UNIVERSIDADE ABERTA

INSTITUTO SUPERIOR TÉCNICO

Monitoring Applications with Process Mining

João Bernardo Salvador de Miranda

Master's dissertation in Information and Enterprise Systems

Dissertation supervised by Professor Miguel Mira da Silva
and Professor Rita Marques

November 2023

CONDITIONS OF USE OF THE WORK BY THIRD PARTIES

ACKNOWLEDGEMENTS

I like to start by expressing my gratitude to my supervisors, Professors Miguel, and Rita, for every hour generously invested in my research, and for their continuous support in this extensive and challenging research.

Additionally, my sincere appreciation to every faculty member, and colleague, who directly or indirectly was involved, all indispensable to achieving this personal academic objective.

I must also mention the essential role of the people involved in getting access to the necessary data, to demonstrate the applicability of the proposed solution.

Finally, to my friends, and specially my close family, your unconditional support was much appreciated, and I am forever grateful.

Para o Duarte

INTEGRITY DECLARATION

I hereby declare that I have conducted my dissertation with integrity. I confirm that I have not used plagiarism or any form of falsification of results in the process of the thesis elaboration.

I further declare that I have fully acknowledged the Disciplinary Regulations of the Universidade Aberta (regulation published in the official journal Diário da República, 2a série, No. 215, 6nov 2013).

Universidade Aberta, 24th of November 2023
João Bernardo Salvador Miranda

RESUMO

Um Método para Aplicar Mineração de Processos (Process Mining) para Suporte e Monitorização de Aplicações de Sistemas de Informação

Esta pesquisa apresenta um novo método para alavancar técnicas de *Process Mining* com o objetivo de monitorizar e suportar aplicações de Sistemas de Informação. Combinando as metodologias de Revisão Sistemática da Literatura e de Pesquisa em Design Science para abordar os objetivos da investigação. A revisão da literatura foi conduzida para explorar a investigação existente de *Process Mining* para monitorizar e suportar aplicações. A revisão de literatura seguiu rigorosos critérios de inclusão e exclusão, selecionando um conjunto de estudos que serviram como base de conhecimento, respondendo às questões de investigação colocadas. Foi descrito o potencial dos atuais métodos, as suas aplicações, limitações e dimensões inexploradas. Com base nesta revisão, a metodologia Design Science Research foi utilizada para desenvolver um novo método que descreve uma abordagem estruturada e sistemática para a aplicação de técnicas de *Process Mining*, especificamente adaptadas para monitorizar e suportar aplicações complexas de Sistemas de Informação. Enfatizando a utilidade prática, o método descreve etapas detalhadas, componentes e diretrizes para uma implementação eficaz. Posteriormente, foi avaliado e validado através de um cenário real de caso de uso, afirmando sua eficácia e potencial impacto em aplicações reais. O processo de avaliação concentrou-se na capacidade de o método identificar ineficiências de processos e fornecer suporte para a tomada de decisões em aplicações de Sistemas de Informação. As conclusões derivadas deste estudo contribuem para a aplicabilidade de *Process Mining*, introduzindo um método que visa melhorar as capacidades de monitorização e suporte de aplicações de Sistemas de Informação. Esta investigação reforça a relevância prática e o potencial transformador da integração do *Process Mining* no domínio da gestão de Sistemas de Informação e estabelece as bases para futuras investigações neste campo.

***Palavras-Chave**: Mineração de Processos, Monitorização de Sistemas, Suporte Aplicacional, Mineração de Dados, Processos de Negócios*

ABSTRACT

This research presents a novel method aimed at leveraging Process Mining techniques for monitoring and supporting Information Systems applications. Combining a Systematic Literature Review and the Design Science Research Methodology to address the research objectives comprehensively. The Systematic Literature Review was conducted to explore the existing landscape of Process Mining for application's monitor and support. The review process followed rigorous inclusion and exclusion criteria, selecting a collection of pertinent studies that served as a foundational knowledge base. Through this review, key insights were reported regarding the current methodologies, their applications, limitations, and the unexplored dimensions within the field. From these insights, the Design Science Research Methodology was then employed to conceptualize and develop a new method. This method outlines a structured and systematic approach to applying Process Mining techniques specifically tailored for monitoring and supporting complex Information Systems applications. Emphasizing practical utility, the method encompasses detailed steps, components, and guidelines for effective implementation. Subsequently, the proposed method was evaluated and validated through a real-world use-case scenario, affirming its efficacy and potential impact in actual application environments. The evaluation process focused on assessing the method's ability to derive actionable insights, identify process inefficiencies, and provide support for decision-making within Information Systems applications. The findings derived from this study contribute to the field of Process Mining by introducing a tailored methodology aimed at enhancing the monitoring and support capabilities of Information Systems applications. This research reinforces the practical relevance and potential transformative impact of integrating Process Mining into the domain of Information Systems management and lays the groundwork for future advancements in this field.

Keywords: *Process Mining, System Monitoring, Application Support, Data Mining, Business Process Model, Design Science Research, Systematic Literature Review*

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF ABBREVIATIONS AND ACRONYMS

BPM ..... Business Process Management

BPMN..... Business Process Management Notation

CRM.....*Customer Relationship Management*

CSV.....Comma-Separated Values

DFG .....*Directly-Follows Graph*

DSR .....*Design Science Research*

ERP..... *Enterprise Resource Planning*

ID .....*Identification*

ILP..... *Integer Linear Programming*

PDM .....*Product Data Management*

PM..... *Process Mining*

SCM .....*Supply Chain Management*

SQL.....*Structured Query Language*

XES.....*Extensible Events Stream*

XML..... *Extensive Markup Language*

1 INTRODUCTION

The increasingly complex and interconnected application systems demand the development of advanced monitoring mechanisms. In an ever-changing technical context, the capability to simultaneously analyse the performance and identify possible applicational issues is indispensable to guarantee the best possible performance, high reliability, and, ultimately, user satisfaction.

Traditional monitoring tools provide valuable information however, they commonly, heavily rely on technical capabilities, providing extremely technical or purely analytical reports. Additionally, those tools often struggle to provide the depth of insight required to address the specific requirements for the efficient management of complex applications.

Considering this optimization opportunity, several alternatives were considered. The innovative and disruptive nature of Process Mining [1] and its potential in Information and Enterprise Systems management, was the principal reason to select this technology as an alternative to the current methods.

Over the past two decades, the discipline's scope expanded and currently, more than 40 commercial Process Mining products are available, in addition to the open-source process mining tools. The adoption of these technical capabilities has been accelerating recently, the process mining market is expected to double every 18 months in the coming years [2].

In this context, Process Mining was considered a promising prospect, by leveraging event data captured within systems, to discover patterns, bottlenecks, and inefficiencies, offering a comprehensive understanding of the underlying processes [1] [2]. The Systematic Literature Review and the Design Science Research methodologies will support this study which intends to propose an innovative method to properly apply the available technical capabilities to large and complex applications. The proposed method's efficacy and validity will be demonstrated and evaluated using a use case from a real application.

The primary objective of this proposal was to construct a systematic and adaptable method to utilize Process Mining to support and monitor Information Systems' applications. The secondary purpose was to evaluate the efficacy, utility, and applicability of the method in a real and complex use-case scenario. By accomplishing these goals, this research intends to contribute to the management and information system's body of knowledge, while also contributing to the Process Mining field, particularly regarding monitoring and supporting strategies for complex applications.

The selected scientific research methodologies offer a structured approach to apply Process Mining as a monitoring instrument for application systems. To support the study, the following research questions were proposed:

- RQ1: How is Process Mining applied as an applicational support tool?
- RQ2: What are the algorithms employed?
- RQ3: What are the Process Mining methods used to effectively identify constraints and errors in applications?
- RQ4: What are the possible limitations of Process Mining as an application support tool?

This research is structured as follows:

- Chapter 2: provides an in-depth review of the relevant literature, offering a comprehensive understanding of the current Process Mining body of knowledge.
- Chapter 3: introduces the methodologies: Design Science Research and the Systematic Literature Review.
- Chapter 4: details the literature review process, planning the research questions, the search and selection process, and answering the proposed research questions.
- Chapter 5: identifies the motivation and research problem, based on the literature review.

- Chapter 6: describes the respective objectives to the identified research problem.
- Chapter 7: delineates the specifics of the developed method design, outlining its components and detailing the steps involved in its application.
- Chapter 8: presents a real-world use-case scenario, to demonstrate the proposed method’s effectiveness in a practical context.
- Chapter 9: evaluates the method’s applicability in the selected use case.
- Chapter 10: summarizes key findings, discusses the identified limitations, and suggests prospects for future research.

2 RESEARCH BACKGROUND

To expand the current practical application of the existing methods, this research is founded on the academically established principles, previously researched and studied.

To determine these theoretical foundations, this chapter introduces the utilized standards, guidelines, parameters, and constraints, divided into three sections, Process Mining, Process Models and Events Logs. Introducing, respectively, the principal concepts of the field, the practical applicability, and the technical characteristics.

2.1 Process Mining

Process Mining is an emerging scientific discipline, developed to establish the bridge between data science and process science. Pioneered by Wil van der Aalst, at the Eindhoven University of Technology, it has progressed since the late 1990s, emphasised by several published studies. The field development is evidenced by the current range of available commercial software, that supports and facilitates the successful practical application of Process Mining (PM) in several organizations. PM is a relatively young research discipline, positioned between data mining and process modelling [1] [2].

The principal purpose [1] of Process Mining is to discover, monitor and improve real processes by extracting knowledge from event logs. PM is commonly utilized by organizations to discover and discern their processes, provide valuable insights, and possibly diagnose problems regarding productivity, effectiveness, compliance, regulatory, and legal issues. Additionally, and when applicable, it has the capability to automatically trigger the required corrective measures or actions [2].

This introductory chapter will briefly describe the fundamental subjects, intrinsic to the Process Mining field, to establish the followed principles and the foundation of our research.

There are three structural types of Process Mining: discovery, conformance, and enhancement. Discovery is the capacity to extrapolate a process model from the available event logs. Conformance compares an existing process model with an event log of the same process, matching and evaluating the observed events with the process model obtained, identifying possible deviations. Enhancement, also referred to as extension, has the objective to extend, improve or repair the process model. Figure 1 displays the established relations between the actual processes, the data generated and the process model, highlighting the alignment and dependency between the digital and physical universes [1].

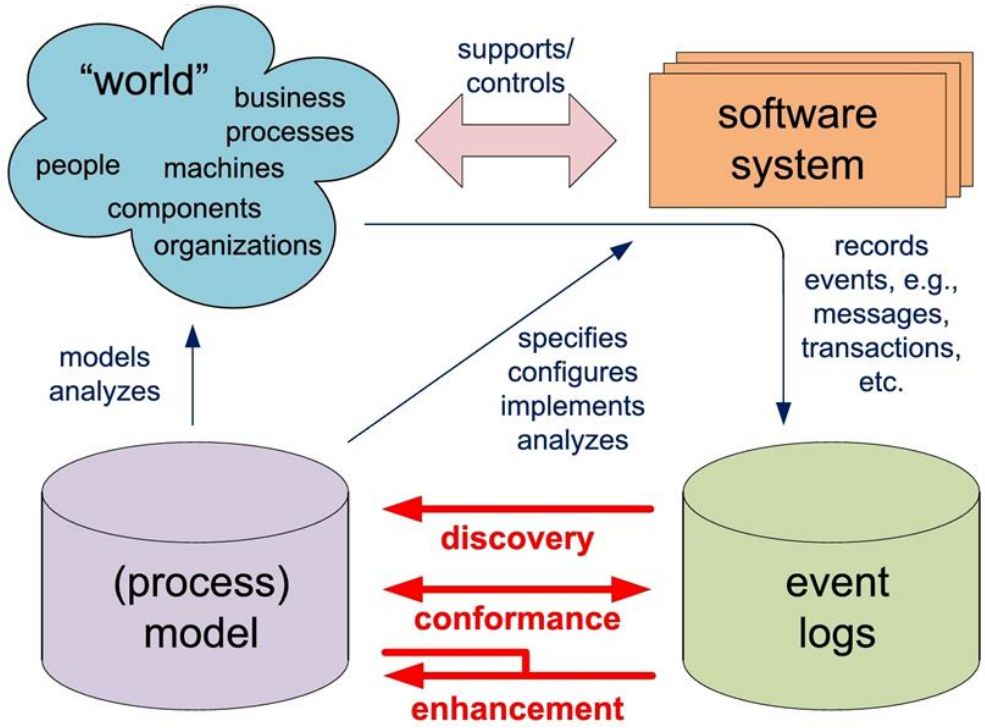

Figure 1 - Positioning of the three main types of Process Mining: discovery, conformance, and enhancement (source [1])

2.2 Process Models

Process models are an intrinsic part of Process Mining, and can be employed for the following purposes:

1. Insight: presenting the data from different perspectives, providing new valuable information;
2. Discussion: supporting and structuring key discussions;
3. Documentation: for instructing or certification purposes;
4. Verification: process models can be analysed to identify errors in systems, applications or procedures;
5. Performance analysis: identifying the critical activities, their relations, their response times, service levels, and other meaningful data;
6. Specification: models can be used to detail and specify the requirements;
7. Configuration: models can be used as a guide to configure a system.

Commonly, there are four widely accepted perspectives: (1) the control-flow perspective, which focuses on the ordering and relations between the activities, to identify a suitable characterization of the possible paths; (2) the organizational perspective emphasises resources, their involvement, role and relation, with the objective to structure the organization or to establish the existing social network; (3) the case perspective is concerned with the cases' properties, a case can be characterized by its activities, order, resources, and can also be characterized by additional values, when available, and finally; (4) the time perspective, examines the timing and frequency of each event, to identify and discover possible bottlenecks, to measure service levels, to monitor the use of resources, and to predict current and future cases' processing time [1].

The evaluation of the discovered process model has four principal criteria: fitness, precision, generalization, and simpleness [2]. Fitness: referred also to as recall, evaluates whether the discovered model suits the event log behaviour. Precision: assesses if the discovered model establishes redundant or unconnected behaviour, absent from the event log. Generalization: prevents the inclusion of every observed behaviour, which will result in an overfitting model. Simplicity: Evaluate if the discovered model is as simple as possible. Relating to Occam's Razor principle, which states that "one should not increase, beyond what is necessary, the number of entities required to explain anything".

To further understand these metrics and the variables utilised, it is useful to comprehend the general formulas for calculating each criterion:

$$Fitness = \frac{Total\ number\ of\ Matching\ Events}{Total\ number\ of\ Observed\ Events} \quad \textit{Equation 1}$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad \textit{Equation 2}$$

$$Generalizations = \frac{Total\ number\ of\ Matching\ Transitions}{Total\ number\ of\ Possible\ Transitions} \quad \textit{Equation 3}$$

$$Simpleness = \frac{1}{Number\ of\ Elements\ in\ the\ Model} \quad \textit{Equation 4}$$

The concepts of precision and generalization are challenging. These notions are quantifiable, however, what they measure is not consensual. The researcher must balance these, to avoid an underfitting or an overfitting model [2].

In Process Mining, the available visualizations provide valuable insights. The graphic representation of process models facilitates the analysis of complex workflows, illustrating the flow of activities, transitions, and dependencies, highlighting behavioural patterns and bottlenecks. Additionally, they provide performance metrics, with an interactive exploration. There are identified limitations, the complexity of some of the interactive features requires a learning curve, and due to the available visualizations, the interpretation may be subjective.

The available notations, or modelling techniques, widely used to visualize a process model are: Business Process Management Notation (BPMN), Flowcharts, Directly-Follows Graph (DFG), Transition Systems, and Petri Nets. BPMN is the standard for modelling business processes, and facilitating workflow creation and visualization. A Flowchart represents the sequence of actions in a process. The DFG, unlike the BPMN model, displays the cycle time and highlights the more frequent paths and activities, valuable for identifying frequency variations. A Transition System is

used to illustrate the potential behaviour of discrete systems, consisting of states and transitions between the respective states. Finally, a Petri Net is a graphic representation of the system's flow, consisting of places, transitions and connecting arcs.

Petri nets are commonly used to analyse complex processes, however, there are identified limitations that can comprise their application in the following situations:

1. Large and intricate processes;
2. Containing non-linear or Stochastic Behaviour;
3. Representing Concurrent Activities;
4. Continuous Behaviour;
5. Dynamic Processes.

For the mentioned reasons, in highly dynamic processes, with complex concurrency, BPMN or other simulation techniques, are more appropriate and effective. BPM (Business Process Management) and PM are clearly related. BPM established a set of principles, methods, and tools, combining information technology, management sciences and industrial engineering to improve business processes [1].

The quality of the discovered model is directly related to the selected visualization. Process models commonly exhibit infinite behaviour, through loops, a behaviour that the model should not contain. The intrinsic characteristics of the process in study will be the decisive factor in selecting the modelling technique.

2.3 Event Logs

Event logs are essential for Process Mining. Each technique can have specific requirements, and or limitations, regarding the necessary data. It is crucial to understand these fundamental prerequisites to adapt the data extracted.

In some cases, the data may be gathered from different data sources, like databases, flat files, message logs, application logs, transaction logs, Enterprise Resource Planning ERP, Customer Relationship Management (CRM), Supply Chain

Management (SCM), Product Data Management (PDM), work-flow management, project management software or document management systems.

In principle, it is possible to apply PM techniques to any event log, providing that each case has a timestamp, a unique case Identification (ID) and the activities, or transactions, are properly identified [1].

Additionally, syntax and semantics are fundamental when extracting and merging the data: both need to be considered. Depending on the research questions or business objectives, different visualizations of the available data could and may be necessary. To provide meaningful information using Process Mining, it is imperative to understand the importance of this initial process, and the implications of potential data quality problems [1].

Four primary data quality dimensions are considered for event logs [2]: (1) missing data; (2) incorrect data; (3) imprecise data, and; (4) irrelevant data, commonly referred to as noise. Applying PM techniques with incorrect data and imprecise data, specifically for key event attributes, like activity labels and timestamps, will impact the quality of the obtained results. Missing data, depending on this statical relevance may also impact the outcome of the analyses. Irrelevant data will needlessly waste resources, additionally, if unnecessary or redundant activities are considered, the obtained process model will be needlessly complicated.

It is possible to identify parallel data quality challenges in event logs and data mining. One significant distinctive aspect present in PM is the necessity to have detailed correlated event data to attain and capture the real process behaviour [2].

The exercise of preparing event logs includes three key techniques: extraction, abstraction, and correlation. Figure 2 demonstrates their association and the fundamental PM concepts. Understanding the data structure benefits the process discovery efficiency.

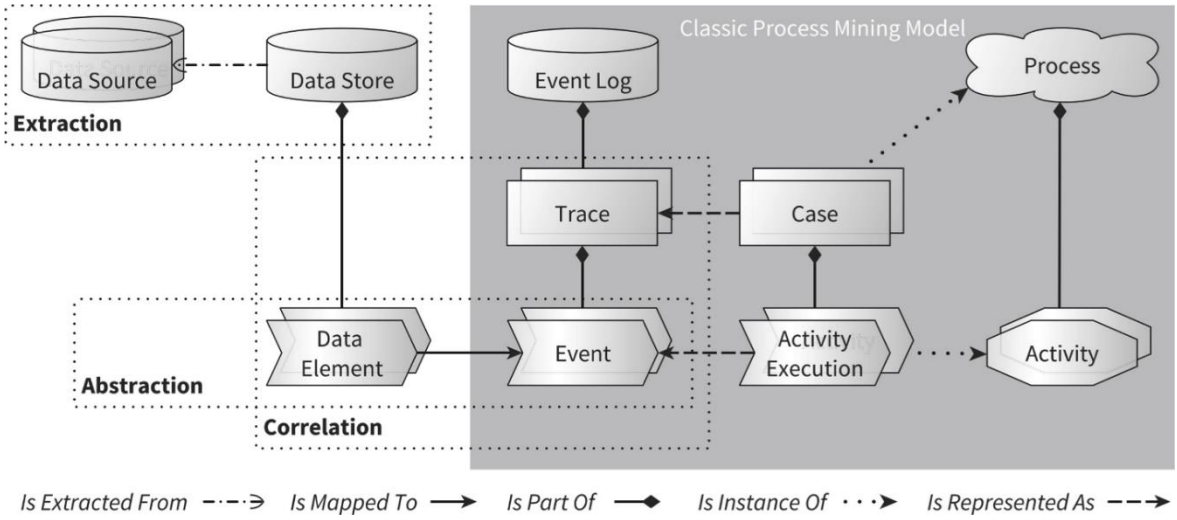

Figure 2 - Event log preparation techniques (extraction, correlation, and abstraction) and their relationship to key Process Mining concepts (source [2]).

The log file more widely utilized is the XES (Extensible Events Stream) standard, providing a file structure in a commonly recognized XML (Extensive Markup Language) format.

3 RESEARCH METHODOLOGY

This chapter provides a clear and concise overview of the research methodologies selected. The following sections will describe each methodology, its principal procedures, techniques, and methods.

3.1 Design Science Research

Following the study and analyses of several possible research methods, and considering the practical characteristic of the selected problem, we have considered that the Design Science Research (DSR) methodology was appropriate and would produce the necessary and applicable results. As defined [3], the science of design and its theories should be general and applicable in several domains, regardless of their specificities.

Design Science Research is a systemic, but flexible, methodology that uses an iterative process with several analytical techniques and perspectives with the purpose of designing innovative or new solutions for the identified problems. DSR has been successfully used in Information Systems research due to its wide applicability, Considering the practical application of the solution, and the perceived need for a continuing review process, the DSR methodology process ensures the necessary agile and adaptive approach for the identified problem.

In DSR, the researcher answers relevant and practical questions by creating innovative artefacts, contributing to the scientific knowledge. To design the artefact effectively, it is valuable and fundamental to understand the identified problem as well as the available information on the subject. These artefacts can be constructs, models, methods, instantiations, or better design theories.

The DSR model (see Figure 3) adapted from the process iteration proposed by Peffers [4] [5] consists of six sequential activities: (1) identify the problem and the motivation; (2) the solution’s objectives; (3) the design and development; (4) the demonstration; (5) the evaluation; and the (6) communication. Additionally, there

are four described [3] entry points, directed or centred on: the (1) problem; (2) objective; (3) design and development; and (4) client and context.

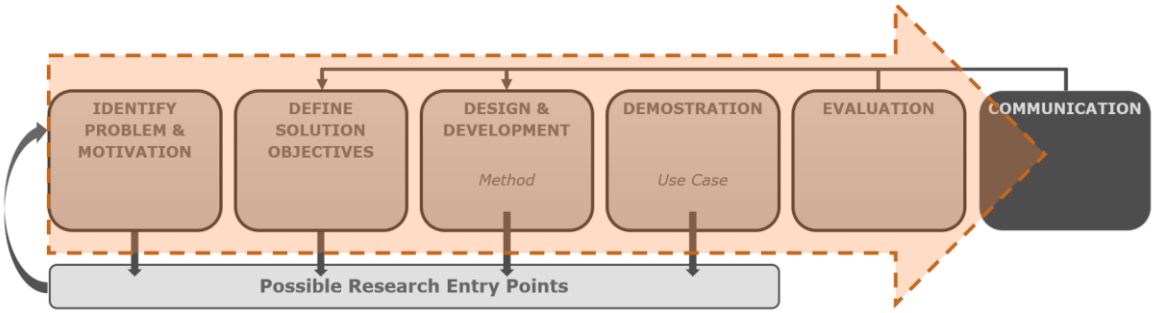

Figure 3 – Design Science Research Methodology adapted from the model proposed by Peffers [4]

The DSR's seven guidelines published by Hevner [6] describe a consistent research framework:

1. Design as an Artefact - A construct, a model, a method, or an instantiation;
2. Problem Relevance - Identify important and relevant business problems;
3. Design Evaluation - Must be demonstrated and evaluated rigorously;
4. Research Contributions - Must provide clear and verifiable contributions;
5. Research Rigor - Rigorous methods to construct and evaluate the artefact;
6. Design as a Search Process - Utilizing the available methods;
7. Communication of Research - Technologic and management oriented.

The documented [7] research activities rapidly iterate the construction of an artefact, its evaluation, and subsequent feedback to refine the design further. The evaluation of the DSR Methodology, proposed by Peffers [4] recognises three principal objectives. Primarily, must be consistent and represented in the literature. Secondly, must provide a nominal conducting process. Thirdly should provide a mental model to present and evaluate the research.

In design science research, evaluation is directed to the created artefacts, which can include design theory, produced artefacts [8] and information systems [9]. Their utility, quality, and efficacy must be demonstrated using rigorous evaluation methods [9], providing evidence of its applicability and whether the new technology

developed achieves the purpose and objective for which it was designed. The significance of conducting a rigorous evaluation is consensual [10].

Artefacts can be evaluated considering their functionality, completeness, consistency, accuracy, performance, reliability, usability, organization fitness and even, if applicable, the artefact's style. Checkland and Scholes proposed five essential evaluative properties ("the 5 E's"): efficiency, effectiveness, efficacy, ethicality, and elegance [9].

The goals proposed by Venable et al. [9] are (1) rigour, identifiable improvement in real situations, (2) efficiency, respecting the available resources, (3) ethics and respect.

Hevner et al. [6] summarized five evaluation classes: (1) observational methods: case studies and field studies, (2) analytical methods: static analysis, architecture analysis, optimizations, and dynamic analysis, (3) experimental methods, controlled experiments, and simulations, (4) testing methods: functional testing (black box) and structural testing (white box), and finally (5) descriptive methods: informed arguments and scenarios.

Prat et al. [11] proposed a holistic view, composed of three main elements, (1) a hierarchy of evaluation criteria, (2) a model providing a high-level abstraction of evaluation methods and (3) a set of generic evaluation methods.

Since the choice of evaluation methods is driven by the choice of artefact. Following the research by Peffers [10], a case study was primarily elected to evaluate the artefact, specifically in Information Systems journals. For this reason, the conducted case study was the primary evaluation method, performed by applying the method to a real-world situation.

Following the method and framework developed and proposed by Venable et al. [9], which was an extension of the framework proposed by Pries-Heje et al. [8], the described [9] four-step method for conducting evaluation research design was applied.

After analysing the context of the evaluation, while considering that the evaluand artefact is a method, despite having several algorithms incorporated, and in nature, this method is a social-technical process, since human interaction is necessary to provide utility.

The proprieties of the method subjected to evaluation are its usability, utility, and effectiveness. The evaluation must be rigorous, considering the proposed method, classifiable as naturalistic, the evaluation was done ex-post, after implementation, using a case study. Regarding both proposed algorithms, the evaluations will be done ex-ante and ex-post utilizing computer simulations. A hybrid methodology will be therefore employed. No comparable artefacts were identified for the proposed method. The main constraint will be time and computing power.

3.2 Systematic Literature Review

The Systematic Literature Review (SLR) is a preliminary study that applies a well-defined methodology (see Figure 4) to identify, analyse and understand the available published evidence concerning the specific Research Questions (RQ) [12] [13].

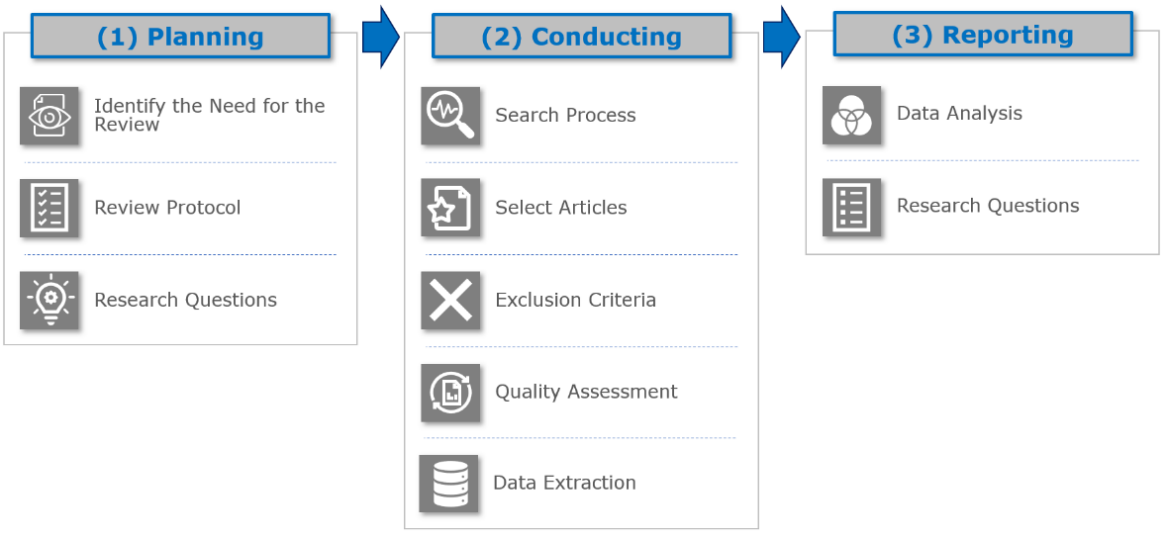

Figure 4 - Systematic Literature Review: Planning, Conducting, Reporting

By applying this methodology, it is possible to review and assess the existing evidences related to a specific subject, summarising the benefits and limitations of certain methods. Is possible to identify gaps in the existing research, and propose areas

or topics for further investigation. This methodology provides a framework to appropriately identify, and position new research activities.

The SLR incorporates the researched topic's current available contributions, providing several metrics and data analysis to support and guide future researches. To guarantee this review is done scientifically, the employed search strategy must ensure the maximum number of relevant articles are identified [12] [13]. The quality assessment process guarantees the relevance of the selected articles, evaluating their significance and precision to answer the proposed questions.

This snowballing method is then applied to the discovered articles until there are no new articles discovered. The snowballing backward method can additionally be employed, consisting of reviewing the articles referenced in every article, however by nature will return only older articles, whereas the forward method will return the more recent articles, in principle, more interesting for the literature review.

Following the article selection, and exclusion, dependent on the number of valid identified articles, it may be necessary to employ the mentioned snowballing forward method, that is to search which articles reference each selected article, using Google Scholar(1), 'cited by' option, after reviewing each new article, the same inclusion and exclusion criteria are applied.

(1) https://scholar.google.com/

4 SYSTEMATIC LITERATURE REVIEW

Following the systematic literature review method, the search for academic journals and peer-reviewed publications aims to answer the research questions.

As described in this chapter, in the planning stage, the research questions are elaborated, following the conducting phase, defining the search process, inclusion and exclusion criteria, quality assessment and search results, finally in the reporting stage the proposed research questions are answered.

4.1 Planning

To guide the research scientifically, elaborated the following research questions:

RQ1 *How is Process Mining applied as an applicational support tool?*
The question pretends to evaluate and assess the current applicability of PM and to identify possible solutions to apply the current techniques with more effective and promising results.

RQ2 *What are the algorithms employed?*
The question's purpose is to identify the available algorithms and their respective utility. The open-source environment intrinsic to the PM technical development offers several alternative algorithms, meaning it is relevant to identify their characteristics, properties, and possible limitations.

RQ3 *What are the Process Mining methods used to effectively identify constraints and errors in applications?*
The question intends to identify and evaluate the possible solutions for the application of these tools. PM is applied across distinct fields, with different applications, the research has focused on information systems and specifically on the identification of errors and problems in complex applications, using gathered event logs.

RQ4 *What are the possible limitations of Process Mining as an application support tool?*
The purpose of this question is to understand the effectiveness of the existing methods and identify possible solutions to mitigate or avoid negative outcomes. The

intention is to understand if and where compromises were made and to assess the solutions, or lack thereof, proposed for each limitation identified.

4.2 Conducting

In this section the search process is detailed, referencing the inclusion and exclusion criteria, the quality assessment process, and the obtained search results.

4.2.1 Search Process

The search was conducted in EBSCOhost Web(2), using full and truncated search terms, and utilizing comparable expressions to guarantee the results included every article related to the research questions.

The following search string was applied to the abstract of the articles:
(“process mining" or "process discovery" or "conformance checking") AND (application or software or system) AND (monitoring or monitor or assessment or errors or evaluate or control or issues or problems)

4.2.2 Inclusion Criteria

The selected criteria used for the inclusion of the discovered articles were the following:

1. Applied and described Process Mining applications;
2. Identified and explained research methods used;
3. Acknowledged conformance checking as a research subject;
4. Reported and experimented with different process discovery algorithms.

Criteria 1-2 were selected to assess the distinct applications of PM. Criterion 3 intends to analyse the potential of conformance checking applications. Criterion 4 to identify the distinct algorithms applied.

(2) https://search.ebscohost.com

4.2.3 Exclusion Criteria

The initially selected articles were excluded for the following criteria:

1. Not peer reviewed;
2. Not published in an academic journal;
3. Full article not available in PDF;
4. Duplicates;
5. Full text not written in English;
6. Process Mining not applied;
7. Out of scope or not applicable;
8. The evaluation methods or the conclusions lacked justification or evaluation.

Criteria 1–2 were selected to guarantee high-quality results. Criterion 3, to guarantee the articles were publicly available. Criterion 4 excluded duplicate articles, in most cases published in multiple sources. Criterion 5 excluded non-English articles. Criteria 6-7 intended to focus the research on the applicability of Process Mining specific to applications and excluded articles that used simplistic processes not considerably applicable to the current research. Criterion 8 intended to exclude articles that did not properly evaluate or explain their findings regarding their contributions to the research questions.

Considering the researched field has the initial contributions dating from 2005, and since there was not a clear reason to not include articles dated after any particular year, no articles were excluded based on this criterion.

4.2.4 Quality Assessment

The quality assessment method was applied to guarantee the significance of the selected articles and to assess the evidence of a substantial and reliable method for the selected review questions [12] [13]. For each article included in this review, and for each of the following five criteria, were conferred a score between 1–3 (where 3 is high, 2 is medium and 1 is low):

1. How is Process Mining utilized as a monitoring tool?
2. How is conformance checking utilized as a monitoring tool?
3. How appropriate are the methods and analysis conducted?
4. How applicable is the proposed solution in a large scale and in a real application?
5. How reliable is the evaluation method utilized?

After calculating the resulting scores for each selected article, being the minimum 5 and the maximum 15, it is possible to identify articles that clearly do not enhance or complement the research. Articles with a score lower than 8 were removed. No comparison with other studies was possible, due to the inability to identify comparable research.

4.2.5 Research Results

The search was conducted in February 2023, applied the identified search string resulted in a total of (n=595) articles (see Figure 5), after applying the selected limiters, peer reviewed, published in an academic journal and with full text available, corresponding to the criteria 1-3, (n=206) remained. Subsequently removing the duplicates (n=4), and articles not translated to English (n=4), criterion 4 and 5 respectively, remained 198 articles, of which read and studied the title and abstract.

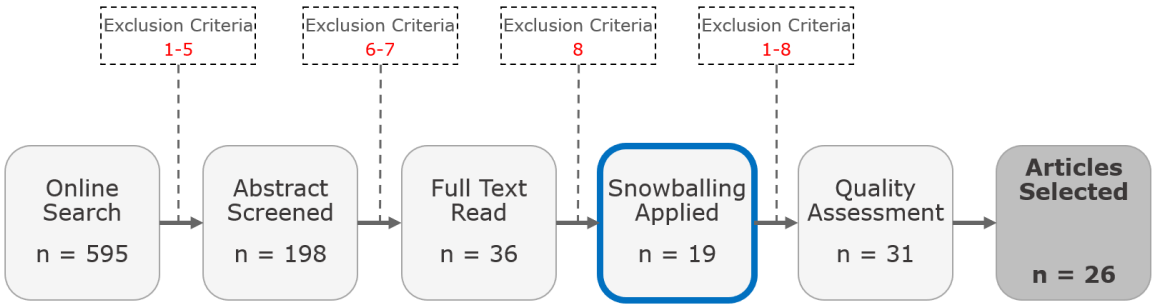

Figure 5 – Article Selection Process

From these articles, applying criterion 6, where PM was absent (n=11), were excluded, and articles that were considered out of scope or not applicable (n=151) were also removed, corresponding to criterion 7. Considering the percentage of articles excluded for this reason is relevant to mention that most articles were focused

on human based processes, where the optimization process is not transferable, therefore, articles that did not use application event logs were excluded, and when the number of instances or activities was too minimalistic or simple to be considered.

The full texts of (n=36) articles were read to identify possible cases where the methods or contributions were not evaluated or explained, or the selected evaluation was not considered applicable (n=17), criterion 8. In most cases, the article was not previously excluded since this evaluation was only possible after a complete reading, the absence of an evolution method in the abstract won't necessarily mandate that there was not being applied, however in several cases, that turns out to be the case. The resulting articles were (n=19), listed as P1 to P19 (see Table 7 in Appendix II – SLR Quality Assessment).

To increase this number, and to extend the scope of the research, the snowballing forward method was applied. From the (n=19) selected articles, using Google Scholar to search for the articles that referenced them, there were (n=3417) resulting articles, including duplicates (see Figure 6).

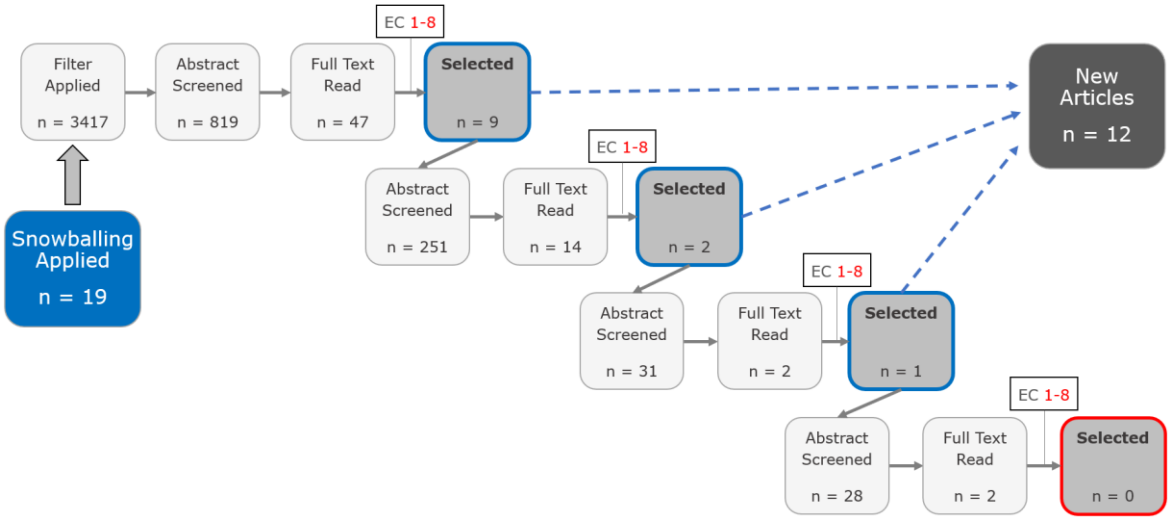

Figure 6 – Snowballing Process

Considering that the vast majority (n=2970) were obtained from a small (n=5) number of sources, a filter *[allintitle ("Process Mining" or reliability or monitor)],* was applied to the search of these articles, excluding a large (n=2605) percentage. After this filter was applied, the total (n=819) of the remaining articles' titles, and for pertinent cases their abstracts', were screened and evaluated. Resulting in further

(n=47) possible articles, these articles were evaluated using each criterion 1-8, resulting in this first snowballing stage, in new (n=9) selected articles. These, where referenced by (n=251), applying the same method as previously, from the preselected (n=14) articles, (n=2) were selected. From these, referenced by (n=31), only (n=2) were preselected, and (n=1) was selected. The last article, referenced by (n=28) did not result in any new article.

By applying the snowballing method to the initial identified (n=19), it was possible to identify new articles (n=12) that met the defined criteria requirements, listed as P20 to P31, totalling 31. When performing the quality assessment, and since one of the articles was excluded, P27 was the snowballing source of another article, P29, and following the best practices, the article subsequently discovered with this method was also removed. A total of 5 papers were removed after the quality assessment, resulting in a total of 26.

Further analysing the resulting studies, it is relevant to address the fact that a statistically considerable number (n=7) are written by, or with the collaboration of, Wil M.P. van der Aalst, evidencing his significant contribution to the Process Mining field. Table 1 demonstrates the publishers of the selected articles, evidencing the weight of Elsevier (n=13), and the relevance of Springer (n=4) and Willey affiliated (n=4) publishers.

Table 1. Publishers

Publisher	Number of Articles
Elsevier	13
Springer	4
John Wiley & Sons, Ltd	3
Taylor & Francis Group	2
Wiley Periodicals, Inc.	1
IEEE	1
JMLR	1
IOS Press	1

The analysis of the keywords used in each article was also done, demonstrated in Table 7 in Appendix I – SLR Search Results, identifying the most used keywords and illustrating the number of articles that focused their research in Process Mining (n=19), Business Process Management (n=10), Process Discovery (n=6). Petri Nets (n=6) and Conformance Checking (n=5).

After performing the quality assessment, the results are presented in Table 8 n Appendix II – SLR Quality Assessment, resulting in the exclusion of the articles with a score under 8. One key conclusion identified is the fact that only a few (n=4) articles score equal to or above 12.0.

4.3 Reporting

In this section, the researched data is analysed with the objective of answering each research question. The respective studied literature and individual contributions are identified.

4.3.1 RQ1 - How is Process Mining applied as an applicational support tool?

Table 2. Process Mining Applications

Applications	Description	Articles
Conformance Checking	Identify deviations between the event logs and the process model.	[14] [17] [22] [31] [32] [33] [34] [36]
Applicational and Operational Support	Identify errors or issues in applications.	[16] [17] [18] [23] [25] [28] [33]
Process Discovery	Discover processes from event logs.	[16] [17] [18] [25] [32] [34] [37]
Model Processes	Model and visualize processes from event logs.	[17] [18] [19] [22] [24]
Monitor and Control Processes	Automatized monitored and controlled processes.	[17] [20] [21] [22] [26]

Workflow Management	Manage workflow from registered event logs.	[15] [28] [29]
Process Management	Manage processes from registered event logs.	[30]

Most organizations already use a variety of process-aware information systems, like ERP, CRM, SCM, PDM and more. These systems generate event logs where Process Mining can be applied [14] [15]. Event data is everywhere, and PM techniques help to unlock the value of such data [16]. PM combines data-centric disciplines, with process modelling and analysis, with the purpose of discovering, monitoring, and improving these processes [17].

The focus of this research was the applicational support potential. PM is applied to discover software execution and model component behavioural [18] [19]. Can support industrial control systems [20] [21]. Applicable to audit information systems [22] to conduct software process appraisals [23], and to model unstructured processes [24]. Additionally, PM is utilized for reengineering applications and operational support [16] [25].

Regarding human-driven processes, PM is applicable as a monitoring tool to control workflow, performance indicators and resources [26]. PM has been applied successfully in a wide variety of domains, besides information systems, like healthcare, manufacturing, tourism, education, finance, logistics, security, and several others [25] [27]. Capable of managing processes and workflow [28] [29], and by combining concepts from workflow management and social network analysis, is possible to analyse, discover and optimize social networks [15]. It is also utilized to evaluate and manage business processes [30].

Process discovery and conformance checking are considered the more notorious PM tasks, process discovery derives a process model from an event log and conformance checking diagnoses disparities and inconsistencies within the observed behaviour [14] [31] [32]. A process model defines the control-flow of the process,

identifying execution dependencies between every activity [33]. PM techniques can therefore be used to investigate and assess the alignment of the process model, optimizing the information system and business process [28].

A comparative study [34] of the quality metrics for fitness, precision, and generalization, concludes that there is insufficient empirical evidence on the behaviour of the several existing evaluation research metrics, to evaluate properly their correlations both within and across the different quality dimensions. The most significant relation identified was that, in general, metrics within the fitness and precision dimensions, are respectively finding only differences in sensitivity on a local scale, although no agreement was found between the generalization metrics considered.

Conformance checking has been recently studied from several different perspectives. Two dimensions of conformance: fitness and appropriateness have been proposed. Fitness is defined by the extent to which the log traces are consistent with the process model, quantifying the control flow's conformance. Appropriateness, measures the accuracy of the process model's description of the observed behaviour, considering the degree of clarity in which it is represented [35].

Conversely, recently the two measures in conformance checking more widely accepted are precision and recall, the latter often referred to as fitness [17]. Precision guarantees all the traces are considered and recall estimates the number of instances where the model does not correspond to the registered event logs. As described, the research on the applicability of PM in applicational support has concurring approaches, commonly dependent on the studied process and the objective.

4.3.2 RQ2 - What are the algorithms employed?

Table 3. Process Mining Algorithms

Algorithms	Articles
Heuristic Miner	[16] [19] [21] [22] [23] [24] [25] [30] [31] [32] [33] [34] [36] [38] [39]

Alpha Algorithm	[14] [16] [18] [23] [24] [25] [32] [34] [35] [36]
Inductive Miner	[16] [18] [23] [24] [25] [33] [34]
Genetic Miner	[14] [21] [23] [24] [25] [31] [38]
Fuzzy Miner	[22] [23] [30] [31] [36]
ILP Miner	[16] [23] [31] [32] [34]
Multi-Phase Miner	[14] [23] [24]
Declare Miner	[19] [22] [36]
Flower	[32] [34]
All Mining	[24]
Region Based Miner	[20]

The more commonly applied process discovery algorithms are the following: Alpha algorithm, Heuristic miner, Inductive miner, Genetic miner, The Fuzzy miner and the ILP (Integer Linear Programming) miner.

A comparative analysis [24], has considered the inductive miner the more dependable, supporting process incompleteness, parallelism, non-free choice structure, loops, hidden activities, and input log noise. The Inductive Miner algorithm is therefore considered the more applicable for process discovery [18].

Another study [27] identified that the heuristic miner was by far the most widely used, a fact supported by this research, however, since the date of the research papers was not considered, it is difficult to relate this preference with the outcome, since different algorithms were discovered several years apart.

Heuristic-based algorithms, generate models with frequency based ordered events. These algorithms are more appropriate for handling noise in event logs, however, low-frequency events may be ignored [21]. The Genetic miner is a search-based algorithm that tries to mimic the process of evolution [21].

Most discovery algorithms can be categorized into three principal types of algorithms, region-based, genetic, and traditional. Region-based and genetic algorithms'

purpose is to break the restrictions and limitations of the traditional, however, their mining efficiency is evidently lower than the traditional algorithms [36].

Other commonly used algorithms are the Fuzzy miner and Declare miner, recommended for highly unstructured processes, which represent most of the event logs from complex applications, returning manageable results [22].

The algorithm All Mining has been proposed [36], stating it is faster than Inductive-Miner and Declare-Miner, which avoids performing multiple database scans. The research shows that the model quality of All Mining outperformed the counter state-of-the-art process model mining algorithms, and the primary identified reason was its superior derailed methodology.

4.3.3 RQ3 - What are the Process Mining methods used to effectively identify constraints and errors in applications?

Table 4. Methods used to identify application's malfunction and errors

Methods Utilized	Description	Articles
Process Discovery	Discover a process model from event logs, to describe and represent the registered events.	[17] [18] [20] [21] [22] [25] [27] [31] [37] [38]
Conformance Checking	Analysis with the objective to detect inconsistencies between a process model and its corresponding event logs.	[17] [18] [21] [22] [25] [27] [32] [33] [35] [37]
Process Enhancement	Analysing existing business processes to optimize their performance.	[17] [18] [22] [25] [27] [37]
BPMN Notation	Business Process Model and Notation is standard to diagram business process.	[25]
Fog-Computing	Decentralized computing, data, computing, and storage are located between the data source and the cloud.	[19]

Negative Events	In a certain position in an event sequence, a specific event cannot occur.	[38]
Delta Analysis	Comparing the overall change in values.	[28]
Sample-Based Conformance Checking	Using trace samples to analyse the processes' deviation, to optimize the method.	[33]
Entropy-Based Conformance Checking	Utilizing exact and partial matching to identify disparities between the process model and the application event logs.	[17]
Pattern Mining	Identifies rules to describe specific patterns.	[36]

It is possible to categorize most studied methods as process discovery, conformance checking, or process enhancement [25] [27]. In real-world applications, the process discovery method must be adaptable and capable of adding new expected behaviour. Using process discovery and conformance checking to detect software errors, allows for the efficient reengineering of software systems supporting business processes [37].

Conformance checking is described as an analysis with the objective of detecting inconsistencies between a process model and its corresponding event logs, quantifying the conformance using several metrics [35]. Delta analysis compares the discovered model with some predefined process models identifying deviations [28].

As mentioned, precision and recall can be utilized as performance measures for information retrieval systems [17]. Using entropy-based conformance checking, with exact and partial matching it is possible to identify and quantify disparities between the process model and the application event logs.

Additionally, could discover the behavioural model for each software component, to improve the overall performance, by evaluating the event logs, and applying the

metrics to evaluate fitness, precision, and complexity [18]. Using conformance checking is possible to compare the discovered process model, often in Petri-net form, with the recorded event log, by replaying the event log to determine the model fitness [21]. Extracting the process mapping and performance, it is also possible to identify and fix issues in the process, using event logs to develop an efficient process monitoring system [20].

The concept of negative events, meaning that in a certain position in an event sequence, a specific event cannot occur, assuring that will not occur in the process model to be learned, can be useful in the discovery process [38].

A sample-based conformance checking was researched [33], using trace sampling to improve the efficiency, focused on the overall fitness, deviation distribution, and observed contextual deviations related to resource assignments, introducing result approximation to reduce the computation of conformance results. Trace sampling and result approximation were instantiated for the following conformance results: fitness as a conformance measure, identifying the deviation distribution that underlines individual activities non-conformance hotspots and deviations.

Applying frequent itemset mining to extract patterns is a popular method that could indicate how resources and activities are frequently used, however, the results may not be reliable [36].

Furthermore, the use of the BPMN notation has been proposed [25] to make the gathered information comprehensible, by mapping conformance and performance info. Process Mining techniques can be integrated with the existing applications, while BPMN models can be relevant to understand the results, and to guide process discovery by setting a suitable class of target models.

This research has focused on studying the PM's ability to be a support tool for applications, though, most studies in this field use as case studies business processes [28] [39] or applications that lack the necessary complexity. In the context of this research, to be considered applicable, proven scalability is crucial [29].

This research did not identify any study that has proven the necessary adaptability and scalability of an existing PM framework, applicable in a complex application, with intricate and singular characteristics.

Another critical aspect that was not possible to properly evaluate during this research was the computing demands required to efficiently implement the proposed methods. Considering that it is inevitable that, with large volumes of data from event logs, there is a threshold where possible compromises may be necessary.

The exception was the proposed [19] method, which uses fog-computing technologies, regarded as a complement to cloud-computing, to balance the computing and network capabilities, handling complex data sets, and providing accurate log contexts with lower overhead. This method was designed to discover process models from event logs. considering active concept drift detections.

Although PM has been widely researched, there are few studies that have studied the potential benefits of its application to monitor transactional applications and provide useful data for developers and managers.

4.3.4 RQ4 - What are the possible limitations of Process Mining as an application support tool?

Table 5. Identified Limitations

Limitations	Description	Articles
Incomplete Logs	Not all possible behaviour is evidenced in the available event logs.	[14] [15] [17] [32] [38]
Scalability	Applicable on a large scale.	[17] [29] [30] [33]
Inaccurate Information	Erroneous or misleading behaviour is represented in the event logs.	[15] [38]
Prior Knowledge	Knowledge about concurrency or parallelism, locality, or exclusivity of activities	[15] [38]

Runtime vs. Accuracy	Compromises between the speed and accuracy of the process.	[33]
Computational Cost	The associated cost to run processes and applications	[27]

In principle, Process Mining is applicable in every application or process with an event log [28], which is the starting point. Each event refers to an activity and is related to a particular case. The case events are ordered by time and can be seen as one “run” of the process. An event log contains a partial behaviour, one cannot assume that all possible runs have been observed and may only contain an unquantifiable fraction of every possible behaviour [32]. The absence of a complete process log, including every possible scenario, can be challenging, and unlike in information retrieval processes, where the collections are finite, the collection of possible traces encoded in a discovered process model, particularly in complex applications, are almost certainly infinite [17].

The fact that, in complex applications, some behaviour is extremely infrequent, signifies that, for most studies, this fringe behaviour is purposely ignored. However, the objective of this research is to be able to identify every possible issue, including the more sporadic behaviour. Since a perfect fit is not possible, existing PM techniques try to avoid overfitting by generalizing the model to allow for more general behaviour. This generalization is often achieved by representation language and basic assumptions about completeness [14]. It is a complex problem since the specificity is unquantifiable due to the absence of negative information. Discovered process model should allow the observed behaviour, and not include unintended and unnecessary behaviours, not identified in the event log.

Commonly identified challenges include accuracy, expressiveness, noise, incomplete logs, and prior knowledge [38]. Accuracy requires a trade-off between specificity and recall. Expressiveness describes the ability to comprehensively summarize an event log. Noise is the low-frequent behaviour unwanted in the process

model. Conversely, incomplete logs represent the other side of this challenge, since event logs do not contain the complete set of possible sequences. Prior knowledge refers to knowledge about concurrency or parallelism, locality, or exclusivity of activities [15] [38].

Another identified limitation of the discovery algorithms relates to the Petri net language bias, the basis for several algorithms, which is caused by a structural limitation and the complexity of transforming and adapting to the process model [27].

Finally, when employing PM, especially when the computing power is limited, some trade-offs may be necessary, considering the process runtime, to the detriment of its accuracy [33]. It is documented that several algorithms involve high computational costs, compromising their applicability [27]. Another limitation of some of the proposed methods is the scalability issue [33] [30], to be applicable on a large scale [29]. Additionally, several algorithms implemented required high computational cost and effort in their optimization [27].

4.4 Summary

The reliability of digital applications directly impacts services' quality, and user satisfaction. Additionally, the financial and reputational risks associated with applicational instability indicate it must be a priority in information systems management. Valuable and reliable information is critical to guarantee that the resources are effectively allocated and that the service provided is within the requirements.

After performing the literature review it was possible to identify the main contributions, and to identify the need for a new method to address the missing link between the technical capabilities of PM and the real complex and particular aspects of each application, with customizable and adaptable settings. The developed solution has the objective to apply some of the developed techniques to large organizations, with complex processes and high a number of activities.

5 RESEARCH PROBLEM

Process Mining research, specifically regarding process discovery and conformance checking, is commonly focused on process optimization [1] [2], with limited conclusive articles dedicated to PM as an applicational support tool. This research was developed to identify new opportunities to apply the existing PM techniques in a real context, identifying the encountered challenges and the methods to overcome them.

The field of PM has extensively studied the applicability of this tool to discover process models effectively and then use conformance checking analysis to identify deviations between the discovered model and the generated applicational event logs [17] [22] [33]. However, in most cases, the number of activities and processes analysed is minimal when compared to real world applications, which diminishes the prospect of a successful practical application.

One absent key aspect of the studied research papers is the necessity to regularly update the discovered process model, namely due to alterations or new features implemented. In large organizations, several transformational or technical developments and projects occur simultaneously, increasing the possibility of incidents impacting applicational services, and generating unexpected behaviour when the discovered process is not adequately updated. The everchanging and evolutive aspect of most applications will imply a constant need to evaluate the analysis methods applied. Additionally, no consideration was identified for the possibility of having different categories of users, with possible access constraints or privileges.

The conducted literature review was not able to identify the adequate consideration of the necessary computing power required for most of the proposed solutions, disregarding the need to compromise or make concessions.

The lack of ability to differentiate sub-processes severely complicates the analysis, and depending on their number may compromise it. Additionally, it is not possible to monitor and analyse specific processes that may be more susceptible to issues. This capability is critical to precisely monitor, novel or modified features or processes, to guarantee their correct implementation. Furthermore, the frequency of a

process may be completely unrelated with to its significance, and without an external validation, there is no way to guarantee every critical activity is included, and that necessary, and limited, resources are being efficiently applied.

One last critical aspect that seems to elude most researchers, or at least is not clearly and widely reported, is the importance of data quality. If the process model discovery is done using event logs with erroneous and unintended behaviour, it is necessary to manually identify and manage these cases. This analysis will be more intricate depending on the number of activities and the complexity of the discovered process model. The quality of the data extracted has therefore a decisive impact.

The fact that Process Mining has been successfully applied to discover unconformities in event logs, evidences its applicability, the missing contribution is a method to apply the technical capabilities to be adaptable to real applications.

6 RESEARCH OBJECTIVES

The research objective is to develop and implement a method to properly address the necessary customization to adapt Process Mining techniques to real and complex applications. The purpose of this method is to allow for specific organizational requirements, considering that each applicational will demand a specific and adaptable configuration. Also, the computational power is taken into account, reason why we have tested the method with a large data set. This method allows, when necessary, for a direct and adjustable compromise between costs and outcomes, adjustable for different configurations.

Regarding the conformance checking, and since a non-segmented approach would neglect less frequent processes, sample-based conformance checking will be employed for selected processes, providing additional statistical information. Additionally, and considering that the frequency of a process may be completely unrelated to its importance, the more reliable method to efficiently assign the necessary resources efficiently, is to classify and define them accordingly. One of the key differentiating characteristics of the proposed method is the segmentation of the different activities accordingly to its criticality, allowing for distinct, customized, and periodical analysis, without compromising the reliability of the method.

Moreover, considering a rapidly transformational environment, the proposed method pretends to guarantee that the process model is updated and accurately represents the existing processes. The recurring process model development resulting from the continuous update process will provide meaningful historic and evolutive analytical data. If the process model is obtained from corrupted or erroneous event logs, it must be corrected, updated, or manually modified.

The purpose of the informational outcome of the proposed method is to directly support business and technical decisions, identify application errors, provide performance analysis and user behaviour patterns, and produce periodic reports. In practical terms, each application has different priorities, setting them will have to be a business decision. The objective of this method is to provide the necessary guidance to efficiently manage real applications utilizing Process Mining capabilities.

7 DESIGN AND DEVELOPMENT

In this chapter, we have detailed the design and development of the proposed model, explored in these fundamental and correlated subjects: Business Analysis, Data Analysis, Process Discovery, and Conformance Checking. In each, we have detailed our progress, relating the proposal with the practical general application, identifying the possible issues or constraints, and the respective solutions.

To provide a visual representation, the proposed method is summarized in a BPMN model (see Figure 7), serving as a guideline for the implementation of the proposed solution.

The selection of each algorithm was the result of several experiments, the suitable options were evaluated concerning two key metrics: (1) accuracy, using metrics for calculating the fitness and precision; and (2) comprehensibility, evaluated by their simplicity.

7.1 Business Analysis

The visual conceptualization of the method provided by the BPMN diagram was developed to facilitate the modelling of several possible characteristics, and it pretends to illustrate the level of dependency of business input, critical to identify, characterize and differentiate each sub-process.

The business input is critical to initially identify and classify the available activities and processes, and, if applicable, categorize the users. This categorization will impact the quality of the results, any issue will undermine future analysis, is therefore necessary to be able to critically examine the results and repeat the process if unexpected problems are identified. The selected period will also be a business decision.

The method allows for distinct paths to create, or update, the process model, and to replicate the analysis. This flexible characteristic of the method was designed to support the necessary customization.

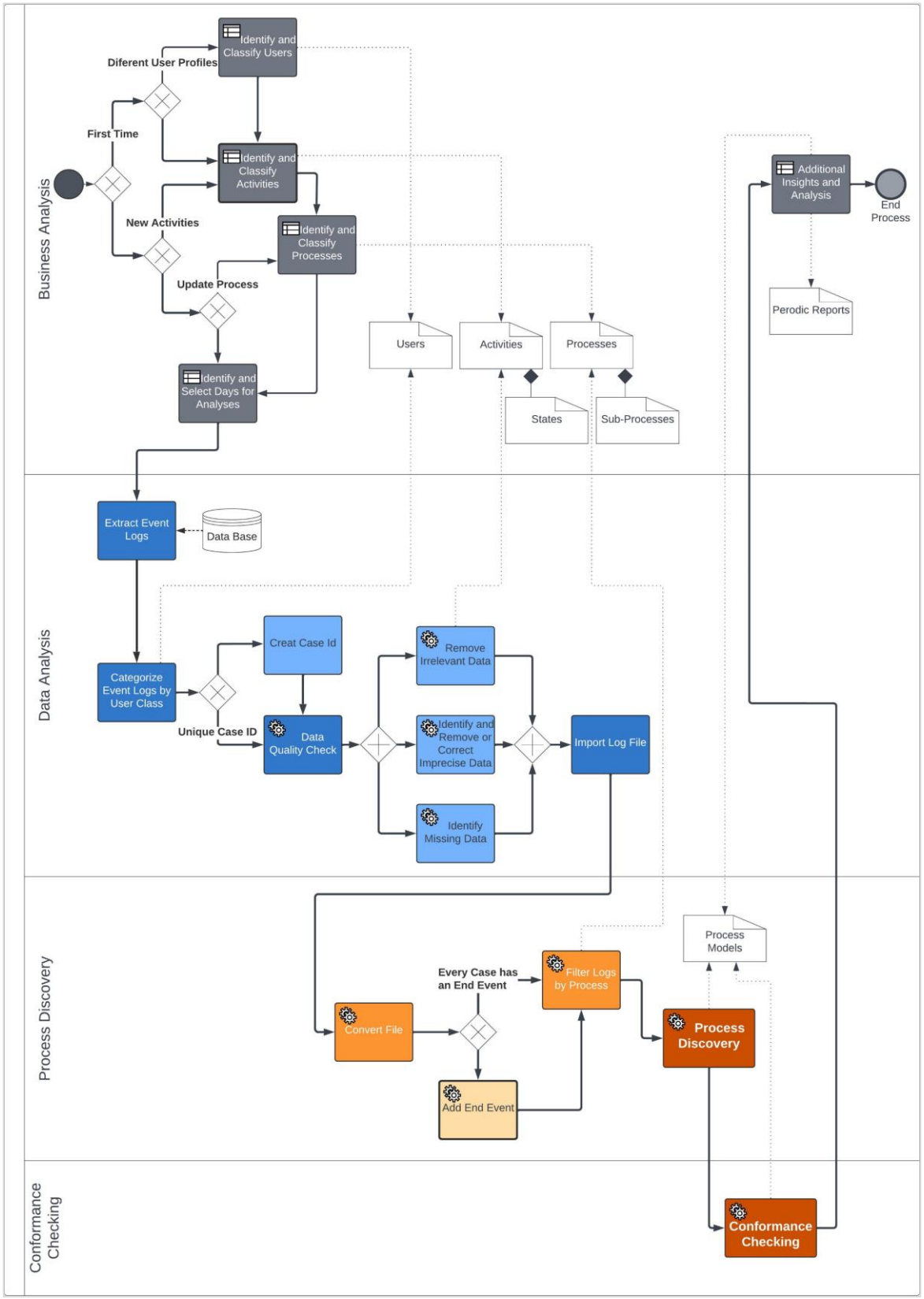

Figure 7 - Proposed Method - BPMN Model

As illustrated, the final step of the method, is categorised also as business analysis, utilising the generated data to create periodic reports. This task is just a representation, since the potential will be related to particular organizational requirements. However, its inclusion is demonstrative of the significant dependency of this method in the technical and organisation alignment. The input and output data are therefore defined according to business objectives.

7.2 Data Analysis

This procedure begins with the extraction of an event log according to the defined configurations. Concerning the elementary structure of the file, to perform any PM task, every log must have a time stamp, an activity, and a case ID. However, the case ID is often not present, with only a unique user identification. In these cases, considering that one user could have more than one session in the event log, it is necessary to differentiate them. A simple way to achieve it is to order by user identification, and time stamp, and to assume that, naturally, each user will start a new session in each new access, replacing the user identification for the required case ID.

The extracted event log, will commonly contain data quality issues. The next step should be removing unnecessary information since it will impact the performance and efficiency of the subsequent tasks. Subsequently, and critically, it is necessary to identify if the logs contain imprecise or incorrect data. A common example is activities that do not represent the user's actions. It is necessary to understand the possible reasons that justify each instance, and remove, or correct, them, accordingly. The last step is to verify if there is missing data: unique end or initial events, are commonly absent.

When possible, any incorrect, imprecise, or irrelevant data should be excluded from the extraction process. Specifically for error identification purposes, it is critical to minimise the time consumed. The extraction and conforming checking must be as efficient as possible, especially when analysing critical processes. Subsequent data

quality adaptations are performed using Process Mining algorithms further discussed.

The quality of the event log's data will determine the necessary effort either in the extraction process or in the application of PM algorithms. The available data, and respective sources, will directly impact the possible and necessary adjustments.

7.2.1 Algorithms

The first plugin applied was 'Convert CSV to XES', authored by F. Mannhardt, N. Tax and D. Schunselaar, the plugin converts the CSV (Comma-Separated Values) log events file into an open XES XLog object [40]. The XES standard log contains traces, and each trace contains events. Logs, traces, and events have attributes. The standard does not impose a set of mandatory attributes for each element and each event may include any number of attributes. Extensions provide semantics for these attributes [1].

The second analytical plugin applied was 'Add Artificial Events', developed by J. Claes, a simple plug-in that adds to every trace an additional start and, or end event. It is one of the most used PM techniques [41].

The filtering plugin selected was the 'Filter Log using Simple Heuristics' established by H. Verbeek, an essential tool to [42] filter the selected traces or events from the event log.

7.3 Process Discovery

In the context of an application where the user is free to navigate and select each option, the order of most of the activities is not imposed, the reason why it was necessary to first extrapolate the several processes within the application to be able to apply PM discovery processes and apply conformance checking, otherwise, several sub processes would be undermined. Several factors may complicate this initial effort, mainly the number of total activities and the quantity and complexity of the existing sub-processes. When analysing logs from complex applications, while

considering that each user action is an activity, it is apparent that each case will only contain a fraction of the available activities, and to increase the accuracy and efficacy, it is recommended to expand the number of cases to perform this analysis.

To extrapolate each process there are several possible approaches, however, due to the lack of research in complex applications, it is wise to experiment with several alternatives to try to establish the relationship between each activity and possible sub processes. Our approach starts by discovering the complete process, to identify the possible sub processes and their dependencies. The possible approaches are completely dependent on the number and complexity of the available processes. This effort is only necessary for the first analysis or when there are significant changes in the processes.

7.3.1 Algorithms

To automatically generate a process model, the ‘Heuristics Miner’ algorithm employs a set of predefined heuristics, allowing for process discovery from event logs with noise, or incomplete. This is extremely useful in real-world scenarios, where data quality may be an issue. The BPMN visualization has the advantage of being easily understandable, accurately, and effectively displaying the discovered model, for highly dynamic processes, with complex concurrency.

The ‘BPMN Miner’ authored by R. Conforti, despite being classified as analytical, is supported by the following discovery algorithms: ‘Induction Miner’, ‘Heuristics Miner’, ‘ILP Miner’ and ‘Alpha Algorithm’. The process model discovered provides a visual representation, in several available configurations.

The plugin was employed using the ‘Heuristics Miner’, specifically the ProM 6 version, with the default parameters, all tasks connected, no long-distance dependency and ignoring loop dependency thresholds. Additionally, we applied the algorithm ‘Mine for a Heuristics Net using Heuristics Miner’, authored by A. Weijters [43] [44].

Since the principal discovery algorithms are heuristic-based, the discovered model may not perfectly represent the effective behaviour, ignoring infrequent behaviour, reason why other modelling alternatives were also employed.

Using the filtering plugin selected, ‘Filter Log using Simple Heuristics’ it is possible to visualize the model by applying the available visualizations, based on their applicability have selected the ‘Log Skeleton’ and ‘Trace Variants’. The ‘trace variants’ visualization is suitable for analysing the more frequent behaviour, particularly in less complex models, where a small number of variants can represent a significant majority of the total existing traces. It is the most suitable alternative to analyse individual traces, particularly for user behaviour analyses.

The Log Skeleton based approach is an advanced mining algorithm that returns, a ‘log skeleton’, a declarative process model. This algorithm visualizer plugin is selected using the “Visualize Log as Log Skeleton”. From an event log, the algorithm extends it with artificial start and end activities and discovers from the extended event log the collection of initial specific constraints, keeping only relevant constraints. It also provides statistical information for each activity, the total occurrences in the event log and the minimum and maximum number of instances in one case.

7.4 Conformance Checking

This research was focused on the control-flow perspective to understand each process and the influence and importance of the activities. After establishing process models individually, and comparing them with subsequent process models, it is possible to identify deviations and unexpected behaviours. Especially for critical processes, it is possible to continuously apply this technique to validate and evaluate the performance of the application.

To accurately identify issues in complex models, it is necessary to understand the historic data, and to adapt and update the model accordingly. Each organisation’s objectives and demands will vary, this analysis is therefore dependable on those requirements.

7.4.1 Algorithms

The selected plugin to perform Conformance Checking was the ‘iDHM' (Interactive Data-Aware Heuristics Miner) developed by Mannhardt et al. [45], which supports and enhances process discovery, as an interactive approach with several real time adjustments and configurations, with immediate quality feedback to create and analyse process models perfectly aligned with the observed behaviour.

The interactive nature of the plugin, offering several distinct alternatives, requires experimenting with the plugin's interface and functionality. Highly interactive plugins can demand higher computational power, to guarantee the performance with large or complex event logs.

From these experiments we have concluded that the DFG graphical representation was the more suitable. It illustrates the frequency of direct transitions between activities, identifying the most common paths and sequences.

The plugin and visualization were selected due to their applicability in an organisational context, both are compatible with dynamic processes, with missing events, loops, or concurring activities. The frequency of the included activities is adjustable, to exclude, from specific analysis, infrequent behaviour.

In the next chapter, the algorithms utilized in each step of the applied method in the case study are clearly identified.

8 DEMONSTRATION

This chapter details the implementation of the use-case divided by Business Analysis, Data Extraction and Quality, Process Discovery, Conformance Checking, and Additional Insights. Relating to the conceptual model previously proposed, the demonstration will be mostly focused on ‘Data Analysis’, and ‘Process Mining’ (see Figure 7), the latter represented by Process Discovery and Conformance Checking. As identified in the method, the final section is related to the final business analysis.

8.1 Introduction to the Use-Case

To demonstrate the practical application of the proposed method, we have applied the proposed solution to real data from a large organization in the financial sector, Company X, with more than 100k monthly users in the selected application. The event logs utilized represent the available operations, transactions, or queries, of a transactional mobile application. The objective of this use-case is to apply the developed proposal to a real application, to identify possible limitations, and especially, unidentified or unresearched potential practical applications.

8.2 Business Analysis

Started by analysing the available user’s categories, in the selected use-case there are several available configurable settings that can condition the available transactions, with access to the selected application there are 4 possible configurations. To avoid over complicating future analysis and considering that a single category represented most users, only this was considered.

For confidential constraints, the efforts of selecting and categorizing each activity and sub-processes are not properly justified and detailed in this research, evidencing only their relation in the discovery section. Following this categorization, the naming of the activities considered their subsequently identified sub-process.

Selected several sets of event logs, for the following purposes, two sets representing equal and consecutive time periods, were utilized as the principal and the control data sets, the latter to minimize eventual unidentified issues or problems within the selected dates. Additionally, several sets were extracted from periods with identified

issues that affected the normal use of the application, to be utilized in the conformance checking analysis. A final data set was extracted, from several uninterrupted days, with one million traces, for performance analysis purposes, detailed in the evaluation of the demonstration. The selected approach was the conclusion of several experimental tests, with different strategies.

To be consistent with the proposed method, the final business analysis input, will be addressed in the final section of this chapter.

8.3 Data Extraction and Data Quality

No user data was utilized, and the event logs were previously anonymized, to guarantee total anonymity while providing significant data.

The data was extracted using Microsoft SQL (Structured Query Language). The selected data extracted from the data base were only: the mandatory timestamp, the unique case ID, the activities, and their respective state. In this use-case, only a user identification was directly present in the log, which could be feasible by applying the solution proposed in the research and development chapter. However, opted for using a unique session ID, comparable to the case ID, which was available in the additional data of each event, in XML format, using the cast expression in SQL. The events logs were separated in SQL by the selected user category.

The complete event logs for the selected application contained lots of additional information for each event. Most of the data was removed for confidentiality constraints, and classified as irrelevant for the research. The data utilised for process discovery was selected, randomly, from several days, however, to ensure the absence of erroneous behaviour, or possible errors, any periods with identified constraints were excluded from process discovery.

To avoid misrepresenting user behaviour, it is necessary to understand what each activity represents. In this case study, several activities were not originated by users, and were considered internal transactions, classified as imprecise data that

distorted the perceptions of the process model without providing any meaningful information, and therefore removed.

The included activities all represented an action performed by the user, varying from logging in to performing an operation, which may require several actions for successful completion.

Additionally, in an application with several recurring user sessions, the selected initial time of the event logs will coincide with an active undergoing user session, consequently, when extracted, these traces will not include the log on activity. It is necessary to apply the filtering plugin to remove imprecise data, specifically traces that didn't start with the established initial activity.

Another significant aspect, already mentioned, was the fact that the studied event logs contained additional critical data, representing the state of each activity, The state was treated as if it was a subcategory of the activity. Since several of the activities have distinct possible states, the impact of including them is the increase in the number of activities. This data will be utilized accordingly, the respective activities with a state that were considered either imprecise and/or irrelevant data were removed from the extraction process. Additionally, some states were considered equivalent, in cases where it was possible, we have categorized these, and grouped them, accordingly, resulting in the following three states: "Done"; "Rejected"; "Error".

The state "Done" represents the completion of the activity. It is the expected state for every successful activity. The state "Rejected" is expected in certain scenarios, meaning it may only signify the operation is not available with the requested parameters, and may be also a user's input error, more important than identifying rejected activities is the proportion of successful activities and their comparative absolute number. Conversely, the state "Error" is not an expected behaviour, it may be explainable, and not considered an issue depending on the circumstances. However, in most cases the state "Error" signifies there is an application issue, which may or not be related and contingent to a restricted number of activities. The meaning and context of each state are particularly relevant in the conformance checking section.

One of the principal purposes of this research is to clearly identify applications' issues. In this pursuit, one important aspect to consider is that, amongst general errors, some will impact every user, while others may only impact a percentage of users, and the same principle is applicable to each activity.

Following the described method, as exposed (see Figure 8) from the total available distinct activities (n=39), when considering all the possible states, resulted in the total of available activities (n=130). From these, we have identified and removed every internal, irrelevant, or redundant activity, resulting in the remaining essential activities (n=68). The extracted data was then thoroughly analysed, identifying each critical activity (n=47). From the original distinct activities (n=39), only about half (n=20), were included.

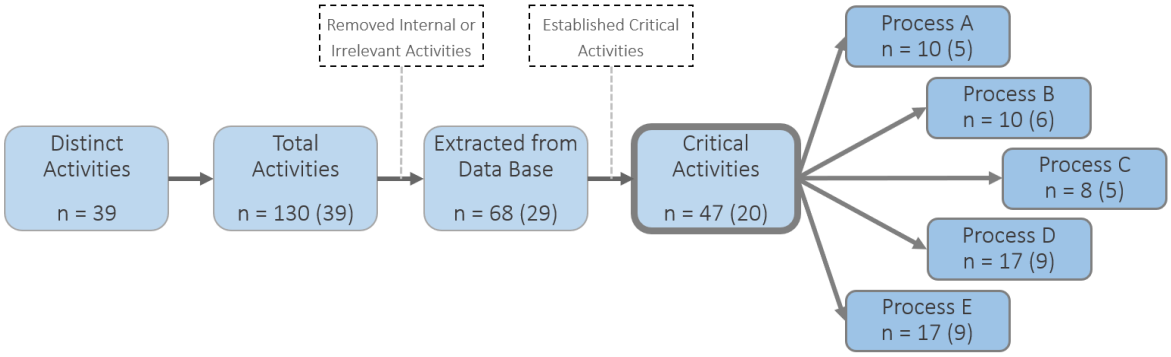

Figure 8 - Activity Selection Process

The evident issue with blindly selecting the activities based only on their frequency is to neglect potentially critical operations, and to overemphasise irrelevant activities. To illustrate this issue, we have extracted with SQL, from several days, the total count of each respective activity, and have calculated their relative frequency (*rf*), and cumulative relative frequency (*crf*).

As exposed in Table 9 in Appendix III – Activities Statistical Analysis, only the activities with the respective letter and number were included, the activities that appear as "--" were considered irrelevant. The states were excluded, since the purpose was only to demonstrate that the frequency of an activity does not directly impact its relevance, particularly in unstructured and complex processes, where the order and frequency of most of the critical activities vary.

In this case study, the total number of registered activities considered irrelevant in the event logs is higher (57%) than the number of included activities. Considering the top 10% of activities (n=13), highlighted, only about half (n=7) have been included. Additionally, if we had considered only the 99,5% more frequent behaviour, excluding every activity, and respective states that cumulatively occurred less than 0,5%, which would normally be more than appropriate, about two-thirds (n=33) of the total critical included activities (n=47) would have been ignored. As demonstrated, applying PM to complex and large processes, without analysis, categorisation, and segmentation, would either ignore part of the activities, or would be incomprehensible.

Ultimately, a CSV file was extracted from SQL, and imported to the selected PM application, ProM, where every subsequent analytic filtering process discovery and conformance checking was performed, starting by converting the extracted CSV file to XES.

As mentioned, there are several available commercial applications, we have selected ProM(3) 6.12 (Revision 45684) since it is an open-source tool widely used in academia [1] [2] [40] [46].

8.4 Process Discovery

In the selected use-case, and usually in other mobile applications, the log-off is not a registered activity, and as mentioned, in an applicational context, the logging off operation is not mandatory since it is possible that the user will simply leave the application. However, several algorithms require a unique end event, to guarantee their combability, therefore an end event was added.

In the studied use-case, the first challenge was to overcome the complexity and intricacy of analysing a process model with an unresearched number of activities. After removing the imprecise and irrelevant data, still had several (n=47) activities. To illustrate the extension of the inherent complexity, an example of a process

(3) https://promtools.org/

model discovered from the complete event logs was attained, excluded due to the minute scale which prevented any possible visual analysis in this context. Consequently, to efficiently discover a process model with practical applications, our approach was to subdivide each underlining process, ending with 5 defined process models. Each represents a fundamental process within the application. Noteworthy, and expected, that some processes must share common activities.

The activities were then named according to their principal process and considering their expected order, when appropriate. Conversely, the letter attributed to each process did not follow any order or reasoning. Identifying and categorizing each activity is a demanding initial effort, however, it will only be required and repeated in analysis when there are significant alterations in the process model.

Due to the interchangeable characteristics of some of the activities, from dissimilar processes, it is not possible, at least with the currently available tools, to automatically extract each sub-process from the complete event logs. The need to interrelate the same activity to different processes was studied, without it, some processes would be incomplete, and, critically in errors affecting these transposable activities, the underlying cause could be misrepresented.

There are tools to auto discover sub processes, and several unsuccessful attempts were made. The results were invariably incomprehensible or over simplified to be considered appropriate. The method applied, using the plugin “Mine Local Process Models”, did not generate any process with five or more activities, and, with four activities, the process with the highest score did not represent any of the relevant observed traces. Considering that the first and last activities are shared by most of the processes, this method proved completely unsuccessful. None of the resulting processes were useful.

To initiate the extrapolation of the existing subprocesses, the starting point was the discovery of the full model, containing every selected activity, but only considering the state ‘Done’. Using the ‘Heuristic Miner’ was possible to extract a comprehensible BPMN model (see Figure 9). The number of activities (n=29) confuses the initial

analysis, however, after the categorization and filtering efforts, the resulting process is intricate but manageable. This preliminary stage allowed the clear identification of each individual process, highlighted with the respective colours, that were selected without any particular reasoning.

There are clear similarities between ‘Process D’ and ‘Process E’ activities, which could even be considered a single sub process. One reason to divide it into two sub processes was primarily the total number of combined activities. The reasoning for the selection done according to their respective established category, considering the type of operation, was to create an efficient method to forward each issue to the corresponding team. Commonly, different operations are supported by different services, and teams.

Is also possible to extrapolate the relation between each process, particularly the dependency of the ‘Process A’ in the processes B, D and E. The activity Z1, is the initial step of every process, except B. The BPMN model provides significant and interesting structural information; however, due to the showed complexity, it does not effectively represent the observed behaviour.

This effort is highlighted in the resulting segmentation of the five sub processes. To divide the event logs into processes, the filtering plugin was applied resulting in the 5 respective event logs. From this moment, each event log was treated separately, underlining the segmentation that characterises the proposed method, allowing for distinct settings and objectives defined accordingly. The method applied for each process is discussed subsequently.

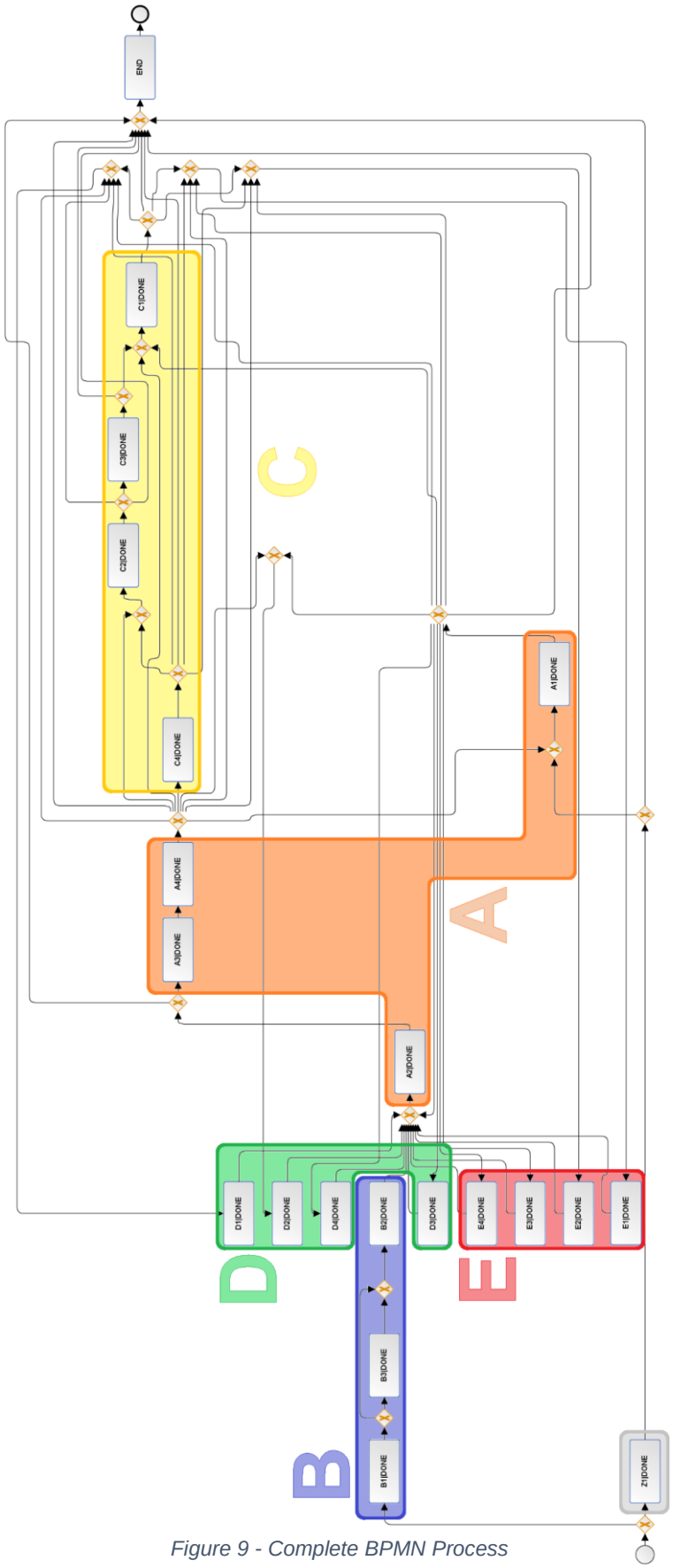

Figure 9 - Complete BPMN Process

8.4.1 Process A

In this use-case, ‘Process A’ is critical and essential, and for this reason we have selected the four core activities, [A1, A2, A3, A4], which are expected to be performed in this sequence. Employing the described method, which resulted in the filtered event log, initiated the process discovery analysis, by viewing the process, as a Log Skeleton (see Figure 10).

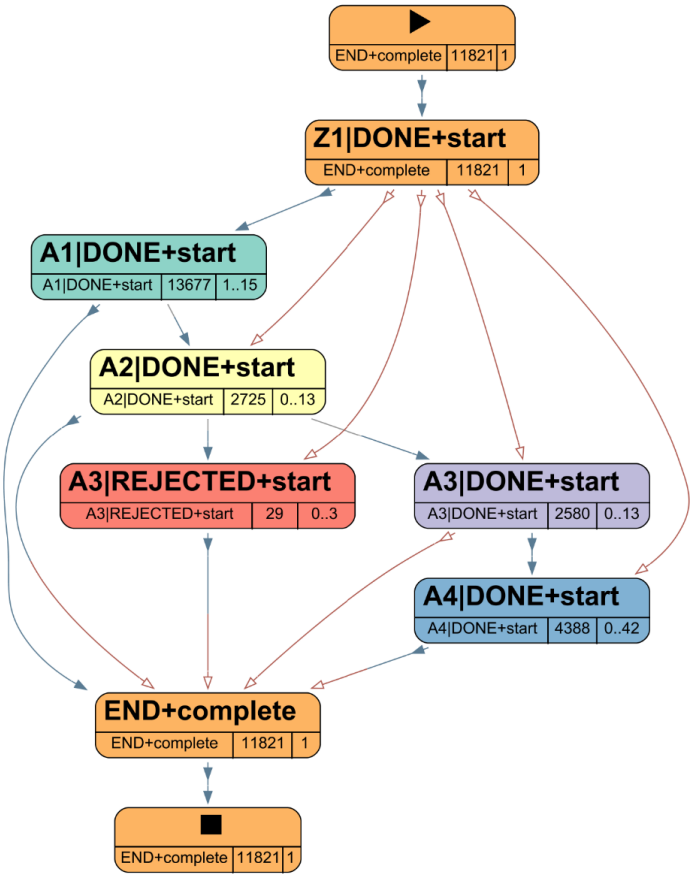

Figure 10 - Process A - Log Skeleton

By analysing the log skeleton is possible to extrapolate valuable information: the number of completed operations, represented in A3-Done; the number of total attempts, in A2, the number of rejected activities, A3-Rejected; the total transactions in A4; and the total number of cases analysed, in Z1 or END. One important remark is the last activity, A4, is expected to be repeated since it supports multiple operations simultaneously.

Subsequently, we have applied the heuristic miner to extract a BPMN model (see Figure 11), the principal discovery visualization utilized. Previous analyses were necessary to ascertain and reinforce the validity of this model. This representation also supports the gathered data. From a normal observed behaviour, only one activity, A3, can have an accepted 'Rejected' state, explained by the user's input error.

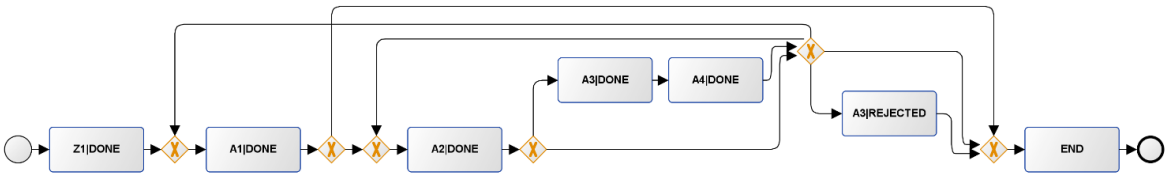

Figure 11 - Process A - BPMN

As mentioned, the user can conclude the process after activity Z1, and after, A1 or A4. Conversely, when the activity A2 is initiated, it is expected to go through to the activity A3-Done followed by A4. When the activity A3 is rejected, the user must return to the previous activity, A2, to continue the process.

This flow will be conserved in future processes that include these three activities, A2, A3 and A4, and the evidence of the importance of this process will be further apparent in three of the following four processes. Issues in this 'Process A', are critical and will impact most (n=15) of the distinct activities (n=20) supported by this application. This process must be constantly monitored.

8.4.2 Process B

The second identified process, 'Process B' was chosen for its critical factor, its particularity and relatability. In this use-case, this is the only process that has a complete and predictable order for successful processes, with only one, variant, a non-mandatory activity, B3. The complete process would be, [B1, B2] or [B1, B3, B2] followed by [A2, A3, A4]. For these reasons is by far the more accessible process, and it is easier to identify possible issues.

Arguably other researched alternatives could have been utilised for this case, however, to prove the method's consistency and validity, followed the exact same principle, applied previously, we started by viewing the process as a Log Skeleton (see Figure 12).

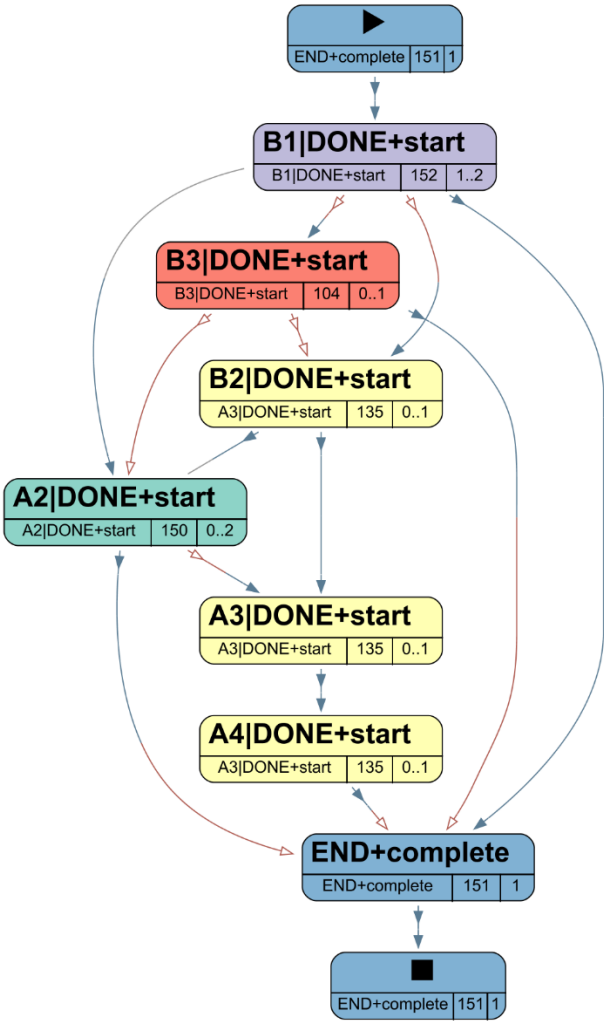

Figure 12 - Process B - Log Skeleton

For this process the Log Skeleton also provides significant statistical information, being able to identify clearly: The number of initiated processes in B1, the processes that required this optional activity B3, the completed process in A4, and finally, the abandoned processes, subtracting A4 from B1. The B3 activity is not mandatory within the process analysis, however, it is required for some users. The absence or frequency reduction may indicate a possible issue.

Unexpectedly, the log skeleton representation, identifies a strange path from B1 to A2, complicating the analyses. And, when analysing the trace variants, an outlier was identified, this one trace [B1, A2] was considered to be a possible error, had to be manually analysed and proved to be an infrequent but possible behaviour. This specific situation served as a real example of the applicability of this approach, and the need to understand the underlying application.

In this process, due to its simplicity, we also extracted the Trace Variant (see Figure 38 in Appendix IV – Process Discovery – Process B) since every variant is visible. As evident, is a comparatively infrequent process, however, it is critical, and represents an important operation. Due to its infrequent behaviour, from the selected data set, no rejected activities were identified, however, the user's input errors may generate a normal and expected reduced number of rejected activities. Additionally, the reason why it is necessary to include in this process the core activities from 'Process A' [A2, A3, A4] is the fact that errors in this process may occur in one of these final activities.

The generated BPMN model (see Figure 13) purposely ignores the outlier behaviour, providing the expected process model from the event logs. As the visualization suggests, is an extremely simple process, with the only variant being the option activity B3.

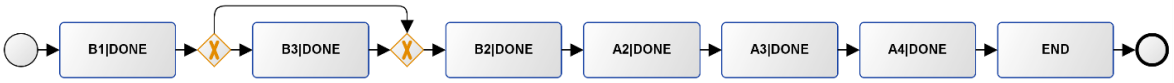

Figure 13 - Process B - BPMN

Finally, it is necessary to emphasise that since it is the only process that does not start with the initial Z1 activity, even if every activity was considered, with this manual assessment, it would be removed or ignored from most, if not all, the traditional approaches.

8.4.3 Process C

The 'Process C' is the only process that does not include any activities that belong to other processes. The activities selected represent the remaining critical activities

available in this mobile application that are not directly related to any of the other processes. Therefore, this process is very unstructured. Only two of the activities have an imposed order, [C2, C3], otherwise, the number and order of activities are arbitrary.

One particularity is the fact that activity C4 is also represented in ‘Process E’ and ‘Process F’, since this activity is utilised in dissimilar processes, and errors may be only applicable for some. It is consequently necessary to include it in every process to identify the real possible reasons for each error.

The generated Log Skeleton (see Figure 14) confirms the expected behaviour. The log skeleton translates the frequency of the C1 activity, and the number of maximum activities in one trace, which for C1 was a surprizing value (n=603). This value will be further discussed in the user behaviour section. Additionally, only the activity C4 has rejected activities in a comparatively reduced number.

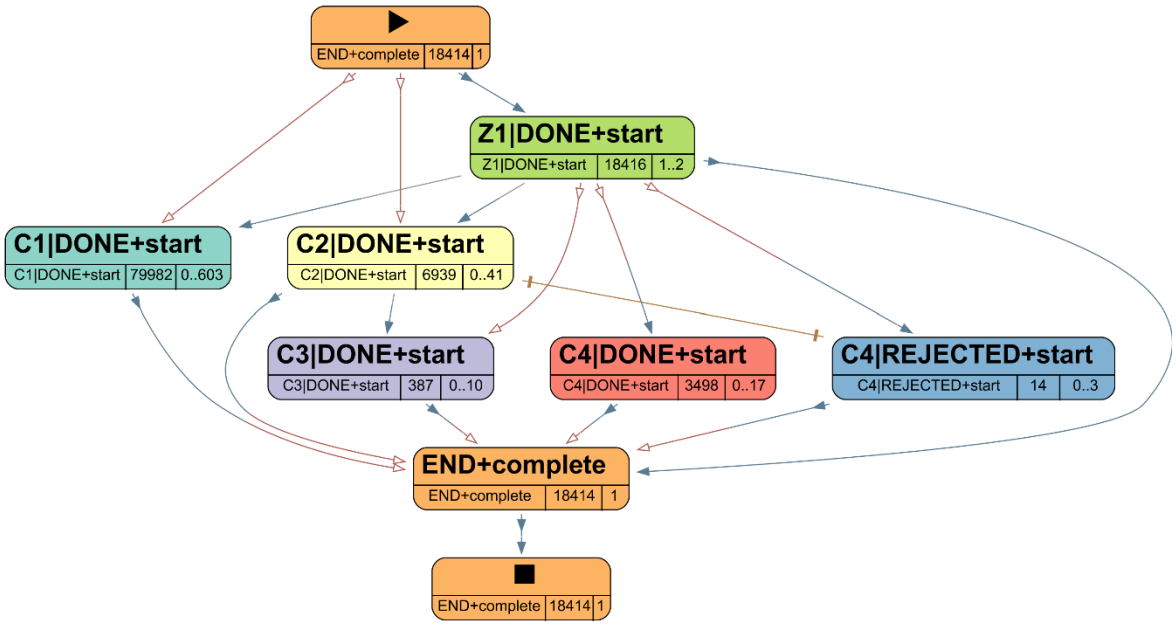

Figure 14 - Process C - Log Skeleton

The ‘Process C’ BPMN model (see Figure 15) clearly illustrates the unstructured characteristic of the process, as evidenced, by only two of these activities having an imposed order C2 and C3 are sequential. Otherwise, every other activity is completely independent, and can be performed exclusively.

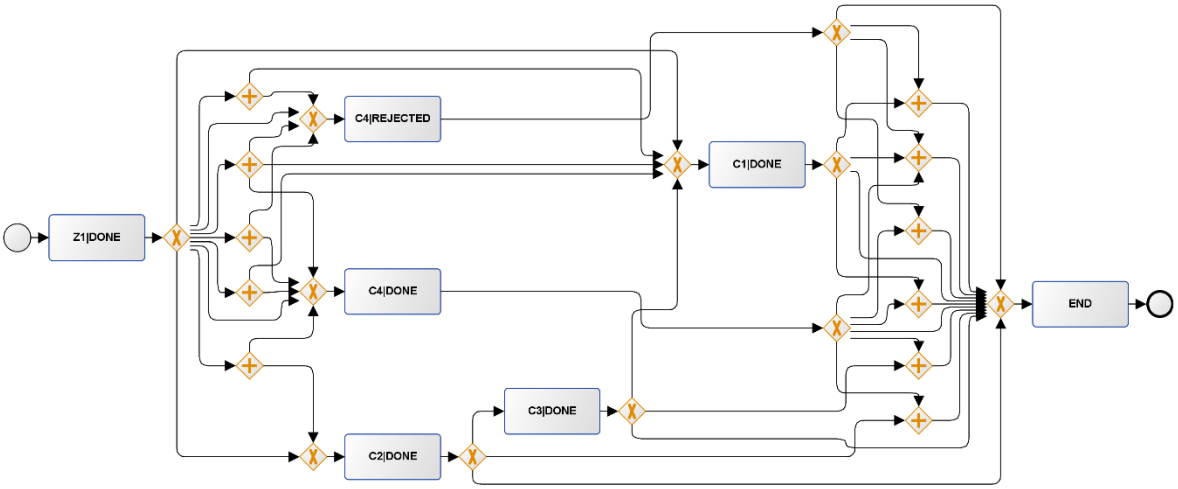

Figure 15 - Process C - BPMN

The indefinable characteristic of this process, harshly contrasting with the previously analysed, excluded any possible conventional analysis.

8.4.4 Process D

The ‘Process D’, as well as ‘Process E’, represents several distinct but comparable operations, each represented by a single activity, D1, D2, D3, D4. To conclude each operation, the following core activities [A2, A3, A4] must be concluded. Additionally, the activity C4 was included, it is optional but with significant representation, and identifies possible issues only related to this process.

The complete process was visualized as a Log Skeleton (see Figure 39 - Process D - Log Skeleton in Appendix V – Process Discovery – Process D), due to the number of activities, and possible variants, when analysing this data, it is possible to discern that one of the principal activities, D1, is more statistically significant.

The more relevant conclusion was the confirmation of the importance of the included activities, A2, A3, A4 and C4. The Log Skeleton provides also significant statistical information regarding the frequency of each of the activities, D1 (n=1021), D2 (n=146), D3 (n=23), D4 (n=83). The historical information of these respective values, at different time periods, will be valuable to identify possible deviations.

We also extracted the complete BPMN model (see Figure 40 in Appendix V – Process Discovery – Process D), corroborating the relation between this process and the included additional activities. However, this model, due to the number of traces and variants of infrequent behaviour, does not accurately describe the observed behaviour.

Since the visualization of this process is complex when every state is included, the process to extract the BPMN Model (see Figure 16) was repeated excluding the "Rejected" activities to overcome the identified issues. Additionally, the 'Process A' activities were merged.

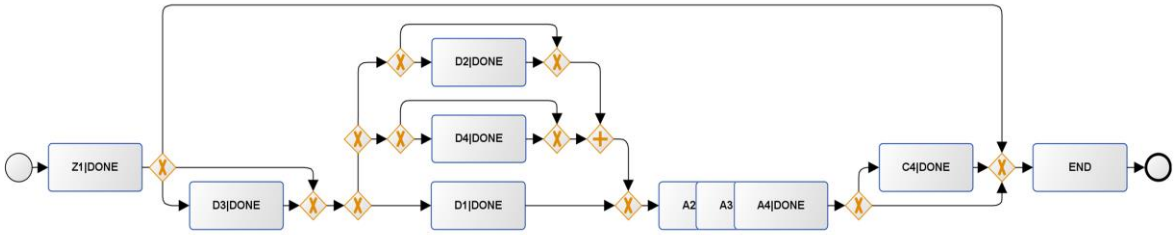

Figure 16 - Process D - BPMN

By analysing the simplified BPMN Model, it was possible to successfully discern the accurate process model, without any strange or unexpected behaviour. The different levels of detail considered in each visualization demonstrate the importance of this task, and the need to use distinct approaches.

Additionally, especially for this process, as well as 'Process E', it was necessary to use relatively large data sets, otherwise, with a smaller data set, the less frequent behaviour was grossly misrepresented, and some expected activities were not even present.

8.4.5 Process E

As described, 'Process E' is extremely similar to 'Process D', sharing several intrinsic characteristics. It also represents several distinct operations, each represented by a single activity, E1, E2, E3, E4. The activities A2, A3, A4 and C4 were included for the same reason previously mentioned.

The method applied followed the same principles utilized in ‘Process D’, starting by visualizing the complete process as a Log Skeleton (see Figure 41 - Process E - Log Skeleton in Appendix VI – Process Discovery – Process E). Similarly, to ‘Process D’, the Log Skeleton does not accurately describe the observed behaviour, providing principally significant statistical information regarding the frequency of each of the activities, E1 (n=2724), E2 (n=263), E3 (n=96), E4 (n=60).

The generated BPMN Model (see Figure 42 in Appendix VI – Process Discovery – Process E) accurately represents the observed behaviour, only demanding to analyse due to the number of activities, and scale. Serves as another example to advocate for the segmentation of included activities.

As done in the previous process, the BPMN Model (see Figure 17) was discovered excluding the “Rejected” activities, providing a simpler visualisation of the process, that ignores input errors, focusing on the successful activities.

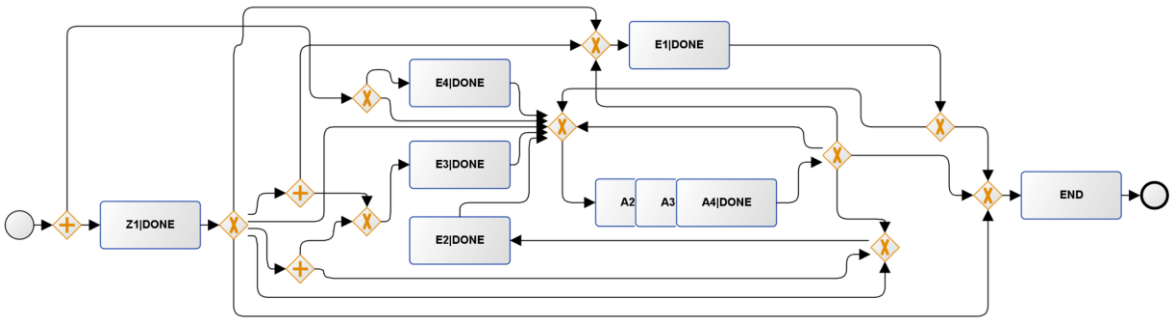

Figure 17 - Process E – BPMN

As demonstrated, the intrinsic characteristics of each process and respective activities are indispensable to effectively discover each model from the observed behaviour.

The selection of the utilized BPMN visualizations considered the technical and organisation requirements and proved to be applicable and valuable for the studied use-case.

8.5 Conformance Checking

The application of conformance checking is done for several purposes. In this proposal, and applied to the selected use-case, the principal objective is to identify possible errors, issues or deviations from the established expected behaviour represented in each sub process model.

Following several experiments, we concluded that, for complex and unstructured processes, considering the level of knowledge required to successfully categorize each process, it was more effective to manually identify issues by comparing the process models with the observed behaviour, to identify missing, unexpected, or abnormal number of certain paths or activities.

For ‘Process B’, due to its simplicity, it could be possible to use a different approach with the purpose of automatically identifying missing or unexpected activities. However, to guarantee a uniform and more widely applicable approach, we have used similar methods to each process to demonstrate their efficacy in a wide application.

The only conformance plugin applied, was the iDHM, applied individually to each sub process. The visualization selected was a DFG which facilitated the correlation with every model discovered. Since the purpose is to identify every issue, even in infrequent activities, the frequency setting was always set to 1.

For each respective process, we have analysed specific previously identified issues, to demonstrate the application of the method. These are real scenarios, the demonstration is done with past events, however, the process would be the same in a live situation. Most time periods selected have the same duration, the number of expected events for each process will vary depending only on the time and day.

To demonstrate distinct situations, we have selected a total of 14 cases, for the 5 identified processes. For some, it was necessary to use additional examples to demonstrate the applied reasoning, this necessity, and the number of cases per

process, varied for several further detailed reasons. The total number data sets utilized was 18.

These cases were selected to represent every identified incident. Only similar incidents were excluded since the underlying analysis and insights would be the same.

One additional remark: commonly in most examples the incident starts in the middle of the data set, time wise; the decision to select these event logs from a predefined time stamp was to be a realistic simulation; in a live situation, the issue and initial time would not be known beforehand.

Regarding the other commonly studied perspectives, only the time perspective was considered to have interesting potential, however, in this research, it was not possible to study the possible practical applications. The organizational perspective was not possible to employ using the selected case study, since required critical data, is absent from the event logs. The case perspective could have been employed, however, for this case study no sufficient prospective value was identified to further research this topic in such a complex application as it could undermine the efficiency of the research.

Performance analysis is defined [1] in different ways. Three typical dimensions of performance are: time, cost, and quality. Each has several performance dimensions, however, for the studied use-case, the focus was only on the quality component. The cost dimension was not applicable, at least with the available data, and was not considered relevant. The time element is obviously present, however, as explained it was not currently considered for performance analysis.

One key identified aspect that differentiates this case study is regarding infinite loops, a commonly faced issue, when the event log does not contain the possible model behaviour, however, was classified as normal behaviour and it was not an issue.

8.5.1 Process A

The analytical data can be sufficient to identify possible errors in this case, any unexpected activity not represented, will be a worrying sign. Other possible errors can be detected by the proportion of the A3 rejected activities, by the number of abandoned operations subtracting A3-Done by A2-Done and by the number of successful transactions A4. Additionally, it is possible to establish an expected number of transactions from the total number of cases, significant alterations could indicate possible issues.

It is essential to understand what those activities are, since there is always a percentage of errors that must be tolerated. In an online mobile application, the lack of completion of an operation, may not signify an applicational issue; other possible, and common, explanations are, issues in the user's connection, equipment issues, or any personal impediment or interference. It is therefore necessary to establish an adaptable threshold, based on the normal frequency of abandoned processes to accurately identify possible issues.

To identify these errors in this process, the more efficient method was to analyse the log skeleton and BPMN model to establish a baseline and then compare with the future observed behaviour to recognise possible deviations. To demonstrate the practical effectiveness in 'Process A', and since this is the more fundamental and critical process, three distinct situations were selected to perform these analyses.

In 'Case_1', the following DFG (see Figure 18) was extracted, and the effects of this constraint are evidenced by several errors in the activities A3 and A4, highlighted. While using the DFG visualization it is possible to focus on part of the activities. Despite having several successful activities, these errors indicate that part of the process is not being completed successfully.

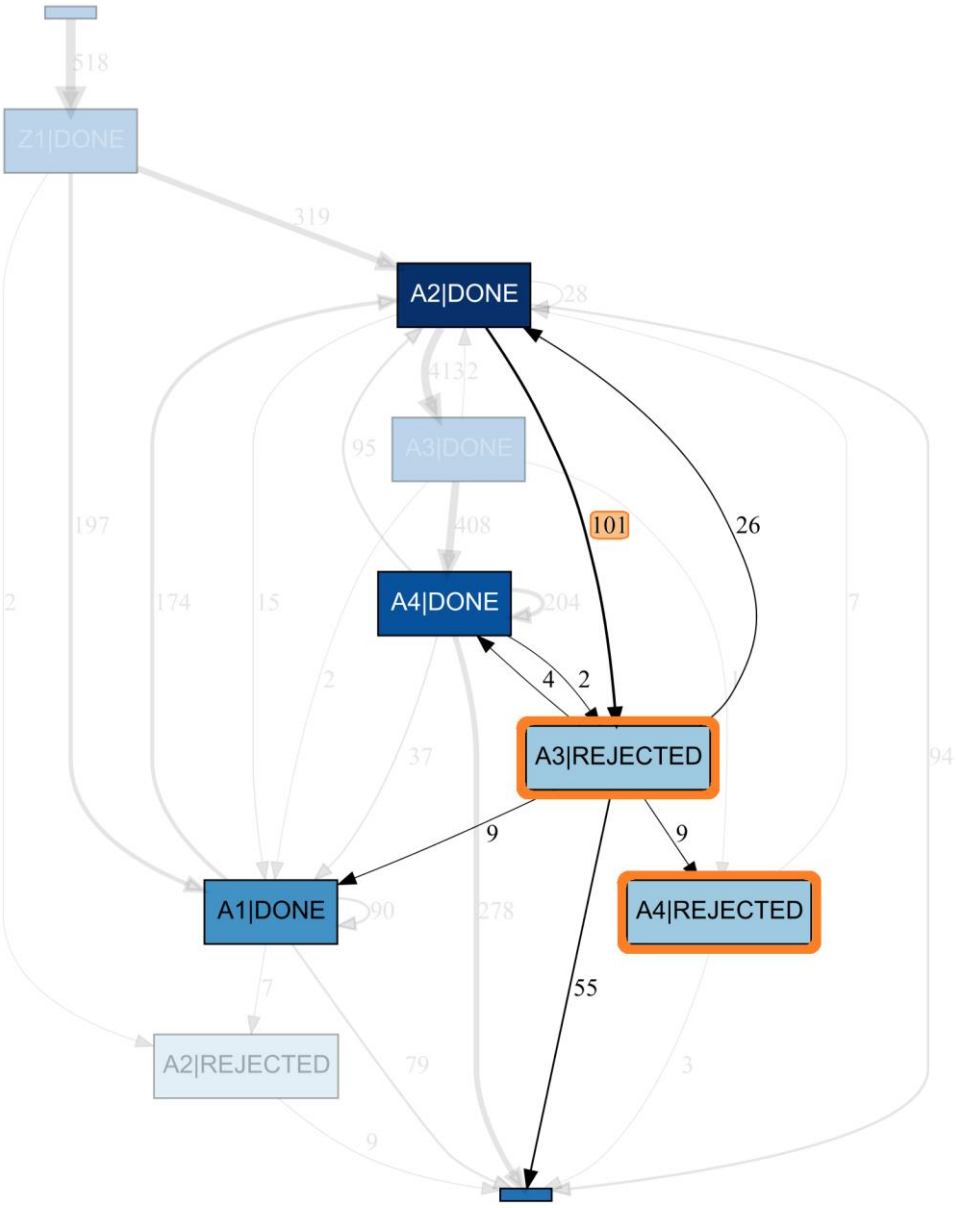

Figure 18 - Case_1 DFG - Number of Rejected Activities

As mentioned, errors in this process, will undoubtedly affect all processes, except 'Process C'. It is necessary to differentiate the origin of each problem. For this reason, different situations were analysed. In 'Case_2' (see Figure 19) there are no unexpected activities. If only the same reasoning was applied, it would apparently be an uneventful case. However, the error's evidence is the reduced number (n=259) of A4-Done activities, compared with (n=396) A3-Done.

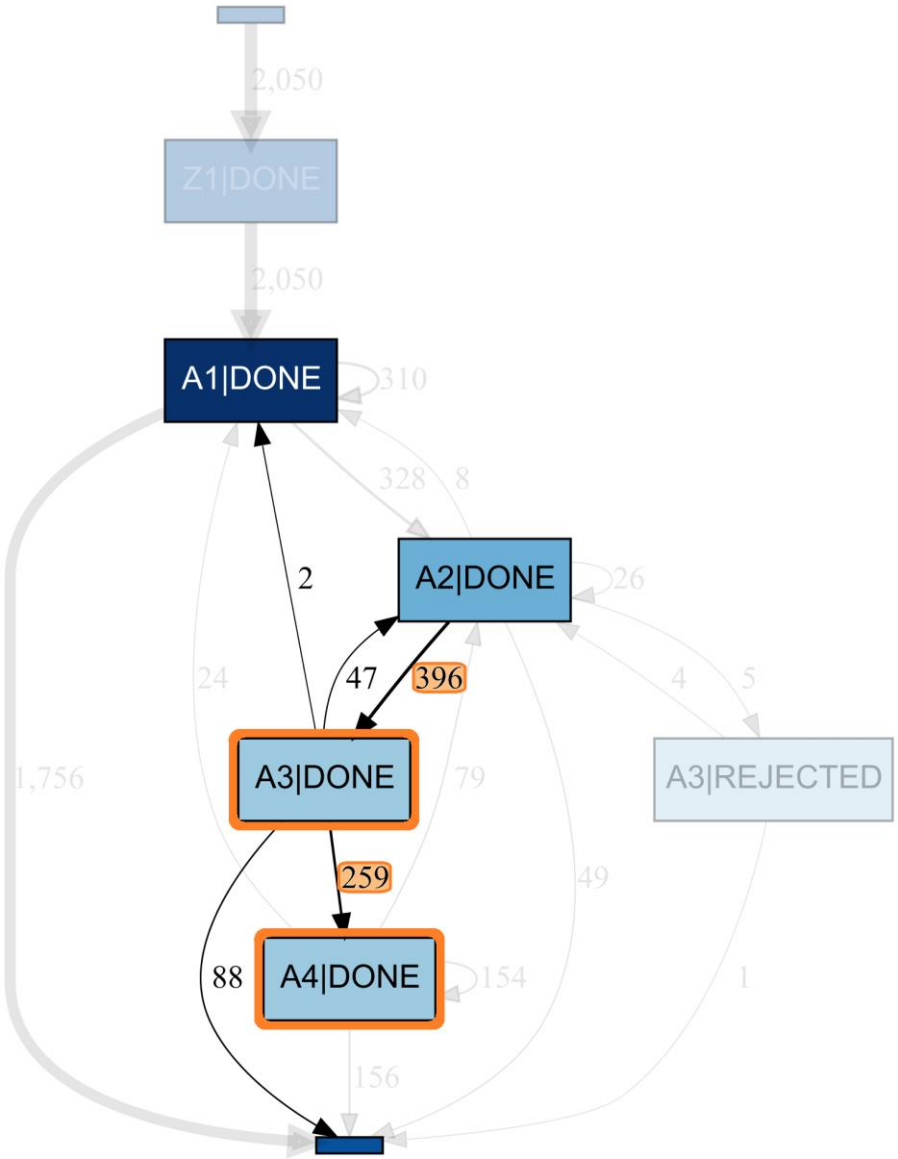

Figure 19 - Case_2 DFG - Reduce Number of Expected Activities

In this process this number should be always higher, since one A3 successful activity can generate several A4, represented in the duplicated values (n=154). One could easily overlook this issue without a complete understanding of the expected behaviour.

Finally, for the ‘Case_3’ (see Figure 20) we have selected one issue that is even less apparent, since the relation of each activity is not far from the expected, and there are no unexpected registered activities.

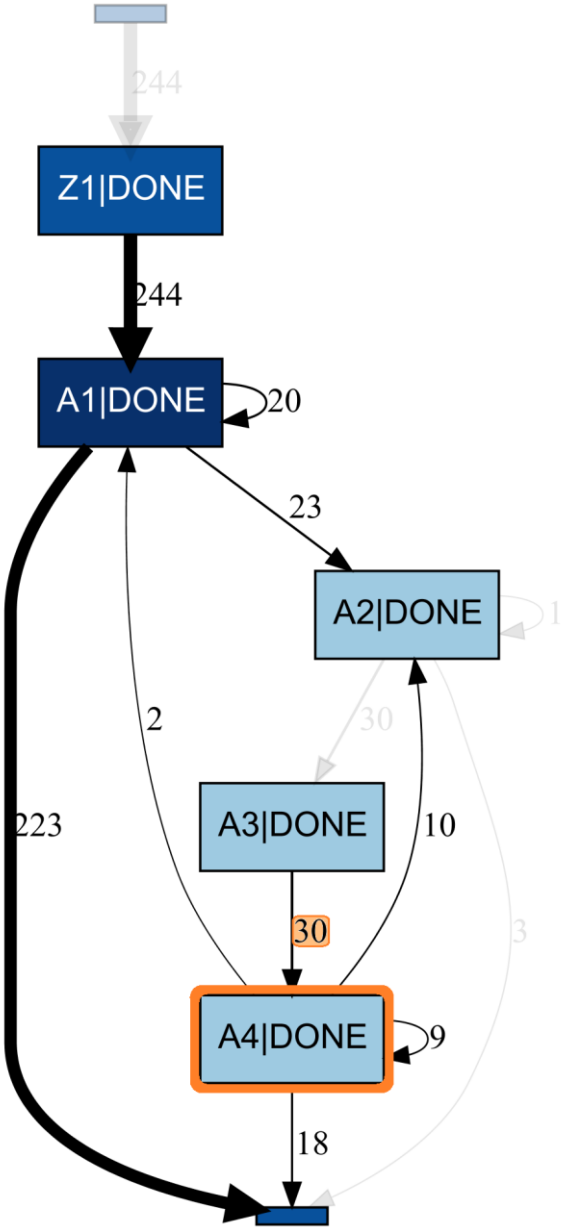

Figure 20 - Case_3 DFG - Without Clearly Identifiable Errors

This issue can only be identified by comparing the total number of activities with a comparable period, ‘Case_3.1’ (see Figure 21). As an example, we could compare the number of Z1 (n=244) and A3/A4 (n=30), with the control data set, respectively (n=1250) and (n=177), only one-fifth of the expected activities were present.

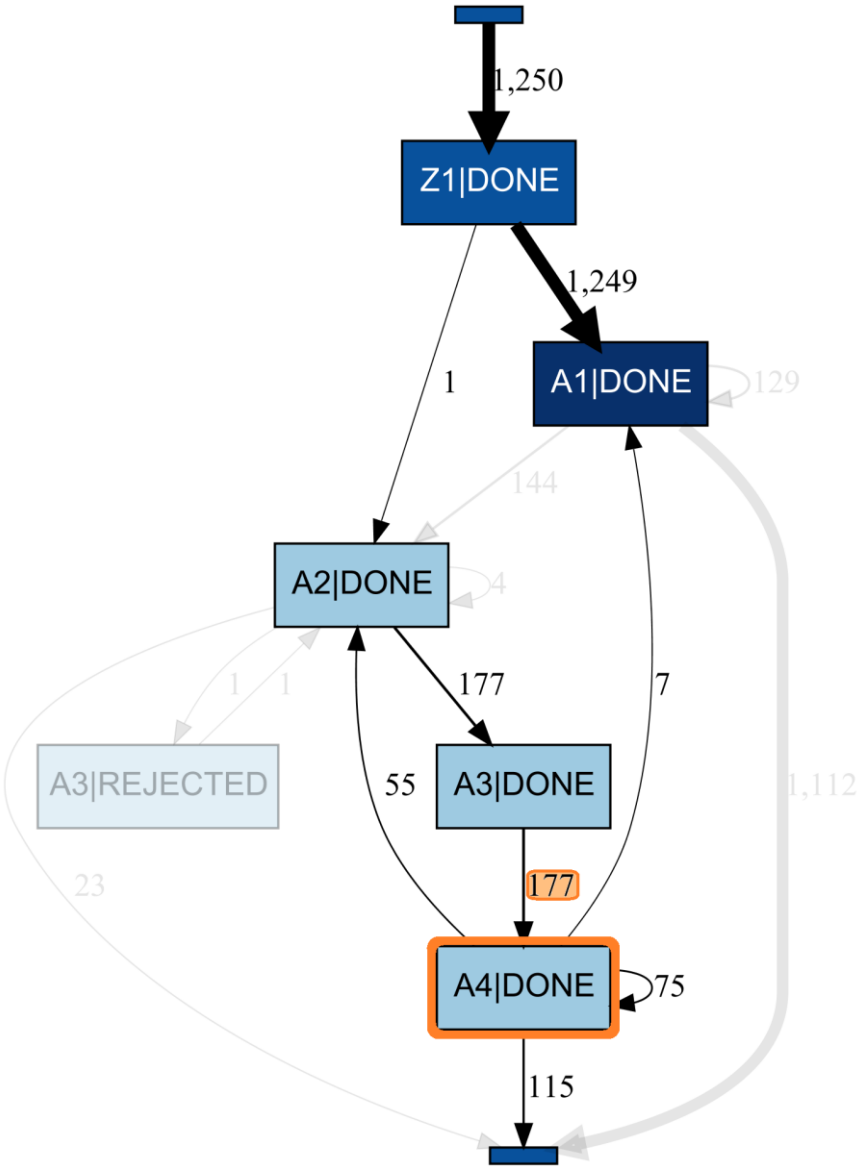

Figure 21 - Case_3.1 DFG - Expected Behaviour

This analysis concluded that there are three principal ways to identify possible errors, (1) the presence of unexpected activities, (2) the reduced proportion of certain activities and (3) the reduced number of observed activities. For the last case, a control dataset, from a comparable period must be analysed.

8.5.2 Process B

Due to the particular infrequency of 'Process B', the time period of data sets selected was increased, to guarantee a realistic representation of this process. Arguably, for

this process, the identification of most issues is simple, when using only the respective traces, any absence or reduction of the number of expected activities signifies a possible error.

Following the same principle, we have visualized the respective dataset where critical errors were subsequently identified. By observing the ‘Case_4’ DFG (see Figure 22) one immediate conclusion is evident, which is the missing final transaction, and subsequent connection to ‘Process A’ activities. And obviously, it is apparent that only a minute number of traces are identified.

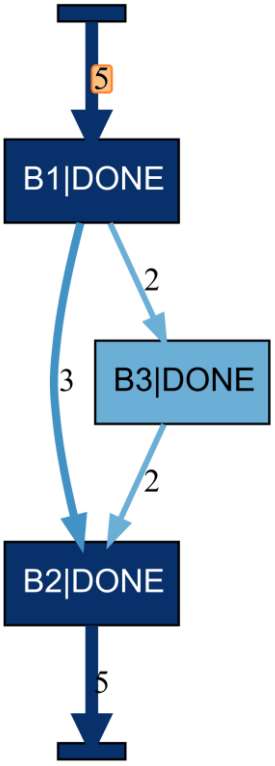

Figure 22 - Case_4 DFG - Reduced Number of Expected Activities

Additionally, it is possible to identify the extremely reduced number of activities. A dataset was extracted, ‘Case_4.1’ (see Figure 23) from a comparative and homologous period, to evidence this notable reduction in the number of expected activities.

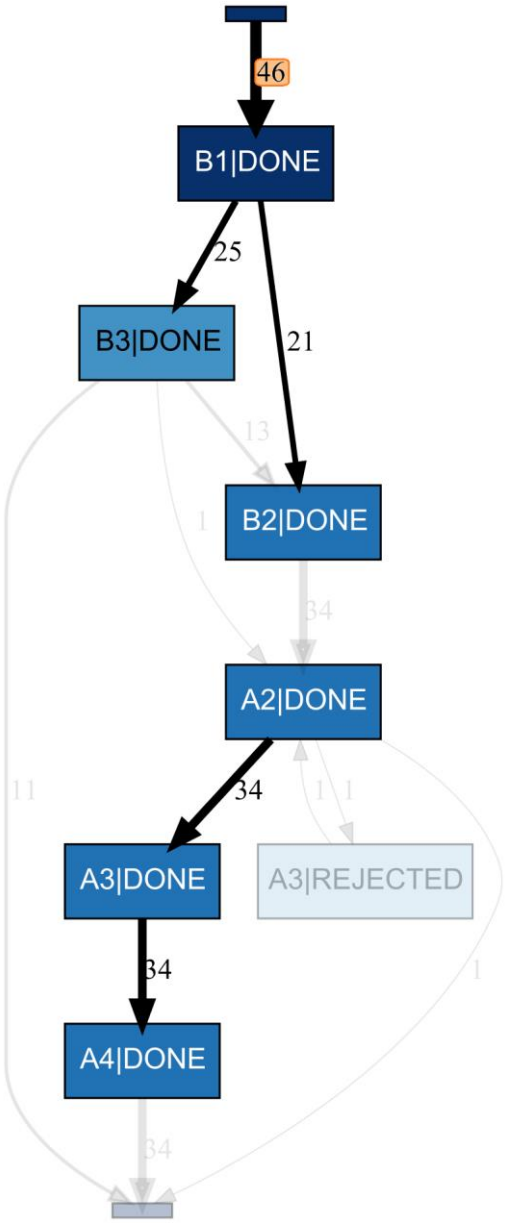

Figure 23 - Case_4.1 DFG - Expected Behaviour

In this process, due to its infrequent behaviour, it is possible to demonstrate the need to have a segmented approach to error identification. As an example, for the same selected case, we have included the traces beginning with Z1, which basically would add every 'Process A' activity from other processes. As presented in 'Case_4.2' (see Figure 24) this error, which is completely apparent in the sub process context, would be difficult to identify.

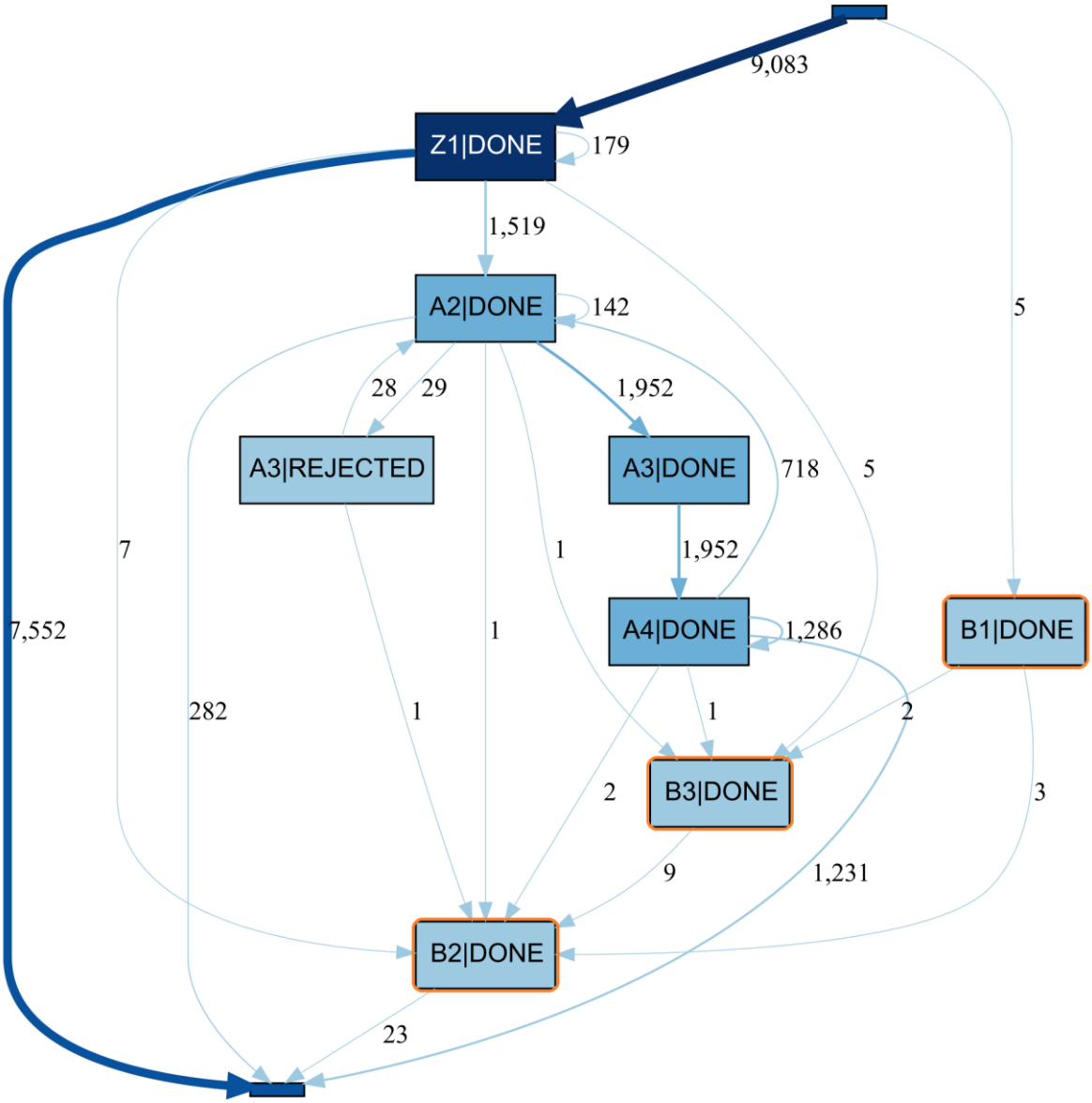

Figure 24 - Case_4.2 DFG - Model Without Necessary Segmentation

One other significant case was selected, though a less frequently identified incident, but particularly relevant since this process has a predetermined path. The following 'Case_05' exemplifies a commonly identified issue affecting only the final activity, A4, which is not present (see Figure 25)

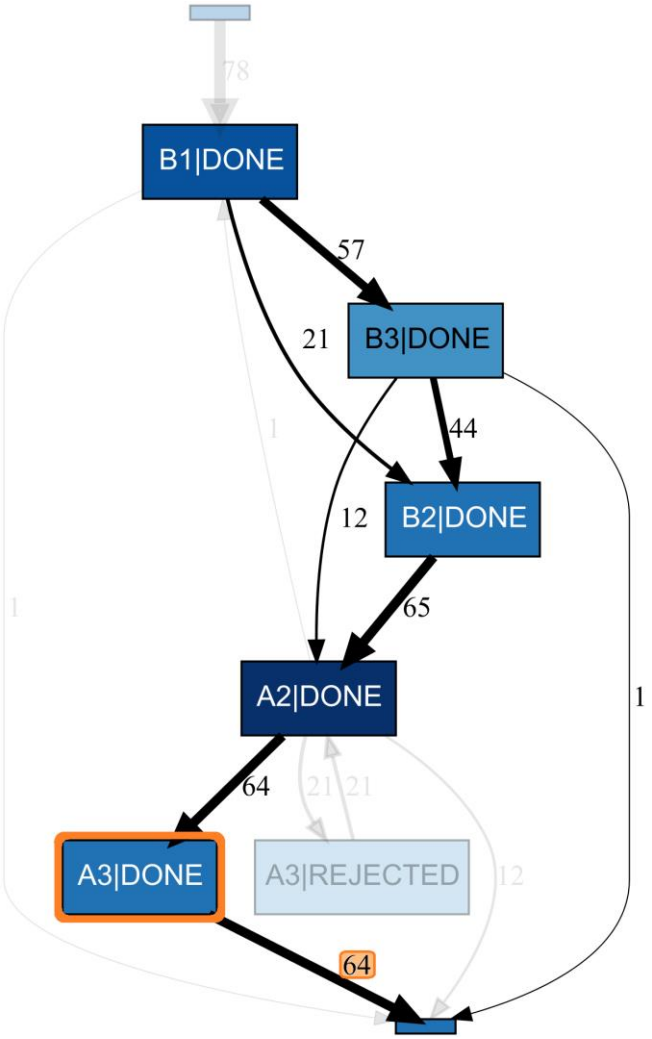

Figure 25 - Case_5 DFG - Missing Activity

Finally, for this process, we have identified one issue that only affected part of the users, impacting only the activity B3, ‘Case_06’, represented in (see Figure 26). A significant case to illustrate the practical advantage of performing a process-based error identification visual analysis. As mentioned, this activity is not mandatory, however, as evidenced in this case, the users that followed this path were not able to successfully proceed to the following expected activity.

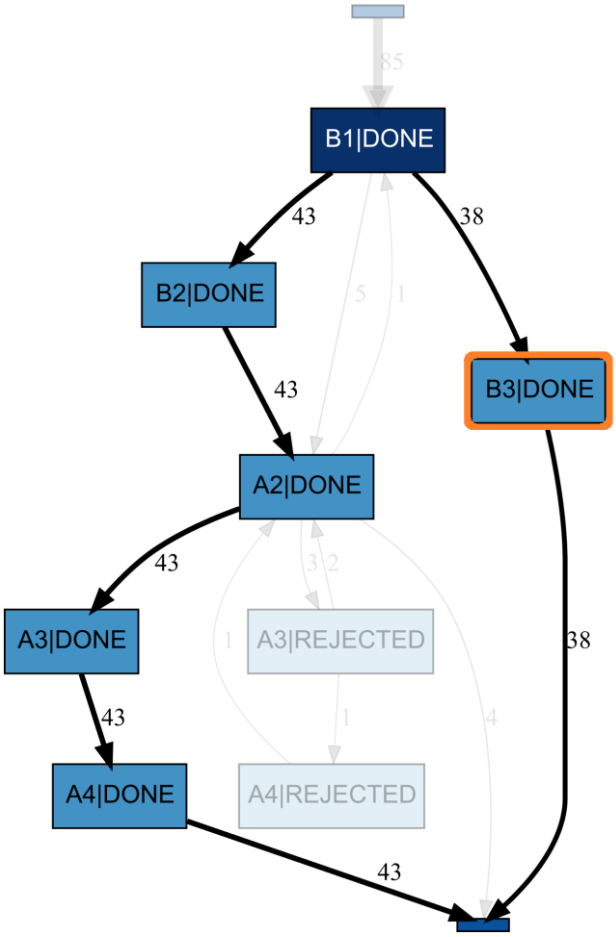

Figure 26 - Case_6 DFG - Only One Impacted Activity

In this process, each case clearly exemplifies how it is necessary to follow distinct approaches to the identification of each issue.

8.5.3 Process C

The unstructured nature of this process makes it simple to analyse, from an error detection perspective. With only two consecutive activities, the principal worrying signs are either, missing, or error activities. The selected ‘Case_07’ (see Figure 27), represents an incident where both situations are present, affecting the activity C4. The proportion of the C4-Rejected is unusually high, and the number of the C4-Done activity is abnormally low when compared to the number of registered traces.

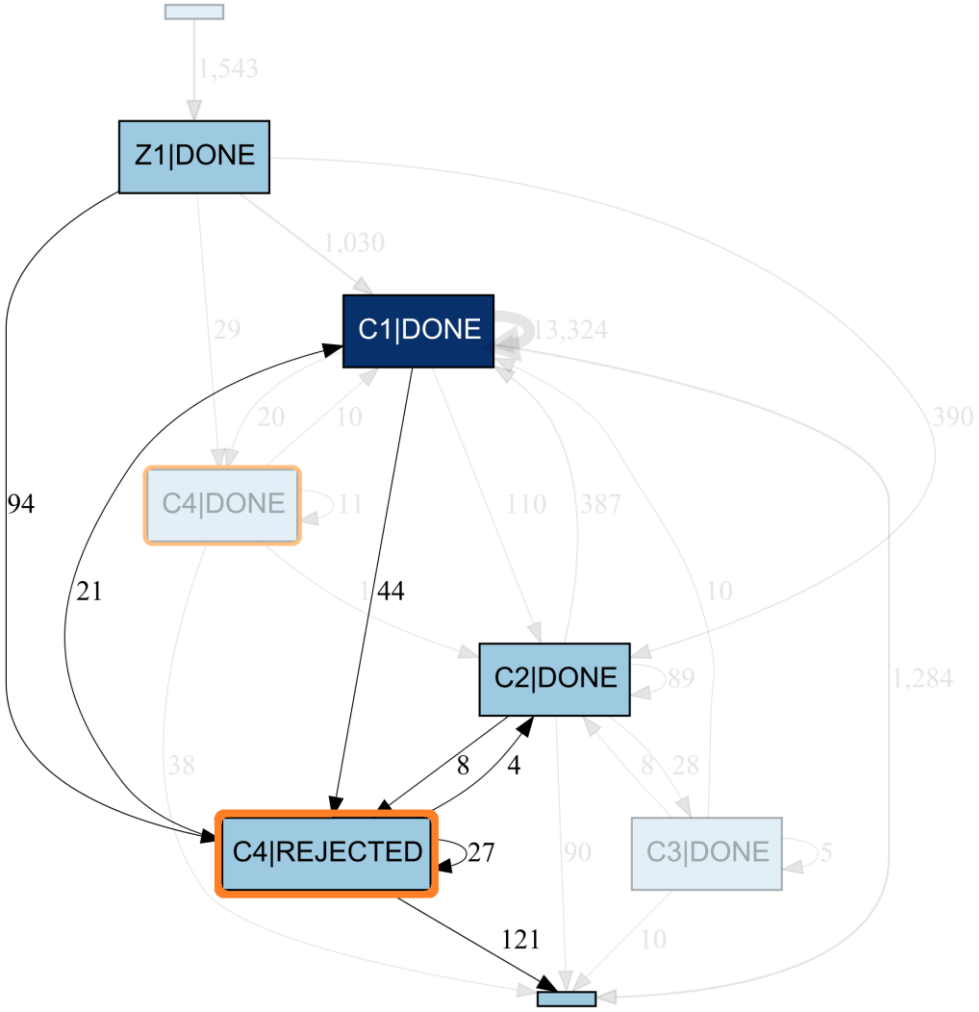

Figure 27 - Case_7 DFG - Unexpected Errors and Reduced Number of Successful Activities

Almost all the analysed historical cases affecting this process had similar characteristics; one distinct case was identified, only affecting activity C1. As mentioned, this activity has the characteristic to be registered consecutively, with an expected high frequency. In the selected 'Case_8' (see Figure 28), this activity does not exhibit the expected behaviour, only identifying a single instance per trace, implying that part of the process had an issue.

This singular process analysis, reinforces the efficiency of the proposed method, clearly and easily identifying the recorded issues, even in an unstructured process.

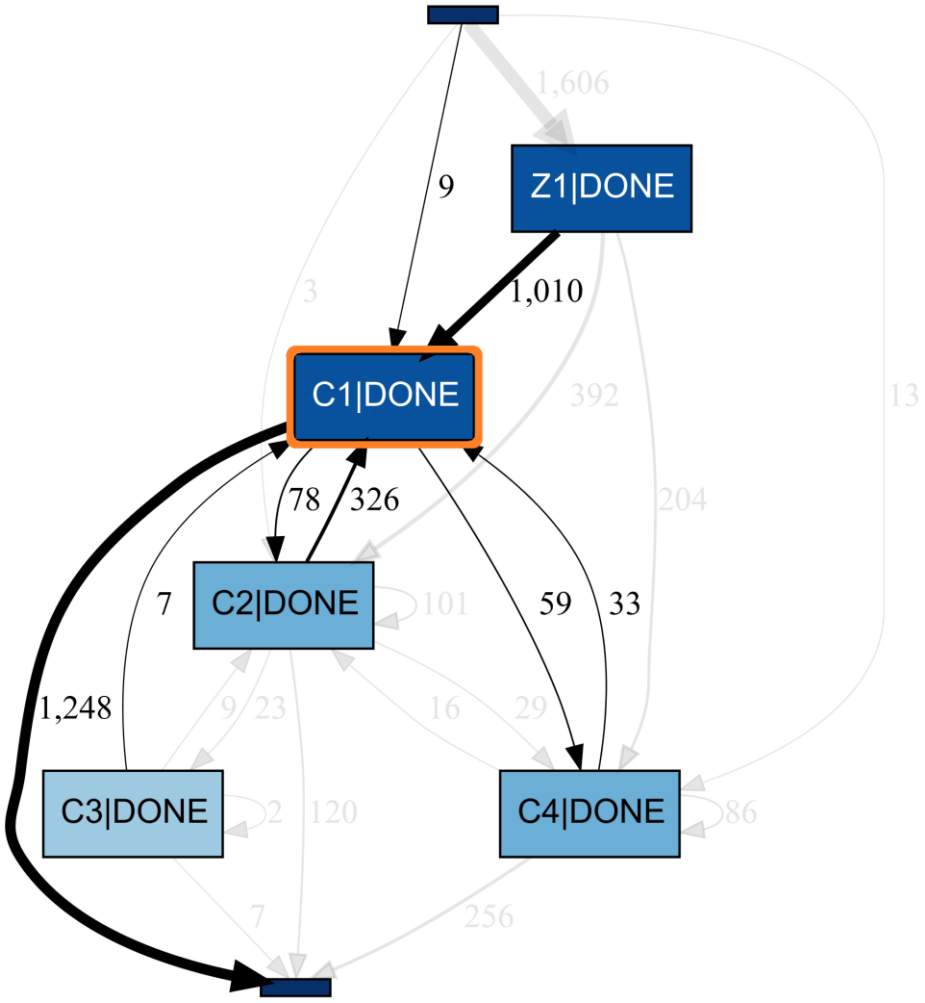

Figure 28 - Case_8 DFG - Unexpected Behaviour

Arguably for this process, it would be possible to obtain the same results analytically, however as previously mentioned, the objective was to guarantee the method's consistency.

8.5.4 Process D

As established in the Process Discovery section, 'Process D', as well as 'Process E', have four core activities. The error identification will inevitably be focused on these, particularly on the number of successful activities, number of errors, and identification of unexpected behaviour. In 'Case_9' (see Figure 29), it is possible to clearly identify an issue in the number of activities with errors, and, unexpectedly,

the number of successful operations is small. As observed, there are still successful operations, and related ‘Process A’ activities do not display any issues.

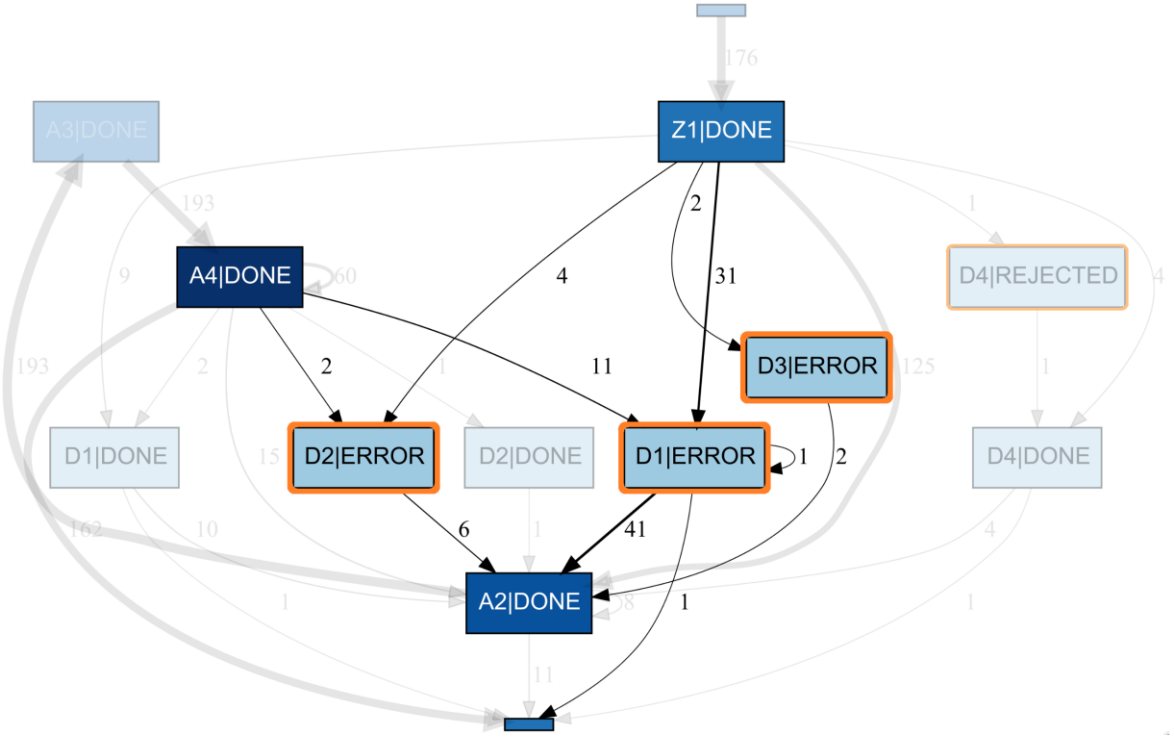

Figure 29 - Case_9 DFG - Activities with Errors

In the following example, ‘Case_10’ (see Figure 30), the error is similar in nature to the previous, however, only affects one of the activities, D1, which would undoubtedly complicate the identification without the proposed segmentation. In this case, for this activity, the number of operations exhibiting errors is comparable to the number of successful activities, nevertheless, still unusually high.

Additionally, selected ‘Case_11’, is an example of a clear and easily identifiable error (see Figure 31). In this case every D activity is either rejected or in an error state. As evidence, the issue is distinct between the activity D1, and the activities D2, D3 and D4. For D1, about half of the activities represent errors, and since it is the most frequent activity is effortlessly recognisable.

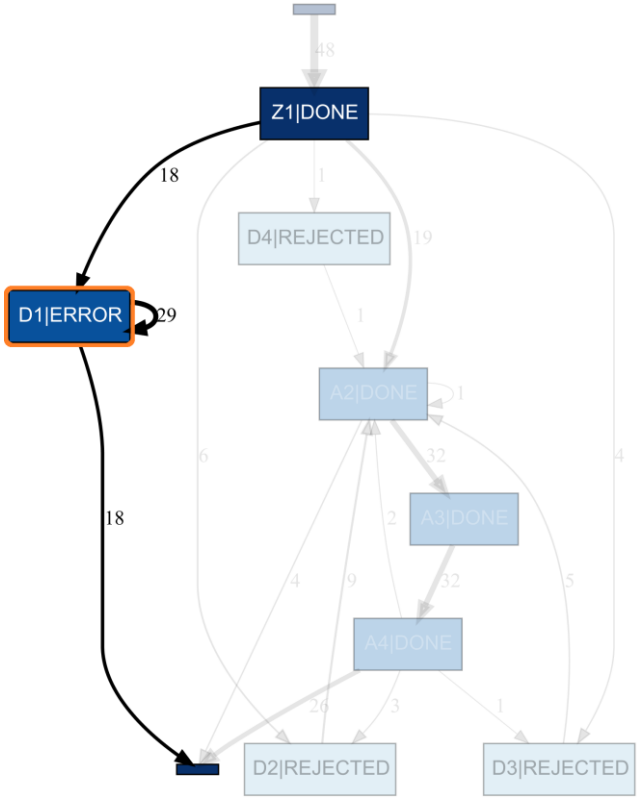

Figure 30 - Case_10 DFG - One Activity with Errors

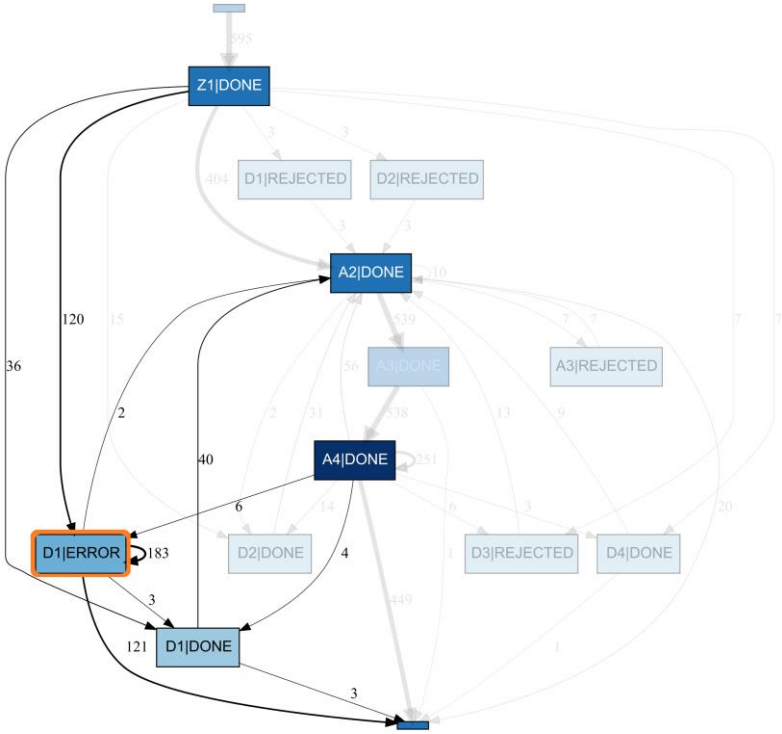

Figure 31 - Case_11 DFG - Every Activity with Errors, Evident in the most Frequent Activity

Finally, for this process, we selected the ‘Case_12’, to emphasize the perception of the singular characteristic of some identified issues. This case, (see Figure 32) represents an incident that also only affected part of the activities, not interfering with the more common D1 activity, opposite to the previous case. These situations are normally arduous to identify since the more representative activity was unaffected, reaffirming the value of this method.

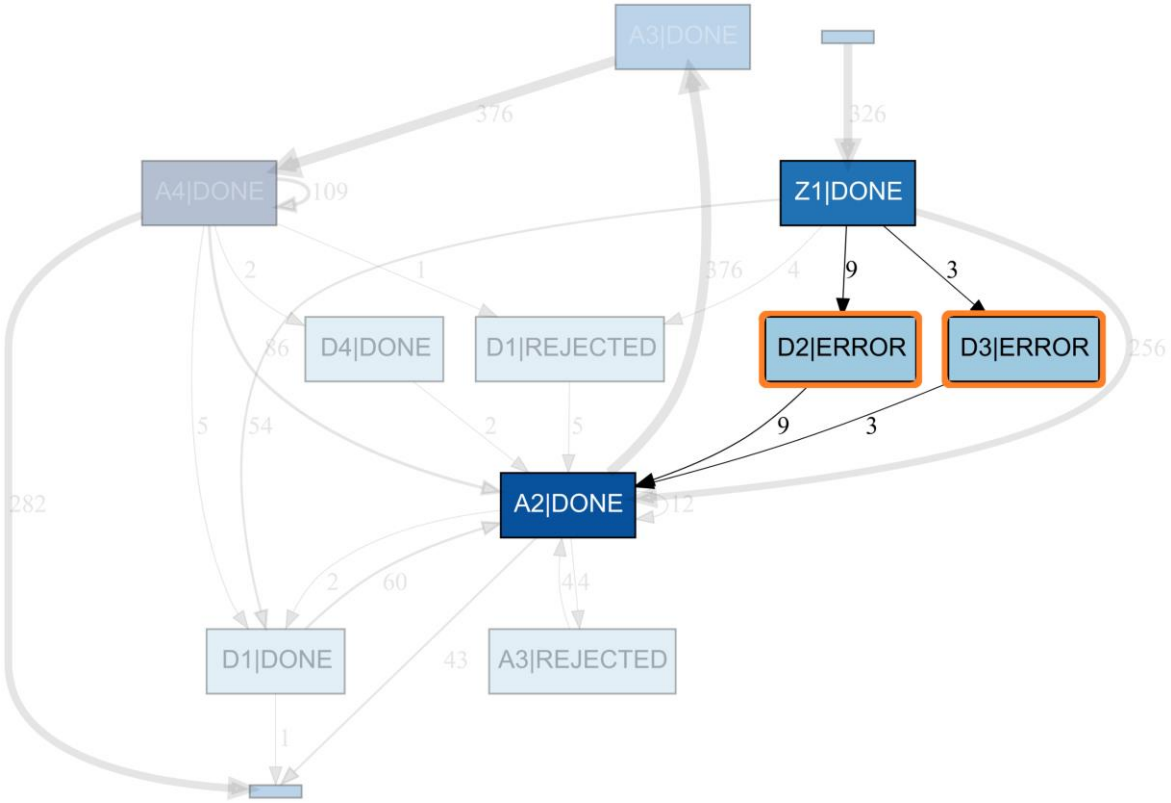

Figure 32 - Case_12 DFG - Only less frequent activities have errors.

By comparing the observed behaviour of the activities D2 and D3, between the last and the immediately previous case, it is possible to identify that there is no difference in modelled behaviour from these states, “Error” and “Rejected", and the normal, expected “Done” activities’ behaviour, besides, obviously, the respective state.

8.5.5 Process E

Unlike with process discovery, the conformance checking has few similarities with ‘Process D’. There are two principal distinctions, primarily, each ‘Process E’ activity can have particular characteristics, and secondarily, depending on this, some will

have time constraints, and are not available every day and every hour of the day, meaning that outside these intervals, rejected and error activities are expected. The analysis of historical issues considered these characteristics ignoring expected errors in the identified time frame.

In ‘Case_13’, represented in (see Figure 33), the number of done activities is within the expected numbers; however, the presence of error activities indicates there is an underlying problem that affects a percentage of the operations. Since there are several successful activities, it could be affecting either a significant percentage of users or be an intermittent application issue.

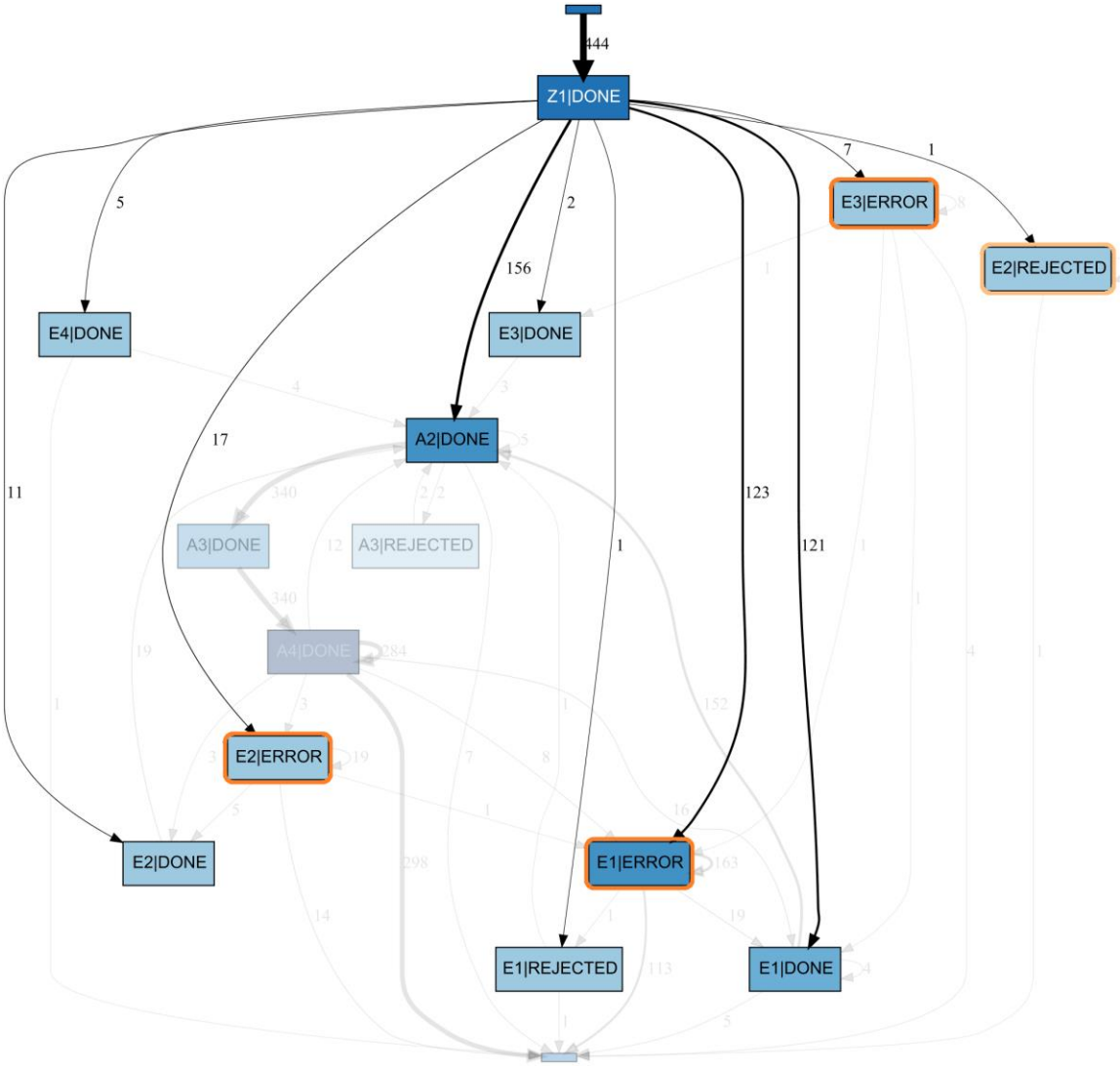

Figure 33 - Case_13 DFG - Unexpected Errors

Finally, to exemplify a more challenging issue, in ‘Case_14’ no error activities were identified (see Figure 34). The nost registered transaction seems to be concluded successfully, and the number of rejected activities are within the expected values.

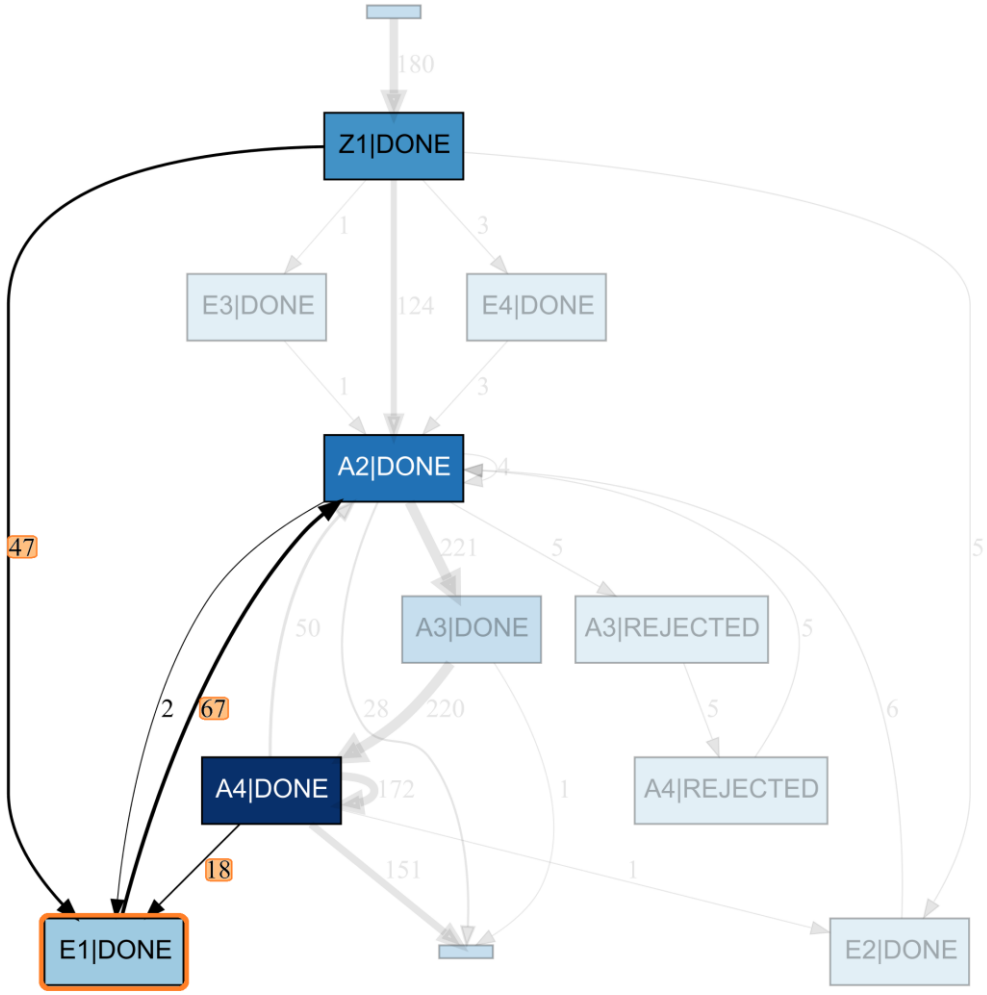

Figure 34 - Case_14 DFG - Unexpected Reduction in the Number of Activities

To identify the issue, was necessary to compare it with another model, extracted from a homologous period (see Figure 35). As evidence, when analysing the most frequent activity, E1, the observed activities (n=67) are about half the number of expected operations (n=119). This case emphasises that the absence of errors is not enough to guarantee the lack of issues.

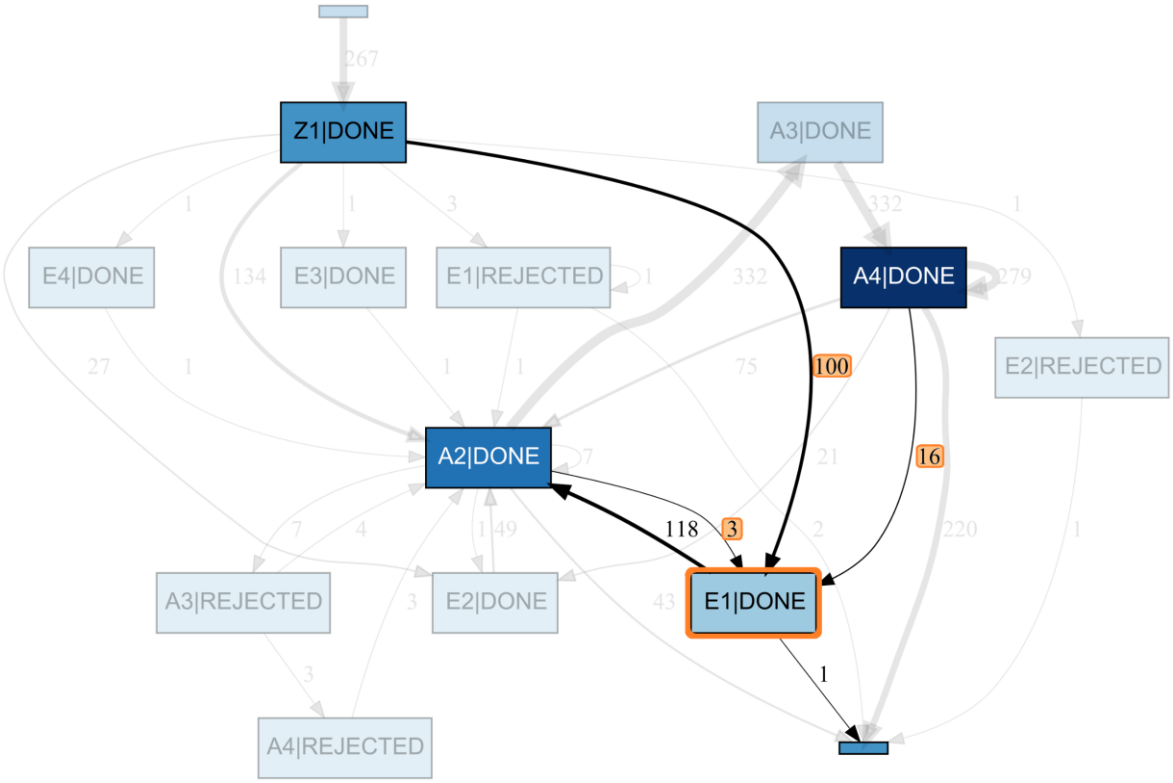

Figure 35 - Case_14.1 DFG - Expected Behaviour

8.6 Additional Insights and Analysis

Another parallel research followed was to analyse the complete process, disregarding the state, and to aggregate some of the 'Process D' and 'Process E' activities, that could be considered analogous (see Figure 36).

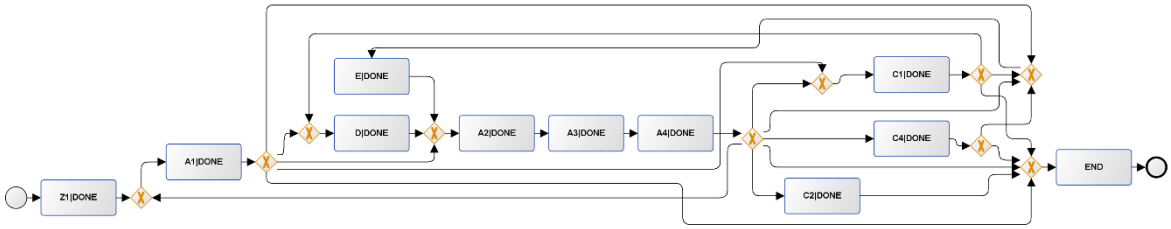

Figure 36 - BPMN Model, Agregating Process D and E Activities

This approach had the objective to properly view the entire process model, useful for statical purposes and overall performance, interesting in a business environment, however less relevant for academic research.

The analysis of the trace variants was only utilised for ‘Process B’, due to the structured nature of the process, to expose the real complexity of most processes, the following Table 6 displays the total number of traces, events, event classes and variants of each process.

Table 6. Trace Variants Analysis

	COMPLETE	PROCESS A	PROCESS B	PROCESS C	PROCESS D	PROCESS E
Traces	30.703	11.821	151	18.414	5.105	5.149
Events	213.017	47.041	962	127.650	34.859	36.859
Event Classes	31	7	7	7	14	14
Variants	1.942	169	8	837	303	346

As evidenced and observed, the difference in the number of variants from ‘Process A’, ‘Process B’ and ‘Process C’, while having the same number of classes, is an indication of the complexity, and unstructured nature, of each process. As exposed, the number of possible variants varies extremely.

There is also a high percentage of abandoned processes, partially explained by the processes’ complex nature. And also results in an astonishing number of possible different successful paths. In such a varied process, simple traces, like the abandoned ones, seem to be more statical significant than they really are. This observation is apparent in the individual process analyses, substantiated by analysing the complete event trace.

Unrelated, and despite not being a chosen strategy, there was a valuable compliance prospect for the proposed method. The ability to extract data from specific users, and activities, allows for a direct and effective method to identify possible constraints or unrestricted operations that should not otherwise be available.

Also, there is an interesting prospect to identify the causes of the constraints, not discussed since they are strongly related to the available technical structure. It is valuable to understand the respective technical architecture to rapidly identify the

possible origin. As an example, it is possible to randomly affect a fraction of the activities when backend components are parallelly installed, supporting independent activities. In these cases, it is possible to identify issues by evaluating the percentage of affected activities. As an example, if an application has four servers and twenty critical components, if either 25% or 5% of the activities are affected, it is probable that the percentage of the users or activities affected is related respectively to one server, or one component.

9 EVALUATION

The evolution of the proposed artefact was achieved by applying the framework based on the five following system dimensions proposed by Prat et al. [11], resumed (see Figure 37): the (1) goal, and objective will be evaluated regarding its efficacy, validity, and general application. The (2) environment evaluates the consistency with people, organization, and current technology. The artefact (3) structure will be assessed regarding its completeness, simplicity and level of detail, clarity, style, homomorphism, and consistency. The (4) activity will be measured concerning its completeness, consistency, accuracy, performance, and efficiency. And finally, the (5) evolution will be evaluated for robustness and learning capability.

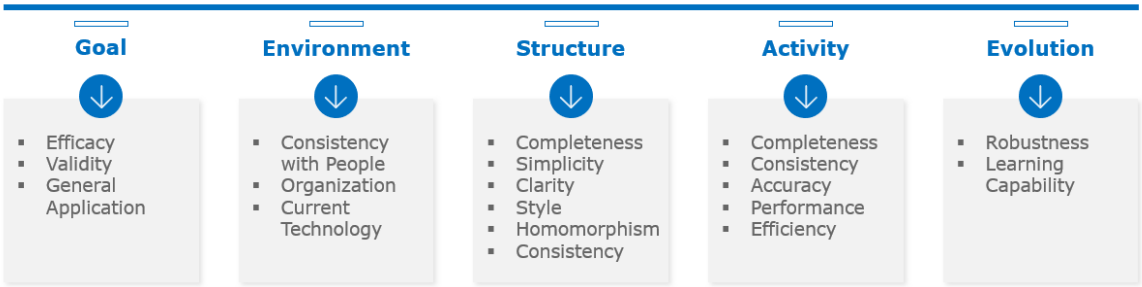

Figure 37 - Adapted from the Five Systems Dimensions proposed by Prat et al. [45]

9.1 Goal

The successful implementation of the proposed artefact, while using a use-case of a real complex application for the demonstration, confirms the validity of the proposed method. As shown, it was able to effectively identify every sub process, and every critical error in each studied case. Adjustable to the organization's needs, and technical capabilities, supporting and providing additional statistical and modelling data to the existing development and management teams.

The sample-based conformance checking proposed is demonstrably generally applicable, and effective for the two principal objectives, the discovery of the processes and the identification of every identifiable issue. The extensive approach to each process validates the method's applicability in a real and complex application.

9.2 Environment

This method is completely adaptive, suitable for any organisation context and applicable on a large scale with minimal investment and costs. The proposed method does not change the current organisation's technical structure, serving as additional data that can be utilised to efficiently identify issues and their origin. However, the method requires a significant understanding of the studied case.

The identification and visualisation of the individual, and the complete, BPMN process models, provide a unique perspective of the application users' behaviour, offering valuable insight into user experience, development, and management perspectives.

Arguably, the application selected, ProM, requires a particular level of technical proficiency that is not comparable to other alternatives [46]. Since this application was only utilized within the use-case, for the demonstration, we have disregarded the application selected. The proposed method is applicable independently of the application, with minor adjustments.

Considering all the currently available applications, some with a modern and intuitive front end, providing for a more suitable user experience, the method, independent of the application, is adjustable to every organisation.

9.3 Structure

The completeness and the simplicity were the principal factors in designing the method structure, to guarantee a wide application while maintaining an understandable and practical design. Related concepts are the level of detail and the clarity, which can be opposing the purpose was to present the data with a high level of detail while guaranteeing it was widely apprehended. The method's homomorphic and adaptable nature is granted by the possibility to adjust, remove, or add steps.

The evaluation of the method's style can be subjective. Superficially, the objective was to display a visual and appealing representation, while maintaining the intended

integral structure, consistent with the proposed objectives. The plugins utilised were described and detailed, to guarantee that the method is replicable.

Regarding the use-case, the required anonymization may also have interfered with the interpretation of the models. For that reason, the activities have been named using single letters and accordingly to their principal process.

9.4 Activity

The successful implementation of the selected use-case, with the complexity and intricacy demonstrated, had the purpose of guaranteeing the method's applicability. Conversely, a single use-case was selected, and since such complex real event logs are not normally available, the completeness of the method is difficult to access. It was undoubtedly complete for the selected use-case. The method demonstrated to be consistent for the proposed solution, proving to be reliable to accurately discover each process, and identify every issue in every used case.

The method's efficiency must be evaluated, considering its applicability. The data extraction, and especially the data quality, was not a very efficient process, requiring several manual interventions. The discovery process could also have been a more efficient process, although it may be visually overcomplicated, provided valuable insight, either for the actual discovery process, or for the subsequent analyses. The conformance checking process was arguably the more efficient process, however, it was built upon the previous effort. The fact that the preceding steps were purposefully taken, simplified this final proceeding.

Disregarding the manual effort, and concerning the utilized application's performance, we have discriminated the time consumed in each task, as shown in on Table 9 in Appendix VII – Time Analysis. Additionally, to prove that it is a scalable solution, we have experimented and evaluated the complete process, from an applicational perspective, and gathered the results for each 'case', and one 'large data set'. The priors represent the standard event log files utilised in the demonstration,

and the latter is utilised only for this purpose. To avoid disclosing any confidential data, the time perspective is in relation to 1 time unit.

Even when performing the analysis with a large data, it is possible to conclude that the computing power needed is negligible. However, every second saved in the analysis will positively affect the method's efficiency.

Considering the SQL server database utilized, the time consumed may vary with external factors. This analysis served as a statistical guideline to support the consistency of the method. As demonstrated, the total time consumed for each case never surpassed, or came close to the period in analysis, confirming the applicability of the method. The extraction process is the more significant limitation, timewise, still representing less than 1% of the analysed period duration. More importantly, the total time consumed was always a fraction of the period in analysis. The average was calculated considering comparable time periods. This evaluation validates the applicability of this method to support and monitor applications in almost real-time.

In an organisational context, each process would be evaluated with the respective determined recurrence. This configuration must consider the time necessary for the analysis, despite the demonstrated capacity to rapidly analyse each process, it still is a manual intensive method, requiring a fundamental understanding of the models to accurately identify possible issues.

9.5 Evolution

Finally, regarding the evolution prospect, specifically, the method's robustness, proved to be capable to performing the proposed objective. The possibility to automate parts of the process, and the perspective to be able to improve and adapt each task to specific needs, demonstrates an intrinsic evolutive nature.

The method is intended to be used continuously, providing statistical and historical data, and due to the level of detail, is adaptable to the necessary improvements.

10 CONCLUSION

The principal purpose of this research was to design, develop, demonstrate, and evaluate, an effective method to use Process Mining capabilities, in complex applications.

The demonstration of the method was done using real events from a large and complex mobile application. Two primary objectives were achieved, firstly the identification, categorisation and discovery of every critical activity and process, secondly the successful identification of each identified applicational issue.

The results obtained evolved through several iterations of the process, as the DSR methodology proposes, there was a constant return to the design and development phase to adapt the proposed method. Additionally, the demonstration was a progressive and continuous process that demanded adjustments and reiterations.

In this chapter, we have documented the principal contributions, limitations, and opportunities for future researchers.

10.1 Contributions

The initial contribution was the performed Systematic Literature Review, specifically the reported contributions of previous researches, and the reasoning for the identification of the research problem, supporting the subsequent implementation of the Design Science Research methodology.

By applying our proposal to an intricate and specific use-case, we have validated the adaptability of the method, while confirming the importance and necessity of understanding each activity and sub-process. The continuous use of the method, as intended, is an adaptive and evolving process.

Applying Process Mining to a real application and testing the proposed method with a significantly large data set, demonstrated the technical and practical applicability of the solution. As mentioned in the literature review, in most of the reviewed studies the number of cases, and distinct activities, available in those event logs were

negligible when compared to the selected use-case, especially considering the initially identified activities.

The demonstration has proved that critical processes would be undermined or excluded without a previous analysis of each activity and respective available, and valuable, additional data. Demonstrated that the frequency of an activity is unrelated to its importance, reinforcing the clear need to manually categorise each activity and process.

The Process Mining application selected, ProM, is principally used in an academic background, to efficiently present the results, business-oriented visualizations were selected. The objective was to be scientifically minded, while being practical and applicable.

Finally, we have summarized the principal contributions, related to the defined Research Questions: (1) *How is Process Mining applied as an applicational support tool?* Providing an additional method to employ the available techniques in complex event logs. The combination of the data gathered from the discovery and conformance checking, to identify sub-processes, their relevance and the number of incomplete processes, while additional supporting several statistical information. Using Process Mining is therefore possible to display statistical information and complex analysis with a non-technical output; (2) *What are the algorithms employed?* Reinforcing the capabilities of the utilised algorithms, specifically the 'Heuristic Miner'; (3) *What are the Process Mining methods used to effectively identify application's malfunction and errors?* Demonstrating that for such complex processes, is necessary to handle data quality issues, and categorize and segment subprocesses to successfully identify unexpected behaviour. Without prearranged logs, clean of insignificant data, false positives will increase, negatively affecting the model precision; (4) *What are the possible limitations of Process Mining as an application support tool?* Confirming that the computing power is not an issue, even for complex and large data sets, however, unlike most Process Mining use-cases, the

conformance checking method heavily relies on the manual analysis of the observed behaviour.

10.2 Limitations

The anonymization imposed, which from the organization's perspective is completely understandable and prudent, obfuscate the significance of some of the insights.

Regarding the algorithms, plugins, and visualizations selected, it was not possible to document and report every experiment and result, since it would be an overwhelming effort considering the number of available alternatives, each with particular requirements.

Considering the followed approaches, it would be possible to extend the data analysed. For example, the time element is obviously present, however as explained was not considered for performance analysis. A possible approach would be to compare certain variants' time durations to identify possible constraints.

Additionally, the identification of the possible causes for the identified issues was properly not explored in this research, was considered too complex, and lacked perceived value due to the required confidentiality.

Moreover, and due to the level of interactivity of the adopted visualizations, there is a risk of influencing the results, unintentionally, by individual preferences or biases.

Finally, regarding the SLR, due to time constraints, only utilized one digital library to conduct the search, and only free articles with the full text in PDF were considered. Although the screening was done carefully, it is possible that some contributions have been overlook, and perhaps some articles may be incorrectly excluded.

10.3 Future Work

One of the more promising future research prospects is to evaluate the performance of the application by applying the time perspective, assessing the time lapsed in the

process, specifically between each activity. Significant variations in this metric will probably indicate performance issues. The complexity of this type of analysis is to differentiate applicational incidents from expected user interference or delay.

Additionally, the case perspective would probably be a valuable research possibility, especially, if additional case data was included. For the selected use-case several interesting prospects were identified, excluded principally due to the sensitive nature of most of the data, opposing the anonymization necessity. Also, the process enhancement capabilities were not adequately researched, and have the possibility to offer a possible research prospect.

From a more technical perspective, identifying the origin of the errors from the observed behaviour is an inciting possibility. In principle, the cause of a particular issue should be obtainable directly and automatically.

Regarding the discussed limitation concerning the algorithms and plugins selected, the research, experimentation, detailed characteristics, and respective evaluation would be a rewarding and valuable effort for future researchers.

The Enhancement plugins were also not properly explored, these plugins are particularly useful for optimizing processes, and identifying and correcting bottlenecks, however, were not included for absence of identified potential value in such complex and unstructured process.

Finally, the automation of the entire process would be easily achievable, this implementation would be a topic worth experimenting and researching. And, perhaps the more promising, and challenging, research opportunity, would be to develop a new algorithm, or combine other technical capabilities, like machine learning and artificial intelligence, to properly identify and categorise activities and sub processes, in large and complex models.

REFERENCES

[1] W. van der Aalst, Process Mining: Data Science in Action, New York: Springer, 2016.

[2] W. van der Aalst, “Foundations of Process Discovery,” *Business Information Processing,* vol. 448, pp. 37-75, pp. 201, 2022.

[3] A. Hevner and S. Chatterjee, Design Research in Information Systems: Theory and Practice, New York: Springer, 2010.

[4] K. Peffers, T. Tuunanen, M. Rothenberger and S. Chatterjee, “A Design Science Research Methodology for Information Systems Research,” *Management Information Systems,* vol. 24, no. 3, pp. 45–77, 2008.

[5] K. Peffers, T. Tuunanen, C. Gengler, M. Rossi, W. Hui, V. Virtanen and J. Bragge, “The Design Science Research Process: A Model for Producing and Presenting Information Systems Research,” in *1st International Conference, Desrist Proceedings*, 2006.

[6] A. Hevner, S. March, J. Park and S. Ram, “Design Science in Information Systems Research,” *MIS Quarterly,* vol. 28, no. 1, pp. 75-105, 2004.

[7] A. Hevner, “A Three Cycle View of Design Science Research,” *Scandinavian Journal of Information Systems,* vol. 19, no. 2, pp. 87-92, 2007.

[8] J. Pries-Heje, R. Baskerville and J. Venable, “Strategies for Design Science Research Evaluation,” in *ECIS Proceedings 87 Association for Information System*, 2008.

[9] J. Venable, J. Pries-Heje and R. Baskerville, “A Comprehensive Framework for Evaluation in Design Science Research Design Science Research in Information Systems,” *Advances in Theory and Practice,* vol. 7286, pp. 423-438, 2012.

[10] K. Peffers, M. Rothenberger, T. Tuunanen and R. Vaezi, “Design Science Research Evaluation,” *DESRIST Lecture Notes in Computer Science,* vol. 7286, 2012.

[11] N. Prat, I. Comyn-Wattiau and J. Akoka, “Artefact Evaluation in Information Systems Design-Science Research,” in *A Holistic View PACIS Proceedings 23 Association for Information System*, 2014.

[12] B. Kitchenham, “Procedures for Performing Systematic Reviews Joint Technical Report,” Keele University UK Technical Report and Empirical Software Engineering, Australia, 2004.

[13] B. Kitchenham, “Guidelines for Performing Systematic Literature Reviews in Software Engineering Joint Technical Report,” Keele University UK Technical Report and Department of Computer Science of University of Durham UK, 2007.

[14] W. van der Aalst, V. Rubin, H. Verbeek, B. van Dongen, E. Kindler and C. Günther, “Process Mining: A Two-Step Approach to Balance between Underfitting and Overfitting Softw Syst Model,” *Springer,* vol. 9, pp. 87–111, 2010.

[15] W. van der Aalst, H. Reijers and M. Song, “Discovering Social Networks from Event Logs. Computer Supported Cooperative Work,” *Springer,* vol. 14, pp. 549–593, 2005.

[16] W. van der Aalst, “Process Discovery from Event Data: Relating Models and Logs Through Abstractions,” *Wiley Periodicals,* vol. 1244, no. Data Mining Knowl Discov, 2017.

[17] A. Polyvyanyy and A. Kalenkova, “Conformance Checking of Partially Matching Processes: An Entropy-Based Approach,” *Information Systems,* vol. 106, 2022.

[18] C. Liu, “Discovery and Quality Evaluation of Software Component Behavioral Models,” *Transactions on Automation Science and Engineering,* vol. 18, 2020.

[19] T. Huang, B. Xu, H. Cai, J. Du, K. Chao and C. Huang, “A Fog Computing Based Concept Drift Adaptive Process Mining Framework for Mobile APPs,” *Future Generation Computer System,* vol. 89, pp. 670-684, 2018.

[20] G. Özdağoğlu and E. Kavuncubaşı, “Monitoring the Software Bug-Fixing Process Through the Process Mining Approach,” *Software Evolution and Process,* vol. 2162, 2019.

[21] D. Myers, S. Suriadi, K. Radke and E. Foo, “Anomaly Detection for Industrial Control Systems Using Process Mining,” *Computers & Security,* vol. 78, pp. 103–125, 2018.

[22] P. Zerbino, D. Aloini, R. Dulmin and V. Mininno, “Process-Mining-Enabled Audit of Information Systems: Methodology and an Application,” *Expert Systems with Applications,* vol. 110, 2018.

[23] A. Valle, E. Santos and E. Loures, “Applying Process Mining Techniques in Software Process Appraisals,” *Information and Software Technology,* vol. 87, pp. 19–31, 2017.

[24] A. Stefanini, D. Aloini, E. Benevento, R. Dulmin and V. Mininno, “A process Mining Methodology for Modeling Unstructured Processes,” *Knowl Process Manager,* p. 1–17, 2020.

[25] A. Kalenkova, W. van der Aalst, I. Lomazova and V. Rubin, “Process Mining Using BPMN: Relating Event Logs and Process Models,” *Softw Syst Model,* vol. 16, p. 1019–1048, 2017.

[26] M. Fardbastani, F. Allahdadi and M. Sharifi, “Business Process Monitoring via Decentralized Complex Event Processing Enterprise Information Systems,” pp. 1751-1775, 2018.

[27] C. Garcia, A. Meincheim, E. Junior, M. Dallagassa, D. Sato, D. Carvalho, E. Santos and E. Scalabrin, “Process Mining Techniques and Applications: A Systematic Mapping Study Expert Systems with Applications,” vol. 133, p. 260–295, 2019.

[28] W. van der Aalst, “Business Alignment: using Process Mining as a Tool for Delta Analysis and Conformance Testing,” *Springer,* vol. 10, no. Requirements Eng, p. 198–211, 2005.

[29] L. Raichelson, P. Soffer and E. Verbeek, “Merging Event Logs: Combining Granularity Levels for Process Flow Analysis Information Systems,” p. 211–227, 2017.

[30] J. Claes and G. Poels, “Merging Event Logs for Process Mining: A Rule Based Merging Method and Rule Suggestion Algorithm Expert Systems with Applications,” vol. 41, p. 7291–7306, 2014.

[31] D. Repta, M. Moisescu, I. Sacala, I. Dumitrache and A. Stanescu, “Towards the Development of Semantically Enabled Flexible Process Monitoring Systems International Journal of Computer Integrated Manufacturing,” vol. 30, no. 1, p. 96–108, 2017.

[32] W. van der Aalst and H. Verbeek, “Process Discovery and Conformance Checking using Passages Fundamenta Informaticae,” *IOS Press,* vol. 131, p. 103–138, 2014.

[33] M. Bauer, H. van der Aa and M. Weidlich, “Sampling and Approximation Techniques for Efficient Process Conformance Checking,” *Information Systems,* vol. 104, 2022.

[34] G. Janssenswillen, N. Donders, T. Jouck and B. Depaire, “A Comparative Study of Existing Quality Measures for Process Discovery Information Systems,” vol. 71, p. 1–15, 2017.

[35] A. Rozinat and W. van der Aalst, “Conformance Checking of Processes Based on Monitoring Real Behavior Information Systems,” vol. 33, p. 64–95, 2008.

[36] Y. Djenouri, A. Belhadi and P. Fournier-Viger, “Extracting Useful Knowledge from Event Logs: A Frequent Itemset Mining Approach Knowledge-Based Systems,” vol. 139, p. 132–148, 2018.

[37] C. Li, J. Ge, L. Huang, H. Hu, W. B., H. Hu and B. Luo, “Software Cybernetics in BPM: Modeling Software Behavior as Feedback for Evolution by a Novel Discovery Method Based on Augmented Event Logs. The Journal of Systems and Software,” vol. 124, p. 260–273, 2017.

[38] S. Goedertier, D. Martens, J. Vanthienen and B. Baesens, “Robust Process Discovery with Artificial Negative Events,” *Machine Learning Research,* vol. 10, pp. 1305-1340, 2009.

[39] J. Samalikova, R. Kusters, J. Trienekens and A. Weijters, “Process Mining Support for Capability Maturity Model Integration-Based Software Process Assessment, in Principle and in Practice,” *Journal of Software: Evolution and Process,* no. John Wiley & Sons, Ltd, 2014.

[40] F. Mannhardt, N. Tax and D. Schunselaar, “XESLite Managing Large XES Event Logs in ProM,” *BPM Center Report,* vol. 1604, 2016.

[41] J. Claes and G. Poels, “Process Mining and the ProM Framework: An Exploratory Survey,” *Business Process Management,* vol. 132, 2013.

[42] A. Rozinat, A. Medeiros, C. Günther, A. Weijters and W. van der Aalst, “Towards an Evaluation Framework for Process Mining Algorithms,” vol. 224, 2007.

[43] S. Cnuddea, J. Claes and G. Poels, “Improving the Quality of the Heuristics Miner in ProM 6.2,” *Department of Business Informatics and Operations Research,* 2014.

[44] W. van der Aalst, B. van Dongen, C. Günther, R. Mans, A. Medeiros, A. Rozinat, V. Rubin, M. Song, H. Verbeek and A. Weijters, “ProM 4.0: Comprehensive Support for Real Process,” *CATPN,* vol. 4546, 2007.

[45] F. d. L. M. &. R. H. A. Mannhardt, “Heuristic Mining Revamped : an Interactive, Data-Aware, and Conformance-Aware Miner,” in *BPM Demo Track and BPM Dissertation Award, co-located with 15th International Conference on Business Process Management*, Barcelona, Spain, 2017.

[46] W. van der Aalst, B. van Dongen, C. Günther, A. Rozinat, H. Verbeek and A. Weijters, “ProM : The Process Mining Toolkit,” in *Proceedings of the BPM Demonstration Track*, 2009.

APPENDIX I – SLR SEARCH RESULTS

Table 7. Keywords identified in the SLR Search Results

	Process Mining	Business Process Management	Process Discovery	Petri Nets	Conformance Checking	Data Analysis	Event Logs	Security Auditing	Software process Assessment	Workflow Management	Distributed Computing	Component Behav-ioural Model
P1	X				X							
P2			X	X						X		
P3	X	X						X				
P4					X	X						
P5	X	X						X				
P6	X							X				
P7	X											
P8	X	X	X									
P9	X	X	X	X	X							
P10	X					X			X			
P11			X	X								
P12	X											
P13	X								X			
P15	X						X					
P16			X	X								
P17	X			X								
P18	X	X			X							X
P19	X	X		X		X						X
P20	X					X						
P22	X											
P23		X									X	
P24	X	X	X		X						X	
P25												
P26	X	X					X					
P28	X	X					X					
P30						X	X					

APPENDIX II – SLR QUALITY ASSESSMENT

Table 8. Quality Assessment Results

		QA1	QA2	QA3	QA4	QA5	SCORE
P1	[17]	3	3	2	2	3	13
P2	[18]	3	1	1	2	2	9
P3	[20]	3	2	2	2	2	11
P4	[33]	3	3	3	2	2	13
P5	[22]	3	2	3	3	2	13
P6	[21]	3	1	2	1	2	9
P7	[19]	3	2	2	3	2	12
P8	[16]	2	2	2	1	2	9
P9	[25]	3	3	2	1	2	11
P10	[23]	2	2	1	2	2	9
P11	[37]	3	2	1	2	2	10
P12	[31]	3	1	1	2	2	9
P13	[39]	2	1	3	2	3	11
P14	-	2	2	1	1	1	7
P15	[14]	3	1	3	2	2	11
P16	[38]	3	1	2	2	2	10
P17	[35]	2	3	2	2	2	11
P18	[28]	2	3	2	1	3	11
P19	[15]	2	2	2	2	2	10
P20	[27]	3	1	2	1	2	9
P21	-	2	1	1	2	1	7
P22	[24]	2	2	2	3	2	11
P23	[26]	2	1	3	3	2	11
P24	[32]	3	3	2	1	2	11
P25	[34]	2	2	2	1	2	9
P26	[36]	3	1	2	1	2	9
P27	-	2	1	1	1	1	6
P28	[30]	3	1	2	2	2	10
P29	-	-	-	-	-	-	-
P30	[29]	3	1	2	2	2	10
P31	-	2	1	1	2	1	7

APPENDIX III – ACTIVITIES STATISTICAL ANALYSIS

Table 9. Total Available Activities Statistical Analysis

#	label	rf	crf	#	label	rf	crf	#	label	rf	crf	#	label	rf	crf
1	--	27,80993%	27,80993%	34	E3	0,02592%	99,69767%	67	C2	0,00096%	99,98999%	100	--	0,00006%	99,99939%
2	C1	16,50086%	44,31079%	35	B3	0,02333%	99,72100%	68	D3	0,00093%	99,99091%	101	--	0,00005%	99,99945%
3	A1	8,92872%	53,23951%	36	--	0,02143%	99,74243%	69	--	0,00089%	99,99180%	102	--	0,00004%	99,99949%
4	--	8,18551%	61,42502%	37	--	0,02102%	99,76345%	70	--	0,00084%	99,99265%	103	--	0,00004%	99,99953%
5	--	8,13441%	69,55943%	38	--	0,01999%	99,78344%	71	--	0,00062%	99,99327%	104	A1	0,00004%	99,99957%
6	--	8,13324%	77,69268%	39	--	0,01835%	99,80180%	72	A2	0,00052%	99,99379%	105	--	0,00003%	99,99960%
7	Z1	8,13222%	85,82490%	40	A3	0,01601%	99,81780%	73	--	0,00048%	99,99427%	106	--	0,00003%	99,99963%
8	A4	2,11177%	87,93667%	41	E4	0,01564%	99,83344%	74	--	0,00043%	99,99470%	107	--	0,00003%	99,99966%
9	C2	1,96712%	89,90378%	42	E1	0,01415%	99,84759%	75	--	0,00043%	99,99513%	108	--	0,00002%	99,99968%
10	--	1,86260%	91,76638%	43	C5	0,01388%	99,86147%	76	--	0,00042%	99,99555%	109	--	0,00002%	99,99970%
11	A2	1,59262%	93,35901%	44	E3	0,01152%	99,87299%	77	--	0,00041%	99,99596%	110	--	0,00002%	99,99972%
12	A3	1,52369%	94,88270%	45	--	0,01088%	99,88387%	78	--	0,00039%	99,99635%	111	--	0,00002%	99,99975%
13	--	1,00193%	95,88463%	46	--	0,01041%	99,89428%	79	--	0,00037%	99,99672%	112	--	0,00002%	99,99977%
14	C4	0,91163%	96,79626%	47	E2	0,01029%	99,90457%	80	D4	0,00030%	99,99702%	113	Z1	0,00002%	99,99979%
15	E1	0,70506%	97,50132%	48	D1	0,01019%	99,91476%	81	--	0,00029%	99,99731%	114	--	0,00002%	99,99981%
16	--	0,61179%	98,11311%	49	--	0,00918%	99,92393%	82	--	0,00028%	99,99759%	115	--	0,00002%	99,99984%
17	D1	0,28539%	98,39850%	50	D3	0,00902%	99,93295%	83	E4	0,00015%	99,99774%	116	B1	0,00002%	99,99986%
18	--	0,17873%	98,57723%	51	--	0,00842%	99,94137%	84	--	0,00014%	99,99789%	117	--	0,00001%	99,99987%
19	--	0,14592%	98,72314%	52	--	0,00608%	99,94745%	85	--	0,00013%	99,99802%	118	--	0,00001%	99,99989%
20	C3	0,11392%	98,83706%	53	--	0,00578%	99,95323%	86	--	0,00013%	99,99815%	119	--	0,00001%	99,99990%
21	--	0,10289%	98,93995%	54	D2	0,00560%	99,95882%	87	--	0,00012%	99,99827%	120	--	0,00001%	99,99992%
22	--	0,10275%	99,04269%	55	--	0,00516%	99,96399%	88	--	0,00011%	99,99838%	121	D3	0,00001%	99,99993%
23	E2	0,09702%	99,13972%	56	--	0,00492%	99,96891%	89	--	0,00011%	99,99849%	122	--	0,00001%	99,99995%
24	E1	0,09660%	99,23631%	57	C1	0,00413%	99,97305%	90	E4	0,00011%	99,99860%	123	--	0,00001%	99,99996%
25	--	0,09098%	99,32729%	58	C4	0,00321%	99,97625%	91	--	0,00010%	99,99870%	124	--	0,00001%	99,99996%
26	D2	0,07273%	99,40003%	59	--	0,00250%	99,97875%	92	--	0,00009%	99,99879%	125	--	0,00001%	99,99997%
27	--	0,07105%	99,47107%	60	--	0,00203%	99,98078%	93	--	0,00009%	99,99888%	126	--	0,00001%	99,99998%
28	--	0,04938%	99,52045%	61	E3	0,00182%	99,98260%	94	--	0,00008%	99,99896%	127	--	0,00001%	99,99999%
29	B1	0,03632%	99,55677%	62	A4	0,00155%	99,98414%	95	--	0,00008%	99,99904%	128	--	0,00001%	99,99999%
30	B2	0,03115%	99,58793%	63	--	0,00147%	99,98561%	96	D4	0,00007%	99,99912%	129	--	0,00001%	99,99999%
31	D4	0,03039%	99,61832%	64	--	0,00139%	99,98700%	97	--	0,00007%	99,99919%	130	D2	0,00001%	100,00000%
32	D1	0,02706%	99,64538%	65	--	0,00103%	99,98804%	98	--	0,00007%	99,99927%				
33	--	0,02636%	99,67174%	66	E2	0,00099%	99,98903%	99	--	0,00007%	99,99933%				

label: Included Activities are represented by the process' letter.
rf: Relative Frequency
crf: Cumulative Relative Frequency

APPENDIX IV – PROCESS DISCOVERY – PROCESS B

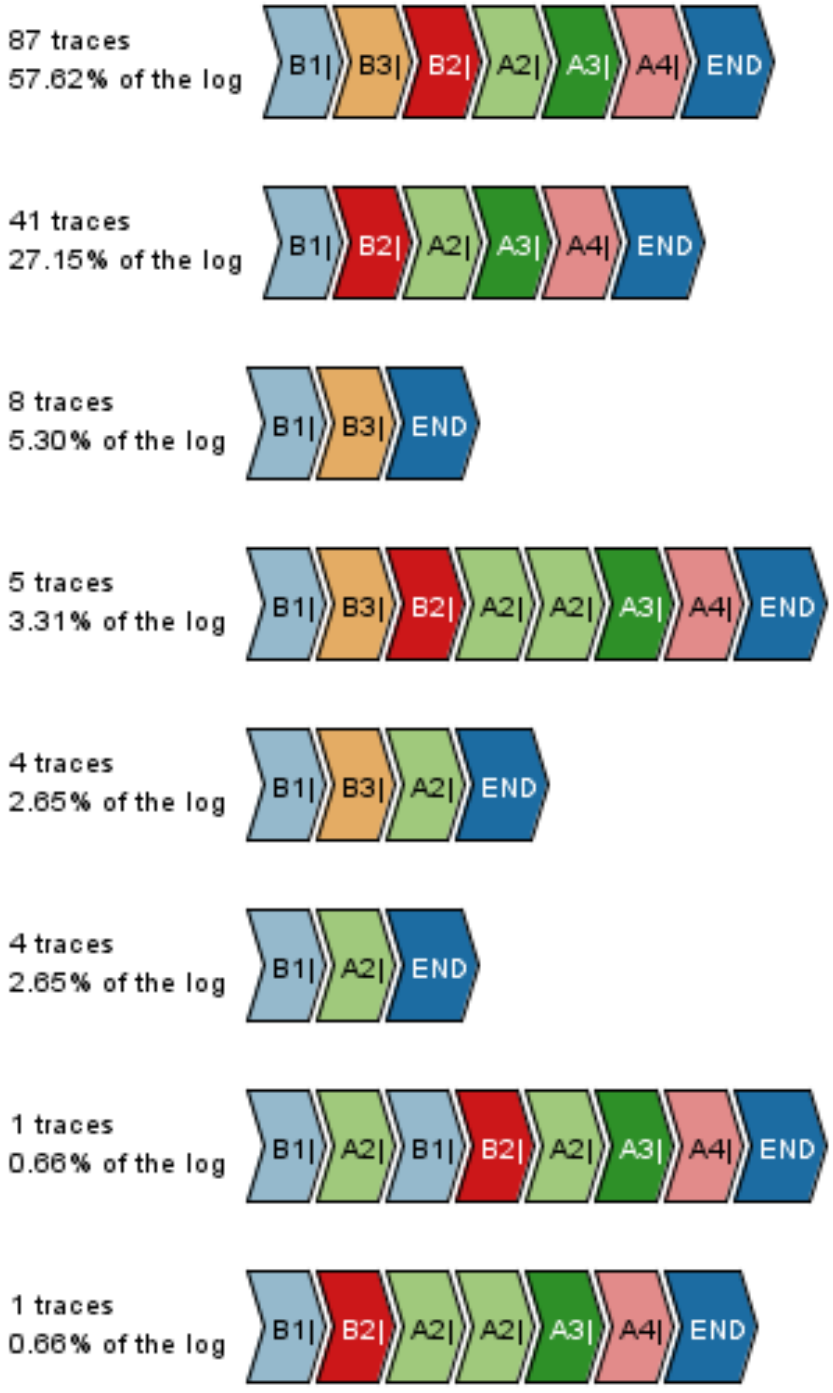

Figure 38 - Process B - Trace Variants

APPENDIX V – PROCESS DISCOVERY – PROCESS D

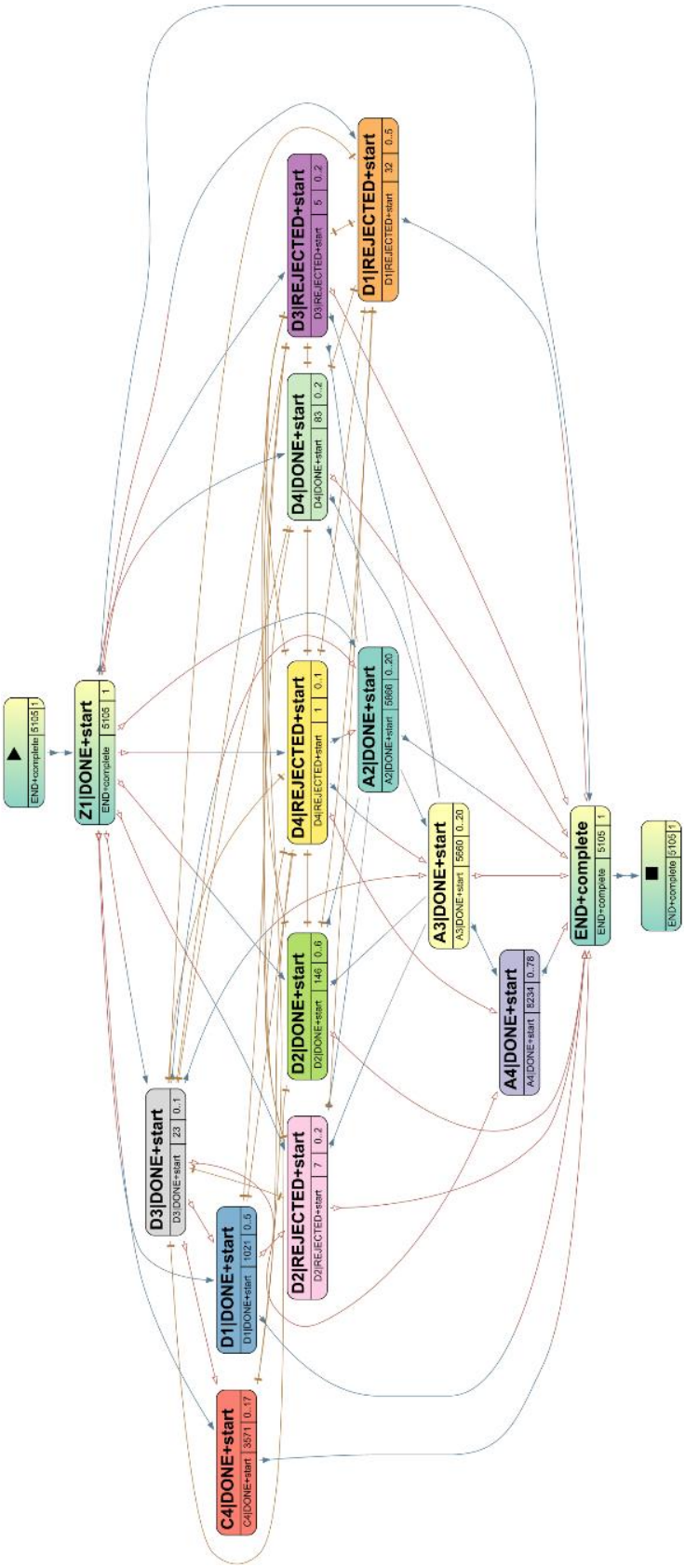

Figure 39 - Process D - Log Skeleton

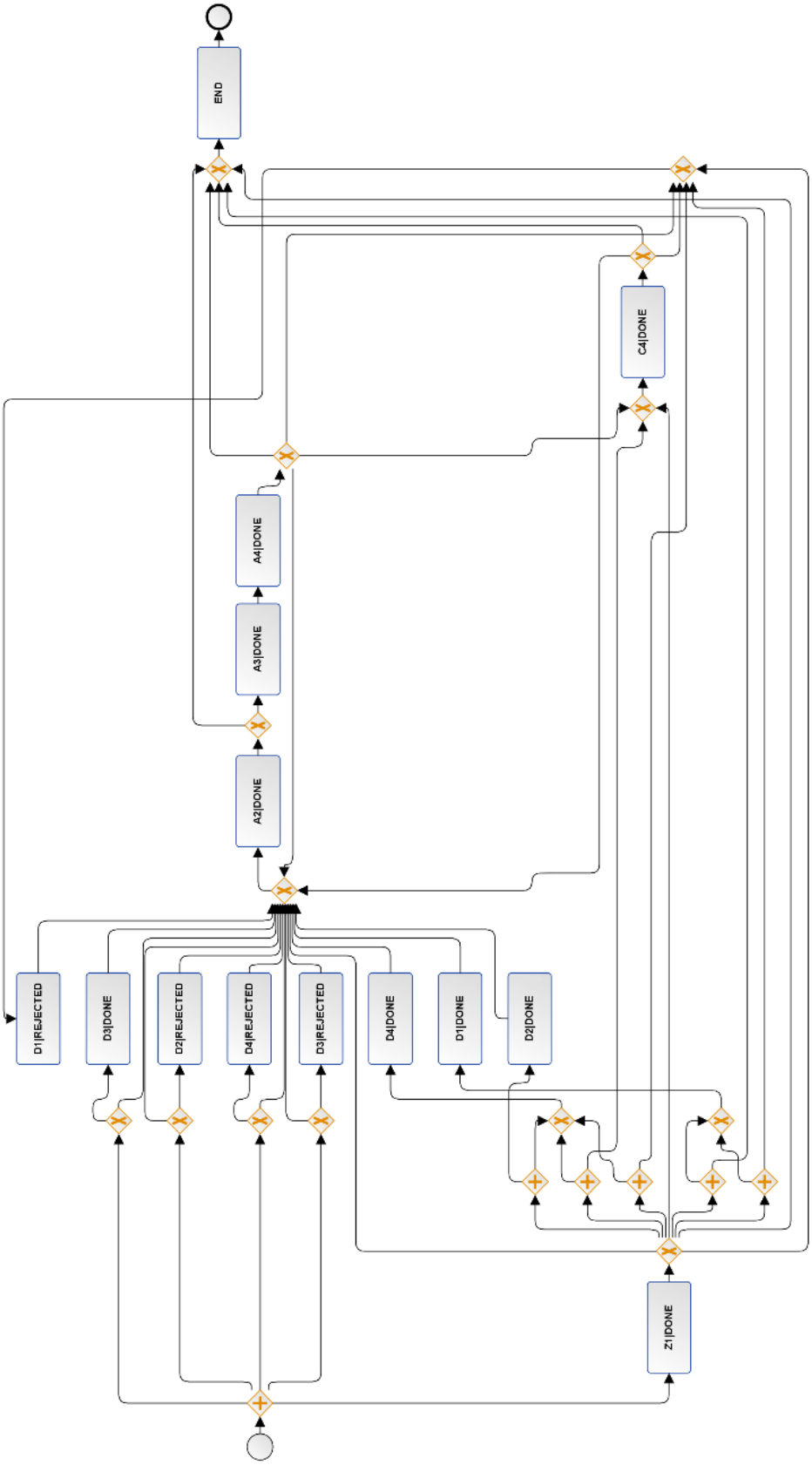

Figure 40 - Process D - Complete BPMN

APPENDIX VI – PROCESS DISCOVERY – PROCESS E

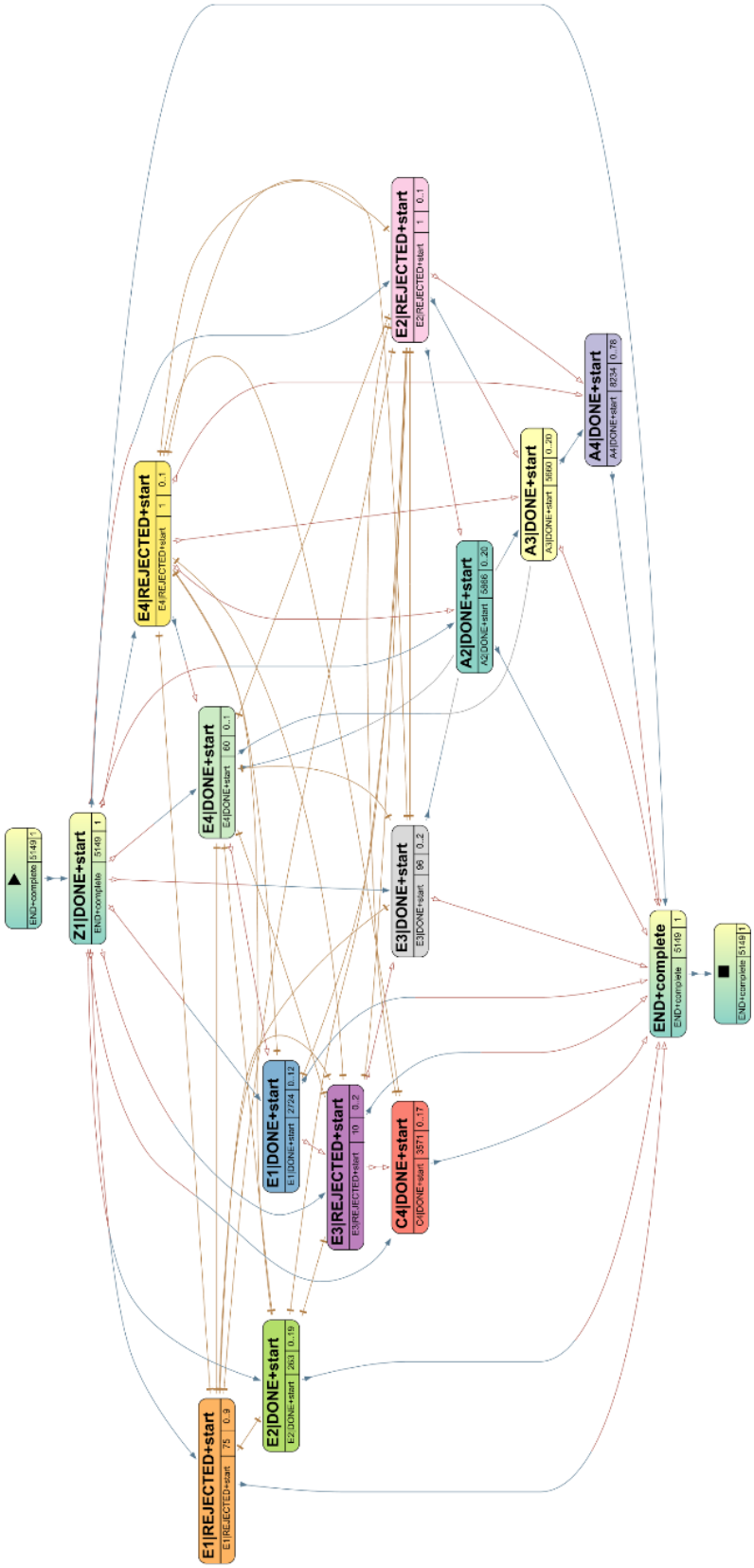

Figure 41 - Process E - Log Skeleton

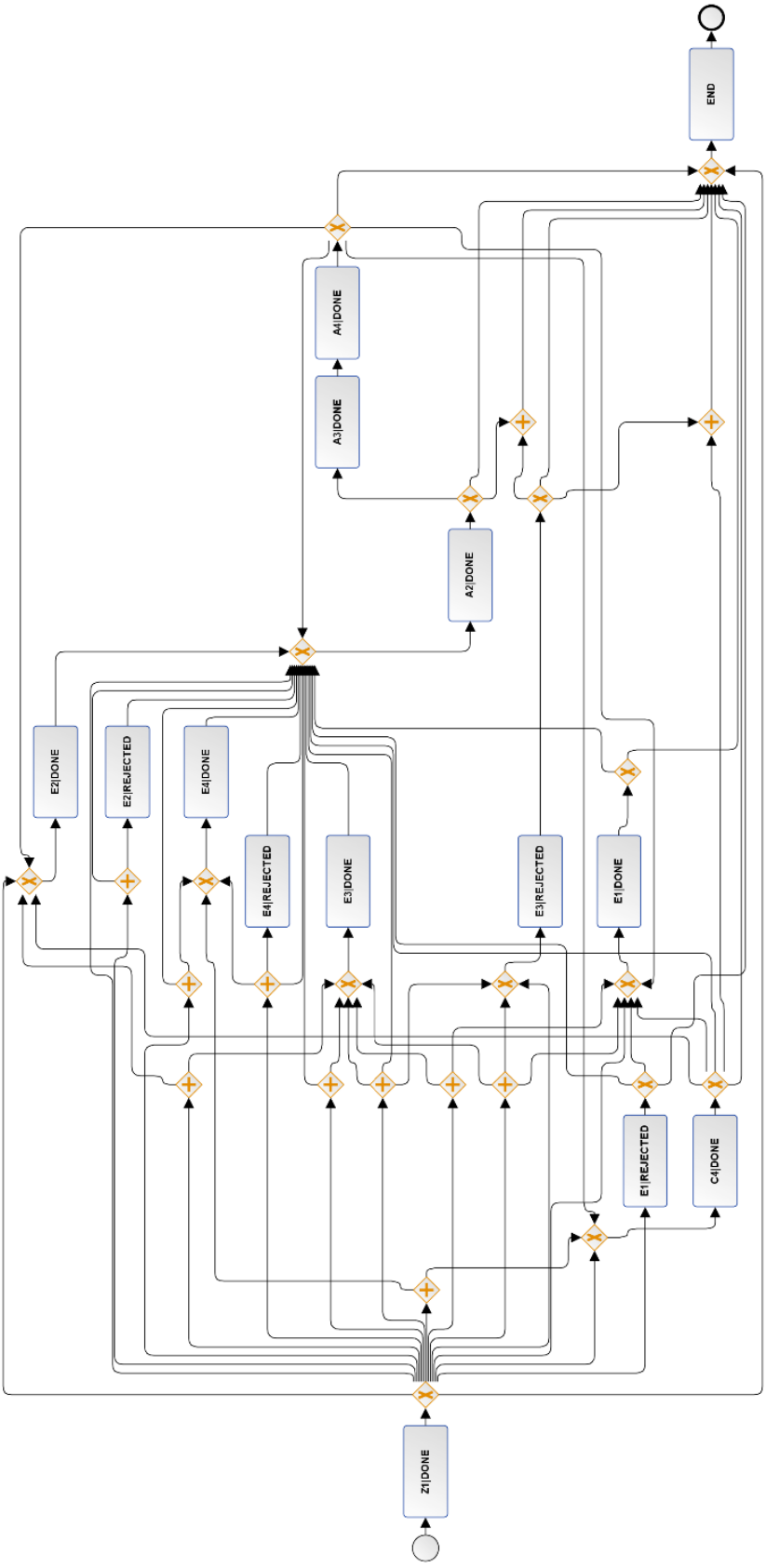

Figure 42 - Process E - Complete BPMN

APPENDIX VII – TIME ANALYSIS

Table 10 Analysis of Time Consumed when Applying the Proposed Method

	DATA SET		DATA				PM	
	TIME PERIODO DURATION	TOTAL NUMBER EVENTS	EXTRACTION PROCESS (SQL)	DATA QUALITY (EXCEL)	IMPORT CSV (ProM)	FILTER EVENTS (ProM)	CONFORMANCE CHECKING (ProM)	TOTAL
CASE_01	1,00	2.893	0,0011	0,0006	0,0003	0,0006	0,0007	0,003
CASE_02	1,00	18.156	0,0069	0,0025	0,0006	0,0022	0,0028	0,015
CASE_03	1,00	4.505	0,0036	0,0008	0,0003	0,0011	0,0014	0,007
CASE_04	1,00	4.483	0,0011	0,0008	0,0003	0,0011	0,0014	0,005
CASE_05	1,00	6.337	0,0019	0,0011	0,0003	0,0017	0,0021	0,007
CASE_06	1,00	627	0,0000	0,0003	0,0003	0,0003	0,0003	0,001
CASE_07	1,00	690	0,0000	0,0003	0,0003	0,0003	0,0003	0,001
CASE_08	1,00	4.902	0,0014	0,0008	0,0003	0,0011	0,0014	0,005
CASE_09	1,00	4.266	0,0017	0,0008	0,0003	0,0011	0,0014	0,005
CASE_10	1,00	5.141	0,0011	0,0011	0,0003	0,0022	0,0028	0,008
CASE_12	1,00	4.354	0,0014	0,0008	0,0003	0,0017	0,0021	0,006
CASE_13	1,00	13.825	0,0044	0,0019	0,0006	0,0022	0,0028	0,012
CASE_14	1,00	5.920	0,0019	0,0011	0,0003	0,0022	0,0028	0,008
AVERAGE	1,00	5853,77	0,002	0,001	0,000	0,001	0,002	0,006
CASE_11	7,00	7.902	0,0025	0,0011	0,0006	0,0022	0,0028	0,009
CASE_15	7,00	12.314	0,0061	0,0019	0,0006	0,0022	0,0028	0,014
CASE_16	7,00	11.384	0,0050	0,0019	0,0006	0,0017	0,0021	0,011
AVERAGE	7,00	10.533	0,0045	0,0017	0,0006	0,0020	0,0025	0,011
LARGE DS	456,00	1.000.000	0,1194	0,0367	0,0011	0,0039	0,0049	0,166