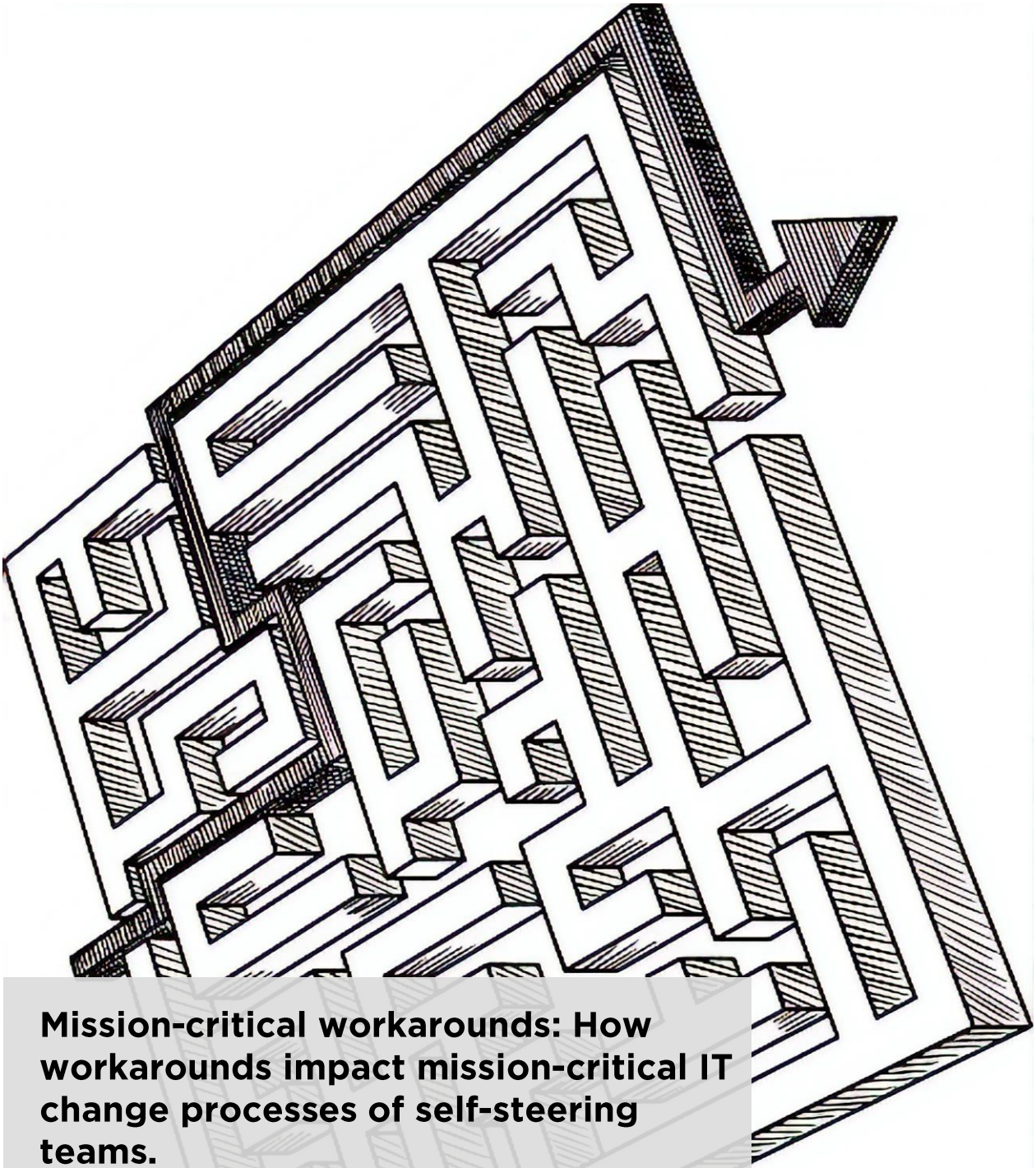


Mission-critical workarounds: How workarounds impact mission-critical IT change processes of self-steering teams.

Master thesis – Business Informatics

Nick A. Gies
n.a.gies@students.uu.nl | ngies@schubergphilis.com

Department of Information and Computing Sciences
Utrecht University, Utrecht, The Netherlands

Mission-critical workarounds: How workarounds impact mission-critical IT change processes of self-steering teams

Master thesis – Business Informatics

N. (Nick) Gies BSc
Graduating student
Student number: 6273408

n.a.gies@students.uu.nl
ngies@schubergphilis.com

Utrecht University
Department of Information and Computing Sciences
Buys Ballot Building
Princetonplein 5, de Uithof
3584 CC, Utrecht

dr. G.C. (Inge) van de Weerd
Assistant professor
First supervisor

g.c.vandeweerd@uu.nl

Utrecht University
Department of Information and Computing Sciences
Buys Ballot Building
Princetonplein 5, de Uithof
3584 CC, Utrecht

I. (Iris) Beerepoot MSc
PhD Candidate
Second supervisor

i.beerepoot@uu.nl

Utrecht University
Department of Information and Computing Sciences
Buys Ballot Building
Princetonplein 5, de Uithof
3584 CC, Utrecht

dr. H.W. (Henk) van der Schuur
Customer Director
First external supervisor

hvanderschuur@schubergphilis.com

Schuberg Philis
Boeingavenue 271
1119 PD, Schiphol-Rijk

Ir. I. (Ilja) Heitlager
Chief Innovation Officer
Second external supervisor

iheitlager@schubergphilis.com

Schuberg Philis
Boeingavenue 271
1119 PD, Schiphol-Rijk

Abstract

While the research field is divided on whether workarounds are a positive or negative aberration to an initial process, it is clear that analysis of workarounds shows to be valuable input for process optimisation. Workaround analysis helps to understand human interaction with processes and thereby gain knowledge about flaws and shine light on potential improvements.

While workarounds are a broadly studied concept, a lot of this research has been executed in a healthcare context. This study focusses on mission-critical IT as within mission-critical IT, time is often of the essence, and there is limited room for failure, which makes it similar to healthcare in its characteristics. Because of this pressure in time and quality, change management is relatively regulated and is expected to have a significant impact on failure.

There seems to be a compelling case to study workarounds in the mission-critical change process that are executed by self-steering teams. Within self-steering teams' authority is decentralised and decision-making responsibilities are transitioned from traditional management to individual team members. It is interesting to analyse what workarounds occur, why they occur, what their impact is, what actions are taken in response, and how this relates to the self-steering aspect of teams.

Given this context, there is value in discovering what kind of workarounds exist within these mission-critical IT change processes and determine how they impact the process. Knowledge about this impact and how to react to it will help mission-critical IT service providers be more in control of their processes and increase the quality and efficiency of their change management processes. Hence the following research question is asked: *How do workarounds of self-steering teams impact mission-critical change processes?*

To answer this question, a systematic literature review, document analysis and two cases studies are conducted. The systematic literature review describes what authority and decisions making are and how self-steering teams manage these concepts. The document analysis provides insight into mission-critical processes and sets a baseline to detect workarounds. Two cases studies were conducted within Schuberg Philis for two customer teams that operate in a finance industry and postal service industry. Within these case studies, the workaround snapshot approach was used, the approach identifies workarounds and helps to determine the motivation, impact and actions taken in response to those workarounds. Lastly, the drivers behind the occurrence of workarounds executed by self-steering teams in mission-critical changes processes were investigated to identify connections between those concepts.

Results show that in most cases, mission-critical workarounds relate to the administrative part of the process or the approval mechanisms. The majority of workarounds have a positive impact on time, costs and flexibility of the process. The impact on quality differs as both a positive, neutral and negative impact was determined. Even though the impact in terms of time, costs, quality, and flexibility show both positive and negative results, teams decided to ignore only one workaround. In most cases, workarounds are motivated by an act in the best interest of the customer, in other cases team member forget to execute activities, but they rarely consciously avoid execution of activities. Whenever workarounds are motivated by an act in the best interest of the customer, they tend to be adopted or lead to a redesign of the process. For these specific workarounds, results show a positive or neutral impact on the quality of the process.

Preface

Here I am at the end of a journey as I hereby lay the last stone to the fundament of my future.
I look back with gratitude but forward full of excitement.

I want to start by thanking both Henk van der Schuur and Ilja Heitlager, my supervisors at Schuberg Philis. You inspired me to excel, and your guidance kept me sane in moments of uncertainty, I can't thank you enough for this.

I also want to thank my direct colleagues at Schuberg Philis: Robbert C, Robbert K, Pieter, Ozge, Gerard, Yorick, Denis, Andrii, Giel, and many more. The moments where I could express my thoughts and validate my decisions helped to keep me on track. Thank you for this, and I look forward to working with you in the near future.

Next, I want to thank Inge van de Weerd and Iris Beerepoot, my supervisors from Utrecht University. The support that that you gave throughout the project was motivating and your feedback always proved to be valuable and helped to lift this research to a higher level.

Finally, I want to thank all the customer teams within Schuberg Philis that were involved in my research. Without your excitement and willingness to help, it would've never been possible for me to achieve this. Thank you.

I sincerely hope that you take just as much joy in reading this thesis as I had in writing it.

Nick Gies
February 2020

Table of contents

List of figures

List of tables

"The improvement of understanding is of two ends; first, our own increase of knowledge; secondly, to enable us to deliver that knowledge to others."

— John Locke

Chapter 1

Introduction

This research thesis gives insight into the topic of process workarounds by self-steering teams and their impact on mission-critical change processes. Its cause is subject to the graduation of Nick Gies for finalising his master's degree in Business Informatics at the Department of Information and Computer Sciences, Utrecht University. The research is conducted and executed over a period of eight months running from June 2019 to February 2020 under supervision of dr. G.C. (Inge) van de Weerd and I. (Iris) Beerepoot MSc and facilitated by Schuberg Philis; dr. H.W. (Henk) van der Schuur and Ir. I. (Ilja) Heitlager. This research explores the field of process workarounds and determines how workarounds impact the mission-critical IT change processes of self-steering teams.

1.1 Problem statement & objectives

Organisations continuously try to optimise their operations by rethinking their processes. One of the most well-known concepts derived from this is business process management (BPM), which is defined as "a body of methods, techniques, and tools to identify, discover, analyse, redesign, execute, and monitor business processes in order to optimise their performance"(Dumas, La Rosa, Mendling, & Reijers, 2018). Although BPM centres the process itself, it often fails to take deviations of the designed process into account. Methods for this exist, such as conformance checking, but these are limited as they require event logs and an IT system to be present and do not consider the human interaction with the process.

In fact, in many studies, these deviations or so-called workarounds are perceived as a negative aberration of a process (Alter, 2014; Ferneley & Sobreperez, 2006; Rathert, Williams, Lawrence, & Halbesleben, 2012). Other studies show that these workarounds can help to understand the human interaction with the process and hereby form a fundament for process improvement (Azad & King, 2012; Cresswell, Mozaffar, Lee, Williams, & Sheikh, 2017; Lalley & Malloch, 2010; Petrides, McClelland, & Nodine, 2004). Aside from the number of studies, most of the studies on workarounds are executed in either the healthcare industry as many other industries are left to be unexplored.

Assessment of workarounds can help in understanding how processes are executed and detect differences in process design and process execution. Hence, guidance is provided for process optimisation by making knowledge of workarounds explicit (Beerepoot & Van De Weerd, 2018). The assessment of workarounds within the context of mission-critical IT has not been studied yet. Within mission-critical IT, time is often of the essence, and there is limited room for failure, which makes it similar to healthcare in its characteristics. Mission-critical refers to services that support the core mission and applications of their customers (Evans, 2013), failure of these services may lead to an inability to complete system or project objectives and have intangible economic costs (Hinchey & Coyle, 2010). Because of this pressure in time and quality, change management is relatively regulated and is expected to have a certain impact on failure.

By specifying the context further to assessment of workarounds in mission-critical change processes executed by self-steering teams there seems to be a compelling case. Within self-steering teams authority is decentralised and decision-making responsibilities are transitioned from traditionally management to individual employees or teams

(Lee & Edmondson, 2017). A particular management layer is often absent, and authority and ownership over tasks of teams in these organisations have been linked to improvements in work processes (van der Vegt, Bunderson, & Kuipers, 2010). Therefor it is interesting to analyse to what degree workarounds occur, what their impact is, what actions are taken in response and how this relates to the self-steering aspect of these teams.

Given this context, there is value in discovering what kind of workarounds exist within these mission-critical IT change processes and determine how these workarounds impact the process. Hence the objective for this research is to provide insight into what kind of workarounds exist in mission-critical IT change processes of self-steering teams, how these workarounds impact these processes, and what actions are taken in response. Knowledge about this impact and how to react to it will help mission-critical IT service providers be more in control of their processes and increase the quality and efficiency of their change management processes. Next to that, the objective is to determine what drives the occurrence of workarounds applied by self-steering teams in mission-critical change processes. These drivers will help to understand why and by what means workarounds occur in this specific context. To satisfy these objectives, the following research question is asked;

> *How do workarounds of self-steering teams impact mission-critical change processes?*

This question structures the research and to helps to investigate the problems statement as described. In the next chapter the relevance of this research is further explained.

1.2 Relevance

As stated by Alter (2015), “workarounds are important for systems analysis and design as ignoring them has negative consequences”. This research provides insight and offers guidance for impact assessment of workarounds in mission-critical IT processes and self-steering teams. It forms an example of how such an assessment could be performed and how results are analysed and helps organisations who want to assess potentially existing workarounds occurring in their processes. This study shows the potential of such research in this industry as it might motivate specific organisations in doing so themselves

As formulated in the problem statement, many studies perceive workarounds as a negative aberration of a process (Alter, 2014; Ferneley & Sobreperez, 2006; Rathert et al., 2012). This research contributes to the belief that workaround assessment and analysation form valuable input for process optimisation and thereby weaken the perspective of workarounds being a negative aberration of a process.

Lastly, the workarounds snapshot approach by Beerepoot and van de Weerd (2018), which is prominently used throughout the research, is designed to assess workarounds and provide various means for process optimisation. This research shows to be of relevance as it expands usage of this approach in a different industry than it has previously been used. Hereby a possibility is created for comparison of results between different industries or management forms.

1.3 Thesis outline

The research described in this document is structured as follows; Chapter 1 introduces the topic and research problem. Chapter 2 gives insight into relevant literature regarding the research topic. Chapter 3 is the research approach; here, the research is structured and described which methods are used. Chapter 4 explores how authority and decision-making are managed in self-steering teams by means of a systematic literature review. Chapter 5 gives insight into the characteristics of mission-critical change processes and sets a baseline to detect workarounds. Chapter 6 presents the two case studies in which workarounds are identified, and their impact is determined.

Chapter 7 shines a light on what the drivers are behind the occurrence of workarounds in mission-critical change processes. Chapter 8 and 9 describe the conclusion and discussion.

Finally, the document contains several appendices, Appendix A shows the consent form used for participant consent. Appendix B shows the interview questions which were used to gather information about workarounds. Appendix C provides the process models of the baseline processes conducted in chapter 5. Appendix D and E are the case study reports and Appendix F shows the survey results.

Chapter 2

Related literature

Business Process Management

In the early 1990s, the concept of business process redesign (BPR) emerged which focussed on improving the performance of businesses by radically redesigning the process as a whole rather than just automating existing ones (Davenport, 1993; Hammer, 1990). The concept later lost traction but laid a fundament and somewhat evolved into the phenomenon of business process management (BPM) which takes a broader and more incremental approach for process improvement. Dumas et al. (2018) define BPM as "the art and science of overseeing how work is performed in an organisation to ensure consistent outcomes and to take advantage of improvement opportunities.". This definition is rather broad, van der Aalst et al. (2003) defines BPM as "supporting business processes using methods, techniques, and software to design, enact, control, and analyse operational processes involving humans, organisations, applications, documents and other sources of information" (van der Aalst, ter Hofstede, & Weske, 2003). These methods and goals were later incorporated in the BPM lifecycle (van der Aalst, 2004). The original BPM lifecycle consisted of four phases; process design, system configuration, process enactment, and diagnosis. Throughout the years the BPM lifecycle has been evolved and has known multiple iterations (de Morais, Kazan, de Pádua, & Costa, 2014; Houy, Fettke, & Loos, 2010; Muehlen & Ho, 2006; Netjes, Reijers, & Van Der Aalst, 2006; Weske, 2007). One of the more recent iterations is that of Dumas et al. (2018), their BPM lifecycle aims to manage all phases of BPM and consists out of; process identification, process discovery, process analysis, process redesign, process implementation and process monitoring.

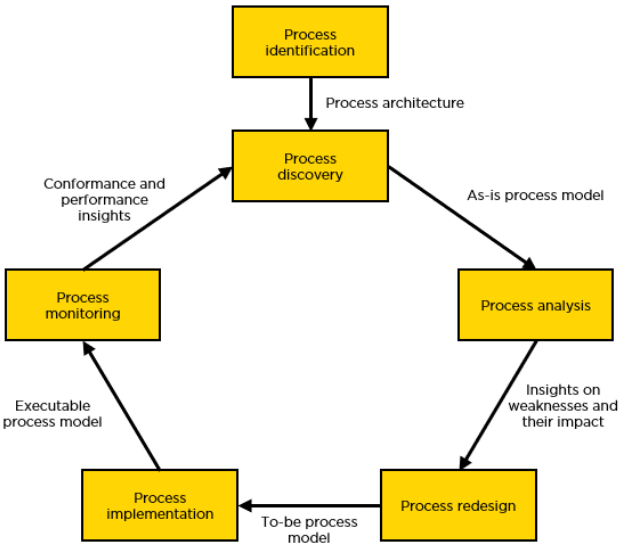


Figure 1. The BPM Lifecycle (Dumas et al., 2018)

Process identification is about identifying processes that are relevant for a posed business problem; the outcome of this is a process architecture that provides insight into the organisation's processes and relationships. Based on this architecture, a decision can be made which processes to manage as an answer to the business problem (Dumas et al., 2018).

Process discovery refers to determining the current state of each process. Typically, this results in an as-is process model (Dumas et al., 2018). Techniques that are used for modelling processes can be classified into three types (Figure 2); mathematical models, diagrammatic models, and business process languages (Vergidis, Tiwari, & Maieed, 2008).
Mathematical models are formal and work from mathematical notation; this provides structure but makes them complicated and harder to understand. Diagrammatic models are a visual display of a process, e.g. flowcharts. While they are easy to understand, they do lack a form of notation which could confuse the reader on what is meant by the model. Business process languages aim to be simple and consistent, are structured by notation, and have the potential to be executed, e.g. BPEL. Between business process languages and diagrammatic models is the combination of both. Visual models that follow a notation and are therefore easy to understand and less likely to be confusing as the notation dictates what each element means (Vergidis et al., 2008).

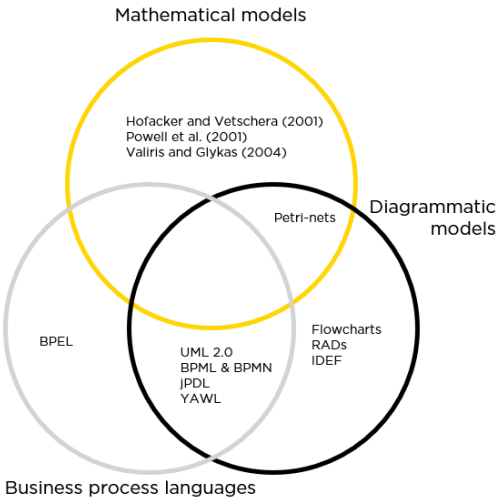


Figure 2. Venndiagram BPM models (Vergides et al., 2008)

Process analysis is the identification and documentation of issues that are associated with the as-is process (Dumas et al., 2018). It aims to provide insights on weaknesses in the process and their impacts. Process analysis is performed by different means; qualitative analysis (e.g. value-added analysis or waste analysis) and quantitative analysis. Qualitative analysis is performed via interviews and desk research and with techniques such as value-added analysis which tries to identify unnecessary steps in a process and eliminating them. Quantitative analysis leans heavy on mathematical calculations as it tries to determine efficiency and the performance of processes by calculating aspects such as throughput time, waiting time or capacity (Dumas et al., 2018)

Process redesign is concerned with identifying potential "changes to the process that would help to address the issues identified in the previous phases" (Dumas et al., 2018). There are many methods and techniques that help in redesigning processes and implement these changes (Gross, Malinova, & Mendling, 2019). For example, the one named before, BPR, which is oriented on radical change or lean, which focusses more on incremental changes. To determine the impact of such changes, the devil's quadrangle (Figure 3) is a useful mechanism. Changes are scored on four dimensions; time, costs, flexibility and quality. The advantage of the quadrangle is that it visualised that even though something is improved on one dimension, it might weaken other dimensions (Dumas et al., 2018).

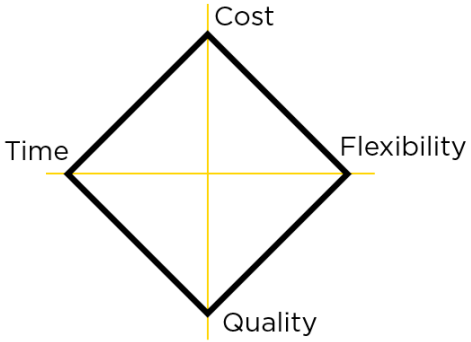


Figure 3. Devils quadrangle (Dumas et al., 2018)

Process implementation refers to the implementation of the redesigned process from the previous phase. While in theory, this would include simple changes such as manual administration, process implementation usually is executed from the perspective of automation via specific systems. These systems are called process-aware

information systems (PAIS). In the early years this started with workflow management systems and domain-specific PAIS (e.g. ERP, CRM), but later shifted more towards business process management systems(van der Aalst, 2010), (BPMS). BPMS are systems that support in managing most phases in the bpm lifecycle; design, analysis, execution and monitoring (Dumas et al., 2018). BPMS usually incorporate business process languages such as BPMN for process design and modelling, as these languages have semi-formal notations it provides opportunities for analysis, execution and monitoring of processes.

Process monitoring uses data generated by process execution in order to provide insights into the performance or conformance of a process. It can be performed on both historical and real-time data (Dumas et al., 2018). Techniques for process monitoring can be classified in statistic-based, which performs statistical analysis on measures such as cycle time, meantime, etc. or model-based which analyses the execution of a process based on its process model. The latter has the opportunity for both performance monitoring and conformance checking while the first mainly focus on performance monitoring (Dumas et al., 2018). In the case of conformance checking, model-based monitoring heavily depends on event logs generated by executing processes. These event logs are generated by BPMS or other PAIS. Conformance checking is a concept that compares a process model with event logs generated by process execution, this way it is possible to detect deviations to the process (Adriansyah, van Dongen, & van der Aalst, 2011). A relatively new but popular concept for conformance checking is process mining. Process mining helps to "discover, monitor and improve real processes by extracting knowledge from event logs" (Van Der Aalst et al., 2012). Aside from conformance checking and performance monitoring, process mining also enables automated discovery of processes based on event logs, variant analysis and diagnostics of differences.

Workarounds

Deviations or so-called workarounds are differences between a designed process and the execution of this process. Alter (2014) defines a workaround as: "A workaround is a goal-driven adaptation, improvisation, or other change to one or more aspects of an existing work system in order to overcome, bypass, or minimise the impact of obstacles, exceptions, anomalies, mishaps, established practices, management expectations, or structural constraints that are perceived as preventing that work system or its participants from achieving a desired level of efficiency, effectiveness, or other organisational or personal goals.". While BPM takes workarounds into account by conformance checking, it is limited in a way that it requires the presence of an IT system to generate event logs. The research field on workarounds significantly broadens the view on workarounds as it takes a more socio-technical stance, asking why and how workarounds occur, and what to do with them (Beerepoot & Van De Weerd, 2018). Although many studies perceive workarounds as a negative aberration of a process (Alter, 2014; Ferneley & Sobreperez, 2006; Rathert et al., 2012). There are also studies that find workarounds a solid ground for process improvement (Alter, 2014; Safadi & Faraj, 2010).

Beerepoot & Van de Weerd (2018) designed a method, the workaround snapshot approach, for process improvement by making knowledge about workarounds explicit; how do people work around the process, why do they do this and what actions should be undertaken. The approach includes the date of the snapshot, the workers that are involved, a description of the workaround, a visualisation of it in the form of a process model, the impact, motivation and at last the potential actions that can be undertaken as a reaction on the workaround. These actions; adopt, redesign, ignore, and prevent are incorporated in the workaround action matrix (Figure 4).

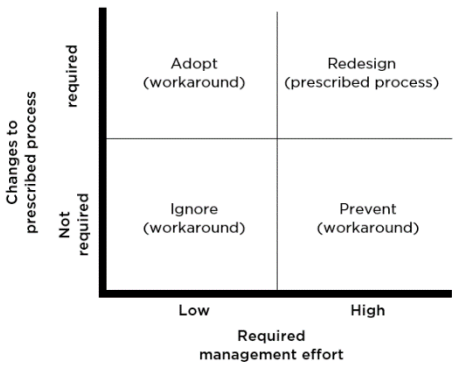


Figure 4. Workaround Action Matrix (Beerepoot & Van de Weerd (2018)

Even though many studies exist on workarounds, the concept itself is mostly studied in healthcare (Azad & King, 2008, 2012; Cresswell et al., 2017; Halbesleben, Wakefield, & Wakefield, 2008). This makes it interesting to expand that by researching workarounds in a different field such as mission-critical IT, as is performed in this study.

Mission-critical IT

Mission-critical IT is in some sense similar to healthcare; time is often of the essence; there is a limited room for failure and a constant demand for high quality. The term itself, Mission-critical, refers to services that support the core mission and applications of their customers (Evans, 2013), failure of these services may lead to an inability to complete system or project objectives (Hinchey & Coyle, 2010) and potentially result in substantial economic losses (Ponsard et al., 2007). Although, how mission-critical is defined differs per organisation. It depends on their needs and the impact on their mission (Paula Fraga-Lamas, 2017). For example, for a bank, it is mission-critical that their customers can withdraw and deposit money while for a postal service organisation, the sorting process is part of its mission-criticality. Identification of mission-critical IT can be difficult as the underlying systems that support these processes are often interdependent, which makes it hard to determine which specific systems or services are mission-critical (Paula Fraga-Lamas, 2017).

These services and systems are, many times, legacy systems that form the organisation's backbone in regards to IT (Bisbal, Lawless, Wu, & Grimson, 1999), thus require high-availability (99.999%+), this making it mission-critical (Gray & Siewiorek, 1991). Given the importance of these systems, it is of priority that they are managed with care and held stable at any moment in time. Characteristics of these kind of systems is that they are as stated before, highly available, secure, maintainable, and reliable (Ciccozzi et al., 2017; Paula Fraga-Lamas, 2017). Change management is one of those concepts that is required to progress and upgrade these systems but forms a threat to the stability of these services. Because of this, change management is fairly regulated and often follows a certain procedure in a mission critical setting.

Change management

Change management is a concept within information technology service management (ITSM), ITSM "focuses on defining, managing, and delivering IT services to support business goals and customer needs, usually in IT Operations." (Conger, Winniford, & Erickson-Harris, 2008), such as mission-critical. One of the most widely used initiatives for ITSM is ITIL (information technology infrastructure library). ITIL is a begin-to-end approach for service management (Arraj, 2013) and offers multiple components. ITIL is formed around five core pillars: service strategy, service design, service transition, service operation, and continual service improvement. Each pillar contains multiple components, change management is part of service transition.(McLaughlin & Damiano, 2007).

Change management itself controls the lifecycle of all changes executed on systems (McLaughlin & Damiano, 2007). "Its objective is to enable beneficial changes to be made with minimum disruption to IT services" (Cater-steel, Toleman, & Tan, 2006). These goals are driven by two statements (Gacek, Giese, & Hadar, 2008):

1. "to provide efficient and error free implementation of changes to assets and process.
2. "to minimise the level of service disruption"

Changes can be classified into three types; standard changes, normal changes, and urgent/emergency changes. A normal change are low-risk requests such as a request for access and happen occasionally. Standard changes are changes that are related to a service or infrastructure and follow the change process, such as an upgrade of a service. An urgent/emergency change is generally unplanned, and a reaction to a particular event, this type of change surpasses the usual test scenarios (Yarberry, 2007).

Self-steering teams

The concept of self-steering teams has emerged in the early 1990s in "response to challenges such as declining productivity and quality, and increases in employee dissatisfaction, absenteeism, turnover and counterproductive behaviour."(Manz & Neck, 1995). Self-steering or self-managing organisations are those whom radical decentralise authority in a formal and systematic way throughout the organisation (Lee & Edmondson, 2017). Nowadays, many organisations or teams deem themselves as self-steering. Although lots of differences occur in how the self-steering concept is executed. Lee and Edmondson (2017) introduced a model for self-managing organisations and stress that there is a spectrum to what degree authority is decentralised. A broader perspective on self-steering teams is provided in Chapter 4.

Chapter 3

Research approach

3.1 Research questions

The main research questions as formulated in the introduction is repeated below;

MRQ *How do workarounds of self-steering teams impact mission-critical change processes?*

In order to answer the main research question, a set of five sub research questions are asked. These sub-research questions drill particular parts of the main research question down with more specific and contextualised questions. The sub-research questions are formulated below;

SRQ1 *How do self-steering teams manage authority and decision making?*

SRQ2 *What characterises a mission-critical change process?*

SRQ3 *What types of workarounds are applied by self-steering teams in mission-critical change processes?*

SRQ4 *What is the impact of each workaround executed by self-steering teams on mission-critical change processes?*

SRQ5 *What drives the occurrence of workarounds executed by self-steering teams in mission-critical change processes?*

SRQ1 aims to get a better understanding of how authority and decision making is managed within a self-steering organisation. Especially these two aspects are interesting as it provides perspective for how decisions could be made in the following SRQs. For example, the way authority is distributed, and decisions are made will impact the way a decision is made regarding what actions can be undertaken in response to the identified workarounds in SRQ4.

SRQ2 aims to identify all the steps and activities that are present in the mission-critical IT change process. This helps to get a better understanding of the process while defining a baseline process on which workarounds can be determined.

With a defined baseline process in place, workarounds occurring in this process can be identified. SRQ3 explores the human interaction with the process and helps to identify workarounds in the process.

SRQ4 investigates the impact of each workaround executed by self-steering teams on the mission-critical change process. This impact is twofold; the consequence of a workaround and what actions are taken in response to the workaround.

SRQ5 explores what the drivers are behind the occurrence of workarounds by self-steering teams in mission-critical change processes. It does so by analysing previous results and incorporating specific questions in the interview protocol.

3.2 Research method

For this research project, I executed a systematic literature review (SLR), document analysis and a case study following the multiple case design (Yin, 2014). The SLR will help to answer SRQ1 as it describes a clear overview of the available literature regarding how authority and decision making is managed within self-steering teams. Document analysis will help to answer SRQ2 as documentation regarding the change process of four self-steering teams is analysed to investigate what characterises mission-critical change processes. For SRQ3, SRQ4, and SRQ5 a case study is executed. Within the case study, the workaround snapshot approach (Beerepoot & Van De Weerd, 2018) is applied. By experiencing first-hand how workarounds are incorporated in the change process and having direct contact with the workers that execute them will help to determine workarounds, their impact, the motivation behind them and what actions can be undertaken in response to these workarounds. The case study counts a total of two cases and is executed at Schuberg Philis, the Netherlands. Table 1 visualises an overview of the projected research.

Table 1. Overview of this research

Research question	Research method	Outcome	Chapter
SRQ1	Systematic literature review	Understanding of self-steering teams regarding authority and decision-making	4
SRQ2	Document analysis	Characterisations of a mission-critical change process	5
SRQ3	Case study	Workarounds occurring in the mission-critical change process	6
SRQ4	Case study	Impact analysis of each workaround	6
SRQ5	Case study	Drivers behind the occurrence of workarounds	7

The research method incorporates four phases; literature study, document analysis, case study and finalisation. To portrait these phases and actions within a process-deliverable diagram (PDD) used, which is presented in Figure 5. PDDs are used to model methods and consists of two integrated diagrams; on the left side, the process view and on the right side, the deliverable view (van de Weerd & Brinkkemper, 2008). The process view explains the flow of activities that are executed within the research project, while the deliverable view shows the concepts that are a result of the executed activities. The activity table is shown in Table 2, and the deliverable table is shown in Table 3.

The literature study phase represents the execution of the systematic literature review. By conducting a search query, a direct scope to the SLR is set. Based on this query literature is gathered and later reviewed. As stated above, this phase helps in answering SRQ1. A detailed description of this SLR is provided in sub-chapter 3.2.1

The document analysis phase aims to identify the change process of self-steering teams and thereby form input to determine what characterises this process. A total of five teams are analysed to identify such processes. A detailed description of this is given in sub-chapter 3.2.2.

The case study phase contains the full execution of the case study and workaround snapshot approach. This includes identifying workarounds, modelling process models, determining the impact, motivation, and actions that should be

undertaken in response to these workarounds. A detailed description of the case study is provided in sub-chapter 3.2.3

The last phase is the finalisation, results gathered from the previous phases are analysed, based on these results, drivers are determined behind the occurrence of workarounds by self-steering teams in mission-critical change processes. This same analysis helps to form a fundament to draw a conclusion on. The final report, in this case, the thesis, is written following the structure provided in the table of contents and is, ultimately, the last deliverable within the research project.

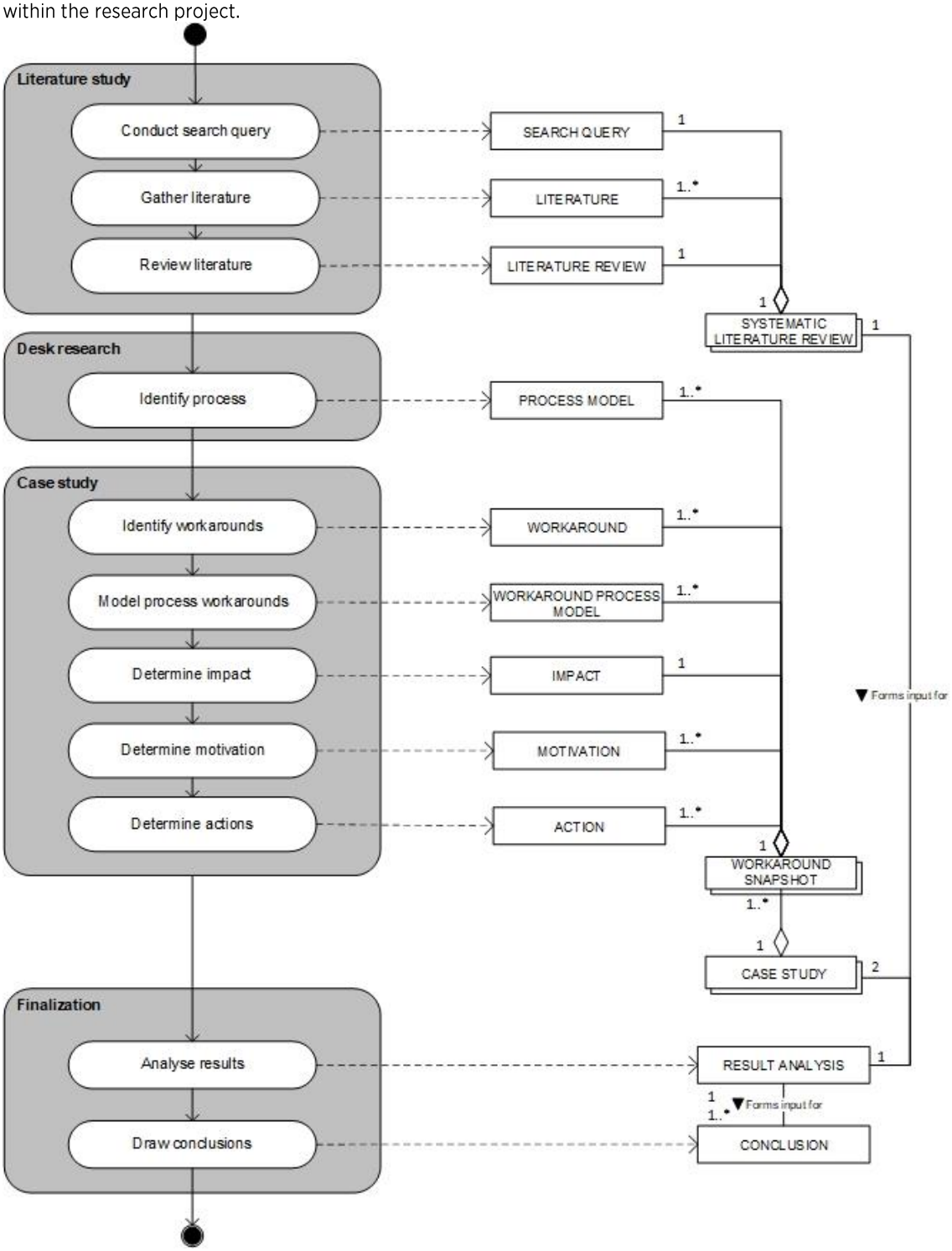


Figure 5. Process Deliverable Diagram (PDD)

Table 2. Activity diagram of the research project (PDD)

Activity	Sub-activity	Description
Literature study	Conduct search query	The conduction of a combination of search terms that together represent the scope of the project.
	Gather literature	LITERATURE is gathered through the execution of the SEARCH QUERY. This LITERATURE is gathered through an academic database.
	Review literature	Review of LITERATURE gathered from the previous step. A LITERATURE REVIEW is executed via a coding cycle.
Desk research	Identify process	Identification of the change management process, the process is identified via available documentation and potentially interviews. As a result, a PROCESS MODEL is created.
Case study	Identify workarounds	Identification of workarounds that occur in the identified process, this activity will result in zero to multiple WORKAROUNDs
	Model process workarounds	Based on the identified WORKAROUNDs, WORKAROUND PROCESS MODEL are created.
	Determine impact	Determining the consequences of a WORKAROUND.
	Determine motivation	Determining the reasoning behind why a person works around the identified process.
	Determine actions	Potential ACTIONs that are undertaken regarding a specific workaround.
Finalisation	Analyse results	Based on the execution, the results are analysed.
	Draw conclusions	Based on the analysation of the results, CONCLUSIONs are drawn.

Table 3. Deliverable diagram of the research project (PDD)

Concept	Description
SEARCH QUERY	A SEARCH QUERY is a combination of search terms that together form an insert for a database.
LITERATURE	A body of written works.
LITERATURE REVIEW	Selection of available documents on the topic, which "contain information, ideas, data and evidence written from a particular standpoint to fulfil certain aims or express certain views on the nature of the topic and how it is to be investigated, and the effective evaluation of these documents in relation to the research being proposed" (Hart, 2018).
SYSTEMATIC LITERATURE REVIEW	"a means of identifying, evaluating and interpreting all available research relevant to a particular research question, or topic area, or phenomenon of interest" (Kitchenham, 2004).
PROCESS MODEL	A visualisation of a process that describes all steps and activities taken that together form the process.
WORKAROUND	"A workaround is a goal-driven adaptation, improvisation, or other change to one or more aspects of an existing work system in order to overcome, bypass, or minimise the impact of obstacles, exceptions, anomalies, mishaps, established practices, management expectations, or structural constraints that are perceived as preventing that work system or its participants from achieving a desired level of efficiency, effectiveness, or other organisational or personal goals." (Alter, 2014)

WORKAROUND PROCESS MODEL	A PROCESS MODEL of the identified WORKAROUND.
IMPACT	A consequence of executing a workaround on time, costs, quality and flexibility of the process.
MOTIVATION	The reasoning behind why a worker executed a specific workaround.
ACTION	A response to a workaround. An ACTION is undertaken in response to the occurrence of a workaround.
WORKAROUND SNAPSHOT	"A snapshot of the essential information about the social and technical aspects of a workaround"(Beerepoot & Van De Weerd, 2018).
CASE STUDY	"A case study is an empirical inquiry that investigates a contemporary phenomenon in depth and within its real-world context, especially when the boundaries between phenomenon and context may not be clearly evident." (Yin, 2014)
RESULT ANALYSIS	The RESULT ANALYSIS is the output of the data analysis part of the case study, described in chapter 3.2.3.4 and 3.2.4
CONCLUSION	A reaction to the analysis of the result. A CONCLUSION is drawn based on the given results of the project.

3.2.1 Systematic literature review

SRQ1 is answered by performing a systematic literature review (SLR). "A systematic literature review is a means of identifying, evaluating and interpreting all available research relevant to a particular research question, or topic area, or phenomenon of interest" (Kitchenham, 2004). The reason for this form of literature review specific for this question is that the self-steering aspect within the scope of the project and case study company requires an extensive understanding of the meaning and influence on the organisation. With this SLR, a representative overview of existing studies is provided on self-steering/self-managing teams or organisations and how they manage decision making and authority. By scoping the SLR to authority and decision making, a fundament is provided to understand potential decisions made by the teams in the case study (chapter 3.2.3)

A key element of such a literature review is that it follows a specific process/protocol (Budgen & Brereton, 2006). The protocol used for this literature review is the PRISMA flow. The PRISMA flow is "designed to help authors improve the reporting of systematic reviews and meta-analysis" (Moher, Liberati, Tetzlaff, & Altman, 2009). A visualisation of this flow is provided in Figure 6. As the academic literature gap has already been explained in chapter 1, the last step of the PRISMA flow, the meta-analysis, is excluded as this is deemed as too much for the given goal.

The identification of literature was performed via the search engine of Scopus. Through the extensive filtering options of Scopus, a specific scope was set in order to find relevant literature. This scope is translated into a search query, which is as follows:

> (TITLE-ABS-KEY (*"self-managing"*) OR TITLE-ABS-KEY (*"self-steering"*) AND TITLE-ABS-KEY (*"organi*"*) OR TITLE-ABS-KEY (*"team*"*) AND TITLE-ABS-KEY (*"authority"*) OR TITLE-ABS-KEY (*"decision making*"*)) AND (LIMIT-TO (SUBJAREA , *"BUSI"*) OR LIMIT-TO (SUBJAREA , *"COMP"*) OR LIMIT-TO (SUBJAREA , *"SOCI"*) OR LIMIT-TO (SUBJAREA , *"DECI"*))

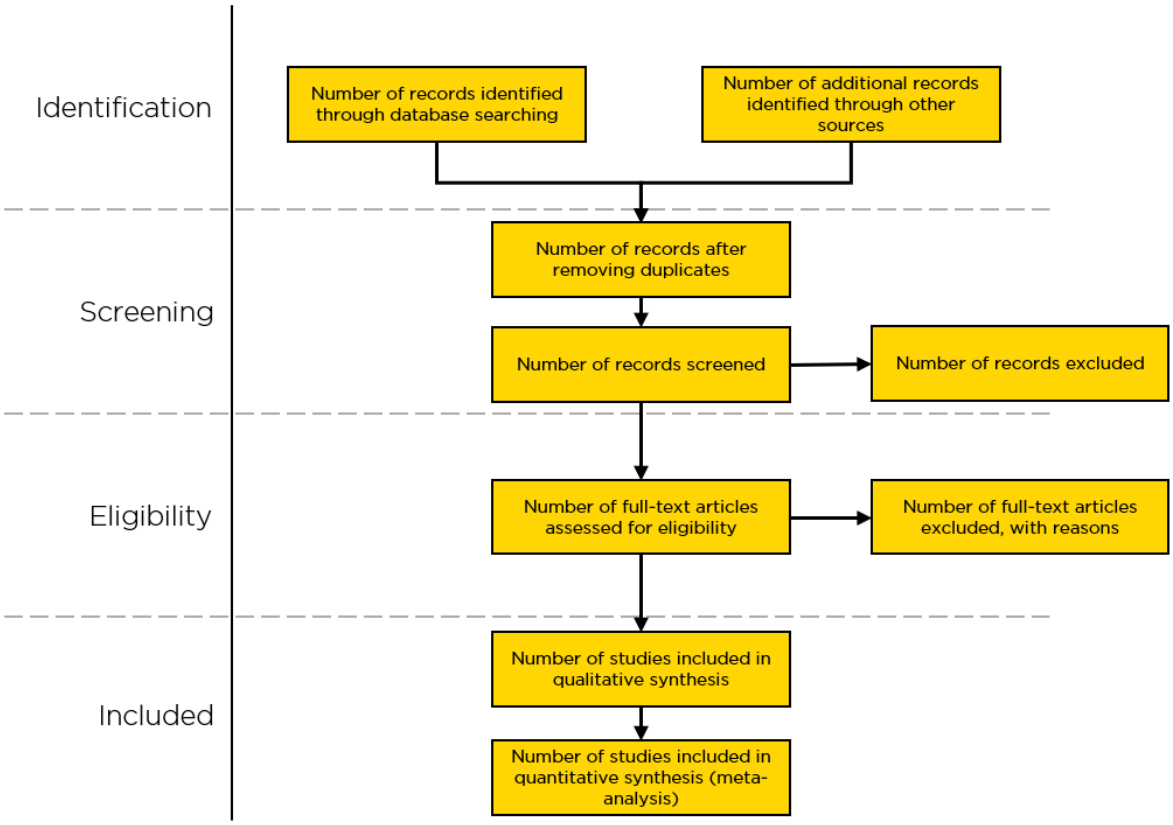


Figure 6. PRISMA Flow (Moher, 2009)

The query searches for titles, abstracts or keywords that include the words “self-managing”, “self-steering” in combination with “organi* or “team*” and “authority” or “decision making*”. The Asterix sign counts as a wildcard. Next to that, the search query limits itself by only looking in the research areas of computer sciences, social sciences, decision sciences or business, management & accounting.

Based on the gathered literature from the search results, duplicates are removed and based on the title and abstract of each paper, it is deemed relevant within the scope of the project or not. Papers are included when they focus on self-steering/self-managing teams or organisations.

The third phase relates to the eligibility of each paper, in order to draw conclusions on this a full-text review is performed at which is concluded whether the paper is relevant for the set scope of the project. Papers are excluded on three occasions; whether the paper is related to self-steering/self-managing teams or organisations, whether it is related to authority or decision making and whether the source is available online. These phases result in a list of included papers which are subject to the SLR.

As a result, 56 papers were found in the identification phase. While screening, 19 papers were excluded as they were either doubles or not related to the topic. Interesting to notice is that of these 19 papers a lot were focussed on self-managing software systems instead of organisations or teams. During the eligibility phase, another 19 papers were excluded; 7 because after a full-text review were not related to the topic (e.g. one paper focussed on firefighting teams), 4 because they were not related to either authority or decision-making. Another 8 papers were excluded as they did not meet the university’s standard and therefore not accessible. Figure 7 visualises this process.

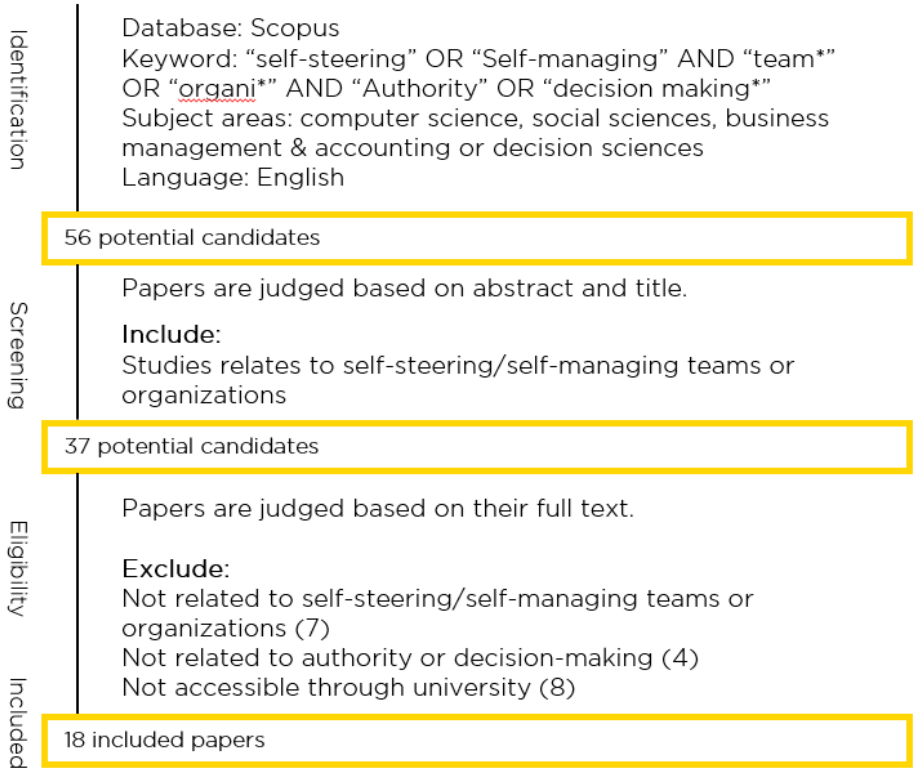


Figure 7. Results from PRISMA Flow

3.2.2 Document analysis

In order to answer SRQ2 and explore what the characteristics of the mission-critical change process of self-steering teams are, document analysis is performed. A total of four teams are analysed. These teams are from the same organisation as to where the case study is conducted. The teams that are subject for this are a postal service customer, grocery store chain customer and two banking customers. Based on available documentation, such as service-level agreements, contracts or other documents related to change management, a BPMN process model is created. These process models are compared to look for overlapping activities to eventually determine specific characteristics of the process. The process will be modelled following the structure of BPMN 2.0. With the use of Bizagi modeller[1]. Business process modelling notation 2.0 is a notation that intends to "provide a simple means of communicating process information to other business users, process implementers, customers, and suppliers." (Object Management Group, 2011). To ensure the validity of the information gathered from the documentation, the process extracted from it is presented to two members of the customer team and asked whether the process model is correct. The two people asked for approval are the customer operations manager and the engineering team lead.

As the determined characteristics could also belong to a standard change process instead of a mission-critical change process. Experts that are interviewed during both case studies are also asked to fill in a survey in which they are questioned about the mission-criticality of each characteristic and whether they would deem it part of a mission-critical change process or rather a standard change process.

[1] https://www.bizagi.com/

3.2.3 Case study

Studying workarounds can be a complicated matter as a real-world context is required that provide both a baseline process and workarounds that occur in the execution of this process. Because of this, a case study is conducted to answer SRQ2 to SRQ5. "A case study is an empirical inquiry that investigates a contemporary phenomenon in depth and within its real-world context, especially when the boundaries between phenomenon and context may not be clearly evident." (Yin, 2014). As described in the introduction, the case study company for this case study is Schuberg Philis, a managed service provider specialised in Mission Critical IT solutions located in the Netherlands. Because of the organisation's structure (described in the next sub-chapter), a choice was made to perform a case study with two cases following the multiple-case design of Yin (2014). By performing two cases, it is possible to detect differences in workarounds, impact and undertaken actions between customer segments. An overview of the multiple-case design is shown in Figure 8.

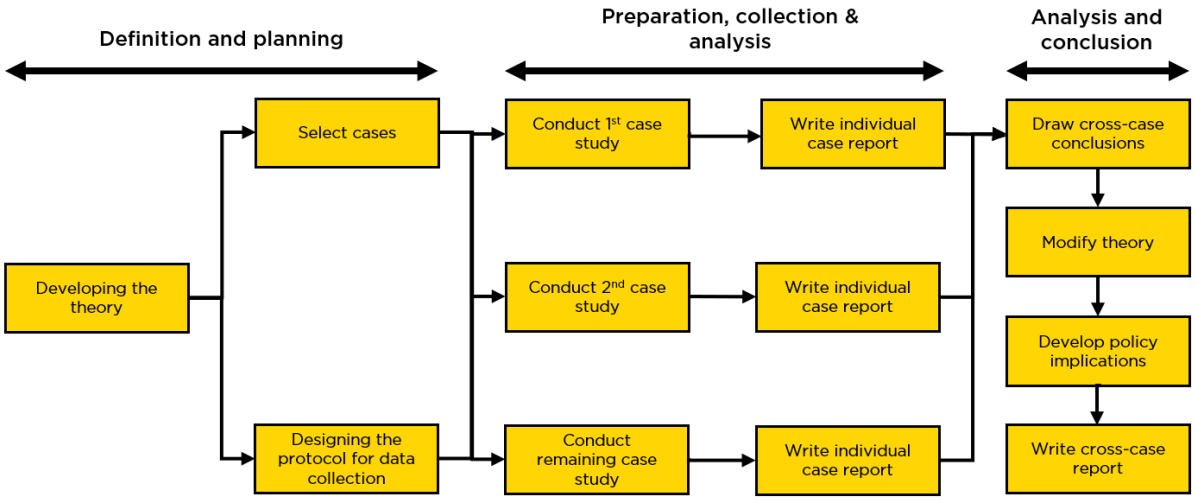


Figure 8. Case study design (Yin, 2014).

The multiple-case design consists of three phases; definition and planning; preparation, collection and analysis, and analysis and conclusion.

Definition and planning start with developing the theory, which is parallel with the creation of this document as it requires the practitioner of the case study to get familiar with the concepts and literature. The other aspects in this phase are the selection of the cases and the design of the case study protocol. Before a selection of the cases is made a description of the case study context is provided below.

3.2.3.1 Case study context

The company at which the case study is performed is Schuberg Philis. Schuberg Philis is a managed service provider specialised in mission-critical IT solutions and has an extensive history in infrastructure and hosting applications with as aim 100% availability. Currently, the organisation counts 286 employees (FTE) and has a revenue of ca. 53mln. Euros. Schuberg Philis is known as a market leader in their industry (Giarte, 2018). They serve a total of twenty-two customers, including large organisations such as Rabobank, Air France-KLM, and BNP Paribas. Each customer is segmented in one of three divisions; finance, asset management, and public utilities & retail (PUR). These segmentations are named greater customer teams (GCT) and are led by managing directors. Each GCT has multiple customer teams. These customer teams are dedicated teams that focus on a specific customer. Customer teams work independently of each other but are facilitated by their GCT. A customer team consists of a sales

representative, a customer operations manager (COM)/customer director and multiple mission-critical engineers (MCE). Aside from the GCT's and customer teams there is also the labs team which focusses innovation and the atom team which is focussed around networking. Aside from the greater customer teams and customer teams, there are also more traditional departments such as finance, HR, and recruitment. These relations are visualised in Figure 9.

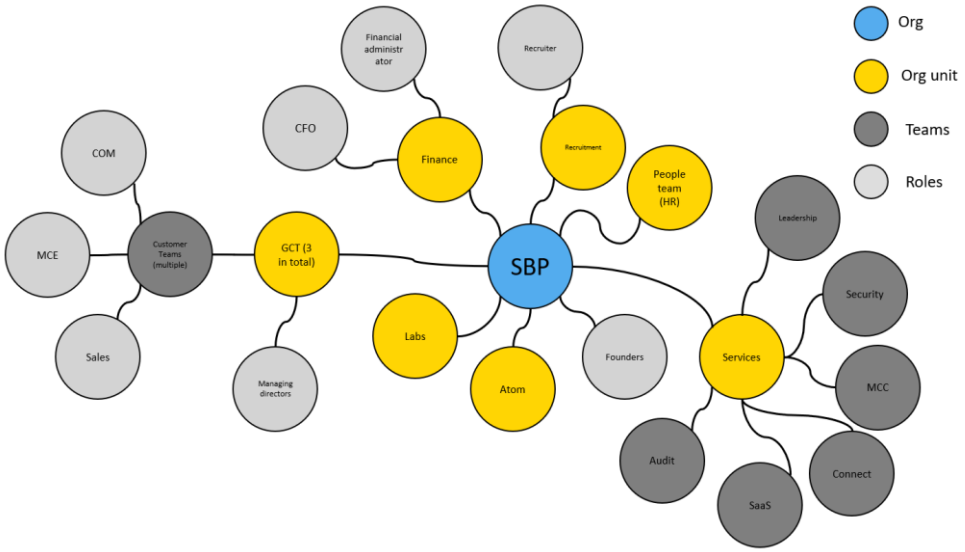


Figure 9. Organisation chart (Schuberg Philis)

Schuberg Philis describes itself as a self-steering organisation, with decentralised authority and no hierarchical management. This means that there are no traditional managers present as teams have the freedom and are required to make their own decisions as a team. This also means that each team is free to decide on how they serve their customer and which technologies they deem necessary in doing so. Next to the self-steering aspect the company has a culture of experts in the lead. This part of their culture, which could be seen as part of their competitive edge, places subject-matter experts as first point of contact for customers. Thereby, they eliminate different layers of bureaucracy between customer teams and the customer. For example, system architects who design a customer's infrastructure also deliver support on it. This is different to traditional managed service providers as they often incorporate helpdesks with several lines of support.

Within Schuberg Philis decision making is handled via the consent method. The consent method means that whenever decisions are made there needs to be consent for this. This used to be explicit consent but as the company grew in numbers this was changed to opt-out consent which assumes people give consent unless they object. People can only object if they are informed, and this is an important aspect of the consent method as who to ask for consent depends on the decision. For example, for decisions that impact one customer the specific customer team is asked for consent. But strategic decisions that impact the whole organisation are asked to the whole organisation. As decisions can be related to specific topics, Schuberg Philis creates circles. A circle is a group of people who have knowledge regarding a specific topic and share an interest on this topic and represent the organisation regarding that topic. If decisions regarding such topics is made, the members of that circle are asked for consent. Examples of such circles are a learning circle or software circle.

As described in the introduction, the freedom that comes with the organisation's structure and culture is expected to have an impact on the number of deviations on the studied mission-critical change processes. The change process describes the flow of actions that are executed when changes are applied to one of the systems. The structure and required activities can be specific to a customer team and could differ between customer teams. These variations are decided upon by the customer and the customer team but are mainly driven by the nature of the customer. For example, customers from GCT Finance are generally operating in environments that are regulated (by law) and overall incorporate more bureaucracy compared to while customers from GCT Public utilities & retail whom are

lesser regulated and deal with lesser forms of bureaucracy. This is expected to have an impact on how the prescribed process is structured and the workarounds that are to be identified.

3.2.3.2 Selection of cases

As described above the case study will consist of two cases. The choice for two cases was made to enable future comparison and because of the availability of the case study companies. In both cases the mission-critical change process of a specific customer team is analysed. The customer teams that are subject in the two cases are selected on the following criteria:

- The customer team consists of at least six members. This number of team members is chosen to prevent a potential lack of data when participants drop out.
- The team must be available to contribute to the research project. This means team members are at least available for a one-hour during interview and a two-hour during focus group session.
- Documentation about the mission-critical change process that describes all activities in the process from start to end must be present.

The customer teams that are the object for the two cases are bank 1 from GCT Finance and the postal service team from GCT Public utility & retail. The data collection procedure is identical for both cases and further described below.

3.2.3.3 Data collection

As the workaround snapshot approach is embedded in the case study, data is collected through various means depending on the stage of the approach. By using this approach, insight is provided on multiple aspects of a workaround. The snapshot includes the date of the snapshot, the workers that are involved, a description of the workaround, visualisation of it in the form of a process model, the impact, motivation and at last the potential actions that can be undertaken as a reaction on the workaround. Table 4 displays this structure.

Table 4. Workaround snapshot component structure (Beerepoot and Van de Weerd, 2018)

Snapshot component	Content
Date of snapshot	The date the snapshot was created
Workers	Roles that are involved in the workaround
Description	Concise textual description of the workaround
Process model	A process model of the workaround
Impact	Impact of the workaround in terms of the devil's quadrangle.
Motivation	Description of the worker's motivation to workaround
Possible actions	Inventory of actions that can be taken; prevent, adopt, redesign, or ignore.

Data collection sources used within the case study range from documentation, interviews, direct observations and focus groups. Below each activity in the case study phase displayed in the PDD (Figure 5) is explained, these activities are derived from the Workaround Snapshot Approach.

Identify workarounds

To identify workarounds the process model from the previous step is taken as the baseline at which workarounds are to be identified. Identification of such workarounds is performed via both interviews and direct observations. Interviews are used to retrieve a general idea of the workarounds that occur and to discover the steps and activities

taken that make it a workaround. Direct observation is then used to validate the workarounds and see whether the worker performs the workaround as were described.

The type of interviews that are used are shorter case study interviews, defined by Yin (2014) as interviews that focussed and took about an hour or so while remaining open-ended but are likely to follow the protocol (or a portion of it) more closely. These interviews generally follow a script. The script created as part of the protocol that can be found in appendix B interview questions. Each interview is later transcribed, and workarounds are marked.

As stated above, direct observation is used to verify what has been described as the workaround in the interviews. Although the reasons for this is of multiple ends; partly verification but also for the researcher to feel familiar with the process and workarounds at hand. Direct observation is fieldwork, where the researcher attends the execution of, in this case, a workaround, to experience and assess the behaviour (Yin, 2014). As changes are executed in regard to an information system, the researcher is likely to sit next to a mission-critical engineer to see how he/she performs the workaround.

For the postal service team, five people were interviewed and for the bank 1 team seven, resulting in twelve interviews in total. For both teams the team lead was interviewed, the other interviewees fulfilled the role of mission-critical engineer. Mission-critical engineers are generally the people that execute changes and have a role in occurring workarounds. Both the team leads and mission-critical engineer that are interviewed have more than ten years of experience in their field of work. All participants are asked to sign the consent form shown in appendix A consent form to get a general consent on the usage of the interview data.

The interviews provide data for different sections of the workaround snapshot. For example, the description, context, and motivation are extracted from these interviews. Both the interviews and direct observations provide information to create a process model of the workaround.

Model workarounds

As workarounds are identified through the means described above, the actual process model of these workarounds is created following the structure of BPMN 2.0 using Bizagi modeller as a tool. The process model of the baseline process is used as a template and workarounds are modelled as an overlay to the original process using a different colour to show the deviations in the process. Each workaround gets a specific process model which is afterwards validated by asking the participant that provided the workaround whether the process model is correct in terms of steps and activities.

Determine motivation

The motivation behind a workaround is gathered through interviews. Questions regarding the motivation are incorporated in the interviews from the first step, depending on whether workarounds are identified. Just like the protocol for the first steps, a description of this motivation is validated by the interviewee afterwards to ensure that data collected is correct.

Specific questions regarding motivation incorporated in the interview are shown in appendix B interview questions. As these questions are asked in the first interview, information extracted from them is possible by the transcription in combination with the marked workaround.

Determine impact

The impact is measured on the accounts of the time, costs, flexibility and quality following the perspective of the devil's quadrangle (Dumas et al., 2018) in the setting of a focus group. A focus group is a technique involving in-depth group interviews at which the group is focussed on a specific topic (Rabiee, 2004). Participants in this focus group are members from the same customer team and preferably the persons that were previously interviewed, in both sessions, the engineering tech lead is also attending. The size of the focus group is aimed to be six including the following roles;

- Engineering tech lead
- Mission-critical engineers

During these sessions, participants are asked to score the impact of each workaround on the original process by measurements of the devil's quadrangle (cost, time, quality & flexibility). In order to decide, the snapshot of each workaround is presented which contains the description of each workaround, the process model and the motivation. Participants score each measurement with a --, -, 0, + or ++. A – means a negative impact on the original process and –- a significant negative impact. A 0 indicates a neutral impact.

Determine actions

Since the case study company is defined as self-steering in its organisation and no hierarchical management level is present the decision on what actions should be undertaken in response to the workarounds is performed via the consensus method used at the case study company. Participants generally vote for their preference but together decide which action is taken on what workaround.

This decision-making process is monitored through a focus group. The focus group consists out of the same people, and therefore the same roles that were present during the focus group sessions for the determination of the impact of each workaround. This is done so that the participants involved are familiar with the workarounds at hand. During the focus group session, a recap will be given of each workaround and its specific characteristics such as the process model, description, impact and motivation.

Based on these characteristics, the participants may vote for an action from the workaround action matrix. Participants are asked to explain their reasoning behind their choice. In the end, the participants need to conclude to which action they seem fit for each workaround.

Identify drivers behind occurrence of workarounds

The workaround snapshot approach helps to identify different aspects of workarounds. Next to this approach, it is observed whether team members feel that there is a connection between the self-steering aspect and the occurrence of workarounds in the mission-critical change process. This is done to identify what drives the occurrence of workarounds executed by self-steering teams in mission-critical change processes.

To determine this connection and what drives the occurrence of workarounds, several questions are added in the interview protocol described in Appendix B. These questions are asked at the end of each interview during the short case study interviews described earlier.

3.2.3.4 Data analysis

The data analysis is separated in two parts: within-case analysis and cross-case analysis. Activities that are part of the workaround snapshot approach are subject for within-case analysis. Workarounds and their different characteristics are determined for both teams and handled in separate cases. For the identification of the drivers behind the occurrence of workarounds a cross-case analysis is conducted.

Within-case analysis

For the identification of the process, a business process model is created as a means of analysis as this creates a general overview of the characteristics and steps taken in the change process. This process model is based on the available documentation of both teams and is the end-result of the previous research method, document analysis.

Data collected to identify workarounds is analysed by coding the data in one cycle (Miles, Huberman, & Saldana, 2014). The interviews and observations are transcribed and marked. Personal data from these interviews and notes is anonymised and ordered per case. Coding is performed using an inductive coding cycle via the process coding. This type of coding uses verbs to connote participant actions/interactions with, in this case, the change process. The motivation behind each workaround is analysed using one cycle coding as well, although now by descriptive coding to summarise where motivations are stated in the transcriptions.

The model of each workaround is based on the process model of the baseline process and shows deviations with BPMN 2.0 activities in red. The actual activities that that make the deviation are extracted from the coding cycle from the previous step. The process model is discussed with the interviewee to verify correctness.

For both impact determination and action determination, a deductive analytical approach is used (Miles et al., 2014). Literature is used to derive impact measures and potential actions. The devil's quadrangle explained in chapter 2, scores each workaround on the dimensions of time, costs, quality and flexibility (Dumas et al., 2018). Comparing this to the initial process will show in-/ or de-creases in each dimension and thereby determine impact. Actions are determined by using the workaround action matrix introduced by Beerepoot and van de Weerd (2018). The action matrix takes four actions into account; adopt, ignore, redesign and prevent. Given the self-steering character and consent method used in the case study company, choices on what action is undertaken is a group decision.

An overview of the data analysis techniques used per step in the workaround snapshot approach is displayed in Table 5. Figure 10 also shows a visual representation of the overview of the within- and cross-case analysis.

Table 5. Data analysis overview

Step in the workaround snapshot approach	Type of data analysis
Identify workarounds	One cycle coding: process coding (Miles et al., 2014)
Model process workarounds	BPMN 2.0 / coding previous step
Determine impact	Deductive coding: Devil's Quadrangle (Dumas et al., 2018)
Determine motivation	One cycle coding: descriptive coding (Miles et al., 2014)
Determine actions	Deductive coding: Workaround action matrix (Beerepoot & Van De Weerd, 2018)

Cross-case analysis

To determine what drives the occurrence of workaround executed by self-steering teams in mission-critical change processes a cross-case analysis is executed based on the collected data.

Data collected to identify these drivers is analysed by two-cycle coding (Miles et al., 2014). As the data collection method is incorporated in the earlier conducted interview protocol, these interviews are already transcribed, marked and made anonymous. Coding is performed inductive, in the first-round open coding is used to identify drivers. Based on the outcome of the first coding cycle, descriptive coding is used. This is done to detect specific themes and thereby define the drivers.

As presented in Figure 10, the within-case analysis forms an in-depth study of both cases. The process characteristics, workarounds and several aspects of these workarounds are extracted. Aside from this the connection between self-steering and the occurrence of workarounds is determined. In the cross-case analysis the characteristics, motivation, impact, actions and determined drivers are further analysed and linked to each other to conclude on whether these drivers' reason with specific findings.

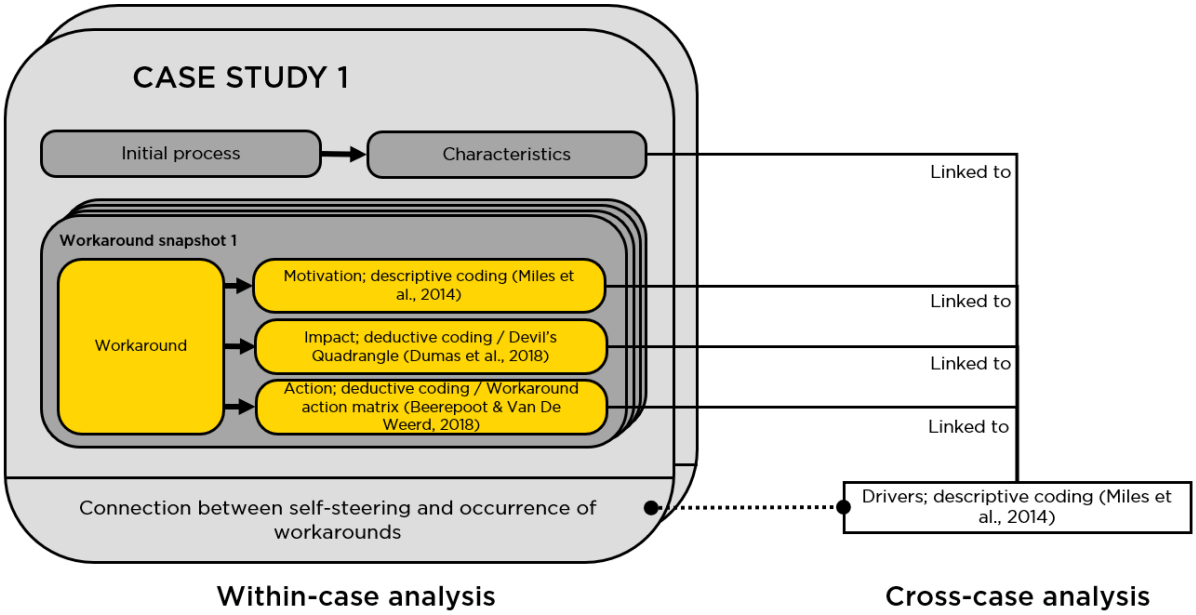


Figure 10. Within-case and cross-case analysis overview

Chapter 4

A closer look at self-steering teams

Systematic literature review

4.1 Introduction

As stated in the research approach, eighteen papers were analysed in order to formulate an answer to SRQ1; *"How do self-steering teams manage authority and decision making?"*.

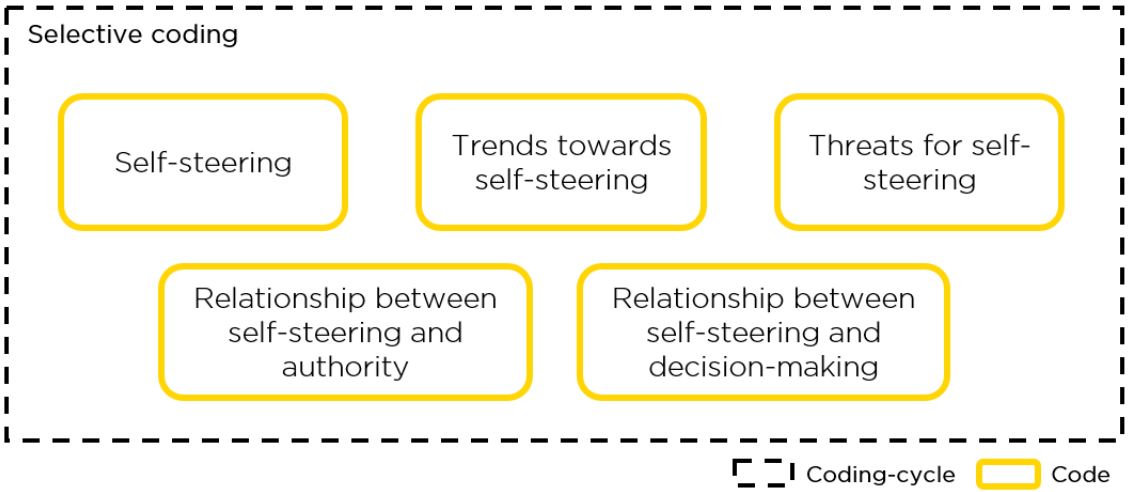


Figure 11. Coding scheme for the SLR

To analyse these papers, I imported them into NVivo and coded them to structure my findings and increase the traceability within the SLR. For this deductive coding is used (Miles et al., 2014), the coding scheme is visualized in Figure 11. In total, five categories were coded; self-steering, the relationship between self-steering and authority, the relationship between self-steering and decision-making and trends towards self-steering and threats for self-steering. The first three were purely designed to align with and answer SRQ1 as well as how self-steering teams manage authority and decision-making might influence the existence of workarounds in mission-critical change processes. The two latter keywords were designed to get a better understanding of how the concept of self-steering established itself as an organisational style and what threats occur in these teams. Examples of each code are provided in Table 6. In total, 107 quotations were coded.

Table 6. Examples of coding results for the SLR

Code name	References	Codes	Example
Self-steering	12	34	"A team was considered to be a self-managing team when "team members jointly decide how the work is to be done" and when "the teams are given responsibility for specific

			products or services."(Park, 2012)
Relationship between self-steering and Authority	7	24	"Decentralization of authority can occur in numerous ways, either by decreasing the number of levels of formal authority (i.e. "flattening" the formal hierarchy) or by creating a more equitable distribution of authority across existing hierarchical levels." (Lee & Edmondson, 2017)
Relationship between self-steering and decision-making	10	31	"They have the autonomy to make decisions that are traditionally the responsibilities of supervisors and managers."(Alper, Tjosvold, & Law, 1998)
Trends towards self-steering	8	13	"One implication of operating in the knowledge economy is that managers rarely have the full expertise needed to solve organisational problems. Instead, individuals at all organisational levels must contribute information and ideas for their organisations to succeed" (Lee & Edmondson, 2017)
Threats for self-steering	3	5	"The issue of leadership of self-managing employees has been particularly challenging. The questions "How does one go about leading employees that are expected to increase lead themselves?" captures the essence of this challenge" (Neck & Manz, 1994).

4.2 Trends

Throughout the year's different organisational styles have emerged and gained popularity. In the 1970s, job enrichment became a thing and provided employees with more responsibilities and more complex tasks. In the 1980s and 1990s surfaced as employee empowerment, which transformed people with relatively no power into those who possess personal control (Liden, Wayne, & Bradway, 1997). Self-steering or self-managing teams have been a well-known phenomenon in relation to employee empowerment while being a longstanding concept it gained more traction in the early 1990s in response to several challenges such as a decline in productivity, quality, and an increase in employee dissatisfaction (Manz & Neck, 1995). More recent research defines three trends towards this less-hierarchical form of organising; first, the increase in the pace of information flows and technological development. Second, the change toward more knowledge-based work where managers are not able to possess the information to solve problems by themselves. Third, the trend where work typically changes to become a place for personal meaning as people look for personal fulfilment through their work more than previous generations do (Lee & Edmondson, 2017). Interesting to notice is that all papers that referred to the effects of self-steering teams showed positive improvements as teams of a self-steering nature were credited with improvements in productivity, job satisfaction, turnover and a decline in absenteeism (Alper et al., 1998; Lee & Edmondson, 2017; Liden et al., 1997; Manz & Neck, 1995; Park, 2012; Sagie & Aycan, 2003) and are in relation to managerial hierarchy more efficient under dynamic conditions (Lee & Edmondson, 2017). Out of the 19 papers, only one paper provided adverse effects, as it stated that the productivity gain only comes from the elimination of supervisors rather than the team itself (Sagie & Aycan, 2003).

4.3 Authority

Lee and Edmondson (2017) define self-steering teams as those who radically decentralise authority formally and systematically. Formal and systematic in a way that it is codified how authority is decentralised through an explicit set of rules or principles. Thus implying that authority is not delegated via informal arrangements between managers and subordinates and that employees hold significant decision rights that cannot be suspended by someone because they have a "higher" position (Lee & Edmondson, 2017). Self-steering teams generally do not have a hierarchical structure (van Riemsdijk, Jonker, Rens, & Wang, 2012) and are fully responsible for specific segments of work (Alper et al., 1998) as team members jointly decide how they want to do their work (Alper et al., 1998; Park, 2012). In practice, the role of a manager is often eliminated. Therefore individuals are more required to possess self-management and self-leadership skills as the team as a whole needs to regulate itself (Moorhead, Neck, & West,

1998). This translates to a relatively low power distance and high levels of individualism (Sagie & Aycan, 2003). As authority is decentralised and power is transitioned to the lower ends of the organisation, this does not directly mean that authority is equalized (Lee & Edmondson, 2017), the need for self-management and self-leadership creates room for more natural leadership. An aspect that is often overlooked is that elimination of a "manager" role does not imply that these teams are devoid of managerial work (Lee & Edmondson, 2017). This type of work still exists and is potentially needed to be successful.

Nowadays, many organisations deem themselves self-steering or incorporate self-steering teams, although Lee & Edmondson (2017) point out that there are differences to what degree these organisations or teams decentralise authority as self-steering is more of a spectrum. This is in line with what Lui, Wang, Hui & Lee (2012) state as they determine a difference between self-steering team climates in the level of control over the processes of deciding who performs which tasks, when to perform them, and how to make work-related decisions" (Liu, Wang, Hui, & Lee, 2012). Aspects such as work execution and monitoring work are likely to be decentralised, but aspects such as firm strategy or resource allocation are less likely to be decentralised. Compared to a traditional managerial hierarchy it is noted that these predominantly do not decentralize authority on any of these aspects (Lee & Edmondson, 2017).

4.4 Decision making

This spectrum to what degree authority is decentralised is tightly connected to how decision making is handled in self-steering teams. As stated earlier, authority is decentralised and responsibilities and decision making power shifts from management to employees at lower levels (Alper et al., 1998; Kuipers & De Witte, 2005). These self-steering teams are a form of participate decision-making (PDM) (Sagie & Aycan, 2003), which translates to the opportunity team members have in exercising control over aspects of their work and the outcomes of their work (Liu et al., 2012) This means that team members now have the autonomy to make decisions that were traditionally the responsibility of supervisors or managers (Alper et al., 1998; Lee & Edmondson, 2017; Moorhead et al., 1998).

This autonomous decision making or local decision making is stated as a key mechanism to self-steering teams (Kuipers & De Witte, 2005; Molleman, 2000). Autonomous decision making can be defined along with three perspectives; the freedom of employees to formulate the objectives of their work, planning, and distribution of their work, and the freedom to choose their working method (Molleman, 2000). Autonomous decision making also shows to be an indicator of the effectiveness of self-steering teams as the increased control over their work drives the perception of psychological ownership (Liden et al., 1997; Liu et al., 2012).

4.5 Threats

Aside from the benefits of self-steering teams, there are also some threats that impact the effectiveness and/or efficiency of these teams. One of these threats is regarding the way decision making is handled and is coined as groupthink (Manz & Neck, 1995). Manz & Neck (1994) define groupthink as "the tendency for group members striving to agree with one another to interfere with rational, constructive decision-making processes.", which ultimately results in dysfunctional decision making. Self-steering teams tend to vulnerable to this as they fall victim more easily to conformity and are more cohesive. Some indicators of groupthink are the self-censorship of team members to avoid deviating from the general consensus, applying pressure to team members who challenge the common beliefs and a shared illusion of unanimity (Manz & Neck, 1995).

Another threat is that of leadership. As members of self-steering teams are expected to possess self-management and self-leadership skills, it could potentially be that these expectations are not met (Manz & Neck, 1995). Druskat and Wheeler (2004) stress that self-steering teams require a unique sort of leadership in the form of external leadership. External leaders are more focussed on facilitating teams rather than leading them as they are more oriented to the team's position in the bigger picture of an organisation (Druskat & Wheeler, 2004).

To refer back to SRQ1 and to conclude on this question, self-steering teams manage authority in a decentralised way where, depending on how far organisations go, employees are given the responsibility to decide how they want to shape most aspects of their work. The lack of a hierarchical structure and often elimination of a manager requires team members to possess self-management and self-leadership skills. While authority is decentralised, it does not necessarily mean it is equalised as there is more room for natural leadership. This highly relates to how teams of self-steering nature manage decision-making, which is executed in a local and autonomous way as decision-making responsibilities are transitioned from management to team members. Teams formulate their own work objectives, method, planning, and distribution of work.

Chapter 5

Characteristics of mission-critical change processes

Document analysis

As stated in the research method, the characterisations of mission-critical change processes are determined by modelling and comparing the mission-critical change process of four customer teams within the case study company. Hereby, I aim to answer SRQ2: *"What characterises a mission-critical change process?"*. Project documentation of earlier projects formed input for this. Four customer teams had specific documents dedicated to describing change management. Aside from this document, all four teams maintained a document named daily agreed procedures (DAP), which also covered the change management process. Table 7 shows which documents were analysed per team.

Table 7. Document analysis

Customer team	Document name
Postal service	Change management process
	Daily agreed procedures (DAP)
Grocery store chain	Change management process
	Change management flow
	Daily agreed procedures (DAP)
Bank 1	Change management process
	Daily agreed procedures (DAP)
Bank 2	Change management process
	Daily agreed procedures (DAP)

While analysing the available documentation of each customer team, it comes to notice that the extensiveness of documentation varies. The overall structure of the documents was similar, although the amount of detail in which the process was described was different. For example, one team only described the general process flow provided with plain explanation. Another team thoroughly described the process and exceptions to certain steps which made it more explicit how the process is structured.

Aside from the differences in extensiveness in the documentation, some of the documents contained contradictions within the process description. For example, one of the Public Utility & Retail customers described the roles and responsibilities incorporated in the process and further in the document showed a process model which contained different roles but described the same process. This could be labelled as sloppy although the document itself is agreed upon by both the customer and the customer team. Because of these contradictions, the models that are created were verified by the customer operations manager (COM) of each team to claim truthfulness and completeness.

Another interesting thing is that the documentation was relatively dated. Only one of the customer teams had documentation that was updated in the past twelve months. Documentation of the other teams was not updated in at least twelve months. While this might raise suspicion to whether the content is these documents is still relevant, the COMs of each team ensured the versions under review are the documents they are held accountable for.

5.1 Process models

The mission-critical change process of four customer teams is in BPMN2.0. For this, the tool Bizagi modeller is used. These models are shown Appendix C Baseline processes.

5.1.2 Public, Utility and Retail

As described in the case study context, the greater customer team public utility and retail (PUR) is focused on serving customers who operate in public utility and retail industry. Customers of this sort include airlines, port operators, postal services and grocery store chains.

Postal service

The mission-critical change process of the postal service customer, displayed Appendix C (Figure postal service), involves four different roles in the process; submitter, 1st change approver, 2nd change approver, and executor. The role of the submitter is fulfilled by either someone from Schuberg Philis or the customer (for example, an engineer, management or a functional admin). The role of change approver is fulfilled by either the customer or Schuberg Philis. Depending on who the submitter is, a person from the opposite party must approve the change. The last role is the executor; this is the person who executes the actual change.

Table 8. Roles and responsibilities (postal service)

Role	Person fulfilling the role	Responsibilities
Submitter	A representative of customer/representative of Schuberg Philis	Submission of change tickets, initiation of risk/impact analysis, create test scenarios, classification of a change, test change outcomes
1st approver	A representative of customer/representative of Schuberg Philis	Reviewing and assessing changes
2nd approver	A representative of customer/representative of Schuberg Philis	Reviewing and assessing changes
Executor	An engineer from Schuberg Philis	Executing changes, communicating results, update relevant documentation and handling the end-stage of change tickets.

The process starts with a change initiation. This initiation can occur by various means such as a patch, new requirement or bug. When a change is initiated, the submitter creates a ticket in Jira. Within the ticket, general information such as title, cause and reason are defined. Next to that, a risk/impact analysis is initiated, and test scenarios are created. The risk/impact analysis is essential as this determines the number of approvals required and test scenarios are used to check whether the change was executed successfully. Once both the risk/impact analyses and test scenarios are ready, the ticket is classified. Changes classified as an emergency will bypass the approval mechanism, as these changes need to be executed as soon as possible.

Non-emergency changes are set to be assessed by the first change approver, as stated earlier, this is someone from the opposite party as the person who submitted the change. If the change approver disapproves a change, it is redirected back to the submitter which has the option to either adjust the ticket and submit it again or close the ticket. The latter option leads in the process to end.

If the first change approver approves the ticket, it directs to either the executor or the second change approver. This is based on the impact of a change, low/normal changes do not require a second approver, but high impact changes do. This form of approval is the same as the previous approval. Although the second approver is someone from the opposite organisation as the first approver. Upon approval, the executor executes the change. After the execution, the results are communicated with each stakeholder and the submitter tests the outcome of the change. The executor then updates the relevant documentation and closes the ticket. The process has hereby ended.

Grocery store chain

The mission-critical change process for the grocery store chain customer follows a different flow than that of the postal service customer. Displayed in Appendix C (Figure Grocery store chain), it shown that within the process, five different roles are involved: the submitter, the change coordinator, the change manager, the change advisory board and an executor.

Table 9. Roles and responsibilities (grocery store chain)

Role	Person fulfilling the role	Responsibilities
Submitter	A representative of customer/representative of Schuberg Philis.	Submission of change tickets, initiation of risk/impact analysis, create a rollback plan, create test scenarios.
Change coordinator	Someone mandated by customer (always someone at the customer).	Reviewing change tickets based on completeness and accuracy and determining priority
Change manager	Someone mandated by customer (always someone at the customer.)	Reviewing and planning changes.
Information officer customer	a person responsible for IT security at the customer.	Final approval security changes
Executor	An engineer from Schuberg Philis.	Execute and potentially rollback changes, option to reject changes.

The role of the submitter is fulfilled by either someone from Schuberg Philis or the customer. In the case of Schuberg Philis, this is either the COM or an engineer. In the case of the customer, it is someone from a specific group who are allowed to submit change requests. The change coordinator is someone who is mandated by the customer to check the validity of a change; this is always someone from the customer side. The change manager is similar to the change coordinator role but is delegated to fewer people. Aside from a change coordinator and change manager, the process also involves the information security officer; this is the person responsible for information security at the customer side. Last is the executor; this is the person who executes the change and is always an engineer from Schuberg Philis.

The process starts with a change initiation. Changes can be initiated by various means; new requirements, software releases and patches or infrastructure changes. Once a change is initiated, the submitter conducts a risk/impact analysis, test scenarios, and a rollback plan. Upon completion, details are documented in a ticket.

From there on the ticket is up for the first approval, this is done by the change coordinator. During the first approval round, the change coordinator reviews the ticket based on completeness and accuracy. When disapproved, the ticket comes back to the submitter who gets the chance to either change the ticket or close it. When closed, the process ends, but when a changed ticket comes back to the change coordinator. Upon approval, the change coordinator determines the priority of the change.

Based on this priority, the ticket follows another approval flow. If the change is labelled as standard, it does not require another approval and is set for the next step, "plan change". If the ticket is not standard, it needs to be assessed by the change manager. When the ticket is disapproved the submitter has once again the option to change it or close the ticket. When approved, the ticket is set for the "plan change" step. However, when the ticket is labelled as a security-related change, then the information officer needs to approve it as well.

Once a ticket has all its required approvals, the change manager can plan the change. All changes are planned inside a maintenance window, except for emergency changes which might be executed outside of these windows. A maintenance window is a specific timeframe in which changes are executed. If the change is planned, the executor can execute the change, the submitter later tests the results and based on whether it was successful, the executor performs a rollback on the change or closes the ticket. Hereby the process has ended.

5.1.3 Finance

The greater customer team finance is oriented towards customers who operate in the finance industry. These mostly include banks throughout the Netherlands.

Bank 1

The change management process of the first bank is shown Appendix C (Figure Bank 1). As shown in this figure the process incorporates a multitude of roles; submitter, change approver, executor, tester SBP and final change approver customer. Unlike other banking customers, Schuberg Philis has a relatively more significant role in the IT landscape of the customer. Aside for the responsibility over the IT infrastructure of this customer they also are their IT department.

Table 10. Roles and responsibilities (Bank 1)

Role	Person fulfilling the role	Responsibilities
Submitter	A representative of customer/representative of Schuberg Philis.	Submission of change tickets, initiation of risk/impact analysis, create a rollback plan, create test scenarios.
Change approver	Someone mandated by customer. (customer or Schuberg Philis)	Reviewing and assessing changes.
Executor	An engineer from Schuberg Philis.	Scheduling, executing and potentially rollback changes, option to reject changes.
Tester SBP	Test engineer from Schuberg Philis	Testing the quality of changes and whether everything works according to plan.
Final change approver customer	Someone mandated by customer. (always someone from the customer)	Reviewing and assessing changes (final approval).

The mission-critical change process of this customer starts with the initiation of a change. Changes are initiated by the submitter. This role is fulfilled by either an employee of the customer or the customer team at Schuberg Philis. Once a change is initiated, the change is classified by the submitter, a risk/impact analysis is initiated and when

required, a rollback plan is created. All these components are stored in a ticket. When the submitter is from the customer, then someone from Schuberg Philis must review the ticket and vice versa. If a change gets disapproved, the submitter has the option to refine the change ticket and ask for approval again or close it and thereby end the process. After the approval, the executor can implement the change to the test environment after which it is tested by the tester. If this test fails, the executor will retry the implementation, and when it succeeds the change approver must approve the change again. Upon approval, the executor implements the change to the acceptance environment. From here the change approver performs a user acceptance test after which the executor plans the change for implementation to the production environment. The final change approver reviews all results and decides whether the change can be implemented in the production environment. Once the final change approver approves the change, the executor can implement the change. If this is successful, he/she can communicate this to all stakeholders, update the relevant documentation and close the ticket. If it fails, the executor will perform a rollback and thereby the process ends.

Bank 2

While this customer is a bank as well, the change management process is different than the previous one. This is because this customer forms its approval mechanism around specific applications or services rather than the impact of changes. This is because Schuberg Philis has a smaller role in the IT landscape of this customer and therefore has more parties to work with in comparison to the previous customer. Appendix C (Figure Bank 2) visualises the mission-critical change process of Bank 2. As seen in this figure multiple roles are incorporated in this process, a description of these roles is provided in the table below.

Table 11. Roles and responsibilities (Bank 2)

Role	Person fulfilling the role	Responsibilities
Submitter	A representative of customer/representative of Schuberg Philis.	Submission of change tickets, initiation of risk/impact analysis, create a rollback plan, create test scenarios.
Digital delivery customer	Someone from the digital delivery department at the customer	Reviewing and approving/disapproving changes related to specific components and able to execute changes to these specific components.
Digital orchestration customer	Someone from the digital orchestration department at the customer	Reviewing and approving/disapproving changes related to specific components.
Schuberg Philis	A Mission-critical engineer at Schuberg Philis	Execute and potentially rollback changes, option to reject changes.
Software vendor	Someone mandated by the software vendor.	Reviewing and approving/disapproving changes related to specific components.
Tester customer	Someone from the customer whose job is to test	Testing the results of changes.

The mission-critical change process of this customer starts with the initiation of a change. Changes are initiated by the submitter who is from either the customer side, Schuberg Philis or an external software vendor. The submitter initiates a risk/impact analysis, test scenarios and a rollback plan. Afterwards, the submitter also classifies the change to which components it is related. This information is stored in a ticket which then needs to be approved. The approval mechanism of this customer is relatively extensive and somewhat complicated as the roles who need to approve a change is based on the component or service that needs a change. Some changes need to be approved by the digital delivery, digital orchestration, Schuberg Philis and the software vendor. This means an 8-eye-principle

is applied. In some situations, only digital delivery needs to approve a change. In total, there are sixteen different components or services and seven different combinations of roles that need to approve a change.
If a change gets disapproved, the submitter has the option to refine the change ticket and ask for approval again or close it and thereby end the process. After the approval, the change gets planned. Depending on the component or service, changes are planned in specific maintenance windows.

After a change is planned, it may be executed. If a change was only required to be approved by digital delivery, then digital delivery is also authorised to execute a change. Schuberg Philis executes all other changes. Once a change is executed, a tester from the customer side performs a user acceptance test (UA). Results of this test are communicated with Schuberg Philis and based on whether it was successful or not, a rollback is performed, or the ticket is closed. A rollback leads to the reinitiating of the UAT and upon closure of the ticket, the process is finished.

5.2 Characteristics

The process models of the mission-critical change process of the four selected customers of Schuberg Philis form ground to determine what characterises such a process. The matrix below compares what activities are taken within each process and shows the differences between the processes of these customer teams. These activities are derived from the process models and in some cases generalised. For example, the assessment and approval/disapproval of a change is generalised to approval mechanism.

Table 12. Activity occurrence matrix

Characteristic	Customer			
	Postal service	*Grocery store chain*	*Bank 1*	*Bank 2*
Documents changes (ticket)	X	X	X	X
Initiate a risk/impact analysis	X	X	X	X
Create test scenarios	X	X		X
Create a rollback plan		X	X	X
Classify changes	X	X	X	X
Approval mechanism	X	X	X	X
Plan/schedule changes	X	X	X	X
Execute change	X	X	X	X
Test change	X	X	X	X
Communicate results	X		X	X
Update relevant documentation	X		X	
Close change	X	X	X	X

As shown in the matrix, many of the same activities are incorporated throughout the different change processes. For example, all customer teams document their changes. Interesting to notice is that changes are documented in either Jira or ServiceNow.

The most significant differences occur in the approval mechanism, shown in Table 13. The postal service customer and grocery store chain customer are less strict on how many approvals are required. In some cases, a two-eye-principle is required for the postal service customer. While customer bank 1 always requires a 6-eye-principle and customer bank 2 varies between a 2-eye-principle up to an 8-eye-principle in case of core applications. This variation is based on the impacted application of that specific customer.

Table 13. Approval mechanism matrix

Approval mechanism	Customer			
	Postal service	*Grocery store chain*	*Bank 1*	*Bank 2*
0-eye-principle	X			
2-eye-principle	X	X		X
4-eye-principle	X	X		X
6-eye-principle		X	X	X
8-eye-principle				X

I suspected that this difference in required approvals was because banks usually manage vast amounts of valuable assets and are therefore more impacted when things go wrong. Although information from the interviews for the case study indicate that this is because banks are heavily audited and will lose certain privileges when they do not comply or have full control over these aspects of their IT landscape. Some of the interviewees that worked in the postal service customer had previously worked in a bank customer team and confirmed that there is a substantial difference in strictness. The postal service customer and more customers in the public utility and retail industry are more focussed on deploying changes relatively fast and without too much "hassle" such as approval mechanisms.

In all cases, changes are scheduled. For some applications or services changes are scheduled via a maintenance window. This is a timeframe in where changes can be executed. It also occurs that changes are executed throughout the day whenever the executor wants. This depends on the impact/risk of a change and whether the architecture of the application or service allows it. Low risk and low impact changes are more likely to be executed throughout the day.

After the execution of a change, all customer teams test whether this was successful. For the PUR customers, this is performed by either the submitter or the executor. For the banking customers, a specific role is incorporated in the process which performs all tests.

Communication of results is performed by three out of four customers teams and updating relevant documentation is performed by only two customers teams, although all customer teams close the ticket as a finalising step in the process.

By generalising the multitude of these actions/activities, it becomes possible to notice and determine the characteristics of these processes. Activities within the determined processes are deemed a characteristic when three out of four customer teams incorporate it in their process. Below these characteristics are described and explained;

- **Each change is documented**; changes are registered through a service management tool in the form of a ticket, in the context of this research these tools were three out of four times Atlassian Jira and one out of four times ServiceNow.

- **Risk and impact of a change are determined**; in order to execute a change, a risk/impact analysis is executed. This determined risk and impact form input for both the prioritisation of a change and the workflow that the change will follow. For example, a change with a high impact and high risk is required to be approved by more people or people in a higher position.
- **Rollback plan**: when executing a change, it is possible for that specific change to fail or not give the expected outcomes. Whenever this happens, there is a plan to mitigate this. This rollback plan describes what and how to act when a change fails.
- **Approval mechanisms**: each change needs to be approved by someone else than the submitter. Depending on the risk and impact of a change, this would be one or two people, following the 2-/4-/6-/8-Eye-principle. Interesting to notice is that for the finance customers' requirements for this are stricter as these require a 6-/8-eye-principle.
- **Changes are planned/scheduled**; based on impact, priority, application/service or infrastructure changes are scheduled to what suits the executor best or designated maintenance windows.
- **Changes are tested**; in order to execute a change, test scenarios are to be created, these test scenarios add structure in testing the outcome of the change and whether the change was successful or not.

The list of determined characteristics does not necessarily conclude to be of a mission-critical change process but could rather belong to a standard change process. To determine whether any of these characteristics are related to the mission-criticality of these processes, a survey was shared with the twelve engineers that were also interviewed in the case study. In this survey questions were asked for each found characteristic whether it was a characteristic for a mission-critical change process or could also be part of a standard change process. Seven responses were collected, results are shown in Table 14, and further specification of the results is provided in Appendix F.

Table 14. Survey results (n=7)

Characteristic	Deemed mission-critical?	Result
Each change is documented	Yes: 28.6% / No: 71.4%	NO
Risk and impact of a change is determined	Yes: 28.6% / No: 71.4%	NO
Rollback plan	Yes: 57.1% / No: 42.9%	YES
Approval mechanisms	Yes: 57.1% / No: 42.9%	YES
Changes are planned/scheduled	Yes: 28.6% / No: 71.4%	NO
Changes are tested	Yes: 71.4% / No: 28.6%	YES

Characteristics are deemed mission-critical if the majority (50%+) of respondents agree with this. Results show that three characteristics are deemed mission-critical: rollback plan, approval mechanisms, and the testing of changes.

The majority of respondents indicate that documenting changes is not specific to mission-critical. One respondent stated, "Each change is (should be) documented, mission-critical or not". Results of the risk and impact determination of a change show the same. This is something that is done in all change processes although the general remarks of one respondent state that "It should be the combination of risk and severity that classifies the change to be mission-critical.", which means that the mission-criticality of a change is rather determined by risk and impact analysis instead of the whole process being mission-critical.

Planning for failure is deemed a characteristic of a mission-critical change process by a small majority, although comments on this characteristic do not give a clear explanation for this other than the existence of rollback mechanisms.

The fact that changes must be approved before execution is deemed a characteristic for mission-critical changes. Although one respondent states that "each change needs to be approved by someone else than the submitter,

mission-critical or not". Another respondent states that the multiple eye principle is the most important characteristic of a change but does not specify whether it is part of a mission-critical change process. Given these comments and the small overhand, it could be stressed that a 2- or 4-eye-principle is common and could be seen as a characteristic of a standard change process but that multiple eye principles such as 6 and 8 are deemed mission-critical. Hence, the extensiveness of the approval mechanism determines the mission-criticality of the process rather than the existence of the mechanism.

The planning/scheduling of changes is not seen as a characteristic of a mission-critical change but also occurs in a "normal" change process. The testing of changes is the most agreed-upon characteristic of a mission-critical change process. One respondent stated that mission-critical engineers make use of DTAP environments in which changes are tested under different conditions before they will be deployed to the production environment.

To refer back to and conclude on SRQ2. While the overall structure of the change process is similar between a mission-critical one and a standard one, the mission-critical change process is characterised by extensive preparations, multiple approval gateways and mechanisms that ensure the success of a change or mitigate failure. Mission-critical changes are tested in lower environments (DTAP) and have specific plans created upfront on how to act when a change fails. At last extensive approval mechanisms ranging from a 4-to-8 eye principle ensure the validity of these changes and make sure no mistakes are made.

Chapter 6

How workarounds impact mission-critical change processes

Case study

This chapter further explores what workarounds occur in mission-critical change processes and how these workarounds impact the process. It does so by answering SRQ3; *What types of workarounds occur in the mission-critical change processes by self-steering teams?* and SRQ4; *What is the impact of each workaround executed by self-steering teams on mission-critical change processes?* In order to answer these questions two cases studies were conducted and analysed. These case studies are described in detail in Appendix D and E.

Out of the four customer teams analysed in the previous chapter, two of them were further analysed in the form of a case study. These are the Postal service team and the Bank 1 team. Their mission-critical change processes which were modelled earlier formed a baseline to detect workarounds upon. In total, twelve team members of these teams were interviewed, following the script in Appendix B. From the Postal service team, five members were interviewed and from the Bank 1 team seven. Interviewees were asked how they execute the mission-critical change process and in what ways it is deviated from. These deviations were extracted, and each deviation was listed as a workaround. In total fifteen workarounds were found of which six belonged to the Postal service team and nine to the Bank 1 team. Table 15 provides a short description of these workarounds and in how many interviews they were detected.

Table 15. Identified workarounds

ID	Workaround	Count
Postal service		
WA1.1	A change is executed without a change ticket.	5
WA1.2	A change is executed, but the change ticket and approvals are asked afterwards.	2
WA1.3	Relevant documentation is not updated after a change.	2
WA1.4	Stakeholders are not informed about a change.	2
WA1.5	Approval from the customer was not given via the system (formal), but the executor did this himself after reaching an agreement with the initial approver via phone.	1
WA1.6	Intentionally changing the submitter of a change to ensure that the pool of people able to approve a change is big enough	1
Bank1		
WA2.1	A change is executed without a change ticket (impact too low, the customer does not understand)	2
WA2.2	A change is executed without formal approval from the customer	4
WA2.3	A change is executed while the submitter is the same person as the first approver.	1
WA2.4	Upon failure of a change, a fix forward is executed instead of a rollback	2

WA2.5	Relevant documentation is not updated after executing a change	5
WA2.6	A change is executed, but there was no rollback plan in place	1
WA2.7	A change is executed based on a verbal approval of the customer, and the customer approved the change after execution via the system (Jira)	4
WA2.8	A ticket is created after the change is already executed.	1
WA2.9	Stakeholders are not informed of the execution of a change	1

6.1 Workarounds

Each workaround is stored in a workaround snapshot that provides insights in which role executes the workaround, why the workaround is executed and to what activity in the initial process the workaround corresponds. Table 17 shows a short version of this, but a detailed overview per workaround is presented in Appendix D (Postal service) and E (Bank 1). To increase the comparability of each workaround, the specific motivation was generalised to a more abstract level.

Interesting to notice is the motivation behind each workaround. As displayed in Table 16, four different motivations were identified: The worker acted in best interest of the customer as working around the prescribed activity would result in a more positive outcome, the worker forgot to execute an activity, the worker intentionally avoided to execute an activity or the worker was unaware that an activity was required to execute.

In 12 of the cases, the reason to deviate from the process was either that the worker acted in the best interest of the customer or that the worker forgot to execute an activity (8/4). Specific to the different teams it shows that for the Postal service team three of the workarounds were executed as an act in the best interest of the customer and in two cases an activity was forgotten to be executed. For the banking customer, this was 5 and 2. Especially acting in the best interest of the customer resonates with the self-steering aspect of the teams where team members have the authority to deviate but are also responsible for consequences. Hence, if they consciously decide to deviate, it is reasonable to assume it is in the best interest of the customer as they will face the consequences.

Table 16. Workarounds per summarised motivation

Motivation (summarised)	Workaround ID
Acted in best interest of the customer	*WA1.2, WA1.5, WA1.6, WA2.1, WA2.3, WA2.4, WA2.7, WA2.8*
Forgot to execute activity	*WA1.3, WA1.4, WA2.5, WA2.9*
Intentionally avoided execution of activity	*WA1.1, WA2.6*
Unawareness	*WA2.2*

Table 17 shows the workaround matrix. For both the Postal service and Banking customer, it shows that the worker or role that executes the workaround is in all cases the executor or submitter. The reason behind this is that the people who were interviewed in order to identify these workarounds could only fulfil these roles.

For each workaround in regard to the corresponding activity in the process it comes to notice that the most common parts of the change process to deviate from is the approval mechanism (5 cases) and the documentation of changes in the form of tickets (4 cases). This is interesting as these are the two parts which relatively take the most time to complete. In the case of the approval mechanism, the worker often relies on the other party such as the customer. In case of the Postal service team avoidance of documenting changes was linked to the fact that this would mean they eventually had to go through the approval mechanism, which takes time.

Table 17. Workaround matrix

ID	Worker	Motivation (summarized)	Corresponding activity in the process
Postal service			
WA1.1	Executor	Intentionally avoided execution of activity	Document changes (ticket)
WA1.2	Executor	Acted in the best interest of the customer / forgot to execute activity	Document changes (ticket)
WA1.3	Executor	Forgot to execute activity / Intentionally avoided execution of activity	Update relevant documentation
WA1.4	Executor	Forgot to execute activity	Communicate results
WA1.5	Executor	Acted in best interest of the customer	Approval mechanism
WA1.6	Executor/submitter	Acted in best interest of the customer	Approval mechanism / Document changes (ticket)
Bank 1			
WA2.1	Executor	Acted in best interest of the customer	Documents changes (ticket)
WA2.2	Submitter/executor	Unawareness	Approval mechanism
WA2.3	Submitter/executor	Acted in best interest of the customer	Approval mechanism
WA2.4	Executor	Acted in best interest of the customer	Failure of change (not identified)
WA2.5	Executor	Forgot to execute activity	Update relevant documentation
WA2.6	Submitter/executor	Intentionally avoided execution of activity	Create rollback plan
WA2.7	Executor	Acted in best interest of the customer	Approval mechanism
WA2.8	Executor	Acted in best interest of the customer	Documents changes (ticket)
WA2.9	Executor	Forgot to execute activity	Communicate results

To refer back and conclude on SRQ3. Based on the fifteen workarounds which were identified throughout the two case studies results show that the occurring workarounds executed by self-steering teams in the mission-critical change process are generally executed by the submitter or executor role. Four different motivations were identified; the worker either acted in best interest of the customer, forgot to execute a workaround, intentionally avoided execution of a workaround or was unaware of the activity. In twelve cases, workers were motivated to deviate as they acted in the best interest of the customer or they forgot to execute an activity. While these workarounds correspond to different activities in the original process, most workarounds occurred in regard to the approval mechanism and documentation of changes in the form of tickets.

6.2 Impact

The impact of the identified workarounds is twofold. First, the impact of each workaround on the original process in terms of costs, time, quality, and flexibility. And second, the team's decision on how they decide to act in response to the workarounds (ignore, prevent, adopt, redesign). These actions are listed as impact as this influences the original process.

As shown in Figure 12, out of the 15 workarounds nine have had a positive impact on costs and only four a negative impact on costs. In regard to time dimension, this difference is more prominent as 14 workarounds were deemed to have a positive or significantly positive impact. In only 1 case a negative impact was determined.

For the majority of these workarounds, the impact of flexibility on the process stayed neutral, was positive or significantly positive. In only 1 case a negative impact of flexibility was determined. While, costs, time and flexibility lean towards an overall positive impact, the impact on quality is more spread. In seven cases, a negative or a significant negative in quality was determined, in five the impact was neutral and in three workarounds had a positive impact.

Differences between the two case studies become visible once the graphs are drilled down. Figure 13 shows the numbers for the Postal service team and Figure 14 for the Banking team.

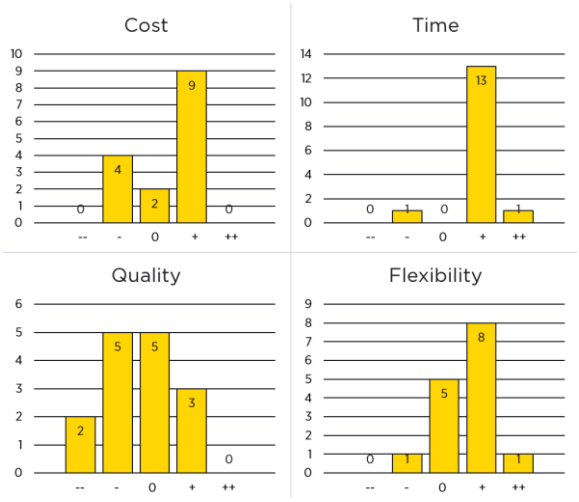


Figure 12. Numbers of impact of all workarounds

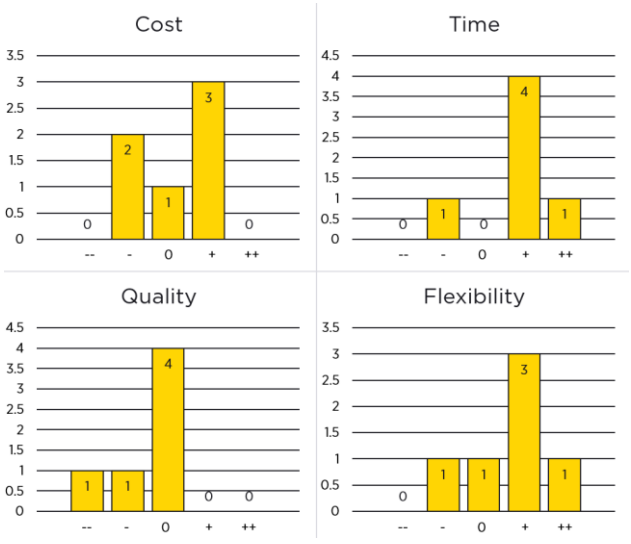


Figure 13. Numbers of impact for the postal service team

The impact of the workarounds on costs are quite similar for both teams. Most workarounds have a positive impact in costs. For both teams in two cases, the workarounds have a negative impact on cost.

In case of the Banking team, a positive impact on time was determined for each workaround. While the majority of workaround of the Postal service team have a positive impact on the time dimension as well, it also occurred twice that a negative impact was determined.

The quality of the process remains neutral in most cases for the postal service team (four). In two cases, a negative impact in quality was determined. While the majority for the workarounds of the banking team have a negative impact in quality (five) as well, there are also workarounds that have a positive impact on the quality (three) of the process.

Comparing the impact of each workaround in term of flexibility, it shows that for the Banking team the flexibility was impacted positively (five) or remained neutral (four). Just as the Banking team the majority of workarounds of the Postal service team had a positive impact in flexibility, although one case was identified which had a negative impact and one which remained neutral.

By looking at what impact per dimension occurs the most, both teams show many similarities. In both teams, there is an overall positive impact in costs and time, but also a positive impact in flexibility. Differences occur in quality; for the postal service team workaround generally have a neutral impact on quality. Although for the banking team, this is split between a negative and a positive impact in quality.

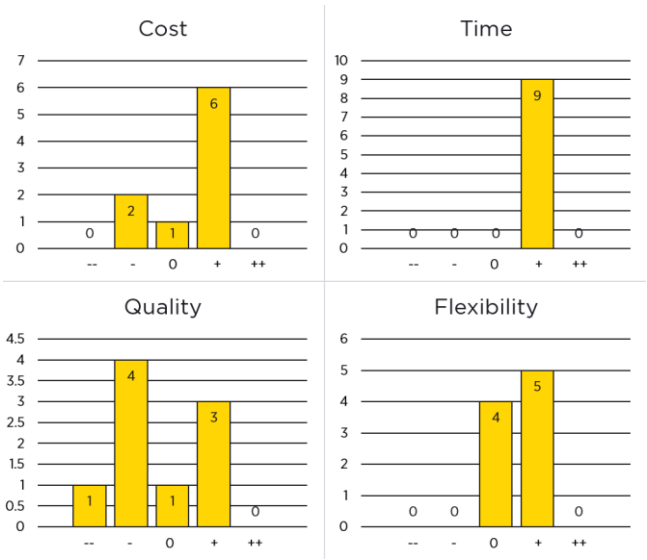


Figure 14. Numbers of impact for the Bank 1 team

The other perspective of impact is that of the actions that are decided to take. This is deemed as a part of impact as these actions are a response to the workaround and impact the original process. While it was expected that most of the identified workarounds would be ignored as team members have the authority to deviate and most of the workarounds were performed in the best interest of the customer, this appears to be not the case. Table 18 shows that only one workaround is ignored and the reasoning behind this was that the participants ought this workaround to occur rarely.

For six of the workaround's participants decide to prevent them. In five cases where it was decided to prevent, the workaround has a negative impact on the quality of the process. In only one case the impact on quality remained neutral. In the six cases where it was chosen to prevent the workaround, five times there was also a positive impact in the time dimension. The motivation behind these workarounds vary but the common factor is that the worker forgot, was unaware or intentionally avoided the execution of an activity.

Workarounds were adopted on three occasions. For each of these three workarounds, a positive impact in time, costs, and flexibility was determined. Quality of the process varied between negative, neutral and positive. However, in all three cases, the worker acted in the best interest of the customer.

A redesign of the original process was determined five times. For all these five workarounds, the workaround has a positive impact in time and costs. In cases where the worker acted in best interest of the customer, this also led to a positive impact. In all other cases, quality remained neutral or had a negative impact. For these workarounds the flexibility either remained neutral or had a positive impact.

Table 18. Workaround actions

Action	Action description	Workaround ID
Ignore	No action in response to the workaround, no effort required.	*WA2.3,*
Prevent	Undertake action to avoid the occurrence of this workaround.	*WA1.2, WA1.3, WA1.6, WA2.2, WA2.5, WA2.6*
Adopt	Accept the workaround and spread it within the team	*WA1.5 WA2.1, WA2.7*
Redesign	Redesign the process.	*WA1.1, WA1.4, WA2.4, WA2.8, WA2.9*

To conclude on SRQ4, workarounds executed by self-steering teams impact mission-critical change processes in different ways. The identified workarounds generally have a positive impact in regard to the dimensions time, costs, and flexibility of the process. In four cases a negative impact in terms of cost was determined, however those four workarounds were decided to prevent. The impact in terms of quality of the process varied the most as in seven cases it was impacted negatively and in eight cases it remained neutral or was impacted positively. Aside from the impact on these four dimensions, only one workaround is ignored, and most workarounds led to a prevention or redesign of the original process.

Chapter 7

Drivers behind workarounds

Case study

This chapter explores what the drivers are behind the occurrence of workarounds executed by self-steering teams in mission-critical change processes and investigates these connections. SRQ5 is formulated as following: *What drives the occurrence of workarounds executed by self-steering teams in mission-critical change processes?*

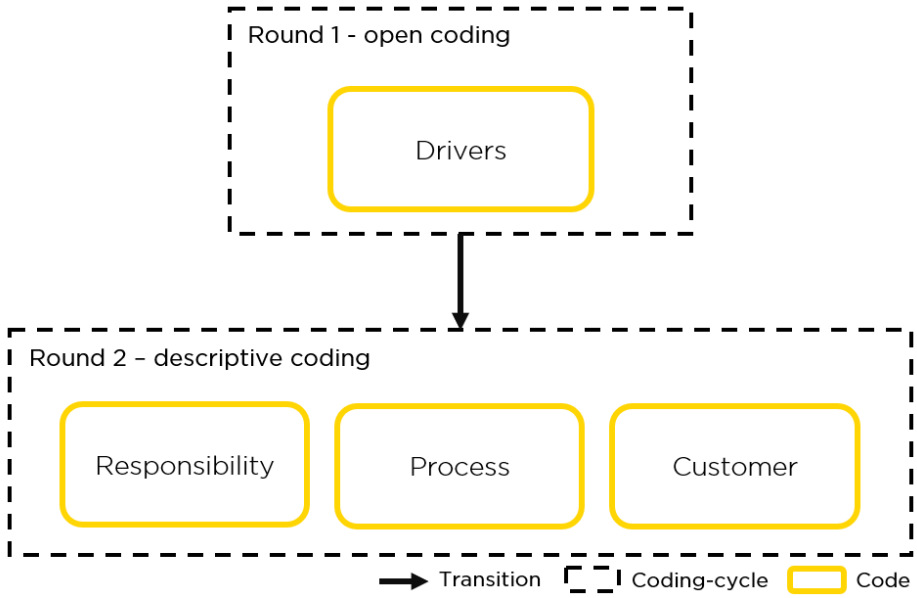


Figure 15. Coding scheme for the drivers

In order to answer RSQ5, specific questions were formulated in the interview protocol. Twelve participants were interviewed after which the interviews were transcribed and coded. Interview transcriptions are coded following a two-cycle coding scheme. Figure 15 visualises this coding scheme. The yellow blocks are codes, the dashed outlines represent each cycle, and the arrow refers to the transition from the first to the second coding cycle in which the second coding cycle is based on quotations coded in the first cycle.
In the first cycle, coding is performed inductive by open coding(Miles et al., 2014). Specific quotes that indicate to be a driver are marked. However, these codes are not specified yet as it is not clear yet what drivers exists. During this cycle 26 quotations are coded. During the second cycle all 26 codes are analysed and descriptive coding (Miles et al., 2014) was used to specify each driver. As a results three different drivers were identified: responsibility, process and customer. Table 19 shows the number of interviewees, codes and examples of these codes.

- **Responsibility** refers to the amount of responsibility workers experience over changes.
- **Process** refers to the characteristics of the change process that impact the occurrence of workarounds
- **Customer** refers to the role of the customer in regard to the process and workarounds

At first four drivers were identified, but the driver “team” was left out as further analysis showed that there was a team component in each of the codes. Next to that, quotations under team differed substantially and had similarities with other codes. Therefore, codes listed under team were split up and listed under a more corresponding code.

Table 19. Examples of coding results for the drivers

Code name	Number of interviewees	Codes	Examples per code
Responsibility	10	16	1) "It makes it easier only, by taking all those ownerships and all those risks, etcetera. It comes with responsibility. Extra responsibility. So, every decision you make. Really make sure your decision isn't a disadvantage for the systems or for your customer." 2) “I think it plays a huge role in terms of responsibility and feeling ownership of a change. It's not just the change. You actually picking up this change, you announce to your team that I'm going to be the one taking responsibility for this change, and, um, I think that makes you more aware about what you're doing. So, you're not doing it automatically. You're not doing it, you know, half.”
Process	4	6	1)“Um, yeah, but I think in a way, because the process is like it's lighter and more. Just to the speed we need at that moment in time and we don't have to wait for planned decision moments” 2) “Our change process is pretty simple and straightforward. It’s still well documented, but it has more flexibility or something like that.”
Customer	3	4	1) “For me, the experience from the finance team is quite different. So, the difference between there is quite big yet. Um, So I do think that *customer name* might benefit from yeah, formalizing a bit more in the change process. But on the other hand, I think it should be something also that's coming from the customer. And yeah, I really don't have that feeling at the moment that they actually care about that.” 2) “it's not from one side of the party that decides to do it differently. There have been cases where I didn't agree with what the customer wanted us to do. Yeah, and we put it up for debate and decided to comply with what they wanted.”

7.1 Responsibility

As was concluded in Chapter 4, self-steering teams manage authority in a decentralised way and decision-making responsibilities are transitioned from traditional management to team members. While this would indicate that team members have the freedom and authority to decide to deviate. Results show that this authority also comes with a certain responsibility, one interviewee stated: "It makes it easier only, by taking all those ownerships and all those risks, etcetera. It comes with responsibility. Extra responsibility. So, every decision you make, really make sure your decision is not a disadvantage for the systems or your customer". Another interviewee stated: "It is easier. But in the end, I think you also feel more responsibility. So deviating is not something I will. Well, it prevents me from deviating because of the responsibility". Next to that several interviewees say that whenever they decide to deviate, they would ask their colleagues for approval to do so. In practice this responsibility prevents team members from working around the process as they are also responsible for the consequences. This is also confirmed when looking at the provided motivation of the identified workarounds as many of these workarounds were motivated by an act in the best interest of the customer. This makes sense as these team members do have the authority and decisive responsibility to deviate, but they will, in most cases, only do so if it benefits the customer.

7.2 Process

Another driver is the design of the mission-critical change process. How the process is designed and what activities and gateways are required to take or pass influences to what degree team members can work around the process. As seen in the process models and confirmed by an interviewee, the change processes in the two cases are generally lightweight and does not incorporate a lot of activities or bureaucracy. One interviewee stated: “but I think in a way, because the process is lighter and more. Just to the speed we need at that moment in time and we don't have to wait for planned decision moments”. Another interviewee compared the process to one of his previous job experiences which had a more traditional hierarchy; he stated: “our change process is pretty simple and straightforward. It's still well documented, but it has more flexibility or something like that. In a previous job, you had this really strict process, and you absolutely cannot deviate from it, even if you would want to do something really fast and want to turn it into a for example, emergency change, Still, you cannot do that emergency change right there on that spot because you need to get all kinds of other people involved.”. This is different from the self-steering team as team members have more freedom and are allowed to make these decisions themselves. However, the elimination of activities and bureaucracy might explain why many of the identified workarounds were motivated by the worker forgetting to execute an activity as there is a reduction of control points.

7.3 Customer

The last driver is the role of the customer. As shown in Chapter 5, Indication of this appeared in the comparison of the process models between the customer of greater customer team finance and public, utility, and retail. There are differences in the extensiveness of these mission-critical change processes as specific aspects and characteristics of these customers impacted the design of the process. For example, the fact that banking customers are heavily audited results in more approval mechanisms in the process. Some of the interviewees had previously worked in teams of other greater customer teams and two of them who worked for the Postal service team had previously worked for a banking team. He stated: “For me, the experience from the finance team is quite different. So the difference between there is quite big yet. I do think that *Postal service team* might benefit from formalizing a bit more in the change process. But on the other hand, I think it should be something that is also coming from the customer. And yeah, I really don't have that feeling at the moment that they actually care about that”. These aspects and characteristics of these customer influences how the process is designed and how people treat the process.

To conclude on RSQ5, based on the analysis of twelve interviews, three drivers behind the occurrence of workarounds executed by self-steering teams in mission-critical change processes were identified. These drivers are:

1) The responsibility that team members of self-steering teams have that eventually prevents them from deviating as they are responsible for the consequences. If they decide to deviate it is debated with colleagues and in majority of the cases an act in the best interest of the customer.
2) The design of the original process, mission-critical change processes of self-steering teams are generally lightweight and therefor incorporate less activities and approval mechanism which influences the number of workarounds that can occur.
3) The role and characteristics of the customer as these influences how the process is shaped and how people treat the process.

Chapter 8

Conclusion & future research

The research at hand explores what the impact of workarounds executed by self-steering teams is on mission-critical processes. In order to do so a systematic literature, document analysis and two case studies were executed. The systematic literature review shines a light on how self-steering teams manage authority and decision making, the document analysis provides insight in mission-critical processes, what characterises these processes and sets a baseline to detect workarounds upon. Two case studies were conducted within Schuberg Philis for a postal service customer team and a banking customer team, within these case studies workarounds were identified, sorted in the workaround snapshot approach and its impact determined. At last, the drivers behind the occurrence of workarounds executed by self-steering teams on mission-critical change processes were investigated.

8.1 Conclusion

As stated in chapter 3.1, the research is specified around the main research question, which is formulated as follows: ***How do workarounds of self-steering teams impact mission-critical change processes?*** To structure this, five sub-research questions were formulated and answered below.

1. **How do self-steering teams manage authority and decision making?**
   Self-steering teams manage authority in a decentralised way, and employees are given the responsibility to decide how they want to shape most aspects of their work. While this is a general characterisation of self-steering teams, it depends on the organisation or team and to what degree this is practiced. For example, decisive responsibilities on firm strategy are less likely to be decentralised while work execution if more likely to be decentralised. While authority is decentralised, it does not necessarily mean it is equalised as there is more room for natural leadership. This is strengthened by the lack of hierarchical structure and often elimination of a manager role which requires team members to possess self-management and self-leadership skills. Decision-making is influenced by this decentralisation of authority and managed local and autonomous as decision-making responsibilities are transitioned from management to team members.

2. **What characterises a mission-critical change process?**
   Analysis of four mission-critical change processes has led to the identification of six characteristics. These characteristics are not necessarily related to the mission-criticality of the process but are common for both a mission-critical change process and a standard change process. 1) each change is documented in form of a ticket, 2) the risk and impact of a change is determined beforehand, 3) there is a plan for when a change fails, 4) changes are approved via multiple-eye-principle, 5) changes are planned and 6) changes are tested before they are implemented. To conclude which of these characteristics are mission-critical seven experts were surveyed. Results show that a mission-critical change process is characterised by the fact that there is a plan for failure, changes are approved via a multiple-eye-principle and changes are tested before they are implemented.

3. **What types of workarounds occur in the mission-critical change processes by self-steering teams?**
   Based on the identification of fifteen workarounds, results show that workarounds executed by self-steering teams in the mission-critical change processes are generally executed by the submitter or executor role. Overall, workers deviate from the process when they are motivated by an act in the best interest of the customer or when they forget to execute specific activities. Only on limited occasions they intentionally avoid execution of an activity or are unaware of it. For majority of the workarounds, deviation occurs in regard to the approval mechanism or documenting changes in form of change tickets.

4. **What is the impact of each workaround executed by self-steering teams on mission-critical change processes?**
   The impact of each workaround executed by self-steering teams on mission-critical change processes is determined in terms of costs, time, quality and flexibility and how the team wants to act in response to each workaround. Results indicate that the identified workarounds have a positive impact in regard to the time, costs, and flexibility of the process. The quality dimension was the most divergent as in seven cases workarounds had a negative impact in terms of quality and in eight it remained neutral or had a positive impact. Actions taken in response to these workarounds show promising results as only one workaround was deemed to be ignored. Most workarounds led to the prevention of a workaround. In some cases, these workarounds were adopted and in the other cases it led to a redesign of the original process. This strengthens the belief that workarounds can form input for process optimization.

5. **What drives the occurrence of workarounds executed by self-steering teams in mission-critical change processes?**
   The occurrence of workaround executed by-self steering teams in mission-critical change process is driven by three drivers: 1) The responsibility that team members of self-steering teams have that eventually prevents them from deviating as they are responsible for the consequences. If they decide to deviate it is debated with colleagues and in majority of the workarounds it is an act in the best interest of the customer. 2) The design of the original process, mission-critical change processes of self-steering teams are generally lightweight and therefor incorporate less activities and approval mechanism which influences the number of workarounds that can occur. 3) The role and characteristics of the customer as these influences how the process is shaped and how people treat the process.

As each sub research question is answered, a conclusion can be drawn on the main research question; *How do workarounds of self-steering teams impact mission-critical change processes?*

Results show that the workarounds of self-steering teams impact mission-critical change processes in multiple ways. In most cases, these workarounds relate to the administrative part of the process or the approval mechanisms. The majority of workarounds have a positive impact on time, costs, and flexibility of the process. The impact on quality varies as in some cases it decreases, remains neutral or increases. Even though the impact in terms of time, costs, quality, and flexibility show both positive and negative results, only one workaround was ignored. When workarounds have a negative impact on quality, they tend to be prevented. Team members of self-steering teams have the authority to deviate. Given this authority, interviewees experience responsibility as well. It appears that because of this responsibility, it prevents these team members, in the majority of the workarounds, only tend to consciously deviate when it is in the best interest of the customer. In other workarounds, team members forget to execute activities, but rarely team members consciously avoid or skip activities to undermine the quality of the process. From the same perspective of motivation, whenever workarounds are motivated by an act in the best interest of the customer, workarounds tend to be adopted or lead to a redesign of the process. For those workarounds, it also shows that the quality of the process generally increases or remains neutral.

8.2 Future research

As this research takes a broad take on workarounds and studies them in a specific context with specific teams, it opens multiple paths for future research. Below we describe some of these paths for future research.

- *In-depth extension of the case study.* Interviews were held with solely mission-critical engineers to identify workarounds. Future research could extend this perspective and focus on interviewing different roles within the process or gather the perspective of the customer. This could be done for both the identification of workarounds as the determination of their impact.
- *Workaround identification by process mining.* An important aspect which was discovered throughout the research is that within the mission-critical change process ITSM tool (e.g. Jira, ServiceNow) are used to track changes in the form of tickets but also to support the approval mechanism. Within this research project, the source of evidence for workaround identification is interviewing. Future research could focus on ITSM tools that incorporate the mission-critical change processes through process mining techniques (Van Der Aalst et al., 2012), and thereby identify and possibly compare workarounds.
- *Changing the case study context.* Subject to the case studies were the two customer teams within Schuberg Philis, and within these case studies, the mission-critical change process was investigated. By changing the case study context, it becomes possible to compare results to other types of companies or processes. For example, future research of this kind could mean identifying workarounds in a standard change process and see how they compare. An alternative option is the identification of workarounds within mission-critical change processes in an organisation that is not deemed self-steering but hierarchical.

Chapter 9

Discussion

In this section, we discuss the results, threats to validity, and practical and scientific implications. This research and specifically the two conducted case studies, have led to the identification of fifteen workarounds. Before interviewing both teams, it was expected that more workarounds would be found in the Postal service team as the Bank 1 team is more bound to rules and audits and teams from GCT finance are generally stricter on following the process in comparison to teams in GCT Public, Utility and Retail. Information gathered from interviewees who had worked in both GCTs before confirmed that customers from GCT finance are stricter and even went as far to say that GCT PUR customers “do not really care much for the procedure”. Although the opposite is the case, my explanation for this would be the extensiveness of each process. While the general structure of the process remains the same, the process of the bank 1 team incorporates more activities and gateways. This also means there are also more activities to deviate from. Also, within the banking customer team, two more people were interviewed compared to the postal service team.

Aside from the number of workarounds identified, the number of times it was identified reflects the initial expectation more closely. The most counted workaround in the Postal service team is that of executing changes without a change ticket (WA1.1) while at the Banking team this was not updating the relevant documentation after a change (WA2.5). Relevant documentation after a change is not assessed during audits or used nor seen by the customer as it is for internal use. However, change tickets are. Not creating a ticket for a change means it is not traceable and is a risk for that specific component or service. The lack of detection of such workarounds could be explained by the pressure of rules and audits in the Banking team compared to the Postal service team.

The impact of these workarounds shows interesting results, the effects vary, and do not necessarily have to be beneficial for the process but could be disadvantageous as well (Beerepoot, van de Weerd, & Reijers, 2019). This was also found for the identified workarounds as some workarounds had an overall positive impact, which therefore appeared beneficial. Although some seemed detrimental as well. However, the workarounds which cause a disadvantage were often related to transparency, which is coined by Beerepoot, van de Weerd, & Reijers (2019). In the context of this research, the loss of transparency translated to a loss in auditability which specifically for finance customers, is essential. We found this particularly true in workarounds where changes were executed without tickets as they become untraceable or auditable. Nadhrah and Michell (2013) state that often in high-pressure environments workarounds can help to complete processes in less time (Nadhrah & Michell, 2013). This is also confirmed by our research as in most cases execution of these workarounds results in a positive impact in time. In all cases where a certain pressure was high (e.g. time impairment), it was also detected that the workaround led to a positive impact of time spent on the process.

By looking at what actions are taken in response to these workarounds, results show that the majority of workarounds are to be prevented, adopted or lead to a redesign of the process and only one workaround was

ignored (*WA2.3*). This supports and strengthens the belief that workarounds can form input for process improvement (Azad & King, 2012; Cresswell et al., 2017; Lalley & Malloch, 2010; Petrides et al., 2004).

As stated before, time is often of the essence, and there is limited room for failure. In this research, the majority of workarounds were motivated by an act in the best interest of the customer or that the worker forgot an activity. However, for most cases where the worker acted for the better of the customer, the workaround was adapted to make this workaround possible, whether it was through adoption or redesign. In healthcare, similar situations occur as workarounds are generally accepted if it contributes to the quality of patient care (Beerepoot, Ouali, Van de Weerd, & Reijers, 2019)

Results gathered throughout the research are interesting and have led to several opportunities for process improvement. However, it is still unclear what the influence of the self-steering aspect of these teams or the mission-criticality of the process is on these workarounds. While drivers were identified, it cannot be concluded that for example, there are fewer workarounds in mission-critical environments or when these processes are executed by self-steering teams. This is because no comparison was conducted between workarounds executed by non-self-steering teams or in a non-mission-critical change process.

The workaround snapshot approach which was used to determine the impact of each workaround and what action to take in response showed to be a useful tool to reflect on each workaround from different perspectives. Usage of this approach, as stated before, has led to opportunities for improvement in fourteen out of the fifteen cases. This supports the statement that "the approach enables organisations to make well-informed decisions" (Beerepoot & Van De Weerd, 2018). Hereby we believe that as a result of this we not only contribute to the research field as we have validated the workaround snapshot approach in a before unexplored industry and strengthened the belief that workarounds are input for process optimisation through various means. We have also contributed to the industry and actively shown that a reflection on the mission-critical change process through the eyes of workarounds leads to opportunities for process improvements in mission-critical change processes. We show that self-steering teams that execute mission-critical changes can learn from reflecting on their mission-critical change process. How these teams interact with this process is a direct demonstration of potential disadvantageous restrictions since workers often act in the best interest of the customer and these workarounds indicate an overall positive impact.

9.1 Limitations

Throughout the research project, it became evident that it has several limitations and threats to the validity of the research. Just like workarounds, these limitations do not necessarily have to be a bad thing but could be seen as opportunities for future research.

First of all, the case study company; Schuberg Philis serves several customers in different industries and segments, for this research two cases were conducted in a Postal service team and banking team. This number of cases could be deemed low as the mission-critical change process is executed in other industries and segments as well. Therefore, results are limited to these specific industries and context. Next to that, in total 12 interviews were held which could also be deemed as low as for one case only five people were interviewed. Since data collection solely relies on these interviews it is hard to claim completeness of results.

Second, for impact determination the devil's quadrangle was used. Measurement of each aspect is done by rating it in contrast to the original process. However, this rate is not quantified in any way. For example, a positive impact on time means less time is spent on the process, although it is not specified how much time this is. It could be 1 minute or 1 hour. As this is not specified participants rate each measure based on their own interpretation of it. This has influence on how they determine the impact but also on what action they deem to take in response to each workaround.

Threats to validity

Several threats to the validity of the research and its results were identified, some of which are derived from the limitations that became clear throughout the research.

Internal validity

The common denominator throughout the research project is the case study company: Schuberg Philis. Even though different customer teams were selected who operate separately and in different industries, this does not alter the fact that these teams still share the same cultural values or beliefs of the organisation under which they operate. This inevitably introduces bias. Next to that, within both cases participants that were interviewed all had an engineering background and were limited to fulfil the role of submitter or executor of a change. While they play a significant role in the mission-critical process, the role of the customer is just as significant. Their perspective has been left out as no customers were interviewed. This might result in a single-sided perspective on, for example, the impact of a workaround or what action should be undertaken in response to a workaround.

External validity

The external validity refers to which extend results could be generalised. For the case study, two cases were conducted at a Postal service team and a banking team. As both the scope of service delivery and design of the change process is tailored to these specific customers it becomes difficult to generalize the results to for example other teams within the organisation or outside the organisation. A similar effect occurs for the generalisability to the level of self-steering teams as the organisation, it's culture and way of working causes bias.

Construct validity

Measuring the impact of each workaround is performed by a focus group session. How impact is divided in four different measures: time, costs, quality, and flexibility. Two threats occur here, the first is construct confounding as the score of these measures might influence each other. Impact on costs and time could influence how the quality is measured. The second threat is regarding how these measures are understood. While time and costs are concepts that are relatively clear, quality and flexibility are more open to interpretation. For example, how is quality impacted, is there a positive impact when the customer is overall happier or when there is more freedom in execution.

Reliability

Within the case study most of the data collection methods rely on the usage of expert interviews. While conduction of these interviews was done with utmost care, the fact that the interviewer had experience working at the case study company could negatively impact the reliability as his knowledge, or assumptions might crease a researcher bias. On the other hand, and something that has been stated before, people generally do not like to talk about how they avoid the standard and agreed upon procedure as it might impact them in a negative way. Interviewees could've provided dishonest answers during the interviews to for example why they deviate in certain cases. While it was tried to mitigate this by asking different variants of the same questions, a respondent bias cannot be excluded.

Bibliography

Adriansyah, A., van Dongen, B. F., & van der Aalst, W. (2011). Towards Robust Conformance Checking. In *Lecture Notes in Business Information Processing* (Vol. 66 LNBIP, pp. 232–237). Spinger. https://doi.org/10.1007/978-3-642-20511-8_23

Alper, S., Tjosvold, D., & Law, K. S. (1998). Interdependence and Controversy in Group Decision Making: Antecedents to Effective Self-Managing Teams. *Organizational Behavior and Human Decision Processes*, *74*(1), 33–52. https://doi.org/10.1006/obhd.1998.2748

Alter, S. (2014). Theory of Workarounds. *Communications of the Association for Information Systems*, *34*(55), 1–30. Retrieved from http://repository.usfca.edu/at

Alter, S. (2015). A Workaround Design System for Anticipating, Designing, and/or Preventing Workarounds. In *EMMSAD 2015* (Vol. and/or Pr). Retrieved from http://repository.usfca.edu/at

Arraj, V. (2013). *ITIL: the basics (white paper)*. The APM Group and The Stationery Office 2013. Retrieved from https://www.mountainview-itsm.com/itil-training/downloads/axelos/8863-WP-ITIL-the-basics.pdf

Azad, B., & King, N. (2008). Enacting computer workaround practices within a medication dispensing system. *European Journal of Information Systems*, *17*(3), 264–278. https://doi.org/10.1057/ejis.2008.14

Azad, B., & King, N. (2012). Institutionalized computer workaround practices in a Mediterranean country: An examination of two organizations. *European Journal of Information Systems*, *21*(4), 358–372. https://doi.org/10.1057/ejis.2011.48

Beerepoot, I., Ouali, A., Van de Weerd, I., & Reijers, H. A. (2019). Working around health information systems: to accept or not to accept? In *Twenty-Seventh European Conference on Information Systems*.

Beerepoot, I., & Van De Weerd, I. (2018). Prevent, redesign, adopt or ignore: Improving healthcare using knowledge of workarounds. *Ecis 2018*.

Beerepoot, I., van de Weerd, I., & Reijers, H. A. (2019). The Potential of Workarounds for Improving Processes. In *Business Process Management Workshops. BPM 2019* (pp. 338–350). https://doi.org/10.1007/978-3-030-37453-2_28

Bisbal, J., Lawless, D., Wu, B., & Grimson, J. (1999). Legacy Information Systems: Issues and Directions. *Software, IEEE*, *16*(5), 103–111.

Budgen, D., & Brereton, P. (2006). Performing systematic literature reviews in software engineering. In *Proceeding of the 28th international conference on Software engineering - ICSE '06* (p. 1051). https://doi.org/10.1145/1134285.1134500

Cater-steel, A., Toleman, M., & Tan, W.-G. (2006). Transforming IT Service Management – the ITIL Impact. *Proceedings of the 17th Australasian Conference on Information Systems*, *6*(August 2014), 1–11. https://doi.org/http://www.ribabookshops.com/uploads/b1e09aa7-c021-e684-a548-b3091db16d03.pdf

Ciccozzi, F., Crnkovic, I., Di Ruscio, D., Malavolta, I., Pelliccione, P., & Spalazzese, R. (2017). Model-Driven Engineering for Mission-Critical IoT Systems. *IEEE Software*, *34*(1), 46–53. https://doi.org/10.1109/MS.2017.1

Conger, S., Winniford, M., & Erickson-Harris, L. (2008). Service Management in Operations. In *Americas Conference on Information Systems*. Retrieved from http://aisel.aisnet.org/amcis2008http://aisel.aisnet.org/amcis2008/362

Cresswell, K. M., Mozaffar, H., Lee, L., Williams, R., & Sheikh, A. (2017). Workarounds to hospital electronic prescribing systems: A qualitative study in English hospitals. *BMJ Quality and Safety*, *26*(7), 542–551. https://doi.org/10.1136/bmjqs-2015-005149

Davenport, T. H. (1993). Process Innovation: Reengineering Work Through Information Technology. *Academy of Management Perspectives*, *7*(2), 100–103. https://doi.org/10.5465/AME.1993.9411302338

de Morais, R. M., Kazan, S., de Pádua, S. I. D., & Costa, A. L. (2014). An analysis of BPM lifecycles: From a literature review to a framework proposal. *Business Process Management Journal*, *20*(3), 412–432. https://doi.org/10.1108/BPMJ-03-2013-0035

Druskat, V. U., & Wheeler, J. V. (2004). How to lead self-managing teams. *MIT Sloan Management Review*.

Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018). *Fundamentals of business process management: Second Edition*. *Fundamentals of Business Process Management: Second Edition*. https://doi.org/10.1007/978-3-662-56509-4

Evans, N. D. (2013). The new scope of mission-critical computing | CIO. Retrieved June 3, 2019, from https://www.cio.com/article/2474862/the-new-scope-of-mission-critical-computing.html

Ferneley, E. H., & Sobreperez, P. (2006). Resist, comply or workaround? An examination of different facets of user engagement with information systems. *European Journal of Information Systems*, *15*(4), 345–356. https://doi.org/10.1057/palgrave.ejis.3000629

Gacek, C., Giese, H., & Hadar, E. (2008). Friends or foes? - A conceptual analysis of self-adaptation and IT change management. In *Proceedings - International Conference on Software Engineering* (pp. 129–136).

Giarte. (2018). *Outsourcing Performance Jaarboek 2018 - Purpose*. Giarte.

Gray, J., & Siewiorek, D. P. (1991). High-Availability Computer Systems. *Computer*, *24*(9), 39–48. https://doi.org/10.1109/2.84898

Gross, S., Malinova, M., & Mendling, J. (2019). Navigating Through the Maze of Business Process Change Methods. In *Proceedings of the 52nd Hawaii International Conference on System Sciences*. https://doi.org/10.24251/hicss.2019.754

Halbesleben, J. R. B., Wakefield, D. S., & Wakefield, B. J. (2008). Work-arounds in health care settings: Literature review and research agenda. *Health Care Manage Rev*, (March), 2–12.

Hammer, M. (1990). Reengineering Work: Don't Automate, Oblirate. *Harvard Business Review*, 104–112.

Hart, C. (2018). *Doing a literature review: releasing the social science research imagination*. SAGE Publications, Inc. Retrieved from http://link.springer.com/10.1007/BF00253969

Hinchey, M., & Coyle, L. (2010). Evolving critical systems: A research agenda for computer-based systems. In *17th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems, ECBS 2010* (pp. 430–435). https://doi.org/10.1109/ECBS.2010.56

Houy, C., Fettke, P., & Loos, P. (2010). Empirical research in business process management - analysis of an emerging field of research. *Business Process Management Journal*, *16*(4), 619–661. https://doi.org/10.1108/14637151011065946

Kitchenham, B. (2004). *Procedures for Performing Systematic Literature Reviews. Keele University Technical Report TR/SE-0401*. Retrieved from http://www.it.hiof.no/~haraldh/misc/2016-08-22-smat/Kitchenham-Systematic-Review-2004.pdf

Kuipers, B. S., & De Witte, M. C. (2005). The control structure of team-based organizations: A diagnostic model for empowerment. *Economic and Industrial Democracy*. https://doi.org/10.1177/0143831X05057504

Lalley, C., & Malloch, K. (2010). Workarounds: The hidden pathway to excellence. *Nurse Leader*, *8*(4), 29–32. https://doi.org/10.1016/j.mnl.2010.05.009

Lee, M. Y., & Edmondson, A. C. (2017). Self-managing organizations: Exploring the limits of less-hierarchical organizing. *Research in Organizational Behavior*, *37*, 35–58. https://doi.org/10.1016/j.riob.2017.10.002

Liden, R. C., Wayne, S. J., & Bradway, L. K. (1997). Task Interdependence as a Moderator of the Relation Between Group Control and Performance. *Human Relations*, *50*(2). https://doi.org/10.1023/A:1016921920501

Liu, J., Wang, H., Hui, C., & Lee, C. (2012). Psychological Ownership: How Having Control Matters. *Journal of Management Studies*, *49*(5), 869–895. https://doi.org/10.1111/j.1467-6486.2011.01028.x

Manz, C. C., & Neck, C. P. (1995). Teamthink: Beyond the groupthink syndrome in self-managing work teams. *Journal*

of Managerial Psychology, *10*, 7–15. Retrieved from https://www-emerald-com.proxy.library.uu.nl/insight/content/doi/10.1108/02683949510075155/full/pdf?title=teamthink-beyond-the-groupthink-syndrome-in-selfmanaging-workteams

McLaughlin, K. A., & Damiano, F. (2007). American ITIL. In *Proceedings ACM SIGUCCS User Services Conference* (pp. 251–254). https://doi.org/10.1145/1294046.1294106

Miles, M. B., Huberman, A. M., & Saldana, J. (2014). *Qualitative Data Analytics: a methods sourcebook* (Third). California: SAGE Publications, Inc.

Moher, D., Liberati, A., Tetzlaff, J., & Altman, D. G. (2009). Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA Statement. *Annals of Internal Medicine*, *151*(4), 264–269. https://doi.org/10.1371/journal.pmed1000097

Molleman, E. (2000). Modalities of self-managing teams the "must, "may, "can" and "will" of local decision making. *International Journal of Operations and Production Management*, *20*(8), 889–910. https://doi.org/10.1108/01443570010332935

Moorhead, G., Neck, C. P., & West, M. S. (1998). The tendency toward defective decision making within self-managing teams: The relevance of groupthink for the 21st century. *Organizational Behavior and Human Decision Processes*, *73*(2–3), 327–351. https://doi.org/10.1006/obhd.1998.2765

Muehlen, M. zur, & Ho, D. T.-Y. (2006). Risk Management in the BPM Lifecycle. In *LNCS* (Vol. 3812, pp. 454–466). https://doi.org/10.1007/11678564_42

Nadhrah, N., & Michell, V. (2013). A normative method to analyse workarounds in a healthcare environment: Motivations, consequences, and constraints. In *ICISO 2013 - Proceedings of the 14th International Conference on Informatics and Semiotics in Organisations, IFIP WG8.1 Working Conference* (pp. 195–204). Retrieved from http://centaur.reading.ac.uk/31983/www.reading.ac.uk/centaur

Neck, C. P., & Manz, C. C. (1994). From Groupthink to Teamthink: Toward the Creation of Constructive Thought Patterns in Self-Managing Work Teams. *Human Relations*, *47*(8), 929–952. https://doi.org/10.1177/001872679404700804

Netjes, M., Reijers, H. A., & Van Der Aalst, W. (2006). Supporting the BPM life-cycle with FileNet. In *CEUR Workshop Proceedings* (Vol. 364, pp. 135–146). Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.142.6750&rep=rep1&type=pdf

Object Management Group. (2011). *Business Process Model and Notation (BPMN)*. Retrieved from http://www.omg.org/spec/BPMN/20100501

Park, R. (2012). Cognitive and affective approaches to employee participation: Integration of the two approaches. *Journal of World Business*, *47*(3), 450–458. https://doi.org/10.1016/j.jwb.2011.05.011

Paula Fraga-Lamas. (2017). Enabling Technologies and Cyber-Physical Systems for Mission-Critical Scenarios. *PhD Thesis*. https://doi.org/http://hdl.handle.net/2183/19143

Petrides, L. A., McClelland, S. I., & Nodine, T. R. (2004). Costs and benefits of the workaround: Inventive solution or costly alternative. *International Journal of Educational Management*, *18*(2), 100–108. https://doi.org/10.1108/09513540410522234

Ponsard, C., Massonet, P., Molderez, J. F., Rifaut, A., Van Lamsweerde, A., & Tran Van, H. (2007). Early verification and validation of mission critical systems. In *Formal Methods in System Design* (Vol. 30, pp. 233–247). https://doi.org/10.1007/s10703-006-0028-8

Rabiee, F. (2004). Focus-group interview and data analysis. *Proceedings of the Nutrition Society*, *63*(4), 655–660. https://doi.org/10.1079/pns2004399

Rathert, C., Williams, E. S., Lawrence, E. R., & Halbesleben, J. R. B. (2012). Emotional exhaustion and workarounds in acute care: Cross sectional tests of a theoretical framework. *International Journal of Nursing Studies*, *49*(8), 969–977. https://doi.org/10.1016/j.ijnurstu.2012.02.011

Safadi, H., & Faraj, S. (2010). The role of workarounds during an opensource electronic medical record system implementation. In *ICIS 2010 Proceedings - Thirty First International Conference on Information Systems*. Retrieved from http://aisel.aisnet.org/icis2010_submissions/47

Sagie, A., & Aycan, Z. (2003). A cross-cultural analysis of participative decision-making in organizations. *Human*

Relations, *56*(4), 453–473. https://doi.org/10.1177/0018726703056004003

van de Weerd, I., & Brinkkemper, S. (2008). Meta Modeling for Situational Analysis and Design Methods. In *Handbook of Research on Modern Systems Analysis and Design Technologies and Applications* (pp. 35–54). https://doi.org/10.4018/978-1-59904-887-1.ch003

van der Aalst, W. (2004). Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management. *LNCS 3098 - Lectures on Concurrency and Petri Nets*. Retrieved from https://link-springer-com.proxy.library.uu.nl/content/pdf/10.1007%2Fb98282.pdf

van der Aalst, W. (2010). Process-Aware Information Systems: Lessons to Be Learned from Process Mining. In *Transactions on Petri Nets and Other Models of Concurrency IV* (Vol. 6550, pp. 1–26). https://doi.org/10.1007/978-3-642-18222-8

Van Der Aalst, W., Adriansyah, A., De Medeiros, A. K. A., Arcieri, F., Baier, T., Blickle, T., … Wynn, M. (2012). Process mining manifesto. In *Lecture Notes in Business Information Processing* (Vol. 99 LNBIP, pp. 169–194). https://doi.org/10.1007/978-3-642-28108-2_19

van der Aalst, W., ter Hofstede, A. H. M., & Weske, M. (2003). Business Process Management: A Survey (pp. 1–12). https://doi.org/10.1007/3-540-44895-0_1

van der Vegt, G. S., Bunderson, S., & Kuipers, B. (2010). Why turnover matters in self-managing work teams: Learning, social integration, and task flexibility. *Journal of Management*, *36*(5), 1168–1191. https://doi.org/10.1177/0149206309344117

van Riemsdijk, M. B., Jonker, C. M., Rens, T., & Wang, Z. (2012). *Negotiation Game for Improving Decision Making in Self-managing Teams*. (W. Ding, H. Jiang, M. Ali, & M. Li, Eds.), *Modern Advances in Intelligent Systems and Tools* (Vol. 431). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-30732-4

Vergidis, K., Tiwari, A., & Maieed, B. (2008). Business Process Analysis and Optimization: Beyond Reengineering. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, *38*(1), 69–82. https://doi.org/10.1109/TSMCC.2007.905812

Weske, M. (2007). *Business Process Management: Concepts, Languages, Architectures*. Springer.

Yarberry, W. A. (2007). Effective change management: Ensuring alignment of it and business functions. *Information Systems Security*, *16*(2), 80–89. https://doi.org/10.1080/10658980601144899

Yin, R. K. (2014). *Case study research: Design and methods*. (V. Knight & J. Young, Eds.) (Fifth edit). California: SAGE Publications, Inc.

Appendices

A Consent form

Consent Form

By completing this form, you give us permission to use this interview for academic purposes and potential publications. Thank you for helping.

Introduction

Research Title:	Mission-critical Workarounds: How workarounds impact the mission-critical IT change process of self-steering organisations.
Researcher:	Nick A. Gies
Research period:	June 2019 – February 2020
Objective:	Based on this interview, and many more, we hope to identify specific workarounds in relation to the defined change process. These workarounds will provide us insight to what degree the execution of the change process differs from the process defined beforehand.
Risks of participation:	We foresee no risks of participation. Although the participant is always, during the research period, free to withdraw from participating without giving any reason.
Data protection:	Your personal data will only be used within the context of this research and will not be shared with any other parties that are not already involved (Schuberg Philis & University Utrecht). Any personal data will be anonymised in further (potential) publications.

Personal information

Full name	
Team	
Function	
Telephone	
Email	

Agreement

I have read and understand the conditions of participating in this research. I understand that my participation is voluntary and that I am free to withdraw at any time without giving any reason. I declare the above to be true and agree to participate by signing this consent form.

Signature _____ Date _____

B Interview questions

Interview questions used within the case study protocol to identify workarounds and the motivation behind them. These questions add structure, but depending on the answers, further questions could be asked.

Introduction

Dear [name], first of all thank you for your cooperation by accepting to be interviewed. I will start with some introductory questions to get a better understanding of both your role at Schuberg Philis and experience in IT.

- What is your role at Schuberg Philis?
    - How long have you been working at Schuberg Philis?
    - For how long have you been in this role?
    - How long have you been working in the IT industry?

- In what customer teams are you a member?
    - What is your role in each team?

Process

The following questions are regarding the mission-critical change process, specifically the one that your specific team uses. We see changes as any additions, modifications, or removals of anything that could have an effect on IT systems or services.

- Do you execute mission-critical changes?
    - How often do you execute mission-critical changes?
    - When do you execute mission-critical changes?

- What has been the most mission-critical change that you executed?

- Could you describe the last change that you executed?
    - What happened?
    - Did it go as planned?
        - If not, how did you deal with it?

- Could you name the activities that you perform while executing a change?
    - Which of these actions are essential?

- Do you think that within a customer team all members follow the same process for executing mission-critical changes?
    - If no, does that mean that everyone is executing a mission-critical change differently?

- Are you familiar with the definition of a mission-critical change process?
    - If so, could you explain what the characteristics or steps of this mission-critical processes are within your team?

- What systems or applications are involved when executing a mission-critical change?
    - Are these systems/applications always involved?

- Are there actions in this process that are specific to the customer that you work for?
    - If so, what are these aspects?

Workarounds

The next questions are about workarounds; workarounds are deviations regarding how a process was designed and intended to be executed. Note that these workarounds are not necessarily a bad thing but could rather be seen as opportunities to optimise processes.

- When you execute a mission-critical change do you always follow the same steps?

- Are there situations where deviate from the process?
    - If so, could you tell me in what way?

- Have you thought of a way to make the execution of a change go faster, more efficient or easier?
    - If so, are these things shared amongst colleagues?

- Do you always register changes?
    - If so, is this always done in the same software system?

Characteristics

These questions help to structure the interview for when the interviewee does not talk a lot or is not directly willing to give specific answers.

- Do you update the relevant documentation after finishing a change?

- Do you execute a risk/impact analysis before executing a change?

- Do you create test scenarios before executing a change?
    - If so, do you execute these test scenarios after a change?

- Do you inform all stakeholders after executing a change?

- Have you ever avoided or skipped the approval mechanism?

Motivations

I'm curious to the reasoning behind these workarounds, why did you execute them, what was it necessary to do this, do you feel that this was better for the process?

- Were you aware that you deviate from the process?

    - If so, is it necessary to deviate from the process?

- Given your experience, what were the situations where you felt obliged to deviate from the mission-critical change process?

- Why did you deviate from the process?

- What would be your evaluation of that decision to deviate?
    - Negative, positive, neutral?
    - In terms of time, flexibility, costs, quality?

Self-steering

Self-steering teams are those who manage authority in a decentralised and systematic way. Decisions are made local and autonomous as decision-making responsibilities are transitioned from management to team members. Teams of such nature formulate their own work objectives, method, planning, and distribution of work.

- Does the freedom that comes with being in a self-steering team make it easier for you to deviate from the set process?

- Do you think there is a connection between being in a self-steering team and how you execute mission-critical changes?
    - If so, could you explain this connection?

- Do you feel you have enough authority to decide to deviate from the standard process?
    - If so, how come do you like you have this authority?

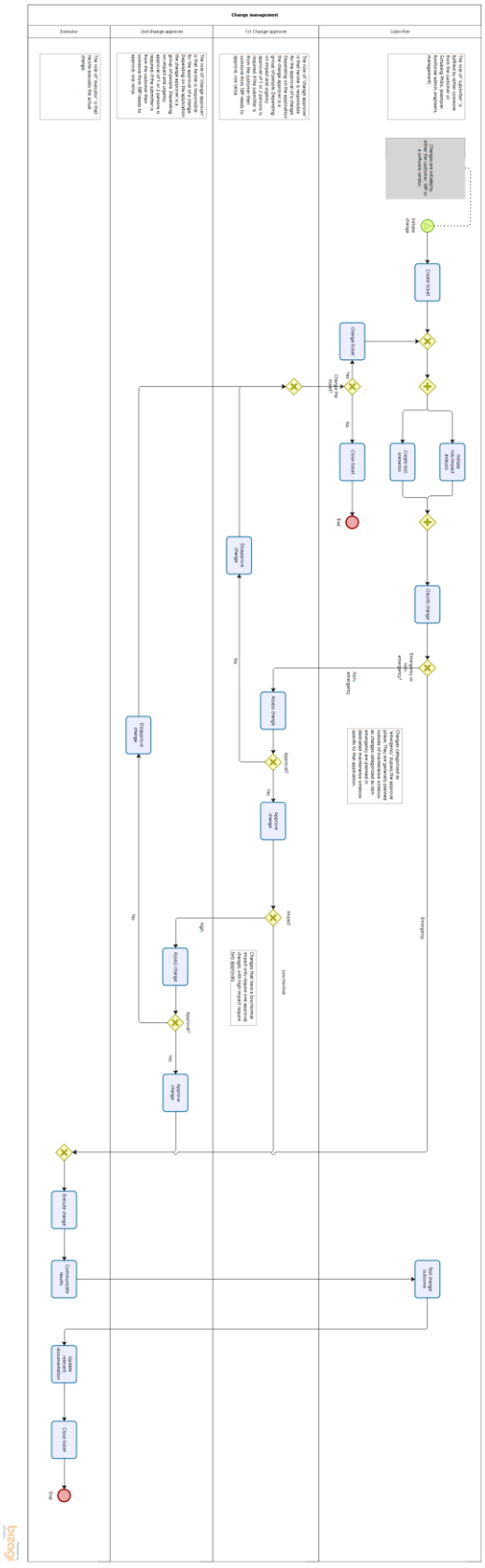
Wrap up

- What else would you like to share?

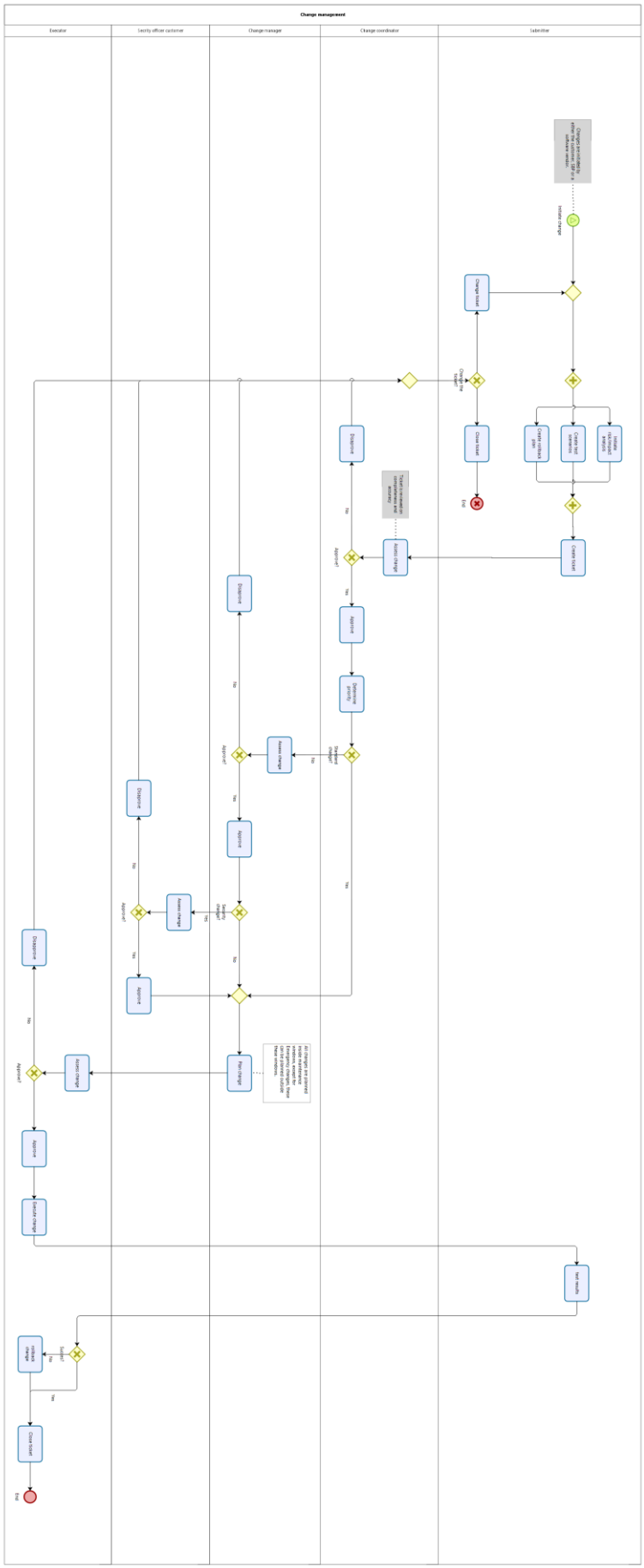
- Any last remarks?

C Baseline processes

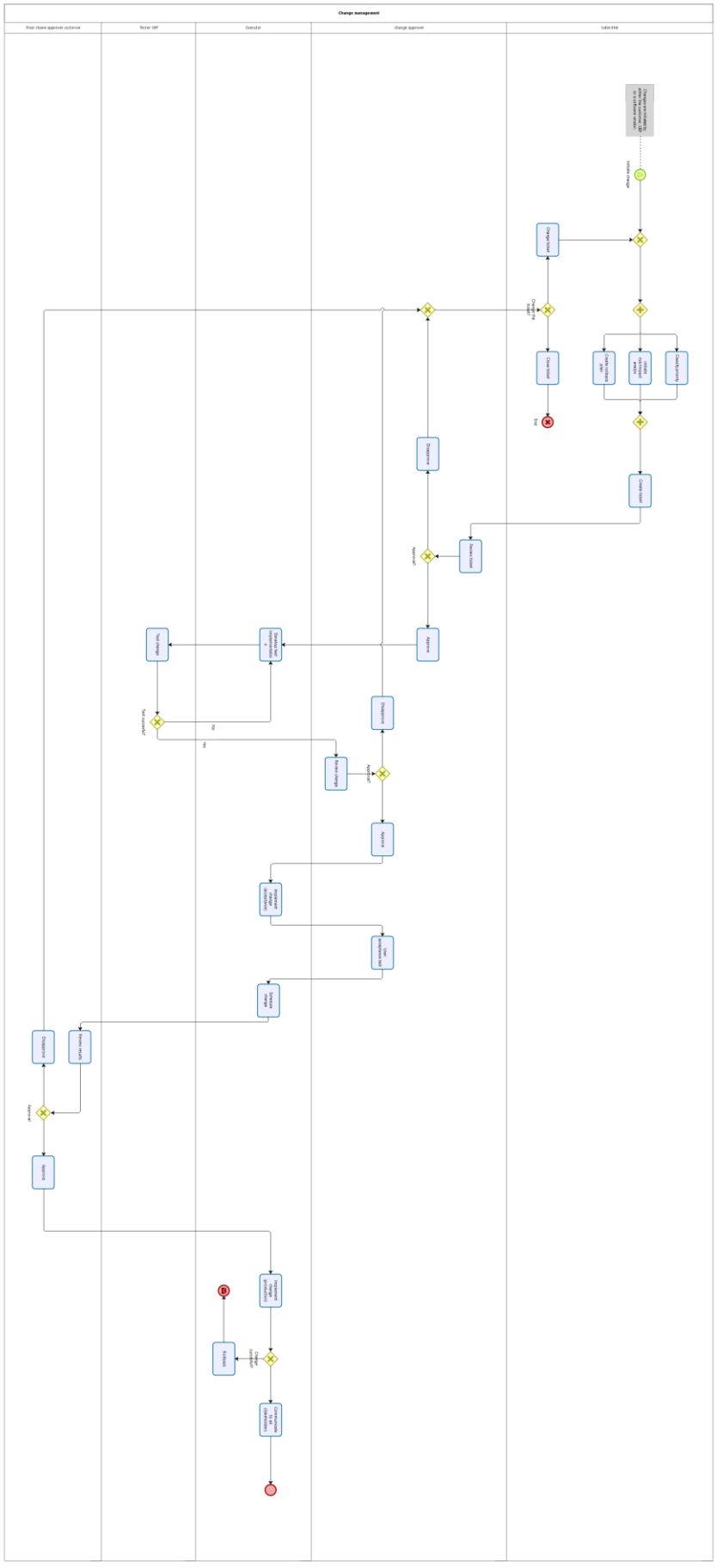
Postal service



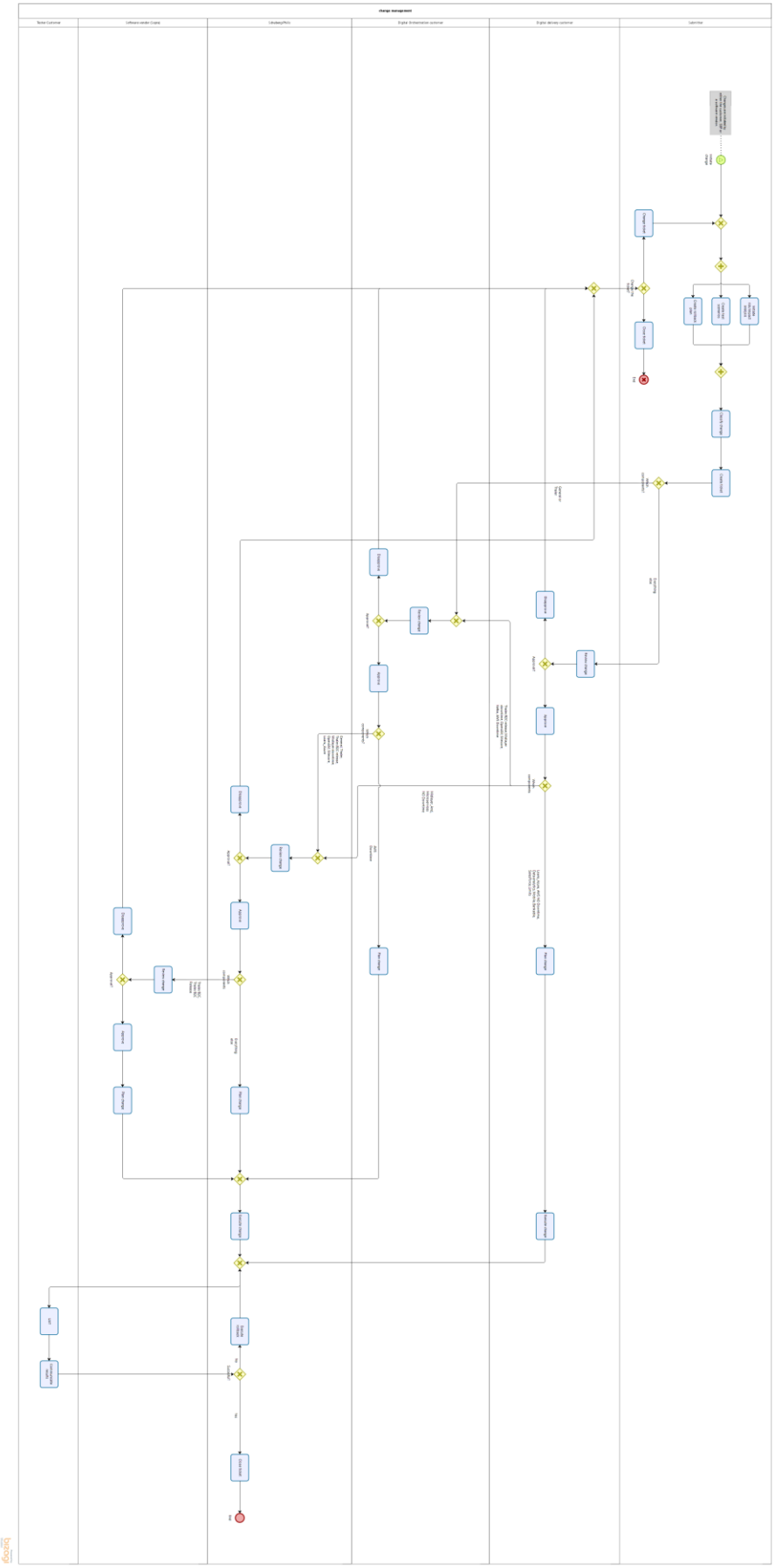
Grocery store chain



Bank 1



Bank 2



D Case study report: Postal service

Introduction

This case study report is written concerning research conducted by N.A. Gies on *How workarounds of self-steering teams impact mission-critical change processes?* A more detailed description of the cause for this research is provided in chapter 1 of the original document.

As stated in the research approach of the original document (chapter 3) the mission-critical change process of four customer teams of Schuberg Philis is modelled following the BPMN2.0 notation. For two of those teams (Postal service customer and a banking customer), the mission-critical change process is further investigated in the form of a case study.

Within this case study, several interviews are conducted following the interview script provided in Appendix B. These interviews were held with multiple mission-critical engineers and aim to determine how these people deviate from the designed mission-critical process. In those interviews, I also tried to identify whether there is a connection between the self-steering aspect of these teams and how they execute the mission-critical change process in regard to workarounds.

Workarounds identified throughout these interviews are shaped via the workaround snapshot approach and visualised via a process model according to BPMN2.0. These workaround snapshots later form input for the focus group session in which all interviewees and a COM of the customer team are involved. During this focus group session, the workarounds are presented, and the team helps determine what the impact of each workaround is based on the devil's quadrangle (time, costs, flexibility and quality). Once the impact has been determined, the team can decide what action they want to undertake in response to each workaround. To structure this, the workaround action matrix is used.

This document is structured in four chapters; the introduction, context, workarounds and impact. Chapter context provides more information about the specific customer team; how many people are involved, what role do they fulfil and what is the scope of business with this customer. Chapter workarounds describe all identified workarounds in detail. Chapter impact shines a light on the focus group session in which the devil's quadrangle is used and lays out the decision-making process and what decisions are taken in response to the identified workarounds.

Context

The case study report at hand is that of the postal service customer. This team consists of sixteen members fulfilling four different roles. The role distribution is described in the table below.

Table 1 roles in team

Role	FTE
Customer operations manager (COM)	1.7
Sales director	0.6
Mission-critical software engineer (MCSE)	2.5
Mission-critical engineer (MCE)	11

The role of COM is a rather broad one as their responsibilities range from pre-sales to operations and maintaining the customer relationship but also team dynamics. They are generally seen as the first point of contact for whenever the customer has questions that do not involve operational casualties.

The sales director is a dedicated sales representative and is traditionally seen as an account manager.

An MCSE is, in essence, a developer, specific to this customer is that Schuberg Philis develops certain plugins and applications for their IT landscape and these are created and maintained by MCSEs from the team

MSE's are responsible for all operations of the applications and systems within the scope of the project and customer. They generally maintain the customers (cloud) infrastructure. This role also is a key stakeholder when executing changes.

In the case of this customer, the scope of delivery is focussed around integrations. This specific customer has a lot of different applications and services in their IT landscape which need to integrate in order to work. Within this IT landscape of the customer, Schuberg Philis is a relatively big partner and are responsible for most of the technical application management of for example the ordering processes of the customer and all applications that make sure packages are packed and delivered. A common characteristic of these services and applications is that the customer is heavily dependent on them if failures occur it will not be possible for this customer to continue operations and is likely to have a financial impact.

Workarounds

Within this specific customer team, seven mission-critical engineers were interviewed. These interviews have led to the identification of 6 workarounds. The table below provides a short description of these workarounds and how many times they were detected.

Table 2 identified workarounds

ID	Workaround	Count
WA1.1	A change is executed without a change ticket.	5
WA1.2	A change is executed, but the change ticket and approvals are asked afterwards.	2
WA1.3	Relevant documentation is not updated after a change.	2
WA1.4	Stakeholders are not informed about a change.	2
WA1.5	Approval from the customer was not given via the system (formal), but the executor did this himself after reaching an agreement with the initial approver via phone.	1
WA1.6	Intentionally changing the submitter of a change to ensure that the pool of people able to approve a change is big enough	1

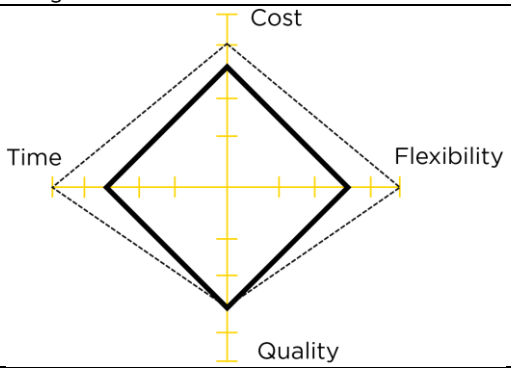
In the tables below, a snapshot is created for all identified workarounds; this snapshot incorporates the workers, description, process model, motivation, impact and action of the workaround.

Referring back to the research method, the impact determination of each workaround is performed via a focus group session with the team members who were interviewed in the previous phase. In this session, four people attended including the researcher. All other people had the role of a mission-critical engineer of which one was the team lead.

During this session, the team was asked to score the impact of each workaround in regard to the measurements in the devil's quadrangle. These decisions were made based on the other aspects of the workaround snapshot which were already in place. For each measurement in each workaround, the team gives a -, --, 0, + or ++. A – on costs means that the workaround leads to a reduction in costs. A + on quality means that the workaround results in an increased quality of the process. As multiple people were involved in this session the final score for each measurement and workaround was a team decision on which each individual gave consent.

Based on the determined impact attendees were then asked to decide how they want to act on each workaround. To structure this, the workaround action matrix was used. Attendees could choose between "adopt", "reengineer", "prevent", and "ignore".

WA1.1

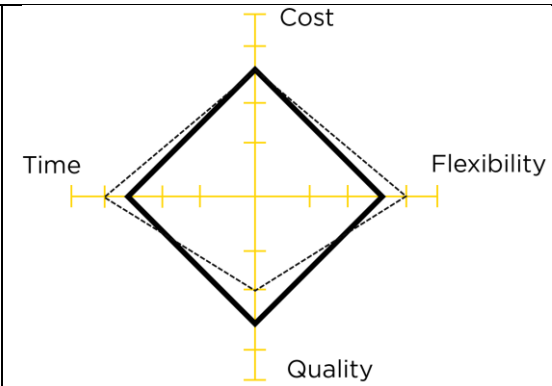
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	A change with very low impact and risk, the executor deployed this to test, acceptance and eventually production. But did not create a ticket whatsoever.
Process model	See **page 86**
Motivation	The executor deemed the impact and risk of this change very low and found that creating a ticket, explaining it to the customer, etc would take more time than the actual change. Therefore he decided to do as stated an illegal change.
Impact	
Action	Redesign

The team determined that this workaround would result in a significant decrease in time spent as creating the ticket, waiting for approval and explaining it to the customer take much time and are not required in this workaround. This also translates to a decrease in costs, which is realised by the reduction of time spent. They found that quality is the same as these changes generally are very low impact and risk. The flexibility however increased significantly as there is more freedom in the process.

The team decided to redesign the process. They understand that these “illegal changes” are not wanted but are keen on altering the process to allow certain components or risk/impact levels to be excluded from approval so that it is documented without losing speed.

WA1.2

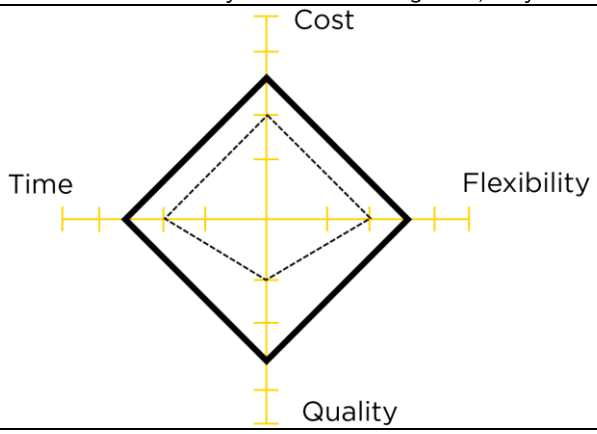
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	A change was required to be executed although there was a time impairment, and the executor was in a hurry. For example, during the night. The executor decided to act first and do the administration later.
Process model	See **page 86**
Motivation	The executor forgot to create the ticket afterwards, although he stated that in 99% of the cases, they do create a ticket afterwards.

Impact	
Action	Prevent

The team determined a neutral effect on costs, pager duty is part of the job, so no extra costs are made there. There is a decrease in time spent as no ticket was required to be created. Flexibility increases because it gives the executor more space to operate in. Regarding quality, they stated that it decreased. This kind of changes are usually ad hoc, and during times when people usually sleep so people are not at their best and prone to make mistakes.

The team decided to prevent this from happening. Although to a certain degree, they do not want to bother the engineer with administrative work during the night so to them it is acceptable to create the ticket the next morning. This is because there is no added value to create it at that specific time as the approver will not be awake and will probably see it the next morning.

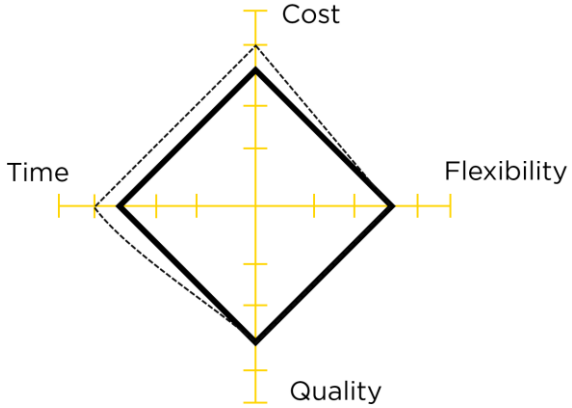
WA1.3

Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	Infrastructure maintained by SBP was being modified but relevant documentation was not updated after the change.
Process model	See **page 86**
Motivation	While the executor says this is often forgotten, they are also not keen on doing this.
Impact	
Action	Prevent

Both time and costs increase as it might spare some time as you do not have to create documentation, although it will increase the time spent when new people on board the team. This also translates to an increase in cost as it costs more to get people up and running. Flexibility is decreased as decisions regarding the system or service that is incorporated is harder to understand or even find.

The team chose to prevent this from happening. Although they stated that the problem with documentation is that the moment you write it, it is already outdated. But having no documentation at all will make it hard for people to get started on projects or join the team.

WA1.4

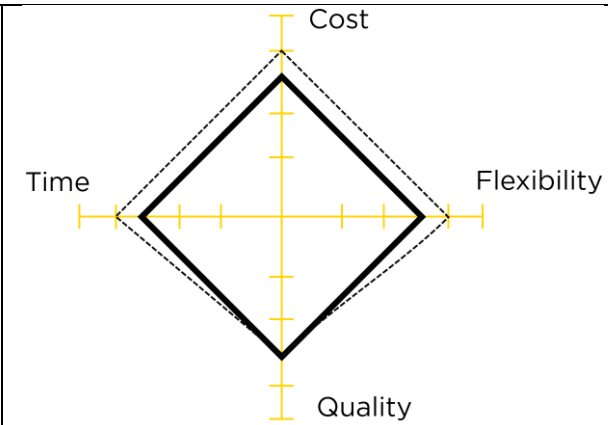
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	Once a change is set for execution the executor informs all stakeholders about the execution in form of a maintenance message. In a certain situation this message was not send out and stakeholders were not informed about the change.
Process model	See **page 86**
Motivation	The executor was unaware that this was not done yet and forgot to do this.
Impact	
Action	Redesign

This one was interesting as the team stated that this is not their job to do. Apparently, It is agreed upon with the customer that they inform all stakeholders regarding changes and not the executor of the change. The impact of this is only visible on costs and time as both decrease since they don't have to spend any time on it and therefore fewer costs.

However, the team did decide to redesign the process so that it is in line with agreements made with the customer.

WA1.5

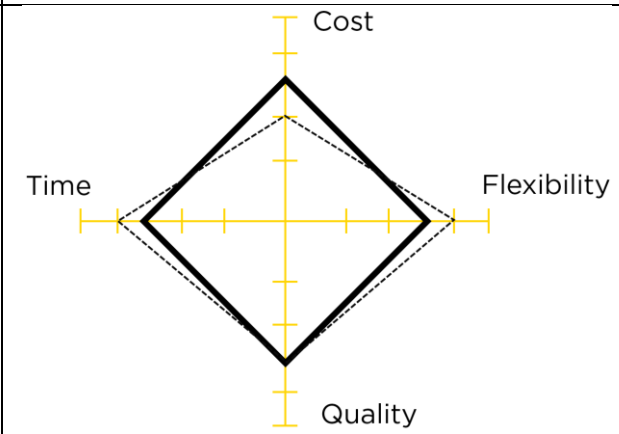
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	The executor wants to execute a change although the change does not have the required approval from the customer yet. At that time it was also not possible for the required approver to approve the change. Instead the executor contacted the approver directly and asked for verbal approval and approved the change himself.
Process model	See **page 87**
Motivation	The executor wanted to execute the change and deemed calling the customer for a verbal approval was also deemed enough for the change.

Impact	
Action	Adopt

A neutral impact on quality was determined as the approval was there; the team does not care whether this is verbal or formal via the system. Flexibility increases as the executor have more options to choose from at that time. Both time and costs decrease as you do not have to wait for formal approval anymore.

The team decided to adopt this workaround and when it is required a verbal approval would suit the approval mechanism and would be enough for the executor to proceed the change

WA1.6

Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor/submitter
Description	Before initiating a change both the executor and customer have contact and decide it is better to let the SBP engineer create the ticket as the opposite party must approve the change. This is because the approval group of SBP is much smaller than that of the customer.
Process model	See page 87
Motivation	This will speed up the initial process as the chance that you have to wait for approval is minimized.
Impact	
Action	Prevent

Time decreases as it will speed up the process, but costs increase as you would have to contact the customer relatively more times. Flexibility also increases, but the quality is neutral.

The team decided to prevent this workaround from happening because it is not scalable.

Types of workarounds

Table 3 compares the workarounds based on the worker, motivation, the action taken and corresponding activity in the initial process. Based on this, it becomes possible to see certain relations.

First to notice is the specification of the worker in each workaround. In each workaround that was identified the worker was the executor of the process and in one workaround it was a shared responsibility with the submitter. This might show relevance, but the group of interviewees in this case study were only able to fulfil the role of executor or submitter as other roles are fulfilled from the customer side.

Interesting to notice is the motivation behind each workaround. When summarised to a more abstract level, it shows that in 50% of the workarounds, the worker acted in the best interest of the customer and therefore deviated from the process. In 33% of the cases, the worker forgot to execute an activity and in the other cases, the executor willingly skipped the execution of an activity.

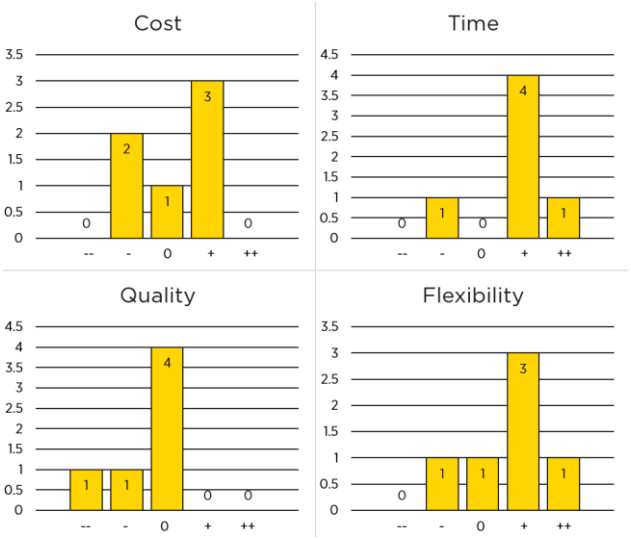
At last, each workaround was connected to a corresponding activity in the initial process to identify any relations. In 66% of the cases, the workaround was related to either documenting changes in the form of tickets or the approval mechanism. This is interesting as the approval mechanism is a part of the process which takes relatively the most time to progress. Even in the cases that it was regarding documenting changes further explanation during the focus group session led to believe that this workaround would reduce the time spent as the executor would not have to wait for the approval mechanism.

Table 3 comparison of different aspects of the identified workarounds

ID	Worker	Motivation (sumarized)	Action taken	Corresponding activity
WA1.1	Executor	Intentionally avoided execution of activity	Redesign	Document changes (ticket)
WA1.2	Executor	Acted in the best interest of the customer / forgot to execute activity	Prevent	Document changes (ticket)
WA1.3	Executor	Forgot to execute activity / Intentionally avoided execution of activity	Prevent	Update relevant documentation
WA1.4	Executor	Forgot to execute activity	Redesign	Communicate results
WA1.5	Executor	Acted in the best interest of the customer	Adopt	Approval mechanism
WA1.6	Executor/submitter	Acted in the best interest of the customer	Prevent	Approval mechanism / Document changes (ticket)

Impact

The impact of the six around workarounds in terms of cost, time, quality and flexibility show interesting results. In three cases costs was impacted positively, twice negatively, and it remained neutral once. In five cases a positive or significant positive impact was determined for the time dimension. While costs and time are generally impacted positive with these workarounds, the quality never increased but instead remains neutral or decreases. On the other hand, the flexibility of the process is impacted positively in four cases.



While only six workarounds were identified within the process, it comes to notice that for no workaround the team decided to ignore it. In 50% of the cases, it was decided to prevent the workaround as an occurrence of it was not beneficial or scalable. For most of these workarounds, it showed a decrease of time spent although the team did state that it might cost less time on a short term. It would increase their workload long term.

Adoption of the workaround was only chosen once. This was because when the context requires it, it does not matter to the team if approval is formalised in the system or whether it was given verbally through the phone. The formal administration was not a priority if it meant that it would slow the process in case of a time impairment.

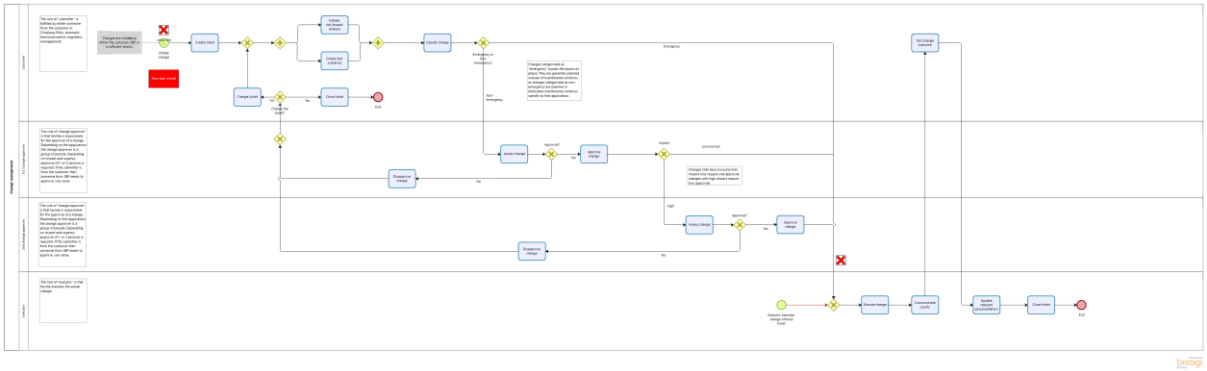
It is also interesting to notice that the workarounds which were decided to lead to a redesign of the initial process were not really influenced by the impact in terms of costs, time, flexibility or quality as these points showed only positive signs. The reason for redesign was rather influenced by the opportunity to automate it or making the process more lightweight.

Table 4 decision in response to each workaround

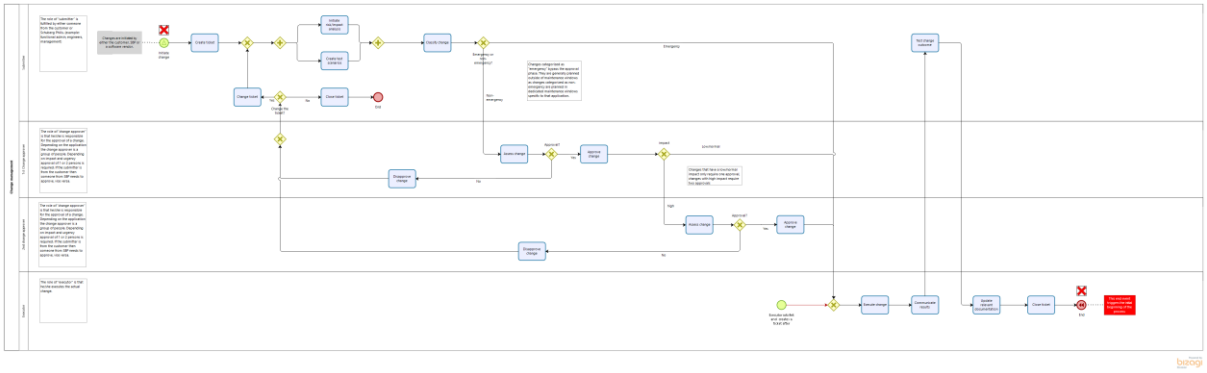
Action	Action description	Workaround ID
Ignore	No action in response to the workaround, no effort required.	
Prevent	Undertake action to avoid the occurrence of this workaround.	WA1.2, WA1.3, WA1.6
Adopt	Accept the workaround and spread it within the team	WA1.5
Redesign	Redesign the process.	WA1.1, WA1.4

Process models

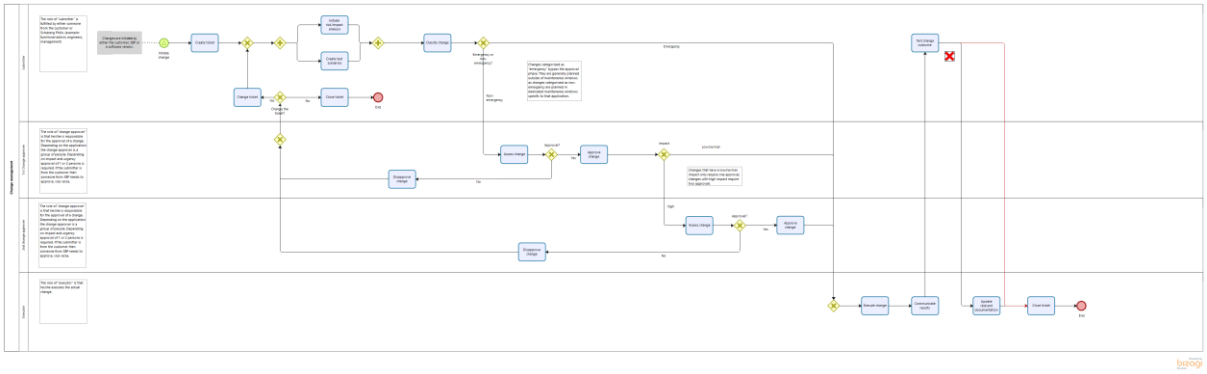
WA1.1



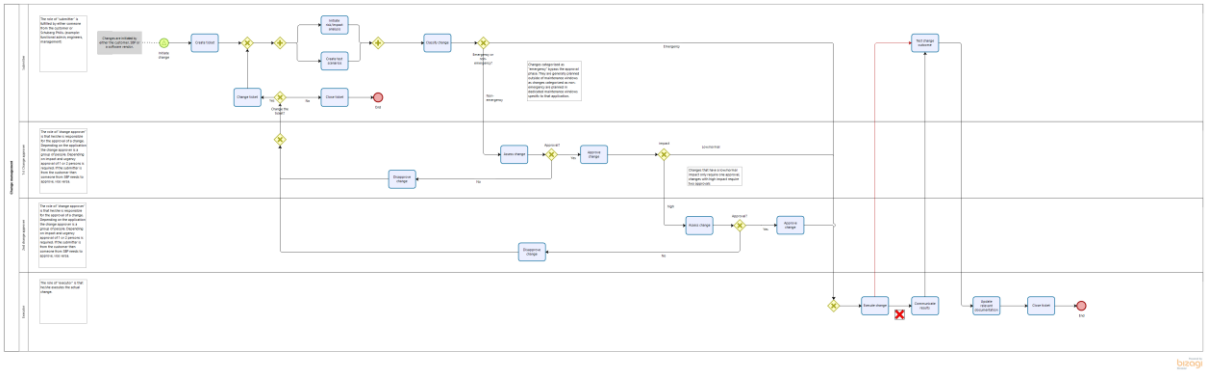
WA1.2



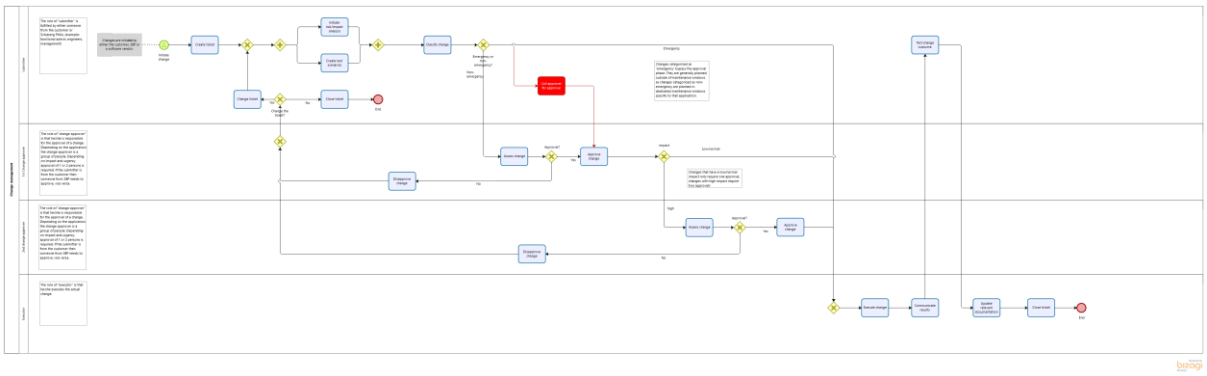
WA1.3



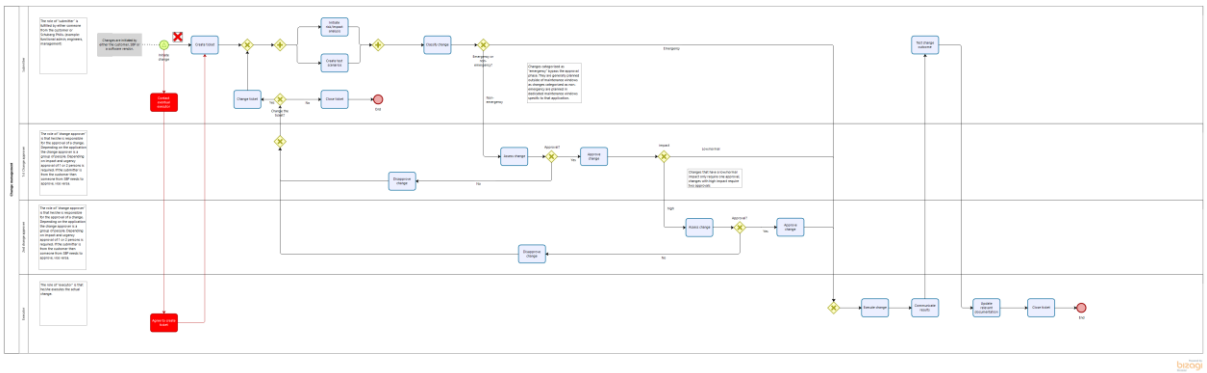
WA1.4



WA1.5



WA1.6



E Case study report: Bank

Introduction

This case study report is written concerning research conducted by N.A. Gies on *How workarounds of self-steering teams impact mission-critical change processes?* A more detailed description of the cause for this research is provided in chapter 1 of the original document.

As stated in the research approach of the original document (chapter 3) the mission-critical change process of four customer teams of Schuberg Philis is modelled following the BPMN2.0 notation. For two of those teams (Postal service customer and a banking customer), the mission-critical change process is further investigated in the form of a case study.

Within this case study, several interviews are conducted following the interview script provided in Appendix B. These interviews were held with multiple mission-critical engineers and aim to determine how these people deviate from the designed mission-critical process. In those interviews, I also tried to identify whether there is a connection between the self-steering aspect of these teams and how they execute the mission-critical change process in regard to workarounds.

Workarounds identified throughout these interviews are shaped via the workaround snapshot approach and visualised via a process model according to BPMN2.0. These workaround snapshots later form input for the focus group session in which all interviewees and a COM of the customer team are involved. During this focus group session, the workarounds are presented, and the team helps determine what the impact of each workaround is based on the devil's quadrangle (time, costs, flexibility and quality). Once the impact has been determined, the team can decide what action they want to undertake in response to each workaround. To structure this, the workaround action matrix is used.

This document is structured in four chapters; the introduction, context, workarounds, and impact. Chapter context provides more information about the specific customer team; how many people are involved, what role do they fulfil and what is the scope of business with this customer. Chapter workarounds describe all identified workarounds in detail. Chapter impact shines a light on the focus group session in which the devil's quadrangle is used and lays out de decision-making process and what decisions are taken in response to the identified workarounds.

Context

The case study report at hand is that of the bank 1 customer. This team consists of twenty-one members fulfilling five different roles. The role distribution is described in the table below.

Table 1 Roles in team

Role	FTE
Customer operations manager (COM)	1.4
Sales director	1
Mission Critical Tester (MCT)	2
Mission Critical Software Engineer (MCSE)	6
Mission Critical Engineer (MCE)	7

The role of COM is a rather broad one as their responsibilities range from pre-sales to operations and maintaining the customer relationship but also team dynamics. They are generally seen as the first point of contact for whenever the customer has questions that do not involve operational casualties.

The sales director is a dedicated sales representative and is traditionally seen as an account manager.

The MCT role is responsible for testing software that is deployed within the IT landscape of this specific customer.

An MCSE is, in essence, a developer, specific to this customer is that Schuberg Philis develops certain plugins and applications for their IT landscape and these are created and maintained by MCSEs from the team

MSE's are responsible for all operations of the applications and systems within the scope of the project and customer. They generally maintain the customers (cloud) infrastructure. This role also is a key stakeholder when executing changes.

In the case of this customer, the scope of delivery is focussed around maintaining, managing and transforming the whole IT Landscape of the customer. This IT Landscape consists out of some core banking applications and several other applications which together form the front and back-end of the bank. A common characteristic of these services and applications is that the customer is heavily dependent on them if failures occur it will not be possible for this customer to continue operations and is likely to have a financial impact.

Workarounds

Within this specific customer team, seven mission-critical engineers were interviewed. These interviews have led to the identification of 9 workarounds. Table 2 provides a short description of these workarounds and how many times they were detected.

Table 2 comparison of different aspects of the identified workarounds

ID	Workaround	Count
WA2.1	A change is executed without a change ticket (impact too low, the customer does not understand)	2
WA2.2	A change is executed without formal approval from the customer	4
WA2.3	A change is executed while the submitter is the same person as the first approver.	1
WA2.4	Upon failure of a change, a fix forward is executed instead of a rollback	2
WA2.5	Relevant documentation is not updated after executing a change	5
WA2.6	A change is executed, but there was no rollback plan in place	1
WA2.7	A change is executed based on a verbal approval of the customer, and the customer approved the change after execution via the system (Jira)	4
WA2.8	A ticket is created after the change is already executed.	1
WA2.9	Stakeholders are not informed of the execution of a change	1

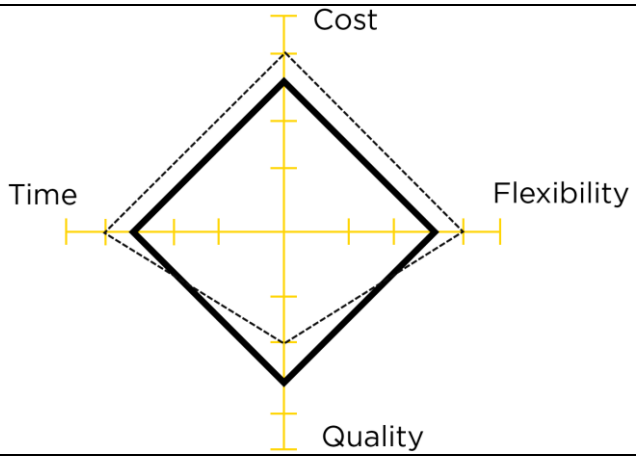
In the tables below, a snapshot is created for all identified workarounds; this snapshot incorporates the workers, description, process model, motivation, impact and action of the workaround.

Referring back to the research method, the impact determination of each workaround is performed via a focus group session with the team members who were interviewed in the previous phase. In this session, seven people attended including the researcher. All six other people had the role of a mission-critical engineer of which one was the team lead.

During this session, the team was asked to score the impact of each workaround in regard to the measurements in the devil's quadrangle. These decisions were made based on the other aspects of the workaround snapshot which were already in place. For each measurement in each workaround, the team gives a -, --, 0, + or ++. A – on costs means that the workaround leads to a reduction in costs. A + on quality means that the workaround results in an increased quality of the process. As multiple people were involved in this session the final score for each measurement and workaround was a team decision on which each individual gave consent.

Based on the determined impact attendees were then asked to decide how they want to act on each workaround. To structure this, the workaround action matrix was used. Attendees could choose between “adopt”, “reengineer”, “prevent”, and “ignore”.

WA2.1

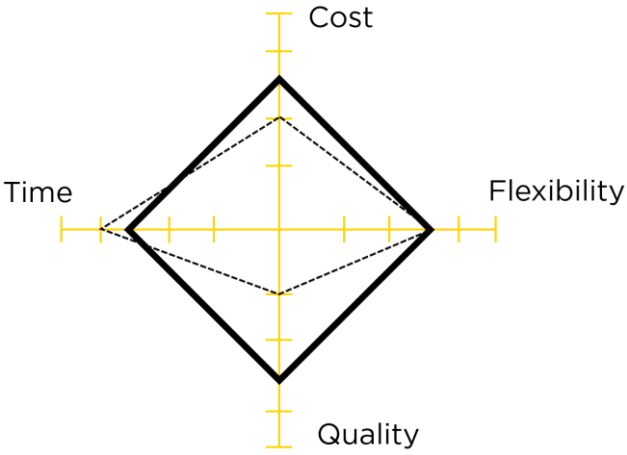
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	a specific change is executed on a chef cookbook (tool for automation of deployments) which does not impact the IT landscape of the customer or has a very small impact that is out of the scope of knowledge for this customer.
Process model	See **page 98**
Motivation	The executor deems this far out of the scope of knowledge of the customer and expects that this specific change will not impact the customer upon failure. It is not worth bothering the customer with this as they will not understand nor care about it too much.
Impact	
Action	Adopt

The team noted that executing this workaround would reduce the costs as they would not have to spend time on creating a ticket for these minor things. For the same reason, they agreed that it would result in a reduction of time spent in the process. On the other side, they do agree that it would reduce the quality of the process as components would be changed without any documentation and as the process prescribes to make it, executing this workaround violates that. The team did see an increase in flexibility as gateways such as approvals are removed, and the executor can act more freely.

As a response, the team chose to adopt this workaround. They agreed that bothering the customer with changes like this does not add any value.

WA2.2

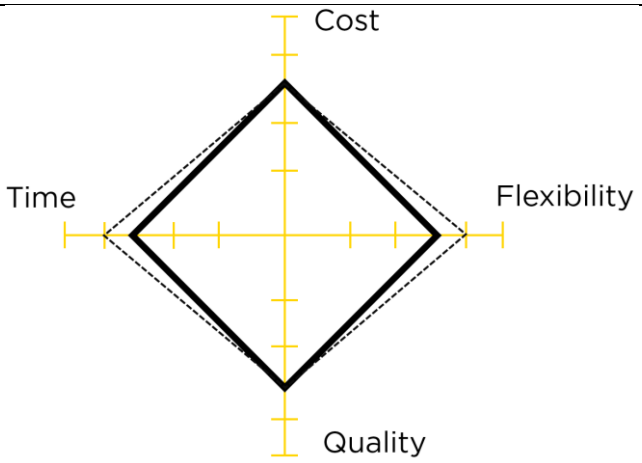
Snapshot component	Content
Date of snapshot	December 2019
Workers	Submitter/executor
Description	A change is executed while the approval mechanism was not started yet.
Process model	See **page 98**
Motivation	The worker didn’t notice this and was unaware of the fact that approval was not provided yet.

Impact	
Action	Prevent

The team agreed that the time spent is reduced as the executor did not have to wait for the customer or colleague to approve the ticket. A neutral impact on flexibility but costs increase as the customer could charge them for it when things go wrong. The quality is decreased significantly because at that time it is not known if the change is any valid as no approval was given.

The team chose that they want to prevent this workaround from happening and will take actions to do this in the future.

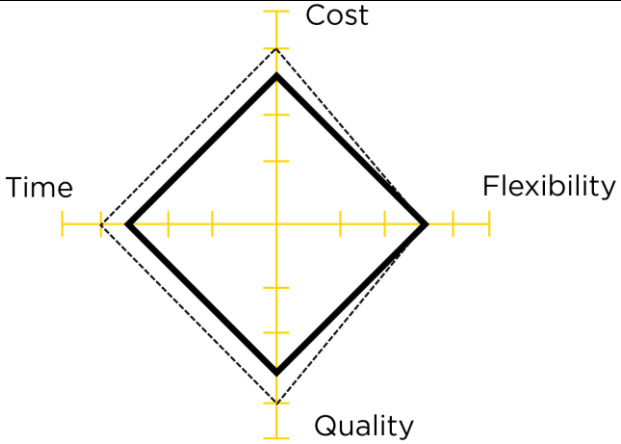
WA2.3

Snapshot component	Content
Date of snapshot	December 2019
Workers	Submitter/executor
Description	Ticket is submitted by an engineer of Schuberg Philis, customer approves the ticket. The second approval which is given by Schuberg Philis is the same engineer that submitted the ticket.
Process model	See **page 98**
Motivation	The change was required to be executed as soon as possible, but no one else was available to approve, so the submitter who is also the executor decided to do this himself.
Impact	
Action	Ignore

The team deems this a neutral impact on both costs and quality. A decrease in time spent as the engineer was able to continue the change and not have to wait for a second approval. This also increases the flexibility of the process.

The team doubted to choose to adopt or redesign as to some degree. It felt wrong for them to ignore it as they set the rules to why a second approval was needed. However, they decided to ignore this workaround as the context of the change required to do deviate from the process and they expected occurrence of this workaround to be rare.

WA2.4

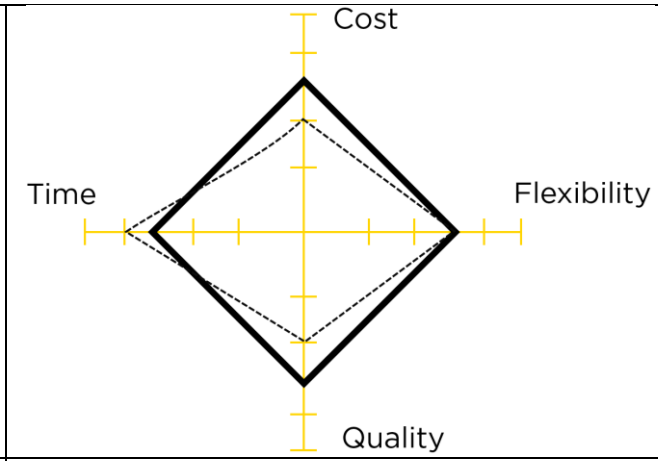
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	A change is executed but does not have the expected outcome and fails, in order to solve this problem, a fix forward is executed instead of a rollback.
Process model	See **page 99**
Motivation	The executor deems that a rollback is not possible or that a fix forwards is likely to be more beneficial, therefore he decides to do that.
Impact	
Action	Redesign

As the motivation behind the workaround indicates that the fix forward would be beneficial the team decides that this workaround has a decrease in cost and in time spent. The impact regarding flexibility was neutral and quality increased.

The team chose to redesign the process and upon failure of a change make it a possibility for the engineer to decide whether it is for the better to rollback or fix forward.

WA2.5

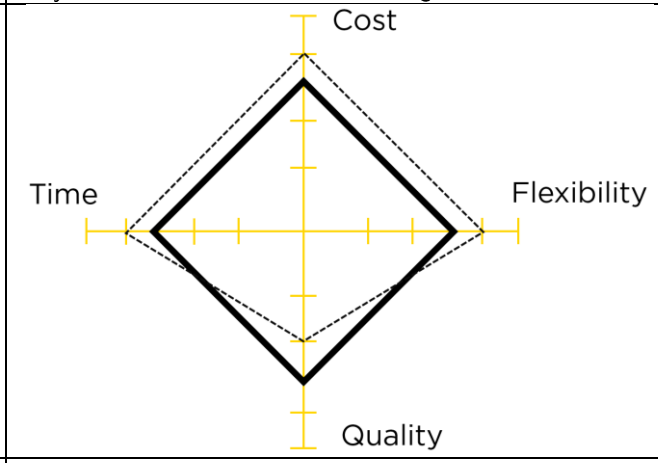
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	a specific change is executed, but the relevant documentation for the system or service that it is applied to is not updated afterwards.
Process model	See **page 99**
Motivation	The executor forgot to do this.

Impact	
Action	Prevent

A neutral impact on flexibility was decided upon. Costs increase as when a situation occurs where something needs to change again, and it becomes more difficult and time-intensive as no relevant documentation is in place or up to date. This also translates to a decrease in quality. Time spent is decreased as writing documentation takes a lot of time.

The team decided to prevent this workaround from happening and be stricter on writing documentation.

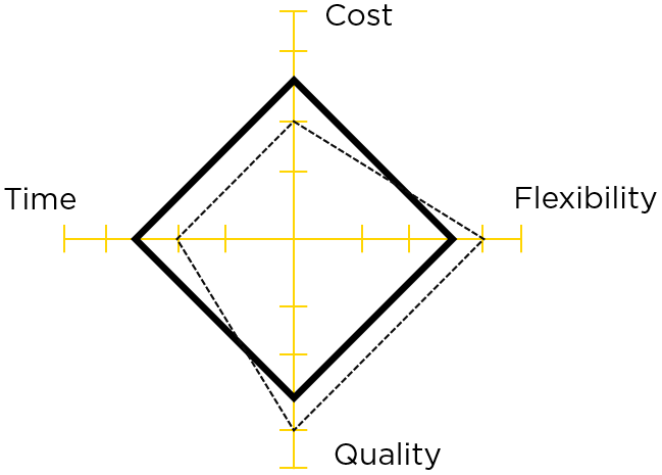
WA2.6

Snapshot component	Content
Date of snapshot	December 2019
Workers	Submitter/executor
Description	A smaller change is executed to a system or service, but upon executing the change, there was no rollback plan for when things go wrong.
Process model	See **page 99**
Motivation	the executor deemed it not necessary to have a rollback plan created as the change was very small and it wouldn't be hard to figure this out if the change failed.
Impact	
Action	Prevent

Costs are twofold, on the short term you might save money as you decrease the time you spent on the change but if it fails, it might cost a lot more as this might cause an extended downtime. It does increase the flexibility as you are freer to adapt to what to do in case of a rollback. Although the quality is decreased.

The team chose to prevent this change. It did not matter to them that it was a smaller change as it was always required to have a plan for this.

WA2.7

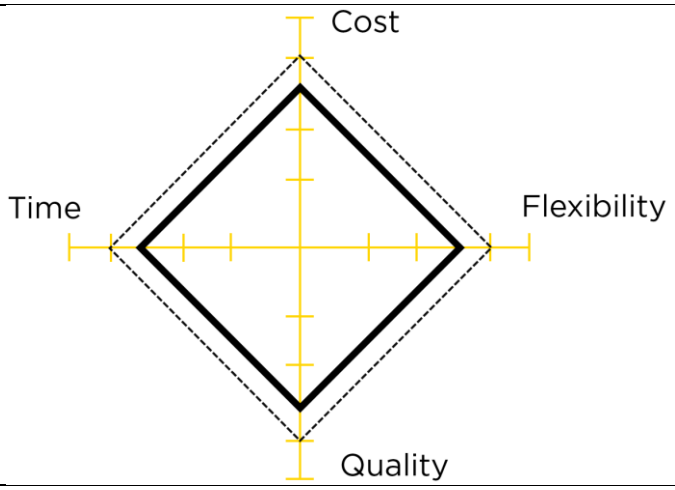
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	A change ticket is submitted, and the executor wants to execute this as there is a time impairment. Instead of waiting for the customer to give formal approval via the system, the executor calls or finds alternative ways to contact someone who is mandated to approve and asks for verbal approval. Based on this, the executor executes the change, and formal approval is given afterwards. Note that all interviewees whom do this said that they will not execute changes without approval but that in some cases this verbal approval is formal enough for them.
Process model	See **page 100**
Motivation	Multiple motivations are given; there could be a time impairment which requires the change to be executed as soon as possible or when it is during the night, and most approvers sleep.
Impact	
Action	Adopt

The team deemed that time spent decreased as they did not have to wait until the approver was available. The costs would also be reduced as a potential downtime was avoided by this. Both flexibility and quality increased as the executor was able to act immediately without negative effects.

The team choose to adopt this workaround and spread it across the team that when in case of emergency, this is the way to handle.

WA2.8

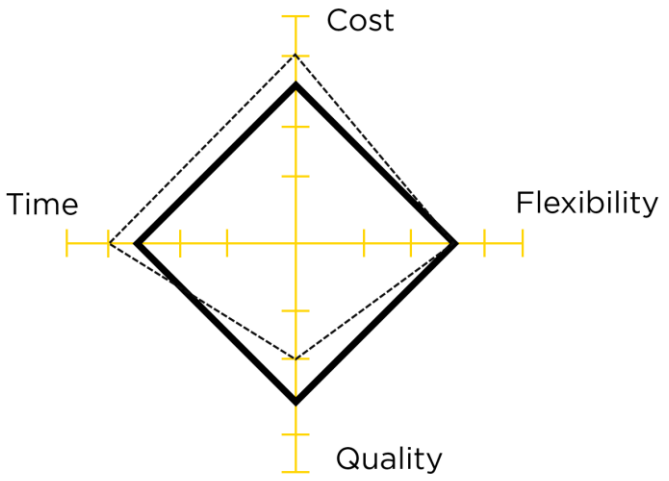
Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	An emergency occurs on any system or service which causes downtime. In order to fix this, a change needs to be executed. The executor acts first and fixes the problem and creates a ticket afterwards and asks for approval then.
Process model	See **page 100**

Motivation	The executor made this decision as time was of the essence, and creating a ticket first would lead in more downtime.
Impact	
Action	Redesign

The team agreed that both flexibility and quality of the process increased as the executor was quick to handle when it was required. Both time and costs decrease as downtime is avoided.

The team chose to redesign the process and make it able to do an emergency change. This would mean the approval workflow would start after executing the change.

WA2.9

Snapshot component	Content
Date of snapshot	December 2019
Workers	Executor
Description	Once a change is set for execution, the executor informs all stakeholders about the execution in the form of a maintenance message. In a certain situation, this message was not sent out and stakeholders were not informed about the change.
Process model	See **page 100**
Motivation	The worker was unaware that this was not done yet and forgot to inform all stakeholders in the process.
Impact	
Action	Redesign

The quality decreases as people might want to use the system and are not informed that it is unavailable. There is no impact on flexibility and a decrease in costs and time as no activity was executed.

The team chose to redesign the process as they saw ways to automate this. For example, the system could send out a maintenance message to all stakeholders based on the timeframe inserted in it.

Types of workarounds

When looking (see table 3) at the workarounds from the axis of who executes it and what is the general motivation why they execute it. It shows two things; execution of these workarounds is mainly performed by the executor and/or submitter and the in most cases the motivation behind a workaround is either the executor forgot to execute the specific activity or they acted in the best interest of the customer and therefor deviated.

The reason behind the fact that executor or submitter was identified as a worker in each workaround is that the all people that were interviewed have an engineering role within the team and can only take on these two roles within the process. This explains why no other roles were identified as a worker within this case study.

The more relevant aspect is that of the motivation. When summarising the motivation and somewhat generalise them to an overall motivation it shows that most of the workarounds identified were executed because the worker either forgot to execute an activity or that he/she acted in the best interest of the customer and therefore deviated. In only one case, the worker willingly skipped to execute an activity and in one case, the worker was unaware.

When combining the motivation and actions taken it comes to notice that for each workaround where the worker acted in the best interest of the customer it was decided to either adopt the workaround or redesign the process to make this more flexible.

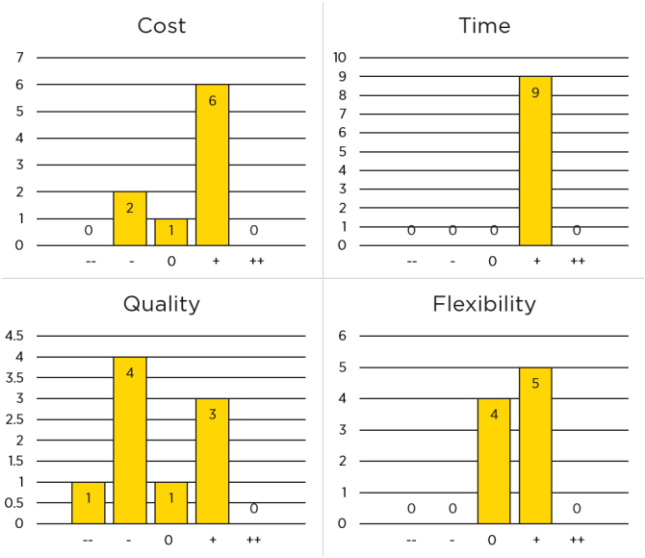
At last, each workaround was connected to a corresponding activity in the process to see if there are any relations. Most of the workarounds were either related to the approval mechanism or documentation of changes in the form of a ticket.

Table 3 comparison of different aspects of the identified workarounds

ID	Worker	Motivation (summarized)	Action taken	Corresponding activity
WA2.1	Executor	Acted in the best interest of the customer	Adopt	Documents changes (ticket)
WA2.2	Submitter/executor	Unawareness	Prevent	Approval mechanism
WA2.3	Submitter/executor	Acted in the best interest of the customer	Ignore	Approval mechanism
WA2.4	Executor	Acted in the best interest of the customer	Redesign	Failure of change (not identified)
WA2.5	Executor	Forgot to execute activity	Prevent	Update relevant documentation
WA2.6	Submitter/executor	Intentionally avoided execution of activity	Prevent	Create rollback plan
WA2.7	Executor	Acted in the best interest of the customer	Adopt	Approval mechanism
WA2.8	Executor	Acted in the best interest of the customer	Redesign	Documents changes (ticket)
WA2.9	Executor	Forgot to execute activity	Redesign	Communicate results

Impact

The impact of these workarounds in terms of costs, time, flexibility, and quality of the process show interesting results. Execution of these workarounds has had a positive effect on costs in six cases. In two cases a negative impact was determined and on one occasion impact remained neutral. The time dimension is the least dynamic as all workarounds had a positive impact in time spent on the process. While a positive impact was determined for time and costs, in five occasions a negative impact was determined for quality. Only three times a positive impact was determined and one time it remained neutral. The flexibility of the process generally increased or remained neutral.



Another part of the impact is how the team reacted to these workarounds and what actions they want to undertake. At first, I expected that the team would choose to ignore a lot of workarounds as in most cases, the executor acted in best interests of the customer. Although only one workaround was ignored as the context required the executor to handle fast and the team expected occurrence of this workaround to be rare.

Three workarounds were chosen to prevent. For all three of these workarounds, the reason was either unawareness, the executor forgot to execute it or willingly avoided it. The team also confirmed that activities regarding these workarounds were somewhat tedious and administrative.

Adoption of the workaround was chosen two times, the team agreed that this was not the official procedure but did not see these workarounds fit for a redesign of the process and would rather spread them through the team as “only do this when there are no options left”.

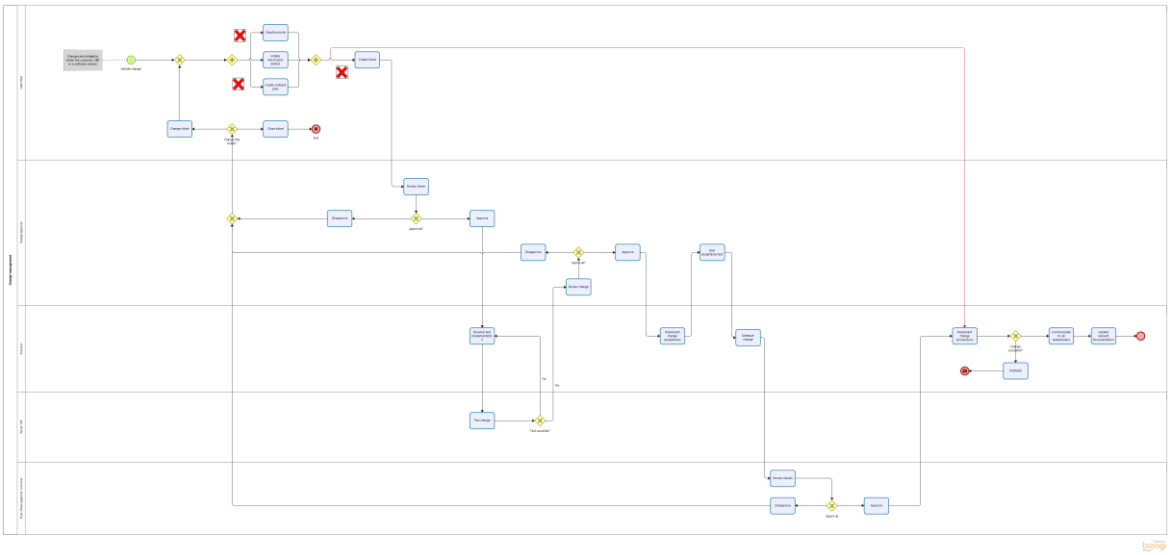
Redesigning the initial process was chosen on three occasions. For workaround WA2.4 the team was actually surprised that it was not part of the process as fixing forward is a common practice and a rollback is not always possible.

Table 4 comparison of different aspects of the identified workarounds

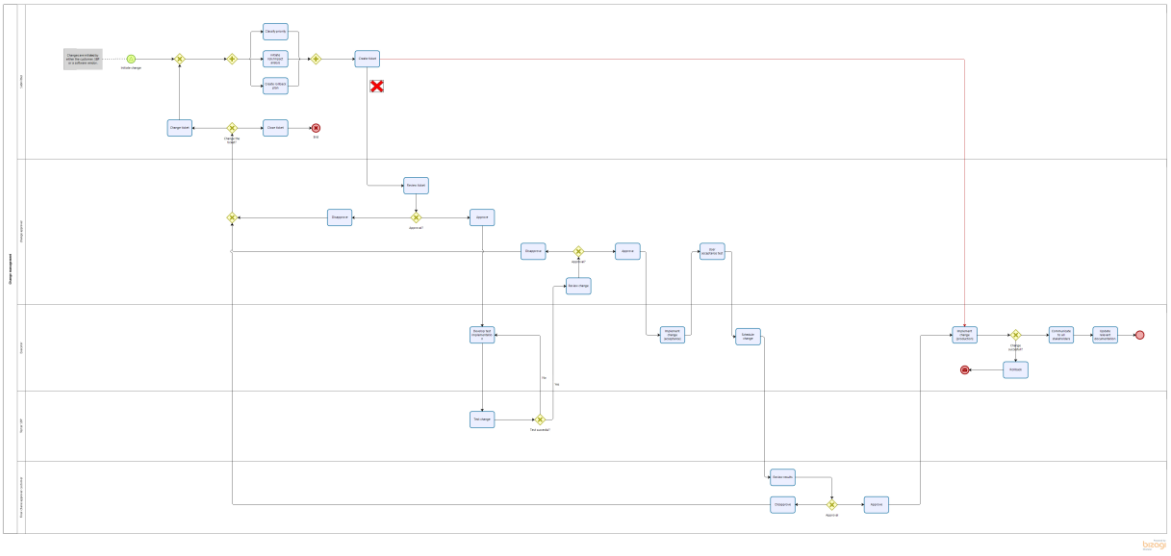
Action	Action description	Workaround ID
Ignore	No action in response to the workaround, no effort required.	WA2.3,
Prevent	Undertake action to avoid the occurrence of this workaround.	WA2.2, WA2.5, WA2.6
Adopt	Accept the workaround and spread it within the team	WA2.1, WA2.7
Redesign	Redesign the process.	WA2.4, WA2.8, WA2.9

Process models

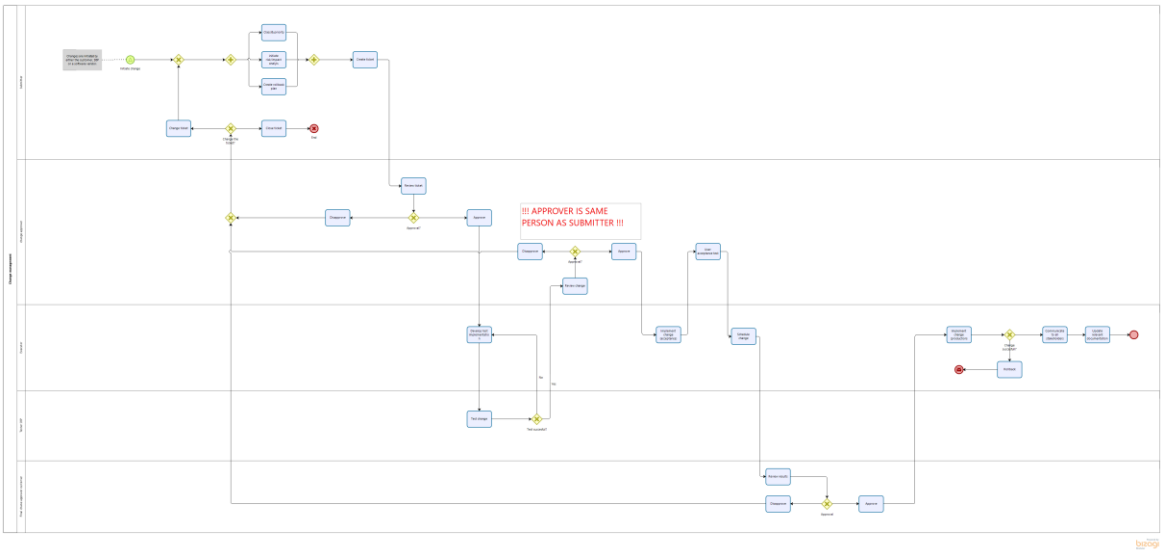
WA2.1



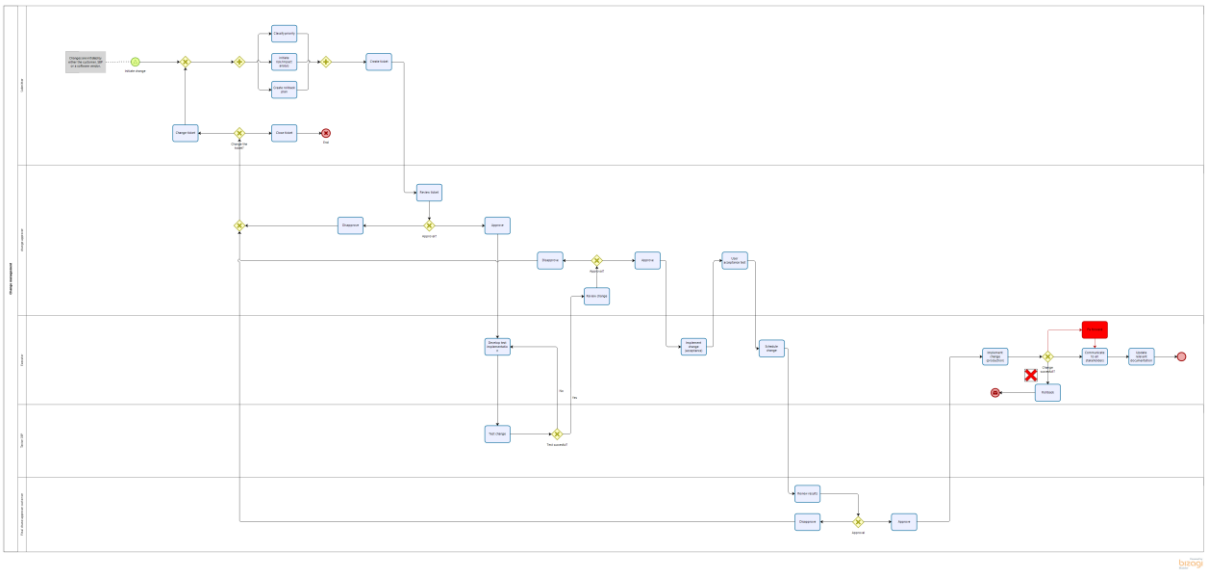
WA2.2



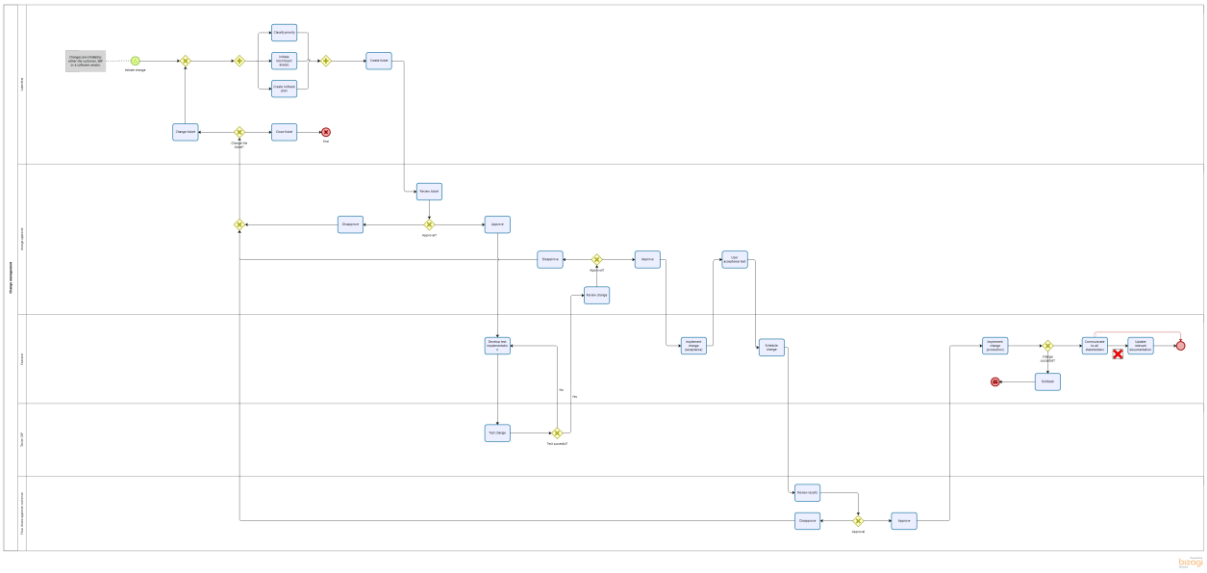
WA2.3



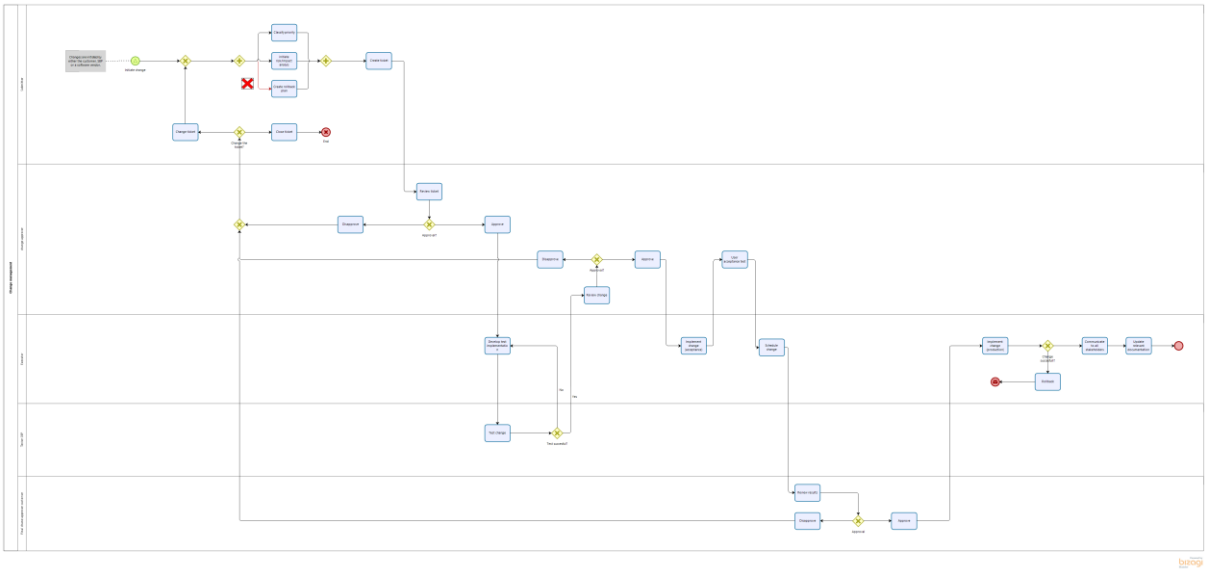
WA2.4



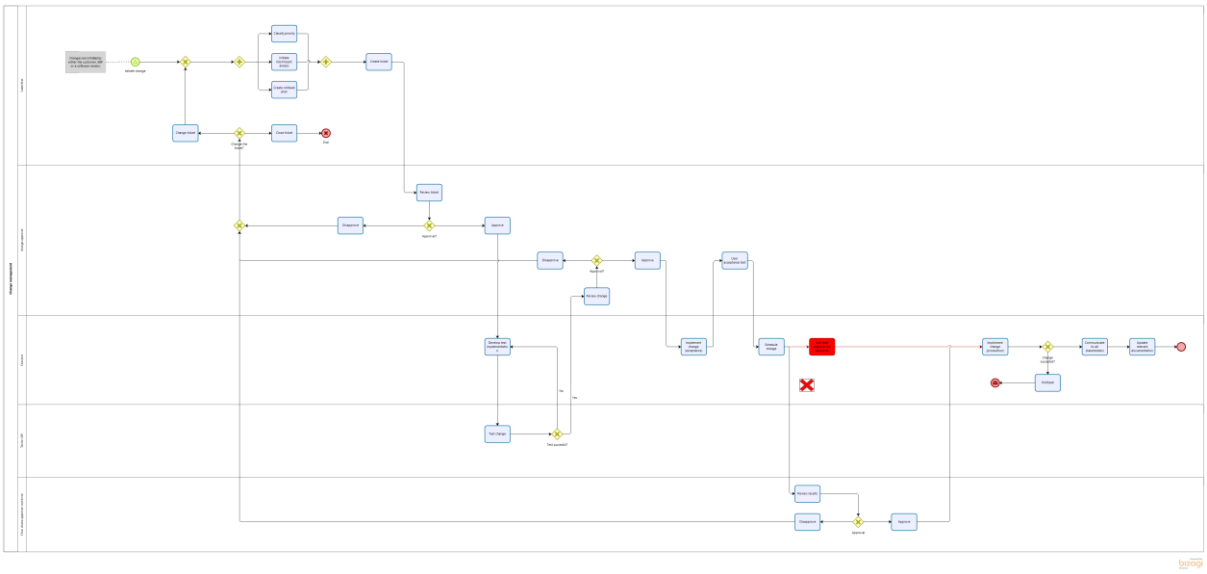
WA2.5



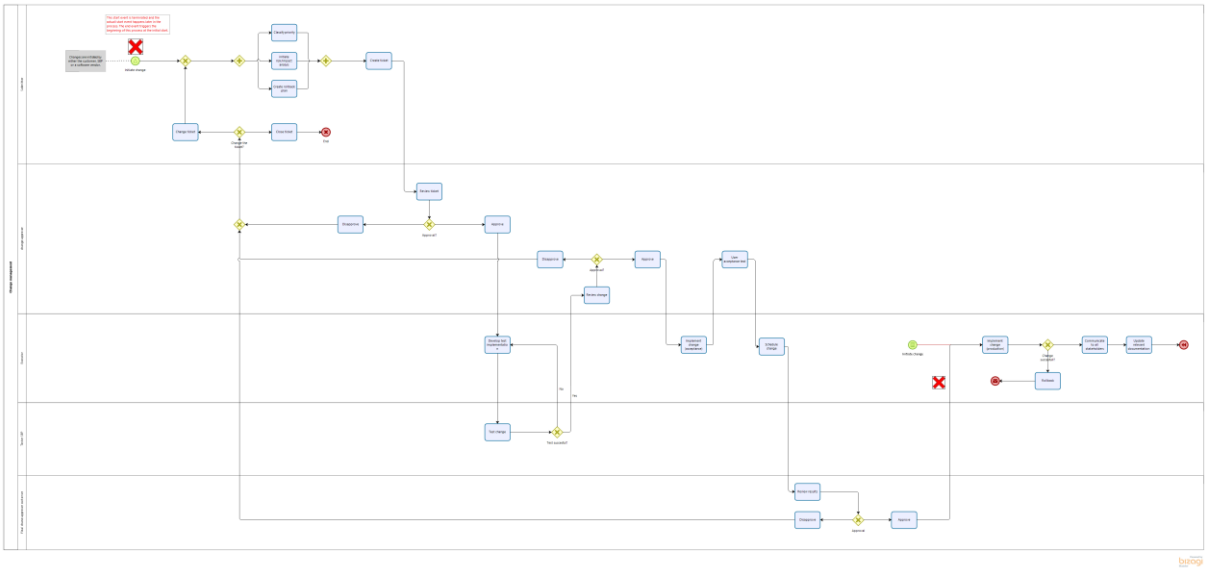
WA2.6



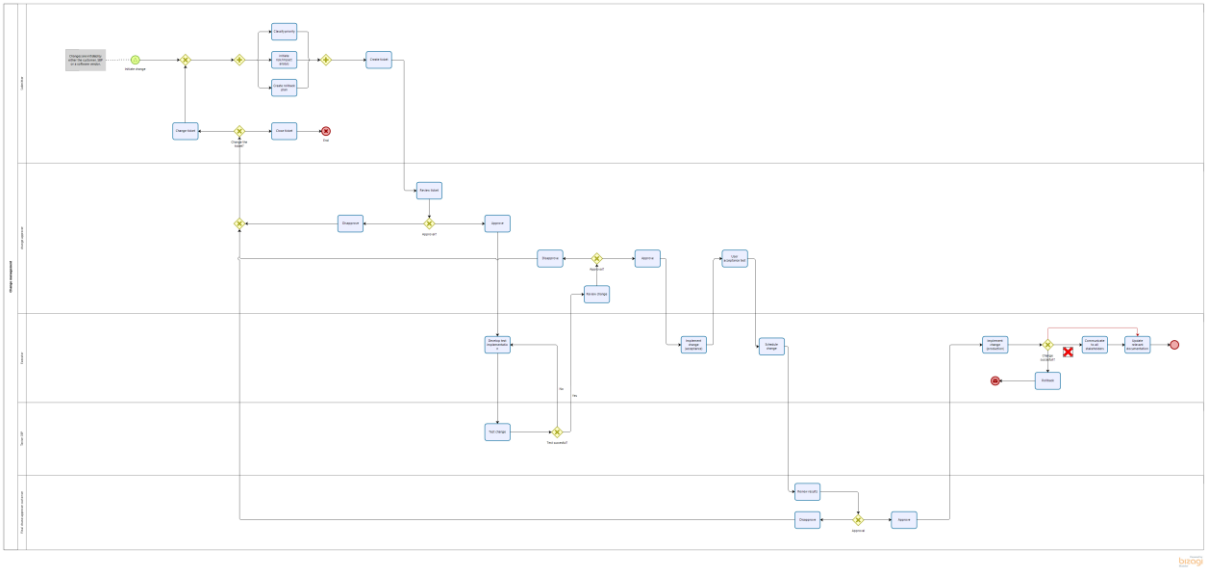
WA2.7



WA2.8



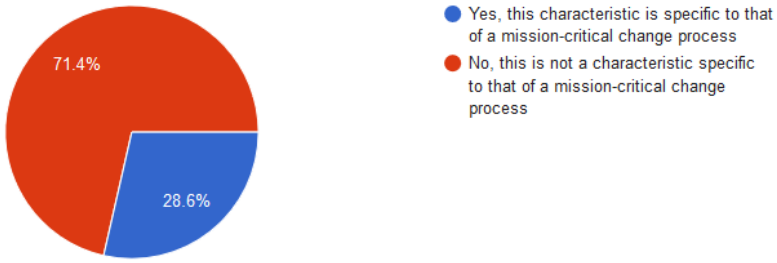
WA2.9



F Survey results

Each change is documented; changes are registered through a service management tool in form of a ticket.

7 responses



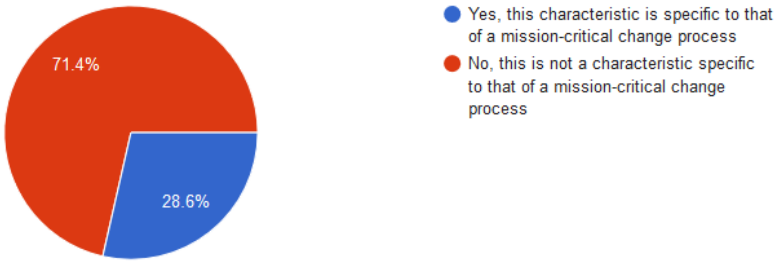
Comment:

1 response

Each change is (should be) documented, mission critical or not.

Risk and impact of a change is determined; in order to execute a change a risk/impact analysis is executed. This determined risk and impact form input for both the prioritization of a change and the workflow that the change will follow. For example, a change with a high impact and high risk is often required to be approved by more people or people in a higher position.

7 responses



Comment:

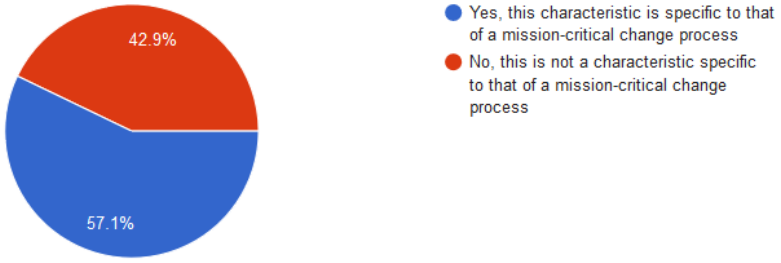
2 responses

Risk and impact of a change is (should be) determined, mission critical or not.

Usually changes are reviewed and approved by at least one person. High impact changes are analysed in groups and once consensus is found then it is approved by one in the name of the others.

There is a plan for failure; when executing a change, it is possible for that specific change to fail or not give the expected outcomes. Whenever this happens there is plan to mitigate this. This plan describes what and how to act when a change fails.

7 responses



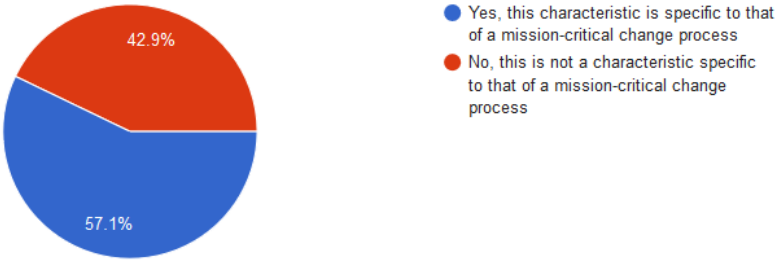
Comment:

1 response

MCEs are usually planning for rollback mechanisms.

Approval mechanisms; each change needs to be approved by someone else than the submitter. Depending on the risk and impact of a change this would be one or two people, following the 2-/4-/6-/8-Eye-principle. Interesting to notice is that for some customers therequirements are stricter as these require a 6-/8-eye-principle.

7 responses



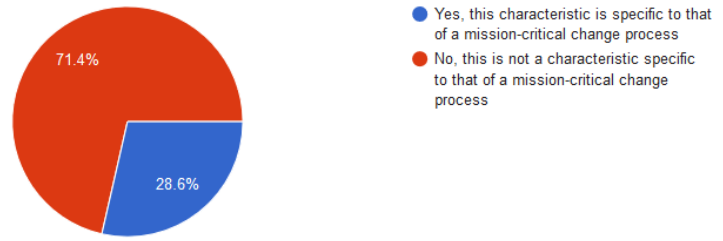
Comment:

1 response

Each change needs to be approved by someone else than the submitter, mission critical or not.

Changes are planned/scheduled; based on impact, priority, application/service or infrastructure changes are scheduled to what suits the executor best or designated maintenance windows.

7 responses



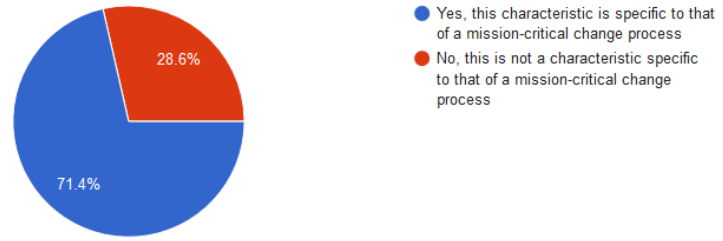
Comment:

1 response

There is usually a 5-7 days time-frame ahead of the maintenance activity when an announcement is made.

Changes are tested; in order to execute a change, test scenarios are to be created, these test scenarios add structure in testing the outcome of the change and whether the change was successful or not.

7 responses



Comment:

1 response

MCEs make use of D/T/A/P environments staged environments before the change reaches prod.

Do you have any other remarks regarding this subject?

2 responses

A change process should not deviate from an agreed process, mission critical or not.
It should be the combination of risk and severity that classifies the change to be mission critical. Still, it should be documented in change system, have a separate approval flow. Fallback plans and multiple eye approvals can be agreed in the process once risk and severity have determined the change to be a mission critical one.

The most important characteristic of a change is the 2+ pair of eyes principle and ahead of time change notifications.