Business Process Intelligence: Assignment 2

Vincent Janssen
382962

June 28, 2018

Contents

1 Introduction

In this second assignment of Business Process Intelligence I will help a financial institute. The financial institute provides me with a log. From this log they would like me to give them some insights in there processes. In the first part I will discover the process models from the application's and proposal's life cycle as well as a combined process model depending on the outcome of the application process. These discovered models have to be verified and substantiated through conformance checking. They would also like to know if there are bottleneck in the application and workflow process and if so how I suggest they could solve them. In the second part the financial institute would like me to answer some questions about there processes. I will end with a conclusion.

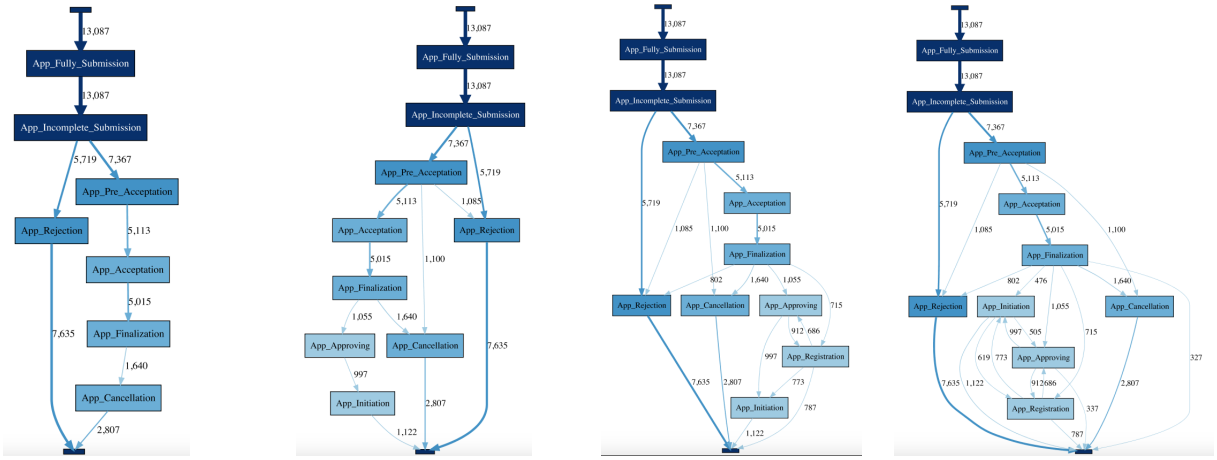
2 Process Models

2.1 Application's life cycle

2.1.1 Process Model

In order to discover the model of the application's life cycle I have to first filter the log. This is done in Prom with the tool "Filter Log on Event Attribute Values" by selecting in the attribute tab "name" the event names that start with "App_". This will give me only the application's activities what is the desired result and I can start discovering the model.

For discovering the process model in Prom I use the tool "Interactive Data-aware Heuristic Miner" and classify on event name. Ones this is computed I can choose the desired process model output type. I chose the "Directly-follows Graph" type because it can give a clear overview of which activities follow each other and with what frequency. The frequency is adaptable to your liking. In total there are 30 directly-follow relations. A number of these relations will be made visible in the graph depending on the chosen frequency. A frequency higher than 0.1 (9/30 relations) is not considered because not only a lot of useful data will be lost but also the graph is not correct any more, the trace doesn't start and end in the, respectively, initial and sink place. As an initial guess I choose the frequency values for which I have 9/30, 14/30, 20/30 and 25/30 directly-follow relations, the result is visible in figure 1. I don't consider the graph with all relations (30/30) because this is just not simplified enough. In the next section I will apply the conformance checking on these four graphs so I can make a decision on which graph to choose.

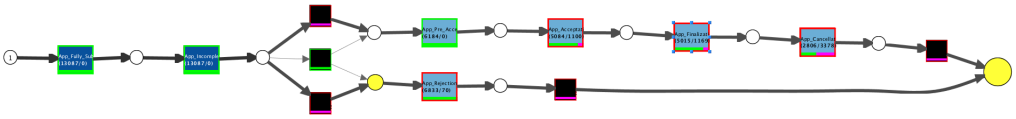


(a) frequency=0.1 (b) frequency=0.07 (c) frequency=0.05 (d) frequency=0.01

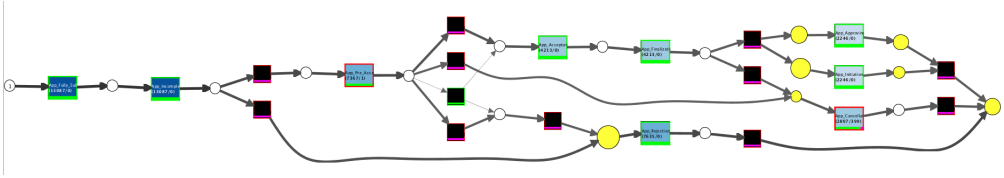
Figure 1: Directly-follows Graph

2.1.2 Conformance Checking

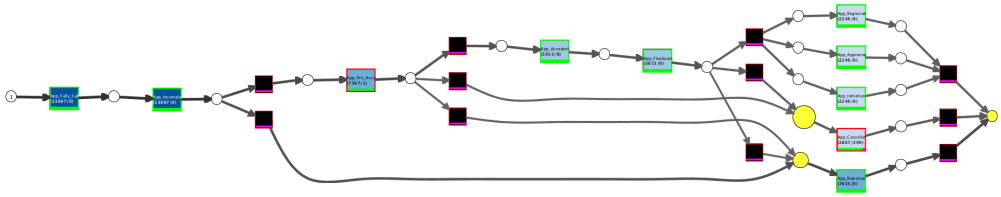
In ProM I can replay the log on a petri net with the "Replay a Log on Petri Net for Conformance Analysis" tool. First I have to generate the petri net for each chosen frequency. The petri net is also discovered with the tool "Interactive Data-aware Heuristic Miner". The output is now the petri net discovered with the "Flexible Heuristics Miner" algorithm. Once I have the petri net I can use this as the input for the "Replay a Log on Petri Net for Conformance Analysis" tool. The visible outcome of the replay can be found in figure 2. I will now explain, in general, what you can see on such an outcome. First of all the petri net discovered by the "Interactive Data-aware Heuristic Miner" tool creates some extra transitions, the black boxes. It is not always clear to me why the tool generates extra (empty) transitions but in some cases it is generated because there is an OR-split pattern, e.q. left most black boxes in figure 5a. In the transition boxes you can see a green or purple line on the bottom. The green line means that it is a synchronous move (in log and model), the purple line means that it is only a move in the model. The red border indicates that move on model happend at least ones. The numbers in the transition boxes quantify these lines. The first number indicates how many traces there are in total, the second number indicates the number of traces where move model only occurred. The yellow places mean that it is a marked place where move log occurred. The white places mean that no move log occurred. It is not visible in the figure but when a place is selected it is possible that an other/the same place turns red. This means that that place is marked when move log occurred on the selected place.



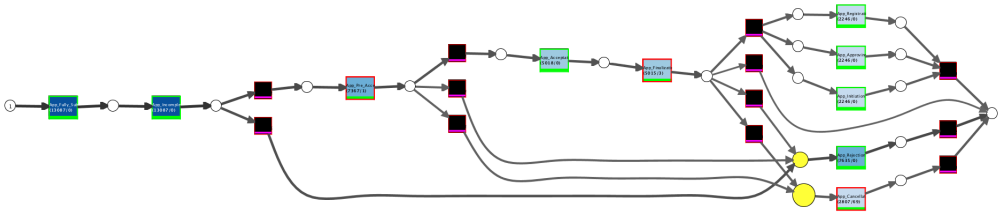
(a) frequency=0.1



(b) frequency=0.07



(c) frequency=0.05

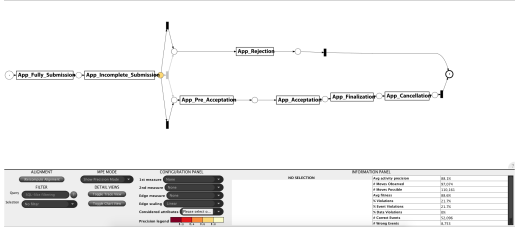


(d) frequency=0.01

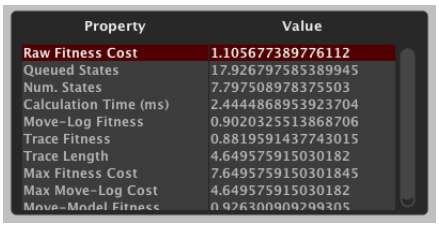
Figure 2: Conformance checking

2.1.3 Conclusion

There are four criteria (fitness, simplicity, generalization and precision) that can be used to determine which model is the best. I use three of the four criteria (fitness, simplicity and precision). Fitness and precision are obtained by the "Multi-perspective Process Explorer" tool in ProM. An example of the outcome can be found in figure 3a. In the alignment-base conformance checking I obtained a table with global statistics. An example can be found in figure 3b. From this table I used the entry "Trace Fitness" as a control of the fitness obtained by the other tool. In table 1 you can find the values for the fitness, precision and simplicity criteria. The simplicity criteria has no value but a sign indicating whether or not it has a (very) good, (+)+, or (very) bad, (-)-, simplicity. I can already discard the frequencies 0.05 and 0.01 as possible models because they have a bad simplicity. Just looking at there directly follow graphs you can notice that it takes quite some time to have a general idea about what is going on and what this model represents. On the other hand the amount of fitness and precision you can gain by decrementing the frequency from 0.07 to 0.05/0.01 does not outweigh the more complex model you get.
I choose the frequency of 0.07 because it has a good fitness and precision, an increase of,respectively, 8% and 6% compared to the frequency 0.1. Choosing 0.07 doesn't lose as much simplicity. So the weight simplicity has, is less than the weight of having a better fitness and precision. That is why I choose the process model of frequency 0.07



(a) Tool 1



(b) Tool 2

Figure 3: Global Statistics

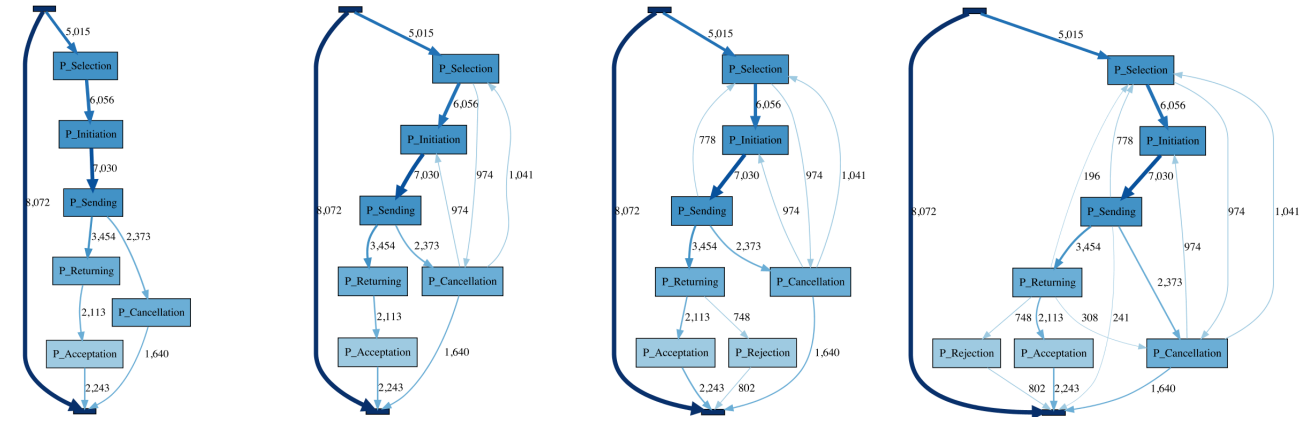
	Fitness	Precision	Simplicity
Frequency 0.1	88.6%	88.1%	++
Frequency 0.07	96.6%	94.8%	+
Frequency 0.05	99.6%	100%	-
Frequency 0.01	99.9%	100%	- -

Table 1: Conformance Table

2.2 Proposal's life cycle

2.2.1 Process Model

The method for discovering the model of the proposal's life cycle is very similar to that of the application's life cycle. First I filtered the log with the ProM tool "Filter Log on Event Attribute Values" and selected all the event names starting with "P_". Now I have all the proposal activities needed for discovering its process model. Like in the application's life cycle I use the ProM tool "Interactive Data-aware Heuristic Miner" to discover the directly follows graph of the proposal activities. Again I can change the frequency to obtain more/less directly-follow relations. In total there are 21 relations. The highest frequency I consider is 0.1 (9/21 relations). I frequency higher than 0.1 is discarded because the directly follow graph becomes incorrect. The trace doesn't start and end in the, respectively, initial and sink place any more. The other frequencies I consider are 0.07, 0.05 and 0.01 which corresponds with, respectively, 12/21, 15/21 and 18/21 relations. I discard the frequency 0.0 because the directly follow graph is just not simple enough any more. The results of the directly follow graphs can be found in figure 4

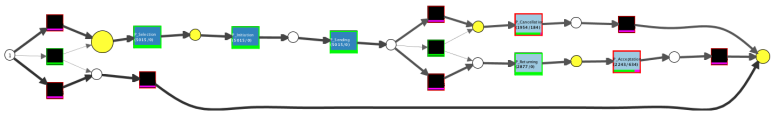


(a) frequency=0.1 (b) frequency=0.07 (c) frequency=0.05 (d) frequency=0.01

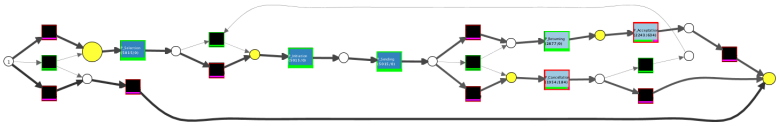
Figure 4: Directly-follows Graph

2.2.2 Conformance Checking

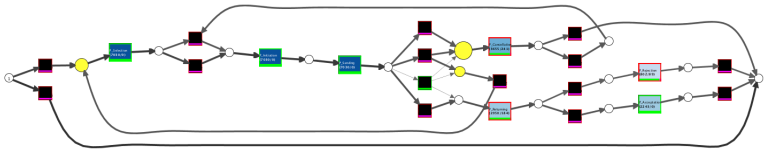
As for the application life cycle the conformance checking is done by the ProM tool "Replay a Log on Petri Net for Conformance Analysis". With the ProM tool "Interactive Data-aware Heuristic Miner" I can generate the petri nets for the different frequencies and use them as input for the replay tool. The petri nets are exactly the same as in the application life cycle discovered. The visible outcome of the replay tool can be found in figure 5 for the different frequencies. The same information about such an outcome as given in the application life cycle conformance checking can be applied here. That is why I don't repeat it here but go directly to the conclusion because that is more interesting.



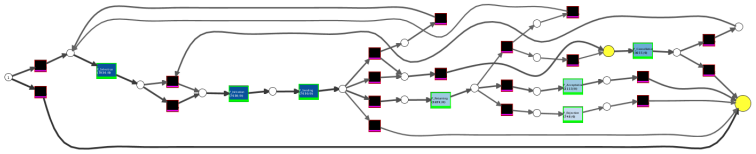
(a) frequency=0.1



(b) frequency=0.07



(c) frequency=0.05



(d) frequency=0.01

Figure 5: Conformance checking

2.2.3 Conclusion

To determine which model is the best/gets my preference I will use the criteria fitness, simplicity and precision. The precision and fitness can once again be computed by the ProM tool "Multi-perspective Process Explorer". The visual outcome of this tool can be found in figure 3a. The values for fitness, precision and simplicity can be found in table 2. Based on this table the model with frequency 0.1 gets my preference. The gain in a better fitness and precision does not outweigh the loss of simplicity, especially for the model with frequency 0.07, there is no gain in fitness and precision. For the models with frequency 0.05 and 0.01 the model gets to complex and the gain in fitness and precision is not significant enough. That is why I choose the process model of frequency 0.1.

	Fitness	Precision	Simplicity
Frequency 0.1	30.3%	80%	++
Frequency 0.07	30.3%	79.2%	+
Frequency 0.05	37.4%	83.5%	-
Frequency 0.01	38%	88.2%	- -

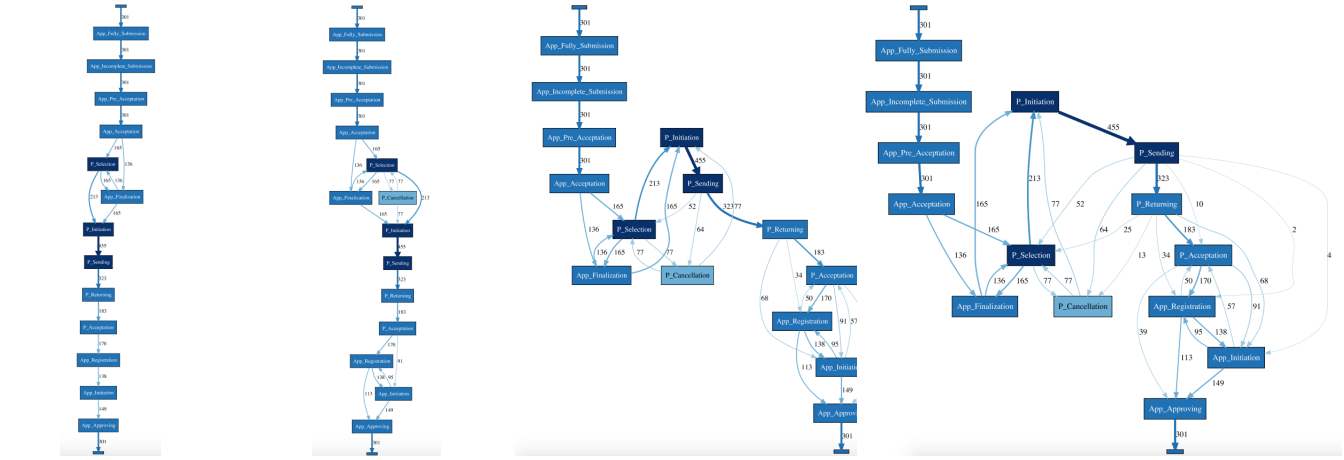
Table 2: Conformance Table

2.3 Application & Proposal's life cycle

2.3.1 Process Model

To discover the desired models I first filter the log so that I only have the event names that start with "App_" and "P_". This is done with the ProM tool "Filter Log on Event Attribute Values" as done before.

Application approved The filtered log is once again filtered now with the ProM tool "Filter Log using Simple Heuristics". I don't change the frequencies here but I only make sure that the final event that is happing will be the "App_Approving" event. After this filter I create the directly follows graph with the ProM tool "Interactive Data-aware Heuristic Miner" as used earlier. Again I can change the frequency so that the model becomes more/less simple. The result is shown in figure 6.



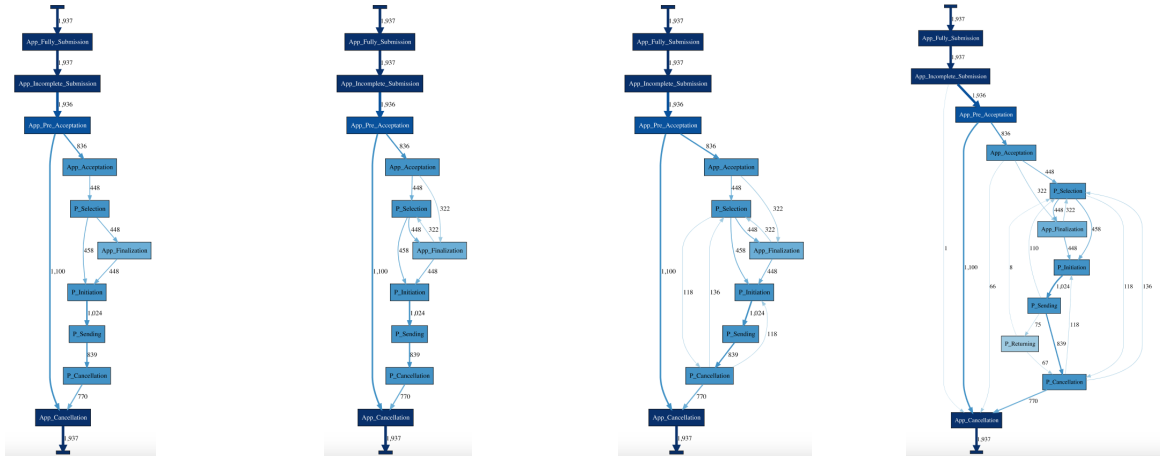
(a) frequency=0.45 (b) frequency=0.25 (c) frequency=0.1 (d) frequency=0.0

Figure 6: Directly-follows Graph

Application cancelled

The log is once again filtered with the ProM tool "Filter Log using Simple Heuristics". I don't change the frequencies here but I only make sure that the final event that is happing will be the "App_Cancelling" event. After this filter I create the directly follows graph with the ProM tool "Interactive Data-aware Heuristic Miner" as used earlier. Again I can change the frequency so that the model becomes more/less simple. The directly-follow graphs are shown in figure 7.

As in the first two questions I will execute a conformance check of the four possible directly-follow graphs. I do this, like in the first two questions, with the replay tool in ProM. I will not show the graphs, like I did in the first two questions, because it is to hard to get some useful data out of them. It is more useful to make a table where I show the fitness, precision and simplicity of every option so that I can make a decision about which model is the best. The precision is calculated, as before, with the ProM tool "Multi-perspective Process Explorer". The results for all the frequencies can be found in table **??**



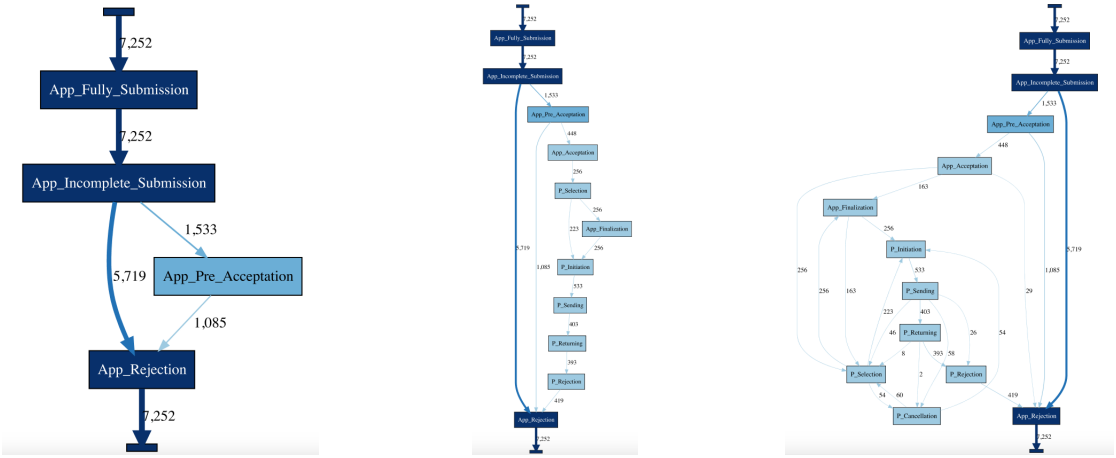
(a) frequency=0.23 (b) frequency=0.1 (c) frequency=0.06 (d) frequency=0.0

Figure 7: Directly-follows Graph

	Fitness	Precision	Simplicity
Frequency 0.23	96.84%	94.6%	++
Frequency 0.1	96.84%	94.6%	+
Frequency 0.06	96.84%	96.2%	-
Frequency 0.0	99.96%	92.5%	- -

Table 3: Conformance Table

Application rejection The log is once again filtered with the ProM tool "Filter Log using Simple Heuristics". I don't change the frequencies here but I only make sure that the final event that is happing will be the "App_Rejection" event. After this filter I create the directly follows graph with the ProM tool "Interactive Data-aware Heuristic Miner" as used earlier. Again I can change the frequency so that the model becomes more/less simple. The result is shown in figure 8.



(a) frequency=0.1 (b) frequency=0.03 (c) frequency=0.0

Figure 8: Directly-follows Graph

2.3.2 Conformance Checking

2.3.3 Conclusion

	Fitness	Precision	Simplicity
Frequency 0.45	90.00%	92.8%	++
Frequency 0.25	93.93%	91.7%	+
Frequency 0.1	99.47%	91.1%	-
Frequency 0.0	99.84%	85.4%	- -

Table 4: Conformance Table Approval

	Fitness	Precision	Simplicity
Frequency 0.23	96.84%	94.6%	++
Frequency 0.1	96.84%	94.6%	+
Frequency 0.06	96.84%	96.2%	-
Frequency 0.0	99.96%	92.5%	- -

Table 5: Conformance Table Cancellation

	Fitness	Precision	Simplicity
Frequency 0.1	96.91%	100%	++
Frequency 0.03	99.6%	98.7%	-
Frequency 0.0	99.99%	99.3%	- -

Table 6: Conformance Table Rejection