.3.)	
2	What is your observation regarding the collaboration of cross-functional team members in the DevOps team(s)?	Opening with a broad question gives the respondent the room to show his or her preferences. This can help the interviewer further on. Furthermore, it can corroborate the conceptual model.	Relevant data, which could help to connect the dots between sub-aspects and main-aspects. Should not take too long.
3	What is the impact of the introduction of DevOps on the way team members collaborate?	This question can already tell a lot about the organization and can help to ask better follow-up questions later on.	
3a	What do you think about the definition of collaboration?	It is relevant to ask the respondent about the definition of the research.	
4	What do you think of the proposed definition?	It is relevant to ask the respondent about the definition of the research.	
5	What is the shared goal of the DevOps team?	This is a question that focusses on a main aspect of collaboration, namely shared understanding. It is important to gather information regarding this concept in an attempt to find out whether the respondent recognizes the concept.	When the respondent is a manager this block of questions is the most important (5-16)

5a	Which elements substantiate this?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
5b	How was this shared goal defined/created?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
5c	Who has influence on it?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
5d	Do you have examples?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
6	What is, according to IT leadership/management the necessity of collaboration?	This is a question that focusses on an indicator of IT leadership. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
6a	Which elements substantiate this?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
6b	How is the necessity communicated?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
7	What is the access of IT leadership/management to higher management?	This is a question that focusses on an indicator of IT leadership. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
7a	Which elements substantiate this?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
8	How is the leadership in and around DevOps teams?	This is a question that focusses on a sub-aspect of shared goal, namely leadership. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
8a	Who are the leaders and what do they do?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
8b	What is the impact of IT leadership on the shared goal of the team?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
9	How strong is the focus of team members on the team?	This is a question that focusses on an indicator of organizational structure. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
9a	How many other tasks do they have?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	

9b	How would you describe the impact of other activities?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
10	What is the influence of the team on day-to-day (operational) decisions?	This is a question that focusses on an indicator of organizational structure. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
10a	How often can the team decide autonomous?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
10b	How often is the management involved?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
10c	What is the impact of deciding autonomous?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
11	How is the organizational structure around DevOps teams?	This is a question that focusses on a sub-aspect of shared goal, namely organization structure. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
11a	What is the impact of this structure on the shared goal of the team?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
11b	What are examples that have an influence?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
12	How are the strategy and the expectations from higher management communicated?	This is a question that focusses on an indicator of shared vision and cadre. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
12a	Who knows the strategy?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
13	How clear are the limitations for the teams?	This is a question that focusses on an indicator of shared vision and cadre. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
13a	How is information received?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
14	What does the organization do to help the DevOps team to remain focused on their responsibility?	This is a question that focusses on a part of a sub-aspect of shared goal, namely shared vision and cadre. This part focusses on the cadre. It is apparent to retrieve data about	

		whether the respondents recognize this concept within their organization.	
14a	How is this focus supported?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
14b	What is the effect of focus on having a shared goal?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
15	What is the vision of the organization?	This a question that focusses on a part of a sub-aspect of shared goal, namely shared vision and cadre. This part focusses on the vision. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
15a	How is the vision shared with the DevOps team?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
15b	What is the impact of knowing the vision on the shared goal of the team?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
16	Which of the three sub-aspects (leadership, organizational structure and shared vision and cadre) has the biggest impact on the creation of a shared goal for your team?	This is a question which aims to help the respondent to order the impact of the sub-aspects of shared goal (for the organization of the respondent).	
16a	Which indicator contributed the least?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
17	What do you think of the proposed definition?	It is relevant to ask the respondent about the definition of the research.	When the respondent is a team member (development or operations) this block of questions is the most important (17-30)

18	How is the degree of shared understanding of the DevOps team?	This is a question that focusses on a main aspect of collaboration, namely shared understanding. It is important to gather information regarding this concept to find out whether the respondent recognizes the concept.	
18a	Which elements substantiate this?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
18b	How was this state of shared understanding created?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
18c	Who has influence on it?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
18d	Do you have examples?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
19	How self-aware are team members?	This is a question that focusses on an indicator of construction. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
19a	What substantiates this degree of self-awareness?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
19b	What is a good example to substantiate your claim?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
20	How easy is it to ask questions to each other?	This is a question that focusses on an indicator of construction. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
20a	Why is that?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
20b	What is a good example to substantiate your claim?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
21	How developed is the team's ability to listen to each other?	This is a question that focusses on an indicator of construction. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
21a	What is a good example to substantiate your claim?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	

22	How is the "construction" of the DevOps team?	This is a question that focusses on a sub-aspect of shared understanding, namely construction. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
22a	What is the impact of the construction on the shared understanding?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
23	How does the team reflect and evaluate?	This is a question that focusses on an indicator of co-construction. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
23a	What is the impact?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
23b	How often is enough?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
23c	What are examples to substantiate your claim?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
24	How complementary are the team members?	This is a question that focusses on an indicator of co-construction. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
24a	What is the effect on the team?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
25	What are the responsibilities and required competencies for the team?	This is a question that focusses on an indicator of co-construction. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
25a	How is this decided?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
25b	How clear is it for the team?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
26	How is the co-construction of the team?	This is a question that focusses on a sub-aspect of shared understanding, namely co-construction. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
26a	What is the impact of the co-construction on the shared understanding?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	

27	How safe is it to honest and share opinions?	This is a question that focusses on an indicator of constructive conflict. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
27a	What does this say about the psychological safety?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
27b	What are examples to substantiate your claim?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
28	How safe is it to ask critical questions and give constructive feedback?	This is a question that focusses on an indicator of constructive conflict. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
28a	What are examples to substantiate your claim?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
29	When do you share information?	This is a question that focusses on an indicator of constructive conflict. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
29a	What is the trigger?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
29b	How often does it happen (and is that enough)?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
30	How is the constructive conflict of the team?	This is a question that focusses on a sub-aspect of shared understanding, namely constructive conflict. It is apparent to retrieve data about whether the respondents recognize this concept within their organization.	
30a	What is the impact of the constructive conflict on the shared understanding?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
31	Which of the three sub-aspects (construction, co-construction and constructive conflict) has the biggest impact on the shared understanding of the team?	This is a question which aims to help the respondent to order the impact of the sub-aspects of shared understanding (for the organization of the respondent).	
31a	Which indicator contributed the least?	This is a follow up question that can be used to retrieve more relevant data or to further guide the respondent.	
32	Any last remarks or things you would like to add?	This is a question that should give the respondent the possibility to add	

		something or set some of his remarks in line when he or she feels the need	
--	--	---	--

2.4 Interview protocol – Shared goal

This section presents the interview protocol that has been used for the respondent group: management. The questions will focus on shared goal. The interview protocol is written in Dutch, because that is the native tongue of all respondents.

Interviewschema – Focus op Shared Goal

INTRODUCTIE

Naam respondent:

Datum: ..-..-....

Plaats: 's-Hertogenbosch

Naam interviewer: Timo Nadorp

Duur interview: 60-75 minuten

Aantal acties voor start van het interview:

- *Voorstellen:* Timo Nadorp, Open Universiteit, Opleiding Business Process Management & IT. Beide partijen.
- *Waardering voor medewerking uitspreken*
- *Onderwerp:* DevOps (collaboration)
- *Doel:* Onderzoeken wat de invloed is van shared goal en shared understanding op de DevOps dimensie collaboration.
- *Akkoord:* Tekenen van de Letter of Consent. Akkoord ophalen voor het opnemen van het interview, gebruiken van de interviewresultaten en het quoten van de respondent
- *Anonimiteit:* Gegevens worden uitsluitend gebruikt voor het onderzoek en worden geanonimiseerd
- *Vragen:* Mogelijkheid voor vragen bieden m.b.t. de procedure van het interview voor het begin van het interview

INTERVIEW

Vraag 1: Wat is jouw rol binnen de organisatie?

- Welke verantwoordelijkheden horen hierbij?
- Hoe lang werk je al voor DevOps1?
- Hoeveel ervaring heb je met DevOps?

Toelichting drie wetenschappelijke begrippen:

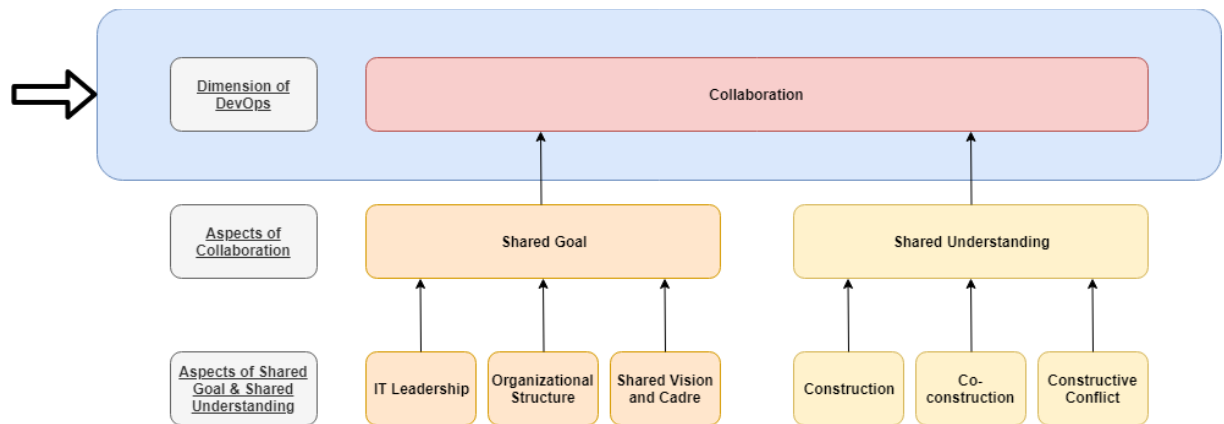
1. Collaboration
2. Shared goal
3. Shared understanding

Toelichten opdeling vragen naar thema's/factoren.

Fase 1: Dimensie Collaboration

Definitie van shared understanding en korte omschrijving:

DevOps bestaat volgens de theorie uit verschillende dimensies, zoals automation en collaboration. In deze studie is collaboration in een DevOps context gedefinieerd als; cross-functionele teamleden die benodigde activiteiten uitvoeren in het belang van een duidelijk doel.



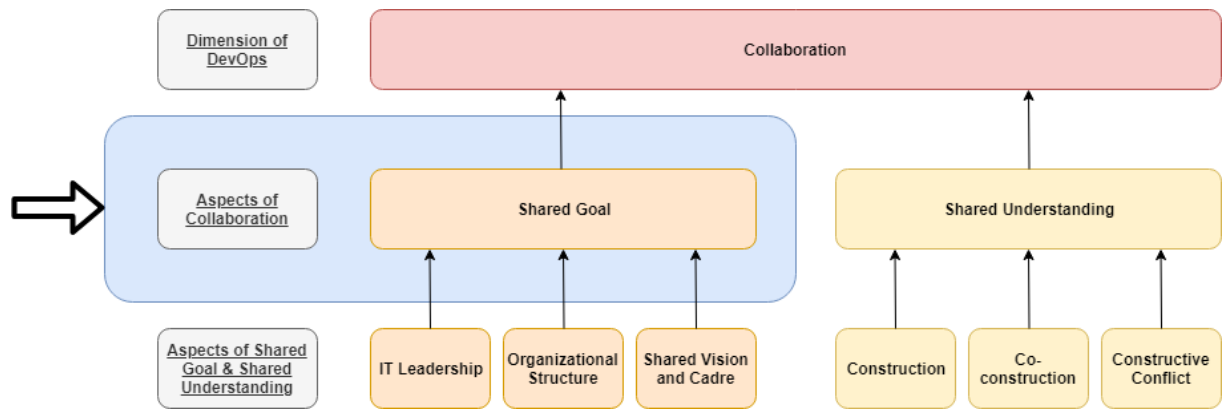
Vraag 2: Wat vind jij van de samenwerking van de cross-functionele teamleden in het DevOps team?

Vraag 3: Wat is de invloed van DevOps geweest op de manier waarop er samengewerkt wordt? Wat vind je van de beschreven definitie?

Fase 2: Hoofdaspect Shared Goal

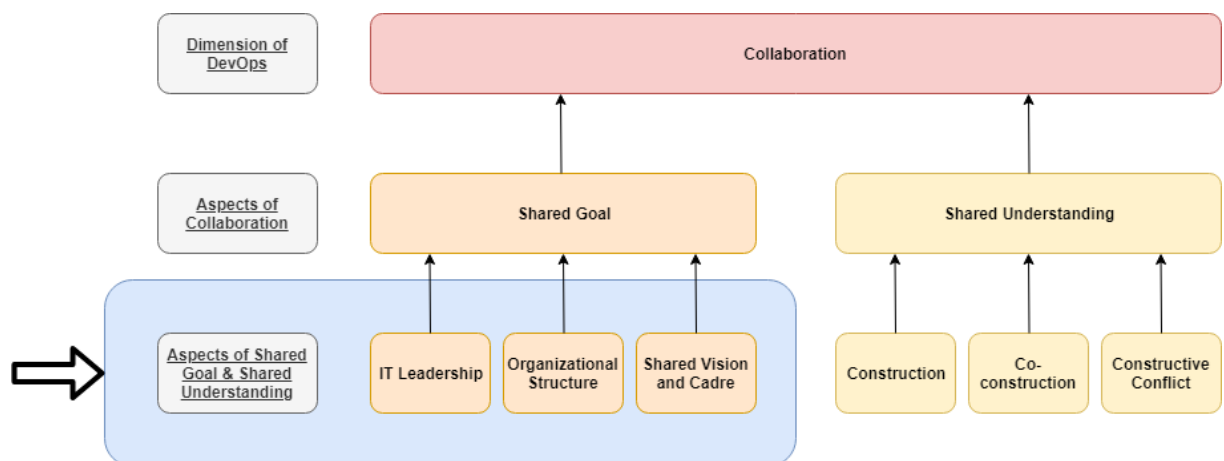
Definitie van shared goal en korte omschrijving:

Collaboration bestaat volgens de theorie uit verschillende aspecten, zoals shared goal en shared understanding. In deze studie is shared goal gedefinieerd als; een organisatorisch instrument dat focus kan creëren door een doel of verantwoordelijkheid te delegeren aan het gehele team. In het tweede plaatje van dit thema vind je de aspecten van shared goal. Dit zijn; IT Leadership, organizational structure en shared vision and cadre.



Vraag 4: Wat vind je van de beschreven definitie voor shared goal?

Vraag 5: Wat is het gezamenlijke doel van het DevOps team? (Waar blijkt dit uit? Hoe wordt het doel bepaald? Wie heeft daar invloed op? Voorbeelden?)



Shared Goal – IT Leadership

Vraag 6: Wat is volgens de IT leider(s)/management de noodzaak/voordeel van samenwerken? Hoe is/wordt dit gecommuniceerd? Waar blijkt dat uit?

Vraag 7: Wat is voor de IT leider(s)/management de toegang tot hoger management? Waar blijkt dat uit?

Vraag 8: Hoe is het leiderschap in en rondom DevOps teams? (Wat is de impact van het IT-leiderschap op het shared goal? Ondersteunt het een team bij het focussen op een shared goal?)

Shared Goal – Organizational structure

Vraag 9: Wat is de focus van de teamleden? Hoeveel andere werkzaamheden hebben de teamleden

Vraag 10: Wat is de invloed van het team op day-to-day beslissing? Hoe vaak is het management hierbij betrokken?

Vraag 11: Hoe is de organisatorische structuur rondom DevOps teams? (Wat is de impact van organisatorische structuur op het shared goal? Ondersteunt het een team bij het focussen op een shared goal?)

Shared Goal – Shared Vision and Cadre

Vraag 12: Hoe wordt de strategie en de verwachting van de organisatie gedeeld? Wie kent de strategie allemaal?

Vraag 13: Wat weet het DevOps team van de limitatie die het team heeft (de kaders). Hoe komen ze aan die informatie?

Vraag 14: Hoe wordt het DevOps geholpen om gefocust en zonder afleiding te opereren? (Hoe wordt dat gedaan? Ondersteunt dit een team bij het focussen op een shared goal?)

Vraag 15: Wat is de visie van het bedrijf? (Productvisie / bedrijfsvisie). Hoe wordt deze gedeeld met het DevOps teams? (Ondersteunt het een team bij het focussen op een shared goal?)

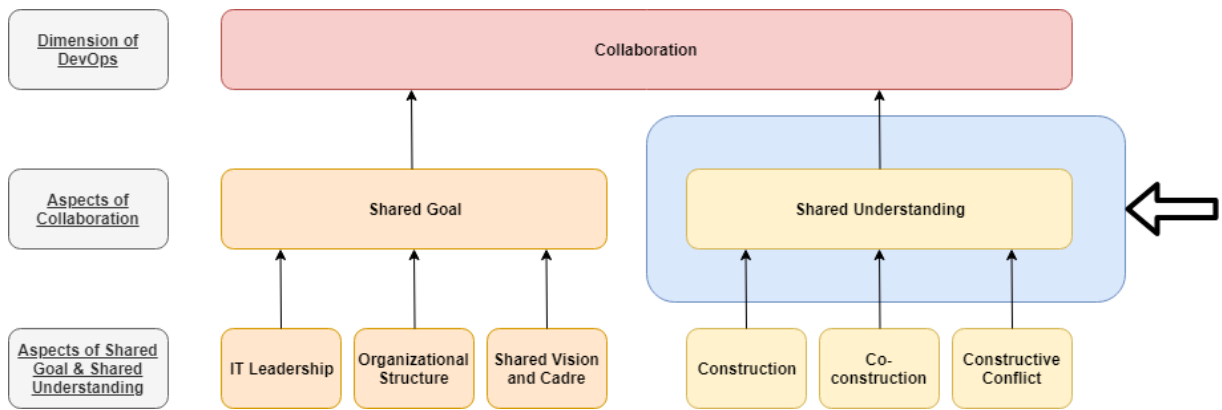
Vraag 16: Welke van de drie sub-aspecten van shared goal heeft het meeste bijgedragen aan het creëren van een gezamenlijk doel? (Welke het minst? Wat zijn andere indicatoren die jij ziet?)

- a) IT Leadership
- b) Organizational structure
- c) Shared vision and cadre

Fase 3: Hoofdaspect Shared Understanding

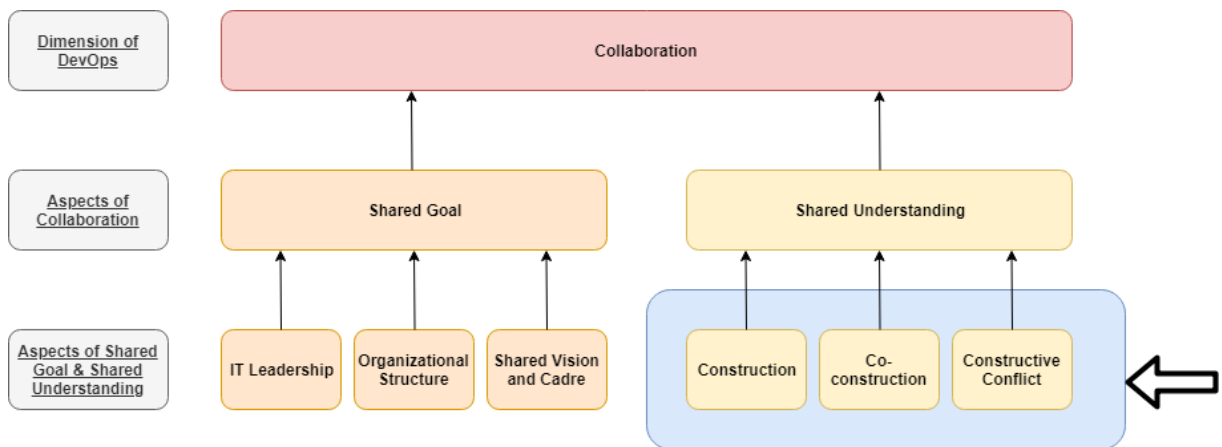
Definitie van shared understanding en korte omschrijving:

Collaboration bestaat volgens de theorie uit verschillende aspecten, zoals shared goal en shared understanding. In deze studie is shared understanding gedefinieerd als; een concept, beïnvloedt door het team, dat opheldering en alignment kan creëren met betrekking tot benodigde activiteiten, verantwoordelijkheden en context. In het tweede plaatje van dit thema vind je de aspecten van shared understanding. Dit zijn; construction, co-construction en constructive conflict.



Vraag 17: Wat vind je van de genoemde definitie voor shared understanding?

Vraag 18: Wat is het gezamenlijke beeld van het DevOps team? (Waar blijkt dit uit? Hoe komt dit tot stand? Wie heeft daar invloed op? Voorbeelden?)



Uitleg van construction, co-construction en constructive conflict

Shared Understanding – Construction

Vraag 22: Hoe is het construction van het team? (Wat heeft dit voor een invloed op het gezamenlijke beeld van het team? Heb je een voorbeeld?)

Shared Understanding – Co-construction

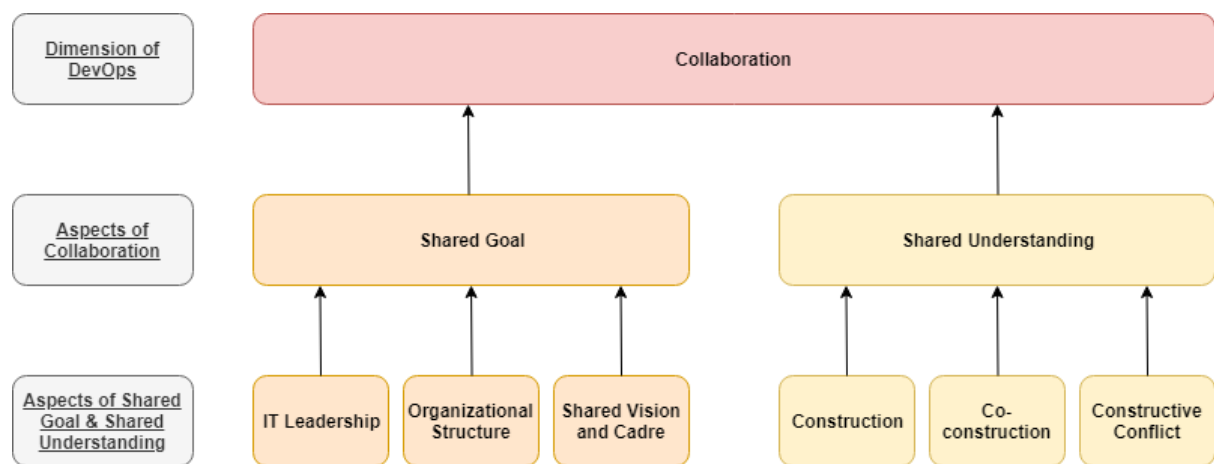
Vraag 26: Hoe is het co-construction van het team? (Wat heeft dit voor een invloed op het gezamenlijke beeld van het team? Heb je een voorbeeld?)

Shared Understanding – Constructive Conflict

Vraag 30: Hoe is het constructive conflict van het team? (Wat heeft dit voor een invloed op het gezamenlijke beeld van het team? Heb je een voorbeeld?)

Vraag 31: Welke van de drie sub-aspecten van shared understanding heeft het meeste bijgedragen aan het creëren van een gezamenlijk begrip/beeld? (Welke het minst? Wat zijn andere indicatoren die jij ziet?)

- a) Construction
- b) Co-construction
- c) Constructive conflict



AFRONDING

Vraag 32: Zijn er nog zaken die wat jou betreft onbelicht zijn gebleven?

Bedanken voor de tijd en bijdrage

2.5 Interview protocol – Shared understanding

This section presents the interview protocol that has been used for the respondent group: team members. The questions will focus on shared understanding. The interview protocol is written in Dutch, because that is the native tongue of all respondents.

Interviewschema – Focus op Shared Understanding

INTRODUCTIE

Naam respondent:

Datum: ..-..-....

Plaats: 's-Hertogenbosch

Naam interviewer: Timo Nadorp

Duur interview: 60-75 minuten

Aantal acties voor start van het interview:

- *Voorstellen:* Timo Nadorp, Open Universiteit, Opleiding Business Process Management & IT. Beide partijen.
- Waardering voor medewerking uitspreken
- *Onderwerp:* DevOps (collaboration)
- *Doel:* Onderzoeken wat de invloed is van shared goal en shared understanding op de DevOps dimensie collaboration.
- *Akkoord:* Akkoord ophalen voor het opnemen van het interview, gebruiken van de interviewresultaten en het quoten van de respondent
- *Anonimiteit:* Gegevens worden uitsluitend gebruikt voor het onderzoek en worden geanonimiseerd
- *Vragen:* Mogelijkheid voor vragen bieden voor het begin van het interview

INTERVIEW

Vraag 1: Wat is jouw rol binnen de organisatie?

- Welke verantwoordelijkheden horen hierbij?
- Hoe lang werk je al voor DevOps1?
- Hoeveel ervaring heb je met DevOps?

Toelichting drie wetenschappelijke begrippen:

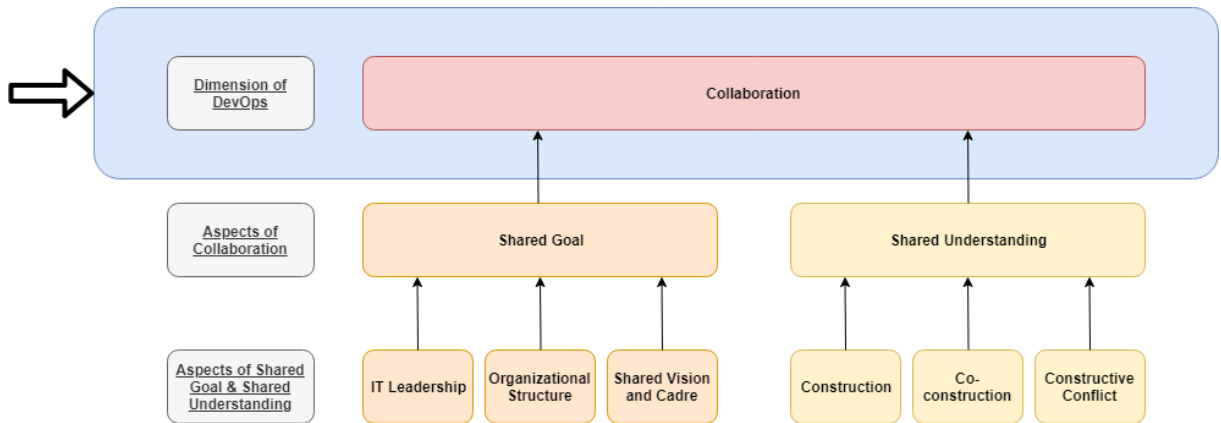
1. Collaboration
2. Shared goal
3. Shared understanding

Toelichten opdeling vragen naar thema's/factoren.

Fase 1: Dimensie Collaboration

Definitie van shared understanding en korte omschrijving:

DevOps bestaat volgens de theorie uit verschillende dimensies, zoals automation en collaboration. In deze studie is collaboration in een DevOps context gedefinieerd als; cross-functionele teamleden die benodigde activiteiten uitvoeren in het belang van een duidelijk doel.



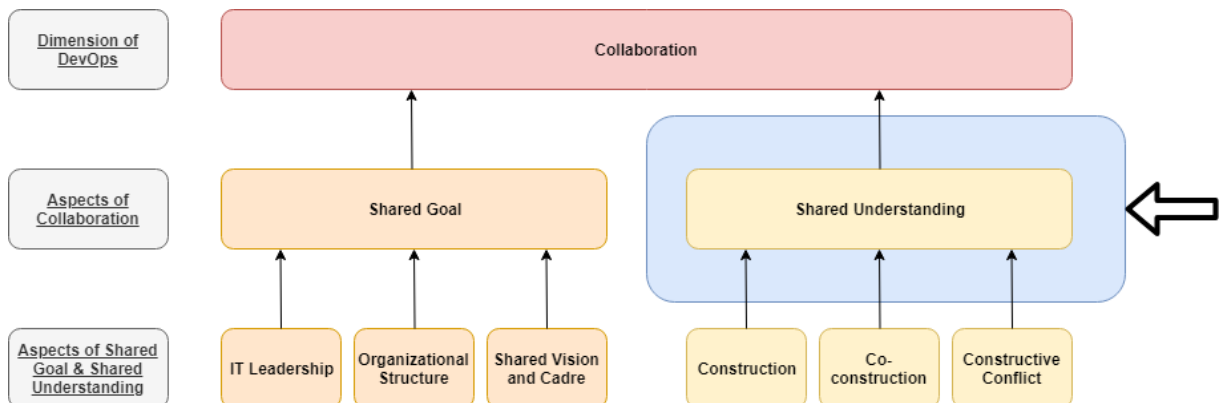
Vraag 2: Wat vind jij van de samenwerking van de cross-functionele teamleden in het DevOps team?

Vraag 3: Wat is de invloed van DevOps geweest op de manier waarop er samengewerkt wordt?

Fase 2: Hoofdaspect Shared Understanding

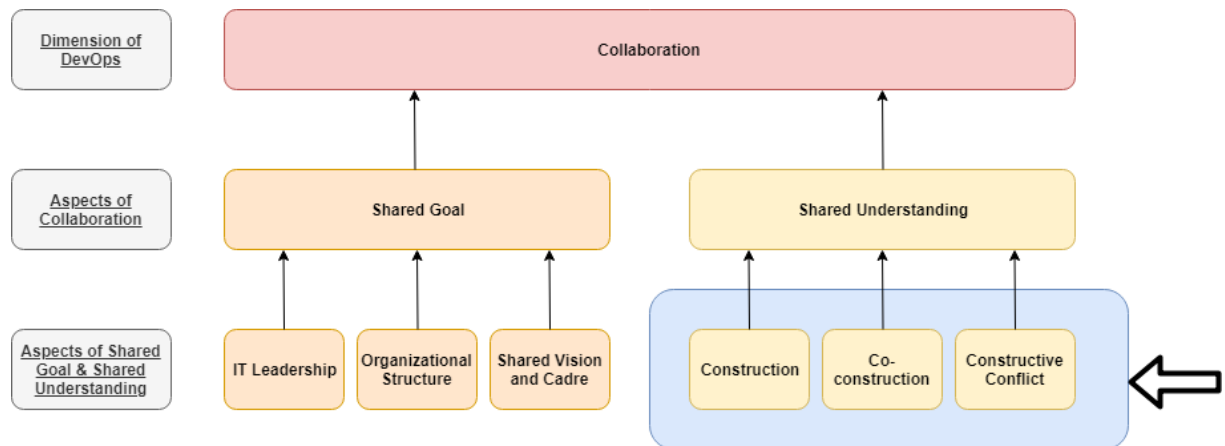
Omschrijving:

Collaboration bestaat volgens de theorie uit verschillende aspecten, zoals shared goal en shared understanding. In deze studie is shared understanding gedefinieerd als; een concept, beïnvloedt door het team, dat opheldering en alignment kan creëren met betrekking tot benodigde activiteiten, verantwoordelijkheden en context. In het tweede plaatje van dit thema vind je de aspecten van shared understanding. Dit zijn; construction, co-construction en constructive conflict.



Vraag 17: Wat vind je van de genoemde definitie voor shared understanding?

Vraag 18: Wat is het gezamenlijke beeld van het DevOps team? (Waar blijkt dit uit? Hoe komt dit tot stand? Wie heeft daar invloed op? Voorbeelden?)



Uitleg van construction, co-construction en constructive conflict

Shared Understanding – Construction

Vraag 19: Hoe self-aware zijn de teamleden? Waar blijkt dat uit?

Vraag 20: Hoe gemakkelijk stellen teamleden vragen aan elkaar?

Vraag 21: Hoe wordt er onderling geluisterd?

Vraag 22: Hoe is de construction van het team?

Shared Understanding – Co-construction

<Introductie van de term>

Vraag 23: Wat is jouw beeld van de momenten van evaluatie? Hoe doen jullie dit? Is dit vaak genoeg? Wat is het effect?

Vraag 24: Hoe complementair zijn teamleden?

Vraag 25: Wat zijn de benodigde competenties en verantwoordelijkheden om als team te slagen. Hoe wordt dit bepaald? Hoe duidelijk is dit voor het team?

Vraag 26: Hoe is de co-construction van het team?

Shared Understanding – Constructive Conflict

<Introductie van de term>

Vraag 27: Hoe veilig is het om je mening te delen? Wat zegt dit over de psychologische veiligheid?

Vraag 28: Hoe geaccepteerd is het om kritische vragen te stellen? Wat zegt dit over de team cultuur?

Vraag 29: Wanneer delen jullie informatie? Hoe ontstaat dat? Gebeurt dit voldoende? Waaruit blijkt dat?

Vraag 30: Hoe is het constructive conflict van het team?

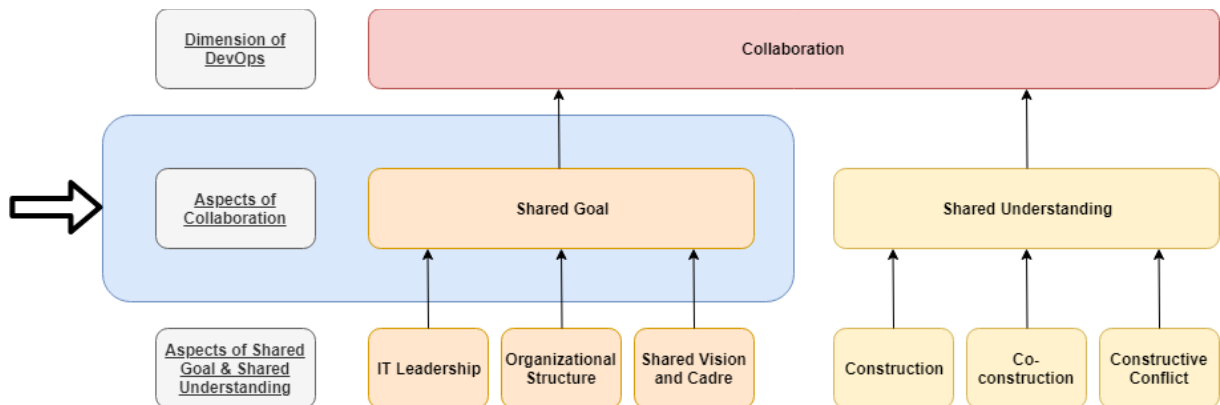
Vraag 31: Welke van de drie sub-aspecten van shared understanding heeft het meeste bijgedragen aan het creëren van een gezamenlijk begrip/beeld? (Welke het minst? Wat zijn andere indicatoren die jij ziet?)

- a) Construction
- b) Co-construction
- c) Constructive conflict

Fase 3: Hoofdaspect Shared Goal

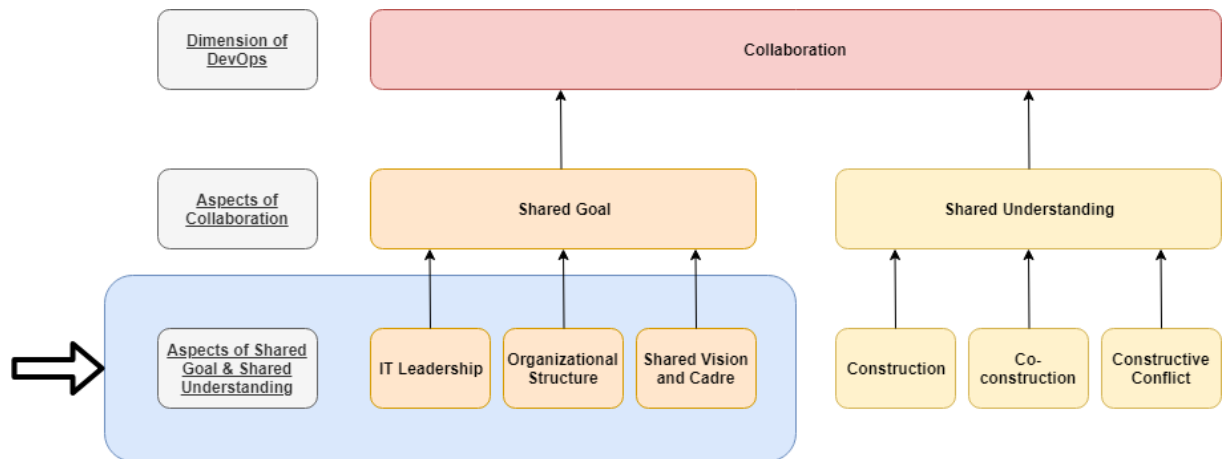
Definitie van shared goal en korte omschrijving:

Collaboration bestaat volgens de theorie uit verschillende aspecten, zoals shared goal en shared understanding. In deze studie is shared goal gedefinieerd als; een organisatorisch instrument dat focus kan creëren door een doel of verantwoordelijkheid te delegeren aan het gehele team. In het tweede plaatje van dit thema vind je de aspecten van shared goal. Dit zijn; IT Leadership, organizational structure en shared vision and cadre.



Vraag 4: Wat vind je van de genoemde definitie voor shared goal?

Vraag 5: Wat is het gezamenlijke doel van het DevOps team? (Waar blijkt dit uit? Hoe wordt het doel bepaald? Wie heeft daar invloed op? Voorbeelden?)



Uitleg van IT leadership, organizational structure en shared vision and cadre

Shared Goal – IT Leadership

Vraag 8: Hoe is het leiderschap in en rondom DevOps teams? (Wat is de impact van het IT-leiderschap op het shared goal? Ondersteunt het een team bij het focussen op een shared goal?)

Shared Goal – Organizational structure

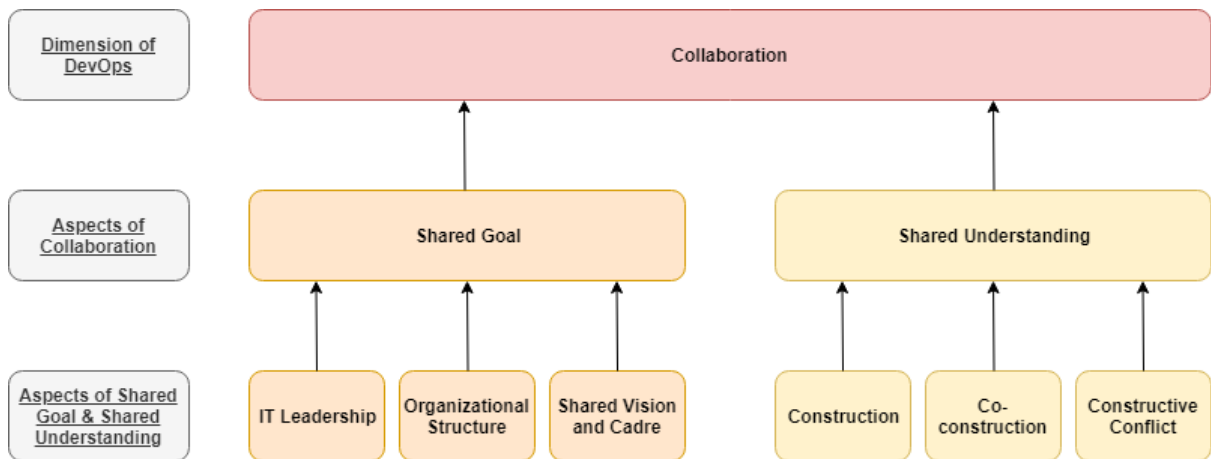
Vraag 11: Hoe is de organisatorische structuur rondom DevOps teams? (Wat is de impact van organisatorische structuur op het shared goal? Ondersteunt het een team bij het focussen op een shared goal?)

Shared Goal – Shared Vision and Cadre

Vraag 15: Wat is de visie van het bedrijf? (Productvisie / bedrijfsvisie). Hoe wordt deze gedeeld met het DevOps teams? (Wie kent de strategie? Ondersteunt het een team bij het focussen op een shared goal?)

Vraag 16: Welke van de drie sub-aspecten van shared goal heeft het meeste bijgedragen aan het creëren van een gezamenlijk doel? (Welke het minste? Wat zijn andere indicatoren die jij ziet?)

- a) IT Leadership
- b) Organizational structure
- c) Shared vision and cadre



AFRONDING

Vraag 32: Zijn er nog zaken die wat jou betreft onbelicht zijn gebleven?

Bedanken voor de tijd en bijdrage

2.6 Letter of consent

This paragraph contains the letter of consent that has been sent to the respondents to explain the research method and the way the data will be used. The goal is to inform the respondents upfront, so they know how the information they share is used. The letter is written in Dutch, because that is the native tongue of all respondents.

Beste <naam respondent>,

Mijn naam is Timo Nadorp en ik ben student aan de Open Universiteit. Hier volg ik de master-opleiding Business Process Management & IT, met de specialisatie Data Science Management. Mijn afstudeeronderzoek richt zich op DevOps. Door jouw kennis en ervaring met of in een DevOps team heeft <naam informant> jou aangedragen als een van de respondenten. Er is inmiddels een afspraak (datum, tijdstip en locatie) bekend waarop het face-to-face interview zal plaatsvinden. Toch wil ik via deze weg benadrukken dat het u vrij staat om niet mee te doen aan dit onderzoek. Het is op vrijwillige basis.

Het interview zal vragen bevatten over drie kernbegrippen; collaboration, shared goal en shared understanding. Er zullen ook twee soorten interviews zijn. Een die zich meer richt op collaboration en shared goal en een die zich meer richt op collaboration en shared understanding. De indeling is gebaseerd op de rol van de respondent. Het interview wordt opgenomen en duurt ongeveer 1 uur. Tijdens het interview heeft u de optie om een vraag niet te beantwoorden. Meer informatie over het interview protocol vindt u in de bijlage. Hier vindt u het type interview dat past bij uw rol.

Het interview wordt opgenomen en getranscribeerd. De analyse zal uitgevoerd worden om resultaten te extraheren. De informatie wordt anoniem verwerkt en quotes zullen zo gebruikt worden dat deze niet herleidbaar zijn tot u, een collega of de organisatie. De opnames worden opgeslagen op een locatie waar alleen ik toegang toe heb.

Na afloop van het interview wordt de uitwerking ter goedkeuring aangeboden. De opname zal alleen toegankelijk zijn voor de interviewer en verstrekte informatie die gebruikt wordt voor het onderzoek wordt anoniem verwerkt. In het geval dat het onderzoek gepubliceerd wordt, dan zal naast de onderzoeker ook een review commissie inzage krijgen m.b.t. de identiteit van de respondenten. Deze commissie mag deze informatie nooit openbaar maken of delen met derden. De geanonimiseerde informatie kan wel gebruikt worden voor publicatie.

Ten slotte is het belangrijk om te weten dat u zich als respondent elk moment kunt terugtrekken. Het onderzoek wordt immers in mutuele overeenstemming afgenomen. Mochten er vragen of opmerkingen zijn, dan hoor ik dat graag.

Door te tekenen gaat u akkoord met bovenstaande informatie.

Met vriendelijke groet,

Handtekening + datum interviewer

Handtekening + datum geïnterviewde

Timo Nadorp

(telefoonnummer <interviewer>)

(emailadres <interviewer>)

Bijlage

- Interviewschema

2.7 Introduction Letter

This paragraph contains the letter that has been sent to the respondents to explain the research subject. The goal is to inform the respondents upfront, so they know better what to expect. The letter is written in Dutch, because that is the native tongue of all respondents.

Beste <naam respondent>,

Mijn naam is Timo Nadorp en ik ben student aan de Open Universiteit. Hier volg ik de master-opleiding Business Process Management & IT, met de specialisatie Data Science Management. Mijn afstudeeronderzoek richt zich op DevOps. Het onderzoek is in het Engels en daarom zal ik tijdens het interview waar nodig Engelse termen gebruiken, aangezien niet alles een op een te vertalen is.

DevOps is een relatief onontgonnen onderzoeksonderwerp. Het doel van de studie is dan ook om een aantal kernbegrippen beter te onderzoeken. Een belangrijke dimensie van DevOps is samenwerking (collaboration). Er zijn volgens de theorie een aantal hoofdaspecten die van invloed zijn op de dimensie collaboration. De studie richt zich op de exploratie van collaboration en zijn hoofdaspecten in de DevOps context.

Er is een conceptueel model opgesteld, aan de hand van wetenschappelijke literatuur. Dit model is een voorlopig antwoord op de onderzoeksvraag. Dit is een kwalitatief onderzoek en het model wordt getoetst door middel van de interviews. De interviews worden geïnterpreteerd en geanalyseerd. Elementen van jouw antwoord kunnen gebruikt worden om een deel van het conceptueel model te substantiëren, dan wel tegen te spreken. De resultaten worden verwerkt in de scriptie, alvorens een conclusie geschreven wordt.

Het interview zal vragen bevatten over drie kernbegrippen; collaboration, shared goal en shared understanding. Er zullen ook twee soorten interviews zijn. Een die zich meer richt op collaboration en shared goal en een die zich meer richt op collaboration en shared understanding. De indeling is gebaseerd op de rol van de respondent.

Voordat het interview begint worden met u een letter of consent en een interviewschema gedeeld. De letter of consent verklaard aan u hoe er met de informatie wordt omgegaan en welke rechten en plichten horen bij de verwerking van de resultaten. Indien u deze brief ondertekent met uw handtekening gaat u akkoord met de voorwaarden. Het interviewschema is om u de mogelijkheid te geven zich voor te bereiden op het interview. U hoeft geen specifieke acties te ondernemen.

Mochten er vragen ontstaan dan hoor ik dat graag.

Met vriendelijke groet,

Timo Nadorp
(telefoonnummer <interviewer>)
(emailadres <interviewer>)

Appendix 3: Coding

The relevant data, retrieved from the semi-structured interviews, is depicted in appendix 4.1, 4.2 and 4.3. In this section of the appendix the coded fragments of the indicators and sub-aspects are depicted. Paragraph 3.1 portraits the indicators and paragraph 3.2 will present the sub-aspects. The questions that focus on the main-aspect are already focused on the main-aspect level, therefore there is no table to show the integration from indicator or sub-aspect to the corresponding main-aspect.

3.1 Coding – Indicators

This paragraph represents the table 3.1.app. It summarizes the coding steps for the questions that relate to the indicators. The first two columns show which respondent and which question from the semi-structured interview are documented. After that column 3 to 5 show the coding steps. The last column explains the reasoning that a respondent has. In some cases that reasoning is not strong enough to build up to a selective category. If that happens the cell is filled with: ‘-’.

Table 3.1.App: coding for the indicator questions

Respondent	Question	Open	Axial	Selective	Reasoning Respondent
<i>Respondent1</i>					
	Interview question 6	Benefit of working together is clear and communicated	IT Leadership	Shared Goal	Communication and clarity are necessary to create guidance. The team understood why we formed the team and what was expected.
	Interview question 7	Leaders know the environment and senior management	IT Leadership	-	Leaders are available, open organization.
	Interview question 10	Decentralized day-to-day Decisions	Organizational Structure	Shared Goal	Leaders should not interfere in daily decisions. Team is responsible for the solution. Decisions that have an unwanted effect will

					result in feedback.
	Interview question 9	Individual team members have focus on the team	Organizational Structure	Shared Goal	The team is 100% dedicated. Sometimes we ask too much flexibility though. We try to protect the team more.
	Interview question 12	Strategy and expectations of the organization are known	Shared vision and cadre	Shared Goal	The team understands the vision. I perceive it like that. They believe in the new path we took and it helps them to help the customers.
	Interview question 13	Opportunities and limitations are known	Shared vision and cadre	Shared Goal	A team needs a cadre to be effective and exercise their responsibilities. The manager is responsible for that. Within the cadre lie the opportunities for the team.
<i>Respondent2</i>					
	Interview question 6	Benefit of working together is clear and communicated	IT Leadership	Shared Goal	Transition team: reflect on the difference between the envisioned effect of the transition and the present. In order to grow you need to communicate that.
	Interview question 7	Leaders know the	IT Leadership	Shared Goal	Created an availability

		environment and senior management			path: developer > architect > lead architect. When necessary we are there.
	Interview question 10	Decentralized day-to-day Decisions	Organizational Structure	Shared Goal	The only centralized themes are education and HR. The rest is for the team and the management is involved as often as the team sees fit.
	Interview question 9	Individual team members have focus on the team	Organizational Structure	Shared Goal	Everyone, except architects, are dedicated in one team. As leadership we try to help the team to bind a long-lasting balance.
	Interview question 12	Strategy and expectations of the organization are known	Shared vision and cadre	Shared Goal	The vision is unit-specific, but the platform is not. We try to be transparent if it is clear what we want. The unit discussions prevent clarity. You want to be clear about what you will be in a few years.
	Interview question 13	Opportunities and limitations are known	Shared vision and cadre	Shared Goal	Everything is possible. If you want to try stuff. Be smart about it, ask a question when you are not sure and try it isolated. Show

					the results. If it doesn't work, fail fast!
<i>Respondent3</i>					
	Interview question 6	Benefit of working together is clear and communicated	IT Leadership	Shared Goal	Communicate reason to team and all stakeholders. Teams need room to excel. Sharing a dream helps.
	Interview question 7	Leaders know the environment and senior management	IT Leadership	Shared Goal	Senior management needs to adapt servant leadership. Should not slow down the team.
	Interview question 10	Decentralized day-to-day Decisions	Organizational Structure	Shared Goal	They decide almost everything. PO is responsible and I am an escalation path. If I am involved it is often due to a blockade: a hot topic or a relational issue. We are more focused on for instance; the right climate to work.
	Interview question 9	Individual team members have focus on the team	Organizational Structure	Shared Goal	Team is above all. We allocate an employee fully to a team. It is possible that will not be the case in a sprint, but the team has to agree on that.
	Interview question 12	Strategy and expectations of the	Shared vision and cadre	Shared Goal	Sometimes the vision is decided behind closed doors.

		organization are known			We should align the bottom up and top down perspectives. It could give more clarity of what you do and do not do.
	Interview question 13	Opportunities and limitations are known	Shared vision and cadre	Shared Goal	We have a just do it mentality. Cadres should be clear, but not written in stone. You want to use their creative skills.
<i>Respondent4</i>					
	Interview question 19	Self-aware team members	Construction	-	During the retrospectives, the team members of Team 69 show that they are self-aware. Sensitive things are often addressed.
	Interview question 20	Environment for asking questions	Construction	Shared Understanding	Nowadays that is definitely the case. That grew since 2016. People will ask for the help they need and therefore will easier get to the required level.
	Interview question 21	Team members listen to each other	Construction	Shared Understanding	They hear each other, sometimes I wonder if they listened. It happens that people do not speak the same language.

	Interview question 23	Moments of reflection and evaluation		Shared Understanding	We are not very good in being honest when something goes wrong. Furthermore, we should also give feedback outside the retro.
	Interview question 24	Team members complement each other and have some level of experience	Co-construction	Shared Understanding	Both teams are well balanced in terms of skills and personalities. They are all mature enough to be effective in the team.
	Interview question 25	Responsibilities and skills should be clear	Co-construction	Shared Understanding	This is somethings that the team knows best. A good example is when we hire a new person. The team needs to be able to describe what they need. It is important to be aware of the gaps. If they feel the responsibility, they will take it.
	Interview question 27	Psychological safety to share opinions	Constructive conflict	Shared Understanding	I think it is safe, although I hope so. We have an experienced population who knows what is necessary to be successful at work.
	Interview question 28	A culture of asking questions	Constructive conflict	Shared Understanding	It can really help to be critical to each

					other. Especially across different roles. In our situation that is definitely the case.
	Interview question 29	Team members that share relevant information	Constructive conflict	Shared Understanding	The cadre, relevant information, is very interesting. So, you need to decide what is relevant for whom. If you share everything you create chaos. It is pretty hard to find the balance between transparency and not creating turmoil.
<i>Respondent5</i>					
	Interview question 19	Self-aware team members	Construction	Shared Understanding	Our team has a lot of humor. However, if you cross a line it will be discussed. That honesty made us more self-aware and helps us as a team.
	Interview question 20	Environment for asking questions	Construction	Shared Understanding	Always! That is not a problem, it is a pre-requisite. We are critical and try help each other.
	Interview question 21	Team members listen to each other	Construction	Shared Understanding	Some team members have a strong opinion. They will stand for

					that opinion and can be pretty stubborn. Others are more like followers. That can result in frustration, because these team members are not always heard.
	Interview question 23	Moments of reflection and evaluation	Co-construction	Shared Understanding	We could focus more on the personal aspect. However, I think we should always give feedback. Then we can fix it immediately.
	Interview question 24	Team members complement each other and have some level of experience	Co-construction	Shared Understanding	Based on the differences we are able to retrieve a lot of information & perspectives and have an informed discussion. We also know which styles a team member prefers.
	Interview question 25	Responsibilities and skills should be clear	Co-construction	Shared Understanding	In general, we have a good grip on that. Especially on the Ops-area we have a gap. There is not a lot of interest in that, so I would say we are more capable in Dev, then in Ops.

	Interview question 27	Psychological safety to share opinions	Constructive conflict	Shared Understanding	We have a playful team, but also a team that is transparent. We share what is going on in our private life and everything is open for discussion.
	Interview question 28	A culture of asking questions	Constructive conflict	Shared Understanding	We have many stubborn team members, but at the end you have created a better image of the problem and the possible solutions. That helps in the decision making.
	Interview question 29	Team members that share relevant information	Constructive conflict	Shared Understanding	Lately, we had several meetings to address certain elements of the application or to create a shared understanding about a subject.
<i>Respondent6</i>					
	Interview question 19	Self-aware team members	Construction	Shared Understanding	Within the team we are aware of our strengths and weaknesses and are willing to share that. I have some habits that can be annoying. If it is within boundaries it is acceptable and we will make it work.

	Interview question 20	Environment for asking questions	Construction	Shared Understanding	Content questions are asked a lot, other questions could be asked more. That could help us as a team to grow closer to each other.
	Interview question 21	Team members listen to each other	Construction		Some are more stubborn than others, but we appreciate the help we give each other.
	Interview question 23	Moments of reflection and evaluation	Co-construction	Shared Understanding	We could be more direct to each other. More like we are in business, instead of friends. In the end you will get better by being direct and open.
	Interview question 24	Team members complement each other and have some level of experience	Co-construction	Shared Understanding	We have a diversified group of individuals. I think that everybody brings in some value. Some are focused on details. Another is a real go-getter. That creates a complete team.
	Interview question 25	Responsibilities and skills should be clear	Co-construction	Shared Understanding	If the attitude and intellect is okay, then it all comes naturally is my experience. If a new person joins the team, we help him. The team

					culture is in place. You will not be left alone, but we get you up to speed.
	Interview question 27	Psychological safety to share opinions	Constructive conflict	Shared Understanding	I think we could be more aware about whether it is psychologically safe for all us and what we need for that. The awareness would be a real improvement.
	Interview question 28	A culture of asking questions	Constructive conflict	Shared Understanding	During code reviews and refinements there are many questions. I like to take it a step further, so; this is good, but this would make it better. A review is not a tool to criticize, it is a way to help each other forward. During retrospectives, with elements like personal relations, it can be slower.
	Interview question 29	Team members that share relevant information	Constructive conflict	Shared Understanding	Most team members test whether the information they received is correct. So, in that way there is a lot of sharing and discussing of information. But often the

					important details are neglected, often due to a lack of oversight.
<i>Respondent 7</i>					
	Interview question 19	Self-aware team members	Construction	Shared Understanding	In general team members are self-aware. There are a few quirks that somebody can have, that might not always be obvious to himself. That is why we have feedback sessions.
	Interview question 20	Environment for asking questions	Construction	Shared Understanding	If we share something within the team, it is up to that person to share it with their environment. This safe space results in a place where we can ask a lot.
	Interview question 21	Team members listen to each other	Construction	Shared Understanding	That depends, there are often conflicts about content. As a team it is ok to disagree. We try to come to an agreement and otherwise let it rest for a while.
	Interview question 23	Moments of reflection and evaluation	Co-construction	Shared Understanding	It is important that you can give each other feedback and that the one who gets feedback is

					willing to listen. It cannot be like pointing fingers and you do this and that and stop it. That does not work.
	Interview question 24	Team members complement each other and have some level of experience	Co-construction	Shared Understanding	You need the right balance in a team. That is also why all these theories are designed on how to build a team. We have different types in our team and that works.
	Interview question 25	Responsibilities and skills should be clear	Co-construction	Shared Understanding	Everybody brings something to the table, with knowledge and/or ideas that they have. If you have a team with only Brainiac's, that will not work. Of course, you might get perfect code, but it might not be maintainable.
	Interview question 27	Psychological safety to share opinions	Constructive conflict	Shared Understanding	A while back we had a conflict that was heated. At a certain moment, emotions started to rise. You should not stop at that moment, but continue. Talk about it. What is going on?

					Those moments are natural in a team I think, it should not be all sunshine. That is almost impossible.
	Interview question 28	A culture of asking questions	Constructive conflict	Shared Understanding	I think it is something we do well as a team. If something is going on, also on the personal level, talk about it. Ask about it if you see it.
	Interview question 29	Team members that share relevant information	Constructive conflict	Shared Understanding	We used knowledge sessions to get people up to speed regarding certain areas of the things we develop. That is a very useful tool for sharing knowledge.

3.2 Coding – Sub-aspects

This paragraph represents a table 3.2.app. It summarizes the coding steps for the questions that relate to the sub-aspects. The first two columns show which respondent and which question from the semi-structured interview are documented. After that column 3 and 4 show the coding steps. The last column explains the reasoning that a respondent has. In some cases that reasoning is not strong enough to build up to a selective category. If that happens the cell is filled with: ‘-’.

Respondent	Question	Axial	Selective	Reasoning Respondent
<i>Respondent1</i>				
	Interview question 8	IT Leadership	Shared Goal	Due to feedback from a team member, the customer or me there is growth. It is often about showing them the impact of what they do.
	Interview question 11	Organizational Structure	Shared Goal	We have interchangeable developers, although everybody has its specialism. That has grown the last year.
	Interview question 15	Shared vision and cadre	Shared Goal	The team understands that it is about more than technology. That is very important.
	Interview question 22	Construction	-	For the team process it is important to talk about values and opinions.
	Interview question 26	Co-construction	Shared Understanding	The dynamic in the team is very nice. Everybody sees it is a real team. They have a way of helping each other.
	Interview question 30	Constructive conflict	Shared Understanding	There is a feedback culture. In the beginning that can be exciting. The good thing of this team is that they hold each other responsible and show respect for each other.
<i>Respondent2</i>				
	Interview question 8	IT Leadership	Shared Goal	A leader who leads by example. Do not blame others, start moving. You can spot a good team easily. When there is a problem they start working harder, together.
	Interview question 11	Organizational Structure	Shared Goal	We have an organization without a lot of hierarchy. Team has three go to roles: team lead, PO and architects and can escalate to me and Respondent3.
	Interview question 15	Shared vision and cadre	Shared Goal	Our business unit should be owner of the platform vision.

				Sadly, we are not and creates chaos.
	Interview question 22	Construction	Shared Understanding	We stimulate people to share. You need to work for a long time. Make sure you know what you want and that others know that.
	Interview question 26	Co-construction	Shared Understanding	Be open, be transparent. That is what we ask of our people. When the skills are present that is often what it takes.
	Interview question 30	Constructive conflict	Shared Understanding	People are important. And people will have conflicts. It depends on the team whether the team leads get involved. However, it helps to grow.
<i>Respondent3</i>				
	Interview question 8	IT Leadership	Shared Goal	IT leadership is important in enabling teams and helping them to overcome blockades and become autonomous.
	Interview question 11	Organizational Structure	Shared Goal	Me and Respondent2 are moving more to a coaching role. The responsibility lies as low as possible. So team level or PO/Architect/Team lead level.
	Interview question 15	Shared vision and cadre	Shared Goal	I should be able to answer whether we will do something by pointing to clear cadres regarding our vision. That's not the case and has a negative impact.
	Interview question 22	Construction	Shared Understanding	Everybody has a good insight in what the other team members want to achieve in life. That process started from being self-aware.
	Interview question 26	Co-construction	Shared Understanding	Again, we can support. By facilitating training in personalities, strengths and weaknesses etc. The teams pick that up.
	Interview question 30	Constructive conflict	Shared Understanding	Without conflict you cannot do DevOps. It should be with respect though. Within the teams this goes pretty well, between the teams and the leadership this could be better.
<i>Respondent4</i>				

	Interview question 8	IT Leadership	Shared Goal	Internally there can be a lot of discussion, especially on unit level. We try to keep that out of the teams. Furthermore, leadership looks whether the role fits the team member.
	Interview question 11	Organizational Structure	Shared Goal	We are doing good due to do the mature teams and the way we organized it. Teams are free to decide a lot.
	Interview question 15	Shared vision and cadre	Shared Goal	The overarching vision is unclear, due to discussions between units. That makes it hard to be transparent.
	Interview question 22	Construction	Shared Understanding	Collaboration has grown a lot. That starts with yourself, as an individual. Who are you and what do you want to achieve? We also ask way more questions.
	Interview question 26	Co-construction	Shared Understanding	The team can do a lot. First, they needed a Scrum master to facilitate their work process and personal connection. Now they do it themselves.
	Interview question 30	Constructive conflict	Shared Understanding	We might avoid conflicts too much. It can be good to be honest to each other.
<i>Respondent5</i>				
	Interview question 8	IT Leadership	Shared Goal	How can we keep it a nice environment for stubborn developers? That what they try to achieve. They also think about how they can make people come up with their own ideas. That really improved this year. I am satisfied with that.
	Interview question 11	Organizational Structure	Shared Goal	We have a flat organization. Especially within the team: there is no order there. Around the team there are several structures. If we disagree with an outcome, they will know it!
	Interview question 15	Shared vision and cadre	Shared Goal	The product vision is blurry. There are many parties who try to have influence on that vision.
	Interview question 22	Construction	Shared Understanding	We had some sessions about what gives you energy. That helps in becoming self-aware.

				Furthermore we are very open for questions.
	Interview question 26	Co-construction	Shared Understanding	We know what we have got and what we can achieve together. And more importantly how we can achieve that.
	Interview question 30	Constructive conflict	Shared Understanding	We are very honest with each other. Hard on the content level, but not on the personal level.
<i>Respondent6</i>				
	Interview question 8	IT Leadership	Shared Goal	I think we have a leadership team, with high technical capacities. We feel support when needed and we do not have to convince them of obvious things. They also expect that we take the responsibility. If it is not good (enough), they will let you know.
	Interview question 11	Organizational Structure	-	Sometimes people talk too much about things that will never happen. I always find it difficult to decide what you need around the teams.
	Interview question 15	Shared vision and cadre	Shared Goal	Sometimes teams do not receive all the information. That can be annoying, because we have to gather that information ourselves. We will also challenge decisions.
	Interview question 22	Construction	Shared Understanding	Being self-aware and sharing the inner thoughts helps in understanding others. Furthermore, we really want to help each other as a team.
	Interview question 26	Co-construction	Shared Understanding	Although we had some personnel changes, I see it improving again. We could be more open about what we really think and feel.
	Interview question 30	Constructive conflict	Shared Understanding	This could be a big improvement point for our team. We would become better individuals and a better team if we did this more often.
<i>Respondent7</i>				

	Interview question 8	IT Leadership	Shared Goal	The leadership thinks about the cadres and they also form a block against stuff that will slow us down. Also, our product owner, whom I see as a leader, helps us a lot in that regard.
	Interview question 11	Organizational Structure	Shared Goal	The structures I see are PO+Architects, who decide upon epics and team leads. That helps us and sometimes leads to frustration.
	Interview question 15	Shared vision and cadre	-	The most important thing for me is that I understand the product vision and that my ideas are heard.
	Interview question 22	Construction	Shared Understanding	At the moment I'm in a coaching process. Others see that in my behavior. That helps me. Secondly, we are real colleagues. Not only work.
	Interview question 26	Co-construction	Shared Understanding	Within the team we try to create team days. Within the work environment we are willing to listen to each other.
	Interview question 30	Constructive conflict	Shared Understanding	On the content level we are very critical. We could do it more on the personal level, but do know how to handle those discussions. Respect for the individual is the most important thing.

Appendix 4: Interviews & content analysis

This section of the appendix presents the data that has been retrieved via the semi-structured interviews and the content analysis. It provides background information for the statements that are made in chapter 4.

4.1 Results - Indicators

The relevant information per indicator is depicted in this section of Appendix 4. The information is portrayed per indicator. Paragraph 4.1.1 presents the relevant data per indicator. The second paragraph portrays the score per respondent, presented in table 4.1. While the last paragraph, 4.1.3, presents a more detailed description of how these scores were emitted.

4.1.1 Relevant information per indicator

In this section we summarize the information per indicator. The more detailed information can be found in appendix 4.1.2 and 4.1.3. These appendixes will also show the information that was used to give the scores Y(es), A(mbiguous), N(o).

Benefit of working together is clear and communicated

All respondents answered convincingly. They communicate and communicated a lot with the teams to create alignment. The benefit of working together is clear and influences shared understanding.

Leaders know the environment and senior management

The scores vary for this indicator. Respondent3 thinks that there is too much hierarchy. The business unit layer creates a lot of friction, which is holding the teams back. Respondent1 disagrees and thinks they are an open organization. Respondent2 has an ambiguous score, he explains that the availability of the leadership team is not as high as he would like.

Decentralized day-to-day Decisions

All respondents claim that the team decides regarding day-to-day decisions. They are responsible for the solution and have to decide how they will meet the requirements, notes Respondent1. They also decide when they seek guidance outside the team. Respondent2 explains that the only elements that are centralized are: HR and education. The leadership, instead, will spend a portion of their time on making sure that there is a good work environment. Respondent3: 'If you are tired, please go home. We trust you have a good reason for it and come back tomorrow'.

Individual team members have focus on the team

All respondents state this is the case. Respondent3 states: 'Team is above all. Team members should also correct each other when they are not behaving like that.' Respondent2 agrees and states that make the team responsible for their own organization. The things that Respondent2 focuses on is

how is the team performing and is there someone who is slowing the team down. Respondent1 states that their team members are also fully focused on the team. However, the stakeholders and organization demand a lot of flexibility of the team. They appointed a manager that functions as a barrier between the stakeholders and the team. This helps them to focus more on the team, but it could be improved.

Strategy and expectations of the organization are known

None of the respondents clearly state that this done in a good manner. According to Respondent2 it is hard to align the vision across business units. The visions are often unit specific, but the platform is not. Respondent3 adds that the decision making is often behind closed doors. He states that this does not help with the adoption. Respondent2 emphasizes that before they share the strategy with the team there often first is a political game.

Opportunities and limitations are known

The respondents score the same on this indicator. Respondent2 and Respondent3 state that: 'everything is possible' and 'we have a just do it mentality'. Respondent3: 'You don't want to control people on input, but on expectations and outcome'. Respondent1 states that it is the job of the manager to make sure the opportunities and limitations are known and respected.

Self-aware team members

The scores regarding this indicator vary. Respondent 4 and 7 are the most optimistic. Respondent4 states that they have retrospectives and that he thinks there is self-awareness in those meetings. Respondent7 tells about a meeting where they had an in-depth feedback session. He says there were no big surprises for anybody and sees that as an argument to state that the team members are self-aware. Respondent5 and Respondent6 are a little bit more reserved. Respondent5 finds it a hard indicator to judge. He thinks that their team is sober and their culture is without a lot of boundaries. Therefore, team members can say a lot to each other, which made them all more self-aware. Respondent6 thinks it varies and things can happen, often due to emotions or lack of experience. They could improve here according to Respondent6.

Environment for asking questions

All respondents state that they have an environment for asking questions. Especially questions regarding content. Respondent5 says that asking is a prerequisite for being part of the team. Respondent7 states that the important thing for him is that he can trust his team members and is able to ask and share what he wants.

Team members listen to each other

This indicator shows several answers. Respondent4: 'They hear each other, but sometimes I wonder if they listened'. It happens regularly that do not speak the same language or do not want to

understand each other. Respondent5 agrees and states that it also depends on the individual. Some team members like to lead and decide by nature. Respondent6 thinks it is on an acceptable level. He thinks that the willingness to ask questions and listen to one another is the reason why the helpfulness so high.

Moments of reflection and evaluation

All the respondents are aware of the importance of reflection and evaluation. Only Respondent7 would describe this as well enough. Respondent4 and Respondent6 feel that the team could share more honestly and should not only do that in a retrospective. Respondent5 contributes that they are able to that, but should reflect more on the personal level.

Team members complement each other and have some level of experience

All the respondents are very positive regarding this indicator. Respondent4 states that both teams have a nice balance, also in terms of personalities. Respondent6 find truth in these words and thinks that everybody in his team brings in some value. Different type of personalities that make the team complete. Respondent5 emphasizes the decision process of his team. 'At the end we can decide better, based on all the input that is delivered from different viewpoints. That is very positive for us as a team'. Respondent5 and Respondent7 both think there is a good mix in their team based on personalities.

Responsibilities and skills should be clear

The respondents agree that they perform well on this indicator. 'This is somethings that the teams know best', according to Respondent4. Respondent6 states that the prerequisite for new employees is that they have the right attitude and level of thinking. If that is on par, then the team will help them to get up to speed. Respondent4 and Respondent5 state that the team members have intrinsic motivation to perform well and make a good qualitative product. Respondent5 and Respondent7 talk about the different roles that individuals take. Respondent7: 'Everybody brings something to the table, with knowledge and/or ideas that they have. If you have a team with only Brainiac's, that will not work. Of course, you might get perfect code, but it might not be maintainable'. Respondent5: 'There is not a lot of interest Ops, so I would say we are more capable in Dev.

Psychological safety to share opinions

The scores vary regarding this indicator. Respondent4: 'we have an experienced population'. He thinks that helps. 'Experienced professional people know what it takes to be successful in the work environment'. Respondent6 states: 'I think we could be more aware about whether it's psychologically safe for all us and what we need for that. The awareness would be a real improvement'. Respondent7 remembers a discussion that took some time. He explained that did not stop and argues that this shows that there is psychological safety. 'We don't make it personal'.

A culture of asking questions

All respondents score high on this indicator. Respondent4: 'It can really help to be critical to each other. Especially across different roles'. During the interview with Respondent5 this was one of the main topics. 'We do this a lot, especially regarding the content. We have many stubborn team members, but at the end you have created a better image of the problem and the possible solutions. That helps in the decision making'.

Team members that share relevant information

The scores vary for this indicator. Respondent4: 'Team members share information. But I feel like you can also create a fuzz if you share too much information'. Respondent6 states that the team members gather and check information. Furthermore, he adds: 'But often the important details are neglected, often due to a lack of oversight'. Due to the complex environment it is hard to know all the relevant knowledge and, therefore, also hard to share all the relevant knowledge. Respondent5 explains that his team mostly share knowledge on the spot. Lately they also have had meetings to discuss certain topics and share the knowledge in order to get a shared understanding. Respondent7 also mentions these meetings: knowledge sessions. 'That is a very useful tool'.

4.1.2 Score per respondent

This paragraph shows and explains the scores per respondent.

Benefit of working together is clear and communicated

Respondent1 explains that they communicated a lot during the transition towards a DevOps team. Before the team was formed, all individuals worked at the client site. It was important to communicate why they wanted to do this. It was a big change to go from individuals who needed to be billable to a team that needs to deliver recurring revenue. It helped that the team agreed with the decision. Based on these quotes and background information the score for Table 4.1 was: "y" for Respondent1.

While Respondent1 had this transition one year ago, Respondent 2 and 3 did this more than three years ago with teams. They both sketch a situation where there was mismanagement, not enough room for development teams and low performance (unacceptable bugs, bad releases etc.). In the beginning of 2016, there were 5-7 people who came into their organization, amongst them were respondent 2 and 3. They professionalized the way teams work and created an IT leadership block between stakeholders and teams.

During that period, a dream was shared; we want to be able to release at every moment and have a stable product. That was quite a shock for the employees who created their own culture. That has an impact, not only positive. Respondent2 notes that it felt like an invasion for them, because the people there thought it was going great. The impact of failures for customers. Respondent2 and Respondent3 helped people to say what was going wrong. Not only what is going wrong? Also, why, and who is responsible and why do not you tell him when it happens. A new way of working was introduced, which was build up and communicated. As an employee you had two options; work with the change or leave. Respondent3 tells that he wanted a culture were the teams goes to the stakeholders with new functionality. Instead of the stakeholders, the team should be in the lead. Give the team the responsibility. This also means, the team needs to take responsibility. That is something you have to guard as leaders, according to Respondent3. They make the team the owner of a whole solution, including questions and implementation. The reasoning behind that was

discussed in depth, including a dream, therefore the scores of Respondent2 and Respondent3 for Table 4.1 are scored as: “y”.

Leaders know the environment and senior management

Respondent1 states that it's an open organization where people feel free to connect and close together, both physically and mentally. Therefore, the score for Table 4.1 is a: “y”. Respondent 2 and 3 are more critical. Respondent2 explains that he has a lot of responsibilities and could be more available for the team. He states that there are two architects who are close to the team and that they are close enough to him. Based on these statements of Respondent2 the score for Table 4.1 is a: “a”. Respondent3 thinks that the organization could be less hierarchic. ‘The team knows what's going on it's often the layers above the team that prevent them from excelling at the highest level’. He also thinks that product owners are thinking too hierarchic and states that it should be the other way around. Leadership should ask the team, as servant leaders, if there is more that they need to excel. So, there is access, but people could take more responsibility. That is something that the organization could improve. That demands something from the leadership team. Based on these remarks it is clear that Respondent3 thinks there is a lot to improve, the score for Table 4.1: “n”.

Decentralized day-to-day Decisions

All respondents claim that the team decides regarding day-to-day decisions. They are responsible for the solution and should decide how they will meet the requirements, notes Respondent1. Based on that the score for Table 4.1: “y”. They also decide when they seek guidance outside the team. Respondent2 explains that the only elements that are centralized are: HR and education. These things are decided by the BLT (team of team leads). Respondent3 explains he is only an escalation option. Most of the execution is with team members and some of the functional responsibilities lie with the PO and technical responsibilities with the architects. They set cadres and within that the team can decide. He explains that there sometimes are hot topics or blockades and then one of the leaders will try to use stakeholder management to solve it. It is our challenge to not solve it, like you instinctively want, and is often expected from you. Another thing that we keep deciding is the balance between working hard and rest. We try to measure on outcome and not on input. Respondent2: ‘People who have a hard time in private life will have more room, they can for instance come in late’. Respondent3: ‘If you are tired, please go home. We trust you have a good reason for it and come back tomorrow’. Based on all these statements of Respondent2 and Respondent3 the score for Table 4.1 is: “y”.

Individual team members have focus on the team

As stated before, the department of Respondent1 made the transition later than the department of Respondent2 and Respondent3. Respondent3 states: ‘Team is above all. Team members should also correct each other when they are not behaving like that.’ Respondent2 agrees and states that make the team responsible for their own organization. ‘Everyone is hardcore in a Scrum team’. The things that he is focusing on is how is the team performing and is there someone who is slowing the team down. He states that one underperformer slows the team down with 25% and that he wants to prevent that effect. The latest step that they took in that regard is giving the team the responsibility to write the job application and find the new team member. So, team members in Core, were

Respondent2 and Respondent3 lead the development organization, have a full focus and responsibility for the team. This all leads to the conclusion that the score of Table 4.1 for Respondent2 and Respondent3 is: “y”. Respondent1 states that their team members are also fully focused on the team. However, there are still a lot of stakeholders who expect something from the team. The organization demands a lot of flexibility of the team. The unit Finance, were Respondent1 leads the development organization, tried to professionalize this. They appointed a manager that functions as a barrier between the stakeholder and the team. This helps them to focus more on the team, but it could be improved. The statements regarding Respondent1 lead to the score of “y” for Table 4.1.

Strategy and expectations of the organization are known

None of the respondents clearly state that this done in a good manner. They all seem to see room for improvement. Respondent1 is the most positive, but he mostly talks about the vision of how the team should work. He states that they know what is expected and believe in the vision of helping the customer. Due to this statement the score for the table 4.1: “a”. The data lacks more concrete information to be a “y”, but he clearly is not all negative towards the indicator. Respondent2 states that the communication towards teams is transparent. The problem lies in the vision across business units. The visions are often unit specific, but the platform that is used by all unit is not. The business unit core is owner of the platform, so they often have discussion with internal and external customers. Respondent3 adds that the decision making is often behind closed doors. He states that this does not help with the adoption. The strategical layer and the executive layer are not always in agreement about what is the best thing to do. Respondent3 calls that a mismatch. Respondent2 and Respondent3 do tell their employees what they can. Respondent3 has several tools, like a 3-month plan that he shares every quarter with the teams and customers. He also has several ambitions and dreams that he shares with the team, to create guidance. Where do we want to be in a few years? Respondent2 emphasizes that before they share the strategy with the team there often first is a political game. Based on the statements of Respondent2 and Respondent3 the indicator scores a “n” for Table 4.1.

Opportunities and limitations are known

Respondent2 and Respondent3 state that: ‘everything is possible’ and ‘we have a just do it mentality’. If needed they will give feedback and if the team has doubts; ask a question. Instruments like; epics, technological roadmap etc. are in place. Respondent2: ‘if you want to try stuff. Feel free. Try it isolated. Show me because I am interested. And if goes wrong, fail fast’! Respondent3: ‘You don’t want to control people on input, but on expectations and outcome. Cadres for that should be clear, but it also important that they are not so strict that they can’t evolve’. Respondent1 states that it is the job of the manager to make sure the opportunities and limitations are known and respected. Leadership decide the limitation and the team can refine them during evaluations. All these statements show that the respondents considered ways to maintain this indicator, therefore the scores for Table 4.1 are all: “y”.

Self-aware team members

The answer regarding this indicator vary. Respondent 4 and 7 are the most optimistic, they score a “y” in Table 4.1. Respondent4 states that they have retrospectives and that he thinks there is self-awareness in these meetings. Respondent7 tells about a meeting where they had an in-depth feedback session. He says there were no big surprises for anybody and sees that as an argument to state that the team members are self-aware. Respondent5 and Respondent6 are a little bit more reserved, resulting in a score of “a” for Respondent5 and “n” for Respondent6. Respondent5 finds it a hard indicator to judge. He thinks that their team is sober, and their culture is without a lot of boundaries. Therefore, team members can say a lot to each other, which made them all more self-aware. Respondent6 thinks it varies and things can happen, often due to emotions or lack of experience. He talks about his trigger point: ‘If somebody made a bad solution and it took a long time; I will start complaining’. He states that it is important to know these kinds of things from yourself and your team members. They could improve here according to Respondent6.

Environment for asking questions

All respondents state that they have an environment for asking questions, therefore they all scored a “y” in Table 4.1. Especially questions regarding content. Respondent5 says that asking is a prerequisite for being part of the team. Respondent4: makes the comparison with 2016 when 6 out of 7 people were new. At that time, it was not that normal, but now it is. He states that nowadays the culture is strong enough to handle a few changes in the team. Respondent6 likes the helpfulness of his team. They do what is necessary to help the other. Respondent says that you can become the center of ridicule when you ask something. Sometimes the other disagree and make jokes. The important thing for him is that he can trust his team members and is able to ask and share what he wants.

Team members listen to each other

Respondent4: ‘They hear each other, but sometimes I wonder if they listened’. It happens regularly that do not speak the same language or do not want to understand each other. Based on these statements the score for Respondent4 depicted in Table 4.1. is “n”. Respondent5 agrees and states that it also depends on the individual. Some team members like to lead and decide by nature. ‘They will stand for that opinion and can be pretty stubborn. Sometimes that can result in frustration’. He is not as negative as Respondent4. However, there seems to be enough room for improvement, resulting in the score: “a” in Table 4.1. Respondent6 thinks it is on an acceptable level. He thinks that the willingness to ask questions and listen to one another is the reason why the helpfulness so high. That resulted in the score; “y” in Table 4.1. Respondent7 shares the following insight: ‘If one person has a fundamentally other idea than someone else, then it’s natural that it creates conflict. Those are often about content and that is fine. Then we discuss and dive into the options. Lately we had a story and there were two people with very different ideas on how to approach it, which would decide more things than just that story. It took a few weeks to agree. What we do is; we discuss it within the limit of the meeting, and if we do not agree, we park the discussion. We will discuss it in the next meeting, but leave the conflict for that meeting’. It shows the ability to listen to each other, also if team members disagree. This is scored as “a” in Table 4.1, because it shows that the team has the ability to listen. However, there seems to be room for improvement since it is focused only on the content level of understanding each other.

Moments of reflection and evaluation

All the respondents are aware of the importance of reflection and evaluation. However, there seems room for improvement. Respondent4 states that they are quite good in giving and receiving feedback when the team is in a good spirit and things go right. They also have fine retrospectives, with attention towards the work process and team development. 'However, when things go wrong it becomes harder. The same goes for compliments. It is also noteworthy that these things are often said in a retro, but almost never outside of it. Especially when it is not about content. We are not that good in stating that somebody is underperforming. Lastly, we had a situation with somebody who was not performing due to personal circumstances. This has been resolved but I do not know if the team would react better when a similar situation arises'. Respondent6, who attends the same meetings, thinks that team members are too sweet for one another. 'I am a direct guy, but I am often holding back. We could be more direct to each other. More like we are in business, instead of friends'. He states that the balance between work and private life that is a hard. It can also be dangerous to know each other privately. 'I am able to have discussion with respondent 2 or 4, but at the end you know, it is okay. We know where we are together, both on the work-aspect and on the private-aspect'. He likes that people get irritated and show passion for something. 'Could be more in our team. To regain that culture, we need coaching in my opinion, and maybe an external coach that creates and discusses conflicts. We could learn a lot from that. I don't know if it's necessary, but it definitely a point for improvement'. Based on these statements the scores for Respondent4 and Respondent6 depicted in Table 4.1. are both: "n".

Respondent5 explains that they always use the same format during a retrospective, normally this format tends to focus on the content. Lately they had a feedback session that was focused on team development and personal relations. That had a positive impact according to Respondent5 and Respondent7. They decided to do this more often. Respondent5 and Respondent7 both stated that they think is important that these moments are not limited to the meeting. It should be something that addressed when it occurs and something that is saved for the meeting. They say they have a culture wherein that is possible, but both seem to think that more focus on the team development and personal relations could improve their team. Based on that notion Respondent5 is scored with a "a" in Table 4.1. Respondent7 adds that it is important that the one who receives feedback is the one who decides whether he or she will do something will it. He, and the team, also believe in positive feedback: 'Sometimes we do that during retro's, then we have bromance letters. If somebody helped you with something, you get a bromance card'. Based on this addition Respondent7 is scored as "y" in Table 4.1.

Team members complement each other and have some level of experience

Respondent4 states that both teams have a nice balance, also in terms of personalities. 'Everybody has his or her main focus or things they find important. That diversification creates a nice balance'. Respondent6 find truth in these words and thinks that everybody in his team brings in some value. Different type of personalities that make the team complete. Respondent5 emphasizes the decision process of his team. 'We are with six persons, so there will exists islands regarding certain matters. At the end we can decide better, based on all the input that is delivered from different viewpoints. That is very positive for us as a team. In that way the team members really complement each other, are able to gather a lot of information and have a good discussion'. Respondent5 and Respondent7 both think there is a good mix in their team based on personalities. They do not think in the colors; red, blue, yellow, green, but they do see that some are more leaders and some more followers.

Respondent7: 'I can be more modest during refinements or retrospectives. I listen a lot. There are also a lot of people who talk a lot and share many ideas. That is a dynamic that works'. Respondent7 also states the team is busy with making sure there are no single point failures, which were abundant in the past. Based on all these remarks all respondents are scored with a "y" in Table 4.1.

Responsibilities and skills should be clear

'This is somethings that the teams know best', according to Respondent4. Respondent6 states that employees we just joined are still learning what is expected. The prerequisite is that they have the right attitude and level of thinking. If that is on the right level, then the team will help them to get up to speed. Respondent4 says: 'a good example is when we hire a new person'. The team has to write the job application. 'At first these applications were very demanding. So, we could not find that person. Nowadays teams say give us someone with the right attitude and we can educate them internally'. Respondent4 and Respondent5 state that the team members have intrinsic motivation to perform well and make a good qualitative product.

Respondent5 and Respondent7 talk about the different roles that individuals take. Some are more interested in tackling that hard, abstract problems and other like to automate. In their team the balance might be little bit off. Respondent5: 'There is not a lot of interest in that and because that interest isn't you can see that we don't do much in that area. So, I would say we are more capable in Dev, then in Ops'. Respondent7: 'Everybody brings something to the table, with knowledge and/or ideas that they have. If you have a team with only Brainiac's, that will not work. Of course, you might get perfect code, but it might not be maintainable'. He also states that it is important to prevent single point of failures. That has been something they focused on the last years. Based on all these remarks all respondents are scored with a "y" in Table 4.1.

Psychological safety to share opinions

Respondent4: 'we have an experienced population'. He thinks that helps. 'Experienced professional people know what it takes to be successful in the work environment'. Therefore Respondent 4 is scored as "y" in Table 4.1. Respondent6 also seems to share the opinion that experience helps in that regard. He also states that there is not a lot of attention for it. 'I think we could be more aware about whether it's psychologically safe for all us and what we need for that. The awareness would be a real improvement'. Based on these statements Respondent6 is scored with a "a" in Table 4.1. He agrees that the professionalism of the workforce helps, but also clearly sees room for improvement.

Respondent5 and Respondent7 both stated several times that it is very much possible to be the center of ridicule. If you say or ask something that is considered "stupid", they will have a culture to make fun of each other. However, they both say it is a positive element and it is always in a playful way. They both think it is not something that creates a blockade for sharing opinions or feelings. However, based on that argument alone the information of Respondent5 is scored as "a" in Table 4.1. Respondent7 states: 'A while back we had a conflict that slumbered for a while. At a certain moment, emotions started to rise between a few team members. You should not stop at that moment, but continue. Talk about it. What is going on? Most of the time there is an underlying issue that needs to be addressed'. This approach helped them to solve the issue. 'Those moments are natural in a team I think, it shouldn't be all sunshine. That is almost impossible. It also creates a certain sharpness. So, you need both, calmness and conflict'. Based on this addition Respondent7 is scored as "y" in Table 4.1.

A culture of asking questions

Respondent4: 'It can really help to be critical to each other. Especially across different roles. As a developer you can be so deep into the content that you don't have the overview anymore'. He feels like everybody can feel and feels free to ask these questions. Although he also states that it is important to stay constructive. He, black and white as he can be, finds that hard sometimes.

Respondent6: 'We ask a lot of critical questions'. For instance, during code reviews. 'A review is not a tool to criticize, it's a way to help each other forward and to make sure that we did the right thing. That is a mentality which is not always there. Especially new people feel like they are getting told what they did wrong, but that's not why we do it'.

During the interview with Respondent5 this was one of the main topics. 'This we do a lot, especially about the content. We have many stubborn team members, but at the end you have created a better image of the problem and the possible solutions. That helps in the decision making'.

Respondent7 adds: 'Ask about it if you see it. We also challenge each other when behavior is not ok or when we need to challenge each other on content'. Based on all these remarks all respondents are scored with a "y" in Table 4.1.

Team members that share relevant information

Respondent4: 'Team members share information. But I feel like you can also create a fuzz if you share too much information. So, the cadre, relevant information, is very interesting. So, you need to decide what is relevant for whom'. He thinks the team and the leadership around teams shares knowledge when it is necessary. 'It is pretty hard to find the balance between transparency and not creating turmoil'. Respondent6 states that the team members gather and check information.

Furthermore, he adds: 'But often the important details are neglected, often due to a lack of oversight'. Due to the complex environment it is hard to know all the relevant knowledge and, therefore, also hard to share all the relevant knowledge. Based on these statements Respondent4 and Respondent6 are scored as "a" in Table 4.1.

Respondent5 explains that his team mostly share knowledge on the spot. When somebody needs it, they receive. Lately they also have had meetings to discuss certain topics and share the knowledge in order to get a shared understanding. Respondent7 also mentions these meetings: knowledge sessions. 'That is a very useful tool for sharing knowledge'. Based on these statements Respondent5 and Respondent7 are scored as "a" in Table 4.1.

4.1.3 Organized information used to score the indicators

This paragraph shows all the information that was used to decide upon the scores per respondent, which are depicted in Appendix 4.1.2. data is translated from Dutch to English, shortened and organized per respondent. For every indicator it starts with a summary, which contains the information from all respondents. After that summary, the paragraph presents the information per respondent. The respondents only answered the indicators that were deemed dominant for their role.

Benefit of working together is clear and communicated

Respondent1 explains that they communicated a lot during the transition towards a DevOps team. Before this team, all individuals worked at the client site. It was important to communicate why they wanted to do this. It was a big change to create a team that needs to create recurring revenue from individuals who needed to be billable. It helped that the team agreed. A colleague, who already did a transition like this, was consulted in how to approach this transition. While Respondent1 had this transition one year ago, Respondent 2 and 3 did this more than three years ago with teams. They both sketch a situation where there was mismanagement, not enough room for development teams, low performance (unacceptable bugs, bad releases etc.). In the beginning of 2016, there were 5-7 people who came into their organization, amongst them were respondent 2 and 3. They professionalized the way teams work and created an IT leadership block between stakeholders and teams.

During that period, a few dreams were shared; like we want to be able to release at every moment and have a stable product. That was quite a shock for the employees who created their own culture. That has an impact, not only positive. It felt like an invasion, because the people there thought it was going great. They did not see the entire picture. The impact of failures. That had a big impact. Help people to say what is going wrong. Not only what is going wrong? Also, why, and who is responsible and why do not you tell him when it happens. A new way of working was introduced, which was build up and communicated. As an employee you had two options; work with the change or leave. Respondent3 tells that he wanted a culture where the teams goes to the stakeholders and says, can you use this? Instead of the stakeholder, the team should be in the lead. Give the team the responsibility. This also means, the team needs to take responsibility. If they make a mistake, then solve it yourself. If a team promises something that we will never do, fix it yourself. That is something you have to guard as leaders. We make teams more and more owner of a whole solution, including questions and implementation. That really works.

Respondent1 (y):

Such a decision has a lot of impact. Internal organization shifts and employees that were billable are now assets that should develop recurring revenue. Communicated very clearly at the beginning. Also discussed whether they saw the benefit and wanted to do this. And they did and agreed that this would be better for everyone. During the transition we used the expertise of a colleague from another business unit who already organized such a transition.

Respondent2 (y):

Three years back a lot changed for DevOps1, unit core. The quality of the product was deteriorating. Unacceptable bugs, bad releases et cetera. That is unacceptable. Installation issues, not being in control. There was mismanagement if you ask me. Employees were not educated. We felt it everywhere, internally and in the projects with customers. A former colleague, an agile enthusiast, and me (more technology oriented) stepped up. We wanted to change, but that meant investing and getting talents away from the customer project and use them to guide the transition. A few of them are respondents of you. We came in with 5-6 man. We knew each other. Trusted each other. We said we are going to do this. And walked in the building. That has an impact, not only positive. It felt like an invasion, because the people there thought it was going great. They did not see the entire picture. The impact of failures. That had a big impact. Help people to say what is going wrong. Not

only what is going wrong? Also, why, and who is responsible and why do not you tell him when it happens. Slowly that creates a different culture.

Respondent3 (y):

When we started in 2016/17 the stakeholders of the team did not give the team the room to become a team. Higher management came in pushed work to the teams. We created a block, with IT Leadership.

So, in 2016 we created teams. Gave them the space to use their ideas. Instead of stakeholders that say I want you to make this now. The team says: I made this; can you use it? That is quite the turn.

You now see the same happening at Public (respondent1), where the team goes through the same development.

This also means, the team needs to take responsibility. If they make a mistake, then solve it yourself. If a team promises something that we will never do, fix it yourself. That is something you have to guard as leaders. Do not make the mistake of doing that yourself, as a manager.

We make teams more and more owner of a whole solution, including questions and implementation. That really works. Make the team owner of the problem that works better. This remains hard. They find me annoying sometimes. They think; I have a customer, so help me out. I think we delivered more value than ever. And that is why we want to work and collaborate like this. It is my job to tell that story. To the teams, but mainly nowadays to the stakeholders.

Respondent4: -

Respondent5: -

Respondent6: -

Respondent7: -

Leaders know the environment and senior management

Respondent1 states that it's an open organization where people feel free to connect and close together, both physically and mentally. Respondent 2 and 3 are more critical. Respondent2 explains that he has a lot of responsibilities and could be more available for the team. He states that there are two architects who are close to the team and that they are close enough to him. Respondent3 thinks that the organization could be flatter. 'The team knows what's going on it's often the layers above the team that prevent them from excelling at the highest level'. He also thinks that product owners are thinking to hierarchic and states that it should be the other way around. Leadership should ask the team, as servant leaders, if there is more that they need to excel. So, there is access, but people could take more responsibility. That is something that the organization could improve. That demands something from the leadership team.

Respondent1 (y):

We are in an open space and everybody can walk to each other. We are close and connection and the distances between individuals are short, both physical and mental. It is very important to have cadres. Cadres about quality, customer care, feedback etc. Within those cadres I find it important that the team decides what to do. Leadership is also in the team.

Respondent2 (a):

Looking at myself. It decreased. I am now responsible for architecture internally and in customer teams. The R&D maturity is higher, so that has not got my focus lately. What I tried to achieve is to boost the availability of people close to me. So, architects are available, but the lead architect (me) is sometimes not available. Although I regularly meet with the architects. There are two architects for the two teams, they are available 50/60% of the time. So, the teams lean on the architects and the architects lean on me. One of the architects organized a meeting every Friday. We talk about the architecture and he demands that I am there. That helps me.

Respondent3 (n):

In my opinion it is not flat enough. We units operates on different band widths. ICT is simple, 0's and 1's. So, if I have to uphold to three bandwidths, that will not work. I have to make choices and don't feel autonomic in that.

Business unit strategies will align, but it takes a lot of discussion.

The teams know what is going on. It is often the layers above the teams that stop the teams from excelling at the highest level.

There is enough access. I would like the PO's to grow in their maturity and attitude. Sometimes they are to hierarchic. We were raised like that and it is hard to get that out of everyone's system.

For me it is a task for higher management to think hierarchic. And what I mean by that is servant leadership. Do you have enough from me to perform your job and if not, what do you need? That is really important if you want to work DevOps.

If I want control, I will use the waterfall method. You can use control over all the phases and use the current traditional management style. If you want DevOps, you need different people with different styles.

Respondent4: -

Respondent5: -

Respondent6: -

Respondent7: -

Decentralized day-to-day Decisions

All respondent claim that the team decides regarding the day-to-day decisions. They are responsible for the solution and have to decide how they will meet the requirements. They also decide when they seek guidance outside the team. Respondent2 explains that the only things that are centralized are: HR and education. These things are decided by the BLT (team of team leads). Respondent3

explains he is only an escalation option. Most of the execution is with team members and some of the functional responsibilities lie with the PO and technical responsibilities with the architects. They set cadres and within that the team can decide. He explains that there sometimes are hot topics or blockades and then one of the leaders will try to use stakeholder management to solve it. It is our challenge to not solve it, like you instinctively want and often expected from you. Another thing that we keep deciding is the balance between working hard and rest. We try to measure on outcome and not on input. Respondent2: People who have a hard time in private life will have more room, they can for instance come in late. Respondent3: if you are tired, please go home. We trust you have a good reason for it and come back tomorrow.

Respondent1 (y):

Me and other leaders do not get involved with the daily decision making. There is a solution architect in public and several in core who can be asked for advice. The team is responsible for the solution that they chose make. They also decide when they ask for advice, from for instance the architect. When they make decisions that are not the best choice, in mine opinion. I will let it happen and will only bring it to the table when it blows back. Then we, as leaders, give feedback.

Respondent2 (y):

Well... HR and education are centralized with the team leads. And even that... if they want an education. Sure, tell us what it is, what you want to do en sent us the costs. We will decide and let you now. The rest is for the team. Management is as often involved as the team sees fit. The impact on autonomy is huge.

Respondent3 (y):

They decide almost everything. PO is responsible and I am an escalation option. I try to decide as less as possible and to be like Switzerland. People remain traditional towards management. They want you to make the choice. That can be a though balance. Sometimes I answer or the lead architect, but we have to improve in that. I do not want to give the answer. Why doesn't the team decide it themselves? They are afraid or not equipped. What can I decide to make an impact? Could be a hot topic. So, I will use stakeholder management on the topic. Searching for the blockade...

We also influence rest and balance. People often feel like they have to be 100% when they are here. But that is not how we look at things. If you are tired. Please go home. Rest. Today you will not deliver value and you have a reason for that. And we trust you on that. We also believe that the next day when you come back to work, that you will be eager to pick it up again. It important to create space for these things. That is again about leadership and about facilitating. A nice example of how we also help the team to relax is that in a 3-month period of 6 sprints we always have one sprint where we do something else. A research week, or a bug squash week, or something with gamification. Or decide for yourself what you do. Be creative, we do not need value from you this sprint. You cannot always run.

Respondent4: -

Respondent5: -

Respondent6: -

Respondent7: -

Individual team members have focus on the team

As stated before, the department of Respondent1 made the transition later than the department of Respondent2 and Respondent3. Respondent3 states: 'Team is above all. Team members should also correct each other when they are not behaving like that.' Respondent2 agrees and states that make the team responsible for their own organization. 'Everyone is hardcore in a Scrum team'. The things that he is focusing on is how is the team performing and is there someone who is slowing the team down. He states that one underperformer slows the team down with 25% and that he wants to prevent that effect. The latest step that they took in that regard is giving the team the responsibility to write the job application and find the new team member. So, team members in Core, were Respondent2 and Respondent3 lead the development organization, have a full focus and responsibility for the team. Respondent1 states that their team members are also fully focused on the team. However, there are still a lot of stakeholders who expect something from the team. The organization demands a lot of flexibility of the team. The unit Finance, were Respondent1 leads the development organization, tried to professionalize this. They appointed a manager that functions as a barrier between the stakeholder and the team. This helps them to focus more on the team, but it could be improved.

Respondent1 (a):

The team is 100% dedicated to the team. There are, however, several stakeholders that can expect something from the team. We ask a lot of flexibility; we need to be able to deliver what the customers want. We want to protect the team more than before. If you compare it with a year ago this has been professionalized. We try to canalize this process by using the manager as a barrier between the customer and the team when this happens. They are limited in their autonomy by the cadres that we set.

Respondent2 (y):

We try to make the team more and more responsible of their own organization. Everyone is hardcore in a scrum team. There are two architects that have a flexible role. Official team members have focus. 70% technological work and 30% ceremonies.

Aside of the scrum ceremonies we try to let them work. If they do not like those then we talk about the why not? If there is no good reason, then it is also like: hey guys we are at work. It is about; give and take. What helps is one on one's. Understanding complex home situations etc. It is a team sport. And it is not a sprint (scrum), but a marathon. If one of your team members is not achieving that takes up around 25% of your productivity. If someone has a bad influence on the team, it is important to let him go. He can be amazing, but if he does not fit, he does not fit. To prevent the 25 reduction, we talked about, we said to Rubic, pick your new team member yourselves. They did the interviews. That is very different than just hearing that you got a new colleague. They are responsible now for making it work.

Respondent3 (y):

Team is above all. Team members should also correct each other when they are not behaving like that. They decide what other actions that can be taken responsibly. We allocate a member fully to

the team. Of course, there can be situations where someone is “borrowed”, but it has to fit in the rhythm. There is one dev who does a lot of consultancy. I do not need to know. If he talked to the team and the agreements are there than it is fine. What I do is: are hours booked, invoices sent? That is servant leadership. Feedback will come when it was not okay.

Respondent4: -

Respondent5: -

Respondent6: -

Respondent7: -

Strategy and expectations of the organization are known

None of the respondents clearly state that this done in a good manner. They all seem to see room for improvement. Respondent1 is the most positive, but he mostly talks about the vision how the team should work. He states that he knows what is expected and believes in the vision of helping the customer. Respondent2 states that the communication towards teams is transparent. The problem lies in the vision across business units. The visions are often unit specific, but the platform that is used by all unit is not. The business unit core is owner of the platform, so they often have discussion with internal and external customers. Respondent3 adds that the decision making is often behind closed doors. He states that this does not help with the adoption. The strategical layer and the executive layer are not always in agreement about what is the best thing to do. Respondent3 calls that a mismatch. Respondent2 and Respondent3 do tell their employees what they can. Respondent3 has several tools, like a 3-month plan that he shares every quarter with the teams and customers. He also has several ambitions and dreams that he shares with the team, in order to create guidance. Where do we want to be in a few years? Respondent2 emphasizes that before they share the strategy with the team there often first is a political game.

Respondent1 (a):

We want products of a high quality/value for our customers. The team knows the vision. Although I perceive it like that. In my opinion they can do everything. They believe in the vision of helping the customers. The new way of working created new energy.

Respondent2 (n):

Actively. If we think it is clear. Communication is transparent. Over the units it can be unclear. So, then we have to figure it out first. That can be about politics as well. Sometimes it is a negotiation. This is the hardest part at this moment. The vision is unit-specific, but the platform is not. We should stop with that. I think that the party that develops the product should have the vision. So not from units like public and finance. We want to help teams with connecting to customers, but platform vision. it should be ours. From that perspective the customer is number 2. You want to be the owner. Just like a team wants to be owner of a component of the platform. It should be carried by the teams. It should be about accepting what the product is and is not. So, it is also about saying no. I like that, most do not. Saying no is not the hard part, accepting no is. That is someone with a problem, who needs a solution. So, you have to be strong. On story level it is easy, on epic level it is

harder, but on initiative level it is very hard. You are directly discussing with customers. That is one of my responsibilities. Public and finance are also customers from us. At least that is how I see it and we are learning them that they are.

Respondent3 (n):

Sometimes vision is decided behind closed doors. This does not help with adoption. That is a shame, because it does not support DevOps. The goal would be to have meetings with an open door. People can sometimes join in the decision making. They do not always agree. If employees disagree you have a mismatch. There is a push top down and a push bottom up and that should match and align, that is something that we could improve. This is an interesting topic. Important for us at this particular moment. Our DevOps vision is clear, but our vision for the platform and company is less clear. Platform: If you make teams autonomous you have to decide what you do and don't do. When is the platform in his strength? You have to be able to say no. Then you need to know what the cadres are, and you need to evaluate that every 6-12 months.

I use three tools to give an insight in what we are going to do. The first one is the ambitions (statements, dreams), these are things that will take years and that we use to motivate and guide. The second is for teams, customers and stakeholders. I always say I will be transparent for 3 months. That is the certainty they have and the way they can see whether we are still doing the right stuff. I want to be flexible beyond these three months and be able to adjust. The third is a list with blocks. All kinds of ideas that I and others have. I do not know when we will make it. We see what we can fit each quarter. So, each 3 months I know for sure and furthermore I have big box with blocks. If there is the feeling on unit level or MD level that we are doing the wrong things than I will hear that. And I am able to adjust, because of the 3 months.

So, importance of vision is clear, but on unit and platform level it could be clearer.

Respondent4: -

Respondent5: -

Respondent6: -

Respondent7: -

Opportunities and limitations are known

Respondent2 and Respondent3 state that: 'everything is possible' and 'we have a just do it mentality'. If needed they will give feedback and if the team has doubts; ask a question. Instruments like; epics, technological roadmap etc. are in place. Respondent2: 'if you want to try stuff. Feel free. Try it isolated. Show me because I am interested. And if goes wrong, fail fast'! Respondent3: 'You don't want to control people on input, but on expectations and outcome. Cadres for that should be clear, but it also important that they are not so strict that they can't evolve'. Respondent1 states that it is the job of the manager to make sure the opportunities and limitations are known and respected. Leadership decide the limitation and the team is able to refine them during evaluations.

Respondent1 (y):

The manager is responsible for maintaining that. He also makes reports about what happens and what the team does. We evaluate these together. Limitations are things outside the cadre. We know how to prioritize, what we expect from each other. Why we do things. The limitations are created by leadership team and refined by the team.

Respondent2 (y):

For me it is clear. Everything is possible. And if you have doubts, ask a question. We write epics, that already covers the functionality aspect. We also have a technological roadmap/backlog, with focus on infrastructure and the future. If they want to try stuff. Feel free. I do not know everything. Make a server, try it isolated, show me the results, because I am like that. And if it goes wrong, fail fast!

Respondent3 (y):

We have a "just do it" mentality nowadays. If it is not okay, you will get feedback.

Cadres should be clear, but it is also important that it is not too strict and it can evolve. It is not black and white and the progress (win/lose) is in the grey. You do not want to control people on input, but on expectations and outcome. Use their creative skills.

Respondent4: -

Respondent5: -

Respondent6: -

Respondent7: -

Self-aware team members

The answer regarding this indicator vary. Respondent 4 and 7 are the most optimistic. Respondent4 states that they have retrospectives and that he thinks there is self-awareness in these meetings. Respondent7 tells about a meeting where they had an in-depth feedback session. He says there were no big surprises for anybody and sees that as an argument to state that the team members are self-aware. Respondent5 and Respondent6 are a little bit more reserved. Respondent6 it is a hard indicator to judge. He thinks that their team is sober and there is culture without a lot of boundaries. Therefore, team members can say a lot to each other, which made them all more self-aware. Respondent6 thinks it varies and things can happen, often due to emotions or lack of experience. He talks about his trigger point: 'If somebody made a bad solution and it took a long time, I will start complaining'. He states that it is important to know these kinds of things from yourself and your team members.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (y): Most of the time they are. Teams have retrospectives. During these meetings sensitive things are addressed and team members show that they are self-aware. For one of the teams, where I regularly join, I definitely see that.

Respondent5 (a):

That is a hard one. We are sober team. We can say everything to one another. Sometimes that is with a lot of humor and sometimes it is serious. There are not many boundaries in our team, we can say a lot. But if you cross the boundary of somebody else, it will be discussed. That has made us all more self-aware.

Respondent6 (n):

I think this varies. Sometimes things happen, often due to emotion or lack of experience. I, for instance, can be very sharp in discussions and react fierce on things that are relatively unimportant. Then it is in a zone which I find important. If somebody made a “bad” solution and took a long time, that triggers me. I will start complaining. My team knows that I can do that and I know it too. They accept it and know that I need to release that. We will fix it together. Everybody has these kinds of things. That you need to be aware of. Within the team we are aware of strengths and weaknesses and able to share this.

Respondent7 (y):

In general team members are pretty self-aware, regarding what their attitude towards others. There are a few quirks that somebody can have, that might not always be obvious to himself. That is why we have feedback sessions. We had a feedback session before our mid-year review, a moment when we need to gather feedback regarding our performance, which we used to give feedback and discuss everybody individually. We used the start-stop-continue method. So, everybody wrote their feedback down for all team members. On a high level, there were no big surprises or weird feedback points. The team members accepted the feedback and saw the resemblance. Also, in our team it can happen that somebody does something that has a very different output on others than intended. Then it really helps that you can give feedback. It should, therefore, be a continuous process and not only something you do in retro or during a mid-year review.

Environment for asking questions

All respondents state that they have an environment for asking questions. Especially questions regarding content. Respondent5 says that asking is a prerequisite for being part of the team. Respondent4: makes the comparison with 2016 when 6 out of 7 people were new. At that time, it was not that normal, but now it is. He states that nowadays the culture is strong enough to handle a few changes in the team. Respondent6 likes the helpfulness of his team. They do what is necessary to help the other. Respondent says that you can become the center of ridicule when you ask something. Sometimes the other disagree and make jokes. The important thing for him is that he is able to trust his team members and is able to ask and share what he wants.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (y): Yes, at this moment that is definitely the case. In 2016 6 out of 7 people were new. At that point it was not that common to ask questions. People now know each other. Nowadays, there are still changes in the team. Especially in one of the teams, but that culture stays the same. Everybody will get the help he or she needs and feels the space to ask questions. Aside that there is a lot of fun, but there will always be room for serious questions.

Respondent5 (y):

Always. We always ask questions. That is not a problem, it is a pre-requisite.

Respondent6 (y):

Content questions are asked regularly. We are also very helpful towards each other. Then we just grab a chair and help as long as possible. Just what is necessary. Other questions are a bit harder sometimes.

Respondent7 (y):

We are not a team where you should be cautious with sharing your thoughts, we will not share confidential information regarding individuals outside the team. If we share something within the team, it is up to that person to share it with their environment. This safe space results in a place where we can ask a lot. You can become the center of ridicule, but that will be in a playful way. The openness and trustworthiness of the team is what makes our team. You have to be able to trust each other, without that trust it is impossible to work together.

Team members listen to each other

Respondent4: 'They hear each other, but sometimes I wonder if they listened'. It happens regularly that do not speak the same language or do not want to understand each other. Respondent5 agrees and states that it also depends on the individual. Some team members like to lead and decide by nature. 'They will stand for that opinion and can be pretty stubborn. Sometimes that can result in frustration'. Respondent6 thinks it is on an acceptable level. He thinks that the willingness to ask questions and listen to one another is the reason why the helpfulness in their team is high. Respondent7 shares the following insight: 'If one person has a fundamentally other idea than someone else, then it's natural that it creates conflict. Those are often about content and that is fine. Then we discuss and dive into the options. Lately we had a story and there were two people with very different ideas on how to approach it, which would decide more things than just that story. It took a few weeks to agree. What we do is; we discuss it within the limit of the meeting and we do not agree, we park the discussion. We will discuss it in the next meeting, but leave the conflict for that meeting'. It shows the ability to listen to each other, also if team members disagree.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (n): They hear each other, but sometimes I wonder if they really listened. There is a big difference between those things. It happens regularly that people do not speak the same language or do not “want” to understand each other.

Respondent5 (a):

This is an interesting one. It varies. It has to do with the characters within the team. Some team members have a very strong opinion regarding certain matters. They will stand for that opinion and can be pretty stubborn. Others are more like followers. Sometimes that can result in frustration, because some of these team members are not heard. The stubborn people are often the ones who decide the course. Partly due to interest and partly due to character.

Respondent6 (y):

That is quite ok. Some are more stubborn than others of course. When you ask for help you are stuck, so you need somebody to step up. We also appreciate that we help each other like that.

Respondent7 (a):

That depends. There are conflicts. If one person has a fundamentally other idea than someone else, then it is natural that it creates conflict. That is logical. Those are often about content and that is fine. Then we discuss and dive into the options. There is always a solution. Lately we had a story and there were two people with very different ideas on how to approach it, which would decide more things than just that story. It took a few weeks to agree. What we do is; we discuss it within the limit of the meeting and we do not agree, we park the discussion. We will discuss it in the next meeting, but leave the conflict for that meeting. As a team you need to prevent that you take that with you during the sprint. That is not good for the vibe in the team. It also not good for a team when there a no conflicts at all. You should not agree all the time. That can slow you down as a team. It is ok to agree most of the time, but sometimes you have to challenge each other. Some team members have more knowledge about a certain area or have different viewpoints. Then it is logical that there will be a conflict. Sometimes we go with one of the opinions of a team member and sometimes we find some middle ground.

Moments of reflection and evaluation

All the respondents are aware of the importance of reflection and evaluation. However, there seems room for improvement. Respondent4 states that they are quite good in giving and receiving feedback when the team is in a good spirit and things go right. They also have fine retrospectives, with attention towards the work process and team development. ‘However, when things go wrong it becomes harder. The same goes for compliments. It is also noteworthy that these things are often said in a retro, but almost never outside of it. Especially when it is not about content. We are not that good in stating that somebody is underperforming. Lastly, we had a situation with somebody who was not performing due to personal circumstances. This has been resolved but I do not know if the team would react better when a similar situation arises. Respondent6, who attends the same meetings, thinks that team members are too sweet for one another. ‘I am a direct guy, but I am often holding back. We could be more direct to each other. More like we are in business, instead of friends’. He states that the balance between work and private life that is a hard. It can also be dangerous to know each other privately. ‘I am able to have a discussion with respondent 2 or 4, but

at the end you know, it is okay. We know where we are together, both on the work-aspect and on the private-aspect'. He likes that people get irritated and show passion for something. 'Could be more in our team. To regain that culture, we need coaching in my opinion and maybe an external coach that creates and discusses conflicts. We could learn a lot from that. I don't know if it's necessary, but it definitely a point for improvement'. Respondent5 explains that they always use the same format during a retrospective, normally this format tends to focus on the content. Lately they had a feedback session that was focused on team development and personal relations. That had a positive impact according to Respondent5 and Respondent7. They decided to do this more often. Respondent5 and Respondent7 both stated that they think is important that these moments are not limited to the meeting. It should be something that addressed when it occurs and something that is saved for the meeting. They say they have a culture wherein that is possible, but both seem to think that more focus on the team development and personal relations could improve their team. Respondent7 adds that it is important that the one who receives feedback is the one who decides whether he or she will do something will it. He, and the team, also believe in positive feedback: 'Sometimes we do that during retro's, then we have bromance letters. If somebody helped you with something, you get a bromance card'.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (n): Retrospectives are a good example of reflection and evaluation. Scrum masters facilitate this process. One of teams has decided that they will rotate the facilitator of a retrospective. That creates new insights and visions. The retrospective is periodically. We address the work process, team development and feedback. Once there was a mid-year review coming up. So, then we used that meeting to give each other feedback. During the meeting, when we are in a good spirit, we are quite good at giving and receiving feedback and asking critical questions. However, when things go wrong it becomes harder. The same goes for compliments. It is also noteworthy that these things are often said in a retro, but almost never outside of it. Especially when it is not about content, but about the relation and/or communication. We are not that good in stating that somebody is not able to perform. Lastly, we had a person who was not doing well, also not in her job. This has now been resolved, but I do not know how the team will react when a similar situation arises.

Respondent5 (a):

As a team we do this every other week on Monday, after we finish the sprint. This is the retrospective. We use every time use the same format and it is more often focused on content, then on personal relations and team development. Lately we had one meeting where we had some depth regarding personal relations. We concluded that it was useful to do and that we should do it more often. On the other hand, I feel like everybody should be able to address his or hers feeling at every time and not only during a retro. So, when there is friction, say it. Then we can fix it earlier.

Respondent6 (n):

We try to do a retro every sprint. This is not always focused on how the sprint went, it also goes about personal stuff, team development or personal ambitions. I do think we are a too sweat for each other. I am a direct guy, but I am often holding back. We could be more direct to each other.

More like we are in business, instead of friends. In the end you will get better by being direct and open. If something is not good, people say something like: yeah that's not very good you should maybe probably if it's possible do something about it". No. It was not good enough, according to our standards. You need to be able to say that, maybe sometimes individually, but nonetheless it should also be possible in the group. This happens less than before. I think we know each other better now and are sometimes too kind to each other. That is the danger of knowing each other well and, also, privately. The balance between work and private life that is a hard one. I am able to have discussion with respondent 2 or 4, but at the end you know, it is okay. We know where we are together, both on the work-aspect and on the private-aspect. But I do not have that with everybody and that is the same for everyone. A while back we had a tester that said to me, I do not like it that you say to me that it is useless what I did. I said that literally, because it was not good or smart enough. He got frustrated and tried better the next time, twice as hard. That is how it should be. Not: could you try to do it better next time. No that is too sweat. We are in the real word. It not a social workplace. I also have no trouble when people get irritated, that is good, raise your voice occasionally. That is ok. Could be more in our team. To regain that culture, we need coaching in my opinion and maybe an external coach that creates and discusses conflicts. We could learn a lot from that. I do not know if it is necessary, but it definitely a point for improvement. We could also be more positive and give compliments. That is also a thing that I could improve. We do not have lead dev's anymore, but I am experienced and know a lot about the organization. I could use this to motivate my team members more.

Respondent7 (y):

See also my answer on question 19. We do this often, sometimes in the retrospectives and sometimes outside the meetings. It is important that you can give each other feedback and that the one who gets feedback is willing to listen. However, it is always up to the team member who gets feedback what he does with it. He or she should say whether it is feedback that he or she will use and why. It cannot be like pointing fingers and you do this and that and stop it. That does not work. Instead of that it might work better to give each other compliments as well, about the work or what is achieved. Sometimes we do that during retro's, then we have bromance letters. If somebody helped you with something, you get a bromance card. That kind of stuff, a way of appreciation. Yeah that is a good thing.

Team members complement each other and have some level of experience

Respondent4 states that both teams have a nice balance, also in terms of personalities. 'Everybody has his or her main focus or things they find important. That diversification creates a nice balance'. Respondent6 find truth in these words and thinks that everybody in his team brings in some value. Different type of personalities that make the team complete. Respondent5 emphasizes the decision process of his team. 'We are with six persons, so there will exists islands regarding certain matters. At the end we can decide better, based on all the input that is delivered from different viewpoints. That is very positive for us as a team. In that way the team members really complement each other and are we able to gather a lot of information and have a good discussion'. Respondent5 and Respondent7 both think there is a good mix in their team based on personalities. They do not think in the colors; red, blue, yellow, green, but they do see that some are more leaders and some more followers. Respondent7: 'I can be more modest during refinements or retrospectives. I listen a lot. There are also a lot of people who talk a lot and share many ideas. That is a dynamic that works'.

Respondent7 also states the team is busy with making sure there are no single point failures, which were abundant in the past.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (y): The two teams that we have internally are pretty well balanced, also in terms of personalities. Of course, everybody has his or hers mean focus or things they find important. That diversification creates a nice balance. Team members are also mature enough, so in our opinion they do not need a full-time scrum master. That was different in the past. When some team members really needed personal growth.

Respondent5 (y):

We have a lot of discussions, especially regarding content. We are with six persons, so there will exist islands regarding certain matters. At the end we can decide better, based on all the input that is delivered from different viewpoints. That is very positive for us as a team. In that way the team members really complement each other and are we able to gather a lot of information and have a good discussion. On the personal level, I would not know which type of DISC-personality everyone has. We do know the styles that people have in way of working and solving a problem. That is a good mix. Some people want to be leaders, others like to follow. Some are more detailed etc.

Respondent6 (y):

We have a diversified group of individuals. I think that everybody brings in some value. Some are very concentrated and detailed. Another is a real go-getter. So, we have different types of personalities who make our team complete.

Respondent7 (y):

Everyone in our team has his specialty. I am more a CICD kind a guy. One is about automation, part of CICD, but more the DEV side of it. I am more the OPS guy. We also have some pretty smart guys, that like to solve the abstract problems. Everybody brings something to the table, with knowledge and/or ideas that they have. If you have a team with only Brainiac's, that will not work. Of course, you might get perfect code, but it might not be maintainable. You need those different viewpoints. I can create a build street which I understand and can work with. But if somebody has no affinity with this topic needs to be able to maintain it and I don't structure it in a logical way... and it's get hard to maintain without in depth knowledge, than it doesn't provide value. You create single point of failures.

If you look at personalities, we do not have the blue-red-yellow-green distinction. We do not use that a lot. What we do know is that some people are more outspoken and some are more silent. I can be more modest during refinements or retrospectives. I listen a lot. There are also a lot of people who talk a lot and share many ideas. That is a dynamic that works. You need the right balance in a team. That is also why all these theories are designed on how to build a team.

Responsibilities and skills should be clear

'Things is somethings that the teams know best', according to Respondent4. Respondent6 states that employees we just joined are still learning what is expected. The prerequisite is that they have the right attitude and level of thinking. If that is on the right level, then the team will help them to get up to speed. Respondent4 says: 'a good example is when we hire a new person'. The team have to write the job application. 'At first these applications were very demanding. So, we could not find that person. Nowadays teams say give us someone with the right attitude and we can educate them internally'. Respondent4 and Respondent5 state that the team members have intrinsic motivation to perform well and make a good qualitative product.

Respondent5 and Respondent7 talk about the different roles that individuals take. Some are more interested in tackling that hard-abstract problems and other like to automate. In their team the balance might be little bit off. Respondent5: 'There is not a lot of interest in that and because that interest isn't you can see that we don't do much in that area. So, I would say we are more capable in Dev, then in Ops'. Respondent7: 'Everybody brings something to the table, with knowledge and/or ideas that they have. If you have a team with only Brainiac's, that will not work. Of course, you might get perfect code, but it might not be maintainable'. He also states that it is important to prevent single point of failures. That has been something they focused on the last years.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (y): This is somethings that the team knows best. A good example is when we hire a new person. The team needs to be able to describe what they need. At first this was very demanding. So, we could not really find that person. Nowadays, teams say give us someone with the right attitude and we can educate them internally. When responsibilities are not available in one team, they sometimes ask questions to the other team. It is important to be aware of the gaps. Team member are intrinsically motivated to perform well, especially on quality and delivering a good product. So, they take these responsibilities quite serious. If they feel that responsibility, they will take it.

Respondent5 (y):

In general, we have a good grip on that. Especially on the Ops-area we have a gap. There is not a lot of interest in that and because that interest is not you can see that we do not do much in that area. So, I would say we are more capable in Dev, then in Ops. In the end we could all do it, together. Most of us does not get energy of it, so it not something we pick up easily. Development motivates us more. So, our responsibilities and skills are clear and we also know where we could improve.

Respondent6 (y):

We work here with very smart people. The employees that just joined our teams are still learning what is expected, but everybody has a logical way of thinking and knows what he or she should contribute. If that attitude and intellect is okay, then it all comes naturally is my experience. If a new person joins the team, we help him. The team culture is in place. You will not be left alone, but we get you up to speed.

Respondent7 (y):

I think we already covered this in the previous question. We know what is going on and what we need to achieve. We also know what it takes to achieve that. Although we have one role: developer. We have different skillsets.

Psychological safety to share opinions

Respondent4: 'we have an experienced population'. He thinks that helps. 'Experienced professional people know what it takes to be successful in the work environment'. Respondent6 also seems to share the opinion that experience helps in that regard. He also states that there is not a lot of attention for it. 'I think we could be more aware about whether it's psychologically safe for all us and what we need for that. The awareness would be a real improvement'.

Respondent5 and Respondent7 both stated several times that it is very much possible to be the center of ridicule. If you say or ask something that is considered "stupid", they will have a culture to make fun of each other. However, they both say it is a positive element and it is always in a playful way. They both think it is not something that creates a blockade for sharing opinions or feelings. Respondent7 states: 'A while back we had a conflict that was heated for a while. At a certain moment, emotions started to rise between a few team members. You should not stop at that moment, but continue. Talk about it. What is going on? Most of the time there is an underlying issue that needs to be addressed'. This approach helped them to solve the issue. 'Those moments are natural in a team I think, it shouldn't be all sunshine. That is almost impossible. It also creates a certain sharpness. So, you need both, calmness and conflict'.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (y): I think it is safe. At least I hope so. We have an experienced population. That is necessary, because we have a heavy and complex product. I think that has an influence on this. Experienced, professional people know what it takes to be successful in the work environment.

Respondent5 (a):

That is always possible. Although, you could say something which we will use (in a funny way) against you for months. That is the funny dynamic which is there. There are many ideas and opinions and sometimes somebody shares something that is not a very good idea or where others have very different opinion. Yeah, then you could be made fun of for a few weeks. That is the kind of humor that our team has and everybody will be handled the same. It should not stand in your way when you want to discuss something and that happens a lot. Also, when it is not about content, but when somebody does something that which goes against the values of somebody else.

Respondent6 (a):

We have a pretty safe environment, I think. I do not know if it is something that we talk about and/or that people are aware of. I think we could be more aware about whether it is psychologically safe for all us and what we need for that. The awareness would be a real improvement.

Respondent7 (y):

That is a funny one. Yes, there is, definitely, safety. On average. A while back we had a conflict that was heated for a while. At a certain moment, emotions started to rise between a few team members. You should not stop at that moment, but continue. Talk about it. What is going on? Most of the time there is an underlying issue that needs to be addressed. It is important that the team does not undermines that discussion. If you do not talk about it becomes bigger. Right now, it is solved. Those moments are natural in a team I think, it should not be all sunshine. That is almost impossible. It also creates a certain sharpness. So, you need both, calmness and conflict.

A culture of asking questions

Respondent4: 'It can really help to be critical to each other. Especially across different roles. As a developer you can be so deep into the content that you don't have the overview anymore'. He feels like everybody can feel and feels free to ask these questions. Although he also states that it is important to stay constructive. He, black and white as he can be, finds that hard sometimes.

Respondent6: 'We ask a lot of critical questions'. For instance, during code reviews. 'A review is not a tool to criticize, it's a way to help each other forward and to make sure that we did the right thing. That is a mentality which is not always there. Especially new people feel like they are getting told what they did wrong, but that's not why we do it'.

During the interview with Respondent5 this was one of the main topics. 'This we do a lot, especially about the content. We have many stubborn team members, but at the end you have created a better image of the problem and the possible solutions. That helps in the decision making'.

Respondent7 adds: 'Ask about it if you see it. We also challenge each other when behavior is not ok or when we need to challenge each other on content'.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (y): I really like critical questions. Everybody can do that. I am a black or white kind of guy. Sometimes that can backfire. It is important to stay constructive. For me as a partly outsider it is easier to ask the critical question. As an architect you are not fully into the content. As a developer you can be so deep into the content that you do not have the overview anymore. It can really help to be critical to each other. Especially across different roles. In our situation that is definitely the case.

Respondent5 (y):

We kind a handled this through all the other questions. This we do a lot, especially about the content. We have many stubborn team members, but at the end you have created a better image of the problem and the possible solutions. That helps in the decision making.

Respondent6 (y):

We ask a lot of critical questions. They are often handled correctly. Sometimes a question might not get the answer and time it deserves, due to pressure. But most of the times we take the time for it. During code reviews and refinements there are many questions, also regarding the way we work. During retrospectives, with elements like personal relations, it can be slower as I said before. During code reviews I like to take it a step further, so; this is good, but this would make it better. A review is

not a tool to criticize, it is a way to help each other forward and to make sure that we did the right thing. That is a mentality which is not always there. Especially new people feel like they are getting told what they did wrong, but that is not why we do it.

Respondent7 (y):

We also talked a lot about this. I think it is something we do well as a team. If something is going on, also on the personal level, talk about it. Ask about it if you see it. We also challenge each other when behavior is not ok or when we need to challenge each other on content.

Team members that share relevant information

Respondent4: 'Team members share information. But I feel like you can also create a fuzz if you share too much information. So, the cadre, relevant information, is very interesting. So, you need to decide what is relevant for whom'. He thinks the team and the leadership around teams shares knowledge when it is necessary. 'It is pretty hard to find the balance between transparency and not creating turmoil'. Respondent6 states that the team members gather and check information. Furthermore, he adds: 'But often the important details are neglected, often due to a lack of oversight'. Due to the complex environment it is hard to know all the relevant knowledge and, therefore, also hard to share all the relevant knowledge.

Respondent5 explains that his team mostly share knowledge on the spot. When somebody needs it, they receive. Lately they also have had meetings to discuss certain topics and share the knowledge in order to get a shared understanding. Respondent7 also mentions these meetings: knowledge sessions. 'That is a very useful tool for sharing knowledge'.

Respondent1: -

Respondent2: -

Respondent3: -

Respondent4 (a): Team members share information. But I feel like you can also create a fuzz if you share too much information. So, the cadre, relevant information, is very interesting. So, you need to decide what is relevant for whom. If you share everything you create chaos. There is information that we, architects, do not share with everyone. Nowadays, we are discussing about the shared vision, what we want to achieve, and we have not shared that with the teams yet. And I think we should not because we are not in agreement ourselves. If we share it, we create chaos, without having certainty. At that point we need to gather information and talk to teams.

The other question is, do you share everything at the same time or do you do it gradually. It is hard to find the balance between transparency and not creating turmoil.

Respondent5 (y):

We do this when it is necessary. Most of the time on the spot when it is important that somebody receives it. Lately, we had several meetings to address certain elements of the application or to create a shared understanding about a subject. That is an effective way to share information/knowledge.

Respondent6 (a):

This varies amongst individuals. Most people in our test whether the information they received is correct. So, in that way there is a lot of sharing and discussing of information. But often the important details are neglected, often due to a lack of oversight. Sometimes, especially new team members, do not know how important a module of software is, and therefore what they are doing with it. That is because of the complex environment.

Respondent7 (y):

We do this a lot. Earlier on we talked about knowledge sessions. These meetings we used to get people up to speed regarding certain areas of the things we develop. That is a very useful tool for sharing knowledge.

4.2 Results – Sub-aspects

The relevant information per sub-aspect is depicted in this section of Appendix 4. Paragraph 4.1.1 presents the relevant data per sub-aspect. The second paragraph portrays the more detailed information regarding the sub-aspects of shared goal. While the last paragraph, 4.1.3, presents that for the sub-aspects of shared understanding.

4.2.1 Relevant information per sub-aspect

This section describes the information that was retrieved from the semi-structured interviews, per sub-aspect. In Appendix 4.2.2 is a more detailed overview of the relevant data regarding sub-aspects. The data that was used for table 4.4 is extracted from questions 8, 11, 15, 22, 26 and 30.

IT Leadership

A leader is someone who will also work with the team through the hard times, according to Respondent2. Respondent3 states: 'we try to keep the outer world outside, to give the team the space to work autonomous'. The leadership team is important in enabling teams to overcome blockades and become autonomous. Respondent2 argues: 'guarding culture becomes easier when its healthy. Before that moment we had to lay off some employees. That's also leadership'. Respondent1 describes; 'sometimes a team (member) will drop the ball. Due to feedback of themselves, the customer and me there is growth'. According to the respondents who work in the teams, the leadership team helps them to create a nice environment to work in and gives cadres. Respondent5 states; 'they think about how they can help people to come up with their own ideas'. Respondent6 describes; 'they block parts of the discussion between stakeholders, so we can focus'. The shared goal could be created by the cadres, room and focus the leadership provided for the team. Respondent5 calls if autonomy and inspiration, Respondent4 states that leadership resides more and more in the teams and Respondent7 explains that the leadership team will often ask the teams how they think about something.

Organizational Structure

Respondent3 sees the importance of a tailored organizational structure. They created two flows. Functionally the structure is: team > PO > Respondent3, and technically it is: team > architect > Respondent2. Another leadership aspect is HR, which is handled by four team leads. Respondent3 explains that every team member could pick their team lead. 'That creates autonomy and trust'. Respondent1 explains that a structure is important and it took time to create that. The teams see the team leads, architects and PO's as the leadership describes it. They were to be for which problem. Sometimes they disagree, especially when the PO's or architects decide something that has an impact on the team. The team members seem to associate organizational structure with processes and are clear in their preference for autonomy. If they feel they need a process it will be implemented. Respondent6; 'I always find it hard to determine what you really need around teams'. Respondent4: 'the leadership has a feedback role'. The team members feel like the impact of an organizational structure on having a shared goal is quite low.

Shared vision & cadre

Knowing the vision helps the team. Everybody agrees on that. It creates guidance. Respondent1: it helps our team to know it is about more than technology. Respondent3 states that he would like a

situation where he could point to the vision and the cadres and explain to someone why we can or cannot do something. Respondent2 agrees. They both see the conflict on the unit level as the main cause. The respondents that work in the teams do agree with the notion that the vision is unclear and that a clear vision would help the teams to create clarity. That would help in doing the right things and having a clear shared goal. Respondent4: 'if you work with several teams on the same product you need an overarching vision'. Respondent7 has a similar notion and Respondent5 states; 'the product vision is quite blurry. In the past the customer decided and we don't want that anymore'. Respondent6: 'we often miss a chunk of information, there is always a knowledge gap between the leadership team and the development teams'.

Construction

According to Respondent3, everybody has an insight regarding the strengths and weaknesses of their team members. The team members corroborate that statement. For instance, Respondent5 knows what gives his team members the energy to come to work and Respondent7 explains that they are not only working. They know each other. Similarly, they corroborate the statement of Respondent2, who says that it is very important to share the good and the bad elements of life. Respondent6 explains that it helps in the acceptance of one another. It is important to be aware of the challenges in private life. Respondent1 resumes that it takes some time to get to know each other and really collaborate. The ability to understand each other is deemed important as a steppingstone in creating a shared understanding.

Co-construction

It is important to be vulnerable, to create trust and to have fun. It is a tricky balance, because Respondent3 feels that they might have become too friendly. That might prevent them to be honest to each other and give feedback. Respondent6 recognizes that. However, the importance of this sub-aspects is apparent for the respondents. Especially for the team members. Respondent7; 'we are willing to listen to new ideas and are open for each other'. Respondent6: 'we are on the right track. The changes set us back a bit. The progress would be in sharing more and be open about what everyone feels and thinks. Respondent5 explains that they know how to achieve the results together and know what is required in the near future to keep doing that. That information on a personal and ability level helps them to create a better shared understanding.

Constructive conflict

Respondent1 and Respondent3 state the importance of a constructive conflict. Otherwise you do not discuss what needs to be discussed. Respondent2 emphasizes that the team culture decides how well a team can do that. Team Rubic often does it themselves, while Team 69 sometimes needs the help of a team lead to stir things up. Team members state that they are critical on the content level, but could be more open regarding the personal level. Respondent4 states that they do avoid conflicts sometimes and Respondent6 thinks it is the biggest improvement point for the team. They should give more feedback, especially the critical kind.

Team Rubic versus Team 69

Table 4.1.app shows the bundled answers of the team. As depicted, they have answered very similar regarding the shared understanding sub-aspects and a bit more diverse regarding the sub-aspect of a shared goal.

Table 4.1.app: Scores of sub-aspects – respondents who did not discuss all indicators of the respective sub-aspect

		Team 69		Team Rubic	
		Respondent 4	Respondent 6	Respondent 5	Respondent 7
Shared Goal	IT Leadership	2	1	1	2
	Organizational Structure	3	2*	3	3
	Shared Vision	1	2	2	1
Shared Understanding	Construction	2*	3	2	2
	Co-construction	1	1	1	1
	Constructive Conflict	2	2	3	3

* This 2, is a “double” 2, because the respondent only chose the most important sub-aspect. Therefore, it is unfair to say that 69 had less agreement here.

4.2.2 Shared goal – Sub-aspects

This paragraph is gathered by analyzing question 8, 11 and 15. The most relevant data is depicted per sub-aspect.

IT Leadership

Respondent1: Sometimes they drop the ball. Due to the feedback of themselves, the customer and me there is growth. Employees can make mistakes, but need to learn and talk about how to prevent it the next time. Next to that I have bilateral meetings with the developers. Most of the time it is about showing them what the impact is.

Respondent2: A leader is someone who will also work with the team through the hard times. Everyone who works here, also the non-executing roles, have worked on the change. Lead by example. But it is more than that it is also about creating a way to reflect. We had Dev meetings where people were talking about stuff that did not worked. Yeah... but what can you do about it? Yeah. I am waiting on that guy. Ok... how can make him do it? And if he cannot do it. Does he need help? Do not blame others, start moving. You can spot a good team easily. When there is a problem they start working harder, together. A bad functioning team starts blaming. If you punish someone for making a mistake, he will not make it anymore. That is true. But he achieves that by taking less risks, he becomes less proactive. Afraid. Less innovative. So, what I do? Say something like; that is a nice mistake you made there. So, what now? How will we move forward? Then you see it is not that big of a deal. Guarding culture is easier when it is good. Before that we also had to let some people go. That is also leadership. It is all about the product/platform. You have to create that mindset.

Respondent3: We gave teams a lot of autonomy and challenged them to take ownership over their processes. We try to keep the outer world outside, to give the team space to work autonomous. IT leadership is important in enabling teams and helping them to overcome blockades and become autonomous.

Respondent4: Internally there can be a lot of discussion, especially on unit level. We try to keep that out of the teams. Every team member has a team lead, which you will speak once every 6 weeks. Furthermore, I believe a big part of the leadership should be in the teams. As architects and PO's, we do check what the teams decide. So, if the chosen technology does not fit, then we will intervene. What happens more, is that the job is too heavy for a team member. Then we will try to offer a different role.

Respondent5: Within our teams there are many seniors. With strong opinions. So, there it can be a challenge to agree. We are the most autonomic team within the unit core. Respondent2 and other architects do not have a big say in what we make. Respondent2 really helps us to be able to make our own stuff. We are responsible for making something from scratch and want to do it in a certain way. The team leads are people who have a say on the high-level, regarding product vision or the vision of the organization. They do not manage the teams, but they do also some HR and education elements. How can we keep it a nice environment for stubborn developers? That what they try to achieve. They also think about how they can make people come up with their own ideas. That really improved this year. I am satisfied with that.

There are a lot of stakeholders within our organization, with influence. That can result in friction. The swift towards more autonomy really inspires us, team members, to come up with smart ways to improve ourselves. Before that it could be a buzzkill when you needed to do something where you did not believe in. There is also an PO+A meeting where they talk about functionalities, epics and bugs. They also plan for the quartal.

Respondent6: I think we have a leadership team, with high technical capacities. There are several technical influences. This helps us as a team. We are easier understood. Also, about what is the next logical step. This also fortifies the team. We feel support when needed and we do not have to convince them of obvious things that are important. That is often different.

Besides that, people like respondent2, 3 and in lesser amounts respondent4 keep the political discussions away from the team. We have no bother of the discussion between units. Some stakeholders want different things for the platform than us. They block that for us, it does not slow us down. They are also people we can spar with. They are technical and therefore they can help you to get out of the details and to check control whether it fits the bigger picture. In general, they leave it up to you. They will not fix stuff for you, they let you fix it. So, they also expect that we take the responsibility. If it is not good (enough), they will let you know. We will not be turned in a scapegoat, but they will let us fix it. Sometimes they will have the conversation with the customers or stakeholders about the problem. Especially when it is a tough moment.

Respondent7: The leadership thinks about the cadres and they also form a block against stuff that will slow us down. Also, our product owner, whom I see as a leader, helps us a lot in that regard. Stakeholders that ask stuff like whether we are performing well is kept outside the team. PO+A will discuss about bugs, feature requests et cetera. If it is sometimes that we will do, then it come in the sight of the teams. Also, innovative ideas, that are not from the team, are handled in that regard. We can be involved earlier, but that will happen when somebody knows a lot about a certain topic. That is why it does not feel like a distant team that decides for us. They do not think they know it all by themselves

BLT is also a form of leadership. They are team leaders. Respondent2,3,4 and another team lead are in that team. I do not know a lot about what they do there. One of them is also in our team and my team lead. He says yeah, we do Bacon Lettuce Tomato, and then we know he will be gone for a

while. They do a bit of coaching, especially for new employees. Because my team lead is also in my team, I do not really see the difference. I can ask budget for a certificate by him. Aside that it is just a developer who works in the team, like any other.

Part of the leadership is in the team. If an epic comes in it is almost empty. We will figure out what to do and decide which user stories to make. A while back PO+A would think about the epics and make a lot of decisions. It took a lot of time for them and we had a lot of comments on their choices. So now we do it ourselves. It takes more time for us, but we also have more freedom. Although we need to discuss it with PO+A still. They more and more bring us the problem. That activates more.

Organizational Structure

Respondent1: The structure has been growing internally. It was a big shift. Now it is good. We have interchangeable developers, although everybody has its specialism. Some guys like test automation, others are technical cracks or like to use backlog and deployment tools.

Respondent2: It is an organization without a lot of hierarchy. We have three teams (two in Holland, one in Romania). Four guys in team lead roles, that is me and respondent 3 and two other guys (an architect/a PO). Every team member has selected its own team lead. That gives autonomy and trust.

Respondent3: Functionally: Team > PO > me. Technically: Team > Architect > lead Architect. Both connected to an epic to help the team. If I get involved it is often blockades/insecurity etc. More and more the architects and PO's are in the lead for leading the team. PO&A is a meeting where we decide what we make. Me and lead architect also a team. And we try to keep each other sharp. We do not want more.

Respondent4: The organizational structure is pretty good. I think that our department is doing the best within the whole organization. That is due to the mature teams and the way we organized it. Teams can decide a lot. If it is bigger, it will cost more and other will also need to think about it. Teams have room to make technological improvements. Lately both teams had to select a tool. They both chose a different tool, which normally would be fine. In this case, one of the tools had an integration problem with Windows. That is a moment when leadership will say, this is not the right tool for the job. An intervention. Otherwise the teams would have been free to choose.

Respondent5: They always say we are a flat organization. I think that is true. Especially in the team, we have no order there. Also, company wise, it is not that bad. It not that responden3 says we will go and do this and that the company follows. Sometimes, for instance the PO+A team, will decide something that does not fall well with the teams, but I think you will always have that. If we really disagree, we will let them know, often our PO is quite capable of doing that for us.

Respondent6: I always find it hard to determine what you really need around teams. Sometimes I feel like people talk too much about stuff, while you know you do not want it as an organization. Most of the time this happens to keep people happy. That could be more efficient, less hierarchic. Discussions over functionality, what not to do, it could be simplified. We could have more people with a real vision, who could set the course. We might miss that now. Furthermore, I do not need more structure, everybody knows what they are responsible for. That makes it a lot easier. The organization is flat, especially in products (part of core). I am glad that our department does not have success managers and other vague functions. I know that respondent1 has them around his team, he might think different about it. And he has a different perspective as a manager. I am glad we do not have them. We handle the accounts ourselves and partly through the leadership.

Respondent7: The only structure in our team is: we have a PO and he brings in stuff. Within the team there is no real structure. Everybody knows what to do. First, we had a Scrum Master role. In 2016 that was also necessary. Nowadays Scrum is more integrated in our team and the scrum master role is more the person who prints out Jira cards. Or prepares a few meetings, that is it. There is one person who takes the lead in that. But if he is not there, we will keep doing it as a team. Everybody knows where the equipment is and knows why we do it. The other structure is the PO+A team around the DevOps team and the BLT team. We discussed that already.

Shared vision & cadre

Respondent1: We want products of a high quality/value for our customers. Knowing the vision helps the team. They know what we want and what the customer wants. Our vision of how to make products needs a certain way of working. The team members also have more consciousness. It is more than technology.

Respondent2: Not enough. Due to conflict at the unit level. I think that the party that develops the product should have the vision. So not from units like public and finance. We want to help teams with connecting to customers, but platform vision. It should be ours. It should be carried by the teams. It should be about accepting what product is and is not. So, it also about saying no. I like that, most do not.

Respondent3: It would be the best situation if somebody asked me a question to make something and I can say no we can't do that and you should have known that because this are the cadres we agreed upon. Nowadays that is, sadly, not yet the case.

Respondent4: Shared vision is important. If you work with several teams you need an overarching vision. That is partly fulfilled by the PO+A meetings. It is important to have a course, otherwise it will not work. Teams also know what we find important. At least for the coming months-year. We still need to discuss more over the period after that. On department level we have the goal to keep the teams together as they are. There have been a lot of swifts lately and a little peace and quiet is required.

Respondent5: The vision is definitely shared. Since last year, actually. The team leads also talked about objective key results. The vision of the company comes also back in that kind of meetings. The same vision that should attract our customers and which explains our added value. The product vision is a bit blurry. People are trying to make that clearer. In the past the customer wishes were leading in that, which we do not want anymore.

Respondent6: I think we have a shared vision. Most of the time we agree. It can happen that teams do not receive all the required information. There is always a little border between the leadership team, PO+A team and the development teams. That remains difficult. Team members sometimes miss a little context and history, of discussions that led to a decision. That can lead to questions and misunderstanding. We will remain critical as a team and often challenge the PO+A team or leadership team about their decisions. We will not build blindly.

Respondent7: If I look at the company and think what the vision is: I do not know. No idea. It is, of course, not something that occupies my day. I like the work. The vision of the product that we develop is more important for me than the vision of the organization. The product vision is also shared. What we are building now should make it easier for the employees/customers that work

with our products. We share that goal. While we are migrating to the new application, we are really trying to improve it.

Back to the company vision. I do not see it as my job to know. Other will think about it, I am sure. They know better, probably. In the three years that I am here I heard 3-4 different kind of visions. So, I do not care. As long as the ideas of the product are good and my ideas are heard. And they are. If someone has an idea, we can always spar about it.

4.2.3 Shared understanding – Sub-aspects

This paragraph is gathered by analyzing question 22, 26 and 30. The most relevant data is depicted per sub-aspect.

Construction

Respondent1: It starts with why you bring people together. Then you want to go get to know each other and collaborate. That is quite the process. The team takes more time during retros to talk about values and opinions then at the beginning.

Respondent2: It is not only Instagram life. Sometimes life is hard. Respect that from another. This is a big core value. You will not be accepted here if you would cross that line. We stimulate that people share. If someone is closed by nature, we help him, ask him to share stuff, at least with the team. It is important for people to know what they can and cannot do with someone. If you know that from everyone, you are good.

Respondent3: Everybody has some insight in what the other team members do and what their strengths and weaknesses are. An example is that I asked the teams to have feedback session for the evaluation and that they all helped each other grow. They organized it and did it their own way. I don't need to know about that. I will see it second hand, in self reflections and I really like that they did that.

Respondent4: Team members have a good image of each other. That started a few years ago. Back than collaboration was not that normal. Those things start with yourself. Getting and taking ownership.

Respondent5: I work more than two year in this team. We got to know each other pretty well. Not only business wise, but also privately. We are very open. Also, about their interests. We also tried to answer for each team member; what gives you energy. With the goal to focus more on that. That is important for yourself and the team. Otherwise the fun is quickly over. Team did not change much. Responsibilities are clear, we correct each other when needed, ask and help each other. We could listen more.

Respondent6: I think we are pretty open. So, we know a lot from each other. Not only the work-related stuff, but also privately. How somebody is doing and what is important in his or her life. That helps with the acceptance of one another. If it is only about the business, you will be more formal and might not accept some of the challenges that people face in the private life. If you also appreciate the person you can handle that in a different way.

Respondent7: Well, that is funny. At the moment, I am busy with a coaching process, in an attempt to learn myself and why I do certain things and what I like about the work. So, I am starting to

become more aware of certain things. This is also something that my colleagues see. They see the things I work on and they say it influences the daily activities. So, my colleagues notice what I do, what my behavior is and what my strengths and weaknesses are. And, in a similar way, I know my colleagues. We are not only working. Like we just come in the office, make 8 hours and go home.

Co-construction

Respondent1: There is a very nice dynamic in the team. They lunch together, have fun, take nerve guns to the office. This environment is growing and they fun in what they do. Everybody sees that it is a real team.

Respondent2: Show yourself. So also, the misery. Showing success is easy. Show how you are when it is bad, what is going on. If you can vulnerable like that than the shield is gone. Creates trust. These discussions take place in several occasions. Retro's, 1-1's or if you know something is up just when it feels good during worktime.

Respondent3: We are in a phase where feedback could be more direct. We became too friendly. So, I'm teasing that now. We really created teams and maybe forgot the opportunity to learn between teams. So, I'm considering challenging them to give feedback from one team to another. We also did personality tests with the teams. So, they have in insight in that. We also gave them to room to pick that up themselves and use when they search for a new team member. We also did a training in how different people are. Who has which strengths and how can you use it? That kind of stuff.

Respondent4: Processes are embedded in the teams. We are at the point that, for instance, a full-time Scrum Master is not necessary. If the team is mature in that regard, like we are, then that can be done more efficient. Also, here it is about giving responsibility.

Respondent5: We know what we can say to each other, and what not. We are able to prevent collisions which are unnecessary. We are sober/normal and know each other for a while. So, we know what we got and what we can achieve together. And more importantly how we can achieve that. We know what we can say to each other, and what not.

Respondent6: I think we are doing this better and better. There are some steps yet to take especially because some of the team members are relatively new. From the 6 people that we have 3 people have changed at the beginning of 2019. From the original team in 2016 only me and one other colleague remain. So, 1/3 of the original team and 1/2 of the team changed this year. The progress will be in sharing more with each other and to be open about what you feel and think. Also, if (you think) somebody does not want to hear it.

Respondent7: Within the organization there are many employees that see each other out of business hours. I do not do that with many teammates, but I do have that relationship with some colleagues. Within the team we try to create team days. That helps in getting to know each other out of the work environment. Within the work environment we are willing to listen to new ideas and each other. We are open for each other and we can challenge each other.

Constructive conflict

Respondent1: There is a feedback culture. Sometimes it can be very clear, in the beginning that can be exciting. The good thing of this team is that they hold each other responsible for agreements that

they made and show respect for each other. All elements that we discussed are important, but I think constructive conflict is the element that made it a real team.

Respondent2: Depends on the team. Rubic is very good at this. They have the culture of holding each other accountable. 69 often needs one of the team leads to intervene somewhere in the process. If they find it hard to talk to someone or to give feedback, we will help you. But try it out and let us know how it went. Team leads are not about solving the conflict but about managing and guiding the process along side of it. People are important. And people will have conflicts.

Respondent3: Sometimes you need conflict. Otherwise you do not discuss it. So have the conflict as fast as possible. Without this you cannot do DevOps. So once a week it fine to have conflict. It should be with respect though. Within the teams this goes pretty well. Also, the respect part. From the teams toward the management / leadership, this could be better. Sometimes they are not seeing how hard me and lead architect have to work to give them the autonomy they deserve.

Respondent4: We could be more open and critical. Especially when it is not about content or when things are hard. We might avoid the conflict sometimes.

Respondent5: We can have conflicts during refinements. We talk about solutions and we have different opinions, so yeah, we are strong enough to have good discussion about that. An informative content focused discussion. So, we will not be personal about it. We will be hard on the content and have a discussion.

Respondent6: I think this is one of the big improvement points for our team. We could really do better. We would become a better team and better individuals if we would do this more often. It can help to be critical and to not walk away from feedback.

Respondent7: We covered this also in other questions. On the content we challenge each other constantly. In such a way that we keep having respect for the individual. Sometimes conflicts are of another nature, more about the personal level. As described in other questions we do that too, but could do it more.

4.3 Results – Main-aspects

The relevant information per main-aspect is depicted in this section of Appendix 4. It contains two paragraphs. 4.3.1 is summary of the relevant questions for the main-aspects. 4.3.2 is the basic information that was used to create 4.3.1.

4.3.1 Summary per relevant question

This paragraph presents the most relevant information of questions 2, 3, 5, 18 and a follow-up questions; what is more important shared goal or shared understanding? The summary is per research question and based on the data from all respondents.

Semi-structured interview – question 2

At question 2 of the semi-structured interview the respondents are asked to elaborate on the collaboration in the cross-functional team. The respondents are very positive about the collaboration in the teams. It is interesting to stress that teams seem to have different maturity levels. The team of Respondent1 has just been working together for over a year. In the years before that moment there was no internal team and developers were working at the customer side. The teams of Respondent2 and Respondent3 exist for over three years. And between the two teams there is a big difference in terms of experience and personnel changes. One is very experienced and had few changes of team members, while the other team is unexperienced and had many changes.

Respondent1 is talking about the positive influence of having the team and how hard it is to develop a team, which makes sense if you take in account how long the team exists. Respondent5 and Respondent7, team members of the experienced team (Rubic), talk about autonomy and ownership. While Respondent4 and Respondent6 talk more about the way of working and how the team (69) can be more effective. Both is understandable if you look at the stability and experience of team Rubic. Team 69, however, had many personnel changes and is reestablishing their principles.

It is also interesting to address that Respondent2 and Respondent3, the leaders of the two teams, perceive two different team cultures. The experienced team, Team Rubic, is perceived as a team that is focused on speed. The other team, Team 69, seems to think about quality first. Respondent2 and Respondent3 are very pleased with the team cultures. Both cultures are commendable and the leadership team believes that the culture helps the teams to perform. Respondent3 states that Team 69 was able to keep performing through all the personnel changes due to the team culture. The teams can also learn from each other, because they have different strengths and weaknesses. Respondent3 explains that the teams were formed based on the preferences of the employees. They had to answer how they like to work and with whom. That was the input for the team layout. The teams get as much autonomy as possible, Team Rubic lately hired a new team member themselves. Respondent3: 'if they hire the new team member, they are responsible for the success'.

All the respondents with a management role stress that it took some investment to get where they are now and that it is important to show what the effect of that investment is. Respondent1 explains that they became owner and therefore they needed a different way of working. The business unit Public started to implement DevOps and became more aware of costs. Business unit Core had a similar transition. They had the dream to be able to release at any moment. This created a whole new way of thinking and created many questions, regarding: the server park, build servers, team autonomy, CICD, non-functionals, monitoring, provisioning, containerization et cetera.

DevOps seems to demand a certain mindset. That was not present at the business unit; core, in 2016. Many of the respondents started to work for the department during that year or quickly after the start of the transition. They had a big impact on the business unit. Respondent2 and Respondent3 were amazed by the mindset of the teams in 2016. There was no passion for the work and failures were accepted easily. Respondent3 mentions an example where a team had a very bad sprint. They said something like: 'O well, we didn't make it' and went back to their desk. The employees that started working there, due to the transition, brought a different mentality. Phrases like; 'if you don't fix it, your colleague has a problem tomorrow', became the standard. The leadership team created a department culture where giving feedback was the foundation.

Respondent4 states that two important changes were division and clarification of responsibilities and more tools and freedom to take care of things themselves. His team member, Respondent6 adds that the most important things is that every team member is all round. Everyone has a preference, that's fine, but team members need to be able to replace each other. He describes the complexity of the work they do; a very abstract and complex platform, operations, development, backend, frontend, CI/CD, deployment models, testing et cetera. He explains that it takes time to perform well as a team member. Due to the recent personnel changes within his team he thinks that they are not able enough to replace each other at this moment.

All the respondents that work in the development teams state that the aforementioned dream had a big impact. It was an enormous stretch for the organization, but created the discussion on how to achieve it. After a while releasing became a team responsibility. Another thing that is mentioned by several team respondents is the improvement of development environments. The dependencies between teams were reduced by adopting more (test) servers, more environments, adoption of CI/CD and smarter build management. Respondent7 recalls that automated tests took too long. He remembers a moment when builds of Tuesday were still running on Thursday morning. He also sees the impact of the improved collaboration. First, they needed a work process, a rhythm, in order to get things done. Now they mastered the rules and know when and how to break them. Respondent5 and Respondent7 explain that they know very well what each team members likes and can do. This helps them in being an effective team.

Semi-structured interview – question 3

At question 3 of the semi-structured interview the respondents are asked to elaborate on the impact of DevOps on the way the teams collaborate.

Respondent1 explains how they had a set up a new way of working. One wherein the customer plays an important role. Feedback loops are implemented. Quality feedback loops that help the developers to improve what they build and session with customers to ensure early feedback on the product. The team has been using DevOps since the start, so biggest difference is the change from project to product. However, none of the developers was used to work like this. Since the team exists customers and employees are more satisfied. Customers state that they feel the team is in a better position to help them. Respondent1 likes that the team can handle more (diversified) work, then 6 individuals; 'they can replace each other and it's scalable.

Respondent3 explains that the DevOps way of working was implemented during the transition. He sees the dream to release whenever it is necessary as an important driver for success. Teams needed to think up front about branching strategy, testing, prioritization of functionalities. It also was the starting point for a different mindset, one based on passion. 'Be proud of your work'.

Respondent3 describes two important roles in that process. One he called the police, like Respondent4 and Respondent6. They will enforce the principles. The other group challenges the police, when the rules do not apply, like Respondent2 or Respondent5. The team is responsible for the whole process, from development to production. Quality is also an important factor. If the team needs a tester, they can utilize that, but the whole team is responsible for the quality of the product. Respondent2: 'they will need to think about a way to ensure quality; unit testing, performance testing, regression testing is all done by the team'. Testers followed a java course and developers learned to structure their code in a way that is testable. Respondent4; 'a team needs people with a quality mindset and operational mindset, if that's a developer that is also fine'. The most important question according to him is: how do you form a team and who do you hire?

Respondent2 and Respondent3 stress the importance for other things than functionality. They created a development mix, which gave teams the freedom to work on technical debt or improvements. There are several backlogs: one technical, one functional et cetera. The teams need to ensure that they use their time in a fashion that relates to the development mix. The leadership team guards this freedom and explains it to higher management.

The team has several channels that feed them work. One is customer support; the issues are discussed with the PO's and architects. If the priority is high enough the team will hear about it. Another channel are customers. You could also call the other business units a customer. Respondent4: 'we try to focus on the product vision and to communicate about that with the other business units. As everywhere, there is more desire than resources. So, we need to make choices.' The epics come from the Roadmap, often created by product owners. Furthermore, there is a technical Roadmap. The ideas can come from almost everybody.

There are several benefits of the way that the teams collaborate nowadays. Several Respondents (1, 2 and 5) stress that the collaboration with the customer has also improved significantly. Respondent1 and Respondent3 describe another benefit, employee happiness. Respondent1 describes that, due to the shift from project to product orientation, they now have the copyright and the maintenance responsibility. That creates the opportunity to re-sell the created module. Respondent2 likes the increased knowledge sharing. 'It is often obvious whether somebody likes Dev or is really Ops-orientated'. That is substantiated by Respondent5 and Respondent7, who state their preference and its impact on the team. By removing traditional barriers between these roles, the pain becomes visible and is discussed earlier in the process. This also creates friction and discussion, which is good according to Respondent2; 'they will start understanding each other'. One example is that the teams now maintain 75 servers themselves, while reducing the costs significantly. The need for operational procedures became apparent to developers.

Respondent2 thinks that it is important for team members to see all the aspects of their work. So, a developer should see the impact of their work by; implementing it at the customer's side, monitoring it in production or testing it when there is an incident. This, together with experience, creates a mature team. A developer should know that it is not enough to build a great feature. If something goes wrong, you will wish that you thought about error handling. Respondent3: 'It's interesting to see that the teams both deliver quality, but have different starting points'. You can see the nature of a developer during a code review. Respondent6 will focus on test cases, while Respondent5 would focus on code structure. In the end the leadership team tries to place the team members in several situations. This will help them to become aware of the impact of their work. One of the changes since 2016 is that the developer has to be billable for a few hours a year. That means that they have to implement their own software at the customer side.

Most respondents see that the effect of DevOps is teams with more ownership. Respondent4; 'they feel more motivation to start doing something, to create. That starts with responsibility'. Another element that several respondents appreciate is the interpersonal relations outside the office. They became close and people tend to stay quite long. Respondent4: 'we have a complex platform, so that can be positive'. Respondent5 adds that bottlenecks became visible. 'We automated some of the pain away and introduced processes where it was absolutely necessary'. That gave them the room to create more autonomy. That started with making things more manageable, according to Respondent5 and Respondent7. Technically speaking, for Respondent7, the turning point is the moment they got their own cloud environment. That was an enabler for more control and responsibility. Secondly, Respondent7, states that they focused on reducing single point of failures. They often did this by creating knowledge sessions, where the expert shared expertise with the other team members. Last, but not least, Respondent6 contributes that they work more as a team now. The team feels the responsibility to ensure that they deliver a good product, that will do as required. Some big improvements were made, like the quality of the build street or the speed of delivery. Respondent6: 'the reason for the changes were the new people. We came and did not accept it and thought about how to improve it. After that, you just have to do it'. They brought the mindset and the knowhow to make the change possible. Respondent6: 'an example is that the lead time has improved tremendously. Now we can fix a bug and release it within hours, while three years back it took a few days'.

Respondent2, Respondent3, Respondent4 and Respondent6 also state an important factor. The bar was raised, which meant that the "old" team members needed to adapt. This was not something that everybody could or would do. Some went on their own account and others were laid off. The people that stayed feel empower now. Respondent6 really likes the way the leadership team supported the change. They gave them the freedom and time to deliver on what was demanded. They supported a culture for continuous improvement. The old management measured on bugs solved, the new management demanded more. They wanted quality improvements. Respondent6: you get what you ask.

Semi-structured interview – question 5

At question 5 of the semi-structured interview the respondents are asked to elaborate on the shared goal of the team(s). The respondents that work in the teams elaborate on their own team goal, while Respondent2 and Respondent3 are able to elaborate on both teams. Respondent1 will talk about his team.

For respondent1 the ultimate goal is successful reusable customer implementations. 'We try to create a situation where the customer pays for initiatives to improve the product, payed R&D'. To do that they have to help the customer. 'We help the customer to place their customers in the center, by empowering the employee'. The team does that by supporting the customer journey, through dynamic case management. So, the shared goal is empowering the employee of our customer by making successful and reusable dynamic case management modules.

Respondent2 and Respondent3 have a similar view on the shared goal that Team Rubic and Team 69 have. There also is a team in Romania, which focusses on libraries and security. However, in the semi-structured interview the focus has been on Team Rubic and Team 69, because they were part of the DevOps transition.

Team Rubic has a very clear and demarcated shared goal. They are responsible for replacing the old studio. Respondent2: 'it creates energy and ownership when you give a team that level of autonomy'. They decide upon everything themselves, also architectural principles. Of course, they need to be able to explain why they choose for something. Respondent5 and Respondent7 are very pleased with the level of autonomy. Respondent5: 'since Q2 of last year our shared goal is very clear. We are, since then, building a new platform. Before that it is was quite scattered. All over the place. We had no sense of ownership or that we could exercise our influence. Now we have that ownership. We feel the autonomy and it helps to stand for something and to really go for it. At that point you will also really start discussing about what is the best way to do things'. Respondent7: we have a few cadres and within that we exercise our autonomy. We do use others for their knowledge and experience. Ask feedback when we have a concrete piece of the puzzle'.

The organization would like to speed up the development of the new studio. Therefore, they might need an extra team. Respondent2: 'that will probably create tension, because of the proudness. The team came up with the solution to split their team into two teams and help the new team members grow faster'. Respondent7 also mentions this option; 'when everybody is on par, we could switch back to the normal teams'. The leadership team likes it when the team comes up with a plan. It creates accountability.

Team 69 has focus on Runtime, there are several components that require attention and per quartile they focus on one of those. Respondent3 describes the year goal for them as; 'work on the Runtime and improve the overall performance and the process engine. According to Respondent6 that makes it harder to concretize the shared goal. He describes it as; 'deliver generic, robust building blocks that are multi-usable and deliver value for our customers and colleagues'. He explains that there are many types of customers; business engineers who work for the customer (colleagues), customers, customers of customers and other business units. Respondent4 also sees a difference between with Team Rubic. He focusses mainly on the department goal, which is deliver a high-quality and well-functioning platform. He adds that the development mix is also an indicator of team goals. Roughly 50% of their work is focused on functionality, which means that other things like support, technical debt, non-functionals and innovation are also very important within the teams.

Respondent3 explains that they work in quartiles and that it is often clear who will work on what. New features are often handled by Team Rubic. While the complex puzzles, for instance in relation with a mainframe from a bank, are handled by Team 69. The team in Romania likes to focus on security. He states that based on the shared goal and preferences of the team everybody knows from each other who is most fit to pick up a story. That identity is important according to Respondent2 and Respondent3. They also both mention the departmental goal; develop a high-quality platform. Another departmental goal is the technical Roadmap, which has been a facilitator for making the dream a reality.

Several of the respondents that work in the teams see a complicating factor. They are often in agreement within the teams, but have a hard time aligning all the stakeholders. Respondent5: it can be a challenge when several customers, or other business units, are trying to influence the way we built the product. Respondent4 and Respondent6 see that same effect. While Respondent6 also points out the legacy; 'we still experience the pain of past decisions'. It is interesting to mention that all the respondents who work in the teams like the complexity of their work.

Semi-structured interview – question 18

At question 18 of the semi-structured interview the respondents are asked to elaborate on the shared understanding of the team(s). The respondents that work in the teams elaborate on their own team understanding, while Respondent2 and Respondent3 are able to elaborate on both teams. Respondent1 will talk about his team.

According to Respondent1 the shared understanding of Public DevOps team is quite high. He thinks they struggle the most with the prioritization. If he would have the pinpoint one improvement point it would be naivety. The team members are too eager to help the customer, but they should try to find a shared understanding on how to handle a new request. Otherwise there are too many back and forwards, that could have been prevented. That is really a how question, so something for the team to solve. Respondent1: 'how can we help them in the most effective way?'

Respondent2 like the private contact. 'For me that's one of the benefits of working here'. He is corroborated by Respondent4 and Respondent5. Respondent2: 'It is not only Instagram life. That is an important element of our culture. Respect for each other problems is a cornerstone. If somebody would disrespect that, he or she would have a hard time here. Lately, one of our HR employees told me that she is always overwhelmed when she arrives at the office. Her mailbox is full, people at her desk et cetera. So, what can we do? Now she comes at the office at 10:30 and everybody knows do not call her between 09:00 and 10:30. Power to you. We try to create a situation where that is possible. That requires transparency and a culture of acceptance'.

A point that could be improved is the level of feedback. Especially Respondent3, Respondent4 and Respondent6 see that necessity. Team 69 has had some personnel changes, so it might make sense that they need to reinstate that practice. Furthermore, Respondent6 sees that not everybody is willing or able to meet the quality norm. Some of the team members are not capable yet to see the bigger picture and therefore introduce mistakes. He understands that and blames the complexity of the platform.

Most respondents like the transparency in the organization. Respondent5; 'if something is going on in your private life, we will discuss that. We are very transparent'. Furthermore, they know what everybody likes and what they want to achieve. Respondent5: 'at the end, it is important that you can do what gives you energy. That helps you and the team. Otherwise it will not last. Respondent7: 'some of the team members inspire other to grow'. He claims that sharing is a very important factor, to be able to replace each other.

Respondent3 likes how the teams will investigate whether they need things to keep their shared understanding on the required level. He recalls an example where one of the teams asked him for a junior, to attain a certain balance. 'Their own idea based on a need. Sometimes I feel like they are more equipped to look into the future than us'.

Shared goal or Shared understanding?

Every respondent was asked whether shared goal or shared understanding is more important. The answers show that everyone sees the impact of both main-aspects. Respondent1, Respondent3, Respondent4 and Respondent6 prefer the main-aspect; shared goal, while Respondent2, Respondent5 and Respondent7 prefer the main-aspect; shared understanding.

According to Respondent1 it starts with a clear organizational purpose, things like grow in NMR or scalability. If that is clear the team can create a shared goal, within their cadre. Elements of that cadre are quality controls and sharing knowledge. If that is in place the team can become more

aware and move towards shared understanding. Respondent1; 'we did that in phases. The last phase is a process and takes a lot of time and effort'. He describes it as a sequential process, wherein shared goal is the most important factor. 'Especially in the beginning'. There should be a need to assemble the team. Respondent4 agrees by suggesting that; 'you know what needs to be done when a goal is clear. If a goal changes everything will shake'. Respondent6 states that without a goal there is no need to create a shared understanding; 'what are you doing? Without a goal you're without guidance'. After that clarity you can start building a shared understanding. That starts within the team and should grow with their stakeholders as well. Respondent4 and Respondent6 state that shared understanding takes time, you can work on that as a team.

Respondent2 has a different notion; 'shared understanding has the biggest impact. If you know each other and the benefit of working together well, then the goal is secondary. However, he stressed that it works the best when you do it both. Shared goal is a way to create guidance and a kickstart. Respondent2: 'therefore I think most managers will say shared goal, but shared understanding is harder to achieve and it's more fragile. You can always create a new goal'. Respondent5 adds that the goal has not always been clear. That they created that bottom-up for their team; 'the autonomy lies with us and we created a long-term goal based on the clarity that came from shared understanding'. Respondent7 refers to a statement of one of his colleagues; 'I could do an uninspiring job with my current colleagues'. A good team helps each other, so you will get through the uninspiring tasks. Both Respondent5 and Respondent7 clearly prefer shared understanding, but they do see the benefit of clarity that a shared goal can provide.

Respondent3 states that a shared goal and a shared understanding have to intertwine. To achieve that you need facilitation, leadership and ceremonies with a clear purpose. Leadership can help in creating cadres, room for teams and individual needs and culture. Respondent3; 'shared goal versus shared understanding is a management-team balance. For me, the shared goal is slightly more important, because it is the starting point. It has an immediate effect, while shared understanding is a process. Nowadays the goal is clear and new team members can fast forward to shared understanding. The team facilitates that acceleration'.

Respondent3 and Respondent6 missed something on the level of shared understanding and shared goal. Respondent3 states the importance to communicate the effect of the DevOps way of working to the organization. This could be translated as, shared success. Respondent6 misses the inter-team collaboration and inter-business unit collaboration. The team entity and the business unit entity could collaborate better. Respondent6 described it as; 'we should share more knowledge. We are working in the same code base! It could reduce mistakes. So, the collaboration between teams could really improve. Furthermore, the visions of business units are so diversified. They are really doing different things. The leadership teams of every unit should align on the most important aspects'.

4.3.2 Overview of relevant data

This contains the information that was used to make 4.3.1. The data is translated from Dutch to English, shortened and organized per respondent.

Respondent1: Summary

Question 2

The collaboration between the customers and us is quite interesting. We try to introduce BusDevOps, so focus on performance, optimizing, server settings, operation and continuous feedback loops with the business. The team has a clear task. If a customer wants something and central R&D (business unit; core) will not make it, they have to make it. They are responsible from need to production. Often small modules.

Previously a developer would be at the client site. Nowadays, they form a team. Authors right is for the customer, as is the maintenance responsibility. We wanted to change that, so we can be scalable and have the option to re-sell. We positioned a product based on the needs of the customer. That changed our requirements and DevOps was introduced. We wanted automated pipelines, continuous testing, version control, documentation, maintenance, support and continuous integration & delivery. Customers also expect that. I am content with this way of working. We deliver value fast to our customers and we own the product. The option to sell it to other customers makes it financially interesting.

The quality mindset is important. We have the same quality rules as central R&D, because the functionality needs to be able to integrate in the main product. Our team improved in automation, so we can deliver the product fast. We implemented test automation, container technology, monitoring, continuous deployment et cetera. Nowadays, because it is a team; the knowledge is spread, code reviews are standard and we can help more customers at the same time. We also try to search for collaboration with the customer in Cloud and container technology. All to deliver faster, have more stable software and reduce costs. In the near future we should be able to create an environment within minutes, test the software and deliver it with the version we want. At that point we would not need an OTAP-street and provisioning would be automated. In the ideal scenario we offer everything from the cloud and influence the upgrade schedule, while customers will not have to migrate after every big release.

Question 3

Demand can come from the customer or be created by us. Firstly, we decide if it is something that we can or want to make and which business unit should do the work. After that the process is like this; requirements, business case, go/no-go, refinements, development, 0.1 version, feedback, development-feedback process (with sprint reviews at the customer site), most valuable product, user acceptance test and delivery. After that moment we build on it with the priority is high enough.

There is one manager, he is responsible for hours and money. The other team members are developers (1 junior, 4 mediors and 1 senior). Team is working Kanban due to the maintenance emergencies that can occur with several customers. What I like about the team is that they can replace each other and its more scalable. Of course, everybody has its strengths and weaknesses, but everybody is willing and able to work on several items and customers. They also learn from each other and will focus on first things first. That is easier as a team. The DoD is sharp and clear, it helped us to improve on automated (regression) testing. Retrospectives, also quarterly with the customer, helped. We implemented automation actions on the parts where we had problems. Customers are very satisfied with the new way of working since DevOps. The product is more tailored to their wishes. We are also more capable to estimate the work and learned to incorporate the entire process in that estimate. Customers also see that we are in a better position to advise them, because every team member sees more customer environments. Another benefit that I like is the employer

satisfaction, they are more satisfied with their work. I think these things are linked. The customer is happy and therefore the team has success, which reflects in their employer satisfaction.

It has been quite the investment to change, especially in terms of technology and process. For instance: internal GIT, regression testing, documentation, compatible with central R&D, monitoring et cetera. We did that in phases, aside the customer wishes. Now we are at a point that an extra investment will not necessarily lead to a better result. That is a balance.

Question 5

The ultimate goal, from my perspective, is successful customer implementations. Every project has technical support and we develop new features. Furthermore, integrations and delivering. These actions need to deliver maximum value to the customers. The team is an important link in that process. I try to achieve a situation where the features we make for one customer are sellable to other customers. That comes from a financial component. We try to create a situation where the customers pay for initiatives to improve the product, payed R&D. It should help customers to get their business processes in order, because then we both win. We have resources to make that happen. We achieve that by the central R&D of Core and our specific features.

We try to help the customer to place the citizen in the center, by empowering the employee. If he can be more effective there is more time for unique cases. At this point in time civilians are often generalized and employees need to do repeating tasks to push papers. We want to automate the repeatable elements and free up time to treat every citizen in a unique way. We excel at that, supporting dynamic processes. No standardization, supporting citizen in their customer journey by dynamic case management. Use the standard technologies for the back-office and integrate us on what you already have. Our team is looking for ways to achieve that for our customers. To help them grow, to a situation that supports that new way of working, in a safe and economical fashion. That is the shared goal of the team.

Question 18

The shared understanding is quite high. I think they struggle the most with prioritizing. Sometimes they are also too naïve and help the customer too fast. That can result in a many back and forwards, which is not effective. So, I think they could improve on having a shared understanding on how we approach customers and how we can help them the most effective.

Shared goal vs Shared understanding

Both contribute to collaboration. The shared goal was, in our case, necessary upfront. It was our motivation to create the team. Shared understanding is the element that creates a team, because when they started that were 6 individuals. For me, that is not only the internal collaboration, but also the external communication. Both improved a lot.

The team is aware of what we want to achieve and therefore they can create a clear shared goal. If that is clear, then it is easier to build to a shared understanding. We, the management, try to facilitate that process. Some customers can go faster than others and we try to help in those

conversations. In an attempt to find the right balance per customer. With one of our biggest customers we had a big success, that also helps in clarifying the goal. It becomes tangible.

For me it starts with a clear organizational purpose. Things like; grow in NMR, scalable in projects et cetera. If that is clear the shared goal of the team is easier to describe. Things like improve quality, share knowledge, remove single points of failures become the elements of a cadre. At that point, a team can start growing and becomes more aware. Looking back, we did that in phases. We started with the shared goal and then moved towards the shared understanding. The last phase is interesting. That is a process, which takes a lot of time and effort. There has been a lot of frustration; not enough protection for the team, no prioritization etc. If that improves you start connecting with the environment. I think shared goal is more important, especially in the beginning. The driver for success is shared understanding. First internally and secondly externally.

Respondent2: Summary

Question 2

In general, the collaboration is good. I always find the term interesting DevOps. It is often very obvious whether somebody like Dev or is a really Ops-orientated. Some employees have real affinity with ops, they love Jenkins-pipelines, docker, deployments, improving et cetera. While others, have no interest for that, but love the development and complex puzzles. It can be quite hard to create a balance of these skills within a team. Furthermore, it is even harder to balance that across teams and for a department.

We have platform and several teams work on that. There are several components. One of the things that determines how a certain component will be build is the team that picks up the feature. We have two internal teams. One of them would start with test scenarios, deployment models. While the other team would start with architecture and conceptual choices, they would consider deployment choices later. I like it when that happens. Let it explode. Why did it occur? How could we prevent it?

It is also fine that people get angry. You also get mad in a relationship, right? You need to challenge each other. You can get angry, as long as you don't damage others fundamentally, it can be very functional. You need to know why you come to work, what drives you. Sometimes there are discussions within a team. Most of the time I try to leave the decisions to the team. However, these team members are specialists, they are often knee deep in the details. It can help to bring in a neutral player, someone who asks why are we doing this again? They forget the bigger picture at that moment, which makes sense, because that is not their role. They excel in the details and sometimes need somebody to show that bigger picture again. The question I often ask is; does your solution helps us to go left or will it push us to the right, and if it does are you then ok with the fact that we will go left? That has nothing to do with DevOps, but with culture and human contact.

When I was developing there was no DevOps, more compartmentalized. Development made an artifact and Operations needed to install it and where responsible for it. They did not know the ins and outs, which results in friction. Once an operations employee said to me; we will not support that new java version, while I needed that version to use the opportunities it offers. These worlds were separated, so that problem became visible at the end of the process. So, that creates a lot of friction. Now we start the game with; this is what we want to achieve; the product needs to go as fast as possible to production and we need these things to make it happen. That is a nice way of working,

because dependencies are known upfront and Dev and Ops understand each other. That is a recipe for progress.

Question 3

What I really like about DevOps is knowledge sharing. I like it when I see that in our organization. We are an organization that builds a product. That product is not live in our production environment, but in that of the customer. Internally, for our product development, we have one, old school, Ops oriented professional. Great guy and we really learned why he could react allergic when we ask for a last-minute server or never giving the sign to kill a server. So, DevOps create a culture of learning to understand why that is a problem and what is important for others. Nowadays we maintain 75 servers ourselves. That is great. We are autonomous in making, maintaining and deleting them. That also means we have to create procedures for it and need clean our stuff and think about patch management. So, there is more understanding and therefore solutions will come.

Furthermore, there is the testing role. For me that is a coach. Teams need to make tests for themselves. We want them to think about quality and which level of testing is required. In some cases, it is necessary to use a tester, but the team need to think about the approach. Unit testing, performance testing, regression testing is all done by the team. We have a senior tester, who thinks more about the long-term approach and which tooling we need. He is also thinking about how to organize test in the right way, so that we have logical blocks. Automation is the key and then it is again about understanding. Our testers all did a java course. Now they understand the developer better. It helps them to connect to the development work and understand code and concepts like maven and quality tools. That works both ways, so a developer needs to think about how he can make his code testable.

By learning from each other you can, and will, both improve. For instance, some of the developers helped with reducing the costs for our servers by automating stuff. We also made a formula to calculate the required CPU per server. Before that, we always asked for too much. Which, annoyed traditional operations. We had servers that costs 500 euro per month and do nothing 99% of the time. If that is visible for a team you can even, make a game of it. Improve it, may the best man win. Today we deliver the same quality, while the costs were decreased with 20.000 per month.

Question 5

We have three teams. Two here and one in Romania. The two teams here are working based on the DevOps principles, so we will focus on them. A team needs cadres, about architecture, deployment model, security and functionality. Architects and product owners deliver those, technically and functionally. I would not say that is a common goal, though. Furthermore, we have a technical backlog. We have made significant steps the last few years. That is a backlog on department level. Our build streets, and therefore our feedback cycles, were too long. So, the goal was to bring that back till under 15 minutes. That results in a lot of stories. That was more a department goal.

If we look more to the teams; Rubic and 69. We divide the work in quartiles. Team Rubic is a real studio team, they work on that every quartile. They have full responsibility for that aspect of the platform. They are the owner, autonomously. Team 69 has a focus on runtime. There are several components that require attention and they work on one of those components per quartile.

Team Rubic really shows what can happen when you give a team full responsibility. It creates energy and ownership. It really works to give them the decision power. They have the task to create a new studio, that will replace the old one. They also decide upon the architectural principles and do the implementation themselves. That creates maturity and it reflects on the way they work together. They are a real team. Spar together, are critical, do things together and are proud on their product. Do not mess with that product. That is a nice thing. We are considering speeding up the development process, by adding a team on this subject. That will probably create tension, because of that proudness. The team came up with the solution to split their team into two teams. They think they can help the new team members to grow faster. I like when they come with the plan. It creates accountability. In the end it is my goal to create an environment wherein my role is not necessary.

Team 69 is very different. Where Rubic is innovative, 69 is quality minded. They focus on automated testing, unit testing, process rules and continuous integration. They work on a more controlled pace and over score on coverage agreements. Teams can decide that for themselves. Rubic will get into trouble once in a while, because they did not focus on quality. 69 will sometimes go too slow, because they focus a lot on quality. It is decided by team culture and (the level of the) team members. Team 69 had a lot of juniors and then it makes sense to focus more on automated testing. To create a cadre for yourself, more need for checks and balances.

Another reason for it can be are you more operations- or development orientated. Developers that do not know the impact of production incidents are more likely to forget automated testing. One of the team members of Team 69, Respondent4, has worked on the department Maintenance & Support. He knows the impact, so he will mitigate that risk and starts with testcases. You can see the nature of a developer based on the feedback that he gives during a code review. One will look at automated testing, while another will comment on code structure. You can influence that by letting team members experience other things. It is good for a development minded person to work a year on the production side. Great that you have built a nice feature, but when your error handling is off, and you receive a high priority incident you would wish that you had made some unit tests. A production incident is expensive and it is something you have to explain to customers. Bottom line: team members should experience all aspects that influence or are influence by their work.

It starts with creating a balanced team. We like to switch positions. A developer that worked a few years on the product, with internal R&D, can be switched to solutions. At that point he has to work with the product at the customer side. The internal R&D teams are also more in contact with the customers. They have a PS goal, so be billable for a limited number of hours. That really helps. There you are, with your own product. Customers looking to you. It creates new ideas. Team members often come back with feedback, for instance regarding the documentation. That they wrote themselves!

Question 18

Our employees also have contact privately. For me that is one of the benefits of working here. We share a lot, also misery. It is not only Instagram life. That is an important of our culture. Respect for each other problems is a cornerstone. If somebody would disrespect that he would have a hard time here. We want to help each other, so speak up your mind and we will find a way to work around the harder times in life. You will not be happy 40 years of your life. We want to know how everybody is feeling and the leadership team stimulates sharing.

It is just an aspect of shared understanding, but for me it is the most important element. Lately, one of our HR employees told me that she is always overwhelmed when she arrives at the office. Her mailbox is full, people at her desk et cetera. So, what can we do? Now she comes at the office at 10:30 and everybody knows do not call her between 09:00 and 10:30. She is doing her own thing and will be available for question at 10:30. Power to you. We try to create a situation where that is possible. That requires transparency and a culture of acceptance.

Shared goal vs Shared understanding

It is hard to decide, but I think the shared understanding has the biggest impact. A few days ago, I had a similar discussion with friends. We all are IT professionals and were thinking about what we could do together. We have no concrete idea. At some point we created an alternative path; could we start a restaurant together. If we would do that shared understanding would be the core, because the goal can be created right? We decided that it was possible. If you know each other well and what is the benefit of working together, the goal is secondary.

However, both are possible. A coach can say I want to win the champions league and gather a team that can do it, or you have a few great players and decide together what you want to achieve. It can be done from both sides and it has to be done that way. Shared goal is a way to create guidance and a kickstart, therefore I think most managers will say shared goal. Shared understanding is harder to achieve and it is more fragile. You can always create a new goal.

Respondent3: Summary

Question 2

Teams feel the responsibility to deliver. Before focus on the teams I want to explain how I perceive teams. A team is interdependent and has a shared responsibility. That holds, for instance, for quality. If an individual has a specific role, to ensure that a team delivers something, you are not autonomous as a team. If you look at the two internal teams that we have, then they have autonomy. They have their own team culture and based on those principles they solved the main challenges. This will not go without conflict, also between teams. I do not see that as a problem. That conflict creates discussion and that helps us to learn from each other. Often the discussion is about; what is the right way?

Both teams take quality seriously, but they deliver it in a different way. The first team, Rubic, will start with delivering value fast and will improve on quality later. The other team, 69, will do that the other way around. They will start with TDD. That way of working is an indicator of team culture. It has nothing to do with right or wrong and the teams help each other to improve on speed or quality. What I find interesting. The teams have assembled themselves like that. I do not think they are all aware of it, but we asked them to give up their preferences. That was 2-3 years back. With whom do you like to work and why? That start was already a base for different team cultures.

The strength of team culture is very visible with Team 69. From the original team of 6 are only 2 members left, but still you see the same principles. The principles were negotiated and are guarded by a scrum master. Culture is hard to change, that is why it works. You could only change it by reshuffling team members. Success could be a reason for that. If we can go to three internal teams, we could decide which team culture we want to copy, for instance: Rubic. Then we could hire some

developers and make two new teams. I would not like to mix members of Team Rubic and Team 69, that would result in a culture clash. Both cultures work and have good elements. Why would you mess with that?

Rubic is a very stable team. Almost no changes. The biggest change there is the level of autonomy. We gave them more and more freedom. That can result in friction. For instance, one of the team members cannot handle the level of responsibilities. This is a team that can handle that themselves. They will address it. The next step is giving them the autonomy to hire a new team member. If they hire them, they are responsible for making it a success. By giving them that right you do not fall back that hard into a forming phase. We are quite steady in the norming phase nowadays, perhaps even performing. I think we are close, but I would have to ask the team members how they feel. I do not know if every team member feels the required trust to speak of a full performance phase.

I started here to help with the transition to DevOps, together with respondent2 and a former agile coach. There was a resignation in the organization. There was one moment I will not forget. There was a sprint board, with only red smileys, the team failed on every level. The team looked at each other and said something like: 'oh well, we didn't make it', and went back to their desks. They were working 1,5 years on a new release and this was the mindset. That is where we came from. We needed some new developers, which we selected from projects at the customer side. Some of them are respondents in this interview.

The former agile coach gave the organization a dream. The dream was: release every two weeks. People said he was crazy, but we asked ok, what is necessary to achieve this? Me, respondent2 and the new developers worked hard on it. If we had a dashboard with red signs, we would not go home until it was fixed. If you do not fix it, your colleague has a problem tomorrow. That mindset. That started to spread. Give each other feedback. If those aspects are improving, there are principles. These principles form a new cadre, based on culture, which creates room to give autonomy. It starts together, building safeguards, processes, security and cadres. After that, teams can find a cadence and build towards an identity. In 2017 and 2018 we worked a lot on reducing team dependencies, that creates even more room to give autonomy to the teams. At that point cadres are also open for discussion. The team might find better solutions.

Question 3

We wanted to be able to push a button and have a new version. It does not matter what is in the version, it has to be ready. We only add fully functioning functionality. That means something. Development teams need to think about branching strategy, testing, prioritization of functionalities upfront. How do I plan my work in such a way that we are able to that? The collaboration also changed significantly, especially in terms of drive. Accepting a bad sprint that easy is not ok! You should be proud of your work. If something does not work, I will address it. Something is blocking the process and that is not good. You need to be able to say that and then solve it together. Respondent4 and Respondent6 have been important people in our transition process. They formed the police. They follow the process, build quality-gates and expect other to do the same. If somebody does not do that, they will challenge them. On the other hand, you also need professionals (for instance Respondent5) who do the opposite and will tell you that the rules do not apply to a specific situation. That the police need to shut up. That balance arose by challenging each other.

Another important point, if you want something you need to allocate time to it. The team has 10-15% of the sprint to work on these items and we make stories that are prioritized in a technical roadmap. The team is responsible and we will give feedback on whether they did it. Again, they decide, so they can focus on other things. However, it might make sense to make up for that choice in a later sprint. That is what happens. It comes in phases, also based on the level of pain that is felt by the teams. The management director and other stakeholders sometimes feel like this is lost productivity. That is my challenge. To guard that time, so we can limit our technical debt.

Question 5

Develop on a high-quality platform is the main objective of the teams. We work in quartiles, so per quartile there is a focus point. For Team 69 this is the performance of the process engine. Their year focus is Runtime, to improve the performance and the process engine. They often do the backend stories. For Rubic we are rebuilding, they are investigating a lot. How can improve our design, often called the studio. They have a lot of autonomy.

The teams have their strengths and weaknesses. Often, it is clear which team will work on which feature. The new features are often handled by Rubic. The complex puzzles, for instance related mainframes from banks, are often handled by 69. Our team in Romania does a lot regarding security and libraries. The teams know who they are. Also, from each other.

Question 18

I think they a complete picture. Teams have their own culture and within that culture there is a shared understanding. They are very aware about what they have to do the coming sprints and quartile. Sometimes I feel like they might be more equipped to look into the future than us, the leadership team.

They also know what is necessary to keep the required level, so also how to keep up shared understanding. Lately, Team Rubic asked me for a younger team member, because it would be good for the team dynamics. So, then I say. Ok, arrange it, write a job application. They had some interviews and they hired someone. Their own idea based on a need.

Shared goal vs Shared understanding

Shared goal and shared understanding have to intertwine. For me, the most important thing is facilitation, leadership and ceremonies with a clear goal. A retro is vital, give feedback. There has to a rhythm, but a team can fluctuate in output. You cannot always be on. We also facilitate in that, research week; gamification, bug squash, creativity weeks or something else. Once every quartile we make time for that. Same with work-life balance. If you are not feeling good, go home.

In the end, regarding all these things, leadership is very important. To create room, cadres and culture, you need strong leadership. Shared goal versus shared understanding is a management-team balance. If you, as a manager, go to the team, it helps to have a concrete goal. If they accept that, they can make it their own and create a shared goal. After that you can focus on strengthening the goal and creating understanding. That is a parallel process. For me, the shared goal is more

important, because it unifies and it is the starting point. We want to deliver high-quality solutions every day.

Shared goal has more influence and shared understanding is a process. What you see now, in comparison to three years back. New employees can attach quickly. They understand the goal and can fast forward to shared understanding, because the team facilitates in that. A funny thing was the outcome of a workshop, moving motivators. The whole department said that the need for process was low and the need for freedom was high. That is the culture that you want in a DevOps cycle.

Shared understanding is a less influential, but still very important. Especially the constructive conflict element. That drives progress.

What I missed in this is the importance to show the shared goal to the stakeholders and to measure and show success. They often state the amount of time it takes to invest in technical debt and deployment cycles. I like to show what you get for it; employee satisfaction, better ideas, faster ideas, happier customers et cetera. Guard the space for the team.

Respondent4: Summary

Question 2

Our teams consist of developers, supplemented with testers or operators. We host nothing, although, not directly for customers. We do have a cloud environment, wherein we can make virtual machines, for test services for instance. The collaboration is good. I think that you have to be responsible for something in order to be effective. So be the owner of something from the beginning to the end. The ideal situation would be that you are the owner of the product until it dies.

We come quite close to that. Our teams build a functionality, fix bugs, handle customer support, operations and functional maintenance. If we can delegate it to the team, we will do that. We starting working conform DevOps principles in 2016. Since then we have two teams at our location: Team 69 and Team Rubic. We also have a team in Romania; UTP.

Before the transition I worked in a project, at the customer side. The team here was a lot smaller and we had many problems. We made some big organizational changes since then. They were working on a release for a long time. That release had a lot of impact and brought many problems. The decision was made to make a change. Roughly 10 employees that were working for customers were brought in. Since then the collaboration has grown, that is a long process.

We needed changes on several layers. For instance, we could not make virtual machines ourselves. There was no clear division of responsibilities, within the product. Nowadays we have a clear split. One team is working on the studio application and the other on the runtime application. That focus has helped.

Question 3

The way that we started that transition has resulted in teams with more ownership. They feel more motivation to start doing something, to create. I think that is quite visible. The main reason for the change is that they got responsibility. After that it became stronger teams. They found a way of working that enables collaboration. I do not know for sure whether it is an improvement in comparison to before we came, but I suspect that it is better now.

Teams organize themselves and start things proactively. They also have more contact outside the office, do activities together. People are comfortable here, so they do not switch often. That has pros and cons. One pro is that we handle complex and abstract information and takes a long time to learn the application, so it is important to retain employees. A con is that we like to shuffle employees once in a while to foster new perspectives. They often do not like that idea.

One team has only developers. They do the testing and operation activities themselves. In the other team there is one tester, who does a part of those activities. In the end it is about balance. The team needs people who think about quality and operations, if one of the developers has that mindset it will also work. That grew automatically. However, you do look at how you can form the team. So, who do you hire and how do you create a balanced team?

There are several inputs for the team. One is customer support. We have an employee who handles customer support. Those issues will go via PO's and architects to the teams. We weekly discuss the bugs, whether it is a bug and if we let the team investigate. Another possible input are other business units, that is quite complex, but you could call them customers as well. We have a meeting with them once every two weeks. There a lot of discussion about who builds what and who is responsible. Sometimes Public of Finance will make something themselves, because we will not build everything. We have also advised them a few times to not make it at all. It can be difficult that there are three business units with an interest in the platform. We try to focus on the product vision and to communicate about that with the other business units. As everywhere, there is more desire then resources. So, we need to make choices. The architects and product owners make those choices, together with the leadership team. The epics come from the Roadmap, often created by product owners. However, the ideas come from many stakeholders; finance, public, customers, own ideas, team members et cetera.

Question 5

They have clear goals. Also, a department goal; work on a good and functioning product. On the team level you will also see it. An indicator is the kind of epics they pick up. Roughly 50% is epic development. The other 50% is support and other things that we deem important. It is possible that a team member is not working on the overarching team goal, based on that division of work. It could be that one developer is working on the technical debt of an application or on safety regulations of a part of the platform. However, the other 50% give the shared goal. For one of the teams that is the development of a new studio. They are really invested in that, working out architectural principles and investigating best approaches. That will keep them busy for a while.

Question 18

With one of the teams I am involved the most. However, not all the time that makes it more difficult to assess. I think I have the best shared understanding with the lead architect and the other architects. Also, in terms of personal goals. The teams have shared understanding, but there are points to improve. I think we could give more feedback.

Shared goal vs Shared understanding

I think the shared goal aspect is very important. If you have a goal, you know what needs to happen. You will see that in an organization. If the goal changes, everything shakes with it. If you formulate a clear vision the rest will follow. The same holds for the shared team goal. Shared understanding is the next step. You can build that up in phases.

Respondent5: Summary

Question 2

You also spoke to Respondent7. He is the person in our team who likes operations. I suspect that he told you that. We all share the idea of automation, containerizing et cetera. All to help us to improve and make life easier. However, respondent7 is the only one in our team who also likes to put time in it. We try to divide the work, but you do see that the infrastructural tasks often are performed by him. The others like to focus on pure development. We are developing a product. For us is a production environment the release that we make every two weeks. That will not go on one of our servers, no data migrations. That makes the ops-aspect a big smaller. It less critical than a real production environment. We are still very reliant on the development and test servers, but if it fails a few hours it does not hurt customers. Therefore, the risk is lower.

As a team, we know who is good in what. What we all like, and, also what we do not like. Those things are considered when we plan our work. That makes us a team that is balanced, we know how we want to play the game. There have not been many changes. Since I am here, now roughly 2 years, there has been one change in the team. That is quite rare in IT.

In 2016 we started focusing more on product development. I was one of the last developers to make that transition. Since then, a lot has changed. In 2016 this has started and we continued and improved on it. For me it is my first job. In the beginning I did not really saw the differences. I was just doing my job. One of the things that had a lot of impact was the goal to do a minor release every month or every two weeks. We did not have the tools to do that. Although, not without a disproportional time investment. Since a year releasing is also a team responsibility. We did not like that. First that resulted in friction, later in ideas to improve it. We are still working on improving that.

There are other elements that improved. The development environments have improved. More (test) servers, more environments which decreased the dependencies between teams. Those servers are still dedicated servers. The next step would be a set of docker image that can be published to a random machine. That is a work in progress and a lot of work.

Question 3

The biggest improvement that I see is the level of control, it is manageable now. If something grows and becomes dynamic, it is more important to automate and introduce some processes. Otherwise it does not scale and then the system does not work anymore. The bottlenecks became visible and we started talking about solutions.

Question 5

Since Q2 of last year this became very clear. We are, since then, building a new platform. Before that it was quite scattered. All over the place. We had no sense of ownership or that we could exercise

our influence on the application. Now we have that ownership. We are working on that and we decide. We feel autonomy and it helps to stand for something and to really go for it. At that point you will also really start discussing about what is the best way to do things.

We are able to agree within the team, but it can be hard to get the support from all the stakeholders. I do not mean the architects, but the project developers, often from other business units. They do not have the complete picture. I think we are at the best position to see long term benefits. Sometimes we do not get the priorities and that can be annoying. We see Finance and Public as customers. It can be a challenge when several customers are trying to influence the way we built the product. While they do not have the knowledge to understand why that is a bad idea.

Question 18

I work here now for two years. We got to know each other well, also privately. If something is going on in our private life, we will discuss it. We are very transparent. Secondly, it is important to know the others interests. In the last half year, we had OKR's (objectives, key, results). The goal was to assess what gives you energy and to focus on that when formulating year goals. At the end, it is important that you can do what gives you energy. That helps you and the team. Otherwise it will not last. I think we have a good insight in these things.

Shared goal vs Shared understanding

I think shared understanding was the most important element. Both is important, though. If I look at our organization, the goal has not always been clear. It has been formed from shared understanding, so bottom up. The autonomy lies with us and we created a long-term goal based on the clarity that came from shared understanding.

Respondent6: Summary

Question 2

The most important thing is that we all can do everything. Everybody has his preference, but we should be able to replace each other. Therefore, we need to keep each other up to speed. When something is new, we need to share the knowledge. That does not always go as I would like to see it. The team has a few new team members, we had quite some changes a few months back. We work on a very abstract and complex platform. Furthermore, we work the frontend, backend, deployment models, cicd street, automation, testing et cetera. So, there is quite a big difference between the levels whereon team members perform. The effect is that team members find a corner where they are comfortable and stay there. Sometimes I try to goad a team member to try something else.

Question 3

I started at DevOps1 during the transition. I was one of the developers that was selected to make a change and came from maintenance and support. The biggest difference between now and then is that we work more as a team. We feel more responsibility that something will keep working. When

we came here almost everything was broken. We really improved, for instance our build street or the speed of delivery.

The reason for those changes is the new people. They saw that and did not accept that. So, we started talking about how to make it a workable situation and then; you just have to do it. That is the main reason we came. Shake it up and bring the knowhow to change it for the better. It was outrageous in 2016. You could not develop without dependencies, based on the branching strategy limitations. Furthermore, it took very long to make a build. The lead time has improved tremendously. Now we can fix a bug and release it within hours, while three years back it took a few days.

The difference that we made was automating simple task and making sure that you can work isolated. That you have a safe environment to pick up stories. Then you will start going faster. Another important aspect was that the established order had two choices; commit to the new way of working or leave. Some did that on their own account and others were told to leave. The people that stayed are very happy with the situation that we have now. They feel empowered now and a few years back they did not have the power to change the situation on their own. Another important enabler was the vigor of the management team. They really supported a culture for continuous improvement and gave us the time to do that. Before the transition management was demanding bug fixes, but no quality improvements. So, you get what you ask.

Question 5

The shared goal is mainly focused on the functionality that we deliver. We make new features in an effective fashion. We want to do fun stuff and deliver value for our customers and colleagues. Rubic has the studio and for us it can be a several things: like a story in the frontend or backend. We are platform for making software, so it is quite abstract. That makes the goal abstract as well. Our business engineers are internal customers, our customers are customers and the customers of our customers are also using our products. So, we have three levels of potential customers. Furthermore, we have two business units who demand things.

We have a very generic block of software, that should be usable in several ways. That requires abstract thinking. It can look quite easy when it is done, but it very tricky to make it right. For me that is why remains fun, the complexity. We do not build the same thing twice. We make something generic and you can do a lot with it. That makes it hard to maintain. You lose the oversight, especially because it is a relatively old product. Many employees have worked on it, with several ways of working. Back in the days they used to hack some things in there. We still experience the pain from those decisions. So not everyone can do this work and takes a long time to adapt. That complexity and generic way of thinking is also an important element in our team goal.

Question 18

Every team member can assess what a certain story should contain; what are the important elements. The oversight across stories is harder to achieve. That creates mistakes. The structure of our team is known and the quality norm where we strive for. That is a given. We make automated tests, document what we do et cetera, we are at the point that we do not have to put that in every story.

What you do see is that the experience and drive of people decides how they perform regarding our quality norm. For instance, if somebody is mainly interested in the outcome of the features than you will see the details and might forget the way it connects with the application. You also need to consider the business goal and why we make it. That also comes with work experience. It helps you to add the right test and see the risks. Nobody knows everything of the product. There are a few developers who know a lot and there are a few experts of parts of the system. So, it is already a challenge to know who knows what, let alone know everything.

Shared goal vs Shared understanding

We need both. I think it also based on personal preferences. For me it starts with a shared goal, without a goal there is no need to create a shared understanding. What are you doing then? Without a goal you are aimless. Shared understanding comes after that, you can work on that as a team.

What I missed is the inter-team collaboration. We focused on the leadership team and the internal team. I think we could improve a lot by sharing more knowledge between teams. We did not do that enough the last years. We should share more knowledge. We are working in the same code base! Similarly, we could understand more of the team in Romania. It could reduce mistakes. So, the collaboration between teams and business units. That could really improve. The visions of business units are so diversified. They are really doing different things. The leadership teams of every unit should align on the most important aspects. Otherwise, you cannot move forward. There are many individual decisions. Should be more about we and not about them and us.

Respondent7: Summary

Question 2

It is balanced, constant. We are together now for over two years. In the beginning it was a challenge. There was group of people who had never work together before. It takes time to make a team out of that. That requires structure. We needed a rhythm, daily standups other ceremonies. We needed to refine our way of working.

During the forming phase those work processes are very important. Otherwise you do not know when things start or end. The teams needed a cadre. Nowadays, we have less need for that structure. It is more a natural process; some things have become second nature and other things are not necessary anymore. You follow process steps, but it is less explicit. We mastered the rules and know when and how to break them.

What helps is that we know each other well. We know what team members like and what their goals are. You do not have that knowledge in the beginning and it helps to enable everyone to be themselves. The planning is also tailored on what we can and like. Compared to 2016 we have more structure. Our CICD was not good enough, it took a long time. Builds took too long. Developers were waiting on each other. At the end of the sprint everybody needed resources and it created long queues. Everything was done on the same CI server; dev pipelines, test releases et cetera. We also had quality checks that took too long. There was a moment that I came Thursday morning and saw builds running from Tuesday. That is insane. Now we have split the goal of resources, one server for builds, one for test releases and one for releases. That reduced the dependencies tremendously. From that base we started improving incrementally.

The collaboration in the team has improved a lot. The team members know each other well. That helps us and enabled us to create a balanced team, wherein everybody contributes.

Question 3

For me, one of the turning points has been the moment that we got our own cloud environment. We did not have that in 2016. Everything was channeled through our system operator. When we got that cloud environment, we got more control and responsibility. We now can make our own servers and are responsible for their performance. What do you do with backups? Many things that need to be solved. Of course, the server, that we already discussed, made a big impact. We also started releasing more often. By doing that more the pain becomes smaller and you will automate more. In the beginning only Respondent2 was able to release, now everybody in the teams should be able to do it. We simplified the process and then they gave the responsibility to the teams. Minor patches are once every two sprints and majors are once a year. By doing it more it becomes a routine and by improving on it the process goes faster and becomes trustworthy. You have to go to the point that you trust what release, blindly.

This step, from Respondent2 to everybody can do it, is important. It reduces single point of failures. In our team we have one team member who knows a lot about one specific part of the application. A single point of failure. Which we really wanted to reduce. So, we asked ourselves; how can we spread his knowledge? So, he can do new stuff and we are not dependable. We introduced knowledge sessions. He started thinking about how he could teach us about this piece of functionality. It started in concepts and then he added layers each session. If a story touched a certain element, he would introduce that element to us and then others started working on it. He was a coach and a reviewer. That was a successful process. It is something to keep an eye on. Someone starts getting an interest and before you know it you have a single point of failure. Others think well he does that well, let him have that story again. It can grow pretty fast. You need to be able to understand and review each other, otherwise you have a problem. A last example was a developer who made a new framework in his spare time. It was so good that we wanted to start using it. So, we needed to learn it. We did that by writing tests for it. That was also a nice way to share the knowledge.

Question 5

We have a very clear shared goal. We are working on an old part of the product and we have the freedom to renew that how we see fit. We have the most knowledge about the product and what can be renewed and what would be the most effective way. Also, in how we can handle the migration process from the old to the new situation. Architects are involved in the process, but we are in the lead. We need to have a good story, able to answer questions. We use them for their knowledge and experience. Ask feedback when we have a concrete piece of the puzzle. It is always good to get feedback from someone who was not involved in the team process. We have a few cadres and within that we can exercise our autonomy.

Currently we are discussing how we could speed up the development process. One of the ideas is to higher some developers and introduce them in our team culture. After that you could reshuffle teams, which creates a knowledge transfer. When everybody is on par, we could switch back to the normal teams. In the end there are several ways to do this, and everything has pros and cons.

Furthermore, I do not think we will get to the point that the other team can do the same as us. They will do simpler stories, especially in the beginning and we will keep the main focus on this application. However, again, it is sharing of knowledge. Which is good. The process would not even change a lot, because we already work on the same major backlog. We might need to collaborate more closely.

Question 18

We all why we do it and make sure that everybody is on par with the required knowledge. Some of our team members inspire others to grow. They know a lot about something and will take the time to help others to the same level. It is not feasible if a team member starts working on its own. We need to share what we make, so we know what happened. One time we made a shift in code language and tools. A team member had a lot of experience with that. So, he thought of a way to introduce it to everybody.

Most of the time one developer writes the code, another reviews it and the one who is testing looks at the functionality. That means that two or three team members have not seen the code. If it is a small fragment that does not really matter. However, if that is a big decisive piece of software, that is not good enough. In those cases, we use the knowledge sessions.

Shared goal vs Shared understanding

Both have an effect. I think one has more to do with the way you work as a team. However, if you do not know what you want to achieve as a team it does not make sense. Lately a colleague said, 'I could do an uninspiring job with my current colleagues'. It is a good team and the team is more important than the activities. A good team helps each other, so you will get through the uninspiring tasks. A bad team will fail, even if the activities are fun. I like that way of thinking. However, I do clearly see the need for a shared goal. If there is no shared vision, it becomes hard to have guidance.

4.4 Results – Content analysis

There are two documents that DevOps1 provided based on table 3.6. One of them focuses on how team members view themselves, the organizations and the future and one shows how the leadership measures and manages the department.

4.4.1 How do employees view the organization

This is a document that is made in collaboration with an organization that provided a workshop. There are a few questions that were asked to all employees and they had to come up with a shared answer. In every situation they needed to describe two situations: 1) how it was now and 2) the preferred situation.

In the following pages the information is depicted. The pictures are in Dutch and the description above are in English.

What is your shared image of the future?

We want to go from:

A lack of guidance and vision, that results into challenges to prioritize and need for experiments

To:

A common vision about the product and organization which helps us to take 100% ownership of the decisions that correspond to guidance.

Wat is jullie gezamenlijke toekomstbeeld?

NiNi's NAAR!



NiNi's VAN!

BRAINERGY

EVO-T, Evolution Together!

NAAR

Door een gezamenlijke visie over het product en organisatie kunnen we zelf 100% eigenaarschap nemen bij beslissingen over de Richting.

PRODUCTS

VAN

Gebreke aan Richting en Visie, waardoor Prioritering lastig is en geëxperimenteerd moet worden.

What are your biggest chances and possibilities?

We want to go from:

Making qualitative good product with competent employees and deliver services

To:

Offer valuable solutions to customers with competent employees based on our vision

The graphic design is enclosed in a blue border. On the left, a vertical text reads "Wat zijn jullie grootste kansen en mogelijkheden?". In the center is a circular logo with a blue character named "NeNe" who has a head shaped like a brain with a speech bubble containing "NE". Above the character is the text "NeNe's NAAR!" and below is "NeNe's VAN!". To the right of the logo are two grey sticky notes. The top note is titled "NAAR" and contains the handwritten text: "Vanuit onze visie met competente mensen voor de klant waarde oplossingen bieden". The bottom note is titled "PRODUCTS" and contains the handwritten text: "VAN Met competente mensen kwalitatief goede producten maken en diensten leveren". At the bottom of the graphic, the text "BRAINERGY" is written in a large, bold, sans-serif font, with "EVO-T, Evolution Together!" underneath it.

What interests do you serve?

We want to go from:

Reactive short-term solutions based on individual unit focus, while thinking too much from the perception of the customer

To:

Build an innovative product that makes us happy and that can solve the real problems of users

The graphic design is divided into several sections:

- Top Left:** A yellow-bordered box containing a cartoon character with a smiling face, wearing a white suit with a yellow heart on the chest, and holding two yellow hearts. Above the character is a yellow circle with the letters 'Fi' inside. The text 'FiFi's NAAR!' is written above the circle, and 'FiFi's VANI' is written below the character. To the left of the box, the text 'Wat zijn de belangen die jullie dienen?' is written vertically.
- Top Right:** A grey sticky note with handwritten text in blue ink: 'NAAR Samen een innovatie product maken waar wij blij van worden en dat de echte problemen van de gebruikers oplost.'
- Middle Right:** A white rectangular box with the word 'PRODUCTS' written in black capital letters.
- Bottom Right:** A grey sticky note with handwritten text in blue ink: 'VAN Reactieve korte termijnoplossingen o.b.v. individuele unit-focus, waarbij u te veel vanuit de perceptie van de klant denken.'
- Bottom Center:** The word 'BRAINERGY' in large, bold, black capital letters, with the slogan 'EVO-T, Evolution Together!' underneath it.

What would you describe as a nice way to collaborate?

We want to go from:

Everybody wants to have influence on every level, what results in ineffective decision making

To:

Trust on each other capabilities, both inter-personal as inter-unit, and find a to build a tantamount relationship with the customers

The graphic design is divided into several sections:

- Top Left:** A vertical text box with the question "Wat is een fijne samenwerking voor jullie?" (What is a nice collaboration for you?).
- Center:** A cartoon character with a yellow heart on its chest, arms raised, and a speech bubble containing the letters "FE". Above the character is the text "FeFe's NAAR!" and below is "FeFe's VAN!".
- Top Right:** A sticky note with the heading "Naar" and the text: "Vertrouwend op elkaars kwaliteiten zowel inter-persoon als inter-unit, een gelijkwaardige manier met onze klanten omgaan".
- Middle Right:** The word "PRODUCTS" in bold capital letters.
- Bottom Right:** A sticky note with the heading "Van" and the text: "Iedereen wil invloed hebben op alle niveau's wat zich uit in inefficiënt besluitvorming".
- Bottom Center:** The text "BRAINERGY" in large bold letters, with "EVO-T, Evolution Together!" underneath.

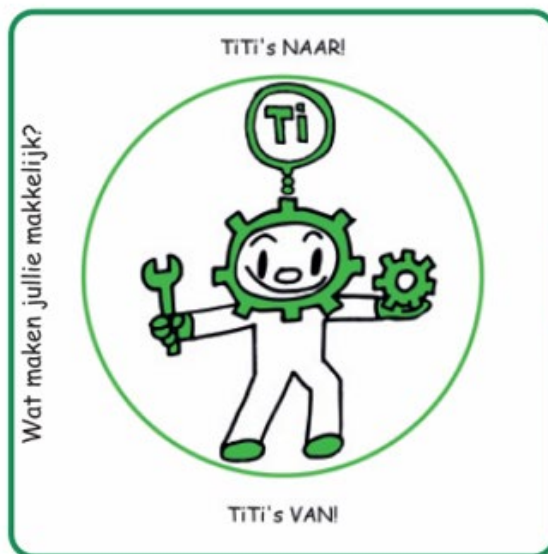
What can you do easily?

We want to go from:

Making a knowledge intensive solution in a bottom-up fashion

To:

Making a knowledge intensive solution, where the user is supported in an optimal way by a maintainable platform



BRAINERGY
EVO-T, Evolution Together!

Naar:
Het maken van een kennisintensieve oplossing waarbij de gebruiker optimaal wordt ondersteund door ons goed onderhoudbare platform

PRODUCTS

Van:
Het maken van een kennisintensieve oplossing op een bottom-up manier

What would be the right steps for you?

We want to go from:

Adding functionality, without thinking about the long term, based on the demand of a customer

To:

Make solutions based on our expertise, that fit the vision of the future, which is often verified with our customers

Wat zijn de juiste stappen voor jullie?

TeTe's NAAR!



TeTe's VAN!

PRODUCTS

naar
op basis van onze expertise oplossingen bouwen die passen in onze toekomstvisie en die we veelvuldig toetsen aan de werkveld van onze klanten

van
In de waan van de dag functionaliteit toevoegen waar de klant nu om vraagt

BRAINERGY
EVO-T, Evolution Together!

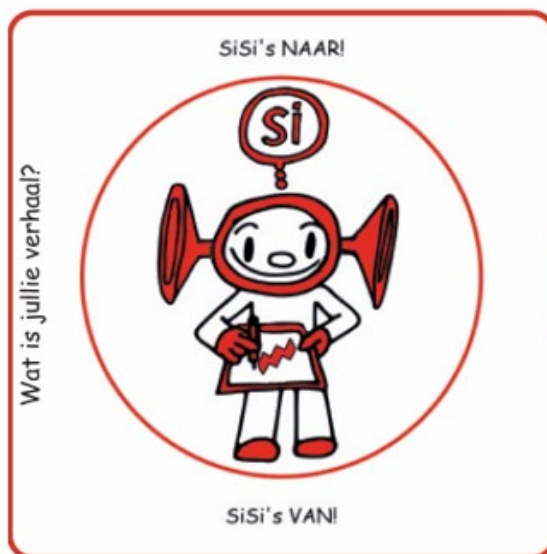
What is your story?

We want to go from:

Lack of a shared and clear vision which causes us to comply with short term wishes of our customers

To:

Make solutions based on a clear vision, which make sus happy and that aligns with our customer in a clear fashion.



BRAINERGY
EVO-T, Evolution Together!

Op basis van een heldere visie oplossingen maken waar wijzelf blij van worden en die recht doet aan onze klanten en dat op een heldere manier

PRODUCTS

Ontbreken van een gedragen duidelijke visie, waardoor we te vaak meegaan in de korte termijn-denken van onze klanten.

What results in the most action in your organization?

We want to go from:

By not using the product in a proper fashion we lost a lot of energy on feature discussion (reactive).

To:

Work on features based on our own vision and ideas and test these features by continuous feedback during development.



BRAINERGY
EVO-T, Evolution Together!

NAAR

Vanuit eigen idee/v
werken aan features en
deze d.m.v. Feedback
tijdens ontwikkeling,
toetsen bij gebruikers

Pro-actief

PRODUCTS

VAN

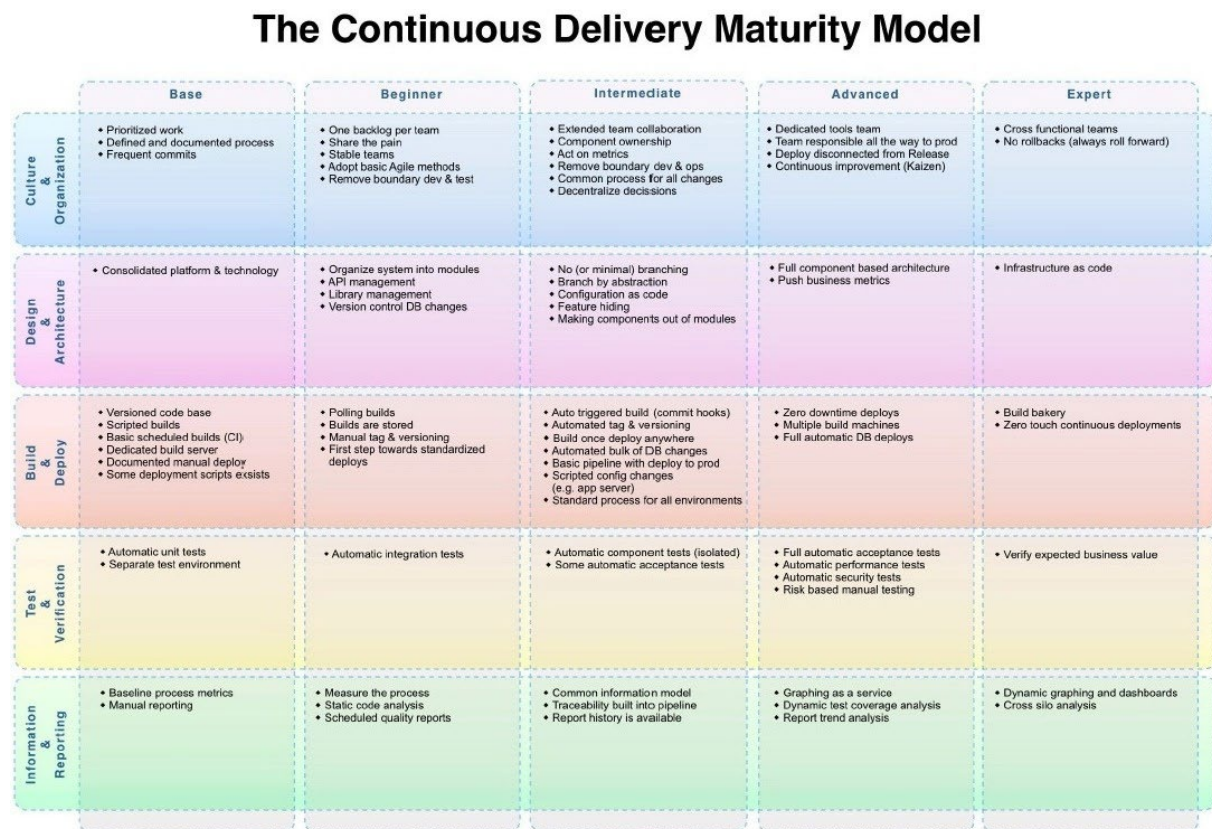
Door niet goed gebruik
van het product Bluebird
Zijn we veel energie
kwijt aan feature/func
discussies.

Reactief

4.4.2 Management information

The management uses a set of tools and metrics to monitor their progress. Figure 4.1.app shows a model that the organization uses to monitor progress. There are techniques in the model that provides the team with guidance. This resembles with opportunities and limitations are known, an indicator of shared vision and cadre. This helps to create a cadre for the team and to assess how they perform.

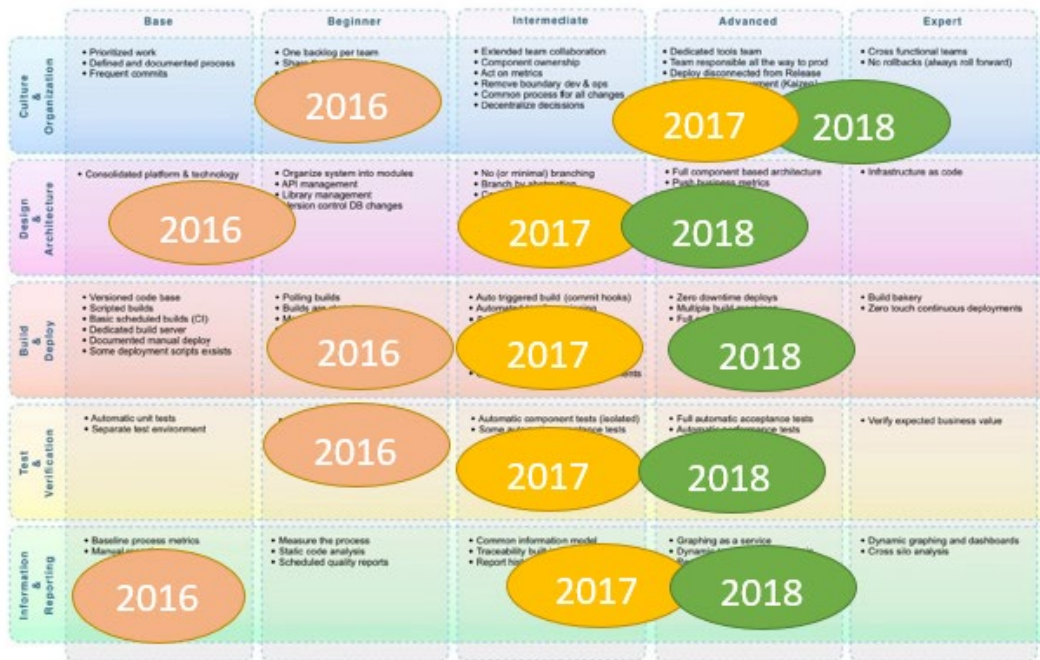
Figure 4.1. Maturity for teams



In the figure 4.2.app the progress of the teams on this model is depicted.

Figure 4.2. Maturity growth of teams

The Continuous Delivery Maturity Model



As depicted in Figure 4.3.app the management is very interested in how the technical debt progresses.

Figure 4.3.app: Technical debt

Improved Technical debt

2017

Jenkins	Progress		
	High	Normal	Low
Checkstyle warnings	0,00%	9,39%	0,00%
Duplicate code	13,22%	0,00%	0,00%
FindBugs	23,94%	14,96%	13,54%
PMD Warnings	13,87%	9,41%	0,00%
Open Tasks	43,06%	29,26%	17,00%
Compiler warnings	0,00%	79,30%	0,00%

2018

	Progress		
	High	normal	low
Checkstyle warnings	0,00%	-41,09%	0,00%
Duplicate code	-29,52%	-100,00%	0,00%
Find bugs	-25,93%	-13,83%	-46,65%
PMD warnings	-12,82%	-66,40%	0,00%
Open tasks	-21,95%	-11,94%	-20,48%
sevadocs (Sprint 43)	0,00%	-0,52%	0,00%

Technical debt:

Technical debt can be compared to monetary debt. If technical debt is not repaid, it can accumulate 'interest', making it harder to implement changes later on.



Goals 2018:

- Embedded awareness in teams
- Continuously pay the interest on our debt

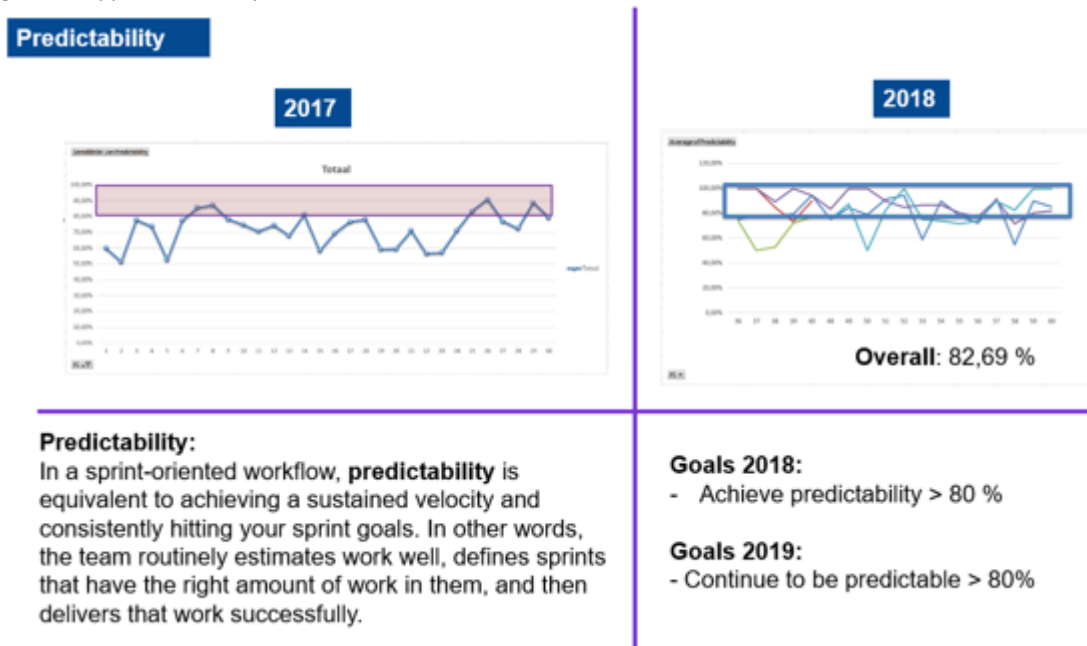
Goals 2019:

- Same as 2018

They give the team time to work in this. The guideline is that teams work for 70% of their time on the portfolio, 15% on support and 15% on technical improvement. Teams are asked to monitor this themselves and it is their choice if they do it different for a certain period. However, they need to be able to explain why that did something else and they should try to achieve this balance in the long run.

Other things the management team measures on are predictability, lead time and cycle time. Figure 4.4.app shows a visual they made regarding predictability.

Figure 4.4.app: Predictability



4.5 Results – Theory and Data

This section contains the comparison between the theory and the data are depicted in this section of Appendix 4. Paragraph 4.5.1 presents the comparison for the indicators and paragraph 4.5.2 for the sub-aspects.

4.5.1 Indicators – Theory and Data

This paragraph contains the comparison between the between the definitions for the indicators, based on the theory, and the data from the semi-structured interviews.

Benefit of working together is clear and communicated

According to Stock et. al. (2013) and Aronson et. al. (2013) individuals need to understand the necessity or benefit of the team structure (Kwak & Anbari, 2009). This indicator is practiced by respondents 1, 2 and 3. Respondent1 states that it was important to communicate with the team. Everybody knew what the intention was and the team agreed. Respondent2 and Respondent3 were part of another transition. They introduced a dream. Something that would not be doable if the organization would not change. One of the changes was working in teams. They needed to become accountable and would also receive more autonomous. The benefit of working together was that the cross-functional teams could create a better situation. Looking back, they all emphasize the importance of communicating that benefit to align the team.

Leaders know the environment and senior management

Thamhain (2009) stated that leaders with a strong affiliation with the environment and the senior management are more likely to create effective teams. Respondent3 states that the team knows what is going on, the top layers are often preventing them from excelling at the highest level. Respondent2 contributes that he is not always available for the team and Respondent1 finds the organization to be open. Respondent3 does not fully agree, he finds that the division of units creates an extra layer of confusion and discussion. Furthermore, the leadership should ask the teams what they can do. That is only done by the middle management, but not by the senior management. So, the Respondent differ in interpretation and perspective, which makes this an interesting indicator for DevOps1. The leadership seems to be aware of their environment and senior management, but two out of three respondents described that the link could be more effective by increasing servant leadership and availability.

Decentralized day-to-day decisions

Zábojník (2002), among others, emphasized the importance of the decentralizing day-to-day decisions. Centralization only makes sense if the manager's signal is better than the worker's signal. It also creates mature teams (Gupta et. al., 2017). All the managing respondents state that they have decentralized the day-to-day decisions. They are responsible for the entire solution; from how it will meet the requirements till how we can maintain a high level of service in production. They can seek guidance, but they decide when and how. Only HR and education are centralized with a team of 4 team leads. The management tries to position themselves as an escalation option and a coach. They

also guide the balance of their employees. All respondents saw the importance of decentralized day to day decisions.

Individual team members have focus on the team

To have an effective organizational structure, teams should be assigned to one team (Tessem & Iden, 2008; Guidice & Condo, 2017). Respondent3 states that the team is above all. Respondent2 agrees and confirms that the team is responsible for their own organization. They both emphasize that everybody is fully dedicated to one team. Teams can ask each other for help, but that should fit with what the team is doing. The team is protected that way. Respondent1 describes a similar situation, the team members are fully focused on the team, although there are many stakeholders with diverse expectations. This demands a lot of flexibility from the team.

Strategy and expectations of the organization are known

Eldor (2019) posed that a team can only have a shared goal when they know the strategy of their organization. They need to have an insight of what would make them successful as a team. All respondents feel that this is an indicator which could, and should, improve. The three units often disagree about the vision, so the strategy is fragile. That blocks transparency, because the leadership does not want to share a strategy which could change in an instant. So, they share what they can, but not as much as they would have wanted. The respondents do see that this has an impact.

Opportunities and limitations are known

According to Eldor (2019) a team needs to know which things they can decide for themselves and when they need to consult the management team. It is interesting that the business units organized this indicator different. Respondent2 and Respondent3 state that everything is possible, just do it. Be smart and try it isolated and inform me, because I am interested. They stress that it is important to fail fast. The team of Respondent1 is in another phase. They started this transition a year ago. He states that it is the job of the manager to ensure that opportunities and limitations are known. It is quite interesting that there are different ways to manage it. Both statements show that the leadership is aware of the importance. However, there are different ways to ensure it. The teams of Respondent2 and Respondent3 might feel more autonomy, while the team of Respondent1 could feel more control.

Self-aware team members

Van den Bossche et. al. (2011) and Bitner & Leimeister (2014) found that team members need to be self-aware in order to align behavior with the environment. The respondents find it hard to judge this for the other team members. They do all seem to understand the importance of this indicator. The organization implemented several sessions to help employees to find their ambition. They did this by stating what gives you energy and asking employees to set targets that fits that. Respondent6 states that it differs, also based on work experience and personality. This fits the other statement, where the more experienced seems to be a bit more self-aware. Respondent4 and Respondent7 are part of a more senior team. They both state that most things can be shared. They think this indicates

that people are self-aware. Respondent7 contributed that an in-depth feedback session of the team did not result in big surprises for the team members. Team members know their strengths and weaknesses. Building on these statements one could argue that the respondents find it an important indicator that could be improved upon.

Environment for asking questions

A team should strive for clarity; therefore, they need to have environment for asking question (Bittner & Leimeister, 2014; Van den Bossche et. al., 2011). All respondents state that this is an important aspect of their team culture. They feel that everybody can feel free to ask questions, especially regarding content. Helpfulness (respondent6), trust (respondent5, respondent7) and strong culture (respondent7) are elements that are mentioned by the respondents. They all are convinced that this indicator is a strong part of their culture and see the relevance.

Team members listen to each other

To use all the individual perceptions a team needs to listen and grasp everyone's perspective (Bittner & Leimeister, 2014; Van den Bossche et. al., 2011). The interviews show that it is more natural for the team to ask question, than to listen to the answer. Respondent4 describes it as: 'they hear each other, but sometimes I wonder if they listened'. He might have found the important difference where the other respondents are struggling with. Almost every respondent state that most of these conversations focus on the content. The underlying cause for the comment that a team member makes might be on another level. Respondent4 described that as team members conversing on different levels. Another observation is that the teams are good at discussing on the content level. They like to discuss about the problem and sometimes it takes weeks to agree. Respondent7 states that it really helps to find the best solution. Respondent5 states that it also very different per person. Some team members are more likely to try and understand the perspective of the other than other team members. A strong opinion can block the flexibility to do that. These statements suggest that there is room for improvement, while they seem to agree that it is an important indicator.

Moments of reflection and evaluation

Teams can learn from reflection and evaluation patterns (Mohammed & Dumville, 2001). The respondents are all aware of the importance of reflection and evaluation. However, most of them do not feel like they are doing this as they should. Especially when things get harder (respondent4) or when it is time to be critical towards someone (respondent6). Respondent6 feels that an external coach might be a way to improve that. There seems to be a difference between teams, because Respondent5 and Respondent7 are a bit more positive. They say there is a culture of giving feedback, however they could also introduce more elements that focus on team development and personal relations. Both teams see the retrospective as the formal moment to reflect and evaluate. The respondents of team Rubic both state that it is important to give feedback whenever it is necessary. Respondent4 states that Team 69 almost never gives feedback outside the retrospectives. It is interesting that all respondents feel that this is an important aspect, but that it is not yet at a place where they are satisfied. Secondly, the difference in teams is noticeable.

Team members complement each other

Team members can contribute most to the team, when they complement each other (Bittner & Leimeister, 2014). All respondents recognize the importance of a diversified team. Things that are stated are: team members have their own focus and differ in terms of personalities (respondent4), based on different viewpoints we collect all relevant input (respondent5), some of us are leaders and other are followers (respondent7). Respondents seem to be aware of the relevance of diversity in order to complement each other. They are also quite satisfied with the balance in the teams. It is interesting to contribute that respondent3 disclosed that the teams were formed based on the input of the team members.

All responsibilities and required skills should be clear and described.

The team should have a clear overview of responsibilities and required skills (Hackman & Wageman, 2015). The respondents are all convinced that the team is in the right position to know this. They know it best (respondent4) and new team members need to have the right mindset and then all prerequisites are there to get up to speed (respondent6). Respondent5 and Respondent7 both state that the responsibility and required skills are clear. There might be a mismatch in focus, most team members like development and only one, respondent7, likes operations. They know this and hold each other responsible to get the job done. The last year they really focused on removing single point of failures. All respondents are aware of the importance of this aspect and are convinced that the team has a good insight.

Psychological safety to share opinions

Psychological safety is an enabler for team members to share divergent views (Bittner & Leimeister, 2014). The respondent5 and respondent6 feel like there could be more attention for this indicator. Respondent6 feels that question like; is it psychologically safe and what do we need for that, are important. He focuses on awareness. Respondent5 states that it possible to be the center of ridicule when a team member shares an unpopular idea. He also thinks it is a fun element of the team that shows their nice team dynamic. Respondent 4 emphasizes the experienced population as a factor for psychological safety. He feels like they have experienced professionals, who know what it takes to be successful in the work environment. Respondent7 notes that an argument can really show the psychological safety. Team Rubic is able to continue talking about it and solve the issue. Most respondents are quite convinced that this is an important indicator, that they recognize in their practice.

A culture of asking questions

A team should give a situation the attention it deserves, which can be ensured by building a culture of asking questions (Kleinsmann & Valkenburg, 2008; Bittner & Leimeister, 2014). All respondents share this believe and they state that they do this. Respondent4 notes that it is important to be critical, especially across roles. Respondent6 emphasizes the critical nature of his team, in an

attempt to help each other to the next level. Respondent5 and Respondent7 also talk about that mentality. They believe it helps in the decision-making process. An indicator that really seems to be an important part of both team cultures.

Team members that share relevant information

Teams thrive more when they have all the relevant information available (Bittner & Leimeister, 2014). There are two opinions regarding this indicator, based on execution at DevOps1. Respondent5 and respondent7 state that they share knowledge on the spot or during knowledge sessions. They share proactively and have processes to facilitate it. Respondent4 and Respondent7 are more hesitant. Respondent4 states it is hard to find the right balance between sharing information and creating turmoil, while respondent7 states the important details are often forgotten. It might be interesting to share best practices between teams regarding the way they share information, because there is quite a difference between the teams on how this is perceived. However, the sharing of (relevant) knowledge is deemed important by all the respondents.

4.5.2 Sub-aspects – Theory and Data

This paragraph contains the comparison between the between the definitions for the sub-aspects, based on the theory, and the data from the semi-structured interviews.

IT Leadership

When an organization requires teams to have a shared goal it is important that there are leaders in the organization that provide clarity (Stock et al., 2013). As described in 2.2.2, the theory states that leadership can be assessed by using the indicators; benefit of working together is clear and communicated and leaders know the environment and senior management.

The data corroborates that. The importance of leadership is clear by reviewing table 4.4. Furthermore, the data suggests that leadership might have a bigger impact than the conceptual model assumes. There are several indications that leadership is also very important for creating a shared understanding. The respondents state that the room created by the leadership team gave them the chance to grow as a team. That enabled shared understanding. Therefore, leadership might be a sub-aspect that influences both main-aspects.

Organizational structure

Another sub-aspect that can foster a shared goal is the organizational structure (Jassawalla & Sashittal, 1998; Kwak & Anbari, 2009). Organizations need to find their balance between centralized or decentralized decision making. As described in 2.2.2, the theory states that an organizational structure can be assessed by using the indicators; decentralized day-to-day decisions and individual team members have focus on the team.

The organizational structure was deemed less important by the respondents. Especially the team respondents state that it has a limited effect on having a shared goal. The managers have another perspective. The difference seems to be that the team members view an organizational structure as

overhead and processes, while the management sees it as the climate to foster teams. An organizational structure which enables autonomy is still an organizational structure according to them. Both indicators are implemented successfully according to all respondents.

Shared vision and cadre

A team can only define a relevant shared goal when they know the strategy of their organization (Eldor, 2019). Therefore, they need to know the mission statement of the organization (Eldor, 2019; Aronson et al., 2013; Lynn & Akgün, 2003). Furthermore, they should have an insights in when they are successful as a team (Gutiérrez, Lloréns-Mones & Bustinza Sánchez, 2009), in order to determine and execute the required tasks (Paris, Salas & Cannon-Bowers, 2000). As described in 2.2.2, the theory states that a shared vision and cadre can be assessed by using the indicators; strategy and expectations of the organization are known and opportunities and limitations are known.

The shared vision is deemed important by all respondents. This is also corroborated by the content analysis. However, they are not positive about the situation at their organization. It should be clarified. The limitations are quite clear and the teams exercise their autonomy within it. Teams have grown a lot and therefore the cadre is less strict. The vision could give them guidance and that does not happen enough.

Construction

Teams should be able to create a construction of the individual. Team members need to be aware of their own mental model (Bittner & Leimeister, 2014). This helps in aligning behavior and understanding how others perceive you. As described in 2.2.2, the theory states that the construction can be assessed by using the indicators; self-aware team members, environment for asking questions and team members listen to each other.

The data shows that construction is the least important sub-aspect for shared understanding according to the respondents. However, the respondents do state that they see the importance of all the indicators. There should be harmony between them, so they can empower each other. Respondent1 and Respondent6 describe that there is a sequence and it starts with construction.

Co-construction

Teams should be able to create a co-construction of the images of two or more team members. They should be able to evaluate the outcome and compare the individual understanding of team members (Van den Bossche et al., 2011). As described in 2.2.2, the theory states that the co-construction can be assessed by using the indicators; moments of reflection and evaluation, team members complement each other and have some level of experience and all responsibilities and required skills should be clear and described.

The team members mention co-construction as the most important sub-aspect for shared understanding. An important indicator, that should be improved, is moments of reflection and evaluation. The respondents explain that there are situations where they could more feedback. The other indicators are more tangible and the team is very capable in complementing each other and knowing the required skills and responsibilities. They see co-construction as the moment that communication starts and they state that it is important to be transparent. The business unit: core

strives for a culture where vulnerability and respect are important. The team members show that they embrace those principles.

Constructive Conflict

Teams should be able to create a constructive conflict in an attempt to understand each other (Van den Bossche et al., 2011). The differences should be discussed. All elements of the “conflict” should be addressed and there should be room for questions and conflict negotiation. As described in 2.2.2, the theory states that the constructive conflict can be assessed by using the indicators; psychological safety to share opinions, a culture of asking questions and team members that share relevant information.

The managers assess this sub-aspect as the most important. However, the team members score it as the lowest. That is quite interesting. The team members emphasize the need to communicate and be transparent. The management does that too, but sees the potential conflict as a way to discuss and learn. Constructive conflict is described as a way of channeling emotions and passion. By doing that two or more ways of thinking collide, which can create understanding.