BPI Challenge 2020

Adem Baran Orhan
Undergraduate Erasmus Student

Business Information Systems
Prof. Paolo Ceravolo
a.a. 2021 - 2022

Contents

1 Description of Case Study

The TU/e staff travels to conferences or other universities for project meetings and/or to meet up with colleagues in the field. Moreover, as many companies, they have procedures in place for arranging the travel and reimbursing costs.

The high amount of domestic or international travel at the University makes it important for our project.

Having optimized and improved flow for the travel process will be what the university wants.

In the case study, we have actors for giving permission or not. On a high level, we distinguish two types of trips, namely domestic and international.

For domestic trips, no prior permission is needed, i.e., an employee can undertake these trips and ask for reimbursement of the costs afterward.

For international trips, permission is needed from the supervisor. This permission is obtained by filing a travel permit, and this travel permit should be approved before making any arrangements.

To get the costs for a trip reimbursed, a claim is filed. This can be done as soon as costs are actually paid or within two months after the trip.

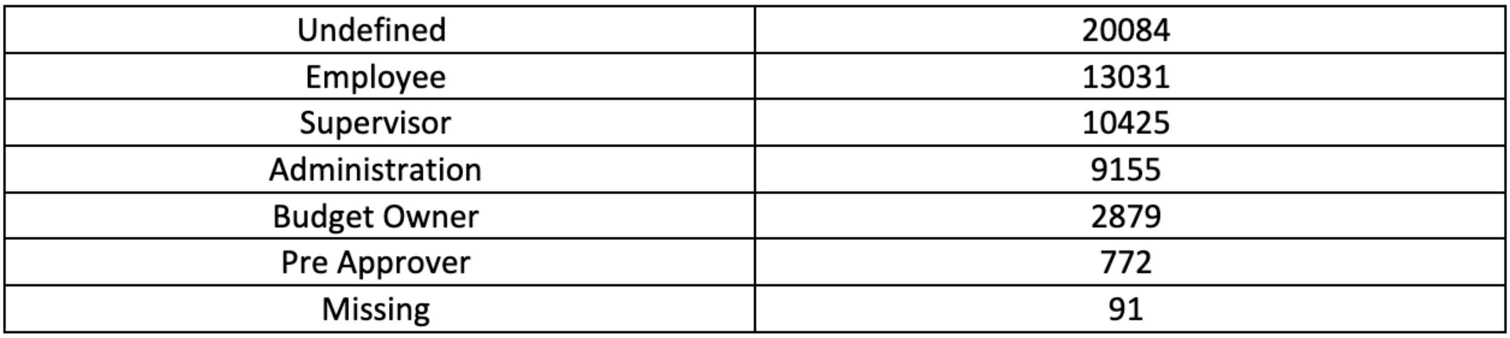
We have two types of trips. Domestic and International

1.1 Dataset

The dataset contains data from 2017 (only two departments) and 2018, the full TU/e.

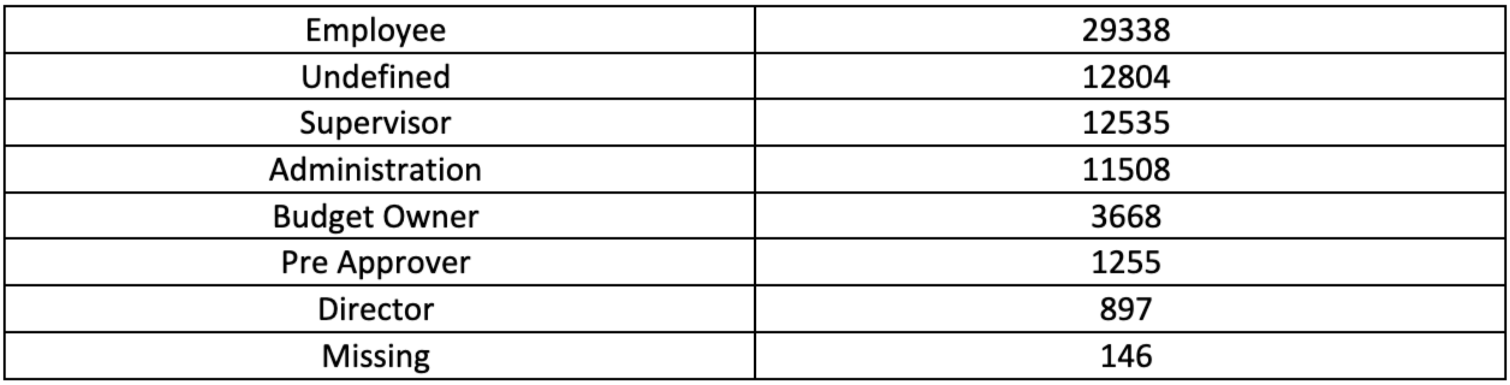
- Domestic declarations - 10,500 cases, 56,437 events
- Prepaid travel costs - 2,099 cases, 18,246 events
- International declarations - 6,449 cases, 72151 events
- Requests for payment - (should not be travel related): 6,886 cases, 36,796 events
- Travel permits (including all related events of relevant prepaid travel cost declarations and travel declarations) - 7,065 cases, 86,581 events

Before filtering the event logs, we will provide information about which roles performed how many activities.



Undefined	20084
Employee	13031
Supervisor	10425
Administration	9155
Budget Owner	2879
Pre Approver	772
Missing	91

Figure 1: Domestic Declarations Role-based activity table



Employee	29338
Undefined	12804
Supervisor	12535
Administration	11508
Budget Owner	3668
Pre Approver	1255
Director	897
Missing	146

Figure 2: International Declarations Role based activity table

The median of the requested amounts from International Declarations and Domestic Declarations is 79.0 euros.

2 Organizational Goals

The project's main goal is to analyze travel reimbursements to obtain important results. By using process mining tools goal is to find deviating processes. Also, optimizing and detecting the unwanted values are among the goals. Using process mining techniques to see the bottlenecks of the process.

Another goal of the organization is to prevent the overspent in cases.

In the challenge, our focus will be on the executions and decisions made by the organization following the intended reimbursement flow or not.

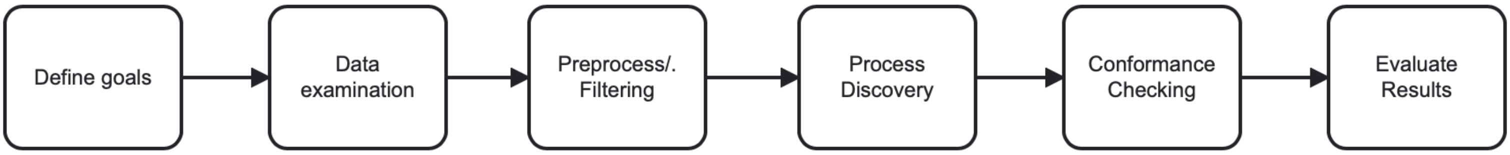


Figure 3: Goal Diagram

2.1 Questions

To understand the goals of the organization, we may look at the questions.

- What is the throughput of a travel declaration from submission (or closing) to paying?
- Is there are difference in throughput between national and international trips?
- Are there differences between clusters of declarations, for example between cost centers/departments/projec etc.?
- What is the throughput in each of the process steps, i.e. the submission, judgement by various responsible roles and payment?
- Where are the bottlenecks in the process of a travel declaration?
- Where are the bottlenecks in the process of a travel permit (note that there can be multiple requests for payment and declarations per permit)?
- How many travel declarations get rejected in the various processing steps and how many are never approved?

Then there are more detailed questions

- How many travel declarations are booked on projects?
- How many corrections have been made for declarations?
- Are there any double payments?
- Are there declarations that were not preceded properly by an approved travel permit? Or are there even declarations for which no permit exists?
- How many travel declarations are submitted by the traveler and how many by a mandated person?
- How many travel declarations are first rejected because they are submitted more than 2 months after the end of a trip and are then re-submitted?
- Is this different between departments?
- How many travel declarations are not approved by budget holders in time (7 days) and are then automatically rerouted to supervisors?
- Next to travel declarations, there are also requests for payments. These are specific for non-TU/e employees. Are there any TU/e employees that submitted a request for payment instead of a travel declaration?

2.2 Questions that answered

- Question- What is the throughput of a travel declaration from submission(or closing) to paying?

  Answer- Using Disco for the mean and median throughput time shows that 12 days with a median of 8.2 days.

- Question- Is there are difference in throughput between national and international trips?

  Answer - Using Disco for the event log we obtain mean and median throughput time for both. For the International Declarations, it is 66 days mean and 86.5 days median.

  For the Domestic Declarations it is 7.3 days median and 11.5 days mean duration.

- Question - Where are the bottlenecks in the process of a travel declaration?

  Answer - After filtering the event log by several techniques and exporting it as a .xes file, it is visualized in Disco using animation.

  As shown in the figure, we have problems with the case of the approval, especially from Supervisor, and declarations submitted from employees are also another bottleneck reason.

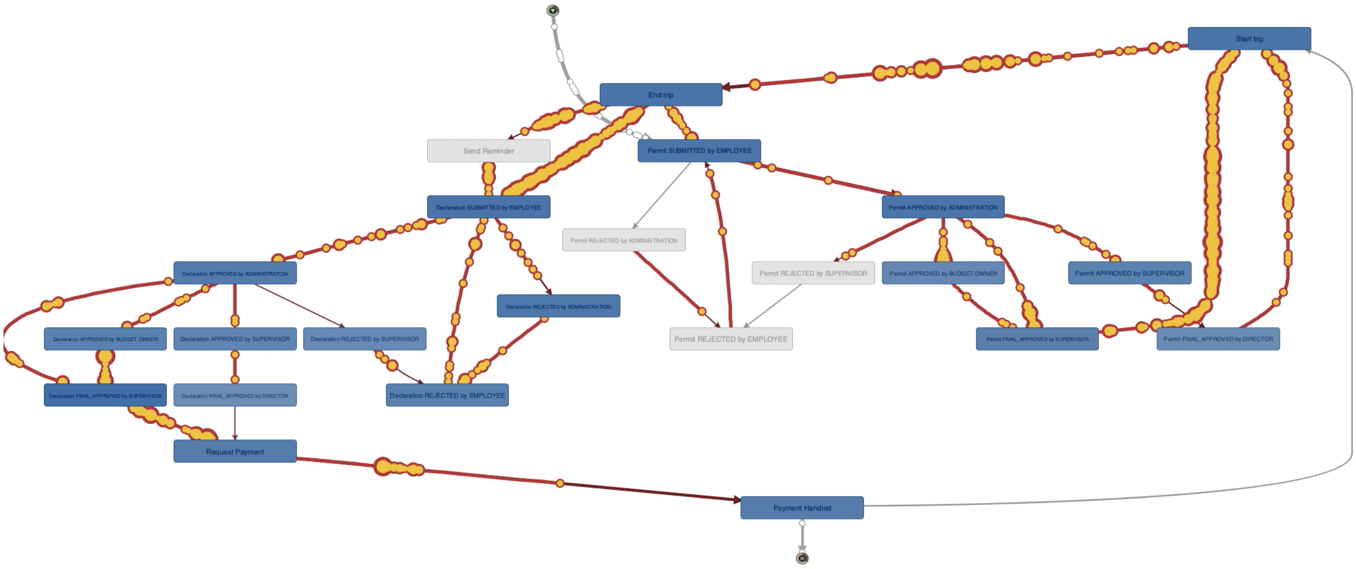

Start trip
End trip
Send Reminder
Permit SUBMITTED by EMPLOYEE
Declaration SUBMITTED by EMPLOYEE
Permit REJECTED by ADMINISTRATION
Permit APPROVED by ADMINISTRATION
Declaration APPROVED by ADMINISTRATION
Permit REJECTED by SUPERVISOR
Permit APPROVED by BUDGET OWNER
Permit APPROVED by SUPERVISOR
Declaration REJECTED by ADMINISTRATION
Declaration APPROVED by BUDGET OWNER
Declaration APPROVED by SUPERVISOR
Declaration REJECTED by SUPERVISOR
Permit REJECTED by EMPLOYEE
Permit FINAL_APPROVED by SUPERVISOR
Permit FINAL_APPROVED by DIRECTOR
Declaration FINAL_APPROVED by SUPERVISOR
Declaration FINAL_APPROVED by DIRECTOR
Declaration REJECTED by EMPLOYEE
Request Payment
Payment Handled

Figure 4: International Declaration Bottleneck analysis

- Question - Where are the bottlenecks in the process of a travel permit (note that there can be multiple requests for payment and declarations per permit)?

  Answer - As shown in the figure, a bottleneck occurs in the declaration approved by the budget owner and final approve by the supervisor.

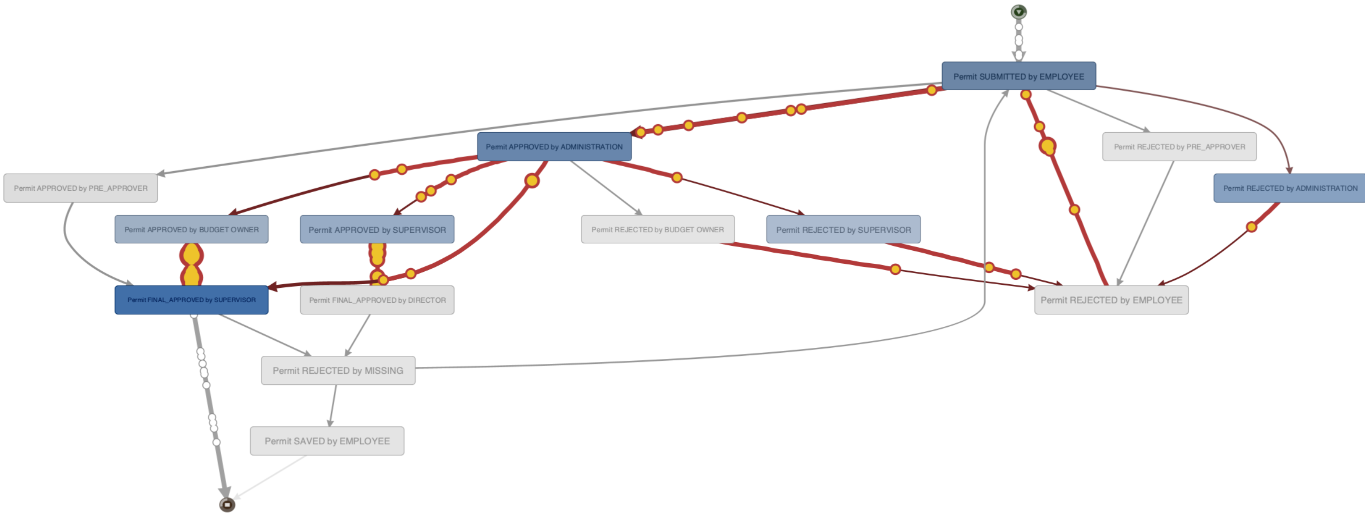

Permit SUBMITTED by EMPLOYEE
Permit APPROVED by ADMINISTRATION
Permit REJECTED by PRE_APPROVER
Permit REJECTED by ADMINISTRATION
Permit APPROVED by PRE_APPROVER
Permit APPROVED by BUDGET OWNER
Permit APPROVED by SUPERVISOR
Permit REJECTED by BUDGET OWNER
Permit REJECTED by SUPERVISOR
Permit FINAL_APPROVED by SUPERVISOR
Permit FINAL_APPROVED by DIRECTOR
Permit REJECTED by EMPLOYEE
Permit REJECTED by MISSING
Permit SAVED by EMPLOYEE

Figure 5: PermitLog Bottleneck analysis

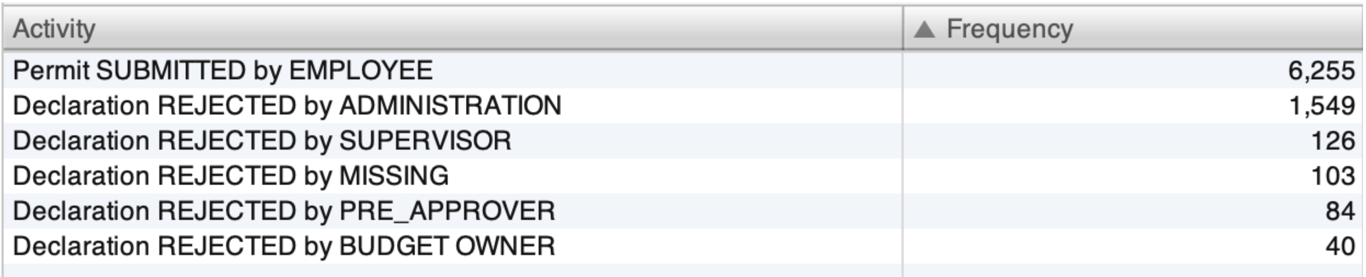
- Question - Are there any double payments?

  Answer - Used function for the checking double payments made or not in the events from Domestic, International, and Permit Logs.

  For the Domestic and International, there is no double payment, but for the permit log, we have 1233 doubles.

- Question - How many travel declarations get rejected in the various processing steps, and how many are never approved?

  Answer - Using Disco for this question We use the filtering Attribute method provided by Disco. Selected only the rejection events,



Activity	▲ Frequency
Permit SUBMITTED by EMPLOYEE	6,255
Declaration REJECTED by ADMINISTRATION	1,549
Declaration REJECTED by SUPERVISOR	126
Declaration REJECTED by MISSING	103
Declaration REJECTED by PRE_APPROVER	84
Declaration REJECTED by BUDGET OWNER	40

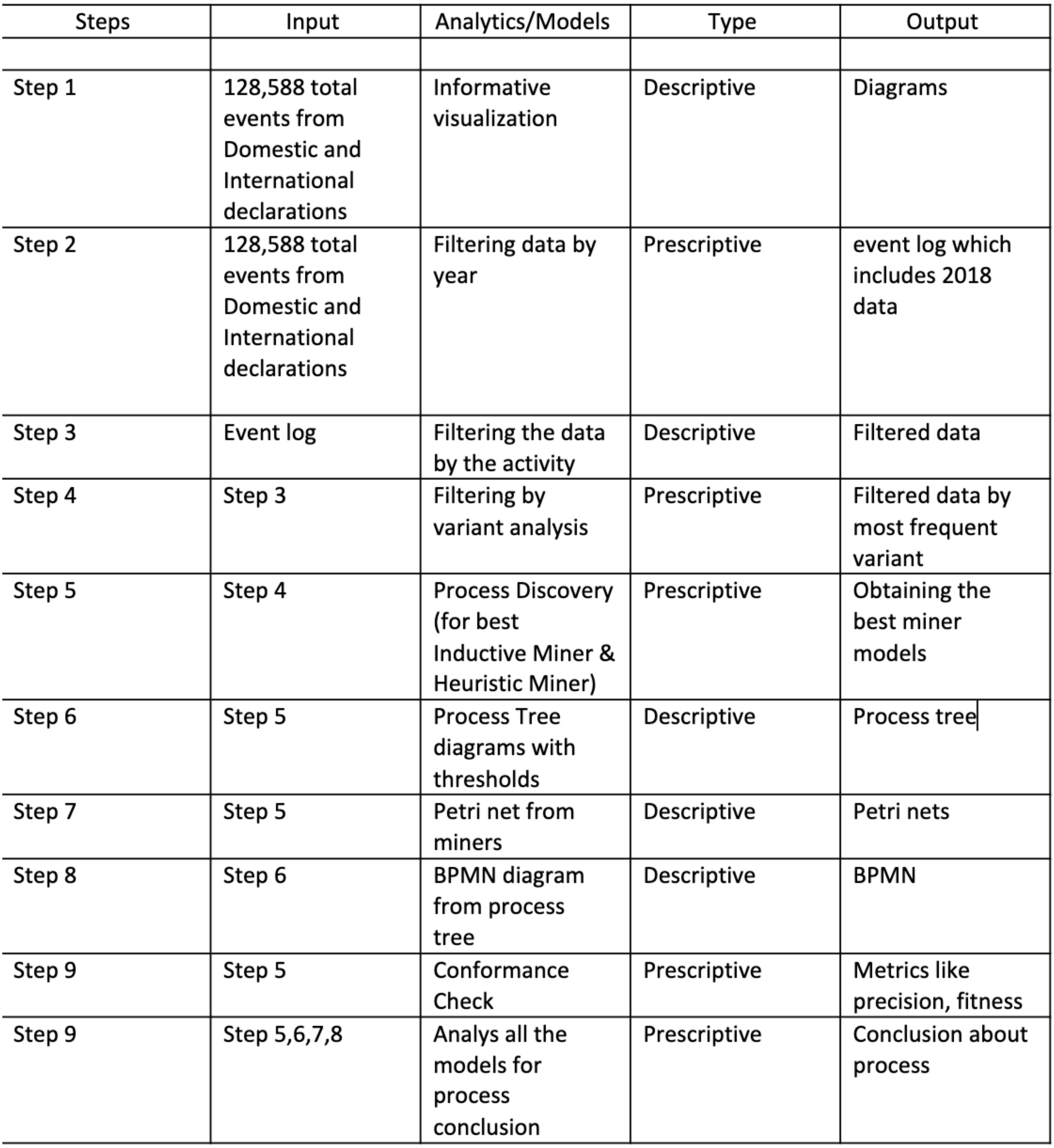
Figure 6: Absolute Frequency of rejections

3 Knowledge Uplift Trail

Before starting process mining, we will filter the data for accurate results. In the KUT part, the steps mentioned in the table are related to International Declarations. In this part, we are using both the pm4py library and the Disco program. Disco will be used mostly for the visualization, and pm4py is used for mostly the process discovery

Descriptive Analytics tells you what happened in the past.

Prescriptive Analytics recommends actions you can take to affect those outcomes.



Steps	Input	Analytics/Models	Type	Output
Step 1	128,588 total events from Domestic and International declarations	Informative visualization	Descriptive	Diagrams
Step 2	128,588 total events from Domestic and International declarations	Filtering data by year	Prescriptive	event log which includes 2018 data
Step 3	Event log	Filtering the data by the activity	Descriptive	Filtered data
Step 4	Step 3	Filtering by variant analysis	Prescriptive	Filtered data by most frequent variant
Step 5	Step 4	Process Discovery (for best Inductive Miner & Heuristic Miner)	Prescriptive	Obtaining the best miner models
Step 6	Step 5	Process Tree diagrams with thresholds	Descriptive	Process tree
Step 7	Step 5	Petri net from miners	Descriptive	Petri nets
Step 8	Step 6	BPMN diagram from process tree	Descriptive	BPMN
Step 9	Step 5	Conformance Check	Prescriptive	Metrics like precision, fitness
Step 9	Step 5,6,7,8	Analys all the models for process conclusion	Prescriptive	Conclusion about process

Figure 7: KUT Table

4 Project Results

First we used "xes_importer" to import xes files to EventLog object.

4.1 Filtering

In the analysis of the data, We filtered years by only 2018 and 2019 using the pm4py filter time range function. The reason behind the filtering year is the process in the 2017 year is not stable.

The BPI Challenge 2020 website also specified that 2017 data contains only two departments/ [2]

After converting the event log to the data, the frame dropped the UNKNOWN values from cases DeclarationNumber, Permit travel permit number, Permit TaskNumber.

The length of the international declaration becomes 2901 after filtering. We also filtered by start and end activities.

4.2 Variants

To reduce the complexity of the process in the University, we use variation analysis and defined groups like the Duration, Resource, Activity and activity list.

Feature	Value
Total case	2901
Handled Cases	2900
Ratio of Handled Cases	0.99
Rejected Cases	759
Double payment for International Declarations	0
Double payment for Domestice Declarations	0

The number of variants for Domestic Declaration is 99.

The number of variants for the International Declaration is 753. (Using Disco with unfiltered data.)

Checked the handled ratio of the event log after filtering and obtained a high ratio, meaning the filter and the data were clean.

Total handled 2900 and total rejection 759

Parameter	Value
Mean	36
Median	11
Mode	1

Table 1: Variant Distribution

By using variants and their frequency, we plotted a variants frequency plot.

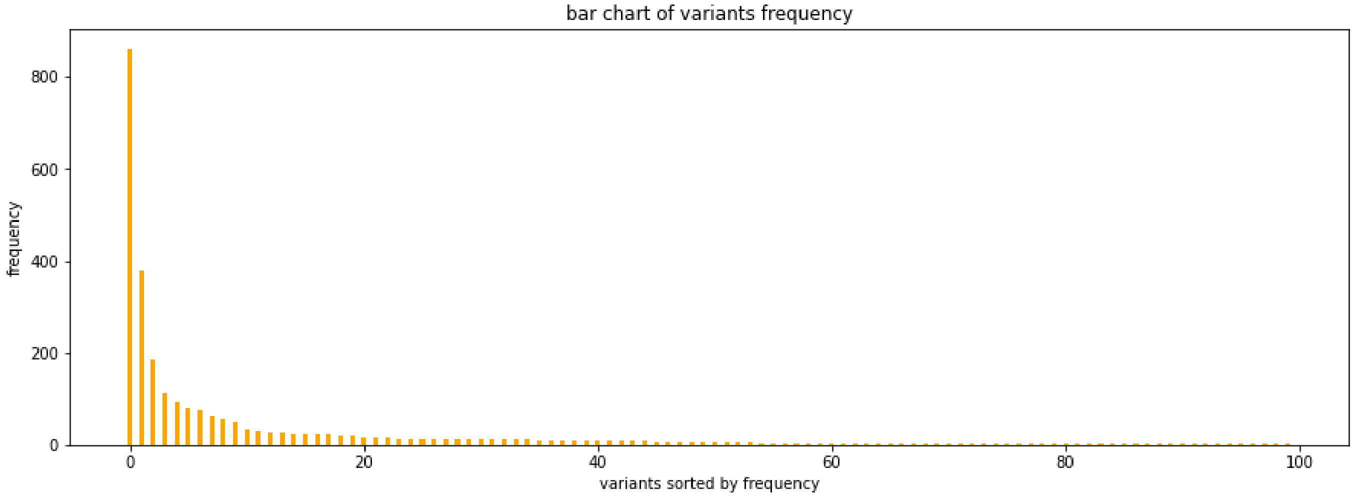


Figure 8: Variant Frequency Table

Mean duration is 71 days from start to end.

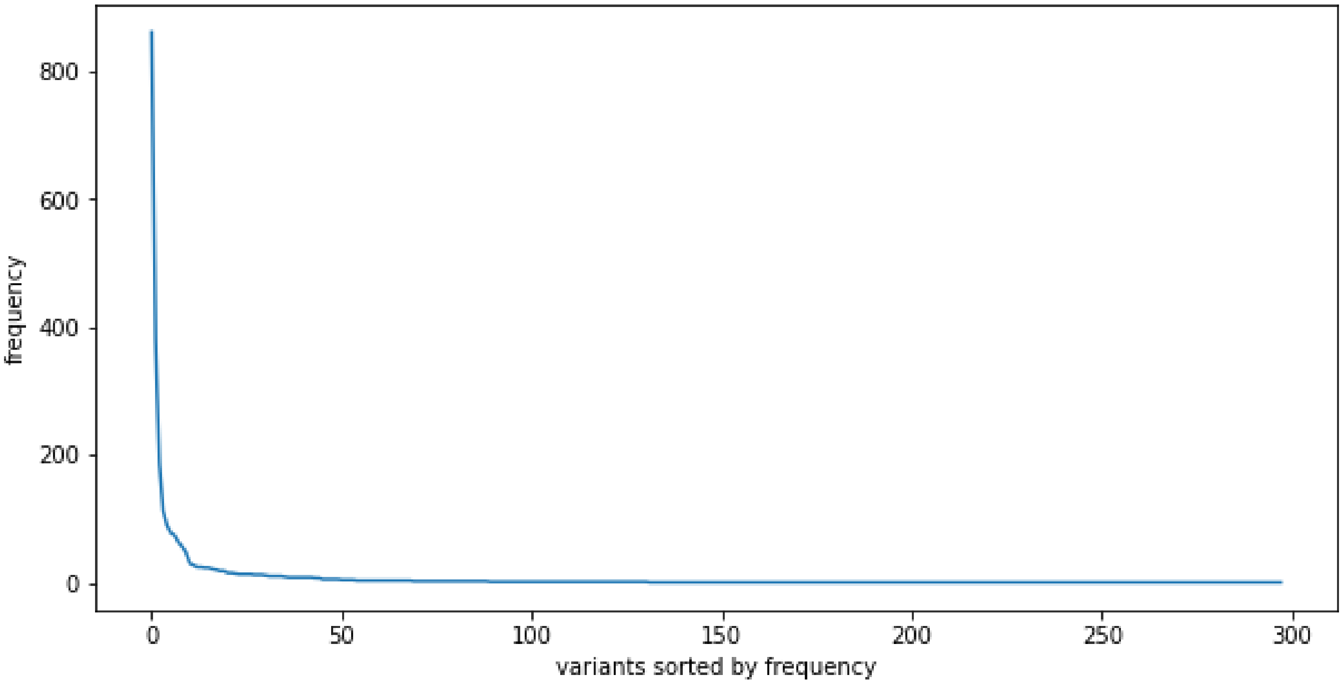


Figure 9: Distributions of variants

After filtering by 100, we are getting a more detailed distribution.

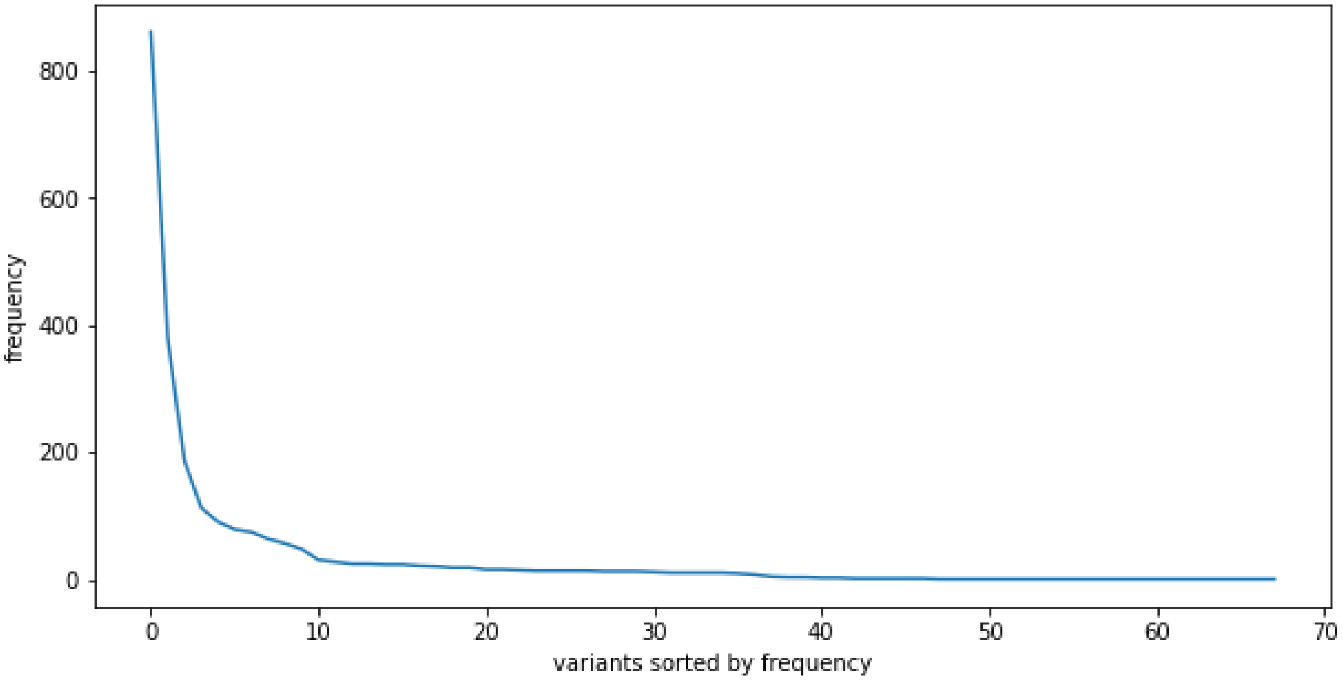


Figure 10: Filtered variant distribution

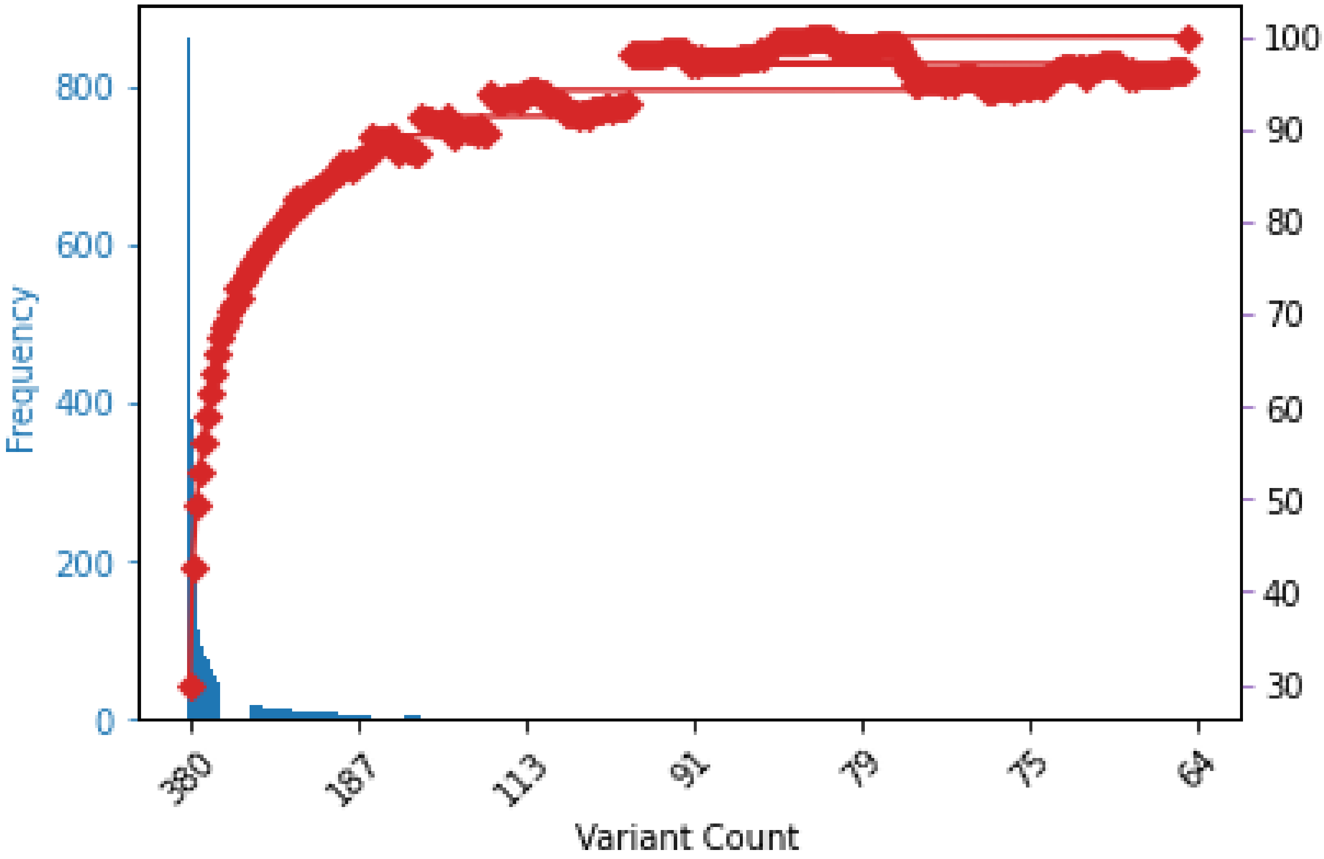


Figure 11: Pareto distribution of variants

Pareto principle is basically saying that 80 percent of the output from a given situation or system is determined by 20 percent of the input.

The red marks similar to the line represent the cumulative percentage of the variants. And the tiny blue bars representing the variants count. We do have not well-distributed variants. Most of the variants distributed on the first 1 or 2 variants.

The first variant with count 861 is:

Permit SUBMITTED by EMPLOYEE,Permit APPROVED by ADMINISTRATION,Permit FINAL_APPR by SUPERVISOR,Start trip,End trip,Declaration SUBMITTED by EMPLOYEE,Declaration APPROVED by ADMINISTRATION,Declaration FINAL_APPROVED by SUPERVISOR,Request Payment,Payment Handled

4.3 Process Discovery

In the dataset, we have two different variants: Domestic Declarations and International Declarations. During the project, we are mostly mining the International Declarations. The reason for using International Declarations is that all the declarations permit is required for international travels. Also, a person can request before the travel. This feature makes International travel more complex. This is the reason why we are using the International Declarations.

Before starting the process of discovery, we have 290 variants. Using Disco for the Domestic declarations, we can see the process is not that complex considering the 2 variants.

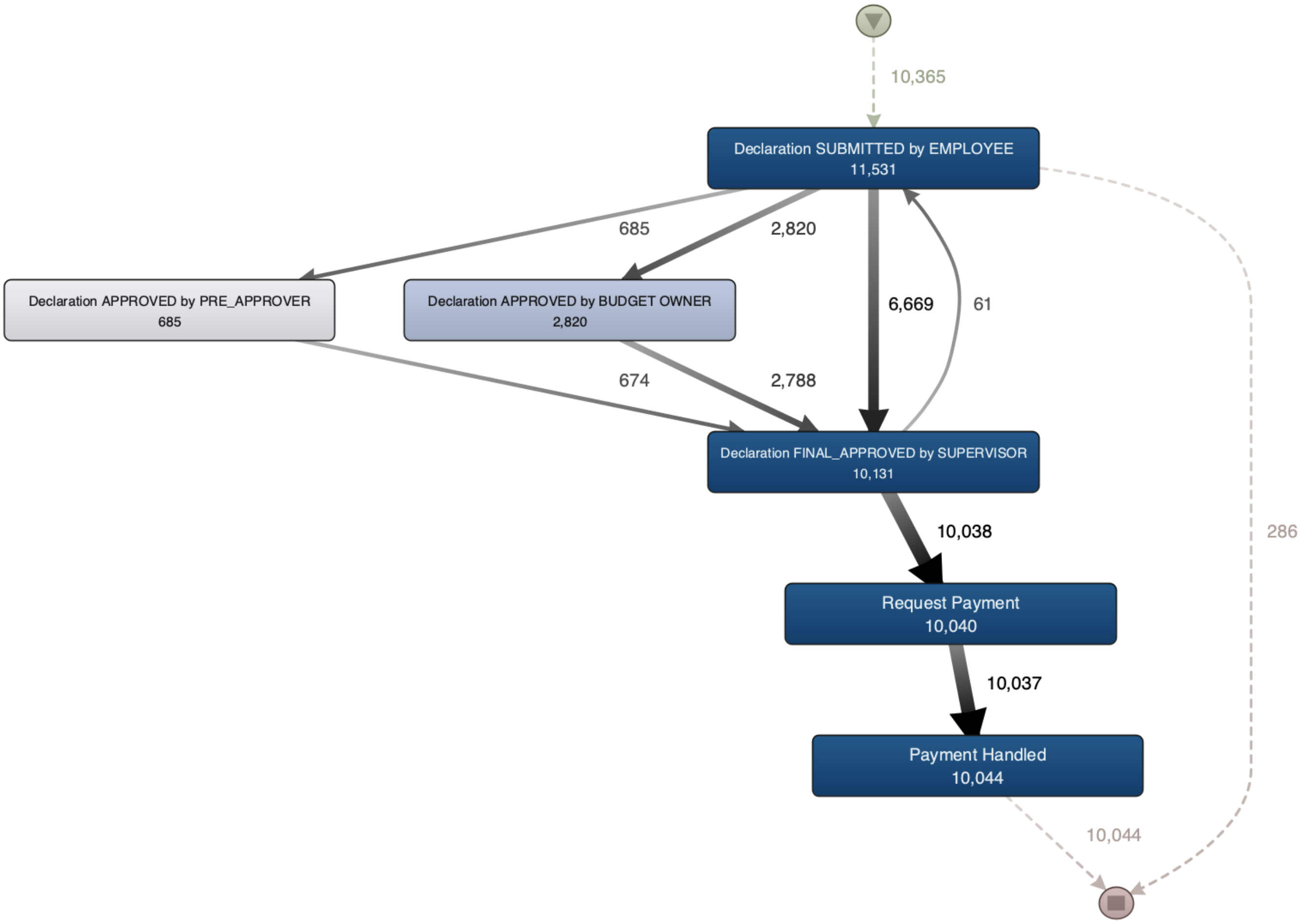

10,365
Declaration SUBMITTED by EMPLOYEE
11,531
685
2,820
Declaration APPROVED by PRE_APPROVER
685
Declaration APPROVED by BUDGET OWNER
2,820
6,669
61
674
2,788
Declaration FINAL_APPROVED by SUPERVISOR
10,131
10,038
286
Request Payment
10,040
10,037
Payment Handled
10,044
10,044

Figure 12: Domestic Declarations with 2 variants

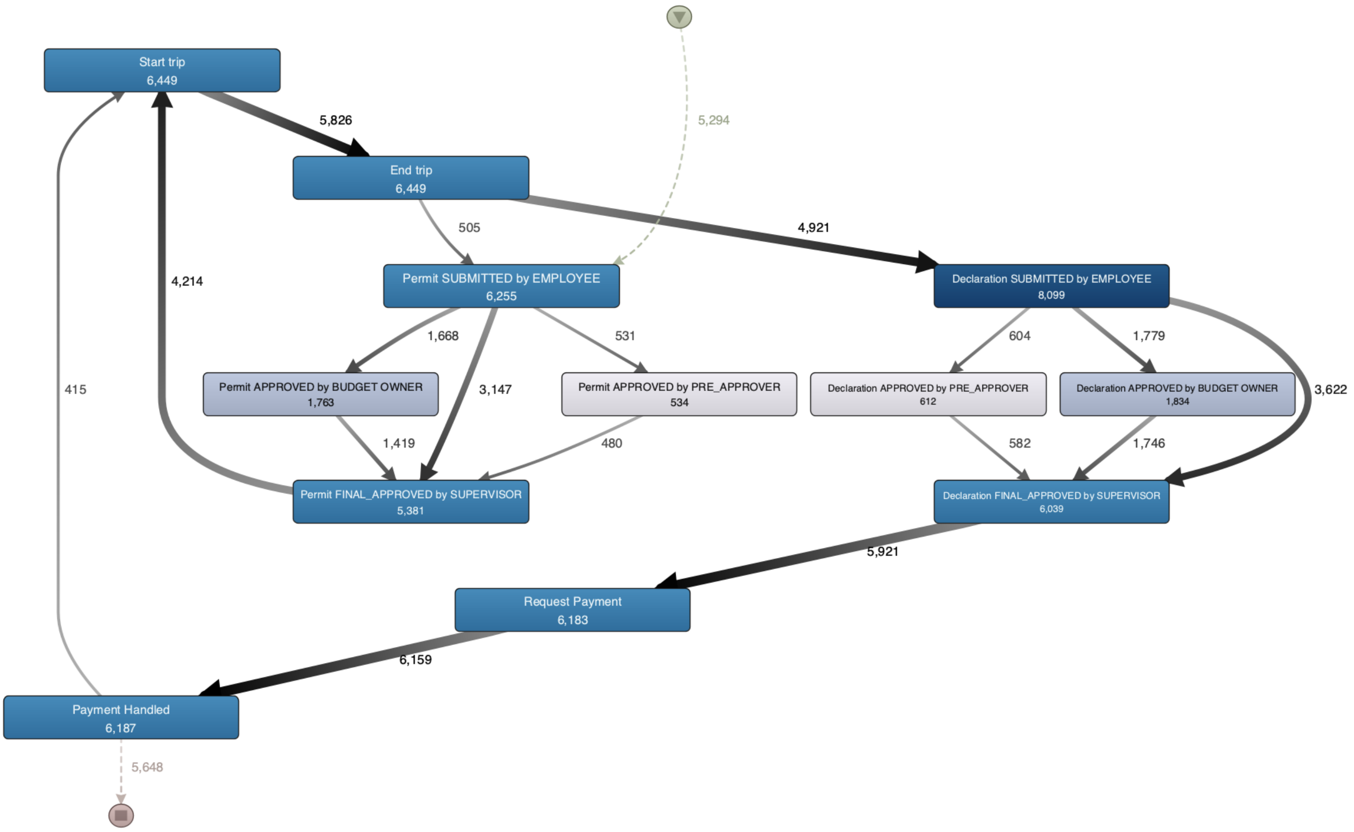

Start trip
6,449
5,826
End trip
6,449
5,294
505
4,921
4,214
Permit SUBMITTED by EMPLOYEE
6,255
Declaration SUBMITTED by EMPLOYEE
8,099
1,668
531
604
1,779
415
3,147
3,622
Permit APPROVED by BUDGET OWNER
1,763
Permit APPROVED by PRE_APPROVER
534
Declaration APPROVED by PRE_APPROVER
612
Declaration APPROVED by BUDGET OWNER
1,834
1,419
480
582
1,746
Permit FINAL_APPROVED by SUPERVISOR
5,381
Declaration FINAL_APPROVED by SUPERVISOR
6,039
5,921
Request Payment
6,183
6,159
Payment Handled
6,187
5,648

Figure 13: International Declarations same variants with Domestic and Permit cases

Permits that need final approval by the supervisor take so long. This kind of long duration can be considered a bottleneck for the organization.

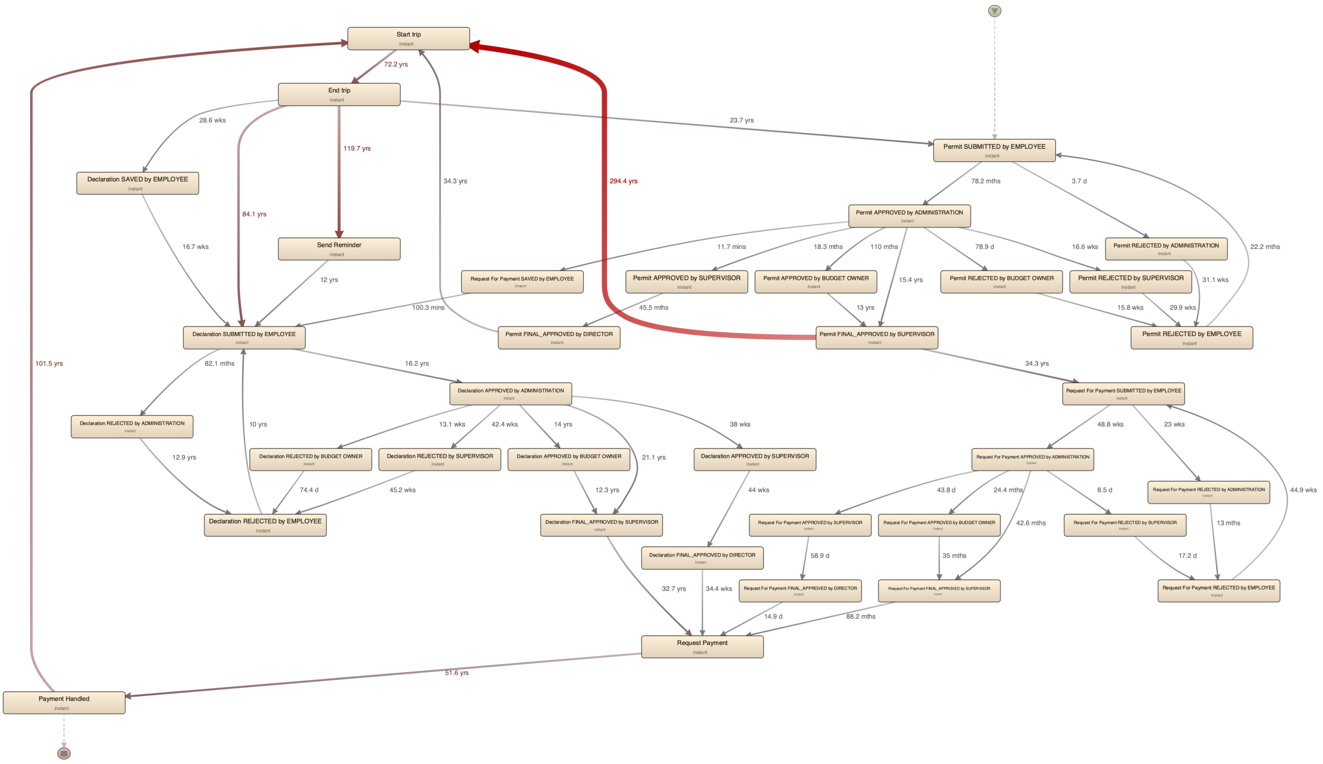
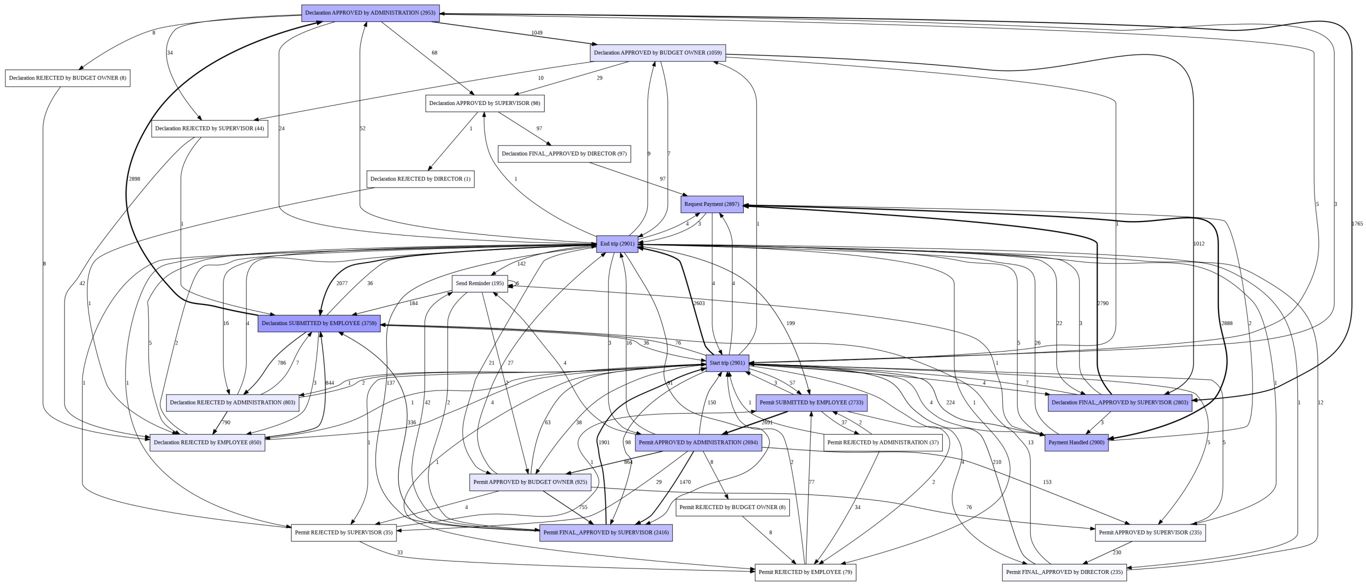

Start trip
instant
72.2 yrs
End trip
instant
23.7 yrs
Permit SUBMITTED by EMPLOYEE
instant
28.6 wks
119.7 yrs
34.3 yrs
294.4 yrs
78.2 mths
3.7 d
Declaration SAVED by EMPLOYEE
instant
84.1 yrs
Permit APPROVED by ADMINISTRATION
instant
101.5 yrs
Send Reminder
instant
11.7 mins
18.3 mths
110 mths
78.9 d
16.6 wks
Permit REJECTED by ADMINISTRATION
instant
22.2 mths
16.7 wks
12 yrs
Request For Payment SAVED by EMPLOYEE
instant
Permit APPROVED by SUPERVISOR
instant
Permit APPROVED by BUDGET OWNER
instant
15.4 yrs
Permit REJECTED by BUDGET OWNER
instant
Permit REJECTED by SUPERVISOR
instant
31.1 wks
100.3 mins
45.5 mths
13 yrs
15.8 wks
29.9 wks
Declaration SUBMITTED by EMPLOYEE
instant
Permit FINAL_APPROVED by DIRECTOR
instant
Permit FINAL_APPROVED by SUPERVISOR
instant
Permit REJECTED by EMPLOYEE
instant
82.1 mths
16.2 yrs
34.3 yrs
Declaration APPROVED by ADMINISTRATION
instant
Request For Payment SUBMITTED by EMPLOYEE
instant
Declaration REJECTED by ADMINISTRATION
instant
10 yrs
13.1 wks
42.4 wks
14 yrs
38 wks
48.8 wks
23 wks
12.9 yrs
Declaration REJECTED by BUDGET OWNER
instant
Declaration REJECTED by SUPERVISOR
instant
Declaration APPROVED by BUDGET OWNER
instant
21.1 yrs
Declaration APPROVED by SUPERVISOR
instant
Request For Payment APPROVED by ADMINISTRATION
instant
74.4 d
45.2 wks
12.3 yrs
44 wks
43.8 d
24.4 mths
8.5 d
Request For Payment REJECTED by ADMINISTRATION
instant
44.9 wks
Declaration REJECTED by EMPLOYEE
instant
Declaration FINAL_APPROVED by SUPERVISOR
instant
Request For Payment APPROVED by SUPERVISOR
instant
Request For Payment APPROVED by BUDGET OWNER
instant
42.6 mths
Request For Payment REJECTED by SUPERVISOR
instant
13 mths
Declaration FINAL_APPROVED by DIRECTOR
instant
58.9 d
35 mths
17.2 d
32.7 yrs
34.4 wks
Request For Payment FINAL_APPROVED by DIRECTOR
instant
Request For Payment FINAL_APPROVED by SUPERVISOR
instant
Request For Payment REJECTED by EMPLOYEE
instant
14.9 d
88.2 mths
Request Payment
instant
51.6 yrs
Payment Handled
instant

Figure 14: Total Duration (Mean+Median) using Disco for the PermitLog

4.3.1 DFG Discovery

A Directly-Follows Graph (DFG) is the simplest representation of the process models. In a directly-follows graph, each node represents an activity, and the arcs describe the relationship between various activities. Typically in a process model, the directly-follows graph has a source and sink representing the start and end activities. An arc in the directly-follows graph between any two activities represents that the source activity is directly followed by the sink activity in the event log. [6]



Declaration APPROVED by ADMINISTRATION (2953)
Declaration APPROVED by BUDGET OWNER (1059)
Declaration REJECTED by BUDGET OWNER (8)
Declaration APPROVED by SUPERVISOR (98)
Declaration REJECTED by SUPERVISOR (44)
Declaration FINAL_APPROVED by DIRECTOR (97)
Declaration REJECTED by DIRECTOR (1)
Request Payment (2887)
End trip (2901)
Send Reminder (195)
Declaration SUBMITTED by EMPLOYEE (3739)
Start trip (2901)
Declaration REJECTED by ADMINISTRATION (803)
Permit SUBMITTED by EMPLOYEE (2733)
Declaration FINAL_APPROVED by SUPERVISOR (2803)
Declaration REJECTED by EMPLOYEE (850)
Permit APPROVED by ADMINISTRATION (2694)
Permit REJECTED by ADMINISTRATION (37)
Payment Handled (2900)
Permit APPROVED by BUDGET OWNER (925)
Permit REJECTED by BUDGET OWNER (8)
Permit REJECTED by SUPERVISOR (35)
Permit FINAL_APPROVED by SUPERVISOR (2416)
Permit APPROVED by SUPERVISOR (235)
Permit REJECTED by EMPLOYEE (79)
Permit FINAL_APPROVED by DIRECTOR (235)

Figure 15: Directly Follows Graph

4.3.2 Alpha Miner

We have really complicated Petri net for the Alpha miner. Alpha miner can give us process models in the form of Peti net. A bad feature of the Alpha Miner is can not handle noise so well.

Another bad feature is invisible, and duplicated tasks can not discover

We do not have a simple Petri net because we have multiple duplicated tasks.

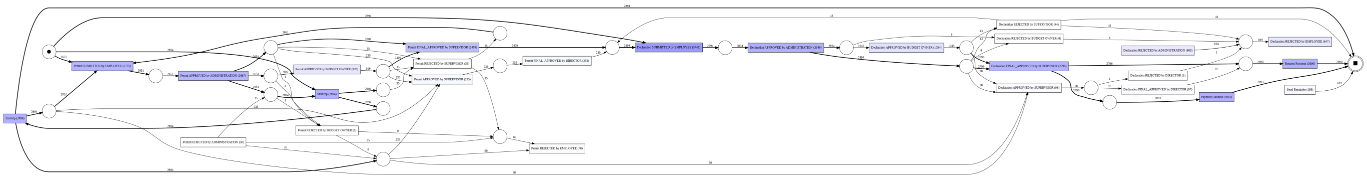


Figure 16: ALPHA Miner Petri Net

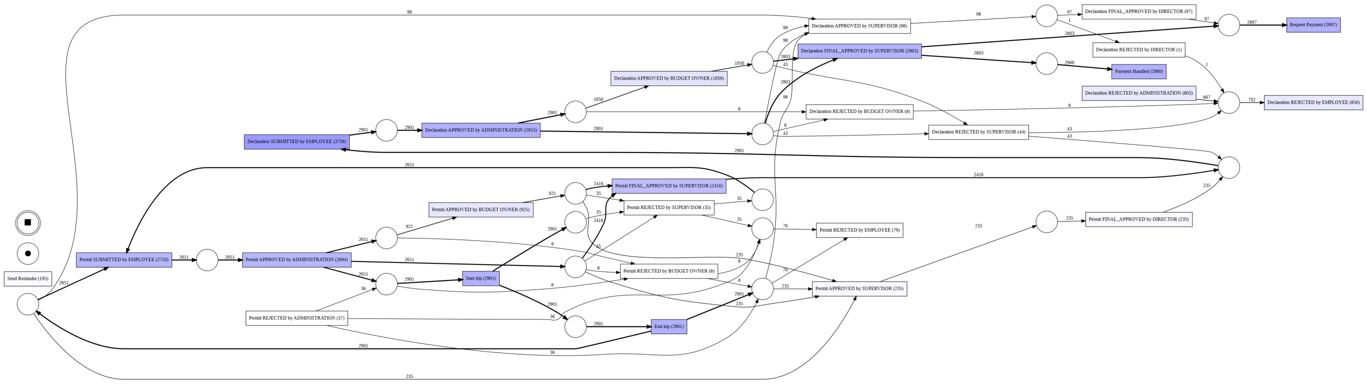

Declaration APPROVED by SUPERVISOR (98)
Declaration FINAL_APPROVED by DIRECTOR (97)
Request Payment (3897)
Declaration FINAL_APPROVED by SUPERVISOR (2803)
Declaration REJECTED by DIRECTOR (1)
Declaration APPROVED by BUDGET OWNER (1050)
Payment Handled (2900)
Declaration REJECTED by ADMINISTRATION (803)
Declaration REJECTED by EMPLOYEE (850)
Declaration SUBMITTED by EMPLOYEE (3700)
Declaration APPROVED by ADMINISTRATION (2910)
Declaration REJECTED by BUDGET OWNER (8)
Declaration REJECTED by SUPERVISOR (44)
Permit FINAL_APPROVED by SUPERVISOR (2416)
Permit REJECTED by SUPERVISOR (35)
Permit APPROVED by BUDGET OWNER (925)
Permit REJECTED by EMPLOYEE (79)
Permit FINAL_APPROVED by DIRECTOR (235)
Send Reminder (195)
Permit SUBMITTED by EMPLOYEE (2733)
Permit APPROVED by ADMINISTRATION (2684)
Start trip (2901)
Permit REJECTED by BUDGET OWNER (8)
Permit APPROVED by SUPERVISOR (235)
Permit REJECTED by ADMINISTRATION (37)
End trip (2901)

Figure 17: ALPHA Miner WITHOUT PROPER FILTERS

4.3.3 Heuristic Miner

Heuristic miner is a good way to handle noise and common constructs. Heuristic miner has its own graph called Heuristics Net, but we will provide Petri net by converting.

Features of the Heuristic miner are that it can consider the frequency, detect short loops provided, and look for skipped events.

The HeuristicsMiner Plug-in mines the control-flow perspective of a process model. To do so, it only considers the order of the events within a case. In other words, the order of events among cases isn't important.[3]

Also, HeuristicsMiner can deal with noise and low-frequency behavior [3]

By increasing the threshold, we can remove instances with a low frequency.

Since we increased the threshold to 0.999, we must also be careful about removing the relevant information.

Dependency measure threshold: minimum value of dependency between events.

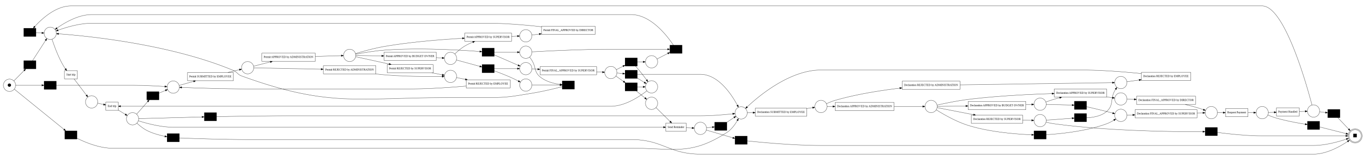

Start trip
End trip
Permit SUBMITTED by EMPLOYEE
Permit APPROVED by ADMINISTRATION
Permit REJECTED by ADMINISTRATION
Permit APPROVED by SUPERVISOR
Permit APPROVED by BUDGET OWNER
Permit REJECTED by SUPERVISOR
Permit FINAL_APPROVED by DIRECTOR
Permit REJECTED by EMPLOYEE
Permit FINAL_APPROVED by SUPERVISOR
Send Reminder
Declaration SUBMITTED by EMPLOYEE
Declaration APPROVED by ADMINISTRATION
Declaration REJECTED by ADMINISTRATION
Declaration APPROVED by SUPERVISOR
Declaration APPROVED by BUDGET OWNER
Declaration REJECTED by SUPERVISOR
Declaration REJECTED by EMPLOYEE
Declaration FINAL_APPROVED by DIRECTOR
Declaration FINAL_APPROVED by SUPERVISOR
Request Payment
Payment Handled

Figure 18: Heuristic Miner process tree with 0.7 dependency threshold

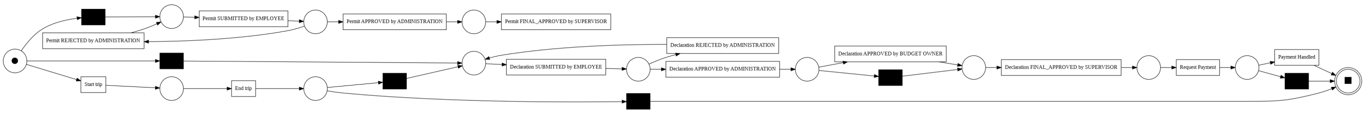

Permit SUBMITTED by EMPLOYEE
Permit APPROVED by ADMINISTRATION
Permit FINAL_APPROVED by SUPERVISOR
Permit REJECTED by ADMINISTRATION
Declaration REJECTED by ADMINISTRATION
Declaration APPROVED by BUDGET OWNER
Declaration SUBMITTED by EMPLOYEE
Declaration APPROVED by ADMINISTRATION
Declaration FINAL_APPROVED by SUPERVISOR
Request Payment
Payment Handled
Start trip
End trip

Figure 19: Heuristic Miner process tree with 0.999 dependency threshold

4.3.4 Inductive Miner

The best part of the Inductive miner is it is good for finding permanent split. After finding the split algorithm look for the sub logs until the base case is found. Doing this by kind of divide-and-conquer perspective

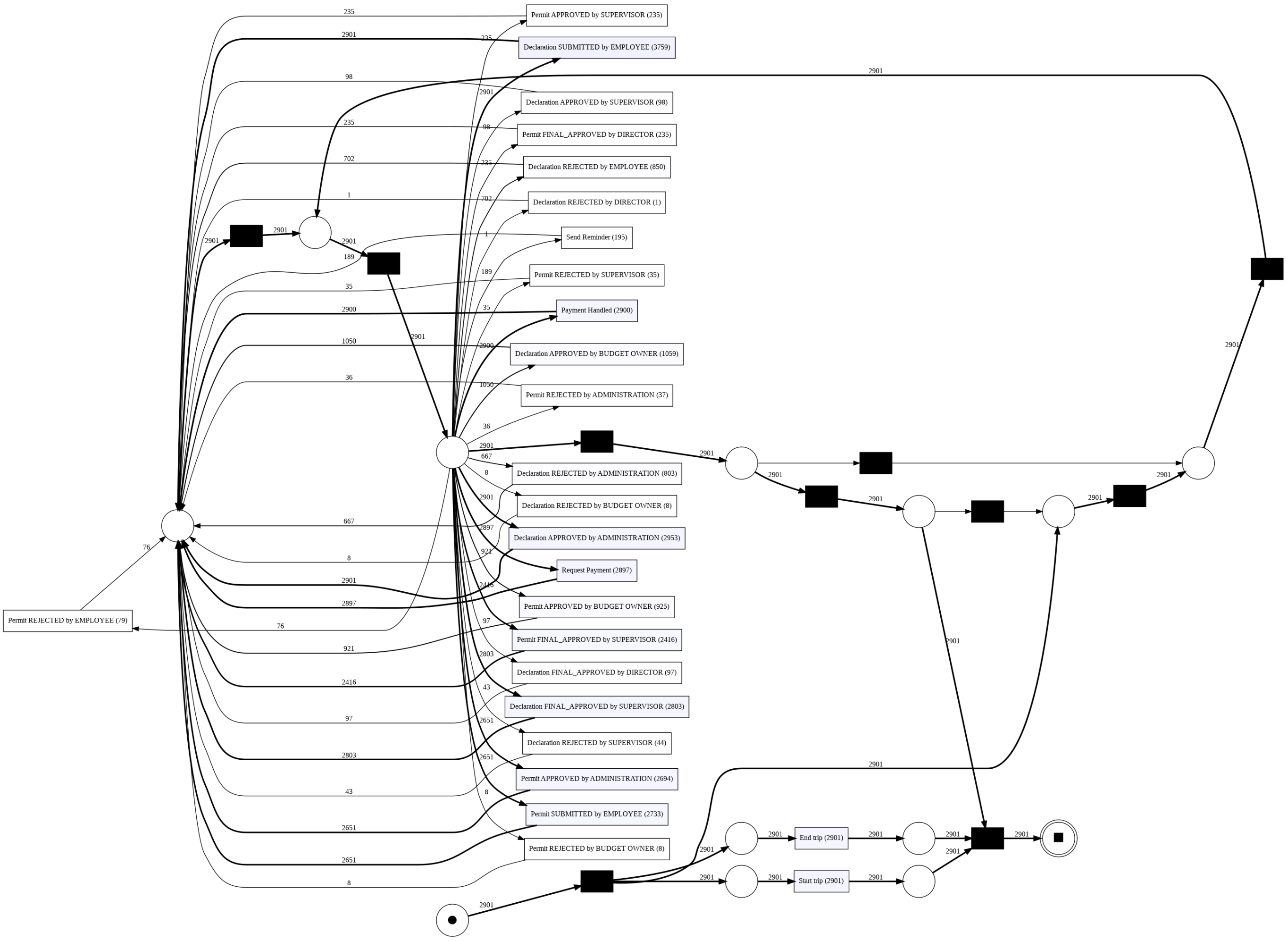

Permit APPROVED by SUPERVISOR (235)
Declaration SUBMITTED by EMPLOYEE (3759)
Declaration APPROVED by SUPERVISOR (98)
Permit FINAL_APPROVED by DIRECTOR (235)
Declaration REJECTED by EMPLOYEE (850)
Declaration REJECTED by DIRECTOR (1)
Send Reminder (195)
Permit REJECTED by SUPERVISOR (35)
Payment Handled (2900)
Declaration APPROVED by BUDGET OWNER (1059)
Permit REJECTED by ADMINISTRATION (37)
Declaration REJECTED by ADMINISTRATION (803)
Declaration REJECTED by BUDGET OWNER (8)
Declaration APPROVED by ADMINISTRATION (2953)
Request Payment (2897)
Permit APPROVED by BUDGET OWNER (925)
Permit FINAL_APPROVED by SUPERVISOR (2416)
Declaration FINAL_APPROVED by DIRECTOR (97)
Declaration FINAL_APPROVED by SUPERVISOR (2803)
Declaration REJECTED by SUPERVISOR (44)
Permit APPROVED by ADMINISTRATION (2694)
Permit SUBMITTED by EMPLOYEE (2733)
Permit REJECTED by BUDGET OWNER (8)
Permit REJECTED by EMPLOYEE (79)
End trip (2901)
Start trip (2901)

Figure 20: Inductive Miner Petri Net

4.4 Conformance Check

Conformance checking techniques need an event log and a model as their input. The output consists of diagnostic information showing differences and commonalities between the model and its log. Conformance checking compares a process model concerning the event log with four main quality dimensions: fitness, simplicity, precision, and generalization. [1]

Conformance checking is done to define similarity and dissimilarity between process model behavior and data behavior. Fitness calculates behavior proportion on event logs in the model[10]

Fitness - the discovered model should allow for the behavior seen in the event log. Typical use case auditing (avoiding "non-fitting" behavior) .

Fitness quantifies the extent to which the discovered model can accurately reproduce the cases recorded in the log. [4]

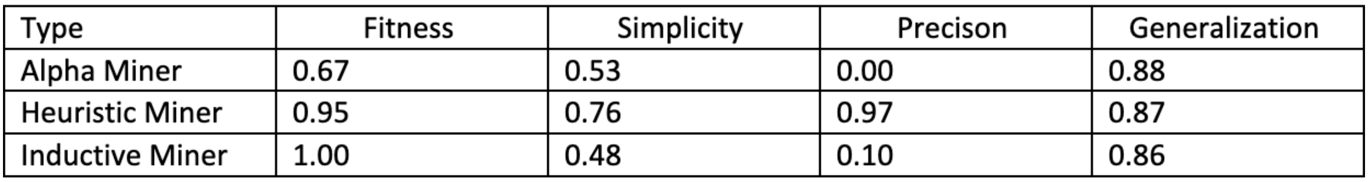
A discovered model with good fitness should allow the replay of (most of) the behavior seen in the event log [5]

Precision - the discovered model should not allow for behavior completely unrelated to what was seen in the event log. The typical use case is optimization. (avoiding "underfitting").

Precision is trivial to discover a simple process model that can reproduce the event log.

Generalization - the discovered model should generalize the example the behavior is seen in the event log. Typical use case implementation (avoiding "overfitting")

Simplicity - the discovered model should not be unnecessarily complex. Typical use case human readability.



Type	Fitness	Simplicity	Precison	Generalization
Alpha Miner	0.67	0.53	0.00	0.88
Heuristic Miner	0.95	0.76	0.97	0.87
Inductive Miner	1.00	0.48	0.10	0.86

Figure 21: Conformance Check Analysis table

4.4.1 Comments on the values

With 0.95 and 1.00 values on the **Fitness** this means that it will allow the replay of behavior seen in the event log.

We have 0 precision on the table because Precision equals the intersection of the model's behavior and the Event log divided by the model behavior. And by 0, it means that there is no common behavior with Model and Event Log

Also, with 0 Precision, we can not reproduce the event log.

We have good **generalization** among all the types, meaning all model subsets are frequently visited.

5 Conclusion

In the challenge, we used process discovery and variant analysis to answer the challenge questions and find the bottlenecks for improving the process.

International declarations focused more than the other event logs, And another reason is there were too many variants compared to the domestic declarations. In the process discovery, we considered only the International declaratons.

Then we used conformance analysis to understand which mining type was better suited for our data. Heuristic miner with 0.95 fitness and 0.97 precision gave the best values. Alpha miner gave the worst result with 0.67 fitness and 0.00 precision. On the other hand, Inductive Miner gave simplicity by 0.48 and 0.86 for the generalization.

Results of the conformance check gave us the result that we still have many variants.

As shown in figure 3, we have bottlenecks related to approvals from budget owners. So the solution for a better process is to reduce the approval time. Or with the same approval time with fewer cases.

References

[1] - TEM Journal. Volume 8, Issue 4, Pages 1232-1241, ISSN 2217-8309, DOI: 10.18421/TEM84-18, November 2019.

[2] - https://icpmconference.org/2020/bpi-challenge/

[3] - http://www.padsweb.rwth-aachen.de/wvdaalst/publications/p314.pdf

[4] https://link.springer.com/chapter/10.1007/978-3-642-33606-5_19

[5] - W.M.P.VanderAalst,A.Adriansyah,A.K.A.deMedeiros,F.Arcieri,T.Baier, T. Blickle, J. C. Bose, P. Van den Brand, R. Brandtjen, J. Buijs, A. Burat- tin, J. Carmona, M. Castellanos, J. Claes, J. Cook, N. Costantini, F. Curbera, E. Damiani, M. de Leoni, P. Delias, B. F. Van Dongen, M. Dumas, S. Dustdar, D. Fahland, D. R. Ferreira, W. Gaaloul, F. Van Geffen, S. Goel, C. Gu nther, A. Guzzo, P. Harmon, A. ter Hofstede, J. Hoogland, J. E. Ingvaldsen, K. Kato, R. Kuhn, A. Kumar, M. La Rosa, F. Maggi, D. Malerba, R. S. Mans, A. Manuel, M. McCreesh, P. Mello, J. Mendling, M. Montali, H. R. Motahari-Nezhad, M. zur Muehlen, J. Munoz-Gama, L. Pontieri, J. Ribeiro, A. Rozinat, H. S. Prez, R. S. Prez, M. Seplveda, J. Sinur, P. Soffer, M. Song, A. Sperduti, G. Stilo, C. Stoel, K. Swenson, M. Talamo, W. Tan, C. Turner, J. Vanthienen, G. Varvaressos, E. Verbeek, M. Verdonk, R. Vigo, J. Wang, B. Weber, M. Weidlich, T. Weijters, L. Wen, M. Westergaard, and M. Wynn. Process Mining Manifesto. In Busi- ness Process Management Workshops, number 99 in Lecture Notes in Business Information Processing, pages 169–194. Springer Berlin Heidelberg, 2012.

[6] - http://processmining.org/process-discovery.html: :text=A%20Directly%2DFollows%20Graph%20(DFG,

[10] - https://www.researchgate.net/publication/301998930_Implementing_Heuristic_Miner_for_Different

[-] - https://icpmconference.org/2020/wp-content/uploads/sites/4/2020/10/ICPM_2020_paper_148.pdf

[-] - https://icpmconference.org/2020/wp-content/uploads/sites/4/2020/10/ICPM_2020_paper_44.pdf