Process Mining in the Financial Industry: A Case Study
The BPI Challenge 2017
Student Category

Peyman Badakhshan[1], João Carlos Maciel[1], Bastian Wurm[2]

[1]University of Liechtenstein, Vaduz, Liechtenstein
{Peyman.Badakhshan, Joao.Maciel}@uni.li

[2]Vienna University of Economics and Business, Vienna, Austria
bastian.wurm@wu.ac.at

Abstract. Over the last years, process mining has proved to be an innovative and efficient method to discover, analyze, and predict business processes' behavior. In this report, we present our findings from the analysis of a personal loan application process from a financial institute in the Netherlands. The case study is conducted in the context of the BPI Challenge 2017. The real-life event log consists of 1,202,267 events and 31,509 cases. We analyzed different aspects of the process based on questions of main interest to the company - throughput times of several parts of the process, the influence of incompleteness to the final outcome, and the comparison of single and multiple offers and the frequency of conversations between the financial institute and customer. Moreover, the analysis is performed using Celonis and Disco process mining solutions.

Keywords: Process Mining, Process Analysis, Control Flow Model, Performance Model, Business Process Intelligence Challenge

1 Introduction

Business Process Intelligence Challenge (BPIC) 2017 is sponsored by "Minit" and "Celonis" held in conjunction with BPM 2017 Barcelona. This challenge is using the data set from the same financial institute with the aim of answering several questions from the process owner of the financial institute. These questions are addressed as:

1. What are the throughput times per part of the process, in particular, the difference between the time spent in the company's systems waiting for processing by a user and the time spent waiting on input from the applicant as this is currently unclear.
2. What is the influence on the frequency of incompleteness to the final outcome. The hypothesis here is that if applicants are confronted with more requests for completion, they are more likely to not accept the final offer.
3. How many customers ask for more than one offer (where it matters if these offers are asked for in a single conversation or in multiple conversations)? How does the conversion compare between applicants for whom a single offer is made and applicants for whom multiple offers are made?

In 2012, the BPI Challenge used data from a financial institute. Data that has since been used by many researchers for various papers. A new workflow system has been implemented in the company, advice from the BPI Challenge 2012 has been implemented and (due to the financial crisis) the case volume has gone up considerably [1].

This paper is structured as follows. First, the information about the data and the process is provided such as the explanation of events and activities, which helps to understand the process and it's the basis of further analysis over the event logs. Second, the analysis and answering of the main questions from the financial institute are provided. For each part of this section the understanding of the question comes in the beginning then follows with the analysis and answer. At the end, if the paper, the conclusion of this report is provided. The findings of the question number 1 are based on results from Celonis and the findings from question number 2 and 3 are based on Disco as a process mining solution.

2 Understanding the Data and the Process

In total, there are 1,202,267 events pertaining to 31,509 loan applications. For these applications, a total of 42,995 offers were created. There are three types of events, namely Application state changes, Offer state changes and Workflow events. There are 149 originators in the data, i.e. employees or systems of the company. For all applications, the following data is available [1]:

- Requested load amount (in Euro),
- The application type,
- The reason the loan was applied for (LoanGoal), and

- An application ID.

For all offers, the following data is available:

- An offer ID,
- The offered amount,
- The initial withdrawal amount,
- The number of payback terms agreed to,
- The monthly costs,
- The credit score of the customer,
- The employee who created the offer,
- Whether the offer was selected, and
- Whether the offer was accepted by the customer.

After importing the event logs into process mining solution, the process model discovered as below:

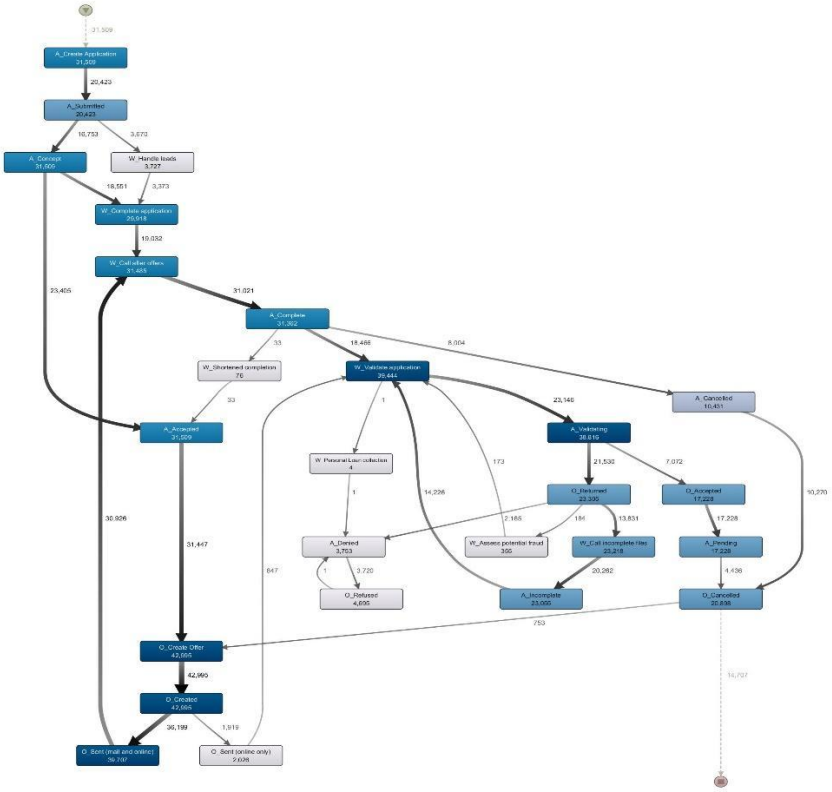

Fig. 1. General Process Model.

According to the model in figure 1, the application has the following events [2]:

- **Submitted:** A customer has submitted a new application from the website. A new application can also be started by the financial institute, in that case, this state is skipped.
- **Concept:** The application is in the concept state, that means that the customer just submitted it (or the financial institute started it), and a first assessment has been done automatically. An employee calls the customer to complete the application.
- **Accepted:** After the call with the customer, the application is completed and assessed again. If there is a possibility to make an offer, the status is accepted. The employee now creates 1 or more offers.
- **Complete:** The offers have been sent to the customer and the financial institute waits for the customer to return a signed offer along with the rest of the documents (payslip, ID etc)
- **Validating:** The offer and documents are received and are checked. During this phase, the status is validating.
- Incomplete: If documents are not correct or some documents are still missing, the status is set to incomplete, which means the customer needs to send in documents.
- **Pending:** If all documents are received and the assessment is positive, the loan is final and the customer is paid.
- **Denied:** If somewhere in the process the loan cannot be offered to the customer because the application doesn't fit the acceptance criteria, the application is declined, which results in the status 'denied'.
- **Canceled:** If the customer never sends in his documents or calls to tell he doesn't need the loan, the application is canceled.

Some of the statuses have a corresponding work item for call agents to process. These are the so-called 'standard work-items': “Handle leads”, “Complete application”, “Validate application”, “Call incomplete files”, and “Assess potential fraud”. Parallel to these standard work-items, there are also 'custom work-items'. These can exist together with the standard work-items. Shortened completion and personal loan collection are examples of that. Shortened completion means the customer has a certain profile that defines as a lower credit risk. These applications are investigated less thorough than higher risk applications [2].

Endpoints in the event logs are playing the important role in of understanding the log and analyze what had happened in the process. The event log provided from financial institute includes end points as below [3]:

1. “O_Canceled” that refers to the offer that was sent to the applicant who did not reply in time.
2. “O_Refused” that refers to two things, Offer refused by the financial institute or offer refused by the applicant.
3. “A_Pending” that refers to the offer that was accepted by the applicant and loan paid by the financial institute.

4. “A_Cancelled” that refers to the application that got canceled completely.
5. “A_Denied” that refers to the application that got denied by the financial institute as it was not in the acceptance criteria.

An offer may be refused individually and within the same application, another offer can be made. But if the total application is denied, it means that the status of all active offers is automatically set to “Refused” as well [4].

3 Analysis of the Main Questions

3.1 Question 1

Q1: What are the throughput times per part of the process, in particular the difference between the time spent in the company's systems waiting for processing by a user and the time spent waiting on input from the applicant?

Understanding the Question.

To analyze the throughput time, each of these questions should be answered per “part of the process”. In this line, we define the following sub-parts of the process, which will serve for the analysis of the throughput time:

1. Total throughput time for the overall process
2. The timeframe between the creation of the application (A_Create_Application) until the application is accepted (A_Accepted).
3. The timeframe between the acceptance of the application (A_Accepted) and the completion of the application (A_Complete).
4. The timeframe for the completion of the application (A_Complete) and any of the three outcomes for an application, which are:

a) Pending payments (A_Pending)/ Application pending
b) Application denied
c) Application cancelled

5. Timeframe between the acceptance of the application (A_Accepted) and the creation of an offer (O_Created)
6. The timeframe between the creation of an offer (O_create) and any of the three outcomes for an offer, which are:

a) Cancellation of the offer (O_Cancelled)
b) Acceptance of the offer (O_Accepted)
c) Refusal of the offer (O_Refused)

Additionally, in the above-given question, we have two perspectives on the throughput time of the process. Thus, we specifically investigate (7) the time spent in the company’s system waiting for processing by a user, and (8) the time spent waiting on input from the applicant.

Analysis Results 1-6

The table below depicts the throughput times of the process parts 1 to 6 as defined above. Results are given as mean and median values of the throughput time.

Table 1. Throughput Times per parts of the process.

Part of the Process	**Throughput Time**
1: Total throughput time for the overall process	21 days (mean) 19.1 days (median)
2: Timeframe between the creation of the application until the application is accepted	1 day (mean) 1 day (median)
3: Timeframe between the acceptance of the application and the completion of the application	0 day (mean)/189 mins 0 day (median) / 4 mins
4.a: The timeframe between the completion of the application and pending payments	16 days (mean) 14 days (median)
4.b: The timeframe between the completion of the application and application denied	15 days (mean) 13 days (median)
4.c: The timeframe between the completion of the application and application cancelled	28 days (mean) 31 days (median)
5: Timeframe between the acceptance of the application and the creation of an offer	2 days (mean) 0 days (median)
6.a: Timeframe between the creation of an offer and the cancellation of the offer	25 days (mean) 31 days (median)
6.b: Timeframe between the creation of an offer and the acceptance of the offer	16 days (mean) 14 days (median)
6.c: Timeframe between the creation of an offer and the refusal of the offer	15 days (mean) 13 days (median)

Particularly interesting is the finding, that the throughput times for the process parts 4.c and 6.a on average span a longer time frame than for the overall process (process part 1). As the process parts 4.c and 6.a both end with the cancellation of the application and the cancellation of the offer, respectively, we can suppose that the financial institute waits longer until they cancel an offer or an application hoping to complete it successfully, eventually.

Furthermore, the completion of the application is probably an automated task. As it seems, all or most activities between the acceptance of an application and the

completion of an application are automated explaining the very low throughput time of 189 minutes (mean)/ 4 minutes (median).

Analysis Results 7 & 8

From the log-file, we can interfere that all work items with the state "start" are related to (7): Strictly speaking, the time spent in the company's system waiting for processing by a user is the difference between the scheduling of the work item and the start of that work item. For reasons of completeness, we also consider cases, which flow directly to the work state "start", as work items might be picked up directly for processing (i.e. scheduling of the task is omitted).

As most of the of the work items are directly processed, i.e. the time they wait in the system is 0. Our analysis with that regards focuses on waiting time greater than 0. For the work items W_Call incomplete files_start, W_Handle leads_start, W_Personal Loan collection _start, W_Shortened completion _start, and W_Validate application _start the waiting time in the system is 0, the waiting time for the work items W_Call after offers_start and W_Complete application _start are depicted in the tables below. Because throughput times are conditional on the path the cases take, also the number of cases in each path are listed.

Table 2. W_Call after offers - start.

Preceeding activity / state	Throughput time	Number of cases affected
W_Call after offers – schedule	0 days (mean) 0 days (median)	31,362
W_Call after offers – resume	0 days (mean) 0 days (median)	51
O_Sent (mail and online) – complete	12 days (mean) 7 days (median)	5
O_Sent (online only) – complete	1 day (mean) 1 day (median)	2
W_Call after offers – start	0 days (mean) 0 days (median)	2

Table 3. W_Complete application – start.

Preceeding activity / state	Throughput time	Number of cases affected
W_Complete application – schedule	0 days (mean) 0 days (median)	14,516
A_Concept – complete	1 day (mean) 1 day (median)	15,065
W_Complete application – resume	0 days (mean) 0 days (median)	135
W_Complete application – start	0 days (mean) 0 days (median)	59
O_Created – complete	1 day (mean) 1 day (median)	26

W_Shortened completion – suspend	1 day (mean) 1 day (median)	2
O_Cancelled – complete	4 days (mean) 3 days (median)	3
A_Accepted – complete	1 day (mean) 0 days (median)	49

Interestingly, more most work items associated with W_Complete application – start are rooted from A_Concept – complete instead of W_Complete application – schedule.

However, (based on business logic) we can only determine one state, which is clearly dependent on input from the applicant and can be labeled as (8): the return of the offer by the customer. Table 4 gives an overview of the time spent waiting on feedback from the applicant regarding the offers proposed by the financial institute. Additionally, the number of cases for each path are listed to determine each path's strength.

Especially, the fact that offers are faster returned if they are received online only is worthwhile to note.

Table 4. O_Returned – complete.

Preceeding activity / state	Throughput time	Number of cases affected
A_Validating – complete	0 days (mean) 0 days (median)	20,406
A_Incomplete – complete	0 days (mean) 0 days (median)	746
W_Call incomplete files – suspend	2 days (mean) 1 day (median)	707
W_Validate application resume	0 days (mean) 0 days (median)	282
O_Sent (online only) – complete	1 day (mean) 1 day (median)	122
O_Sent (mail and online) – complete	3 days (mean) 3 days (median)	100
W_Validate application – suspend	1 (mean) 0 (median)	624
O_Returned – complete	0 (mean) 0 (median)	16
W_Validate application – start	0 (mean) 0 (median)	6
O_Cancelled – complete	2 (mean) 1 (median)	3
W_Call after offers – suspend	0 (mean) 0 (median)	1
W_Call incomplete files – resume	0 (mean) 0 (median)	2

General Observations

Analyzing general characteristics of the throughput time, some interesting observations

become apparent. Especially the data concerning the last month of the log file shows some irregularities, respectively changes in comparison to the data of the proceeding months. Fig. 1 and Fig. 2 contrast the average throughput time distribution for the overall log (Fig. 2) and the last 30 days of the log (Fig. 3).

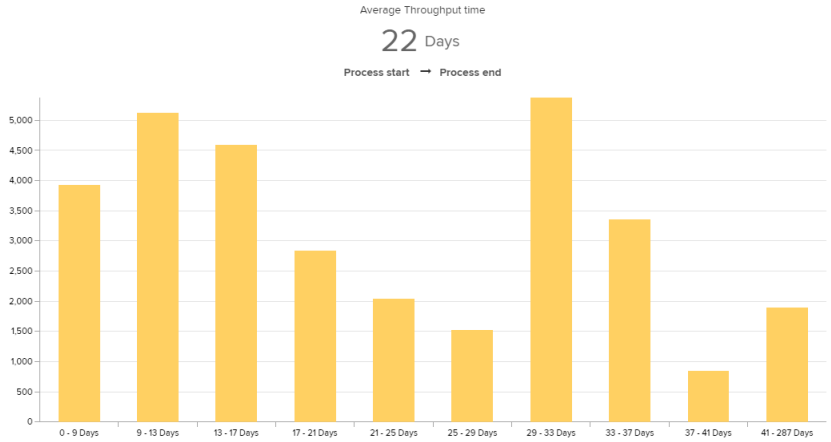

Fig. 2. Throughput time distribution for total duration of the log file available.

Comparing both diagrams, there is a significant change in how the throughput time is distributed. For the complete log, the throughput time distribution has two main peaks at 9-13 days and 29-33 days. Yet, for the last month of the financial institute process, it is largely concentrated in the 31-36 days pillar, as depicted below in Fig. 2. In general, the distribution of the throughput time for the last 30 days is way more unequal than for the complete log. This distribution shift also affects the average throughput time, which grows from 22 days to 28 days to complete the process.

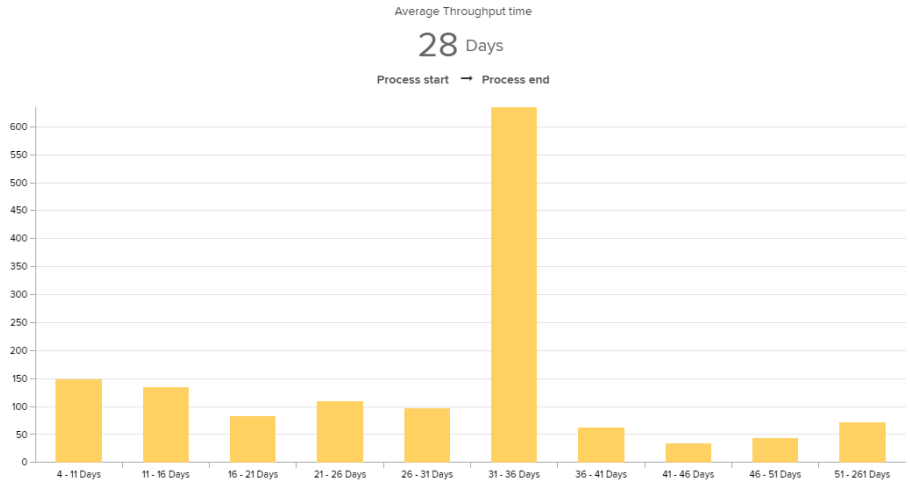

Fig. 3. Throughput time distribution for the last 30 days of the log file available.

In line with this, also the bottlenecks of the financial institute process increase. For the overall log file, there is only one connection, which increases process throughput time considerably and is marked as critical. The connection between W_Call after offers –suspend and A_Cancelled – complete affects 24% of the cases with and average throughput time of 23 days.

Selecting the last 30 days of the log, only, there are three connections, which are classified as considerable bottlenecks by Celonis. The connection between W_Call after offers –suspend and A_Cancelled – complete is still an issue and impacts 534% of the cases with an average throughput time of 23 days. Additionally, the connections from W_Call after offers –schedule and O_Sent (mail and online) – complete to A_Cancelled – complete are listed as bottlenecks for the last 30 days of the log. While the bottleneck between W_Call after offers – schedule and A_Cancelled – complete influences 65% of the cases with an average throughput time of 25 days, the connection between O_Sent (mail and online) – complete and A_Cancelled – complete affects 38% of all cases with an average throughput time of 28 days.

All of the named critical bottlenecks go to A_Cancelled – complete, which might indicate that there are too few resources performing this activity causing the bottleneck or, as stated before, that the financial institute waits longer until they cancel an application hoping to complete it successfully, eventually.

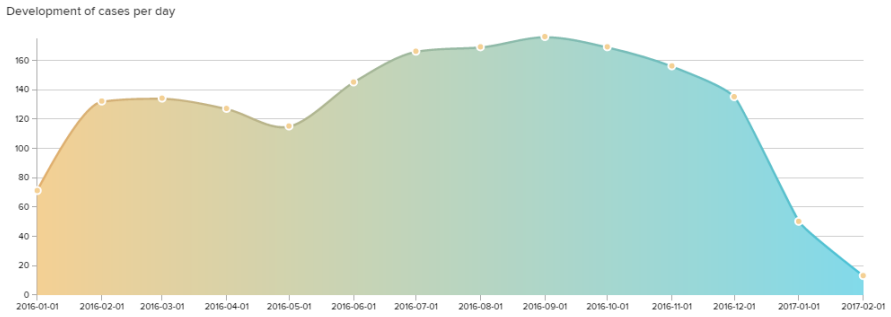

Fig. 4. Case distribution for the log file available.

Further, cases decrease, but throughput time increases significantly in the last 30 days of the financial institute log file provided. One line of argumentation is, that the bottlenecks cause this increase in throughput time. Additionally, we find a cyclic development in the amount of cases per time of the year, which might be connected to yearly closing. However, these are only possible explanations and should be subject to further investigation.

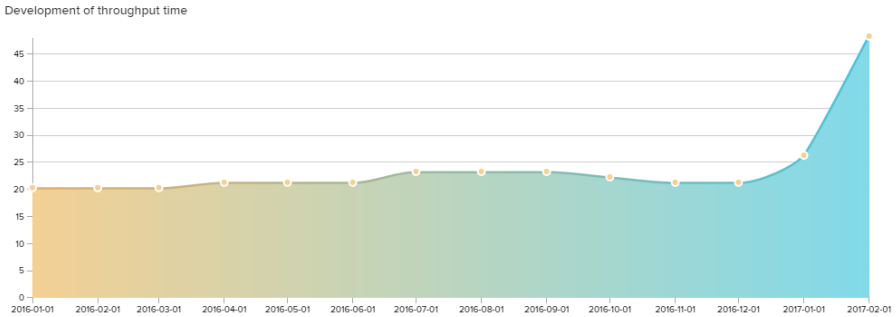

Fig. 5.. Throughput time distribution for total duration of the log file available.

3.2 Question 2

Q2: What is the influence on the frequency of incompleteness to the final outcome? The hypothesis here is that if applicants are confronted with more requests for completion, they are more likely to not accept the final offer.

Understanding the Question

This question aims at finding how the process behaves when an application has a frequency of incompleteness. In addition, we are interested in finding if we should accept or reject the hypothesis that applicants confronted with more requests for completion are more likely to reject the final offer.

By filtering the data to visualize such behavior, we can analyze the process variants and verify if customers confronted with more requests for completion are more likely to not accept an offer. The same behavior can be observed if one takes "W_Call incomplete files" into consideration, which means that every time an application is incomplete, an agent would call the customer to complete the application. Additionally, we are particularly interested in the presence of "A_Pending" in the execution of cases and its relation with the frequency of incompleteness.

Analysis and Results

For the analysis of this question, one may consider three different scenarios:

Scenario 1: Cases with at least one occurrence of "A_Incomplete"
Scenario 2: Cases with no occurrence of "A_Incomplete"
Scenario 3: Cases with occurrence of more than one "A_Incomplete"

Considering these scenarios, we perform the following steps:

1. Find all cases from Scenario 1 – Note that same behavior can be observed when considering "W_Call incomplete files" as well. As both of them lead to the same set of cases, we considered only "A_Incomplete" for the purpose of this analysis.
2. Find how many cases from Scenario 1 have the end point of "A_Pending" – which means an offer was accepted by the customer.
3. Find all cases from Scenario 2, as well as all cases with the occurrence of "A_Pending" from this scenario.
4. Find cases from Scenario 3, as well as the cases with a successful outcome – occurrence of "A_Pending".
5. Compare the results.

In the context of step 1, we created a filter of type "Attribute" with filtering mode "Mandatory" and selected attribute "A_Incomplete" (See Figure *). That means that only cases with the occurrence of "A_Incomplete" will be retrieved by the filter. By applying such filter, one could visualize that, out of the total number of applications from the event log (31509), 15003 cases had the occurrence of at least one "A_Incomplete" status, which represents about 47.6% of the cases.

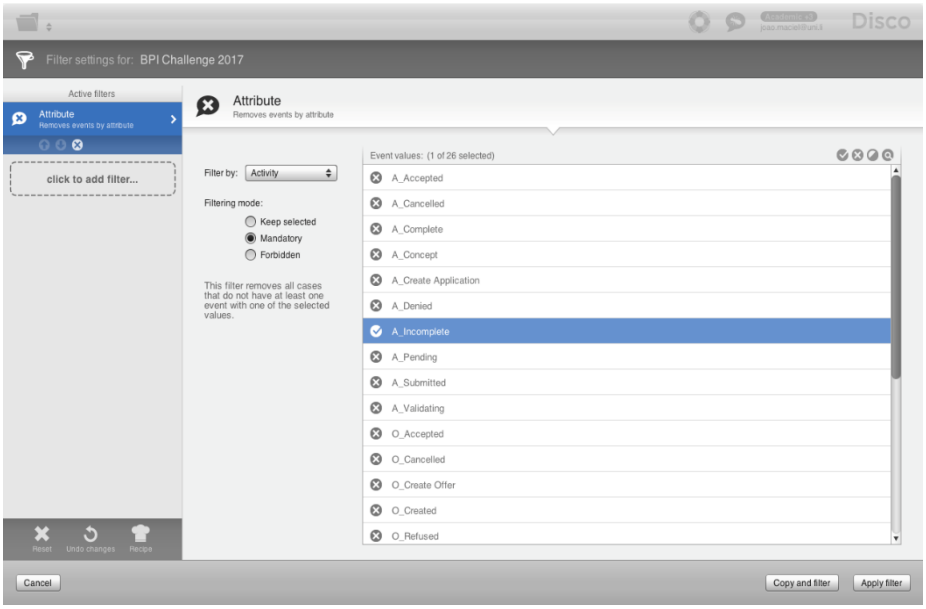

Fig. 6. Filter Settings of Scenario 1.

Furthermore, we continue to step 2, in order to figure out how many cases had at least one offer accepted. In order to do so, we create another filter of type "Attribute", but at this time we make only "A_Pending" mandatory. By filtering those cases, we see that 12647 cases (84.3%) that registered at least one incompleteness had an offer accepted by the customer, triggering the status of "A_Pending".

In order to compare the outcome of cases from the previous steps with the outcome of cases in which an incompleteness never occurred – Scenario 2, we may still consider the filters from step 1 and 2, but switching the filtering mode the of "A_Incomplete" to "Forbidden". That means that we want to visualize now all cases that ended with "A_Pending" without having "A_Incomplete" during their execution.

Out of 16506 of cases retrieved, only 27.75% (4581) of the cases had a successful outcome. That might already indicate that we may reject the hypothesis that applicants confronted with more requests for completion are more likely to not accept the final offer. However, this analysis considered all cases with at least one occurrence of "A_Incomplete". Therefore, we may further analyze the behavior of the process if we consider only the cases where "A_Incomplete" happens more than once – Scenario 3. For this purpose, we created a filter of type "Follower" where reference event must be eventually followed by the follower event. In this case, "A_Incomplete" was selected as both reference event value and follower event value (Figure *).

This filter retrieved 5686 cases. That means that out of the total number of cases (31509), about 18% of applications had more occurrence of "A_Incomplete". To visualize how many of them had a successful outcome, we created a second filter by selecting "A_Pending" as a mandatory attribute. This time, Disco retrieved 4891 cases, which represents approximately 87.6% of cases. If we compare this result with the positive outcome of cases from scenario 1, we even see an increase on the number of

applicants that accepted an offer. Table 1 summarizes the findings for the three scenarios.

Therefore, based on the analysis of the data, we may conclude that the frequency of incompleteness did not negatively influence the final outcome of the process.

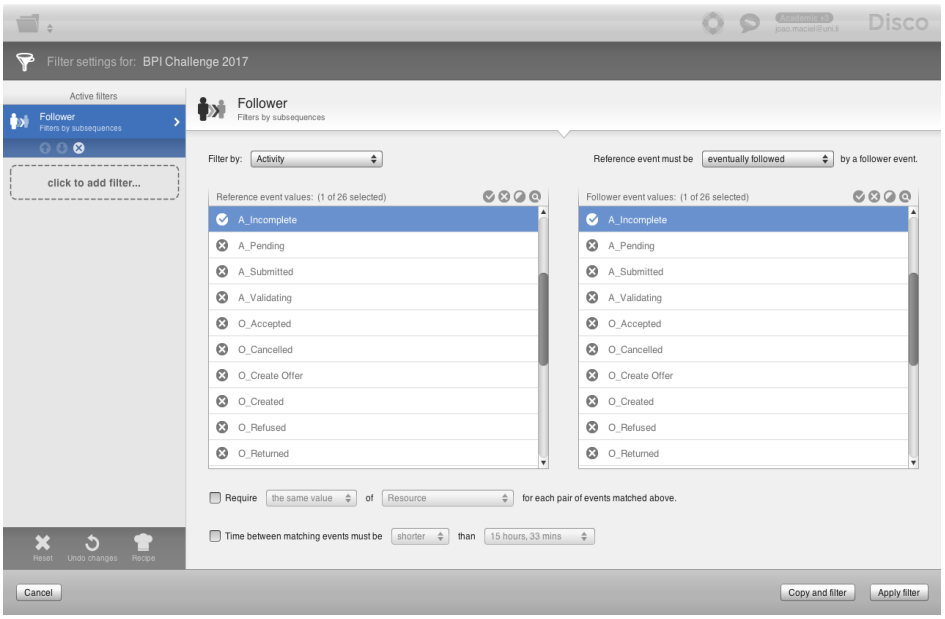

Fig. 7. Filter Settings of Scenario 3.

Table 5. Influence of incompleteness on occurrence of "A_Pending".

	Total Cases		*A_Pending*	
Scenario 1	15003	47.61%	12647	84.30%
Scenario 2	16506	52.39%	4581	27.75%
Scenario 3	5686	18.04%	4981	87.60%

3.3 Question 3

Q3: How many customers ask for more than one offer (where it matters if these offers are asked for in a single conversation or in multiple conversations)? How does the conversion compare between applicants for whom a single offer is made and applicants for whom multiple offers are made?

Understanding the Question

This question can be divided into two sub-questions, which should be analyzed separately. First sub-question is "How many customers ask for more than one offer (where it matters if these offers are asked for in a single conversation or in multiple conversations)?" and second sub-question is "How does the conversion compare between applicants for whom a single offer is made and applicants for whom multiple offers are made?".

To answer first sub-question, we must set some filters on event logs to see how many cases are existing in logs that contain more than 1 offer. It's notable that more than one offer can be made by the financial institute itself or only by the customer or a combination of these two. Afterward, these cases must be analyzed to see how many had happened from the single conversation and how many had happened from multiple conversations. In the end we want to compare which type were more successful (cases that has the end point of "A-pending", where the loan is actually paid out to the customer) for the financial institute and which type was not successful (cases with the end points of "O_Canceled", "O_Refused", "A_cancelled", and "A_denied").

For this purpose, we should calculate the proportion of two different groups then compare the findings for each group and present the results. Group-1 includes more than one offer in single conversation, and Group-2 includes more than one offer in multiple conversations. We must check how many cases in each group were accepted for the loan, therefore we would have proportion per each group.

The second sub-question aims to find cases with multiple offers and cases with the single offers then check which one is more successful and would end to accepting the offer. For this purpose, we should compare the proportion of cases with single offer that ends to "A-pending" and multiple cases that end to "A-pending". In this question, we just need to know the number of cases with multiple offers and it doesn't matter if these offers are from customers, financial institute or both of them. To make a decision over results, we should consider the mean and median case duration of each group next to their success proportion.

Analysis and Results

After analyzing the process and controlling the cases, we understood that the event "W_complete application" stands for the first conversation between the customer and financial institute. If the activity "W_call after offers" occurs after the first offer creation that is labeled as "O_create offer" then we face cases that have more than one offer. Also when two or more offers created eventually after each other with no "W_call

after offers" in between then we face the cases that have offers that initiated by the financial institute itself.

Analysis and results of first sub-question

To analyze event logs and answer this sub-question easily we must follow several steps as below:

1- Find all cases that have multiple offers
2- By applying filters over step 1:
2-1 Find multiple offers that had happened in single conversation
2-2 Find multiple offers that had happened in multiple conversations
3- Find how many cases has end point of "A-pending" in step "2-1" and how many has rejection end points, then calculate the proportion
4- Find how many cases has end point of "A-pending" in step "2-2" and how many has rejection end points, then calculate the proportion
5- Compare the results of step 3 and step 4

8559 cases out of 31509 cases are the ones that have more than 1 offer. This result has reported by applying the filter type of "Follower", when we choose "O_create offer" as "*reference event value"* eventually followed by "O_create offer" as "*follower event value" as it has shown below:*

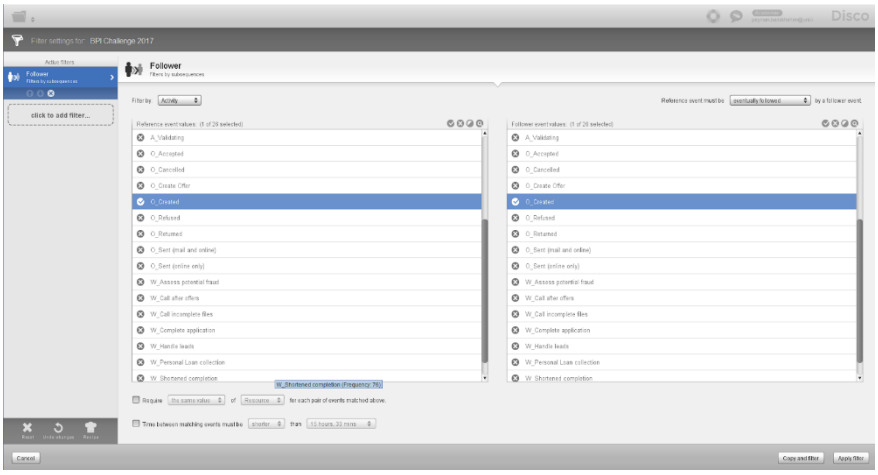

Fig. 8. Filter Settings of Step 1.

We are looking for a way to observe cases that have multiple offers. We must consider that if there is existing event labeled as "W_complete application", it is referring to the first conversation between the financial institute and the customer and if the event "W_call after offers" occurs eventually then we face multiple conversations. Therefore, we continue with step 2 and with the same filtering method. By choosing "W_call after offers" as reference event value that eventually followed

"O_create offer", we find all cases that have more than one offer in multiple conversations. The filter settings for this step is shown as below:

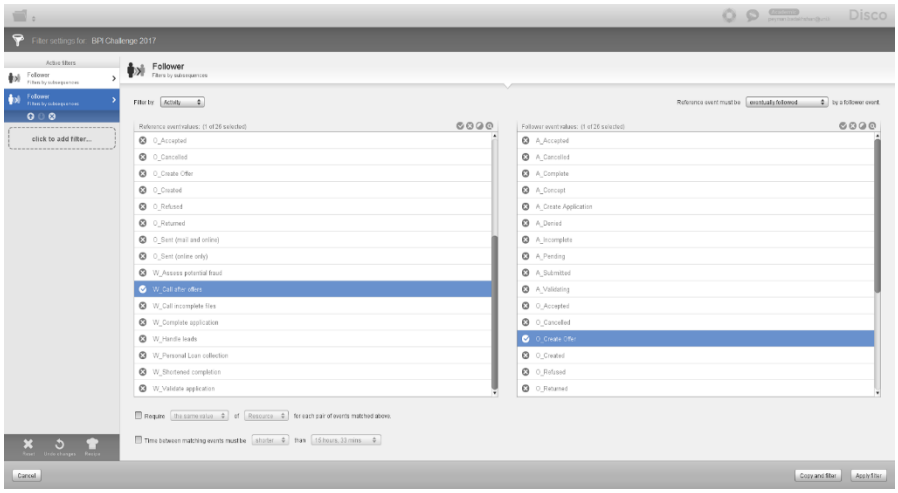

Fig. 9. Filter Settings of Step 2-1.

And by choosing "W_call after offers" as reference event value that **never** eventually followed "O_create offer", we find all cases that have more than one offer in single conversation.

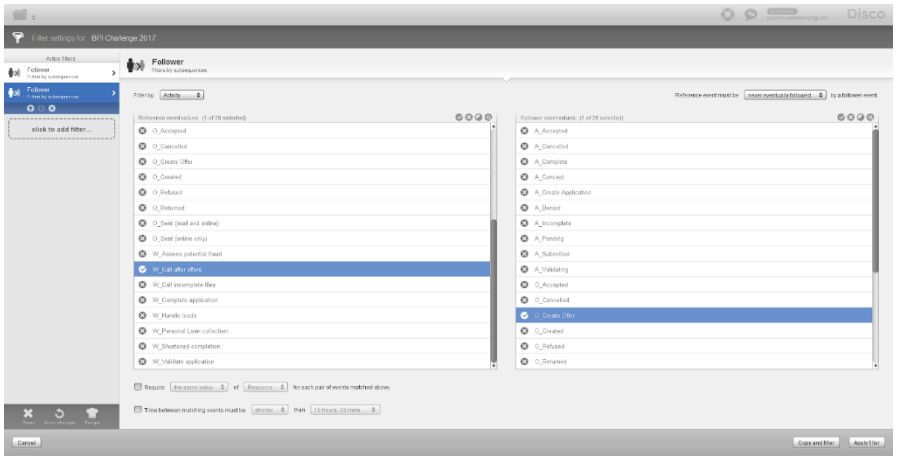

Fig. 10. Filter Settings of Step 2-2.

Considering multiple conversations, 5872 cases has been reported out of 8559 cases of step 1 that refers to multiple offers in multiple conversations. Considering single conversations, 2687 cases out of 8559 cases refers to multiple offers happening in a single conversation that is approximately 31.4%.

By setting a new filter on step 2-1, we would like to find how many cases ended up to “A-pending” and answer the step 3. To do so, in “Follower” type filter, we choose all events as reference event values and “A-pending” as follower event value. This filter ends up to 1231 cases out of 5872 cases of step 2-1, which means approximately 21% of cases were successful.

By setting a new filter on step 2-2, we would like to find how many cases ended up to “A-pending” and answer the step 4. To do so, in “Follower” type filter, we choose “O_Create Offer” event as reference event value and “A-pending” as follower event value. This filter ends up to 3819 cases out of 5872 cases of step 2-2, which means approximately 65% of cases were successful.

Comparing the step 3 and 4 shows that cases with more than one offer in a single conversation have the higher rate of acceptance (65%) than cases with more than one offer in multiple conversations (21%).

Analysis and results of second sub-question

To analyze event logs and answer this sub-question easily we must follow several steps as below:

1- Find all cases that have multiple offers
2- Find all cases that end to “A-pending” in step 1 and calculate the proportion
3- Find all cases that have single offer
4- Find all cases that end to “A-pending” in step 3 and calculate the proportion
5- Compare the results of step 2 and step 4

From the first sub-question, we know that 8559 cases out of 31509 are the ones that have more than one offer. Now we apply the filter over step 1 to find out how many of these cases were successful and ended up with the event “A_Pending”. 5050 cases out of 8559 cases are successful and are the ones that financial institute paid the loan to its customers and means 59% of cases were successful.

To follow with step 3 we need to clear all filters and apply a new filter to get cases that have only single offers. By using filter type of “Follower” and choosing “O_Create Offer” as reference event value which is **never** eventually followed by “O_Create Offer” we reach 22950 cases out of 31509 cases existing in event logs. To find the result of step 4, we need to add a new filter over step 3 by choosing “O_Create Offer” as reference event value that is eventually followed by “A_Pending”. This action ends up to 12178 successful cases out of 22950, which means 53% of cases were successful.

The success rate in single offers is slightly less than multiple offers. It can be assumed that conversion between these two scenarios (Single offers and multiple offers) are quite similar but we should consider that single offers has 22950 cases in total but multiple offers has 8559 cases in total, which means single offers are 72% of total cases and multiple offers are only 27% of total cases. The successful single offers had happened with mean case duration of 16.1 days and median case duration of 13.7 days while multiple offers had happened with the mean duration of 23 days and median case duration of 19.6 days. This can prove that cases with single offers are bringing more benefit to the financial institute.

4 Conclusion

In this paper, we have analyzed a log file from a financial institute. We have investigated the loan process with regards to throughput time, the frequency of incompleteness of loan applications, and how multiple offers relate to successful process completion. Our results indicate interesting starting points for hypothesis development and future inquiries.

We investigated various parts of the given business process log file with respect to throughput time. Interestingly, we found a rapidly increasing throughput time in the last month of the process log, while at the same time number of cases were decreasing significantly and new bottlenecks arose. We would like to highlight this fact and motivate the financial institute to further look into this development.

In the context of the frequency of incompleteness of loan applications, our analysis demonstrated that applications that reached the status of incomplete, either once or more times, were more likely to have an offer accepted by the customer. Therefore, we rejected the hypothesis that the frequency of incompleteness would have a negative impact on the outcome of the process, and we concluded that cases with more frequency of incompleteness had actually a positive impact on the overall acceptance rate of offers.

After analyzing the third question of the financial institute, it is realized that there are two sub-questions available. Considering the first sub-question, it is concluded that cases with more than one offer in a single conversation have the higher rate of acceptance (65%) than cases with more than one offer in multiple conversations (21%). Also, considering the second sub-question, it is concluded that the success rate in single offers is slightly less than multiple offers. It can be assumed that conversion between these two scenarios is quite similar. Therefore, regarding both sub-questions, it is recommended to the financial institute to invest more on multiple offers and try to find the best offer for the customer in one conversation.

Acknowledgements

All authors contributed equally to this paper. The list of authors is structured according to alphabetic order.

References

1. BPI, https://www.win.tue.nl/bpi/doku.php?id=2017:challenge
2. Activity explanation, http://www.win.tue.nl/promforum/discussion/764/activity-explanation
3. End Points - a proposition, http://www.win.tue.nl/promforum/discussion/752/end-points-a-proposition
4. A_denied and O_refused, http://www.win.tue.nl/promforum/discussion/751/a-denied-and-o-refused