

**Project Title:** Business Process Analysis using BPI Challenge 2019 Dataset

**Student Name:** Dauletbay Gulzat

**Project Description:**

The project aims to analyze and gain insights from the BPI Challenge 2019 dataset. The dataset represents a real-world business process in a specific application area. It includes information about various activities, timestamps, actors involved, and other relevant process data. The project will explore the dataset and extract valuable information to support decision-making and process improvement.

The data comes from a multinational company in the coatings and paints industry, based in the Netherlands. The company investigated the purchase order handling process across its 60 subsidiaries. The dataset includes information about purchase orders and their line items.

There are four types of flows related to the line items in the data:

1. 3-way matching, invoice after goods receipt: The value of the goods receipt message is matched against the value of an invoice receipt message. This flow requires both the GR-based flag and the Goods Receipt flags to be true.
2. 3-way matching, invoice before goods receipt: Goods receipt messages are required for these purchase items, but GR-based invoicing is not. Invoices can be entered before goods are received but are blocked until the goods arrive. Users or batch processes can unblock the invoices. Clearance is only allowed if the goods are received, and their value matches the invoice and the initial item value.
3. 2-way matching (no goods receipt needed): Separate goods receipt messages are not necessary for these items. The invoice value should match the initial value, either in full or partially until the purchase order value is consumed. Both the GR-based flag and the Goods Receipt flags are false for these items.
4. Invoice Receipt (IV) --> Match ValuesConsignment: This flow applies to items handled separately, without invoices on the purchase order level. The GR indicator is true, but the GR IV flag is false. Consignment items are not expected to have invoices based on their item type.

Multiple goods receipt messages and invoices can be associated with each purchase item. For example, paying rent may involve one purchase document item but multiple goods receipt messages and cleared invoices, each representing a fraction of the total amount. Similarly, logistical services may generate numerous goods receipt messages for one line item. Compliance requires matching the amounts of the line item, goods receipt messages, and invoices.

- 1. Organizational Goals:**
  - 1.1. Sumamrize data
  - 1.2. Identify goals
- 2. Knowledge Uplift Trail**
  - 2.1. Data preprocessing: Cleaning the dataset, handling missing values, and ensuring data quality.
  - 2.2. Exploratory Data Analysis (EDA): Analyzing the dataset's characteristics, identifying patterns, and visualizing process flows.
  - 2.3. Process Mining: Applying process mining techniques to discover the underlying process model, such as process maps and petri nets.
  - 2.4. Conformance Analysis: Evaluating process performance metrics, such as cycle time, throughput, and bottlenecks.
- 3. Project Results**
- 4. Conclusion**

## 1.1 Project

### Summary of data:

- Total number of purchase documents: 76,349
- Total number of purchase items: 251,734
- Total number of cases (purchase document + purchase item combinations): 251,734
- Total number of events: 1,595,923
- Total number of activities: 42
- Total number of users: 627 (607 human users and 20 batch users)

### Tools used in the project:

Colab for running Python code using libraries pandas and pm4py, also additional software like PMKT and Disco to observe the process map in various ways which makes it easy with different diagrams and filters.

Various **attributes** are recorded for each purchased item, but the **most importance** ones are highlighted in the table below:

**Table: Event Log**

Attribute	Description
Case ID	Combination of the purchase document and purchase item IDs
Purchase Document	Anonymized ID of the purchase document
Purchase Item	Anonymized ID of the purchase item
Event ID	Unique identifier for each event within a case
Activity	Name of the activity performed in the event
Timestamp	Timestamp indicating when the event occurred
User	Identifier of the user who performed the activity (human user or batch user)

## **1.2 Goals**

### **1.2.1. Goals for the project:**

- Analyze the BPI Challenge 2019 dataset from a multinational company in the coatings and paints industry.
- Gain insights into the purchase order handling process across the company's subsidiaries.
- Extract valuable information to support decision-making and process improvement.

### **1.2.2. Research Questions to Address BPI Challenge:**

- Are there any loops or repetitions in the process?
- Are there any bottlenecks or inefficiencies in the process?
- Can we predict the completion time of a process instance based on its attributes?
- How does the process flow differ for each of the four types of flows related to the line items
- Are there any patterns or variations in the process that can be observed from the data?

## 2.1 Data preprocessing

Data preprocessing: Cleaning the dataset, handling missing values, and ensuring data quality.

The initial phase of data preprocessing involves dropping duplicates from the dataset. Before checking for duplicates, there were 11,973 unique variants and 251,734 cases. After removing duplicates, the number of unique variants decreased to 11,319 while maintaining the same number of cases.

The second step includes filtering out incomplete cases that do not reach the "create purchase order item" stage. Following this filtering process, the number of unique variants further reduced to 11,311, representing a decrease of 5.5%. The number of cases decreased to 247,287, which accounts for a reduction of 1.7%.

Next, the time duration considered for analysis was limited to the last three years. After applying this time constraint, the number of unique variants decreased to 11,267 , which corresponds to a reduction of 5.7%. The number of cases decreased to 247209, representing a decrease of 1.8%.

## 2.2 Exploratory Data Analysis

The basic steps help to clean data and reduce it to N%. However, working with 10 thousand variants challengeable task, and also it is hard to identify bottlenecks and analyze throughputs. The solution is to divide the dataset into four types of flows related to the line items in the data.

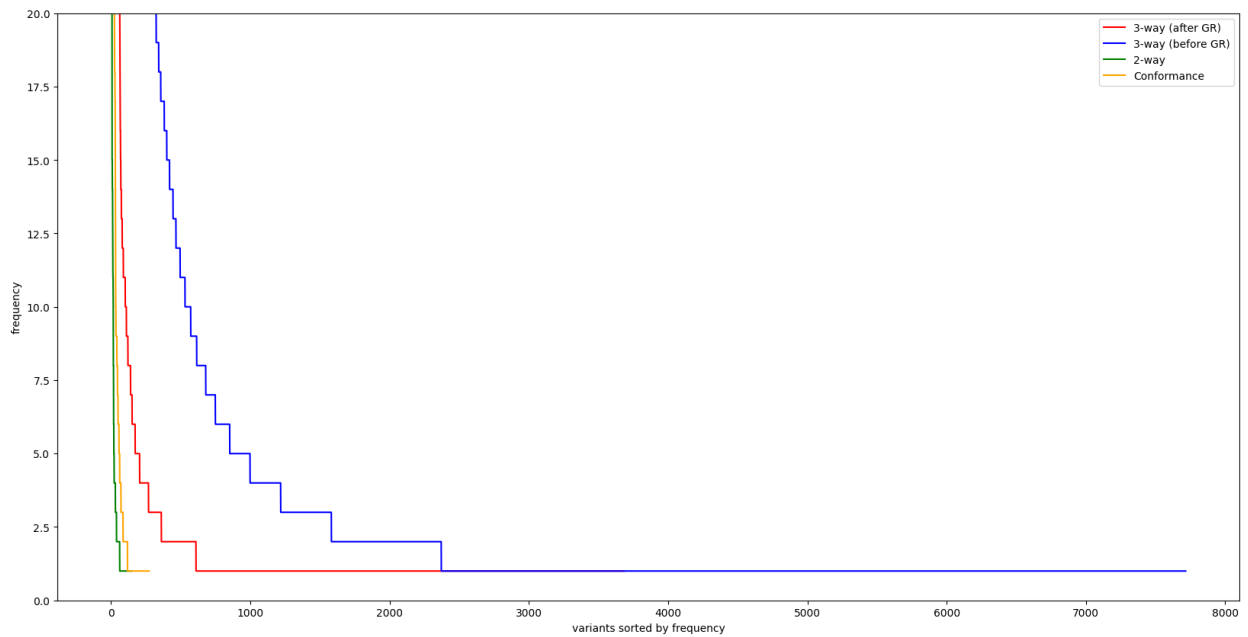
### Illustrations of key differences of types in tables

These table provide a more detailed representation of the categories, their attributes, and the matching process for each category, with separate columns indicating whether the goods receipt value, invoice value, and creation value are matched.

Category Name	GR-based Flag	Goods Receipt Flags	Match Goods Receipt Value	Match Invoice Value	Match Creation Value	3-way matching, invoice after
3-way matching	True	True	Yes	Yes	Yes	Yes
3-way matching, invoice before	False	True	Yes	User or Batch Process	Yes	Yes
2-way matching	False	False	N/A	Yes	Yes	N/A
Consignment	True	False	N/A	N/A	N/A	N/A

### Variant analysis for each type:

- 3-ways matching after GR variants: 3682
- 3-ways matching before GR variants: 7717
- 2-ways matching variants: 147
- Conformance variants: 274

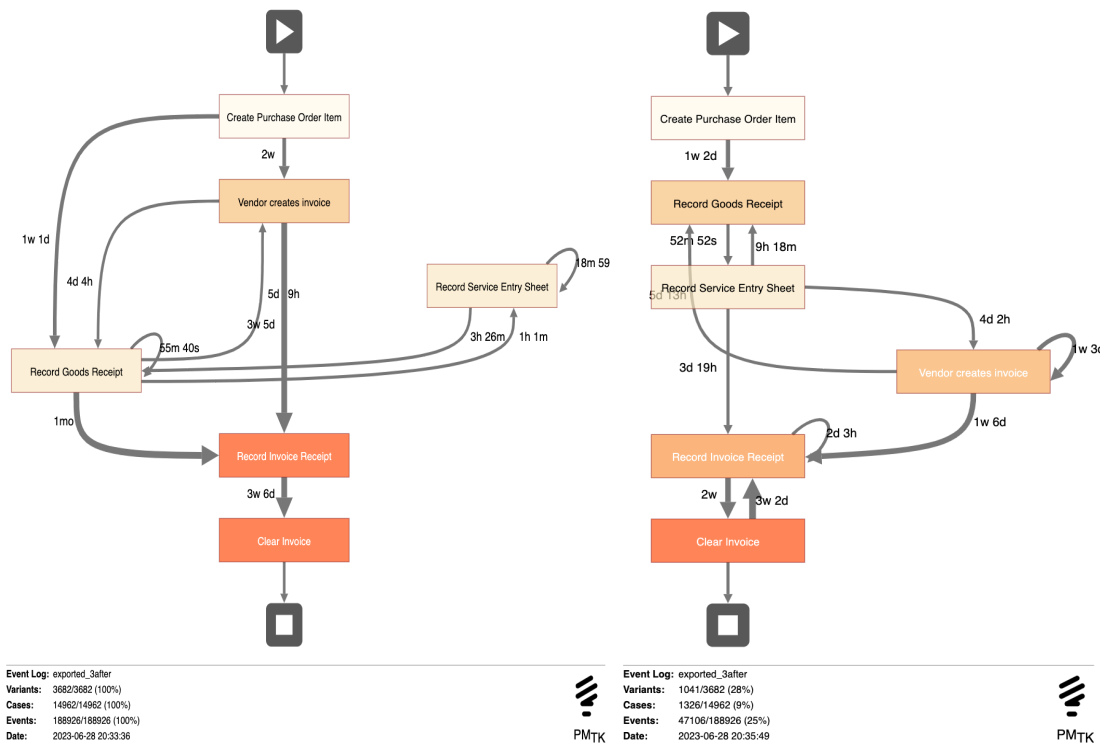


## Process Maps

Process maps provide a visual representation of the sequence and dependencies of activities in a flow, allowing you to identify patterns, bottlenecks, and variations.

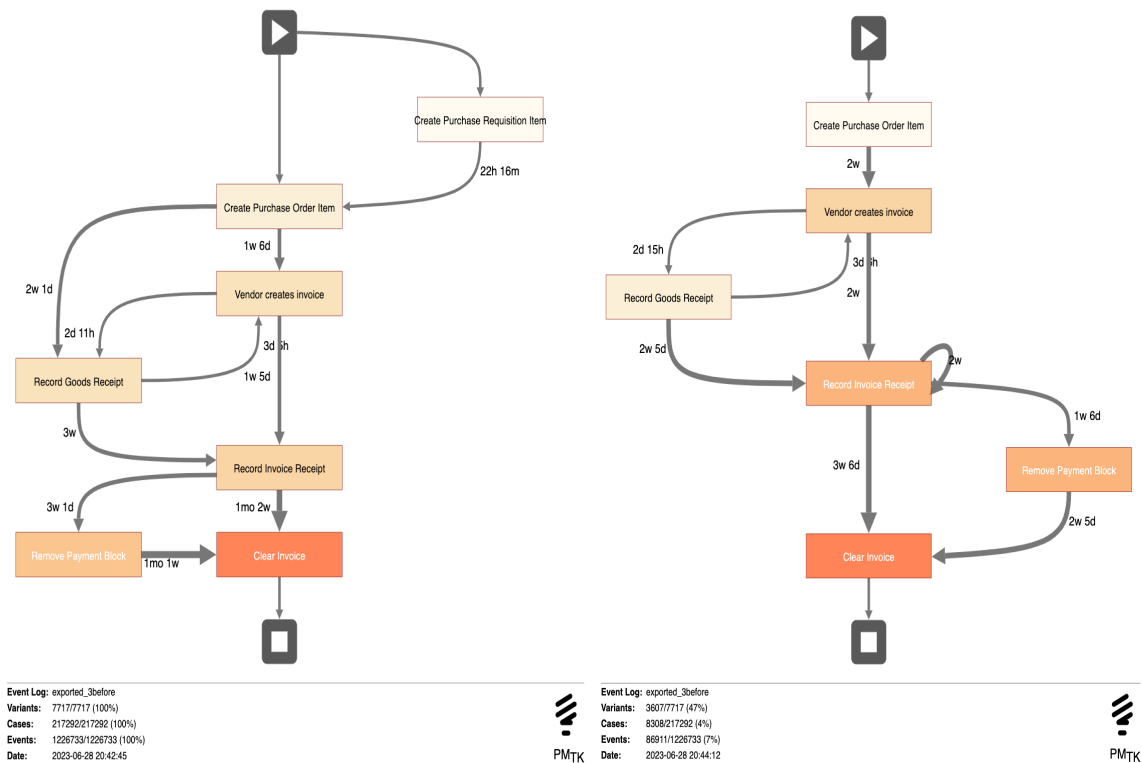
- 3-ways matching after GR variants:

Two pictures from pmkt show us the difference in all data and filtering by reworking have a significant difference between a number of cases and variants and events.



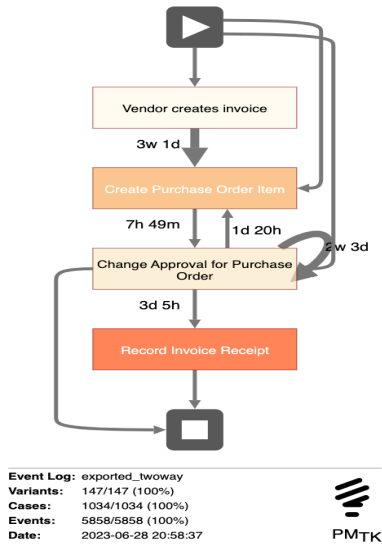
- 3-ways matching before GR variants: 7717

Overall the situation with reworking is repeated again, with different percentage reducing in cases, variants.

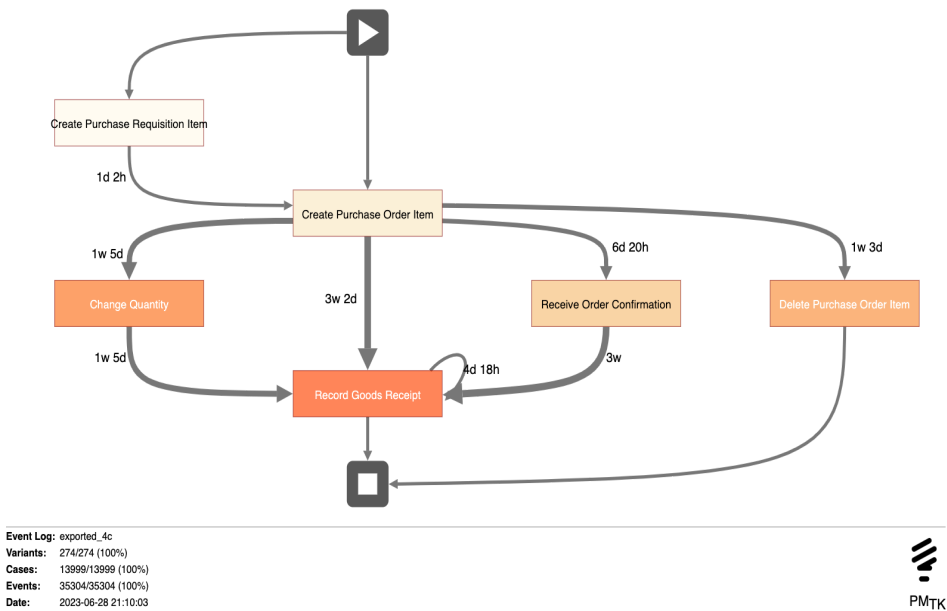




- 2-ways matching



- Consignment



There are no any huge optimizations by time filtering reworks or too distorted the process maps

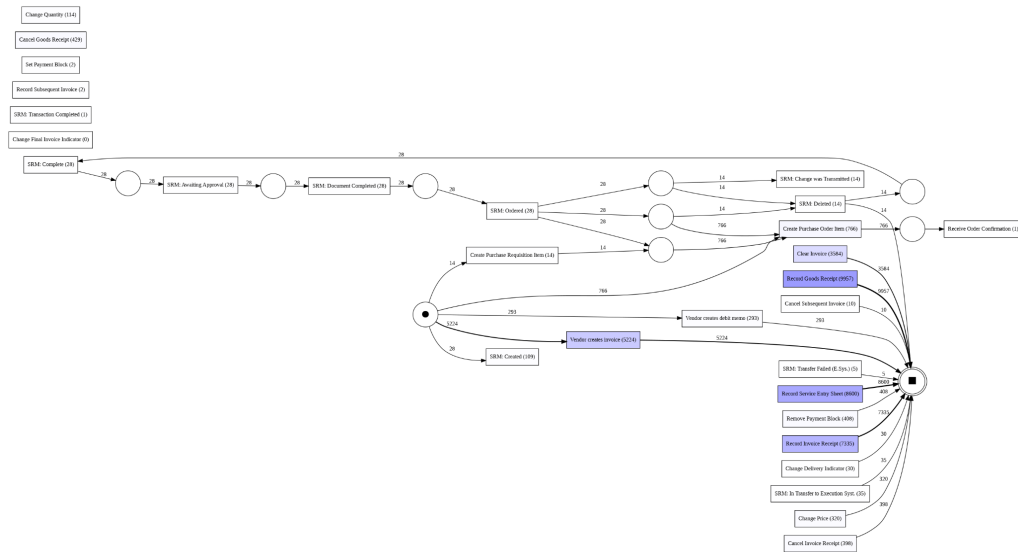
## 2.3 Process Discovery

### 3-way matching after GR

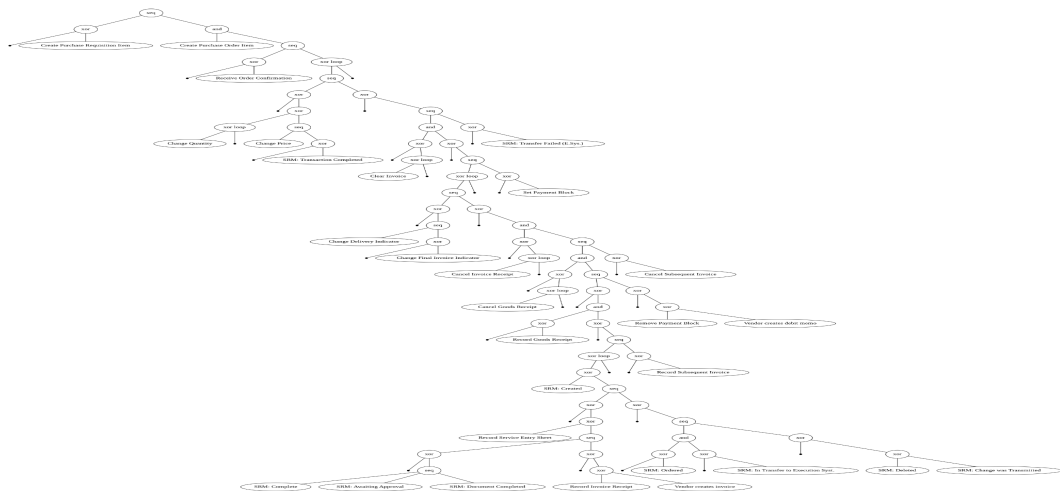
To visualization reduce the reworks on step **"Record Invoice Receipt"**

Below all diagrams Alpha mining, Inductive Mining with petrinets also heuristic mining with flow diagram fo first type. Detailed image will be provided on github repository.

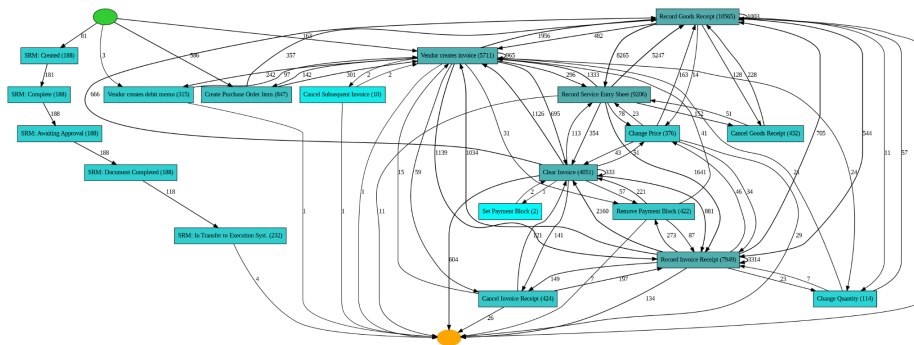
#### 1. Alpha mining- discovers causal relationships between events



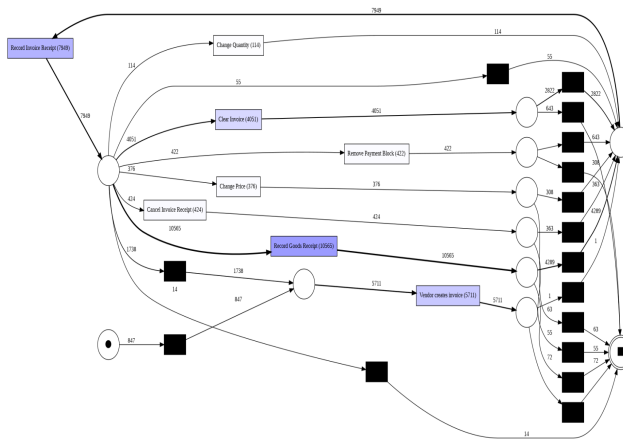
#### 2. Inductive mining representation in tree



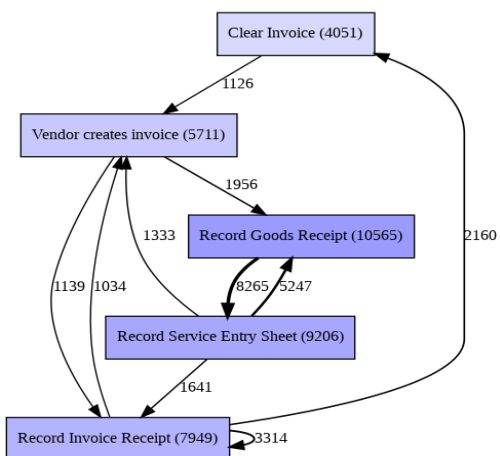
### 3. Heuristic mining relations representation



### 4. Heuristic with frequency representation

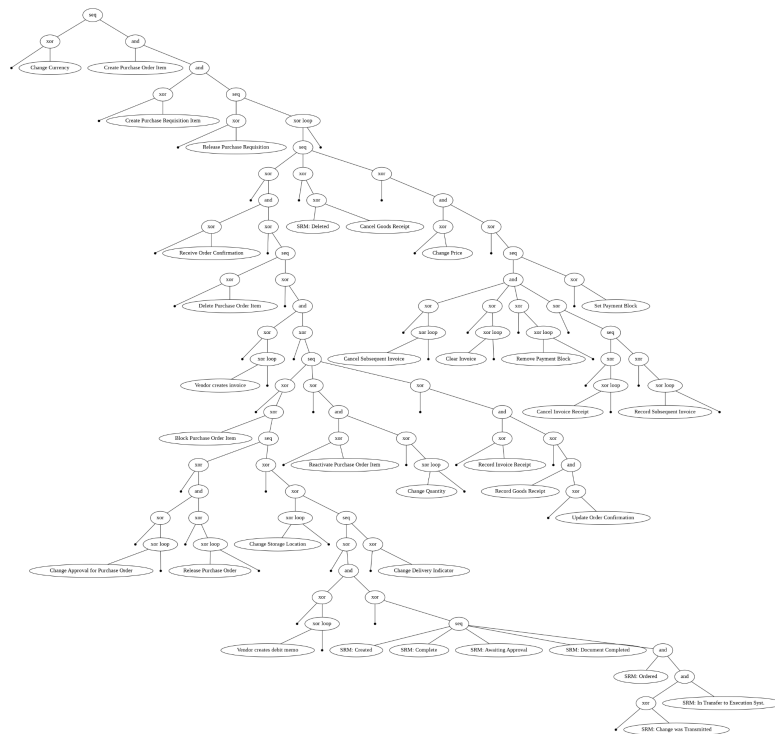


### 5. Graph from logs

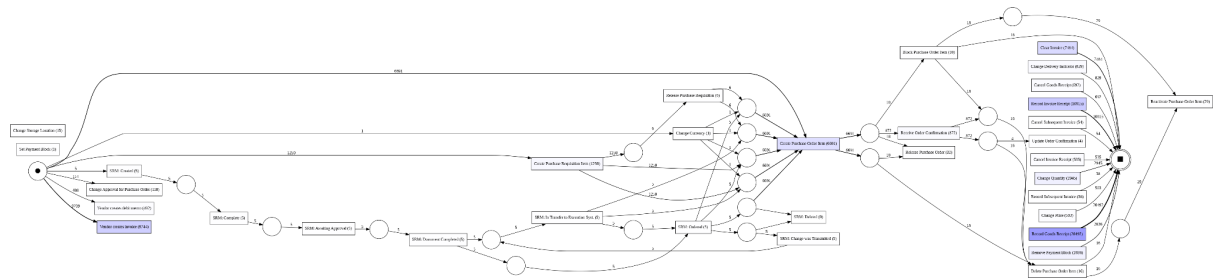


### 3-way matching before GR

#### 1. Inductive mining representation in tree

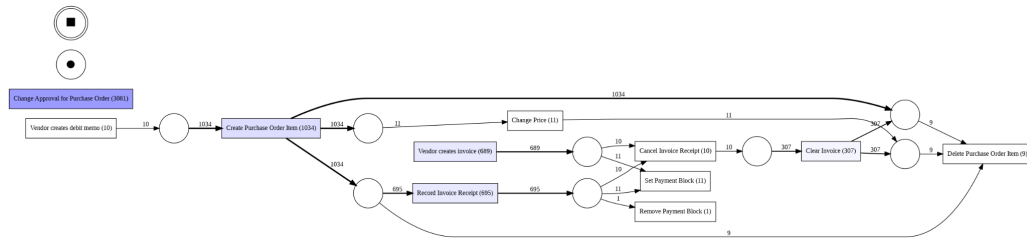


#### 2. The alpha mining graph represents relations between processes and is highlighted by frequency

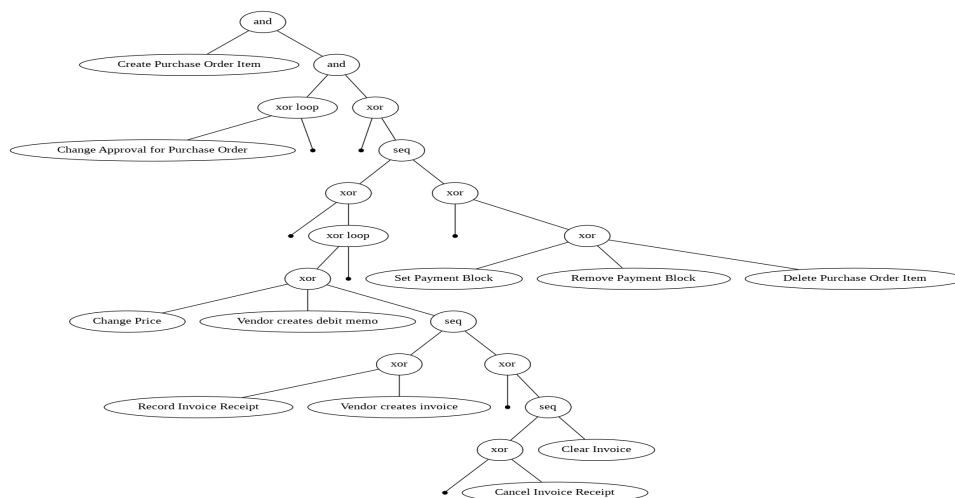


## 2-way matching

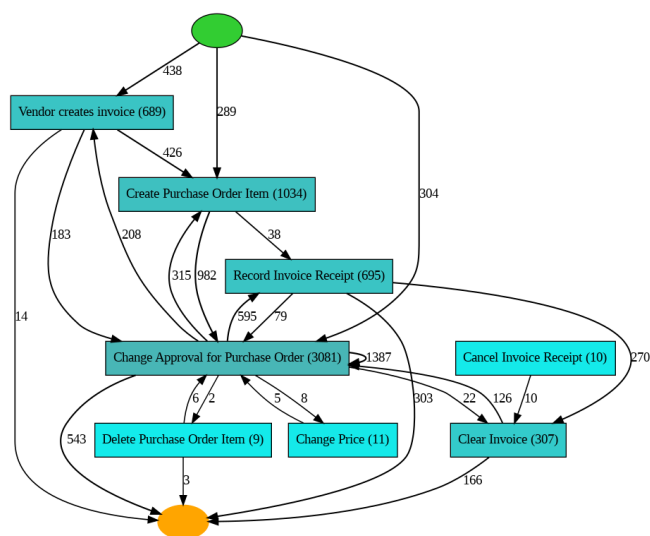
1. Alpha mining- discovers causal relationships between events



2. Inductive mining visualization

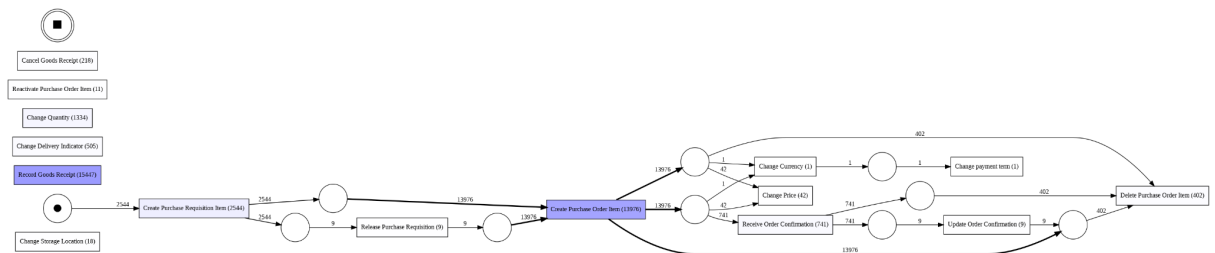


3. Heuristic mining visualization

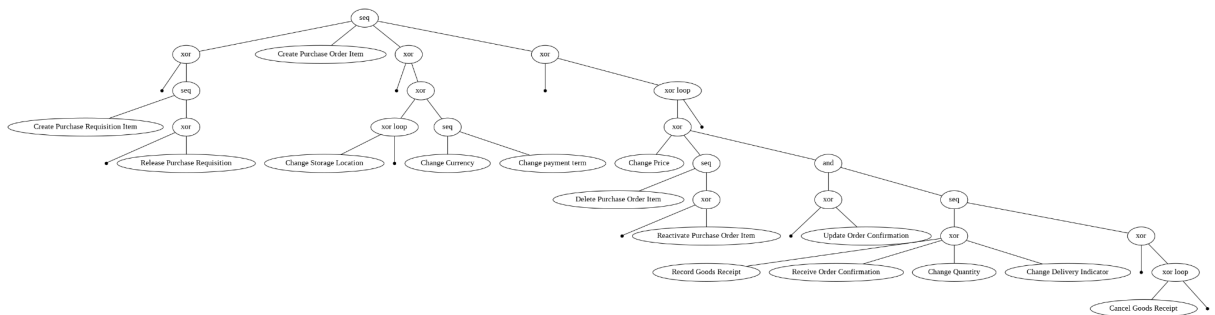


## Consignments

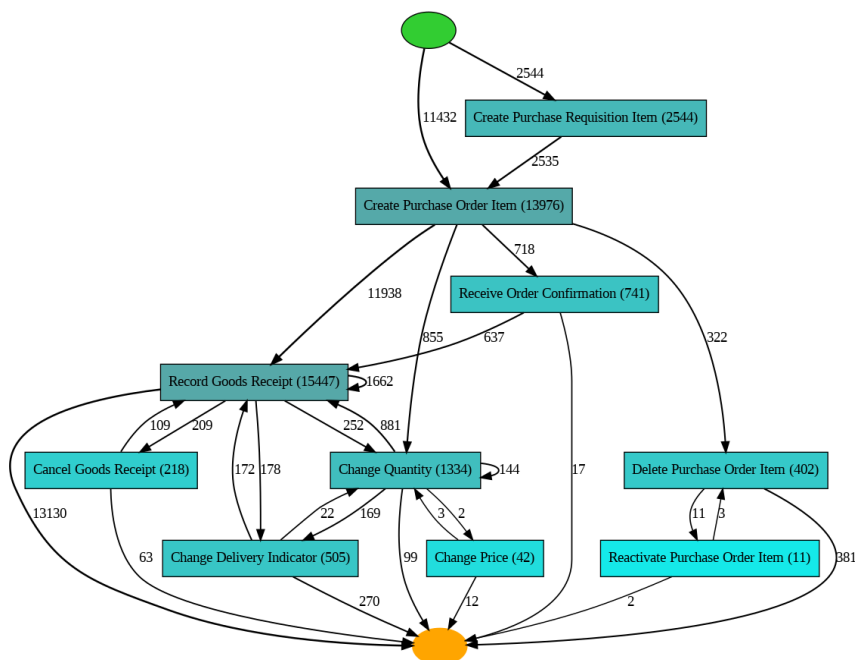
1. Alpha mining- discovers causal relationships between events



2. Inductive mining visualization



3. Heuristic mining visualization



## 2.4 Conformance Analysis

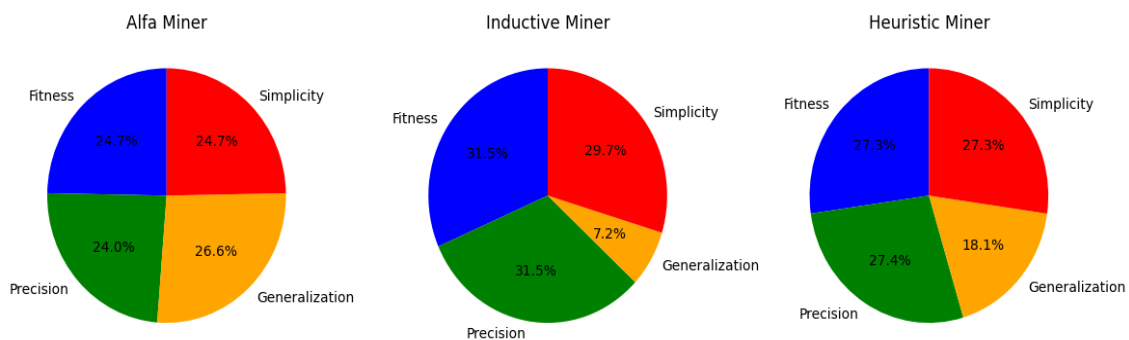
### 3-way matching after GR

Metric	Fitness	Precision	Generalization	Simplicity
Alpha Mining	'perc_fit_traces': 0.0, 'average_trace_fitness': 0.219, 'log_fitness': 0.163, 'percentage_of_fitting_traces': 0.0	0.152	0.776	1.0
Inductive Mining	'perc_fit_traces': 4.959, 'average_trace_fitness': 0.978, 'log_fitness': 0.971, 'percentage_of_fitting_traces': 4.959	0.168	0.890	0.610
Heuristic Mining	'perc_fit_traces': 0.0, 'average_trace_fitness': 0.647, 'log_fitness': 0.671, 'percentage_of_fitting_traces': 0.0	0.540	0.779	0.574

In summary, both Inductive Mining and Heuristic Mining outperform Alpha Mining in terms of fitness and precision, indicating better conformance to the log data and more accurate capturing of behavior. However, Inductive Mining shows a higher level of simplicity compared to Heuristic Mining, while Heuristic Mining demonstrates a higher level of generalization.

### 3-way matching before GR

Metric	Fitness	Precision	Generalization	Simplicity
Alfa Miner	'perc_fit_traces': 0.0, 'average_trace_fitness': 0.265, 'log_fitness':0.258, 'percentage_of_fitting_traces': 0.0	0.265	0.258	0.285
Inductive Miner	'perc_fit_traces':100.0, 'average_trace_fitness': 1.0000, 'log_fitness':1.0000, 'percentage_of_fitting_traces': 100.0	1.000	1.000	0.228
Heuristic Miner	'perc_fit_traces': 10.172 average_trace_fitness':0.942, 'log_fitness':0.946, 'percentage_of_fitting_traces': 10.172	0.942	0.946	0.625



In a comparative analysis of the metrics and visualized pie charts, Inductive Miner stands out with a perfect fitness score of 1.000 and a high precision score of 0.228, indicating a strong conformance to the log data. Alfa Miner shows moderate fitness and precision scores, with a fitness score of 0.265 and precision score of 0.285. Heuristic Miner demonstrates good fitness and precision scores of 0.942 and 0.625, respectively, but falls slightly behind Inductive Miner in terms of overall conformance.



## 2-way matching

title	Fitness	Precision	Generalization	Simplicity
Alpha Miner	'perc_fit_traces': 0.0, 'average_trace_fitness': 0.1585, 'log_fitness':0.1856, 'percentage_of_fitting_traces': 0.0	0.887	0.750	0.9130
Inductive Miner	'perc_fit_traces':100.0, 'average_trace_fitness': 1.0000, 'log_fitness':1.0000, 'percentage_of_fitting_traces': 100.0	0.344	0.837	0.5738
Heuristic Miner	'perc_fit_traces': 0.0, ' average_trace_fitness': 0.4769, 'log_fitness':0.3540, 'percentage_of_fitting_traces': 0.0	0.335	0.750	1.0000

In summary for 2-ways matching, the Inductive Miner algorithm achieves a perfect fitness score, indicating a higher level of conformance than the Alpha Miner and Heuristic Miner. However, the Alpha Miner algorithm performs well in terms of precision and simplicity. The Heuristic Miner algorithm has a moderate fitness score and the highest simplicity value.

### Consignment

Algorithm	perc_fit_traces	average_trace_fitness	log_fitness	percentage_of_fitting_traces	precision	generalization	simplicity
Alpha	0.0	0.0972	0.127	0.0	0.1429	0.758	1
Inductive	100.0	1.0	1.0	100.0	0.8609	0.798	0.628
Heuristic	81.855	0.9610	0.955	81.855	0.9993	0.954	0.647

In the table, the metrics for each algorithm are presented:

- perc\_fit\_traces: Percentage of fitting traces.
- average\_trace\_fitness: Average trace fitness.
- log\_fitness: Log fitness.
- percentage\_of\_fitting\_traces: Percentage of fitting traces.
- precision: Precision metric.
- generalization: Generalization metric.
- simplicity: Simplicity metric.

Based on these metrics, the Inductive Miner generally outperforms the Alpha Miner and Heuristic Miner in terms of conformance and precision for type consignment. However, the Heuristic Miner shows competitive performance with high precision and moderate conformance.

### **3. Project Results**

Each step of the project implemented using Python and the PM4Py, pandas libraries. The code cells in the submitted Colab project is referenced for each analytical step. The detailed results is included:

1. Preprocessed dataset: Cleaned and prepared dataset for analysis.
2. Process maps: Visual representation of the discovered process model.
3. Process discovering: Analysing with 3 type of mining
4. Conformance analyzing: based on previous step compare the best mining for each

## 4. Conclusion

In conclusion, the project "Business Process Analysis using BPI Challenge 2019 Dataset" aimed to analyze and gain insights from the BPI Challenge 2019 dataset, which represents a real-world purchase order handling process in the coatings and paints industry. The project successfully achieved its goals of analyzing the dataset, identifying patterns, and extracting valuable information to support decision-making and process improvement.

The project began with data preprocessing, which involved cleaning the dataset, handling missing values, and ensuring data quality. Duplicates and incomplete cases were removed, and a time constraint was applied to focus on the last three years of data.

Exploratory Data Analysis (EDA) was conducted to understand the dataset's characteristics and visualize process flows. The dataset was divided into four types of flows related to the line items, namely 3-way matching after goods receipt, 3-way matching before goods receipt, 2-way matching, and consignment. Process maps were created for each type, providing a visual representation of the sequence and dependencies of activities.

Process mining techniques were then applied to discover the underlying process models. Three mining algorithms, namely Alpha Miner, Inductive Miner, and Heuristic Miner, were used for process discovery. The results showed that Inductive Miner generally outperformed the other algorithms in terms of fitness and precision, indicating better conformance to the log data and more accurate capturing of behavior. However, Heuristic Miner demonstrated a higher level of simplicity and competitive performance with high precision.

Conformance analysis was conducted to evaluate the discovered process models' conformance to the log data. Inductive Miner showed a strong conformance to the log data, while Alpha Miner and Heuristic Miner demonstrated moderate conformance. The project's results provided valuable insights into the purchase order handling process, including identifying bottlenecks, inefficiencies, and variations in the process. The process maps and conformance analysis highlighted areas where the process could be improved or optimized.

Overall, the project successfully analyzed the BPI Challenge 2019 dataset, provided insights into the purchase order handling process, and laid the foundation for process improvement initiatives. The findings can support decision-making, identify areas for optimization, and contribute to enhancing the efficiency and effectiveness of the process in the coatings and paints industry.