Business processes exceptions in relation to operational performance

Citation for published version (APA):

Dijkman, R., Turetken, O., van IJzendoorn, G. R., & de Vries, M. (2019). Business processes exceptions in relation to operational performance. *Business Process Management Journal*, 25(5), 908-922.
<https://doi.org/10.1108/BPMJ-07-2017-0184>

Document license:
TAVERNE

DOI:
[10.1108/BPMJ-07-2017-0184](https://doi.org/10.1108/BPMJ-07-2017-0184)

Document status and date:
Published: 02/09/2019

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

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

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

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

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

openaccess@tue.nl

providing details and we will investigate your claim.

Business processes exceptions in relation to operational performance

Remco Dijkman, Oktay Turetken and Geoffrey Robert van IJzendoorn
Technische Universiteit Eindhoven, Eindhoven, The Netherlands, and
Meint de Vries
Teradata, Amsterdam, The Netherlands

Abstract

Purpose – Business process models describe the way of working in an organization. Typically, business process models distinguish between the normal flow of work and exceptions to that normal flow. However, they often present an idealized view. This means that unexpected exceptions – exceptions that are not modeled in the business process model – can also occur in practice. This has an effect on the efficiency of the organization, because information systems are not developed to handle unexpected exceptions. The purpose of this paper is to study the relation between the occurrence of exceptions and operational performance.

Design/methodology/approach – The paper does this by analyzing the execution logs of business processes from five organizations, classifying execution paths as normal or exceptional. Subsequently, it analyzes the differences between normal and exceptional paths.

Findings – The results show that exceptions are related to worse operational performance in terms of a longer throughput time and that unexpected exceptions relate to a stronger increase in throughput time than expected exceptions.

Practical implications – These findings lead to practical implications on policies that can be followed with respect to exceptions. Most importantly, unexpected exceptions should be avoided by incorporating them into the process – and thus transforming them into expected exceptions – as much as possible. Also, as not all exceptions lead to longer throughput times, continuous improvement should be employed to continuously monitor the occurrence of exceptions and make decisions on their desirability in the process.

Originality/value – While work exists on analyzing the occurrence of exceptions in business processes, especially in the context of process conformance analysis, to the best of the authors' knowledge this is the first work that analyzes the possible consequences of such exceptions.

Keywords Operational performance, Business process management, Process mining, Conformance, Exception
Paper type Research paper

1. Introduction

Organizations often describe their flow of operations in terms of business process models. Tools for operations support, such as information systems, forms, and manuals, are then developed based on these business process models. However, in practice exceptions can occur that disrupt the normal flow of operations (Reichert and Weber, 2012b). The tools for operations support, which have been defined for the normal flow of operations, cannot be expected to provide optimal support in case of an exception. For that reason, we expect that the performance of business processes deteriorates when exceptions occur.

For example, consider a bank that has a process for applying for a bank account. The application process is supported by several forms that a new customer has to fill out, manuals that explain to the bank employee what to do during the various steps of the application procedure, and information systems for recording customer details and activating the bank account. There is likely to be a normal flow through the process that most customers follow and that can be dealt with efficiently, because the employees have been trained for and have experience with that normal flow, and because all forms, work instructions, and information systems are optimized for it. It is also likely that the bank

pre-defined certain exceptions to the normal flow that are expected to occur incidentally, such as the situation in which a customer fills out a form incorrectly. Such exceptions may lead to delays in the process and a decrease in quality and customer satisfaction, simply because more work is required, but also because employees have less experience with these situations, depending on how frequently they occur. We will show in this paper that exceptions that were not considered when designing a process, form, or information system are also surprisingly frequent. Such exceptions can be expected to further decrease the efficiency and quality of the work. Just remember the time that a helpdesk employee told you: “I am sorry Sir/Madam, but our system does not allow for that.”

The goal of this paper is to investigate the relation between different types of exceptions in a business process and the performance of that business process. To achieve this goal, the paper provides a classification of the different types of exceptions that can occur during the execution of a business process. Subsequently, it presents a statistical analysis of the relations between the different types of exceptions and the performance of a business process – in particular the throughput time. To the best of our knowledge, such an analysis has not been made before.

This has important practical implications. First, there exist many papers that propose solutions for handling exceptions (Schonenberg *et al.*, 2008), based on the premise that exceptions are a fact of life and deteriorate process performance when not properly supported. This study validates that premise. Second, this study provides a theoretical basis for showing organizations how different types of exceptions should be supported in practice.

The study is conducted by investigating five processes in five different companies and a total of nearly 70,000 customer cases that have been executed for these processes. Of these customer cases more than 40,000 contained some form of exception. These numbers are sufficient to do a statistical analysis for each of the business processes. Subsequently, we aggregate the results of the statistical analyses in a qualitative manner.

Against this background, the remainder of this paper is structured as follows. Section 2 presents the background of the research by providing a conceptual background on exceptions and an overview of related work. Section 3 discusses the research methodology. Sections 4 and 5 present the analysis of the data and the results of the study, and Section 6 presents the conclusions.

2. Background

A business process is a structured, measured set of activities designed to produce a specific output for a particular customer (Davenport, 1993). Business processes are the structure by which an organization does what is necessary to produce value for its customers. A business process model is a (graphical) representation of a business process. Figure 1 shows an example of a business process model, in the notation of automated process discovery tool “Disco[1]”. The process starts at the “play” icon and stops at the “stop” icon. Rounded rectangles represent tasks that can be executed. An arrow represents that if the task at

Figure 1.
A business
process model

the source of the arrow is completed, the task at the target of the arrow can be started. When multiple arrows leave a task, this represents a choice between the target tasks. For example, after “register appeal” there is a choice between “request revision,” “register documents,” and “reject appeal.” Multiple incoming arcs to a task also indicate an (exclusive) choice that triggers the task. For instance, “request revision” task can be triggered by “request appeal” or “register documents.”

Each customer case follows a certain path through the process. For example, for the process above, a possible path is: “register appeal,” “reject appeal,” “send advice,” and “archive.” Another possible path is “register appeal,” “request revision.”

Different paths through the process may lead to a different outcome of the process. For example, a path that contains “request revisions” has a different outcome than a path that contains “accept appeal.” We call a collection of paths that leads to the same outcome a scenario. The process from Figure 1 has three different scenarios: a scenario for accepting an appeal, a scenario for rejecting an appeal, and a scenario for requesting revisions. However, there are different paths for each scenario. For example, an appeal can be rejected via the path “register appeal,” “reject appeal,” “send advice,” “archive,” but also via the path “register appeal,” “reject appeal,” “register documents,” “send advice,” and “archive.”

Different paths also have a different frequency in which they are being taken by customer cases. Each scenario has a most frequently followed path, which will be called the normal flow of the scenario in the remainder of this paper. An exception is a disruption of the normal flow of operations (Reichert and Weber, 2012b). While there exist business process modeling languages that have notational elements for modeling exceptions, such as the business process model and notation (Object Management Group, 2011), such notational elements are not frequently used (zur Muehlen and Recker, 2008). Consequently, it is often hard to determine which paths through a process represent normal behavior and which paths represent exceptional behavior (Grigori *et al.*, 2001). In this paper, we define an exception as a path through a process that is not the normal flow of a scenario. This position is based on the observation that non-normal flows are relatively rare, as we will show in Section 4, such that they are exceptional in the dictionary sense of the word. We should note that, this definition adopts a rather restricted view of process exceptions in the sense that it focuses only on the control-flow aspect of processes and does not cover exceptions regarding other aspects such as resources (roles, actors, etc.) or information.

A clearly identifiable form of exception is an exception that causes a customer case to follow a path that is not represented by the process model. For example, if a customer follows the path “register appeal,” “request revision,” “accept appeal,” “send advice,” “archive” through the business process from Figure 1, this would be an exception to the normal flow of operations. Consequently, we define an unexpected exception as a path through a process that is not described by the process model and an expected exception as an exception that is described by the process model.

Looking in more detail at the paths that are taken in the process outlined above, a more fine-grained distinction between exceptions can be made. Considering a normal flow A–B–C–D, we define an exception as:

- an early exit, when we remove an activity (in our case D) from the end of the path;
- a late exit, when we add an activity to the end of the path;
- an early entry, when we remove an activity (in our case A) from the start of the path;
- a late entry, when we add an activity to the start of the path;
- a repeat, when a single activity is repeated (e.g. A–B–C–C–D);

- a step back, when a sequence of multiple activities is repeated (e.g. A–B–C–B–C–D);
- an add, when an activity is added to the normal flow, but it is not a late exit, late entry, repeat, or step back; and
- a skip, when an activity is removed from the normal flow, but it is not an early entry, or an early exit.

There is a large body of literature on exception handling in business processes. Various papers discuss the topic from different perspectives. First, there are papers that conceptualize the notion of exception and related notions (Curbera *et al.*, 2003; Eder and Liebhart, 1998; Strong, 1997). These papers agree that an exception in a business process is a deviation from what is considered to be the normal flow of a business process. It should be noted that the normal flow covers multiple scenarios as explained above. Second, there exist papers that present the various types of exceptions that can occur during the execution of a business process (Russell *et al.*, 2006; Weber *et al.*, 2008). These exception types are defined in terms of small patterns of behavior, such as: redoing a task that was already performed is considered to be an exception. From a technical perspective, there exists a line of research on developing techniques for detecting whether an exception has occurred and even predicting whether an exception will occur for a certain customer case (Grigori *et al.*, 2001, 2004; Hwang *et al.*, 1999; Rozinat and van der Aalst, 2008). In addition to that there exists a line of research on developing techniques for handling exceptions when they occur (Adams *et al.*, 2006; Zeng *et al.*, 2005; Hagen and Alonso, 2000; Rinderle and Reichert, 2006; Gottschalk *et al.*, 2009; van der Aalst, 2009; Hallerbach *et al.* 2010; Meerkamm, 2012; Reichert and Dadam, 1998; Reichert and Weber, 2012a). These techniques have been implemented into business process management tools, such as ADEPT (Reichert and Dadam, 1998) and YAWL (Adams *et al.*, 2006). Schonenberg *et al.* (2008) provide a literature survey of those tools and techniques. In the remainder of our work, we use the definitions and patterns that are identified in this related work. We then contribute to it by investigating the relations between the occurrence of exceptions and exception patterns and throughput time.

3. Research methodology

Based on the discussion from the previous section, the overall research goal can be refined into hypotheses. In particular, the following hypotheses follow from the discussion:

- H1.* Paths through a process that involve an exception have a longer throughput time than paths through a process that do not involve an exception.
- H2.* Paths that involve an unexpected exception have a longer throughput time than paths that involve an expected exception.
- H3.* Paths that involve an exception that adds activities to the normal flow have a longer throughput time than the normal flow.
- H4.* Paths that involve an exception that removes activities from the normal flow have a shorter throughput time than the normal flow.

H1 and *H4* seem to contradict each other. However, note that *H1* concerns all exceptions, while *H4* only concerns a subset of the exceptions; those exceptions that remove activities from the normal flow. If this subset of exceptions is relatively small, it is still possible that, overall, exceptions are related to an increase in throughput time.

For reasons of availability of data, this paper focuses on the operational performance in terms of throughput time. In future work, we aim to study other key performance indicators, such as customer satisfaction.

The following research steps were followed to investigate the hypotheses. First, we selected the processes to be analyzed and gathered process-related data including the process models, where available, and process instance data (cases). We selected five processes from five companies operating in different business domains. For each of the five processes, we gathered information on the customer cases that flowed through the process during a certain time period.

Second, using process-mining techniques, we analyzed the cases for each process to determine the paths that are followed through the process and corresponding throughput time. This information allows us to determine whether the customer case contained any exceptions and to compare the throughput time between customer cases. For two of the five processes, we also have the original process model, which allows us to investigate differences between expected and unexpected exceptions.

Finally, we performed a set of statistical analyses with the data originated from the previous step to test our hypotheses. First, we performed the analysis on the customer cases of each process separately, in which we determined significant differences between the throughput times of customer cases, depending on the types of exception that they contain. Based on this analysis, we established per process and per type of exception whether it is associated with a significantly longer or shorter throughput time, both compared to the normal flow and compared to the flow of work that is described by the process model. Second, we aggregated the results over the processes, using qualitative arguments. When the same type of exception has the same type of relation for multiple processes, it is considered that that type of exception indeed follows that type of relation in the general case.

4. Descriptive statistics

Table I shows the descriptive statistics of the processes that are investigated, including the type of process used, the start date of the measurements, the end date of the measurements, the number of customer cases measured, and the average throughput times of the customer cases and the standard deviation of the throughput times.

The first process is an appeals process at the municipality of Eindhoven. This process concerns appeals by citizens for a revision of a decision that was made by the municipality at an earlier stage. There are five possible outcomes of this process. The appeal can be accepted, rejected, withdrawn by the citizen, sent on to another department, or deferred awaiting a revision of the request from the citizen. These outcomes would seem suitable as scenarios. However, the data show that it is possible to have “multiple outcomes” at the same time, due to the fact that the different parties involved can take decisions independently. For example, it is possible that the citizen withdraws the appeal and the municipality rejects the appeal. Since it is impossible to decide on the exact outcome in such cases, we consider these outcomes separately and count them as different scenarios.

The second process is a change control process at a consumer electronics manufacturing company. This process concerns evaluating whether change requests should be incorporated into a product or not. From a process perspective, there is only one possible scenario in this

Table I.
Descriptive statistics
of processes

Process	Start date	End date	No. of cases	Avg. throughput	Std. throughput
P1 – appeals	October 24, 2006	October 20, 2008	1,268	41 weeks	31 weeks
P2 – change control	March 9, 2006	October 22, 2007	4,014	52 days	55 days
P3 – incident management	September 8, 2008	January 1, 2010	41,902	19 days	21 days
P4 – loan application	October 1, 2011	March 14, 2012	13,085	8 days	12 days
P5 – incident management	May 5, 2011	May 23, 2012	7,554	12 days	29 days

process: the scenario in which the change request is handled. The process does not have different paths for change requests that are rejected and change requests that are accepted. Therefore, we cannot distinguish these as different scenarios. For this process, we originally had nearly two years of data. However, halfway the measurement period, there was a change to the process, such that customer cases that finished before that date were handled differently from customer cases that were finished on or after that date. To not pollute the measurements with this, we only included customer cases from the most recent period.

The third process is an incident management process at an American company. This process concerns handling incidents that are reported to the company, from the moment that they are submitted until the moment that they are handled. This process has two possible outcomes and corresponding scenarios. Either the incident is handled, or the incident report is canceled.

The fourth process is a loan application process at a Dutch financial institution (van Dongen, 2012). This process concerns applications for loans, from the moment that they are submitted by the client until the moment that they are accepted, rejected, or canceled by the customer. This process has three possible outcomes and corresponding scenarios. The loan can be accepted, rejected, or the loan application can be canceled.

The fifth process is an incident management process at Volvo (Steeman, 2013). Similar to the incident management process from the third process, this process also concerns handling incidents, from the moment that they are submitted until the moment that they are handled. However, the number and flow of tasks that are used in this process varies from those in the third process. This process has two possible outcomes and corresponding scenarios. Either the incident report is handled till completion by the helpdesk, or it is handled by the second or third line.

Table II shows the different scenarios that occur in each of the processes that we investigate. For each scenario, it also shows the number of possible paths through that scenario, distinguishing between the normal flow of the scenario (of which there is one by definition) and exceptions to that normal flow (of which there are typically more than one). It then shows the average number of customer cases through each path.

The information in Table II shows that an exceptional path on average is taken far less frequently than a normal flow. This is more clearly illustrated in the graph in Figure 2, which shows the 15 most frequently taken paths through each process and the percentage of customer cases that take these paths. The figure shows that each process has one path that is by far the most traversed path. Except for the second process, the next most traversed path takes less than half of the number of customer cases that the first path takes. After that there are a number of paths that are followed by a distinguishable number of customer cases. The number of customer cases per path then rapidly decreases and the far majority of the paths (98.5 percent) take less than 1 percent of the customer cases.

Figure 2 also shows that there are exceptional paths that are followed as frequently as the normal flows. Therefore, the precise definition of exceptional path can be debated. Clearly paths that are followed twice as often as other paths are normal flows, while paths that are taken by less than 1 percent of the customer cases are exceptions. However, the question remains where between these two extremes we consider a path an exceptional path. There exist rules of thumb, such as the elbow principle (Tibshirani *et al.*, 2001) and the Pareto, or 80/20, principle. However, further analysis shows that these rules do not apply here. Using the elbow principle would mean excluding paths that clearly represent common process behavior, such as rejecting loans. Using the Pareto principle would mean including a large number of paths that are taken by less than 1 percent of the customers. Another logical assumption would be that there is a relation between the paths that are described in the process model and exceptions, but that is also not the case. There are paths in the process models that are taken by less than 1 percent of the customer cases. Moreover, the

Table II.
Scenarios per process

Process	Scenario	Type of path	No. of paths	Avg. no. of cases per path
P1	S1 – Accepted	Normal	1	498
		Exceptions	7	40
	S2 – Withdrawn	Normal	1	78
		Exceptions	9	11
	S3 – Rejected	Normal	1	57
		Exceptions	8	13
	S4 – Deferred	Normal	1	47
		Exceptions	9	5
	S5 – Sent on	Normal	1	27
	S6 – Withdrawn and rejected	Normal	1	13
		Exceptions	6	4
	S7 – Deferred and rejected	Normal	1	4
		Exceptions	1	1
	S8 – Deferred and withdrawn	Normal	1	1
		Exceptions	1	1
P2	S1 – Handled	Normal	1	2,400
		Exceptions	44	37
P3	S1 – Handled	Normal	1	9,913
		Exceptions	934	34
	S2 – Canceled	Normal	1	333
		Exceptions	52	7
P4	S1 – Rejected	Normal	1	5,719
		Exceptions	33	70
	S2 – Accepted	Normal	1	1,535
		Exceptions	30	24
	S3 – Canceled	Normal	1	1,132
		Exceptions	20	84
P5	S1 – Handled by helpdesk	Normal	1	1,749
		Exceptions	29	5
	S2 – Handled by others	Normal	1	524
		Exceptions	2,247	2

Figure 2.
Frequency with which
the 15 most popular
paths are taken

normal path through Process 2 is not in the process model. For these reasons, we used the concept of scenarios to identify normal flows, as explained in Section 2, defining a normal flow as the most frequently taken path to achieve an identifiable result.

For Processes 2 and 5, we also have the process models, such that it is possible to determine the difference between expected exceptions – exception paths that are described

by the process model – and unexpected exceptions – exception paths that are not described by the process model. Table III shows the differences between these two types of exceptions. The table illustrates that there are more unexpected paths than there are expected paths and that each unexpected path is taken by relatively few customer cases. This means that each expected exception occurs less frequently than the normal flow and that each unexpected exception occurs even less frequently.

Looking in more detail at the different types of exceptions, as they are defined in Section 2, Table IV shows the relative frequency with which the different types of exceptions occur in the paths through the process.

5. Analysis and results

In order to investigate the relation of different exception types and the throughput time, we conducted a series of statistical analyses over the customer cases of each process and corresponding scenarios. Such relationships can be analyzed using parametric statistical tests, such as the linear regression, where the exception types can be considered as predictors (independent variables) and the throughput time as the dependent variable. These tests, however, give valid results only if a certain set of assumptions holds (Witte and Witte, 2014), especially the normality assumption (Field, 2013).

The results of our initial analysis showed that there are clear deviations from normality in throughput time for all processes and scenarios (with a Shapiro–Wilk test of normality with $p < 0.01$). This remained as a limitation even after transforming (e.g. log-normalizing) the dependent variable. Therefore, instead of applying parametric statistical tests, we forewent the predictive power of such tests and applied their non-parametric counterparts (Field, 2013). In particular, we used the Kruskal–Wallis H test to evaluate the hypothesis that multiple independent groups come from different populations. In other words, we assessed per exception type if the throughput time for the customer cases with specific types of exceptions is different than the throughput time for the customer cases that belong to the normal flow of a scenario.

Process	Scenario	Type of path	No. of paths	Avg. no. of cases per path
P2	S1 – Handled	Normal	1	2,400
		Expected	15	102
		Unexpected	29	3
P5	S1 – Handled by helpdesk	Normal	1	1,749
		Unexpected	29	5
	S2 – Handled by others	Normal	1	524
		Expected	118	5
		Unexpected	2,129	2

Table III.
Expected and
unexpected exceptions
per process

Process	Early entry (%)	Late entry (%)	Early exit (%)	Late exit (%)	Repeat (%)	Step back (%)	Add (%)	Skip (%)
P1	0	0	25	8	8	4	35	33
P2	0	0	0	0	20	48	65	26
P3	1	0	8	22	73	48	49	29
P4	0	0	0	19	0	67	30	26
P5	2	96	3	1	11	63	48	1

Table IV.
Relative frequency of
types of exceptions

We included in our analyses only those customer cases that have at most two exceptions types occurring at the same time and have a group size larger than 25. In doing so, we aimed to preserve accuracy in the inferences from the Kruskal–Wallis *H* tests and balance the sizes of the groups that are compared.

Table V shows the results of the statistical analysis of the relation of exceptions in a process on the throughput time of that process. It also shows the descriptive statistics of the scenarios that were investigated: the number of process instances in the scenario, as well as the average throughput time of the scenario and the standard deviation, skewness, and kurtosis of the throughput time. Our analysis on the customer cases through these processes reveals that both expected and unexpected exceptions lead to a significantly longer throughput time ($p < 0.001$), thus proving *H1*. Our analysis also shows that processes with expected exceptions and unexpected exceptions have significantly different throughput times ($p < 0.01$ in three disjoint Kruskal–Wallis *H* tests), thus proving *H2*. The pairwise comparisons that were conducted separately for each process show that, when the exceptions are unexpected, the throughput times are significantly longer ($p < 0.002$ and $p < 0.012$, respectively, for processes P2 and P5 in scenario *S2*). Consequently, we can conclude that: paths through a process that involve an exception have a longer throughput time than paths through a process that do not involve an exception (*H1*); and that paths that involve an unexpected exception have a longer throughput time than paths that involve an expected exception (*H2*).

Table VI presents the results from the statistical analyses performed over each process scenario and exception type (the results for each separate analysis is with $p < 0.01$), again including descriptive statistics of each scenario. Our analyses indicate that the throughput time in all process scenarios is statistically different when the paths involve different types of exceptions.

The findings depicted in Table VI can be aggregated over each exception type:

- In all five scenarios, where “step–back” appears, the throughput time is significantly longer than the normal flow.
- Similarly, in all scenarios that “repeat” appears the throughput time is significantly longer.
- The analyses involving paths where “skip” takes place, the throughput times differ significantly than the normal flow, but in different directions. In four scenarios in which it appears as the only exception type, the throughput times are shorter; whereas in two scenarios in which it appears paired with “add” and “late–exit” exception types, the throughput times are longer. This can be attributed to the fact that the exception types that are paired with “skip” (i.e. “add,” “late–exit”) are expected to have process instances with longer throughput times (as discussed below). Hence, assuming that the

Table V.
The relation between
expected/unexpected
exceptions and the
throughput time

Process Scenario	P2 – Change control		P5 – Incident management	
	S1 – Handled		S1 – Handled by helpdesk	S2 – Handled by others
No. of instances analyzed	3,982		1,844	4,742
Avg. throughput time (days)	51.6		0.39	16.2
SD	54.7		5.2	31.9
Skewness	3.09		37.45	7.83
Kurtosis	13.35		1,537.83	77.17
Expected exceptions	↑			↑
Unexpected exceptions	↑		↑	↑
Notes: ↑: increases throughput time (arrow size indicates degree of influence); “empty”: not applicable				

Process	P1 – Appeals			P2 – Change control		P3 – Incident management		P4 – Loan application		P5 – Incident management	
Scenario	S1 Accepted	S2 Withdrawn	S3 Rejected	S4 Deferred	S6 Withdrawn and Rejected	S1 Handled	S1 Handled	S1 – Rejected	S2 – Accepted	S3 – Canceled	S1 Handled helpdesk S2 – Handled others
<i>Exception type</i>											
No. of instances analyzed	775	173	158	90	39	3,982	39,547	8,034	2,244	2,807	1,844
Avg. throughput time (days)	270.0	369.8	305.8	332.6	332.6	51.6	18.4	2.4	16.8	18.7	0.39
SD	223.2	199.7	217.1	165.7	207.8	54.7	18.8	6.5	9.5	14.7	5.2
Skewness	0.27	–0.37	0.08	–0.01	–0.01	3.09	6.61	3.85	1.86	0.58	37.45
Kurtosis	1.31	–1.12	–1.52	–0.66	–1.39	13.35	66.46	18.59	5.95	0.36	1537.83
Early–Entry											↔ (with Add and (with repeat)
Late–Entry											↑ (with Add)
Early–Exit	↓		↓				↓ (with Skip)				↑
Late–Exit			↓ (with Skip)				↓ (with Skip)	↑ (with Skip)			
Add		↓ (with Skip)	↔ (with Skip)	↑		↑	↑	↑ (with Late–Exit)		↑ (with Late–Entry)	↑
Skip		↓	↓ (with Late–Exit) ↔ (with Add)		↓	↑ (with Add)	↓			↓	
Repeat						↑ (with Add)	↑				↑
Step–back						↑ (with Add)	↑	↑	↑	↑	

Notes: ↑: longer throughput time; ↓: shorter throughput time; ↔: no significant difference in throughput time; “empty”: not applicable. All with adjusted $p < 0.01$

Table VI. Different types of exceptions and the throughput times

exception type influences the throughput time, we may consider that the negative influence of the “skip” in these scenarios was weaker when compared to the influence of “add” and “late-exit.”

- “Add” appeared in majority of the scenarios that we analyzed (10 out of 13) – either separately or paired with other exception types. In eight of these scenarios, the throughput time is longer. Only in two scenarios, where the throughput time was shorter or remained the same, it appears with “skip,” which is associated with shorter throughput times (as discussed above). Therefore, we may conclude that paths with “add” exception types have significantly longer throughput time.
- “Late-exit” occurred in three scenarios – always paired with “skip,” the throughput times against the normal flow varied in different directions. Therefore, we do not have sufficient evidence to infer a valid conclusion on the relation between “late-exit” and the throughput time.
- In all scenarios where “early-exit” appears, the throughput time is significantly shorter than the normal flow.
- “Late-entry” appears in two scenarios (in one, paired with “add”). In both scenarios, the throughput time is significantly longer.
- “Early-entry” appears (in sufficient numbers) only in the second scenario of Process 5, and paired with “add” and “repeat.” No significant relation is found between the presence of these pairs and the throughput time. However, as discussed above, our analysis on “add” and “repeat” shows that they are associated with increased throughput time. Therefore, we could argue that the insignificance of these pairs can be attributed to the significant negative influence of the “early-entry” exception, which balances the positive influence of the “add” and “repeat” exceptions and leads to an inconclusive result of the pairs’ aggregated result. Nonetheless, this would still lead to a conclusion that is based on a single (qualitative) observation.

Based on the discussions above, we infer that the exception types “step-back,” “repeat,” “add,” and “late-entry” (i.e. exceptions that add activities to the normal flow) are associated with a significantly longer throughput time, while “skip” and “early-exit” (i.e. exceptions that remove activities from the normal flow) are associated with a significantly shorter throughput time, thus proving *H3* and *H4* to a large extent. For the “early-entry” and “late-exit” exception types, the result is inconclusive and further analysis is required with additional process data.

The conclusions are summarized in Table VII, which shows the conclusions and their support. The support is specified in terms of the number of cases (out of 5) and the number of scenarios (out of 13) that support each conclusion. If there are any cases or scenarios that contradict the conclusion, this is also specified. None of the conclusions is supported by all cases or scenarios, although they all have substantial support with only a few contradicting cases and scenarios for one of the conclusion. However, note that the support is qualitative in nature. To examine the statistical significance of the claims, a much larger number of cases or scenarios must be investigated.

6. Conclusion

This paper has shown the relation between the occurrence of exceptions in a business process and the performance of that business process. It shows that the paths that host exceptions to the normal flow of a business process have higher throughput time in the overall. It also shows that unexpected exceptions (i.e. exceptions that are not taken into account in a pre-defined process model) are associated with a longer throughput time than expected exceptions (i.e. exceptions that are taken into account in a pre-defined process model).

Conclusion	Support (no. of cases)	Support (no. of scenarios)	Contradict (no. of cases)	Contradict (no. of scenarios)
A step-back exception is related to a throughput time that is longer than the normal flow	3	5		
A repeat exception is related to a throughput time that is longer than the normal flow	3	4		
A skip exception is related to a throughput time that is shorter than the normal flow, but only when it does not co-occur with an add or late-exit exception	3	5		
An add exception is related to a throughput time that is longer than the normal flow	5	8	1	2
An early-exit exception is related to a throughput time that is shorter than the normal flow	2	3		
A late-entry exception is related to a throughput time that is longer than the normal flow	1	2		

Table VII.
Relation of exceptions
to throughput time

Finally, it investigates the relation between different types of exceptions and the throughput time, and shows that: paths that have exceptions that involve repeating one or more activities in a business process have a longer throughput time. In addition, paths with exceptions that involve performing additional activities at the start or during a process have longer throughput time; and paths with exceptions that involve skipping activities at the end or during a process have a shorter throughput time.

These conclusions lead to practical implications on policies that can be followed with respect to exceptions. First, since expected exceptions are associated with shorter throughput time than unexpected exceptions, it is advisable to investigate in advance which exceptions may occur during a process. These exceptions should then also be considered when developing operational support, such as forms, work instructions, and information systems. However, there is a trade-off to be made here, because processes typically have too many paths to all be considered up front. Nonetheless, some of the processes that we studied contained unexpected exceptions within the ten most frequently occurring paths, so improvement seems possible. Second, not all exceptions can be related to a longer throughput time. Anecdotal evidence suggests that employees sometimes take shortcuts in order to improve on process performance. For example, another study (Martens, 2009) showed that employees sometimes skipped a mandatory "resource reservation" step and instead agreed with their colleagues that they could use the resource in order to meet a deadline and keep customers happy. However, we also saw situations in which employees deliberately skipped steps with a detrimental effect. Therefore, employees should be enabled and even encouraged to make exceptions to the normal flow if necessary. Exceptions that are made by employees should be periodically reviewed both to detect good ideas, which can be incorporated into the defined process, and to detect bad ideas, which should be prevented in the future. The reason for making an exception can also be reviewed and possibly multiple reasons can be solved by making the same exception instead of different exceptions. The heavy-tail distribution of the different paths that are taken through the processes shows that many exceptions are unique for a single customer. By applying the same exception for multiple customers, the heavy tail can possibly be reduced and the process can be standardized further, thus leading to fewer unexpected exceptions and, consequently, shorter throughput times.

Since our analysis clearly shows that different types of exceptions have a different effect on throughput times, it is tempting to conclude that exceptions that lead to a longer throughput time must be avoided and exceptions that lead to a shorter throughput time must be implemented. However, against this background, it is important to note that in this study correlation certainly does not equal causation. For example, in one process we found that the additional activity “send reminder” was related to a longer throughput time (Jager, 2015). However, it was of course not the “send reminder” activity that caused the longer throughput time. It was rather the fact that the customer was unresponsive that caused both the longer throughput time and the reminder to be sent. This has the important practical implication that no general rules can be created for how to handle exceptions, but that each exception has to be studied by itself to determine how to mitigate its effects.

The research is performed by studying the customer cases of five processes from five different companies. First, it describes a statistical analysis of the relations between exceptions and performance per process, as there are enough customer cases per process to conduct a statistical analysis. Subsequently, it qualitatively aggregates over the results of the individual processes to derive general conclusions. The results of this analysis confirm the expectations regarding the relation between control-flow exceptions and throughput time. To the best of our knowledge, this work is the first to empirically test these expectations and provide quantitative evidence for them.

From a methodological perspective, the two most important limitations are the following. First, this paper defines an exception as any path through a process that is not the normal flow of a scenario. A scenario is a particular outcome of a process, such as “accepted,” “rejected,” or “sent on to other department.” The normal flows through these scenarios can safely be defined as non-exceptional. Also, in all processes that are studied, there are a large number of paths that can safely be defined as exceptional, because they are taken by less than 1 percent of the customer cases. However, there also is a gray area of customer cases that could both be classified as “normal” and as “exceptional.” We chose a rigorous definition of “exception,” but asking domain experts to identify paths as either “normal” or “exceptional” would yield a set of exceptions that is closer to a practical interpretation of exception. In future work, we aim to ask practitioners to identify their exceptions and investigate whether that leads to significantly different results. Second, due to limitations of the available data, this paper only studies the performance of processes in terms of throughput time with the presumption that favors a shorter time. In future work, we also aim to study the effect of exceptions on other performance indicators to provide a better representation of operational performance. Of particular interest are customer satisfaction and cost. We conducted our analyses over 70,000 cases originating from five industry processes. However, future work should consider analyzing instances of additional processes to strengthen the generalizability of the results.

Note

1. <http://fluxicon.com/disco/>

References

- Adams, M.J., ter Hofstede, A.H., Edmond, D. and van der Aalst, W.M.P. (2006), “Worklets: a service-oriented implementation of dynamic flexibility in workflows”, in Debruyne, C., Panetto, H., Meersman, R., Dillon, T., Kühn, E., O’Sullivan, D. and Ardagna, C.A. (Eds), *Proceedings of CoopIS*, Springer, Berlin and Heidelberg, pp. 291-308.
- Curbera, F., Khalaf, R., Leymann, F. and Weerawarana, S. (2003), “Exception handling in the BPEL4WS language”, in van der Aalst, W.M.P., ter Hofstede, A.H.M. and Weske, M. (Eds), *Proceedings of BPM*, Springer, Berlin and Heidelberg, pp. 276-290.

-
- Davenport, T. (1993), *Process Innovation: Reengineering Work through Information Technology*, Harvard Business School Press, Boston, MA.
- Eder, J. and Liebhart, W. (1998), "Contributions to exception handling in workflow management", *Proceedings of the EDBT Workshop on Workflow Management*, pp. 3-10.
- Field, A. (2013), *Discovering Statistics using IBM SPSS Statistics*, Sage Publications, Washington, DC.
- Gottschalk, F., Wagemakers, T.A., Jansen-Vullers, M.H., van der Aalst, W.M.P. and La Rosa, M. (2009), "Configurable process models: experiences from a municipality case study", in van Eck, P., Gordijn, J. and Wieringa, R.J. (Eds), *Proceedings of CAiSE*, Springer, Berlin and Heidelberg, pp. 486-500.
- Grigori, D., Casati, F., Dayal, U. and Shan, M.-C. (2001), "Improving business process quality through exception understanding, prediction, and prevention", in Apers, P.M.G., Atzeni, P., Ceri, S., Paraboschi, S., Ramamohanarao, K. and Snodgrass, R.T. (Eds), *Proceedings of VLDB*, Morgan Kaufmann, San-Francisco, CA, pp. 159-168.
- Grigori, D., Casati, F., Castellanos, M., Dayal, U., Sayal, M. and Shan, M.-C. (2004), "Business process intelligence", *Computers in Industry*, Vol. 53 No. 3, pp. 321-343.
- Hagen, C. and Alonso, G. (2000), "Exception handling in workflow management systems", *IEEE Transactions on Software Engineering*, Vol. 26 No. 10, pp. 943-958.
- Hallerbach, A., Bauer, T. and Reichert, M. (2010), "Capturing variability in business process models: the Provop approach", *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 22 Nos 6/7, pp. 519-546.
- Hwang, S.Y., Ho, S.F. and Tang, J. (1999), "Mining exception instances to facilitate workflow exception handling", *Proceedings of DASFAA*, IEEE Computer Society, Los Alamitos, CA, pp. 45-52.
- Jager, L. de (2015), "Consequences of exceptions in business processes", master's thesis, Eindhoven University of Technology, Eindhoven.
- Martens, A.C.N. (2009), "Relevance of conformance analysis information", master's thesis, Eindhoven University of Technology, Eindhoven.
- Meerkamm, S. (2012), "Staged configuration of multi-perspectives variants based on a generic data model", in La Rosa, M. and Soffer, P. (Eds), *Proceedings of the BPM Workshops*, Springer, Berlin and Heidelberg, pp. 326-337.
- Object Management Group (2011), "Business process model and notation (BPMN) version 2.0", Technical Report Formal/2011-01-03, Object Management Group, Needham.
- Reichert, M. and Dadam, P. (1998), "ADEPTflex – supporting dynamic changes of workflows without losing control", *Journal of Intelligent Information Systems*, Vol. 10 No. 2, pp. 93-129.
- Reichert, M. and Weber, B. (2012a), "AristaFlow BPM suite", *Enabling Flexibility in Process-Aware Information Systems*, Springer, Berlin and Heidelberg, pp. 441-464.
- Reichert, M. and Weber, B. (2012b), *Enabling Flexibility in Process-Aware Information Systems: Challenges, Methods, Technologies*, Springer, Berlin and Heidelberg.
- Rinderle, S. and Reichert, M. (2006), "Data-driven process control and exception handling in process management systems", in Dubois, E. and Pohl, K. (Eds), *Proceedings of CAiSE*, Springer, Berlin and Heidelberg, pp. 273-287.
- Rozinat, A. and van der Aalst, W.M.P. (2008), "Conformance checking of processes based on monitoring real behavior", *Information Systems*, Vol. 33 No. 1, pp. 64-95.
- Russell, N., van der Aalst, W.M.P. and ter Hofstede, A. (2006), "Workflow exception patterns", in Dubois, E. and Pohl, K. (Eds), *Proceedings of CAiSE*, Springer, Berlin and Heidelberg, pp. 288-302.
- Schonenberg, H., Mans, R., Russell, N., Mulyar, N. and van der Aalst, W.M.P. (2008), "Process flexibility: a survey of contemporary approaches", in Dietz, J.L.G. and Albani, A. (Eds), *Advances in Enterprise Engineering I*, Springer, Berlin and Heidelberg, pp. 16-30.
- Steeman, W. (2013), "Log of Volvo IT incident management system", available at: 10.4121/uuid:500573e6-acc6-4b0c-9576-aa5468b10cee (accessed November 13, 2014).

- Strong, D.M. (1997), "IT process designs for improving information quality and reducing exception handling: a simulation experiment", *Information & Management*, Vol. 31 No. 5, pp. 251-263.
- Tibshirani, R., Walther, G. and Hastie, T. (2001), "Estimating the number of clusters in a data set via the gap statistic", *Journal of the Royal Statistical Society: Series B*, Vol. 63 No. 2, pp. 411-423.
- van der Aalst, W.M.P. (2009), "Process-aware information systems: lessons to be learned from process mining", *Transactions on Petri Nets and Other Models of Concurrency II*, Springer, Berlin and Heidelberg, pp. 1-26.
- van Dongen, B.F. (2012), "Event log of a loan application process", available at: 10.4121/uuid:3926db30-f712-4394-aebc-75976070e91f (accessed November 13, 2014).
- Weber, B., Reichert, M. and Rinderle-Ma, S. (2008), "Change patterns and change support features – enhancing flexibility in process-aware information systems", *Data & Knowledge Engineering*, Vol. 66 No. 3, pp. 438-466.
- Witte, R.S. and Witte, J.S. (2014), *Statistics*, 10th ed., John Wiley & Sons, Hoboken, NJ.
- Zeng, L., Lei, H., Jeng, J.J., Chung, J.Y. and Benatallah, B. (2005), "Policy-driven exception-management for composite web services", *Proceeding of CEC on E-commerce Technology*, IEEE Computer Society, Los Alamitos, CA, pp. 355-363.
- zur Muehlen, M. and Recker, J. (2008), "How much language is enough? Theoretical and practical use of the business process modeling notation", in Bellahsène, Z. and Léonard, M. (Eds), *Proceedings of CAiSE*, Springer, Berlin and Heidelberg, pp. 465-479.

Further reading

- Weber, B., Reichert, M., Mendling, J. and Reijers, H.A. (2011), "Refactoring large process model repositories", *Computers in Industry*, Vol. 62 No. 5, pp. 467-486.

Corresponding author

Remco Dijkman can be contacted at: r.m.dijkman@tue.nl