

A Process Mining Starting Guideline for Process Analysts and Process Owners: A Practical Process Analytics Guide using ProM

Fitri Almira Yasmin^{1,2}, Rob Benthuis^{1,3}, Moustafa Elhagaly⁴, Fons Wijnhoven^{1,3}, and Faiza Allah Bukhsh^{1,3*}

¹ University of Twente, PO Box 217, 7500 AE Enschede

² falmirayasmin@gmail.com

³ {r.h.benthuis,a.b.j.m.wijnhoven,f.a.bukhsh}@utwente.nl

⁴ moustafa.elhagaly@outlook.com

Abstract. This report concerns a white paper on the demonstration of practically-oriented guidelines for process mining users. The main target group of this document is novice process mining users (e.g., those who are willing to start using process mining). After proposing the guideline, we present a few exercise questions.

Keywords: Process Mining · Demonstrability · ProM.

1 The Guideline

A business process entails different organizational aspects, such as functions, business artifacts, humans, and software systems [1]. These various aspects of a business process can be called process perspectives. The ordering of activities in process execution is called the control-flow perspective. When we focus on the resources of an organization, such as an employee or information system that executes the process, then we consider the organizational perspective of a process. Besides that, we can also focus on the frequency and timing of process execution. For example, how many times a particular activity is conducted within a timespan. This can be named the time perspective of a process.

One way to capture different aspects of a business process is through process mining. Process mining comprises data analytics techniques intending to extract process-related information. This field of study is introduced by Wil van der Aalst in 2004 [2]. Today, numerous process mining techniques, tools, and plugins are available. Key in conducting process mining analysis is to understand how the techniques, tools, and plugins can be applied in a meaningful and sound manner. Different permutations of steps may deliver different analysis results. These wide varieties pose process analysts with the concern about which process mining approach to choose and how to execute them appropriately. The purpose of this work is to propose a guideline that is able to give a quick and general

* corresponding author

process diagnostics overview of business processes, as well as to provide (potential) process mining users an idea about capabilities process mining offers. The proposed guideline is intended for novice process mining users, but the guideline may also be of interest to other users (e.g., those who have some experience, but want to refresh their knowledge). A key motivation behind the development of the guideline and the selection of this target audience is the broad spectrum of the process mining discipline. The vast options currently available might be overwhelming for novice process mining users to get started in using process mining tools.

An inability of novice users to perform process analysis using process mining tools can result in low perceived ease-of-use. Perceived ease-of-use is the degree to which a person believes that using a system would be free of effort. In turn, a low perceived ease-of-use can result in a low intention to use the application (or system). A way to increase the perceived ease-of-use is through the demonstration of expected results or outcomes, and that is the goal of our artifact. We want to demonstrate a practical step-by-step process analysis using process mining techniques. The tools, techniques, and plugins used in the presented guideline are mainly based on the exploration of the current state-of-the-art of process mining practices.

In this document¹, we will demonstrate the implementation of the guideline using an event log from www.processmining.org. The reader will follow the guidelines to perform process analysis for an example event log. In this section, the event log named *teleclaims.xes*² is used, but in the exercise section the respondents will use a different file from the same source (namely, the file *reviewing.xes*). The teleclaim event log contains records of the handling of different types of the insurance claim. For evaluation purpose, please follow the exercise that is included in this report and fill out the additional questionnaire about the guideline³.

1.1 Preparation

Typically, a process analyst first needs to acquire an event log of the selected process. Also, the process analyst needs to install the tools. We suggest the latest version of ProM (at the time of writing, this is version 6.9) as a tool. ProM is open-sourced and can be downloaded from www.promtools.org. ProM can load XES, MXML, and CSV files. To extract files from other data sources, ProM Import⁴ can be used [3]. The format of the event log we demonstrate in this report is in the form of XES file.

¹ This document is based on the paper: F. A. Yasmin, “Enhancement in process mining: guideline for process owner and process analyst,” Enschede, University of Twente, 2019

² Download *chapter_8.zip* from http://www.processmining.org/event_logs_and_models_used_in_book

³ This url is not included in the public version

⁴ ProM Import is available at <http://www.promtools.org/promimport/>

Figure 1 shows the page that will appear when users open ProM (the so-called Workspace). The import button is located at the upper right of the page. There are several import options. Just to mention some, the Naive one uses the most memory but is fast and lightweight, the sequential one is also fast but has some (minor) limitations, and the disk-buffered by MapDB import puts little information in the memory and can be a good choice when you have large event logs [4].

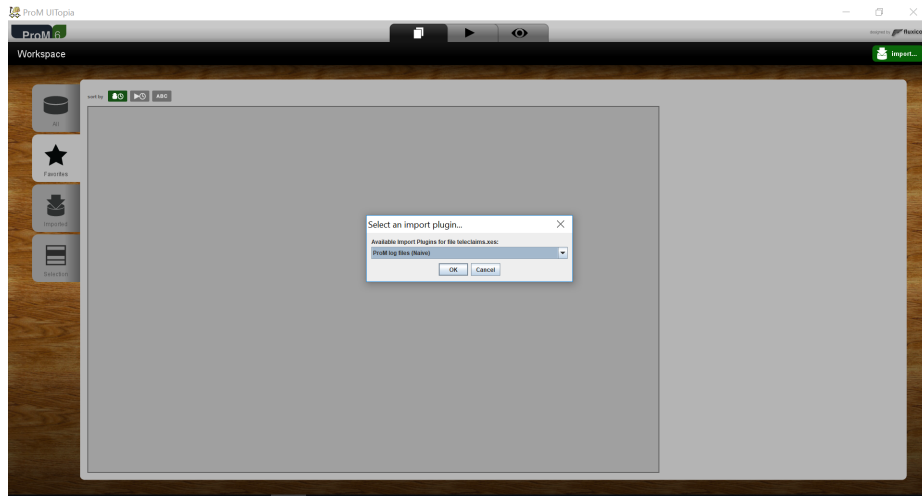


Fig. 1. Import event log in ProM

The user can import the *teleclaims.xml* file using the Naive import (which is also the default option for log import). Now, the user should click on the ‘view resource’ button on the right (the eye icon). One would see a similar interface as shown in Figure 2. Note that this figure is based on a different event log (the teleclaims log should contain 46138 events related to 3512 cases, the claims). The dashboard displays general information such as the number of cases, events, event classes, and originators. It also displays the minimum, maximum, and average number of events per case. In the ‘inspector’ view, you can view the detailed ordering of events per case. The ‘summary’ view lists the available event classes, the starting events, the end events, and the resources of the process.

1.2 Process Discovery (Control-Flow)

The traditional way of process modeling through meetings and discussions typically takes from a half to two years, depending on the size of the organization and the time the managers participating in the team can allocate to do the work [5]. An alternative approach is to discover a process model by using process mining, which provides a process model based on the observed behavior [6].

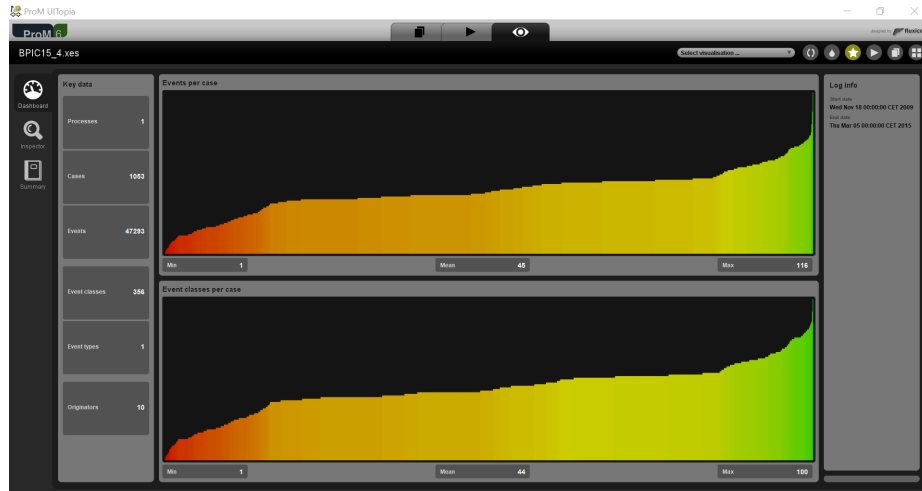


Fig. 2. Log info visualization in ProM

The discovery task of the control-flow perspective in process mining is usually referred to as process discovery. A process discovery algorithm is a function that maps event logs onto a process model such that the model is “representative” for the behavior seen in the event log [3].

There are at least 16 process discovery algorithm plugins available in ProM. In a concise literature review, we found four commonly used plugins for process discovery, namely alpha miner, inductive miner, fuzzy miner, and heuristic miner. Fluxicon [7] recommends a fuzzy miner and heuristic miner to be used in practice for users that just started using ProM. The alpha algorithm was one of the first process discovery algorithms, but, in general, it is not perceived as a practically useful mining technique as it has challenges with noise, infrequent or incomplete behavior, and complex routing constructs [3]. On the contrary, inductive mining techniques can handle infrequent behavior and deal with huge models and logs while ensuring formal correctness criteria such as the ability to rediscover the original model [3]. Furthermore, the inductive miner is one of the few algorithms that guarantees a sound process model [4]. That is, it will always produce a model that can replay the whole event log. In other words *fitness* is guaranteed [3]. Lastly, while both the alpha miner and the inductive miner can produce a Petri Net as output, fuzzy miner and heuristic miner do not produce this.

The user should go back to the Workspace menu by clicking on the document icon (the most left icon) in the navigation tab. Make sure to select the teleclaims file after which one can proceed by clicking on the “action” button. This button is located in the upper middle part of the display with the triangle symbol (and also on the right next to the ‘view resource’ button). We will use the “Mine Petri Net with Inductive Miner” plugin (with default settings) as process discovery algorithm (see Figure 3).

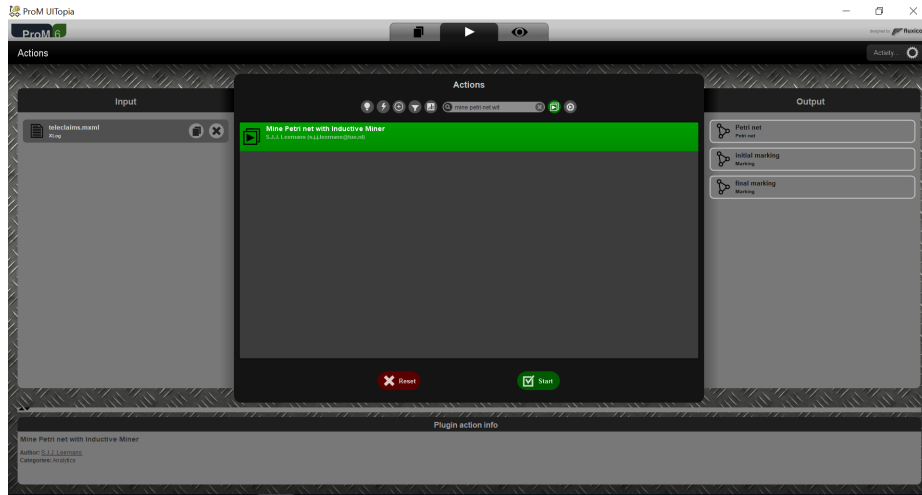


Fig. 3. Inductive Miner in ProM

Figure 4 shows the discovered process model of the event log. As shown, the discovered model is concise and can be interpreted with relative ease. We can see that the claim process starts with the customer call to the Brisbane or Sydney call center. Then, each call center checks whether the information is sufficient. If this is the case, then the claim is registered. After the assessment, the claim will be either paid or reimbursed. However, if the assessment determines that the claim is not eligible for payment or reimbursement, then the case will be ended, i.e., the claim is rejected.

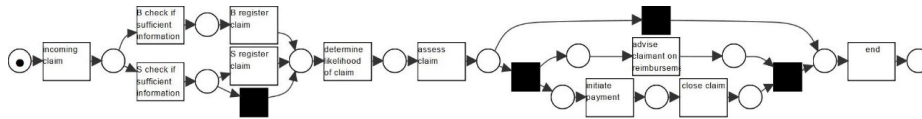


Fig. 4. Mined process model after using the Inductive Miner

1.3 Conformance and Performance Analysis (Time)

The discovered model of a process can be compared to the behavior recorded in the event log to analyze commonalities and discrepancies between them. Regarding the purpose, there are two types of process models: normative and descriptive. When a model is intended to be descriptive, then the discrepancies indicate that the model needs to be improved to capture reality better [3]. When the discovered model is intended to be normative, the discrepancies mean that there

are deviations in the process execution [3]. For example, when an employee needs to conduct unforeseen activities that are not prescribed in the existing process model, or when an employee bypasses a certain procedure and violates regulations. In this case, the process itself, not the model, need to be improved.

It is described that a process mined model aims to produce a process model that is “representative”. Representativeness of a process model can be operationalized by requiring that the model can replay all behavior in the log [3]. This is the so-called “fitness” prerequisite, which is often perceived as an important quality dimension of a process model. Process model conformance uses the recorded behavior to verify how well the process model conforms with the observed behavior or vice versa, and it also indicates where the actual execution differs from the process model [6].

The performance of a process can be considered from three dimensions: time, cost, and quality. Here, we analyze the time dimension of a process. The time perspective is concerned with the timing and frequency of events [3]. By replaying executed traces on a process model, the timing information of the different steps in the process becomes available [6]. When events bear timestamps, it is possible to discover bottlenecks, measure service levels, monitor the utilization of resources, and predict the remaining processing time of running cases [3].

There are at least 10 different plugins for conformance or performance analysis in ProM. At least two of them (e.g., the “Replay Log on Petri Net for Conformance/Performance Analysis” by Adriansyah and “Multi-perspective Process Explorer” by Mannhardt) can explore performance and conformance metrics. The “Multi-perspective Process Explorer” is a plugin that integrates multi-perspective process mining techniques for discovery and conformance checking [8]. In this demonstration part, we will use that plugin (see Figure 5) for performance and conformance analysis in ProM. One may proceed from the previous step (the mined Petri Net from Figure 4) by clicking on the ‘Use resource’ button in the upper right part of the interface. Alternatively, one can also go back to the ‘Actions’ tab (middle bottom in the upper display) and use as input *teleclaims.xml* and the mined Petri Net.

In contrast to the simple and static output obtained after using Adriansyah’s plugin, the “Multi-perspective Process Explorer” comes with a configuration panel that enables users to choose measures and mode preferences. For example, the plugin allows different mode choices. The fitness mode describes whether the model represents the behavior of the event log. After selecting the “simple configuration”, the result shows an 60.9% average fitness score of the discovered process model (see Figure 6). One can observe from the graph that more than half of the cases are reimbursed or paid, while the rest are rejected.

Figure 7 shows a performance analysis based on the aligned event log (the user may want to try to reproduce the figure). The darker blue color indicates the longer average time of the activity. As shown, the activity “determine the likelihood of claim” takes the longest average time with 2.9 hours per case.

For performance analysis, in general, the fitness of the process model with respect to the event log needs to be high. One reason for this is that timing

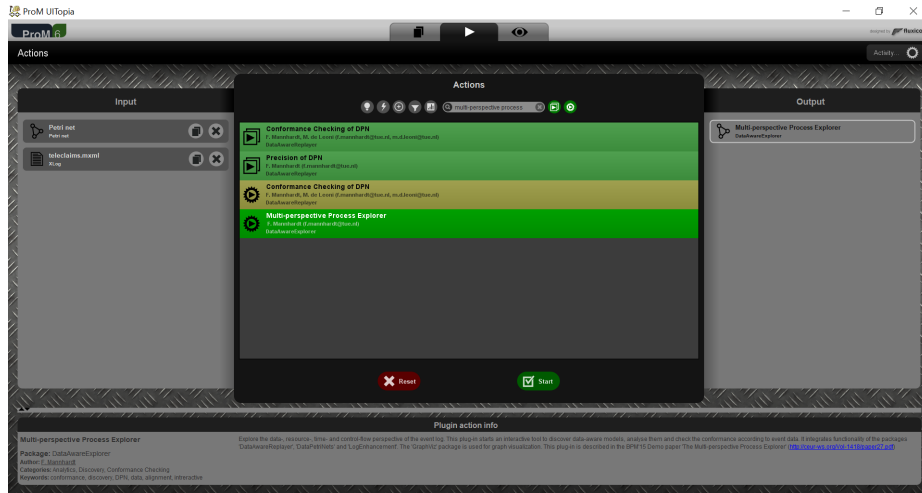


Fig. 5. Multi-perspective Process Explorer plugin in ProM

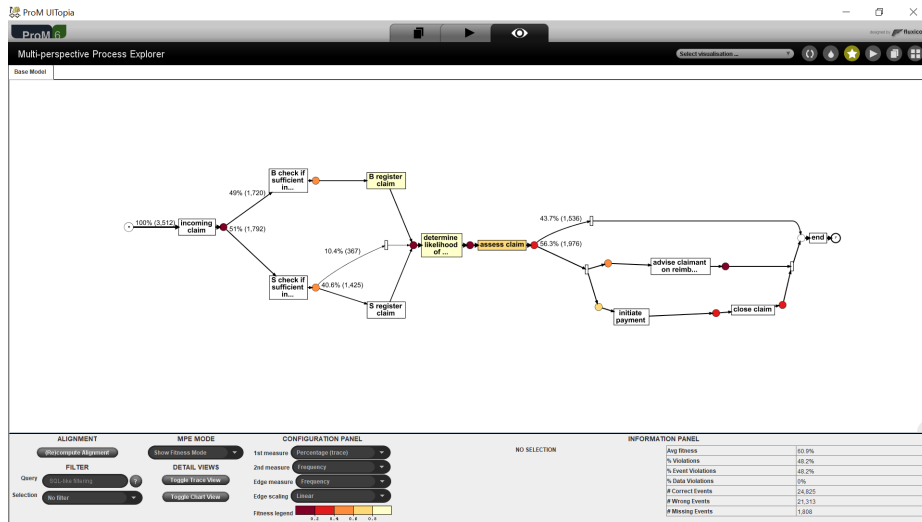


Fig. 6. Fitness score as given by the Multi-perspective Process Explorer

information of traces can then be appropriately replayed on the process model [6]. A good practice is to calculate the performance of the model with a minimum of 99% fitness to the log. In the previous step, one could observe that the average fitness of the discovered model is not that high. To this end, one can use the “Align Log to The Model” plugin. After the alignment, the fitness of the model to the log is at 100%.

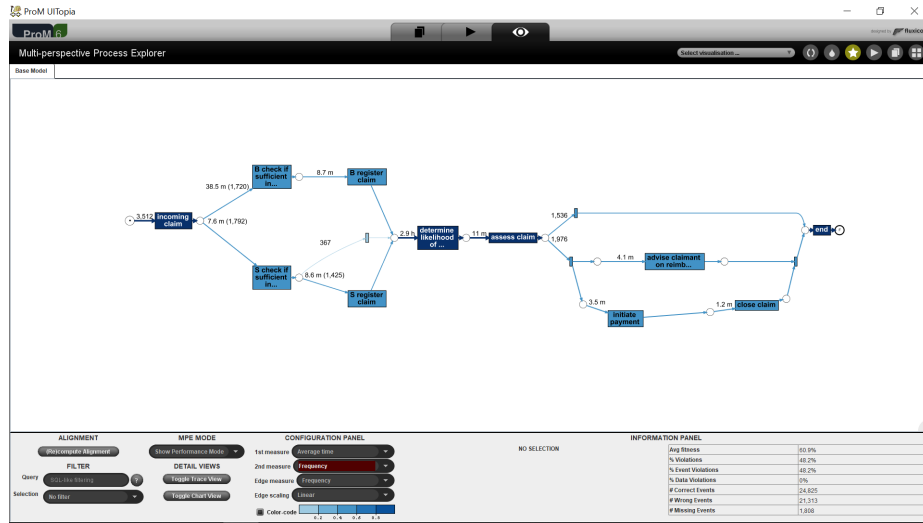


Fig. 7. Performance score as given by the Multi-perspective Process Explorer

1.4 Social Network Analysis (Organizational)

Organizational mining focuses on organizational perspectives. The starting point for organizational mining is typically the resource attribute in the event log [3]. An organizational entity in process mining can be, for example, a person, a department, a role, or a system that performs the activity. Before performing analysis by considering the organizational perspective, it is important to know what activities are performed by a certain resource. A dotted-chart can help to visualize this information.

In the ‘view’ tab, one can find the dotted chart visualization of an event log. The dotted chart is an example of a visual analytics technique where users visually identify patterns and trends in (large) datasets [3]. To access the ‘view’ tab, one can select the imported event log and then select the eye-shaped button. Next, as visualization (see upper-right selection box) one can select the “Dotted Chart”. In the dotted chart page, the user can choose the “resource” attribute at one axis and the “event” at the other (one may also want to adjust the color attribute). Figure 8 shows an example of a dotted chart for *teleclaims.xes*.

In this document, social network analysis is used to present the entities’ relationships in the form of a graph. Social network analysis views relationships in terms of network theory, consisting of nodes that represent individual actors within the networks, and arcs that represent the relationships between the actors [9]. Arcs and nodes may have weights that indicate its importance. There are several types of social network analysis based on the construction of the network from the event log.

A commonly used metric is the “handover-of-work”. Handover-of-work describes work exchange between individuals. The more frequent individual x per-

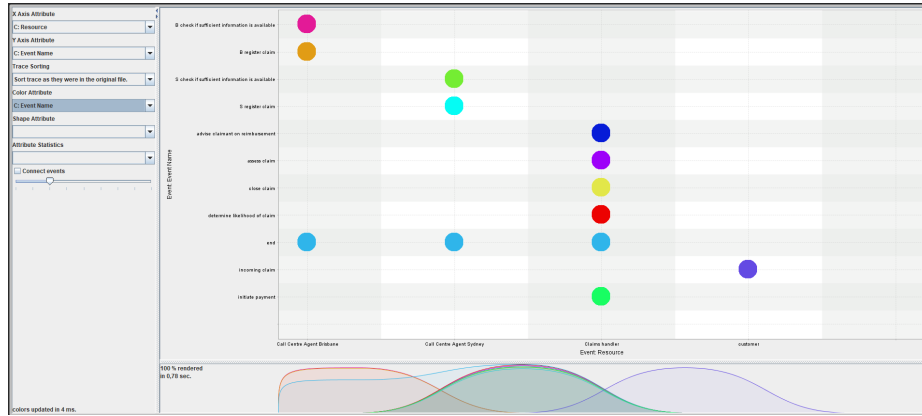


Fig. 8. Event and resource dotted chart of the teleclaims case

forms an activity that is causally followed by an activity performed by individual y, the stronger the relation between x and y usually is [10]. One can use the “Mine for a Handover-of-Work Social Network” plugin to analyze the organizational perspective. For this, go back to the ‘Actions’ menu and select the miner while using as input the original event log file (make sure that the mined process models/Petri Nets are removed from the input list).

Figure 9 shows an example of a social network graph of the teleclaims case, based on the handover-of-work metric. Such a graph provides visual imagery behind the basic concept of relations between the actors [11]. One can choose a ‘view option’ to visualize the importance of the resources and their relationship. For example, one can check the ‘size by ranking’ and ‘show edge weight values’. The size of the nodes indicates the weight of the nodes. The weight of the edge is written in numbers. Also, users can choose a different layout for the social network. In the example as shown in Figure 9, we selected the ‘Ranking View’ so that we can see the degree of centrality of each resource. Figure 9 shows that the customer can contact the call center in Sydney or Brisbane. After that, both branches direct the claim to the claim handler. By looking at the size and the centrality of the nodes, one may say that the claim handler is one of the “most” important resources in the process.

A problem one may face with the social network graph is that it illustrates only the patterns of connections and no (or limited) information about the causality between the resources. To provide information about the relations between the actors, the reader is suggested to use the “Discovery of Resources Causal Matrix” plugin to accompany the social network analysis graph. In ProM, one can choose the “Discover Matrix” plugin (in case there are multiple plugins, try to find the one used in this report) and choose the “resource” attribute as classifier (the miner can be the default one). An example is given in Figure 10. Notice that one can also change the visualization to a different representation, such as the one shown in Figure 11.

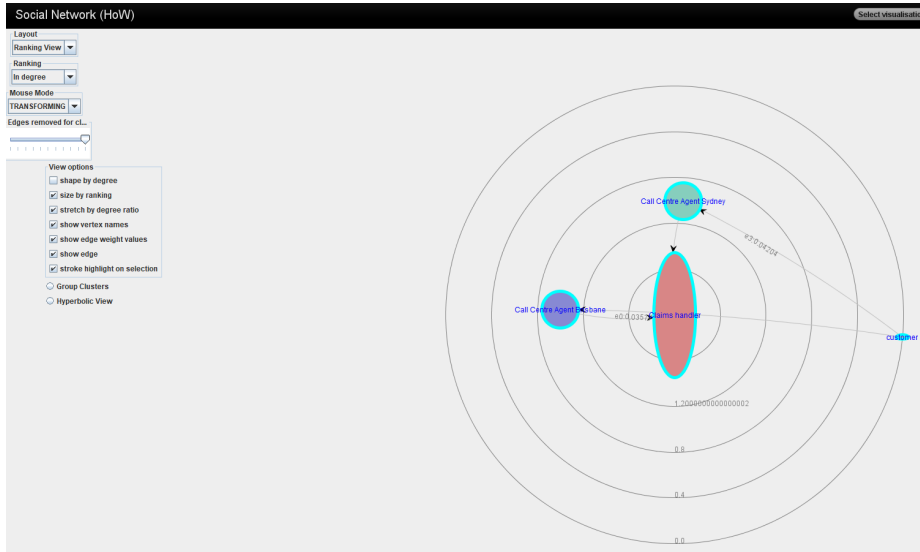


Fig. 9. Social network analysis of the teleclaims case

Causal Activity Matrix				
Matrix	Call Centre Agent Brisbane	Call Centre Agent Sydney	Claims handler	customer
Call Centre Agent Brisbane	0.0	-1.0	0.002476361056225	-1.0
Call Centre Agent Sydney	-1.0	0.0	0.002307419276302	-1.0
Claims handler	-1.0	-1.0	0.0	-1.0
customer	0.000413042075305	0.0004450705103652	-1.0	-0.11133891860494582

Fig. 10. Causal matrix using colored table

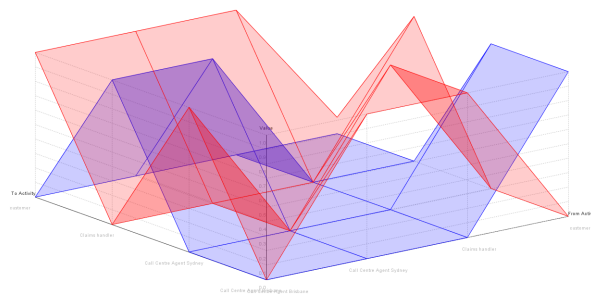


Fig. 11. Causal matrix using grid

1.5 Decision Mining (Case)

Decision mining discovers why particular cases take a specific path [12]. To analyze the choices in a business process, one can first identify those parts of the model where the process is split into alternative branches. These parts are called decision points [13]. In a Petri Net, a decision point is a node with more than one outgoing arc. After identifying a decision point, the influence of case data to the decision can be evaluated, i.e., whether cases with certain properties follow a specific route.

The idea is to convert each of the decision points into a classification problem where the classes are the different decisions that can be made [13]. A classification technique, such as decision tree learning, can be used to find decision rules [3]. In a classification technique, one often focuses on target attributes for prediction [14]. The prediction is called a response variable, and a target attribute is called a prediction attribute. There are three main types of prediction attributes [14]:

- **Numeric:** describes the quantitative value of the attribute (e.g., the number of orders in a case);
- **Ordinal (or categorical):** provides the ordering of the category but without the exact measurement of the distance between them. An example is the classification of socioeconomic status in: poor, middle class, and rich;
- **Nominal:** the values are merely distinct names or labels with no meaningful order by which one can sort the data. For example, the customer’s country of origin.

In contrast to the control-flow perspective, which is one of the main research fields in process mining, there is limited support and less attention for the case perspective of the process [13, 12]. One of the first plugins in ProM that supports the case perspective is the “Decision Miner”, which was implemented in 2006 [15]. However, this plugin cannot always deal with event logs which are characterized by deviating behavior and more complex control-flow constructs [12]. A more recent plugin named “Discovery of the Process Data-Flow (Decision-Tree Miner)” was developed to overcome (a part of) the limitations of the “Decision Miner” plugin. It can discover an accurate data flow in combination with event logs that contain non-conforming traces [12]. In this document, we will use the “Discovery of the Process Data-Flow (Decision-Tree Miner)” plugin for decision mining of the case perspective.

Figure 12 shows a Petri Net with data discovered by using the plugin. Since the data properties of the case are important, it is prudent to check the data types of the attributes in the log in decision mining. Often, if there are formatting errors in the data, the decision-tree miner will end up running without ends. This is what happened when one tried to mine the decision tree based on imported *teleclaims.xes*. One solution to overcome the problem that the decision-tree miner cannot process the import event log, is to repair the event log.

One can use the “Repair Type of Event Attributes in the Log (Discrete, Continuous, Boolean, Date)” plugin for the purpose. For this, we leave a detailed implementation to those interested in the subject. However, to give you

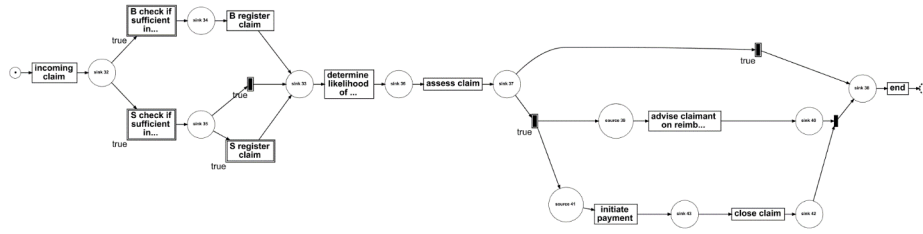


Fig. 12. Decision Tree of the teleclaims case

some impression on how one can obtain a decision-tree, one can simply use the "Discovery of the Process Data-Flow (Decision-Tree Miner)" plugin that uses as input a Petri Net and an event log. Consequently, in the configuration panel one can select as variable only the location.

Figure 12 shows an output one can obtain after applying the decision-tree miner. The plugin discovers guards for transitions at a decision, instead of decision logic for an entire decision point [16]. However, the decision logic that is extracted by the decision-tree miner can be in unclear formats, thereby reducing the utility of decision-mining plugins for daily practice [16]. In fact, many of the decision miners either directly use or are based upon open source algorithms. Thus, for the case perspective, the result should be considered with carefulness.

1.6 Recommendation

After the analysis, a process analyst may summarize the results and provide input for (further) process improvement [4]. However, we do understand that organizations typically have their own key performance indicators and business goals. In addition, the interpretation of the analysis depends on the domain expertise of process owners. Therefore, we will not prescribe exact improvement actions process owners may think of. Instead, we will mention some general examples of improvement actions that can result from analysis using process mining techniques, as addressed by [3]:

- **Redesign:** insights attained in the analysis phase can prompt changes to the process. For example, if the analysis discovers that the performance of the business process is poor, then it indicates that the process needs to be redesigned.
- **Adjust:** after applying process mining techniques, one can recommend (temporary) adjustments. Here, the process is not redesigned; only predefined controls are used to adapt or reconfigure the process. For example, a company might define in their contingency plan that as long as a criteria are met, they will allocate certain resources to accommodate the cases.
- **Intervene:** process mining may also reveal problems related to particular cases or resources. For example, the process owner can decide to terminate a

designated problematic case or take disciplinary action for an employee that violates compliance regulations.

- **Support:** based on historical information, process mining techniques can be used to predict the remaining flow time or recommend the action with the lowest expected costs.

2 Exercise

In this exercise, please answer the questions below based on the analysis of the *reviewing.xes* business process. Please go to this link⁵ to download the event log for the exercise.

2.1 Event Log Overview

How many cases are in the reviewing process?

How many events are in the reviewing process?

How many events are there per case on average?

What is the maximum number of events per case?

What is the minimum number of events per case?

How many originators/resources are recorded in the event log?

2.2 Process Discovery

Please briefly explain the process of reviewing based on the discovered process model:

⁵ http://www.processmining.org/event_logs_and_models_used_in_book

2.3 Conformance and Performance Analysis

What is the percentage of log that conforms to the discovered process model (i.e., fitness)?

Please list the events that needs to be concerned by the process owner (e.g., events that cause bottleneck or took the longest time). Please also include a reason why:

2.4 Social Network Analysis

Please name the “most important” resource as defined by this document and explain why:



Please list the activities that are performed by one of the “most important” resource(s):



Please list the resources that are working with one of the “most important” resource. Also mention the causality degree(s) to this resource(s):



2.5 Decision Mining

Please list the decision points of the process or motivate why this is not possible:



2.6 Recommendation

Please give an improvement recommendation for the reviewing business process:



References

1. M. Dumas, M. La Rosa, J. Mendling, and H. A. Reijers, *Fundamentals of Business Process Management*. Springer Berlin Heidelberg, 2013, vol. 1.
2. W.M.P. van der Aalst and A.J.M.M. Weijters, “Process Mining: a Research Agenda,” *Computers in Industry*, vol. 53, no. 3, pp. 231–244, 2004.
3. W.M.P. van der Aalst, *Process Mining*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016.
4. Buijs. Introduction to Process Mining with ProM. <https://www.futurelearn.com/courses/process-mining/0/steps/15643>, accessed 2020-07-18.
5. P. Harmon, *Business Process Change: a Business Process Management Guide for Managers and Process Professionals*, 4th ed. Morgan Kaufmann, 2019.
6. A. Adriansyah and J. C. A. M. Buijs, “Mining Process Performance from Event Logs,” in *International Conference on Business Process Management*. Springer, 2012, pp. 217–218.
7. Fluxicon, “ProM Tips - Which Mining Algorithm Should You Use? - Flux Capacitor,” <https://fluxicon.com/blog/2010/10/prom-tips-mining-algorithm/>, accessed 2020-08-17.
8. F. Mannhardt, M. De Leoni, and H. A. Reijers, “The Multi-perspective Process Explorer,” *BPM (Demos)*, vol. 1418, pp. 130–134, 2015.
9. D. Passmore, *Social Network Analysis: Theory and Applications*, 2011, http://train.ed.psu.edu/WFED-543/SocNet_TheoryApp.pdf, accessed 2020-08-17.
10. W. M. P. van der Aalst, H. A. Reijers, and M. Song, “Discovering Social Networks from Event Logs,” *Computer Supported Cooperative Work (CSCW)*, vol. 14, no. 6, pp. 549–593, 2005.
11. J. Scott, “What is Social Network Analysis?” *London: Bloomsbury Academic*, 2012.
12. M. De Leoni and W. M. P. van der Aalst, “Data-aware Process Mining: Discovering Decisions in Processes using Alignments,” in *Proceedings of the 28th annual ACM symposium on applied computing*, 2013, pp. 1454–1461.
13. A. Rozinat and W. M. P. van der Aalst, “Conformance Checking of Processes Based on Monitoring Real Behavior,” *Information Systems*, vol. 33, no. 1, pp. 64–95, 2008.
14. L. Rokach and O. Maimon, *Data Mining with Decision Trees: Theory and Applications*. World scientific, 2014, vol. 81.
15. A. Rozinat and W. M. P. van der Aalst, “Decision Mining in ProM,” in *International Conference on Business Process Management*. Springer, 2006, pp. 420–425.
16. S. Peeters, “Decision Mining in ProM: From Log Files to DMN Decision Logic,” Ph.D. dissertation, Ghent University, 2016, http://lib.ugent.be/fulltxt/RUG01/002/273/963/RUG01-002273963_2016.0001_AC.pdf, accessed 2020-08-17.