BizDev Teams in Agile Software Development

A case study of coordination between business and development

Helga Nyrud

Thesis submitted for the degree of
Master in Software Engineering
60 credits

Department of Informatics
Faculty of Mathematics and Natural Sciences

UNIVERSITY OF OSLO

Summer 2017

BizDev Teams in Agile Software Development

A case study of coordination between business and development

Helga Nyrud

Summer 2017

2017

BizDev Teams in Agile Software Development

Helga Nyrud

http://www.duo.uio.no

Trykk: Reprosentralen, Universitetet i Oslo

Abstract

Background: There is a need for better collaboration between business and development units in product producing organizations, within in-house development. This need has emerged because developers would often know which ideas that are possible to implement and which are not, while business representatives have insights into the customer needs, the competitors and the market. Consequently, both units would benefit from the other's insights, and a closer collaboration can be achieved through effective and appropriate coordination.

Aim: The aim of the thesis is to investigate the challenges between business and development in agile software development, and identify possible solutions to the challenges. Additionally, the aim is also to investigate how task uncertainty, task interdependence and size of work unit change when business and development merge into one team. The scope of this thesis is coordination between business and development in a BizDev team, where BizDev simply is a team consisting of both business representatives and developers.

Considering coordination is achieved through coordination mechanisms, such as meetings, those areas are examined.

Method: A qualitative case study was conducted in an organization where business and development merged into a BizDev team. Data was collected by conducting 11 interviews, observing 51 meetings and 21 entire days with the team, and gathering various documentation.

Results: The results revealed that there were 24 coordination mechanisms present in the team. These could be grouped into three main categories; impersonal, personal and group mode. The most prominent challenges they experienced were differing needs in the working environment, they were not present at the same meetings or used the same methodologies or tools, and their team dynamics were of a working group, not a team.

Conclusion: It is possible to increase the collaboration between business and development by changing the implementation of the daily stand-up meeting, creating a team meeting that includes all members, safeguard and arrange for the use of story mapping, and facilitate for informal ad hoc conversations because they create speed and fast decision-making. These changes could lead to less status reporting, a stronger team spirit, a common understand of what is being creating, and lead to speed and fast decision-making. Furthermore, merging business and development into one team lead to increase in all the factors; how task uncertainty, task interdependence and size of work unit.

Acknowledgements

Writing this master thesis has been a challenging, yet rewarding experience. Completing this thesis would not have been possible if it wasn’t for a number of people. First, I am incredibly thankful to my supervisor Viktoria Stray for invaluable guidance, immensely helpful contributions and tremendous support. I am also thankful to my supervisor Yngve Lindsjørn for valuable feedback and good help. Additionally, I am incredibly thankful to Nils Brede Moe for giving me the opportunity to write this thesis, for excellent discussions, and exceptional input. Furthermore, I am forever grateful to all of the participants in the study, for welcoming me into their organization and including me in their activities and workplace. I thank my fellow students and the employees of the research group Programming and Software Engineering for the valuable discussions, distractions and help. In particular, I thank Kristin Brænden for having read my thesis and suggested valuable improvements.

Lastly, I am grateful to my family and friends for their support and optimism. I am especially thankful to my mother for proofreading and providing valuable contributions, in addition to giving endless encouragement. A special mention goes to my late father, whom is greatly missed and always believed in me. Particularly, I thank my dear Sikander for tremendous patience, love and comfort.

Helga Nyrud
Oslo, July 2017

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

There is broad consensus that there is a need for a holistic approach to software development (Bogsnes, 2016; Dingsøyr, Moe, Fægri, & Seim, 2016; Fitzgerald & Stol, 2017; Karvonen et al., 2016; Leffingwell, 2007; Overby, Bharadwaj, & Sambamurthy, 2005). In this case, the holistic approach refers to a closer collaboration between the various actors that contribute to the development of software. The actors include for example legal representatives, customer service agents, business representatives, designers, developers, testers and maintainers.

Another way of describing the holistic approach is by considering the key concept of 'flow' from the Lean development method (Fitzgerald & Stol, 2017, p. 2), where the goal is to achieve 'flow' between the various actors. Such a collaboration has been proposed in research already, namely between development and operations, and *DevOps* is a proposed concept and mindset (Debois, 2011; Fitzgerald & Stol, 2017; Humble & Molesky, 2011). Meanwhile, there is still a missing collaboration between the business and development units, which has been expressed in research in various ways(Burn & Szeto, 2000; Dingsøyr & Lassenius, 2016; Willcoxson & Chatham, 2004).

However, as far as I know, after a thorough research, there seems to be a gap in research when it comes to studies performed on coordination between business and development within agile software development. I would argue that this gap amplifies the need for this thesis.

In order to clarify and define the scope of this thesis, when the terms business and development are being mentioned, it is referred to each their respective units within a product producing company. This thesis is therefore about in-house development.

One of the main challenges between business and development that I am addressing in this thesis is that the business unit typically act as a buyer, while the IT (development) unit is being viewed as a supplier, meaning that the business representatives order an information system that they have specified for the developers. The developers have little influence on the functionality of the finished system. This is an unfortunate situation because developers have the technical insights and domain knowledge that is useful to take into consideration when developing a system. For example, the developer would often know which ideas that are possible to implement and which are not, as well as which ideas are easy to implement and which are difficult. Furthermore, the business representatives have insights into the customer needs, the competitors and the market. Consequently, both units would benefit from the other's insights, and a closer collaboration between the business and development is needed in order to develop better software systems.

Furthermore, as both stated in agile software development research (Dingsøyr, Nerur, Balijepally, & Moe, 2012) and Lean Startup (Ries, 2011), it is rarely possible to specify an entire software system in advance due to the high degree of uncertainty. Therefore, frequent releases of the software, and a close dialog with the customer have been proposed (Dingsøyr et al., 2012). In many cases, the business unit is the customer of a software system, and has the role of the product owner (Schwaber & Beedle, 2002). Since one of the concepts in agile software development is the 'on-site customer' (Vinekar, Slinkman, & Nerur, 2006) that is an example of the need for business being present in development.

An important benefit from a closer collaboration between the two organizational units is increased speed, which Bosch (2016, p. 84) calls for: "companies today must respond to new customer needs and requests at unprecedented speeds, which required a level of enterprise-wide agility that's often exceedingly difficult in traditional, hierarchical organizations".

The closer collaboration and 'flow' between the various actors stated above can be achieved through efficient and appropriate *coordination*. Coordination is also an important aspect to study when investigating the challenges in the collaboration between business and development. Coordination can be understood as the way a group of people is organized in order to be productive. Therefore, coordination is a very important part of establishing organizational structures and processes. Since agile software development methods have been criticized for only being applicable to small teams (Dybå & Dingsøyr, 2008), the coordination between organizational units in agile software development is even more crucial.

Coordination is achieved through coordination mechanisms, such as different types of meetings. In this thesis, I will examine coordination through studying the different coordination mechanisms between business and development.

Furthermore, since I have chosen to look at coordination as a way of integrating two organizational units, I will study the notion of cross-functional teams, as I will look into how business and development can be integrated as an interdisciplinary team.

1.2 Research Area

The research area of this thesis is coordination between business and development within agile software development. The process of coordination will be studied by examining coordination mechanisms.

There are several different types of coordination mechanisms, such as meetings and tools. Since many of the different coordination mechanisms are dependent on the software development methodology, as for example the daily stand-up meeting, parts of the agile software development methodologies and practices will be examined.

Lastly, I will study cross-functional teams consisting of people from both business and development.

1.2.1 Research Questions

With the previous described motivation and research area, the research questions are the following:

RQ1: What are the challenges and possible solutions in the coordination between business and development in agile software development?
RQ2: How does task uncertainty, task interdependence and size of work unit change when business and development merge into one team?

1.3 Thesis Structure

Chapter 2: Background contains all the necessary theories that make up the theoretical background needed in order to answer the research question. The Chapter includes theory on *software development methodologies, coordination, the BizDev concept* and *teamwork* theory.

Chapter 3: Research Method outlines what research method was applied in order to study the coordination mechanisms in a cross-functional team consisting of business and development.

Chapter 4: Research Context presents an overview of the unit under study in order to provide a context to the study.

Chapter 5: Results includes an overview of the observed coordination mechanisms in the study, as well as an extensive description of each one of them.

Chapter 6: Discussion contains a discussion of the result and theory in order to answer the thesis' research questions. Furthermore, the limitations of the study will be outlined.

Chapter 7: Conclusion and Future work presents the conclusion to the research question and propose what future work should be conducted on the field.

2 Background

This thesis is based on theories on software development. First some relevant software development methodologies are presented, as the methodologies affect coordination by providing ceremonies and various arenas for coordination. Then, theories on coordination are presented, because research on coordination has been used as a basis for outlining how different organizational units work together to develop software. Lastly, theory on cross-functional teams is presented since one way of facilitating coordination between two different units is by merging them.

2.1 Software Development Methodologies

Agile software development is a newer approach to software development, as a contrast to the traditional waterfall model. Software development belongs to the engineering discipline called software engineering (Wohlin, Šmite, & Moe, 2015). A theory suggests that software engineering is a balancing act between three capitals; human, social and organizational (Wohlin et al., 2015). Furthermore, software development is the production of software, which consists of a sequence of fundamental activities called a software process (Paulk, Curtis, Chrissis, & Weber, 1993). The software process is the systematic approach used in software engineering, and in addition to activities, consists of methods and practices used to create software and other products such as plans, documents and tests (Paulk et al., 1993).

A simplification and abstraction of these software development processes are represented in software process models, which are used to explain different approaches to software development (Kaur & Sengupta, 2012).

The first software process model was the traditional waterfall model. This methodology is named the waterfall method due to its plan-driven nature, because one phase has to be planned and executed before proceeding to the next phase. The phases in the waterfall model includes requirements analysis, design, and development phases (Larman & Basili, 2003).

Moreover, as a response to the plan-driven process model, the agile methods emerged (Highsmith, 2002). The agile methods were created because there was a need for methods to take into account the unpredictability of the world, including the higher rates of change and feedback (Dybå & Dingsøyr, 2008). Furthermore, the agile methods involve the customer much earlier, during development, and they focus on reducing time spent on overhead (Dybå & Dingsøyr, 2008). Agile software development has become the mainstream development methodology (Dingsøyr et al., 2012).

While there are many different agile methods, this thesis will include elements from Scrum, Extreme Programming, Continuous Software Development, Lean Software Development, Kanban, Scrumban and Lean Startup. The reasons why these methods are included are because they are either directly or indirectly present in the case study.

2.1.1 Scrum

Scrum is an agile software development process based on empirical process control theory, and is meant for situations where there is a lot of uncertainty about the finishing result and changes are likely to occur (Schwaber & Beedle, 2002). Scrum has a list of different practices, including: The Scrum Master, Product Backlog, Scrum Teams, Daily Scrum Meetings, Sprint Planning Meeting, Sprint, and Sprint Review.

Scrum Master

The Scrum Master is a management role and the role is "responsible for ensuring that Scrum values, practices, and rules are enacted and enforced" (Schwaber & Beedle, 2002, p. 31).

Product Backlog

The Product Backlog is "an evolving, prioritised queue of business and technical functionality that needs to be developed into a system" (Schwaber & Beedle, 2002, p. 32). The Backlog includes every feature or change that any stakeholder would want made to the product (Schwaber & Beedle, 2002).

Scrum Teams

The Scrum Team is a self-organizing team, and is the unit of people in which produces the software and completes the current tasks from the Product Backlog (Schwaber & Beedle, 2002). Furthermore, the team is cross-functional and should include the necessary roles to complete their tasks, such as designers and programmers (Schwaber & Beedle, 2002, p. 37), and they should consist of seven people, plus or minus two (Miller, 1956, as cited in Schwaber & Beedle, 2002, p. 36).

Daily Scrum Meetings

The Daily Scrum Meetings is a practice put in place in order to ensure communication within the team, because “software development is a complex process that requires lots of communications” (Schwaber & Beedle, 2002, p. 40). The practice has multiple names, and the frequently used daily stand-up meeting originates from extreme programming (Stray, Sjøberg, & Dybå, 2016). The daily stand-up meeting is a rather popular practice, because a study shows that 87% of those who employ agile methods use the daily stand-up meeting (Stray, Moe, & Bergersen, 2017). The duration is maximum 15 minutes, and every team member answers the three following questions each:

1. What have you done since last Scrum?
2. What will you do between now and the next Scrum?
3. What got in your way of doing work? (Schwaber & Beedle, 2002, p. 43).

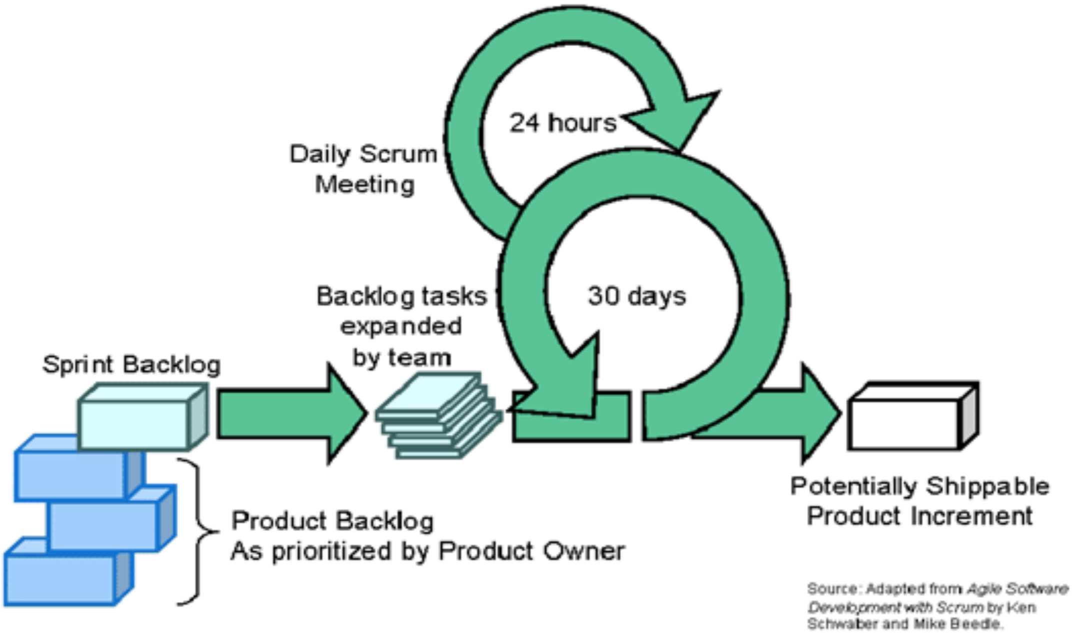
The purpose of answering these questions is in order to identify any obstacles the team members might encounter, and share it with the team in case they can help each other (Schwaber & Beedle, 2002). Moreover, Stray et al. (2016) found it to very important to be standing during these meetings, because their results show that the meetings lasted considerably shorter when everyone were standing compared to if some people were sitting.

Sprint Planning Meeting

At the Sprint Planning Meeting “customers, users, management, the Product Owner and the Scrum Team determine the next Sprint goal and functionality” (Schwaber & Beedle, 2002, p. 47). The attendees choose which tasks from the Product Backlog that is supposed to be completed during the subsequent Sprint (Schwaber & Beedle, 2002).

Sprint

A Sprint is a fixed period of time in which the Scrum team works, and is from two to four weeks long (Schwaber & Beedle, 2002). During one Sprint the Scrum Team produces a product increment, therefore it is said that a Scrum Team works in increments (Schwaber & Beedle, 2002). Figure 1 illustrates a Sprint, with the related practices Product Backlog and Daily Scrum Meetings.


24 hours
Daily Scrum Meeting
30 days
Backlog tasks expanded by team
Sprint Backlog
Product Backlog As prioritized by Product Owner
Potentially Shippable Product Increment

Source: Adapted from *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle.

Figure 1: The sprint in Scrum

Sprint Review

The Sprint Review is an informational meeting where "the team presents to management, customers, users, and the Product Owner the product increment that it has built during the Sprint" (Schwaber & Beedle, 2002, p. 54). The Sprint Review is therefore held after the end of a Sprint, and Scrum Master is responsible for conducting it (Schwaber & Beedle, 2002).

2.1.2 Extreme Programming (XP)

Extreme Programming (XP) was one of the most commonly practiced and researched agile software development methodology nine years ago (Dybå & Dingsøyr, 2008). However, at the present date agile practitioners state that less than 1% use XP as their agile method (VersionOne, 2017).

Nevertheless, XP is mentioned here because one of its practices, continuous integration, is crucial to the creation of the continuous software development. Therefore, only a superficial presentation of the methodology will suffice.

The name, XP, was given because the methodology pushed already acknowledged good practices to extreme levels (Beck, 2000). The practices that together constitute the XP methodology are the following: incremental planning, small releases, simple design, test-first development, refactoring, pair programming, collective ownership, continuous integration, sustainable pace and on-site customer (Dybå & Dingsøyr, 2008).

2.1.3 Continuous Software Development

The notion of continuous software development started with the practice of continuous integration in the Extreme Programming methodology (Beck, 2000). However, it expanded into an umbrella term for continuous integration, continuous delivery, continuous testing and continuous deployment.

2.1.4 Lean Software Development

Lean software development was derived from the lean manufacturing, especially the Toyota production system, because the methodology was successful in the car manufacturing industry (Poppendieck & Poppendieck, 2003). Overall, the main goal in lean development is to reduce and eliminate all kinds of waste, which includes everything that does not produce value for the customer (Poppendieck & Poppendieck, 2003), the methodology consists of seven main principles: eliminate waste, amplify learning, decide as late as possible, deliver as fast as possible, empower the team, build integrity, and see the whole (Poppendieck & Poppendieck, 2003, as cited in Dybå & Dingsøyr, 2008).

2.1.5 Kanban

Kanban is originally a lean tool to managing production operations (Liker, 2004). Recently it has however been more popular in software development (Ahmad, Markkula, & Oivo, 2013). The methodology consists of the following principles: "visualise the workflow, limit work in progress, measure and manage flow, make process policies explicit, improve collaboratively (using models and the scientific method)" (Anderson, 2010, as cited in Ahmad et al., 2013). "Kanban is a Japanese word meaning a signboard, and it is used in manufacturing as a scheduling system. It is a flow control mechanism for pull-driven Just-In-Time production, in which the upstream processing activities are triggered by the downstream process demand signals" (Liker, 2004, as cited in Ahmad et al., 2013).

2.1.6 Scrumban

Scrumban is a hybrid between Scrum and Kanban, as the title suggests, and is a rather new method in the world of software development (Reddy, 2015). There are therefore several different understandings and uses of the method as described by Reddy (2015) “some believe it to be nothing more than using virtual Kanban systems within the Scrum framework, while others believe it to be new software development framework that combines “the best” elements of Scrum and the Kanban Method” (Reddy, 2015). However, Reddy (2015) emphasizes that neither of those understandings are correct, but that Scrumban is a new method applying Kanban from the Scrum methodology.

In simple terms, Scrumban makes use of several of the Scrum practices, such as backlog, planning and review, combined with Kanban, such as the board, flow and the pull principle (Ladas, 2009). Making use of the Kanban board includes having a limited number of tasks in progress, because “a task card without a limit is not a Kanban in the same way that a photocopy of a dollar bill is not money” (Ladas, 2009).

2.1.7 Lean Startup

The Lean Startup was coined by Ries (2011) in his book named “The Lean Startup: How Constant Innovation Creates Radically Successful Businesses”, and is a newer addition to the lean methods. Eric defined a Startup as being a “human institution designed to create a new product or service under conditions of extreme uncertainty” (Ries, 2011), in which means that this approach is meant for any kind of organization.

Validated Learning

With the term validated learning Ries (2011) argues that organizations need to conduct experiments on customers in order to test their hypotheses and vision. Creating prototypes is a way of conducting an experiment.

Feedback loop

The Lean Startup approach includes a process called the “Build-Measure-Learn feedback loop” (Ries, 2011). He argues that every product producing organization wants to turn an idea into a product, then measure how the customers respond, then lastly learn from the measurements in order to know whether to continue as before, or change the course (Ries, 2011).

The build phase of the feedback loop implies creating a minimal viable product (MVP), which Ries (2011, p. 77) defines as: “that version of the product that enables a full turn of the Build-Measure-Learn loop with a minimum amount of effort and the least amount of development time.”

Furthermore, the measure phase of the feedback loop includes establishing metrics and milestones, and performing methods to see whether the efforts are leading to actual progress (Ries, 2011). An example of such a method is the experiments and prototypes mentioned above.

Lastly, the learn phase of the feedback loop involves examining the resulting data from the measure phase, and make a decision on “whether to pivot the original strategy or persevere” (Ries, 2011, p. 77).

2.2 Coordination

Since coordination can be understood as the way a group of people is organized in order to be productive, theory on coordination is outlined here.

Theories on coordination

There are several definitions of coordination, from numerous fields, such as organization theory, economics and computer science. From the field of sociology, Van de Ven, Delbecq, and Koenig Jr (1976, p. 322) define coordination as "integrating or linking together different parts of an organization to accomplish a collective set of tasks.”

In 1988, Malone (1988, p. 5) proposed the definition of coordination as: “the additional information processing performed when multiple, connected actors pursue goals that a single actor pursuing the same goals would not perform.” However, in 1994, Malone and Crowston (1994, p. 90) introduced the following definition: “Coordination is managing dependencies between activities.”

Theories on coordination mechanisms

Coordination is achieved through coordination mechanisms. In addition to define coordination, Van de Ven et al. (1976, p. 69), also identifies three categories of coordination mechanisms; impersonal, personal and group, in which are described detailed in Section 2.2.1.

Similarly, Mintzberg (1979), from the field of organization theory, also defines six coordinating mechanisms: mutual adjustment, direct supervision, standardization of work processes, standardization of outputs, standardization of skills and standardization of norms.

Moreover, Jarzabkowski, Lê, and Feldman (2012, p. 908) refers to the term coordinating mechanisms instead of coordination mechanisms because of its dynamic nature, and defines coordinating mechanisms as "dynamic social practices that are under continuous construction” (Jarzabkowski et al., 2012, p. 907).

Both the approaches from Mintzberg (1979) and Van de Ven et al. (1976) are useful in order to study coordination mechanisms. However, Mintzberg (1979) speaks of coordination of tasks, while Van de Ven et al. (1976) speaks of coordination of different parts of an organization. The definition by Van de Ven et al. (1976) is highly suitable in this case study, where the main focus is challenges and possible solutions in the integration between business and development (in order to produce better software systems). As a result, the approach by Van de Ven et al. (1976) is more suitable for this case, and therefore their framework was followed.

Since coordination is achieved through coordination mechanisms, such as different types of meetings, I will examine coordination through outlining the different coordination mechanisms between business and development. Since Van de Ven et al. (1976) also propose a framework for mapping out coordination mechanisms, I used their proposed framework in my collected data.

2.2.1 Coordination mechanisms

Following the definition of coordination of Van de Ven et al. (1976), they identify three modes of coordinating work activities: impersonal, personal and group (Van de Ven et al., 1976, p. 322). Furthermore, they discuss how situational factors such as task uncertainty, task interdependence and unit size might predict which of the three modes that are being used (Van de Ven et al., 1976, p. 322).

The impersonal mode is coordination by programming, such as the use of "pre-established plans, schedules, forecasts, formalised rules, policies and procedures, and standardised information and communication systems" (Van de Ven et al., 1976, p. 323).

The personal and group modes are types of coordination by feedback, which is defined as mutual adjustment based upon new information (Van de Ven et al., 1976, p. 323). In the personal mode, the individual serves as the mechanism for making the mutual adjustment, while for the group mode, a group of individuals serves as the mechanism for making the mutual adjustment (Van de Ven et al., 1976, p. 323).

Furthermore, the personal mode is divided into two channels of communication: vertical and horizontal. The vertical channel is typically communication with line managers or unit supervisors (Thompson, 1967, as cited in Van de Ven et al., 1976, p. 323). The horizontal channel is communication between two co-workers in a non-hierarchical relationship (Van de Ven et al., 1976, p. 323).

Moreover, the group mode is divided into scheduled and unscheduled meetings (Van de Ven et al., 1976, p. 323). The scheduled meetings are the planned, more routine forms of communication, such as team meetings, meanwhile the unscheduled meetings are the unplanned forms of communication, such as the informal, more spontaneous conversations between more than two team members about a work-related problem (Hage, 1974, p.151, as cited in Van de Ven et al., 1976, p. 323).

2.2.2 Task uncertainty, task interdependence and size of work unit

In addition to proposing a way in which to group coordination mechanisms, Van de Ven et al. (1976, p. 322) introduces three hypotheses regarding task uncertainty, task interdependence and size of work unit within organizations, concerning the different modes of coordination. They describe how these situational factors determine when the different coordination mechanisms are used (Van de Ven et al., 1976, p. 323). They used data to test the three hypotheses, and gathered results that either confirmed or rejected the hypotheses.

Task uncertainty

Task uncertainty is being defined by Van de Ven et al. (1976, p. 324) as: "the difficulty and variability of the work undertaken by an organisational unit", and their hypothesis is the following:

A. Increases in the degree of task uncertainty for an organisational unit is associated with
   1. a lower use of the impersonal coordination mode
   2. a greater use of the personal coordination mode
   3. a significantly greater use of the group coordination mode (Van de Ven et al., 1976, p. 324).

The data that Van de Ven et al. (1976, p. 329) Van de Ven et. al. used to test the hypothesis showed that there was a substantial decrease in the impersonal coordination mode (plans and rules), a large increase in the horizontal communications and group mode (scheduled and unscheduled meetings), while the vertical communications remained invariant. Therefore, hypotheses A1 and A3 can be considered confirmed, while hypothesis A2 would be considered partly confirmed because the vertical communications part of personal coordination mode remained unchanged. Their results are shown in the Figure 2.

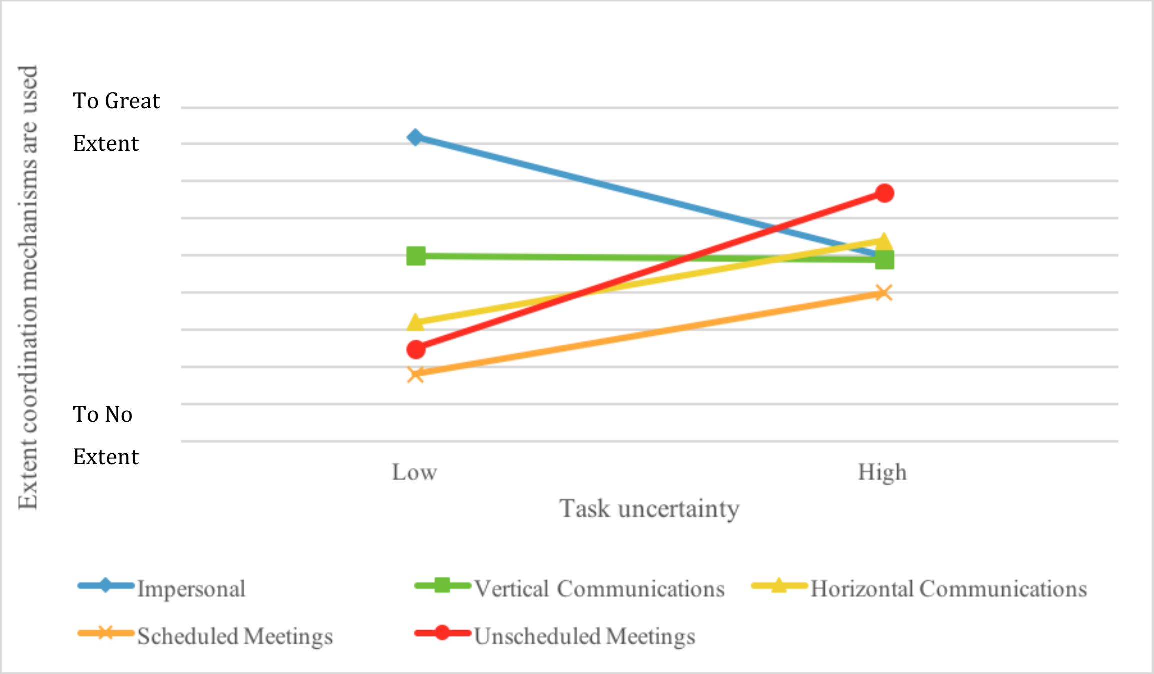


Figure 2: Coordination mechanisms and task uncertainty (Van de Ven et al., 1976)

Task interdependence

Van de Ven et al. (1976, p. 324) define task interdependence as: “at the work unit level of analysis is the extent to which unit personnel are dependent upon one another to perform their individual jobs,” and their hypothesis is the following:

B. Increases in work flow interdependence from independent to sequential to reciprocal to team arrangements will be associated with
   1. small increases in the use of impersonal coordination mechanisms
   2. moderate increases in the use of personal coordination mechanisms
   3. large increase in the use of group coordination mechanisms (Van de Ven et al., 1976, p. 325).

Within this hypothesis I decided to only add the results from reciprocal to team arrangements because those are relevant to the results collected in this thesis. Reciprocal is the state to which business and development was at before merging into a team, while team arrangement was what they changed into. Reciprocal occurs when “the output of one department becomes the input of another, with the addition of being cyclical” (Thompson, 1967). And the team arrangement refers to work flow situations where the work is undertaken jointly by the team members and they collaborate in order to complete the work (Van de Ven et al., 1976).

The data that Van de Ven et al. (1976, p. 330) used to test the hypothesis showed that there was an increase in all of the coordination mechanisms, except for the impersonal coordination mode and the vertical communications part of personal coordination mode.

Therefore, hypothesis B1 would be considered rejected because there is a decrease in the impersonal mode. However, the hypothesis B2 would be considered partly confirmed because the increase in the horizontal communications exceeds the decrease in the vertical communications. Moreover, the hypothesis B3 would be considered confirmed because there is an increase in both scheduled and unscheduled meetings. Their results are shown in the Figure 3.

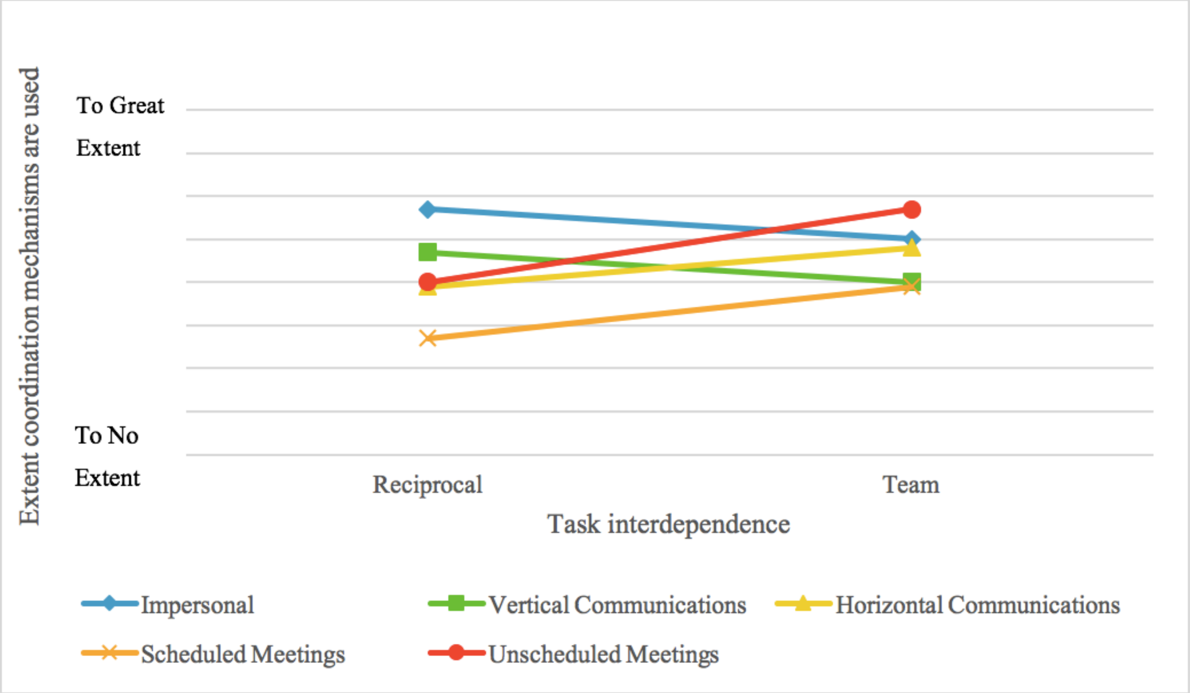


Figure 3: Coordination mechanisms and task interdependence (Van de Ven et al., 1976)

Work Unit Size

The size of work unit is defined as: “the total number of people employed in a work unit.” (Van de Ven et al., 1976, p. 324), and their hypothesis is the following:

C. An increase in work unit size is associated with

1. a decrease in use of group coordination
2. an increase in use of personal coordination
3. a significant increase in use of impersonal coordination mechanisms (Van de Ven et al., 1976, p. 326).

Within this hypothesis I decided to visualize the results as small and large work unit size because Van de Ven et al. (1976, p. 331) makes the distinction between the results gathered on the work unit size of 2-10 people, and above 10 people. Since this thesis examines two separate units consisting of five and seven respectively, which merged into one unit consisting of thirteen, that distinction is applicable to this thesis as well.

The data that Van de Ven et al. (1976, p. 331) used to test the hypothesis showed that there was an increase in the use of impersonal coordination mode, while there was a decrease in the rest of the coordination mechanisms. Therefore, hypotheses C1 and C3 would be considered confirmed, while hypothesis C2 would be considered rejected because the personal coordination mechanisms (vertical and horizontal communications) decreased. Their results are shown in the Figure 4.

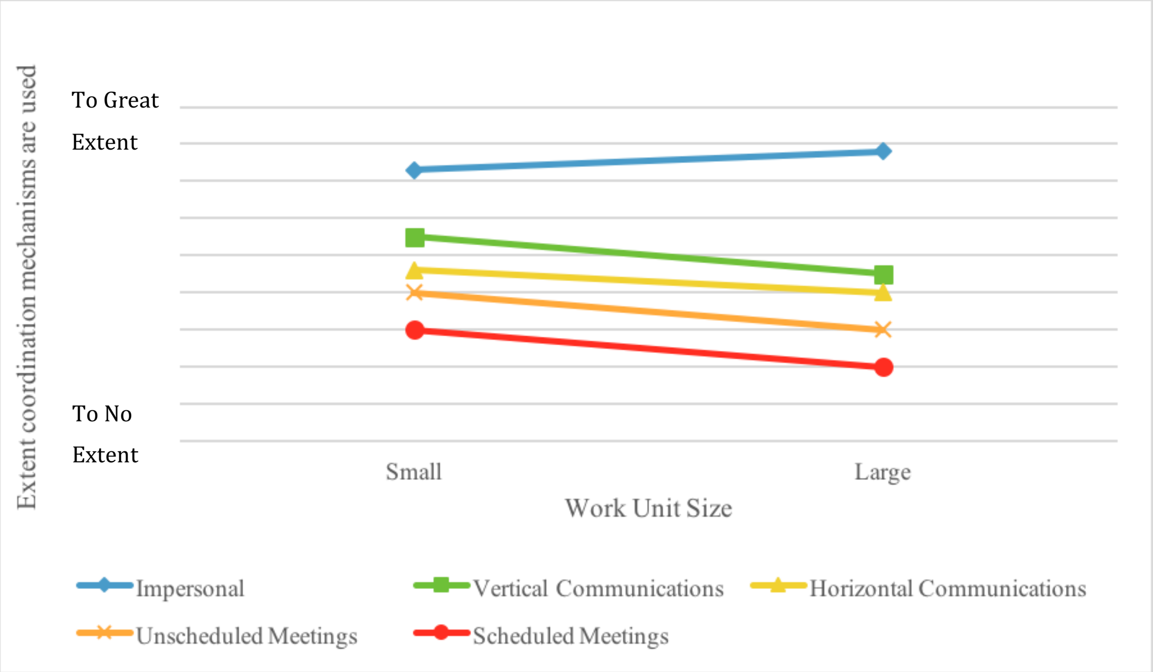


Figure 4: Coordination mechanisms and work unit size (Van de Ven et al., 1976)

2.3 BizDev

Fitzgerald and Stol (2017) created the term "BizDev" in order to facilitate integration between business strategy and development, as a continuation of DevOps and Continuous Integration, illustrated in Figure 5.

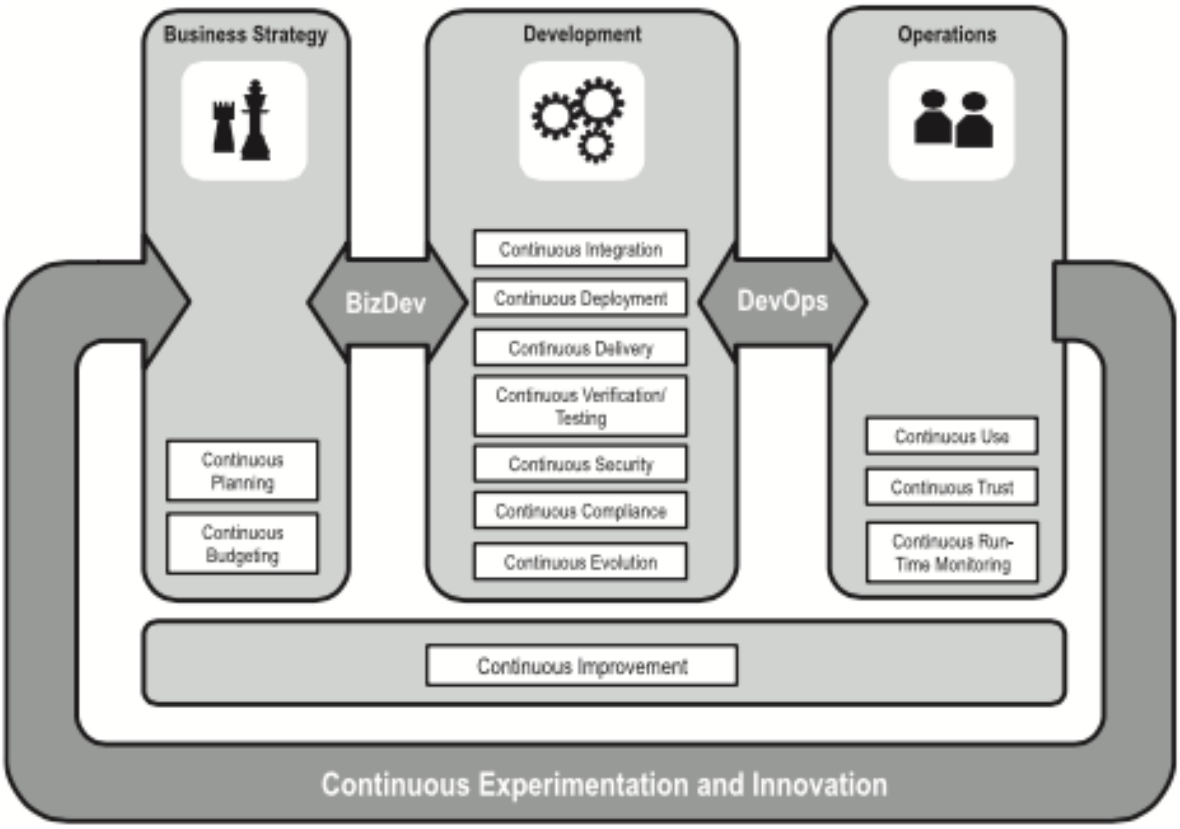


Figure 5: A holistic view on activities from business, development, operations and innovation (Fitzgerald & Stol, 2017).

Furthermore, Fitzgerald and Stol (2017, p. 10) argues that the various continuous movements, described above, in Chapter 2.1.3, reveal the need for a continuous flow from a customer demand to the fast delivery of a solution.

The purpose of BizDev is outlined as: “The age-old disconnect between the business strategy and technical development components is recognised in the BizDev concept which seeks to tighten this integration” (Fitzgerald & Stol, 2017, p. 10). Moreover, the Scrum role of Product Owner is acknowledged as a recognition for the connection between business strategy and development by the agile methods, but that it does not go far enough (Fitzgerald

& Stol, 2017, p. 10). And since they do not explicitly define how to create this connection, I propose that it is by creating a cross-functional team called a BizDev team.

In general, a cross-functional team can be defined as: “a group of people with a clear purpose representing a variety of functions or disciplines in the organization whose combined efforts are necessary for achieving the team’s purpose” (Parker, 2003, p. 6).

DevOps

As the Figure 5 illustrates, BizDev is a similar concept to DevOps. DevOps was created in order to bridge the gap between development and operations.

2.4 Teamwork

I am examining the way of achieving better collaboration between business strategy and development by creating a cross-functional team, as a way of attempting to secure coordination between the two different disciplines. Therefore, it is important to define and understand what a team is. Hence, the following section describes theory on teamwork.

2.4.1 Team

An established definition of a team is “a small number of people with complementary skills who are committed to a common purpose, set of performance goals, and approach for which they hold themselves mutually accountable” (Katzenbach & Smith, 1993).

Moreover, when it comes to the optimal number of team members, studies show that small teams consisting of three to five members are more productive, while teams of nine or more members are significantly less productive than small teams (Hoegl, 2005; Putnam, 1978; Rodriguez, Sicilia, García, & Harrison, 2012).

2.4.2 Team vs. Group

However, not all units working together are teams. As the definition of a team is rather narrow, all collection of people working together is not teams, but groups, as teams deliver extra performance (Katzenbach & Smith, 2005, p. 2).

Katzenbach and Smith (2005) describe that “the best working groups come together to share information, perspectives, and insights; to make decisions that help each person do his or her job better; and to reinforce individual performance standards”.

Table 1 shows a comparison of a working group and a team proposed by Katzenbach and Smith (2005, p. 4):

Table 1: Working group vs. team (Katzenbach & Smith, 2005)

Working group	**Team**
Strong, clearly focused leader	Shared leadership roles
Individual accountability	Individual and mutual accountability
The group’s purpose is the same as the broader organizational mission	Specific team purpose that the team itself delivers
Individual work products	Collective work products
Runs efficient meetings	Encourages open-ended discussion and active problem-solving meetings
Measures its effectiveness indirectly by its influence on others (such as financial performance of the business)	Measures performance directly by assessing collective work products
Discusses, decides, and delegates	Discusses, decides, and does real work together

3 Research Method

In this Chapter I present and provide the reasons for the chosen research method and technique for data analysis.

3.1 Qualitative Research

When investigating the research questions, I started by developing the appropriate research design. On one hand, there is the quantitative research methods which attempts to maximize replicability, objectivity and generalizability of findings, (Harwell, 2011, p. 149) and are frequently characterized as assuming that there is a single "truth" that exists, independent of human perception (Lincoln & Guba, 1985). On the other hand, there is the qualitative research methods that focus on discovering and understanding the experiences, perspectives and thoughts of participants (Harwell, 2011, p. 148), that it explores meaning, purpose and reality (Hiatt, 1986). As Manstead and Semin (1988, as cited in Robson, 2002, p. 80) points out, the chosen strategy depends on the type of research question put forth. Therefore, since my research questions involve understanding coordination between humans from different parts of an organization, the research design of quantitative methods seemed most fit.

When conducting a research project there are various elements that needs to be thoroughly examined, and Robson (2002, p. 81) put forth a model with the following components:
purpose, theory, research questions, methods and sampling strategy. Table 2 contains my approach to Robson's components which was begun before the study was conducted, and emerged underway:

Table 2: My approach to proposed components in a research design (Robson, 2002)

Robson's Components	**Conducted in this study**
Purposes	This study is trying to understand how business and development work together in a cross-functional team. I am also trying to assess whether the way they work together is effective or not. The study is trying to contribute to solutions to the problems that have been seen between business and development. Hopefully, the study can contribute to more effective coordination between business and development in the future.
Theory	Theory within the agile software development will inform my study. Theory on coordination will help me in outlining and assessing the situation. The coordination framework on coordination mechanisms by Van de Ven et al. will guide me in collecting data and analyzing them.
Research questions	The question this research is geared towards answering is what arenas contribute to effective coordination between business and development? And how do these arenas provide to effective coordination? These questions are to be answered in the context of a team consisting of both business and development (BizDev).
Methods	I will use semi-structured interviews, participant observation and documents to collect data. The data will be analyzed by using the framework by Van de Ven et al., as well as importing all of the data into a tool for qualitative analysis and coding them accordingly. I will ensure the trustworthiness of the data by triangulation as a way of improving validity.
Sampling strategy	I will be present in an organization that has undergone an organizational change and created such BizDev teams. First, I will observe some meetings in order to get to know the business and the context. Then I will be present every day over a period of time, until I believe that I have understood the situation sufficiently. Then I will conduct interviews. Lastly, I will organize a feedback session in order to ensure that I have understood the team and organization correctly.

Within the field of qualitative research methods (Robson, 2002, p. 87) presents two research design strategies; fixed or flexible. While the fixed design strategies require the researcher to know all of the components above very rigorously before conducting the study, the flexible design strategy allows the researcher to know the answers underway by interacting with the research subjects and thus form the research from that understanding of the subjects. Since some of my answers above were not thoroughly outlined before the research started, but completed and shaped along the way, I followed a flexible design strategy.

Within the field of flexible design strategies there are three traditional strategies, namely case study, ethnographic study and grounded theory study (Robson, 2002, p. 89), where I conducted a case study. The reasoning for why I chose a case study follows in the next Section.

3.1.1 Case Study

A case study was my chosen approach to conducting this research because the objective of this thesis was to study a single case (BizDev) in its context (an organization) (Robson, 2002, p. 89). Furthermore, I conducted a case study because it is preferred when the researcher's aim is to get an in-depth understanding of a real-life phenomenon (Yin, 2009, p. 18). Since a case study has a distinct advantage at answering "why" and "how" questions (Yin, 2009, p. 13), it is highly appropriate at answering my research questions regarding how business and development coordinate themselves, and why certain coordination mechanisms are more effective.

Since the presence of the BizDev team is a unique case (Yin, 2009, p. 47), the case study design is the single-case (holistic) design. The unit of analysis is a group of people, composed as a so-called BizDev team. However, in addition to the unit of analysis, later called team Alpha, there was another team, later called team Beta, that had recently merged into a BizDev team, and I therefore collected data on team Beta in order to better understand team Alpha. Because of how recent team Beta had merged into a BizDev team, as well as less data collected on them, I decided to not treat the two teams as equal in an embedded case study design, but rather use the data from team Beta to support my analysis of team Alpha.

Figure 6 shows the context of this study, that both the business unit (Biz) and the development unit (Dev) were working in the same product area before and after the merger, and that the only difference was that the two units merged into a BizDev team.

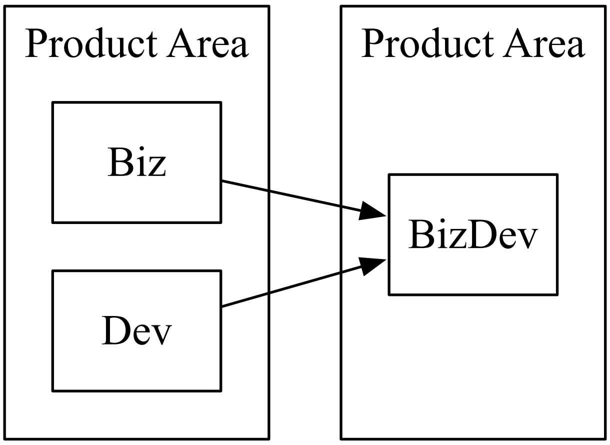


Figure 6: The context of the merge between business and development

3.2 Data Collection

The data that was collected as this thesis' sources of evidence is participant observation, semi-structured interviews and documents.

The three principles of data collection put forth by Yin (2009, p. 114) were followed when collecting the data, which are outlined in the Table 3. The principles where followed in order to increase the construct validity and reliability, detailed in Section 3.4.

Table 3: Three principles of data collection (Yin, 2009)

Principle	My approach
1: Use Multiple Sources of Evidence	I used participant observation, semi-structured interviews and documentation.
2: Create a Case Study Database	I organized and collected all of the raw data into a tool for qualitative data analysis. Additionally, I created case study notes daily in the form of reflection notes that included observations and relevant information.
3: Maintain a Chain of Evidence	I kept all of the raw data in the analysis tool organized with the time and place. Additionally, since I first coded the data, then grouped them into concepts, then categories, it is effortless to trace the evidentiary process backwards. This has also been maintained by keeping a clear distinction between the results and the discussion.

3.2.1 Observation

Through this study I was able to observe parts of an organization in order to get insight into their way of working, and therefore I observed various types of meetings, see Table 4. Furthermore, I spent entire days in the open work area where the relevant teams were situated in order to observe them in their natural habitat, conducting their everyday business. In total, I spent 21 entire working days in the organization's offices, as well as I observed 51 meetings. My observations were guided by an observation protocol based on Spradley (1980) and Stray et al. (2016), see Appendix D. I took notes during the meetings, and immediately after, where I included general information, such as number of attendees, start and end time. I also noted information related to coordination, such as whether they coordinated themselves, and how.

Table 4: An overview of the meetings observed

Observations	**Team Alpha**	**Team Beta**	**Team Gamma**	**Other**
Daily Stand-Up	18	9	12	0
Retrospective	2	1	0	0
Other meetings	3	4	2	4
Total:	**23**	**14**	**14**	**4**

Yin (2009) distinguishes between direct and participant observation. In this case I was merely a passive observer in the meetings, however, I did partake in casual social interactions such as lunch and coffee breaks. Therefore, my observations can therefore be regarded as participant observations. The value from being able to "perceive reality from the viewpoint of someone 'inside' the case study" (Yin, 2009, p. 112) was considered as large enough that it outweighed the risks of conducting participant observation. One significant risk is the researchers' bias, which is discussed in Section 3.4 and 6.3.

3.2.2 Interviews

As interviews are "one of the most important sources of case study information" (Yin, 2009, p. 106), I conducted eleven interviews, shown in Table 5. The interviews lasted from 33 to 53 minutes, depending on how detailed the interviewees answered.

The interviews were semi-structured. I followed an interview guide (Appendix A, B and C) in order to ensure that certain topics were covered, while it was possible to include other topics if relevant or desired by the interviewee. I prepared multiple interview guides depending on the interviewee, such as for the manager, business developer and developer. However, all of the interview guides had some core topics in common.

Table 5: An overview of the conducted interviews

Who	From	When	Duration
Digitally responsible	Team Alpha	January 2017	37 min
Project manager	Team Alpha	January 2017	46 min
Manager	Team Alpha	January 2017	52 min
Business developer	Team Alpha	January 2017	53 min
Team lead	Team Alpha	January 2017	51 min
Tech lead	Team Alpha	January 2017	50 min
Developer	Team Alpha	January 2017	38 min
Developer	Team Alpha	January 2017	33 min
Digitally responsible	Team Alpha	February 2017	34 min
Developer	Team Beta	February 2017	52 min
Manager	Team Beta	March 2017	45 min

Before I conducted the interviews, an application was sent to the Data Protection Official for Research, at the Norwegian Centre for Research Data, which was approved (see Appendix E). Therefore, with the acceptance of the interviewees, the interviews were recorded and stored on a secured platform, as in compliance with the application.

After the interviews were conducted, a hired consultant and I transcribed the interviews. The interviews that the consultant transcribed were still handled by me, because I listened to the interviews afterwards and made sure that the transcriptions were correct. To go over the transcriptions was additionally a valuable exercise for me because it refreshed my memory. Additionally, since the interviews were in Norwegian, I translated the selected quotes to English.

3.2.3 Documentation

Another source of evidence was documentation in the form of documents, systems, pictures and presentations. Table 6 contains a brief description of the documentation. These types of documentation were an important data source as they contributed to enhancing my understanding of the case's context.

Table 6: An overview of the collected documentation

Documentation	Description
Documents	Plans Reports
Systems	The software systems the unit under study was developing.
Pictures	Sketches of future systems.
Presentations	Progress reports Plans

Additionally, at most of the stand-ups I drew dialogue maps, which illustrates who spoke to whom. Every participant was represented with a circle, and then a line was drawn from one circle to the other when people spoke to each other, out of the regular round robin turn-taking. In that way, possible tendencies could emerge and be easier to notice.

3.3 Data Analysis

Before I started collecting data, I chose a strategy for data analysis. Yin (2009, p. 135) proposes four general strategies: "relying on theoretical propositions, case descriptions, a dual use of both quantitative and qualitative data, and rival explanations." My analysis was built on the theories presented in Chapter 2 on coordination, software development methodologies and teamwork.

The framework proposed by Van de Ven et al. (1976) was used to aid me in getting a thorough overview of the field of coordination and thus get a better understanding of the whole range of coordination mechanisms. This framework in particular, formed the basis for the data collection, just as Yin (2009, p. 130) states that theory helps focus attention on certain data and ignore other data. The results in Chapter 5 are therefore organized according to the framework (Van de Ven et al., 1976). An overview of the coordination modes based on the framework proposed by Van de Ven et al. (1976) follows in Figure 7.

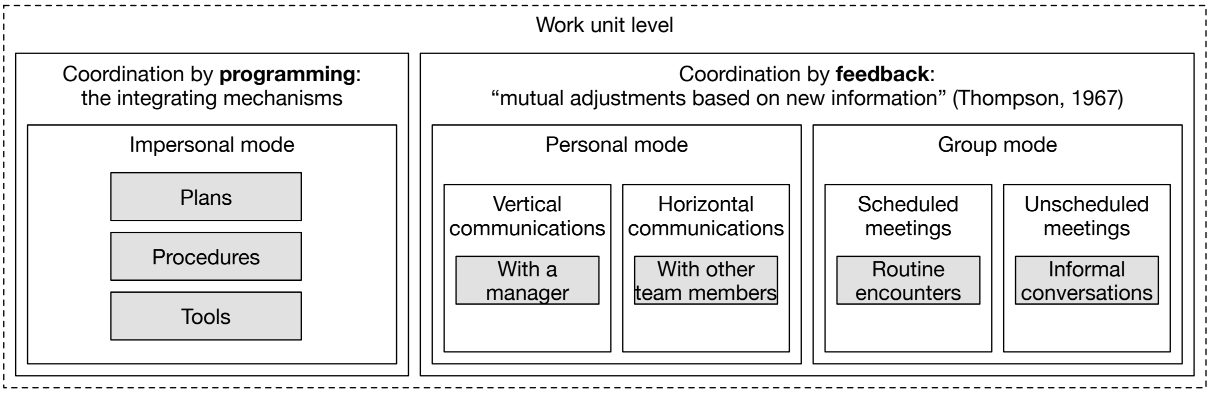


Figure 7: Coordination modes with examples in grey, based on Van de Ven et al. (1976)

Furthermore, in addition to following the framework outlined above, all of the data sources were uploaded into a program called NVivo[1], which is qualitative data analysis software. I coded the data using descriptive codes (Saldana, 2009). I created a start list of codes prior to observing the organization, as Miles and Huberman (1994, p. 58) recommends. During the study I revisited, edited and added codes when I considered that a code did not apply or if something new emerged. Throughout the whole process I kept a structure in the codes, based on the theories that formed the basis of the study, especially the framework (Van de Ven et al., 1976).

I coded both sentences and paragraphs, and grouped them into concepts, then categories, based on the method proposed by Glaser (2011). In that way, I organized my data and analyzed them on the topic of the different coordination mechanisms, as well as other relevant topics. By coding the data, I ensured that every claim I proposed was supported by data. An example of the coding process and the abstraction levels are shown in Figure 8.

[1] NVivo is a registered trademark of QSR International, www.qsrinternational.com

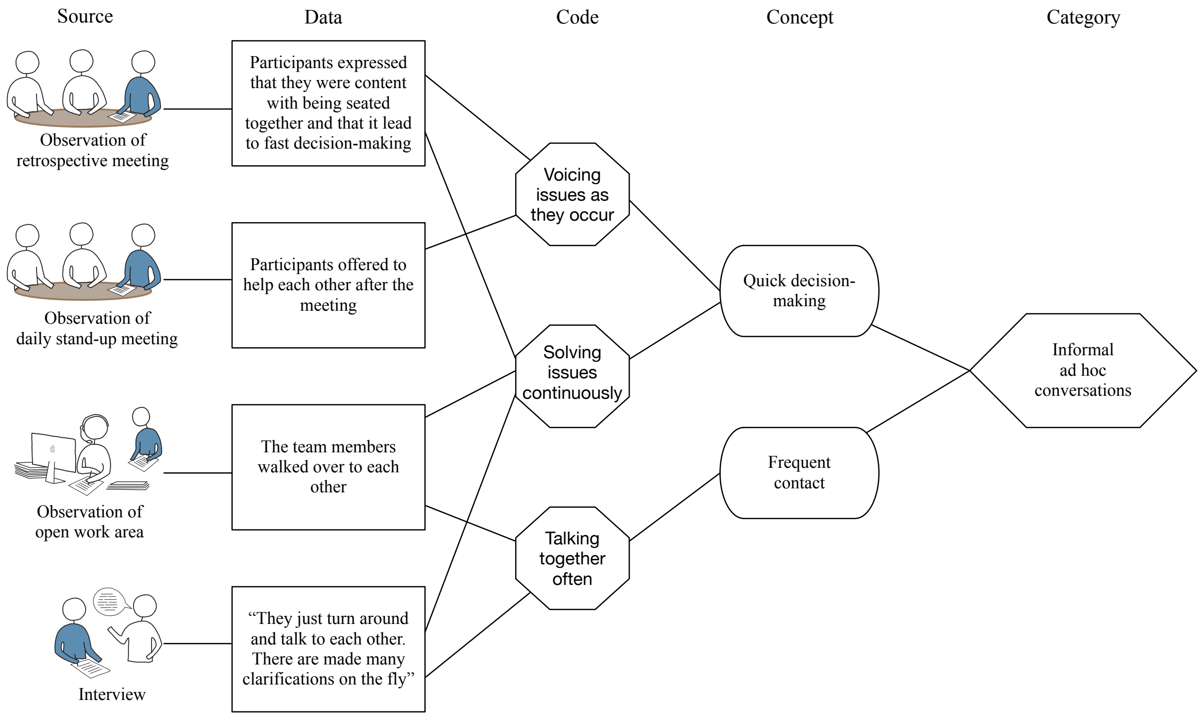

Source
Data
Code
Concept
Category
Observation of retrospective meeting
Participants expressed that they were content with being seated together and that it lead to fast decision-making
Observation of daily stand-up meeting
Participants offered to help each other after the meeting
Observation of open work area
The team members walked over to each other
Interview
"They just turn around and talk to each other. There are made many clarifications on the fly"
Voicing issues as they occur
Solving issues continuously
Talking together often
Quick decision-making
Frequent contact
Informal ad hoc conversations

Figure 8: An example of the coding process performed in this study

3.4 Validity

There are multiple threats to the validity of a qualitative case study, and my aim is to reduce the threats and subsequently increase the value of my thesis. The thesis' validity increases by reducing the threats such as reactivity, researcher bias and respondent bias (Robson, 2002).

The validity can be improved in various ways, as proposed by Wohlin et al. (2012, p. 69), by:

> "triangulation; developing and maintaining a detailed case study protocol; having designs, protocols, etc. reviewed by peer researchers; have collected data and obtained results reviewed by case subjects; spending sufficient time with the case, and giving sufficient concern to analysis of "negative cases", i.e. looking for theories that contradict your findings."

My approach to increasing the validities is outlined in the limitations section of the discussion in Chapter 6.

Construct validity

The construct validity is concerned with whether the researcher has used subjective judgment throughout the case study (Yin, 2009). Yin (2009) proposes three tactics to increase the construct validity of case studies. The first tactic is to use multiple sources of evidence during data collection (Yin, 2009), which by some is also called data triangulation (Patton, 2002). The second tactic is to establish a chain of evidence during the data collection, which was mentioned above as the third principle of data collection (Yin, 2009). The third tactic proposed by Yin (2009) is to have the key informants review a draft of your case study report.

Internal validity

The internal validity is relevant when causal relations are examined (Wohlin et al., 2012). Therefore, the internal validity is not relevant as this study is not trying to find causal relations.

External validity

The external validity is concerned with to what extent the findings from the case study are generalizable (Wohlin et al., 2012). It is important to distinguish statistical generalization from analytic generalization, because case studies only relies on the latter (Yin, 2009). The use of a theory increases the external validity (Yin, 2009).

Reliability

The reliability of a study is concerned with that the study would produce the same findings and conclusions if another researcher were to conduct the same study (Yin, 2009). The objective is to reduce errors and biases, such as the researchers' bias. Yin (2009) proposes two tactics to increase the reliability of a study. The first tactic is to use a case study protocol, whereas the second is to use a case study database (Yin, 2009). Overall, the general idea behind these tactics is that it should be possible to complete all the steps taken in a research in order to create the same results.

4 Research Context

The organization under study is a Norwegian bank, pension and insurance organization with more than 2,000 employees. This organization was chosen partially because it is part of a larger research project on agile methods for global software development. Moreover, it was chosen because they had merged a group of business representatives and IT developers into cross-functional teams. The research subjects under study are therefore two teams within this company called team Alpha and Beta. Both teams are multidisciplinary, and thus consist of both IT and business representatives. Therefore, they develop their products from the conceptual phase to the finished solution, the only difference being that one team develops for the retail market, and the other team for the business customers. In addition, these two teams cooperated with an offshore, outsourced team called Gamma, located in India.

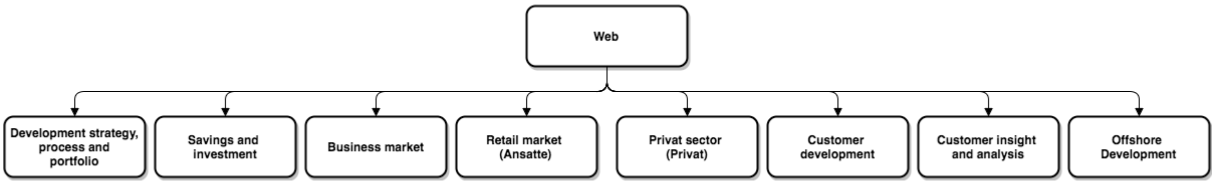


Figure 9: The organizational map of the organization under study

4.1 Team Alpha

The merging of business representatives and IT developers that lead to the interdisciplinary team Alpha took place in February 2016, hence the team is relatively young and inexperienced in working together across areas of expertise.

Team members

The team consists of seven business representatives, one project manager and five developers. However, some of the team members possess multiple roles. Of the seven business representatives there is one manager, three business developers, one digital designer and two with a role they called digitally responsible, meaning that they were responsible of keeping track of the IT development. Of the five developers from the IT department, there is one team lead, one tech lead, three developers and one tester.

In addition to the team members in Norway, team Alpha has two developers as a part of their extended team in the capital of an Eastern European country.

Moreover, the team lead in team Alpha also serves as the liaison between Norway and the Indian team Gamma, which is presented later on, and therefore also serves the role of Tech Liaison.

Seating arrangement

Even though the team consists of both business individuals and IT developers, they are located in the IT department. The two disciplines are sitting right next to each other and somewhat mixed, see Figure 10, except the test lead which is located with other testers, not that far away. One could therefore say they are rather closely knit.

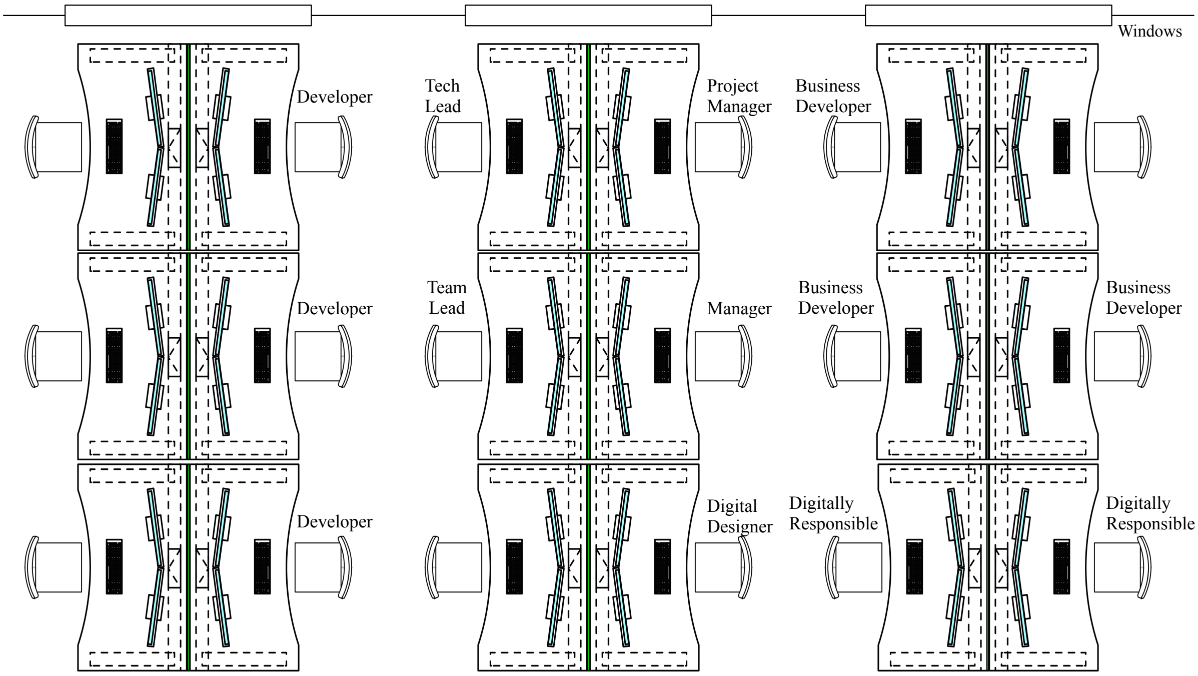


Figure 10: Team Alpha's seating arrangement

4.2 Team Beta

Whereas team Alpha was established in February 2016, team Beta united their developers and business representatives in November 2016. Team Beta is therefore even more inexperienced regarding the new business model than Alpha.

Team members

As with team Alpha, Beta has the same roles, the only difference being that Beta only has one digitally responsible, not two, and they do not have two developers offsite.

Seating arrangement

Team Beta is also located in the IT department, rather close to team Alpha, but not directly next to them. However, the teams are placed in open-plan offices and thus perceive closer to each other. All of the team members are located next to each other, including the test lead, see Figure 11.

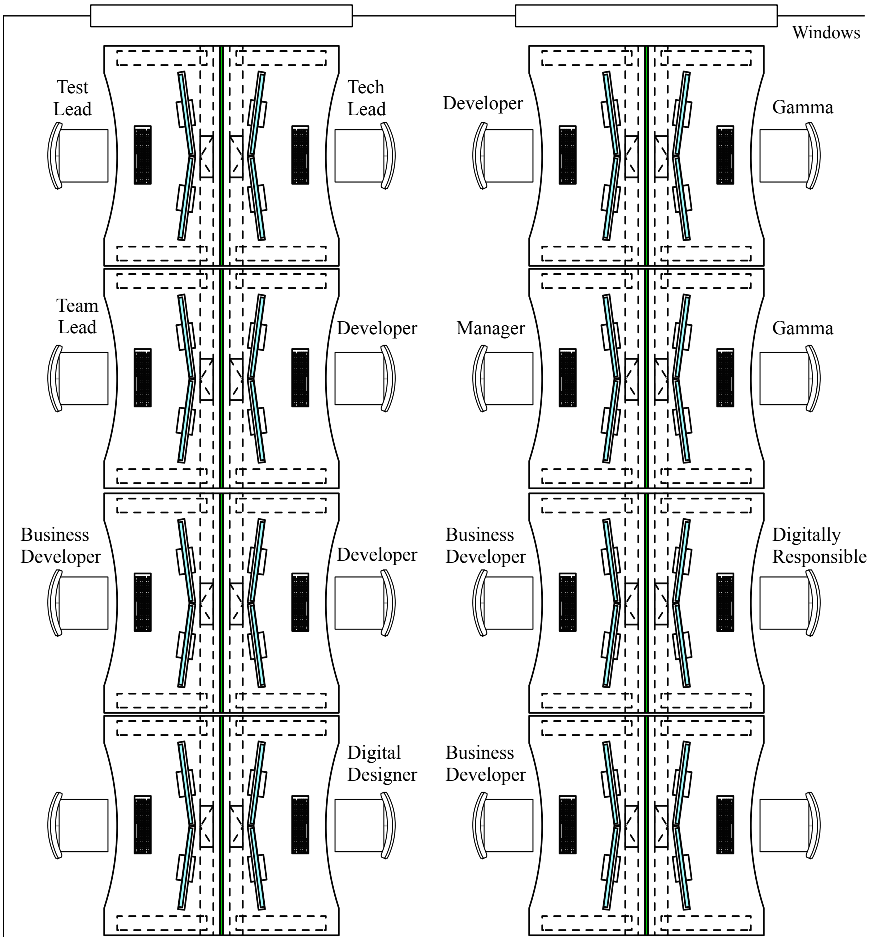


Figure 11: Team Beta's seating arrangement

4.3 Team Gamma

In addition to the two multidisciplinary teams above, the company has as of recent outsourced some of the team's work to an external team in India consisting of 9 developers as part of an outsourcing agreement. Two thirds of Gamma's human resources are allocated to work for team Alpha, while one third is allocated to tasks from team Beta.

Team members

The team Gamma consisted of an on-shore team lead, an off-shore team lead, an offshore architect, a project manager, developers, and testers.

Seating arrangement

While this team is outsourced, there are some resources on-site. There is one on-site team lead from India sitting in Norway. Additionally, team Gamma's project manager belongs to the Norwegian organization under study, and is therefore also situated on-site.

4.4 Roles

Since real life cases deviate from typical definition explanations, it is necessary to outline how the different roles are carried out in this case. An overview of the different roles is presented in Table 7.

Table 7: An overview of the different roles in the teams

Roles	Department	Team Alpha	Team Beta	Team Gamma
Manager	**Business**	1	1	
Business Developer	**Business**	3	3	
Digitally Responsible	**Business**	2	1	
Digital Designer	**Business**	1	1	
Project Manager	**Project Manager**	1	1	1
Team Lead	**IT**	1	1	1
Tech Lead	**IT**	1	1	
Developer	**IT**	3	3	
Test Lead	**Test**	1	1	
Offsite Developers	**IT**	2	0	9

Manager

As illustrated in the organizational chart (Figure 9) there are 8 different departments within the Web section of the firm, and in each of those departments there is one manager. In general, this person has a commercial responsibility for the team. Consequently, it means that the manager is responsible for keeping track of the team's stakeholders, which includes attending regular meetings, updating and following up on the team's commercial goal and monitoring how the team is delivering terms of reaching those goals. Moreover, it means that the manager is responsible for the commercial direction of the team. Additionally, the manager has the HR responsibility of the people employed in the business department, which is less than half of the team members.

Business Developer

The typical task a business developer works with, is the realization of an idea, meaning to propose a product or service as a possible solution to a known problem or desire in order to meet the team's goal. A business developer generally works with an idea from the concept phase all through to having sketches of a finished product, with help from the Digital Designer.

Digitally Responsible

The digitally responsible are employed in the business department and thus their knowledge and expertise is therefore within the field of business. However, the difference between the digitally responsible and the business developer is that the digitally responsible has more knowledge and experience of how to develop a business through digital means. Consequently, it includes knowing how customers use their different digital gadgets, having knowledge of different technologies and knowing what is possible in the world of technology.

Digital Designer

The digital designer works with creating sketches of possible user interfaces of products and services that the business developers and digitally responsible proposes. The digital designer has typically knowledge of or experience with user experience design (UX) and graphical design.

Project Manager

The project managers are organized in a separate department. In general, the project manager is responsible for the planning and execution of a certain project, whereas in this case the project manager is responsible for following up on the projects that the developers in that particular team are working on.

Team Lead

The team lead is the head and organizer of the developers. One of the team lead's functions is to shield the developers against unwanted noise and disturbances. External requests typically go through this person and are forwarded to the appropriate person. However, if the developers do not have capacity to complete the request, it might be denied. Additionally, the team lead is also a developer.

Tech Lead

The tech lead is responsible for keeping track of the different technologies used, as well as having in-depth knowledge of their services' APIs. Additionally, the tech lead is also a developer.

Test Lead

The test leader is responsible for thoroughly testing the solutions the developers create in order to find errors and bugs. The test lead is organized in a separate department, with the other testers in the organization.

Tech Liaison

The tech liaison is the intermediary role between the Norwegian team leads and the onsite team lead for the offshore team. The purpose of this role is to ensure good communication between the onsite and offsite developers.

Developer

The developers in this case consist of both front-end and back-end developers. The developers do not work together on the same systems, but separately.

Similarly to the on-site developers, the off-site developers also consist of both front-end and back-end developers. There are two different types of off-site developers in this case; those that belong to the extended team, the second site of team Alpha, and those that belong to the external team that is the outsourced team.

5 Results

This chapter describes the different coordination mechanisms using a framework proposed by Van de Ven et al. (1976). I have focused on describing the various coordination mechanisms because the research question examines coordination in a BizDev team, and coordination mechanisms are a representation of coordination. As stated in Chapter 3 regarding the research method, these findings are discovered in team Alpha, but with findings from team Beta and Gamma to help further understand team Alpha.

As outlined in Chapter 2, Van de Ven et al. (1976) Van de Ven et al. divides coordination [mechanisms] into three main categories; impersonal, personal and group mode. Furthermore, the personal mode is split into vertical and horizontal communications, and group mode into scheduled and unscheduled meetings. Table 8 shows an overview of the coordination mechanisms that I uncovered in this case study, 24 in total. My findings are based on the conducted interviews, observations and documentation that I gathered.

After the overview follows the coordination mechanisms, listed according to which category they belong to, and what I found on each of them. Unless otherwise specified, the coordination mechanisms were present in both team Alpha and Beta.

Table 8: Overview of the coordination mechanisms in this case study

No.	Coordination Mechanisms					
		Programming	Feedback			
		Impersonal Mode	Personal Mode		Group Mode	
			Vertical	Horizontal	Scheduled	Unscheduled
1.	Project Management Tool	X				
2.	Wiki	X				
3.	Instant Messaging	X				
4.	Communication and Collaboration Tool	X				
5.	Monitor Application Performance Tool	X				
6.	Measure Application Performance Tool	X				
7.	Methodology: Lean Startup	X				
8.	Customer Journey Map	X				
9.	Deployment Rules	X				
10.	Quality Assurance Rules	X				
11.	User Support	X				
12.	Agile Process	X				
13.	Open Work Area	X				
14.	One-on-one Meeting		X			
15.	Informal ad hoc conversations			X		X
16.	Daily Stand-Up Meetings				X	
17.	Story Mapping				X	
18.	Retrospective				X	
19.	Department Meetings				X	
20.	Team Meetings				X	
21.	Demo for Stakeholders				X	
22.	Demo with Gamma				X	
23.	Backlog Grooming with Gamma				X	
24.	Sprint Planning with Gamma				X	

5.1 Impersonal Mode

Typically, impersonal coordination mechanisms include plans, procedures, standardized information and communication systems. I found 13 coordination mechanisms that could be categorized as impersonal coordination mechanisms.

Project Management Tool - Jira[2]

Jira is a project management tool where tasks are created and organized, then assigned to the person responsible for that particular task. Before a task is entered into Jira, the team members estimate how long it will take to complete the task, usually at a Story Mapping session. Then the digitally responsible writes user stories for each task, which are entered into Jira, where it is assigned an identifiable number. As an impersonal coordination mechanism, Jira was used both as a communication tool and as the tool that organized the team's tasks and issues at hand.

Wiki - Confluence[3]

Confluence is a team collaboration tool. In this study, they used it mainly to store different system's documentation. During the interviews, the manager mentioned that it was used for storing a system's technical specifications. The tech lead stated that it contained the tech overview of the associated components and to how to operate simple management. The tech lead also stated that the main purpose for sharing this information in such an arena was in order for people to find information themselves without having to always ask the developers, and to reduce the personal dependence.

Instant Messaging - HipChat[4]

The team made use of several tools for instant messaging, such as Skype[5] and HipChat. The project manager stated that these tools were mainly used for information exchange. Examples of information shared, stated by the tech lead, were reminders to the other team members, or informing if something in the system was down or delivered. In addition, the arena was also

[2] Jira is a registered trademark of Atlassian, Inc., www.atlassian.com/software/jira.

[3] Confluence is a registered trademark of Atlassian, Inc., www.atlassian.com/software/confluence.

[4] HipChat is a registered trademark of Atlassian, Inc., www.atlassian.com/software/hipchat.

[5] Skype is a registered trademark of Skype Technologies, www.skype.com/en.

rather social, stated the project manager, with the example that a developer could share a picture if his/her feature was put into production from home. Moreover, the project manager stated that to a large extent, the developers used HipChat only amongst themselves. However, a digitally responsible had joined the platform, meanwhile the rest of the business representatives were hesitant, and had not joined. Once the team organized their daily stand-up in appear.in[6] through sending each other the link on HipChat. At several occasions, the team members were observed having important work-related conversations with the developers off-site.

Enterprise Communication and Collaboration Tool - Workplace[7]

While HipChat was used almost exclusively by the developers, Workplace covered the same need, but was used organization-wide. More or less all employees were members of Workplace, however it was a much less relevant coordination mechanism within teams, but rather relevant for inter-team coordination.

Monitor Application Performance Tool - New Relic[8]

New Relic, and more specifically the module called Insights[9] provided real-time analytics of how the software performed. In this case the tool helped the team understand and learn about their applications performance. It made data available about the customers' behavior and how the customers use their solutions. In addition to creating performance data, the tool had the benefit of working as a coordination mechanism that gathered the colleagues, as a developer pointed out:

> *"It makes things more visible. You see that the team members gather around a computer screen and discuss New Relic-statistics, and that what you're making is being used and that it creates enthusiasm. That's good."*

Furthermore, at a team meeting, the manager used the results from New Relic Insights to show that the system that the team had developed was doing well.

[6] Appear.in is a registered trademark of Telenor Digital AS, www.appear.in.
[7] Workplace is a registered trademark of Facebook, www.facebook.com/workplace.
[8] New Relic is a registered trademark of New Relic, Inc., www.newrelic.com.
[9] Insights is a registered trademark of New Relic, Inc., www.newrelic.com/insights.

Measure Application Performance Tool - Google Analytics[10]

While New Relic was an initiative taken within the team, Google Analytics was requested by the marketing department. This tool was primarily used in order to get an overview of the different applications' traffic.

Methodology: Lean Startup

Both team Alpha and Beta followed the Lean Startup way of thinking when developing their ideas and concepts. However, they called it lean innovation, but when asked the question, they meant Lean Startup. Team Alpha started adopting the methodology earlier in 2016, while team Beta began at the end of the year. A business developer described how they worked with this specific approach:

> *"I think about how we can build this step by step, as a Lean Innovation process; what is the minimum viable product that will create value for us, in order to prove that what we are producing creates additional value? How do we get there without making this into a large project that we never will manage to deliver?"*

Moreover, a manager described how the methodology helped the team make a wiser use of their resources:

> *"The business side always test an idea on customers before it is sent to development and having resources allocated to it, in order to make sure it is viable. That way of working is the result of following the Lean Innovation methodology."*

Furthermore, the conceptual testing is also an important part of the methodology, which a digitally responsible stated:

> *"We start with finding the concepts, then start the work on the concepts, then test the concepts in order to see what we will continue working on."*

A manager also pointed out how the methodology helped them to start prototyping, and how important testing is in order to create the right product:

[10] Google Analytics is a registered trademark of Google, www.google.com/analytics.

> *"That's when we started with prototyping, the simple testing against customers in order to get feedback, then go back, iterate and continue testing. We spent some time late last year learning how to speed up the process."*

Customer Journey Map

The customer journey methodology was introduced late 2016, therefore the approach was less established. In this case the tool was divided into seven customer journeys, where the manager mentioned the following: become customer, be a customer, passive customer, settlement, and terminate. With some experience with the tool, a manager described it as such:

> *"We have an end-to-end perspective on what we create, which makes the scope very wide. The customer journey map methodology helps us narrow it down, by dividing the entire journey into smaller parts. It helps us discover what stakeholders needs to be involved when and what we need to solve to make it a solid solution."*

Deployment Rules

The team lead in large part completes the deployment because the person knows the rules well, which means that the developers can continue their tasks while the team lead handles the process.

Quality Assurance Rules

When a developer has finished coding a technical solution, a different developer has to complete Quality Assurance (QA) on the code and is conducted by following a set of rules. This activity is performed in order to ensure a certain level on the source code.

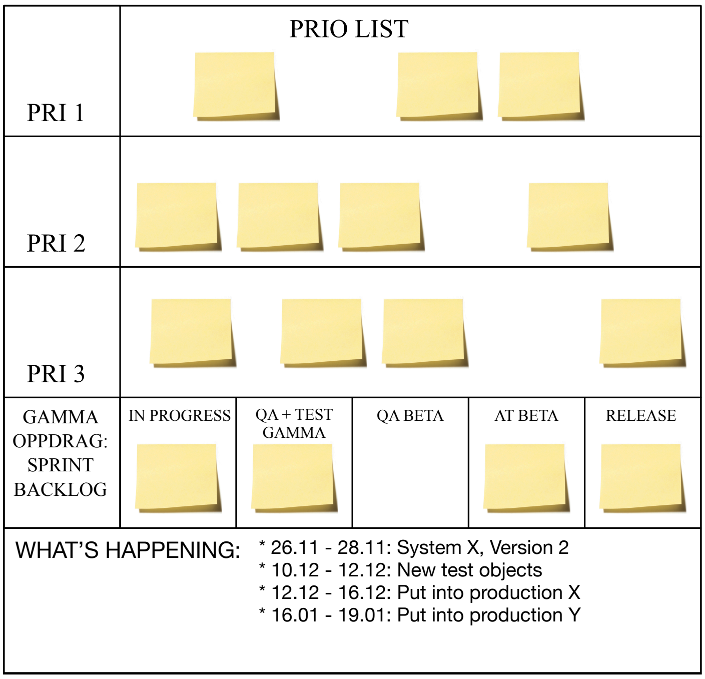
User Support

User support is the procedure for handling errors in production. The errors were reported by customers calling to the customer service or employees such as the key account managers and testers. The team lead described the process as:

> *"The user support system is much more formalized now. Before, the business representatives forwarded the emails containing the reported errors to individual developers, and they simply piled up in the inbox, because there was no system."*

Agile process

At first, the teams were asked to follow Scrum, but after a period of time both Alpha and Beta decided to change to Scrumban. They wanted the flow from the Kanban board, but continue to follow the Scrum ceremonies such as the daily stand-up meeting, retrospective and having the scrum master. The main reason for the change was that they did not want to work in sprints. The Figure 12 illustrates how their Scrumban board is divided into priorities with the different lanes typical to a Kanban board.



* 26.11 - 28.11: System X, Version 2
* 10.12 - 12.12: New test objects
* 12.12 - 16.12: Put into production X
* 16.01 - 19.01: Put into production Y

Figure 12: The Scrumban board

Open work area

As described in Section 4.2.2 and 4.3.2, the teams are seated in an open work area, both business and development together, see Figure 13 for a picture of the area. This arrangement facilitates coordination by providing easy access to the other employees and an area where the team members can discuss their tasks and solutions. The coordination mechanism is an enabler for the highly valued informal ad hoc conversation, outlined in the section below. The seating arrangement was highly valued, as confirmed by a team member:

> *"That is what is good now, that we are seated together. It is not necessary to organized meetings to figure things out, because we talk to each other. And that is a much better way of working."*

However, even though the open work area seating arrangement was highly valued, it also brought along the challenge of shielding the developers. It became evident rather early, after business and development was situated together, that they have different needs in their working environment. On one hand, business wanted a very open work area with the possibility of discussions and temporary seats for guests to be seated, for example. On the other hand, development wanted designated seats because their desktop computer contained specific hardware and multiple monitors, and they needed a quiet zone for focusing on their programming.

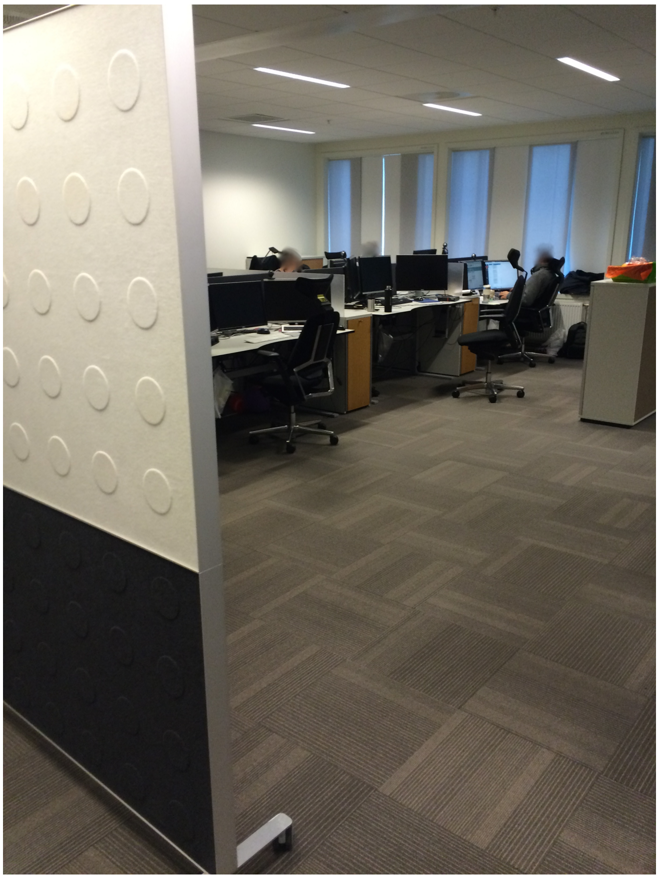


Figure 13: The open work area where the team members work

5.2 Personal Mode

As outlined in Chapter 2.2 on coordination, the personal mode includes coordination mechanisms between two individuals; communication with a line manager or a co-worker. See Table 9 for an overview of the personal coordination mechanisms and their frequency.

The present coordination mechanisms that belong to this category are therefore described here.

Table 9: The personal coordination mechanisms discovered in this study

Coordination Mechanisms	**Frequency**
One-on-one Meeting	Every other week
Informal ad hoc conversations	All the time

5.2.1 Vertical communications

One-on-one Meeting

While the developers had a more traditional employee review meeting with their supervisor, the manager of the business representatives had a regular meeting called one-on-one, every other week. This meeting did not revolve around personal or professional evolvement, but on operational issues. The business developers could give a status update on their tasks, and they could discuss different areas that pose challenges, such as cooperation, stakeholders and their solutions.

5.2.2 Horizontal communications

The horizontal communications represent the regular communication between co-workers in an organization. Therefore, before reporting on these findings, it is crucial to understand the team dynamics in order to properly understand how the team members communicate to one another.

Introduction to the horizontal communications

As outlined in Chapter 2.4 on Teamwork, there are different characteristics on a team and a working group, and according to those characteristics I am mapping where the unit under study belongs. According to the findings in team literature, the characteristics of a team involve the unit size, whether it has a common goal, a set of performance goals, a common purpose and whether they hold themselves mutually accountable. This introduction to the horizontal communications therefore outlines those characteristics in the unit under study.

Common goal

When asked the question if the team has a common goal, almost all of the team members said yes, however none of them could actually provide the common goal. Some members admitted to not knowing the goal, with for example that a business representative answered that he/she hoped there was one. Other team members provided a very thin explanation of the goal, such as this person:

> *"I guess we have quite clear goals on what to achieve. Then we try to work towards that. Everyone has set up a plan, not that I know where they get these estimates from, but that's just the way they want it to look."*

Moreover, one team member even stated that the person did not have the impression that they had a common goal:

> *"No, I would say that it does not feel that way. At least not for 2016, because we worked on such different things. So no, I do not think so."*

Knowledge of other's work

During the interviews, there was a reoccurring tendency that the interviewee did not know what the other team members were working on. Out of three members that were asked, two members did not even know what business area six of the other team member was working in, while one member did not have any knowledge of the domain details or specification in which one of the other team members worked on. Business area refers to the general name of an area in which they are operating, such as bank, finance, etc. Moreover, domain details or specification refers to more in-depth knowledge of either the business area for the business representatives, or the technical solution for the developers. It is therefore safe to say that in general, the team members had little or limited knowledge of the other member's work. Multiple team members stated that they did not know what the other team members were working on, both business and development. One example is the following statement:

> *"I have no idea what he/she is doing, and he/she has no idea what I am doing. I have no overview."*

And when a team member was asked whether he/she knew that the other developers were working on, the team member answered:

> *"I know what they are working on, or at least I think I know."*

Communication

As stated in the interview guide, I identified the different lines of communication within the team by asking whom the team members talked to on a day-to-day basis when solving their tasks. Figure 14 shows the team members, except the project manager and the manager, because the purpose of this information was to outline how the team members worked together to solve their tasks. A manager does not work on a task per se, but administers the others and facilitates so that they can do their job the best possible. The blue color means that the person is a business employee, while the red color stands for IT. The grey fill means that the person was not interviewed and therefore did not get the chance to give their input.

As the figure is visualizing, the team was quite divided, in two. One can argue that there were signs of subgroups within the larger group. Roughly speaking, there was one developer per two business developers. Both the interviews and observations are the sources to Figure 14. Figure 14 illustrates that the group was very divided.

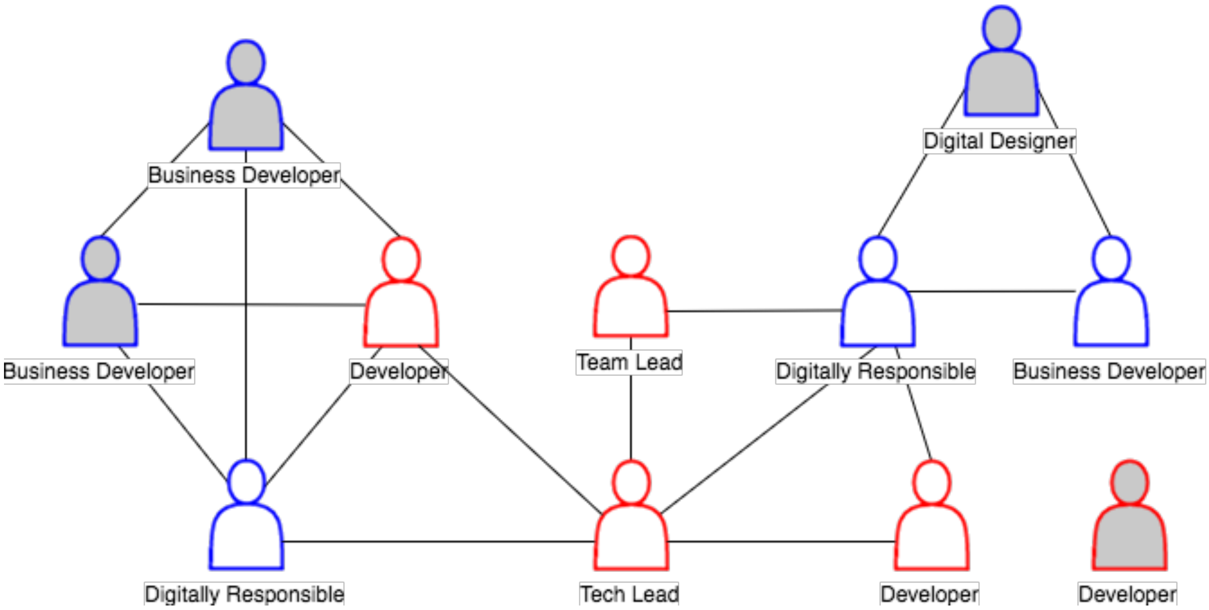


Figure 14: The communication lines in the unit of analysis

The interviews are the source to the Figure 14, however, the participant observations that were done on a daily basis backs up the figure to a large extent. To a certain degree the team members seemed to be working even more divided than what the figure depicts.

One role that it is important to take note of is the digitally responsible, because this role, to a great extent, bridged the gap between business and development. Therefore, even though the team was quite divided and to a large part did not talk to each other, several of the team members did inform that they talked to this role, both the business developers and the developers. During both the interviews and the participant observation, this role emerged very clearly as the "glue" between the IT developers and the business developers.

Prioritizations

Furthermore, in addition to the team characteristics above, it is crucial to understand how the team receive new tasks from external parts of the organization, in order to fully understand how the team operates. Figure 15 illustrates what happens in the unit of analysis when external priorities are being set. In general, these activities are put into place if there is a change in the external priorities, not if they are the same as before. Typically, it is the manager above the team's manager that decides the external overall prioritization, and then the team is able to decide its own priorities, within the limits set externally. Then the digital designer proposes sketches on what the new products could look like, which is a contributor to the discussions at the story mapping session. There are often iterations on those three activities because information discovered during story mapping could affect the team's internal priorities. After story mapping, the digitally responsible writes the user stories on the first MVP that is going to development. Last, if it is not evident which developer does what, they discuss and distribute the tasks amongst themselves.

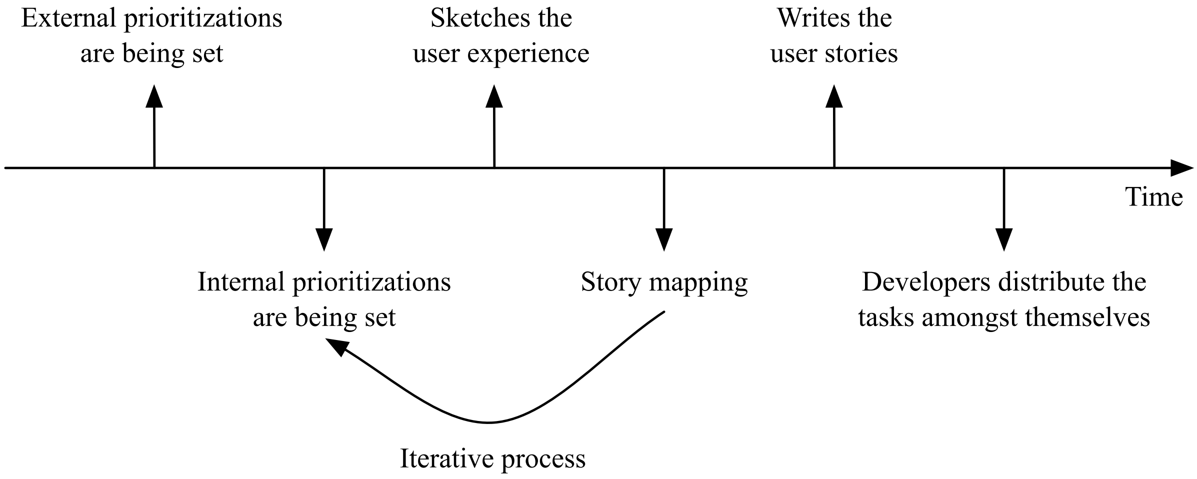

External prioritizations are being set
Sketches the user experience
Writes the user stories
Time
Internal prioritizations are being set
Story mapping
Developers distribute the tasks amongst themselves
Iterative process

Figure 15: A timeline of how new tasks reached the unit of analysis

Informal ad hoc conversations

Even though this coordination mechanism is located under the personal mode of coordination, it is also an example of group mode coordination. The informal conversations took place between two people and more. I observed the team members engage in informal ad hoc conversations very frequently, several times a day. And when asked how often this occurred, the team lead stated that it was very many times a day, every day. They made quick decisions, which created speed for the development. The project manager confirms the fast decision making:

> *"They just turn around and talk to each other. There are made many clarifications on the fly."*

Additionally, a business representative answered the following when asked in which arenas the collaboration between business and development took place:

> *"I would say that it is mostly informal collaboration. We make decisions along the way, while we work on our tasks, when it is needed. That is our largest arena for collaboration, I believe. If there is something which is under development, and an issue occurs, it happens that we handle it informally by talking with the developers about solving it differently."*

Furthermore, it seems that informal ad hoc conversations were regarded as the most efficient way of coordination. A manager claimed that sitting together had made the dialogue very much better, and that the most important communication is in person, and not digitally. The efficiency was also stated by a team member:

> *"I think to make the communication more informal and have less regularly scheduled meetings is perhaps the way to go."*

In addition to being regarded as the most efficient coordination mechanism, it also seemed to be preferred, in which a developer stated:

> *"I simply just walk over and talk, it's more effective than writing a Jira issue with the overhead it entails. As a consequence, I have very few e-mails, very few meetings, and that's great."*

5.3 Group Mode

As outlined in Chapter 2.2 on coordination, the group mode includes coordination mechanisms between three or more individuals, in scheduled or unscheduled meetings.
See Table 10 for an overview of the group coordination mechanisms and their frequency.
The present coordination mechanisms that fall under this category are therefore described here.

Table 10: The group coordination mechanisms discovered in this study

Coordination Mechanisms	**Frequency**
Daily Stand-Up Meetings	Daily
Story Mapping	Differing, every quarter
Retrospective	Differing
Department Meetings	Once a month & upon request
Team Meetings	Every week
Demo for Stakeholders	Differing
Demo, Gamma	Every other week
Backlog Grooming, Gamma	Every other week
Sprint Planning, Gamma	Every other week

5.3.1 Scheduled meetings

Daily Stand-Up Meetings

Team Alpha started their stand-up meeting every morning at 10:45a.m., in which they followed up with lunch afterwards. Team Alpha conducted their stand-up meeting in a meeting room because they had the extended team joined via video.

Meanwhile, team Beta carried out their stand-up at 9:45a.m. Team Beta carried out the meeting in their open work space, adjacent to their seats, and used a permanent board as their Scrumban board containing their current tasks.

The interviewees that participated in the stand-ups, which was eight persons, were asked to rate how satisfied they were with the ceremony on a scale from one to five where one was not satisfied, and five very satisfied. The average rating was 3.3. However, the two developers that had no additional role such as tech lead, team lead or scrum master, gave the meeting a score of two. One developer stated that they are working on such different things and that there have not been many times where they have benefited from sharing with each other what they do. Another developer stated similarly:

> *"Everyone in the team work with different applications, so there is not much valuable information really. I think there's too little discussions regarding obstacles or opportunities to receive help, it's more like giving a status report."*

In contrast, the highest rating was four, and while the team lead stated that the stand-up often turned into a status update in order to know what the team is working on, the person also believed that the meeting provides a good way of getting a team spirit. Meanwhile, the project manager gave his/her reasoning for the rating with:

> *"I need to know the status."*

Another team member that was satisfied with the meeting was the team lead, which gave the following reasons:

> *"I am pleased with stand-up because it forces me to reflect on what I spend my time on. And it uncovers if someone is stuck on the same problem over time."*

It is evident that stand-up more often function as a status update, and less frequently actually solve problems. This claim is supported by the ratings, because the team members with roles such as team lead, tech lead and project manager were more satisfied than the developers. That is, the roles that include a more overall responsibility for the team and their progress, were more satisfied. However, there are still multiple reasons for why stand-up is adding value to the team, such as by creating and enhancing a team spirit and solving issues.

The dialog diagrams I drew also show that there was little coordination at the stand-up meetings. It was seldom that something other than the regular round robin turn-taking took place. Figure 16 illustrates a typical stand-up meeting where four people said something of value out of the regular turn, as shown with the lines drawn between them. The circles are the team members, SM stands for scrum master and the square means that the person was connected by a screen.

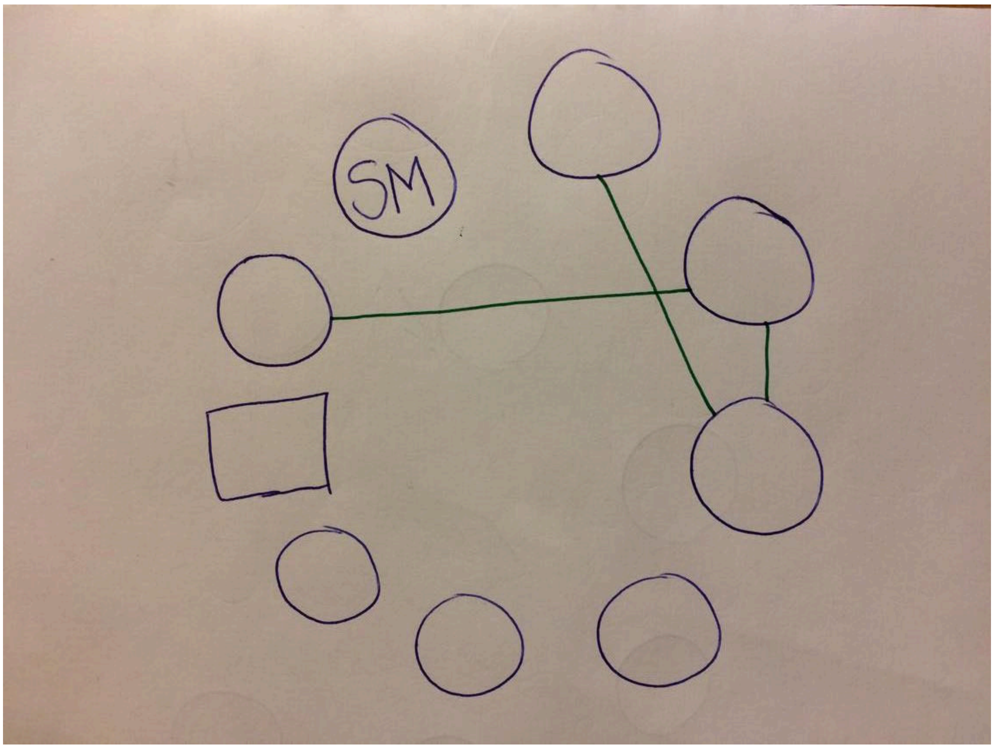


Figure 16: An example of a dialogue diagram

However, team members from team Alpha and Beta also participated in Stand-up meetings with team Gamma. Every morning, from 9:00am to 9:30am, the whole Gamma team had two stand-ups consecutively. The first meeting was with members from Alpha, the second with Beta. The members that participated from the Norwegian team, Alpha or Beta, were the tech liaison, the test manager, the relevant business representative and sometimes the project manager. These stand-ups were conducted so that the off-site members of Gamma were participating through a video call, while the rest were seated in a meeting room. The meeting room was equipped with a round table and two TV screens, where one screen showed the video call and the other showed their Jira board. The off-site team members could also see both the people in Norway, as well as the Jira board.

Story Mapping

Story mapping is a tool used to uncover the minimum viable product (MVP), known from Lean Startup (Ries, 2011). It was organized as a workshop where the team reached an agreement on what to work on and the order of the different tasks, conducted by the project manager, and was the basis for the quarterly planning. This workshop included both the business representatives and the developers, and therefore worked as the only common meeting arena for all team members. A picture of the MVP created during a story mapping session follows in Figure 17. The coordination mechanism was also regarded as a well-functioning technique by the manager:

> *"It has worked very well in terms of dividing a large system into minimum viable products, which is used a starting point for prioritizing tasks and knowing what we are working on at all times."*

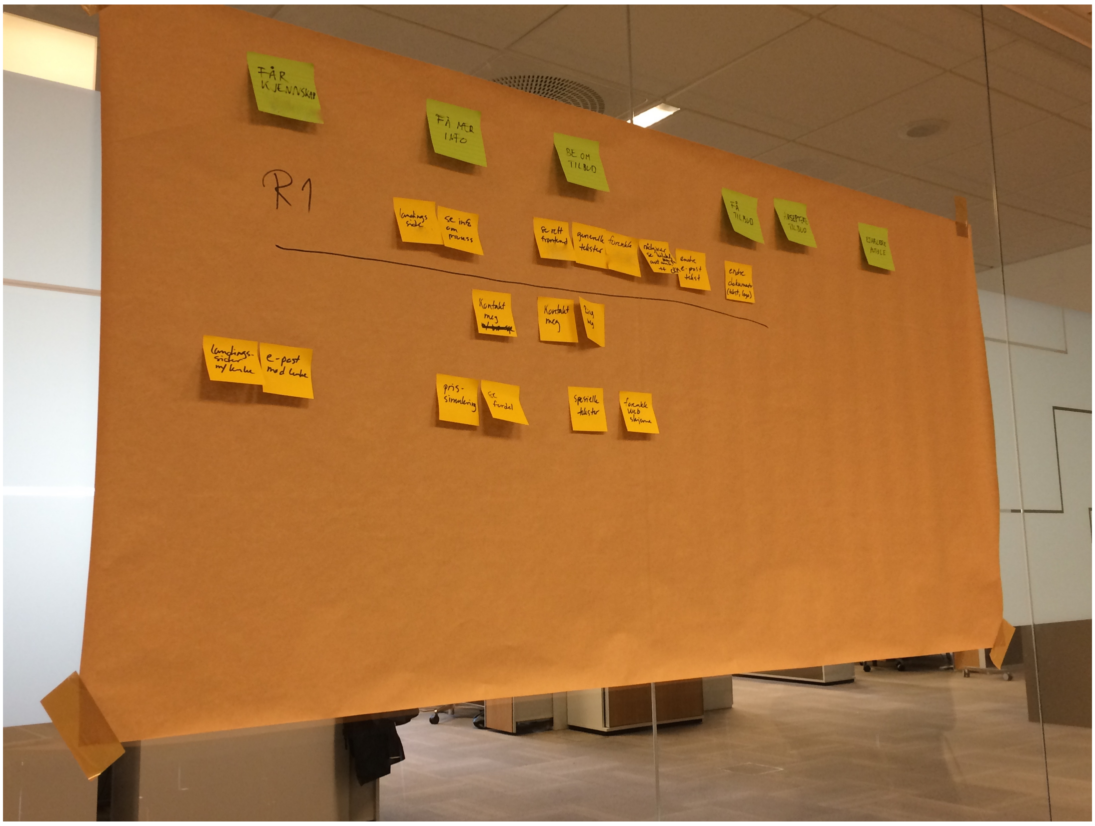


Figure 17: Creating the minimum viable product in a story mapping session

Moreover, since the resulting MVPs from the story mapping were the basis for the quarterly planning, the plans derived from this activity were important to the team. The plans created as a result of a story mapping session hung on the wall next to the team in the open work area.

Retrospective meeting

The retrospective meetings have been held periodically, by the project manager, although rather seldom. The team lead stated:

> *"We have retrospectives on a regular basis, although we should have it more often."*

Team Alpha had a retrospective meeting in March 2016 where they wanted synergy with a different development team, and they also proposed the idea that merging the team with theirs was a possible solution. Consequently, the two teams were merged in October 2016. This is one example of the purpose of the retrospective meeting, and where the execution was successful.

One possible exercise in the retrospective meeting is called the weather forecast, where the team members draw their own vision of the future work situation in terms of a weather forecast. A picture of the exercise conducted with the unit of analysis in Figure 18.

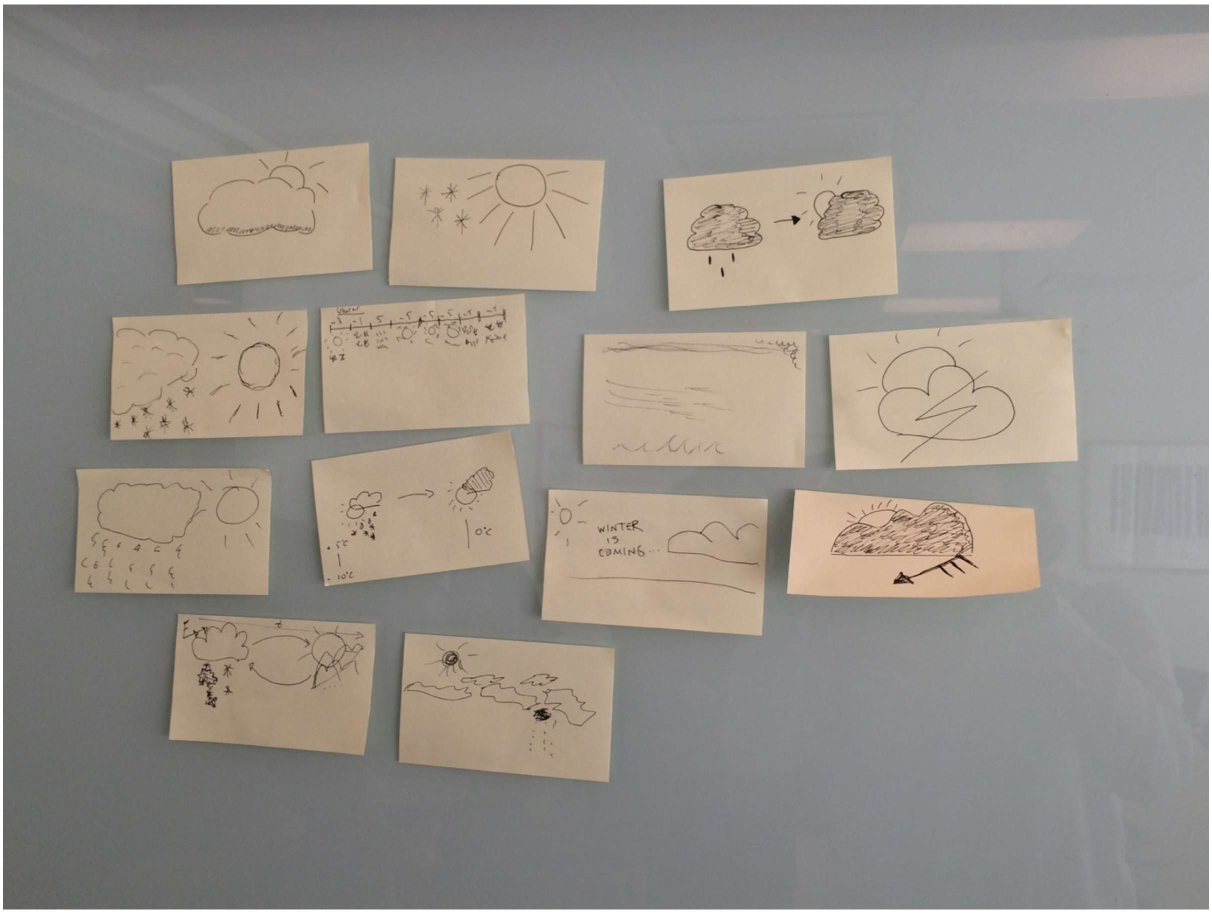


Figure 18: The weather forecast exercise in a retrospective meeting

Department Meetings

The business representatives had regular meetings, once a month, with different external stakeholders, such as the sales department, the marketing department and the product department. In general, all of these departments were interested in the solutions that the teams created, because they in one way or the other were affected by changes to the company's product portfolio.

The meetings with the sales department were held for mutual exchange of information. The business representatives from the team informed the sales department on what they were working on. This information was exchanged because the sales department had to know of changes to their products, and understand them, because the solutions were created for the customers, but also so that the sales department, including the customer service and the advisers, could use them. A business representative stated:

> *"We have many meetings with the sales department, because they need to know and understand as soon as we make a change because customers might contact them. They need to know that what we create is in order to help them sell."*

Moreover, the sales department informed the business developers of what they typically were talking to the customers about in order to provide them with an insight into the customers' issues and wants.

Furthermore, since the marketing department launched and promoted the different solutions the team created, these regular meetings were of great importance. The manager expressed this importance by stating that the marketing department is their most important stakeholder.

The business representatives had a considerable amount of communication with this department, as stated by a business representative:

> *"There is a lot of contact with the marketing department, because they are the ones launching the systems that we create. The might have very strong opinions on what the solutions should look like. Therefore, we try to have very regularly scheduled meetings with them in order to understand each other and build trust. There is a lot of coordination with them."*

The product department was the formal owner of the products the company provides to their customers. Therefore, the meetings with them were important if they needed assistance with such as product rules, as a digitally responsible stated:

> *"They are responsible for making sure that our products follow the statutory rules, and the product rules that the organization has created."*

Team Meetings

Table 11 illustrates which team meeting each of the team members attend.

Table 11: An overview of the team meetings' attendees

Attendants	**The weekly meeting**	**Team meeting**
Manager	X	X
Business developers	X	X
Digitally responsible	X	X
Project manager	X	1/2
Team lead	X	-
Tech lead	X	-
Developers	-	-

There were two different team meetings, including different people. One was called the weekly meeting and contained all of the team members, except the developers. The purpose of this meeting was for the business representatives to get a status update on development, from the team lead of the developers. The team lead informed about what the developers had completed, and more generally on how far they had come on the prioritized tasks. After the team lead had given the update, the business representatives informed about what they were working on, which was at earlier stages, and therefore not ready for development yet. Examples are ideas they were exploring, prototyping or still gathering information on.

The goal of the meeting was to create transparency between business and development, and amongst the business representatives, as stated by the manager:

> *"We have tried to make it as transparent as possible, so that no one is very surprised when changes happen. I think it has worked to some extent, while it still is a challenge. When someone is not involved, it creates a divide in terms of taking ownership of the things that the team creates."*

Additionally, the team had a department meeting for the business representatives because the manager had the line management responsibility for these team members, named team meeting. This meeting did therefore not include the project manager, team lead, tech lead, or developers. One could say the name team meeting was misleading. These meetings lasted for two hours, and were divided in two. The manager stated what the first hour contained information about:

> *"What meetings have I been involved in, and what information have I received from my manager in order to ensure transparency in terms of what is happening in the periphery of the team."*

The project manager was invited to the last hour because that contained a professional update in which the manager thought was useful for the project manager to accompany. Overall, the theme of these updates was how the company makes money and understand the business of how they make money on insurance, for example.

Demo for Stakeholders

The team had regular demos that included a bigger perspective on what they were to present. These demos were primarily for their different stakeholders: team lead

> *"We arrange demos for the stakeholders that includes some information on the business reasoning on why we work as we do, and how we are going to reach our sales goals."*

Coordination Mechanisms with Gamma

At the following coordination mechanisms were members from team Alpha and/or Beta present. An overview of the mechanisms follows in Table 12.

Table 12: The coordination mechanisms with Gamma discovered in this study

Attendees	Demo	Backlog grooming	Sprint planning	Retrospective	Stand-up
Team Alpha/Beta					
Business Representative	X*	X*	X*	X*	X*
Project Manager	X*	X*	X*		X*
Tech Liaison	X	X	X	X	X
Test Manager	X	X	X	X	X
Team Gamma					
Project Manager	X	X	X	X	X
Onshore Team Lead	X	X	X	X	X
Offshore Team Lead	X	X	X	X	X
Developers					

*** = when relevant.**

Demo with Gamma

At the Demo meeting, Gamma demonstrated the product they had developed with members such as the technical domain expert, test lead and business representatives, from the Norwegian team that owned that specific product. The demo facilitated coordination because it provided an arena for creating common expectations and understanding the finished product.

Backlog Grooming with Gamma

The purpose of the Backlog grooming was to ensure that Gamma had a thorough understanding of the new business requirements. The meeting therefore consisted of a presentation of the business requirements provided by a business representative, and a run through of technical recommendations given by the technical domain expert, both from the Norwegian team that was responsible for that specific area. The meeting ensured coordination between the teams because that was the arena where Gamma was provided with in-depth knowledge about the requirements.

Sprint Planning with Gamma

At the Sprint Planning meeting team Gamma selected the tasks they were going to develop in the upcoming sprint. In that way, the meeting ensured the focus of the next sprint. These decisions changed rarely, close to never, during the sprint. The sprint planning meeting ensured coordination between the teams because that was where the tasks were allocated amongst the members in team Gamma, based on the priorities made by the Norwegian team that was responsible for that specific area.

5.3.2 Unscheduled meetings

Informal ad hoc conversations

This coordination mechanism is described in full under the Chapter 5.2.2 because the conversations take place both between two people, and more than two people.

6 Discussion

In this Chapter, the results outlined in Chapter 5, and the related limitations, are discussed in light of the theory and related research in order to answer the thesis' research questions:

RQ1: *What are the challenges and possible solutions in the coordination between business and development in agile software development?*
RQ2: *How does task uncertainty, task interdependence and size of work unit change when business and development merge into one team?*

Since this thesis is limited to the coordination mechanisms in a team consisting of business and development, the answer to the research questions will be within those topics.

Before answering the proposed research questions, I will discuss whether the unit of analysis is a team or a working group, because it has implications on how to analyze the results. After having collected the different sources of data on the unit of analysis, it is evident that even though this group of people calls themselves a team, the presented data contradicts that.

When viewing this case in relation to the theory on working group versus team proposed by Katzenbach and Smith (2005, p. 4), summarized in Table 1 in Chapter 2.4, this unit of analysis matches the characteristics of a working group. The group had a strong and focused leader, individual accountability, individual work products, efficient meetings, measured their effectiveness in terms of financial performance and delegated their work. Therefore, this unit of analysis does not meet the characteristics of a team (Katzenbach & Smith, 1993).

Realizing that the unit of analysis is a working group instead of a team is helpful when assessing the value of the different coordination mechanisms. Since the group worked on different applications, the value of conducting the daily stand-up meeting was limited, because the information that one person had about his or her application was not relevant for the others.

6.1 Challenges and possible solutions in the coordination of BizDev teams

This Section discusses and answers the first research question;

RQ1: *What are the challenges and possible solutions in the coordination between business and development in agile software development?*

The differing needs in the work environment was a great challenge. Business representatives wanted an open and social work environment that would facilitate discussion, while developers wanted a closed work environment that would facilitate fewer interruptions when they write code.

A possible solution to the challenge of different needs in the work environment is a physical solution. Since the open work area was highly valued by both business and development, it is important not to change that. However, the developers could be seated at the innermost part of their open work area so they are shielded from the noise, while the business representatives could be seated at the outermost part of the area. In that way, the temporary seats for guests could be located close to the business representatives. That could be one way of avoiding to interrupt the developers, as research by Parnin and Rugaber (2009) and van Solingen, Berghout, and van Latum (1998) state that developers typically need more than 15 minutes or more to resume programming after an interruption.

Another observed challenge is that the regularly scheduled team meetings are arenas where only the business side of the team, and the project manager, team lead and tech lead, are present. Meaning that the developers do not attend these team meetings. Since coordination takes place at these arenas, it proves difficult to coordinate themselves when they are not both present. I would also claim that this meeting created a bigger divide in the team rather than unifying the team, as one would assume is intended for a team meeting. As Moe, Dingsøyr, and Røyrvik (2009) propose five dimensions to be addressed when improving teamwork; shared leadership, team orientation, redundancy, learning and autonomy. Of the five, Stray, Moe, and Dingsøyr (2011) focus on, amongst others, team orientation and stresses that a team needs information sharing, shared goals, and to work together to achieve their goals. I would therefore argue that regular scheduled team meetings for the entire team is where the information sharing and facilitate working together takes place. Therefore, I consider the team to have low team orientation.

An obvious solution to this challenge is to establish a team meeting for the entire team. However, it is a difficult balancing act to participate in the right number of meetings, because the number of meetings affect employees well-being (Luong & Rogelberg, 2005). On one hand meetings are valuable and the source of communication, on the other hand too many meetings are seen as a waste of time (Rogelberg, Scott, & Kello, 2007). Additionally, there has been conducted several studies which states that a meeting is an interruption and that it takes time for people to get back to their tasks after a meeting (Allen et al., 2012; Parnin & Rugaber, 2009; Zijlstra, Roe, Leonora, & Krediet, 1999). Therefore, I would argue that the already existing team meetings should be reconsidered, and that perhaps the new proposed team meeting could replace the existing ones.

In addition to establishing a shared meeting, the team should exploit the potential the story mapping meetings has, to increase their team orientation. The story mapping meeting was an arena where business and development together broke down the system that was to be developed, and used the Lean Startup mindset in order to create a minimum viable product (MVP). In this meeting both units worked together and coordinated the work amongst themselves. Since this meeting was perceived as very valuable and lead to effective coordination, I believe that the lack of coordination mentioned above could be gathered at story mapping. Moreover, story mapping's main activity is finding the MVP, which is one way of conducting experiments with the customers (Ries, 2011). Experimentation is proposed by Bosch (2016) as being what organizations need to transition to, in order for them to be able to deliver on today's software needs. Additionally, by following the Lean Startup mindset and practices, the team is strengthening their customer relationships by shortening their feedback loops and contributing to fast communication and coordination, which is one of the recommendations, proposed by Olsson and Bosch (2016). Therefore, I would argue that the coordination mechanism story mapping is a very important meeting and highly valuable for the team.

Besides the divided team meetings, the daily stand-up meeting was neither a meeting for all the team members. At the regularly scheduled stand-up meetings were the developers and the digitally responsible present, but not the business representatives. This regularly scheduled meeting contributed very seldom to coordination between business and development. However, it was regarded by many as an arena for reporting status. A study of daily stand-up meetings by Stray et al. (2016) also showed that the meeting too often was regarded as a status event. Moreover, another study shows that developers in large teams view the meeting as less valuable than developers in small teams, which may be because the work in large teams is often loosely coupled (Stray et al., 2017). I would argue that the work in this case indeed was loosely coupled since the developers worked on different systems.

A possible solution to the challenges with the daily stand-up meeting is to focus on only the two last questions Scrum questions, as proposed by (Stray et al., 2016), where they suggest to discontinue the first question because it addresses what the team members have done, and in large parts leads to status reporting. Moreover, the team does not necessarily need to conduct the meeting on a daily basis, but still regularly (Stray et al., 2016).

By not having the daily stand-up meeting on a daily basis, the team would have more time for their informal ad hoc conversation, in which was regarded as the most important and most effective coordination mechanism, as outlined in the findings. A study by Kraut and Streeter (1995) also confirms that the informal communication is needed for coordination, because it is especially useful in the face of uncertainty, which is how software is characterized. Additionally, the frequent conversations contribute to effective coordination (Strode, Huff, Hope, & Link, 2012). Moreover, the informal ad hoc conversations create speed, as Bosch (2016) proposes as one of the key elements to success. Furthermore, it leads to fast decision-making. Since the regularly scheduled meetings are dominated by status reporting, these conversations are needed in order for the decision-making to take place. However, since the ability to create informal ad hoc conversations are depending on the person, it is important to be aware that these types of conversations might not always work as successful as in this case.

The interviewees also stated that the informal ad hoc conversations were the most frequently used coordination mechanism. The large amount of time spent in such conversations is also reported in a study by Herbsleb and Mockus (2003), which states that the developers spent an average of 75 minutes a day in unplanned interactions. However, in cases where the unplanned interactions, or informal ad hoc conversations, does not lead to fast decision-making, I would assume that spending such a large amount of time on these interactions would not be preferred.

Not only the scheduled meetings, but also the impersonal coordination mechanisms were different between business and development. Within the instant messaging tool, the developers used HipChat, while the business representatives did not. I would argue that it created a barrier between the two groups because they did not receive the same information. Salas, Sims, and Burke (2005) express that the exchange of information is invaluable in teamwork. Therefore, I regard this as a great challenge.

In hopes of solving the challenge, I propose to introduce a new instant messaging tool that neither business, nor development used before, such as Slack[11]. Since the conducted interviews revealed that the business representatives were hesitant to join HipChat, I believe changing to a new arena, which is not very technical, is a good middle ground. Moreover, changing to an entire new tool will give none of them the impression that they have to change their way of working, while the others do not.

Furthermore, business and development used different methodologies. Business used the customer journey map and Lean Startup, while development used the agile process of Scrumban. The customer journey map and Lean Startup (Ries, 2011) address more the specification part of development, meaning all the activities performed before the backlog, such as how to understand the customer's needs and what part of the finished system is needed in a minimum viable product. The agile process of Scrumban addresses the activities pertaining to development, meaning the actual coding of the system. Since business and development follow different methodologies that pertain to separate parts of development, they work in different ways.

Even though it is a challenge that members of the same team work differently, it is also inevitable when the team is cross-functional. I believe the team should continue working with their own methodologies, but still learn and understand how the other team members are working in order to bridge the gap.

[11] Slack is a registered trademark of Slack Technologies, www.slack.com.

Lastly, as mentioned in the introduction to this discussion, there are several reasons why this team in fact is a working group. Some of those reasons are also challenges in this case. The high degree of specialization means that the team members does not actually work together, but on different systems or different parts of the same system Stray et al. (2011). Multiple studies observed similarly that the developers had little competence to solve tasks outside their specific area, which lead to low team redundancy (Levesque, Wilson, & Wholey, 2001; Moe, Dingsøyr, & Dybå, 2010; Stray et al., 2011, p. 159). In accordance with Stray et al. (2011), I would recommend them to identify new ways of increasing the redundancy.

The absence of a common goal lead to limited team spirit and less effective work. This is not an uncommon challenge, as Stray et al. (2011) also reported it in their study of agile teams. Moreover, they also propose that improving teamwork by increasing team orientation can be achieved by creating shared goals (Stray et al., 2011). It could therefore be possible to establish team orientation by allocating enough time, resources and facilitation, as suggested by Stray et al. (2011, p. 158).

6.2 The change in task uncertainty, task interdependence, and size of work unit

This Section discusses and answers the second research question;
RQ2: *How does task uncertainty, task interdependence and size of work unit change when business and development merge into one team?*

First, I will discuss what change the merge between business and development lead to, in terms of coordination, meaning whether the different coordination mechanisms increased or decreased in usage. Then, I will discuss how the merge between business and development lead to a change in task uncertainty, task interdependence and size of work unit, and what that change was. Finally, I will compare my results to the ones of Van de Ven et al. (1976) and other relevant findings.

6.2.1 Change in the coordination mechanisms used

Table 13: The coordination mechanisms before and after the merge

When	Impersonal	Personal		Group	
		Vertical	Horizontal	Scheduled	Unscheduled
Before the merge	Jira	One-on-one		Daily Stand-Up	
	Wiki			Retrospective	
	Instant Messaging			Department Meetings	
	Communication Tool			Team Meetings	
	Customer Journey Map			Demo for Stakeholders	
	Deployment Rules			Demo Gamma	
	Quality Assurance Rules			Backlog Grooming Gamma	
	User Support			Sprint Planning	
	Agile Process				
Added after the merge	Monitor Tool		Informal ad hoc conversations	Story Mapping	Informal ad hoc conversations
	Measure Tool				
	Lean Startup				
	Open Work Area				

Figure 19 contains a visualization of the change in the use of the different coordination mechanisms in this study.

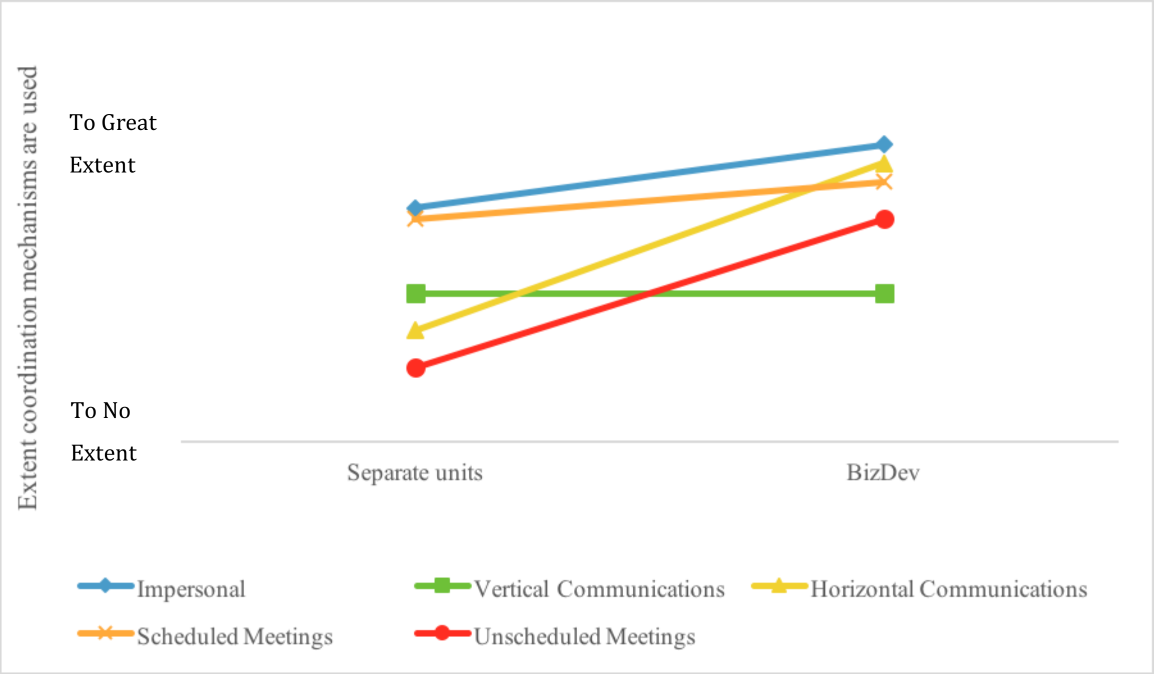


Figure 19: The change in coordination mechanisms over time in this study

Impersonal coordination mechanisms

As shown in Table 13, the impersonal coordination mechanisms increased with four mechanisms, from nine to thirteen, when business and development merged into one BizDev team. The mechanisms New Relic (monitor application performance tool), Google Analytics (measure application performance tool), the methodology Lean Startup and the open work area were the mechanisms that they adopted when they merged business and development into one team.

The methodology Lean Startup (Ries, 2011) was adopted when they merged because it provided a technique for both business and development to work more effectively together. Blank (2013) stated that the Lean Startup practices are meant for large organizations as well, but that they require new organizational structures, being what had been carried out in this case. The concept such as the minimum viable product (MVP) is an activity that both business and development carried out together in order for both to better understand what they were developing, and what features were necessary.

The "build-measure-learn feedback loop" (Ries, 2011) is another example of a mindset that Lean Startup contributed to the BizDev team. New Relic and Google Analytics were tools that were implemented in order to provide data to the 'measure' part of the feedback loop, and were therefore also implemented after business and development merged.

Business and development physically moved in order to be seated together, and therefore occurred the coordination mechanism named open work area. The open work area facilitated informal ad hoc conversations.

In total, there was a rather substantial increase in the impersonal coordination mechanisms. The use of the impersonal mechanisms was rather great for each of the units before they merged, therefore the use of these coordination mechanisms was relatively high both before and after the merge.

Personal coordination mechanisms

There was not established any additional vertical communications coordination mechanism when business and development merged into BizDev. I would propose that was because the team members' managers' role did not change, and the way in which they coordinated with their managers remained the same. However, within horizontal communications, because the BizDev team was co-located, the open work area facilitated the informal ad hoc conversations, which made it a much more frequently used coordination mechanism.

In total, the extent to which the vertical communications were used remained unchanged, while the horizontal communications increased substantially because the informal ad hoc conversations were used continuously, while it barely existed when the two units were separated.

Group coordination mechanisms

Within the scheduled meetings the meeting called story mapping was introduced after business and development merged, partly due to the introduction of Lean Startup. Story mapping is the activity that is performed when a large system is divided into smaller MVPs.

In addition to being a personal coordination mechanism, the informal ad hoc conversations also took place between three or more people; therefore, the coordination mechanism is regarded as an unscheduled meeting within the group coordination mechanisms.

In total, the use of scheduled meetings increased to a certain extent, because of the introduction of story mapping. Meanwhile the unscheduled meeting increased significantly because the informal ad hoc conversations between three or more took place very often, and barely existed before.

6.2.2 Comparison between my study, Van de Ven et al. (1976) and others

Van de Ven et al. (1976) looked at task uncertainty, task interdependence, and size of work unit in 197 work units by distributing a questionnaire to each unit supervisor and unit members, and completing a standardized follow-up interview with each unit supervisor.

Task uncertainty

I propose that the degree of uncertainty increased after business and development merged into one team. Before, they worked in a way where business specified all of the features before it was handed over to development. That way of working implied that they knew what they were creating. However, after they merged into one team, they accepted that they did not know the solution to what they are producing. They introduced the methodology Lean Startup, which is meant for units that are creating systems under conditions of extreme uncertainty (Ries, 2011). That is an indication that they acknowledged that they do not necessarily know what is the best system for their customers.

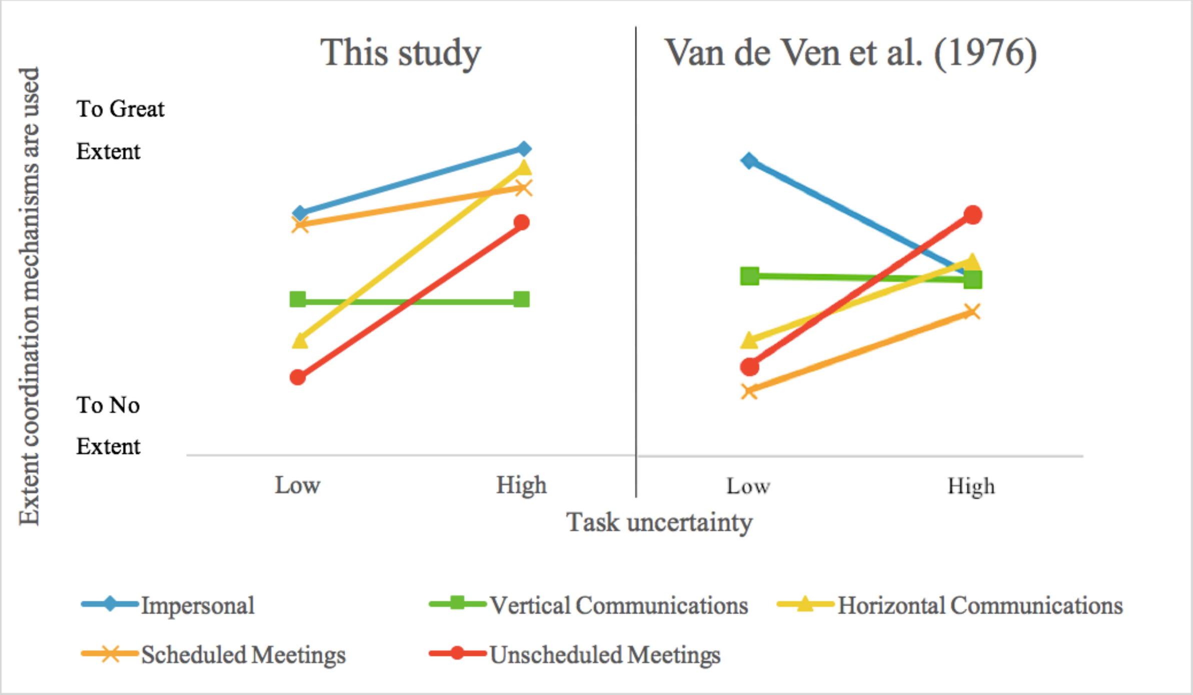


Figure 20: Coordination mechanisms and task uncertainty, in my study and Van de Ven et al. (1976)

Similarly, Van de Ven et al. (1976) also looked at how the use of the different coordination mechanisms changed when there was an increase in task uncertainty, therefore my findings are compared to theirs in Figure 20.

The similarity between the two studies, that the vertical communications remained unchanged, indicates that the use of a manager in order to coordinate the work does not increase even though the uncertainty increased because the two units merged.

Additionally, the similarities that there was an increase in the use of scheduled meetings, unscheduled meetings and horizontal communications in both studies are also supported by multiple studies that claim that mutual adjustments are performed when tasks become more variable and less analyzable (Hoegl, Weinkauf, & Gemuenden, 2004; Kraut & Streeter, 1995; March & Simon, 1958; Perrow, 1970; Thompson, 1967). Moreover, a study performed on IT projects, by Nidumolu (1995), investigated how coordination mechanisms and project uncertainty affected project performance, showed that coordination by mutual adjustments and communication between team members had a positive impact on the project performance. Meaning that communication, as a way of exchanging knowledge, and adjustments for the changes is needed.

However, the use of impersonal coordination mechanisms increased in my case study, while they decreased in the result of Van de Ven et al. (1976). Some claim that solving complex tasks require 'richer media' than plans and schedules, as examples of impersonal coordination (Daft & Lengel, 1986). I would argue that the substantial difference in the use of impersonal coordination mechanisms is in large parts due to the present technological tools. Since tools such as New Relic and Google Analytics are rather new inventions, they aid in analyzing today's customers and markets.

Task interdependence

Before business and development merged into a BizDev team, the work flow between the two units can be described as reciprocal, because the output of the business unit became the input of development, such as when they created the specifications of a system and handed it over to development. Additionally, the work flow was cyclical because the process of finalizing the specifications was completed in iterations.

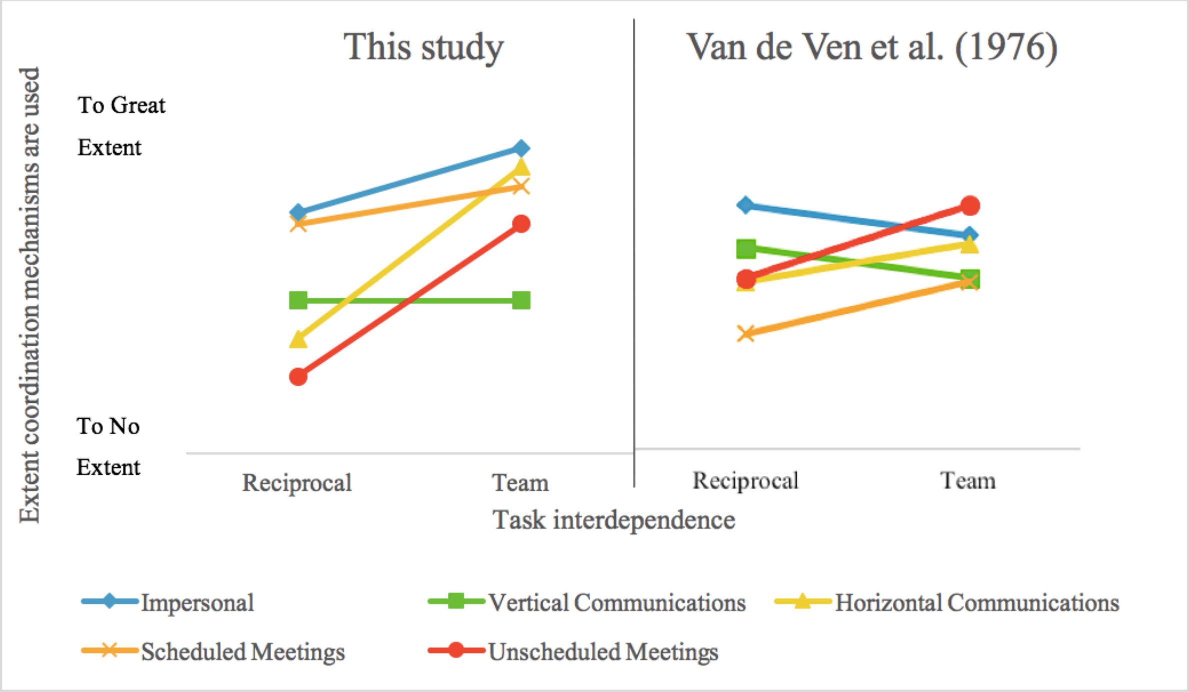


Figure 21: Coordination mechanisms and task interdependence, in my study and Van de Ven et al. (1976)

Similarly, Van de Ven et al. (1976) also looked at how the use of the different coordination mechanisms changed when there was an increase in task interdependence, therefore my findings are compared to theirs in Figure 21.

The use of vertical communications remained unchanged in my study, while it decreased in the results of Van de Ven et al. (1976). I would argue that a change in task interdependence would not affect the need for coordination with a manager, because it would rather affect the team members.

In both cases there was an increase in the use of scheduled meetings, unscheduled meetings and horizontal communications, which is in large parts supported by Thompson (1967). Moreover, Dietrich, Kujala, and Artto (2013) state that personal coordination is emphasized when the interdependency is high. Furthermore, a study by Andres and Zmud (2001) show that coordination strategies such as informal and cooperative are more successful when the interdependence is high between organizational units. I propose that the higher task interdependence calls for more mutual adjustments because the team members have to talk to each other because they are dependent on each other's work.

However, the use of impersonal coordination mechanisms increased in my study, while it decreased in the results of Van de Ven et al. (1976). Similarly, Dietrich et al. (2013) found that high interdependencies meant a reduction in the impersonal coordination mechanisms. I would argue that the use of impersonal coordination mechanisms increased because the team members had an increased need to coordinate themselves with each other, which they did by the use of for example, the instant messaging tool, the Lean Startup methodology, and the open work area.

Work Unit Size

There has been an increase in the size of work unit since business and development merged into one unit. Business and development was seven and five people, respectively, and merged into a BizDev team of twelve. Van de Ven et al. (1976) characterized groups containing up to ten people for a small group, therefore the BizDev team would be called a large group.

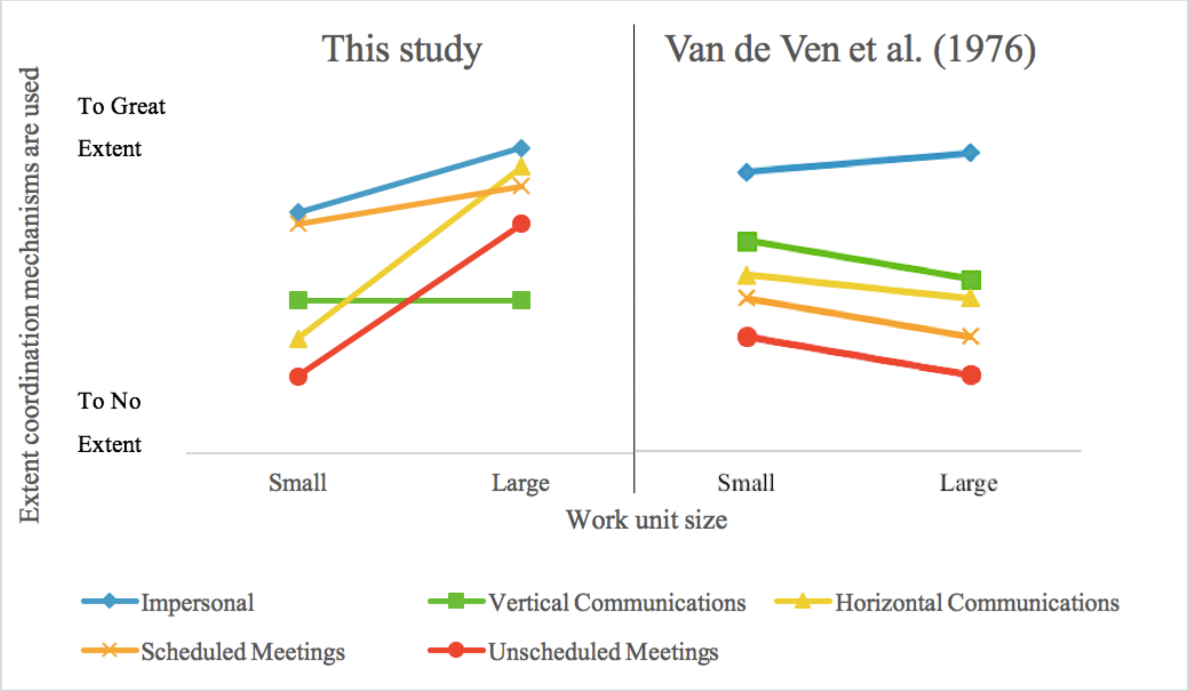


Figure 22: Coordination mechanisms and work unit size, in my study and Van de Ven et al. (1976)

Similarly, Van de Ven et al. (1976) also looked at how the use of the different coordination mechanisms changed when there was an increase in work unit size, therefore my findings are compared to theirs in Figure 22.

Both results show an increase in the use of impersonal coordination mechanisms, which Van de Ven et al. (1976) call an impersonalizing effect. Several studies confirm this, and state that formalization increase when the work unit size increase (Blau, 1970; Hickson, Pugh, & Pheysey, 1969; Kraut & Streeter, 1995).

The use of vertical communications remained unchanged in my study, while it decreased in the results of Van de Ven et al. (1976). The reliance on vertical communication also decrease in other studies (Blau, 1970; Hickson et al., 1969). However, I would argue that when business and development merged into one team, the coordination with their managers remained unchanged, while in other cases it decreased because one manager would not have the capacity to have the one-on-one contact with a large number of team members.

However, the use of both unscheduled meetings, scheduled meetings and horizontal communications increased in my study while they decreased in the results of Van de Ven et al. (1976). I would argue that they increased in my case because new meeting arenas were established, while in other cases I assume that the number of team members would be too large for it to be profitable to conduct such meetings.

6.3 Limitations

In this part of the thesis, I discuss possible limitations to the study.

Theory by Van de Ven et al. (1976)

The results presented by Van de Ven et al. (1976) are collected in 197 work units, while my results are based on one single study. Moreover, the results by Van de Ven et al. (1976) are based on completed questionnaires and one interview in each unit, while my study is based on extensive observations and several interviews. Therefore, it is important to keep in mind the difference in scale and type when comparing my study to Van de Ven et al. (1976).

Other frameworks

There are other frameworks that could have been chosen, such as the theory proposed by Mintzberg (1979). If a different framework was followed, the results and discussion could be different, because the approach would be different.

Coordination over time

Since "coordination mechanisms are dynamic social practices that are under continuous construction” (Jarzabkowski et al., 2012) it is evident that the results could be different if the data was collected at a different point in time. It would therefore also be interesting to observe the team over a longer period of time, both before business and development merged, and after they became a BizDev team. In particular, I would want to collect more data on how business and development were coordinating themselves before they merged.

Developers’ point of view

This study examined BizDev from the developers’ point of view. However, I tried my best to be neutral and address the situation from both sides. I believe that my educational background, also having a Bachelor’s degree in Commerce with a major in Global Business Management, helped me in understand the situation from the business’ side as well.

Construct validity

As mentioned in Chapter 3.2 on data collection I followed the first principle of Yin (2009) of using multiple sources of evidence by using three sources of evidence, which indicated that I used the strategy of data triangulation.

Furthermore, the presentation that I held for the unit of analysis, where they had the opportunity to correct me and add relevant information, reduced the threat to construct validity. This presentation contributed to a more accurate understanding of the case and a more correct interpretation of the data, and therefore also reduced the threat to construct validity.

External validity

Since the use of a theory increases the external validity, I used the theory of coordination proposed by Van de Ven et al. (1976) both during the data collection and during the data analysis.

Reliability

I created a case study database and maintained a chain of evidence, as outlined in Chapter 3.2, in order to increase the reliability of the study.

Moreover, this study has been reviewed by peer researchers, as the strategy of observer triangulation. It was upheld by discussing the case and the data with three researchers that are involved in flexible design research. This strategy increases the likelihood that someone else will reach the same conclusions as me when presented with the same case.

7 Conclusion and Future work

This Chapter is presenting this thesis' conclusion and proposes suggestions for future work.

7.1 Conclusion

The aim of this master's thesis was two-fold. The first research question aimed at mapping out the challenges and possible solutions in coordination between business and development. The three main challenges discovered are that they had differing needs in their working environment, they were not present at the same meetings nor used the same methodologies or tools, and their characteristics were of a working group, not a team. Overall, it is difficult to coordinate themselves when they are not both present in the same arenas where the coordination takes place. The possible solutions to the three main challenges are: to reorganize where the team members are seated in the open work area, to focus more on impediments during the daily stand-up meeting, to create a team meeting that includes all members, to continue and enhance the use of story mapping, and facilitate for informal ad hoc conversations. Overall, these changes would lead to less status reporting, a collected team, a common understanding of the product they are creating, and speed and fast decision-making.

The second research question looked at how task uncertainty, task interdependence, and size of work unit change when business and development merge into a team. In this study, I found that they all increased, but due to different reasons. Task uncertainty increased because the team acknowledged that they do not necessarily know what they are creating for their customers by introducing the methodology Lean Startup. Moreover, the task interdependence increased because business and development got more dependent on each other's work as they continuously worked together, in coordination mechanisms such as story mapping and informal ad hoc conversations. Lastly, the size of work unit increased because they merged into one large team of twelve team members.

7.2 Future work

There are several topics that could be studied further.

BizDev is a new term that would benefit from more investigation. What would be the results different if the BizDev team was in fact a team, and not a working group?

The challenge that the units have different needs in their working environment was raised in this study. The developers wanted a quiet zone for focusing on programming and designated seats, while business wanted a very open work area for discussions and temporary seats for moving around. It seems that the developers need to be shielded, while business need to get in touch with people. How is it possible to get cross-functional teams with business and development to work well together when their way of working is so different?

Bibliography

Ahmad, M. O., Markkula, J., & Oivo, M. (2013). *Kanban in software development: A systematic literature review.* Paper presented at the Software Engineering and Advanced Applications (SEAA), 2013 39th EUROMICRO Conference on.

Allen, J. A., Sands, S. J., Mueller, S. L., Frear, K. A., Mudd, M., & Rogelberg, S. G. (2012). Employees' feelings about more meetings: An overt analysis and recommendations for improving meetings. *Management Research Review, 35*(5), 405-418.

Andres, H. P., & Zmud, R. W. (2001). A Contingency Approach to Software Project Coordination. *Journal of Management Information Systems*, 41-70.

Beck, K. (2000). *Extreme programming explained: embrace change*: addison-wesley professional.

Blank, S. (2013). Why the lean start-up changes everything. *Harvard Business Review, 91*(5), 63-72.

Blau, P. M. (1970). Decentralization in bureaucracies. *Power in organizations*, 150-174.

Bogsnes, B. (2016). *Implementing beyond budgeting: unlocking the performance potential*: John Wiley & Sons.

Bosch, J. (2016). Speed, data, and ecosystems: the future of software engineering. *IEEE Software, 33*(1), 82-88.

Burn, J. M., & Szeto, C. (2000). A comparison of the views of business and IT management on success factors for strategic alignment. *Information & management, 37*(4), 197-216.

Daft, R. L., & Lengel, R. H. (1986). Organizational information requirements, media richness and structural design. *Management science, 32*(5), 554-571.

Debois, P. (2011). Devops: A software revolution in the making. *Journal of Information Technology Management, 24*(8), 3-39.

Dietrich, P., Kujala, J., & Artto, K. (2013). Inter‐Team Coordination Patterns and Outcomes in Multi‐Team Projects. *Project Management Journal, 44*(6), 6-19.

Dingsøyr, T., & Lassenius, C. (2016). Emerging themes in agile software development: Introduction to the special section on continuous value delivery. *Information and Software Technology, 77*, 56-60.

Dingsøyr, T., Moe, N. B., Fægri, T. E., & Seim, E. A. (2016). Exploring software development at the very large-scale: a revelatory case study and research agenda for agile method adaptation. *Empirical Software Engineering*, 1-31.

Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *The Journal of Systems & Software, 6*(85), 1213-1221.

Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology, 50*(9), 833-859.

Fitzgerald, B., & Stol, K.-J. (2017). Continuous software engineering: A roadmap and agenda. *Journal of Systems and Software, 123*, 176-189.

Glaser, B. G. (2011). *Getting out of the data: Grounded theory conceptualization*: Sociology press.

Harwell, M. R. (2011). ReseaRch Design in Qualitative/Quantitative. *The Sage handbook for research in education: Pursuing ideas as the keystone of exemplary inquiry*, 147.

Herbsleb, J. D., & Mockus, A. (2003). An empirical study of speed and communication in globally distributed software development. *IEEE transactions on Software Engineering, 29*(6), 481-494.

Hiatt, J. F. (1986). Spirituality, medicine, and healing. *Southern medical journal, 79*(6), 736-743.

Hickson, D. J., Pugh, D. S., & Pheysey, D. C. (1969). Operations technology and organization structure: An empirical reappraisal. *Administrative science quarterly*, 378-397.

Highsmith, J. A. (2002). *Agile software development ecosystems* (Vol. 13): Addison-Wesley Professional.

Hoegl, M. (2005). Smaller teams–better teamwork: How to keep project teams small. *Business Horizons, 48*(3), 209-214.

Hoegl, M., Weinkauf, K., & Gemuenden, H. G. (2004). Interteam coordination, project commitment, and teamwork in multiteam R&D projects: A longitudinal study. *Organization Science, 15*(1), 38-55.

Humble, J., & Molesky, J. (2011). Why enterprises must adopt devops to enable continuous delivery. *Cutter IT Journal, 24*(8), 6.

Jarzabkowski, P. A., Lê, J. K., & Feldman, M. S. (2012). Toward a theory of coordinating: Creating coordinating mechanisms in practice. *Organization Science, 23*(4), 907-927.

Karvonen, T., Suomalainen, T., Juntunen, M., Sauvola, T., Kuvaja, P., & Oivo, M. (2016). *The CRUSOE Framework: A Holistic Approach to Analysing Prerequisites for Continuous Software Engineering.* Paper presented at the PROFES.

Katzenbach, J. R., & Smith, D. K. (1993). *The discipline of teams*: Harvard Business Press.

Katzenbach, J. R., & Smith, D. K. (2005). The discipline of teams. *Harvard Business Review, 83*(7), 162.

Kaur, R., & Sengupta, J. (2012). Software Process Models and Analysis on Failure of Software Development Projects. *International Journal of Scientific & Engineering Research, 2*(2).

Kraut, R. E., & Streeter, L. A. (1995). Coordination in software development. *Communications of the ACM, 38*(3), 69-82.

Ladas, C. (2009). *Scrumban-essays on kanban systems for lean software development*: Lulu. com.

Larman, C., & Basili, V. R. (2003). Iterative and incremental developments. a brief history. *Computer, 36*(6), 47-56.

Leffingwell, D. (2007). *Scaling software agility: best practices for large enterprises*: Pearson Education.

Levesque, L. L., Wilson, J. M., & Wholey, D. R. (2001). Cognitive divergence and shared mental models in software development project teams. *Journal of Organizational Behavior, 22*(2), 135-144.

Liker, J. K. (2004). *The toyota way*: Esensi.

Lincoln, Y. S., & Guba, E. G. (1985). *Naturalistic inquiry* (Vol. 75): Sage.

Luong, A., & Rogelberg, S. G. (2005). Meetings and more meetings: The relationship between meeting load and the daily well-being of employees. *Group Dynamics: Theory, Research, and Practice, 9*(1), 58-67.

Malone, T. W. (1988). What is coordination theory?

Malone, T. W., & Crowston, K. (1994). The interdisciplinary study of coordination. *ACM Computing Surveys (CSUR), 26*(1), 87-119.

March, J. G., & Simon, H. A. (1958). Organizations.

Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook*: sage.

Mintzberg, H. (1979). *The structuring of organizations* (Vol. 12): Prentice-Hall Englewood Cliffs, NJ.

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, 52*(5), 480-491.

Moe, N. B., Dingsøyr, T., & Røyrvik, E. A. (2009). Putting agile teamwork to the test–an preliminary instrument for empirically assessing and improving agile software development. *Agile processes in software engineering and extreme programming*, 114-123.

Nidumolu, S. (1995). The effect of coordination and uncertainty on software project performance: residual performance risk as an intervening variable. *Information Systems Research, 6*(3), 191-219.

Olsson, H. H., & Bosch, J. (2016). No More Bosses? In P. Abrahamsson, A. Jedlitschka, A. Nguyen Duc, M. Felderer, S. Amasaki, & T. Mikkonen (Eds.), *Product-Focused Software Process Improvement: 17th International Conference, PROFES 2016, Trondheim, Norway, November 22-24, 2016, Proceedings* (pp. 86-101). Cham: Springer International Publishing.

Overby, E., Bharadwaj, A., & Sambamurthy, V. (2005). A framework for enterprise agility and the enabling role of digital options. *Business agility and information technology diffusion*, 295-312.

Parker, G. M. (2003). *Cross-functional teams: Working with allies, enemies, and other strangers*: John Wiley & Sons.

Parnin, C., & Rugaber, S. (2009). *Resumption strategies for interrupted programming tasks.* Paper presented at the Program Comprehension, 2009. ICPC'09. IEEE 17th International Conference on.

Patton, M. Q. (2002). *Qualitative Research & Evaluation Methods*: SAGE Publications.

Paulk, M. C., Curtis, B., Chrissis, M. B., & Weber, C. V. (1993). Capability maturity model, version 1.1. *IEEE Software, 10*(4), 18-27.

Perrow, C. B. C. B. (1970). *Organizational analysis: A sociological view*. Retrieved from

Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit: An Agile Toolkit*: Addison-Wesley.

Putnam, L. H. (1978). A general empirical solution to the macro software sizing and estimating problem. *IEEE transactions on Software Engineering*(4), 345-361.

Reddy, A. (2015). *The Scrumban [r] evolution: Getting the Most Out of Agile, Scrum, and Lean Kanban*: Addison-Wesley Professional.

Ries, E. (2011). *The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*: Crown Books.

Robson, C. (2002). Real world research. 2nd. *Edition. Blackwell Publishing. Malden*.

Rodriguez, D., Sicilia, M., García, E., & Harrison, R. (2012). Empirical findings on team size and productivity in software development. *Journal of Systems and Software, 85*(3), 562-570.

Rogelberg, S. G., Scott, C., & Kello, J. (2007). The science and fiction of meetings. *MIT Sloan Management Review, 48*(2), 18.

Salas, E., Sims, D. E., & Burke, C. S. (2005). Is there a "big five" in teamwork? *Small group research, 36*(5), 555-599.

Saldana, J. (2009). *The Coding Manual for Qualitative Researchers*: SAGE Publications.

Schwaber, K., & Beedle, M. (2002). *Agile software development with Scrum* (Vol. 1): Prentice Hall Upper Saddle River.

Spradley, J. P. (1980). *Participant Observation*: Holt, Rinehart and Winston.

Stray, V., Moe, N. B., & Bergersen, G. R. (2017). *Are Daily Stand-up Meetings Valuable? A Survey of Developers in Software Teams.* Paper presented at the International Conference on Agile Software Development.

Stray, V., Sjøberg, D. I., & Dybå, T. (2016). The daily stand-up meeting: A grounded theory study. *Journal of Systems and Software, 114*, 101-124.

Stray, V. G., Moe, N. B., & Dingsøyr, T. (2011). *Challenges to teamwork: a multiple case study of two agile teams.* Paper presented at the International Conference on Agile Software Development.

Strode, D. E., Huff, S. L., Hope, B., & Link, S. (2012). Coordination in co-located agile software development projects. *Journal of Systems and Software, 85*(6), 1222-1238.

Thompson, J. D. (1967). *Organizations in action: Social science bases of administrative theory*: Transaction publishers.

Van de Ven, A. H., Delbecq, A. L., & Koenig Jr, R. (1976). Determinants of coordination modes within organizations. *American sociological review*, 322-338.

van Solingen, R., Berghout, E., & van Latum, F. (1998). Interrupts: just a minute never is. *IEEE Software, 15*(5), 97-103.

VersionOne. (2017). VersionOne 11th Annual State of Agile Report. Retrieved from http://stateofagile.versionone.com/

Vinekar, V., Slinkman, C. W., & Nerur, S. (2006). Can agile and traditional systems development approaches coexist? An ambidextrous view. *Information systems management, 23*(3), 31-42.

Willcoxson, L., & Chatham, R. (2004). Progress in the IT/business relationship: a longitudinal assessment. *Journal of Information Technology, 19*(1), 71-80.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). *Experimentation in software engineering*: Springer Science & Business Media.

Wohlin, C., Šmite, D., & Moe, N. B. (2015). A general theory of software engineering: Balancing human, social and organizational capitals. *Journal of Systems and Software, 109*, 229-242. doi:http://dx.doi.org/10.1016/j.jss.2015.08.009

Yin, R. K. (2009). Case study research: Design and Methods. SAGE publications. *Thousand oaks*.

Zijlstra, F. R., Roe, R. A., Leonora, A. B., & Krediet, I. (1999). Temporal factors in mental work: Effects of interrupted activities. *Journal of Occupational and Organizational Psychology, 72*(2), 163-185.

Appendix

Attachment A: Interview Guide, Manager

Part	Question
Introduction	Present myself Thank the person for participating Confirm confidentiality and anonymity Ask for permission to record the interview Inform about sharing the document when transcribed
General	What in general are you working on now? What role do you have? Including tasks and responsibilities. What is the status of that project?
Teamwork	
Coordination	Who do you talk to in order to solve your tasks? What do you normally clarify?
Coordination mechanisms	Which of the following mechanisms do you use during your workday? • Personal coordination one on one, with your leader or other team members. • Group coordination with more than two, in scheduled or unscheduled meetings. Which of the following mechanisms do you use or affects the way you work? • Pre-established plans, rules, guidelines, standardized information and communication systems.
Prioritizations	How are tasks prioritized? What stakeholders are important to the team? How are the team's tasks communicated to the team? How do you document the prioritizations of tasks to the team? Do you use any tools?
Closing	What do you think has changed in the past year? Why? What caused this? What has happened? What challenges have you experienced with the new team composition? Is there anything you would like to add that we did not discuss?

Attachment B: Interview Guide, Business representatives

Part	Question
Introduction	Present myself Thank the person for participating Confirm confidentiality and anonymity Ask for permission to record the interview Inform about sharing the document when transcribed
General	How long have you been working for this company? How long have you been working in this team? What in general are you working on now? What role do you have? Including tasks and responsibilities. What is the status of that project?
Teamwork	Do you have an overview of what the other team members are doing? Do you have a common goal? Do you collaborate with other team members?
Coordination	Who do you talk to in order to solve your tasks? What do you normally talk about? What do you spend a lot of time on? And not? When the coordination is with business, how is it? Per mail, in meeting, in person, etc.?
Coordination mechanisms	Which of the following mechanisms do you use during your workday? • Personal coordination one on one, with your leader or other team members. • Group coordination with more than two, in scheduled or unscheduled meetings. Which of the following mechanisms do you use or affects the way you work? • Pre-established plans, rules, guidelines, standardized information and communication systems.
Stand-up	If participating in Stand-up: On a scale of one to five (where one is not satisfied, and five is very satisfied), how satisfied are you with Stand-up? Why? What value do you experience from attending? When you do not participate in the meeting, what is the reason? If not participating in Stand-up: Why do you not participate in Stand-up? What do you think Stand-up consists of?
Closing	What do you think has changed in the past year? Why? What caused this? What has happened? What challenges have you experienced with the new team composition? Is there anything you would like to add that we did not discuss?

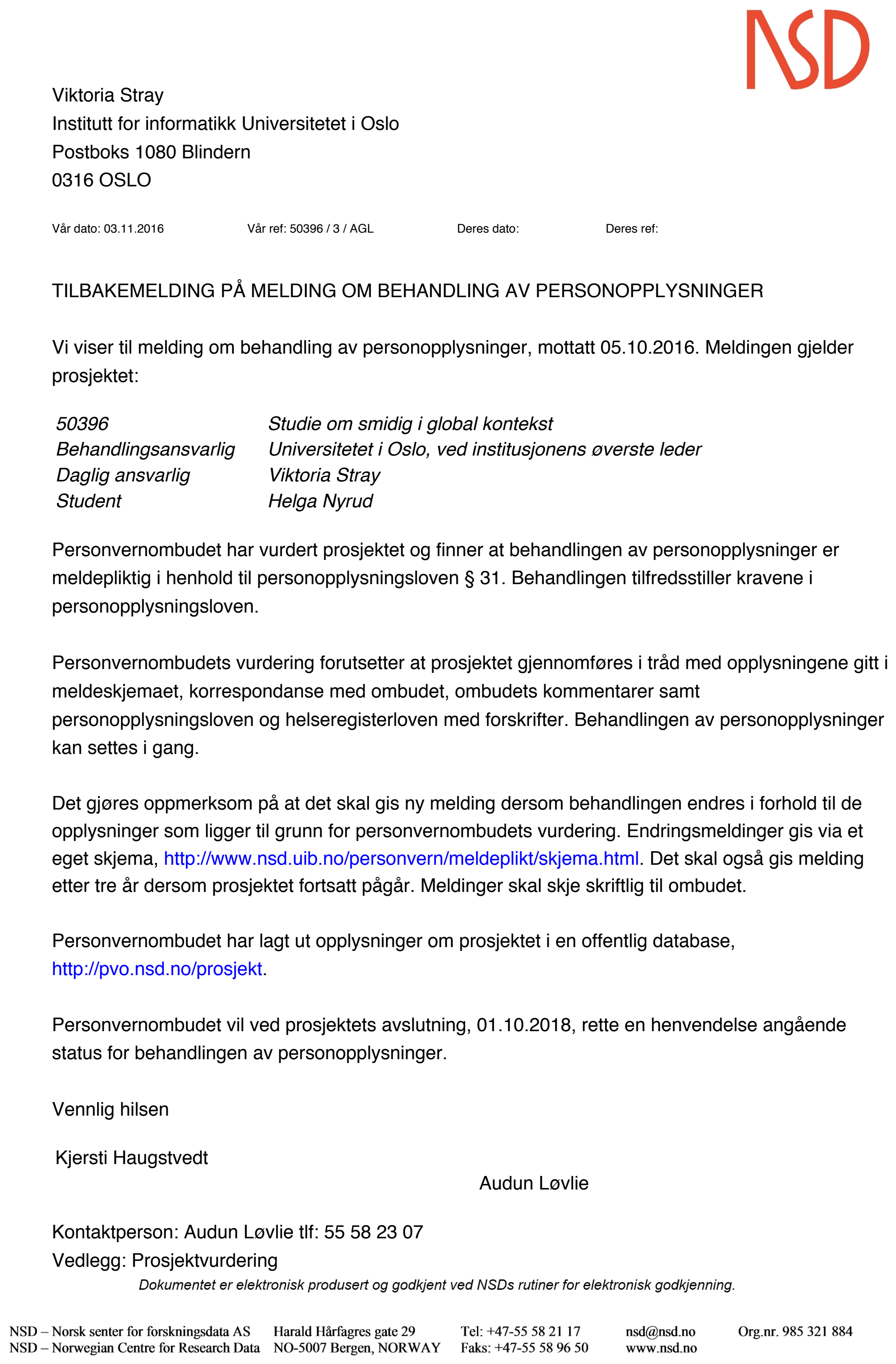
Attachment C: Interview Guide, Developers

Part	Question
Introduction	Present myself Thank the person for participating Confirm confidentiality and anonymity Ask for permission to record the interview Inform about sharing the document when transcribed
General	How long have you been working for this company? How long have you been working in this team? What in general are you working on now? What role do you have? Including tasks and responsibilities. What is the status of that project?
Teamwork	Do you have an overview of what the other team members are doing? Do you have a common goal? Do you collaborate with other team members?
Coordination	Who do you talk to in order to solve your tasks? What do you normally talk about? What do you spend a lot of time on? And not? How much time do you spend in meetings, and coding? When the coordination is with business, how is it? Per mail, in meeting, in person, etc.?
Coordination mechanisms	Which of the following mechanisms do you use during your workday? • Personal coordination one on one, with your leader or other team members. • Group coordination with more than two, in scheduled or unscheduled meetings. Which of the following mechanisms do you use or affects the way you work? • Pre-established plans, rules, guidelines, standardized information and communication systems.
Stand-up	On a scale of one to five (where one is not satisfied, and five is very satisfied), how satisfied are you with Stand-up? Why? What value do you experience from attending? When you do not participate in the meeting, what is the reason?
Closing	What do you think has changed in the past year? Why? What caused this? What has happened? What challenges have you experienced with the new team composition? Is there anything you would like to add that we did not discuss?

Attachment D: Observation protocol

Topic	Question
Space	What is the layout of the physical space? Where are the different actors situated?
Actors	Who are the different actors involved? What are the actors' names and roles?
Activities	What are the different activities and discussions?
Objects	What physical objects are used?
Acts	How do the different actors interact and behave towards each other? Are there any specific individual actions?
Events	Are there anything unexpected
Time	When does the activity start? What is the sequence of events? When does the activity end?
Goals	What are the different actors' goals?
Feelings	How is the atmosphere in the particular contexts? What are the emotions?

Attachment E: NSD Approval



Viktoria Stray
Institutt for informatikk Universitetet i Oslo
Postboks 1080 Blindern
0316 OSLO

Vår dato: 03.11.2016 Vår ref: 50396 / 3 / AGL Deres dato: Deres ref:

TILBAKEMELDING PÅ MELDING OM BEHANDLING AV PERSONOPPLYSNINGER

Vi viser til melding om behandling av personopplysninger, mottatt 05.10.2016. Meldingen gjelder prosjektet:

50396	*Studie om smidig i global kontekst*
Behandlingsansvarlig	*Universitetet i Oslo, ved institusjonens øverste leder*
Daglig ansvarlig	*Viktoria Stray*
Student	*Helga Nyrud*

Personvernombudet har vurdert prosjektet og finner at behandlingen av personopplysninger er meldepliktig i henhold til personopplysningsloven § 31. Behandlingen tilfredsstiller kravene i personopplysningsloven.

Personvernombudets vurdering forutsetter at prosjektet gjennomføres i tråd med opplysningene gitt i meldeskjemaet, korrespondanse med ombudet, ombudets kommentarer samt personopplysningsloven og helseregisterloven med forskrifter. Behandlingen av personopplysninger kan settes i gang.

Det gjøres oppmerksom på at det skal gis ny melding dersom behandlingen endres i forhold til de opplysninger som ligger til grunn for personvernombudets vurdering. Endringsmeldinger gis via et eget skjema, http://www.nsd.uib.no/personvern/meldeplikt/skjema.html. Det skal også gis melding etter tre år dersom prosjektet fortsatt pågår. Meldinger skal skje skriftlig til ombudet.

Personvernombudet har lagt ut opplysninger om prosjektet i en offentlig database, http://pvo.nsd.no/prosjekt.

Personvernombudet vil ved prosjektets avslutning, 01.10.2018, rette en henvendelse angående status for behandlingen av personopplysninger.

Vennlig hilsen

Kjersti Haugstvedt

Audun Løvlie

Kontaktperson: Audun Løvlie tlf: 55 58 23 07
Vedlegg: Prosjektvurdering

Dokumentet er elektronisk produsert og godkjent ved NSDs rutiner for elektronisk godkjenning.

NSD – Norsk senter for forskningsdata AS Harald Hårfagres gate 29 Tel: +47-55 58 21 17 nsd@nsd.no Org.nr. 985 321 884
NSD – Norwegian Centre for Research Data NO-5007 Bergen, NORWAY Faks: +47-55 58 96 50 www.nsd.no

Attachment F: Article in Press

Inter-Team Coordination Mechanisms in Large-Scale Agile

Helga Nyrud
University of Oslo
Gaustadalléen 23B
0373 Oslo, Norway
helgany@ifi.uio.no

Viktoria Stray
University of Oslo
Gaustadalléen 23B
0373 Oslo, Norway
stray@ifi.uio.no

ABSTRACT

Coordination is an important but challenging success factor in large-scale software development. Large-scale is particularly demanding because it involves several teams, and therefore we chose to study coordination between teams. Our aim of this case study was to investigate inter-team coordination mechanisms. By using an established framework from the field of sociology, we found eleven different coordination mechanisms that we mapped into the five different categories. We interviewed four project members and observed 26 meetings in three different teams as part of a larger digitization project. Our results show that even though there are many impersonal mechanisms in the form of various guidelines and rules, the most important mechanism seem to be ad-hoc conversations between two or more project members. The large-scale project had many regularly scheduled meetings. However, the majority of these meetings tended to be, to a greater extent, about reporting status and not about coordinating work. The scheduled meeting that was the most time-consuming, and also involved little coordination, was the daily stand-up meeting with two teams. Future work should investigate how to increase the value of daily stand-up meetings in large-scale programs, and whether the meetings can be organized in a way that enables more inter-team coordination.

Keywords

large-scale software development, coordination practices, large project organization, daily stand-up meeting, communication, software engineering, multiteam systems

1. INTRODUCTION

An important topic in large-scale agile is inter-team coordination [1][2]. In large-scale settings, achieving effective coordination is difficult because of the complexity different sites and teams and their dependencies introduce, especially when dependencies between activities are tightly coupled [3] [4]. Dingsøyr et al. classifies large-scale agile as development projects of two to nine teams where coordination is achieved through a new forum where the teams participate [5].

Agile software development has become the mainstream development methodology [6]. According to a recent survey [7], more than 70% of developers work in agile software projects and almost 40% work in distributed teams. Agile methodology introduced a shift in how teams coordinate their work. For example, most agile teams conduct daily stand-up meetings [8]. Representatives from more than one team often attend daily stand-up meetings to achieve inter-team coordination in large-scale agile and the meetings are often called Scrum-of-Scrums [9].

With large projects, new organizational forms such as "team-of-teams" are being used, referred to as multiteam-systems (MTS). An MTS is defined as a minimum of two teams that work interdependently and are linked by shared super-ordinate goals [10]. Scheerer et al. [11] studied coordination strategies in large-scale agile with an MTS perspective and call for more exploratory research on coordination in large-scale software development.

In this paper, we have conducted a case study that draws on a well-known framework of coordination mechanisms proposed by Van de Ven et al. [12].

We investigate the following research question: *What coordination mechanisms facilitate large-scale agile development?*

2. METHOD

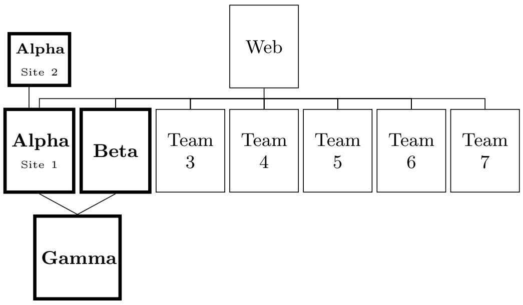


Figure 1: Organizational chart

XP '17 Workshops, May 22-26, 2017, Cologne, Germany
© 2017 ACM. ISBN 978-1-4503-5264-2/17/05...$15.00
DOI: https://doi.org/10.1145/3120459.3120476

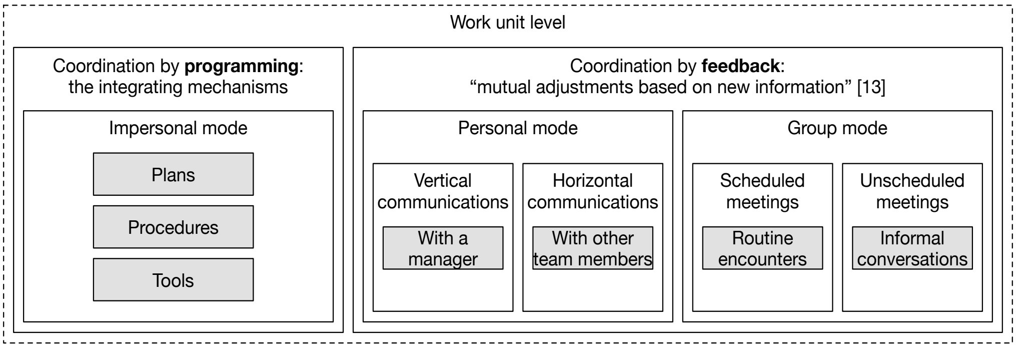


Figure 2: Coordination modes with examples in grey, based on Van de Ven et al. [12]

The study was conducted in a digitization project in a Norwegian pension, bank and insurance company with more than 2,000 employees. The project had eight teams that worked on different solutions, and we studied three of these teams, marked in bold in Figure 1. The three research subjects were two Norwegian teams, Alpha and Beta, and one offshore, outsourced Indian team, Gamma. Additionally, the Alpha team also had an extended team offshore consisting of two developers, named Site 2 in Figure 1.

Team Alpha developed the company's services for the retail marked, while team Beta developed services for the company's business customers. Both team Alpha and Beta consisted of three developers, one team lead, one tech lead and one test lead. Six of team Gamma's members were allocated to work on tasks from team Alpha, while three individuals worked on solutions handed to them by team Beta. Gamma was typically handed tasks that required less domain knowledge and understanding of Norwegian law and language. Even though Gamma was an outsourced team with most of their team members offshore, they still had two members on-site, seated close to both Alpha and Beta. All three teams followed agile methodology. The three teams conducted daily stand-up meetings, and we observed 20 of these as shown in Table 1. We also observed six other meetings, such as workshops and team meetings, as well as the project members working in their natural habitat. Additionally, we conducted four interviews, with representatives from team Alpha and Beta. The interviews lasted for approximately one hour and were audio recorded and transcribed. Both notes from observations and the interview transcripts were imported into a tool for qualitative analysis and were coded on the topic of inter-team coordination.

There has been done several studies on coordination mechanisms; hence there are also multiple frameworks on the subject. Thompson [13] proposed the theory that coordination mechanisms consist of three parts: standardization/rules, plans/schedules, and mutual adjustment. The framework proposed by Van de Ven et al. [12] expanded on the theory by Thompson [13] by adding a fourth part which included the group aspect of coordination, namely the team. This is represented in the rightmost part of Figure 2, and called Group mode. Hence, we chose this framework, because it takes into account the team coordination mechanisms as well.

Table 1: Data collection

Data	*Alpha*	*Beta*	*Gamma*
Daily Stand-Up Meeting	9	5	6
Other meetings	2	2	2
Interviews	2	2	0

Furthermore, Mintzberg [14] has a similar theory. His theory states that coordination mechanisms are divided into five: mutual adjustment, direct supervision, and the standardization of work processes, of work outputs, and of work skills. While Mintzberg's [14] framework is on an organizational level, Van de Ven et al's [12] framework is on the work unit level of an organization, which makes it more suitable for this study of the inter-team coordination mechanisms in the digitization project.

3. RESULTS

We have mapped coordination mechanisms in the large-scale project according to a framework from classical organisational theory [12], see Table 2. The framework [12] identifies three main coordination categories: impersonal, personal and group mode, depicted in Figure 2. The impersonal mode represents all the forms of coordination by *programming*, meaning the integrating mechanisms such as formalized rules, pre-established plans, standardized information and communication systems [12]. The personal and group mode are both types of coordination by *feedback* and is defined by Thompson [13] as "mutual adjustments based upon new information." All of the present coordination mechanisms are displayed in Figure 3 according to when they happened during the two-week iteration, in order to present when the different mechanisms took place.

3.1 Programming

The project used five different coordination mechanisms of type programming and impersonal mode, as shown in Table 2.

All three teams followed an agile process. The offshore team Gamma followed the Scrum methodology and conducted two-week iterations (called sprints). The main pur-

Table 2: Inter-team coordination mechanisms

Coordination Mechanisms	**Programming**	**Feedback**			
	Impersonal mode	**Personal mode**		**Group mode**	
		Vertical	Horizontal	Scheduled	Unscheduled
Agile process	X				
Jira	X				
Rules for QA	X				
Open work area	X				
Instant messaging	X				
Informal ad hoc conversations			X		X
Stand-Up Meeting				X	
Retrospective				X	
Demo				X	
Sprint planning				X	
Backlog grooming				X	

pose why the off-site team worked with Scrum's time boxes were to secure predictability as to when their solutions and products were delivered to the Norwegian teams for testing. Thus the use of time boxing enabled coordination because it simplified planning the different coordinating mechanisms. However, the Alpha and Beta teams followed a Scrumban methodology where they task boxed their assignments.

The project management tool Jira enabled coordination between different teams. Jira was a crucial part of team Gammas daily stand-up meetings because their meetings were organized so that the on-site Team Lead ran through the tasks that were in the "In Progress" column. Furthermore, they had columns in Jira for Quality Assurance (QA) for the off-site team and for the Norwegian team that owned the technical solution; team Alpha or Beta. Additionally, they had one last column called Acceptance Testing. Moreover, Jira also enabled coordination between the teams by providing the option to comment and give feedback to the different tasks.

The on-site teams Alpha and Beta did QA on the product delivered by the Gamma team. In order for the teams to do QA they had to follow a set of guidelines to ensure that the quality of the delivered product was of a certain level.

Open Work Area was practiced in all three teams. The on-site members of team Gamma were seated in the open work area in the close vicinity to both team Alpha and Beta. The open area enabled faster coordination across the teams because team members got hold of each other instantly.

The teams used various communication tools to instantly message each other. The tools facilitated coordination on a frequent basis and provided an important platform for communication and collaboration between the teams.

3.2 Feedback

The coordination mechanisms of type feedback is divided into *personal* and *group* mode. In the *personal mode* the individual serves as the mechanism and this mode is further divided into vertical (communication with a manager) and horizontal (communication with other team members). *Group mode* involves a group of individuals, and is divided into scheduled and unscheduled meetings. This division is mainly in order to differentiate between the more routine encounters and the informal conversations between co-workers [12].

Informal ad hoc conversations, between two or more people, typically happened by the team members' seats. This coordination mechanism was the one that happened most frequently and was regarded as very valuable by the team members. The mechanism also facilitated quick coordination and decision-making because the conversations took place instantly. This is illustrated by the following statement by a business developer:

> "I think to make the communication more informal and have less regularly scheduled meetings is perhaps the way to go."

A developer also commented that informal ad hoc conversations were preferred:

> "I simply just walk over and talk, it's more effective than writing a Jira issue with the overhead it entails. As a consequence I have very few e-mails, very few meetings, and that's great.".

Every morning Gamma conducted a daily stand-up meeting with representatives from Alpha and Beta, therefore this meeting was a form of Scrum-of-Scrums. The first 15 minutes of the meeting was with the Alpha team, the subsequent 15 minutes was with the Beta team. The meeting was quite formal and rigid. They were organized so that the on-site team lead ran through each task in Jira so that the responsible developer could give status on the task. The meeting lead to little coordination, but was more of a status update, as one developer commented:

> "I think there's too little discussions regarding obstacles, it's more like giving a status report."

Furthermore, both Alpha and Beta had their own stand-up meetings with their respective team members, these meetings were more informal. Because Alpha was distributed with two developers off-site their stand-up meetings were conducted using video. However, the meetings in Alpha were as valuable as those in Beta even though the meetings in Alpha were distributed, where the use of video was a positive contributor. Table 3 shows an overview of what roles participated in the different daily stand-up meetings. Some roles had to attend several daily stand-up meetings every day. For example, the Tech Liaison who attended three daily stand-up meetings, as well as the whole Gamma team which was present at two daily stand-up meetings. The

Table 3: Roles participating in the different team's daily stand-up meetings.

Roles	Teams			
	A	B	A & G	B & G
Team Alpha:				
Project Manager	X			
Team Lead	X			
Tech Liaison	X		X	X
Developers	X			
Team Beta:				
Project Manager		X		
Team Lead		X		
Test Manager		X		X
Developers		X		
Team Gamma:				
Project Manager			X	X
Team Lead			X	X
Developers			X	X

purpose of the Tech Liaison role was to create and maintain consistency across teams and across the technical platform, in which the role was considered valuable and important in a large-scale perspective.

The goal of the Retrospective was to improve the way the teams worked together. The meeting therefore included a review of what they did well, and what they could improve. All members from Gamma were present, as well as members from the relevant Norwegian team. This meeting ensured a continuous improvement of the inter-team coordination mechanisms.

At the Demo meeting, Gamma demonstrated the product they had developed with members such as the technical domain expert, test lead and business representatives, from the Norwegian team that owned that specific product. The demo facilitated coordination because it provided an arena for creating common expectations and understanding the finished product.

At the Sprint Planning meeting team Gamma selected the tasks they were going to develop in the upcoming sprint. In that way the meeting ensured the focus of the next sprint. This decision changed rarely, close to never, during the sprint. The sprint planning meeting ensured coordination between the teams because that was where the tasks were allocated amongst the members in team Gamma, based on the priorities made by the Norwegian team that was responsible for that specific area.

The purpose of the Backlog grooming was to ensure that Gamma had a thorough understanding of the new business requirements. The meeting therefore consisted of a presentation of the business requirements provided by a business representative, and a run through of technical recommendations given by the technical domain expert, both from the Norwegian team that was responsible for that specific area. The meeting ensured coordination between the teams because that was where Gamma was provided with in-depth information about the requirements.

4. DISCUSSION AND CONCLUSION

In this section we discuss our research question: What coordination mechanisms facilitate large-scale agile development?

In a study of 12 co-located development teams, Dingsøyr et al. [2] reported 14 inter-team coordination mechanism. We found that 11 inter-team coordination mechanisms was present in the project. In large-scale settings, one would expect that meetings provide the best opportunity for coordination. In our study, informal ad hoc conversations were more important and also more frequently used. Our findings are consonant with those of Dingsøyr et al. [2], who found that informal communication was the most important mechanism. Informal communication was supported by the open work area (ibid). Even though informal ad hoc conversations in our study facilitated coordination between the teams in an efficient way and led to quick problem-solving, a negative aspect is that it was dependent on each individual. In this program, the ad hoc conversations were necessary because the scheduled meetings were more about reporting status than coordinating work. Individuals in large-scale projects will not have necessary knowledge regarding the whole system, therefore they need to rely on their network for solving complex tasks [15]. However, some people are outgoing and know many people and will naturally contact other teams while others are more prone to working alone. For example, new team members do not necessarily know who to ask and will benefit more from scheduled meetings. Another benefit of using scheduled meetings as a coordination mechanism is that everyone is involved and receive the same information.

There was little coordination of tasks in the daily stand-up meetings. While we would expect that the daily stand-up meetings with two teams (”scrum of scrum-meetings”) would have more coordination than the local stand-up meetings, the opposite was the case. However, there was little coordination in both types of meetings. Our findings support an earlier study of daily stand-up meetings which found that coordination constituted only 7% of the meeting [16]. Additionally, if they had organized the stand-up by letting people speak in turns instead of focusing on the tasks in Jira, the meeting might have led to more coordination because it would have had less focus on reporting status on each of the tasks.

The framework of Van de Ven et al. [12] was valuable as a tool to map coordination mechanisms in a large-scale program. In particular, it was valuable in terms of mapping impersonal modes of coordination such as rules, plans and communication systems (programming), because the framework makes you aware of what to look for and to understand the concept of coordination. These coordination mechanisms are usually harder to uncover because the researcher has to specifically ask for them. While meetings are visible, programming type coordination mechanisms are written rules and guidelines that are followed. It is more difficult to know that a team member works a certain way because of these guidelines, they may have turned into norms and routines several months earlier.

Norms are essential in influencing how agile team members work [17]. For example, the norm in the daily stand-up meetings was that the team lead asked for status for each of the tasks in the sprint, which resulted in little coordination in the meetings. Agile teams should incorporate norms and routines that facilitate coordination in all the different arenas. Most of the coordination in the program happened in backlog grooming and sprint planning meetings. In these meetings, tasks were assigned to each of the team members and rarely changed during the sprint. Coordination of de-

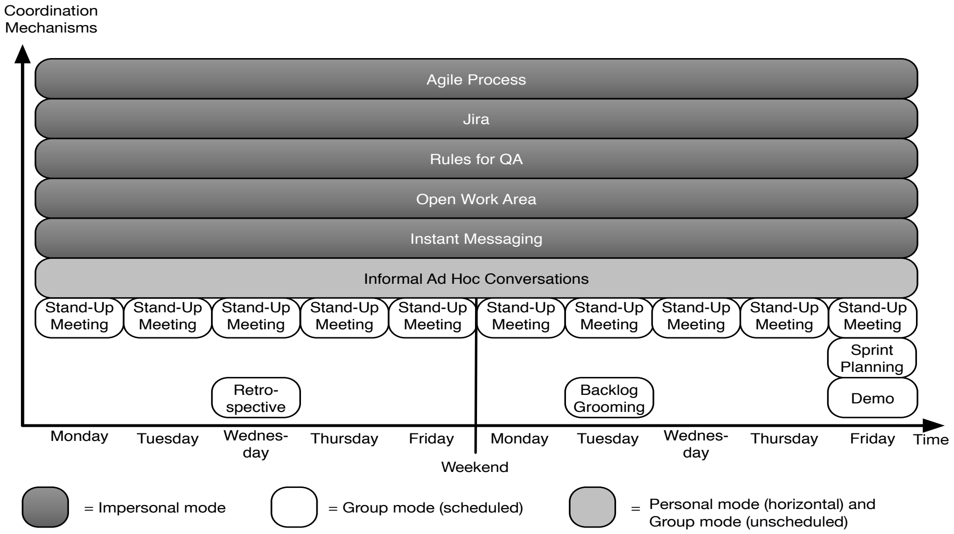


Figure 3: The coordination mechanisms when they were present during a sprint.

pendencies between tasks mostly happened through informal ad hoc conversations and by instant-messaging.

In large scale agile development, many practitioners create new roles, such as "Chief Product Owner" (CPO) in order to have someone responsible for the end product and the inter-team coordination [11]. In our study they created a role called Tech Liaison. The individual having this role possessed the technical insights into the entire product portfolio and served as a link between the different teams. The role was therefore particularly valuable for coordinating teams and facilitating the large-scale development.

In addition, many practitioners also conduct daily stand-up meetings/Scrum of scrums with participants from more than one team. However, in this case, those meetings did not lead to effective coordination. The structure of the meeting was a rigorous run-through of the current Jira tasks, which lead to a reporting of status instead of actual coordination. This style of reporting status on each Jira task gave rather the impression that the project manager on-site wanted control over the work performed off-site, rather than coordinating themselves. However, the use of video was perceived as contributing positively to the meeting.

4.0.1 Acknowledgements

We thank Gunnar R. Bergersen for his comments on an earlier version of this paper. This work was supported by the Smiglo project, which is partly funded by the Research Council of Norway under the grant 235359/O30.

5. REFERENCES

[1] Nils Brede Moe, Helena Holmström Olsson, and Torgeir Dingsøyr. Trends in Large-Scale Agile Development. In *the Scientific Workshop XP2016*, New York, New York, USA, 2016. ACM Press.

[2] Torgeir Dingsøyr, Nils Brede Moe, Tor Erlend Fægri, and Eva Amdahl Seim. Exploring software development at the very large-scale: a revelatory case study and research agenda for agile method adaptation. *Empirical Software Engineering*, pages 1–31, 2017.

[3] Grady Booch. The future of software engineering: keynote. In *Proceedings of the 37th International Conference on Software Engineering-Volume 2*, pages 3–3. IEEE Press, 2015.

[4] James D. Herbsleb and Audris Mockus. An empirical study of speed and communication in globally distributed software development. *IEEE Transactions on software engineering*, 29(6):481–494, 2003.

[5] Torgeir Dingsøyr, Tor Erlend Fægri, and Juha Itkonen. What is large in large-scale? a taxonomy of scale for agile software development. In *International Conference on Product-Focused Software Process Improvement*, pages 273–276. Springer, 2014.

[6] Torgeir Dingsøyr, Sridhar Nerur, VenuGopal Balijepally, and Nils Brede Moe. A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6):1213 – 1221, 2012.

[7] Viktoria Stray, Nils Brede Moe, and Gunnar R. Bergersen. Are daily stand-up meetings valuable? A survey of developers in software teams. In Hubert Baumeister, Horst Lichter, and Matthias Riebisch, editors, *Agile Processes in Software Engineering and*

Extreme Programming: 18th International Conference, XP 2017, Cologne, Germany, May 22-26, 2017, Proceedings, pages 274–281, Cham, 2017. Springer International Publishing.

[8] Viktoria Stray, Dag I. K. Sjøberg, and Tore Dybå. The daily stand-up meeting: A grounded theory study. *Journal of Systems and Software*, 85:101 – 124, 2016.

[9] Maria Paasivaara, Lassenius Casper, and Ville T. Heikkilä. Inter-team coordination in large-scale globally distributed scrum: Do scrum-of-scrums really work? In *Proceedings of the 2012 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ACM*, pages 235–238. ACM, 2012.

[10] JE Mathieu, Michelle A Marks, and Stephen J Zaccaro. Multi-team systems. *International handbook of work and organizational psychology*, 2:289–313, 2001.

[11] Alexander Scheerer, Tobias Hildenbrand, and Thomas Kude. Coordination in large-scale agile software development: A multiteam systems perspective. In *System Sciences (HICSS), 2014 47th Hawaii International Conference on*, pages 4780–4788. IEEE, 2014.

[12] Andrew H Van de Ven, Andre L Delbecq, and Richard Koenig Jr. Determinants of coordination modes within organizations. *American sociological review*, pages 322–338, 1976.

[13] James D Thompson. *Organizations in action: Social science bases of administrative theory.* Transaction publishers, 1967.

[14] Henry Mintzberg. Structure in 5's: A synthesis of the research on organization design. *Management science*, 26(3):322–341, 1980.

[15] Darja Šmite, Nils Brede Moe, Aivars Šāblis, and Claes Wohlin. Software teams and their knowledge networks in large-scale software development. *Information and Software Technology*, 86:71 – 86, 2017.

[16] Viktoria Stray, Nils Brede Moe, and Aurum Aybüke. Investigating daily team meetings in agile software projects. In *Software Engineering and Advanced Applications (SEAA), 38th EUROMICRO Conference on.*, pages 274–281. IEEE, 2012.

[17] Viktoria Stray, Tor Erlend Fægri, and Nils Brede Moe. Exploring norms in agile software teams. In *Proceedings of the International Conference on Product-Focused Software Process Improvement*, pages 458–467. Springer International Publishing, 2016.