

# Enhancing Event Log Manipulation and Insight Discovery through Querying Process Representations with DFGs

María Salas-Urbano<sup>1</sup>

<sup>1</sup>University of Sevilla

## Abstract

In this doctoral thesis project we will address some limitations of the current process mining tools to analyze business processes. To achieve this, we propose to develop and evaluate a tool based on a query language for analyzing and visualizing business processes from event logs.

## Keywords

LoVizQL, process mining, query language, Directly-Follows Graph

## 1. Introduction

Process mining techniques use event logs to discover, analyze, and optimize business processes [1]. Current process mining tools offer several functionalities, such as data filtering or process visualization using Directly Follows Graphs (DFGs).

Process mining analysts often perform a frequent data analysis that involves a significant manual effort to obtain several multiple sets of traces from an event log. Additionally, the analysis implies identifying specific subsets that meet certain criteria, which necessitates repetitive actions and comparisons between DFGs, and relies heavily on the user. For instance, as observed in prior research [2, 3], a typical workflow for analysts involves comparing different case subsets (e.g., cases grouped by product category within a procure-to-pay process) to identify patterns or behaviors in the process data (such as cases containing transitions with unusually high cycle times).

The substantial user workload stems from the fact that current process mining tools are not prepared to simultaneously handle multiple DFGs in a consistent manner. Conducting this type of analysis using existing process mining tools is typically a time-consuming task that involves several steps. Initially, the analyst filters the event log to isolate cases associated with a specific product. Then, the analyst configures and explores the DFG to uncover insights related to those cases. This process is repeated for all products, which may be dozens or hundreds. Often, these DFGs are compared either with each other or with a specific pattern of interest to the analyst,

---

ICPM Doctoral Consortium and Demo Track 2023


\*Corresponding author.

✉ msurbano@us.es (M. Salas-Urbano)

ORCID 0000-0003-0620-1615 (M. Salas-Urbano)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

such as transitions with a high cycle time. This comparison is usually performed applying filters back and forth because most process mining tools can visualize only one process at a time.

Our goal is to support the analyst in carrying out this labor-intensive analysis by developing Log data Visualization Query Language (LoVizQL), a query language to obtain collections of DFGs that satisfy certain conditions desired by the user. With our approach, users can discover insights about the process without manually manipulating the event log, explore the data, and compare the various visualizations that are generated during the analysis. For instance, in a single LoVizQL query, users can filter the event log traces by each organizational unit involved in the process, obtain the corresponding DFGs for each data subset, and search for those DFGs where the frequency of activity rework exceeds the average.

To address its, this PhD thesis is driven by the following research question:

**RQ:** How can the manipulation of event logs and the discovery of aspects of interest in the DFGs be facilitated for users?

This research question can be answered by addressing the following objectives:

**OBJ1:** Identify the frequent workflows followed by analysts to compare different subsets of cases (e.g., cases grouped by product category of a purchase-to-pay process) and to identify interesting patterns or behaviors in process data through the use of DFGs.

**OBJ2:** Develop a query language to manipulate collections of DFGs and discover those that may contain relevant information.

**OBJ3:** Develop a support tool to effectively use the query language in order to visualize and analyze business processes from event logs.

**OBJ4:** Validate the tool with real scenarios and real users. The objective is to determine if results similar to those obtained with typical process mining tools are obtained in a more agile way.

The rest of the document describes the methodology that will be followed to address these objectives, details of the proposal and its current state of development, and an analysis of the work related to this doctoral thesis project.

## 2. Methodology

Design Science (DS) is the research methodology to be followed in this work. This methodology serves several purposes: it aligns with existing literature, provides a nominal process model and a mental model for presenting and evaluating DS research in information systems [4]. To achieve the target objectives and in accordance with the steps of DS, we pursue the following milestones: 1) to identify the problem and to motivate it, 2) to define objectives and solutions, 3) to design and development a support tool for process mining analysts, 4) to demonstrate the utility of the tool, 5) to evaluate its utility, and 6) to communicate and promote the obtained results.

## 3. Proposal

Regarding to the steps of DS mentioned in Section 2, we have already identified the problem in Section 1 and explained why it poses a challenge for analysts. In addition, we have also defined

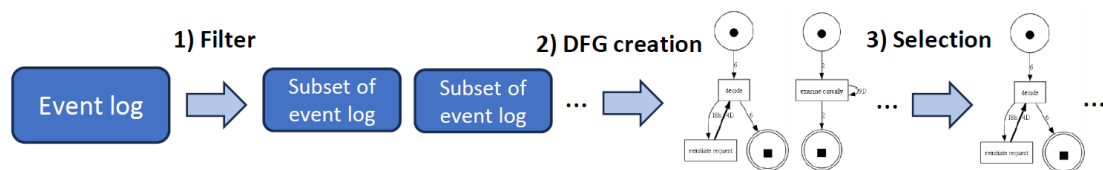
our research question and four objectives to address it. Currently, we have addressed OBJ1 and OBJ2.

In relation to the first objective (OBJ1), we have relied on the results published in the BPM 2022 [3]. In this work, we used Business Process Intelligence Challenge (BPIC) to discover how process analysts answered to specific business questions related to time performance. We coded 110 answers to time performance questions in more than 60 process mining reports. As a result, we identified 55 different operations with 137 variants used in them. We analyzed the types of answers and their similarities and examined how contextual information as well as existing process mining support may have affected them. These results provide an overview of the state of practice at that time in addressing questions related to time performance and have revealed opportunities for enhancing process mining tools. For instance, the study identified the iterative use of filters on event logs and the comparison of multiple DFGs from various subsets of traces as time-consuming tasks when utilizing these tools

In addition, we carried out an extended study of this work that is under review in a journal. Through a mixed-method approach, the study analyzes operations performed by process analysts in response to such questions using the previous reports and 12 screen/audio recordings. The research provides a detailed and fine-grained characterization of these operations, allowing for classification, comparison, and assessment of how contextual information influences the analysis.

Regarding the second objective (OBJ2), we have designed using Python a first version of a language called LoVizQL [5] based on this previous work [6]. LoVizQL aims to automatically generate collections of DFGs containing insights about a process without the need for manual manipulation and visualization comparisons. The user can categorize the characteristics of each resulting DFG collection using the query fields defined in each query row (cf Figure 1). Specifically, the user can determine how to manipulate the data (*Filter* step) and how to create the collection of DFGs (*DFG creation* step), that is, the characteristics of the DFGs (metric used, nodes, percentage of activities and paths shown). In addition, users can define the conditions that the DFGs must meet to be returned (*Selection* step).

On top of this language, we aim to develop tool support to help the users to visualize and analyze business processes from event logs.



**Figure 1:** Query language steps

We have already used LoVizQL to solve a provided questions in a BPIC and obtained results similar to those of some participants. Next, we plan to develop a tool to facilitate the use of this query language by users, and we plan to evaluate the future tool with real users and real scenarios, through experiments, addressing real analytic questions, and comparing the performance and effectiveness of the tool against current process mining tools such as Disco.

Finally, we plan to disseminate the results and promote the use of process mining, with the publication of the results in high-impact journals and participation in different conferences.

## 4. Related work

In the last decade, specific query languages have been developed for business process domains to obtain useful information about processes and assist in their executions. The process querying framework [7] has categorized these languages into various groups. Some of them have been categorized as event log query languages [8], encompassing diverse subject areas.

Some query languages focus on event log data, treating it as graphs to discover hierarchies and summarize information, such as [9]. Others [10] aims to simplify query writing, combining process and data perspectives for easier selections and insights. Some languages [11] facilitate querying Key Performance Indicators (KPIs) and Process Performance Indicators (PPIs) of activities or cases. Additionally, certain languages handle complex relations (constraints) between process elements, while a software company developed its own language for formalizing business questions as queries.

However, none of these languages allow users to iteratively filter event log data and select instances meeting specific conditions through comparisons. Existing process mining tools often require manual modification of Directly-Follows Graphs (DFGs) for specifying conditions, resulting in a tedious trial-and-error process. Inspired by data science, where query languages have addressed similar challenges in exploratory data analysis and visualization, we have designed LoVizQL, extending concepts from [12].

On the other hand, some works related to the identification of actions in process analysis have already been carried out. [2] qualitatively analyzes BPIC reports to understand how process analysts perform their work by focusing on visual representations. We complete this investigation by focusing on identifying all specific low-level operations to understand how specific issues are addressed. Furthermore, [13] carries out an empirical study to understand how analysts perform a process mining task, focusing only on the initial exploratory phase of process mining.

## 5. Acknowledgments

This research is partially funded by projects PID2021-126227NB-C21 (PERSEO), RTI2018-100763-J-I00 (CONFLEX) and TED2021-131023B-C22 (ORCHID) granted by MCIN/ AEI/ 10.13039/501100011033/ and ERDF A way of making Europe.

This PhD thesis is supervised by Manuel Resinas Arias de Reyna and Cristina Cabanillas Macías from the University of Seville.

## References

- [1] W. M. P. van der Aalst, A practitioner's guide to process mining: Limitations of the directly-follows graph, *Procedia Computer Science* 164 (2019) 321–328.

- [2] C. Klinkmüller, R. Müller, I. Weber, Mining Process Mining Practices: An Exploratory Characterization of Information Needs in Process Analytics, in: Business Process Management (BPM), 2019, pp. 322–337.
- [3] C. Capitán-Agudo, M. Salas-Urbano, C. Cabanillas, M. Resinas, Analyzing how process mining reports answer time performance questions, in: Business Process Management (BPM), Springer, 2022, pp. 234–250.
- [4] K. Peffers, T. Tuunanen, M. A. Rothenberger, S. Chatterjee, A design science research methodology for information systems research, *Journal of management information systems* 24 (2007) 45–77.
- [5] M. Salas-Urbano, C. Capitán-Agudo, C. Cabanillas, M. Resinas, Lovizql: A query language for visualizing and analyzing business processes from event logs, in: International Conference on Service-Oriented Computing (ICSOC), 2023, p. In press.
- [6] M. Salas-Urbano, C. Capitán-Agudo, C. Cabanillas, M. Resinas, A query language for exploring directly-follows graph collections, in: Jornadas de Ciencia e Ingeniería de Servicios (JCIS), 2022.
- [7] A. Polyvyanyy, C. Ouyang, A. Barros, W. M. P. van der Aalst, Process querying: Enabling business intelligence through query-based process analytics, *Decision Support Systems* 100 (2017) 41–56.
- [8] A. Polyvyanyy, *Process Querying Methods*, Springer, Cham, 2022.
- [9] A. Beheshti, B. Benatallah, H. R. Motahari-Nezhad, S. Ghodratnama, F. Amouzgar, A. Beheshti, S. Ghodratnama, F. Amouzgar, B. Benatallah, H. R. Motahari-Nezhad, *Process Querying Methods*, Springer, Cham, 2022.
- [10] E. González López de Murillas, H. A. Reijers, W. M. P. van der Aalst, Everything you always wanted to know about your process, but did not know how to ask, in: *BPM Workshops*, volume 281, 2017, pp. 296–309.
- [11] J. M. P. Álvarez, A. C. Díaz, L. Parody, A. M. R. Quintero, M. T. Gómez-López, *Process Querying Methods*, Springer, Cham, 2022.
- [12] P. S. Abril, R. Plant, The patent holder’s dilemma: Buy, sell, or troll?, *Communications of the ACM* 50 (2007) 36–44. doi:10.1145/1188913.1188915.
- [13] F. Zerbato, P. Soffer, B. Weber, Initial insights into exploratory process mining practices, in: *BPM Forum*, 2021, pp. 145–161.