Homework 2

Miriam Wagner
373045

July 10, 2018

1 Introduction

In this assignment I have to analyze for a financial institute data and make suggestions for the company. The event log is taken from the information system while it is running.

All researches were done with ProM as ecpected from us. The data contains three types of activities (taken from the Assignment):

1. Event names that start with "App_" refer to status changes of applications. People make applications for a loan online.

2. Event names that start with "P_" refer to status changes of proposals, i.e. if the financial institute sees an opportunity to sell a loan to a customer, a proposal is sent to the customer.

3. Events that start with "W_" refer to activities performed by employees in the process, for example the call-center employees

The data is analyzed in different steps to get a better idea of the process.

2 General insights about the data set

The given data set is generated between 1st Oct 2011 0:38:44 (Saturday) and 14th Mar 2012 16:04:54 (Wednesday) and contains 13087 cases with 262200 events executed. One case contains minimal 3 events and maximal 175. In average 20.035 events.

There are 4366 different variants of cases. There are 36 different events possible "W_Complete_Application + COMPLETE" (9.141%), "W_Complete_Application + START" (8.967%), "W_Quotations + COMPLETE" (8.763%), "W_Quotations + START" (8.545%), "App_Fully_Submission + COMPLETE" (4.991%), "App_Incomplete_Submission + COMPLETE" (4.991%), "W_Handeling_Incomplete_Dossiers + COMPLETE" (4.35%), "W_Handeling_Incomplete_Dossiers + START" (4.348%), "W_Validation + COMPLETE" (3.011%), "W_Validation + START" (3.01%), "App_Rejection + COMPLETE" (2.912%), "W_Complete_Application + SCHEDULE" (2.811%), "App_Pre_Acceptation + COMPLETE" (2.81%), "P_Initiation + COMPLETE" (2.681%), "P_Selection + COMPLETE" (2.681%), "P_Sending + COMPLETE" (2.681%), "W_Quotations + SCHEDULE" (2.53%), "W_Handling_Leads + COMPLETE" (2.249%), "W_Handling_Leads + START" (2.249%), "App_Acceptation + COMPLETE" (1.95%), "W_Validation + SCHEDULE" (1.916%), "App_Finalization + COMPLETE" (1.913%), "W_Handling_Leads + SCHEDULE" (1.82%), "P_Cancellation + COMPLETE" (1.394%), "P_Returning + COMPLETE" (1.317%), "App_Cancellation + COMPLETE" (1.071%), "W_Handeling_Incomplete_Dossiers + SCHEDULE" (0.909%), "App_Initiation + COMPLETE" (0.857%), "App_Approving + COMPLETE" (0.857%), "App_Registr + COMPLETE" (0.857%), "P_Acceptation + COMPLETE" (0.855%), "P_Rejection + COMPLETE" (0.306%), "W_Fraud_Dete + COMPLETE" (0.103%), "W_Fraud_Detection + START" (0.103%), "W_Fraud_Detection + SCHEDULE" (0.047%) and "W_Changing_Contact_Details + SCHEDULE" (0.005%).

The complete, start and shedule are more details about the state of the event. Just about the Workflow events we have more information about the state.

All processes start in "App_Fully_Submission + COMPLETE" (100.0%).

The processes can end in 13 different events:

"App_Rejection + COMPLETE" (26.202%), "W_Validation + COMPLETE" (20.975%), "W_Handling_Leads + COMPLETE" (17.07%), "W_Complete_Application + COMPLETE" (14.816%), "W_Quotations + COMPLETE" (9.849%), "App_Cancellation + COMPLETE" (5.005%), "W_Handeling_Incomplete_Dossiers + COMPLETE" (3.454%), "P_Cancellation + COMPLETE" (2.132%), "W_Fraud_Detection + COMPLETE" (0.436%), "W_Changing_Contact_Details + SCHEDULE" (0.031%), "W_Validation + START" (0.015%), "App_Registration + COMPLETE" (0.008%) and "W_Quotations + START" (0.008%).

What you already see here, is that the process not always ended. All the process ending with a start event are probably not done now. This was to expected, because the data is picked during the running process.

Having a look at the whole data it can be seen, that there are just athe following events executes: "W_Complete_Application", "W_Handling_Leads", "App_Incomplete_Submission", "P_Cancellation", "App_Fully_Submission" and "App_Rejection". Furthermore is there a constant input of new cases. The events are mostly during the week executed same amount of times just "W_Fraud_Detection" and "W_Changing_Contact_Details" have really deeps and ups. What is not surprising, because both are outlier behavior actions.
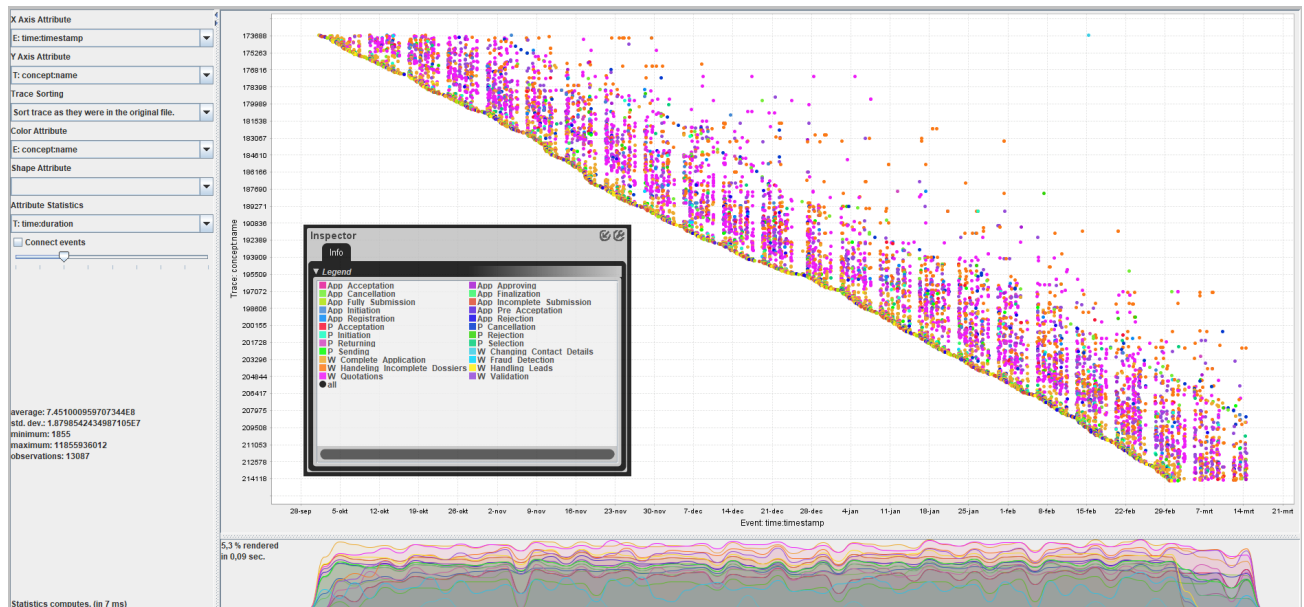
Figure 1: Dotted chart of the data set

3 Process Models

3.1 Exploring process

For exploring a good model of a data set I first filtered the data with the (**Filter log on event attribute names**) tool for extracting the required. Then I used **Interactive data heuristic miner** and **Inductive Miner** on the filtered data set to discover different models. After comparing the outcomes I decided to concentrate on **Interactive data heuristic miner** and their directly followed graphs and petri nets for understanding the lifecycle. I think the petri net is the best with basic configuration and just different frequency filters. The conformance checking where done with **Replay a log on Petri Net for conformance analysis** tool on the petri net and the filtered data. For the precision check I applied the **Multi-perspective Process Explorer** tool on the petri net and the filtered data and chose "show precision mode" in the tool with basic configuration.

3.2 Application data set

Just the events beginning with "App_..." are required. The resulting data set is saved as "Filtered App".

3.2.1 General details of the data set

The data set is collected between the 1st of 0ct 2011 (Saturday), 00:38:44 and the 14th of Mar 2012 (Wednesday), 15:33:57. It contains 13087 cases with 60849 executed events.
10 different events appear (occurences relative), all events are complete so I deleted this extra information: "App_Fully_Submission" (21.507%), "App_Incomplete_Submission" (21.507%), "App_Rejection" (12.547%), "App_Pre_Acceptation" (12.107%), "App_Acceptation" (8.403%), "App_Finalization" (8.242%), "App_Cancellation" (4.613%), "App_Initiation" (3.691%), "App_Approving" (3.691%) and "App_Registration" (3.691%). Just "APP_Fully_Submission" appears to be a start event. However there are 8 end events possible: "App_Rejection" (58.34%), "App_Cancellation" (21.449%), "App_Initiation" (8.573%), "App_Registration" (6.014%), "App_Approving" (2.575%), "App_Finalization" (2.499%), "App_Pre_Acceptation" (0.527%) and "App_Acceptation" (0.023%).
A trace contains maximal 8 different events and minimal 3. The mean is 4.65. In total there are 17 different variants of traces.
In figure 2 the dotted chart can be seen. Having a closer look at this chart you see gaps, which are always on a sunday. Those gaps do not appear for "APP_Pre_Acceptation" and "APP_Incomplete_Submission". Furthermore in the lower left corner the following information can be found: average duration of a case, 8 days 1 hours 55 minutes and 14.86 seconds, and the maximum duration, 91 days 10 hours 55 minutes and 36.16 seconds. Both are given in milliseconds.

3.2.2 Discover and evaluate models of the application lifecycle

To find an accurate model I tried different frequency filters, 3. First I chose the frequency 0.1 and had a look at the resulting petri net. I also checked 0.2 and 0.51 to get a better understanding of the model(not shown here is the petri net with 0.6 filtering, which was the next filtering after 0.1, that changed the net, but I decided, that it does not contain enough cases). For comparsion in the end I had also a look at the original petri net found by the **Interactive data heuristic miner**. In figure 3 the 4 considered petri nets can be seen. My first choice was the petri net with 0.1 as threshold for frequency, because it was a simple model that still tells us a lot about the main process (90% of the cases) and is not too specific (still has an acceptable generalization). Obviously the original graph does not fullfill the criterium of simplicity and also the graph with
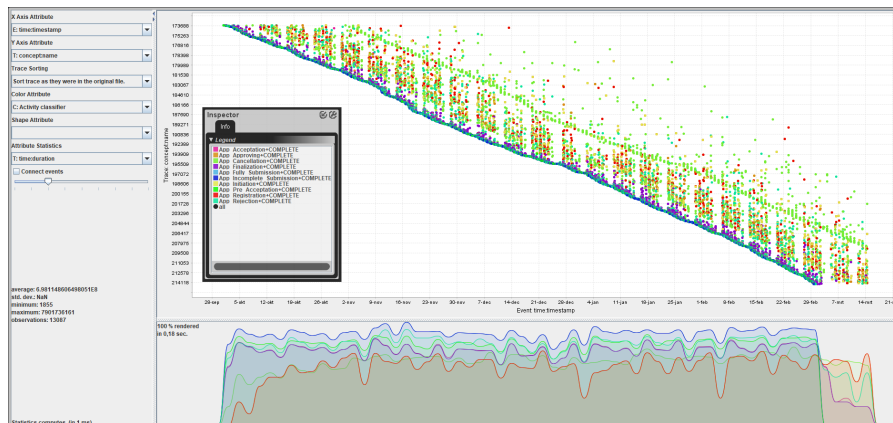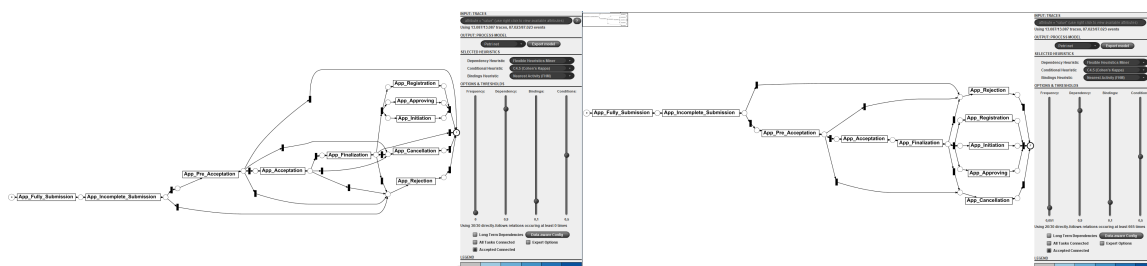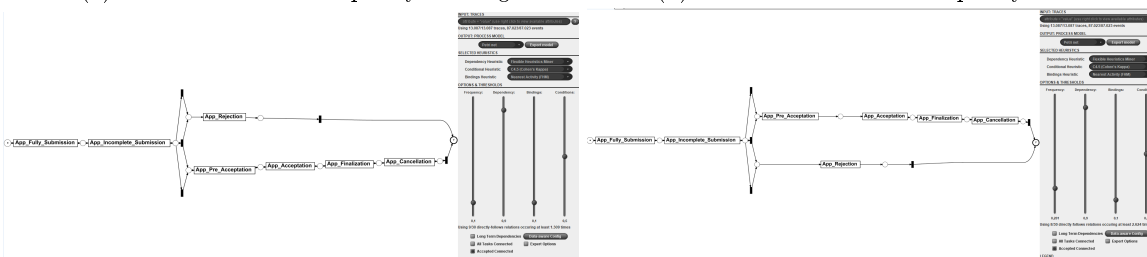
Figure 2: Dotted chart showing the time of events

(a) Petri net without frequency filtering

(b) Petri net with 0.051 frequency filter

(c) Petri net with 0.1 frequency filter

(d) Petri net with 0.2 frequency filter

Figure 3: Considered petri nets

frequency 0.051 still looks not as simple. I do not consider the 0.2 filtering in the next steps, because it is the same petri net as the 0.1 filtered one.
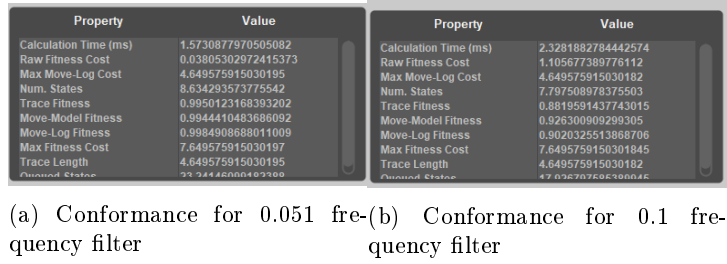

(a) Conformance for 0.051 frequency filter

(b) Conformance for 0.1 frequency filter

Figure 4: Conformance checking

The results of the conformance checking are shown in figure 4.

Avg activity precision	100%	Avg activity precision	88,1%
# Moves Observed	144.789	# Moves Observed	97.074
# Moves Possible	144.789	# Moves Possible	110.161
Avg fitness	99,6%	Avg fitness	88,6%
% Violations	0,8%	% Violations	21,7%
% Event Violations	0,8%	% Event Violations	21,7%
% Data Violations	0%	% Data Violations	0%
# Correct Events	60.849	# Correct Events	52.096
# Wrong Events	0	# Wrong Events	8.753
# Missing Events	498	# Missing Events	5.717

(a) Precision for 0.051 as frequency filter

(b) Precision for 0.1 as frequency filter
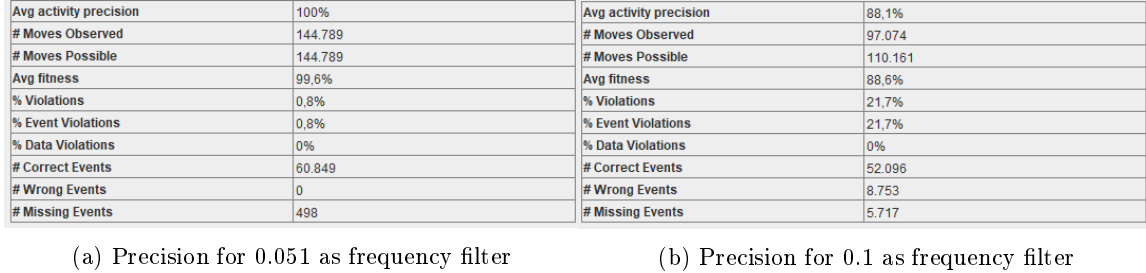
Figure 5: Precision checking

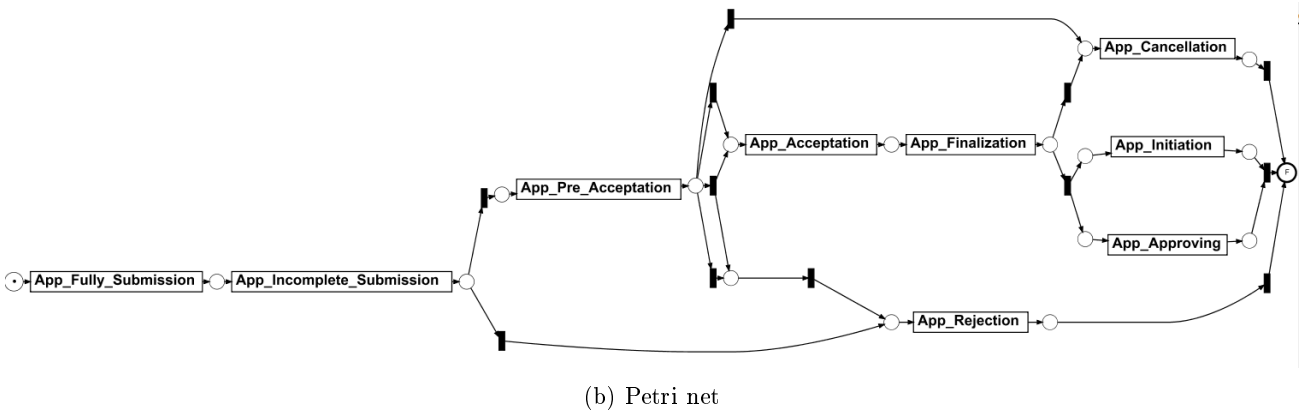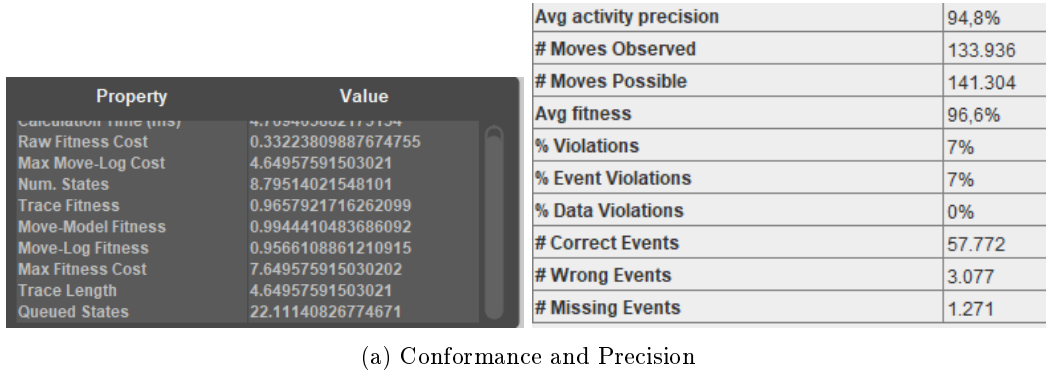(a) Conformance and Precision

(b) Petri net

Figure 6: Frequency 0.069

Having a look at the precision, figure 5, I could see, that the precision of 0.051 filtering is 100%, but of the 0.1 filtering just 88.1%. In combination with the results before, I came to the conclusion, that 0.1 filtering is not good enough as model and 0.051 would be good enough, but did not fullfill my simplicity criterium complete. Starting by 0.051 for filtering I again tried different frequency filters starting by 0.075 to find a model with a similar simplicity as the 0.1 frequency model, but a better conformance and precision than this simpel model. The best result I found was with 0.069 filtering. This model is simpel enough to understand the main lifecycle, but still has good result for conformance (fitness 96.58%) and precision(94.8%),6. Based on this I decided to choose this petri net for the lifecycle of the application data.

3.2.3 Discussion of the lifecycle

Simplifying the discussion I will not write "App_.." in begin of all events, but every event mentioned in this section is from the application data set. To also have details about the occurency of the steps in the model I used the replay result, 7.
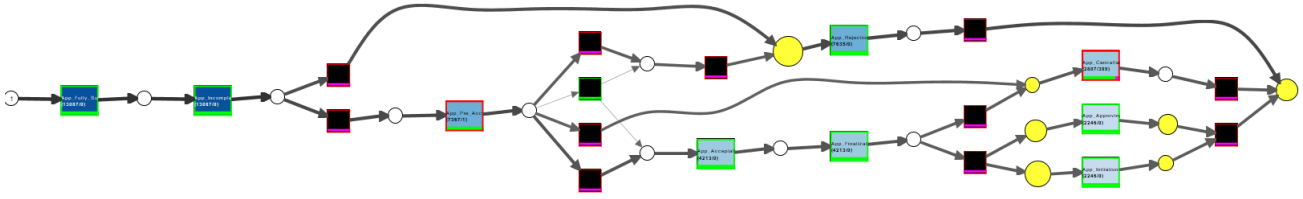

Figure 7: Replay on petri net with 0.069 frequency filter

The process always starts with "Fully_Submission" and "Incomplete_Submission" (13087 cases). After this there are two different outcomes:

1. "Rejection" (5719 cases, executed in total in 7365 cases)

2. "Pre_Acceptation" (7367 cases)

"Pre_Acceptation" has 4 different successor events

1. "Rejection" (1916 cases)

2. "Rejection" and "Acceptation" (0 cases)

3. "Acceptation" (4213 cases)

4. "Cancellation" (1239 cases)

"Acceptation" is followed by "Finalization" (4213). "Finalization" is followed in (1967 cases) by "Cancellation", which is in total executed 3206 times, and otherwise by "Approving" and Initation (2246 cases). All the differences between incoming and outcoming cases can be explained through the filtering by frequency for building the model.
Looking at the **Time between Transition matrix** of the replay result it can be seen, that the transition to "Approving" and "Initiation" need in average ~16 days, which is the second longest transition time. "Cancellation" has the worst transition time with in average ~20 days.

3.3 Proposal data

Just the events beginning with "P_..." are required. The resulting data set is saved as "Filtered P". Having a look in the data it was seen, that there were a lot of empty traces, so I filtered it again with the **Filter Log using Heuristics** tool, without applying any filter just everything to 100% and saved this as "Pdoublefiltered". I use the second data set in the next steps.

3.3.1 General Details of the data set

The data set is collected between the 1st of 0ct 2011 (Saturday), 10:44:40 and the 14th of Mar 2012 (Wednesday), 15:50:59 and contains 5015 cases with 31244 events. Looking at the visualization you can see that there are gaps in the workflow.

Figure 8: Dotted chart showing the time of events

In figure 8 the dotted chart can be seen. When the gaps in the dotted chart are studied closely, it's clear that these gaps illustrate the situatioin on sundays. What does not show this behaviour so clear is "P_Cancellation" and "P_Initiation". Furthermore in the lower left corner the following information can be found: the average duration of a case, 17 days 4 hours

20 minutes and 24.85 seconds, the minimum duration is 667 milliseconds and the maximum duration 89 days 13 hours 10 minutes and 6.16 seconds. Both is given in milliseconds.

The data set has 7 events: "P_Initiation" (22.5%), "P_Sending" (22.5%), "P_Selection" (22.5%), "P_Cancellation" (11.698%), "P_Returning" (11.055%), "P_Acceptation" (7.179%) and "P_Rejection" (2.567%). There are 169 different variants of traces.

All traces start with "P_Selection". And the end events are distributed like this: "P_Acceptation" (44.726%), "P_Cancellation" (32.702%), "P_Rejection" (15.992%), "P_Sending" (4.806%) and "P_Returning" (1.775%).

Maximal 30 events are executed in a case and minimal 3. The mean of events per class is 6.23.

3.3.2 Discover and evaluate models of the proposal lifecycle

Applying the steps on the proposal data set gave me in first instance 4 models I wanted to have a closer look at.
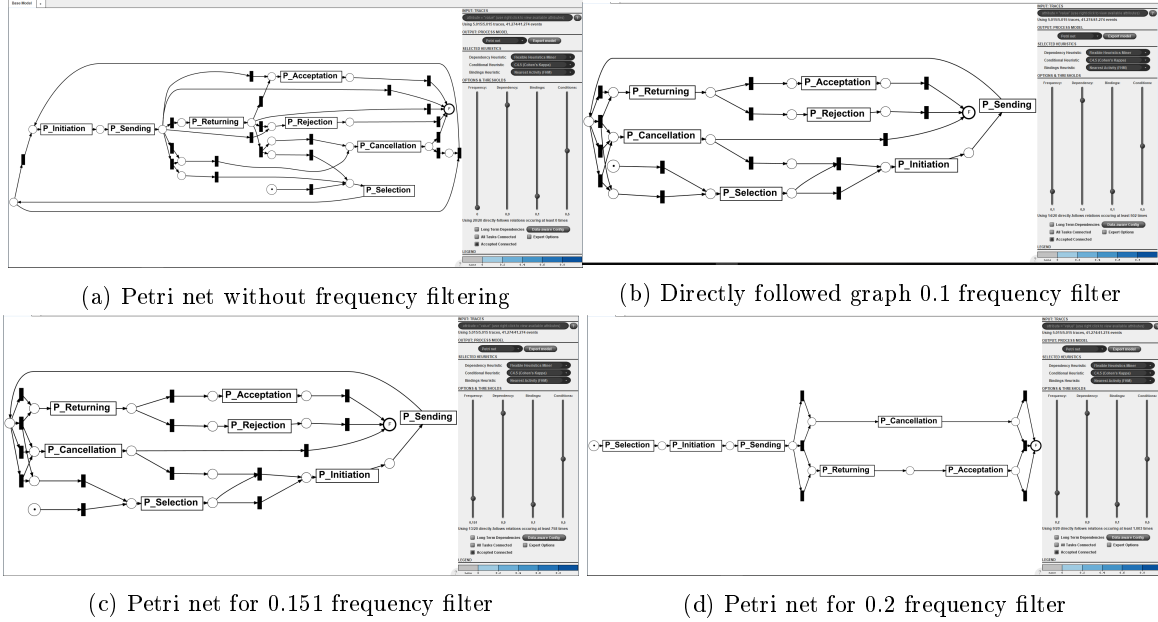

(a) Petri net without frequency filtering

(b) Directly followed graph 0.1 frequency filter

(c) Petri net for 0.151 frequency filter

(d) Petri net for 0.2 frequency filter

Figure 9: Considered petri nets

Because of simplicity reasons I picked out of those, figure 9, just the 0.151 and 0.2 filtered petri nets.

Property	Value
Calculation Time (ms)	1.07050450707070645
Raw Fitness Cost	0.20299102691924234
Max Move-Log Cost	6.230109670987042
Num. States	14.854237288135613
Trace Fitness	0.9824694366187957
Move-Model Fitness	0.9837550634114102
Move-Log Fitness	0.9894060148078491
Max Fitness Cost	10.230109670987057
Trace Length	6.230109670987042
Queued States	39.14217347956131

(a) Conformance for 0.151 as frequency filter

Property	Value
Calculation Time (ms)	3.394013333333302
Raw Fitness Cost	1.84187437686939
Max Move-Log Cost	6.230109670987028
Num. States	11.126021934197395
Trace Fitness	0.8633619014528089
Move-Model Fitness	0.9655433698903272
Move-Log Fitness	0.8301506678744179
Max Fitness Cost	10.230109670987046
Trace Length	6.230109670987028
Queued States	23.692123629112654

(b) Conformance for 0.2 as frequency filter

Avg activity precision	79,6%
# Moves Observed	81.810
# Moves Possible	102.778
Avg fitness	98,4%
% Violations	3,2%
% Event Violations	3,2%
% Data Violations	0%
# Correct Events	30.740
# Wrong Events	504
# Missing Events	514

Avg activity precision	91,9%
# Moves Observed	45.844
# Moves Possible	49.872
Avg fitness	85,6%
% Violations	28,5%
% Event Violations	28,5%
% Data Violations	0%
# Correct Events	23.202
# Wrong Events	8.042
# Missing Events	1.195

(c) Precision for 0.151 as frequency filter

(d) Precision for 0.2 as frequency filter

Figure 10: Conformance and precision checking

The conformance and precision outcomes(see in figure 10) told me, that the 0.2 frequency model (fitness 86.33%, precision 91.9%) has a better outcome for the precision as the 0.151 filtered model(precision 79.6%), but a worse for the fitness

(98.25%). So I searched for an other model, which has a similar simplicity as the 0.2 and 0.151 filtered models and might have a better fitness or precision. The model fullfilling simplicity was 0.175 filtering, but the fitness was worse, than for 0.2 filtering. I tried this with several other filterings, but the best was 0.2 filtering, so I chose this one.

3.3.3 Discussion of the lifecycle

Simplifying the discussion I will not write "P_.." in begin of all events, but every event mentioned in this section is from the proposal data set.
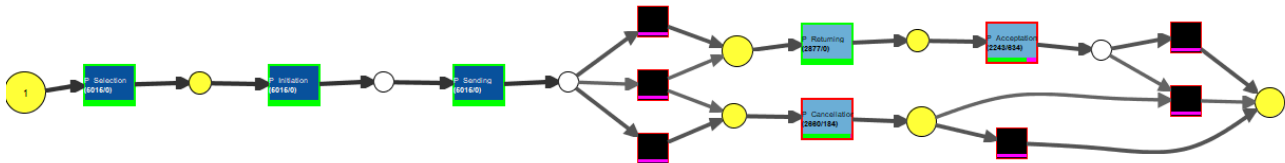

Figure 11: Replay on petri net with 0.2 frequency filter

5015 cases start with "Selection". Then "Initiation" event is performed (5015 cases). "Initiation" is followed by "Sending" (5015 cases), which has 3 different succesor actions.

1. "Returning" (2171 cases)

2. "Returning" and "Cancellation" (706 times)

3. "Cancellation" (2138 times)

"Returning" is in 2243 cases followed by "Acceptation". "Cancellation" and "Acceptation" have to be joined again if it was an end split before

3.4 Combined Model

For combined Models I first filtered the data to have a dataset with all proposal data combined with the application data. This data I filtered with Heuristic filter (all configurations to 100% and just deciding what the endstate is) with the outcomes/endstates "APP_rejected" or "APP_cancelled" or "APP_approved". I saved them under the names "Filtered P App with approv", "Filtered P App with canc" and "Filtered P App with rej".

3.4.1 Endstate APP_Approved

General Details of the data set

The data set is collected between the 1st of 0ct 2011 (Saturday), 12:36:08 and the 14th of Mar 2012 (Wednesday), 15:23:32. The set contains 301 and 4550 events executed.
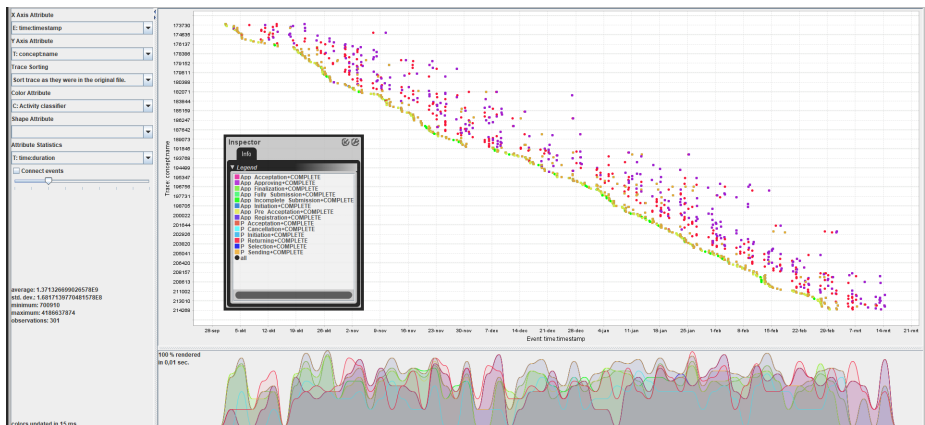

Figure 12: Dotted chart showing the time of events

In figure 12 the dotted chart can be seen. There is a gap visible on every sunday. This is more clearly to see, when changing the color to days of the week. There you can see, that it is always a sunday. Furthermore in the left lower corner the following information can be found: average duration of a case is, 15 days 20 hours 55 minutes and 26.70 seconds, the minimal duration, 11 minutes and 40.91 seconds, and the maximum duration, 48 days 27 hours 23 minutes and 14.69 seconds. Those are given in milliseconds in the figure.

There are 14 different events: "P_Initiation" (10.0%), "P_Sending" (10.0%), "P_Selection" (10.0%), "P_Returning" (7.099%), "App_Pre_Acceptation" (6.615%), "App_Initiation" (6.615%), "App_Approving" (6.615%), "App_Registration" (6.615%), "App_Finalization" (6.615%), "App_Fully_Submission" (6.615%), "App_Incomplete_Submission" (6.615%),

"App_Acceptation" (6.615%), "P_Acceptation" (6.593%), "P_Cancellation" (3.385%). The percentages are the relative occurences. There are 73 different variants of traces.

All events start with "App_Fully_Submission".

Maximal 34 events are executed in a case and minimal 12. The mean of events per class is 15.116.

Discover and evaluate models of lifecycle with Endstate APP_Approved

Like for approved and proposal I first checked different frequency filter setting to have a first idea, which models fullfill simplicity. Then for every chosen frequency in begin I checked conformance and precision.

	Frequency			
	0	0.1	0.2	0.3
Simplicity	-	-	+	++
Fitness	99.84	99.47	93.93	93.93
Precision	93.3	94.5	91.7	94.8

Figure 13: Results for approved as endstate

Based on the results, 13, I chose the model with 0.3 filtering. This one has a high simplicity, but still has surprisingly good results.
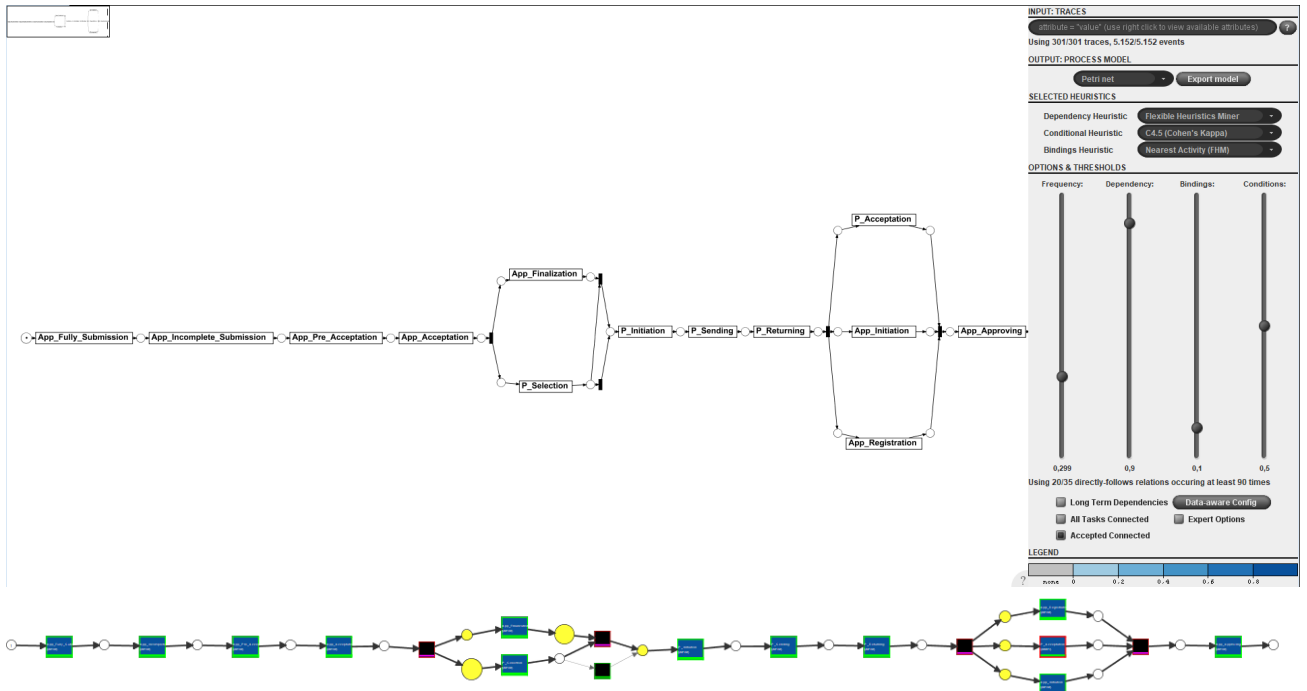
Discussion of the model

Figure 14: Endstate APP_Approved

In figure 14 the petri net and the result of the replat can be seen. It is hard to clearly read the details, but I describe it in detail here.

All 301 cases start with the same 4 events: "App_Fully_Submission" →"App_Incomplete_Submission" →"App_Pre_Acceptation →"App_Acceptation". Then there are two events parallel: "App_Finalization" and "P_Selection". Before and after those actions are events skipped in the model, that occur in the original log. Afterwards "P_Initiation" →"P_Sending" →"P_Returning" are executed in all cases. Again 3 events are parallel "App_Registration", "App_Registration" (just in 300 cases and "App_Initiation". The last event is "App_Approving".

When I controlled the time matrix for the transitions I saw, that there are a lot of transitions taking place between 12 and 16 days.

3.4.2   Endstate APP_Cancelled

General Details of the data set

The data set is collected between the 1st of Oct 2011 (Saturday), 09:45:25 and the 14th of Mar 2012 (Wednesday), 15:30:47. The set contains 1937 and 13524 events executed.

In figure 15 the corresponding dotted chart can be seen. On every sunday there are just two events executed "App_Pre_Acceptation" and "App_Incomplete_Submission". Furthermore in the left lower corner the following information can be found: the average duration of a case is, 16 days 22 hours 23 minutes and 7.31 seconds, the minimal duration, 1 minute and 5.067 seconds, and the maximum duration, 90 days 3 hours 48 minutes and 38.29 seconds. All three values are given in milliseconds in the figure.

There are 11 different events (relative occurences): "App_Cancellation" (14.323%), "App_Fully_Submission" (14.323%), "App_Incomplete_Submission" (14.323%), "App_Pre_Acceptation" (14.315%), "P_Cancellation" (7.572%), "P_Initiation"
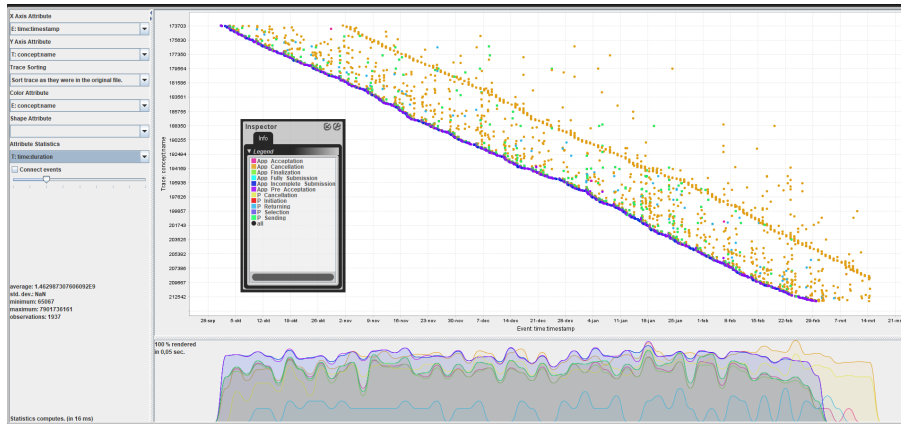
Figure 15: Dotted chart showing the time of events

(7.572%), "P_Sending" (7.572%), "P_Selection" (7.572%), "App_Acceptation" (6.182%), "App_Finalization" (5.694%) and "P_Returning" (0.555%). There are 46 different variants of traces.

All traces start with "App_Fully_Submission"

A maximum of 32 events is executed in a case and minimal 3. The mean of events per class is 6.982.

Discover and evaluate models of lifecycle with Endstate APP_Cancelled

	Frequency		
	0	0.049	0.1
Simplicity	-	+	++
Fitness	99.96	99.39	96.84
Precision	92.5	91.8	94.6

Figure 16: Results for cancelled as endstate

Like for APP_Approved I checked different configurations and based on the results, 16, I decided to pick 0.1 filtered frequency model. The fitness and precision is still higher than 90%, but it is also the most simpel model.

Discussion of the model

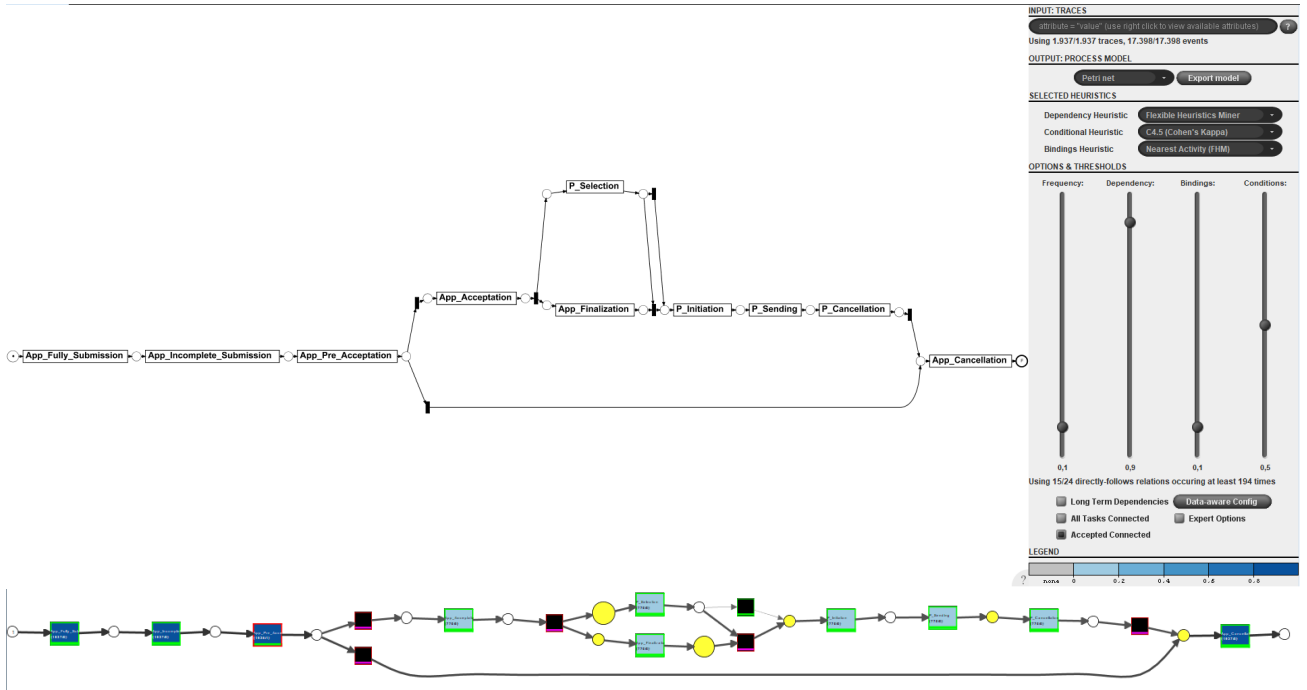
Figure 17: Endstate APP_Cancelled

In figure 17 the petri net and the result of the replay can be seen.

All cases start with "App_Fully_Submission" →"App_Incomplete_Submission" →"App_Pre_Acceptation" (this is just executed in 1936 cases). Then in 1167 cases "App_Cancellation" follows directly. Otherwise "App_Acceptation" (770 cases), followed by "P_Selection" and "App_Finalization" in random order. After those are executed, "P_Initiation" →"P_Sending" →"P_Cancellation" →"App_Cancellation" is executed.

The time transition matrix showed me, that the transition to "App_Finalization" and "P_Cancellation" took the most time (between 17 and 22.5 days).

3.4.3 Endstate APP_Rejected

General Details of the data set

The data set was collected between the 1st of Oct 2011 (saturday), 08:11:08 and the 14th of Mar 2012 (Wednesday), 15:20:23. The set contains 7252 and 26691 events executed.
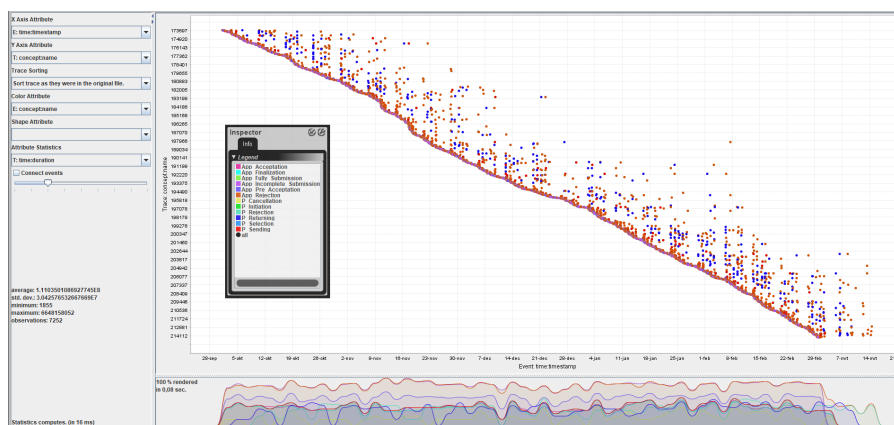

Figure 18: Dotted chart showing the time of events

In figure 18 the dotted chart can be seen. On every sunday there are just two events executed "App_Rejection" and "App_Incomplete_Submission". Furthermore in the left lower corner the following information can be found: the average duration of a case is, 1 days 6 hours 50 minutes and 35.01 seconds, the minimal duration, 1.855 seconds, and the maximum duration, 76 days 22 hours 42 minutes and 38.05 seconds. All three values are given in milliseconds in the figure.

There are 12 different events (relative occurences): "App_Rejection" (27.17%), "App_Fully_Submission" (27.17%), "App_Incompl (27.17%), "App_Pre_Acceptation" (5.744%), "P_Initiation" (1.997%), "P_Sending" (1.997%), "P_Selection" (1.997%), "App_Acceptation" (1.678%), "App_Finalization" (1.57%), "P_Rejection" (1.57%), "P_Returning" (1.51%), "P_Cancellation" (0.427%). There are 30 different variants of traces.

All traces start with "App_Fully_Submission"

A maximum of 12 events is executed in a case and minimal 3. The mean of events per class is 3.681.

Discover and evaluate models of lifecycle with Endstate APP_Rejected

The last analysis is of the models ending in APP_Rejected. In the first step I checked different frequency filters and decided based on simplicity and traceability I had to take a closer look at 0, 0.025 and 0.1.

	Frequency		
	0	0.025	0.1
Simplicity	-	+	+++
Fitness	1.00	99.60	96.91
Precision	99.3	98.7	100

Figure 19: Results for cancelled as endstate

Based on 19 I chose 0.1 filtered frequency model as the best. It is really easy to follow and has a good fitness.

Discussion of the model

In figure 20 the model and the replay can be seen.

The process starts with "App_Fully_Submission" →"App_Incomplete_Submission" (7252 cases). This is or followed directly by "App_Rejection" (5719 cases) or first by "App_Pre_Acceptation" (1533 cases) and then "App_Rejection". No details presented for "App_Rejection" due to filtering.

The worst transition is with 5.51 days from "App_Pre_Acceptation" to "App_Rejection".

3.5 C-net of the proposal process

Based on the results of my analysis of the model before I chose the same frequency filtering for the C-net of the proposal process.

In figure 21 I show two C-nets of the proposal process for comparsion. 21a is the one I picked and I will discuss in detail. In 21b the original C-net of the process is to see and obviously it is much more complicated and not so intuitive than the filtered one.

3.5.1 Analysis of the C-net

For simplicity reasons I will not write "P_" as prefix of every activity.

The first thing I did is having a look at the maximum number of bindings. This is for sending with 3 possible bindings. The input is always from initiation, but there are 3 different outputs possible. All trace possibilities:

These 5 pathes are also what you would expect a little bit, but you clearly see, that there are missing details. But depending on the further results those pathes are the main pathes.
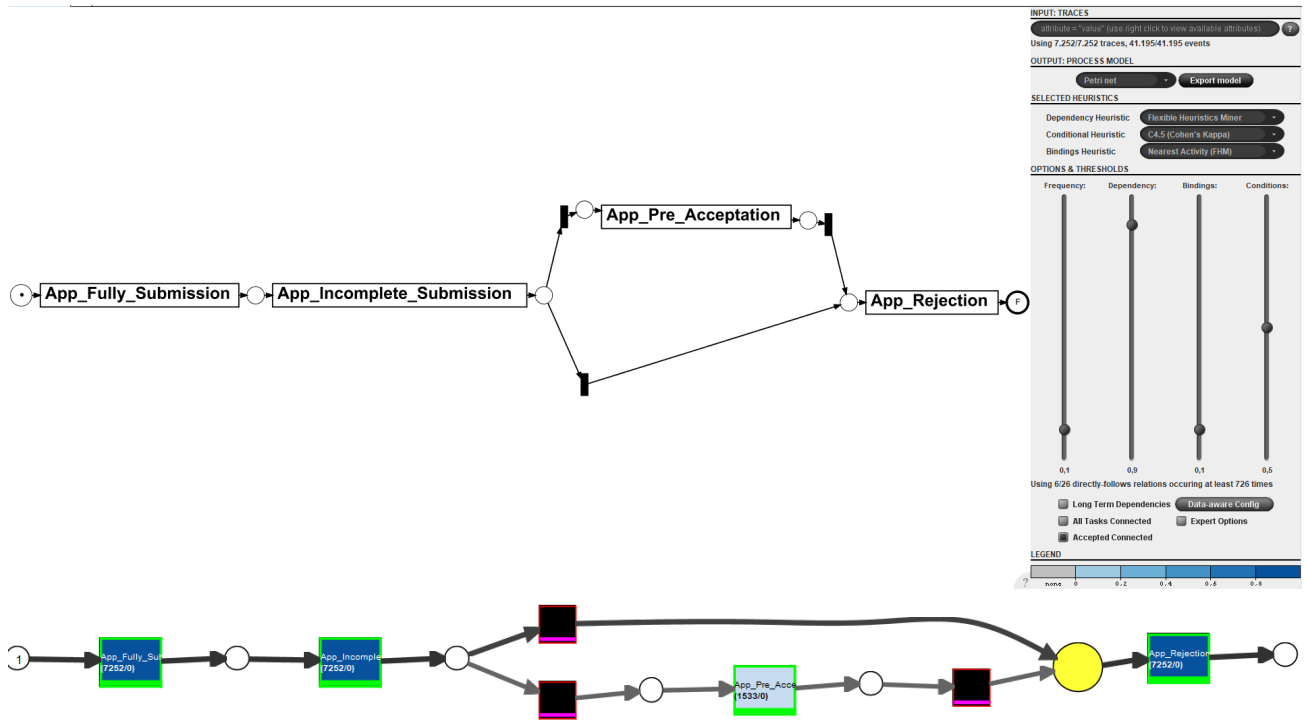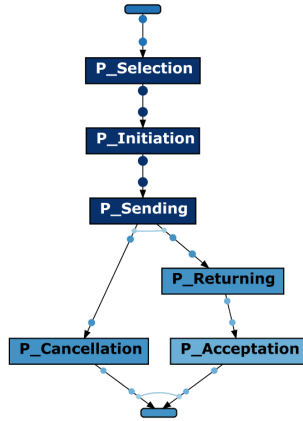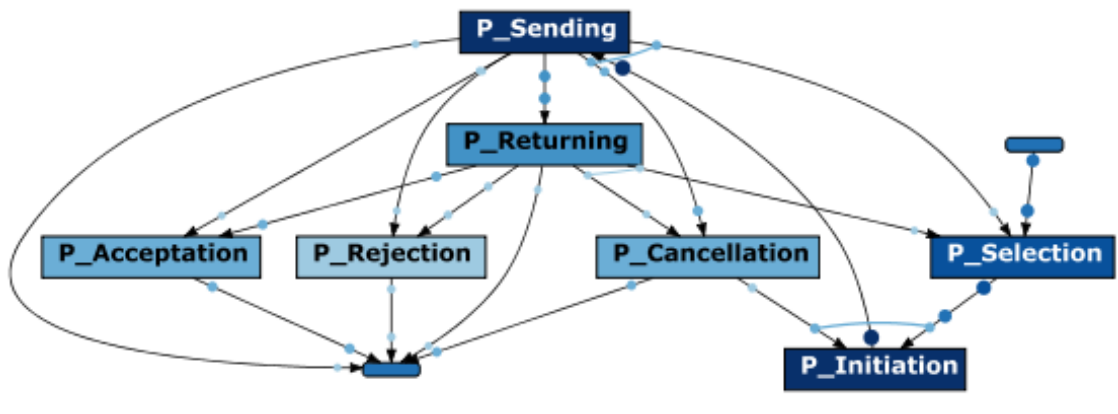
Figure 20: Endstate APP_Rejected

(a) C-net with 0.2 filtered frequency

(b) C-net without filtered frequency

Figure 21: C-Nets of the proposal process

- Selection →Initiation →Sending →

  - Returning →Acceptance →Done
  - Cancellation →Returning →Acceptance →Done
  - Returning →Acceptance →Cancellation →Done
  - Returning →Cancellation →Acceptance →Done
  - Cancellation →Done
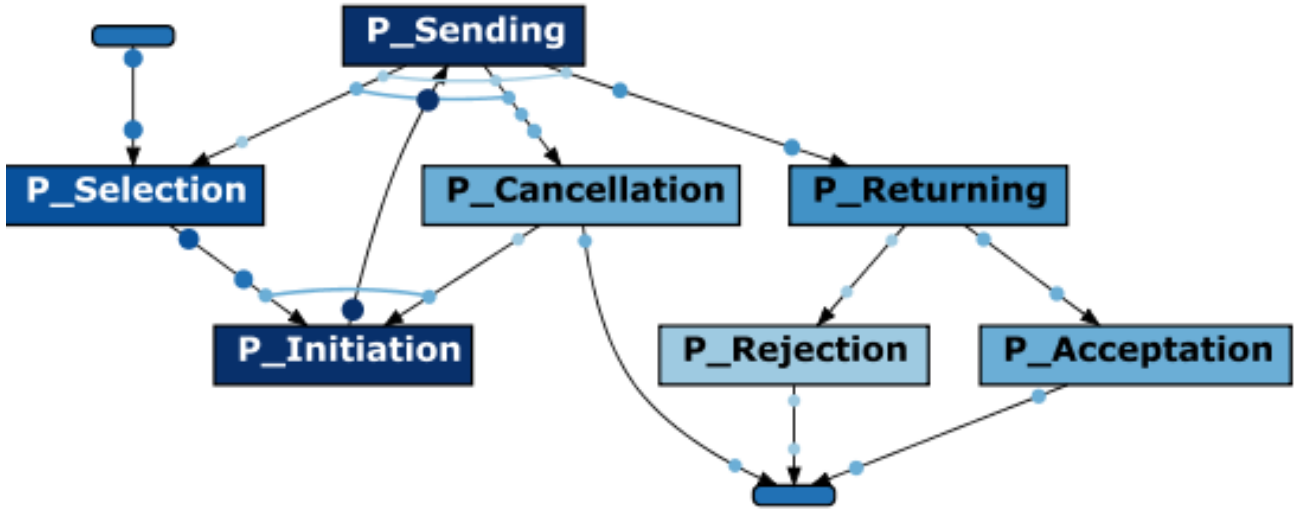
3.5.2 Extra C-net

Figure 22: C-net with 0.151 filtered frequency

Just for comparsion I also checked the 0.151 filtered c-net and there you can see, that in real data a loop can be found between Sending, Cancellation, Initiation and Selection. So this behaviour makes the processes probably slower.

3.6 Own Petri net of the proposal process

Based on my analysis of the C-net and also the petri net.The first problem I saw, was that the start event can occur in between again. Because of that you have a deadlock if you would build it without transitions, that can occur more than once. Also the cancellation and selection or just cancellation or split makes problems.
Then just checking the different variants and the occurences I found out, that already just the traces occur at least 1% of the time have "P_Selection" as start state and as in between state. So I decided just to emphasize the main three main variants. I think this is the behaviour as it should be.
The traces that I considered were:
"P_Selection"→"P_Intiation" →"P_Sending"→

- "P_Returning" →"P_Acceptation" (1.537 cases, 11.74%)

- "P_Returning" →"P_Rejection" (1.132 cases, 8.65%)
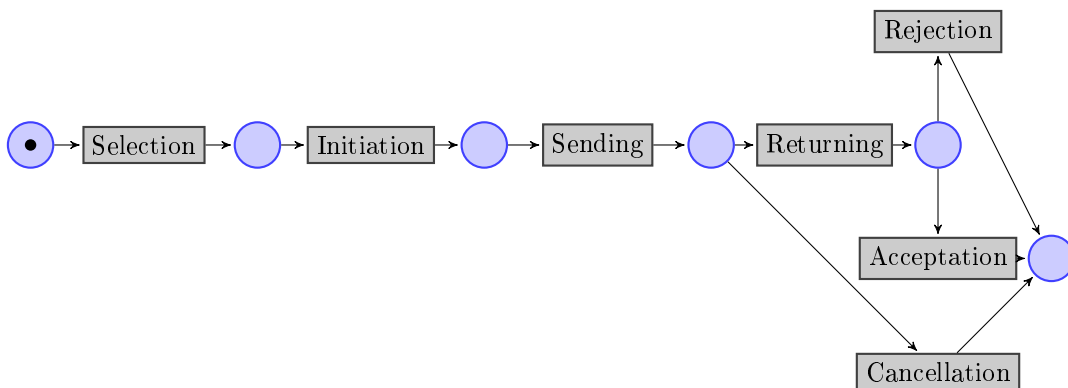
- "P_Cancellation" (574 cases, 4.39%)

Figure 23: Own petri-net

3.7 Analysis of the performance of Application and work process

Just the events beginning with "App_..." or "W_..." are required. The resulting data set is saved as "Filtered Work and App". I also just considered the workdataset (data beginning with "W_..") saved as "Filtered Work"
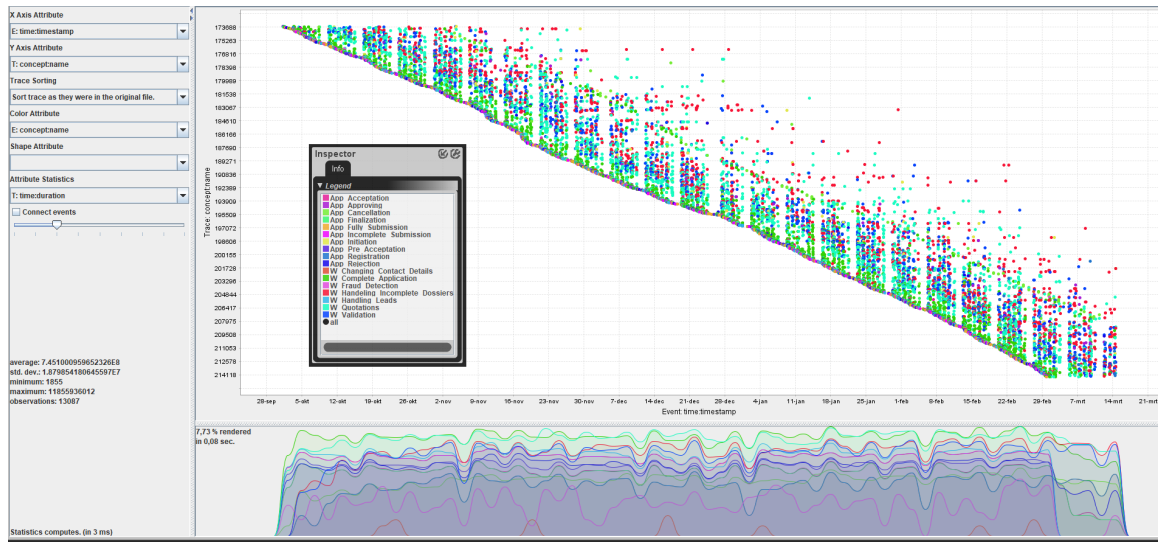
Figure 24: Dotted chart showing the time of events

3.7.1 General Details of the data set

The data set is collected between the 1st of Oct 2011 (saturday), 00:38:44 and the 14th of Mar 2012 (Wednesday), 16:04:54 and contains 13087 cases with 230956 events.

In figure 24 the dotted chart can be seen. Having a closer look at this I saw again gaps on sunday. On sundays just "App_Incomplete_Submission" and "App_Rejection" is been executed. Furthermore is in the left below corner to see what is the average duration of a case, 8 days 14 hours 58 minutes and 20.10 seconds, the minimum duration, 13.087 seconds, and the maximum duration, 137 days 5 hours 18 minutes and 56.01 seconds. Both is given in milliseconds.

The data set has 29 events (occurence relative): "W_Complete_Application" (10.377%), "W_Complete_Application" (10.18%), "W_Quotations" (9.948%), "W_Quotations" (9.701%), "App_Fully_Submission" (5.666%), "App_Incomplete_Submis (5.666%), "W_Handeling_Incomplete_Dossiers" (4.939%), "W_Handeling_Incomplete_Dossiers" (4.936%), "W_Validation" (3.418%), "W_Validation" (3.417%), "App_Rejection" (3.306%), "W_Complete_Application" (3.192%), "App_Pre_Acceptation (3.19%), "W_Quotations" (2.872%), "W_Handling_Leads" (2.554%), "W_Handling_Leads" (2.553%), "App_Acceptation" (2.214%), "W_Validation" (2.175%), "App_Finalization" (2.171%), "W_Handling_Leads" (2.066%), "App_Cancellation" (1.215%), "W_Handeling_Incomplete_Dossiers" (1.032%), "App_Initiation" (0.972%), "App_Approving" (0.972%), "App_Regis (0.972%), "W_Fraud_Detection" (0.117%), "W_Fraud_Detection" (0.117%), "W_Fraud_Detection" (0.054%) and "W_Changin (0.005%).

There are 3668 different variants of traces.

Maximal 162 events are executed in a case and minimal 3. The mean of events per class is 17.648.

All cases start with "App_Fully_Submission", but there are 12 different outcomes: "App_Rejection" (26.202%), "W_Validation" (20.975%), "W_Handling_Leads" (17.07%), "W_Complete_Application" (14.816%), "W_Quotations" (9.895%), "App_Cancellat (7.091%), "W_Handeling_Incomplete_Dossiers" (3.454%), "W_Fraud_Detection" (0.436%), "W_Changing_Contact_Details" (0.031%), "W_Validation" (0.015%), "App_Registration" (0.008%) and "W_Quotations" (0.008%).

3.7.2 Bottlenecks of the workflow-net

Using the **Mine with inductive visual miner** I first explored the workflow process. The configuration were changed to 100% of the pathes and show paths and sojourn time in the first step.
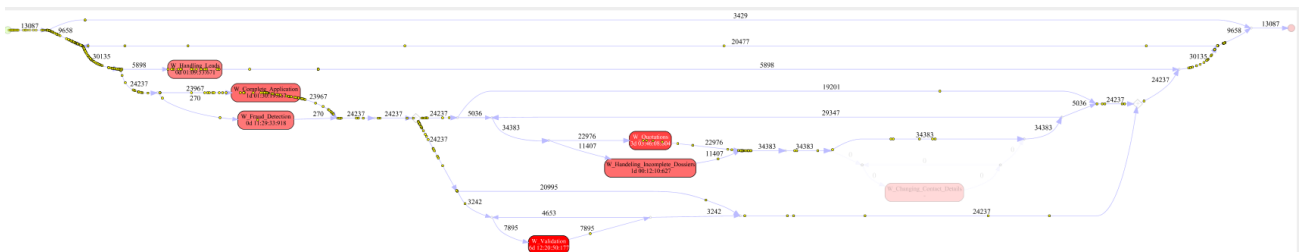
Figure 25: Workflow net with sojourn time

The dark red transitions are the transitions with a long preoceeding time, so our searched bottlenecks. The biggest problem we get from the "W_Validation" step (6 days 12 hours 20 minutes 50 seconds and 177 milliseconds). The second slow trensition is "W_Quotations" (3 days 3 hours 46 minutes 46 seconds 804 milliseconds). The rest is around 1 day. Having a look at the amount of times the two slow transitions has to be executed it is clear, that "W_Quotations" makes the process slow. Having a look at the event log visulization you can see, that in the 10 most common processes there are 2, appearing in 1.3% of the traces, that have at least one time "W_Quotations" and they also have "W_Validation" in it.

13

Changing the view to paths and queue lengths and having a look at the longest queues over time I had the same 2 as bottlenecks.

The next thing I controlloed was how many different persons this event can execute, but the result was that both event are executed by 53 for "W_Quotations" and 54 for "W_Validation". Conspicuous is, that for "W_Validation" the first most frequent appearing ressources execute the event in already 57% of the cases.

3.7.3 Recommendations based on the workflow-net

Having a look at the bottlenecks of the workflow-net it can be seen, that for the two bottlenecks a lot of the cases execute the events at least 2 times. This ist obviously to see, because there are just 3242 cases incoming to "W_Validation", but in total it is executed 7895 times. This make the waiting queue longer and so the overall executing time, too.

For "W_Quotations" the problem is similar.

So the first advice I would give is to overthink the procedure in both cases, such that it has to be done just one time, but then on a good manner. Also maybe the workload can be better spreaded if it is not specific or for "W_Validation" concentrated such that a few people know how to do it and just concentrate on this.

3.7.4 Bottlenecks of the application-net

Using the same tool I also had a look at the application process.

Figure 26: Application net with sojourn time

As to see in figure 26 the most events are done pretty fast, but the events in the end are really slow.

Figure 27: Zoomed in application net with sojourn time

Having a closer look at this it can be seen, that the events there take roughly 16 days. "Cancellation" even takes almost 18 days.

Taking a look into the ressources used it is clear, that the work is concentrated on a few persons and not good distributed in between. ("App_Registration", "App_Approving", "App_Initiation" 9 of 61) Just "App_Cancellation" is done by 58 of 61, but it still takes a high amount of time. Probably, because the peoble are busy with other stuff.

3.7.5 Recommendations based on the application-net

Based on the results I would recommend to check, if there is a possibility to accelerate the decision steps.

Figure 28: Application and workflow net with sojourn time

3.7.6 Combined model

Just to be sure I also executed the process for the complete data set.
In figure 29 the whole net with the sojourn time can be seen. For better understanding I zoomed in at the different critical parts.
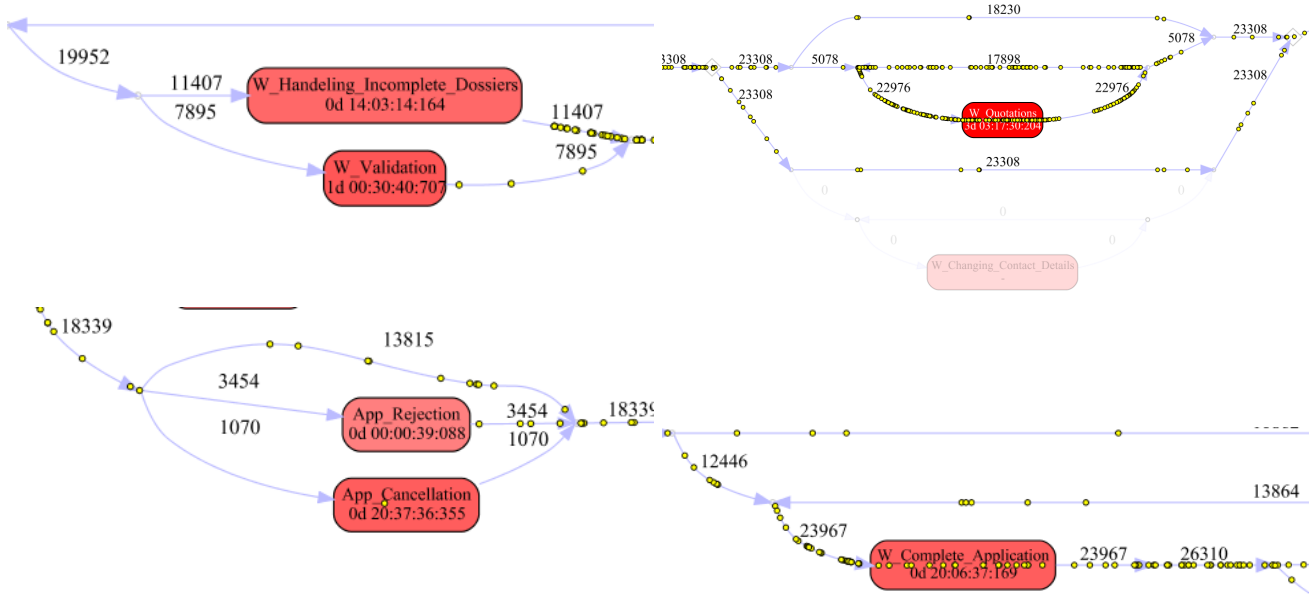

Figure 29: Application and workflow net with sojourn time

Like we expected the bottlenecks of the whole net are the same than for the two smaller ones. Just the sojourn times are smaller, because there are more events, that are executed in bettwen and the timestamps are closer to each other. Furthermore it is clear, that the workflow bottlenecks are worde, than the application bottlenecks. The amount of times those events have to be executed is for the workflow events much higher. This clarifys a little bit the high soujoun time, but the process has to be optimized on this parts.

3.7.7 Recommendations based on the combined net

Noticeable is, that there are a lot of traces going back and the part has to be executed again. This makes the process slower and also increases the workload for the different steps. So this backloop should be rethinked.

4 Further questions

4.1 Are some decisions in any of the models driven by the application's amount?

For answering this question I had a look at the dotted charts of the whole data set and the three filtered ones, "Filtered App", "Pdoubledfiltered", "Filtered Work". As x-axis I chose "C:Activity classifier" and y-axis "T:AMOUNT_REQ". Also I sorted the traces by the "AMOUNT_REQ". Just for the layout I changed the color to "C: Activity classifier".
Figure 30 shows clearly, that there is **no correlation** between activity and requested amount.

4.2 Are there clear indications that the same employees are always involved in cases that are declined, approved or cancelled?

For this question I concentrate on the application data set and filter it three time: "Filtered App with approv", where as end event just "APP_Initation", "APP_Registration" and "APP_Approved" are possible as end event, based on the result of the 6th question before, where you can see, those three always happen together, "Filtered App with Rej", end event just "APP_Rejected" and "Filtered App with Canc"
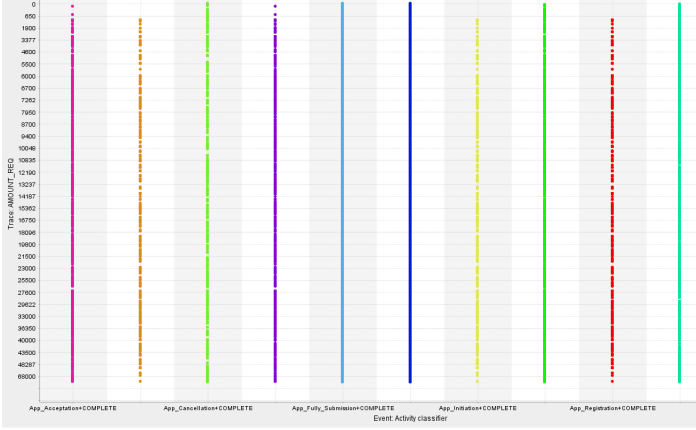Those filtered data sets I used for the **Interactive Data aware Heuristic Miner**, where I chose resource in place of activity to create a model.
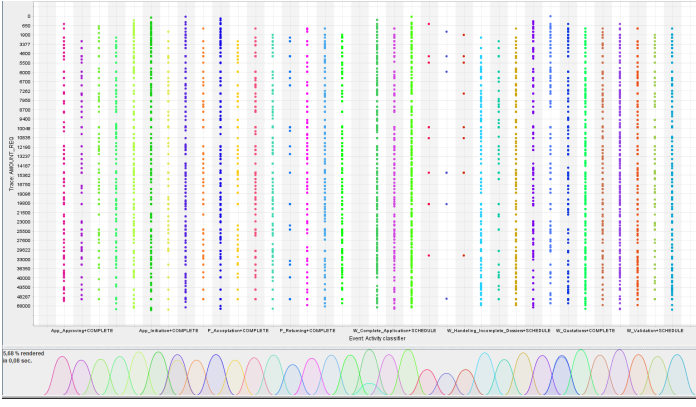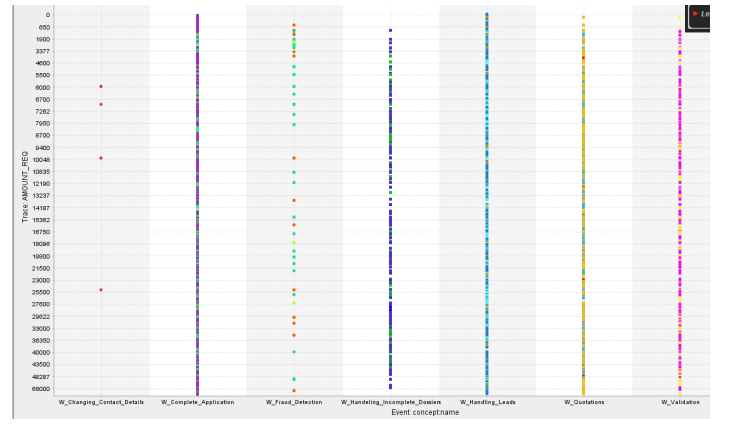
(a) Whole data set
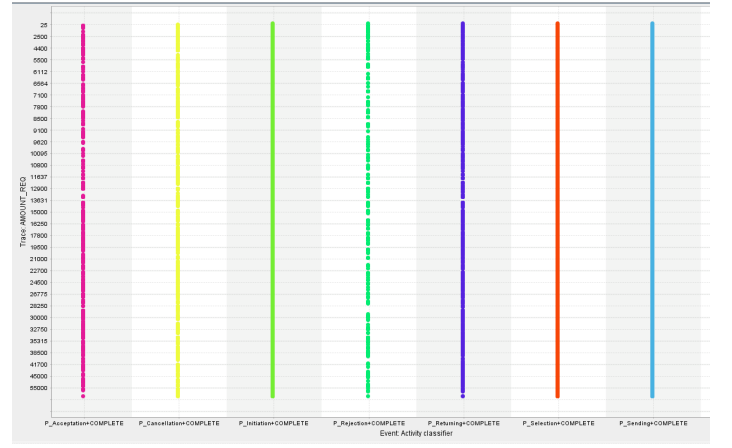
(b) Proposal data set

(c) Application data set

(d) Workflow data set

Figure 30: Dotted chart for the application's amount

4.2.1 Approved

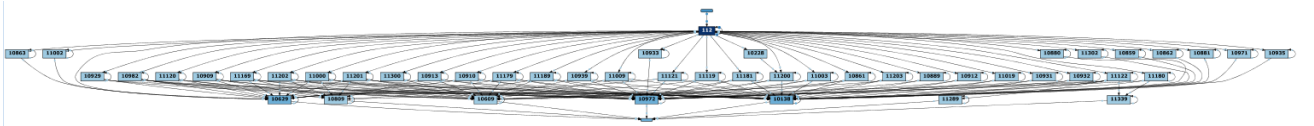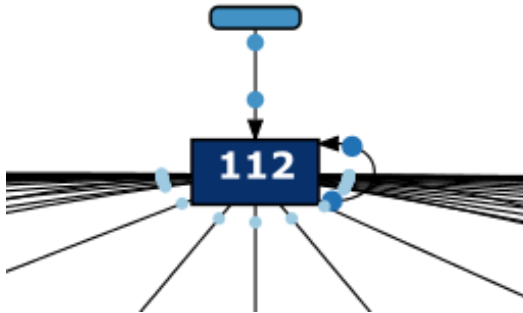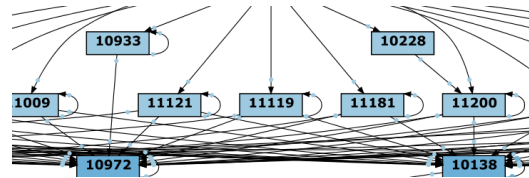To see what are the main resources used I had a look at the c-net.

Figure 31: Causal net of the approved application lifecyle without filtering

(a) Beginning of the traces

(b) Middle of the traces

(c) End of the traces

Figure 32: Zoomed in the c-net

There are three parts prominent, 32:

1. All traces start via **112**, 32a

2. 2 resources, 10933 and 10228, just have two successors, 32b

3. 7 resources execute the last event, 32c

4.2.2 Rejected

(a) Causal net of the reject application lifecyle without filtering

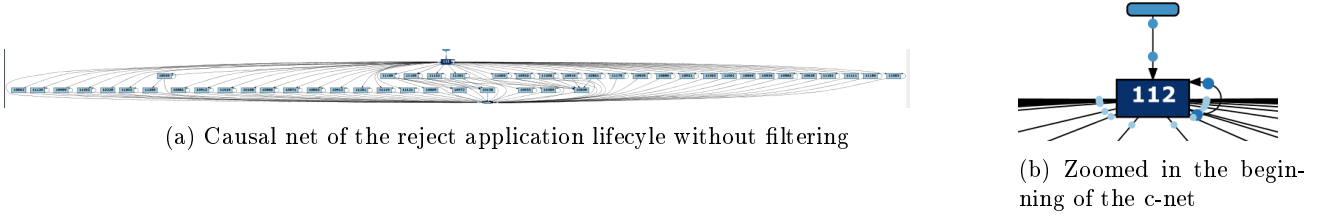(b) Zoomed in the beginning of the c-net

Figure 33: Rejection resource c-net

Figure 33a shows the whole c-net and in 33b it is more clear, that again **112** starts all traces.

4.2.3 Cancellation

(a) Causal net of the cancellation application lifecyle without filtering
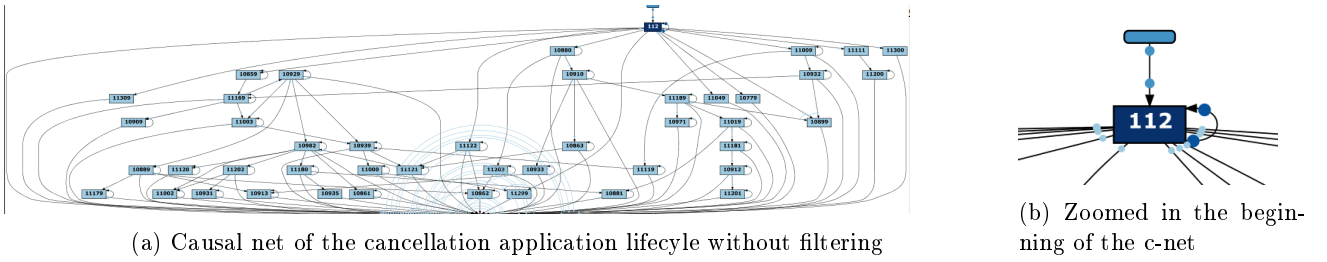

(b) Zoomed in the beginning of the c-net

Figure 34: Cancellation resource c-net

Figure 34a shows the whole c-net and in 34b it is more clear, that again **112** starts all traces.

4.2.4 Summary of those results

So what gets pretty clear is that **112** decides over the next steps in the beginning. Having a look at the resources following it gets obvious, that different ressources work on the 3 sorts of traces. So it seems to depend on 112, which decision will be made.

4.3 Comparsion of the throughput times

For this investigation I created a data set with the combination of rejected and cancelled traces, "Filtered App with rej and canc".
Then I looked the times up in the dotted chart like I did before.

	Minimum	Mean	Maximum
Approved	700910	$1.4393 * 10^9$	7419735534
Rejected of cancelled	1855	$5.613 * 10^8$	7901736161

Figure 35: Throughput times

In figure 35 it can be clearly seen, that an approved traces takes 2.56 times so much time, than a rejected or cancelled trace.

5 Conclusion

Recapping all results from the last parts I would say, that there a few points throughout the event, where it could be done better. There are 2 big bottlenecks in the process, which should be optimzied. Also the use of the resources could be optimized. The difference between the fastest trace is much faster than the slowest one, which should be analyzed in more detail for all process parts.