Exceptions and flexibility in business processes

examining the effects of exceptions on key performance indicators and the use of flexibility

van IJzendoorn, G.R.

Award date:
2013

ACCENTURE & EINDHOVEN UNIVERSITY OF TECHNOLOGY, IEIS

Exceptions and Flexibility in Business Processes

Examining the Effects of Exceptions on Key Performance Indicators and the Use of Flexibility

van IJzendoorn, G.R.

10/28/2013

First Supervisor TU Eindhoven: **Remco Dijkman**

Second Supervisor TU Eindhoven: **Oktay Türetken**

Supervisor Accenture: **Meint de Vries**

Abstract

This research presents a qualitative and quantitative analysis of the effects that exceptions have on the performance of business processes measured with key performance indicator(s) and how flexibility techniques can help enable these exceptions appropriately. For organizations in highly volatile and competitive environments the ability to enable or restrict exceptions through flexibility techniques in business processes is an increasingly important competitive advantage. The speed and effectiveness of (re-)design, implementation, execution and monitoring of their processes to satisfy customers becomes increasingly important. Companies seek out a balance in their business processes between standardization, i.e. making processes as routine as possible, and flexibility, i.e. enabling exceptions.

The analysis of effects which exceptions have on key performance indicators with which business process performance is measured and the ability of flexibility techniques in business processes to enable these exceptions provide insights which will be beneficial in finding this balance. Currently it is often still not clear what the effects of exceptions are on key performance indicators that measure business process performance, and how organizations should enable exceptions in their business process management systems through business process flexibility. In other words, it is often not clear what the balance between standardization and flexibility in business processes should be. This is a problem for which this research aims to find a solution.

In order to make that balance clear, a research approach is designed. First a literature study is conducted to find canonical flexibility techniques in business processes. Flexibility techniques in business processes are then assigned to flexibility categories. After that the extent to which exceptions are considered relevant in practice is researched through case study interviews. The case study interviews also show to what extent practitioners enable exceptions or try to avoid having to use them. The key performance indicators they pay attention to regarding the measuring of the performance of business processes are then identified. Finally the result of the case study interviews is a set of hypotheses which are tested through quantitative means. The hypotheses are tested through data analyses on big data sets using process mining techniques and linear regression analyses to measure the actual effects exceptions have on the previously identified key performance indicators. And finally by aligning the flexibility techniques with exception types input is provided on how the identified exceptions should be enabled.

The flexibility categories that have been identified in the literature study were: flexibility by design, flexibility by deviation, flexibility by underspecification, flexibility by change, and flexibility by configuration. They each have a number of flexibility techniques of which the relevant ones are used to show how flexibility techniques are aligned with identified exception types. The relevant ones are shown in Table II. The case study interviews were conducted for five cases: two Dutch banks, a Dutch quango, the insurance service for US military and a US company distributing consumer electronics. Seven interviews divided over five cases were conducted and it became clear that throughput time is a consistently relevant key performance indicator. Practitioners indicated they would like to have throughput time decreased as much as possible and it is concluded that they aim to prevent exceptions or otherwise handle them in the design phase of business process models. This resulted in the hypotheses mentioned in Table I. Five data sets containing event logs have been analyzed from a Dutch industry company dealing with middleware for a change control board, a big US company and Volvo both dealing with incident management, the Eindhoven municipality handling appeals, and a

Table I: Hypotheses, whether they have been confirmed, rejected or found inconclusive and the effects the exception typed had on throughput time

Hypothesis	Confirmed, rejected or inconclusive	Reasoning in relation to effects the exception types have on throughput time
Main hypothesis Exceptions in business process models cause an increase of throughput time.	Rejected	Correlations between exceptions in general and throughput time are both positive and negative.
Sub-hypothesis 1 Unexpected exceptions in business process models cause an increase of throughput time.	Rejected	Correlations between unexpected exceptions and throughput time are negative where they should be positive to confirm the hypothesis.
Sub-hypothesis 2 Expected exceptions in business process models cause an increase of throughput time.	Confirmed	Correlations between expected exceptions, which includes the Early Exit exception, and throughput time are positive.
Sub-hypothesis 3 The influence of an exception in a business process model on throughput time depends on the type of exception.	Confirmed	The standard beta coefficients from the regression analyses are different per type of exception.
Sub-hypothesis 3a The exception type "skip" in business process models causes a reduction of throughput time.	Confirmed	The standard beta coefficients from the regression analyses having the skip exception type are all negative.
Sub-hypothesis 3b The exception type "early exit" in business process models causes a reduction of throughput time.	Rejected	The standard beta coefficient from the regression analyses having the early exit exception type is negative where it should be positive to confirm the hypothesis.
Sub-hypothesis 3c The exception type "iterate" in business process models causes an increase of throughput time.	Confirmed	The standard beta coefficients from the regression analyses having the iteration exception type are all positive.
Sub-hypothesis 3d The exception type "loop" in business process models causes an increase of throughput time.	Inconclusive	There have been no significant results for the loop exception type in the regression analyses which means the hypothesis cannot be confirmed nor rejected.
Sub-hypothesis 3e The exception type "analysis" in business process models can have both an increasing and a decreasing influence on throughput time.	Inconclusive	There has been only one significant result for the standard beta coefficient of the analysis exception type in the regression analyses where both positive and negative results were needed to confirm the hypothesis and several results of all positive or all negative were needed to reject the hypothesis. This means the hypothesis cannot be confirmed nor rejected.

Dutch financial institute handling loan applications. Five exception types resulted from the process mining analyses: skip, iteration, loop, analysis, and early exit. The effects they had resulted in a conclusion on the hypotheses from Table I. The effects of the exception types on throughput time, positive or negative, and what consequences they had regarding the hypotheses can be read in Table I. The flexibility techniques of which it was concluded that they enable the identified exception types most appropriately are shown in Table II.

Table II: Identified exception types and which flexibility techniques of which flexibility technique best enable these exception types for the analyzed cases

Cases from data analysis	**Exception Types**	**Flexibility Category**	**Flexibility Technique**
Dutch industry	Analysis (Expected)	Flexibility by Design	Choice
			Interleaving
		Flexibility by Configuration	Enable
			Insert
	Skip (Unexpected)	Flexibility by Deviation	Skip Task
	Iteration (Unexpected)	Flexibility by Deviation	Redo
	Loop (Unexpected)	Flexibility by Deviation	Undo
			Redo
Volvo	Analysis (Expected)	Flexibility by Design	Choice
			Interleaving
		Flexibility by Configuration	Enable
			Insert
	Skip (Unexpected)	Flexibility by Deviation	Skip Task
	Iteration (Unexpected)	Flexibility by Deviation	Redo
	Loop (Unexpected)	Flexibility by Deviation	Undo
			Redo
Eindhoven Municipality	Early Exit (Expected)	Flexibility by Design	Choice
			Interleaving
			Cancelation
		Flexibility by Configuration	Enable
			Insert
Big US company	Early Exit (Expected)	Flexibility by Design	Choice
			Interleaving
			Cancelation
		Flexibility by Configuration	Enable
			Insert
	Skip (Unexpected)	Flexibility by Deviation	Skip Task
	Iteration (Unexpected)	Flexibility by Deviation	Redo
	Loop (Unexpected)	Flexibility by Deviation	Undo
			Redo
Dutch financial institute	Early Exit (Expected)	Flexibility by Design	Choice
			Interleaving
			Cancelation
		Flexibility by Configuration	Enable
			Insert
	Skip (Unexpected)	Flexibility by Deviation	Skip Task
	Iteration (Unexpected)	Flexibility by Deviation	Redo

This research provides not only results that are a starting point for a wider research on the effects of exception types on throughput time (and other key performance indicators) and on flexibility techniques for business processes that are most appropriate to handle them, but also an approach on how to analyze the effects of exceptions through both process mining and linear regression analyses. Especially the approach to combine process mining techniques, data mining and linear regression analyses to measure the effects exception types have on throughput time has not been done before. The results are a starting point to having a repository of exception type effects on throughput time and the approach to aligning flexibility techniques with these exception types provide insights which organizations might use in how they manage their business processes. The results such as a skip causing reduction of throughput time and the Skip Task technique to be the technique that enables it are perhaps not always ground-breaking. But these results, and especially the approach to getting these results, provide a start of a solution to solve the problem of the balance between standardization and flexibility in business processes being unclear.

Preface

This research is a Master thesis project to conclude the Master study of Operations Management & Logistics at the Eindhoven University of Technology with the Information Systems group at the department of Industrial Engineering and Innovation Sciences.

The project is done in cooperation with Accenture in Amsterdam at the department of Digital Data & Analytics though the research received input from various departments from Accenture. Accenture is one of the largest consulting firms measured by revenues and contains a strong knowledge base in Information Technology, data analysis, business process management and business intelligence which are relevant fields for this research. Besides Accenture, the Information Systems Group at the faculty of Industrial Engineering & Innovation Sciences at the Eindhoven University of Technology was involved through guidance, feedback, and general support. The total duration of the Master Thesis project is 6 months.

My personal aim of this Master Thesis was for me to get more familiar with the field of business process management and its overlap with business intelligence via process mining. On top of that the knowledge available from consultants at Accenture working in the field of business process management and the people I have worked with was a great source to tap from. Combining both qualitative and quantitative means of analysis in the setting of a Master Thesis on Business Process Management and flexibility proved to be a very pleasant and great learning experience. It gave me a platform to incorporate skills I had acquired from my study and outside of it which I gained through extracurricular activities, but also a platform to absorb more knowledge and improve my skills further.

The progress was done as it was planned out, though the planning was done iteratively in such a way that when a hitch would appear new plans would be made accordingly. The literature study performed for this research was already done prior to this Master Thesis as a preparation, which means that it was not included in the planning of six months of working on this research.

The interviews that have been performed were based on a convenience approach as due to non-disclosure no clients could be contacted directly without this research having interviews as main method. That means that only Accenture consultants were available for interviews. External people that were not clients to Accenture could have been approached too, but that option was omitted due to the great amount of knowledge already available in Accenture and the time that would have to be spent on it resulting in it falling out of the scope of this research.

To measure the effects of exceptions on throughput time, a process mining software tool was used. Several vendors offered to provide their process mining software tool for free during the research as long as the purpose of the usage was purely academic and not commercial. Disco from the company Fluxicon had been chosen to perform the process mining analysis because of its ease of use but also because of the Fuzzy algorithm that is used in Disco being appropriate for this research. Furthermore Microsoft's Excel and IBM's SPSS have been used to further process the data for statistical analyses as I already knew both programs quite well. The big downside in the usage of Excel in combination with big data was that there were enormous lags in the processing of the data which caused enormous delays due to Excel not being able to handle such large sets of data quickly. On top of that, when later on in the analysis another approach was chosen to manipulate data, this would not automatically change in all three programs together. This would each time mean manipulations would have to be done manually in each program separately, thus adding to the delays already caused by the laggings. Fortunately a large buffer was incorporated in the planning for this part of the research which means it caused very little real delay time schedule-wise. Several data sets are used coming from a variety of sources working in the process mining community. The sources that have been used are academic, coming from the Eindhoven University of Technology, as well as corporate, coming from both process mining software vendor Perceptive Software and indirectly from Swedish car manufacturer Volvo.

Finally I would like to thank the supervisors who helped with this thesis through their guidance which gave me the occasional nudge in the right direction, but still left me with the great amount of room to develop and grow individually. I also would like to voice my great appreciation to the variety of people who provided me with the needed data for my thesis – namely the ones involved from Accenture, Perceptive Software and the Eindhoven University of Technology. They helped either through interviews or by sending me data sets. For the free use of Disco I would very much like to thank the people from Fluxicon; Anne and Christian, and I want to thank Anne once more as she was so eager to share her knowledge.

But the biggest appreciation of all goes to Christine and my mother, who have both been there for me when I needed it and also just because, motivating me all the way, who made me proud to have them in my life, and without whom this thesis would not have been made possible.

Table of Contents

Abbreviations

BPM	=	Business Process Management
CCB	=	Change Control Board
KPI	=	Key Performance Indicator
LN()	=	Log-normalized
PAIS	=	Process-Aware Information System
RQ	=	Research Question
Z()	=	Standardized

List of Figures

List of Tables

1. Introduction

This thesis investigates the relation between expected and unexpected events that occur in an organization during daily operations, and the operational performance of that organization. It familiarizes the reader with this relation, but also existing techniques that ensure these events can be enabled, i.e. that it is possible that they can occur.

The research problem and its relevance are explained in more detail in Section 1.1. The situation in the field of research is explained, resulting in a problem statement and the overall direction that is considered in solving the problem. Section 1.2 goes into the definition of exceptions in the context of this thesis, whereas section 1.3 goes into process mining in order to establish how they are relevant in this research. Furthermore a research goal and research questions are stated as a basis to analyze the problem which is outlined in Section 1.4. Finally in Section 1.5 a description is given on how the research goal is achieved and the research questions are answered through a research design which will be followed throughout the rest of the thesis. In Section 1.6 the thesis structure is explained so that it becomes clear where one can find the outcome of which part of the research design, conclusions and so on.

1.1 Problem outline and relevance

This section explains the trade-off between standardization and flexibility in business processes and why the focus on flexibility in the context of exceptions in this trade-off is necessary.

An exception is defined as a special occurrence which causes a business process to not be executed as desired (Reichert and Weber 2012). Business Process Management (BPM) is defined as the methods, techniques and tools to support the design, enactment management and analysis of operational business processes (Van Der Aalst, Ter Hofstede et al. 2003). Flexibility in this context is the ability to respond to exceptions of business processes in the operating environment without necessitating a complete redesign of the underlying model (Schonenberg, Mans et al. 2008). The economic success of an enterprise depends on its ability to react to changes, such as new laws or new customer demands, in a quick and flexible way (Weber, Reichert et al. 2008). Furthermore, for companies in highly volatile and competitive environments the ability to make rapid changes in business processes is an increasingly important competitive advantage. Firms in such environments must be able to quickly (re-)design, implement, execute and monitor their processes to satisfy customers (Kirchmer, Gutiérrez et al. 2010).

At the same time companies seek out to standardize their business processes in order to improve their business processes as is made clear in Chapter 3 while there is still a need to allow for flexibility in business process models in different ways and for this flexibility to be handled appropriately as is shown in Chapter 2. A challenge is to ensure the trade-off between business process standardization and business process flexibility tailors to the business needs and to where most business value can be achieved (Genovese 2007, Hill 2012). When to focus on incorporating flexibility in business processes and the extent to which it should be incorporated depends on the effects exceptions have on the operational performance of an organization and also what exactly has to be measured with regards to the operational performance as flexibility can control these effects and by extension the operational performance (Reichert and Weber 2012). That is why in the trade-off between standardization and flexibility, the next point to look at is exceptions, which key performance indicators matter in practice, and the effects exceptions have on these key performance indicators (KPIs).

The decision-making on which exceptions in business process flows to enable is an emerging topic in process mining. This is the case because process mining provides a means to make the effects of exceptions on key performance indicators tangible. Therefore process mining is relevant for this research. Gartner describes process mining as "a set of techniques and tools for uncovering undocumented process paths in existing systems, thereby speeding up the identification of "as-is" processes. It is invaluable in zeroing in on wasteful exceptions and, ultimately, in optimizing processes." (Jones 2012). Therefore process mining can be used to establish what the relation is between exceptions and KPIs in business processes. In turn this will help to establish the importance of flexibility techniques to handle these exceptions.

The previous problem outline regarding the unknown implications of the effects of exceptions on business process performance in relation to the balance between standardization and flexibility, results in the following problem statement:

Problem Statement
It is not clear what the effects of exceptions are on key performance indicators that measure business process performance and how organizations should handle exceptions in their business process management systems through business process flexibility.

If it becomes clear from practice whether exceptions are relevant and what KPIs most optimally measure the (positive or negative) effects of exceptions according to practitioners, the scope of the research can be further funneled to measuring the effects of exceptions on these specific KPIs and a link can be made between flexibility techniques that enable exception types.

The focus of this thesis mainly lies in the previously mentioned point which states that the effects of exceptions on KPIs must be made tangible, but another challenge is to find out how to integrate flexibility in business processes and how to determine which exceptions should be taken into account when (re-)designing a business process model. Therefore an understanding of the different flexibility techniques is needed as well (Chapter 2). However, first the concepts of exceptions and process mining are elaborated upon in order to further make the context of them in this thesis clear.

1.2 Unexpected versus Expected Exceptions and Core Flow

This section elaborates on exceptions in BPM to outline how they are relevant to this thesis. There are several types of exceptions that may have a different effect on business performance. In particular we distinguish between expected and unexpected exceptions (Reichert & Weber, 2012). Literature on the two types of exceptions goes into how Process-Aware Information Systems (PAIS) enable handling these exceptions. A Process Aware Information System is a process support information system that allows for separating process logic and application code. At build-time the process logic is defined in a meta-model whereas in run-time the processes are then orchestrated according to the defined logic (Weber, Reichert et al. 2008, Weber, Reichert et al. 2009, Weber, Reichert et al. 2010). Workflow Management Systems are examples of these systems (van der Aalst 2009). The literature on the two types of exceptions does however not answer the question on how to measure the effects of exceptions in business processes on pre-specified KPIs that measure process performance, nor how to use flexibility techniques to handle exceptions based on their effects on these KPIs. Here the difference between core flow, expected exception and unexpected exception is explained to provide context for later use in this research.

Expected exceptions are exceptions that have been recognized during build-time of the business process model, whereas unexpected exceptions were not recognized during build-time and require flexibility at run-time of the business processes. The distinction between expected and unexpected exceptions in a PAIS is more detectable once a pre-designed (or pre-specified) process model is available and as such this distinction will also be made in this research. Exceptions are, however, not always detectable as this highly depends on how events have been logged and *if* they have been logged even (Reichert & Weber, 2012).

In Figure 1 the difference between a core flow, expected exception and unexpected exception is illustrated. A Pre-designed Process Model is the model that has been pre-specified before it is made operational, designed at build-time. In this model, business process model designers will have taken into account the core flow, which is defined as the most desirable route where they expect or at least hope to have most cases to go through. They will also have taken into account the expected exceptions, which are defined as not desirable paths, but which still are necessary to maintain flexibility in the PAIS. When the processes are running, the reality may show a discrepancy, which is also illustrated in the figure in orange. At run-time paths that have not been accounted for which are exceptions are defined as unexpected exceptions. The real process model in the system is then a combination of the Pre-designed Process Model and the unexpected exceptions added to it. In the figure the process towards defining what the real process model is is shown through three cases that have been extracted from an event log. Each case has a certain order of activities that occur and the combination of these cases is modeled in the real process model which is supposed to allow for all of the cases' paths to be possible.

Figure 1: distinction between unexpected and expected exceptions and the core process flow in a business process model with an example

Without a pre-designed process model, distinguishing between expected and unexpected exceptions becomes difficult. In that case, if no domain expert is there to point them out, one will have to make assumptions on what are exceptions and what the core process flow or main flow really is, e.g. with 80% of the cases running through a certain set of routes this combination can be considered the core flow, which is defined as the most desirable route in business process flow. The validity of measuring the effects of exceptions on KPIs does, however, go down severely once exceptions are exceptions by interpretation rather than exceptions by deviation from the pre-designed process model.

1.3 The Definition of Process Mining in this Thesis

To discover exceptions process mining is a tool in order to discover the process model derived from the system and to check its conformance with the process model that has been pre-designed (or interpreted as main flow) (Samalikova, Kusters et al. 2011, Jones 2012, Van der Aalst, Adriansyah et al. 2012). Before determining what effects exceptions have on business process performance measured with KPIs, and therefore generating input to design better business process models, process mining and other steps in business process re-design in relation to analyzing exceptions in this thesis should be made clear.

As mentioned in Section 1.1, Gartner describes process mining as "a set of techniques and tools for uncovering undocumented process paths in existing systems, thereby speeding up the identification of "as-is" processes. It is invaluable in zeroing in on wasteful exceptions and, ultimately, in optimizing processes." (Jones 2012). Van der Aalst, Adriansyah et al. (2012) see process mining as a broader concept in which the "process mining" as defined in Gartner is only seen as part of this broader concept of process mining. Van der Aalst, Adriansyah et al. (2012) state that Gartner's process mining is merely the process discovery due to Gartner interchanging the use of process mining and automated business process discovery and the actual broader sense of process mining includes two other steps. The first additional step is conformance checking, which is defined as checking for discrepancies between pre-designed process models and actual process models after they have been operational. The other additional step is model enhancement which is re-designing the process model after diagnostics have shown discrepancies and opportunities to improve the process model. In this thesis Gartner's definition of process mining is used, which means it is the (automated business) process discovery using data sets that contain events in PAISs. A distinction of process mining as defined by Van der Aalst, Adriansyah et al. (2012) and Gartner (Jones 2012) is illustrated in Figure 2.

Figure 2: Process Mining as defined by Van der Aalst et al. (2012) and Gartner (Jones, 2012) with the latter used as definition in this thesis

Process mining is becoming a more and more accepted way to analyze business process models derived from data sets (Jones 2012, Reichert and Weber 2012, Van der Aalst, Adriansyah et al. 2012) and is thus relevant as analysis tool to analyze flexibility in business process models. There are several Process Mining vendors which provide good tools in order to analyze data and to derive process models from them. Disco from Fluxicon is relevant as its fuzzy mining algorithm is appropriate for large data sets from real-life cases and the software provides the needed tools to manipulate the data for analysis in a very intuitive way (Günther and Van Der Aalst 2007). The Process Mining software mines event logs, which contains at least case IDs, activities performed on the cases and the times the activities occurred, for a business process model in a matter of seconds, instantly providing visualization and thus automatically discovering the business process model as illustrated in Figure 3 (Van der Aalst, Adriansyah et al. 2012).

Figure 3: Process Mining visualization – from event log to process model

Despite process mining providing information on what the business process flow in an organization really looks like, the connection between the real world and what is happening within a PAIS is not always clear (De Medeiros, Pedrinaci et al. 2007). It is a challenge to provide a complete overview of the real situation. A holistic overview of business processes can be achieved by including both input from people in practice and input from data in the systems that are used (De Medeiros, Pedrinaci et al. 2007, Reichert and Weber 2012), but as getting input from the obtained undisclosed data provided to be impossible getting additional contextual input is considered out of the scope of this thesis.

1.4 Research goal and questions

From the problem statement an overall goal is derived from which four research questions are derived which, when answered, should aid in achieving the overall goal.

The overall goal is as follows:

> ***Overall Goal***
> *Analyze the effect of exceptions in business process models on process performance measured with Key Performance Indicators and define which flexibility techniques for business process modeling can be linked with exceptions to handle different exception types.*

From the overall goal four sub-goals are identified: firstly a sub-goal aims to familiarize with flexibility techniques as defined in previous researches, secondly another one aims to identify the perceived relevance of exceptions on KPI(s) in practice and how flexibility helps in handling them, thirdly another one aims to identify the actual effects exceptions have on the relevant KPI(s), and fourthly a sub-goal aims to use the previous sub-goals to link the right flexibility techniques with the right exceptions to handle these exceptions. The sub-goals are formed into the following four research questions:

Research Question 1
What are canonical techniques to enable flexibility in business processes according to academic literature?

Research Question 2
What do practitioners claim regarding if and how exceptions of business process flows occur, if and how they should be enabled and which KPI(s) matter when enabling exceptions?

Research Question 3
What is the actual measured effect of exceptions in business process flows on the KPI(s) that matter to practitioners?

Research Question 4
How and which flexibility techniques can be linked to exception types that have been measured by answering RQ3?

There has been some extensive research already on flexibility techniques in PAIS, which means that a literature study can provide insights into these techniques and answer the first research question. The extent to which business process flexibility and exception handling are relevant and how the exceptions' influence on KPIs is relevant, however, should be gained from people working in the field of Business Process Management as exceptions in the contexts of measured effects on KPIs have not been researched as far as the author of this thesis is aware of it. Based on input from domain experts their claims on exceptions, KPIs they consider relevant to measure, and relevance of incorporating flexibility to handle exceptions provide a basis for hypotheses which are used for further scoping of this research. The hypotheses are provided by answering the second research question, which in return can be tested quantitatively using process mining techniques and statistical analyses which in return answers the third research question. The fourth research question is then answered by using results from the previous research questions.

Research questions 1 and 2 (RQ1 & RQ2) are thus answered through qualitative means while research question 3 (RQ3) is analyzed quantitatively and thus measured through metric means. Research question 4 (RQ4) then goes into how the information derived from the answers to RQs 1, 2 and 3 can be linked as relevant output. For all four RQs the way to getting answers to them is provided in section 1.5.

1.5 Research Design

Now that the research goal and the research questions used to reach the research goal have been defined, a research design provides a framework to answer the research questions. The research design which is used is illustrated in Figure 4.

Figure 4: Research design used to answer the research questions from Section 1.4

Theory on flexibility techniques as available in literature is outlined through the results of a literature study in Chapter 2, answering RQ1. RQ2 is answered in Chapter 3 using the output of the Case Study Interviews, i.e. the hypotheses regarding what the effects of exceptions in business process flows are on KPI(s). The flexibility techniques from the literature study provide a means to scope the case study interviews. The relevant KPI(s) will also be determined in Chapter 3. RQ2 then also provides means to scope for RQ3 by determining the relevant KPI(s). RQ3 is answered using the output of Process Mining & Statistical Analysis, i.e. the results of what the effects of exceptions are in business process flows on the KPI(s). RQ4 is then answered by linking the canonical flexibility techniques found in Chapter 2 with the exception types discovered in Chapter 4, thus providing the last step to reach the overall research goal presented in Section 1.4.

1.6 Thesis Structure

With the research design explained, this section elaborates on how this thesis is structured. Figure 4 in Section 1.5 shows which part of the research design is presented in which chapter, and additional chapters are mentioned to provide an overview of the entire thesis structure.

Chapter 2 provides the results of the literature study on canonical flexibility techniques in BPM, mainly based off of Schonenberg, Mans et al. (2008). In Chapter 3 the results of the case study interviews are provided. Chapter 4 elaborates on the results of the analysis through process mining and linear regression. This part of the research is the biggest step in the research design with the most information to be processed. In order to properly structure the output of this part of the research, the focus lies on the context of the data sets, including an explanation on how the process models are interpreted, and the end results of the linear regression analyses. Graphical and tabular output of the regression analyses are put in the appendices, but the tables showing the end results of all regression analyses are shown. The link between flexibility techniques and exception types regarding how exceptions should be handled and how flexibility can help is left for Chapter 5. Each of these chapters contains an introduction with a more detailed elaboration on the approaches per step as presented in the Research Design of Section 1.5.

In Chapter 6 conclusions are derived from the results of Chapters 2 to 5. There the answers to the research questions of Section 1.4 are mentioned and a conclusion is made on whether or not the research goal has been achieved. Furthermore limitations that reduce the validity of this research are pointed out. To cover the limitations further research that can aid in taking away the validity-issues are provided and possibilities to further explore new areas as a result of this study are pointed out.

Appendices and references can be found at the end of this thesis.

2. Taxonomy of Flexibility Techniques in Business Processes

This chapter is the result of a literature study on canonical flexibility techniques in business processes. It provides a taxonomy of flexibility techniques in business processes which is used to look at how exceptions are handled according to literature. The paper of Schonenberg, Mans et al. (2008) is used as basis for this taxonomy along with additional literature found in the literature study. Section 2.1 provides the methodology that is followed for the literature study and Section 2.2 contains the results of the literature study thus answering research question 1.

2.1 Literature Study Methodology

In this section the methodology used for the literature review is explained. It is based off of the guidelines for ICT related literature research portrayed by Kitchenham (2004), which uses literature research methodology from the medicine field and transfers that to the ICT field, the field this literature review belongs to. The approach followed in this literature study to find relevant literature is illustrated through Figure 5.

Figure 5: search protocol implemented for this literature review

The review objective is to find canonical flexibility techniques of business processes in literature. Based on that key terms are derived, which after a quick search on Google Scholar lead to a first bunch of papers. Based on those search terms are determined as well as top researchers in the field who are often mentioned. With relevant online academic libraries determined, together they provide the basis for executing a final search.

The final search includes a number of inclusion/exclusion criteria to increase the quality of the final papers which are used as a basis to provide a categorization of flexibility techniques. First of all a study design hierarchy is chosen in which levels are defined indicating the level of depth in which a paper covers flexibility techniques in PAISs. The papers in the lowest level are excluded. For more detailed information on what the levels entail, one can be referred to Van IJzendoorn (2013). Another inclusion/exclusion criterion is the average number of citations of a year the paper was published in. A lower number of citations indicates a lower relevance of the paper to the field and therefore the papers with a below average number of citations are omitted. As Schonenberg, Mans et al. (2008) is used as a basis, only papers since 2008 are considered. The inclusion/exclusion criterion is depicted in Figure 6. As a third inclusion/exclusion criterion, between papers with a great similarity, content-wise and research group-wise, the older publication is omitted.

Figure 6: citations filter applied to literature which passed the study design hierarchy filter

After relevant literature is found, i.e. the result of the approach followed in Figure 5, including the inclusion/exclusion criteria, that literature is structured in a framework based off of four flexibility categories defined by Schonenberg, Mans et al. (2008) (Flexibility by Design, Deviation, Underspecification and Change) and other flexibility categories derived from the literature itself as is shown in Figure 7. In the end only one additional flexibility category is identified among the relevant literature, which is Flexibility by Configuration. Lastly, as can be seen in Figure 7 a number of flexibility techniques is allocated to the flexibility categories.

An elaborate explanation of the literature study approach can be found in Van IJzendoorn (2013). An explanation of the flexibility categories and the flexibility techniques allocated to them resulting from this approach are provided in Section 2.2.

Figure 7: decision tree to categorize papers

2.2 Categories of Flexibility Techniques in Business Processes

Schonenberg, Mans et al. (2008) provide four main categories of flexibility approaches in process modeling. They improve the ability of business processes to respond to changes in their operating environment without necessitating a complete redesign of the underlying model. They differ in the timing and the manner in which they are applied. The categories are also applicable to both imperative and declarative approaches. Imperative approaches focus on how a task must be carried out with the order of the tasks explicitly defined. Declarative approaches focus on what should be done rather than how, using constraints to restrict task execution orders. Another flexibility category, Flexibility by Configuration, is not covered in Schonenberg, Mans et al. (2008), but it receives significant coverage in some of the newly found papers (Van IJzendoorn 2013). The five flexibility categories are:

1. Flexibility by Design
2. Flexibility by Deviation
3. Flexibility by Underspecification
4. Flexibility by Change
5. Flexibility by Configuration

Each flexibility category's section first describes the category and consequently the technologies to incorporate this type of flexibility into a process model. The first four categories are explained using Schonenberg, Mans et al. (2008) and Flexibility by Configuration by using the literature that was found for this literature study.

2.2.1 Flexibility by Design

Flexibility by Design answers the flexibility issue at the design time of a process model, encoding flexibility possibilities offering several execution paths of which the most appropriate one can be selected at run time. The most common options to realize flexibility by design are:

- Parallelism: executing tasks in parallel
- Choice: selecting one or more tasks as next execution from a set of available tasks
- Iteration: the ability to execute a task repeatedly
- Interleaving: the ability to execute each of a set of tasks in any order so that they do not execute concurrently

- Multiple instances: the ability to execute multiple concurrent instances of a task
- Cancellation: being able to withdraw a task from an execution at any time now or in the future.

2.2.2 Flexibility by Deviation

Flexibility by Deviation touches the necessity of flexibility in the operating environment at run time when deviations of the regular process model are necessary. This is done without altering the original process model and can only encompass changes to the execution sequence of tasks in the process for a specific process instance. The following options characterize support for deviation by imperative languages which indicate how the program should take actions:

- Undo task: moving control to the moment before the task was carried out, but without the carried out task reversed or actually undone in the literal sense of the word undo.
- Redo task: executing a disabled, but previously executed task without moving control. It provides the ability to repeat/improve/update an already executed task.
- Skip task: moving control to the point after an enabled task, effectively skipping that task.
- Create additional instance of task: at the moment of task instantiation some instances are created. Next to that an additional instance can be created in parallel.
- Invoke task: allows a task to be started before it is enabled while it also had not executed yet.

For declarative languages, which indicate how the program should act, a violation of constraints is seen as a way to be flexible. However in Schonenberg, Mans et al. (2008), the full range of these violations is out of the scope of their research. Yet for evaluation purposes the overall range is considered as a flexibility technique.

2.2.3 Flexibility by Underspecification

Flexibility by Underspecification focuses on the cases where future paths cannot be specified yet. It allows a process model to be executed at run time while it is still incomplete. It does not allow for any change to be made on the defined tasks, but it leaves flexibility for the future so that the process model can be completed while an instance of the process has been executed already.

There are two ways of underspecifying a model. They are kept underspecified through so-called placeholders, which are nodes marked as underspecified. Besides placeholders, there are also two moments for realization during the process execution, which are recognized.

The two placeholders are:

- Late binding: one process fragment can be chosen from a selection of predefined process fragments during process execution. The restriction is that the approach does not allow for new process fragments to be added to the selection.
- Late modeling: a process fragment is constructed from scratch or by combining several existing ones in the placeholder which is kept in place to indicate a specific but still unmodelled task is to be carried out there.

Realization types:

- Static realization: the moment where the process fragment chosen to realize the placeholder during the first execution is used to realize the placeholder for every following execution

- Dynamic realization: where the realization of a placeholder can be chosen once more for every following execution of the placeholder

These can be subdivided into two techniques each: static realization before and at placeholder respectively, and dynamic realization before and at placeholder.

2.2.4 Flexibility by Change

Flexibility by Change deals with those situations when during execution it is discovered that the process model must be different at some points. This is different from deviation as it changes the process itself. Flexibility by Change is thus the ability to modify a process model at run time, so that all currently executing process instances will be transferred to the new process model which is newly made at design time.

There are three variation points defined for Design by Change. Firstly Effect of Change which defines whether the changes are performed on process model level or process instance level. Secondly Moment of Allowed Change specifies the moment at which the changes of a process model or process instance(s) can be introduced. Thirdly Migration Strategy specifies what needs to be done with the running process instances that are affected by an evolutionary change.

Effect of Change types:

- Momentary change: the change affects the execution of a selected number of process instances, but does not affect future process instances.
- Evolutionary change: a change made in the process model that affects all new process instances

Moment of Allowed Change types:

- Entry time: a change can be performed only at the moment the process instance is created. For momentary change this means only the given process instances are affected and for evolutionary change this means that the running instances are not affected but the newly created process instances are.
- On the fly: changes can be performed at any time during process execution. For momentary change this means the process instances can be customized during execution. For evolutionary change this means that the new process instances will be affected, but also the running ones will have to be migrated to the new process model.

Migration Strategy types:

- Forward recovery: affected process instances are aborted.
- Backward recovery: affected process instances are aborted (compensated if necessary) and restarted.
- Proceed: old process instances continue as if the changes did not happen, whereas the new ones will be affected by the changes.
- Transfer: the old process instances are transferred to a corresponding state from the new process model

2.2.5 Flexibility by Configuration

With Flexibility by Configuration at configuration time, after build time and before run time, the extent to which the model can be configured before run time is decided. This gives a wide variety of process variants which at configuration time are selected (Weber, Reichert et al. 2008). At configuration time a reference model is used as initial workflow model in which variants can be chosen to change the reference model into the desired to-be model (Gottschalk, Wagemakers et al. 2009, van der Aalst 2009, Hallerbach, Bauer et al. 2010, Weber, Reichert et al. 2010, Meerkamm 2012). The reference model, a standardized process model which was built at build time, is essentially tweaked during configuration time, having entire process variants fitting to specific situations.

Process variants are the different process models a configurable reference model, a family of process models, can lead to. They are dependent on the process context which results from the requirements of the specific situation (e.g. in car production a different model would lead to a different context of the car manufacturing process) (Hallerbach, Bauer et al. 2010).

In Gottschalk, Wagemakers et al. (2009) and van der Aalst (2009) three techniques are mentioned to incorporate Flexibility by Configuration, *block*, *hide*, and *enable* (or "allowed"). The inflowing (join) and outflowing (split) paths' connection to an action are called ports. They depend on the splitting behavior indicated by AND (joint inflow or outflow), XOR (one of several in or outflowing paths can be chosen), or OR (a specific number of in or outflowing paths are to be taken). The ports are the configurable elements (see Figure 8 and Figure 9) and act as *switches* which can be turned on or off.

Port switches:

- Block: input port prevents a task or action from being executed whereas a blocked output port prevents corresponding output paths from being triggered (Gottschalk, Wagemakers et al. 2009).
- Hide: the hidden input port prevents a corresponding action from being executed, but the process continues thereafter (Gottschalk, Wagemakers et al. 2009). This way of skipping is different from the skip for one process instance at Flexibility by Deviation as its resulting process variant has been set up at configuration time before run time and continues to skip the action after configuration.
- Enable: without being blocked or hidden, the action is enabled and will be part of the process in the process model (Gottschalk, Wagemakers et al. 2009).

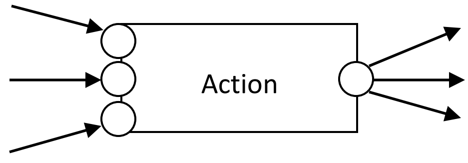


Figure 8: XOR/OR join, AND split

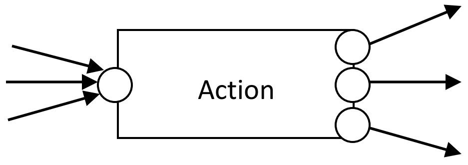


Figure 9: AND join, XOR/OR split

Hallerbach, Bauer et al. (2010), (Meerkamm (2012)), and (Weber, Reichert et al. (2010)) identify *insert* as a technique which adds a given action to a process model on a specified spot. They also identify *move* as a way to adapt a model by changing the location of an action. On top of that they mention *delete* as a means to adapt a configurable reference model into a process variant by removing an action. These techniques act as animations from the configurable reference model towards a process variant. By looking at the amount of animation steps (insert, move or delete) from a reference model to a process variant, the average change distance can be measured. By keeping the process model's average change distance to its variety of process variants at a minimum, the best possible configurable reference model can be found (Weber, Reichert et al. 2010).

Animation techniques:

- Insert: add a given action to a process model on a specified spot.
- Move: adapt a model by changing the location of an action to another place in the process model.
- Delete: remove an action from the process model entirely.

3. Case study interviews to determine hypotheses

The purpose of this part of the research is to come to hypotheses on claimed effects of exceptions on KPIs and thus indirectly the effect of enabling them through flexibility techniques on KPIs. The case study interviews help to funnel the scope of the research. The analysis of the interviews shows whether or not flexibility was relevant in the cases the interviewees worked on when it comes to enabling or preventing exceptions. It also shows which KPIs matter.

The case study interview approach is based off of Voss, Tsikriktsis et al. (2002), Gibbert, Ruigrok et al. (2008) and Dul and Hak (2008) which is explained in Section 3.1. Sections 3.2 to 3.5 focus on the data collection preparation for the case study interview with 3.2 going into case study context and research framework, 3.3 going into the interview questions, 3.4 going into case selection and 3.5 into research instruments used. Data collection, checking and presenting as next step is divided into sections 3.6 to 3.8 with 3.6 focusing on doing interviews and transcribing them, 3.7 on checking for data completeness and reviewing the interviews, and 3.8 on coding the data. Sections 3.9 and 3.10 provide respectively the analysis and hypotheses resulting from the case study interviews. Research question 2 will then be answered through this section.

3.1 Introduction to the Case Study Interview Approach

Voss, Tsikriktsis et al. (2002), Gibbert, Ruigrok et al. (2008) and Dul and Hak (2008) indicate how to properly utilize a case study methodology. They have some overlap and complement each other, emphasizing different perspectives in an overall similar (generalized) action plan. This section combines these in order to generate a complete protocol in which all components of the different perspectives have been included. Only the relevant steps are explained as for example the first steps of implementing a literature study and defining the purpose of the study – and thus the theory preparation – have already been done in Chapter 2.

The case study research protocol is divided into three main steps:

- Data collection preparation
- Data collection, checking and presenting
- Analysis

This is shown in Figure 10 which in Figure 4 in Section 1.5 (the research design framework) represents the Case Study Interviews step.

Once the research framework and case study context have been established, the questions for the interviews can be prepared based on the literature study. Besides that a selection of candidate cases is made. That is; the interviewees or informants that will be approached should be decided upon. Finally as preparation the research instruments, i.e. the material with which the interviews are conducted, are determined.

For the data collection, checking and presenting, data is collected by conducting the interviews and then transcribing the interviews. Once the interviews have been transcribed, the content can be verified for completeness and correctness by the interviewees themselves. Afterwards the texts will have to be coded in order to perform data mining to detect relations of variables. The coded data can then be presented in a table or graph.

Finally, the interviews are analyzed using the coded data and based on the analysis hypotheses can be formed.

Figure 10: Case study interview protocol model

3.2 Case study context and research framework

The interviews come from five different projects and seven people in total. The projects each come from a different industry and have people in different roles in the projects. This is to find common grounds for organizations in different industries on how they handle flexibility.

The choice to select only people from Accenture as key informants is a necessary convenience-based decision as;

- there is direct access to people involved in business process management in Accenture,
- they have a wide variety of experience and work internally with the clients,
- and there is no direct access to clients, but there is access indirectly through Accenture consultants who worked in service for them.

Gibbert, Ruigrok et al. (2008) call an internal benchmarking in case studies between departments of one company a *nested case study approach*, which is the alternative to a *multi-case study approach* which handles cases from different organizations. The reality of this case study research is in between that, as the projects do take place at different organizations, but they are performed by people within one company. So for clarity, to call it what it is, the approach in this case study research is called a *project-based case study approach*.

3.3 Interview questions

The research framework is that of a case study research for theory building. The questions can be found in Table 1 (English) and Table 2 (Dutch). The first question ensures the contexts of the projects are clear. The interviews themselves provide insights, but later on more information might be asked by email in order to find what environmental factors might have influenced the effects of the independent variables on the dependent variable. The context provides information on the industry the project was in.

The next questions have been based on the literature study in Chapter 2, explaining categories of flexibility techniques in PAISs. Five categories have been defined and each of them is used as input for a question as independent variable for the hypotheses that will further provide focus on the quantitative part of this research.

The last two questions are based off of finding out the effects on business goals, and thus finding out what the dependent variables are. This makes the first question fall under non-causal type of questions as defined in Voss, Tsikriktsis et al. (2002). Questions 2-6 are non-causal situational, and question 7-8 are causal according to Voss et al’s (2002) question types.

The hypotheses will have to be derived after coding the answers to the questions and as information gaps appear, more questions are asked later to the key informants by email to ask for the team size and project duration to provide more information on the context.

Table 1: Questions used for case study interview (English)

#	Questions
	Context elaboration
1	Could you describe what the purpose of the project was and how workflow management was involved, as well as your role in the project?
	Management of irregularities from the standard way of working in the client's organization
2	Were irregularities/exceptions accounted for before the implementation of the company's workflow standard and how?
3	Were there times irregularities/exceptions occurred which were NOT accounted for in the company's workflow standard and how did you handle them?
4	Can you name instances where activities of a process were specified at run time, e.g. because the workflow system had to go live already?
5	How did you change the old workflow standard to the new workflow standard?
6	Did you provide process variants out of a repository which the client (or you on behalf of the client) could configure?
	Consequences of (lack of) presence of flexibility in BP Models
7	How did you determine or measure the effects of workflow irregularities/exceptions on business goals or SLAs?
8	What about the solutions to the irregularities/exceptions?

Table 2: Questions used for case study interview (Dutch)

#	Questions
	Context elaboration
1	Kun je omschrijven wat het doel van het project was waar je aan werkte en hoe workflow management daar een rol bij speelde? Welke rol had jij in het project?
	Management of irregularities from the standard way of working in the client's organization
2	Werd er rekening gehouden met mogelijke onregelmatigheden, uitzonderingen of afwijkingen vóór de implementatie van de workflow standaard en zo ja, hoe?
3	Kwamen er onregelmatigheden, uitzonderingen of afwijkingen voor waar je van te voren geen rekening mee had gehouden in je proces standaard en hoe ging je daar mee om toen ze plaatsvonden?
4	Werd een processtandaard/standaard werkwijze al geïmplementeerd voordat een aantal activiteiten nog niet duidelijk waren gespecificeerd, zodat je de activiteiten later alsnog kon specificeren? Hoe werd daarmee omgegaan?
5	Als je een nieuwe standaard werkwijze/workflow wilde implementeren, welke stappen heb je ondernomen zodat de nieuwe standaard workflow in gebruik werd genomen?
6	Was er een set proces varianten uit een repository die je kon configureren naar de wens van de klant en hoe paste je deze aan naar de wens van de klant?
	Consequences of (lack of) presence of flexibility in BP Models
7	Hoe had je bepaald of hoe had je gemeten wat het effect was van onregelmatigheden, uitzonderingen of afwijkingen in de processtandaarden op de doelstellingen die gehaald moesten worden (SLAs)
8	En hoe deed je dat nadat je de processflow had verbeterd?

3.4 Case selection

The selection of cases is convenience-based and goes in three concurrent steps after which people are approached as depicted in Figure 11. As a starting point internally within the department in which this research project is operating, names are requested of people who have worked on projects related to business process management (step A). These people can then also forward to others. Furthermore concurrently requests are sent on the internal portal of Accenture to which all Accenture employees have access (step B). By tagging the request with keywords, people who have indicated an interest for such topics are able to see them. Thirdly a list of names is set up through this same portal in which a database can be found of Accenture employees (step C). The first 100 people, who indicated “business process management” as a skill within the Amsterdam office to favor face-to-face meetings, are selected and put on the list.

Figure 11: model depicting the actions that were taken to obtain sources of people to interview in Accenture

Once steps A-C have been done, a selection of cases is made, based on the willingness of people to give time for an interview. The most effective approach was step A with six out of seven informants being selected. Although some of them had overlap with the list from step C, they had first been contacted via others within the organization. One informant responded to the action from step B.

3.5 Research instruments

In order to gather more in-depth knowledge interviews are chosen as the method for data gathering in this case study approach. As Accenture consultants are generally caught up in a ton of work and their time is very precious, the scheduled meetings were set for approximately half an hour each. This also helped in convincing potential key informants to schedule time as one hour seemed too much.

Two types of interviews were held: face-to-face and by phone/VoIP. Only one informant happened to be available for a face-to-face meeting and others were not able to meet in person and called. In all cases the conversation was recorded using basic sound recording software. The sound record was then transcribed to text for verification and then analysis purposes, as well as for the reliability of this research.

3.6 Conduct and transcribe interviews

The data was collected over a course of approximately two weeks with seven interviews being spread over these weeks. This left room for the sound records to be immediately transcribed and to also notice information gaps if they might occur. None were immediately discovered, however after the interviews a first overview provided insights into the necessity to gather more data; size of the project teams and duration of the project.

The size of the project team as well as the duration is used as an indicator for project size (and costs involved). The bigger the project team and the longer it takes to finish it, the bigger the project size.

3.7 Check data completeness and review data

To check whether or not the data is complete as well as for the transcribed interviews to be reviewed, the transcripts are sent to the key informants themselves for verification. This was done within one week of the interviews taking.

In four of the seven cases an answer was received regarding the content being correct or regarding sentence structures that was meant differently. These were mainly nuance differences if any correction was done at all.

3.8 Data coding

To code the data one should provide information about the type of cases the hypotheses that result from the case study are based on. This information can then also be used as basis for more investigations on flexibility in business process management in the future. Context-based information extracted from the interviews and later emails is team size, length of the project and the industry context. Additionally the country where the project took place is added.

The factors to be analyzed when it comes to being an indicator for success are the five flexibility categories from Chapter 2. As this is a case study and the key informants are homogeneous as they come from one organization, though working in-house at different companies, it should be made clear that not having incorporated a flexibility technique of a certain category in these cases negates its effect on achieving business goals. They do help in forming hypotheses which can then be tested through different means. When coding the flexibility categories are shown as present or not within the business process models of the projects.

Business goals are investigated through the last questions in the interviews. One goal that came forward consistently was that of reduction of throughput of business processes. The reduction of exceptions was also mentioned, but mainly as a mean to improve business processes (by reducing throughput). The follow-up question posed in the interview if not already implicitly answered was whether or not the new system improved the business goal. This last question has been coded by the effects having been positive or not. Once again a big criticizing note should be given as the informants answering the questions have a large interest in their own work being successful in the projects. That can provide a bias, and on top of that some projects have not been running for long or have only been tested for success without having been implemented yet.

The coded data is presented in Table 3.

Table 3: coded data from interviews

Context	Type of company	Consumer electronics leasing company	Bank	Bank	Quango	Insurance customer service for military
	Country	USA	Netherlands	Netherlands	Netherlands	USA
	Team size	10	17	5	10	10
	Project time (years)	0,42	2	0,33	0,67	1
Indicators	Incorporated Flexibility by Design	Y	Y	Y	Y	Y
	Incorporated Flexibility by Deviation	Y	N	N	N	Y
	Incorporated Flexibility by Underspecification	N	Y	N	N	N
	Incorporated Flexibility by Change	Y	Y	Y	Y	Y
	Incorporated Flexibility by Configuration	N	N	N	N	Y
Effects	Reduction of exceptions	Y	Y	Y	Y	Y
	Reduction of Throughput	Y	Y	Y	Y	Y

3.9 Analyze data

As can be read from Table 3 in the previous section, in every project flexibility was incorporated in the design phase and they made room to incorporate Flexibility by Change. Every project team's task was to design and implement a for the company new system. There they often already built in the option for flexibility and also to restrict the movement of process instances to what was designed. Exceptions were seen as something to be avoided and only to be taken for a next evaluation round if the exceptions were causing too much of a loss on profit. Exceptions, coming from the context of the interviews, were mainly seen as deviations from the core process flow, which is also the definition as it is used for this research.

The aim coming with standardization was to reduce complexity in all cases which are caused by exceptions and all cases aimed to reduce throughput time as well. Furthermore when looking at Table 3: coded data from interviewsTable 3 there was consistency in Flexibility by Design and Change to incorporate flexibility and other categories were used as well. Despite some categories being more popular than others, all have in common that they were considered as a necessity when one would rather prevent the necessity by incorporating standardization through automation. All indicators can therefore be aligned with one common and underlying variable which was considered important, namely that of standardization. Through the standardization mainly the easy majority of the cases should be considered. As was mentioned in one case; you should not invest 20% of your time to design a workflow management system for 99% of your customers and then invest the remaining 80% of your time for the last 1%, *"because if you will design for the last 1% of the population, then it will become too expensive no matter what. And even though you might gain a little from that one percent, that is not where you will get the margins."*. Adding to that that complexity increases throughput time and standardization reduces throughput time (Griffin 1997, Münstermann, Eckhardt et al. 2010) the conclusion is drawn that the interviewees aimed to reduce the number of exceptions so that the process flow remains less complicated, the majority of cases goes through a standardized process flow and the result is that throughput time is reduced. The inverse of that conclusion is thus that they aim to reduce exceptions and by doing that reducing throughput time.

3.10 Hypotheses

As a result from the analysis of the coded interviews, one hypothesis is generated from which several sub-hypotheses are derived which are explained in this section.

Main Hypothesis: Exceptions in business process models cause an increase of throughput time.

Sub-hypotheses that are derived from that are based on the literature study in Chapter 2 regarding flexibility techniques in business process models. Firstly, literature points out the distinction between exceptions that were accounted for in the design of a business process model and exceptions that have not been accounted for, respectively expected and unexpected exceptions, as is indicated in Section 1.2. Furthermore, as indicated in Chapter 2, there are several flexibility techniques per flexibility category. The effects that exceptions have can be a reflection of the flexibility technique or exception type they belong to.

Two sub-hypotheses are derived in relation to exceptions having been added to the pre-designed process model or not (Sub-Hypotheses 1 and 2), and another one (Sub-Hypothesis 3) in relation to the influence of an exception belonging to a specific type. After all, as the interviewees in the case studies specified the need to simplify the process flow and reducing exceptions as much as possible as well as the need to reduce throughput time as much as possible, it is relevant to know what specific exception types have what kind of influence on throughput time. The Sub-Hypotheses are as follows:

Sub-Hypothesis 1: Unexpected exceptions in business process models cause an increase of throughput time.

Sub-Hypothesis 2: Expected exceptions in business process models cause an increase of throughput time.

Sub-Hypothesis 3: The influence of an exception in a business process model on throughput time depends on the type of exception.

Sub-Hypothesis 3 can be further broken down into the specific influence a certain exception type has on throughput time. Five exception types are found in different combinations in five different data sets that have been analyzed as described in Chapter 4: skip, iteration, loop, analyze and early exit.

Skip and *early exit* are exceptions of which one would intuitively assume that it causes a reduction of throughput time. With skip a jump is made to a later point in the process model than where a case would go if it would simply sequentially follow the next activity. Early exit means that from any point in the process model a case follows an early exit route after which the process flow for this case is ended.

However, skipping activities can cause a serious delay for the next activities performed due to a poorly processed case, or even a different exception type: iteration where the case is sent back in the process flow to have skipped work done anyway. Naturally this would then cause the throughput time to be increased. Still as skip exceptions mean a jump to a later point in the process model, it is hypothesized that generally skip exceptions cause a reduction in throughput time.

Early exit might also intuitively result in reduction of throughput time, but once a case is taken out of the regular process flow, the early exited case might not receive much attention anymore as it might be e.g. a lost customer which is put very low on the priority list of customers to be processed. Though e.g. cancellations, which are a type of early exit, is assumed to cause a reduction of throughput time as it causes a case to be stopped before it reached the end of the process flow if it would have not been cancelled.

Contrary to skip and early exit, *iteration* and *loop* would intuitively cause an increase of throughput time as a jump to the same spot or back to an earlier place in a process model is made. The exception of *analyze* however could intuitively very well lead to a reduction or increase of throughput time, as analyzing a case can result in the discovery of e.g. an incident which has been resolved previously, thus effectively becoming a skip-by-analysis. Alternatively an analysis can result in a case having to be rerouted to an early activity as it is discovered that e.g. an incident was not documented properly, effectively generating iteration-by-analysis.

Many factors can co-cause a certain effect on throughput time in a process flow and the more factors there are, the more one would need to go into detail on case-level to determine what caused these effects. Though based on the previous reasoning the direction – albeit an increase or decrease – of the influence of a type of exception on throughput time can be hypothesized, thus resulting in sub-hypotheses to Sub-Hypothesis 3:

Sub-Hypothesis 3a: The exception type “skip” in business process models causes a reduction of throughput time.

Sub-Hypothesis 3b: The exception type “early exit” in business process models causes a reduction of throughput time.

Sub-Hypothesis 3c: The exception type “iterate” in business process models causes an increase of throughput time.

Sub-Hypothesis 3d: The exception type “loop” in business process models causes an increase of throughput time.

Sub-Hypothesis 3e: The exception type “analysis” in business process models can have both an increasing and a decreasing influence on throughput time.

While Sub-Hypothesis 3 indicates ambiguity regarding the influence that exceptions have on throughput time as one would have to specify the type of exception, this sub-hypothesis does not contradict Sub-Hypotheses 1 or 2 and also not the Main Hypothesis. Given the case-study interviews in which a need for reduction of exceptions as well as simplification was specified, a possibility could be that from experience the interviewees have come to realize that even though some exception types cause a reduction of throughput time, they do not outweigh the increase of throughput time caused by other exception types.

4. Data Analysis of exception effects on throughput with Process Mining and Linear Regression

In this Chapter the results of the quantitative analysis are presented. The approach as explained in Section 4.1 is followed for each case. The cases' contexts are explained per section after which the results are presented and interpreted for the case. Section 4.2 presents the results of a middleware system implementation case, Section 4.3 shows the outcome of a case from the municipality of Eindhoven in the Netherlands, Section 4.4 presents results of a big company in the USA, 4.5 has results of the case of a Dutch financial institute, and in conclusion Section 4.6 has the results of the case done for Swedish car manufacturer Volvo. To analyze all of the cases together to see the overall perspective of the results, Section 4.7 provides an overview of how the results are interpreted, thus providing input to research question 3.

4.1 Introduction to the Analysis through Process Mining and Linear Regression

To test the hypotheses derived from the approach in Chapter 3 several data sets are acquired for a quantitative analysis of the effects of exceptions in business process flows on the throughput time, the KPI that has been determined in Chapter 3 for further analysis in this chapter.

Once the data sets have been acquired the process as described in Figure 12 is followed in order to analyze the data sets for the purpose of answering research question 3. The process followed in Figure 12 has been made more detailed in Figure 13 including a more detailed explanation. In Figure 12 the analysis starts with determining the context of the data. This is to provide insights in how the data should be interpreted and how it should be manipulated for further analysis. The second step is to derive a process model through process mining. Thirdly exceptions can be detected as a conformance check is performed, comparing the mined process model with a pre-designed process model if available and the main process flow that has been determined. As a fourth step, based on the exceptions found, the process model is filtered in a process mining software program, exported and imported into Excel to set up variables that can be used for a statistical analysis. Finally a statistical analysis is performed from which conclusions can be drawn regarding whether the hypotheses from Chapter 3 are confirmed or not.

In both figures a distinction is made between whether a process model which was already designed has been made available. Once it is available, this provides a way to check conformance between what is happening in the system and what should happen according to the pre-designed process model (Van der Aalst, Adriansyah et al. 2012). That consequently provides an indication on whether an exception is expected or unexpected, which is used as input for the next step of data interpretation. To truly define whether an exception is expected or unexpected, input should be given by domain experts who are knowledgeable when it comes to the analyzed processes. The actual difference in the steps needed to analyze the data set between the ones with and the ones without a pre-designed process model come later in the approach as can be seen in Figure 13 by looking at the orange and green colors given to the different steps. Keeping that in mind, the methodology can be explained.

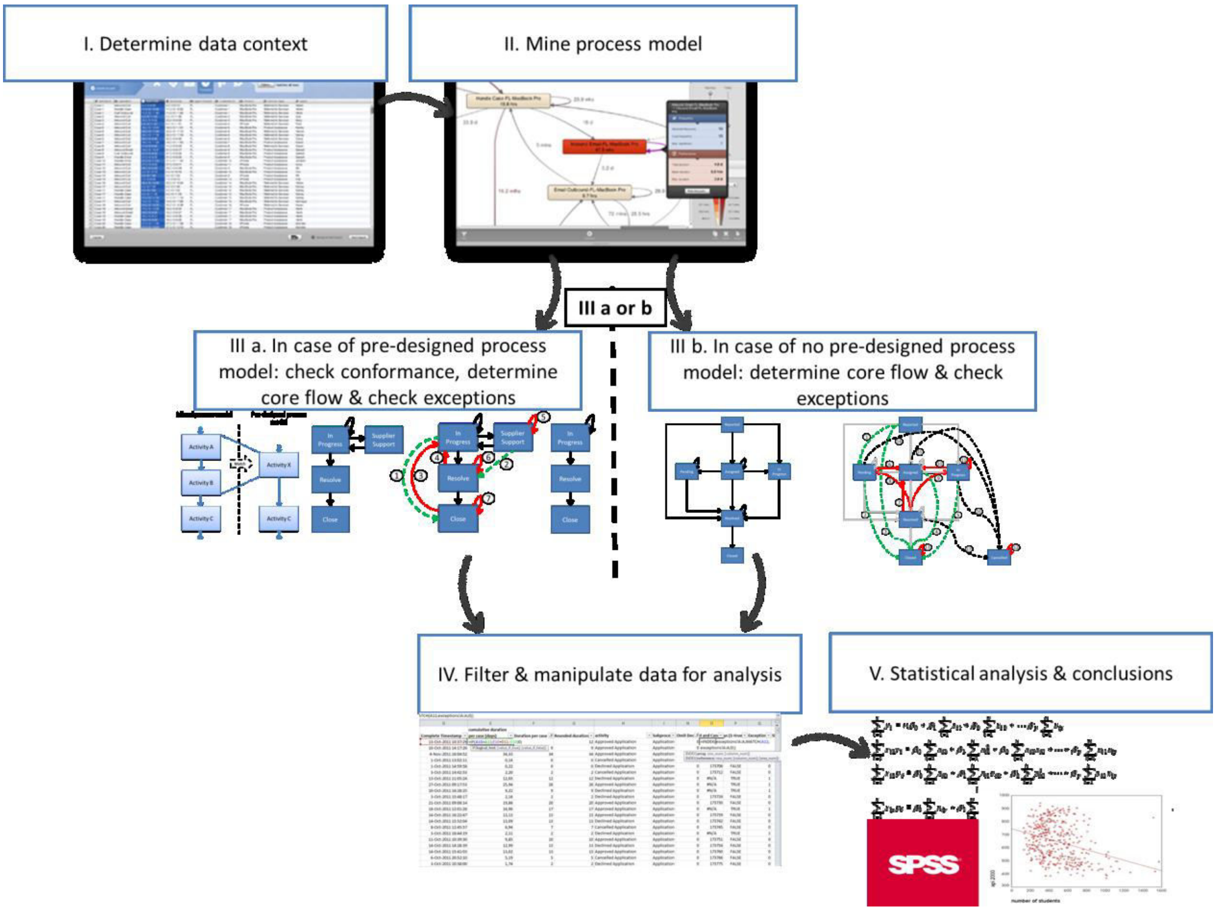


Figure 12: approach used to quantitatively analyze data for effects of exceptions on throughput time using process mining with Disco, manipulating data with Excel, and linear regression analysis with SPSS

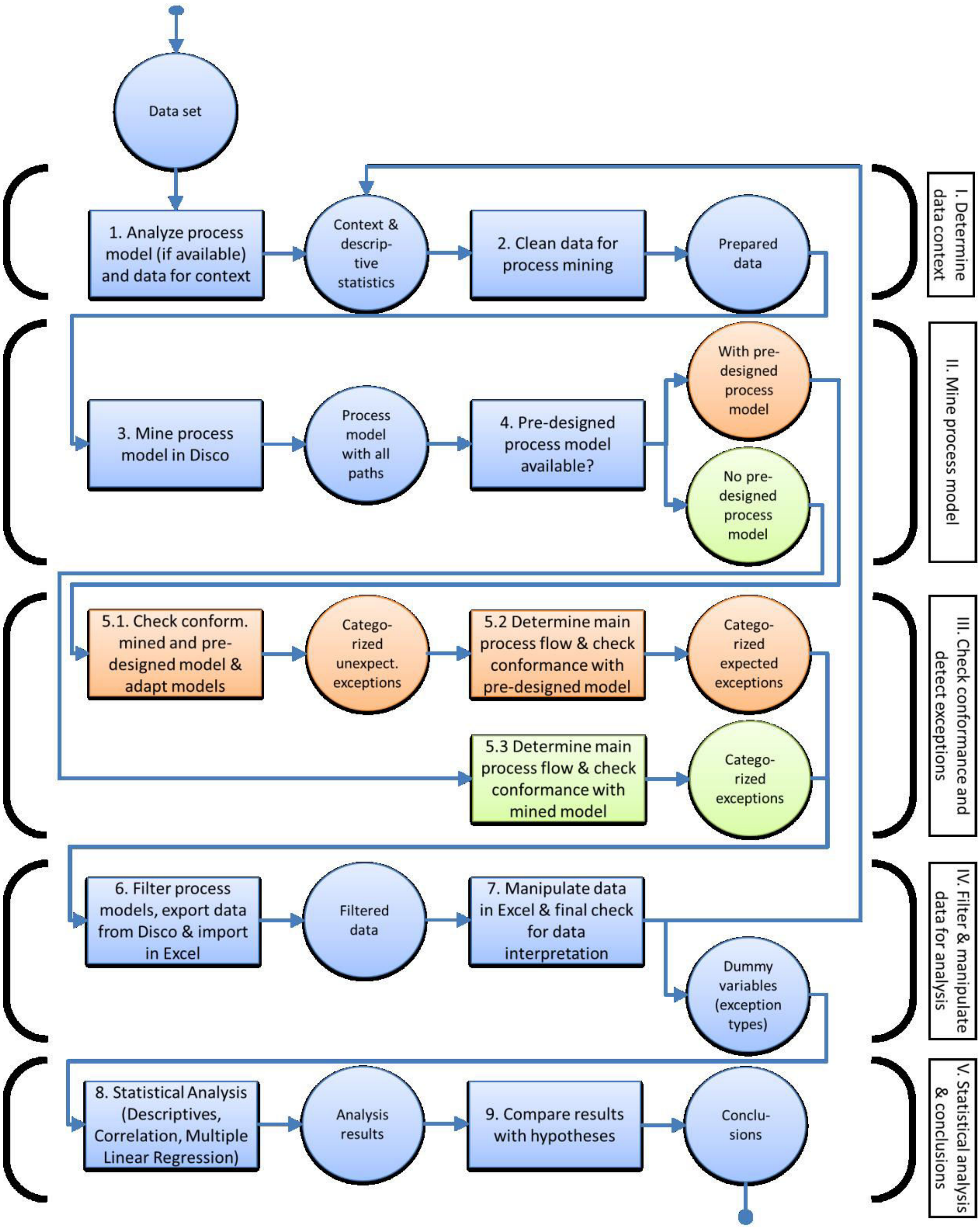


Figure 13: detailed model of the approach followed as depicted in Figure 12

In the methodology for process mining and linear regression, data sets are firstly analyzed for context. Optimal would be if a description of the data is provided in writing, in person through an explanation, as well as by going "on site" to get familiar with the processes as they run. However, in the scope of this research that is near to impossible with limited disclosure of where the data came from. This has as a consequence that assumptions are made when it comes to how the data should be interpreted. By inserting the data in SPSS descriptive statistics can be carried out in order to provide insights, outliers and discrepancies in e.g. throughput time.

Next the data should be cleaned based on the context and how the data are interpreted. To prepare them properly for process mining, event logs should be available. That means that per event that occurs in the system, a case ID, an activity and the timestamp at which the activity on a case was carried out must be available. If this is not available, a process model cannot be extracted through process mining. Often extra contextual information is available, such as resources that were involved in the event.

As the data are prepared for process mining, now the data can be read with a process mining program. For this research the program Disco is used, as explained in Section 1.3. The process is then different for the data sets with a pre-designed process model. For the data sets with pre-designed process model, first the mined process model must be analyzed to see which activities fit to activities from the pre-designed process model as can be seen in the example in Figure 14. This is once again dependent on interpretations and assumptions. Based on the mined process model as well as possible descriptions of the business processes combined, the data set is then further manipulated. If activities in the mined process model are interpreted as belonging to an activity of the pre-designed process model, these activities are renamed in the original data set in order for it to fall under the activity of the pre-designed process model. In the example of Figure 14 activities A and B would be renamed Activity X. Vice versa is, however, also possible where in the mined process model a higher hierarchy Activity X is mined with which a certain activity A and B from the pre-designed process model should be matched.

Figure 14: How activities in process models are matched between mined process model and pre-designed process model

Now that the mined process model and pre-designed process model are synchronized, the unexpected exceptions can be detected. They are identified by looking at paths that do not occur at the pre-designed process model, which is also named conformance checking (Section 1.2). In some cases it is also possible that an activity did not belong to a pre-designed process model activity and has its input and output path(s) all as exception paths. The exceptions are categorized based on similarities and based on these similar characteristics the name is determined.

For both the expected exceptions in the data sets with the pre-designed process models as well as exceptions in general in those without the pre-designed process models the main process flow must be determined. The main process flow is the flow which consists of routes that are considered regular. Without extra information on the business processes themselves, it is difficult to determine whether or not the business process flow which is considered the main flow is in fact the main flow. But based on input from a process mining software developer a process flow which covers 80% of the cases is a rule of thumb chosen to consider what constitutes the main flow and verified as rule of thumb by process mining software developer Fluxicon. In Disco it is possible to filter the process model in such a way that 80% of the cases are still considered. Here it is however assumed that the core flow should be part of the pre-designed process model in case a pre-designed process model is available (Section 1.2). This and other filtering considerations which are executed on top of this rule-of-thumb method can have as a consequence that eventually a smaller percentage of the original cases is covered by the core flow. For the data sets without a pre-designed process model only other filtering considerations will have this consequence. Just like with the unexpected exceptions, here exception types are determined based on similarities in characteristics in the business process model.

With exception paths determined, albeit unexpected, expected or general, and with the types known, Disco can be used to filter the process models in such a way that cases that contain these exception paths are excluded. This can be done with several exceptions at the same time. With the functionality of exporting data, the data that are left after filtering out a specific exception type are exported as Comma Separated Value document, which is put in Excel for further processing.

Using the data set that has been mined for a full process model the extracted filtered data sets can be compared with it in Excel. This can be done using either Excel's functionality of VLOOKUP or the combined functionality of INDEX and MATCH. This way presence of a certain exception is detected, thus resulting in a dummy variable (1 for exception present in a case, 0 for no exception present). With the data sets being as big as they are (i.e. big data), the computations take a lot of processing space, causing every new change made to generate a long waiting time until the computations have been processed. By occasionally switching the calculation options from automatic to manual, it is prevented that even a single click causes such lag. Needless to say, the lag indicates that either the methodology is not optimal and/or that the material and programs are not optimal. However, with the author of this paper being relatively skilled in the mentioned programs and the programs being able to carry out the needed tasks, no other programs were considered. In Chapter 3 it is made clear that throughput time is a consistently important KPI for domain experts. Further manipulation in Excel to determine the throughput time per case must thus be carried out as well.

Next, the data can be (re-)inserted into the statistics software program SPSS. In SPSS the data is once again analyzed for discrepancies, now by also taking exceptions and throughput time into consideration. If here it becomes clear that certain data should have been interpreted differently based on new insights, an iteration is made back to interpreting the context of the data set before it should be once again run in Disco for process mining.

After the data have been cleaned for one last time, a descriptive analysis is done and next the data is analyzed for correlation between exceptions (expected, unexpected or general) and throughput time. After that a multiple linear regression analysis is carried out using the exception types as predictors and the throughput time as dependent variable. For each regression analysis the F-test and adjusted R squared are used to determine goodness-of-fit of the model, and t-tests are used to determine the significance of the individual standardized beta coefficients ascribed to the predictor variables after regression. See also Field (2005) and Hair Jr and Black (2010).

The model is furthermore controlled for violations of regression analysis assumptions. The assumptions for regression analysis aid in giving insight whether or not the model actually predicts the dependent variable even though other tests have indicated a good fit. Violations can give unreliable significant results which means that the relations cannot really be generalized (Field 2005, Hair Jr and Black 2010). The violations provide room for interpretation however.

With the regression analyses performed, the results can be processed in tabular form for a comprehensive overview of the relations between exceptions and throughput time. Based on the results conclusions are drawn which are then compared to the hypotheses. Finally, based on the literature study, matches are made between exception types and flexibility techniques that are able to enable these exception types.

4.2 Software Component Developer for Consumer Electronics – Change Control Board

This section is regarding a Change Control Board (CCB) in a Dutch software component developer for consumer electronics which checks change requests for the migration of configuration items (components and versions) from one software version to the next. The CCB analyzes change requests in order to monitor their status, plan necessary activities in the project and to predict outcomes of the development process.

In Section 4.2.1 the data is mined in order to find out the context of the data. Based on what is found in the data certain assumptions are made and the data will be manipulated for it to be able to be processed through the process mining software and the mined process model with the distinction between exceptions is presented. In Section 4.2.2 results are presented on unexpected exceptions for this data set and in Section 4.2.3 on expected exceptions for this data set.

4.2.1 Descriptive Data Analysis

In this section the data are analyzed in order to determine how they should be manipulated before they are mined for a process model. This data set is accompanied with a pre-designed process model, which means a distinction can be made between an expected and an unexpected exception. But first the data must be filtered for open cases and outliers. This section thus consists of an explanation of which data have been filtered, the exception types that have been identified and a list of the most common process variants.

In order to understand what the closing activity is, the pre-designed process model and the activities in it are explained. The Pre-designed process model is illustrated in Figure 15. It shows a clear ending activity, namely “Concluded”. All cases that do not end with Concluded are therefore omitted from further analysis.

Furthermore, the event log shows unusual behavior as many cases have been closed on the 2006-11-22 where the other dates show approximately a similar frequency. Perhaps an incentive was attached to concluding cases that day. The cases on that particular day are considered unreliable because of that. Looking at the average throughput time of that particular day compared to all the other days, there is a large difference: 157.84 days versus 52.13 days. The incentive for the cases to be concluded that day, might have been that these cases have been running for too long, someone forgot to conclude them or what have you. This day is considered very unusual and cases concluded that day are therefore omitted from further analysis.

Figure 15: Pre-designed model of CCB case, **Figure 16: Mined model with unexp. exceptions** **Figure 17: Core flow A** **Figure 18: Core Flow B**

The throughput time shows a (1/x) reciprocal curve (see Figure 19) which indicates that log normalizing should probably be used to transform the data for linear regression purposes. A boxplot is then considered for the log-normalized throughput times per case in order to check for unusual data. Outliers are looked into further to see if these data can be considered unusual for a reason. The boxplot depicted in Figure 20 shows how outliers can be detected.

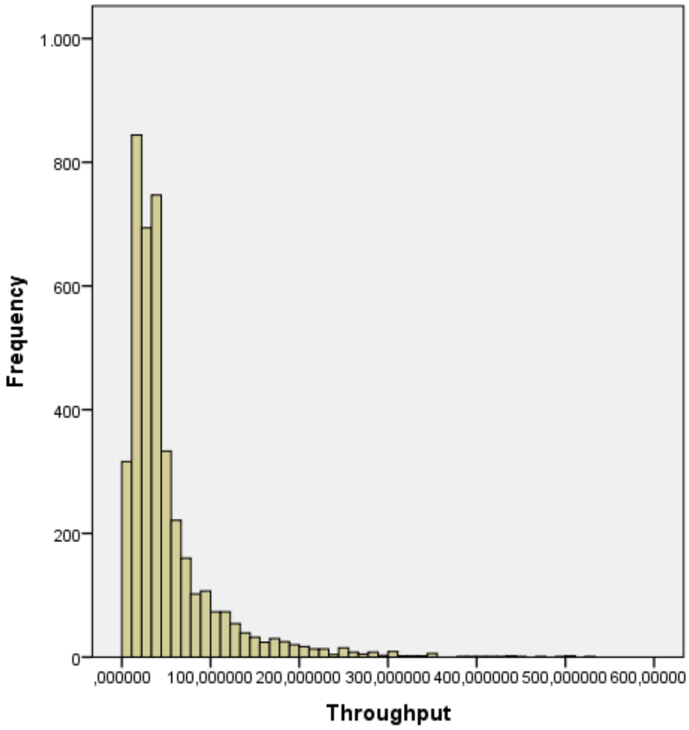


Figure 19: Histogram of throughput time for CCB cases

There are two extreme outliers as is shown in the boxplot with throughput time is log-normalized. Case numbers 2212 and 2215, with case IDs 6335-ehv68 and 6341-ehv68, are extreme outliers and checked. Both their time stamps show that every activity was done within one minute per activity in one day, all one after another. As other cases show that this is unusual behavior, these two cases are omitted as they might have been entered incorrectly into the event log. The other less extreme outliers balance out on both the upper and down side of the plot. They may be realistic cases of people potentially focusing on one case or handling easy cases (lower side outliers) or potentially cases which get increasingly lower priority or which are increasingly more difficult to deal with (upper side outliers). These cases remain in the analysis.

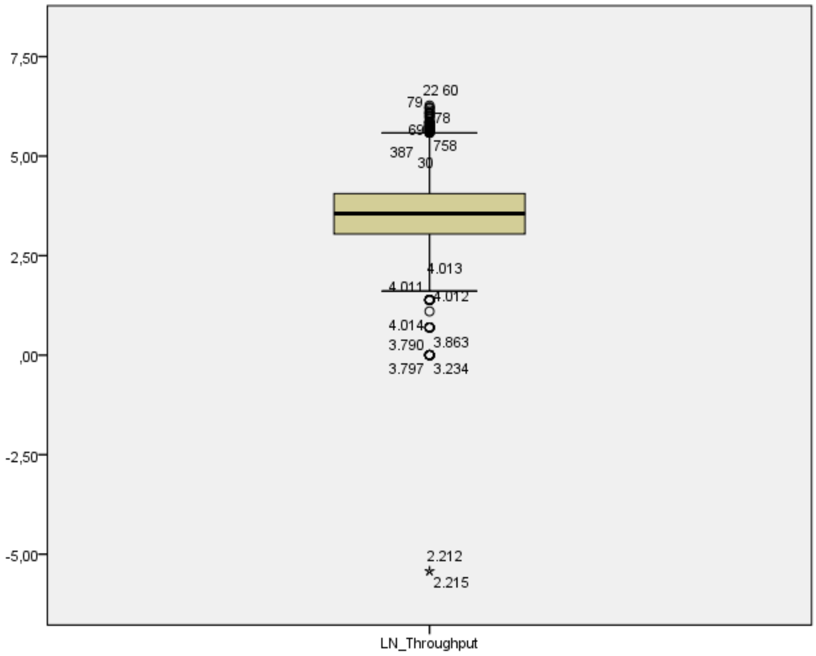


Figure 20: Boxplot for throughput time log-normalized for the CCB case

When the outliers and open cases have been omitted 4012 cases are left for an analysis which includes process mining and linear regression analysis. The mined process model is shown in Figure 16 which makes a distinction between the pre-designed process model of Figure 15 and unexpected exceptions. There are a total of 9 unexpected paths, but there are three exception types identified among them: **skip**, **iteration** and **loop**.

Skip consists of paths 5-9 in Figure 16, and shows behavior in which one or more activity is skipped. Iteration consists of paths 2-4 in which one or more steps are taken towards an earlier activity in the pre-designed process model. Loop is only shown by path #1 and shows behavior in which an activity is performed another time after this same activity has just been performed on a case.

The main flow is determined by looking at 80% of the cases that go through the most common process variants of the pre-designed process model. The most common process variant actually contains an unexpected exception which is thus not possible when following the pre-designed process model which is shown in the previous section. As the rule of thumb as mentioned in Section 4.1 is to look at the combination of process variants which include 80% of the cases, when an attempt is made to filter for that, process variant 1 in Table 4 is the most popular process variant and the second most common process variant is process variant 2 in Table 4. Any combination of these does not lead to a combination of process variants with 80% of the cases running through them. Process variant 1 contains an unexpected exception, but not considering that variant for the core flow causes a large chunk of the cases to be taken out of the analysis. However, to test the hypotheses where a distinction is made between expected and unexpected exceptions, this distinction can be made within the analyzed data as well. Possibly the pre-designed process model simply had a major design flaw and skipping the activity of “analysis” should be part of the core flow as well. Both versions of the core flow (only process variant 2, 28.49% of the cases, and both process variants 1 and 2 combined, 97.51% of the cases, as seen in Table 4) are therefore considered when analyzing the effects of expected exceptions as illustrated in Figure 17 and Figure 18.

Table 4: Process Variants in CCB data set

Process Variant Number	Process Variant Path	Percentage of Cases Going through Process Variant
1	Submit Resolution Evaluation Concluded	69.02%
2	Submit Analysis Resolution Evaluation Concluded	28.49%
3	Submit Analysis Evaluation Concluded	0.48%
4	Submit Analysis Analysis Resolution Evaluation Concluded	0.23%
rest	--	1.78%

Core flow A with only process variant 2 in Table 4 and illustrated in Figure 17, consists of five activities which follow one another sequentially. Comparing that with the pre-designed process model in the previous section, all paths to and from the activity “CCB Analysis” are expected exceptions. The same goes for core flow B with both process variants 1 and 2 except that Analysis can now also be skipped as illustrated in Figure 18. The data set from Volvo in Section 4.3 also contains an activity which is used for additional analysis: Supplier Support. Both activities’ expected exceptions are therefore referred to as **analysis exception**. Loop now also occurs, but as they are part of the cases that are considered unreliable, it is not an expected exception that is analyzed. For core flow A 1527 cases remain after cases have been excluded and for core flow B 3903 cases remain.

4.2.2 Interpretation of Linear Regression Analysis – Unexpected Exceptions

First the Pearson correlation is shown between the log-normalized throughput time and all unexpected exceptions together in Table 5. One can conclude that the throughput time log normalized and exceptions have some significant (at level 0.01) negative correlation.

Table 5: Correlations for unexpected exceptions of CCB cases

		LN_Throughput	exception
LN_Throughput	Pearson Correlation	1	-,143**
	Sig. (2-tailed)		,000
	N	4012	4012
exception	Pearson Correlation	-,143**	1
	Sig. (2-tailed)	,000	
	N	4012	4012

**. Correlation is significant at the 0.01 level (2-tailed).

Performing the regression with all three types of exceptions of skip, iteration and loop yields insignificant results for the loop exception. The loop exception, which accounts for only one case, is therefore omitted from the regression.

Looking at the results of the linear regression analysis with log-normalized throughput time as dependent variable and the dummy variables of skip and iteration, but not loop as independent variables, the model shows a low R^2 of 4.2% (Table 6) which means the predictors predict the throughput time poorly. This is, however, not a big problem as throughput can be influenced by many factors such as which resources are used, time of day, bottlenecks, etc. Together with the significance of the model shown through the F-test and the t-tests for the beta coefficients for the two predictors, all at probability level 0.01, the model does show there is a significant relation between the skip and iteration exceptions and the throughput time (Table 7). For skip the relation is negative (i.e. reduces throughput time), whereas for iteration this is positive (i.e. increases throughput time). The assumptions of linear regression analyses have not been violated as can be seen in Appendix I.

Table 6: Model Summary[b] for unexpected exceptions of CCB cases

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,205[a]	,042	,041	,85768	,042	87,793	2	4009	,000	1,221

a. Predictors: (Constant), itration, skip

b. Dependent Variable: LN_Throughput

Table 7: Coefficients[a] for unexpected exceptions of CCB cases

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.	95,0% Confidence Interval for B Lower Bound	95,0% Confidence Interval for B Upper Bound	Correlations Zero-order	Correlations Partial	Correlations Part
1	(Constant)	3,730	,022		170,229	,000	3,687	3,773			
	skip	-,281	,028	-,156	-10,052	,000	-,335	-,226	-,148	-,157	-,155
	itration	1,159	,126	,142	9,193	,000	,912	1,406	,133	,144	,142

4.2.3 Interpretation of Linear Regression Analysis –Expected Exceptions, Core Flow A

In this section the expected exceptions are considered when comparing to core flow A in Figure 17. As only the analysis exception is considered when it comes to the expected exceptions, the correlation of expected exceptions in general with log-normalized throughput time is also the R of the linear regression model with log-normalized throughput time as dependent and the dummy variable of analysis exceptions as independent variable in Table 8 and standardized beta coefficient in Table 9.

Table 8 shows the model has a low fit as the R^2 has a value of only 4.6%. It does pass the F-test and t-test though (significant at level 0.01), which means there is a significant relation. There is a significant positive relation between the analysis exception variable and the throughput time as can be read from Table 9. The assumptions for linear regression analysis are not violated as can be read from Appendix II.

Table 8: Model Summary[b] for expected exceptions of CCB cases, core flow A

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics R Square Change	Change Statistics F Change	Change Statistics df1	Change Statistics df2	Change Statistics Sig. F Change	Durbin-Watson
1	,213[a]	,046	,045	,86703	,046	72,725	1	1525	,000	1,011

a. Predictors: (Constant), Analysis

b. Dependent Variable: LNThroughput

Table 9: Coefficientsa for expected exceptions of CCB cases, core flow A

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.	95,0% Confidence Interval for B Lower Bound	95,0% Confidence Interval for B Upper Bound	Correlations Zero-order	Correlations Partial	Correlations Part
1	(Constant)	3,695	,023		163,792	,000	3,651	3,739			
	Analysis	1,063	,125	,213	8,528	,000	,819	1,308	,213	,213	,213

4.2.4 Interpretation of Linear Regression Analysis –Expected Exceptions, Core Flow B

Here the expected exceptions as discovered when comparing the pre-designed process model with core flow B of Figure 18 are considered. Also here, the analysis exception is considered when it comes to the expected exceptions and the correlation of expected exceptions in general with log-normalized throughput time is also the R of the linear regression model with log-normalized throughput time as dependent and the dummy variable of analysis exceptions as independent variable in Table 10 and standardized beta coefficient in Table 11.

Now Table 10 shows there is an even lower fit with R^2 of 2.4%. With the F-test and t-test passed (significant at level 0.01) there is a significant relation and Table 11 shows there is a significant positive relation between the analysis exception variable and the throughput time. The assumptions for linear regression analysis are not violated as can be read from Appendix III.

Table 10: Model Summary[b] for expected exceptions of CCB cases, core flow B

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,156[a]	,024	,024	,85629	,024	96,784	1	3901	,000	1,233

a. Predictors: (Constant), Analysis

b. Dependent Variable: LNThroughput

Table 11: Coefficients[a] for expected exceptions of CCB cases, core flow B

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	3,533	,014		256,048	,000	3,506	3,560			
	Analysis	1,187	,121	,156	9,838	,000	,951	1,424	,156	,156	,156

4.3 Volvo – Incident Management

This section is regarding the incident management in Swedish car manufacturing company Volvo to handle clients' problems. The incident management process registers the incidents, allocates problems to the right team, attempts to find a solution to solve the problem, and close the problem.

In Section 4.3.1 the data is mined in order to find out the context of the data. Based on what is found in the data certain assumptions are made and the data will be manipulated for it to be able to be processed through the process mining software and the mined process model with the distinction between exceptions is presented. In Section 4.3.2 results are presented on unexpected exceptions for this data set and in Section 4.3.3 on expected exceptions for this data set.

4.3.1 Descriptive Data Analysis

As this data set is accompanied with a pre-design process model just like the previous data set, a distinction can be made between unexpected and expected exceptions. However, the data set and the pre-designed process model are not as easily linked as they were in the previous section. Activities in the pre-designed process model have to be interpreted from activities in the data set that are named differently and more ambiguously. Figure 21 illustrates the original pre-designed process model on the left side. The original has the three activities Match?, Investigate & Design, and Resolve also layered to three levels of the organization based on the difficulty of the problem. However, in order to simplify the process model, they have been flattened to one level. The original data set has several activities of which it is assumed they belong to the three activities of Register/Classify, Match? and Investigate & Design. For the sake of simplifying the pre-designed process model they have been renamed to In Progress as illustrated in Figure 21. With the three activities of the original process model also following each other, a loop is added in the new pre-designed process model for the activity of In Progress. The first two activities of the original pre-designed process model of Incident? and Other Routines are not represented in the data set and are therefore omitted from the new pre-designed process model.

With the new pre-designed process model determined (Figure 27), the data can be filtered for open cases and outliers. This section thus consists of an explanation of which data have been filtered and how, the exception types that have been identified and a list of the most common process variants. Though the outliers and process variants will be analyzed for two separate data sets as will be shown later in this section. The pre-designed process model shows a clear ending activity, namely "Close". All cases that do not end with Close are therefore omitted from further analysis.

When looking at the time span in which the cases have been finished, illustrated in Figure 22, it seems that the cases have all been closed in the month of May 2012, even though they have been started at various other times. It also seems like there is roughly a seasonal trend of one week. This is not weird as people work on incidents during week days and not weekends. The low frequency dates at which cases have been closed are on or near the weekends. It is unclear why the third week of May in 2012 had a lot less incidents which had been closed. It might have been due to holidays although this would be peculiar on a global level. The days that are missing are the 17th and 18th of May which happen to be Ascension Day (Thursday the 17th) and a bridge day to the weekend.

Figure 21: linking the pre-designed process model with the data set, thus generating a pre-designed process model that can be used to detect exceptions with the data set

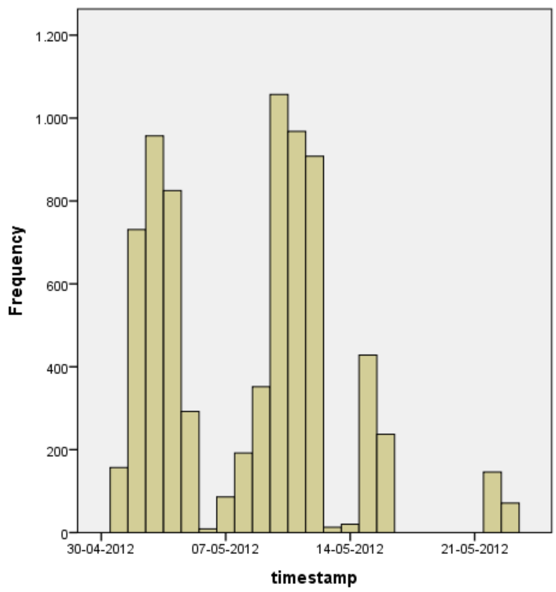


Figure 22: Frequency of Volvo cases per time stamp

The throughput time is log-normalized as it shows a (1/x) reciprocal curve (Figure 23). When looking at a histogram of the dataset (Figure 24) which shows the frequency of the log-normalized throughput time, two bell curves are detected. One with values of LN(Throughput time) roughly below -2.5 and another with values of LN(Throughput time) roughly higher than -2.5 but with an unusual peak around the value 2.

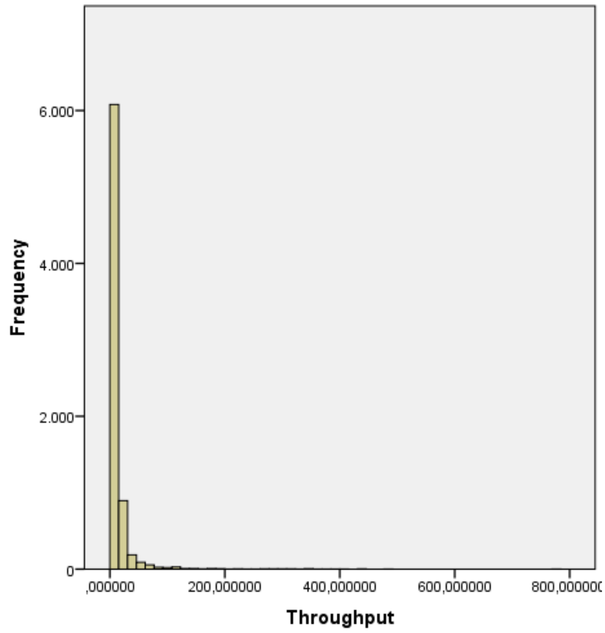


Figure 23: Histogram of throughput time of the Volvo case

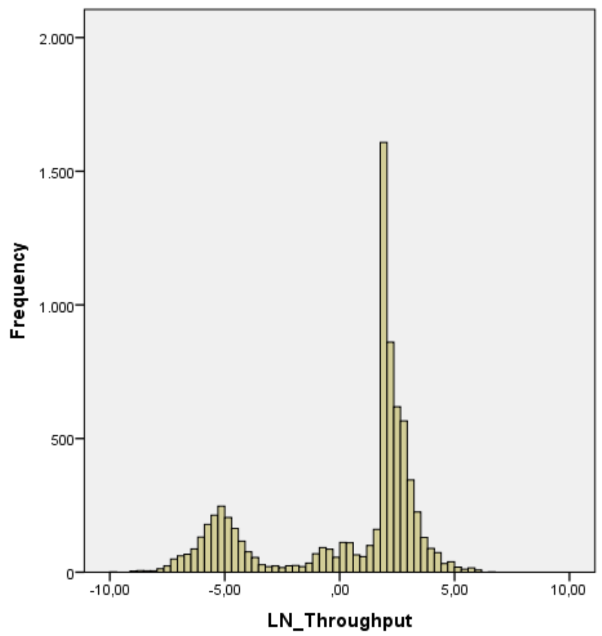


Figure 24: Histogram for the throughput time log-normalized for the Volvo case

One reason for the two bell curves could be that the data sets represent roughly two different types of incidents: one type which is generally solved or handled faster and another which is generally solved slower. From a quick perspective it does not become clear what attribute is linked to these different types of incidents. Through a regression analysis it will be checked whether exceptions are having an influence on the cases' throughput time. The reason for the unusual peak around LN(Throughput time) of 2 can be due to the previously mentioned reason of the system being offline: a holiday. When looking at the event logs, the time stamps of cases with values around

LN(Throughput time) of 2 show that most cases are closed in the week prior to the week with Ascension Day. The week before the peak week of cases with LN(Throughput time) of 2 show a relatively low frequency. It is assumed that employees worked extra hard on closing incidents the week prior to the week with holidays. This might already give an indication that the values and with that the errors of dependent variable LN(Throughput time) will not be independent from each other which would violate one of the assumptions for linear regression analysis.

As the data revolve around incidents that need to be solved, it is natural that a learning curve exists and that this could cause that steps such as “resolve” are skipped as it is already clear how the incident should be resolved. Possibly this behavior in the data shows there are simple cases with a smaller throughput time and complex cases with a larger throughput time, which would also show why there are two bell shaped curves in the histogram. The data are further analyzed to check for the influence of exceptions on the “simple” and “complex” data sets, roughly divided between LN(throughput time) higher than and equal to -2.5 and lower than -2.5. These two separate data sets are further analyzed for outliers and process variants. For the data set of simple cases one outlier is detected as can be seen in Figure 25. The outlier is removed from further analysis. For the data set of complex cases also one case is detected (Figure 26) as extreme outlier and removed.

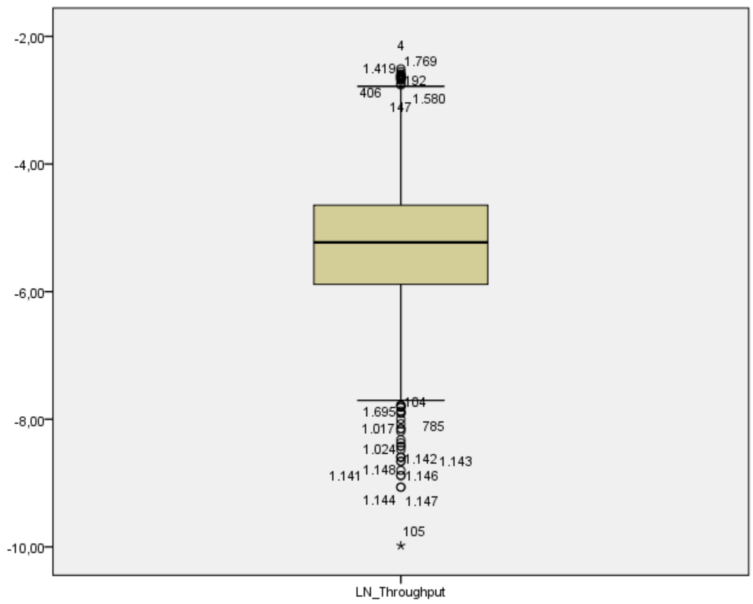


Figure 25: Boxplot for the log-normalized throughput time for the simple Volvo cases

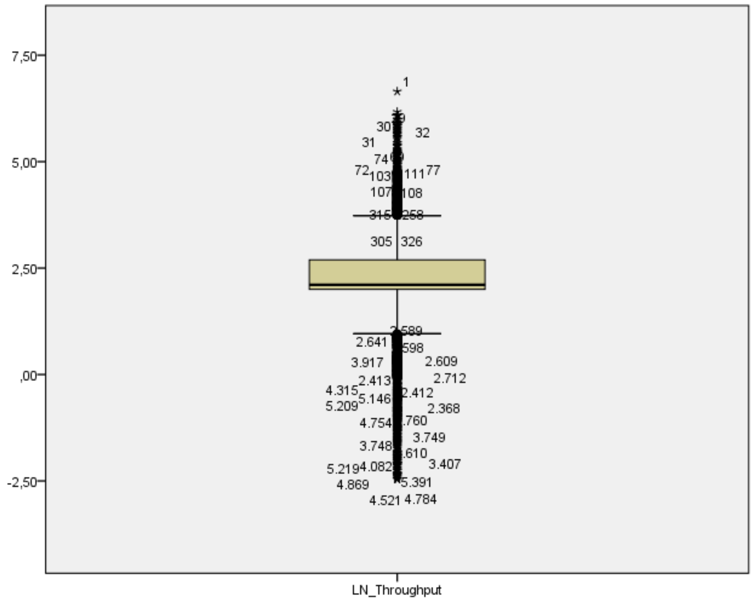


Figure 26: Boxplot for the log-normalized throughput time for the complex Volvo cases

Mining the data for the two sets yields the mined process model illustrated in Figure 28. Just like the previous analysis in Section 4.2 also this time the exceptions of skip (paths 1 and 2), iteration (paths 3 and 4) and loop (paths 5-7) are detected. And also like in Section 4.2, the core flow (Figure 28) compared with the pre-designed process model (Figure 27) shows an expected exception of the analysis type. Supplier Support is defined as a step in which the incident is so complex that an analysis must be performed by a supplier in order to resolve an issue. The core flow consists of the 12 most common process variants containing 80% of all cases. The simple case data set has 1787 cases and the complex case data set has 5661 cases.

Figure 27: Pre-designed process model **Figure 28: Mined process model with unexpected exceptions for Volvo case, and** **Figure 29: Core process flow for Volvo case**

The first process variant only contains a little over 15% of the cases and 23 process variants of the mined process model are all variants of a number of loops from In Progress to itself before continuing to the Resolve activity. It is only the 24th process variant with only 0.4% of the cases using that route which contains Supplier Support as activity. As no “simple case” has the expected exception of the analysis type, for the expected exception only the complex data set is considered, which contains 4959 cases.

4.3.2 Interpretation of Linear Regression Analysis – Unexpected Exceptions

This section is divided in two, namely Sections 4.3.2.1 and 4.3.2.2, to go into the linear regression analyses of respectively the simple case data set and complex case data set.

4.3.2.1 Interpretation of Linear Regression Analysis – Unexpected Exceptions: Simple Cases

The Pearson correlation is shown between the log-normalized throughput time and all exceptions in the form of dummy variable in the simple case data set in Table 12. The correlation between them is negative with a value of -0.158 which is significant at the 0.01 level.

Table 12: Correlations for unexpected exceptions simple Volvo cases

		LN_Throughput	exception
LN_Throughput	Pearson Correlation	1	-,158**
	Sig. (2-tailed)		,000
	N	1787	1787
exception	Pearson Correlation	-,158**	1
	Sig. (2-tailed)	,000	
	N	1787	1787

**. Correlation is significant at the 0.01 level (2-tailed).

The loop exception does not occur for the “simple” incidents and is therefore automatically removed from the regression analysis. The predictors are therefore the dummy variables of the skip and iteration exceptions and the dependent variable is the log-normalized throughput time per (simple) case. Performing the linear regression analysis for this data set shows the assumptions for linear regression analyses (normality, linearity, homoscedasticity, and independence) have not been violated (see Appendix IV).

Table 13: Model Summary[b] for unexpected exceptions simple Volvo cases

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,182[a]	,033	,032	1,04127	,033	30,403	2	1784	,000	1,744

a. Predictors: (Constant), iteration, skip

b. Dependent Variable: LN_Throughput

Table 14: Coefficients[a] for unexpected exceptions simple Volvo cases

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	-3,942	,200		-19,674	,000	-4,335	-3,549			
	skip	-1,372	,202	-,158	-6,795	,000	-1,768	-,976	-,158	-,159	-,158
	iteration	1,805	,466	,090	3,870	,000	,890	2,719	,089	,091	,090

As Table 13 shows, the model has a low R^2 of 3.3% which means the predictors predict the throughput time poorly. This is, as mentioned also in Section 4.2, expected as throughput can be influenced by many factors such as which resources are used, time of day, bottlenecks, etc. Together with the significance of the model shown through the F-test and the t-tests for the beta coefficients for the two predictors, the model does show there is a significant relation between these exceptions and the throughput time at level 0.01. The relation between the skip exception and throughput time log-normalized turns out to be negative (i.e. reduces throughput time), whereas for iteration this is positive (i.e. increases throughput time) with standardized beta coefficients of respectively -0.158 and 0.090 (see Table 14).

4.3.2.2 Interpretation of Linear Regression Analysis – Unexpected Exceptions: Complex Cases

For the complex cases there does not seem to be a significant correlation between the dummy variable of all exceptions together and log-normalized throughput time (Table 15).

Table 15: Correlations for unexpected exceptions complex Volvo cases

		LN_Throughput	exception
LN_Throughput	Pearson Correlation	1	-,004
	Sig. (2-tailed)		,782
	N	5660	5660
exception	Pearson Correlation	-,004	1
	Sig. (2-tailed)	,782	
	N	5660	5660

Skip, iteration and loop exceptions in the form of dummy variables are used as predictors and log-normalized throughput time as dependent variable for this linear regression analysis. Except from linearity, all assumptions for linear regression analysis (normality, independence, and homoscedasticity) are violated (see Appendix V), which means the model is not good in order to predict the throughput time of these complex cases.

4.3.3 Interpretation of Linear Regression Analysis – Expected Exceptions

The "simple" cases do not contain any analysis exception and they are not analyzed further. Also in this section, just as in Section 4.2.3, as only the analysis exception is considered when it comes to the expected exceptions, the correlation of expected exceptions in general with log-normalized throughput time is also the R of the linear regression model with log-normalized throughput time as dependent and the dummy variable of analysis exceptions as independent variable and standardized beta coefficient. However, despite that the Pearson correlation is significant at level 0.01 between the analysis exception dummy variable and log-normalized throughput time (Table 16), the linear regression analysis shows the assumptions are violated and the results for linear regression analysis are unreliable (Appendix VI).

Table 16: Correlations for expected exceptions complex Volvo cases

		LNThroughput	AnalysisException
LNThroughput	Pearson Correlation	1	,070**
	Sig. (2-tailed)		,000
	N	4932	4932
AnalysisException	Pearson Correlation	,070**	1
	Sig. (2-tailed)	,000	
	N	4932	4932

**. Correlation is significant at the 0.01 level (2-tailed).

4.4 Eindhoven Municipality – Appeals

The data represent what is recorded in a system of the municipality of Eindhoven and its citizens' appeals for permits. Unlike the previous two data sets, no pre-designed process model is available to check conformance of the designed model and what is shown through process discovery in process mining software. Therefore no distinction can be made between expected and unexpected cases and only exceptions in general are analyzed. To determine the core flow, just like with the two previous cases, 80% of the cases caught in the most common process variants are used to distinguish when a path is an exception or not.

Section 4.4.1 contains the descriptive analysis of the data set, describing manipulations that have been performed on the data set for process mining and the exception types that have been discovered. Section 4.4.2 contains the results of the linear regression analysis between exceptions and throughput time.

4.4.1 Descriptive Data Analysis

First the data is analyzed in order to check whether manipulations are necessary which are then performed. The core flow, using 80% of the cases going through the most popular process variants, is shown in Figure 30. Cases that have not gone through the closing activity are removed and so are outliers when they are detected. Register Appeal is always the starting point, but there is no consistent end point. Archive looks like a common and logical end point and is seen as the end point and therefore filtered for that.

Exceptions are detected by looking at the mined process model from Figure 30 and comparing it with Figure 31. When comparing the main flow with the flow that includes all exceptions an additional activity is shown: Revision. Cases with this activity are seen as exception in which an update is made in the administration so that when archived, it is clear that a new appeal will be requested, but then revised.

Withdraw Appeal is seen as a cancellation of the process. Comparing the main flow with the flow that includes all exceptions shows that all except for one exception path are either an early or late withdrawal of the appeal, or a revision of the case. The one exception which does not belong to either of these two exceptions is not taken into consideration as it follows an illogical route. It is first rejected and after that processed for further analysis on what to do with the appeal as if it were not rejected. Possibly it was a mistake when entering the data. Both Withdraw Appeals and Revision are a type of activity which usually results in the case having an early exit. It could be considered similar to skip, except for that this is not simply a path which skips one or several activities. Besides skipping activities just like the skip exception does, this type of exception could also follow the rest of the normal process flow without having skipped any activity. No other exception is detected and therefore only the relation between throughput time and one exception, named **Early Exit**, is considered.

Unlike the two data sets in Sections 4.2 and 4.3 which also have a process model, this data set does not have most cases with small throughput after which larger amounts are gradually getting less frequent. Log normalization is therefore not appropriate. As can be seen in Figure 32, the range shows frequencies for throughput which are irregular, showing there is also no normal behavior of any sort.

Figure 30: Core flow of Appeals cases, and

Figure 31: Mined process model of Appeals cases with black dotted numbered arrows indicating exceptions

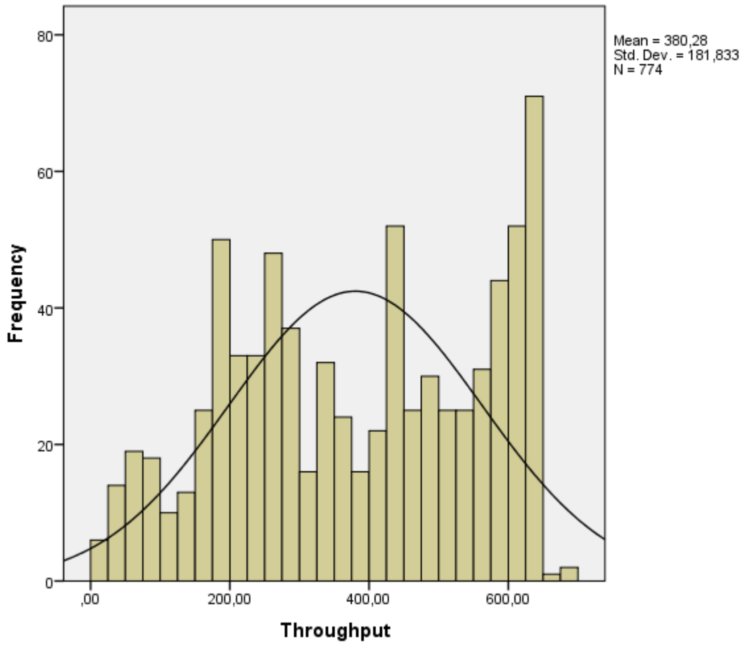


Figure 32: Throughput time frequency histogram

To later compare the values between the different data sets, this data set is still standardized. A boxplot shows there are no outliers in the data set (Figure 33). 774 cases are being analyzed for linear regression in Section 4.4.2.

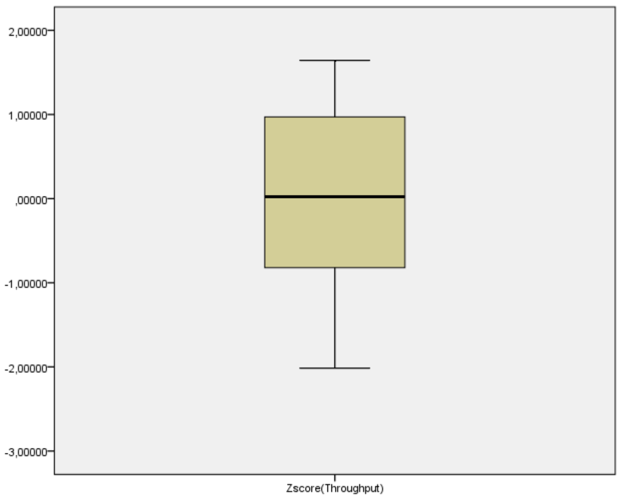


Figure 33: Boxplot of throughput time standardized

The most common process variants of the mined process model can be found in Table 17.

Table 17: Most common process variants from process mining of appeals data set

Process Variant Number	Process Variant Path	Percentage of Cases Going through Process Variant
1	Register appeal Confirm reception Register receipt of documents Result hearing/write advice Send advice Create dossier Process decision Archive	58.15%
2	Register appeal Confirm reception Register receipt of documents Result hearing/write advice Withdraw appeal Archive	8.8%
3	Register appeal Confirm reception Register receipt of documents Withdraw appeal Archive	6.66%
4	Register appeal Confirm reception Rejected Write draft advice Send advice Create dossier Process decision Archive	6.42%
rest	--	19.97%

4.4.2 Interpretation of Linear Regression Analysis –Exceptions

Table 18: Correlations for exceptions Eindhoven Municipality Appeals cases

		Zscore(Throughput)	EarlyExit
Zscore(Throughput)	Pearson Correlation	1	-,164**
	Sig. (2-tailed)		,000
	N	774	774
EarlyExit	Pearson Correlation	-,164**	1
	Sig. (2-tailed)	,000	
	N	774	774

**. Correlation is significant at the 0.01 level (2-tailed).

There is some significant (at level 0.01) negative correlation between the Early Exit exception (the only type of exception present), and the throughput time standardized as can be seen in Table 18. When a linear regression analysis is performed, however, the assumptions (normality, linearity, homoscedasticity and independence) are violated clearly (see Appendix VII).

4.5 Big US company – Incident Management

Much like Section 4.3 with the incident management system of Volvo, this data set also contains event logs of an incident management system which deals with certain problems in a big US company. Incidents are reported, assigned, need to be progressed, be queued, resolved and closed. For this data set no pre-designed process model was available and therefore no conformance checking can be done. However, by defining the core flow, the exceptions can be identified.

Section 4.5.1 contains the descriptive analysis of the data set, describing manipulations that have been performed on the data set for process mining and the exception types that have been discovered. Section 4.5.2 contains the results of the linear regression analysis between exceptions and throughput time.

4.5.1 Descriptive Data Analysis

The data is analyzed in order to check whether manipulations are necessary. The core flow, using 80% of the cases going through the most popular process variants, is shown in Figure 37. Closed cases are those cases that ended with the Closed activity or Canceled activity.

The data is checked for whether or not it should be transformed for linear regression. A histogram of throughput time, as illustrated in Figure 34, shows an (1/x)-type shape, so the data are log-normalized in an attempt to generate a normal curve with the resulting data set. Setting up a histogram then results in two separate groups, each showing a (1/x)-type curve once again, as is shown in Figure 35. The log-normalization makes a distinction between two groups clearer.

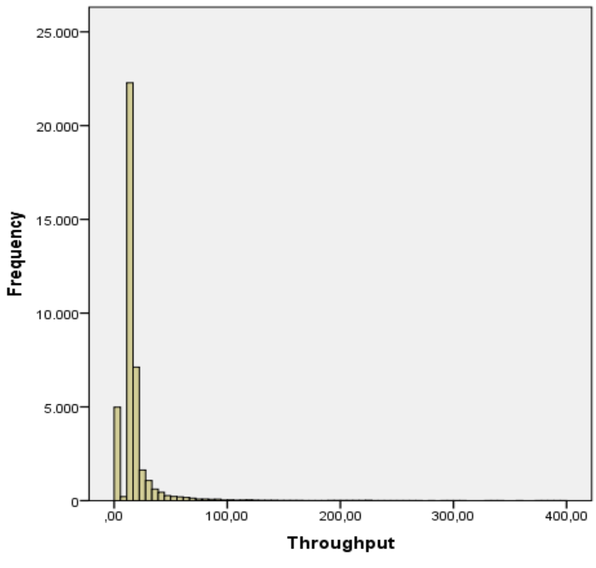


Figure 34: Histogram with frequency of throughput time for incident management cases big US company

Putting the log normalization of the throughput time against time, as shown in Figure 36, it also shows that over time there are two groups of incidents: those with a higher throughput time, and those with a lower throughput time. Just like the Volvo case, it might be a matter of complex and simple cases. It is discovered by approximation that the groups can be divided in those with a value higher than and equal to 2.72 and a value below 2.72, as LN(Throughput time) of 2.72 shows a sudden spike in frequency after the values lower than that have a consistently lower frequency, as is also seen in the scatterplot (Figure 36). The frequency of cases ending their processes is quite equal across time in the given data set, as seen in Figure 39, except for a lower frequency in the very beginning. The data is split in two for simple and complex cases, the same arguments are implied as used in the distinction of simple and complex cases for the incident management cases of Section 4.3 for Volvo.

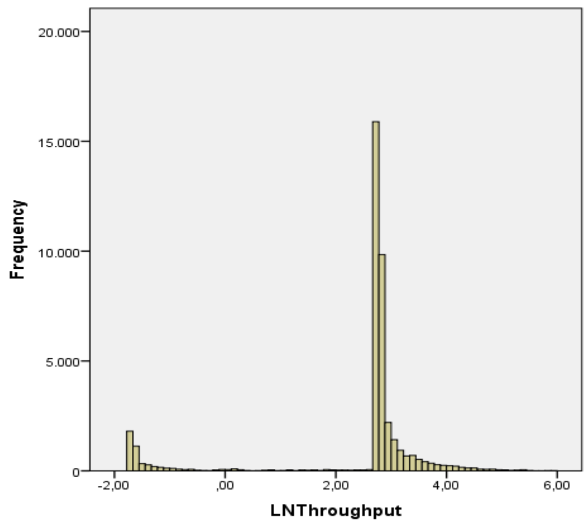


Figure 35: Histogram with frequency of log-normalized throughput time for incident management cases big US company

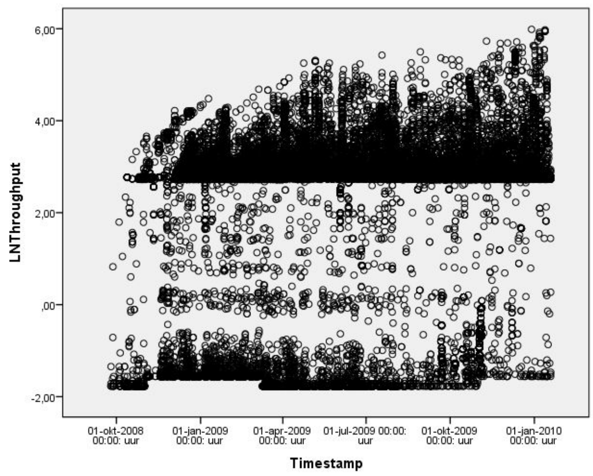


Figure 36: Timestamp per case and its throughput time log-normalized for big US company case

When the data set is mined to show exceptions (see Figure 38), all exception paths are identified based on the mined process model with four different categories defined: **skip** (paths 1-5), **iterate** (paths 6-10), **loop** (paths 11-13), and **early exit** (paths 14-17). In this case early exit is an activity which says “cancel”, but of which the paths are considered exceptions with the same purpose. Two paths going from pending to in progress and vice versa are considered exceptions belonging to the exception type of iteration, as putting work in progress once more after having the case pending for some time is interpreted as having to take a step back in the process in order to work some more on solving the incident.

Figure 37: Core flow of Big US company incident management, and

Figure 38: Mined process model of Big US company incident management with, 1-5 (green): skip, 6-10 (red): iteration, 11-13 (red-loop): loop, 14-17 (black): early exit

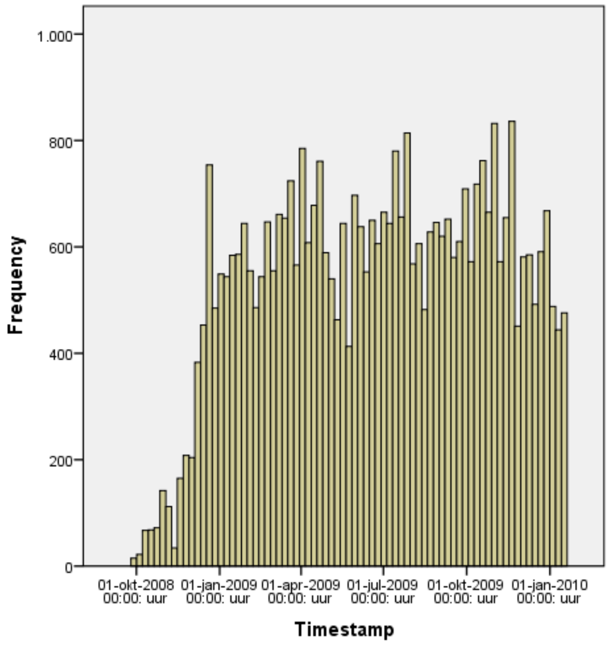


Figure 39: histogram with frequency of timestamps of the cases ending for big US company case

The most common process variants of the mined process model can be found in Table 19. The simple cases have a total of 5414. The simple cases follow the two most popular process variants of Reported, Resolved, Closed in 1/3rd of the cases and Reported, Assigned, Resolved, Closed in 1/6th of the cases shown in Table 19. It is assumed this is, as indicated earlier, due to these incidents being simple and solutions are either already known, or they are easy to find. 34742 cases remain to be analyzed for the complex cases. The complex cases' main process model is the same as the process model that includes all cases.

Table 19: Most common process variants from process mining of big US company incident management data set

Process Variant Number	Process Variant Path	Percentage of Cases Going through Process Variant
1	Reported Assigned Resolved Closed	26.7%
2	Reported Resolved Closed	16.61%
3	Reported Assigned Assigned Resolved Closed	14.79%
4	Reported Assigned In Progress Resolved Closed	7.34%
rest	--	34.56%

4.5.2 Interpretation of Linear Regression Analysis –Exceptions

This section provides results of the linear regression analysis for both the simple cases and complex cases, with the exception types as dummy variables as predictors and log-normalized throughput time as dependent variable. They both do not yield any significant results for the linear regression analysis with the assumptions for linear regression analyses of normality, linearity, homoscedasticity and independence clearly violated (see Appendices VIII and IX). Therefore only the correlations between all exception types together in one dummy variable and the log-normalized throughput time are considered.

Table 20: Correlation for exceptions of simple big US company cases

		LNThroughput	Exception
LNThroughput	Pearson Correlation	1	,254**
	Sig. (2-tailed)		,000
	N	5414	5414
Exception	Pearson Correlation	,254**	1
	Sig. (2-tailed)	,000	
	N	5414	5414

**. Correlation is significant at the 0.01 level (2-tailed).

There is a significant (at level 0.01) positive correlation between throughput log normalized and the general exception dummy variable for the simple cases as can be read from Table 20. For the complex cases this is also the case as can be read from Table 21.

Table 21: Correlation for exceptions of complex big US company cases

		LNThroughput	Exception
LNThroughput	Pearson Correlation	1	,293**
	Sig. (2-tailed)		,000
	N	34742	34742
Exception	Pearson Correlation	,293**	1
	Sig. (2-tailed)	,000	
	N	34742	34742

**. Correlation is significant at the 0.01 level (2-tailed).

4.6 Dutch Financial Institute – Loan Applications

This section shows the result for the analysis of the effects of exceptions on throughput time for a Dutch financial institute's loan application process. Again with this data set no pre-designed process model was provided with the data set and the data set was left for the own interpretation of the researcher. The activities include the acceptance or declining of applications, and then the processing of the accepted applications till offers are made and sent.

Section 4.6.1 goes into the descriptive analysis of the data set in which the data will be analyzed and pre-processed for process mining so that exceptions can be detected. Section 4.6.2 will then go into the linear regression analysis of the effects exception types have on throughput time.

4.6.1 Descriptive Data Analysis

In this section the data are prepared for process mining. Only the cases that have been closed are considered. In this case the activities Declined Application, Cancelled Application and Approved Application are considered end activities for which cases are considered closed. Without a pre-designed process model, a core flow must be determined by using 80% of the cases going through the most common process variants (see Figure 40).

When looking at the frequency of the throughput time, a large amount of cases seem to have had a throughput time of around 0 days. Looking closer at the data, it seems that this is mainly due to the system being the resource involved in the process rather than a person. This means that it was probably automated. It often shows the route of Process Variant 1 in Table 22. The system is involved in 86.65% of these cases prior to filtering (which is why the percentage differs from Table 22) which accounts for 3429 cases with a throughput time of almost 0 days. Besides that variant, the process also seems automated for Process Variant 4 in Table 22, although the throughput time is about 1 month for all such cases. In 76.39% of the activities of the cases the system was the resource handling it. Alternatively all cases within the main flow also have the system involved as user.

As the cases which are only influenced by the system and the clients' interaction with it, which make the throughput time rather fixed (without having any exception to them), these cases are omitted from the analysis, as it is assumed these are automated process variants on which other resources do not have much of an influence. Looking at the throughput time, there are unusual peaks at 0 days and 30 days, and it quickly becomes clear that cases that are declined following one of the core flow routes or cancelled performed by the system are responsible for these peaks. For the 30 days peak it is assumed it is due to automation where for example at all times, when nothing has happened after one month, the system automatically cancels the application. Omitted are the core flow routes ending with Declined Application and routes ending with Cancelled Application performed by the system, which can be read by looking at which user handled these activities.

When checking the frequency of the throughput time values, it first looks like there are two groups showing some kind of (1/x) type curve (see Figure 42). From the value of around 7 days the values show a distinction between the groups. Two groups are defined for cases that are dealt with within one week (simple cases) and a group that takes longer to be dealt with (complex cases).

Figure 40: Core flow of loan application data set of Dutch financial institute, and

Figure 41: Mined process model of Dutch financial institute loan application with, 1-3 (green): skip, 4 & 5 (red): iteration, and 6-11 (black): early exit

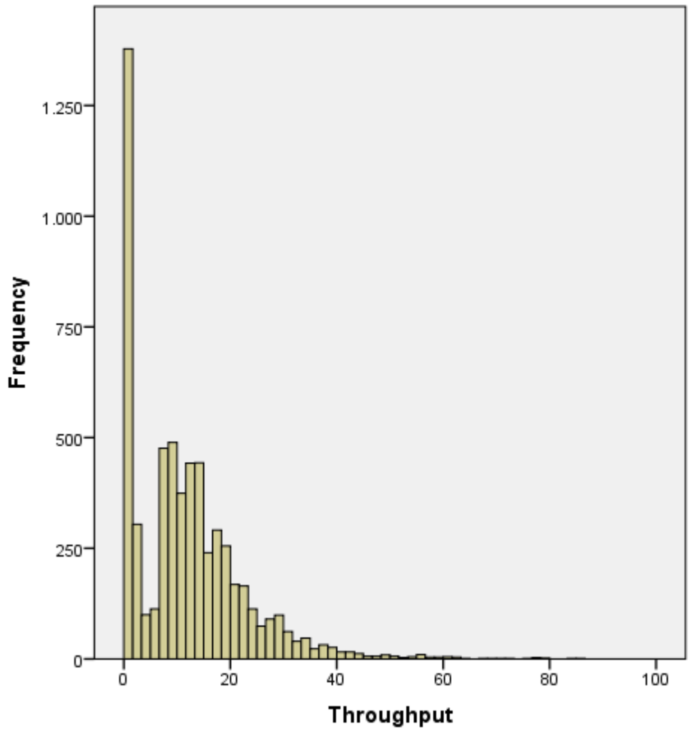


Figure 42: Frequency of throughput time for loan applications Dutch financial institute

Outliers and timeline are checked for the different groups individually. Looking at the frequency for the simple cases (throughput time less than a week), the histogram shows a sloppy (1/x) reciprocal curve (see Figure 43).

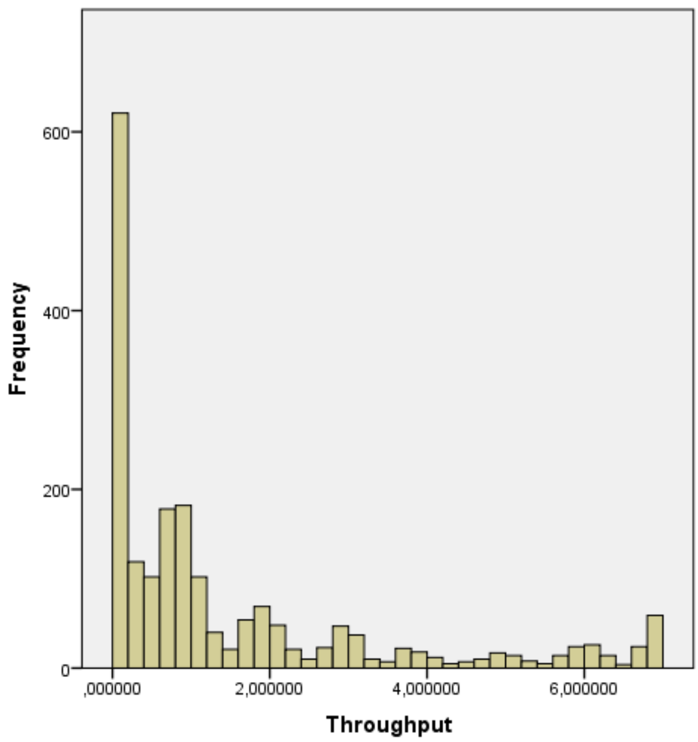


Figure 43: Frequency of throughput time shorter than a week for loan applications Dutch financial institute

Making the dependent of throughput time easier to compare to what effects the independents have, a log-normalization is performed on the throughput time. This results in a frequency graph which looks a bit more like a bell shaped curve at least till -1 and then it starts varying a lot. Though when performing a boxplot on those also outliers are detected of values lower than LN(throughput time)=-6 (see Figure 45). These data points are omitted resulting in 1947 cases for analysis of simple cases. Omitting outliers does not change the shape of the histogram though.

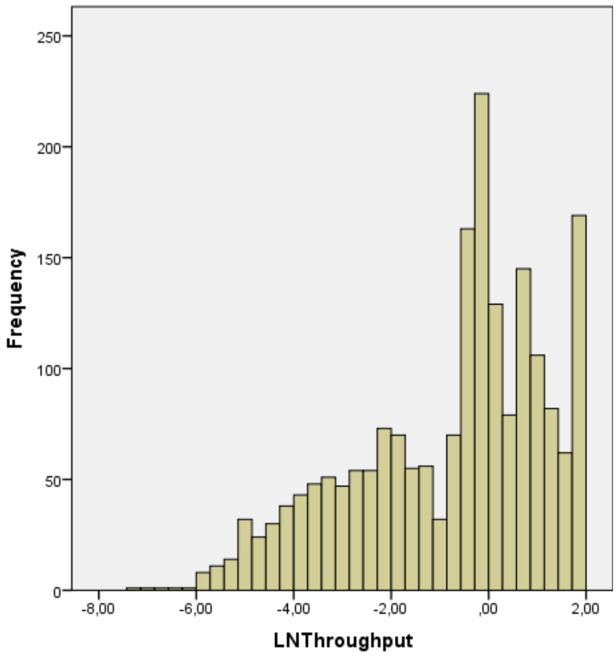


Figure 44: Frequency of log-normalized throughput time shorter than a week for loan applications Dutch financial institute

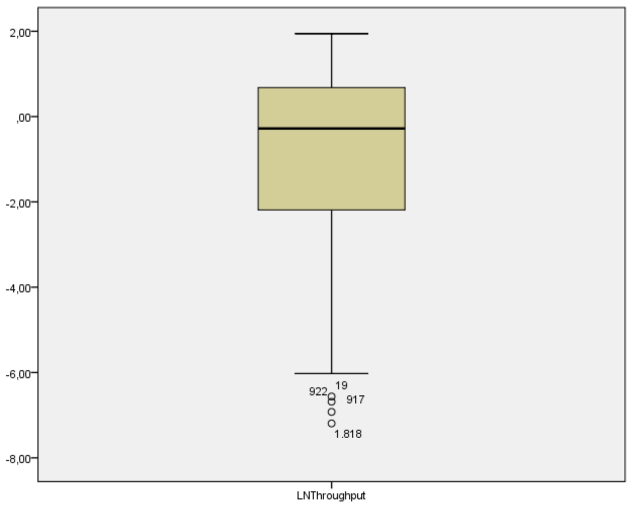


Figure 45: Boxplot for log-normalized throughput time shorter than a week for loan applications Dutch financial institute

For the complex cases with throughput time longer than a week, log-normalizing results in a more bellshaped curve than it did for the simple cases (Figure 46). Performing a boxplot shows there are outliers (see Figure 47), which are removed and consequently leaves the complex cases with a total of 3947.

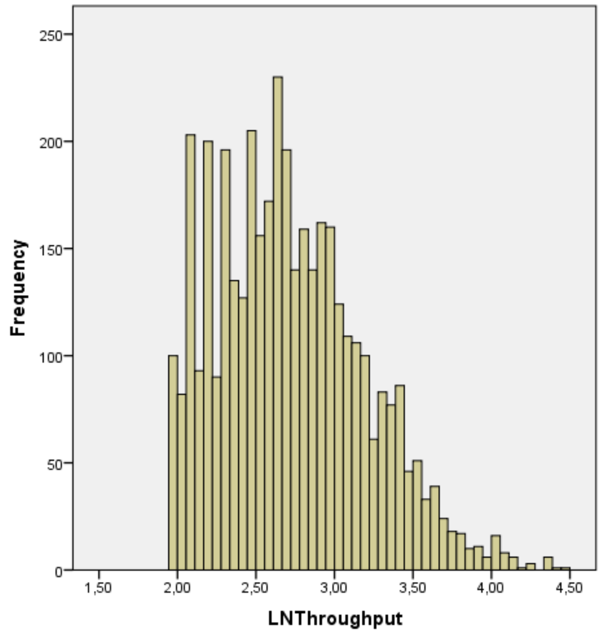


Figure 46: Frequency of log-normalized throughput time longer than a week for loan applications Dutch financial institute

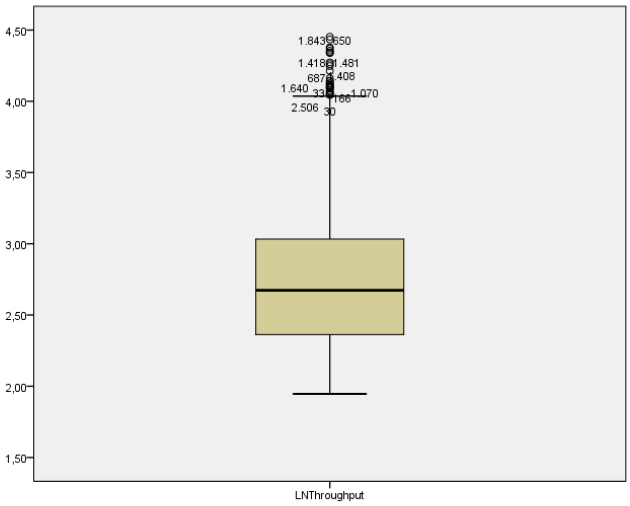


Figure 47: Boxplot for log-normalized throughput time longer than a week for loan applications Dutch financial institute

Mining for processes then yields the process model of Figure 40 which can be used to check conformance with the core flow that is mined using 80% of the cases in Figure 40. All exception paths are identified based on the mined process model with three different exception types defined: **skip** (paths 1-3), **iteration** (paths 4 & 5), and **early exit** (paths 6-11). In this case early exit is two activities which say “cancelled” or “declined”. Cancelled and Declined Application are activities which show an end to the process which would otherwise follow the path through the setting up of an offer. This abrupt ending of the flow can be caused by the (lack of) initiative of the client, resulting in a cancellation, and by the company, resulting in a decline. The most common process variants of the data set can be found in Table 22.

Table 22: Process variants in loan application data set

Process Variant Number	Process Variant Path	Percentage of Cases Going through Process Variant
1	Submitted application Partly submitted application Declined application	45.08%
2	Submitted application Partly submitted application Pre-accepted application Accepted application Selected offer Created offer Sent offer Returned offer Accepted offer Approved application	12.1%
3	Submitted application Partly submitted application Pre-accepted application Accepted application Selected offer Created offer Sent offer Cancelled application	8.92%
4	Submitted application Partly submitted application Pre-accepted application Cancelled application	8.67%
rest	--	25.23%

4.6.2 Interpretation of Linear Regression Analysis – Exceptions: simple cases

This section looks at the results of linear regression analysis for the simple cases of the loan application process. There is a significant correlation at level 0.01 between the log-normalized throughput time and the presence of an exception in such a way that exceptions generally increase the throughput time, as can be read from Table 23. For this data set the assumptions for linear regression analyses are too violated for the results to be considered reliable (see Appendix X).

Table 23: Correlation for exceptions of simple Loan Application cases

		LNThroughput	Exception
LNThroughput	Pearson Correlation	1	,121**
	Sig. (2-tailed)		,000
	N	1969	1969
Exception	Pearson Correlation	,121**	1
	Sig. (2-tailed)	,000	
	N	1969	1969

**. Correlation is significant at the 0.01 level (2-tailed).

4.6.3 Interpretation of Linear Regression Analysis – Exceptions: complex cases

Here the results are presented for the complex cases of the loan application process when it comes to linear regression analysis.

Table 24: Correlation for exceptions of simple Loan Application cases

		LNThroughput	Exception
LNThroughput	Pearson Correlation	1	,229**
	Sig. (2-tailed)		,000
	N	3947	3947
Exception	Pearson Correlation	,229**	1
	Sig. (2-tailed)	,000	
	N	3947	3947

**. Correlation is significant at the 0.01 level (2-tailed).

There is a significant (at level 0.01) positive correlation between the exceptions as dummy variable and the log normalized throughput (Table 24).

The linear regression analysis with dependent variable of log-normalized throughput time and predictors skip, iteration and early exit resulted in more reliable results for the "complex" cases with the assumptions not (strongly) violated (see Appendix XI). When looking at all cases and the outliers which mostly belong to "simple" cases, one can conclude that the simple cases have a lot of factors that affect them whereas with the longer cases the effects of exceptions on throughput time seem more apparent. Just like with the other data sets that have been analyzed, skip reduces throughput time and iterate increases throughput time (see Table 26). Cancel also increases throughput time in this case, possibly as a result of the cancelled cases getting a very low priority in regards to how quickly they should be handled once they have been cancelled. The R^2 of the model is with 8.5% low (see Table 25), but better than most results in this thesis. The F-test for model fit and t-test for all three exception types' standardized beta coefficients are significant at level 0.01 (see Tables 25 and 26).

Table 25: Model Summary[b] for exceptions of simple Loan Application cases

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,292[a]	,085	,084	,42960	,085	122,151	3	3943	,000	1,933

a. Predictors: (Constant), EarlyExit, Iterate, Skip

b. Dependent Variable: LNThroughput

Table 26: Coefficients[a] for exceptions of simple Loan Application cases

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	2,614	,009		294,080	,000	2,597	2,632			
	Skip	-,151	,040	-,060	-3,739	,000	-,229	-,072	,022	-,059	-,057
	Iterate	,298	,016	,298	18,707	,000	,267	,330	,279	,286	,285
	EarlyExit	,063	,017	,058	3,769	,000	,030	,095	,055	,060	,057

4.7 Comparison of results from the linear regression analyses of the five data sets and relation to the hypothesis

In this section the results of the linear regression analyses of the five data sets from Sections 4.2 to 4.6 are compared to one another and benchmarked. Section 4.7.1 provides a general analysis of the results with the benchmark. Section 4.7.2 uses the results to reflect back on the hypotheses from Chapter 3.

4.7.1 Benchmarking results of linear regression analyses

In order for the benchmarking to be done, it is important to look at the standardized beta coefficients that resulted from the regressions which provide a relative effect so one can compare even though the range of a variable, in this case throughput time, is completely different in different data sets (Field 2005). The exception types have all been converted to dummy variables which simply indicate presence of an exception type within a case or lack of presence. Furthermore a distinction is made between the data sets that did come with a pre-designed process model, thus allowing for a distinction between unexpected exceptions and expected exceptions, and those that did not.

Table 27 shows the results of all linear regression analyses with the previously mentioned dimensions incorporated in it. Information that is compared per case is the Pearson correlation between throughput time (of which the specific type is provided too) and the presence of exceptions in general, the R^2 which indicates how well the regression model with exception type predictors predict the independent variable of throughput time, and the standardized beta coefficients of each exception type. The greyed out numbers indicate the linear regression analysis's assumptions have been violated, as indicated per section of the respective data set, and "n.s." indicate that a standardized beta coefficient has been found to be non-significant. A dash ("-") indicates that the exception type is not present in the data set.

The results from Table 27 show that only three out of the five data sets yielded reliable results for the linear regression analysis as the assumptions for linear regression analysis (normality, linearity, homoscedasticity and independence) have not clearly been violated. The two data sets that did not yield a reliable fit between the regression model and the predicted data points they predicted were the data sets of the Eindhoven municipality and the big US company. Both of them were not accompanied with a pre-designed process model and heavily relied on the interpretation of the data sets. The skip and iteration exceptions are the only exceptions of which there are significant results for beta coefficients in more than one data set, which means that allows for a comparison. The loop exception was not significant for the beta coefficient in any of the results. Both the analysis and early exit exception yielded significant results for their beta coefficients in only one data set each. For the latter two that at least means there is an indication of the effects these exceptions have on throughput time, but there are no other results to compare them with.

First when looking at correlations between exceptions in general and throughput time, it can be said that the direction of the effect is not always either positive or negative. Looking at the standardized beta coefficients, the skip exception however always provides a negative effect on throughput time. That is it reduces throughput time. Iteration consistently increases throughput time through a positive effect. The beta coefficients of analysis and early exit are positive, as can be read from Table 27.

Table 27: Results of linear regression analyses of all five cases with standardized effects of exceptions on throughput time per exception type

	Data set	Exception Expected and/or Unexpected	Model	Dependent (y_i)	Correlation with all exceptions	R squared	Standardized Beta Coefficients of Exception Types				
							Skip	Iteration	Loop	Analysis	Early Exit
With process model	Dutch industrial company (CCB)	Unexpected Exceptions	$y_i = \beta_0 + \beta_1 x_{skip} + \beta_2 x_{iterate} + \beta_3 x_{loop} + \varepsilon_i$	LN(Throughput time)	-0.143	4.2%	-0.156	0.142	n.s.	-	-
		Expected Exceptions	$y_i = \beta_0 + \beta_1 x_{Analysis} + \varepsilon_i$	LN(Throughput time "core flow A")	0.213*	4.6%*	-	-	-	0.213*	-
				LN(Throughput time "core flow B")	0.156*	2.4%*	-	-	-	0.156*	-
	Volvo (Incident Management)	Unexpected Exceptions	$y_i = \beta_0 + \beta_1 x_{skip} + \beta_2 x_{iterate} + \beta_3 x_{loop} + \varepsilon_i$	LN(Throughput time "simple cases")	-0.158	3.3%	-0.158	0.090	-	-	-
				LN(Throughput time "complex cases")	n.s.	5.0%	-0.200	0.172	n.s.	-	-
		Expected Exceptions	$y_i = \beta_0 + \beta_1 x_{Analysis} + \varepsilon_i$	LN(Throughput time "simple cases")	-	-	-	-	-	-	-
				LN(Throughput time "complex cases")	0.070	0.5%	-	-	-	0.070	-
Without process model	Eindhoven Municipality (Appeals)	All Exceptions	$y_i = \beta_0 + \beta_1 x_{EarlyExit} + \varepsilon_i$	Z(Throughput time)	-0.164	2.7%	-	-	-	-	-0.164
	Big US Company (Incident Management)	All Exceptions	$y_i = \beta_0 + \beta_1 x_{skip} + \beta_2 x_{iterate} + \beta_3 x_{loop} + \varepsilon_i$	LN(Throughput time "simple cases")	0.254	8.5%	0.132	0.191	0.063	-	0.173
				LN(Throughput time "complex cases")	0.293	10.9%	0.113	0.203	0.088	-	0.202
	Dutch financial institute (Loan Applications)	All Exceptions	$y_i = \beta_0 + \beta_1 x_{EarlyExit} + \varepsilon_i$	LN(Throughput time "simple cases")	0.121	2.4%	n.s.	0.129	-	-	0.061
				LN(Throughput time "complex cases")	0.229	8.5%	-0.06	0.298	-	-	0.058

"n.s." = non significant; grey color = regression analysis assumptions (clearly) violated

*the results of the CCB case (core flow A and B) cannot both be used for analysis simultaneously as either core flow A or core flow B should be chosen for analysis

The size of the standardized beta coefficients is different per data set for skip and iteration but always show the same direction: positive or negative (respectively -0.156, -0.158 and -0.06 for skip and 0.142, 0.090 and 0.298 for iteration). The sizes of the standardized beta coefficients for analysis and early exit are respectively 0.213 and 0.058, which means that for analysis the effect seems relatively big for these beta coefficients, while that of early exit seems relatively small. Both exception types are types that belong to separate additional activities. These activities would not have been possible if they were not already in the system which means that is an indication these exception types can be considered expected exceptions.

The model fit is between 3.3% and 8.5% which shows the predictors predict the throughput time poorly, but this is expected as many factors influence throughput time, such as effectiveness of the system that is used, availability of resources, how trained employees are etc.

4.7.2 Reflection with results on hypotheses

This section will look back at the hypotheses from Chapter 3 and state whether the hypotheses are to be rejected or not, or whether a conclusion can even be made. A bad model fit as was the case for the data sets of which the results have been greyed out in Table 27. It will not be used to confirm or reject a hypothesis. In the case of the analysis exception having only one significant result per exception type provides evidence but not enough to confirm or reject hypotheses. This is because both positive and negative results are necessary to confirm the hypothesis or all positive or all negative results are necessary to reject the hypothesis. Using the results from Table 27, Table 28 shows how they affect the results for the outcome of confirmation, rejection or inconclusiveness regarding the hypotheses.

First the main hypothesis is considered. The main hypothesis is “Exceptions in business process models cause an increase of throughput time”. Looking at the Pearson correlations in Table 27, based on the results, it can be said this hypothesis should be rejected. Exceptions can cause both an increase and decrease of throughput time, depending on the underlying effects the individual exception types have in a specific case but especially on the weight of the exception type’s effect. The last statement is the case as the overall effect of exceptions can still be e.g. an increase of throughput time, but that this is because the weight of the underlying effect of exception types that cause a decrease is simply too weak to work against increasing exception types.

However, this does not mean that the sub-hypotheses are automatically rejected as well. Looking at sub-hypotheses 1 and 2; “Unexpected exceptions in business process models cause an increase of throughput time “, and “Expected exceptions in business process models cause an increase of throughput time “, we have two cases where the effects of unexpected exceptions on throughput time can be read from and one case where the effect of expected exceptions can be read from (see Table 27). However, as mentioned in the previous section, both the analysis and early exit exception are defined as expected exceptions, which means there are two results that indicate expected exceptions cause an increase in throughput time. For unexpected exceptions, looking at the Pearson correlations in Table 27, one can read that sub-hypothesis 1 is to be rejected, as for unexpected exceptions there seems to be a decrease of throughput time. With an increase of throughput time with expected exceptions in process models when looking at the Pearson correlation in Table 27 for the CCB case and at least the early exit exception in the case of the Dutch financial institute, sub-hypothesis 2 is confirmed. Two results are, however, still meager evidence.

Table 28: how the results from the regression analyses in Table 27 affect if the outcome of the hypotheses from Chapter 3 are confirmed, rejected or inconclusive

Hypothesis	Confirmed, rejected or inconclusive	Reasoning, only looking at significant results in black in Table 27
Main hypothesis Exceptions in business process models cause an increase of throughput time.	Rejected	Correlations between exceptions in general and throughput time are both positive and negative.
Sub-hypothesis 1 Unexpected exceptions in business process models cause an increase of throughput time.	Rejected	Correlations between unexpected exceptions and throughput time are negative where they should be positive to confirm the hypothesis.
Sub-hypothesis 2 Expected exceptions in business process models cause an increase of throughput time.	Confirmed	Correlations between expected exceptions, which includes the Early Exit exception, and throughput time are positive.
Sub-hypothesis 3 The influence of an exception in a business process model on throughput time depends on the type of exception.	Confirmed	The standard beta coefficients from the regression analyses are different per type of exception.
Sub-hypothesis 3a The exception type "skip" in business process models causes a reduction of throughput time.	Confirmed	The standard beta coefficients from the regression analyses having the skip exception type are all negative.
Sub-hypothesis 3b The exception type "early exit" in business process models causes a reduction of throughput time.	Rejected	The standard beta coefficient from the regression analyses having the early exit exception type is negative where it should be positive to confirm the hypothesis.
Sub-hypothesis 3c The exception type "iterate" in business process models causes an increase of throughput time.	Confirmed	The standard beta coefficients from the regression analyses having the iteration exception type are all positive.
Sub-hypothesis 3d The exception type "loop" in business process models causes an increase of throughput time.	Inconclusive	There have been no significant results for the loop exception type in the regression analyses which means the hypothesis cannot be confirmed nor rejected.
Sub-hypothesis 3e The exception type "analysis" in business process models can have both an increasing and a decreasing influence on throughput time.	Inconclusive	There has been only one significant result for the standard beta coefficient of the analysis exception type in the regression analyses where both positive and negative results were needed to confirm the hypothesis and several results of all positive or all negative were needed to reject the hypothesis. This means the hypothesis cannot be confirmed nor rejected.

Sub-hypothesis 3, "The influence of an exception in a business process model on throughput time depends on the type of exception" can be considered confirmed when looking at the results from Table 27. The exception types do not all have the same direction, increasing or decreasing the throughput time. Going deeper into the sub-hypothesis, i.e. looking into sub-hypotheses 3a to 3e, 3a and 3c can be considered as confirmed. 3a hypothesized that a skip exception would cause a decrease of throughput time whereas 3c hypothesized an iteration exception would cause an increase of throughput time. However, also for these two hypotheses, just like sub-hypothesis 2, each only have two results per sub-hypothesis which means that it is meager evidence. Of early exit, belonging to sub-hypothesis 3b, it was hypothesized that it would cause a decrease of throughput time, which should be rejected. Sub-hypothesis 3d, which hypothesizes loop exceptions will cause an increase of throughput time cannot be confirmed nor rejected. There are no significant results for the loop exception. For the analysis exception there are not enough results to confirm or reject sub-hypothesis 3e.

5. Flexibility Techniques Used to Handle Exceptions

This Chapter goes into how flexibility techniques from Chapter 2 can be used to handle the exception types that have been discovered in Chapter 4. The exception types will be revisited and per exception type the flexibility techniques that enable the exception type are mentioned and a link is made, describing why they enable the exception type. Section 5.1 goes into which flexibility category handles exceptions when the time to decide which technique should handle which exception becomes a factor, using the BPM lifecycle (Figure 48). Section 5.2 provides the exception types from Chapter 4 and flexibility techniques that are appropriate to handle them in tabular form accompanied with an explanation as to why the techniques and exception types are linked. Section 5.3 then concludes with which flexibility categories are relevant for which case from Chapter 4.

5.1 Flexibility Categories to Handle Expected and Unexpected Exceptions

This section will go into why which flexibility category is appropriate to handle certain exceptions.

What can already be said is that the flexibility category of flexibility by underspecification is not suitable for the exception types discovered in Chapter 4 in their specific contexts. The data sets that have been used in Chapter 4 are all data of routine processes in relatively standardized systems. It is assumed that incident management with routine problem solving, appeals with a limited number of appeals that routinely handled at a municipality, and loan applications which can be routinely handled in a system do not need parts in the process model that have not been specified (flexibility by underspecification) as the activities are straight-forward. The CCB case which has a qualitative part where the people involved need to analyze changes is besides the non-automated part of their activities considered as routine as well, as their actions have been put in a system and have been generalized enough. Besides flexibility by underspecification, techniques from flexibility by change can all be incorporated once a decision has been made to change a process model to handle exceptions. It is just a matter of deciding about when and how the change should be made when it comes to these techniques. Which technique would be appropriate is something that the process owners should determine themselves and therefore this flexibility category is also not considered for further analysis. The other flexibility categories (flexibility by design, deviation, and configuration) all have techniques that can help in dealing with the range of exceptions of this thesis.

The flexibility categories in Chapter 2 occur at three different times in the BPM lifecycle as illustrated in Figure 48 (Weske, van der Aalst et al. 2004), predominantly at process design time (flexibility by design), system configuration time (flexibility by configuration), and process enactment/run time (flexibility by deviation, change and underspecification). When an exception is expected, it can be assumed that one searches for techniques during the design time of the process model (flexibility by design), or perhaps also during configuration time (flexibility by configuration). When it is unexpected, it can be assumed that flexibility should be incorporated at run time and therefore handled by flexibility techniques from flexibility by deviation and change (and underspecification, although this is not appropriate for the data sets that have been used).

Figure 48: BPM lifecycle

5.2 Flexibility Techniques that are Appropriate to Handle Exception Types discovered in Data Analysis

This section will go over each exception type from Chapter 4 and link flexibility techniques to them which handles them. For an explanation on the flexibility techniques and what they do see Chapter 2.

Table 29: Flexibility Techniques that can be used to handle Exceptions

Exception Type	Flexibility Techniques				
	Flexibility by Design (Expected)	**Flexibility by Configuration (Expected)**	**Flexibility by Deviation (Unexpected)**	**Flexibility by Underspecification**	**Flexibility by Change**
Skip	Choice, Interleaving	Enable, Hide	Skip Task	-	-
Iteration	Choice, Iteration, Interleaving	Enable	Redo	-	-
Loop	Choice, Iteration, Interleaving	Enable	Undo, Redo	-	-
Analysis	Choice, Interleaving	Enable, Insert	-	-	-
Early Exit	Choice, Interleaving, Cancellation	Enable, Insert	-	-	-

Table 29 shows which flexibility techniques handle which exception type. For all exception types at design time flexibility can be incorporated if the exception is expected. Choice which allows for a set of activities to be available as next step of which one can be chosen is usable for all exception types in this thesis. For skip this would mean that a activity further down the process model should be available as an option, for iteration one before the activity that was carried out the last, for loop the one that was carried out the last, and for analysis and early exit this should be the activity associated with these exceptions. The same goes for interleaving which allows for activities to be carried out in any order whatsoever. Iteration as flexibility technique allows for a task to be executed repeatedly. That can be the task carried out the last (thus allowing for a loop), but also a task which was already carried out a lot earlier (i.e. iteration exception). Cancellation allows for a task to be interrupted after which it can go to an early exit activity, which means it can be appropriate for the early exit exception.

At configuration time (see Figure 48) Flexibility by Configuration is appropriate. Every exception type can be handled through flexibility by configuration as is shown in Table 29. Enable as flexibility technique is appropriate for all exceptions as it simply allows for activities to be executed in general as opposed to them being blocked. This is needed for the activities between the exceptions to actually be executed. The Hide flexibility technique enables skip as it prevents a corresponding activity to be executed. Insert is a flexibility technique that allows for an activity to be inserted into the process model at configuration time, meaning that exceptions that are dependent on exception activities can happen, which is why they are linked to analysis and early exit.

Flexibility by deviation allows for flexibility at process enactment (run time) as illustrated in Figure 48. The analysis and early exit exceptions do not have a flexibility technique in the deviation category to enable them. One cannot suddenly decide to have another activity in the system. The system must already allow for these activities to be executed. The skip exception is enabled by the skip task which does exactly as the name says, enabling what the skip exception needs to do: jump to a later activity in the process model. The Redo technique is appropriate for the iteration exception as it allows for a task that has been previously executed to be repeated. This is also why it is appropriate for loop, but the Undo technique is also appropriate for the loop exception as it lets a just carried out task to be done directly again.

As mentioned in Section 5.1, flexibility by underspecification is not appropriate for any of the exception types as in none of the exception types it is necessary to leave how they should be enabled during run time undetermined. Flexibility by change can both be appropriate or not, but that depends on how any change should be enabled, i.e. what the process owners prefer in light of how they prefer the process performance is (not) affected.

5.3 The Flexibility Categories that are relevant per Case from Chapter 4

This section specifies how the flexibility categories are relevant to enable the exceptions that occur in the context of the cases mentioned in Chapter 4. For that Table 29 which shows which flexibility techniques best enable exception types (when expected or not) is considered.

For all cases the expected exceptions are determined during the build time, which is done by flexibility by design. Especially as all cases are very routine cases, it makes sense to determine which exceptions should (not) be enabled already then. This is also what the practitioners mentioned in Chapter 3 indicated. However, this only goes for the analysis and early exit exception types as they were expected during build time, as shown in Table 30. Skip, loop and iteration were not expected exceptions in the cases of the Dutch industry company (CCB case), Volvo (incident management), the big US company (incident management) and the Dutch financial institute (loan applications). Therefore, for the exception types of skip, loop and iteration in these four cases, flexibility by design is not appropriate. To enable these exception types, flexibility by deviation is appropriate as flexibility by deviation also enables these exception types, but when they are unexpected (see Table 30).

Flexibility by configuration can be appropriate for all cases from Chapter 4 if the system allows for configuring process flows at configuration time. This would be appropriate for the expected exceptions: analysis and early exit. The practitioners mentioned in Chapter 3, however, indicated they mostly did not consider this flexibility category. With flexibility by design already enabling the analysis and early exit exception types of all cases in Chapter 4 flexibility by configuration then also becomes less of a priority which is why in Table 30 this category is mentioned between brackets to indicate a lower priority.

Table 30: Flexibility categories that are most appropriate to handle the exception types in the cases mentioned in Chapter 4

Cases (Ch. 4)	Exception Types				
	Analysis (Expected)	**Early Exit (Expected)**	**Skip (Unexpected)**	**Iteration (Unexpected)**	**Loop (Unexpected)**
Dutch industry	Flexibility by Design (and Configuration)	-	Flexibility by Deviation	Flexibility by Deviation	Flexibility by Deviation
Volvo	Flexibility by Design (and Configuration)	-	Flexibility by Deviation	Flexibility by Deviation	Flexibility by Deviation
Eindhoven Municipality	-	Flexibility by Design (and Configuration	-	-	-
Big US company	-	Flexibility by Design (and Configuration	Flexibility by Deviation	Flexibility by Deviation	Flexibility by Deviation
Dutch financial Institute	-	Flexibility by Design (and Configuration	Flexibility by Deviation	Flexibility by Deviation	-

Tables 29 and 30 show that when it is known which exception types must be enabled at which moment in the BPM lifecycle, the right flexibility techniques and by extension the right flexibility category can be chosen. This information can then be used to determine whether or not the current PAIS is appropriate when looking at the requirements determined by the exception types. Or vice versa: with the current PAIS in used and the knowledge of what effects these exception types have on throughput time one can get a better perspective on how the current system influences throughput time.

6. Conclusion

This chapter concludes the thesis and is divided in three sections. Section 6.1 reflects on the research questions and overall goal of this research, Section 6.2 focuses on the limitation of the research and Section 6.3 provides options for further research.

6.1 Discussion on Research Questions and Overall Goal

This section focuses on the answers to the research questions, thus drawing conclusions and going towards the outcome of this thesis. Section 6.1.1 goes into answering research question 1, Section 6.1.2 goes into answering research question 2, Section 6.1.3 is for research question 3 and Section 6.1.4 answers research question 4. All of the research questions are answered by using the outcome of Chapters 2, 3, 4 and 5 with each chapter having provided input for respectively research questions 1, 2, 3 and 4. Finally Section 6.1.5 reflects on the overall goal from which the research questions have been derived and how this research contributes to that goal.

6.1.1 Canonical techniques to enable flexibility in business processes according to academic literature

In this section research question 1 is answered. Research question 1 was defined in Section 1.4 as follows:

> ***Research Question 1***
> *What are canonical techniques to enable flexibility in business processes according to academic literature?*

The canonical techniques to enable flexibility in business processes according to academic literature have been found through the literature study as presented in Chapter 2 and are presented in Table 31. Schonenberg, Mans et al. (2008) have provided an already extensive study on canonical flexibility techniques up until 2008. The additional literature study for this thesis focused on canonical flexibility techniques from since 2008. Through the additional literature study an extra flexibility category has been defined. That makes the total number of flexibility categories five. The number of techniques that have been covered per category are the most commonly found techniques, and therefore indeed provide an answer to the research question which aims at uncovering the canonical techniques. That also implies this does not cover all techniques. And on top of that canonical in itself can be seen as ambiguous term. However, within the scope of this research the first research question is considered as answered with the outcome of Chapter 2 and the defined fifth category of Flexibility by Configuration is a new contribution to the area of enabling flexibility in business processes.

Table 31: Canonical Flexibility Techniques according to Literature study of Schonenberg et al (2008) and Van IJzendoorn (2013)

Flexibility Category	Flexibility Techniques
Flexibility by Design	• Parallelism • Choice • Iteration • Interleaving • Multiple Instances • Cancellation
Flexibility by Deviation	• Undo Task • Redo Task • Skip Task • Create Additional Instance of Task • Invoke Task • Declarative Languages
Flexibility by Underspecification	• Placeholders ○ Late Binding ○ Late Modeling • Realization Types ○ Static Realization ○ Dynamic Realization
Flexibility by Change	• Types of Effects of Change ○ Momentary Change ○ Evolutionary Change • Moment of Allowed Change types ○ Entry Time ○ On the Fly • Migration Strategy types ○ Forward Recovery ○ Backward Recovery ○ Proceed ○ Transfer
Flexibility by Configuration	• Port Switches ○ Block ○ Hide ○ Enable • Animation Techniques ○ Insert ○ Move ○ Delete

6.1.2 Practitioners' claims on exceptions and managing effects on KPIs through flexibility

In this section research question 2 is answered. Research question 2 was defined in Section 1.4 as follows:

> ***Research Question 2***
> *What do practitioners claim regarding if and how exceptions of business process flows occur, if and how they should be enabled and which KPI(s) matter when enabling exceptions?*

The answer to the second research question is based on the case study interviews with consultants from Accenture who have worked or currently work in business process management projects, explained more in detail in Chapter 3. Seven practitioners divided over five BPM projects provided input. The five BPM projects were for a consumer electronics leasing company in the USA, two banks in the Netherlands, a Dutch quango, and the insurance customer service for the US military.

The claims the practitioners made, gave a clear statement that exceptions do indeed occur and that they should be handled, either by preventing them or by enabling them. They also mentioned that there was a preference to reduce complexity and increase standardization. They consistently do this at build time (flexibility by design) and when they incorporate a change (flexibility by change), but also other flexibility categories were considered. The following four claims from both practitioners and triangulated with literature resulted in the hypotheses that exceptions increase throughput time which answer research question 2:

- Whatever the preferred flexibility category was, the common denominator for all flexibility category usages was that practitioners all aimed to use it to standardize and reduce complexity in order to increase business process performance.
- The KPI practitioners consistently looked at to measure the performance of the business processes was throughput time.
- Reduced complexity and increased standardization both reduce throughput time (Griffin 1997, Münstermann, Eckhardt et al. 2010).
- Reduction of exceptions reduces complexity and increases standardization (Reichert and Weber 2012).

The result was the main hypothesis of which the ones focusing on exception types in sub-hypotheses 3a-3e were also inspired by literature and process mining on data sets that have been analyzed in Chapter 4. The hypotheses can be found in Table 32.

Table 32: Hypotheses that resulted from the analysis of the case study interviews in Chapter 3

Hypothesis number	Hypothesis
Main hypothesis	Exceptions in business process models cause an increase of throughput time.
Sub-hypothesis 1	Unexpected exceptions in business process models cause an increase of throughput time.
Sub-hypothesis 2	Expected exceptions in business process models cause an increase of throughput time.
Sub-hypothesis 3	The influence of an exception in a business process model on throughput time depends on the type of exception.
Sub-hypothesis 3a	The exception type "skip" in business process models causes a reduction of throughput time.
Sub-hypothesis 3b	The exception type "early exit" in business process models causes a reduction of throughput time.
Sub-hypothesis 3c	The exception type "iterate" in business process models causes an increase of throughput time.
Sub-hypothesis 3d	The exception type "loop" in business process models causes an increase of throughput time.
Sub-hypothesis 3e	The exception type "analysis" in business process models can have both an increasing and a decreasing influence on throughput time.

6.1.3 Actual measured effect of exceptions in business process flows on throughput time

This section aims to provide the concluding answer to research question 3. The research question, as stated in Section 1.4, is as follows:

> ***Research Question 3***
> *What is the actual measured effect of exceptions in business process flows on the KPI(s) that matter to practitioners?*

The actual measured effect of different exception types on throughput time has been uncovered through process mining and linear regression analyses in Chapter 4. Five data sets have been used of which two were of the incident management type (Volvo and a big US company), one on loan applications (Dutch financial institute), one on appeals in a municipality (Eindhoven municipality) and one on the process in a CCB middleware system (Dutch industry company).

Five different exception types were discovered in the data sets: skip, iteration, loop, analysis, and early exit. The measured effects they each have on the throughput time have their consequences in regards to the hypotheses that resulted from answering research question 2. The answer to research question 3 is answered with Table 33.

Table 33: Revisited: How the measured effects of the exceptions affect if the outcome of the hypotheses from Chapter 3 are confirmed, rejected or inconclusive

Hypothesis	Confirmed, rejected or inconclusive	Reasoning, only looking at significant results in black in Table 27
Main hypothesis Exceptions in business process models cause an increase of throughput time.	Rejected	Correlations between exceptions in general and throughput time are both positive and negative.
Sub-hypothesis 1 Unexpected exceptions in business process models cause an increase of throughput time.	Rejected	Correlations between unexpected exceptions and throughput time are negative where they should be positive to confirm the hypothesis.
Sub-hypothesis 2 Expected exceptions in business process models cause an increase of throughput time.	Confirmed	Correlations between expected exceptions, which includes the Early Exit exception, and throughput time are positive.
Sub-hypothesis 3 The influence of an exception in a business process model on throughput time depends on the type of exception.	Confirmed	The standard beta coefficients from the regression analyses are different per type of exception.
Sub-hypothesis 3a The exception type "skip" in business process models causes a reduction of throughput time.	Confirmed	The standard beta coefficients from the regression analyses having the skip exception type are all negative.
Sub-hypothesis 3b The exception type "early exit" in business process models causes a reduction of throughput time.	Rejected	The standard beta coefficient from the regression analyses having the early exit exception type is negative where it should be positive to confirm the hypothesis.
Sub-hypothesis 3c The exception type "iterate" in business process models causes an increase of throughput time.	Confirmed	The standard beta coefficients from the regression analyses having the iteration exception type are all positive.
Sub-hypothesis 3d The exception type "loop" in business process models causes an increase of throughput time.	Inconclusive	There have been no significant results for the loop exception type in the regression analyses which means the hypothesis cannot be confirmed nor rejected.
Sub-hypothesis 3e The exception type "analysis" in business process models can have both an increasing and a decreasing influence on throughput time.	Inconclusive	There has been only one significant result for the standard beta coefficient of the analysis exception type in the regression analyses where both positive and negative results were needed to confirm the hypothesis and several results of all positive or all negative were needed to reject the hypothesis. This means the hypothesis cannot be confirmed nor rejected.

Table 33 shows the results from the linear regression analyses and process mining have been relayed back to the hypotheses in order to reject or confirm them, or otherwise for the hypotheses to be considered inconclusive. The conclusion here is that the actual effects of exceptions in business process flows on throughput time depend on the specific type of business processes. Of some exception types it can be said that there is evidence it has a consistent effect on throughput time, i.e. skip causing decrease of throughput time, iteration causing an increase, and expected exceptions causing an increase of throughput time, as all hypothesized by respectively sub-hypotheses 3a, 3c, and 2. But, for the rest of the hypotheses no conclusion can be drawn (sub-hypotheses 3d and 3e) or the hypothesis has to be rejected completely, i.e. the main hypothesis, sub-hypothesis 1, sub-hypothesis 3, and sub-hypothesis 3b. Most notably it must be concluded that exceptions do not always cause an increase in throughput time.

However, it is not claimed here that there is a discrepancy between what practitioners claim when it comes to what effects exceptions have on throughput time. This does provide an indication that one must tread lightly in generalizing exceptions and asking for a general explanation of how one thinks exceptions should be handled and why. Another example where one must be careful with generalization is when looking at the early exit exception and expected exceptions. Early exit was in the context of the data where it was discovered seen as expected exception of which it was hypothesized that they would cause an increase of throughput time. It was hypothesized that early exit would decrease throughput time, which means that sub-hypotheses 2 and 3b are conflicting. When hypothesizing the effect of early exit on throughput time, the fact that they would stem from already existing activities in the system was overlooked and sub-hypothesis 3b resulted from the assumption of what effect early exit would have when an early exit would take place directly without first going through an additional activity.

An important contribution of this part of the research is that exception types' effects on throughput time have been measured, which when more research is done on exceptions' effects on throughput time, can provide a first step towards getting even more insights in effects per exception type. More important is the approach to which the exception types' effects on throughput time have been analyzed. A combination of process mining, data mining and linear regression analyses on big data has not been done before as far as the researcher is aware, and therefore this approach can be a first step towards a more refined approach in which this type of analysis is automated (and less prone to human error).

6.1.4 Flexibility techniques linked to exception types that have been measured

This section aims to provide the answer to research question 4, which is as follows:

> ***Research Question 4***
> *How and which flexibility techniques can be linked to exception types that have been measured by answering RQ3?*

The last research question, research question 4, is answered by concluding which canonical flexibility techniques in business processes from the literature study results are appropriate to enable the exception types of skip, iteration, loop, analysis and early exit in the contexts of the projects they resulted from.

Table 34 shows which flexibility techniques are appropriate to handle which exception type in the contexts of the data sets where they have been discovered. The expected analysis exception in the Dutch industry case (CCB case) and Volvo case are best enabled through the flexibility techniques of

Table 34: The flexibility techniques that are appropriate to handle exception types in the contexts of the data sets where they have been discovered

Cases from Chapter 4	Exception Types	Flexibility Category	Flexibility Technique
Dutch industry	Analysis (Expected)	Flexibility by Design	Choice
			Interleaving
		Flexibility by Configuration	Enable
			Insert
	Skip (Unexpected)	Flexibility by Deviation	Skip Task
	Iteration (Unexpected)	Flexibility by Deviation	Redo
	Loop (Unexpected)	Flexibility by Deviation	Undo
			Redo
Volvo	Analysis (Expected)	Flexibility by Design	Choice
			Interleaving
		Flexibility by Configuration	Enable
			Insert
	Skip (Unexpected)	Flexibility by Deviation	Skip Task
	Iteration (Unexpected)	Flexibility by Deviation	Redo
	Loop (Unexpected)	Flexibility by Deviation	Undo
			Redo
Eindhoven Municipality	Early Exit (Expected)	Flexibility by Design	Choice
			Interleaving
			Cancelation
		Flexibility by Configuration	Enable
			Insert
Big US company	Early Exit (Expected)	Flexibility by Design	Choice
			Interleaving
			Cancelation
		Flexibility by Configuration	Enable
			Insert
	Skip (Unexpected)	Flexibility by Deviation	Skip Task
	Iteration (Unexpected)	Flexibility by Deviation	Redo
	Loop (Unexpected)	Flexibility by Deviation	Undo
			Redo
Dutch financial institute	Early Exit (Expected)	Flexibility by Design	Choice
			Interleaving
			Cancelation
		Flexibility by Configuration	Enable
			Insert
	Skip (Unexpected)	Flexibility by Deviation	Skip Task
	Iteration (Unexpected)	Flexibility by Deviation	Redo

choice and interleaving of the flexibility by design category. They can also be enabled through flexibility by configuration with enable or insert techniques, but flexibility by configuration is less commonly used according to the practitioners that have been interviewed in the case study interviews. The same goes for the expected analysis exceptions discovered in the other three data sets. The unexpected skip, iteration and loop exceptions which, besides the loop exception, all occur in the Dutch industry case, Volvo case, big US company case and Dutch financial institute case are best enabled through flexibility by deviation. For the skip exception Skip Task fits best, for iteration Redo best enables the exception, and for loop both Redo and Undo are appropriate. As they are not expected, flexibility techniques from flexibility by design that could otherwise be appropriate for these exception types are now not appropriate.

6.1.5 Reflection on the overall goal and the overall contribution of this thesis

In this section a reflection is given on the overall goal which as presented in Section 1.4 is as follows:

> ***Overall Goal***
> *Analyze the effect of exceptions in business process models on process performance measured with Key Performance Indicators and define which flexibility techniques for business process modeling can be linked with exceptions to handle different exception types.*

As indicated in the previous sections of this chapter only a certain number of exception types have been analyzed as the discovery of them have been dependent on the data at hand. On top of that the flexibility techniques that have been defined to handle these exceptions have been dependent on the literature that is available. The KPI that was clearly important to measure business process performance and which is also easily measured from the data is throughput time, which makes the analysis of the effects of exceptions to focus on that KPI. Overall it can be said that the goal has been achieved as an analysis was performed and flexibility techniques have been linked with exceptions to handle exception types. But the quality of the analysis is as strong as the resources that have been available. Unfortunately they have been limited severely, which will be further elaborated upon in Section 6.2.

6.2. Limitations of the Research

This chapter goes into the entire research and how each part of it has been limited and due to which factors they were limited. It does so by going into each part of the research design, namely Chapter 2 for the literature study in Section 6.2.1, Chapter 3 for the case study interviews in Section 6.2.2, Chapter 4 for the measured effects exceptions have on throughput time in Section 6.2.3 and Chapter 5 for the flexibility techniques that have been chosen to handle exception types in Section 6.2.4.

Overall the biggest constraint in this research was the limited availability to contextual information for the data that were received. Many assumptions had to be made because of that which severely reduces the validity of the research. Furthermore, the overall goal would reach its full potential of contribution to the field of business process management and exception handling once more focus would be given per underlying goal of the individual research questions. This research is however a mere first step to reaching the overall goal.

6.2.1 Limitations of the literature study

This section reflects upon the literature study. Chapter 2 showed the research by Schonenberg, Mans et al. (2008) with a focus on that area has been a comprehensive basis for this literature review. Through a survey on canonical flexibility techniques in business process management it partly answers the first research question regarding what the canonical techniques are to enable flexibility in processes. It does however not guarantee the completeness of that survey. Schonenberg, Mans et al. (2008) also indicate this in their paper. That could be considered a limitation of this literature study, as other popular techniques to enable flexibility in business processes from before 2008 might not have been taken into account. However, the literature which was found in the literature study of this research consistently provided examples of techniques fitting to the taxonomy of Schonenberg, Mans et al. (2008), save for the category of Flexibility by Configuration. But that could be due to configuration not yet being a well-known methodology to induce flexibility in process modeling. A search on Google Scholar with the terms of “Configurable Reference Model” and “Flexibility” provided 7 papers from before 2008 and 73 papers since 2008 at the time the literature study was carried out. That could be an indication that configuration was indeed not yet broadly considered in process modeling and flexibility and thus that the taxonomy by Schonenberg, Mans et al. (2008) can still be seen as a reliable source to start the literature study with.

6.2.2 Limitations of the case study interviews

This section goes into the limitations of the case study interviews in Chapter 3. The biggest and most profound limitation is the homogeneity of the interviewees, which is that they have all been from Accenture. This does not necessarily mean the input they provided is completely invalid, but it does question the validity of this research a lot. The fact that three of the cases from the case study interview only provided one interviewee also severely makes one question the validity of this research. The interviewees with possibly the best intentions to inform the interviewer might have some of their own interpretations of facts. Lack of validation by peers in the projects they worked contributes to lack of validity. Another limitation is the lack of proper testing of the interview questions. One brief meeting with a consultant of Accenture was the only testing done which means that there is a possibility for the questions to have not been understood properly. The interviewees have for example also not been tested for their knowledge on the underlying information in the questions on business process management. The interview questions also provide another limitation in consistency as the interviews were performed in two languages, Dutch and English of which

English was not the native language of the interviewer. Information might be lost in translation. The lack of validation by peers and academics did not occur due to the limited amount of time assigned to this part of the research. This was a conscious choice however and can be justified on convenience-based actions – also justified in literature (Dul and Hak 2008) – as the overall goal also required testing of data for measured effects of exceptions.

Section 6.3 will provide input on how a case study interview when done more thoroughly could increase validity in future research.

A limitation in the reasoning behind the hypotheses as in how the results from the case study interviews relate to them is that the link is weak to say the least. The questions in the case study interviews were based off of the flexibility categories that were discovered in the literature study. The underlying aim behind inducing flexibility through these categories, however, had always been to standardize and reduce complexity in the business processes and to reduce throughput time. Flexibility by design can directly be linked to enabling or restricting exceptions at build time, which fits to enable or restricting expected exceptions. As the results showed, expected exceptions do indeed increase throughput time and in retrospect this could have been hypothesized based on the results from what the practitioners mentioned rather than merely focusing on the general underlying aim the practitioners from the case study interviews mentioned. The consequences of the weak reasoning for the results are not grave, but they make the rigor of the hypotheses very low. In extension it can be said that it is important to seek out academics and practitioners who can challenge the reasoning towards the hypotheses actively; something that has not been done well enough in this research.

6.2.3 Limitations of the process mining and linear regression analysis

This section provides the limitations of the analysis of the effects exceptions have on throughput time. For the analysis of the effects exceptions have on throughput time the biggest limitation is the lack of context of the big data sets. Many assumptions had to be made regarding how the data was to be interpreted. This seriously decreases the validity of this research and therefore reduces the research to having its main contribution in the approach rather than the actual results. Input was asked from the sources the data were obtained from, but they did not have additional contextual information on the data. Still there were significant results that came out of the linear regression analyses which provides an indication that (some of) the data may have been interpreted correctly. The linear regression analysis assumptions of normality, linearity, homoscedasticity and independence may are prone to subjectivity due to how the graphs (see Appendices) are interpreted, though. This means that a model may have found to be “fitting” while the assumptions were assumed too easily. Assumptions related to the interpretation of the data included but were not exclusive to discrepancies between pre-designed process models and activities as they were named in the data sets, what actually can be considered an exception and what not, whether pre-designed process models can be considered as poorly modeled or not, and if outliers are indeed outliers and the results of wrongful manipulations of data or if they were the results of normal situations.

On top of that data sets as they have been logged depend on the people using the system. It is possible that people do not log the data correctly, e.g. waiting a day until an ending time stamp has been given whereas an activity has already been finished earlier. Normally outliers would provide indications data have been entered incorrectly, but detecting outliers is not a fully reliable way to indicate wrongfully logged data, especially if data have been logged incorrectly in a consistent way. Finally, when dealing with big data, also in the research itself with a lot of manual manipulations performed mistakes happen more easily than when everything is fully automated.

6.2.4 Limitations of how flexibility techniques have been linked to exception types

This section covers the limitations of the approach followed in Chapter 5. Here the limitations are very much the result of the limitations from the previous parts of the research. Chapter 5 was dependent on the outcomes of Chapters 2 to 4, which each have their own limitations, and therefore lack of validity in those chapters flows through to the approach meant to answer the last research question. An important limitation is the interpretation of flexibility techniques as they were interpreted from literature and not from how they actually operate in practice.

The conclusion on which flexibility techniques can actually handle which exception type is however purely theoretical as they are based on interpreted information which was gained from literature rather than learned from practice. This severely reduces the validity of the results in Chapter 5 and specifically what has been depicted in Table 28. It is however once again in the grand scheme of this thesis part of an approach that can be helpful in practice to decide which flexibility technique is appropriate for which exception type. Once this is tested and verified in practice, this provides input for business process managers to base their decisions on when it comes to which systems they should purchase that manages business processes. After all, if a business process management system does not have the techniques to enable a certain exception that is otherwise crucial to maintain quality in processing products or what have you, this can hurt the profitability of an organization.

6.3 Further Research

This section concludes with what more research needs to be done in order to cover the limitations from Section 6.2. For further research there are several points where more is to be uncovered. Chapters 3 to 5 each can be extrapolated to their own in-depth research.

For the literature study in Chapter 2 the literature that was researched had been found in a very systematic way, which means that overall the literature should cover the canonical flexibility techniques in BPM quite well and that no further in-depth research is seen as necessary. However, with the literature study being dependent on Schonenberg, Mans et al. (2008) one could also still consider to redo the literature study without using that paper as a basis, i.e. search literature from all years and not just from since 2008.

The case study interview of Chapter 3 can be done in a more valid way with more validation within cases by getting three or more interviewees per case with the cases also being from different organizations. With more time allocated to seeking the right cases with its interviewees, as well as with more time allocated to setting a lot longer appointments than has been done in this research, more relevant information can be found. Also the testing of the questions with test interviews can be more extensive to increase the quality of the questions and more importantly the link between the results of the case study interviews and hypotheses should be better thought through.

Chapter 4 can be its own research by focusing on a greater number of data sets that can be analyzed. More importantly in future research contextual information on the data should be found to increase validity of the interpretation of the data and in extension to that also the validity of the results that are dependent on the interpretation of the data. Lack of context has proven to cause not only a delay in analyzing the data as it is more difficult to zone in on anomalies in the data, but also a great many assumptions that had to be made.

Chapter 5 had been the result of the outcomes of the previous chapters, but when it is done as separate research, more ground can be uncovered. Firstly, one point that weakened Chapter 5 was that the flexibility techniques were not learned from practice in this research, but from literature. When PAISs are tested thoroughly for their flexibility techniques and how they handle business processes in which exceptions have been detected, it will become clearer which PAIS fits better to which type of situation when it comes to flexibility. When it comes to PAIS programs, ADEPT1, ADEPT2, METEOR, CAKE2, WASA2, and TRAM (Li, Reichert et al. 2008, Weber, Reichert et al. 2009, Reichert and Weber 2012) are mentioned to increase ease of use while maintaining soundness and flexibility. Flexibility in process modeling is incorporated by having variants of process models with apt process models that can be executed at the right time. PAIS can support this, keeping the robustness of business processes (Reichert and Weber 2012). AristaFlow BPM Suite (Reichert and Weber 2012), ProCycle (Weber, Reichert et al. 2009) (both Flexibility by Change) and GPS (Reuter, Dadam et al. 2012) (Flexibility by Underspecification extension to ADEPT) base their own program off of ADEPT1 and ADEPT2 showing the importance of the ADEPT tools when it comes to flexibility in PAISs. Looking at Chapter 2, and the flexibility categories they cover, it is best to look a bit further for ways to compare process variants and integrate them into a uniform model, thus deciding which exception types should be enabled. When it comes to Flexibility by Configuration there is the possibility to have built a configurable reference model and to execute it in a PAIS which is extended with configuration features such as YAWL (Gottschalk, Wagemakers et al. 2009) or ARIS (van der

Aalst 2009). Hallerbach, Bauer et al. (2010) developed their own framework called Provop, also used by Meerkamm (2012) providing a new tool to cover Flexibility by Configuration. These researches and the PAISs developments provide a good basis to start up a research on how they enable exceptions through their flexibility techniques. Combined with the measured effects of exceptions on throughput time, this information could then be very relevant for companies as a basis to choose the right PAIS or for PAIS vendors to improve their PAIS.

For this research no focus was put on each part of the research as is suggested for further research. A conscious choice was made to analyze in general to what extent exceptions matter regarding their effects on KPIs and how flexibility techniques in business processes can be used to enable exception types. By doing so it became clear how each part is related to each other and how additional in-depth researches are beneficial to this research and what approaches can be used best. It showed how interesting the entire field of BPM flexibility and exception handling is and how much more interesting it can become.

References

De Medeiros, A. A., et al. (2007). An outlook on semantic business process mining and monitoring. On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops, Springer.

Dul, J. and T. Hak (2008). Case study methodology in business research, Routledge.

Field, A. (2005). "Discovering statistics with SPSS." London: Sage, Last access: August **27**: 2012.

Genovese, Y. P., Daryl C. (2007). "Introducing Process Integrity: Critical to Business Applications, SOA Compositions and Processes." Gartner.

Gibbert, M., et al. (2008). "What passes as a rigorous case study?" Strategic Management Journal **29**(13): 1465-1474.

Gottschalk, F., et al. (2009). Configurable process models: Experiences from a municipality case study. Advanced Information Systems Engineering, Springer.

Griffin, A. (1997). "Modeling and measuring product development cycle time across industries." Journal of Engineering and Technology Management **14**(1): 1-24.

Günther, C. W. and W. M. Van Der Aalst (2007). Fuzzy mining–adaptive process simplification based on multi-perspective metrics. Business Process Management, Springer**:** 328-343.

Hair Jr, J. and W. Black (2010). "Babin, BJ, & Anderson, RE." Multivariate Data Analysis: A Global Perspective.

Hallerbach, A., et al. (2010). "Capturing variability in business process models: the Provop approach." Journal of Software Maintenance and Evolution: Research and Practice **22**(6-7): 519-546.

Hill, J. B. (2012). "Business Process Management, the Discipline." Gartner's Hype Cycle for Business Process Management, 2012.

Jones, T. (2012). "Automated Business Process Discovery." Gartner's Hype Cycle for Business Process Management, 2012.

Kirchmer, M., et al. (2010). "Process mining for organizational agility." Industrial Management (Des Plaines) **52**(1): 19.

Kitchenham, B. (2004). "Procedures for performing systematic reviews." Keele, UK, Keele University **33**: 2004.

Li, C., et al. (2008). "On measuring process model similarity based on high-level change operations." Conceptual Modeling-ER 2008: 248-264.

Meerkamm, S. (2012). Staged Configuration of Multi-perspectives Variants Based on a Generic Data Model. Business Process Management Workshops, Springer.

Münstermann, B., et al. (2010). "The performance impact of business process standardization: An empirical evaluation of the recruitment process." Business Process Management Journal **16**(1): 29-56.

Reichert, M. and B. Weber (2012). "AristaFlow BPM Suite." Enabling Flexibility in Process-Aware Information Systems: 441-464.

Reichert, M. U. and B. Weber (2012). Enabling flexibility in process-aware information systems: challenges, methods, technologies, Springer.

Reuter, C., et al. (2012). Guarded process spaces (GPS): A navigation system towards creation and dynamic change of healthcare processes from the end-user's perspective. Business Process Management Workshops, Springer.

Samalikova, J., et al. (2011). "Toward objective software process information: experiences from a case study." Software Quality Journal **19**(1): 101-120.

Schonenberg, H., et al. (2008). "Process flexibility: A survey of contemporary approaches." Advances in Enterprise Engineering I: 16-30.

Schonenberg, H., et al. (2008). Process flexibility: A survey of contemporary approaches. Advances in Enterprise Engineering I, Springer**:** 16-30.

van der Aalst, W. (2009). "Process-aware information systems: Lessons to be learned from process mining." Transactions on Petri Nets and Other Models of Concurrency II: 1-26.

Van der Aalst, W., et al. (2012). Process mining manifesto. Business process management workshops, Springer.

Van Der Aalst, W. M., et al. (2003). Business process management: A survey, Springer.

Van IJzendoorn, G. R. (2013). "Business Process Model Flexibility: a literature review on flexibility techniques in process modeling." EINDHOVEN UNIVERSITY OF TECHNOLOGY, IEIS, INFORMATION SYSTEMS.

Voss, C., et al. (2002). "Case research in operations management." International journal of operations & production management **22**(2): 195-219.

Weber, B., et al. (2010). "Refactoring large process model repositories." Computers in Industry **62**(5): 467-486.

Weber, B., et al. (2008). "Change patterns and change support features–enhancing flexibility in process-aware information systems." Data & knowledge engineering **66**(3): 438-466.

Weber, B., et al. (2009). "Providing integrated life cycle support in process-aware information systems." International Journal of Cooperative Information Systems **18**(01): 115-165.

Weske, M., et al. (2004). "Advances in business process management." Data & knowledge engineering **50**(1): 1-8.

Appendices

In this Chapter the appendices are provided to which is referred in Chapter 4 where linear regression analyses have been performed. In each appendix the basis for the acceptance or rejection of four assumptions is provided. The four assumptions are that of normality, linearity, homoscedasticity and independence (Field 2005, Hair Jr and Black 2010).

For normality a histogram of the standardized residuals should show a bell-shaped curve and a P-P plot with observed cumulative probability put against expected cumulative probability should show a straight diagonal line. Alternatively a sample larger than 200, which is the case for all data sets, would make this assumption less important, but it is analyzed nonetheless (Field 2005, Hair Jr and Black 2010).

For the independence assumption which assumes that the errors of the dependent do not correlate and that the data points are not dependent on each other the Durbin-Watson value is consulted. It should show a value between 1 and 2 (Field 2005, Hair Jr and Black 2010).

The variance of the standardized residuals of the regression should be relatively equal across time for the assumption of homoscedasticity to be accepted. Another way to check for this assumption is to look at standardized residuals versus standardized predicted values of the regression. The variance of the standardized residuals should be relatively equal for the assumption to not be violated. A violation of this assumption is not considered as grave as it is for violating that of independence or linearity, but it will have its influence on the other assumptions (Field 2005, Hair Jr and Black 2010).

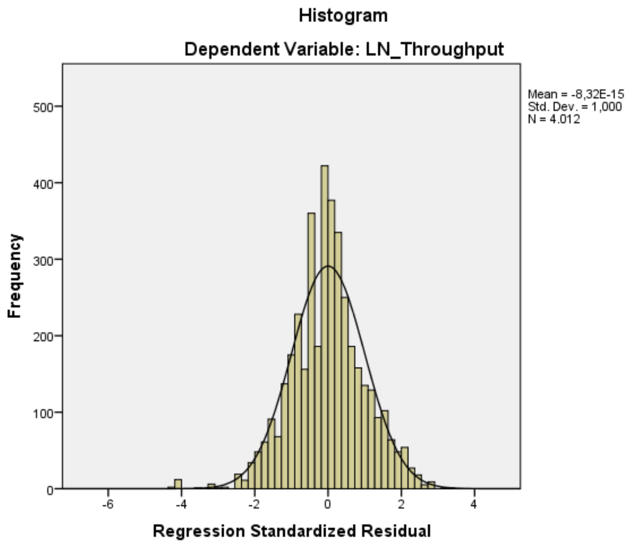
The assumption of linearity is not violated once in the plot of standardized residuals versus standardized predicted values of the regression the standardized residuals are symmetrically divided across a horizontal line (Field 2005, Hair Jr and Black 2010).

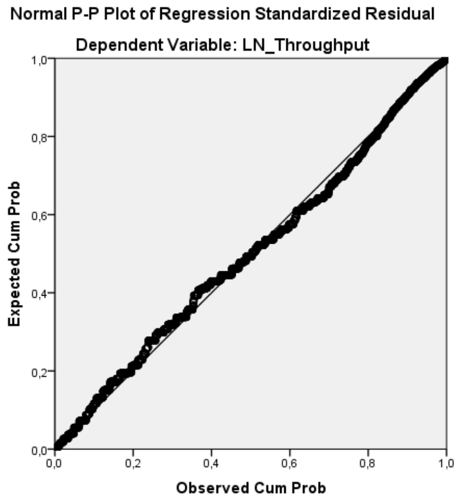
For each appendix the graphs are provided in the previously mentioned order:

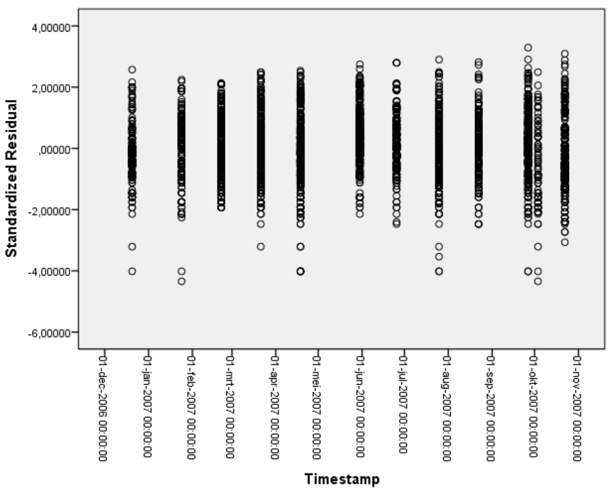
- Histogram with frequency of the standardized residual from the regression of the dependent variable
- P-P plot with observed cumulative probability put against expected cumulative probability
- A plot of time versus the standardized residuals of the regression
- A plot of the standardized residuals versus standardized predicted values of the regression

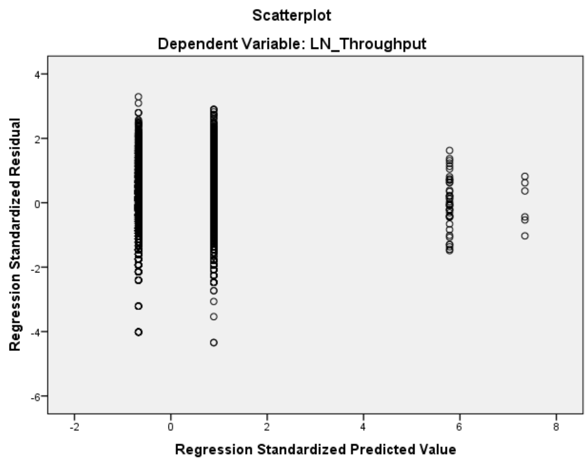
After that the Durbin-Watson value of the linear regression analysis is mentioned after which an overall conclusion is made regarding whether or not the assumptions have been violated too clearly. Also the model summary and coefficient table is provided from which the R^2, F-test, t-tests, Durbin-Watson test and standardized beta coefficients have been read.

Appendix I. Linear Regression Analysis assumptions for the CCB – case: unexpected exceptions









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,205[a]	,042	,041	,85768	,042	87,793	2	4009	,000	1,221

a. Predictors: (Constant), itration, skip

b. Dependent Variable: LN_Throughput

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	3,730	,022		170,229	,000	3,687	3,773			
	skip	-,281	,028	-,156	-10,052	,000	-,335	-,226	-,148	-,157	-,155
	itration	1,159	,126	,142	9,193	,000	,912	1,406	,133	,144	,142

Despite the size of the sample being well over the minimum of 200 after which normality becomes less of an issue, the sample does not violate this assumption as can be seen in the histogram with normality plot and P-P plot.

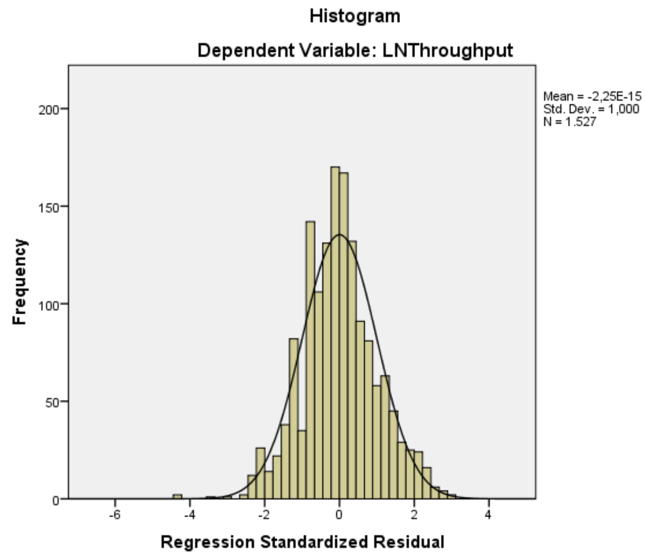
As can also be seen in the plot of the standardized residuals versus standardized predicted values of the regression, there is a fairly clear linearity between the standardized residuals of the regression and the predicted values, showing symmetry across a horizontal line, which means the assumption is not violated.

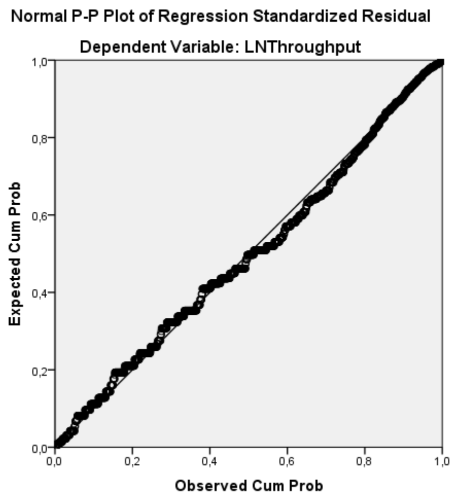
The variance is not entirely equal over the plot of standardized residuals versus time. With standardized residuals versus the predicted value this is a lot clearer. The assumption of homoscedasticity is violated which means that the confidence intervals of the beta coefficients can be off. But this does not have to be a problem.

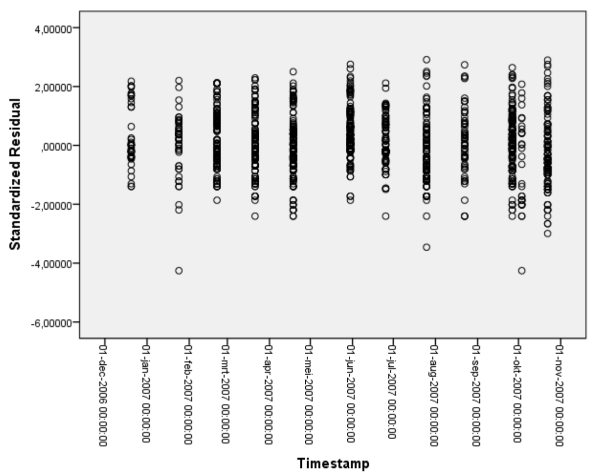
The Durbin-Watson test which shows independence if the value is between 1 and 3, indicates there is some concern about whether the data points are independent or not as the value is at 1.221, showing there might be a slight positive correlation between the errors of the dependent. However independence can be assumed.

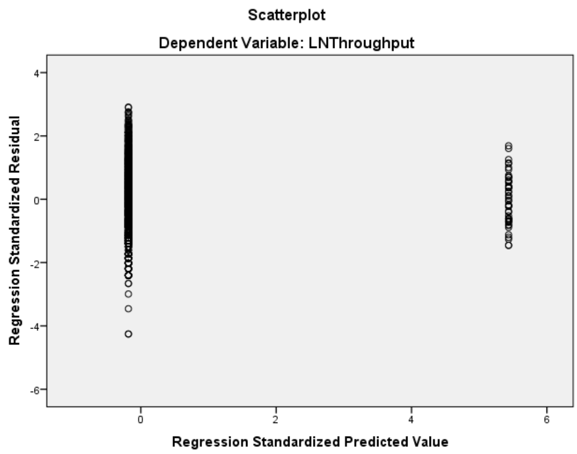
Judging from the assumptions overall the model is seen as appropriate.

Appendix II. Linear Regression Analysis assumptions for the CCB – case: expected exceptions, core flow A









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,213[a]	,046	,045	,86703	,046	72,725	1	1525	,000	1,011

a. Predictors: (Constant), Analysis

b. Dependent Variable: LNThroughput

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	3,695	,023		163,792	,000	3,651	3,739			
	Analysis	1,063	,125	,213	8,528	,000	,819	1,308	,213	,213	,213

Despite the size of the sample being well over the minimum of 200 after which normality becomes less of an issue, the sample does not violate this assumption as can be seen in the histogram with normality plot and P-P plot.

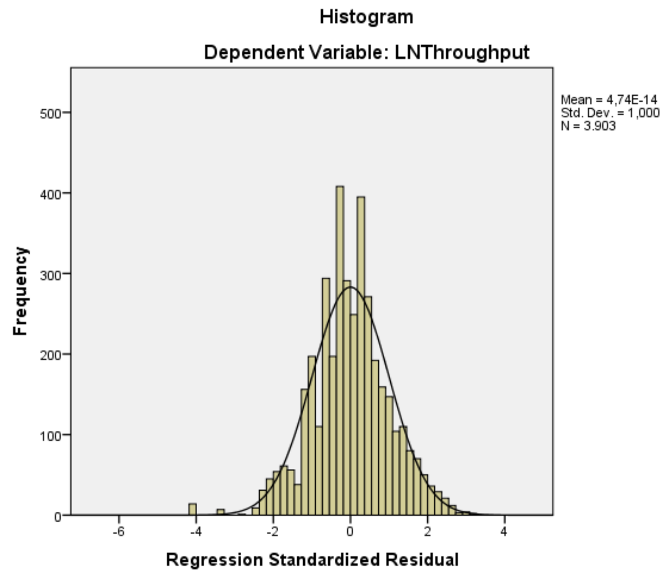
As can also be seen in the plot of the standardized residuals versus standardized predicted values of the regression, there is a fairly clear linearity between the standardized residuals of the regression and the predicted values, showing symmetry across a horizontal line, which means the assumption is not violated.

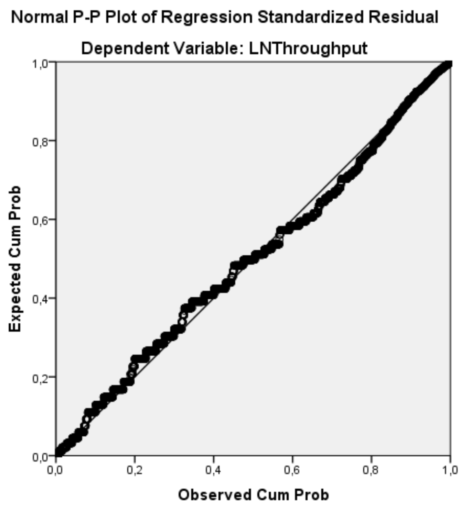
The variance is not entirely equal over the plot of standardized residuals versus time. With standardized residuals versus the predicted value this is a lot clearer. The assumption of homoscedasticity is violated which means that the confidence intervals of the beta coefficients can be off. But this does not have to be a problem.

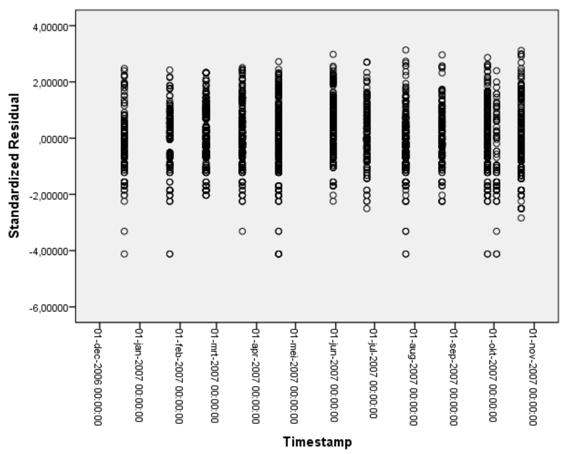
The Durbin-Watson test which shows independence if the value is between 1 and 3, indicates there is some concern about whether the data points are independent or not as the value is at 1.011, showing there might be some positive correlation between the errors of the dependent. However independence can be assumed.

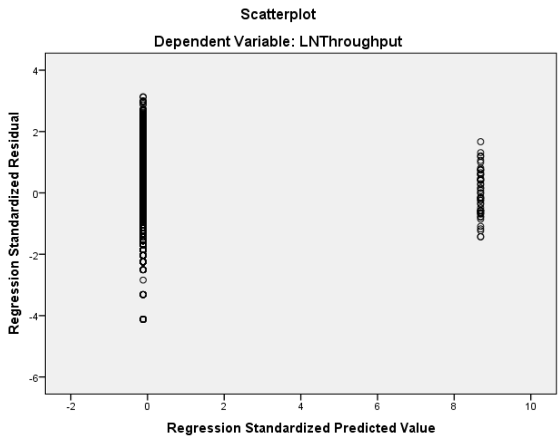
Judging from the assumptions overall the model is seen as appropriate.

Appendix III. Linear Regression Analysis assumptions for the CCB – case: expected exceptions, core flow B









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,156[a]	,024	,024	,85629	,024	96,784	1	3901	,000	1,233

a. Predictors: (Constant), Analysis

b. Dependent Variable: LNThroughput

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	3,533	,014		256,048	,000	3,506	3,560			
	Analysis	1,187	,121	,156	9,838	,000	,951	1,424	,156	,156	,156

The sample does not violate the assumption of normality as can be seen in the histogram with normality plot and P-P plot.

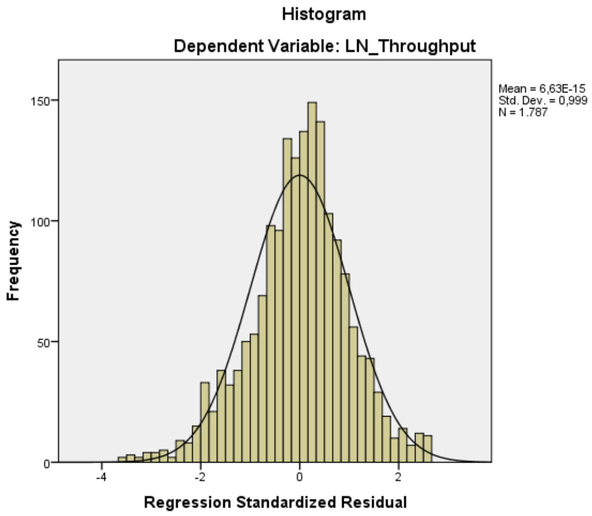
As can also be seen in the plot of the standardized residuals versus standardized predicted values of the regression, there is a fairly clear linearity between the standardized residuals of the regression and the predicted values, showing symmetry across a horizontal line, which means the assumption is not violated.

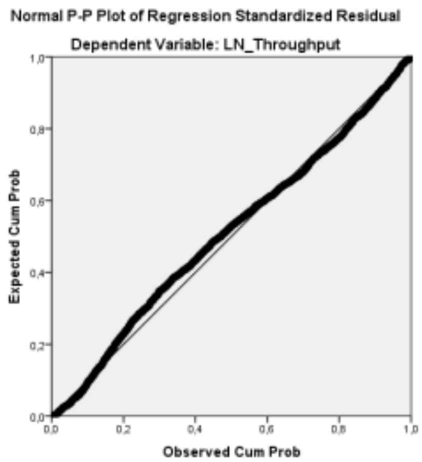
The variance is not entirely equal over the plot of standardized residuals versus time. With standardized residuals versus the predicted value this is a lot clearer. The assumption of homoscedasticity is violated which means that the confidence intervals of the beta coefficients can be off. But this does not have to be a problem.

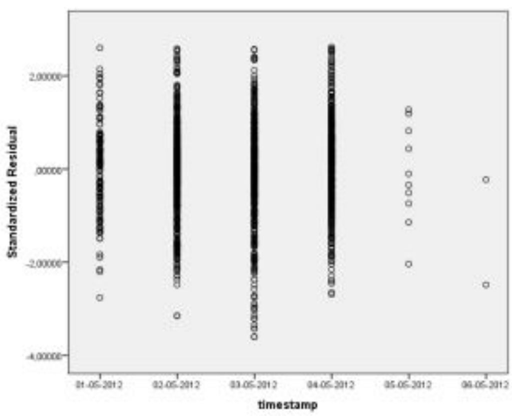
The Durbin-Watson test which shows independence if the value is between 1 and 3, indicates there is some concern about whether the data points are independent or not as the value is at 1.233, showing there might be some positive correlation between the errors of the dependent. However independence can be assumed.

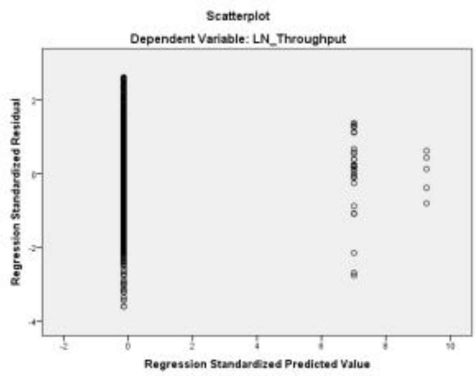
Judging from the assumptions overall the model is seen as appropriate.

Appendix IV. Linear Regression Analysis assumptions for the Volvo case: unexpected exceptions, simple cases









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,182[a]	,033	,032	1,04127	,033	30,403	2	1784	,000	1,744

a. Predictors: (Constant), iteration, skip

b. Dependent Variable: LN_Throughput

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	-3,942	,200		-19,674	,000	-4,335	-3,549			
	skip	-1,372	,202	-,158	-6,795	,000	-1,768	-,976	-,158	-,159	-,158
	iteration	1,805	,466	,090	3,870	,000	,890	2,719	,089	,091	,090

Despite the size of the sample being well over the minimum of 200 after which normality becomes less of an issue, the sample does not (very much) violate this assumption as can be seen in the histogram with normality plot and P-P plot.

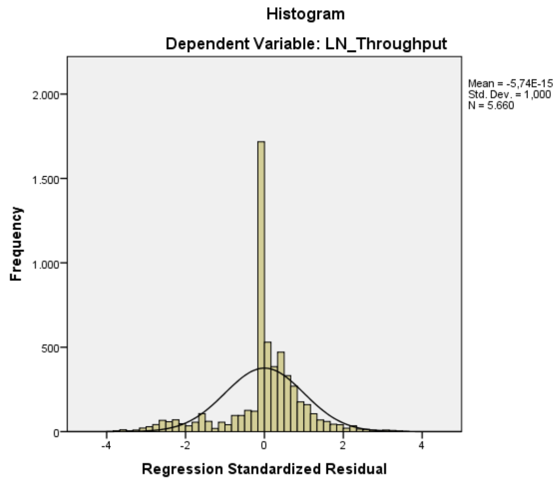
As can also be seen in the plot of the standardized residuals versus standardized predicted values of the regression, there is a fairly clear linearity between the standardized residuals of the regression and the predicted values, showing symmetry across a horizontal line, which means the assumption is not violated.

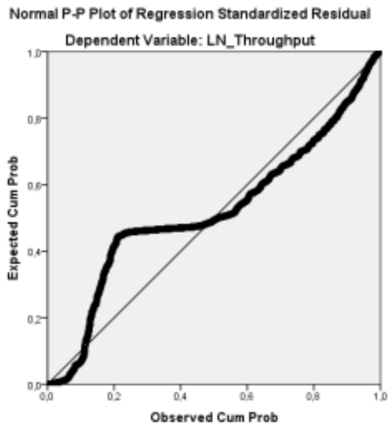
The variance is not entirely equal over the plot of standardized residuals versus time. With standardized residuals versus the predicted value this is a lot clearer. The assumption of homoscedasticity is violated which means that the confidence intervals of the beta coefficients can be off. But this does not have to be a problem.

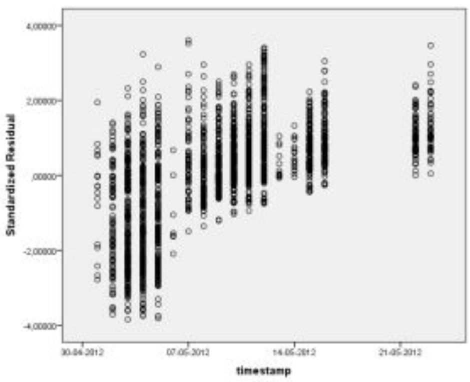
The Durbin-Watson test which shows independence if the value is between 1 and 3, indicates there is some concern about whether the data points are independent or not as the value is at 1.744, showing there might be some positive correlation between the errors of the dependent. However independence can be assumed.

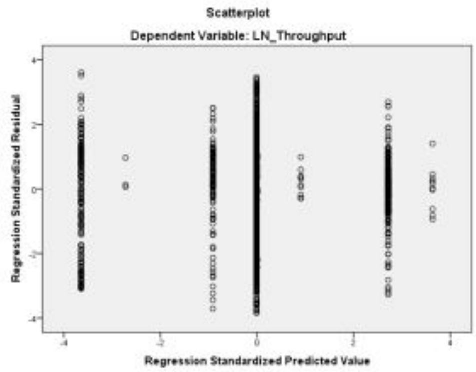
Judging from the assumptions overall the model is seen as appropriate.

Appendix V. Linear Regression Analysis assumptions for the Volvo case: unexpected exceptions, complex cases









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,223[a]	,050	,049	1,16846	,050	99,124	3	5656	,000	,980

a. Predictors: (Constant), loop, skip, iteration

b. Dependent Variable: LN_Throughput

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	2,118	,017		128,347	,000	2,086	2,151			
	skip	-,972	,066	-,200	-14,674	,000	-1,102	-,842	-,147	-,192	-,190
	iteration	,730	,058	,172	12,585	,000	,616	,844	,117	,165	,163
	loop	,246	,183	,018	1,344	,179	-,113	,605	,017	,018	,017

The normality assumption has been violated when looking at the first two graphs. The size of the sample is well over the minimum of 200 after which normality becomes less of an issue, so the sample does not (necessarily) violate this assumption.

There is no linearity between the standardized residuals of the regression and the predicted values, with no symmetry across a horizontal line, which means the assumption is violated.

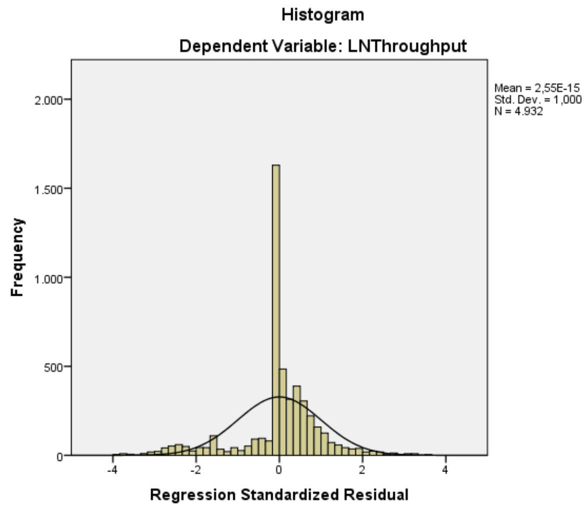
The variance is not equal over the plot of standardized residuals versus time. With standardized residuals versus the predicted value this is also the case. The assumption of homoscedasticity is clearly violated which means that the confidence intervals of the beta coefficients are off.

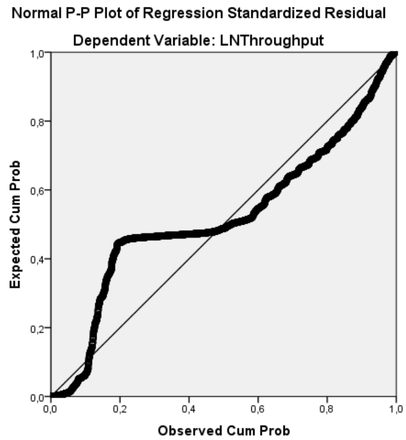
The Durbin-Watson test does not show independence with the value below one. There is a positive correlation between the errors of the dependent with a value of 0.980. The assumption of independence is clearly violated.

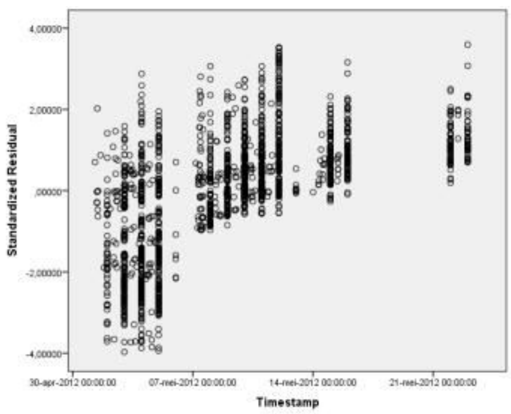
Judging from the assumptions the model is not fit for a linear regression analysis.

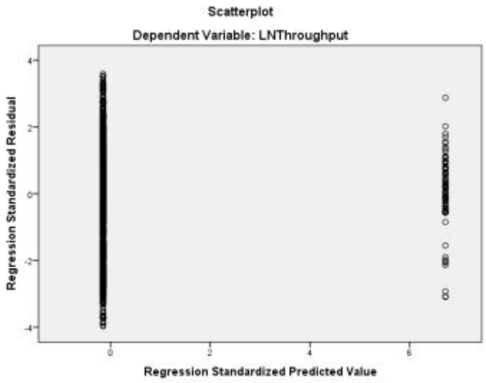
Appendix VI. Linear Regression Analysis assumptions for the Volvo case: expected exceptions, complex cases

No analysis was done for simple cases when it comes to expected exceptions as no exceptions were found in the simple cases.









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,070[a]	,005	,005	1,12904	,005	23,934	1	4930	,000	,914

a. Predictors: (Constant), AnalysisException

b. Dependent Variable: LNThroughput

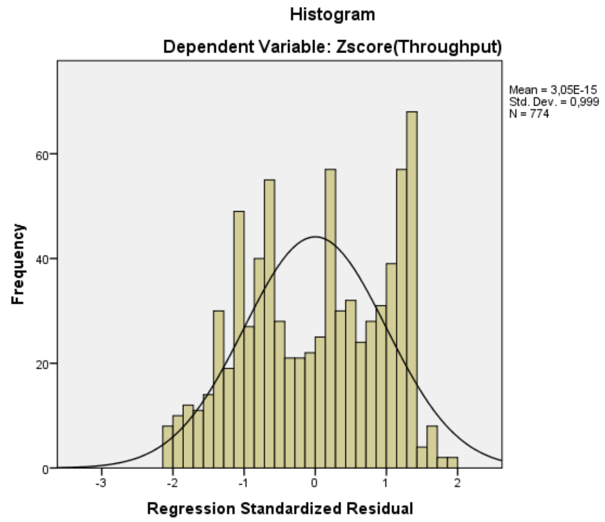
Coefficients[a]

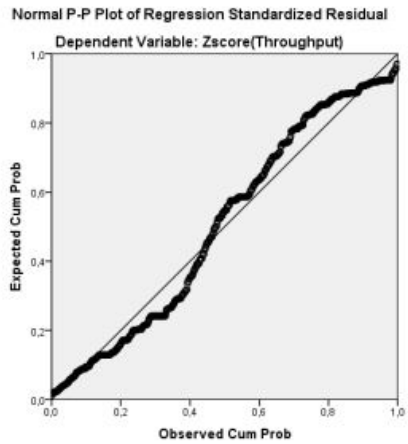
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	2,111	,016		129,861	,000	2,079	2,143			
	AnalysisException	,540	,110	,070	4,892	,000	,324	,756	,070	,070	,070

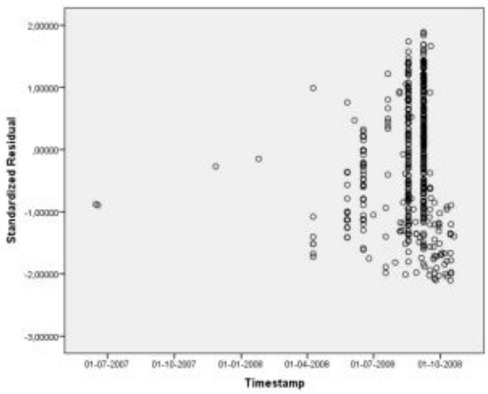
The simple linear regression with throughput time log normalized and the extra analysis exception shows the exception cannot predict the throughput time well. The assumptions of normality, independence (Durbin-Watson value below 1) and homoscedasticity (no equal variance of standardized residuals) are violated, although with a sample of over 200 the violation of normality is less of a concern. Only the assumption of linearity is not violated and even then the relation between the exception type of extra analysis and throughput time is very minimal.

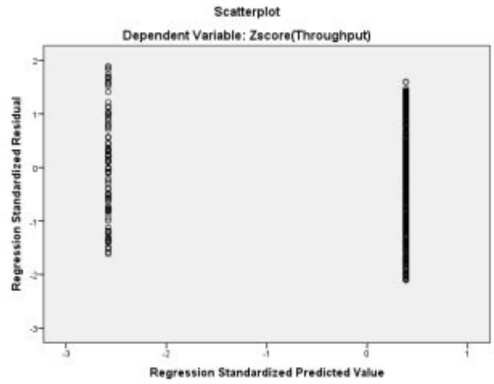
Judging from the assumptions the model is not fit for a linear regression analysis.

Appendix VII. Linear Regression Analysis assumptions for the Eindhoven Municipality Appeals case: early exit exceptions









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,164[a]	,027	,026	,98710257	,027	21,332	1	772	,000	,090

a. Predictors: (Constant), EarlyExit

b. Dependent Variable: Zscore(Throughput)

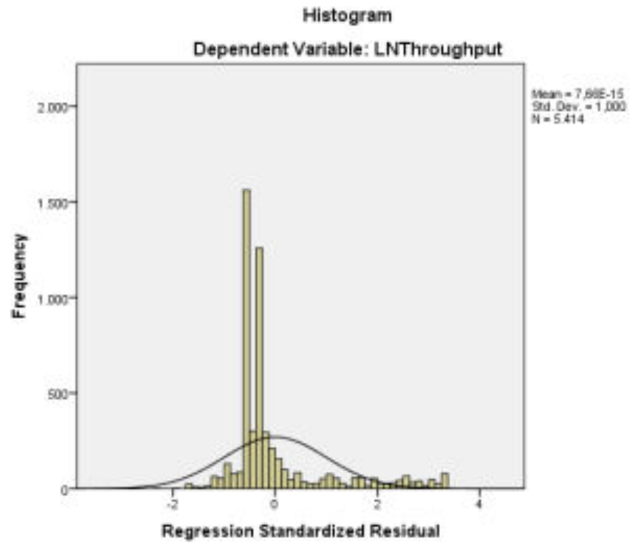
Coefficients[a]

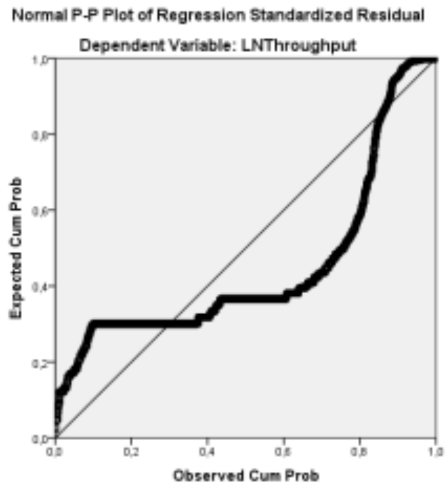
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	,063	,038		1,668	,096	-,011	,138			
	EarlyExit	-,486	,105	-,164	-4,619	,000	-,693	-,280	-,164	-,164	-,164

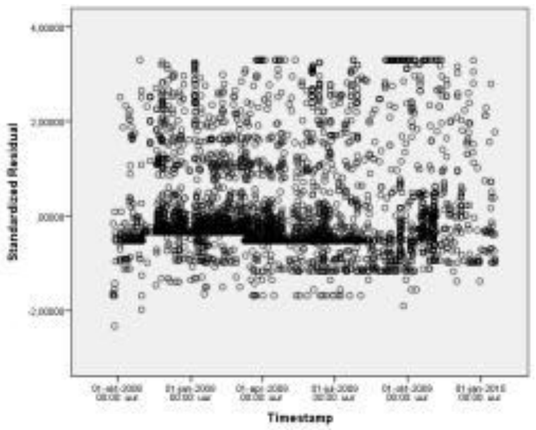
All assumptions have been violated with a heavy violation of independence and homoscedasticity of the standardized residuals of the dependent variable. Over time the variation of standardized residuals fans out completely and the Durbin-Watson value is only 0.090.

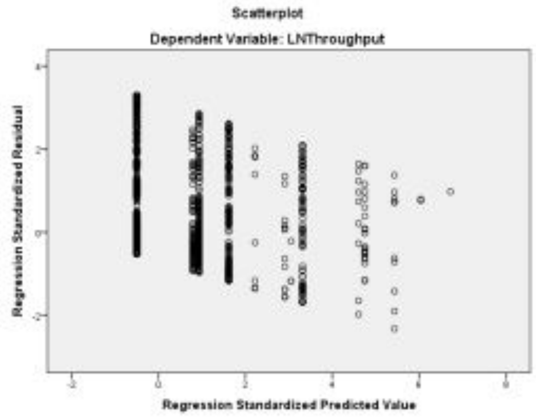
Judging from the assumptions the model is not fit for a linear regression analysis.

Appendix VIII. Linear Regression Analysis assumptions for the Big US Company Incident Management: exceptions, simple cases









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,291[a]	,085	,084	1,17661	,085	125,336	4	5409	,000	1,371

a. Predictors: (Constant), EarlyExit, Loop, Iterate, Skip

b. Dependent Variable: LNThroughput

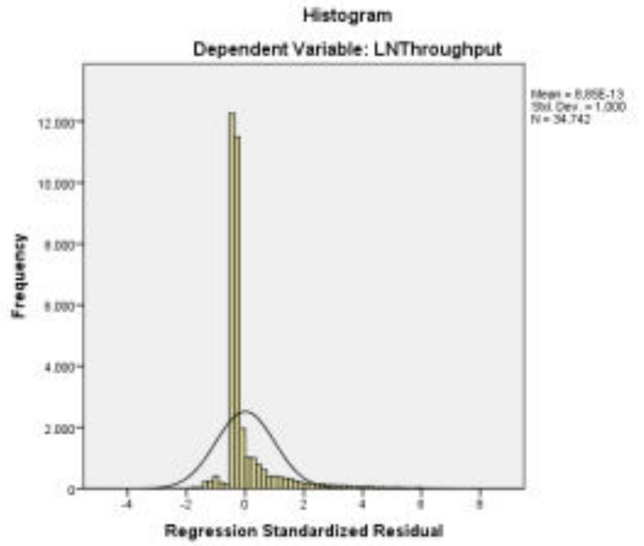
Coefficients[a]

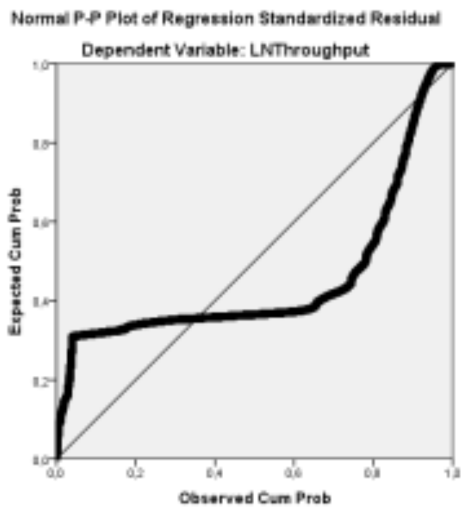
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	-1,158	,018		-63,334	,000	-1,193	-1,122			
	Skip	,513	,051	,132	10,110	,000	,414	,613	,117	,136	,132
	Iterate	1,366	,093	,191	14,663	,000	1,183	1,548	,195	,196	,191
	Loop	,462	,095	,063	4,868	,000	,276	,647	,071	,066	,063
	EarlyExit	,758	,057	,173	13,255	,000	,646	,870	,159	,177	,172

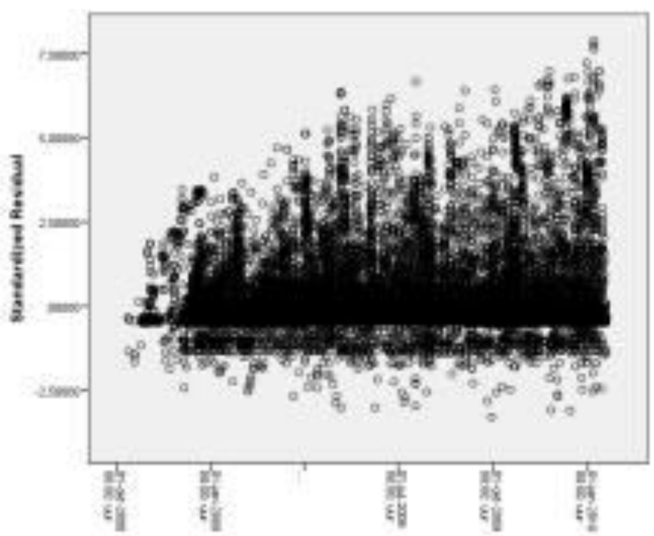
Except for independence with a Durbin-Watson score of 1.371, although the dependent showing some positive autocorrelation, as well as homoscedasticity, with the variance of standardized residuals being equal across time, the normality and linearity assumptions are violated. The F-test, is however passed, showing a model fit. Yet this cannot be taken as reliable results.

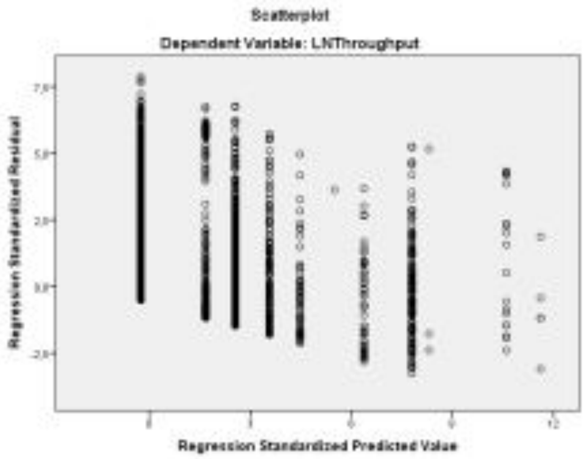
Judging from the assumptions the model is not fit for a linear regression analysis.

Appendix IX. Linear Regression Analysis assumptions for the Big US Company Incident Management: exceptions, complex cases









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,330[a]	,109	,109	,37755	,109	1061,787	4	34737	,000	1,785

a. Predictors: (Constant), Cancel, Loop, Skip, Iterate

b. Dependent Variable: LNThroughput

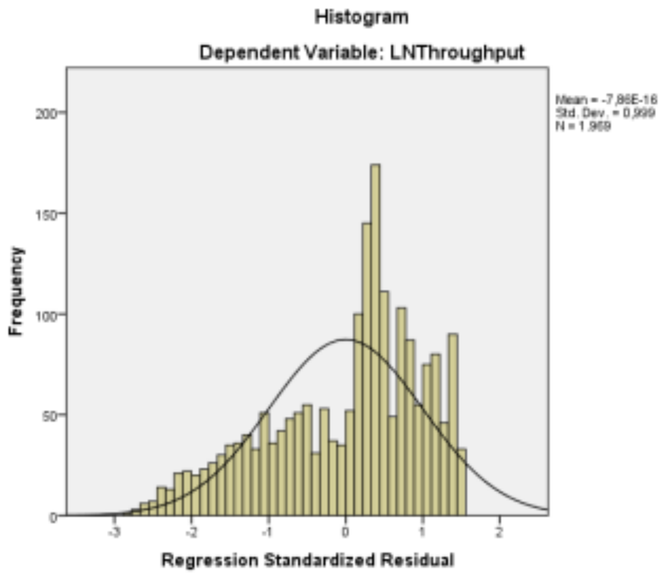
Coefficients[a]

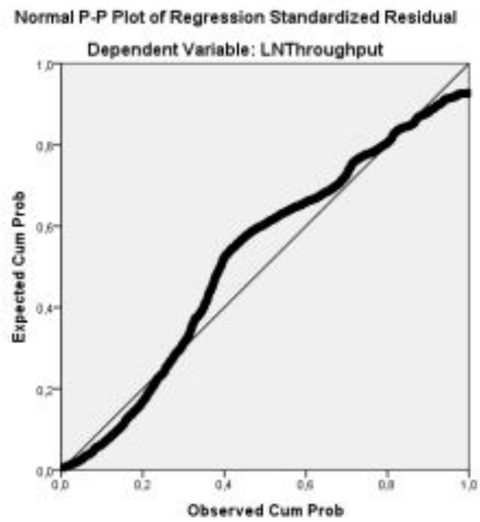
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	2,909	,002		1380,946	,000	2,905	2,913			
	Skip	,505	,023	,113	22,271	,000	,460	,549	,133	,119	,113
	Iterate	,371	,009	,203	39,824	,000	,352	,389	,218	,209	,202
	Loop	,254	,015	,088	17,427	,000	,226	,283	,093	,093	,088
	EarlyExit	1,063	,027	,202	39,888	,000	1,011	1,115	,205	,209	,202

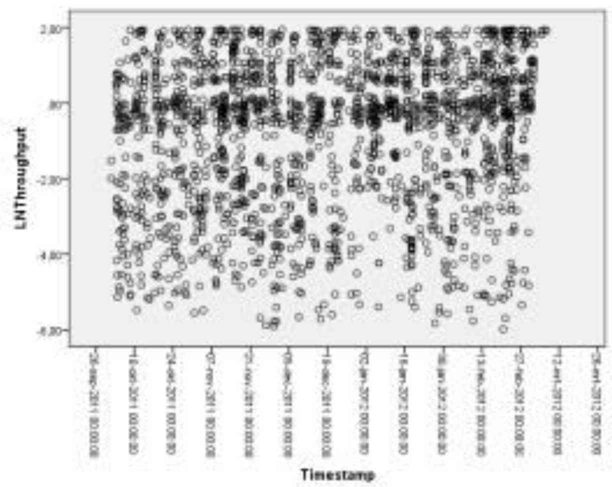
Except for independence with a Durbin-Watson score of 1.785, all assumptions are violated. The F-test, is however passed, showing a model fit. Yet this cannot be taken as reliable results.

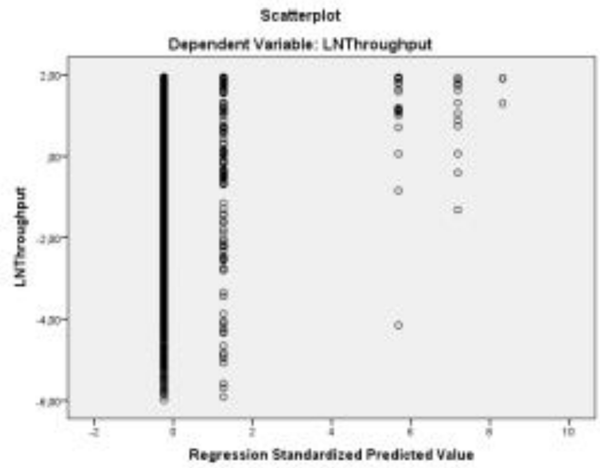
Judging from the assumptions the model is not fit for a linear regression analysis.

Appendix X. Linear Regression Analysis assumptions for the Dutch financial institute's loan application process: exceptions, simple cases









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,154[a]	,024	,022	1,91227	,024	15,919	3	1965	,000	1,658

a. Predictors: (Constant), Cancel, Skip, Iterate

b. Dependent Variable: LNThroughput

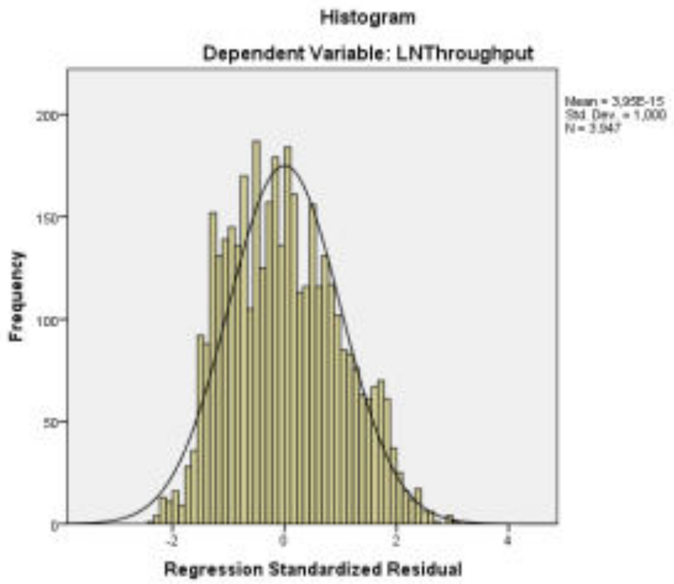
Coefficients[a]

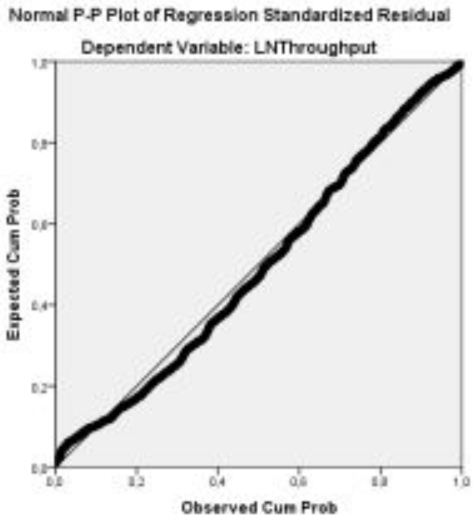
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	-,840	,045		-18,651	,000	-,928	-,751			
	Skip	,784	1,149	,016	,682	,495	-1,470	3,039	,050	,015	,015
	Iterate	1,765	,321	,129	5,500	,000	1,135	2,394	,141	,123	,123
	EarlyExit	,447	,166	,061	2,687	,007	,121	,773	,078	,061	,060

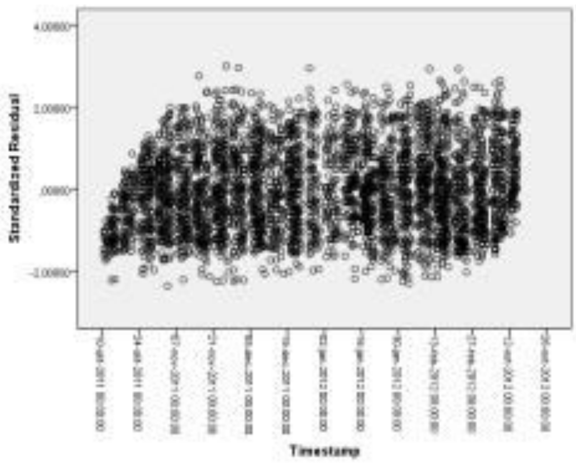
The assumptions of normality, linearity and homoscedasticity are violated, but not that of independence with a Durbin-Watson value of 1.658. That means that even though the Adjusted R^2 show there is a (little bit of) model fit, this is not to be trusted.

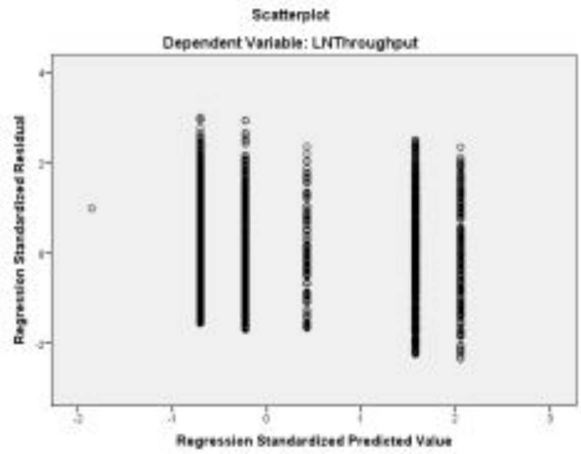
Judging from the assumptions the model is not fit for a linear regression analysis.

Appendix XI. Linear Regression Analysis assumptions for the Dutch financial institute's loan application process: exceptions, complex cases









Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	,292[a]	,085	,084	,42960	,085	122,151	3	3943	,000	1,933

a. Predictors: (Constant), EarlyExit, Iterate, Skip

b. Dependent Variable: LNThroughput

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95,0% Confidence Interval for B		Correlations		
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part
1	(Constant)	2,614	,009		294,080	,000	2,597	2,632			
	Skip	-,151	,040	-,060	-3,739	,000	-,229	-,072	,022	-,059	-,057
	Iterate	,298	,016	,298	18,707	,000	,267	,330	,279	,286	,285
	EarlyExit	,063	,017	,058	3,769	,000	,030	,095	,055	,060	,057

The assumption of normality is not clearly violated and neither is that of linearity, though there is some indication that the residuals are not spread perfectly symmetrically over a horizontal line, but a very slightly sloped one. Homoscedasticity is also not clearly violated with in the beginning less variation, but for the most part a constant spread of standardized residuals over time. The independence assumption with a Durbin-Watson value of 1.933 is not violated.