



UNIVERSITÀ
DEGLI STUDI
DI MILANO

Business Information System

Project Report

BPI Challenge 2019

Miad Alavinezhad

Prof. Paolo Ceravolo

2023 - 2024

Contents

Business Information System	1
Project Report	1
BPI Challenge 2019.....	1
Introduction	3
Data	3
Variants.....	7
Different Flows in the Event Log (Item Category).....	10
Process discovery for Item categories	12
Conformance checking.....	17
Github Repository:	20

Introduction

For the BPI Challenge 2019, we have the data from a large multinational company operating from The Netherlands in coatings and paints. The report gives a detailed analysis of the unfiltered provided data (log) and then tries to gain more information about the process model.

The goal is to analyze the data and get insights into the processes involved in the organization to find patterns that are being executed in them, using data mining methods and tools.

Data

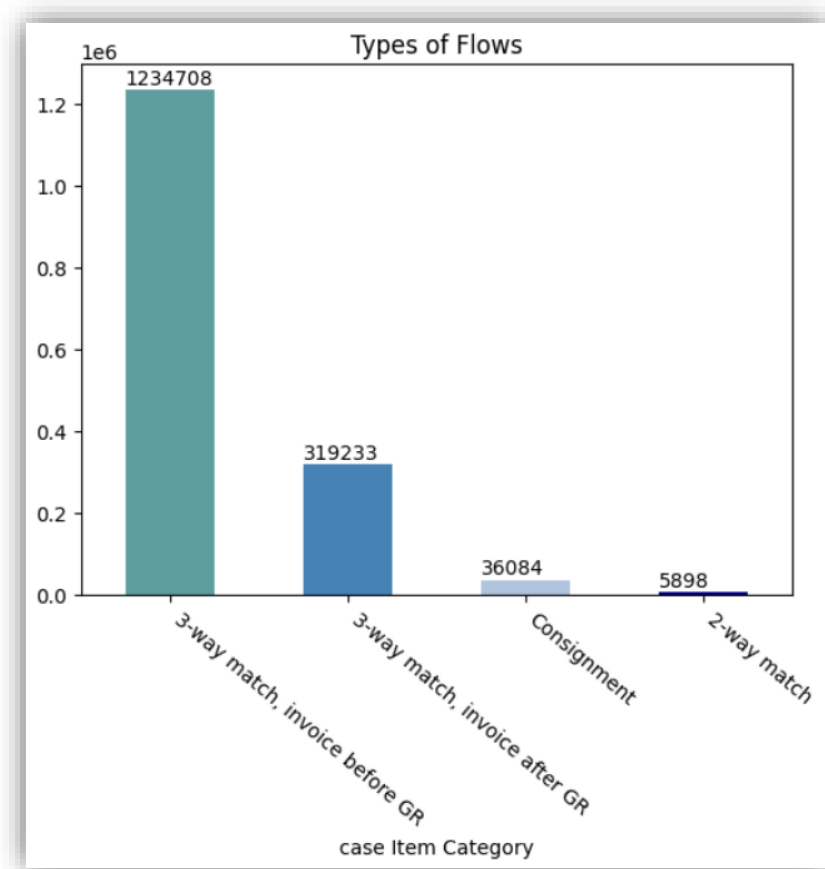
- The data is a log of the events. The following information is obtained using pm4py python module: The log has 1,595,923 events of 42 different classes of events and 251,734 cases. The first event starts from 26.1.1948 and the last one occurs on 09.04.2020. Few events are out of this scope of time.
- In the event log, each purchase order (or purchase document) contains one or more-line items. For each line item, there are roughly four types of flows in the data:

3-way matching, invoice after goods receipt: For these items, the value of the goods receipt message should be matched against the value of an invoice receipt message and the value put during creation of the item.

3-way matching, invoice before goods receipt: For such purchase items, invoices can be entered before the goods are receipt, but they are blocked until goods are received.

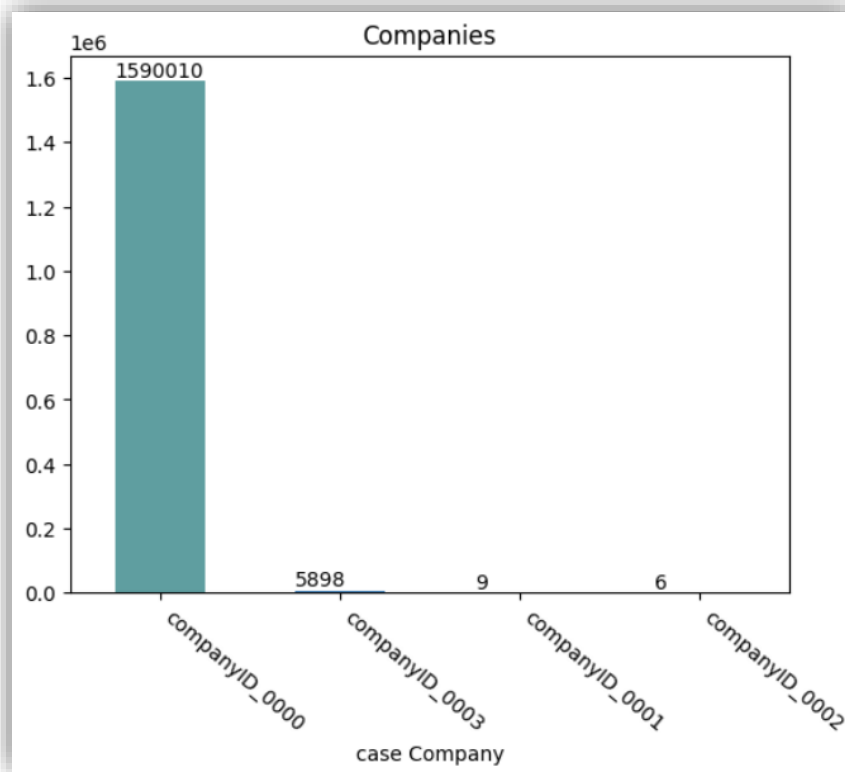
2-way matching (no goods receipt needed): For these items, the value of the invoice should match the value at creation (in full or partial until PO value is consumed), but there is no separate goods receipt message required.

Consignment: For these items, there are no invoices on PO (Purchase Order) level as this is handled fully in a separate process.



The most common flow type is for consignment items, with nearly 1.2 million flows. 3-way match (invoice before GR) has the second highest number of flows, at around 320,000. 3-way match (invoice after GR) has about 361,000 flows, and 2-way match has around 59,000 flows. Consignment have the least number of flows, at around 5,900. Both the 3-way match after and before have most of the item category types and therefore more events are in the process of completing these 2. The graph has been made using matplotlib module.

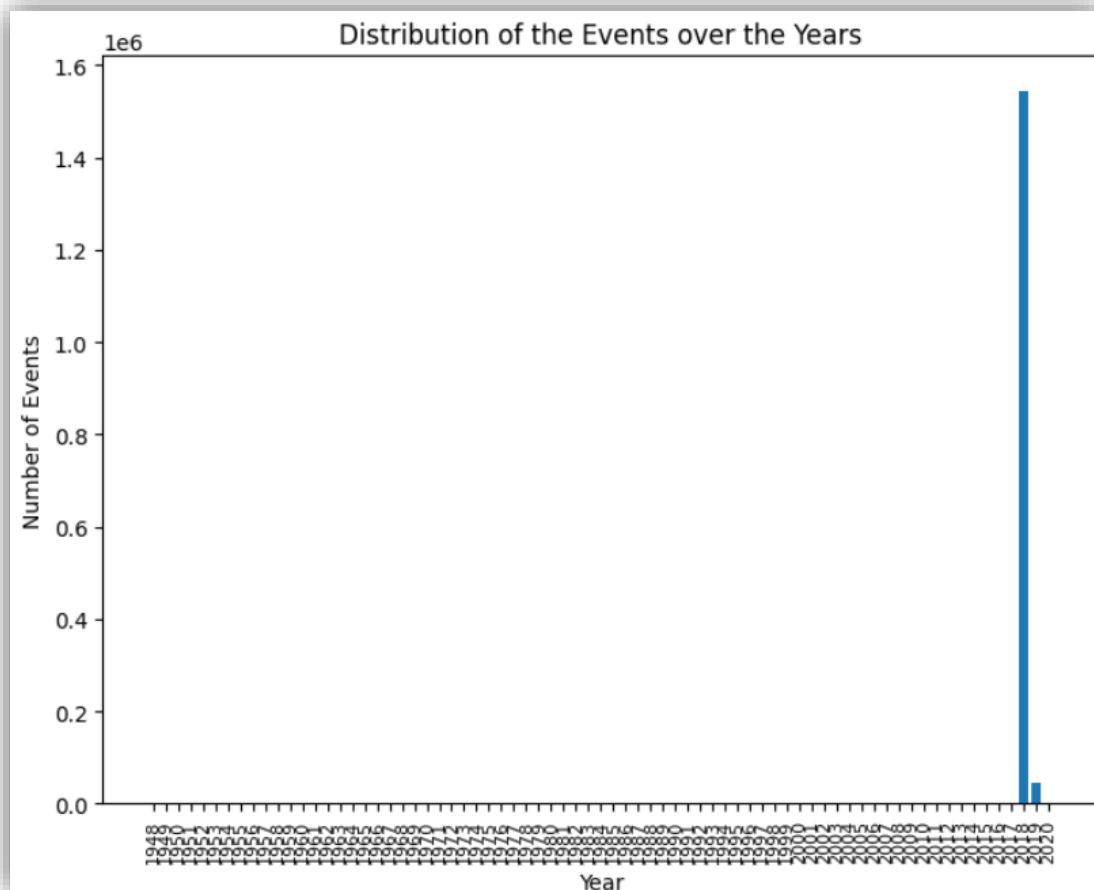
- Next, we have the companies involved in the event log which are companies with IDs from 0000 to 0003 (four companies). The distribution is as follows:



We can see that almost all events are in contribution with companieID_0000 with about 1.6 million events and next comanieID_0003 with 5900 events and next 2 companies with less than 10 which shows the importance of the contribution of companieID_0000.

- There are different types of purchase orders (PO) in the log: Standard PO, Framework Order and EC Purchase Order. More than 1.5 million events are for Standard PO and Framework Order and EC PO have about 33000 and 22000 events respectively.
- For the duration of the events, the minimum and maximum length of extracted form the log are 0 and 37137009 minutes (more than 70 years). This indicates that some filtering needs to be done. Some events end at the time of start, and some seems like has never been finished. The average time for events is about 171 days which probably change after some filtering.

- The First **Filter** on the log is to filter out events with 0 durations. After the filter, 3585 events have been filtered out.
- Distribution of events over the years is as follows:

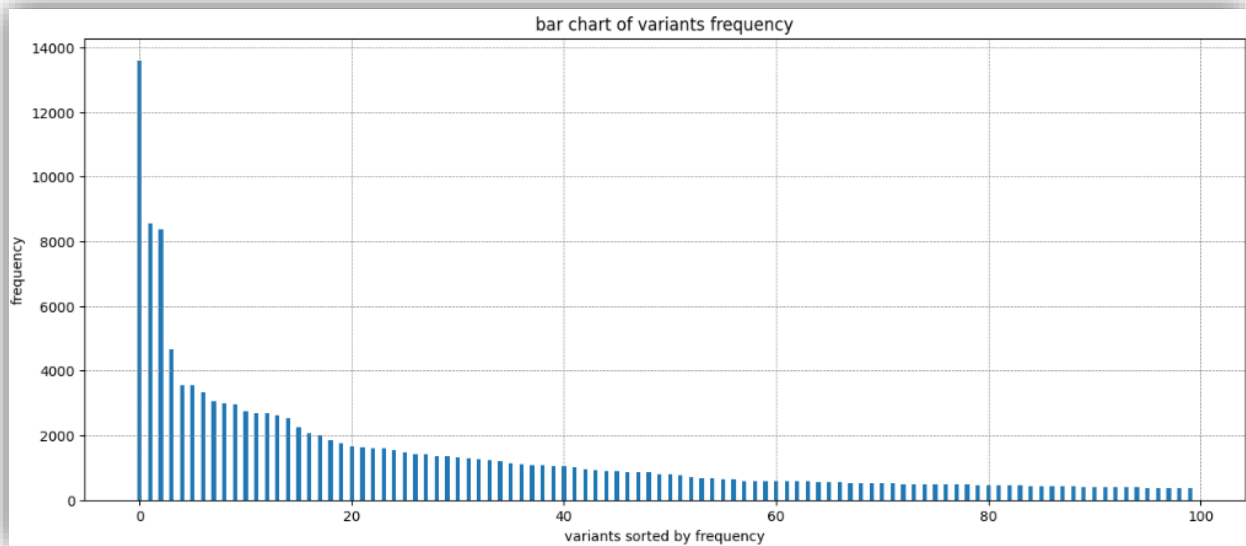


Almost all events are in 2018 and some are in 2019 which can be because some events started in 2018 and ended in 2019. As we saw at least one event (which can be multiple) occurred in 1948. That is the reason that the plot started from 1948 but number of events occurred from 1948 to 2018 are so low that we cannot see them on the chart and can be filtered out. The year of these events may be mistakenly inputted in the event log. Because of this reason we filter events from before 2018 and after 2019. So we consider events in the period of 2 years of time.

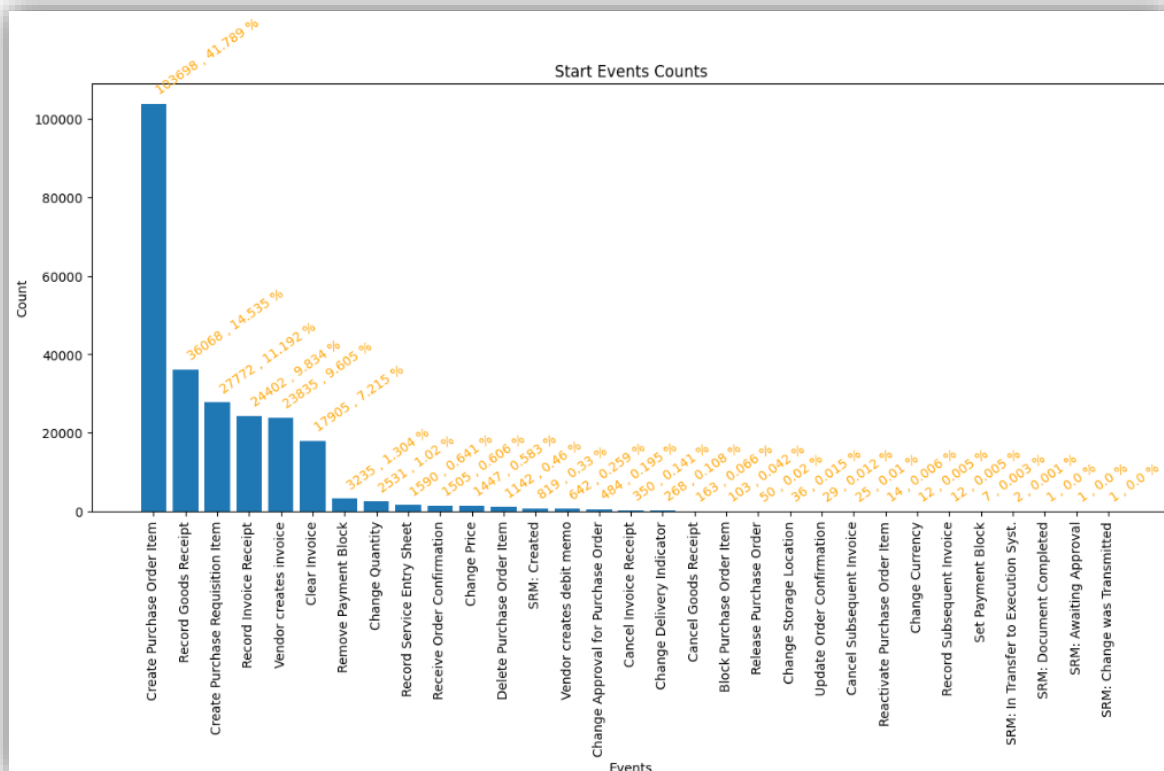
Variants

- The log has 24787 variants. The number has been calculated using `case_statistics.get_variant_statistics()` function. Below we will see the 10 most common variants in the log with their number of occurrences:
 - 1 - Create Purchase Order Item -> Vendor creates invoice -> Record Goods Receipt -> Record Invoice Receipt -> Clear Invoice -> **13607**
 - 2 - Create Purchase Order Item -> Record Goods Receipt -> **8558**
 - 3 - Create Purchase Order Item -> Record Goods Receipt -> Vendor creates invoice -> Record Invoice Receipt -> Clear Invoice -> **8384**
 - 4 - Create Purchase Order Item -> Delete Purchase Order Item -> **4651**
 - 5 - Record Goods Receipt -> Create Purchase Order Item -> **3544**
 - 6 - Create Purchase Order Item -> Vendor creates invoice -> Record Goods Receipt -> Clear Invoice -> Record Invoice Receipt -> **3539**
 - 7 - Create Purchase Order Item -> Receive Order Confirmation -> Record Goods Receipt -> Vendor creates invoice -> Record Invoice Receipt -> Clear Invoice -> **3327**
 - 8 - Clear Invoice -> Create Purchase Order Item -> Vendor creates invoice -> Record Goods Receipt -> Record Invoice Receipt -> **3050**
 - 9 - Create Purchase Requisition Item -> Create Purchase Order Item -> Record Goods Receipt -> **2998**
 - 10 - Create Purchase Order Item -> Vendor creates invoice -> Record Invoice Receipt -> Record Goods Receipt -> Remove Payment Block -> Clear Invoice -> **2958**

In the chart below we can see the distribution of the variants in the event log. The first 3 variants have relatively high distance from the rest of the variants and are more common. You can see the events in these variants in the list above. The number given to variants on x-axis is based on their frequency.

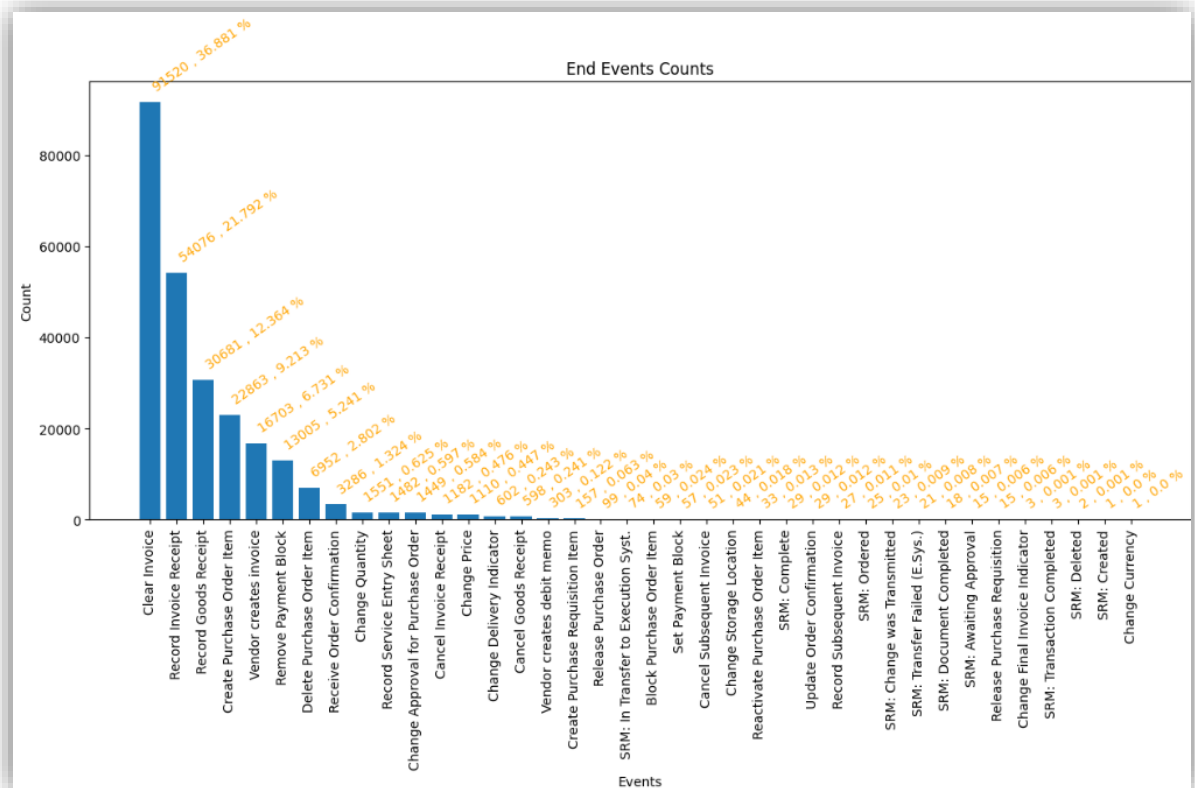


- Next, we analyze the start and end activities (events) and see which of them are usually starts and end variants.



Most frequent start activity is 'Create Purchase Order Item' covering about 41% of all start activities. Since the event logs the records of

purchase orders and its procedures, it is expected that most of the cases start with creating a new purchase order from buyer. Next frequent start activities are 'Record Goods Receipt', 'Create Purchase Requisition Item', 'Record Invoice Receipt', 'Vendor creates invoice' and 'Clear invoice' which cover from about 14% down to 7% of all starts.



On end activities we have 'Clear invoice' on about 36% (most frequent). As the name suggests, clear invoice means that the related invoice has been completed (probably reviewed and the payment was made) which indicated the end for a process of purchasing an item. At the next frequent end activities, we have 'Record invoice receipt' and 'Record Goods Receipt'.

It seems odd that roughly 9% of end activities are 'Create Purchase Item' as it is most common event to start a case. This end activity can imply that the cases has only one event and both start and end activity is 'Create Purchase Item' which mean the PO has been placed but it did not continue.

Different Flows in the Event Log (Item Category)

- It is important for the report to investigate more into the item categories since they are the 4 different flows that the process of purchase can be done in the company.
- First, for each category we extract most common start and end activities:
 - 3-way before:

Most frequent start event of 3-way before: ('**Create Purchase Order Item**', 90247)

Most frequent end event of 3-way before: ('**Clear Invoice**', 86471)
 - 3-way after:

Most frequent start event of 3-way after: ('**Create Purchase Order Item**', 5396)

Most frequent end event of 3-way after: ('**Clear Invoice**', 4929)
 - 2-way:

Most frequent start event of 2-way: ('**Create Purchase Order Item**', 399)

Most frequent end event of 2-way: ('**Change Approval for Purchase Order**', 541)
 - Consignment:

Most frequent start event of 2-way: ('**Create Purchase Order Item**', 399)

Most frequent end event of 2-way: ('**Change Approval for Purchase Order**', 541)
- Now we extract common variants for each category since in the most common variants for all the event log, some variants that may be common in some categories maybe absent because of low occurrence of them in the whole event log.
 - 3-way before:

1 - Create Purchase Order Item -> Vendor creates invoice -> Record Goods Receipt -> Record Invoice Receipt -> Clear Invoice -> **12969**

2 - Create Purchase Order Item -> Record Goods Receipt -> Vendor creates invoice -> Record Invoice Receipt -> Clear Invoice -> **7953**

3 - Create Purchase Order Item -> Delete Purchase Order Item -> **4243**

4 - Create Purchase Order Item -> Vendor creates invoice -> Record Goods Receipt -> Clear Invoice -> Record Invoice Receipt -> **3423**

5 - Create Purchase Order Item -> Receive Order Confirmation -> Record Goods Receipt -> Vendor creates invoice -> Record Invoice Receipt -> Clear Invoice -> **3327**

○ 3-way after:

1 - Create Purchase Order Item -> Vendor creates invoice -> Record Goods Receipt -> Record Invoice Receipt -> Clear Invoice -> **638**

2 - Create Purchase Order Item -> Record Goods Receipt -> Vendor creates invoice -> Record Invoice Receipt -> Clear Invoice -> **431**

3 - Create Purchase Order Item -> Record Goods Receipt -> **381**

4 - Create Purchase Order Item -> Record Goods Receipt -> Record Invoice Receipt -> **250**

5 - Create Purchase Order Item -> Delete Purchase Order Item -> **171**

○ 2-way:

1 - Create Purchase Order Item -> Change Approval for Purchase Order -> **226**

2 - Vendor creates invoice -> Create Purchase Order Item -> Change Approval for Purchase Order -> Record Invoice Receipt -> **126**

3 - Create Purchase Order Item -> Change Approval for Purchase Order -> Record Invoice Receipt -> Vendor creates invoice -> **73**

4 - Change Approval for Purchase Order -> Create Purchase Order Item -> Change Approval for Purchase Order -> Change Approval for Purchase Order -> **52**

5 - Vendor creates invoice -> Create Purchase Order Item -> Change Approval for Purchase Order -> Record Invoice Receipt -> Clear Invoice -> **50**

○ Consignment:

- 1 - Create Purchase Order Item -> Record Goods Receipt -> **5941**
- 2 - Record Goods Receipt -> Create Purchase Order Item -> **2835**
- 3 - Create Purchase Requisition Item -> Create Purchase Order Item -> Record Goods Receipt -> **1259**
- 4 - Record Goods Receipt -> Create Purchase Requisition Item -> Create Purchase Order Item -> **407**
- 5 - Create Purchase Order Item -> Record Goods Receipt -> Record Goods Receipt -> **339**

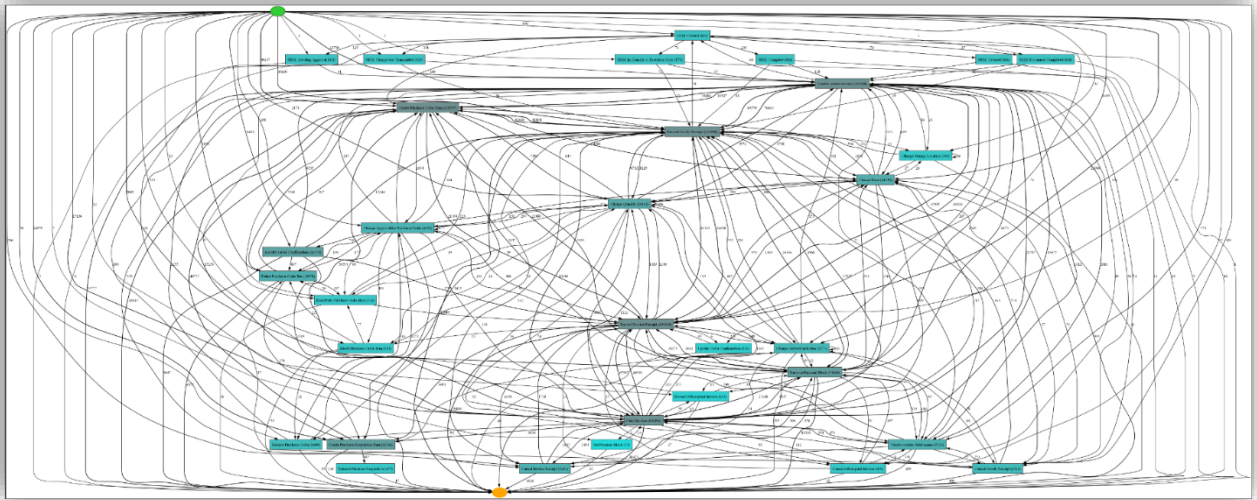
Process discovery for Item categories

Here we show for each flow their most frequent variants (which tell us what to expect in each process) and also show them using **alpha miner**, **heuristic miner** and **petri net**.

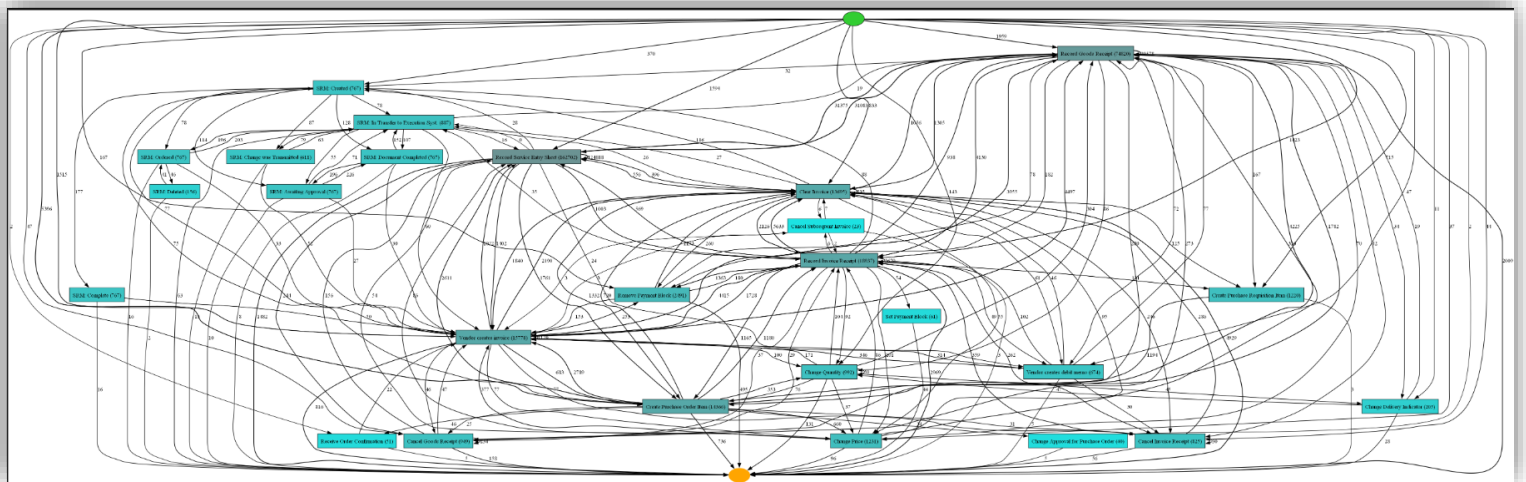
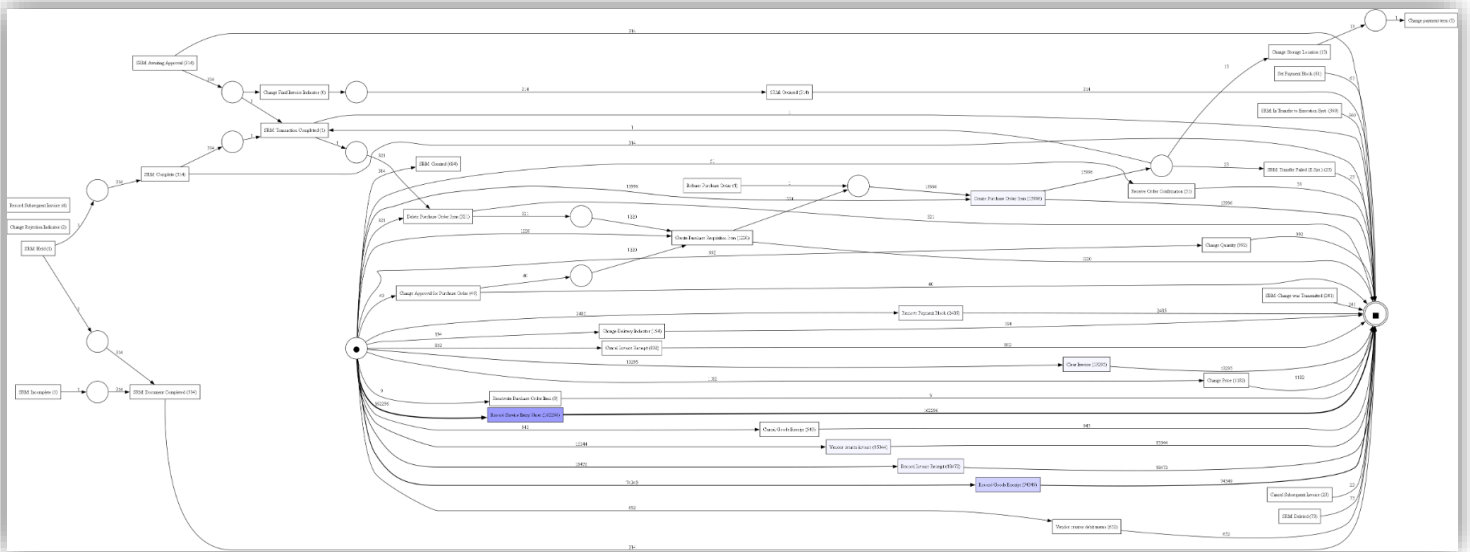
Alpha Miner provides us with a **Petri net model** where all the transitions are visible and unique and correspond to classified events, an initial marking that describes the status of the Petri net model in the start of the execution and a final marking that describes the status of the Petri net model at the end of the execution.

Heuristics Miner acts on the Directly-Follows graphs, providing a way to find common constructs and relationships. It relies on analyzing the event log to discover causal relationships between activities. It examines how frequently activities appear together in specific sequences within different process instances. Its output is an object containing the activities and the relationships between them.

- 3-way before Heuristic miner and Alpha miner (with petri net)

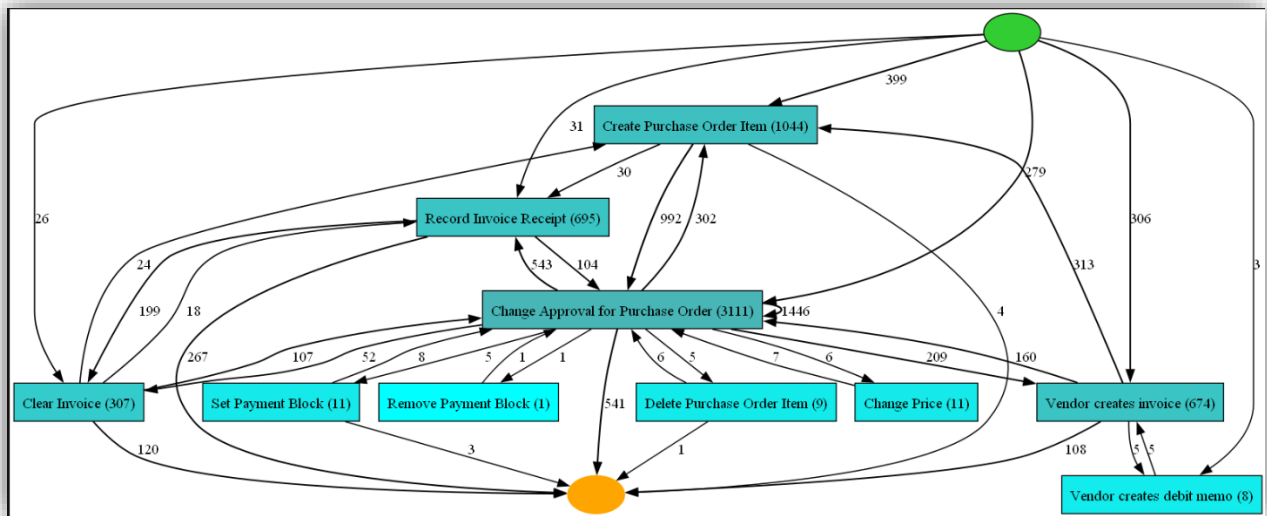
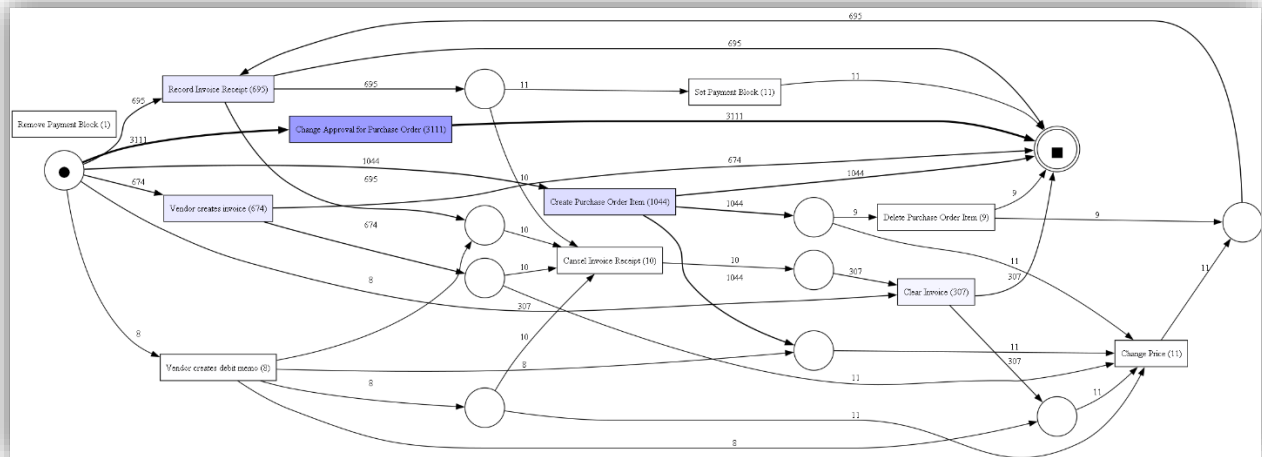


Graph for 3-way before with heuristic miner showing the core flows of this category.



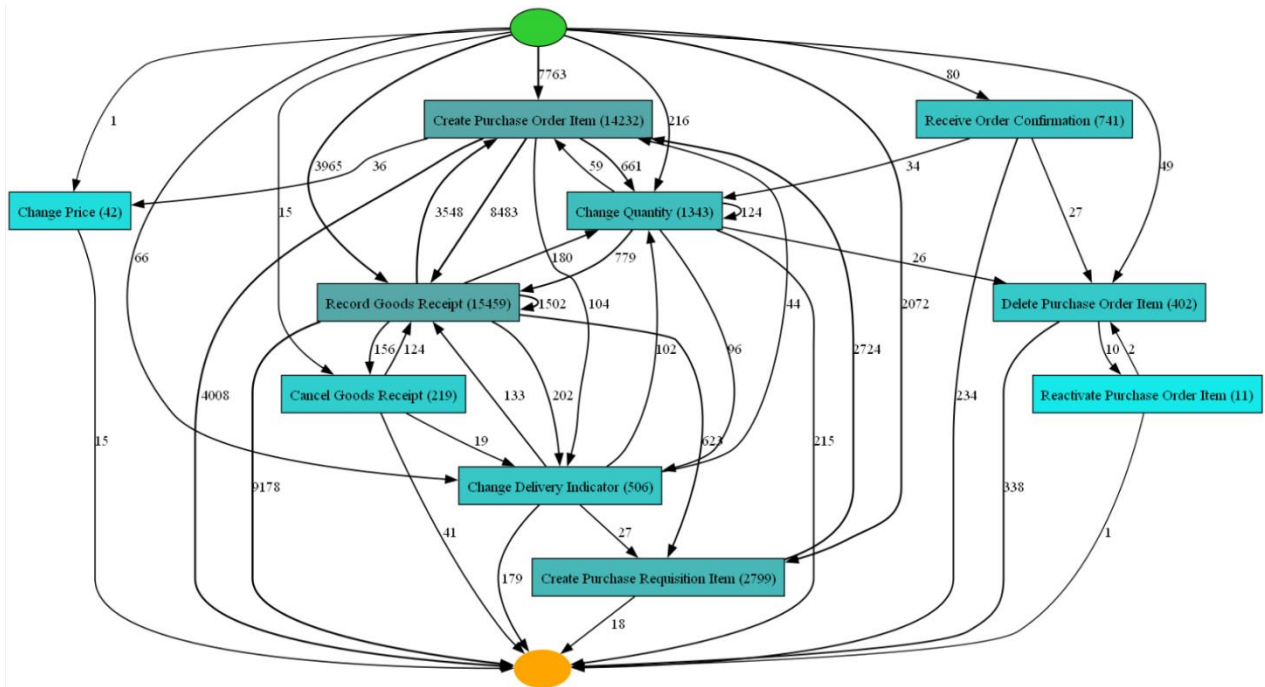
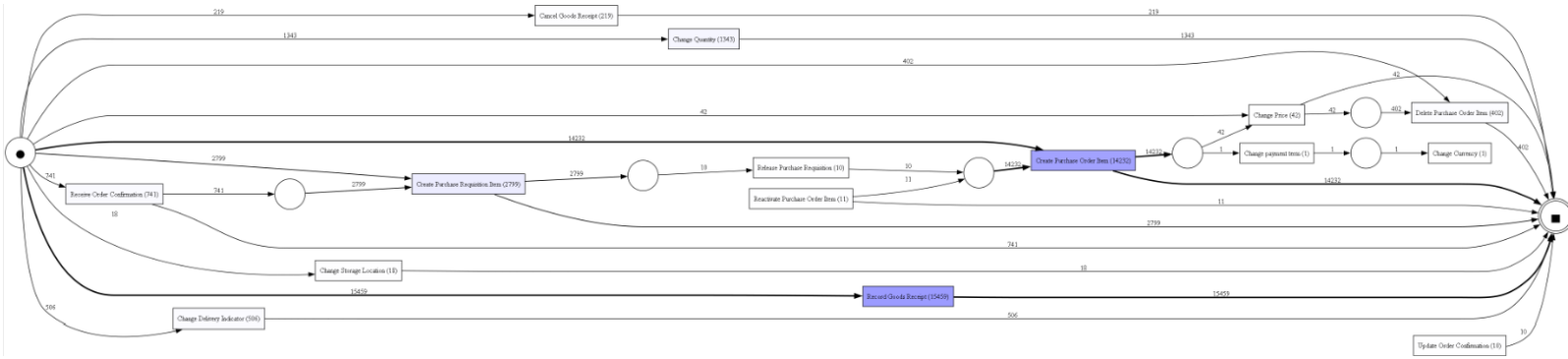
- 2-way:

The Alpha miner and heuristic miner on 2-way:



- Consignment:

The Alpha miner and heuristic miner on consignment:



Conformance checking

This section is devoted to conformance checking which is a technique to compare a process model with an event log of the same process. The goal is to check if the event log conforms to the model and, vice versa. It tells us how well the actual execution of a business process aligns with the expected or desired behavior.

The average fitness for each category is as follows:

- 3-way before Average Trace Fitness: 0.27001288489727454

	is_fit	consumed_tokens	remaining_tokens
0	False	12	14
1	False	12	14
2	False	12	14
3	False	12	14
4	False	12	14
...
218502	False	8	7
218503	False	8	7
218504	False	5	5
218505	False	5	5
218506	False	5	5

- 3-way after Average Trace Fitness: 0.22946234023422193

	is_fit	consumed_tokens	remaining_tokens
0	False	14	15
1	False	17	18
2	False	8	9
3	False	14	16
4	False	13	14
...
14361	False	155	153
14362	False	7	3
14363	False	7	3
14364	False	10	6
14365	False	62	60

- 2-way Average Trace Fitness: 0.29475998959442706

	is_fit	consumed_tokens		remaining_tokens
0	False	9	10	
1	False	10	12	
2	False	15	11	
3	False	15	11	
4	False	3	3	
...	
1039	False	3	3	
1040	False	3	3	
1041	False	3	3	
1042	False	3	3	
1043	False	3	3	

- Consignment Average Trace Fitness: 0.44150240859900836

	is_fit	consumed_tokens		remaining_tokens
0	False	5	3	
1	False	4	2	
2	False	4	2	
3	False	4	2	
4	False	4	2	
...	
14227	False	4	2	
14228	False	4	2	
14229	False	4	2	
14230	False	4	2	
14231	False	4	2	

Github Repository:

Codes used in this report to extract the data can be reached in the repository below:

https://github.com/miadalavinezhad/BPI_2019_Project/tree/main