Process Mining:

Application to a case study

by

Luís Filipe Nascimento da Silva

Master thesis in Data Analysis and Decision Support Systems

Overseen by

Professor João Gama

Faculdade de Economia

Universidade do Porto

2014

Acknowledgments

To my wife Fátima Silva and son Luís Silva who inspire me every day.

To my parents Manuel Silva and Albertina Nascimento and my brother João Silva who always supported me and made my professional and academic path possible.

To Prof. João Gama for his excellent orientation and availability to help. His suggestions and guidance given throughout this year greatly improved the value of this dissertation.

To my mentors, peers and colleagues, who I've been lucky enough to meet in my professional career, that challenged and gave me support during the last years: Peter Kavanagh, Elaine Wheatcroft, Helen Lewis, Jason Lenhart, Joe Kijowski, Robert Meissl, Marcus Guentert, Christopher Seeh, Nuno Silva, Rui Morais, João Magalhães, João Dias, Helder Gomes, Diogo Carneiro, Pedro Guerner, João Moreira, Sérgio Silva, Filipe Lourenço, Jorge Ferreira, Nuno Mota and many others. And a special thanks to Jim Dawson for showing the light.

Also to all my friends especially the ones that accompanied me during all the latest adventures and stayed with me through all endeavors: Tiago Silva, Jorge Fonseca, Eurico Moreira, Tiago Baldaia, Geminiano Piedade, Rogério Martins, Luís Pereira, Apolónia Neusa, Miguel Brandão, Célio Soares, Bruno Carneiro, and of course to my long time and best friend Bruno Monteiro.

This work was supported by Sibila research project (NORTE-07-0124-FEDER-000059), nanced by North Portugal Regional Operational Programme (ON.2 O Novo Norte), under the National Strategic Reference Framework (NSRF), through the Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT), and by European Commission through the project MAESTRA (Grant number ICT-2013-612944).

Abstract

The practical relevance and importance of Process Mining is increasing with the growth in availability of event data. Process Mining techniques aim to discover, monitor and improve real business processes by extracting knowledge from event logs, also known has business transactions. The three most prominent features of Process Mining are process discovery, i.e., learning a process model from example behavior recorded in an event log, conformance checking, i.e., diagnosing and quantifying discrepancies between observed behavior and modeled behavior and performance analysis, i.e. review of the process execution metrics. The broad, practical and relevant appliance to the real business world processes may make Process Mining one of the most important fields in data mining, but still much proof and improvement is needed.

In this particular endeavor, the objective to apply Process Mining techniques using ProM framework tool to an practical case, by extracting business transactional data regarding customer order handling from SAP ERP system and applying a process discovery algorithm to find the real business process model. Data cleaning and transforming is the first and most important step to any analytics and is the first stepping stone to build a process model based on event records and using Process Mining techniques and tools. Noise (outliers and rare activities), incompleteness (missing events), and event correlation (gaps or missing links between events and events and activities) have proven to be big challenges for the application of this technique. This paper will focus on how to execute a particular case of Process Mining and record the user experience and future improvement needs.

Table of contents

Table and graphics index

Diagrams

Figures

Tables

1. Introduction

This dissertation is divided into six main chapters. The goal of this structure is to enable the introduction of this topic, telling the story from theory to execution, to anyone interested in Process Mining. The chapters will cover the following topics:

Act One

1. Problem Definition
2. State-of-the-Art

Act Two

3. Theory
4. Applications

Act Three

5. Execution
6. Conclusions

In the first chapter we will understand the motivation and what has driven this paper. It also covers the thought process for scope definition, tool selection, and gives a quick preview of challenges and opportunities based on previous knowledge. This is basically the starting point of this adventure. To complement the first chapter, in the second we shall cover the history and background of Process Mining. The major advances, contributions and blocks. We will also try to describe and define Process Mining based on recent literature. This ends act one, the introduction of all the characters and motivations in the story.

Second act is composed by two chapter Theory and Applications. In this chapter we will go a little deeper in the definition of Process Mining by describing the techniques, types, perspectives, goals and objectives. The Applications chapter is still a theoretical approach and serves as a bridge to the execution, namely defining what we should be expecting of the Process Mining techniques, the requirements and features of such tools.

Third and final act is composed by Execution and Conclusions. Here we will have the climax of the paper, by applying the methods described before to a real case. This may also be used as a road book, guide or user manual for someone starting in this topic. From the extraction and conversion of data, to the selection of algorithms, execution and creation of process models. In the end we will try to sum it all up in a few words.

Some topics will be repeated on purpose, the first time they will be approached in less detail, superficially, because there is no background and the second time they will be drill down and

understood in more detail in light of the gained background knowledge obtained from the previous chapters. Like in many other subjects we need a few iterations before grasping the full concept. In the conclusions chapter we will see some notes and comments about the whole process and more important some light on potential improvement opportunities and/or alternatives to Process Mining dedicated tools.

Hope you enjoy reading this paper as much as I enjoyed writing it.

2. Problem definition

2.1. Introduction

This chapter will cover the dissertation goals, expected challenges and planned execution method. Here we will understand the motivations, objectives, expectations and approach of this work. This will also cover some topics about Process Mining and ETL.

2.2. Goal

The goal of this work is to apply Process Mining theory, that is, to capture document flow and transactional data for a selected process scope and obtain insights about the "real" process. The process scope will cover part of the Sales and Distribution range, namely the Customer Order Handling from a known ERP system (SAP). The objective is to create a process model using Process Mining techniques and ProM Framework tool, obtain relevant information about it and compare it to the theoretical design.

In this path, we will attempt to turn event data into real value by discovering the process model. This paper aims to test the value of Process Mining in a concrete case and evaluate its application viability. The company name will be maintained confidential. This company is focused in the development and production of specific solutions for other businesses, that is, industry to industry.

The whole data set will be extracted from a single data source. The event log base data will be therefore extracted directly from ERP standard tables, in this case SAP. After defining the process scope and identifying the tables containing relevant process data, ODBC Connections and/or Data Browser transactions will be used to capture the data. Afterwards, this data will be organized and transformed to fit the event logs layout predefined by the ProM tool. The transformation will be done using simple Visual Basic language in a user-friendly interface that will allow the selection of attributes from a source table using simple Access queries. The selection of the source queries will be done from an Excel interface that will contain all the needed routines. The transformation process means picking up a record / table line and copying it down into a list format (event log), grabbing the relevant attributes and associate all records to each specific case. No hardcode will be used, so the approach will be applicable for any type of source data with some minor adjustments.

The starting point for Process Mining is an event log that results from the transformation of SAP tables containing process transactions / event records and document flow links. Each event refers to a process instance (case), a customer order, and an activity, ranging from customer order entry to sales invoice. Events are ordered and additional properties (e.g. timestamp or originator) may be also included.

The result will be a Customer Order Handling process model that can be seen as a “map” describing this operational process. We will see that operational processes like this leave trails in the information systems supporting them. But we need to keep in mind, whereas an event log extracted from the system describes example behavior of the underlying process, the resulting process model aims to describe an abstraction of the same process.

2.3. Challenges

The completion of this goal posts some challenges. For process mining techniques to work it has to be possible to sequentially record events such that each event refers to an activity (well-defined step in some process), and is related to a particular case (a process instance). Additional information like resource performing an activity (person or device), the date and time (timestamp), and other relevant information like start and end time, volume, revenue or cost may also be stored in the event logs. All these extra information’s are very useful to perform process mining and retrieve meaning from data, but in the beginning is better to start with a very simple skeleton of the process with no meat at all. The first challenge is trying to find and fit standard ERP tables that are not originally projected to be used for Process Mining; they are not true event logs, into the described requirements. There are some modules like Process Observer that may facilitate this task, but they are not implemented or available, therefore the data needs to be converted outside the system.

As described in the first chapter, data capturing and transforming, i.e. data extraction, selection of relevant attributes and transforming it into an event log format readable by the different platforms is the first big concern of this case study. The first efforts will be focused in the creation of a simple method to select and transform standard SAP tables into an event log readable format. More shall be said about this format in the next section.

The second challenge will be data cleanup and preparation. Of course some event log quality checking and cleaning will be needed. This will be performed in the data selection process, namely verifying the following criteria:

1. Validity: represent real events;
2. Accuracy: represent how event happened;
3. Completeness: no events missing in scope for each case;
4. Restricted-access: cover privacy and security requirements;
5. Structure: convert data based on the defined semantics.

The goal will be to clear out the usual data issues like noise (outliers and rare activities) and incompleteness (missing events), solving other problems like event correlation (gaps or missing links between events and events and activities) and putting the whole dataset in the right format.

To sum it all up and in a more broad analysis, in Process Mining, at some point in time you will have to deal with the following aspects:

- Scoping: selecting relevant data distributed through thousands of tables;
- Transformation: picking up non structured data and transforming it into a readable format;
- Snapshot: filling the gaps of partial recording of events;
- Ordering: event need to be ordered, using for example timestamp when available;
- Correlation: relating events to each other and to the same case may be a big issue;
- Granularity: different granularity from the activities need to be reviewed.

2.4. Tool Selection

Nowadays there are several commercial (paid / licensed) tools for process mining and a few academic free versions. Commercial versions include software like Aris PPM and HP BPI and academic versions include solutions like ProM, EmiT and Little Thumb. With this last one's already discontinued, ProM seems to be the most up to date free tool around.

The ProM open source initiative started in 2003, after some early prototypes. ProM is a plug-in architecture, so you need to install packages that contain several plugins that may work together with plugins from other packages. Today there are over 300 plugins available that can be divided in families or categories taking into account their main function: mine, analyze, import, export, convert and filter. This functions and some of the available plugins are (in parenthesis the number of plugins available in each function):

1. Mining: alpha miner, heuristics miner, fuzzy miner, etc (+40)
2. Analysis: verification, SNA, LTL, conformance checking, etc (+70)
3. Import: EPC loading, Petri Nets, YAWL, BPMN, etc (+20)
4. Export: EPC storing, Petri Nets, YAWL, BPMN, BPEL, etc (+40)
5. Conversion: Translating EPC or BPMN into Petri Nets (+45)
6. Filter: remove events, etc (+25)

Additionally, ProM Import is a complementary tool that should be used together with ProM. It is also free and can be used to extract the MXML file from several types of applications and formats.

The goal of ProM framework is, like the name implies, the extraction of knowledge from event logs like prescribed by Process Mining. ProM can be used against audit trail logs obtained from Workflow

Management (WFM), ERP or CRM systems, for example. This tool is a "pluggable" environment for process mining. It is a flexible framework in respect to the inputs and out formats. It is also open enough to allow for easy reuse of code during the implementation of new process mining ideas.

Process mining has been around for more than a decade, and it has proven to be a very fertile and successful research field, but may need to prove itself in the market. Part of this academic success can be associated to the contributions of the ProM tool, which combines most of the existing process mining techniques as plug-ins in a single tool and it is free and open. This is why ProM was selected for this case study.

Different business tools use different formats for reading or storing log files (input) and show their results in different ways (output). It is often very difficult to combine the different tools although it would be very useful. Typically researchers working on process mining are forced to build a mining infrastructure from scratch or test their techniques in an isolated way, disconnected from any practical application.

In respect to the input format, ProM tool like any other tool needs to be fed with a specific type of file, in this case a MXML (Mining eXtensible Markup Language) type with a specific format/layout. This layout allows the user to populate attributes or proprieties in a list format. This proprieties are all associated with the same case, therefore can enrich the final result, with information about who performed the task, when the task was performed and the type of task performed.

The free version of ProM available requires the use of an MXML file that may be created using ProM Import and is composed by four different source dimensions that can be translated in a database environment as four different tables:

Table 1 – ProM Import Input Tables

Tables:
Process Instances table
Audit Trail Entries table
Data Attributes table for the Process Instances
Data Attributes table name for the Audit Trail Entries

Each table contains different attributes for each data type:

Table 2 - ProM Import Input Table Fields: Process Instances

Process_Instances	
Name	**Description**
PI-ID	Unique process instance (case) identifier
Description	Process instance description

Table 3 - ProM Import Input Table Fields: Process Instances Attributes

Data_Attributes_Process_Instances	
Name	**Description**
PI-ID	Foreign-Key (Process instance unique identifier)
Name	Process instance name
Value	Process instance value

Table 4 - ProM Import Input Table Fields: Event Trails

Audit_Trail_Entries	
Name	**Description**
PI-ID	Foreign-Key (Process instance unique identifier)
ATE-ID	Audit trail event unique identifier
WFMElt	Task, event, trail or activity name (Workflow Element)
EventType	Task, event, trail or activity type
Timestamp	Date and time
Originator	User or task responsible

Table 5 - ProM Import Input Table Fields: Event Trails Attributes

Data_Attributes_Audit_Trail_Entries	
Name	**Description**
ATE-ID	Foreign-Key (Audit trail event unique identifier)
Name	Audit trail event name
Value	Audit trail event value

Besides understanding the tables and their layout it is relevant to understand how they relate to each other that is what are the primary and foreign keys. The following diagram shows the table relationship and a brief description of their content follows.

The basic idea is a Process Instance (PI_ID) contains several Events (ATE_ID) and each of them may contain several attributes. Every Event must be allocated to a single case, therefore the key PI_ID is a primary key for the Process Instances table and a foreign key for the Event Entries table. In Process Mining a process model describes the life-cycle of a case of a particular type. This means events need to be related to cases and all events or activities in a conventional process model correspond to a status change of the case. Therefore each trace describes a sequential list of events corresponding to a change of a particular case.

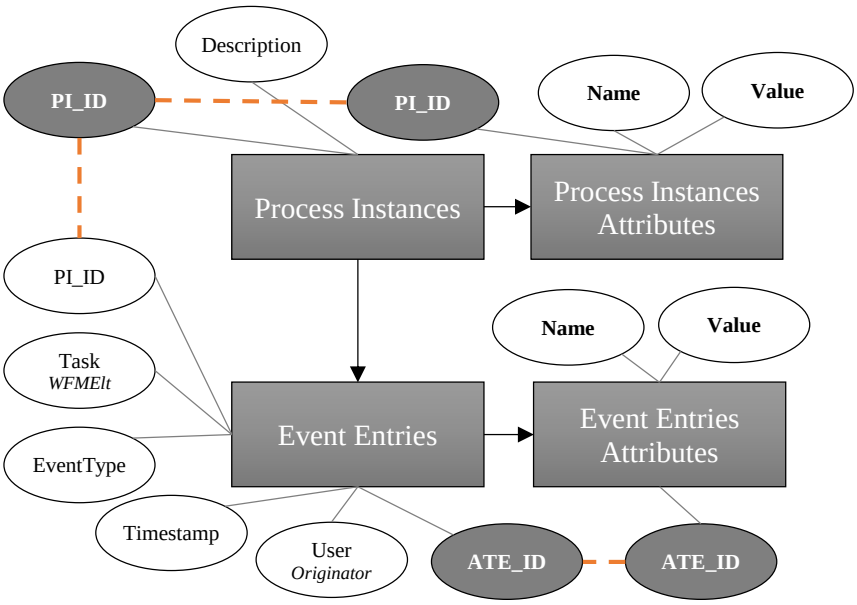


Diagram 1 - ProM Event Log Data Structure

In this structure, each case or process instance has a process id (PI_ID), may have a description and additional attributes like name and value. Each case or process instance has an audit trail, i.e. a record of event entries. Each event is like a task or activity that is performed in sequence. This sequence is uniquely identified by the audit trail entry id (ATE_ID). Like the process instances events may have additional attributes, like name and value.

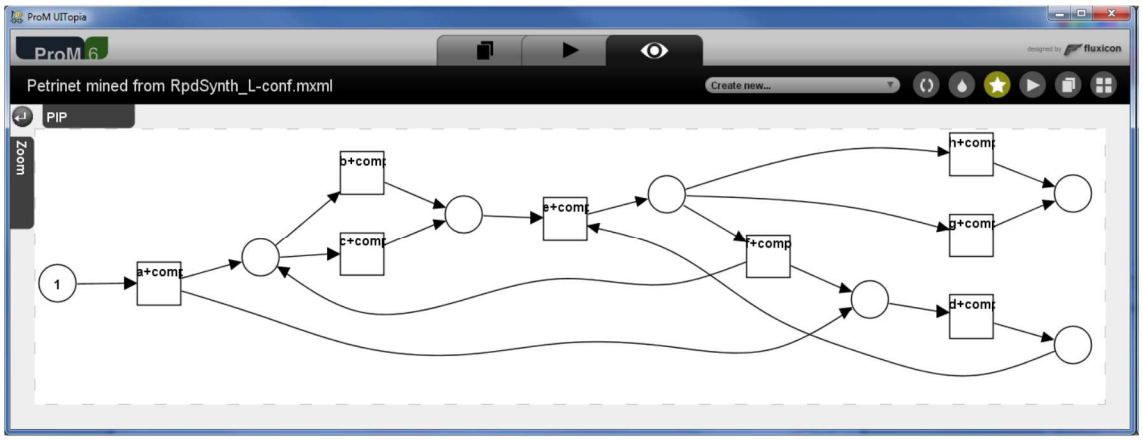


Figure 1 - ProM Tool Output Example (Petri Net)

Herein using a process mining discovery algorithm the result of ProM tool will be a Process Model, for example translate into a Petri Net type of process model as shown in the examples below. We need to keep in mind there are several process model notations like BPMN, UML and EPCs, for

example. It is possible to convert between process models once a discovery algorithm is used. This resulting Petri Net will describe the real design of the selected business process.

In ProM the process mining algorithm is able to discover a Petri Net type of process model by identifying process patterns in collections of events. The Petri Nets are only one of many available process modelling notations. Basically Petri Nets are composed of only three different elements: places, transitions, and arcs. In simple terms, each node represents a transition (events) or a place (conditions), and arcs (traces or flows) describe which places are conditions for which transitions (events).

2.5. Process Scope

In this case study Customer Order Handling was the process chosen to be analyzed and reviewed. Each customer order may consist of multiple order lines as the customer may order multiple products in one order. One customer order may result in multiple deliveries or shipping's. One delivery may refer to order lines of multiple orders. Hence, there is a many-to-many relationship between orders and deliveries and a one-to-many relationship between orders and order lines. Given a database with event data related to orders, order lines, deliveries and invoices, different process models can be derived from it. The following diagram shows the macro level process view that will be the scope of this case study. This case study will focus on this main flow, from the moment the customer submits an order to the moment the product is shipped from the company's warehouse and is invoiced.

Diagram 2 - Case study scope

One can extract data with the goal of describing the life-cycle of an individual order. However, it is also possible to extract data with the goal of discovering the life-cycle of individual order lines or the life-cycle of individual deliveries. The outcome of this case study will be a process model from Customer Order to Customer Invoicing following the life-cycle of individual orders, aggregating order lines and considering an order is completed when all order lines are closed.

2.6. Data Source

This process is supported by the companies ERP system, in this case SAP. The data from SAP that covers this process is integrated in the Sales and Distribution (SD) module. The main tables are the Sales Documents headers and details (VBAK and VBAP), the Delivery Documents header and details (LIKP and LIPS), the Invoice Documents header and details (VBRK and VBRP) and the

Sales Document Flow log (VBFA). More about the SAP data model, how the tables relate to each other, what information each of them contains about the process in analysis and how we can turn it into and event log using ProM Import tool will be developed in the following chapters.

2.7. Work Plan

Before starting any practical endeavor it is important to delineate a work plan and a work methodology. In this case study the Process Mining Life-Cycle shown below will be taken into account as a guideline for the conclusion of this work.

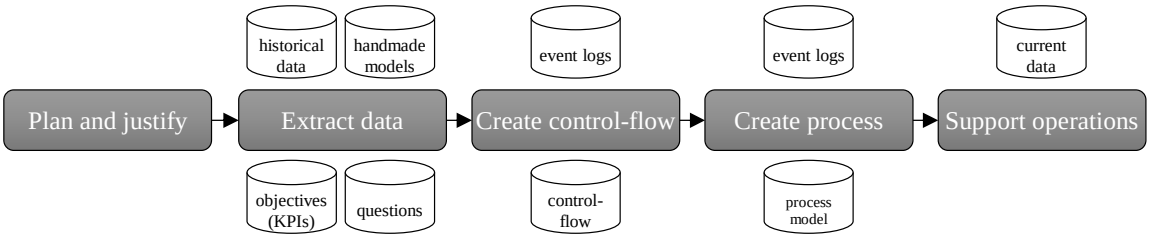


Diagram 3 - Process Mining Life-Cycle

Like in every problem solving assignment we start with a question and then try to find a solution for it. In this case the problem is finding or discovering the hidden business process model for Customer Order Handling. This process is hidden in the transaction records and document flow of SAP that will be extracted for a 3 to 6 month period, depending how long a case takes to be solved on average. Some iterations and repetitions may be needed since noise (outliers and rare activities), incompleteness (missing events) and gaps (missing links between events) need to be cleaned-up manually. Therefore some adjustments may be need in the source data.

3. State-of-the-art

3.1. Introduction

In this chapter we will cover the basic definition of process mining, its context, background and history and current challenges, giving some light about its future. The definition section summarizes all of the current references to Process Mining and explains what Process Mining comprises. In the history section we will explore its context and relation to other disciplines like Business Process Management and Data Mining and its short history. In the end of this chapter, looking at the future of process mining, some of the current main challenges and improvement opportunities are listed to show all the potential of growth for this fresh new discipline.

3.2. Basic Definition

In short words, Process Mining refers to an automated creation and construction of process models based on information system event logs and other event records.

The aim, like shown in the next diagram is to automatically create a process model (abstraction) of the real process (unknown) based on recorded events that in most cases covers only part of reality or possible behaviors.

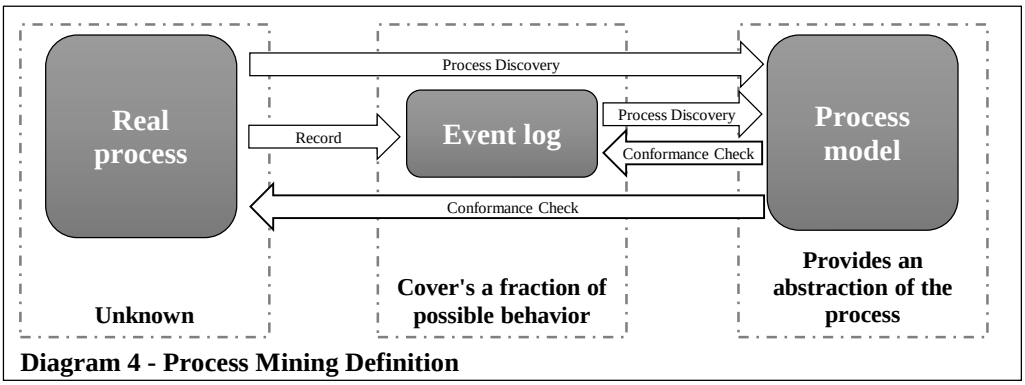


Diagram 4 - Process Mining Definition

On the one hand, automated means it is supported by software applications that receive an event log (input) reflecting the real process and produce, based on different algorithms, a process model representation. On the other hand, a process model refers to the visual and structured representation of processes, for example business processes such as customer order entry and complaint management, and other processes like health treatments, city traffic, etc. This structured view of a process is in fact an abstraction and simplification of the real process, the world, and the behavior of its intervenient enabling insights and knowledge gathering for decision making that the complex reality of things obscures. The starting point of Process Mining is therefore the event log, that is, a record of the sequential transactions, events or behaviors that took place in a process. This event logs

may be structured and maintained in a system, like an ERP or Workflow, or may be scattered and informal and maintained in files or different machines that record events that partially explain a process.

This definition maybe short and succinct and considered incomplete, since the authors, developers and creators of Process Mining also refer to the comparison of process models that are already structured for compliance or enhancement purposes and the execution of performance analysis. But in truth the process model creation feature is the main attribute of this research field and all other are a natural consequence of the first, they are applications and outcomes of the Process Discovery.

With that said, we shall consider that Process Mining techniques allow to extract knowledge from event logs. For example, the audit trails of a workflow management system or the transaction logs of an enterprise resource planning system (ERP) can be used to discover models describing processes, organizational roles and networks, etc. Furthermore, it is possible to use Process Mining to monitor process deviations, for example by comparing the observed events (reality) with predefined models or business rules such as SOX (Sarbanes Oxley).

Process Mining can be considered a new area in business intelligence that aims to learn processes from recorded actions. Of course this can only be done for structured processes, that is, behaviors that follow a consistent rule and sequence and can be associated to the same case. If random and unexplained or anarchic events took place it would be impossible to understand the process, because there is no process, no cause and effect, no relation between an action and its consequences.

The events relationship needs to be rational, meaning that there is a known or unknown logic behind them that will be understood and structured via process modeling. Also these events need to be observed, that is, we should be able to record and monitor the execution of activities or message exchanges.

As explained before process mining is not limited to control-flow discovery, the discovery feature is just one of the three basic forms of process mining and the scope is not limited to control-flow; the organizational, data, case and time perspectives also play an important role in this discipline. This topic will be detailed in the next chapter about process mining applications.

In the next section we will understand the background of process mining, but for now we should understand that Process Mining is not just a specific type of Data Mining, but it can be seen as the “missing link” between the data-driven Data Mining and the traditional model-driven Business Process Management (BPM) areas of study, since most data mining techniques are not process-centric and BPM doesn’t build process models from data and recorded events.

The following diagram shows the possible relationship between the three different subjects on the left the model-based BPM, on the right the data-based DM and in the middle the data and process-based Process Mining:

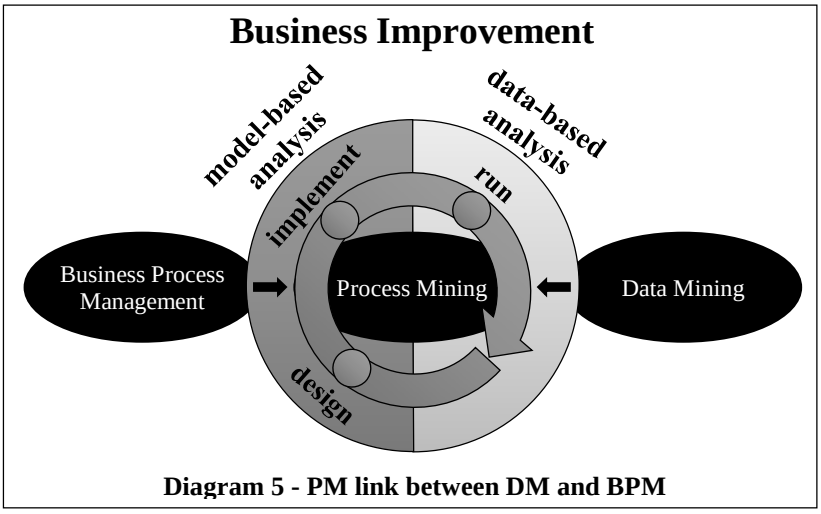


Diagram 5 - PM link between DM and BPM

Also Process Mining is not limited to offline analysis, although knowledge extraction is done from historical event data, were post-mortem data is used, the outcome can and should be applied to running cases, i.e. pre-mortem, for example the completion time of a partially handled customer order can be predicted using a discovered process model.

In the rest of this paper we will have a chance to explore and go into all of the detail about process mining, where and how it is applied, types and perspectives, goals and issues and also the existing tools that make this approach real and tangible in the company or business organization.

3.3. History and Background

If Process Mining is the missing link between Data Mining and Business Process Management it is worth to understand both histories and their relationships.

The origins of Data Mining remote to the XI century when we developed basing methods for decision making based on, for example, means (descriptive statistics) and inference (statistical inference). By the middle of the last century (XX) we had created the first intelligent machines (artificial intelligence) that apply human-thought-like processing to statistical problems and systems that could learn from data (machine learning).

On the other side of the spectrum Business Process Management as we now today is quite a fresh concept dates to the beginning of this century (XXI). The first software applications were developed in the 60' and the first database systems in the 70'. User interfaces were only introduced in the 80' and the first BPM systems only started appearing in the XXI century. But the ideas themselves are

older. Petri Nets were developed by Carl Adam Petri in 1962 and the Information Systems Theory Project in 1968 by Anatol Holt. However the ideas of Workflow Management and Business Process Management in practical terms only saw the light of day in the first decade of this century (XXI).

Parallel to the development of Business Process Management ideas the first principles of Process Mining appeared in the 60' with Mark Gold's language identification theories and also Baum-Welch (1970) and Viterbi (1967) on the algorithms to learn hidden Markov models. Myhill-Nerode theorem (1958) and Biermann/Feldman algorithm (1972) are other examples of the concepts and ideas that were developed at that time. But, as in BPM only in the last decade in this century Process Mining has really been driven and brought to life with some practical applications. Previous theories weren't able to deal with some of the biggest problems with Process Mining, namely: concurrency, noise, incompleteness, end-to-end scope and precise and formal semantics. This new devolvement's are centered in Eindhoven University of Technology, namely Wil van der Aalst who is making a lot of progress in this area of study, namely with his book Process Design by Discovery: Harvesting Workflow Knowledge from Ad-hoc Executions (1999). We can say this new developments are being driven by Aalst, however there are also other developers that built process mining tools before ProM (Process Mining), but most of them are discontinued or are short in terms of what is expected for a Process Mining tool. The articles and papers publicly available about this subject clearly indicate that at least at an academic point of view the biggest and most active player in the Process Mining area is Aalst and center all major developments are occurring in Eindhoven. The first process mining tool to support the alpha algorithm for process discovery was the MiMo (Mining Module) tool based on ExSpect, only later EMiT and ProM were developed and released. However both MiMo and EMiT are considered inactive and discontinued today. More about the tools for process mining will be detailed in the Tool section of the Problem Definition chapter.

In the last decade process mining emerged as a new scientific discipline concerning the interface between process models and event data. The conventional Business Process Management (BPM) and Workflow Management (WFM) approaches and tools are typically model-driven with little or no consideration for data, in this case event data. On the other hand, Data Mining (DM), Business Intelligence (BI), and Machine Learning (ML) focus more on data without considering the end-to-end process model. With that said, Process Mining aims to bridge the gap between BPM and WFM on the one hand, and DM, BI, and ML on the other. The rationale is to turn streams of event data sometimes referred to as "Big Data" available in today's systems into valuable insights related to performance and compliance. The outcome of these techniques can be used to identify and

understand bottlenecks (effectiveness), inefficiencies (optimization), deviations (compliance), and risks (control). Therefore Process Mining will help organizations, public or private, to explore or mine their business processes enabling them to discover, monitor and improve the real processes by extracting knowledge from event logs.

It is therefore worth referring how Process Mining comes to be, namely how it relates to the other disciplines like Business Process Management and Data Mining. In the first place Business Process Management (BPM) is focused in process modeling and analysis, its enactment or simulation, review, verification and improvement. In the second place Data Mining (DM) is focused in clustering, classification and rule discovery based on recorded data. Bringing the two together we have Process Mining (PM), i.e. a data mining supported process modeling discipline, a mixture of both DM and BPM. Process Mining aims to build process models based upon the same methods of Data Mining, i.e. finding the rules that oriented the flow of processes in the recorded events (data). The difference would be most process models are built upon "tribal knowledge", i.e. the empiric know-how, and subjective experience of the resources that perform or manage the different activities and/or the creativity of the resources, sometimes consultants and internal auditors, who design and document the process. There are several risks of misinterpretation of what is happening, what is observed, what is described and finally what is designed and documented as the process model. Process Mining uses real events to show us how processes developed in reality, based on the history of transactions performed by different resources from the beginning till the end of the process.

All this related disciplines like Business Process Reengineering, Business Intelligence (BI), Operations Management, Workflow Management (WFM), Data Modeling and Office Automation are associated to Process Mining and Business Process Modeling, in their own way. These areas of study focus on process control, optimization and improvement. But we should keep in mind that there is no way we can review and optimize a process or make decisions without knowing how things work. First of all we need to define and understand the business process and at some point monitor and control it, before we can truly start to improve it. That is to say we can only deliberately improve what is defined, controlled and measured. Improvements based on unknown, undefined, non-controlled and unmeasured processes are mainly a result of chance and/or arithmetic's, for example when the resources involved try to show the glass half full not understanding the drivers and not having any accurate starting point and insight to identify opportunities, undertake corrective actions and improvement measures and monitor its results and outcomes. Most managers base their decisions on rough indicators and mainly their own orientation, self-interests, experience and what can

subjectively be called professional sensitivity. Sometimes the results are good and compliments and performance incentives are given based on the good work. Other times results are not that good, but there is always an unforeseen variable given to justify it. The main factors of success cannot be at the same time the drivers for induces. Experience and professional sensitivity cannot support success and also be the reasons for failure when it blinds the decision making. This type of informal and subjective knowledge is basically based on informal historical information maintained in the managers own database, most of the times some unstructured notes and mostly the brain, what was previously named “tribal” knowledge. What if we can pull information directly from the business to make better decisions? What if we can control a process and detect deviations before they happen based on the main behavior variables and drivers? Success may not be possible to plan but surely an organization’s administration can improve their chances if they base their decisions on existent and structured data, what will be called “organizational” knowledge. This is information that is constructed and maintained throughout the business transactions without any additional relevant and deliberate effort that can be used to build meaningful insights for decision making. Why ignore it?

Contemporary information systems like Workflow Management (WFM), Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), Supply Chain Management (SCM), and Business to Business (B2B) systems record business events in so-called event logs. Business process mining means to take these logs to discover process, control, data, organizational, and social structures. Although many researchers are developing new and more powerful process mining techniques and software vendors are incorporating these in their software, few of the more advanced process mining techniques have been tested on real-life processes.

3.4. Context and Integration

The first name for this section was Process Mining versus The World, because basically it will cover the context, complementarity and relationship with related areas, like Business Intelligence, Data Mining, Standard Query Tools, etc.

Starting with the first and wide-ranging concept, Business Intelligence is a very large concept that includes many areas from ETL process, Data Warehousing and Reporting and Analytics. The following diagram tries to describe the full BI process:

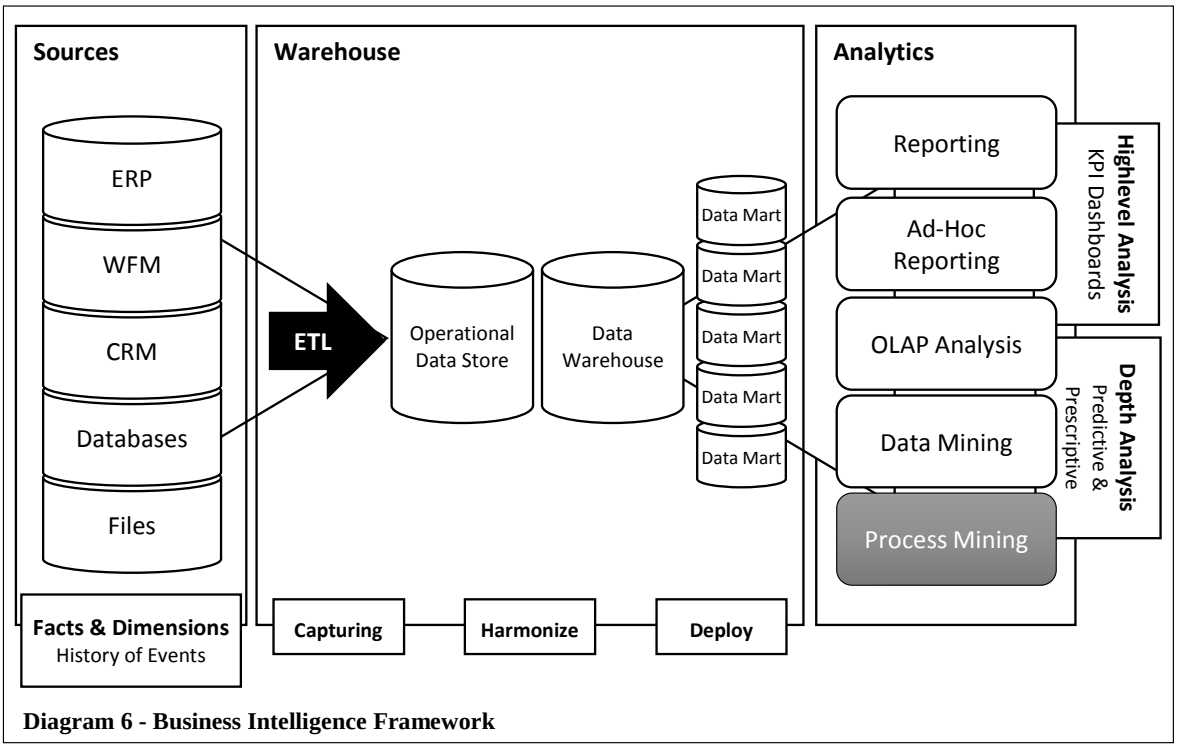


Diagram 6 - Business Intelligence Framework

In this context Process Mining with Data Mining could fit in the end field that is called Reporting and Analytics, but in my experience it is also helpful before to help harmonized, clean and enrich the source data. BI tools include SAP BW (Business Warehouse) and BO (Business Objects), Cognos Business Intelligence (IBM), Oracle Business Intelligence, Hyperion, SAS Business Intelligence, Microsoft Business Intelligence, Jaspersoft, Pentaho BI Suite (this last two are open source), etc, but usually are data-oriented and therefore may cover some Data Mining features, but not at all Process Mining. This tools support mainly dashboards, reports, scorecards and also data transformation (slice and dice), data mining, etc.

By name and concept we should also understand the different or similarity with Data Mining. Process Mining uses basically the same concepts and ideas of Data Mining, but it can be considered a kind of spin off. It joins Process Modelling concepts and requirements with Data Mining type algorithms and ideas to create what the authors and developers determine to be a new academic subject. In my notion it fits in with many other subjects of Data Mining. The objective of Data Mining is broader and therefore can include Process Mining in its family tree. For example, nobody would think to start understanding Process Mining without starting with the basic Data Mining concepts, therefore they are not independent, but hierarchy connected.

In relation with other Process Modelling features, Process Mining does not simulate processes, it may try to predict the success probability and time of the flow. In fact it may for Process Simulation

tools by creating the starting Process Model based on real events flows. Process Mining tries to, above all, understand the process, and with that understanding it may allow predictions and recommendations (prescriptions), but not the simulation of scenarios and considering variables like event order, resources or the event flow itself. For simulation purposes we may try out free tools like CPN Tools.

Another field that is related to Data Mining and Process Mining is Data Querying. Standard Query Tools like SQL, Oracle, MySQL, MS Access, etc are not process oriented, that is a fact. They are flexible tools that allow the user to manipulate, filter, select, query, join, merge, relate and make complex calculations based on the existing data. It may be used in association with scripts, macros or routines to allow some computation and recursiveness, that is the application of a set of rules or calculations in succession or in other words to introduce successive executions of a specific calculation that may use previous results. This way, even though it is not process oriented, Standard Query Tools are flexible enough for the user to create its own calculations and queries over an Event Log to determine the most common path, bottleneck, average time of execution, success probability of a specific type of case based on its DNA (e.g. most inquiries result in a quotation, but not all quotations result in an order, it is useful to understand if a type of product or price is more likely to be reject by a certain type of customers). The authors, developers and vendors may try to dissociate the tools and explain they are complementary, inferring they could be in the same level, just like against Data Mining. But as explained before at least the ETL process will need some type of Standard Query Tool to support it, therefore the basic event log data is based and extracted and converted based on this type of tools, making Process Mining somewhat and in some cases dependent on this types of tools. The authors may try to put them side by side, but like the example of Data Mining, nobody that works with data, that needs to extract, query or manipulate data, namely to prepare it for some Data Mining or Process Mining analytics, knowledge and insights would start from the analytics skills without having some good understanding of the whole BI or more specifically ETL and Data Warehousing processes and how to work with Standard Query Tools, that is, how to execute data queries.

In one occasion Process Mining would be able to escape this dependence, namely when the source of the Event Log is already oriented for Process Mining and produces Event Log bases on the requirements of the Process Mining tool. But in most cases this won’t happen, or at least not by default, only when the source system programmers develop the audit log, document flow, event log or transaction history in such a way it fits the Process Mining tool you could at some level ignore the

usage of query tools. When this does not happen there is no way to escape the Standard Query Tools, therefore Process Mining requires knowledge and understanding of this type of tools before starting the mining process (analytics). The user would need first to acquire data querying and manipulation skills before being introduced in Process Mining. If not he would find the task of reviewing results impossible and frustrating.

At some point in the paper we may also conclude that most of the features, besides the process visualization, could be fulfilled with Standard Query Tools and some basic programming. Most of the business requirements and needs can be fulfilled without a dedicated Process Mining tool and that could be a reason why this tools are not widespread since the concepts started to takeoff in 1998, more than 15 years ago.

3.5. Challenges and Opportunities

Process Mining is an important technique and tool for modern organizations that want to handle and control non-trivial operational processes. On the one hand, there is an incredible growth of event data, what is sometimes called “big data”. On the other hand, processes and information need to be aligned perfectly in order to meet requirements related to compliance, efficiency, and service levels. Despite the applicability of process mining there are still major challenges that need to be addressed. These challenges illustrate that process mining is still a rising and developing discipline.

The increasing volume of event data provides both an opportunity and a challenge for process mining, since existing process mining techniques have problems dealing with large event logs referring to many different activities. Also, many of the existing process mining algorithms cannot deal with concurrency. Other typical problems are the existence of duplicate activities, hidden activities, non-free-choice constructs and others. On top of this real-life logs contain noise (for example, exceptions or incorrectly logged events) and are typically incomplete, i.e. the event logs contain only a fragment of all possible behaviors. Since the event logs are the major starting point for Process Mining, the development of standardized logs and creation of a systematic approach for event logging from the behalf of the software developers and researchers is the first biggest challenge in the next years for this area.

Some of the other challenges in this new emerging subject are distributing process mining problems to cope with big data, on-the-fly process mining for operational support, dealing with concept drift (i.e. process evolution and change through time), cross-organizational and comparative process mining, context aware process mining and sponsor and guarantying support for the process of process mining.

Of course, the first issue an organization will find to implement and perform Process Mining is capturing, transforming and cleaning event data. The information may be distributed over a variety of data sources, the data may be incomplete, not including all elements of a process (containing process gaps), it may contain outliers (i.e. exceptions that deviate from the rule but are so uncommon that they should be ignored for the sake of simplicity and understandability), may contain different levels of granularity across the process, i.e. part of the process may have a lot of detail for compliance reasons, but may lack the necessary information to support management decisions in other parts. In addition it is difficult to deal with complex event logs that hold different characteristics, they may be too big to handle or to small to make trustworthy conclusions.

In a more general view and from the development and research point of view there is a great need for high-quality benchmarks, that is, example data sets and representative quality criteria to compare and improve the various process mining tools and algorithms.

Researchers are also trying to deal with the concept drift predicament. It is expect for the process to change in the middle of the analysis, sometimes resulting from the analysis own insights. In my professional experience, while exploring, understanding and writing business processes management becomes aware of risks and issues that are solved right away (quick-win). Sometimes seasonality, process maturation, technology advancements, system developments and business evolution induce process changes. If the diagnosis takes too much time it may be incorrect and of no use in the end.

A common problem with process modeling subject matter is the representation bias or prejudice for process discovery, i.e. process understanding and visualization. The outcomes need to be aligned with the user's requirements; therefore caution is needed to ensure relevant and useful results from process mining.

Besides managing the representation bias, when building process models one needs to balance four conflicting and competing forces of process modeling. Fitness that is to say the ability to explain the observed behavior. Precision by avoiding underfitting and being able to clarify the relevant details and insights of a process. Generalization by avoiding overfitting and being able to generalize. Finally simplicity, respecting Occam's Razor principle and selecting the simplest possible solution. Most of the times there is a trade-off between this quality criteria's or dimensions and the ambition is to achieve the best score in all of them.

Looking at the challenges placed by current business relations, we can see that today's organizations are more and more connected. For example on time delivery imply that organization's work together in the same process instance, sharing information about safety stocks, product need's, delivery lead

times, promise dates, production planning, product shipping and invoicing. Procurement, collection, production, accounting and many other services may be externalized and shared between organizations that may or may not use the same systems and infrastructure, or the same methods and rules at all. Sharing a process between two or more companies is clearly a big challenge for process mining, as it is for other are like auditing and performance improvement.

But even if cross-organizational processes are non-relevant, ignored or solved, knowing Process Mining is not restricted to offline process analysis, it should be able to support online operational management, particularly being able to detect deviations and issues, predict outcomes and recommending solutions and alternatives.

One tool will not be enough, and to support operations and management decisions it is important to combine the automated process mining techniques with other areas like simulation, visual analytics, data mining and optimization methods to obtain more business insights.

In Process Mining, like in all other management tools, that sometimes only specialists can apply, explain and understand, there is a great need to improve the usability and understandability of process mining techniques and tools for non-experts, by hiding the sophisticated and complex, sometimes scaring, algorithms and mathematics behind good-looking and user-friendly interfaces that allow for the configuration and parameterization, suggest suitable analysis and review and help understand the results, using suitable representation and producing reliable results. This is the only way to have the company's administration or management roles to sponsor the implementation of Process Mining in tools and techniques in their organizations.

4. Theory

4.1. Introduction

In this chapter we will cover the theory fundamentals. We will deep dive on the definition of Process Mining, look at its goals and objectives, tools and features, deployment and data challenges, and different modelling alternatives.

4.2. Advanced Definition

The best way to describe something is to put the narrative in the form of questions. To better understand what is Process Mining, how it works, what is a process and how it is determined let us try to answer the fundamental questions that follow.

1. What is Process Mining?

   It is a scientific subject, mostly developed and studied by computer scientists, that studies the extraction of knowledge from event logs (recorded activities) available in the business systems for process discovery, monitoring and improvement.

2. How does Process Mining fundamentally work?

   That is, how for example are petri nets and other process representations built based on Event Logs? Process Mining uses the rules and concepts of Data Mining, namely association rules based on event and event sequence frequencies. This allows to ignore events with low occurrence rate and determine the relevant and even the strongest connection between events. In Process Mining tools, several packages, plug-ins or functions may be developed to help the user determine relationships and clean up the logs for once, add more data details to the discovered process model or execute different process model analysis (e.g. control-flow versus social or organizational).

3. What is a process?

   A process is a trace of events with a beginning and an end, basically a sequence of tasks with a defined objective. The trace between two events is determined by the numbers of times an event is preceded by another. The dependency type is determined by the number of times one events precedes another minus the times the other event precedes the one event divided by the total times each event is related.

4. What does it mean to mine a process?

   Process Mining just like Data Mining drives the results and models based on data, in this case event logs or audit logs. It basically mines the data to find insights and knowledge. It uses concepts like frequency, distance, correlation, association and clustering just like Data

Mining. But it focuses on business / organizational processes and social networks. It facilitates Process Modelling by allowing an unbiased discovery of the "real" process model based on real facts. The first real contains double quotes, because everything is a model and the model does not exist, even if based on facts, because it depends on the interpretation of the facts and also on the collection of the whole universe of related and relevant facts and their relationships.

5. Therefore, is Process Mining business analytics?

   Process Mining should belong to the business analytics are of Business Intelligence. Process Mining is Business Analytics. Business Intelligence is a big and sometimes ambiguous business jargon. It comprises the collection and extraction of data from different data sources, transformation or conversion and uploading into a data warehouse, the construction of different data marts, info providers and cubes (depending on the terminology or technology used), that allow business analytics. Business analytics can be simply descriptive, that is simply reporting the facts and past events. It can also be prescriptive and predictive, using OLAP and other exploring tools it should be possible to extract knowledge, that is, relevant business insights to support business decisions, allowing the business to navigate through the unknown. Data Mining and Process Mining specifically have a great role in this component of Business Intelligence. Their method of inference and prediction allow to better understand and control unknown business variables and recommend potential actions based on simulations, for example. Data Mining is also useful for data cleaning and enrichment in the transformation / conversion step, but that is another topic.

6. Does it support operations?

   Process Mining supports Operational and Business Support. When executing, operating and driving the business we may become too focused on what is next to us, therefore blindsided and vulnerable. Process Mining desires to improve that and help navigate the business, allowing the manager and conductor of the business to see the road ahead and drive the business with more certainty, efficiency and effectiveness, that is, to get to destination using the shortest and fasted path without a great risk of failing.

7. Reality is too complex to be controlled. How can we manage this?

   Every businessman has an idea of what he wants and how to get it. When confronted with reality it finds that it is really more difficult to achieve its goals and sometimes the process results in different outcomes. There is a learning process, feedback is obtained, changes are

made, and the results hopefully improve. New processes, new systems, new methods, new structures, new procedures and developments emerge to ensure the desired outcomes. But then reality pulls a prank on the businessman changing the path, changing the target, driving other competitors to fight for the same goal. Change, innovation, evolution, progression, development, it is impossible to control this variables. The businessman needs to continually adjust the route and fight of opponents at the same time. Sometimes the businessman even changes the goal, the perspective or approach, the way things should be done to ensure the concretization of the main objectives. Well, reality is just too complex to fit in a box and to fully grasp in its whole. It changes frequently and is too big and complex to capture and interpret based on real facts. You could start right now studying it, but it would be very different when you've finished to cover it all. Therefore we use models, simplifications and abstractions of reality, less time consuming and easier to remember, to update and put in practice. We try to make things simpler, to find the biggest tendencies, the best links, correlations and associations and figure how to act to achieve the desired goals in each context. Since we won't know everything, we cannot avoid our interpretation bias nor changes, our models lose their value through time, they have a shelf life, an expiration date. We need to update them to maintain their value. Models can be easily improved, adjusted and updated to fit most of the common and most important behaviors. Even this is hard, time consuming, expensive and tiring process to whom it is assigned to if the right tools are not available. Some people when confronted with change may try to resist and fight against it. But the entropy principles tell us that nothing will remain the same once change is introduced to a system, no matter what you try to do. You may try to freeze hot water with cold water, in return you'll have warm water. People want different things, they drive their lives in different ways to achieve different goals, with Process Mining perhaps we can improve this learning process, and continuously maintain our models of reality, simplifications and abstractions, more accurate, timely, relevant, complete and valid. With this said, the businessman has a new tool to help him in his endeavor.

4.3. Goals and Objectives

As referred before Process Mining aims to improve the extraction of knowledge from event logs by providing techniques and tools for the discovering process, organizational, social, and performance information from event logs. It is a method of distilling a structured process description from a set of real executions.

Process Mining tools aim to discover, check, predict and recommend based on objective and systematic data. Their techniques attempt to extract knowledge from event logs recorded by an information system and automate process discovery, that is, extracting process models from an event log, execute conformance checking, namely monitoring deviations by comparing a model with a log, social network/organizational mining, automated construction of simulation models, case prediction, and history-based recommendations.

Every organization will support some kind of transactional information, even if the process management maturity level is in its lower level (ad-hoc or informal processes). This information combined with some mining techniques can be used to get more insight about the company's process. It is always possible to create event logs with event or transactional data. On the other hand, this event logs can be used to construct a process specification, which adequately models the behavior registered in the systems.

Process Mining has basically to categories of features based on the time perspective: Backward-Looking or Forward-Looking. Backward-looking seeks to understand the present by looking into the past events. Looking at the known flow of historical events allows us to identify the current process model and detect deviations in past events (offline or post-mortem) or even occurring events (online or pre-mortem). Looking at known events we may execute Cartography, namely: discovery, by modelling an existing process that is, learning the process model, enhance, by repairing and extending and existing process model and diagnose, by verifying and checking the process model. Looking back can also allow Auditing, namely to detect (pre-mortem), i.e. identify deviations online and generate alerts (monitor execution), to check (post-mortem), i.e. identify deviations offline and quantity compliance (compliance check) and to compare, i.e. highlight differences and commonalities between two models (a de jure - the law vs a de fact - the fact).

On the other hand, by looking forward we are able to Navigate through the present by predicting future events, guiding the operator or user by predicting the future outcomes and alternatives (e.g. success probability of an action) and suggest suitable actions (e.g. best options or alternatives available). This can also be called Operational Support as referred before. This category of features include exploration, to visualize a running case and compare to similar recent case (help trigger actions), prediction, to calculate the flow time and success probability (the resulting performance of the case) and recommendation, to guide the operator by predicting and ranking next steps aiming to the most promising outcome (minimize cost and time - efficient, and maximize success probability - effective).

So, Process Mining should be the organizations GPS system because it aims to help organizations navigate the business. That is, understand all the alternative routes, recommend the best path, predict the time of arrival, give on time and online metrics like distance to go and speed, and in the case of a road block an alternative way out.

But to discover, predict and recommend is only a means to an end. It all comes down to continuous Process Improvement, call what you may, Cost Reduction, Revenue Growth, Business Restructuring, Process Optimization, among others, the objective is the same, increase productivity, reduce costs, enhance service level including service quality and ultimately improve the financial outcomes, the shareholder value, return on equity, the business profit, basically to make money and continue doing it in a sustainable way. To do this we need first to understand the business value chain, that is, how it creates or adds value (at least perceived). In an industrial type organization, for example, this starts from the design and development of products, selection of suppliers, purchasing raw materials and equipment, hiring labor, stocking and balancing demand requirements, transforming the raw materials based on specific and unique or common, standard and scalable processes and recipes, marketing the products, making them available and selling them of the shelves and finally collecting the bill. Each segment of the value chain will have their own operational rules and context, they will have internal and external requirements and will be supported by different systems or modules, functions, business knowledge, technical skills and processes. When an organization is already operational you may want to improve the organizational structure, methods, processes and systems. You may want to start from scratch or to understand the status quo, usually called the as-is, identify the pain points, redesign and implement solutions / improvements. The first option is hardly executable, because that may imply stoppage or even parallel operations, and also there are always lessons learned from how things are run know and there is no need to reinvent the wheel in most cases. To understand the as-is operations you may observe and document events, you may ask someone to describe them in an interview, workshop or brainstorm session, you may also re-execute the process, that is, do a walkthrough and experience the flow, you may even do them all. It sounds good, but it takes time and is restricted your individual and the interlocutors experience and worst subjective interpretation. To cut it short, you may simply collect the companies event records and reconstruct the past event in an automated and friendly way. Well that is exactly the final objective of Process Mining. You may want then to check the current process against organizational guidelines and also obtain some performance metrics to understand where the biggest pain points in the process are and therefore find the flaws and improvement opportunities based on the real execution. Later on

you may want to make changes and monitor the results. These are other topics that Process Mining tries to cover. It is easy to understand how Process Mining may support Process Improvement in every stage from Diagnosis, Design, Implement and Monitor (observe and check progress).

We've witnessed a big expansion of technology and information systems capabilities as referred in Moore's Law, by which the number of components in integrated circuits would double every year. In the last fifteen year we've seen a lot of progress in the information technology, making more and more the digital world aligned with the real world. That is, more and more business transactions and social activities are recorded in ERP systems and Social Networks, making it easier to apply data mining and business intelligence techniques to extract meaningful knowledge from all the data available.

Like in all known social networks and business organizations increased their digital universe recording almost every transaction in workflows, ERP's and every machine making it possible to record and analyze events. Recorded events therefore can be a powerful source of knowledge to understand, control, monitor and improve its geneses, the reality (World) that the events represent in a never ending circle of continuous improvement like shown in the next diagram.

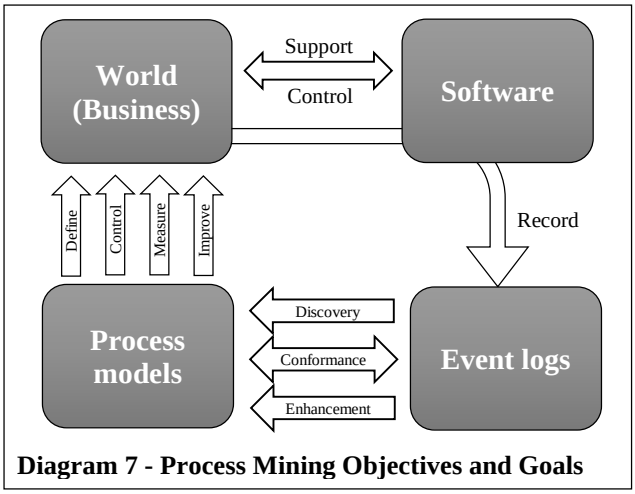


Diagram 7 - Process Mining Objectives and Goals

The biggest ambition of Process Mining is to exploit event data in a meaningful way, for example, to provide insights, identify bottlenecks, anticipate problems, record policy violations, recommend countermeasures, and streamline processes. That is to say untap the reservoir of knowledge already maintained by the organizations about the way people conduct every-day business transactions.

The practical relevance of Process Mining is increasing as more and more event data become available. Process Mining techniques aim to discover, monitor and improve real processes by extracting knowledge from event logs, given that events logged by some information systems can be used to extract information about activities and their causal relationships. The two major Process

Mining tasks, like described in the previous sections, are process discovery, i.e. learning a process model from example behavior recorded in an event log and conformance checking, that is, diagnosing and quantifying discrepancies between observed behavior and modeled behavior. The increasing volume of event data provided both opportunities and challenges for process mining.

The topic of process mining has attracted the attention of both researchers and tool vendors in the Business Process Management (BPM) and Business Intelligence (BI) vectors. But BPM as described before is model-centric and doesn't construct models based on event data and, one the other hand BI may focus on data and includes many buzz words in its umbrella but mostly can be summarized by simple management reporting and dashboard tools.

A lot of technologies can be included in BI. Business Activity Management (BAM) that enables real-time monitoring of business processes. Complex Event Processing (CEP) processes large amounts of event to help the management monitor, guide and optimize the business on the fly. Corporate Performance Management (CPM) measures the performance of a process or organization. Also some management frameworks and approaches can be included like Business Process Improvement (BPI), Continuous Process Improvement (CPI), Six Sigma, Total Quality Management (TQM) and many others. In all this approaches, frameworks and disciplines the business process is put in spotlight, put under scrupulous analysis and meticulously reviewed to identify issues and improvement opportunities. Basically Process Mining should be an enabling technology and desires to be a requirement for all these process management frameworks. Ultimately Process Mining serves the same purposes as all the BI tools and frameworks (management approaches) that aim to improve operational performance, for example reduce lead times, production defects and customer complaints.

Besides performance improvement, in today's business organizations a lot of attention is being given to corporate governance, risk management and compliance (financial, health, security, working conditions, quality, diverse certifications and others). Recent scandals and publicly known corruption cases and bankruptcies can destroy a business organization from one day to another. A lot of international and national legislation like Sarbanes-Oxley Act (SOX) and Basel II Accord and other private certification demonstrate the focus on compliance issues. Process Mining also offers the means to a more scrupulous process analysis for compliance check and audit to ascertain completeness, accuracy, validity and restricted access of the business transactions. This allows building more reliable information for business assurance of the organization's core processes.

Also in finance assurance, financial auditors usually review journal entries, i.e. logging of transactions into accounting journal items, but focus their work on analysis techniques that are purely data oriented, not process oriented. This means they look at each set or limited group of data sets that cover partial business processes rather that performance a complete business assurance review of the end-to-end business process, to cover all of the organization's core processes.

In abbreviation, the Process Mining techniques extract knowledge from event logs or business transactions recorded in the systems and other formats by discovering the process models, i.e. designing the process flowchart, monitor and improve business activities in the process model for fraud, cost and performance issues review for example. But in this case, not like in BPM and BI techniques the goal is to improve real processes based on recorded actions, not assumed or theorized processes.

Process Mining techniques should be supported and support other methods, including the described buzz words, ranging from Data Mining, Business Analysis, Visual Analytics, Business Process Management, Business Process Improvement and all the Business Intelligence techniques.

In more detail and practical terms, three operational support activities can be identified: detect, predict, and recommend. The moment a case deviates from the predefined process, this can be detected and the system can generate an alert. Historical data can be used to build predictive models. These can be used to guide running process instances. For example, it is possible to predict the remaining processing time of a case. Based on such predictions, one can also build recommending and suggestion system that proposes particular actions to reduce costs or shorten the flow time. Predictions and recommendations based on models learned using historic information can therefore be used to influence running cases. Similar forms of decision support can be used to adjust processes and to guide process (re)configuration.

Process mining techniques can also be used to learn a simulation model based on historical data. Subsequently, the simulation model can be used to provide operational support. Because of the close connection between event log and model, the model can be used to replay history and one can start simulations from the current state thus providing a “fast forward button” into the future based on live data.

It is therefore desirable to combine process mining with visual analytics. Visual analytics combines automated analysis with interactive visualizations for a better understanding of large and complex data sets. Visual analytics exploits the amazing capabilities of humans to see patterns in unstructured

data. By combining automated process mining techniques with interactive visual analytics, it is possible to extract more insights from event data.

With that said, processes are everywhere. Organizations have business processes to manufacture products, provide services, purchase goods, handle applications, manage systems and other resources, etc. Additionally, in our daily lives we are intrinsically involved in a multiplicity of processes, when we use our home appliances or when we book a hotel and flight in a website. Although such operational processes are omnipresent, they are at the same time intangible and most of the times ignored, since they are less concrete and more dynamic nature than products, resources or even data. However, more and more information about these processes is captured in the form of event logs by contemporary systems ranging from our machines (e.g. printers, personal computers and medical devices) to enterprise information systems and cloud infrastructures. These events can now be used to make processes visible. Using process mining techniques it is possible to discover processes. This provides the insights necessary to manage, control, and improve processes. Process Mining has been successfully applied in a variety of domains ranging from healthcare and e-business to high-tech systems and auditing.

To recap, the main objective of process mining and building process models is to solve compliance-oriented on one side and performance-oriented questions and problems by focusing on how things happen and come to be. A process is what is between inputs and outputs; it is the middle and intermediary between a trigger and driver, that is, a starting point, for example, a customer complaint and its resolution, closure and end. It explains how inputs come to be outputs, what happens between a cause and a consequence. Process models are maps of different ways and routes for a resolution or closure. Knowing these intrinsic maps and logics will allow us to detect problems, predict results and understand and solve issues before or after they happen.

4.4. Process Mining Tools

What are the Process Mining Tools available and in what categories do they fit?

There are several Process Mining tools already available in the market. Some are free to use and even open source (e.g. ProM) and others are free to try (e.g. Disco). Most of the developments in this area of data mining and process discovery technique is based in Eindhoven, in the Netherlands, were most of the new publications are written and new approaches are developed. Many of the concepts of ProM have been embedded in commercial tools such as Fluxicon's Disco (www.fluxicon.com), Perceptive Process Mining (www.perceptivesoftware.com), Celonis (www.celonis.de), Aris, BPM One (Pallas Athena), Interstage (Fujitsu), Futura Reflect, Comprehend (OpenConnect), Process

Discovery Focus (iontas), Enterpise Visualization Suite (Businesscape) and QPR ProcessAnalyzer (www.qpr.com).

Probably the most well-known and popular process mining tool available is ProM, an open source toolkit developed at Eindhoven University of Technology. ProM is a good choice to explore process mining, because it has consistently been at the forefront of that technology. Most of all, it is free.

The list below contains most of the tools available today, some of which may have been in the meanwhile discontinued:

Table 6 – Process Mining Tools

Name	Category
ProM	Academic
InWoLvE	
Process Miner	
MinSoN	
ExperDiTo	
ServiceMosaic	
Rbminer/Dbminer	
Genet/Petrify	
Aris PPM	Commercial
HP BPI	
ILOG JViews	
Comprehend	
Discovery Analyst	
Flow	
Enterprise Visualization Suite	
Interstage Automated Process Discovery	
OKT Process Mining suite	
Process Discovery Focus	
ProcessAnalyzer	
Reflect|one	
Futura Reflect	
Disco	

4.5. Deployment Challenges

You won't have a short straight answer with Process Mining. Before you can obtain a usable Process Model you need to manipulate parameters, sometimes in the dark, to try and get something that fits the expectations. But the proposition of Process Mining is the possibility of unbiased discovery of the real process model. The reality is you'll need to understand the process previously, identify the supporting systems, configure or create event logs based on the implemented processes and when

executing the discovery algorithm you may need to adjust settings and parameters and try to "correct" the model trying to fit the discovery result with our own model or expectation. In this aspect it goes against what was set to be one of the major advantages of this techniques.

The level of system and process maturity may change from organization to organization and perhaps for most organizations there will be some limitations to try and apply this type of methodology. This is also true for any analytics, the results and very much dependent on the quality and extension of the information available. Being one the highest maturity level and three the lowest, we will try and understand what kind of scenarios we may find in a company regarding process definition and process supporting systems:

1. Explicit process descriptions of the way the work is organized and this description is supported by a process aware information system like, for instance, a Workflow Management System (WFM). The system enforces and guides the user. Options are all predicted and enclosed in the designed process.
2. Explicit process descriptions of the way the work exist, the system supports the process, but the practical way of working can differ considerably from the prescribed way of working because there is some sort of flexibility in the process or system (open doors) that allows for individual decision making (ad-hoc).
3. No or only a very immature process description is available. The existing systems support some business activities, but not the whole business process (non-existing or incomplete).

It seems that most of the companies that would benefit from this type of methods are in the third stage, that is, they have rudimentary informal processes that work based on tribal knowledge and some group sense of how things work and the systems support this way of working in the same rudimentary and sometimes ad-hoc way. This does not mean there is no process in place, just that it is informal and the system those not give full cover to it by consequence. Therefore there will not be data or event logs to cover the process in full. Even if the system is standard and widely implemented, forcing some kind of flow, this is as far as it goes.

One the other hand, organizations that are on the first level of maturity, most probably don’t need Process Mining, basically the process is enforced and most probably the workflow system already gives performance metrics and allows online monitoring. There is no need to discover the process, because the process defined and enforced.

Basically only in the second stage there is available and reliable data and a real benefit in implementing this kind of method. It shortens a little bit the list of companies that verify the criteria and really may benefit from this kind of methodology.

4.6. Data Challenges

No algorithm alone in Process Mining is able to cover all data issues and challenges. Some attack a specific issue, but to truly cover the business needs we need to cover them all. So, Process Mining struggles to deal with several types of complexities, namely:

1. Concurrency (parallelism): two steps working concurrently, together (complementary) or against each other (substitutes);
2. Complex Routing and Complex Control-Flow Constructs: spaghetti/unstructured like processes with one or all of the following aspects like lots of steps, lots of existing combinations, not a clear core or primary trace/flow, lots of loops between steps/activities or to the same activity, explosion of events, i.e. one event results in a variety of new events with the number of new events being almost random and depending on each case. Many algorithms are unable to deal with non-free-choice constructs and complex nested loops;
3. Noise: the existence of bad history (inaccurate records) mixed with good history. Difficult to acknowledge what is an outlier from good history records against what is simply inaccurate history that may deviate our conclusions;
4. Event gaps or silent steps: missing events or records in a case (process instance). Things that are not recorded cannot be discovered;
5. Trace gaps: missing flows / trace (links between events). It is imperative we have a full understanding of the process to really measure and improve it. Some types relevant activities are not recorded at all or are recorded in a way it is not possible to associate or correlate the events;
6. Event correlation: it may be difficult to associate all events to the correct process instance. Each record may not be linked to its original case (e.g. returns, price adjustments) or even worst a record/event may be linked to several cases (e.g. credit note for several orders). In this last case the solution it to increase the level of granularity and looking at the record at a single item level (when possible). This solution increases the complexity and may limit the usage of some algorithms (e.g. Heuristics Miner does not work well with this). Another solution for this last type of cases would be to artificially split the record using the allocation principles of weights just like in costing;

7. Duplication: in the event log it is not possible to distinguish between activities that are logged in a similar way, i.e., there are multiple activities that have the same “footprint” in the log;
8. Underfit or overfit: many algorithms have a tendency to overgeneralize, i.e., the discovered model allows for much more behavior than actually recorded in the log. In some circumstances this may be desirable. However, there seems to be a need to flexibly balance between “overfitting” and “underfitting”.

Process Mining tools allow us to obtain some process insights from event data, but for this to happen some requirements must be filled, namely:

1. It is possible to create event logs with event or transactional data;
2. Each event refers and can be mapped to an activity (i.e. well defined step in the process);
3. Each event refers and can be mapped to a case (i.e. process instance);
4. Each event has a performer, that is, person or system executing or initiating the activity (i.e. origination);
5. Each event has a timestamp and can be ordered.

To fulfill all the requirements we need to meet all the following criteria:

1. Correlation: Events in an event log are grouped per case. This simple requirement can be quite challenging as it requires event correlation, i.e., events need to be related to each other;
2. Timestamps: Events need to be ordered per case. Typical problems: only dates, different clocks, delayed logging.
3. Snapshots: Cases may have a lifetime extending beyond the recorded period, e.g., a case was started before the beginning of the event log. Before starting to model it is necessary to ascertain the best time frame to cover the full process life cycle.
4. Scoping: It is necessary to determine which information is relevant for the defined purpose, that is, to cover all the underlining questions that motivated the implementation of the model.
5. Granularity: The events in the event log must be in the same level of granularity and this level must be aligned with business needs. Most times the event log may contain a different level of granularity than the activities relevant for end users.

Understanding that Process Mining is supported in real business data and knowing that not all events are recorded, may not be recorded in a structured way and may contain errors and inaccuracies (noise), we can conclude that the Process Mining techniques cannot focus only in using the existing data, but they should also cover cleaning and enhancements, like event correlation and case allocation, to be really effective in the real world.

4.7. Modelling Languages

Petri Net is just one of many Process Modelling languages. In management BPMN and UML are usually the excepted model. With that said, there are several Process Modelling languages available, each of the manager, company or author will find their model to be the best, they will highlight the benefit of their version and find holes in all others. But being agnostic about this subject, here is a basic list of the generally used modelling languages, some of which covered by ProM Framework:

1. Petri Net
2. BPMN
3. BPEL
4. UML
5. StateCharts
6. C-Nets
7. Heuristic Nets

5. Applications

5.1. Introduction

This chapter captures the application definitions of Process Modelling. It figures the features, business requirements,

5.2. Features

Process Mining features include representing the process in a diagram, a Petri Net and any other process modelling language, monitoring the execution of the process, predict results of the process and compare process models, basically executing a conformance checking.

The baseline requirements for a Process Mining tool should be aligned with the implementation objectives. That is, the solution should help the business:

1. Gather insight about the how a business process is defined, understand how it works and what are the limitations and issues;
2. Support the discussion and brainstorming about the business process. It should create a common baseline, acceptable and concrete to start the discussion;
3. Document and instruct, that is, enhance transparency and knowledge sharing, informing all relevant resources of their tasks, responsibilities and the impact of their actions;
4. Audit and verify the process, review and look out for errors and mistakes in the design, the systems or even in individual actions;
5. Analyze process performance and drive improvements, by understanding the process and also being able to determine the key pain points the solution should enable the determination of the key change drivers that improve the service levels;
6. Animate and roll play models, allowing users to explore and play out the process bringing new insights and feedback about the process;
7. Process design for system or process specification, that is, the model can serve as a binding contract or list of requirements for system setup, on the one hand, and as the baseline for process change.

Taking into account this basic requirements, in a Process Mining tool you should be able to find the following features, they the translation of the business requirements for this type of solution:

1. Process Discovery – Find how the process is:
   a. Draw diagram with all the paths for presentation and comprehension purposes
   b. Determine process metrics (e.g. time, frequency, and count metrics)
   c. Determine most used path

d. Allow to adjusts the number of events represented (e.g. based on frequency)

e. Identify potential bottlenecks

f. Identify potential loops or redundancies in the process

g. Determine the fastest and slowest path to complete the process

2. Process Predictions and Recommendations - Monitor the process execution:
   a. Determine process execution metrics (e.g. time, frequency, and count metrics)
   b. Compare process execution metrics with standard / target metrics
   c. Create process alerts and flags if thresholds have been compromised
   d. Potentially recommend actions and alternative paths to finish the process
3. Conformance Checking - Review process model and corporate rules:
   a. Get objective information on whether it is actually followed as prescribed
   b. Determine gaps and issues in process (model against reality)

Process Mining does not cover everything, for example Process Simulation, i.e. understand how process would work (prediction) based on predetermined criteria and scenarios, and determine end result and metrics of process changing variables (e.g. adding resources, changing event relationships) and perform all other feature identified in the process discovery but for the simulated version.

All business analytics are driven by results, metrics and indicators, they should produce outputs that may support and bring some business insight to proactively improve it. With that said, Process Mining performance metrics and outputs should include, among others, the following:

1. Process bottlenecks
2. Most frequent path
3. Shortest path
4. Longest path
5. Processing time (of event)
6. Flow time (between events)
7. Gaps and missing links
8. Outliers and weird paths
9. Deviations highlights
10. Conformance measures for conformance review
11. Execution alerts to support monitoring tasks
12. Process automated representation for visual validation / check
13. Tasks performers

14. Segregation of duties conflicts

5.3. Types of Process Mining Techniques

In this section the three types of Process Mining techniques will be explained. Their relationship is shown in the next diagram, where we see the Event Log being the major input for all types of Process Mining shown in the white oval shapes (Discovery, Conformance and Enhancement). For Discovery we need only the Event Log to create a Process Model. With both the Event Log and the Process Model we can do conformance checks (diagnosis) or enhancement and improve the current process model.

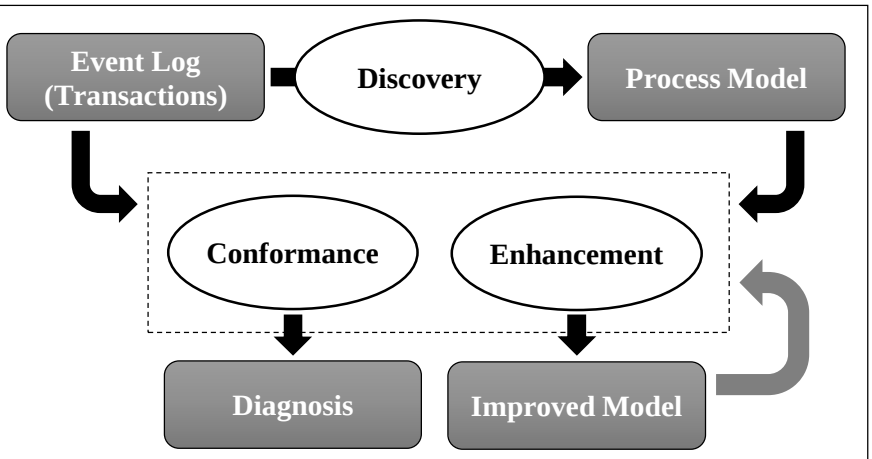


Diagram 8 - Process Mining Types

Following the diagram above, the first type of Process Mining techniques or types is the process model discovery. This technique takes an event log and produces a model without using any a-priori information. Process discovery is the most important process mining technique, since it allows the discovery of real processes based on example recorded activities in event logs. This technique takes an event log (input) and produces a process model (output). The resulting discovered process is usually structured in a process model like a Petri Net, BPMN, EPC, or UML activity diagram. Social networks could be other ways of describing a discovered process. The Process Mining that will be used in this case study (ProM) structures business process models using a Petri Net notation. The Petri Net is one of the simplest and most practical ways of representing a process, containing only three basic elements (transitions, places and arcs). An example of a Petri Net is shown in the next diagram.

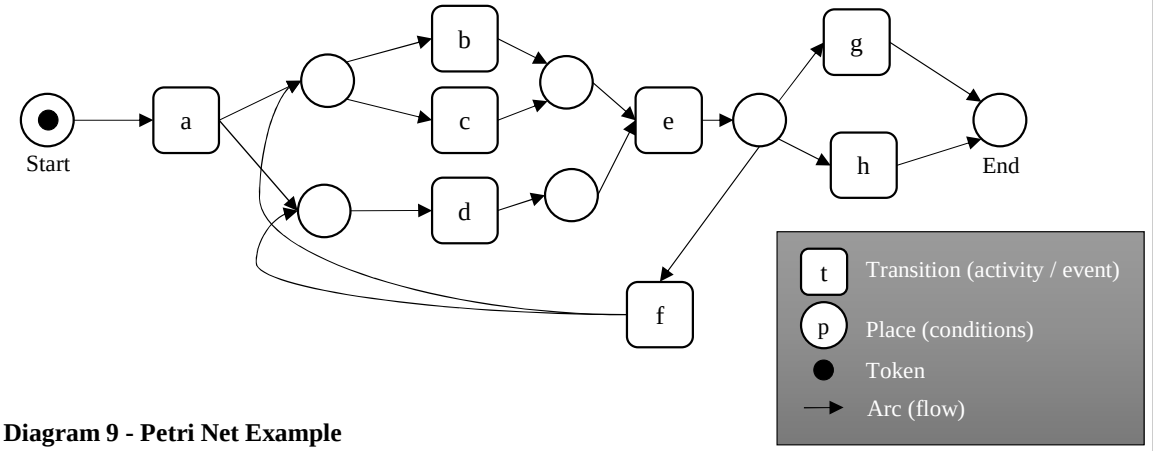


Diagram 9 - Petri Net Example

Each square is a transition or activity put in to action by specific resources, the circles are places or conditions that determine when an activity must take place or what is the follow-up of a activity and the arrows are the arcs that connect transitions and places representing the flow or direction of a process.

Secondly we have the conformance technique, where an already existing process model is compared with an event log for the same process. This technique can be used to check if the real process, recorded in an event log, is conformant with a model or if the model is conformant with the process. In simple terms it aims to measure the alignment between model and reality. This technique takes an event log and pre-existing process model (input) and makes a diagnostic about the process, identifying differences and commonalities between both inputs (event log and process model).

Finally, enhancement aims to extend or improve an already existing process model using information about the actual process recorded in some event log, that is change or extend the a-priori model. The extension of the model could mean the addition of an attribute like task responsible (user) or timestamps to allow the calculation of service levels and identification of bottlenecks. Like conformance checking, enhancement needs an event log and a-priori process model (input) to improve or extend the process model (output).

5.4. Perspectives

Process Mining has three main perspectives:

1. Process: the process perspective focuses on the control-flow, i.e., the ordering of activities. The goal of mining this perspective is to find a good characterization of all possible paths, expressed in terms of, for instance, a Petri Net.
2. Organizational: the organizational perspective focuses on the originator information, i.e., which performers are involved in performing the activities and how they are related. The goal is to

either structure the organization by classifying people in terms of roles and organizational units or to show relations between individual performers, building a social network. It is possible to derive relations between performers of activities, thus resulting in a sociogram. The main questions this perspective tries to answer are: Who is doing what? Who is working with who? What are the roles in an organization?

3. Data: the data perspective focuses on properties of cases. Cases can be characterized by their path (control-flow) in the process or by the originators (organization) working on a case. However, cases can also be characterized by the values of the corresponding data elements. The data perspective, also referred in some literature as case perspective, tries to establish relations between the various properties of a case. The main questions this perspective tries to answer are: How does data flow from one task to another? What data is influencing decisions? What are the data-driven business rules?

This perspectives also bring some light on the meaning of Process Mining and on the understanding of the goals and objectives. Sometimes process modeling and Process Mining only associated with the construction of flowcharts and business process diagrams, also known as control-flows. But Process Mining is able to cover all sort of different perspectives as referred before, not only control-flow (i.e. ordering of activities) by finding a good characterization of all possible paths, but also identifying the resources involved and their relationships by classifying resources in terms of roles and organizational units (i.e. structuring the organization) or showing the social network. This perspective is very important as it allows the organization to understand the different roles played in a specific process and identify their relationships. The roles for a discovered process, for example regarding customer order handling may include the following:

Diagram 10 - Organizational Perspective

On the other hand, the data perspective can focus for example on time (i.e. timing and frequency of event) by discovering bottlenecks, measuring service levels, monitor resource utilization or predict remaining processing time for running cases allowing to characterize cases based on the values of different related data elements.

5.5. Business Requirements

Before deciding to implement Process Mining techniques the analyst should first try to understand the business driver for this type of application. This means answering the business questions a business director or officer would put. This means, the Process Mining requirements are those that determine its existence, what justifies its implementation. So, what drives the Process Mining implementation and investigation? Here are some of the questions the business requires Process Mining applications to answer:

1. Discover:
   What happened?
   How do things work around here?
   What is the way or path?
2. Check:
   Are we executing according to the agreed guidelines, procedures and rules?
   Are we meeting the requirements?
   Are we in conformance with the law or agreed principles?
3. Review:
   What are we doing wrong?
   Where are we failing or stopping?
   What are the bottlenecks?
   Where are we failing?
   Are there redundant or even unnecessary tasks?
4. Predict:
   What do we expect will happen?
   At what time will the case be closed?
   Will it be late?
   Will the outcome be good or disappointing?
5. Improve:
   How can we change the outcome?

How can we improve the probability of success?

Can we improve execution performance and efficiency?

Where should we attack first?

Usually process modelling is more an art than a science. Answering this questions is much more than running algorithms on that and returning metrics. Usually process modelling depends on who is describing, who is documenting, and their knowledge and bias of the real process. Process mining intents to create the process model based on real life events, that is, real business transactions recorded in the systems, from order entry, to credit review, to production and product shipping and invoicing. A lot of data is stored in the systems for financial or management purposes. This information allows us to create real process models unbiased of our previous knowledge of reality. It tells us what the process is currently like based on what has happened, that is real events. Usually there are several problems with the standard process modelling approaches, namely:

1. Describing an idealized version of reality;
2. Over simplifying reality;
3. Inability or incapacity to capture all the details and human behavior;
4. Incapacity to drill down and up different degrees of granularity (from macro to micro).

But, Process Mining should cover all this inabilities or limitations.

5.6. Process Mining Algorithms

If you start up ProM for the first time to try out some Process Mining techniques on your data, the number of available plugins (almost 300) can be daunting. Just to mine a process model you can range up to 40 different plugins with the same goal of process discovery, from the theoretical Alpha Algorithm, to the Fuzzy Miner, Heuristics Miner, Genetic Miner, and in each of them having more than one variant.

The several packages and associated plug-ins available for ProM provide the user different features. It would be good to have some kind of simple summary and comprehensive view of their main functions, their level (basic or advanced), relationships and connections between them, a debrief on how they work, what are the inputs and output, etc. Today the user may need to read a thesis, working paper or equivalent for each and try to make some sense and build the big picture out of it before making a decision. Or just try them all and make some sense out of it, if this is even possible. In the next chapters we will bring some light on this topic, namely try to explain the main plugins. In this section we will give an overview on the available plugins in ProM. To make this less theoretical,

here is a small list of things you may find in ProM, in grey some of the ones we will describe or mention further one:

Table 7 - ProM Plugin Short List

Type	Function	Name
Import	Import Event Log	ProM Log Files (Disk-buffered by MapDB)
Import	Import Event Log	ProM Log Files (Naive)
Import	Import Event Log	ProM Log Files (Lightweight and Sequential IDs)
Quality	Prepare Event Log	Add missing Events
Quality	Prepare Event Log	Add identities to log
Quality	Prepare Event Log	Add artificial events
Quality	Prepare Event Log	Add noise to log filter
Quality	Prepare Event Log	Calculate log meta data
Quality	Conformance Check	Check compliance of a log
Quality	Prepare Event Log	Concept drift
Quality	Prepare Event Log	Enrich log for temporal compliance checking
Quality	Prepare Event Log	Estimate Completeness
Quality	Prepare Event Log	Filter log by attributes
Quality	Prepare Event Log	Filter log on event attribute values
Quality	Prepare Event Log	Filter log on trace attribute values
Quality	Prepare Event Log	Filter log using prefix-closed language (PCL)
Quality	Prepare Event Log	Filter log using simple heuristics
Quality	Prepare Event Log	Fix attributes
Quality	Prepare Event Log	Fix timestamps
Quality	Prepare Event Log	Remove extensions from log
Quality	Prepare Event Log	Rename/merge events
Quality	Prepare Event Log	Remove events from log traces
Quality	Prepare Event Log	Prepare Durations for Correlation Testing
Conversion	Prepare Event Log	Convert Cheetag Log
Conversion	Prepare Event Log	Convert log to generator (tree-based)
Conversion	Prepare Event Log	Convert log to generators (sequential)
Discovery	Process Discovery	Declare Maps Miner
Discovery	Process Discovery	Declare Maps Miner no Hierarchy
Discovery	Process Discovery	Declare Maps Miner no Reductions
Discovery	Process Discovery	Declare Maps Miner no Transitive
Discovery	Process Discovery	Data-aware Declare Miner
Discovery	Process Discovery	Discover matrix
Discovery	Process Discovery	Discover with ILP using Decomposition
Discovery	Process Discovery	Mine for a Petri Net using ILP
Discovery	Process Discovery	Guide Tree Miner (plugin)

Type	Function	Name
Discovery	Process Discovery	Mine a Process Tree with ETMd
Discovery	Process Discovery	Mine for a Causal Net using Heuristics Miner
Discovery	Process Discovery	Mine for a Fuzzy Model
Discovery	Process Discovery	Mine for a Handover-of-Work Social Network
Discovery	Process Discovery	Mine for a Heuristics Net using Heuristics Miner
Discovery	Social Network	Mine for a Reassignment Social Network
Discovery	Social Network	Mine for a Similar-Task Social Network
Discovery	Social Network	Mine for a Subcontracting Social Network
Discovery	Social Network	Mine for a Working-Together Social Network
Discovery	Process Discovery	Mine Transition System
Discovery	Process Discovery	Select Best Fuzzy Instance
Discovery	Process Discovery	Fuzzy Animation
Discovery	Process Discovery	Convert Heuristics Net into Petri Net
Discovery	Process Discovery	Convert Heuristics Net into Flexible Model
Discovery	Process Discovery	Get Model from Pair
Discovery	Process Discovery	Remove edge points
Discovery	Process Discovery	Unpack Aggregate Type
Discovery	Process Discovery	Mine a Petri Net using Flower Miner
Predict	Process Monitoring	Operational Support Client
Predict	Process Monitoring	Start not another workflow language system
Review	Performance Analysis	Perform Predictions of Business Process Features
Review	Performance Analysis	PN Performance Analysis (interactive)
Review	Performance Analysis	PN Performance Analysis (batch)
Design	Process Design	Start Process Tree Editor
Design	Process Design	Start Process Tree Editor Beginner
Design	Process Design	Test Driver - PLG Process Generator

As we can see there are dozens and even hundreds of plugins, but even more complex if to determine their function in the whole process, and redundantly execute process mining on process mining techniques. You would probability find the path between the different plugins, like the ones that will be followed in the case study:

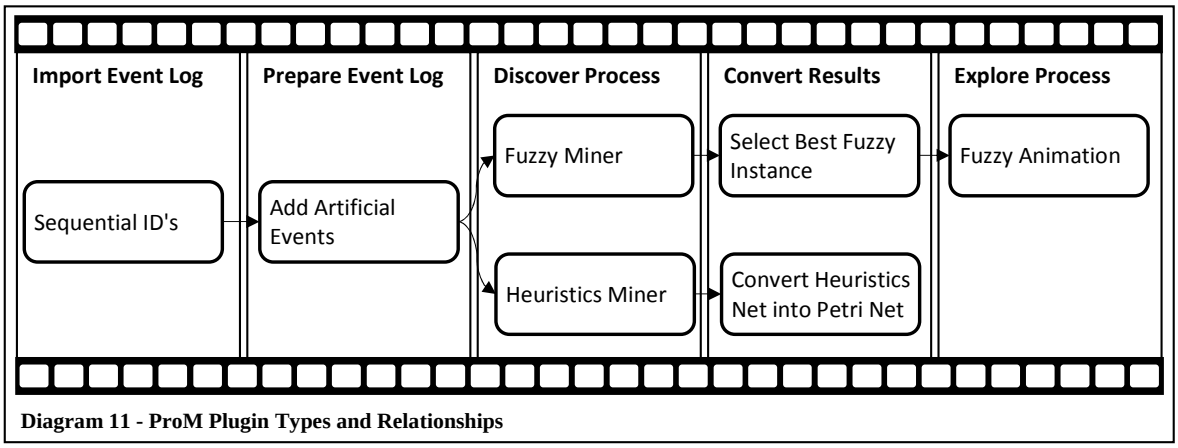


Diagram 11 - ProM Plugin Types and Relationships

A single package in ProM may contain several plug-ins and may depend on other packages or more specifically plugins. For example the Heuristics Miner depends on Flex, Log, Petri Nets and BPMN packages and contains at least 5 plug-ins:

1. Visualize Heuristic Net with Semantics Split/Join Points
2. Flexible Heuristics Miner
3. Convert Heuristics net into Flexible model
4. Convert Heuristics net into Petri net
5. Visualize Heuristic Net with Annotations

It is not very easy to apply this techniques. The first time the user may become overwhelmed with so many options and won't understand how to start or find valuable the time to figure it out. So, what are the algorithm, package or plug-in types, groups or families based on their scope, their features and their outputs?

1. Control-flow discovery
    a. Alpha Algorithm
    b. Heuristics Miner
    c. Fuzzy miner (non-structured processes)
    c. Genetic Algorithm
2. Organizational perspective
    a. Social network miner
    b. Organizational miner
3. Data perspective
    a. Decision miner

In this chapter we will cover at least three of the most relevant plugins regarding Process Mining Discovery Techniques, namely: Alpha Algorithm, Heuristics Miner and Fuzzy Miner. There are several technique types namely algorithmic, genetic or region based as shown in the diagram below:

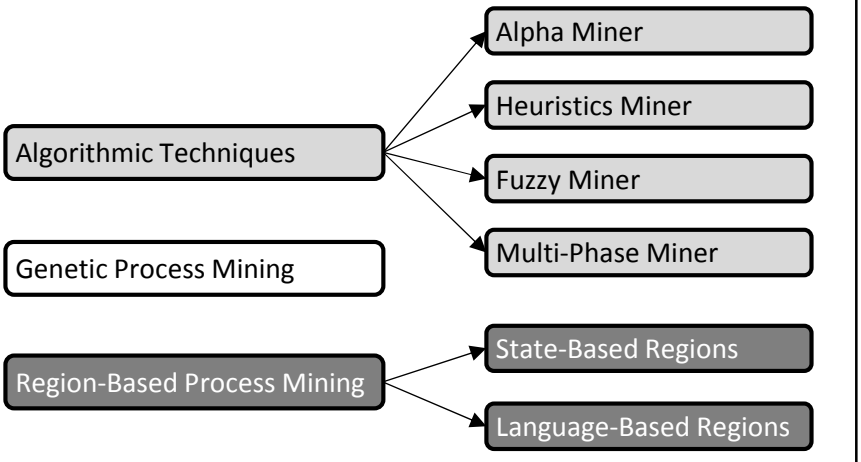


Diagram 12 - Process Discovery Techniques

For example the Genetic Process Mining technique will not be covered in this paper but it follows a data flow like this:

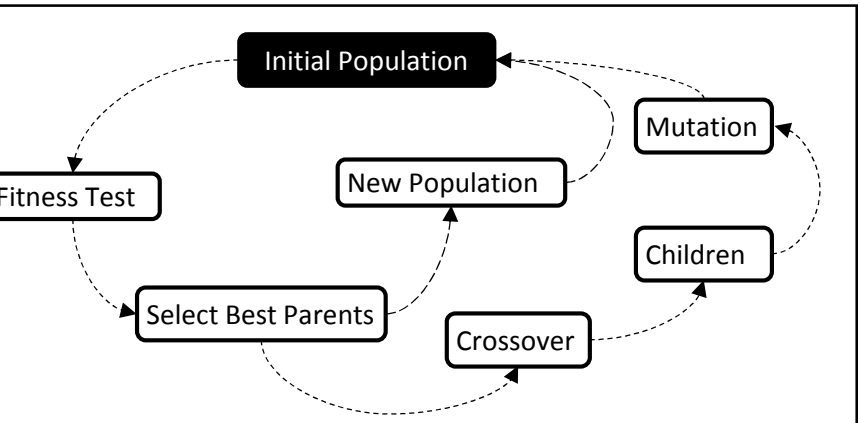


Diagram 13 - Process Discovery Techniques: Genetic Mining

Another note about the Genetic Process Mining is that it:

1. Requires a lot of computing power;
2. Deals with noise, infrequent behavior, duplicate tasks, invisible tasks, etc;
3. Allows incremental improvement and combinations with other approaches (e.g. heuristics post-optimization).

Let us now cover three of the main algorithmic techniques for Process Mining. The Alpha Algorithm was the first process discovery technique able to discover concurrency. But the basic algorithm has

many limitations including the inability to deal with noise, particular loops, and non-free-choice behavior. Basically the α-algorithm scans the event log for particular patterns, like if event A is followed by B but B is never followed by A, then it is assumed that there is a causal dependency between A and C.

Many people have read about the alpha-algorithm in some paper, or in the ProM tutorial. The alpha-algorithm is interesting from a scientific perspective, because it can be formalized in 8 lines and because interesting properties can be proven around it. But it may not be the best starting point.

For real-life logs, the alpha-algorithm is almost never the right choice. It won't work, it will return a result but not a good one.

To sum it up:

4. Mainly of theoretical interest. It is too simple to be applicable to real-life logs;
5. Does not address issues such as noise, etc;
6. Do not use as a benchmark;
7. Reveals the basic process mining ideas and concepts (8 lines of code);
8. Reveals theoretical limits of process mining.

Taking another route the Heuristics Miner plug-in implements a heuristics driven algorithm, which is especially useful for dealing with noise, by only expressing the main behavior present in a log. This means that not all details are shown to the user and exceptions are ignored. The Heuristics Miner plug-in mines the control-flow perspective of a process model. To do so, it only considers the order of the events within a case. In other words, the order of events among cases isn't important.

The Heuristic Miner was the second process mining algorithm, closely following the alpha algorithm. It was developed by Dr. Ton Weijters, who used a heuristic approach to address many problems with the alpha algorithm, making this algorithm much more suitable in practice.

Heuristics Miner is a heuristics driven process mining algorithm, that is, a practical applicable mining algorithm that can deal with noise, and can be used to express the main behavior (i.e. not all details and exceptions) registered in an event log.

Event and executor relationships are determined using weighted frequencies. That is, how many times each event is followed by another event weighted by the number of times the event occurs as a result of any event. A and B are subsequent to C. A is followed by C two times, while B results in C 8 times. The connections between B and C is stronger (80%) than A to B (20%). Therefore, to find a process model on the basis of an event log, the log should be analyzed for causal dependencies, for example, if an activity is always followed by another activity it is likely that there is a dependency

relation between both activities. In a practical situation we never know if a trace is noise or low frequency. Heuristics Miner handles this using three available threshold parameters:

1. Dependency threshold;
2. Positive observations threshold;
3. Relative to best threshold.

Last but not least, the Fuzzy Miner is one of the younger process discovery algorithms, and was developed by Fluxicon co-founder Christian W. Günther in 2007. It is the first algorithm to directly address the problems of large numbers of activities and highly unstructured behavior.

The Fuzzy miner uses significance/correlation metrics to interactively simplify the process model at desired level of abstraction. Compared to the Heuristic miner it can also leave out less important activities (or hide them in clusters) if you have hundreds of them.

To end this section it is worth to explain in summary how an event relationship is determined, regardless of the technique or method used. There are four basic relation types between two events. They are:

1. Direct Succession: X > Y (some case X is directly followed by Y)
2. Causality: X -> Y (X > Y and not Y > X)
3. Parallel: X||Y (X > Y and Y > X)
4. Choice: X#Y (not X>Y and not Y>X)

6. Execution

6.1. Introduction

In this chapter we will cover the execution of this case study. We walk through the whole process from identifying the data sources, converting the data into the required format, importing it into the ProM tool, review the event log and execute Process Mining techniques, namely discovery algorithms.

6.2. Sources of Data

There are several potential sources of data for event logs, but in most of the companies they may not be easily available or convertible. ERP and Workflow management systems would be the most probable choices, but ERP systems are more used and common, therefore developing methods and mechanisms to pull, transform and convert data from the organizations top management system would be the best choice for anyone starting and wanting to sell the idea of Process Mining.

Process Mining tools can be fuelled by any event log in transactional information systems ranging from WFM, ERP, CRM, SCM, and B2B systems, the so called Process Aware Information Systems (PAIS). But still the ERP systems are the most commonly available and reliable source of data for Process Mining in most companies. Available and reliable because it supports most of the financial reporting obligations. To sum it up ERP systems are business management software, usually a suite of integrated applications, that almost every business organization implements to collect, store, manage and interpret data from a wide range of business activities. Although process oriented tools like Workflow Management and other related BPM tools, including Process Document Management tools would be the best sources of data for Process Mining, most of the times these applications are absent or cover only partial processes.

There is a wide range of ERP systems available and used by most of the companies in the world like SAP, Oracle (JD Edwards), Sage, Microsoft Dynamics (Navision), Infor and many others. In 2013 SAP, one of the most used ERP systems worldwide had only about 29% of the market share. Almost 40% of the market share belongs more than 20 different management systems and none of the mentioned ERP software solutions. Big companies may implement different management software to cover different processes, for example Finance or Human Resources may be covered by different solutions.

There are more than 100 Workflow or Business Process Management (BPM) tools that feature different functionalities like Approval Process Control, Asset Management, Automatic Notifications, Configurable Workflow, Dashboard, Event-Based Notifications, Graphical Workflow Editor, Help

Desk Management, Resource Management and User Activity Monitoring. Not all of the solutions cover all the features described, but mainly the different from the common business management systems (ERPs) is that BPM tools are focused in improving corporate performance by managing and optimizing a company's business processes.

6.3. Available and Reliable Source

The ERP system may be only one of the available sources of data, but probably in most cases it is the only reliable one. Most companies don't invest in Process Workflow Management systems that are really process oriented, not like ERP systems that are accounting oriented. Sometimes this management tools, when implemented, only cover parts of the process that are not adequately covered by the ERP. This tools may be disconnected from the main process of the ERP system. The disconnection will happen because the systems belong to different vendors, have different corporate owners or are technically hard to connect. In the perfect world a perfect system would allow an organization to manage and enforce a process, to manage documents and records and at the same time support accounting, management and reporting, this concept would probably substitute the current known ERP systems. Reality as proven this to be at least a bit difficult task. There are at least a few big differences regarding accounting and management, financial and management reporting. The focus and objectives are different and sometimes even incompatible. On one hand the first desires transparency, to build an accurate picture of the company, but management may want to do another analysis to help and understand the business in more detail, creating performance metrics, outlook of events and a better understanding past events at the light of new rules. Agglutinating both concepts in a system would be so complex and difficult to maintain, that it probably will never happen. Most times, the systems only complement themselves and cover different parts of the process. If they were perfect substitutes there would be no reason to maintain separate systems and duplicate the work and record keeping for both accounting and process management. On the other hand, such Business Process Management tools that complement the ERP systems are too expensive to buy and maintain for the majority of the companies. You would need specialized workers or consultants to maintain the system and pull relevant and valuable outputs. The problem for Process Mining is that the ERP systems being in most cases the only reliable data source are mostly accounting oriented, that means the ability the track documents and follow the process end-to-end is not mandatory, that is the user does not need to follow a specific, restrict and clear process flow with defined logic and process oriented rules and also the process flow may not cover all the relevant tasks like reviews, analysis, decisions, approvals, etc. Additionally the system may have process gaps and

back doors that allow users to bend the rules and standard defined procedures. This happens all the time, even though everyone promises that the system is bullet proof or at least everything the company owners want and need. For example, if not adequately configured it is possible to invoice a fully delivered order with differences in the confirmed, delivered and invoiced quantities without any need for formal approval and justification (reason code) to allow an adequate audit or process review. That means ERP system configuration when done loosely will allow inconsistencies in the process. Another example of this type of gap and inconsistencies is the execution of the sales return without the need or requirement to allocate and link it to the original order document. This means we would lose the full picture of the process and the Process Mining results would be found useless. Or not… on the contrary, this is where the Process Mining tools and techniques may come handy and prove useful and value adding to the company. With a simple demonstration the process owners will find all the gaps, inconsistencies and non-compliant events, allowing them to act on it, with the full knowledge and understanding of what, when, why, who and how it happened. So bad information sometimes is a good thing if it reflects what really happened or even problems and missing controls in the system, therefore allowing to enforce new controls, adjust the configuration of the system and improve the process.

The processes are not perfect and do not represent exactly and all the time the utopia that consultants and managers define. It takes only a change in a user, in a rule, in a behavior, in a decision, in the process to make create a different process flow. The normal standard processes will follow a normal path, but the true value is in understanding what is not normal and known and be able to act on it. We could be talking about the need to review the process design to include new possibilities, fraud or undesired behavior detection, process improvement opportunities or finding ways to explain the past and predict the future. This means for example, the system process flow may be non-compliant with internal and external regulatory requirements. Looking at the true process flow, understanding the gaps, the turns, redundancies and different variables will allow us to fix the systems or teach the users how to properly develop the process.

Understanding that the ERP system has a lot of gaps and holes. Although not the best solution for process mining is the only reliable source of data. It does not enforce a process workflow allowing as many process event combinations imagination can cope with. And there is no clear, unique and simple interpretation and comprehension of the system process flow.

6.4. SAP and ERP definition

At this point, before we go any further, we should understand the meaning of ERP system and specifically SAP. SAP means Systems, Applications and Products in Data Processing and is the third largest software company in the world and the largest business application and Enterprise Resource Planning (ERP) solution software provider in terms of revenue. Founded in 1972 as Systems Applications and Products in Data Processing, SAP has a rich history of innovation and growth that has made us the recognized leader in providing collaborative business solutions for all types of industries.

In a more broad sense, Enterprise resource planning (ERP) is business process management software that allows an organization to use a system of integrated applications to manage the business and automate back office functions. ERP software integrates all facets of an operation, including product planning, development, manufacturing processes, sales and marketing. It is a business management software (usually a suite of integrated applications) that a company can use to collect, store, manage and interpret data from many business activities, including:

1. Product planning, cost and development
2. Manufacturing or service delivery
3. Marketing and sales
4. Inventory management
6. Shipping and payment

6.5. ProM Inputs

In this section we will cover specifically the definition of the Event Log input file, namely its dimensions. That is, we will deep dive on the definition of tables and fields that serve as inputs for the creation of the MXML file.

To sum it up there are four tables, two main tables, one with the list of case and another with the list of events assigned to each case, and two secondary tables with possible additional attributes about the cases or the events. The tables and fields are listed below:

1. Process Instances: list of process instances (cases - e.g. customer orders, patient admissions). This table contains:
   PI-ID - identifier of a certain process instance (primary key)
   Description - process instance description (if applicable)
2. Data Attributes Process Instances: additional information about each process instance (data attributes). This table contains:

PI-ID - identifier or a certain process instance (foreign key)

Name - name of the data attribute

Value - represents the value of the data attribute

3. Audit Trail Entries (Events/Tasks): data about tasks that have been performed during the execution of a process instance. This table contains:

   PI-ID - identifier or a certain process instance (foreign key)

   ATE-ID - unique identifier for each audit trail entry (primary key)

   EventType - event type, e.g. start, complete

   WFMElt - name of the task

   Timestamp - time in which the task changed its state

   Originator - person or system that caused the change in the task state

4. Data Attributes Audit Trail Entries: covers additional information about each audit trail entry. This table contains:

   ATE-ID - unique identifier for each audit trail entry (foreign key)

   Name - name of the data attribute

   Value - represents the value of the data attribute

To run the first Process Mining iteration you only need to fill some of the information, namely Case, Event and Time. If possible add the Originator, to facilitate the follow-up actions. Let us take a closer look at each of this main dimensions.

1. Process Instance or Case ID: a case is a specific instance of your process. In this case study we look at the sales process therefore handling one order is one case. Each event needs to be mapped or referred to a case allowing the process mining tool to compare several executions of the process. This is used to uniquely identify a single execution of your process. The range of the case, that is, from where it starts to where it ends determines the scope of the process analysis. With this said you may look at a group of events and determine different cases depending on the process perspective or the scope you want to address. So you look at each customer order as an individual case or you may want to see every order item for more detail, that is, every product-to-customer combo, or you may want to look at the customer perspective. You may also want to look at delivery scheduling and look at partial or bundled deliveries and in depth analysis of carrier services, routes, delivery times and supply-to-demand balance (evaluate your ability to serve), etc. For each specific analysis you may need

to create different subsets of the Event Log, use different granularities or change the perspectives of the analysis.

2. Activity or Event Name: one step in your process, for example Design, Create, Update, Test, Submit, Approve, Deploy, Request Rework, Request Enhancement, Revise, Publish, Discard, Communicate and Archive. The steps may occur many times and/or they may not occur all in the same case. A case may be composed of relevant and irrelevant events that may be considered noise. This may imply the need for data cleaning. It is the event that determined the level of granularity or detail. In order handling you may look at each order as a whole (it contains several supply requests) or at each item in an order and every load and pack process (may be several for each supply request if the delivery is split because of pure volume reasons, carrier requirements, transportation limitations or supply/stock insufficiency to supply the full order).

   It is possible to cluster the activities by region, source (direct electronic, sales professional, service agent, etc), asset group or product family, customer service team, business unit, etc. In this cases the activity name needs to be composed by both the operation or execution and the additional differentiated attribute (e.g. business unit, source, function role, sales office). The resulting process flow will then be divided by the additional attribute and you'll end up type A orders and type B orders with potential different paths.

3. Event Timestamp: this determines when the event took place and also the sequence / order of the events in the same case. For example, for repeating events or activities this will distinguish which event took place first and were in the whole flow. The time of the events allows to determine the sequence in a process instance, it is the order attribute. An event or activity may be composed of several events or different tasks depending on the level of granularity you look at. If an activity is composed of both start and finish time you'll be able to determine the processing time for each activity / event, that is, the time spent by a specific resource in the execution of a particular task.

   To sample a good process you may need to use the timestamp to select and frame a specific period in time. The minimum should be 3 months, but 6 to 9 months would cover almost every chance of seasonality and spot behavior. This needs to be revised depending on the volume of activity and the specificity of the industry process.

4. Originator or Resource: each activity is executed by a specific role, group of people, team, cost center, function, department or individual operator. This dimension helps you determine

who performed or handled the activity or task. This could be also a good way to split events by type taking into account the function that executed the task. Important or premium customer may take more time and specialized agents and technical expertise and even more tasks and approvals, while C customer may have a standard streamlined execution process. Other dimensions or attributes may be included, namely the value of the order, the product category (e.g. product hierarchy, brand, class, asset group, origin group), business unit, customer group or parent DUN, etc. Process Instance, Event and Time are the base and required fields, all other, including the originator or resource enrich the analysis.

The Event Log MXML file is the base input of ProM Framework tool. Let us take a close look at its basic structure. This is particularly useful for those developing new ways of creating this file from raw or converted system data using different types of connections and hosts (e.g. ODBC Connection, Proprietary System Connectors, SQL Databases) without using ProM Import (free) or Disco (not free).

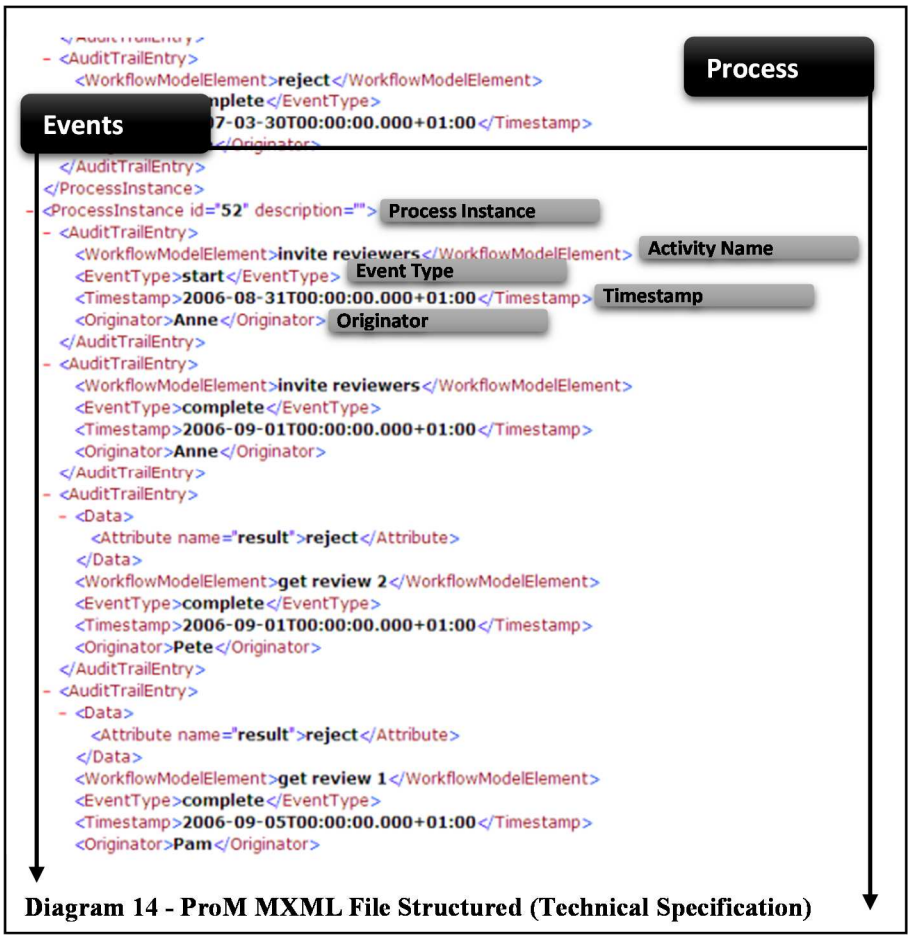


Diagram 14 - ProM MXML File Structured (Technical Specification)

Disco has a demo version available that allows only 100 events per file. And even with a paid versions you still need to track the origin of every document and put it in a list layout in accordance with the requirements of the tool you wish to use. Mapping additional attributes needs a depth knowledge and understanding of all process related tables and the respective matching keys. For sales and distribution process in SAP there are more than 20 related tables with different pieces of the puzzle. Activity records are scattered through many tables. Several tables from the standard sales documents to delivery and billing documents. This will be covered in the next section.

6.6. ETL Process

This case study focused on the SAP Sales and Distribution process, specifically order handling. Before running any process mining technique you need to identify all the main and relevant tables, understand the relevant contents and relationships and created an Event Log out of it. In other words and relating to the case of this paper to identify the main tables is to list them:

1. VBAK (Sales Document Header);
2. VBAP (Sales Document Item);
3. LIKP (Delivery Document Header);
4. LIPS (Delivery Document Item);
5. VBRK (Billing Document Header);
6. VBRP (Billing Document Item);
7. VBBE (Sales Document Business Requirements);
8. VBEP (Sales Document Delivery Schedule Lines);
9. VBUK and VBUP (Sales Document Status Header and Item);
10. VBFA (Sales Document Flow).

This are the standard tables that contain most of the relevant information about the sales process in SAP, from order to bill. The process is obviously a little more complex and may differ from implementation to implementation, having more or less controls and steps in the middle, but the standard tables remain the listed. All this tables have all the sales document attributes and characteristics and allow the users to follow-up, end-to-end, on the documented process in the system. The only restriction to create a process flow or event log with all this transactional data and document information is to understand which documents precede or follow in each step of the process and associate them to a specific case or process instance. For that the system needs to have a specific configuration that records the document flow. This event relationship table is called Sales Document Flow recorded in the standard SAP table VBFA. Without it the Event Log creation effort would

become quite difficult. This table contains all the tracking, preceding and subsequent document flow in the system, allowing the user in the front-end application to have a quick and simple view for each document selected, namely what are the following and preceding documents and their status. We should make a translation at this point and understand that each document generated in the system represents an event in the process. A customer asks the company for a price or budget (quotation), this produces an inquiry document in the system which is followed by the creation of a quotation document. This may result in a sales order, a delivery, picking list, packing and loading, shipping and finally a bill document. The following diagram is a very simplistic view of the SAP Sales and Distribution process flow:

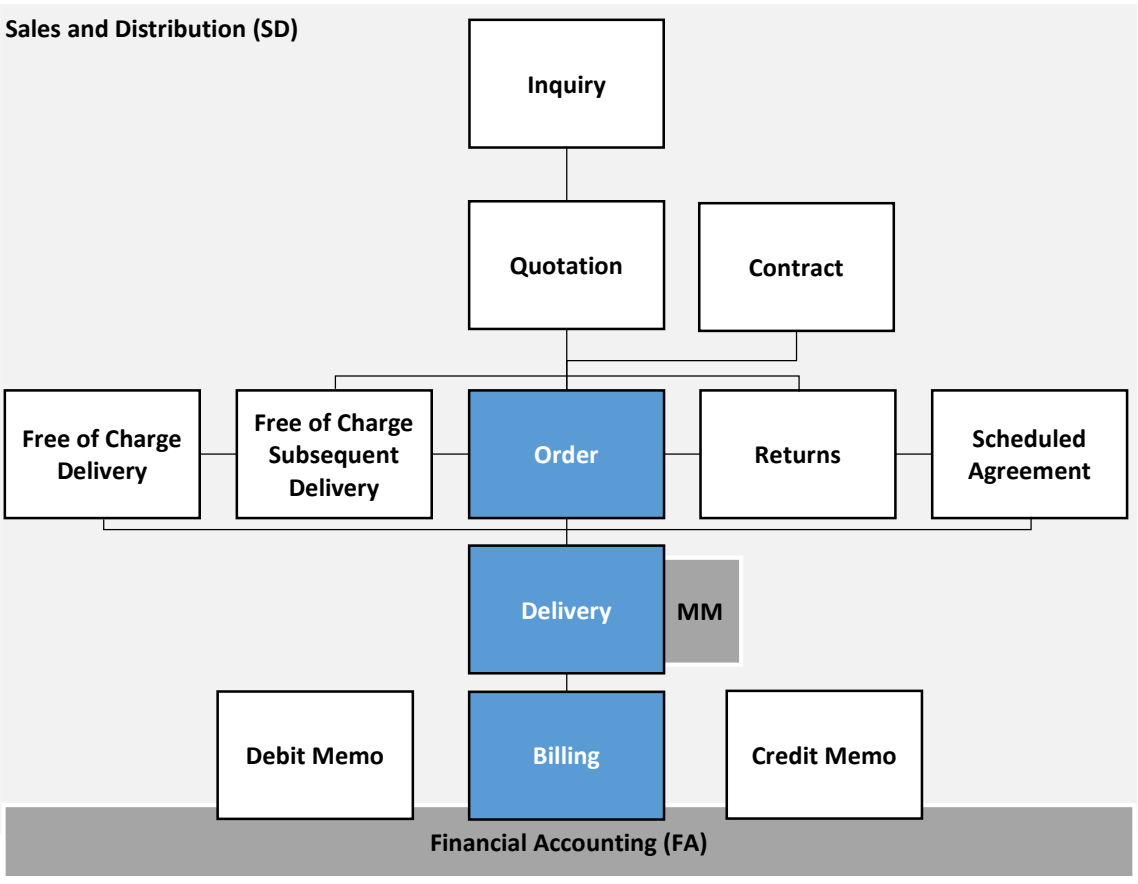


Diagram 15 - SAP Sales and Distribution Process Design

In this diagram we can also see that other SAP modules may come into play, namely Material Management (MM) for Goods Transfer and Issue and of course Financial Accounting (FA) when an invoice/bill, debit or credit memo is produced and accounted for as a result of the sales process.

The ETL process consists in picking up the extracted flat files, mainly table dumps, from the source systems, if no ODBC or other automatic connection is available, selecting and formatting the relevant data and creating the ProM input tables that are described next.

Although this may seem simple, it is in fact the heaviest part of the work, because the information may be scattered through a big number of tables, the process miner needs to have a great knowledge and understanding of the real process and how the system records and documents this process, which tables are used to record those events, which are the relevant attributes, their data types and how to relate all the information in a comprehensive way to create a simple event log with all document flow sequentially listed with all the major attributes mapped.

In my application we need to explode all related events or system documents. Each document in the system is a recorded event, but the system does not create a simple audit log, it only relates two events per line in the Sales Document Flow. This means we need to find the first document in the process and drill it down till the end of the process in a loop routine or script that associates every related event to the same process instance creating an audit log of all the related documents.

For you to have the full picture of the sales event flow you need to join two perspectives, the document flow and the document data. The first one gives you for each document its preceding document. The second one gives you all the details for each document.

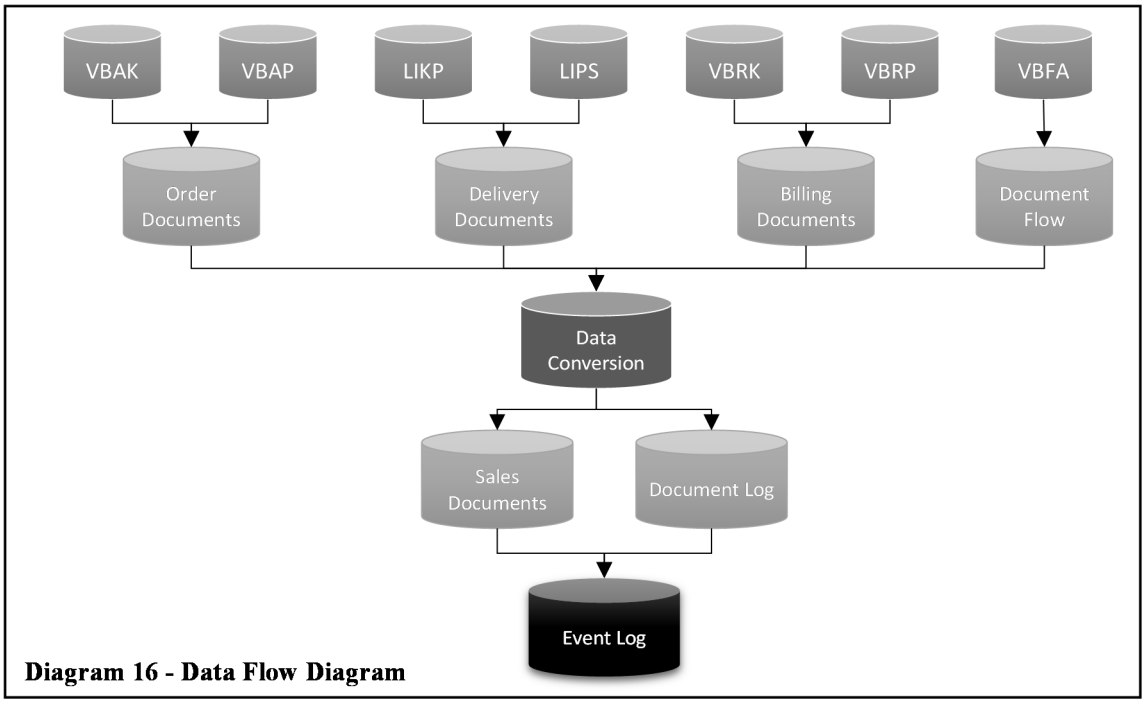


Diagram 16 - Data Flow Diagram

Using a recursive program you'll be able to associate every document to single and unique case based on the individual link between subsequent and preceding documents. Start from the document / record that does not have any subsequent events but only preceding. From that one drill down and uncover the whole trace of events. When you have that skeleton fill it with the meat from the document data that may be scattered through many related tables. This data will tell you every detail of each single transaction.

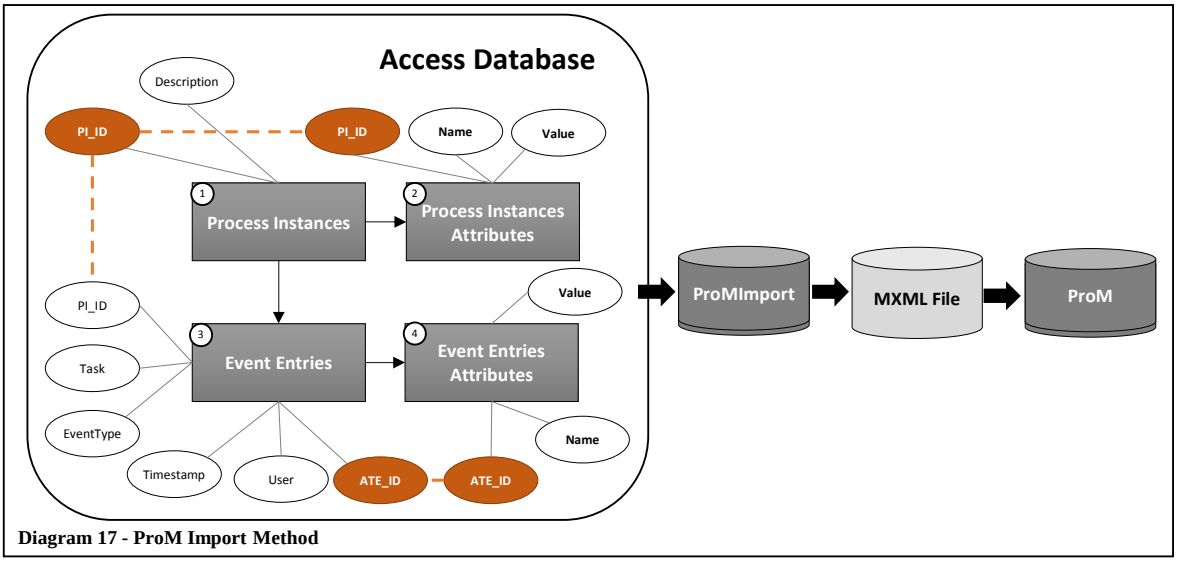


Diagram 17 - ProM Import Method

The main table is called Process Instances, technical name Process_Instances (1). It is the list of all Process Instances and determines all the Process Mining process flow diagram creation. If we have events that are not connected to an existing process instance they will be excluded from the analysis. It is the index of all events in the audit trail or document flow.

Associated with the Process Instance table we have the Process Instance Attributes (2) table is named Data_Attributes_Process_Instances and should contain all relevant characteristics and attributes of the process instance itself. In this case each sales process flow is a process instance, this means attributes would include the type (sales order, sample shipping, product return, etc) the value and volume of it (net weight, gross weight and expected revenue win or loss), and others. This is not a mandatory table, this means to create a process flow diagram this table doesn't need to be populated.

The second most important table contains the Process Instance Event Entries (3), that is to say, the Process Audit Log and is named Trail_Entries. It contains all events for each process instance, the type, timestamp and originator. It is in fact the list of sequential events in a process instance.

The last table is the Event Entries Attributes, named Data_Attributes_Trail_Entries (4) that includes all additional attributes relative to the events in each process instance. It is a list of infinite attributes

like the type, duration and other characteristics of the event. The table is composed of the event id, the name and value of the attribute. This is not a mandatory table, this means to create a process flow diagram this table doesn't need to be populated.

The names of the tables are irrelevant for ProM Import. This means, when creating the source tables from ProM Import only the layout and data types are relevant for the process, in the setting you can refer to whatever name you which to give the tables. In this application the source tables are:

1. Process_Instances
2. Data_Attributes_Process_Instances
3. Trail_Entries
4. Data_Attributes_Trail_Entries

In a nutshell you need to successfully extract relevant raw data from the source system SAP, transform it in to an Event Log, in this case performed using Access and VBA programming, convert it into the input format MXML and import it into ProM, before being able to run any algorithm and sharing the results. The whole process is shown in the following diagram:

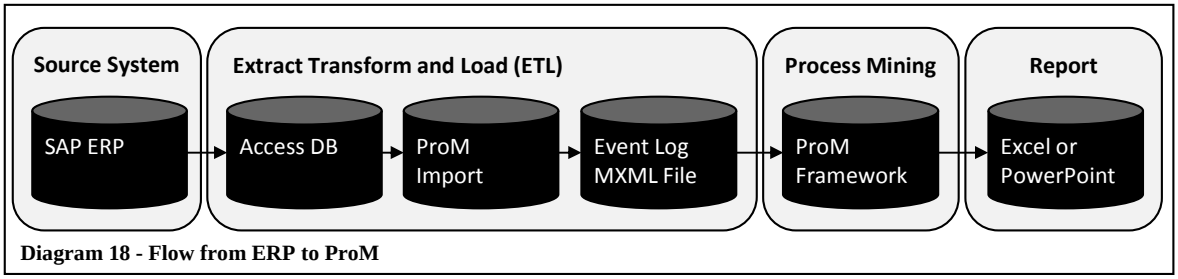


Diagram 18 - Flow from ERP to ProM

In this case study a method was developed using Access Queries and VB programming. This allowed to link the different SAP tables, select and rename the relevant fields and develop a routine that allows the construction of the Event Log. This routine is explained in the next diagram:

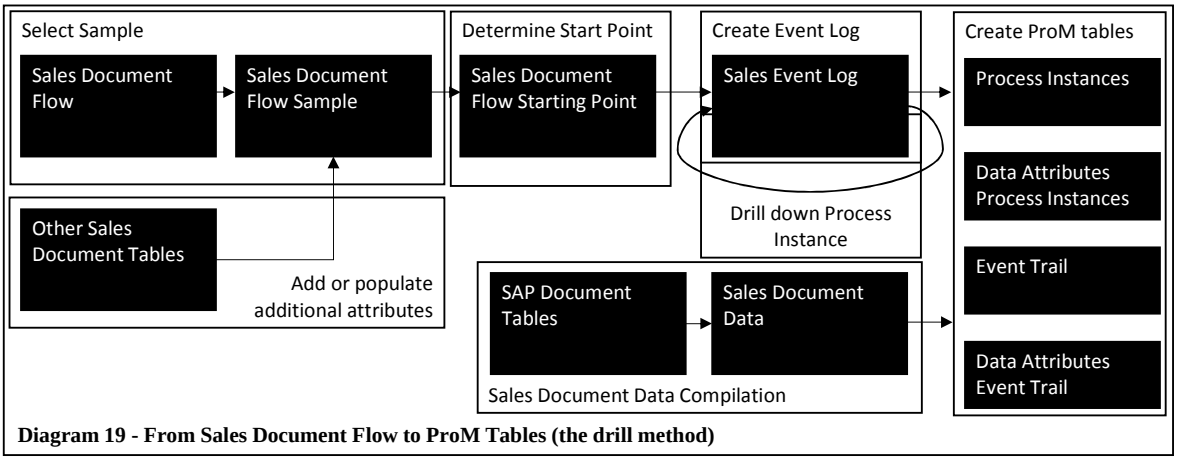


Diagram 19 - From Sales Document Flow to ProM Tables (the drill method)

Understanding the data structure and sources of information (system tables) and their role in the whole analysis is the most critical and probably the most difficult step for Process Mining execution. This statement applies not only for Process Mining techniques but for all Data Mining and Business Analysis jobs. The ETL (Extract, Transform and Load) is the foundation of any business data analysis. Sometimes this is done by external consultants or engineers that don't really understand the business process or how to interpret the data and in most cases don't have a clear vision or opinion about the desired outcomes of the whole data and business analysis practice. They are not responsible for the outcomes, sometimes they don't have contact with them or receive any end-user feedback, so there is no reason for them to cross that bridge between the creation of raw data and the business analysis. This in the end results in poor quality raw data and even worst results and reports.

The first step for good quality raw data for Process Mining is to understand the different sources of data available for the same process. Recognize the different structures, layouts, formats and how they complement or substitute each other. Find that some sources can and others cannot be used to produce adequate raw data, in this case an Event Log for ProM Framework Tool. Some sources are structured and can be related and easily mapped to the business process, but many other sources are bits and pieces of information with ambiguous connections to the process itself, not allowing the Process Miner to take full advantage of all the relevant information available. For example, writing a comment about a business contact, an opportunity or insight in a piece of paper, even if the comment is really important and relevant to the process, is useless if there is no adequate relationship with a process instance or is in an invalid format. In this case paper or notebook notes would need to be optically recognized, text mined and linked to rest of the process maintained in the system. In most cases the cost is too high and it would be preferable to manually add the information in the system. Most times the information is lost, it belongs to the Tribal Knowledge category and is recorded and archived in the brains of some people, and a gap is produced in the process. Most departments prosper with Tribal Knowledge, because that allows them to serve as the bridge for that gap and appear more valuable to the company. Besides notes in a notebook also the general use of flat files by different process contributors as well as other software tools dispersed in the organization to fulfill specific tasks and complement the ERP systems in fragments of the process may not in the end be undoubtedly linked and used to map and understand the complete process flow. That means in most cases we will have a process full of holes. With that said, when using the company's main system, usually the ERP or accounting system, it is critical to have the correct and complete understanding of the document process flow in the and other related software tool and sources of information that

cover the process flow, understand the logic behind how the records are created, archived and tracked and how to interpret the resulting data. There is no machine intelligence that allows a data miner to blindly extract data from files, libraries and system databases, hope it is compliant with the needs of a Process Mining tool and support Process Flow Diagram without any previous understanding, mapping, transforming and cleaning. That would be a great misconception concerning Data Mining and Business Analysis work. Many think that the available software solutions allow us to pick up random pieces of data, pull them together without any previous understanding of relationships, layouts and contents and build reports and outputs with a push of a button or swing of a magic wand. Because of this in the last weeks I've spend a lot of time reading about the SAP Sales Order Process. Found that sometimes even consultants, developers and users don't agree or have a clear and simple vision of specific parts of the process in the system and relevant concepts for its understanding.

6.7. Event Log Review

Once you have created an event log and opened it in ProM, you may want to clean, enhance or change it in some way. This is usually referred to as Event Log Filters. This filtering is done for two main reasons: cleaning the data or narrowing down the analysis. Sometimes there are also technical reasons, like plugin requirements, that may require the use of a filter before running the algorithm. This filters change the event log in four ways:

1. Remove process instances (cases)
2. Add events
3. Remove events
4. Modify events

The first one is clear. You may filter out some cases based on different attributes, for example, take out a specific type of customers, products, companies, etc. This task can all be one by removing some types of events. With that said, let us cover the basics on this topic and develop a little more the subject, specifically adding, removing and modifying events with some practical examples.

To Add Events implies you enhance or extend you base Event Log, this can be done for example by adding artificial start and end events. This is one of the most important event log preparation techniques. Most of the event logs will show every event connected to each other, not being clear where the process starts or ends, because all activities are connected. Therefore you can create a clear start and end point in your process model, you can use the so-called Add Artificial Start Task Log Filter and the Add Artificial End Task Log Filter in ProM for this effect. After executing the plugin you may verify the results in the log inspector and you will find a Start event connect to the event

were most cases start or the event that never as a predecessor, depending on the method. In fact, some of the mining algorithms (such as the Heuristic miner) assume that there is an identical start and end event for each case. So if you don't use this technique the quality of the result may be reduced. It is strongly advised to add these start and end events in most situations.

Regarding the Remove Events type in most cases it consists in filtering out case. You won't really wish to ignore events in the process, you can simply cluster them in the discovery process, but you may want to filter out incomplete cases. So regarding this topic you may filter out cases by filtering cases based on how they start and end. Instead of adding artificial start and end events you may select only those process instances that start and/or end with particular activities in the first place. After extracting data and producing an event log one gets a data extract of the complete process logging in a particular time frame. So, the event log most likely contains some process instances that are incomplete because they were started before the data extract begins, or they were not yet finished when the data extract stops. To clean up your data you should remove those incomplete process instances from the log. This way you may determine which is the real and end events for your process and throw away all cases that do not fulfill this filtering criteria. This can also be done when preparing the Event Log.

The last event changing task you may perform is Modify Events. For example, use the remap element log filter. This one is also useful and quite powerful. You can remove or change the name of events based on regular expression matching. This is handy if you have several low-level events that you would like to project on the same higher-level activity. This is a kind of clustering, you put several low-level event in the same umbrella. The preview shows you the effect of your matching rules, so you can check whether you got the expressions right when you try it.

Another modifying tasks you may perform is filtering duplicate tasks. This filter removes direct repetitions of events with the same name. You can use it if you need to get rid of duplications, or if you have used the remap filter mentioned before to combine several lower-level activities into one.

Using repetitions-to-activity filter adds a start and complete event to each activity, allowing to execute performance analysis and collect performance metrics, attribute value filter the is used to keep only events that have a certain attribute value.

Process instance length filter if you wish to focus on those cases that needed more activities than others to be closed, namely to cover and understand rework cases, this filter lets you specify which process instances to keep based on a threshold on the number of events in the sequence.

With the enhanced event log filter both events and process instances can be filtered based on an activity-based frequency percentage threshold. This is useful if there are hundreds of different events to, for example, focus only on activities that occur in most of the cases.

To finish this topic we are left with one last event log potential filtering technique called event log split up. There is no specific log filter to perform this task, but you may use LTL Checker plug-in. This will allow you to split an event log in two logs, for example one log that contains all instances that executed a specific activity "A" and a second log that contains all the other instances.

6.8. Algorithm Selection

Each algorithm and/or ProM plugin may require different features in the Event Log, for example. Heuristics Miner does not need the Event ID, it relies in the timestamp to establish the trail. Each plugin may require different information or input to be run effectively and the same event log requirements may change from algorithm to algorithm / plugin to plugin. For example, Heuristics Miner doesn't need the Event ID, only used the Timestamp to relate events. Do not use the Alpha Algorithm. It is a good theoretical basis, but with no practical relevancy.

1. Fuzzy Miner: the Fuzzy miner is one of the younger process discovery algorithms, and was developed by Fluxicon co-founder Christian W. Günther in 2007. It is the first algorithm to directly address the problems of large numbers of activities and highly unstructured behavior.

   Input: Event Log

   Output: Fuzzy Model

   Application scope: complex and unstructured log data or objective it to simplify the model in an interactive manner

   About the algorithm:

   The Fuzzy miner uses significance/correlation metrics to interactively simplify the process model at desired level of abstraction. Compared to the Heuristic miner it can also leave out less important activities (or hide them in clusters) if you have hundreds of them.

   Known limitations:

   The fuzzy model cannot be converted to other types of process modeling languages, but you can use it to animate the event log on top of the created model to get a feeling for the dynamic process behavior.

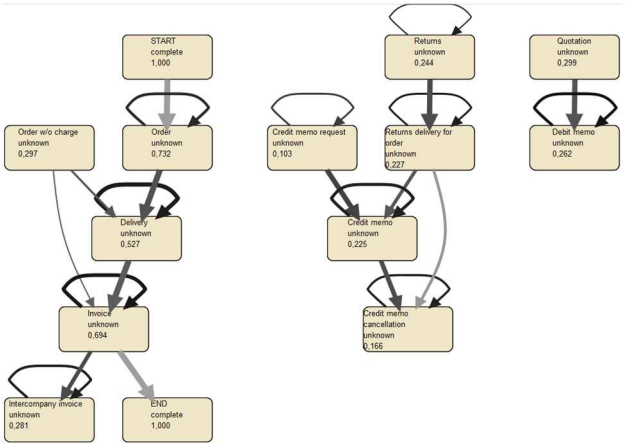


Figure 2 - ProM SAP Order Handling Fuzzy Mined Model (Example)

You can really manipulate almost every parameter to adjust the granularity (cluster non-significant events) and other fitness variables

2. Heuristics Miner

   Input: Event Log

   Output: Heuristic Net

   Application scope: This algorithm should be used when real-life data with not too many different events, or when you need a Petri net model for further analysis in ProM (as a starting point to run another algorithm).

   About the algorithm:

   The Heuristic miner (previously Little Thumb) derives XOR and AND connectors from dependency relations. It can abstract from exceptional behavior and noise (by leaving out edges) and, therefore, is also suitable for many real-life logs. One of the advantages is that a Heuristic net can be converted to other types of process models, such as a Petri net for further analysis in ProM.

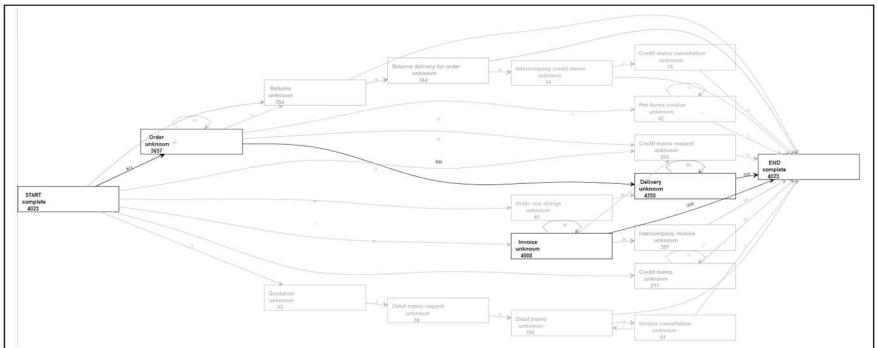


Figure 3 - ProM SAP Order Handling Heuristic Net (Example)

Using the Heuristics Miner with a lot of event divisions is not a good thing. It thinks the second delivery item follows the first delivery item when both come from the order, but because of the date and time in the event it tries to infer they are connected.

3. Multi-Phase Miner

   Not tried in this case study, the Multi-phase miner was the first algorithm to explicitly use the OR split/join semantics, as found in EPCs, enabling it to express complex behavior in relatively well-structured models. It was developed by Dr. Boudewijn van Dongen, a process mining veteran and longtime leading developer of ProM.

   Input: Event Log

   Output: Event-driven Process Chain (EPC)

   Application scope: simple and structured log data and you want to export the mining result to Aris. The Multi-phase miner folds XOR, AND, and OR connectors from so-called runs and displays the resulting model as an EPC. The EPC can then be exported to Aris (e.g., in Aris graph format) and further processed from there.

   About the algorithm:

   One of the advantages of the Multi-phase miner is that it constructs a model that always "fits" the complete event log (more on that in a later post). However, it is seldom useful for more complex processes because the model becomes unreadable.

Social Network algorithms were not covered in this case study. Also, there are a lot of log filters, log enhances and log cleaners that can be used before and after running different plugins and algorithms. In the previous chapter we've covered the basic list of some relevant plugins available in ProM. For example, regarding log conversion, cleaning and enhancement you have such plugins as:

Table 8 - ProM Event Log Preparation Plugins

Name
Add missing Events
Add identities to log
Add artificial events
Add noise to log filter
Calculate log meta data
Check compliance of a log
Concept drift
Enrich log for temporal compliance checking
Estimate Completeness
Filter log by attributes

Name
Filter log on event attribute values
Filter log on trace attribute values
Filter log using prefix-closed language (PCL)
Filter log using simple heuristics
Fix attributes
Fix timestamps
Remove extensions from log
Rename/merge events
Remove events from log traces
Prepare Durations for Correlation Testing

In this case study we used the "Add artificial events" to add a start and end event to our base event log before each discovery run.

6.9. Execution Guide

The execution of Process Mining follows a defined set of steps, starting from the identification of the data sources, extracting and converting that data, importing and running programs or algorithms and interpreting and making decisions based on the results. Every adventure and effort has to have an outcome or impact in the organization, this guides, supports and sponsors the implementation project. For Process Mining to be worth a while it must allow a company to improve process efficient (less resources) and effectiveness (more goals) and mitigate risks (low risk of failure).

If you are trying to apply Process Mining this is where you should start. In this section we will cover the 10 step method for executing and applying this techniques:

1. Select the process or scope:

   Pick a process that you want to analyze. This process should be clearly defined, i.e. you know the flow in advance, and it should be a common and frequently executed process. If possible start with a quick-win, that is, a process that is simultaneously simple, relevant and has margin to improve. Simple means less combinations and different flows or traces from the starting events till the completion of the case. There are several perspectives and ways of defining the business process scope. In this case study we've selected the Order Handling process that is a subset of the Order to Cash stream. You could eventually look at it this way:

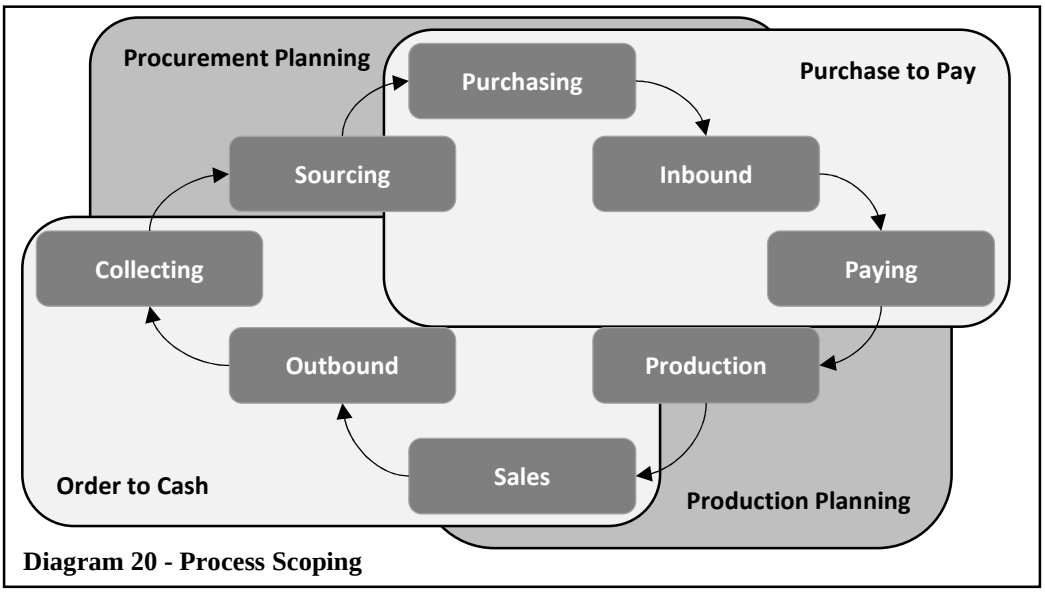


Diagram 20 - Process Scoping

2. Define the questions you want to answer:

   Want to discover the process?

   Want to obtain some performance and execution metrics like the most frequent and/or slowest paths and identify the activities responsible for it?

   Need to discover and review the organizational structure namely the flow between the different departments?

   The questions above will determine the next step, that is, what to look at.

3. Determine which systems and other files support the process:

   Follow and walkthrough the process to identify were events are recorded through the organization. All IT systems involved in the execution of the process are candidates, they should contain relevant data. This systems may include software like ERP, CRM, WFM, etc. Other customs tools, spreadsheets and databases may be used to obtain relevant information about the execution of the process. The data will be scattered amongst several sources and even in the same source through several tables. The data may be stored in databases or flat files. You may not be able to simply extract an Event Log, Document Flow or Audit Log. Firstly you need to configure each system to record such data, that is, save the execution steps of the process, namely the preceding and subsequent events (trace), timestamp, originator and case or process instance, the unique identifier. Even a big ERP may not have this by default, so alternatively or complementary you may need to construct your own event log based on transactional data, just need to find the link between the records.

4. Extract the data:

After identifying the data requirements you need to extract it in a form of a data dump that may be used to create the MXML input file. In this step you may also need define the time frame of analysis. 6 to 3 months is a good sample, but to capture the full process life cycle you may need to extract a full year for example.

5. Structure the Event Log:

   After extracting the data from the source systems you need to convert it into the right format. In the case the information is not an Audit Log type you may also need to do some additional conversions and even recursive programming to associate each associated event to the same case. In the end of this process you need to create a table that contains at least the Process Instance or Case Unique Identifier, Activity Name and Timestamp. The user or originator and the activity identifier may also be useful to enrich the final results.

6. Create/construct the Event Log MXML input file:

   There are several tools that help the analyst or data user convert a file format from CSV, Microsoft Access Database Tables or even based on an ODBC connection directly to the source systems into MXML file. Tools and applications like XESame, Disco, ProMimport do the trick. Use a licensed application like Disco or the free and available options ProM Import and XES tool or create your own tool with or without complementary usage of an existing free tool. For example, in this case study, a routine was developed to first put the information in the right format and second associate each event to the right case before using ProM Import to create the MXML input file. Applications like Disco may facilitate the work but at a price. When creating the Event Log you need to know the basic required fields: case identifier or process instance, activity or event name, resource / agent or originator and timestamp or time reference for the event. If the event runs for a specific duration you may want to specify the start and end time;

7. Import / Run the Event Log MXML input file in ProM:

   This means you can now open the ProM tool, select the import option, and load the log file in ProM. Select one of the available import plugins, try the Sequential;

8. Inspect the Event Log:

   After loading the event log file it is necessary to look at the result of the load. Namely the number of cases, the number of events, the numbers of different events, the average number of events per case and its distribution, the number of originators, the start and end that to

validate the time frame. You may also look at specific events in detail using the Inspector option. Here you may see the full sequence of events in each case;

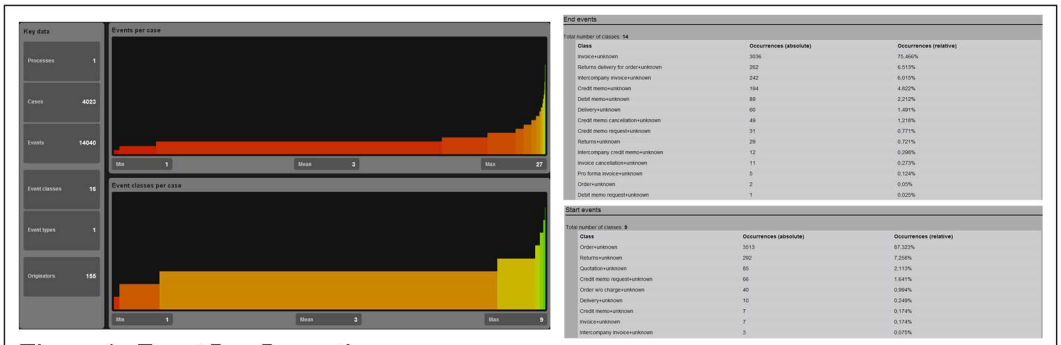


Figure 4 - Event Log Inspection

9. Mine a Process Model:

   Start with the Heuristics Miner and Fuzzy Miner. The advantage of the Fuzzy Miner is that it allows the user to change the parameters on the fly, while the Heuristics Miner has to be run several times with different parameters. You may change the level of accuracy of the displayed model. The numbers in the event boxes show the relative frequency of that task and on the other hand the thickness of the arcs indicates how often two activities have been executed after another;

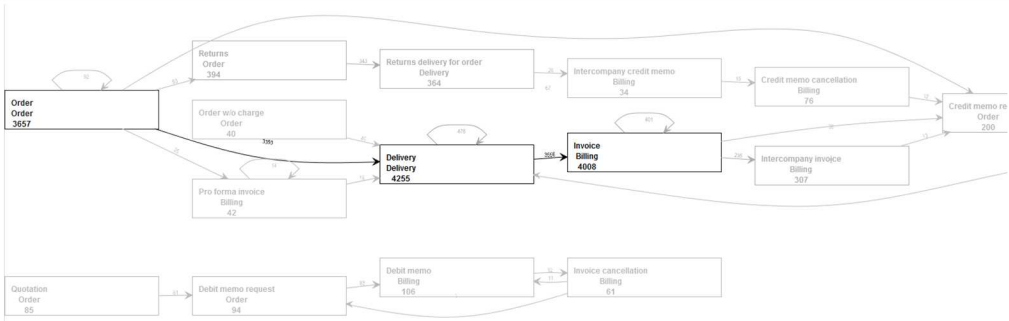


Figure 5 - ProM Mined Process

10. Animate the Process Model:

    Create an animation of the activities in the event log directly in the process model that was created from the same event log. Mine with Fuzzy Miner, use the resulting resource to Select the Best Instance and the apply Fuzzy Animator to create a good representation of the process flow. In the settings pull the look ahead slider to the very left, if you want to see just one token moving around the process for each case.

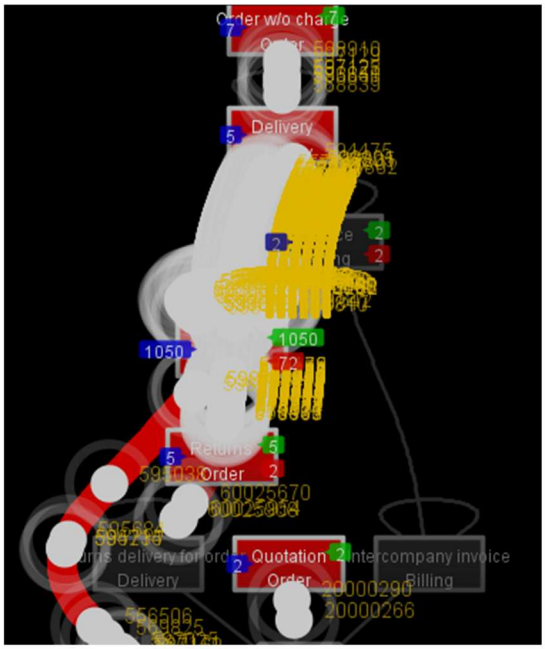


Figure 6 - ProM Animated Process

In the beginning of this section we started by describing that what drives a project is its outcomes and results. With this said, in the end, after you followed the steps above and executed and applied the Process Mining techniques you should be able to execute the following actions:

1. Interpret Results:

   Get insights from the information returned by the techniques. Understand better the process and find were the key points, drivers and majors concerns are in the process you choose.

2. Make Decisions:

   After obtaining insights and a clear understanding of the issues you should develop an action plan to cover those problems or risks. Design solutions, plan the execution, allocate responsibilities, get sponsorship and budget and apply the changes to improve the process as desired.

3. Monitor Actions:

   Monitor the execution and implementation of the actions defined in the previous step. Review the process before, during and after the changes and evaluate the impact of the actions. If needed redesign the recommendations and improve the resolutions before closing the topic. Continue monitoring the process till it stabilizes, before changing the focus to a new project.

Furthermore, there are some advices for any one developing a Process Mining project or initiative and wanting to use ProM. Before running the discovery techniques like Fuzzy Miner or Heuristics Miner, really look at and review the Event Log, this will save you a lot of time, pain and frustration. Before starting any other algorithm use the Inspector and Summary to see the frequency of events, start event, end events and the originator allocation to each of this types of events. That is, review the list of start events, end events, the users initiating and finishing cases and the average number of event per case, for example, look at the biggest and smallest cases to evaluate if they are real or just bad data. Also, looking at the case duration and relationship between events is also relevant. In ProM, the Discover Matrix plugin is an easy way to have a quick summary of relationships and to validate the quality and structure of the Event Log, This allows you to have a quick look at the relationship score between events right away. This should be the first task to assert the quality of the input data. So, start with a Discovery Matrix to understand the causality relationship between the events and validate the reliability of your trail capturing system (e.g. audit log).

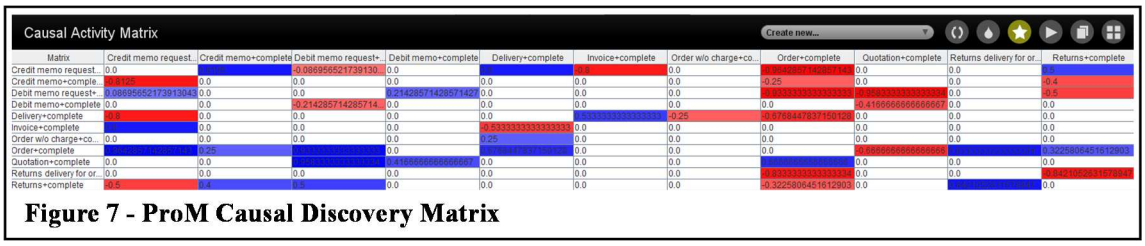


Figure 7 - ProM Causal Discovery Matrix

For this case study you can make some conclusions and validation based on the Discovery Matrix, namely:

1. Almost every Debit Memo / Credit Memo Request is preceded by an Order;
2. The only event that precedes an Order is a Quotation;
3. Following and Return we have Return Delivery, Credit Memo and Debit Memo;
4. Invoices are preceded only by Deliveries;
5. Some invoices don't have a preceding event;
6. Credit Memo Request generate almost every time a Credit Memo and a Delivery;
7. Most times a Credit Memo Request is preceded by an Order and Invoice;
8. Some Credit Memos results in a Return.

Another preparation task that should be performed before moving to the discovery phase is event log filtering, that is, for example deal with partial process instances by filtering the event log based on start and end events. Before running any Process Discovery algorithm you should begin with some Event Log enhancers and/or cleaners. First of all you should cut-off incomplete process instances that resulted from your sample extraction (e.g. pre-mortem cases that have not finished when the

extraction was made). The second enhancer you should use is to add artificial start and end events. Different cases may start and end in different activities (e.g. quotation or order). This plugin basically determines that one event always starts and has no predecessor (e.g. quotation) naming it the start event, even though most cases start with an order (not all order have a preceding quotation). This topic was already mentioned in the previous section called Event Log Review. Take a look and see any other filtering technique that you may find useful for your case study.

Before moving on to the next topic, there are some additional details the user should take into account when developing a Process Mining project. Namely, when creating the Event Log you may need to maintain different versions, for example different scopes or granularities. Some versions may have more details and some with more granularity and others may cover a bigger time frame. Basically, by common sense, you should start with the basics, this means low granularity, a simple process and with low details and a relatively small time frame. This would be version one (gamma version). Based on this, if the structure fits the ProM requirements you are able to progress and introduce more details and more granularity to the Event Log. In this case study you may start looking at the events only through the Document Flow, to build the skeleton, with structure but no data. This can be done at a header level or item level. Start with the headers only. This means for example one order may have different item lines (e.g. products), but in the gamma version you would assume it is a single order. In the beginning, just to establish the basic structure ignoring most of the details. While you progress you should add more details, the last version should include Document Data and may also include the whole document details with the whole item list and the most relevant information like volume, value, product family, customer business unit and/or segment, etc. In the last version, for example, each order would contain different lines and each of this lines would result in different deliveries and this deliveries would result in one single or several invoices (in some cases even an invoice would include orders from another case).

6.10. Case Study Results

Now let us take a look at the particular results of our case study covering the SAP Order Handling Process. Following the steps described in the previous section let us cover this case study execution:

1. Select the process or scope:

   SAP Sales and Distribution in particular the Order Handling sub-process, from quotation or order to billing of sales to the customer.
2. Define the questions you want to answer:

How does the Order Handling process in SAP looks like? How are orders processed in SAP? What are the different paths an order or sales document may follow before being invoiced? What is the most frequent path?

3. Determine which systems and other files support the process:

   The whole order to bill process is covered by SAP. Part of the process may start in an electronic platform used by the customer to introduce the requisition, but the orders are all uploaded and process in SAP. This means the same system cover the whole scope, from order to bill.

4. Extract the data:

   The whole data set for this process was extracted using the Data Browser transaction from SAP. This transaction was used to extract the following tables: VBAK, VBAP, LIKP, LIPS, VBRK, VBRP and VBAF.

5. Structure the Event Log:

   The event log source tables were created using Access Queries and Visual Basic routines. This process was used to link all events to each respective case based on the document flow trail and also map all relevant data for each event based on transactional data details, namely activity name, originator and timestamp.

6. Create/construct the Event Log MXML input file:

   The Event Log MXML file was created using ProM Import. This was executed using an ODBC connection to the source Access Database were the Event Log data was structured.

7. Import / Run the Event Log MXML input file in ProM:

   In this case study both Naive and Sequential ID import plugins were used. The results you'll see next were supported on the Sequential ID import method.

8. Inspect the Event Log:

   Now the fun begins and we can start looking at some concrete results. In the next output example we will verify the integrity and accuracy of the Event Log creation process. Taking a look at the some of the main metrics:

   a. Frequency of events: minimum, average and maximum number of events per case;
   b. Starting events;
   c. Ending events;
   d. Users or originators;
   e. Time frame for case events;
   f. Relationship between events.

Let us take a look at it, images are worth more than words:

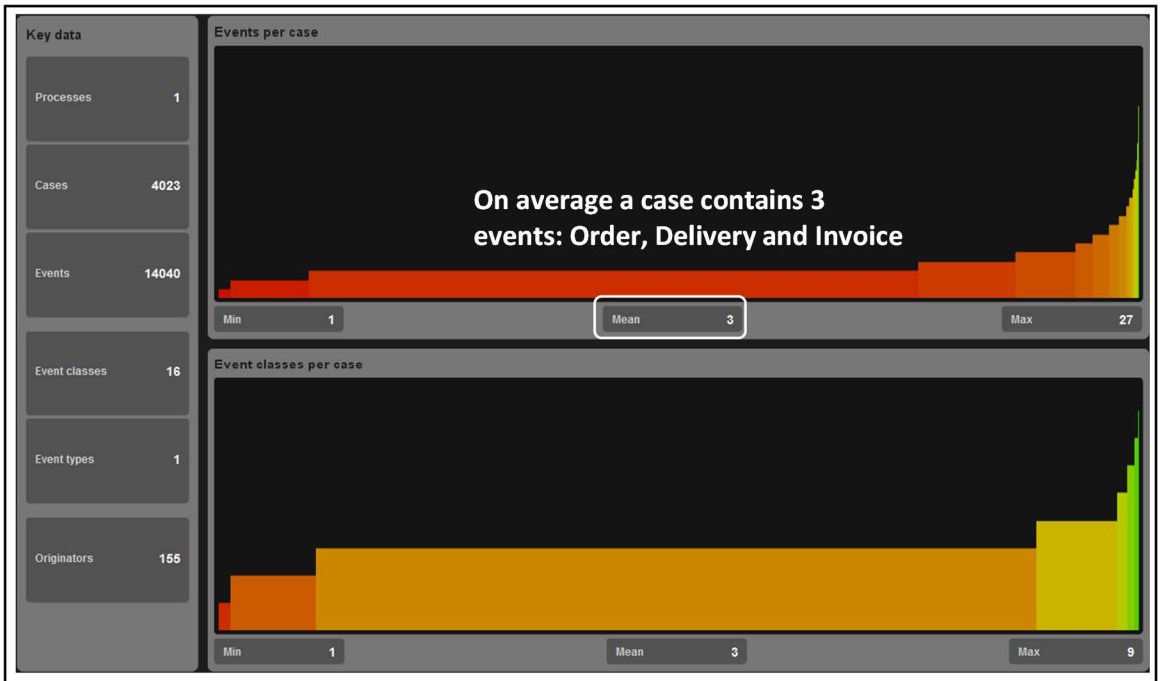


Figure 2 - ProM Event Log Import Dashboard

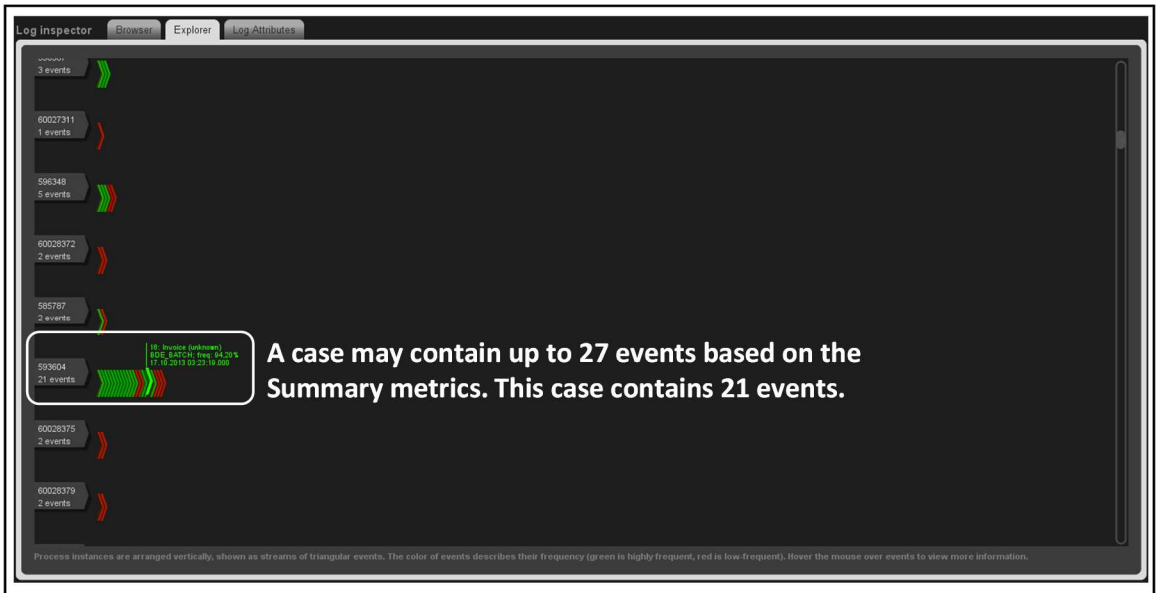


Figure 3 - ProM Event Log Inspector Explorer

Start events

Total number of classes: **9**

Class	Occurrences (absolute)	Occurrences (relative)
Order+unknown	3513	87,323%
Returns+unknown	292	7,258%
Quotation+unknown	85	2,113%
Credit memo request+unknown	66	1,641%
Order w/o charge+unknown	40	0,994%
Delivery+unknown	10	0,249%
Credit memo+unknown	7	0,174%
Invoice+unknown	7	0,174%
Intercompany invoice+unknown	3	0,075%

87% of the Cases start with an Order

Figure 4 - ProM Event Log Summary: Start Events

End events

Total number of classes: **14**

Class	Occurrences (absolute)	Occurrences (relative)
Invoice+unknown	3036	75,466%
Returns delivery for order+unknown	262	6,513%
Intercompany invoice+unknown	242	6,015%
Credit memo+unknown	194	4,822%
Debit memo+unknown	89	2,212%
Delivery+unknown	60	1,491%
Credit memo cancellation+unknown	49	1,218%
Credit memo request+unknown	31	0,771%
Returns+unknown	29	0,721%
Intercompany credit memo+unknown	12	0,298%
Invoice cancellation+unknown	11	0,273%
Pro forma invoice+unknown	5	0,124%
Order+unknown	2	0,05%
Debit memo request+unknown	1	0,025%

75% of the Cases end with an Invoice

20% of the Cases end with Returns, Intercompany Invoices, Credit or Debit Memos

Figure 5 - ProM Event Log Summary: End Events

End events

Total number of classes: **96**

Class	Occurrences (absolute)	Occurrences (relative)
BDE_BATCH	699	17,375%
BFR_BATCH_SB	620	15,411%
BFR_BATCH_BL	404	10,042%
BITSDALL	303	7,532%
BCH_BATCH	297	7,383%
BDEEVOGE	207	5,145%
BES_BATCH	189	4,698%
BFRBATCHSBIV	130	3,231%
HPTTBAST	104	2,585%
BDEAMAN	86	2,138%
BFRBATCHBLIV	77	1,914%
BFI_BATCH	77	1,914%
BFRGSTEP	73	1,815%
BUK_BATCH	63	1,566%
BFRSLUX	60	1,491%
BDEAHOR	52	1,293%
BDEJKLIN	48	1,193%
BCHFKURZ	31	0,771%

More than 60% of end events are performed by batch users. This is expected, since most Invoices are created by a batch job

Figure 6 - ProM Event Log Summary: End Events Users

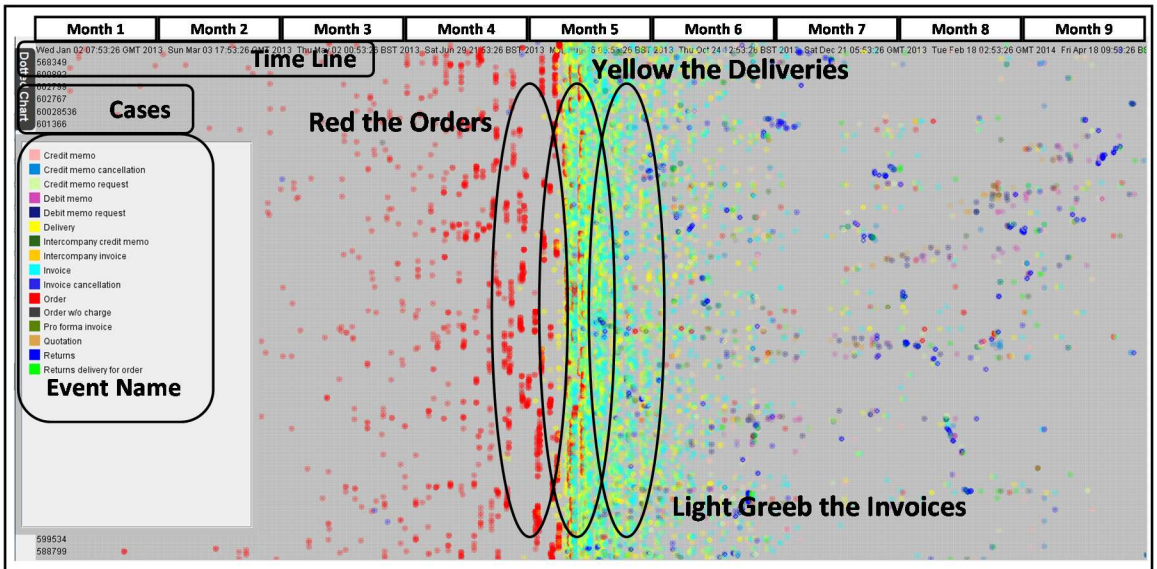


Figure 7 - ProM Event Log Dotted Chart: Unordered

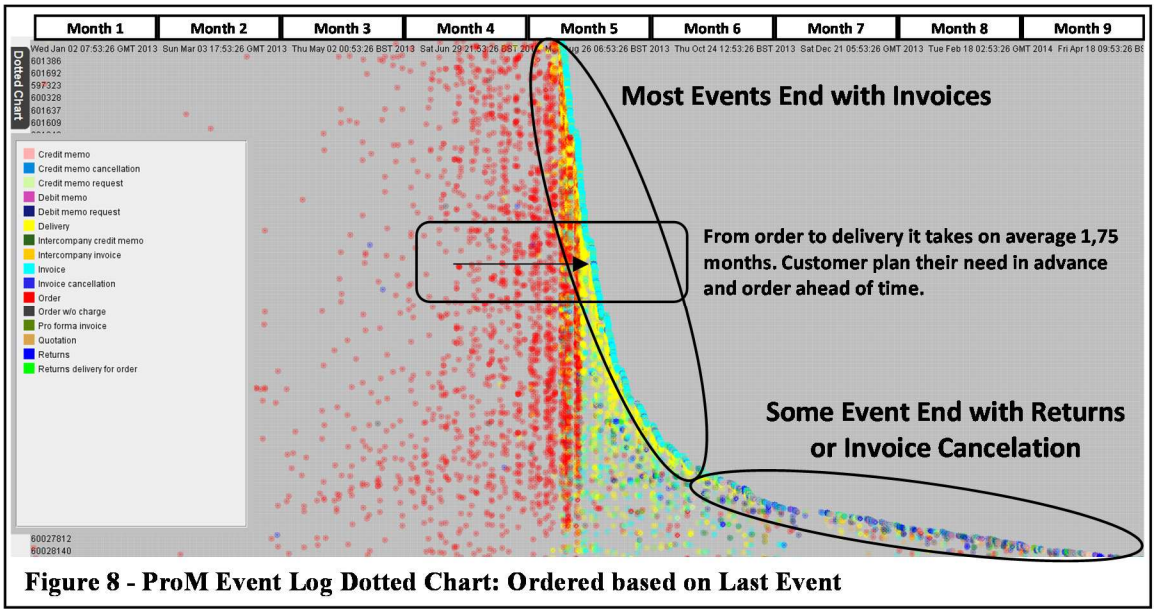


Figure 8 - ProM Event Log Dotted Chart: Ordered based on Last Event

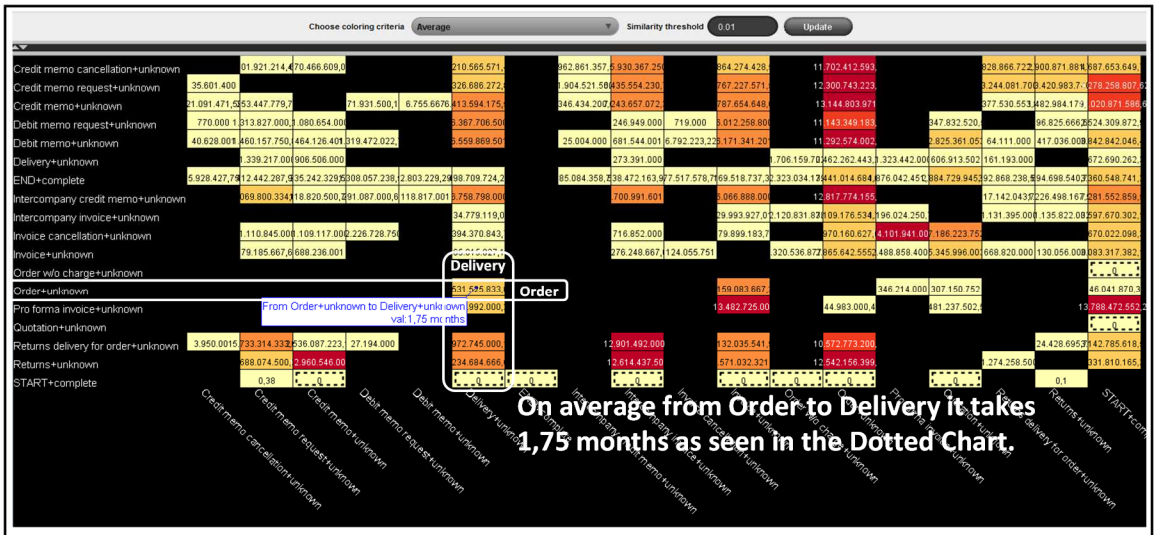


Figure 9 - ProM Event Log Synchronous Activity Analysis

9. Mine a Process Model:

In this case study we were able to run several discovery algorithms, specifically the Fuzzy Miner and Heuristics Miner. The Heuristic Net was converter to a Petri Net using the conversion plugin. Also we are able to compare the results before and after adding the artificial start and end events and also the difference of the output when certain parameters like significance are adjusted. Here are the results:

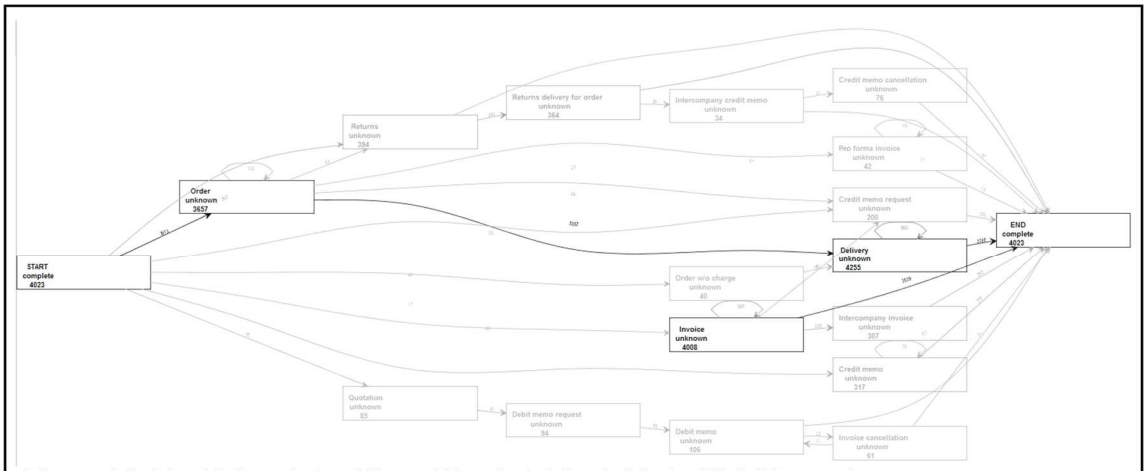


Figure 16 - ProM Heuristics Miner: Heuristic Net (with Artificial Events)

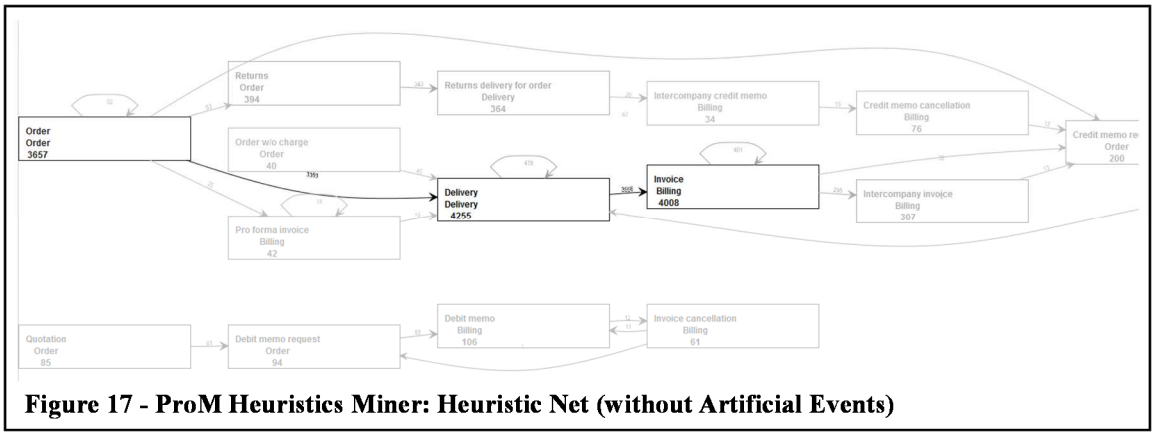


Figure 17 - ProM Heuristics Miner: Heuristic Net (without Artificial Events)

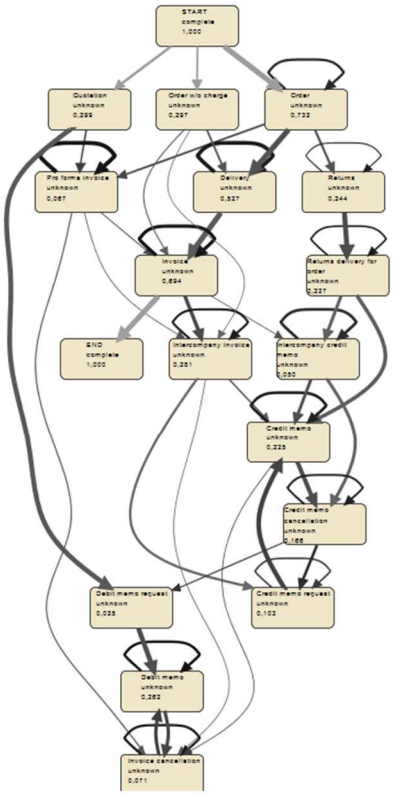


Figure 18 - ProM Fuzzy Miner: Mined Model (No Adjustments)

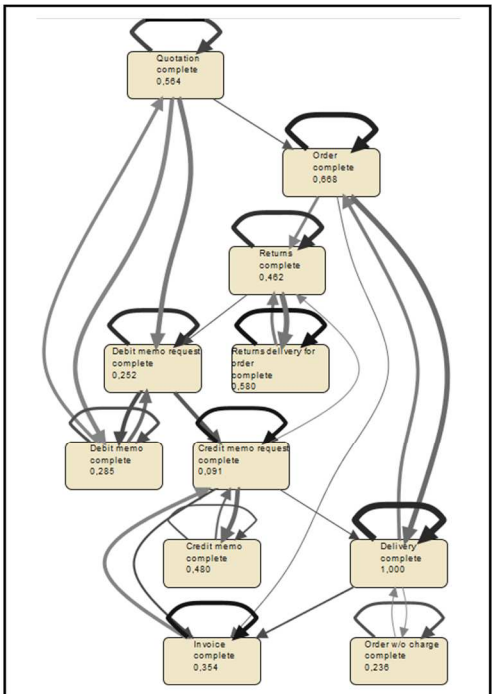


Figure 19 - ProM Fuzzy Miner: Mined Model (Without Artificial Events)

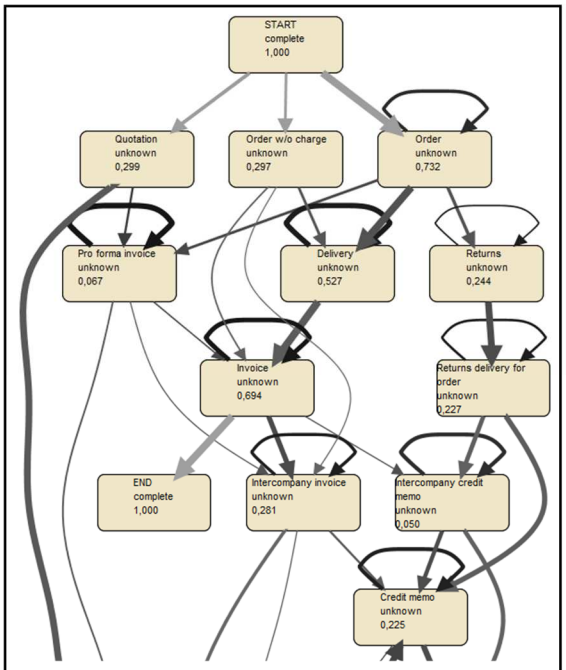


Figure 20 - ProM Fuzzy Miner: Mined Model (No Adjustments Close Up)

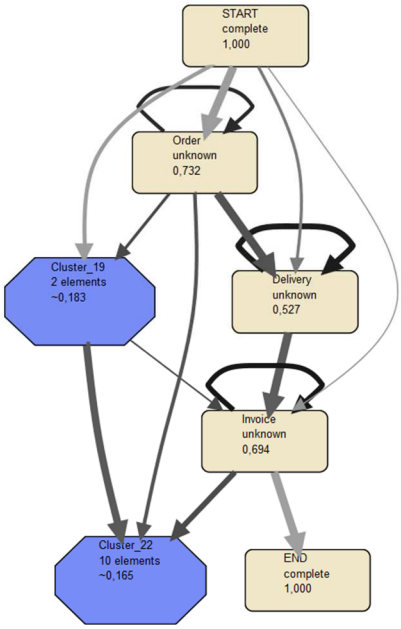


Figure 21 - ProM Fuzzy Miner: Mined Model (Significance CutOff)

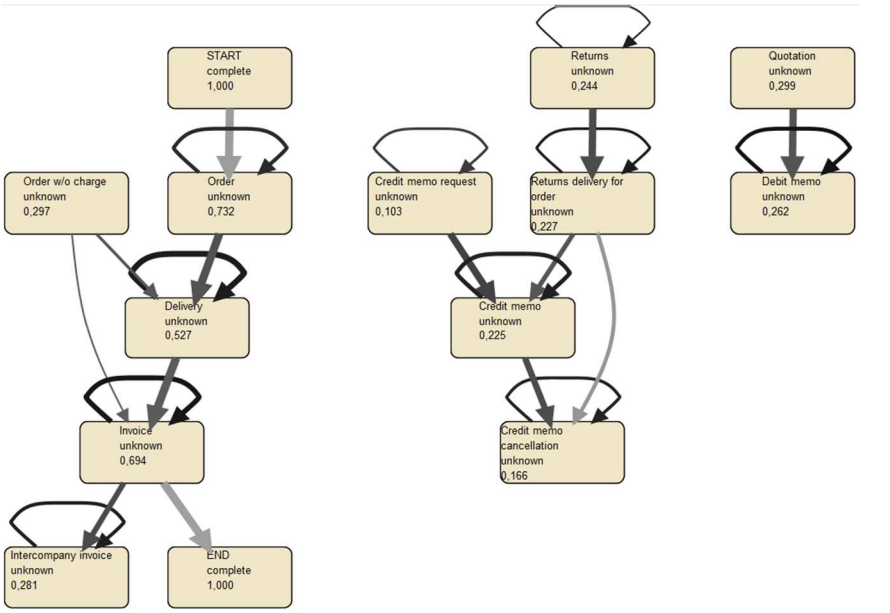


Figure 22 - ProM Fuzzy Miner: Mined Model (Broken Up)

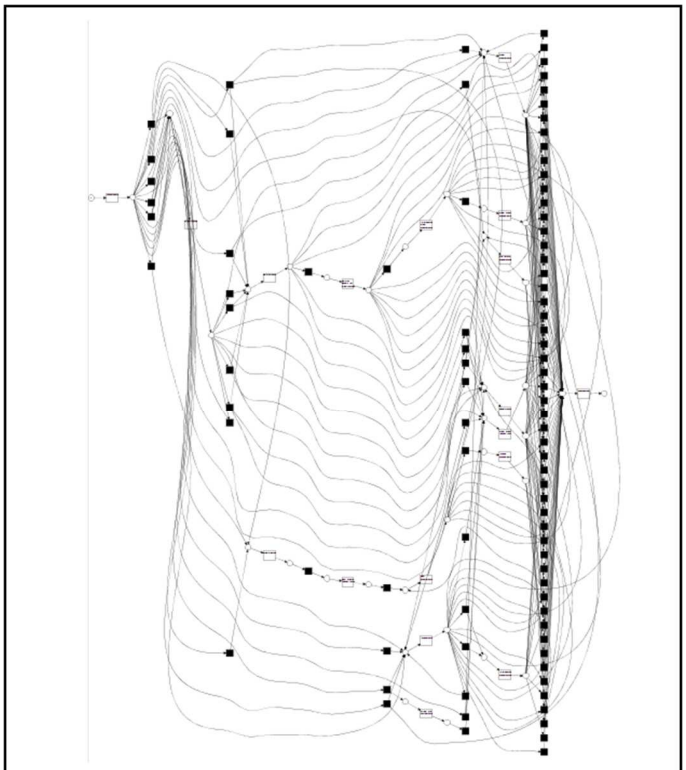


Figure 23 - ProM Heuristic Net to Petri Net Conversion

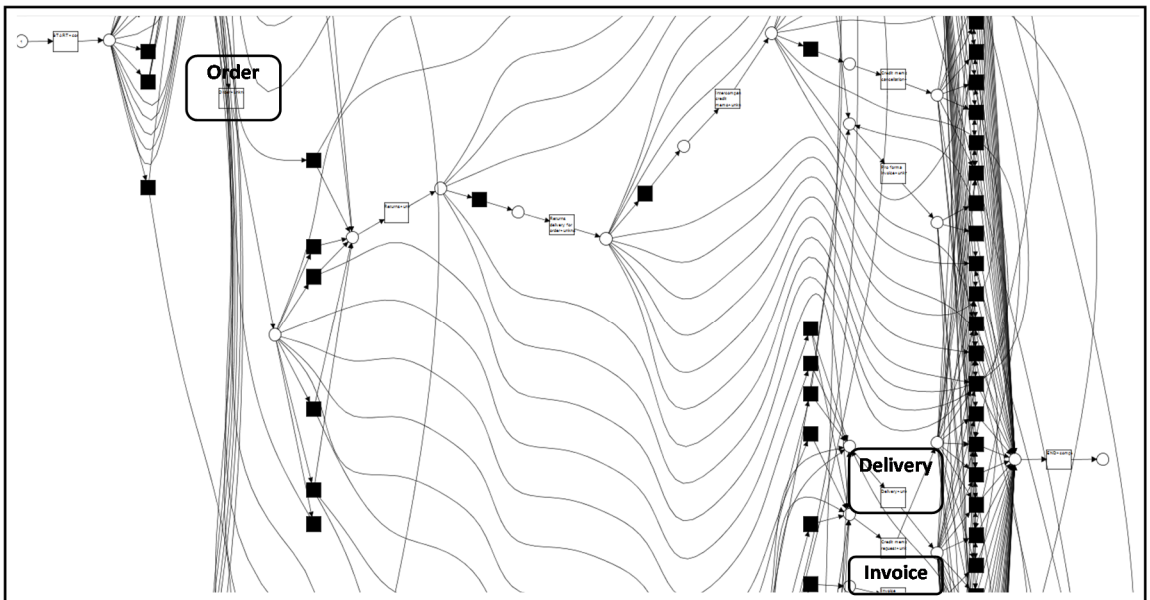


Figure 24 - ProM Heuristic Net to Petri Net Conversion (Close Up)

10. Animate the Process Model:

After using Fuzzy Miner run the Select Best Fuzzy Instance plugin and then the Fuzzy Animator to produce an amusing of your process flow to show the execution of the different cases in the time

frame you selected. Change the speed and look head parameters to improve the visibility of certain events. Here are some shots of the animation based on this case study data:

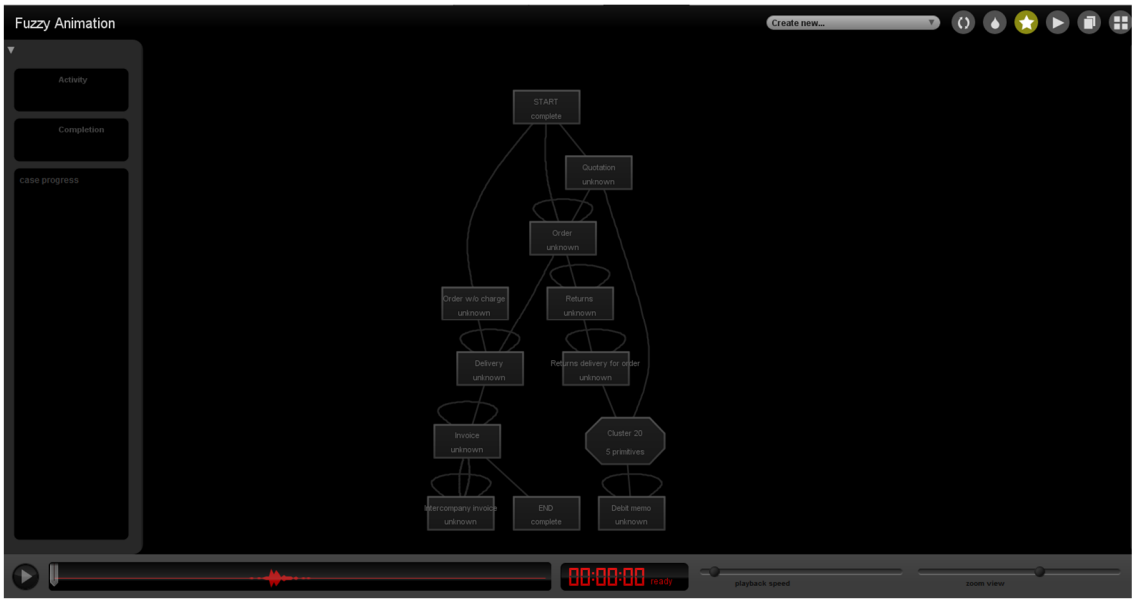


Figure 25 - ProM Fuzzy Animation (Beginning)

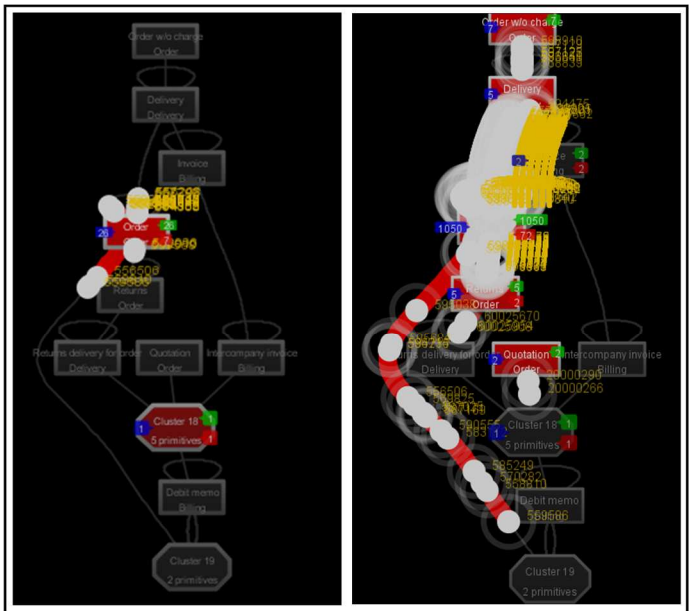


Figure 26 - ProM Fuzzy Animation (Running)

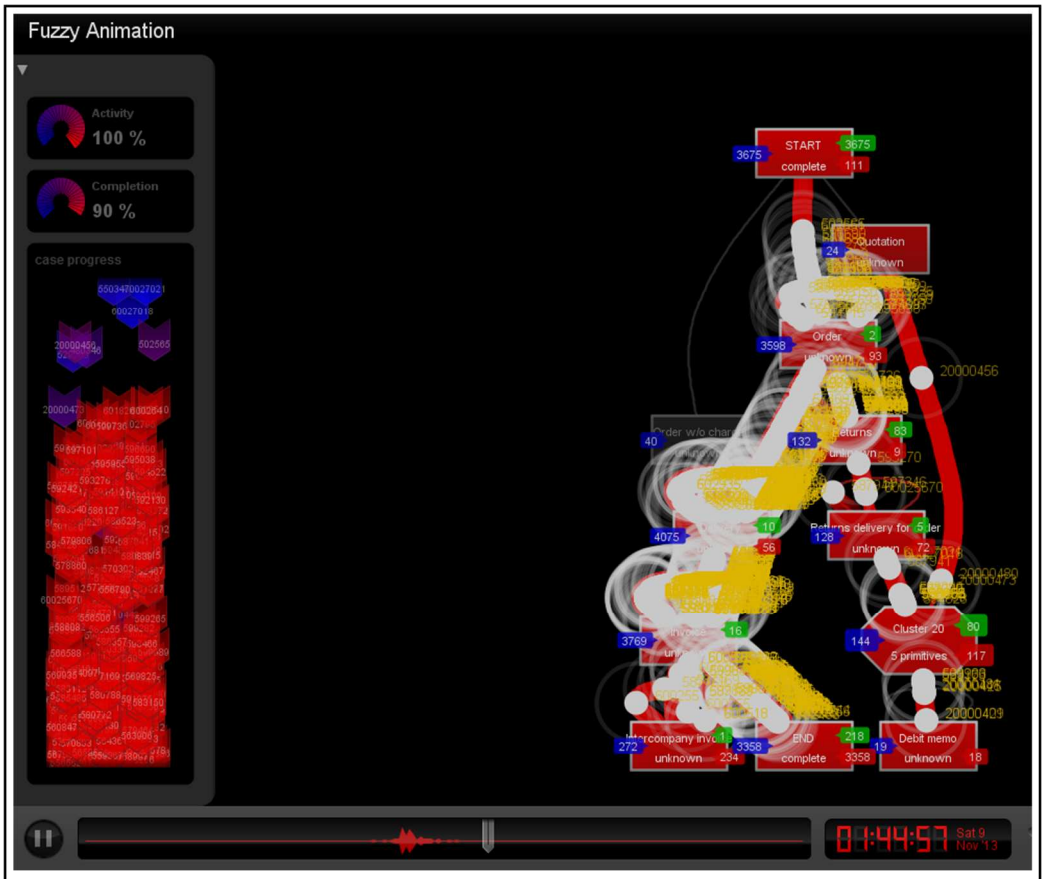


Figure 27 - ProM Fuzzy Animation (Explosion of activity)

This type of animation clear gives some visual clues about the process flow and also the existence of workload level picks during the analysis period.

7. Conclusions

7.1. Introduction

In the last months many lines of SAP sales data have been covered and mined. For this Process Mining application, SAP served as the core data source and the Sales and Distribution module the focus of all the analysis. The sole purpose was to extract raw sales data from the system, map and transform it into an event log that could be read by ProM Process Mining tool. In previous chapters we've covered the basic functions of ProM, the algorithms and packages available and the different outputs it produces.

7.2. ProM Review

Reality is dynamic, independent and complex, Powerpoint and Visio diagrams are bias, too simple and static. Process Mining techniques may top uncover the "real" process flow as shown below:

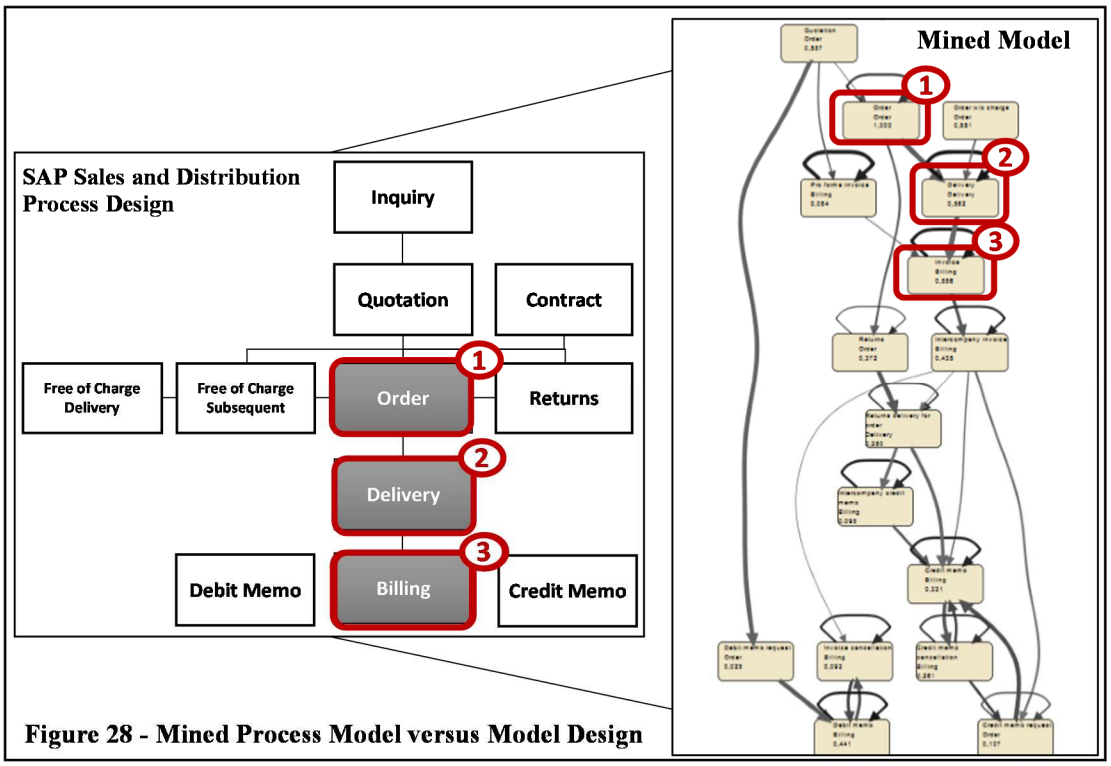


Figure 28 - Mined Process Model versus Model Design

Before implementing a tool, software or system we should first try to understand if it is worth to take the endeavor. That is, if the benefits of the implementation are greater than the cost of putting it in place. Why isn't every business and management consulting company using this type of tool to understand their customer's business process? Why aren't all auditors, compliance officers, quality

certifiers and business managers using this type of tool to understand, monitor and improve the company's processes?

So why hasn't Process Mining been fully disseminated and successful (yet)? After designing and producing an event log from a business management system is it required to implement a Process Mining dedicated tool? The objective of the tool should be to help the analyst discover the business process without being dependent on previous and subjective understanding and knowledge of people executing it. But in reality you would need to understand the business process model before you could really extract the data or create an event log, interpret the process instances and event attributes and review the data driven process model from a Process Mining tool. So this means at least the subjective understanding of the process and the system that supports the process and the objective data driven method of process mining would be complementary, not substitutes as many papers would insinuate when trying to sell the idea.

To extract the data from a system you would need to understand the process and how the system records the events for each process instance and their relevant attributes. It is not a blind process of picking a black box of data and magically the tool interprets it for the user. You would need to install or configure an audit log or document flow feature before you could create an event log or alternatively build your own event log based on business transactional data and document flows using SQL type queries. So the argument that the tools are easy to use and you could autonomously create your own business process analysis based solely on available unstructured event data and a tool that does it all is not realistic or even true. To determine event correlation, associate every event to a single case or process instance, identify and capture all of the relevant data is something that requires in-depth analysis and process knowledge. Yes, Process Mining tools could help to facilitate the job. Even after we extracted the data and produced a process model in ProM we still need to make adjustments before the final output is ready to be used.

Maybe that is one of the major obstacles and why this type of solutions are not widespread through all the organizations. Most internal audit, compliance and quality departments still rely on interviews, observation, walkthrough type meetings and basic data analysis like journal entries and audit log reviews to perform their jobs. Business consultants still apply the same method as well.

Additionally, ERP solutions like SAP already have built-in some tools and modules to help understand and monitor the process, like Process Observer and Business Process Monitoring / Business Process Analytics (BPMon/BPA). Even if you only need the basic information about the process, like what is the average and most frequent process time, the average or most frequent

number of events till a process instance ends or even detect patterns and associations to make predictions and prescriptions and finally identify bottlenecks you wouldn't need a complex and totally different new tool. With a sample of related events in the system, knowing how to extract the data and having some reasonable ability with SQL or even Access Queries you could do the trick. Even cleaning up the missing links and creating some machine learning processes using some programmable scripted routines is possible. Creating the diagram or process representation itself would be the hardest trick to pull with the normal available tools, but there are free tools like Aris Express that perform that task. With that said we can still argue if the effort and cost of implementing it is worth the trip.

The main idea is that process discovery is not a blind task. Most of the authors would say that using standard data queries you would need to know what to look for, regarding the standard data query tools. Not true, there are open exploratory questions and closed specific questions. When reviewing records, like when interviewing someone, because we don't know what we are looking for, we start with open questions, from those we get clues, and then try to investigate more specific topics that were highlighted by the first queries, in an iterative knowledge building process. Event data and event logs in ERPs are not random. They resulted from a process implementation that needed to be configured. The tables where the records are kept follow specific structures and the whole process that the ERP supports was in the mind of the developers. It was not a blind construction. The system knows where to put the records that result from specific process transactions and those are kept in specific process oriented archives, clusters of tables that are somewhat process oriented. SAP like most ERP systems contains Accounts Payable, Accounts Receivable, General Accounting, Sales Documents, Delivery Documents, Billing Documents, Planning and Production Records, Warehouse Movements, Purchasing Records, etc. This are modules or clusters where records are put in a logical order and sometimes even connected to each other. Each Goods Receipt from Purchasing is mapped to a related movement in the Warehouse Movements table, it's consumption in production is reflected both in the Planning and Production and Warehouse Movements tables and in the end when the finished goods is shipped to the customer a similar but now Goods Issue document is created with Delivery and Billing Document.

7.3. Process Mining Alternatives

Based on some experience as a consultant and data analyst for the last 8 years I would have some difficulty proving right away the benefit of this type of tools to any company, as it stands right now. It can be argued that most of the benefits can be obtained and deployed more quickly using already

available standard tools or even internally developing a low cost solution to fulfil this information gap. For example, creating a process flow diagram based on an activity or task index or dictionary that could be inferred from the event log can be accomplished with some simple programming.

With this said, some of the features suggested by Process Mining can be performed at some level by other tools or techniques. The question is then made: Is there really a need for another tool and area of expertize? Does that explain the delay in the implementation of this type of tools?

Taking a look at the business requirements, features and major outputs of Process Mining tools we may conclude that some or most of them can be solved using or developing customized tools. The next features can be easily managed with without a Process Mining dedicated tool:

Process Discovery

1. Draw process diagram based on data: pick up an event log, summarize an activity dictionary and use available tools to draw the process diagram (e.g. Aris Express)
2. Determine process metrics using simple functions in standard database queries

Conformance Checking:

1. Check for approvals and validate weird transactions (e.g. round numbers, holidays or late night entries) using Journal Entry and Data Analysis techniques
2. Validate task sequence for critical tasks or events (e.g. each customer order over 10K€ has credit checking)
3. Review system process controls, that is, if gates and user limitations are in place
4. Check user access profiles and event originators for critical tasks

Monitor Process Execution:

1. Determine online process metrics in running queries and compare them against previous calculated metrics and targets
2. Create process alerts and flags if thresholds have been compromised using dashboard and e-mail notifications
3. Recommend actions and alternative paths to finish process based on contingency set of actions planned and defined previously

Are the Process Mining tools able to really prove the benefit is greater than the cost, that is, it is worth the pain of understanding new methods rather than using already known methods or even the standard Data Mining methods?

My advice to every person interested on this topic is to start with Data Mining concepts, focus on Event Correlation and/or Causality Inference topics, acquire some standard query tools skills, for

example, SQL Server, for optimized and virtually unlimited usage, and MS Access, for quick and easy deployments (very useful to create prototypes and proof of concept type of results). Also and of course Process Modelling, namely understand the business model, the objectives and purposes of modelling, how to map processes to comply with those requirements and the different approaches (e.g. different process flow notations). Only then, if none of the previous subjects was sufficient to answer all the questions, look at the Process Mining algorithms, understand what they accomplish, theirs features, their requirements and the way they work. In this process you may come to conclude you can simply understand how the different process mining approaches fix the obvious type of issues like missing event traces or links (process gaps), noise (bad data or outliers), concurrency (possibility of two event happening in parallel), or loops (the never ending process, activities that depend on themselves, repetition of the same activity in the same process instance in different moments in time), splits, join, disjunction (two or more alternatives) and conjunctions (two or more activities working for the same process), among many other process related topics. After understanding the methods and techniques you may find that the tools you already have can fix the problem. Maybe this is too simplistic, but simple and result oriented solutions is what the business needs and wants. Nobody likes a joke that needs explanation, even if it is the best joke in the world.

7.4. Methodology Review

Is Process Mining more than it says? Is it biting more than it can chew?

There are many inconsistencies in the whole selling argument of most of the sponsors of Process Mining tools. One of which, and probably the most problematic, is the idea of discovering a process without any previous knowledge of how the process should look like. This is a bit far fetch. You need to select a process, understand it’s relevancy to the organization, you need to find out how it is executed and recorded in the system or systems. You need to identify relevant data, and by relevant meaning you've asked the right questions and therefore need to have some previous knowledge of the process. Extracting the data may mean you need to develop or configure an event log, with that said you need to follow the data trace to capture the full length of events relating to the case study. The process is intrinsic in the systems that support them, by nature and default, else they wouldn't support them at all. In the remote chance you could select a process, ask the right questions, identify the data source and extract relevant information without any previous in depth knowledge of the process being studied you would then need to load and review the event log. It would be quite difficult to have an opinion or educated guess with no knowledge and therefore be able to really evaluate the quality of the extracted and imported data, the quality of the conversion process the

completeness, accuracy and validity of the event log. One more time, if again you would be able to perform this task effectively without any profound knowledge of the process, the systems, the business and the data, then you would need to make several decisions to enhance, convert or clean the imported Event Log, or to decide not to do anything at your own risk. You would identify the best algorithm or plugin to discover the process without understanding the existence of concurrency, noise, loops, complexity and granular events, etc. In the end, if by luck, chance or suggestion you were able to select a good plugin (e.g. Fuzzy Miner) you would still have to adjust several parameter to have a good view of the process. Some algorithms need the parameters definition before the plugin is run, others may allow you to adjust on the fly. At this point there would be no chance, even by trial and error, you would be able to evaluate the fitness, precision, simplicity and generality of your output, without knowing the process or studying it before. To end this painful walkthrough, you would have stopped in the data extraction and conversion bit, this is enough for the user to find the need to really understand fully the whole scope of the process and the details of what is recorded or not recorded and where and how, who is involved, and what available information there is elsewhere. Even if you would have time to repeat the whole process 1000 times you would probably never get it right or you would have no way to know when you were right.

The tools help, but there is much work before and beyond applying Process Mining techniques. It is nor the start, nor the end of the Process Improvement and Business Intelligence endeavors.

7.5. User Experience

Perhaps ProM tool is not the most user-friendly experience you'll ever find. There seems to be a lot of focus in having a lot of plugins and a lot of options and parameters and unknown interconnections and dependencies between plugins. In this case, less would be more. We can conclude that for the 99,9% of the people that are not Computer Scientists and/or Data Miners working with this tool will be a very hard and painful experience.

On recommendation would be to have two or three environments or versions, one for developers, one for testing new stuff and one called production with only the robust and already tested models. This last environment should only have 5 to 10 options available. The options that are related, like cleaning a log or fixing it from noise and missing events should appear before the options to run a process discovery algorithm. Finally following this last one all the options to convert the process representations notations between Petri Net, Heuristic Net or BPMN or to execute performance analysis based on the mined model.

Basically there is insufficient guidance and too many options. It is like working in the dark, putting in some data, trying out randomly, based on the smartest name or some literature reference, the plugins or algorithms. These are all in the same list although they perform different tasks. Maybe they should be categorized based on what they accomplish (e.g. discovery, conformance, data analysis) and/or hierarchized and structured based on their particular function (e.g. data cleaning, process discovery, process model conversion).

Besides this, there is no comprehensive list of the existing packages or plugins and what they accomplish. You would need to read the dissertations, papers, presentations or any other document by developers, and even then you may not have the basics covered.

The descriptions or explanations of the parameters to configure and run a package or plugin are scarce, complex or non-existing at all. If there are some they are incomplete and difficult to get to.

Additionally, ProM free version may appear somewhat instable. For example, in my case it wouldn't run in every computer. Sometimes it freezes and shuts-down when starting and even when running the available plugins some of them just don't run and return insignificant errors messages like "java.lang.reflect.invocationTargetException" that basically doesn't say anything for the normal end user.

At some point, it may feel like an arcade game. Graphically it as improved a lot and even in the user experience, but it is still a rudimentary tool, nonprofessional tool like we would sadly expect from most of the open source academic collaborations tools. The things is, most students and professionals don't like them and get a bad first impression. Most likely they will never pick it up again after their first experience, namely if they have a chance to do something better and easier. This is based on the path of less resistance logic. Sometimes it is hard to understand if it is the sport that makes the athlete or the other way around. For sure if the first experiences you have with something are so bad, if you really don't perform, your mind will send a message to the body not to waste any more energy or time on the subject and move on, to protect the self-esteem and ultimately be more efficient working on a subject were you would perform better. With this said, it would be better to simplify the public version of the tool and make the experience as smooth and rich as possible. If then the user wants to deep dive, having a boost of confidence on the first run, he could access the full list of plugins and options for more advanced analysis.

Beside all of this, the tool also behaves like an arcade game because every time the tool is closed we lose the history or at least that is what happens in every computer it was run at. You would have to save every single output produced and import it every time if you wanted to reuse it. So it is like you

have to start from scratch every time, there is no idea of project. Perhaps there is an explanation in a book, dissertation, working paper or forum about that topic, but again, it is not clear or direct and it results in more pain for the user. The frustrating thing is to find that some basic topics are not cover widely and sometimes you find something useful very late in your work because the guidance is scarce.

Finally, like in an arcade game we also don't know the purpose or were we are heading, everything is a surprise. The tutorials and package descriptions are very basic, so when you try the tool for yourself you won't find answers. The plugins have no detailed full explanation of the features. You basically click on everything, try every option, change the settings and try to understand the outcomes of it, in a trial and error process. You go to forums and wikis just like you would to in RPG games to find were the secret door is and how to grab the special solution you need to obtain your objectives.

In the beginning it is not clear which algorithm is correct for each case or type of event log. For example, it seems at first the Heuristics Miner is not the best option for SAP Sales and Distribution process event logs, for example. Not acknowledging that each delivery item may be related to one order item for the same case, but only looking at the timestamp of the events for the same case the algorithm may determine that every delivery line is related to each other, but in reality they may be related to the first event that is the order line. This approach wouldn't be correct, it is just a split from the first event, one starting event producing two or more (up to 6 or 10) new events. So several events result from one and are performed in different moments in time, but not correlated between themselves, only correlated against the starting order line. The tool doesn't offer a full user guidance and the tutorials don't go that far to explain the full picture. Even the papers seem to go around the subject in theory, with complex notations and formulas, but avoiding and never showing how to put it in practice.

Another topic is that the requirements of the Event Log input files are not at all clear for each plugin nor is the output it produces and the implications of not having the right information in the right order or with the right structure or something else. A plugin may need the events to be ordered before creating the MXML import file, on the other hand the Heuristics Miner does not look at the Event ID at all, it only needs the Event Timestamp to discover the process.

The reason why some packs or plugins are unavailable from time to time (color orange) is not clear or explained before using the tool. For the user it is an unknown dependency between a precursor algorithm and the unavailable one, you need to follow the path and try to make new features available on your way. There may be some event log requirements that are not being fulfilled or we need

additional information. For example, sometimes you need an Event Log combined with a Process Model. It is not clear and it is frustrating to not know or understand at least the main structure and how to run a good Process Model based on a specific type of Event Log, convert this Process Model, calculate some performance metrics on it, animate it and review the process. It should be that simple. So yes, you'll need to invest a big amount of time to execute something for which you may not know the real benefit of or if it fits your own case study and business requirements.

To sum it all up, it is not a commercial solution, but it also isn't a good publicity for Process Mining. It is only an academic tool in which most of the discovery processes don't give much information about the process in the first run, at least there won't be any big surprises if you already know your processes and data.

7.6. Further Investigation

One of the biggest doors opened for further investigation in association and collaboration with Process Mining is the concept and methodology for Event Correlation and Causality Inference.

Event Correlation and Causality Inference aim to discover meaningful causalities between events. When we are missing event flows or traces we need to infer the causality or correlation between existing events based on the moment they happen and other characteristics. For example, in SAP it is not mandatory to associate a Credit Memo Request (Complaint) and the potential resulting Return to an existing Order or Invoice. It would be extremely helpful to have the full picture of the process end-to-end and bridge those missing trails between events. Although these two events are not connected we may infer they relationship using causality inference methods. If it is a full return the quantities will match, the product and customer entities must be the same and the return should happen in a reasonable period of time from the shipping, say 3 months. The time for a complaint can even be limited by contract. We could theoretically associate both events in the same process instance. Additionally, with more complexity, if there is a price adjustments, for example we had invoiced at a higher price than agreed with the customer, another Credit Memo Request is created but no return associated. In this case we could cross check Price Lists or latest or more contemporaneous prices to, first of all trigger that the first transaction or flow of events would potentially result in a price adjustments (e.g. different price from Price List or from latest price) and second if there exists a price adjustment in the near future for that product or customer we could associate the first transaction and document with the new event (Credit Memo Request) and therefore connect it to the rest of the flow. In the case of a Return, if the production batch is filled in the record it wouldn't be necessary to create a complex algorithm to associate this two events, but it would still

be necessary to do some programming, namely develop recursive algorithm to check every potential transaction and establish the trail when all of the defined criteria were met.

7.7. Summary

In nutshell, this case study hopefully brought some light and insight on the application of Process Mining techniques. It is a practical user oriented approach on the topic that may be completed and improved in the near future. The list below is a summary of the most relevant steps and achievements of this work, most of which are key to anyone starting on this subject:

1. Understand the general sales order process flow;
2. Understand the sales order process flow in SAP: how system records process;
3. Identify the key tables with relevant information for the sales process flow in SAP;
4. Extract and map relevant system tables in a relation database (Access);
5. Create ETL process: mechanism to produce an Event Log from a Document Flow and Data;

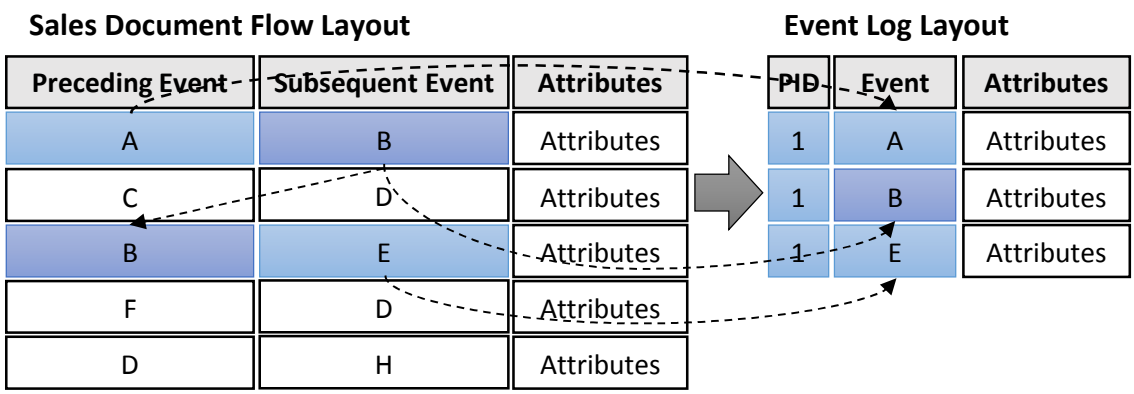
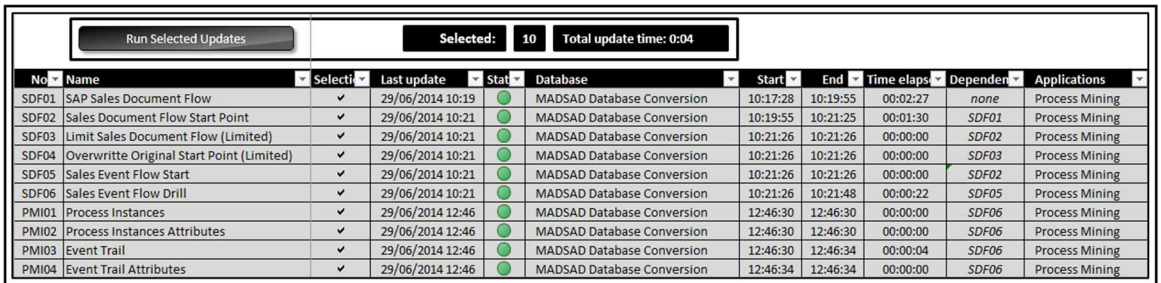


Diagram 21 - Sales Document Flow transformation into Even Log

6. Automate ETL process: developed a script that builds an event log table from the SAP tables;



No	Name	Selecti	Last update	Stat	Database	Start	End	Time elaps	Dependen	Applications
SDF01	SAP Sales Document Flow	✔	29/06/2014 10:19		MADSAD Database Conversion	10:17:28	10:19:55	00:02:27	*none*	Process Mining
SDF02	Sales Document Flow Start Point	✔	29/06/2014 10:21		MADSAD Database Conversion	10:19:55	10:21:25	00:01:30	*SDF01*	Process Mining
SDF03	Limit Sales Document Flow (Limited)	✔	29/06/2014 10:21		MADSAD Database Conversion	10:21:26	10:21:26	00:00:00	*SDF02*	Process Mining
SDF04	Overwritte Original Start Point (Limited)	✔	29/06/2014 10:21		MADSAD Database Conversion	10:21:26	10:21:26	00:00:00	*SDF03*	Process Mining
SDF05	Sales Event Flow Start	✔	29/06/2014 10:21		MADSAD Database Conversion	10:21:26	10:21:26	00:00:00	*SDF02*	Process Mining
SDF06	Sales Event Flow Drill	✔	29/06/2014 10:21		MADSAD Database Conversion	10:21:26	10:21:48	00:00:22	*SDF05*	Process Mining
PMI01	Process Instances	✔	29/06/2014 12:46		MADSAD Database Conversion	12:46:30	12:46:30	00:00:00	*SDF06*	Process Mining
PMI02	Process Instances Attributes	✔	29/06/2014 12:46		MADSAD Database Conversion	12:46:30	12:46:30	00:00:00	*SDF06*	Process Mining
PMI03	Event Trail	✔	29/06/2014 12:46		MADSAD Database Conversion	12:46:30	12:46:34	00:00:04	*SDF06*	Process Mining
PMI04	Event Trail Attributes	✔	29/06/2014 12:46		MADSAD Database Conversion	12:46:34	12:46:34	00:00:00	*SDF06*	Process Mining

Diagram 22 - Automated ETL Control Dashboard

7. Transform resulting event log into the ProM Import table's format;
8. Convert the original event log into the ProM Framework input MXML file;
9. Upload event log and execute several process mining algorithm;

10. Debug and review the process. The corrections and optimizations include:
    a. Case sensitive sales document categories: use string compare function to join tables;
    b. Create unique event key based o preceding and subsequent document;
    c. Define end of process instance based on “no subsequent event rule”;
    d. Solve or exclude inconsistencies in document flow (e.g. father is son of son).
11. Understand ProM basic functionalities: main algorithms, packages and outputs;
12. Create sampling method: limit number of process instances to facilitate first exploratory analysis;
13. Review weird process instances (e.g. sales returns not allocated to original sales document);
14. Map and add additional attributes based on transactional data (e.g. timestamp, originator);
15. Review and explain results of Process Mining algorithm application;
16. Document the whole process from MXML file to Process Model;
17. Understand the limitations of transactional data for process mining;
18. Define the most critical step on this process: ETL process is 90% of the workload;
19. Identify the best sources of data: the ERP is one of the most reliable sources of information;
20. Summarize experience, opportunities and alternatives.

One of the biggest challenges in Process Mining and this particular case study is data extraction and conversion, from transactional data to the required ProM Import format, if this is the method of choice to create the Event Log MXML file for ProM.

Understanding the interlinks and the features of Process Mining techniques and how they are applied in the ProM tool is the second biggest challenge. There is a lot of information about individual subjects, some of it not user oriented, disperse in several papers and presentations. Surely hope that this paper creates some value on this subject and helps to development some common understanding, by compiling most of the existing information and turning it slightly more end user focused and friendly.

References

Adriansyah, A et al. (2011), "Conformance Checking using Cost-based Fitness Analysis" Enterprise Distributed Object Computing Conference 15th IEEE International (pp 55-64)

Goedertier, Stijin et al. (2008), "Process Mining as First-Order Classification Learning on Logs with Negative Events" Business Process Management Workshops Lecture Notes in Computer Science, Volume 4928 (pp 42-53)

Medeiros, A.L. Alves (2003), "Workflow Mining: Current Status and Future Directions" On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE Lecture Notes in Computer Science, Volume 2888 (pp 389-406)

Mendling, J. et al. (2006), "Errors in the SAP Reference Model" BPTrends (pp 1-5)

Mendling, J. et al. (2006), "A Quantitative Analysis of Faulty EPCs in the SAP Reference Model" BPM Center Report (pp 6-8)

Nakatumba, J. et al. (2009), "Analyzing Resource Behavior Using Process Mining" Proceedings of the 5th International Workshop on Business Process Intelligence, BPI

Rozinat, Anne et al. (2006), "Decision Mining in ProM" Business Process Management Lecture Notes in Computer Science, Volume 4102 (pp 420-425)

van der Aalst, W.M.P. (2000), "Process Design by Discovery: Harvesting Workflow Knowledge from ad-hoc executions" ACM Conference on Computer Supported Cooperative Work (pp 2-6), USA

van der Aalst, W.M.P. (2003), ""Process Mining: The next step in Business Process Management"" Australasian Process Days (Keynote)

van der Aalst, W.M.P. (2005), "Process Mining Discovering processes from event logs" CTIT BPM day, University of Twente (Keynote)

van der Aalst, W.M.P. et al. (2005), "Process Mining" Process-Aware Information Systems: Bridging People and Software through Process Technology (pp 235-255) Wiley & Sons

van der Aalst, W.M.P. et al. (2007), "Business Process Mining: An Industrial Application" Information Systems 32(1) (pp 713-732)

van der Aalst, W.M.P. et al. (2007), "Business Process Analysis with ProM" Seventeenth Annual Workshop on Information Technologies and Systems (WITS'07) (pp.223-224), Montreal, Canada

van der Aalst, W.M.P. (2009), "Process Mining: Beyond Business Intelligence" Gartner Business Process Management Summit, London (Keynote)

van der Aalst, W.M.P. (2010), "Beyond Process Mining : From the Past to Present and Future" Advanced Information Systems Engineering, Proceedings of the 22nd International Conference on Advanced Information Systems Engineering (CAiSE'10), volume 6051 of Lecture Notes in Computer Science, pages 38-52, Springer-Verlag, Berlin

van der Aalst, W.M.P. (2010), "Process Discovery : Capturing the Invisible" IEEE Computational Intelligence Magazine, 5(1) (pp 28-41)

van der Aalst, W.M.P. (2010), "Beyond Process Mining: From the Past to Present and Future" 22nd International Conference on Advanced Information Systems Engineering (CAiSE'10). June 2010, Hammamet, Tunisia

van der Aalst, W.M.P. (2011), "Using Process Mining to Bridge the Gap between BI and BPM" Using Process Mining to Bridge the Gap between BI and BPM. IEEE Computer, 44(12) (pp 77-80)

van der Aalst, W.M.P. (2011), "Process Mining: Discovery, Conformance and Enhancement of Business Processes" Springer, Berlin

van der Aalst, W.M.P. (2011), "Process Mining (Manifesto)" Keynote Lecture Multi-Agent Organisation (MAO), Leiden

van der Aalst, W.M.P. et al. (2012), "Process Mining Manifesto" Proceedings of the BPM 2011 Workshops, Part I, Volume 99 of LNBIP (pps 169-194), Springer-Verlag, Berlin

van der Aalst, W.M.P. (2013), "Process Mining Manifesto (flyer)" IEEE Task Force on Process Mining

van der Aalst, W.M.P. (2013), "Decomposing Petri nets for process mining: A generic approach" Distributed and Parallel Databases, 31(4) (pp 471-507)

van der Aalst, W.M.P. (2013), "Mine Your Own Business: Using Process Mining to Turn Big Data into Real Value" Keynote 21st European Conference on Information Systems (ECIS 2013), June 6th 2013, Utrecht, The Netherlands

van der Aalst, W.M.P. (2013), "Process Cubes: Slicing, Dicing, Rolling Up and Drilling Down Event Data for Process Mining" Keynote Asia Pacific conference on Business Process Management, Beijing, China

van der Aalst, W.M.P. (2013), "Processes @ your Service: Using Process Mining to Turn Big Data into Real Value" Keynote International Conference on Web Engineering (ICWE), Aalborg, Denmark

van der Aalst, W.M.P. (2013), "Process Mining: A historical Perspective" Keynote Process Mining Camp, Fluxicon, Eindhoven

van Dongen, Boudewijn F. et al. (2004), "EMiT: A process mining tool" Lecture Notes in Computer Science, Vol. 3099 (pp 454-463), Springer, Berlin

van Dongen, Boudewijn F. et al. (2005), "The ProM Framework: A new era in Process Mining Tool Support" Lecture Notes in Computer Science, Vol. 3536 (pp 444-45), Springer, Berlin

van Dongen, Boudewijn F. (2005), "The ProM framework" ATPN 2005, Miami

van Dongen, Boudewijn F. et al. (), "Verification of the SAP Reference Models using EPC Reduction, State Space Analysis, and Invariants" Computers in Industry 58(6) (pp 578-601)

van Giessel, M. (2004), "Process Mining in SAP R/3: A Method for Applying Process Mining to SAP R/3" Master's Thesis. Technische Universiteit Eindhoven, Eindhoven, The Netherlands

Verbeek, H.M.W. et al. (2010), "ProM 6: The Process Mining Toolkit" BPM 2010 Demo

Verbeek, H.M.W. et al. (2011), "XES, XESame, and ProM 6" Lecture Notes in Business Information Processing, Vol. 72 (pp 60-75), Berlin: Springer

Weijters, A.J.M.M. et al. (2001), "Process mining: discovering workflow models from event-based data" Proceedings of the ECAI Workshop on Knowledge Discovery and Spatial Data (pp 283-290)

Weijters, A.J.M.M. et al. (2007), "Process Mining with ProM" Proceedings of the 19th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC)

Glossary

Activity	A well-defined step in the process. Events may refer to the start, completion, cancelation, etc. of an activity for a specific process instance
Business Intelligence	Broad collection of tools and methods that use data to support decision making.
Business Process Intelligence	A branch of Business Intelligence focusing on Business Process Management. Also called short Process Intelligence.
Business Process Management	The discipline that combines knowledge from information technology and knowledge from management sciences and applies both to operational business processes.
Concept Drift	The phenomenon that processes often change over time. The observed process may gradually (or suddenly) change due to seasonal changes or increased competition, thus complicating analysis.
Conformance Checking	Analyzing whether reality, as recorded in a log, conforms to the model and vice versa. The goal is to detect discrepancies and to measure their severity. Conformance checking is one of the three basic types of process mining.
Cross-Organizational Process Mining	The application of Process Mining techniques to event logs originating from different organizations.
Data Mining	The analysis of (often large) data sets to find unexpected relationships and to summarize the data in ways that provide new insights.

Deadlock	A deadlock is a situation in which two or more competing actions are each waiting for the other to finish, and thus neither ever does. In computer science a deadly embrace is a deadlock involving exactly two competing actions. It is a term more commonly used in Europe. In a transactional database[disambiguation needed], a deadlock happens when two processes each within its own transaction updates two rows of information but in the opposite order. For example, process A updates row 1 then row 2 in the exact timeframe process B updates row 2 then row 1. Process A can't finish updating row 2 until process B is finished, but it cannot finish updating row 1 until process A finishes. No matter how much time is allowed to pass, this situation will never resolve itself and because of this database management systems will typically kill the transaction of the process that has done the least amount of work.
Event	An action recorded in the log, e.g., the start, completion, or cancelation of an activity for a particular process instance.
Event Log	Collection of events used as input for process mining. Events do not need to be stored in a separate log file (e.g., events may be scattered over different database tables).
Fitness	A measure determining how well a given model allows for the behavior seen in the event log. A model has a perfect fitness if all traces in the log can be replayed by the model from beginning to end.
Generalization	A measure determining how well the model is able to allow for unseen behavior. An “overfitting” model is not able to generalize enough.
Institute of Electrical and Electronics Engineers	IEEE's core purpose is to foster technological innovation and excellence for the benefit of humanity. IEEE will be essential to the global technical community and to technical professionals everywhere, and be universally recognized for the contributions of technology and of technical professionals in improving global conditions.

Journal Entries	A journal entry, in accounting, is a logging of transactions into accounting journal items. The journal entry can consist of several recordings, each of which is either a debit or a credit. The total of the debits must equal the total of the credits or the journal entry is said to be "unbalanced". Journal entries can record unique items or recurring items such as depreciation or bond amortization. In accounting software, journal entries are usually entered using a separate module from accounts payable, which typically has its own subledger that indirectly affects the general ledger; journal entries directly change the account balances on the general ledger.
Model Enhancement	One of the three basic types of process mining. A process model is extended or improved using information extracted from some log. For example, bottlenecks can be identified by replaying an event log on a process model while examining the timestamps.
Moore's Law	Moore's law is the observation that, over the history of computing hardware, the number of transistors on integrated circuits doubles approximately every two years. The law is named after Intel co-founder Gordon E. Moore, who described the trend in his 1965 paper. His prediction has proven to be accurate, in part because the law is now used in the semiconductor industry to guide long-term planning and to set targets for research and development.
Murphy's Law	Murphy's law is an adage or epigram that is typically stated as: Anything that can go wrong will go wrong.
MXML	An XML-based format for exchanging event logs. XES replaces MXML as the new tool-independent Process Mining format.
Occam's Razor	Occam's razor (also written as Ockham's razor from William of Ockham (c. 1287 – 1347), and in Latin lex parsimoniae) is a principle of parsimony, economy, or succinctness used in logic and problem-solving. It states that among competing hypotheses, the hypothesis with the fewest assumptions should be selected.

Operational Support	On-line analysis of event data with the aim to monitor and influence running process instances. Three operational support activities can be identified: detect (generate an alert if the observed behavior deviates from the modeled behavior), predict (predict future behavior based on past behavior, e.g., predict the remaining processing time), and recommend (suggest appropriate actions to realize a particular goal, e.g., to minimize costs).
Precision	Measure determining whether the model prohibits behavior very different from the behavior seen in the event log. A model with low precision is "underfitting".
Process Discovery	One of the three basic types of process mining. Based on an event log a process model is learned. For example, the α algorithm is able to discover a Petri net by identifying process patterns in collections of events.
Process Instance	The entity being handled by the process that is analyzed. Events refer to process instances. Examples of process instances are customer orders, insurance claims, loan applications, etc. Also called case.
Process Mining	Techniques, tools, and methods to discover, monitor and improve real processes (i.e., not assumed processes) by extracting knowledge from event logs commonly available in today's (information) systems.
Representational Bias	The selected target language for presenting and constructing Process Mining results.
Simplicity	A measure putting in practice Occam's Razor, i.e., the simplest model that can explain the behavior seen in the log, is the best model. Simplicity can be quantified in various ways, e.g., number of nodes and arcs in the model.
XES	Is an XML-based standard for event logs. The standard has been adopted by the IEEE Task Force on Process Mining as the default interchange format for event logs.