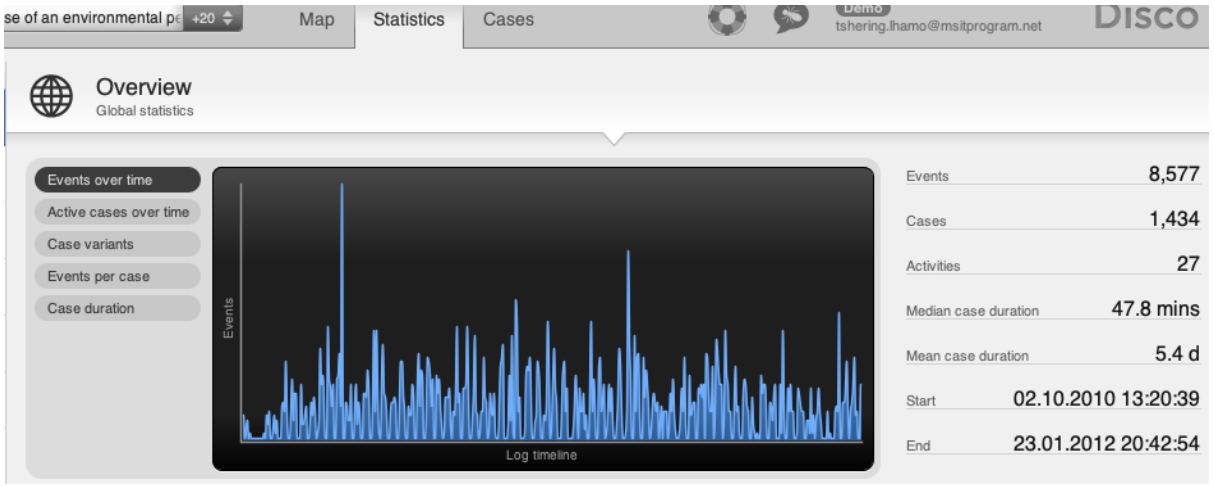
Applying Process Mining on Real Data

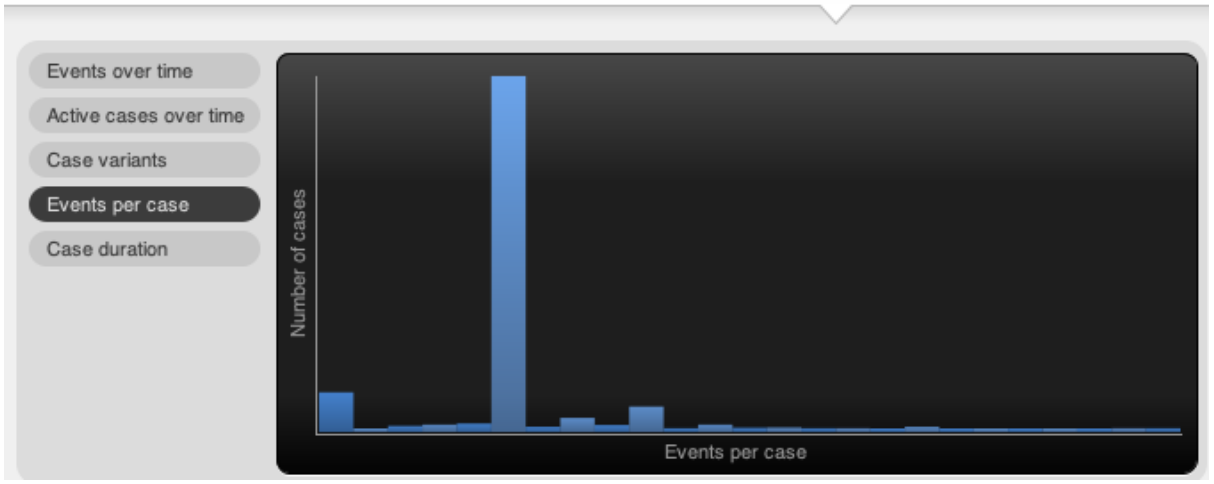
Q. 1

Open the event log ('Receipt phase of an environmental permit application process (_WABO_) CoSeLoG project.fbt') in Disco and switch to the 'Statistics' view.

Without switching to other views, use the statistics view to answer the following three sub-questions:

1. How many events are there on average per case?



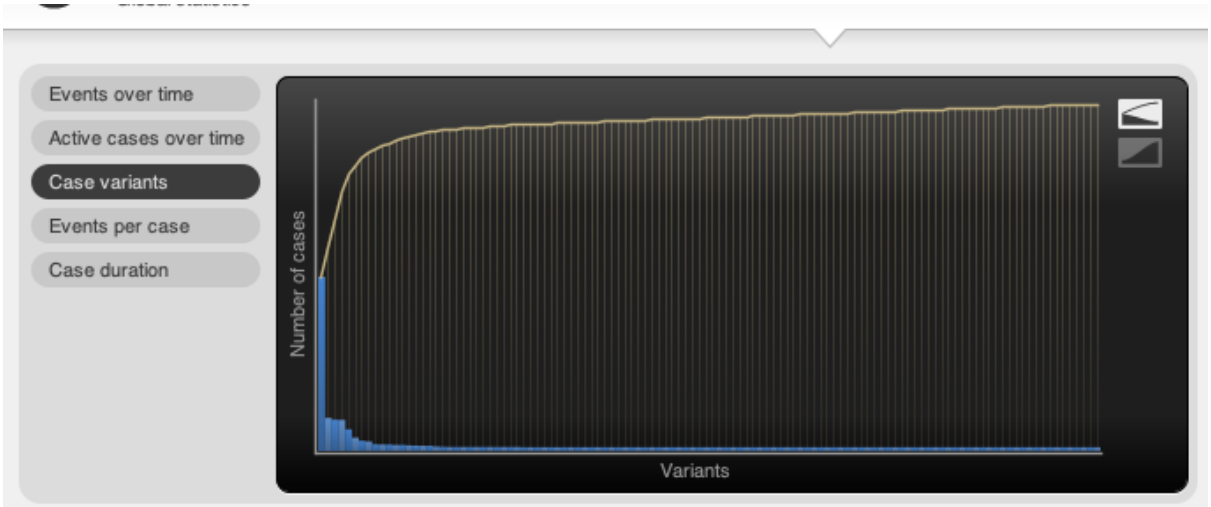


Ans: Total events = 8577

Total cases = 1434

Average events/case = 8577/1434 = 5.98

2. Can you indicate whether each case seems to be unique or whether many cases follow the same activity sequence?



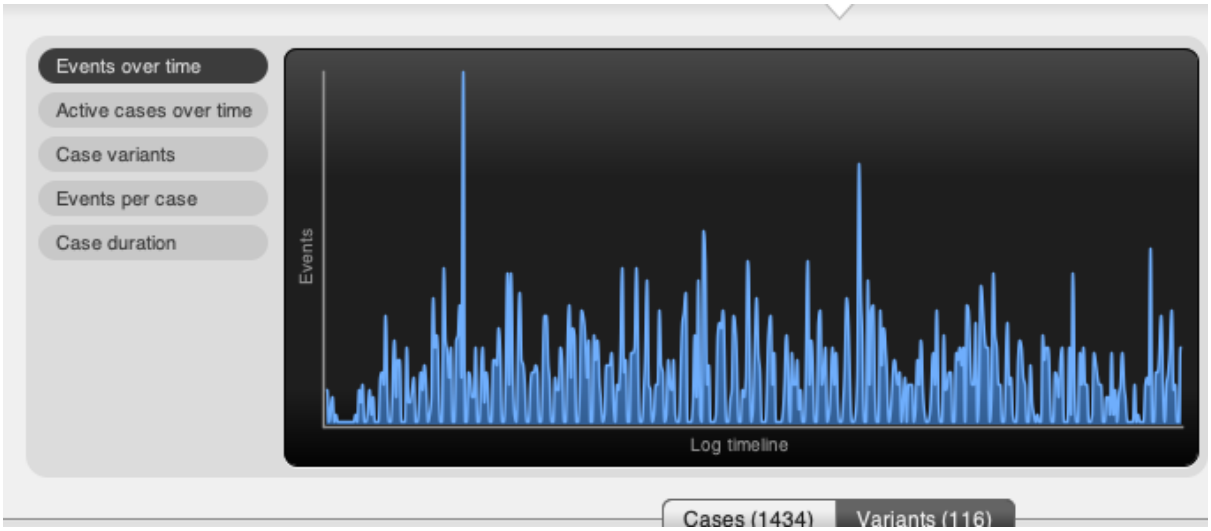
Cases (1434) | Variants (116)

Variant	Cases	Events	Median duration	Mean duration
Variant 1	713	6	8 mins, 1 sec	4 days, 8 hours
Variant 2	123	6	1 day, 23 hours	5 days, 17 mins
Variant 3	116	1	0 millis	0 millis
Variant 4	115	6	1 day, 4 hours	4 days, 2 hours
Variant 5	75	6	2 days, 3 hours	4 days, 19 hours
Variant 6	40	6	7 mins, 54 secs	5 days, 12 hours
Variant 7	28	6	7 mins, 17 secs	2 days, 23 hours
Variant 8	25	10	10 days, 3 hours	15 days, 2 hours
Variant 9	13	6	15 mins, 22 secs	1 day, 5 hours
Variant 10	12	5	7 days, 14 hours	26 days, 17 hours
Variant 11	12	6	4 mins, 50 secs	4 days, 2 hours
Variant 12	10	10	12 mins, 2 secs	1 day, 9 hours
Variant 13	10	6	9 hours, 12 mins	2 days, 4 hours
Variant 14	8	8	19 days, 6 hours	18 days, 16 hours
Variant 15	7	4	2 mins, 54 secs	15 mins
Variant 16	6	8	10 days, 16 hours	8 days, 22 hours

Variant	Cases	Events	Median duration	Mean duration
Variant 31	1	9	10 mins, 17 secs	10 mins, 17 secs
Variant 32	1	10	20 hours, 37 mins	20 hours, 37 mins
Variant 33	1	7	5 days, 19 hours	5 days, 19 hours
Variant 34	1	4	22 days, 5 hours	22 days, 5 hours
Variant 35	1	5	11 days, 21 hours	11 days, 21 hours
Variant 36	1	3	37 days, 23 hours	37 days, 23 hours
Variant 37	1	3	2 mins, 41 secs	2 mins, 41 secs
Variant 38	1	10	16 mins, 12 secs	16 mins, 12 secs
Variant 39	1	8	6 hours, 43 mins	6 hours, 43 mins
Variant 40	1	14	7 days, 2 hours	7 days, 2 hours
Variant 41	1	10	1 day, 23 hours	1 day, 23 hours
Variant 42	1	10	11 days, 22 hours	11 days, 22 hours
Variant 43	1	8	8 days, 5 hours	8 days, 5 hours
Variant 44	1	8	5 days, 31 mins	5 days, 31 mins
Variant 45	1	14	9 days, 22 hours	9 days, 22 hours
Variant 46	1	9	24 days, 23 hours	24 days, 23 hours

Ans: There are 116 sequence variants out of 1434 cases. About 50% of all cases follow one activity sequence. so, most cases are *not* unique, but variants 31 to 116 are single case variants, therefore, these cases follow a unique activity sequence.

3. What is the main observation that can be made from the 'Events over time' graph?



Ans: From the graph, I noticed that the amount of activities per day is in the range of 0 to 147. There are many groups of 2 days or more with zero activity, so it must be weekends and holidays.

Q. 2

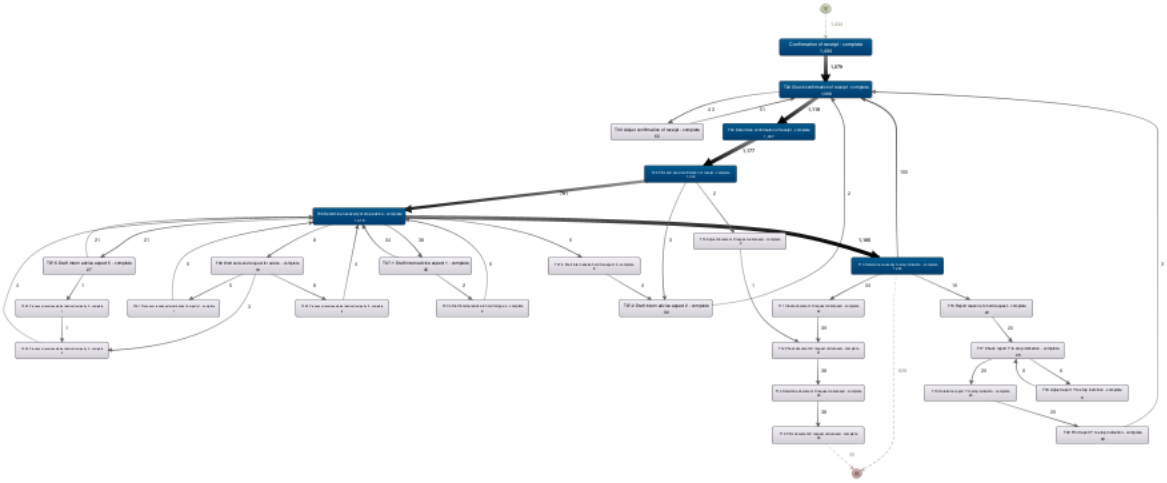
While still in Disco, switch to the 'map' view to display a process map.

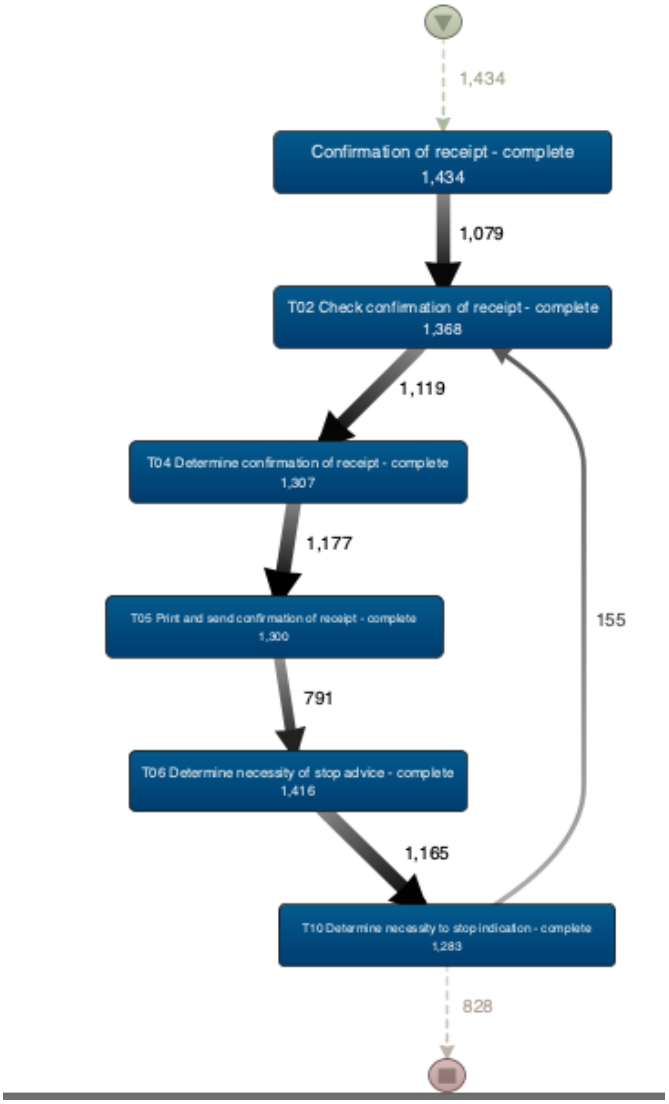
Using the map view, change the activity and path detail settings in order to create a comprehensible process map (e.g. a process map that could be printed on one A4 or letter paper or shown on a single computer screen while still being readable in full). In your answer, include the settings you used for both the activity and path sliders.

1. Discuss this process map, what is the main process?

   Ans: The setting of the process map for 100% activity and 0% path sliders contains too many different activities. On the other hand, only a minor number of traces finish at the end event. So to comprehend a little better the number of activities is decreased and the number of paths is increased.

   The best setting is at 50% activities and 17% paths in the slider. Now nearly 90% of traces finish at the end event and the most important activities are visible.



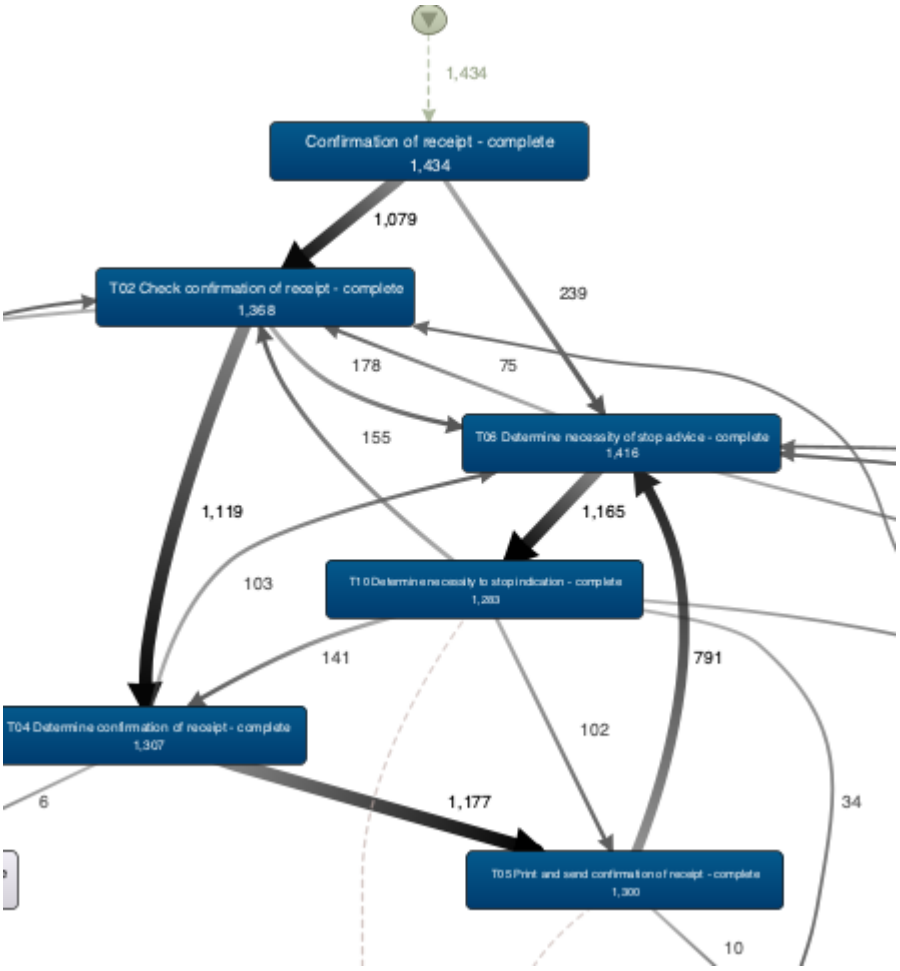


The main parts of the process are ‘Confirmation of receipt’ and ‘Determine the necessity of stop’. ‘Confirmation of receipt’ is self-declaring, whereas ‘necessity of stop’ is not, so it is difficult to comprehend what might be stopped.

2. Which activities and paths between activities are frequent?

   Ans: The main process consists of 2 groups of most frequent activities, conducted a few times parallel but most times in sequence. About 90% of all cases contain these 6 activities.

   1. Confirmation of receipt
   2. Check confirmation of receipt
   3. Determine confirmation of receipt
   4. Print and sent confirmation of receipt
   5. Determine necessity of stop advice
   6. Determine necessity of stop indication



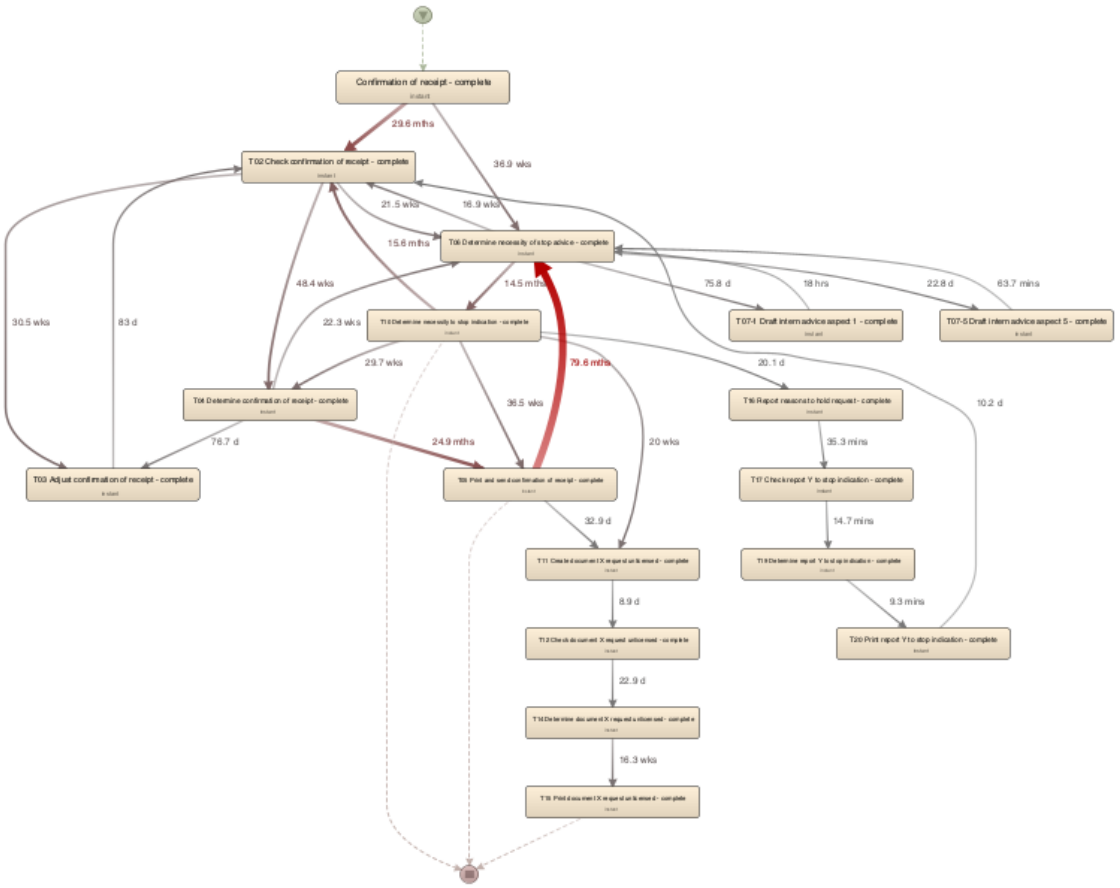
Q. 3

While still in Disco, and while using the same process map (e.g. do not change the activity and path settings), switch to the performance projection.

Discuss where the process takes the most time, e.g. where there are possibilities for improvement. Relate these times (of the bottlenecks) to the time spent in other parts of the process. In other words, discuss how severe the bottleneck is with respect to the time spent on other activities.

Also, explicitly mention the performance metric chosen (e.g. total, mean, median, or max) and why you have chosen this setting.

Ans:

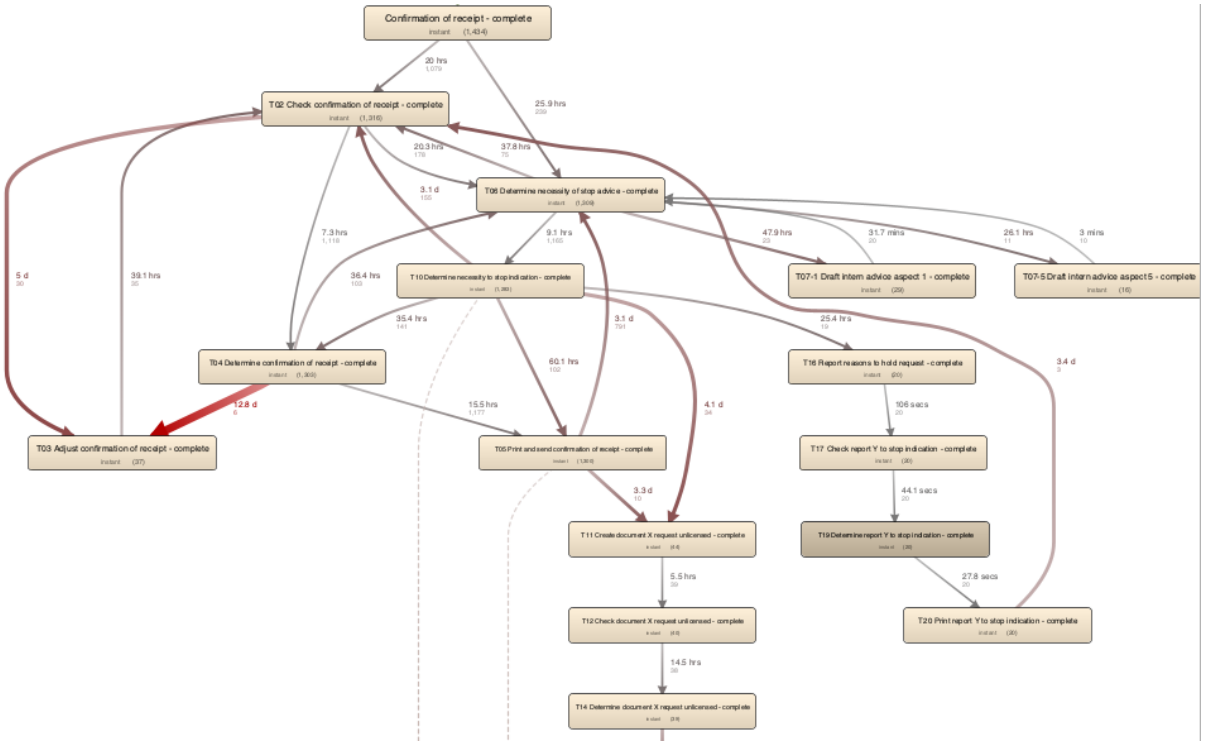


Settings kept at 50% activities and 17% path in the slider. The graph above shows the performance projection of the mean duration. Secondary metrics is ‘Absolute frequencies’

Most of the time spent affecting a majority of cases is the “T06 determine necessity to stop advice activity” as can be seen in the graph, 79.6 months in 791 activities. If this could be shortened significantly, the response times would be improved and stakeholders do have to wait long which in turn reduces cost.

The mean duration of activities and cases (full traces) is relevant if you want to find out how long external stakeholders of a case have to wait to finish the activity.
There are other longer events but are less frequent thus, they are not significant as the above-mentioned.

Secondary metrics as case frequency



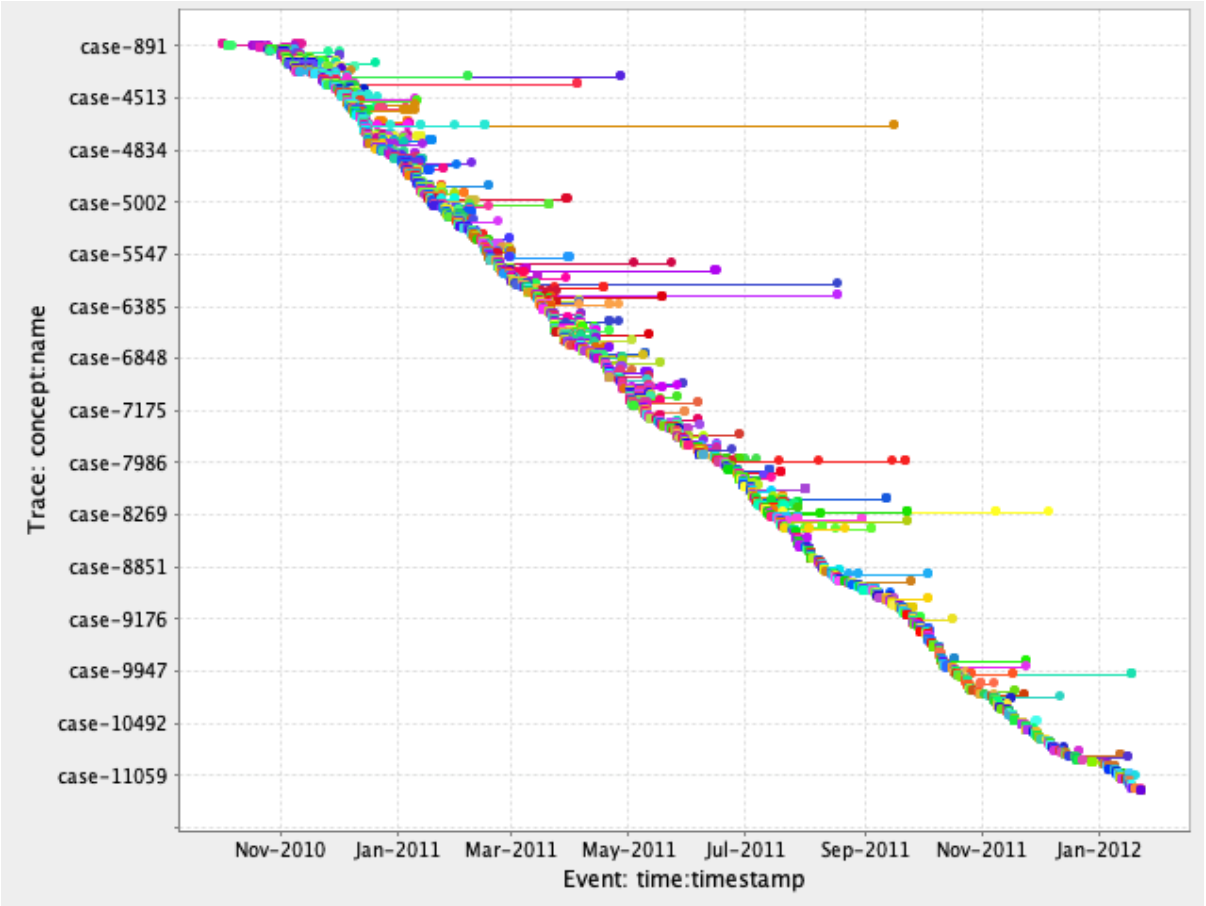
Q. 4

Now load the original event log in ProM. Visualize the event log using the Dotted Chart or XDottedChart visualizer (by pressing the 'eye'-icon with the event log selected and switching to the Dotted Chart or XDottedChart visualizer).

Using the Dotted Chart, answer the following questions:

1. Is the arrival rate of new cases constant? If not, when are there fluctuations? If yes, how can we see this from the Dotted Chart?

Ans:

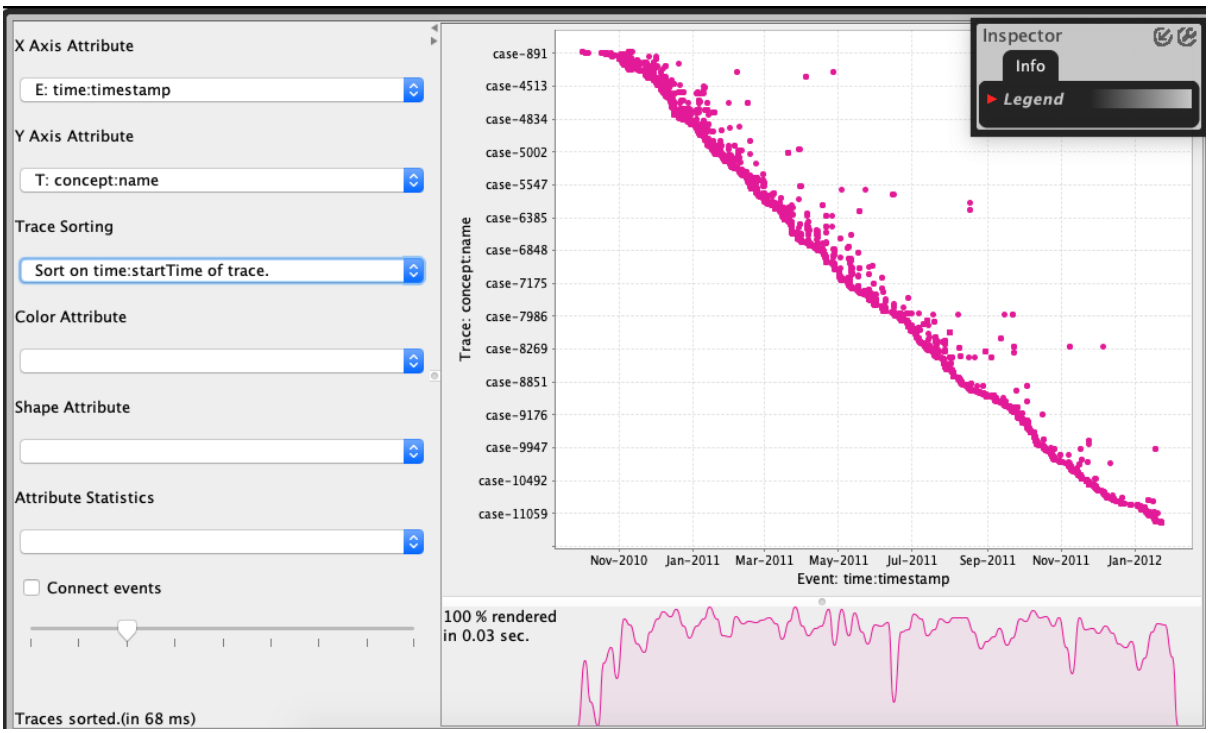


There are small fluctuations in the arrival rate as shown in the graph above. New cases have a slightly lower rate from the September month.

2. Can you observe a change in the global process?

Note that you don't need to change the component, time or coloring settings. You can however re-sort the traces on the time of the first event, and zoom in or out if you want.

Ans: While sorting by time: startTime of Trace gives an overview of case start rate. It is magnified here:



Q. 5

You are now asked to discover a Petri net on the event log. However, the unfiltered event log results in an incomprehensible Petri net. Therefore, you are allowed to run the 'Filter log using simple heuristics' plug-in *once* on the original event log to discover a Petri net on the filtered event log.

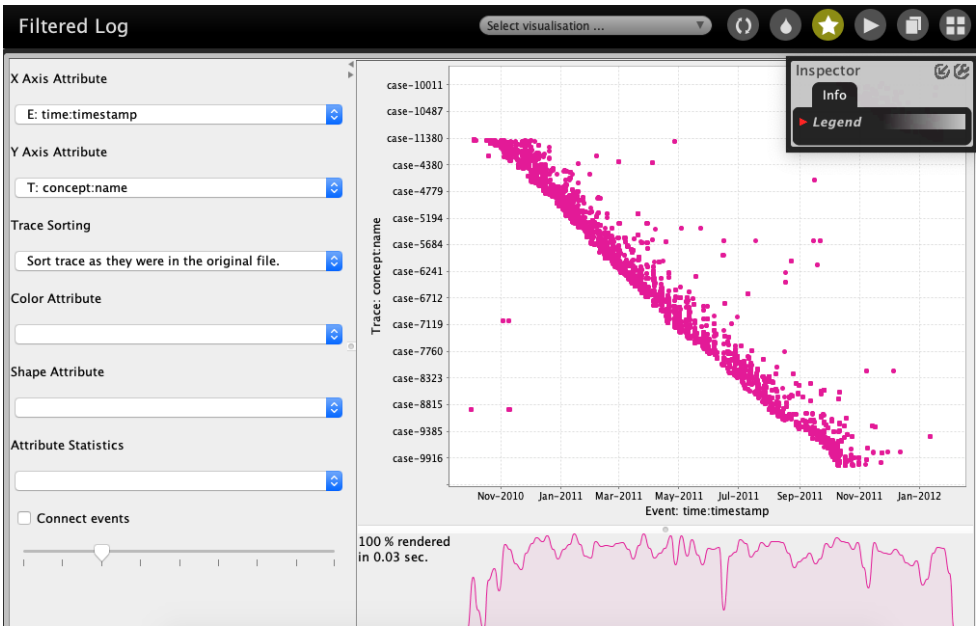
1. Clearly indicate which settings you have used for the 'Filter log using simple heuristics' plug-in.

2. Explicitly motivate the filtering settings chosen, why did you pick this percentage or selection of activities?

Ans: **Settings used for the 'Filter log using simple heuristics' plug-in.**

1. Click on “Actions” icon.
2. Search for “Filter Log”.
3. Select “Filter Log using Simple Heuristics”.
4. Click on “Start” button.
5. Change Log name to “Filtered Log” .
6. Click on “Next” button.
7. Select “Select top percentage” to 100% because there is only 1 Start event.

8. Click on “Next” button.
9. Select “Select top percentage” to 100% because ideally keeping all End events would be critical in understanding the process.
10. Click on “Next” button.
11. Select “Select top percentage” to 96% because this Event filter criterion discards many events and therefore many arcs in the resulting Petri net.
12. Click on “Finish” button.



3. Discuss and argue which plug-in (or chain of plug-ins) you have used to discover a Petri net, for instance by comparing two or more plug-in results and arguing why one of the Petri nets is better.
4. Explain the (best) Petri net: what is the main process and what are notable parts of the Petri net?

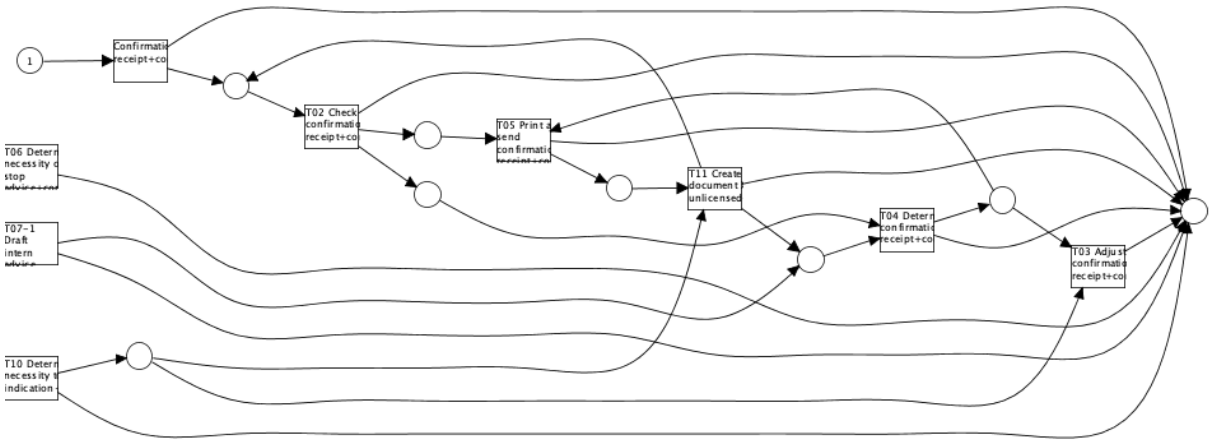
   Note that this question requires you to experiment with different filtering settings and discovery plug-ins. You are not required to describe *everything* you have tried but found unsuccessful. Only describe the successful combination of plug-ins and its result(s) and argue why your final result is 'good'.

   Suggested list of plug-ins or plug-in chains to produce a Petri net:

   - Mine for a Petri Net using Alpha-algorithm
   - Mine for a Petri Net using ILP
   - Mine for a Heuristics Net using Heuristics Miner followed by Convert Heuristics net into Petri net
   - Mine for a Petri net with Inductive Miner

Ans:

Using Alpha-algorithm



1. Click on “Actions” icon.
2. Add filtered log to “Input”.
3. Select “Mine for a Petri Net using Alpha-algorithm” plug in.

4. Click on “Start” button.

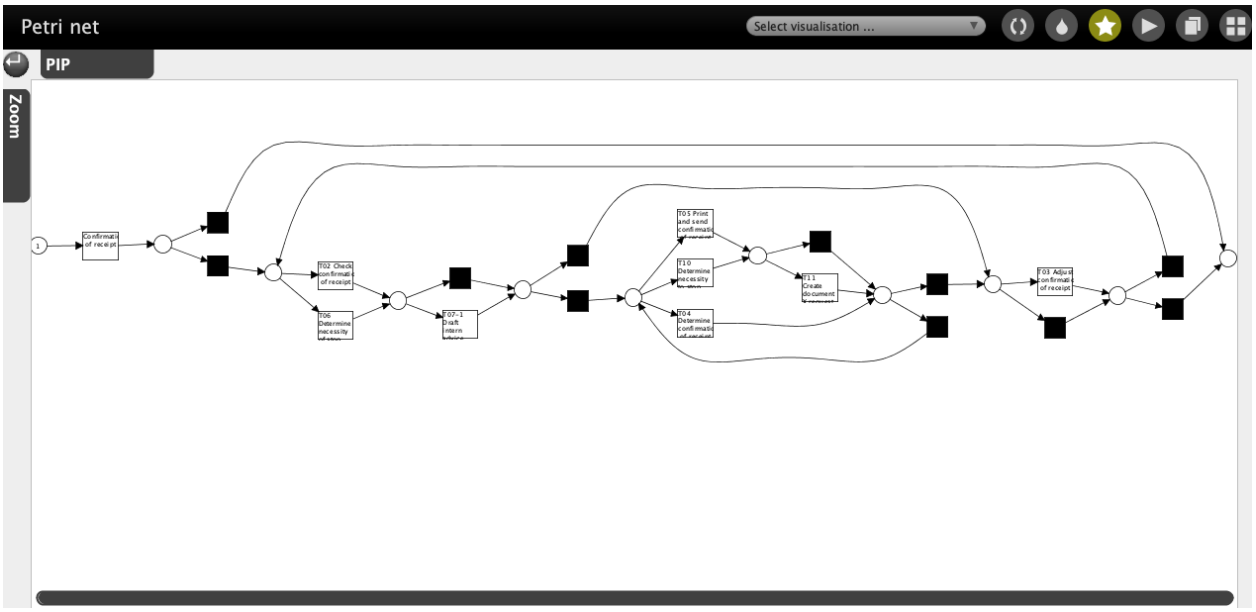
The Alpha algorithm has discovered 9 transitions & 9 places. However, transitions T06, T07-1 & T10 are not integrated well into the rest of the control-flow.

However “Mine for a Petri Net using Alpha-algorithm” plug-in is not robust to logs that contain noisy data (like real-life logs typically do).

The mining plug-ins “Flexible Heuristics Miner”, “Inductive Miner”, and “Fuzzy Miner” is good to go for real-life logs. When you have real-life data with not too many different events, or when you need a Petri net model for further analysis in ProM, Heuristics miner is best.

U**sing Inductive Miner**

1. Click on “Workspace” icon.
2. Select filtered_event log.
3. Click on “Actions” icon.
4. Search for “Inductive” plug-in.
5. Select “Mine Petri net with Inductive Miner” plug-in.
6. Click on “Start” button.
7. Change “Variant” option from default of “Inductive Miner - infrequent” to “Inductive Miner” because the default option drops T04 transition probably due to infrequent cases containing it. We want to keep this transition so that we can compare the different Petri nets with the same set of transitions.
8. Click “Finish” button.



Nearly as good results can be achieved with other miners, but none gave a better petri net and none could be used to find the optimal filter settings for the best compromise of detail and drop of distorting traces.

The essential step to discover a comprehensible Petri net from the supplied log data is to filter out classes of traces with only a few class members but which are complicating the process model without giving extra insights.
The major challenge is setting the filters to such a level that an optimum of the real process variants is captured. Whether a class of traces is included or not for a good process model should not depend on miner chosen for a Petri net. But there are differences in features a process discovery method is able to guarantee.

Q. 6

The organization has a process model that describes the 'should be' process (i.e. a normative process model). Load the file 'normativeModel.pnml' into ProM and apply conformance checking on this process model, and on the full unfiltered original event log.

1. Include a screenshot of the part of the normative process model, with the conformance information projected onto it, that shows where most of the deviations occur.
2. What is the replay fitness (the 'trace fitness' statistic) of the event log on the normative process model?
3. Select the transition 'T06 Determine necessity of stop advice+complete' (on the top left of the model) and discuss its element statistics: how many times is the transition executed correctly and how many times incorrectly?
4. Using the element statistics of transition 'T06 Determine necessity of stop advice+complete', what can you say about the (in)correct execution of this activity?

Instructions to align the process model with the event log:

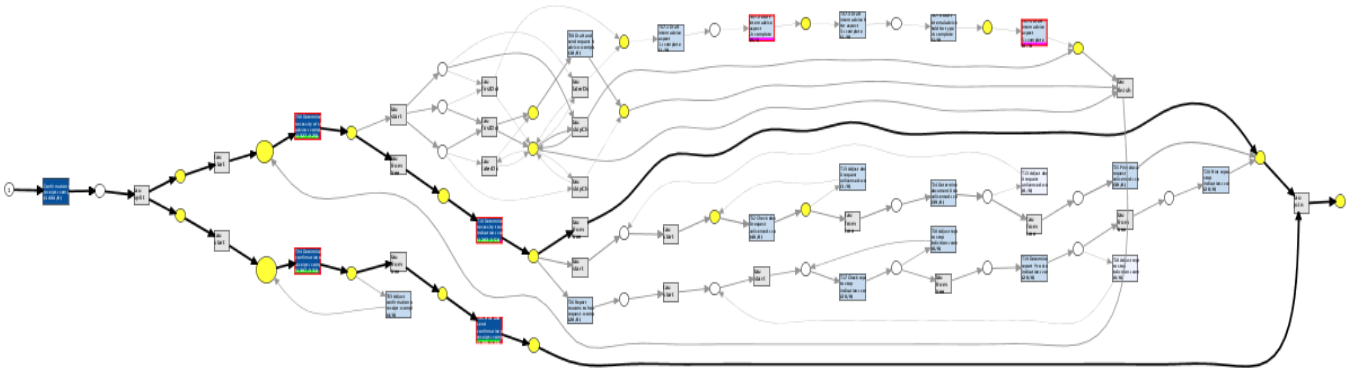
1. Import the normative model using the 'PNML Petri net files' importer.
2. Select the imported normative Petri net and the event log, start the plug-in called ‘Replay a Log on Petri Net for Conformance Analysis’ (not the variant with performance!), and click 'yes' in the 'No Final Marking' pop-up.
3. Select the 'sink' place on the left (note: do not select '0-sink' etc.) and click the button 'Add Place >>' to add the place 'sink' to the candidate final marking list. Now click 'Finish'.
4. Click 'Finish' in the mapping wizard.
5. Click 'No, I've mapped all necessary event classes' to indicate that some events are not present in the normative model.

6. Now click 'Next' and 'Finish'. The normative process model is shown with conformance information projected onto it.

If you followed these instructions exactly you do not need to mention these steps in your answer.

More information regarding this conformance technique is provided in lecture 4.7: 'Aligning observed and modeled behavior' (and to a lesser extend in the lectures 4.3 through 4.6).

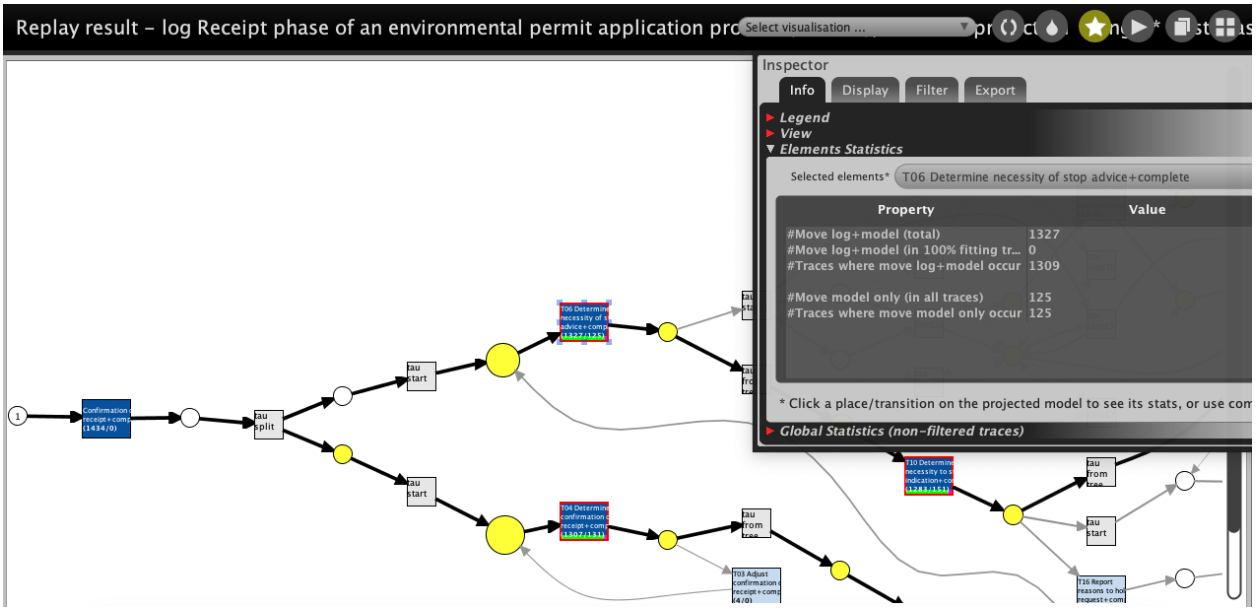
Ans:



The replay fitness (the 'trace fitness' statistic) of the event log on the normative process model is 0.84254. T10 has the maximum deviations (151). T06 has the minimum(125) amongst the frequent trace variants.

The transition 'T06 Determine necessity of stop advice+complete' (on the top left of the model) was tested with 1,434 traces in the event log. Out of those 1,309 (91%) were synchronous moves in both the model & log.

Amongst those 1,309 traces, T06 was fired synchronously for 1,327 times (i.e. some traces fired T06 multiple times). For 125 traces, T06 was fired in the model only i.e. T06 was not fired in the event log 125 times when it was supposed to.



Q. 7

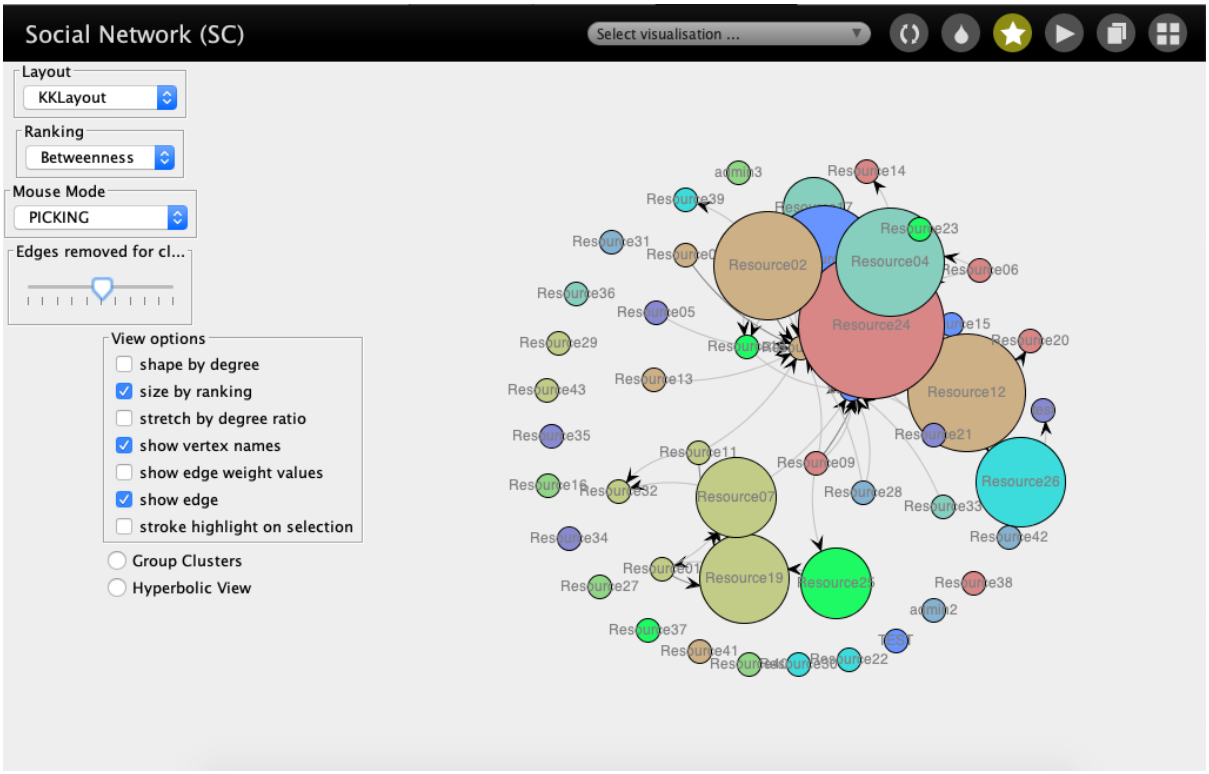
The final analysis you have to perform on the original event log is a resource analysis, e.g. looking at the user behavior in the event log.

1. Use the plug-in 'Mine for a Subcontracting Social Network'. Note that subcontracting means that if individual *j* frequently executed an activity in-between two activities executed by individual *i*, then individual *i* subcontracted work to individual *j*.

   Answer the following question using this view: Can two or more groups of users be distinguished? Explicitly discuss the settings you have used in the resulting visualization.

2. Again use one of the two Dotted Chart plug-ins. For the XDottedChart change the component type to 'org:resource'. If you use the Dotted Chart visualizer change the 'Y Axis Attribute' to 'C: Resource classifier' and the color attribute to 'C: Activity Classifier'. Answer the following two questions using this view:
3. Are all users executing activities from the start of the event log, or are some users joining later?
4. Are users mainly executing particular activities or are most users executing most of the activities?

Ans:



1. Settings which I have chosen for resulting visualization
    1. Ranking: “Betweenness” shows resources acting in-between
    2. Edge removed for - to get distinguishing colors
    3. Size by ranking - to display the in-betweenness by size

We can distinguish 3 types of resource involvement as:

1. With big subcontracting involvement, e.g. Resource 24, 02, 12, 08, . . .
2. Several with some subcontracting, e.g. 33, 15, 13, ..., admin, . . .
3. Many with no subcontracting at all. 36, 30, 2, . . .

Observations:

1. A few resources were active only later, e.g. since Sept. 2011.
2. Some resources are involved only in one or very few different activities.
3. About half of the resources are active most times, the other half only occasionally.
4. Some conducted only one activity.