topicus in finance

Process Mining & Simulation

Public Version

UNIVERSITEIT TWENTE.

Stefan Hessels – S0176869
Business Analytics
6/29/2016

Preface

And there you have it, my Master thesis. As many may know, my Bachelor thesis took a little bit longer than average, so I was preparing for a long ride while writing this thesis. But, to be honest, writing this thesis went by in a flash. In a little over six months I went from pressing Ctrl+N in Word to a document with over 20.000 words, a lot of data analysis, some Process Mining and automatically process generating BPS-model.

As many students have experienced when writing a thesis, you sometimes feel you are alone on a boat drifting to Graduation Island. But you are not, because several others have helped guiding the boat to the right direction. First of all I want to thank Topicus and especially Sander and Ferry for providing for all the help they could give helping me graduate. Especially the (almost) weekly stand-ups have always been a good guidance throughout the graduation process.

Furthermore I could like to thank both my supervisors at the University of Twente, Maurice and Maria. Both have given me very practical advice to improve my thesis to the level it currently is. Especially the tips for improving the structure of the paper have been welcomed with open arms, since, as some may know, structuring things formally is not always my forte.

Clearly, I have done something well while writing this thesis, since I was offered a job here at Topicus. After a brief vacation I will start as an Analyst and become fully part of a team, so I'm officially not alone on my boat any more.

Management Summary

Context
Currently there is little insight in the FORCE process; it is unknown how various cases flow through the process and how this flow affects the various cases and departments. Insight in this process can contribute to the identification of bottlenecks and can show opportunities for improvement. During a preliminary report, business questions were listed from the input of various stakeholders. Some of these questions have been answered directly by using Process Mining. However, some need extra research in order to answer them. To answer these questions, Business Process Simulation was proposed as a solution. Contrary to "traditional" simulation model development, this research is conducted at the supplier side of the information system and the amount of contact with the bank is limited. With that in mind, the created simulation model is largely based on the data that is available within the FORCE system, with no input from the bank itself.

Research Problem
How can we use the data that exists within the (FORCE) system to build a valid BPS-model?

Deliverables
During this research various components of a simulation model were distinguished: the activity, the resources and their related roles, the queue, the entity and the sequence flow. The better we describe (the characteristics of) these components, using the data, the better the simulation model will be. We have used various approaches to describe the various characteristics of these components, using existing approaches, adapting existing approaches and coming up with our own approaches.

The table-output of these methods has been used as input for our BPS-model. This model has been created in such a ways, that it can easily be altered by changing the values in the tables. This had led to a BPS-model that has the potential to answer the various business questions, has a valid arrival rate, a valid flow through the process, but unfortunately an invalid processing speed. Due to this, the current model can answer the business questions about the process that exist within Topicus, but we will not know whether these answer will represent what will happen in the future. It probably will give some hint in which direction they real answer will be, but not more than that.

Readers guide
This thesis can be interesting for multiple parties. For each of these parties' different parts of this method are interesting. For one: the academic community. In the Process Mining Manifesto it was proposed that Process Mining should be used in combination with Business Process simulation. The entire paper is about this process, so for them I suggest to start reading at the top. For fellow BPS-model developers, two aspects can be interesting: firstly, how can one use an event log to give valuable insight in the process. For them I especially recommend Chapter 6 and how they relate to a BPS-model can be found in Figure 27. For this community it can also be interesting to see how Process Mining can be used to assess whether or not their BPS-model is valid. For them I recommend Chapter 8.

Table of Contents

1 Introduction

1.1 Topicus

Topicus is an ICT service provider with more than 550 employees. The company is based, among others, in Deventer, Enschede, Amsterdam, Leiden and Zwolle. The vision of Topicus is that its products are based on Chain integration, where all parties within the chain work together. These products are offered to the market in the form of a SaaS (Software as a Service) concept. Topicus is currently active in four areas: healthcare, financial services, government and education.

Topicus Finance is, with approximately 250 employees, the largest division of Topicus. In this division, products for mortgages, business lending, savings and investments and pensions are developed. The main development takes place in Deventer, with branches in Zwolle (FINAN) and Amsterdam (Jungo).

One of the products delivered by this division is FORCE, which consists of mid and back office software products for mortgages. From a business process point-of-view: FORCE is used for requests for, or mutations of, mortgage contracts.

1.2 Straight-Through Processing (STP)

FORCE thrives to get an as high as possible degree of Straight-Through Processing (STP) in all mortgage processes. Straight-through processing refers to handling cases without human involvement [1]. It is a set of business processes and technologies that is used to create an infrastructure for automated real-time transaction processing [2]. The purpose of STP is to create efficiencies, eliminate mistakes, and reduce costs by having machines instead of people process business transactions [3]. The use of (online) STP is the trigger in a shift that is of crucial importance to cost effective banking in an ever turbulent and changing (financial) world [3].

The main goal of the FORCE engine is to maximize the STP percentage by using "Chain Integration", where FORCE is connected through API's[1] with various other systems: external systems, like the Bureau Kredietregistratie (BKR) and the Nationale Hypotheekgarantie (NHG) system, but also to internal systems of the client, such as front office or HR systems. Depending on the needs of the customer, the workflow process can be customized.

For a Bank, the process is constructed as followed: first a new request is made manually or through a mortgage-offer request message, the latter is also known as a HDN-message. After that the information within the request is extracted and connected to the CRM-system of the bank. Next the request is checked for correctness and completeness, which is followed by several checks such as the BKR and the NHG. The process is completed with an examination whether further investigation is needed or can automatically be accepted or declined. If the first happens to be the case, the request is send to the responsible

[1] Application program interface (API) is a set of routines, protocols, and tools for building software applications. An API specifies how software components should interact and APIs are used when programming graphical user interface (GUI) components. 4. Webopedia,(2016). *API*. 2016; Available from: http://www.webopedia.com/TERM/A/API.html.

department. If a request fits within the guidelines of the bank, an offer is generated and sent to the customer. An overview of this process is provided in Figure 1.

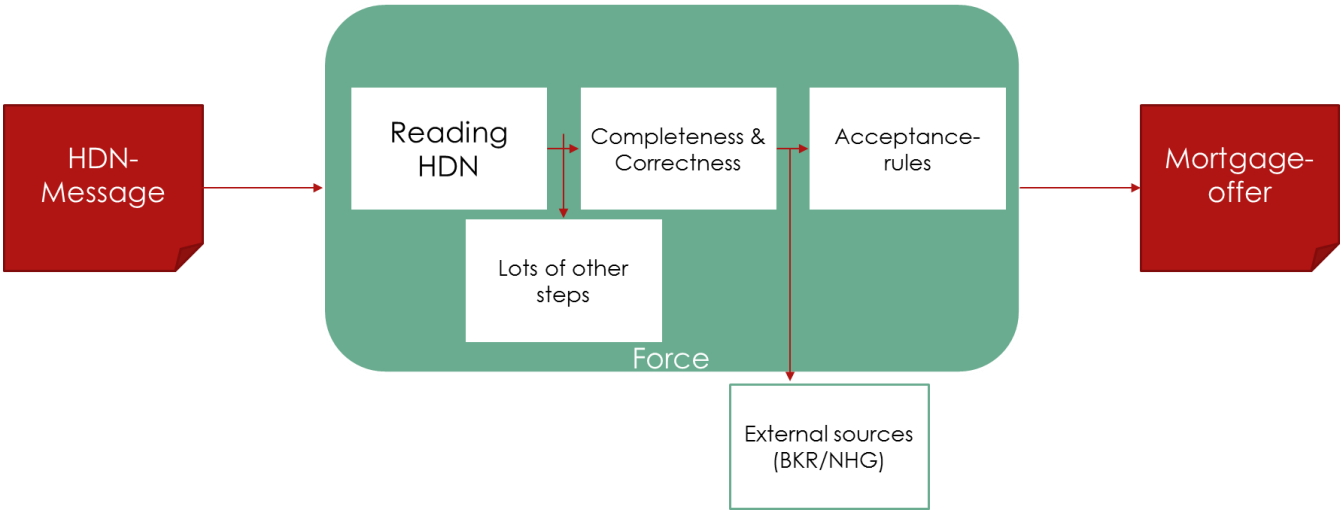


Figure 1- Mortgage Process

A request has a certain status, which relates to a specific step in the process. With the use of state transitions, the request is guided through the process. For example, the last column in the event log from Figure 2 corresponds with the highlighted flow in Figure 3.

Image deleted for Privacy reasons
Figure 2 - Example event log

Image deleted for Privacy reasons
Figure 3 - Part of the flowchart

Most of the state transitions or activities are automatic, but some have to be executed manually. Within Topicus, the workflow that leads to a signed offer consists of only automatic statuses, thus is without human involvement and referred to as the primary path or the “Happy Flow”. Throughout this report, to keep a positive vibe, the latter term will used.

The Happy Flow and all the alternative routes do already have a Business Process Model. It is a very large model, with over 140 activities and over 200 traces.

1.3 Business questions

Currently there is little insight in the FORCE process; it is unknown how various cases flow through the process and how this flow affects the various cases and departments. Insight in this process can contribute to the identification of bottlenecks and can show opportunities for improvement.

During a preliminary report, business questions were listed from the input of various stakeholders. Some of these questions have been answered directly by using Process Mining. However, some need extra research in order to answer them. The remaining questions are:

1. What happens with capacity needed for the various departments if external events happen that changes the number of arrivals, in order to keep the same throughput time?
2. What happens with the throughput time if a large change in the process, from conditional to unconditional offering will take place?
3. For most requests, a throughput time of 5 days is promised. However, some requests do not meet this Service Level Agreement (SLA). What happens with the % of requests that meet this SLA if we apply small changes in the process?

These three questions have the same structure: "What happens with ….(1) if ….(2)?"
(1) refers to some KPI we want to make predictions upon
(2) refers to some scenario that might/will occur in the future

To answer questions with that structure there are two approaches:
1. A mathematical approach (such as algebra, calculus, or probability theory), where we can obtain an exact to such questions.
2. A simulation, where we use a computer to evaluate a model numerically, and data are gathered in order to estimate the desired true characteristics of the model.

If the relationships that compose the model are simple enough, it may be possible to use the first approach containing mathematical methods to obtain exact information on questions of interest; which is an analytic solution. However, most real-world systems are too complex to allow realistic models to be evaluated analytically. Which leaves the second option, studying the models by means of simulation [5, 6]. The process we will investigate has 142 activities and 240 traces, which can be classified as a complex system.

1.4 Process Mining & Business Process Simulation

Business Process Simulation (BPS) models are created by simulation experts and based on insights from information sources such as process documentation, business experts interviews and process observations [7]. Process Mining (PM) is a process management technique that allows for the analysis of business processes based on event logs[8]. An event log is a large table which states what action was performed, by whom and on which time.

Traditionally these two techniques are used separately from each other, where PM is used mostly for tactical decisions and BPS mostly for strategic decisions[5]; however both techniques are rarely incorporated with each other. We will elaborate on both BPS and PM in Chapter 5.

In [9] three common pitfalls in current simulation approaches were presented:
1. Modelling from scratch rather than using existing artefacts. This leads to mistakes and unnecessary work.
2. Focus on design rather than operational decision making, which is helpful for the initial design of a business process but less suitable for operational decision making and continuous improvement.
3. Insufficient modelling of resources: the behaviour or resources is typically modelled in a rather naive manner.

[10] also lists some disadvantages, which mostly are directly related to the human involvement while developing a simulation model:

1. Process documentation might deviate from real-life process behaviour
2. Interviews with business experts can result in contradictory information
3. Interviewees perception tends to be biased to a certain extent
4. When using observational data, the Hawthorne effect[2] can occur

Together all these disadvantages will contribute to a discrepancy between the behaviour of the simulation model on the one hand and the real-life process on the other hand. As a result, efforts to improve the realism degree of simulation models are valuable as they will enhance the representativeness of analysis results and hence its relevance for management support [10]. In other words: the more BPS-models represent reality, the better the model can act as a decision tool for management. By using more objective data that is retrieved from real-life usage of the system, the realism of the model can be improved, as it reflects the real-life behaviour more.

The first attempt, found in scientific literature, to combine Process Mining and Simulation was done by Rozinat [12]. In his paper design-, historic- and state information are merged in order to construct an accurate model based on observed behaviour. This rather than a manually-constructed model which approximates the workflows anticipated behaviour. [12] states that Process Mining can be used to view simulated and real processes in a unified manner, where PM and BPS-models together are used together to extract much more detailed and dynamic data from processes; more than traditional data warehousing and business intelligence tools. Furthermore a unified view of real-life logs and simulation logs enables the validation of the simulation model by re-analysing the simulation logs.

Besides the potential value for simulation research, the potential value of process mining in a simulation context is also recognized within the process mining community as it is explicitly marked as a research challenge in the Process Mining Manifesto[13].

[2] The Hawthorne effect is a type of reactivity in which individuals modify or improve an aspect of their behaviour in response to their awareness of being observed. 11.Landsberger, H.A.,(1957), *Hawthorne Revisited: A Plea for an Open City*. 1957: Cornell University..

2 Glossary

2.1 Topicus FORCE related

Acceptance frame: This is a check whether a mortgage request can automatically be (dis)approved, or that further investigation is needed, for example by the risk or the fraud department.

BKR: Bureau Krediet Registratie, a bureau that has information about an individual's debts.

CCC Check: Check "Controle correctheid en completeheid". It checks whether all information for a request is present in the request and is correct.

FORCE: The product that is used for processing mortgage requests.

HDN Check: Check whether the HDN message is a format that can be used in FORCE.

HDN Message: An XML document that is used for the communication between mortgage requesters and the mortgage suppliers.

NHG: Nationale Hypotheek Garantie, an "insurance" mortgages can have, for when the house owners are unable to pay their mortgage fee or have to sell their house with a loss

Status: A specific state in the process that defines where a mortgage request is in the process.

State transition: After an activity is done, the status of a request is changed to a next status in order to make it enforceable for another activity.

2.2 Other terms

Activity: An automatic or manual execution of task(s).

BPS: Business Process Simulation, the usage of simulation to improve business processing.

Event: An activity X that is performed by employee/system Y on timestamp Z.

Event log: An event log is a multi-set of traces, where a trace represents a single case. Each trace consists of various events, which represent pieces of work performed for the trace.

Process: a collection of activities cutting across various departments, producing a valuable output for the customers.

PM: Process Mining, the usage of (event) data to increase insight in (business) processes.

Workflow: a technical realization of the process.

3 Research Problem & Questions

3.1 Research Problem

Several business questions exist within Topicus, concerning some scenario "What happens with …. if ….". To answer such type of questions, Business Process Simulation (BPS) is identified as a possible solution.

Contrary to "traditional" simulation model development, this research is conducted at the supplier side of the information system and the amount of contact with the bank is limited. With that in mind, the created simulation model is largely based on the data that is available within the FORCE system, with no input from the bank itself.

3.2 Research Questions

The research problem provides us with the main research problem:

How can we use the data that exists within the (FORCE) system to build a valid BPS-model?

In order to provide a BPS-model that reflects the reality as good as possible, every aspect of such a model should be described with relevant characteristics. For this reason we need a meta-model to use as a base for our BPS-model. This provides us with the following sub-question:

1. *Is there a Meta model for the design of a Business Process Simulation? Which components are part of this model?*

We mentioned that every building block should be described with relevant characteristic(s) in order to provide a valid BPS-model. This provides us with the following sub-question:

2. *How can we use the data that is available in order to generate information about these components in order to help to retrieve the simulation model?*

With the answers of sub-questions 1 & 2 we have the input for the BPS-model. Next, we have to use this input to build a BPS-model. But why stop there? Why not build a BPS-model that is generated from solely the output of sub-questions 1 & 2 and code that can be applied at every business process? This provides us with the following sub-question:

3. *Can we build a BPS-model, solely using code and data that has been generated from Process Mining?*

Now we have a BPS-model. However, we can only use this model to make predictions about the future if this model shows (near) real life behaviour. Thus, we have to know whether the model is valid. This provides us with the following sub-question:

4. *Does our method provide us with a valid model?*

If we can answer these four sub-questions, we can answer our final research question: *How can we use the data that exists within the (FORCE) system to build a valid BPS-model?*

4 Research Methodology

4.1 Design Cycle

During this research an iterative process is used throughout the development of the simulation. The Design Cycle, as proposed by Wieringa[14].

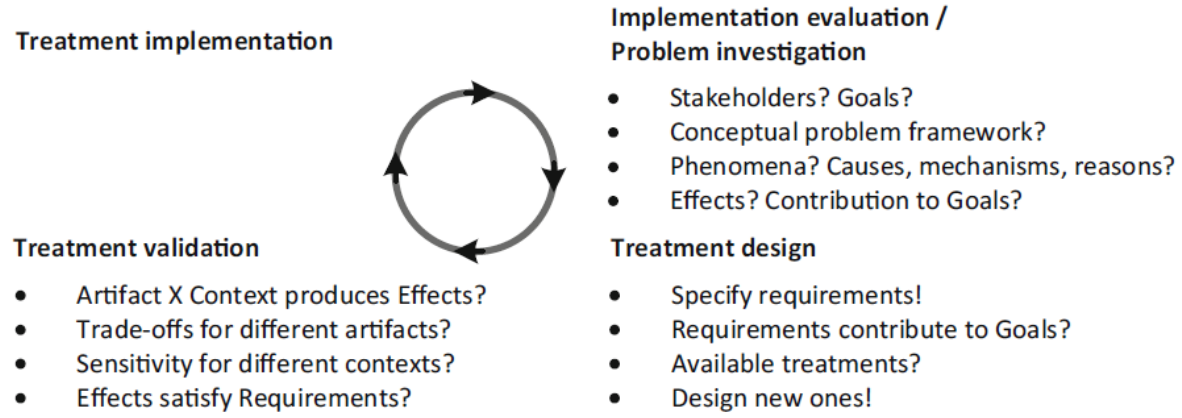


Figure 4 - The Design Cycle (Wieringa, 2012)

Wieringa use some non-conventional concepts in his Design Cycle. Three concepts are important to understand, since we will use them throughout this paper:
Artifact: The method, tool, software that is to be designed by the designer. Everything within this artefact can be designed by the designers and does not depend on external factors.
Context: the context is the "real world" the artefact interacts with, such as the people that are affected by the artefact (stakeholders), but also laws, values, norms, values, desires, fears, goals, norms, and budgets appear in the context of an artifact and cannot be designed by a design researcher. They are given to the design researcher, as part of a problem context, and the researcher must investigate these elements of the context in order to understand them, but not to change them.
Treatment: The treatment is the interaction between the artifact and the problem context to treat a real-world problem. It originates from the medical world where an artifact (medicine) interacting with a problem context (the human body) to treat a real-world problem (contribute to healing).

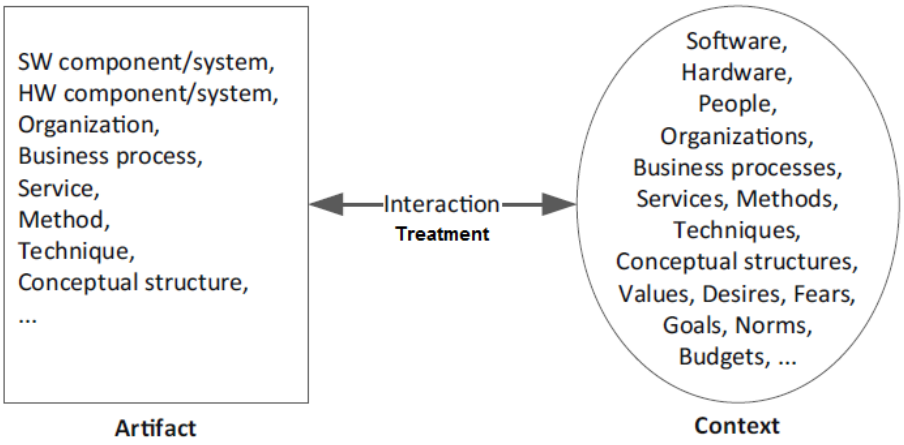


Figure 5 - Artefact, Context &Treatment

With the use of this cycle it is possible to iterate a design process and continuously improve the model. Each step of the cycle contributes to this goal[14].

- Problem investigation: What phenomena must be improved? Why?
- Treatment design: Design one or more artefacts that could treat the problem.
- Treatment validation: Would these designs treat the problem?
- Treatment implementation: Treat the problem with one of the designed artefacts.
- Implementation evaluation: How successful has the treatment been? This may be the start of a new iteration through the engineering cycle.

4.2 Structure of this paper

When we translate the design cycle to our problem, the structure is as followed:

Problem investigation:
Chapter 5 Problem investigation: Here we will discuss the stakeholders and define our conceptual problem framework; hence elaborating on the (academic) research environment we will conduct our research in (Process Mining and Business Process Simulation)

Treatment design:
Chapter 6 Requirements Specification & Artefact Design for simulation input: Here we will design the methods used for generating input for the BPS-model. This input will consist of one or more data tables.

Chapter 7 Artefact design: Building the Simulation model: Here we will elaborate on the design choices made when building the BPS-model.

Treatment validation:
Chapter 8 Artefact validation: In this chapter we will provide and analyse metrics that will validate (or not) whether our designed BPS-model shows real-life behaviour.

The treatment implementation is out of the scope of this paper. Each of these chapters refer to one of the sub-questions. During each of these chapters information-output is generated during a process, which is input for one (or more) of the following chapters. With the final output, a (possibly) valid BPS-model, we can answer our research question:

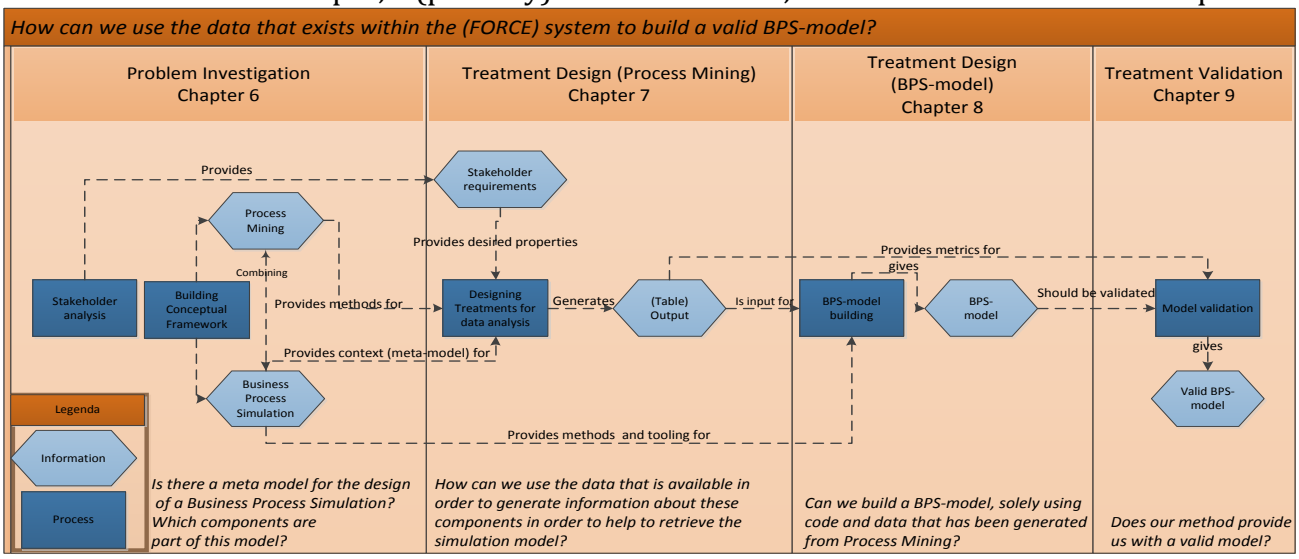
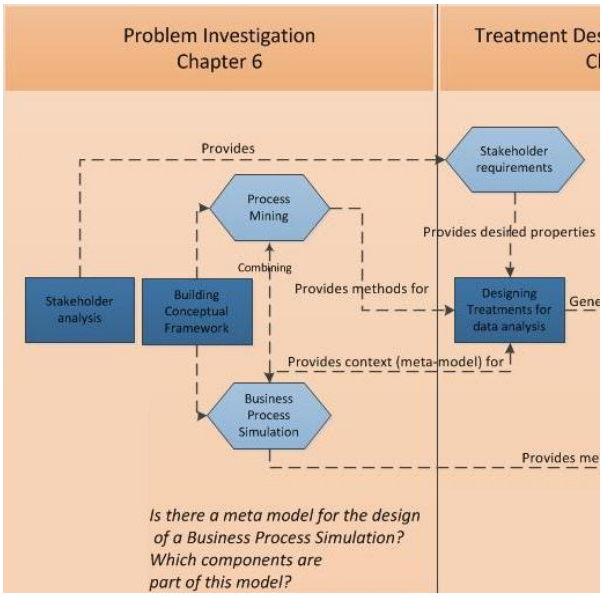


Figure 6 - Overview of the Design Cycle

5 Problem investigation: Process Mining and Simulation



The first step according to the Engineering Cycle is the Problem investigation. This research will take place at the overlap between Process Mining and Simulation. On these two subjects will be elaborated in this chapter. Furthermore combining Process Mining with Business Process Simulation will be discussed. Some research has already been done on this combination, but these papers are scarce and have a rather conceptual nature [10]. In this chapter the current state of research is determined. But first a small stakeholder analysis is conducted in order to determine how this research affects the stakeholders.

5.1 Stakeholder analysis

In this research we distinguish two stakeholders that can contribute from the artefact created during this research: the bank and Topicus. In this chapter we will discuss how they could be affected by this research.

Bank

The bank is the provider of the used data. Because of privacy issues, some data is scrambled, but this will not directly affect this research, since the scrambled information, such as names and addresses, is not relevant for this research. By answering the business questions as stated in Chapter 3.2, the business could be improved, possibly resulting in improved customer satisfaction (because of a shorter throughput time), more efficient resource allocation (because of better capacity planning) and a smaller workload (because of more efficient processing).

Topicus

Topicus would not directly benefit from the artefact, since they solely provide it as a service in order to increase their customer satisfaction. They could also sell some consultancy related advice that is derived from the model. Furthermore, this essay can act as a preliminary research for the development of features that increase the insight in processes.

5.2 Process Mining

There are two point-of-views that can help to understand the STP-process. The first is a data-centric approach, as is used in data mining which can be used to find patterns that will increase our understanding of the process. The second is a process-centric approach which can be used where Process Modelling is applied to understand the process. For each of these two approaches there are several advantages and disadvantages when investigating the current process.

Data Mining techniques use data in order to extract patterns representing knowledge implicitly stored in large databases, data warehouses, the Web, other information repositories or data streams. Contrary to “traditional” research [15], it does not use a

hypothesis to base the research upon, but uses the data to find the relation(s) [16]. This type of research is best applied in an unstructured (Business) Environment, where there are no general theories, especially where one has large quantities of data containing noisy patterns [17]. A well-known example is the systematically extraction of biological meaning from large gene/protein lists [18]. Using Data Mining we can extract relevant information for the process. Examples are: which are characteristics that determine the throughput-time of a request? Or, what is the productivity of employee X? Answering these questions can provide insight about the process, but do not provide a comprehensive understanding of the end-to-end processes. For example, which path do requests take that have a long throughput time?

Business process modelling (BPM) is the activity of representing processes of an enterprise, such that the current process can be analysed or improved. The business objective is often to improve aspects of the process, such as process speed, cycle time and quality. Business Process Modelling techniques are concerned with 'mapping' the 'workflow' to enable understanding, analysis and positive change. Diagrams - essentially 'flow diagrams' - are a central artefact of this methodology. An example of such a flow chart has been shown in Figure 2. These charts can provide insight in the process. However, the reality is not always the same as the should-be situation described in flow charts. Furthermore, flow charts do not provide insight in the as-is situation.

Because data mining techniques are too data-centric to provide a comprehensive understanding of the end-to-end processes within an organization, the algorithms used within this discipline are not fully usable when analysing processes [8]. On the other hand Process modelling relies heavily on experts modelling should-be processes and do not help the stakeholders to understand the as-is processes. Besides, they only provide the relations within a process and do not give direct insight how often a certain path is used [8].

Process Mining is a relative young discipline started in 2001 and is therefore somewhat in its infancy. Process Mining is a process management technique that allows for the analysis of business processes based on event logs. The basic idea is to extract knowledge from event logs recorded by an information system [8].

Process Mining builds on process model-driven approaches and data driven approaches, by combining these two, using the process-centric view of Process Modelling with the data-driven approach of Data Mining (Figure 7). Example of study are about Risk Management[19], Process improvement[20], Fraud Mitigation[21] and Resource Management[22]

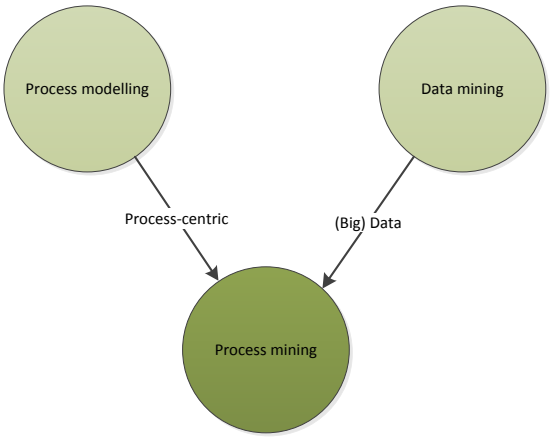


Figure 7 - Process Mining

5.3 Business Process Simulation (BPS)

Simulation of business processes creates added value in understanding, analysing, and designing processes by introducing dynamic aspects, in other words the development of process and resource performance in reaction to changes or fluctuations of certain environment or system parameters[23] . It provides decision support by anticipation of future changes in process design and improves understanding of processes. The results provide insights supporting decisions in process design or resource provision. The goal is to improve factors such as process performance, process and product quality, customer satisfaction or resource utilization [24]. A "discrete-event simulation model" is one in which the state of the model changes at only a discrete set of time points [25]. Most of the times BPS is used in order to facilitate decision making for re-engineering of Business Processes, such as [26, 27] and capacity planning[5].

5.3.1 Components of a simulation model

In order to create a viable simulation, as much information as possible should be used in the model. In this chapter we identify a meta-model for a simulation model, in order to categorize the information in a comprehensive way. Furthermore, we derive challenges we will possibly meet during the development of the simulation, while solely using the data that is available as input.

A simulation model consists of various components that work together in order to create the simulation model. Each of these components has various attributes that we should be aware of in order to create a plausible simulation model. Mes and Bruens [27] distinguish three components for a simulation model: entities, resources, and activities.
Entity: An entity is a moving part that requires processing. In the case of this research an entity is an application for a mortgage.
Resource: An asset of the company that acts on an entity. A resource is a necessity for the manual activities in order to process the various entities. In a simulation model only the resources are used that will impact the throughput of a process. In this case the resources are the various employees of the bank.
Activity: The activities are the services required by the entities. In our case these are the various activities, such as the HDN-check, the check for correctness and completeness (CCC) and the testing within the acceptance frame.

Martin [10] provides some additional components that should be addressed:

Gateway: An element that influences which activities will be executed on an entity. There are three possibilities:
An XOR split is a scenario where an activity is followed by at most one of the following activities.
An OR split is a scenario where an activity is followed by any number of the following activities.
An AND split is a scenario where an activity is followed by all of the following activities.
Sequence flow: A relation between activities and gateways that defines the sequence an entity has to follow. It is mostly represented using arrows.
Queue: Model component containing entities for which the required resources to perform an activity are not (yet) available. Every queue has a queue discipline, which determines which entity is picked from the queue when a resource becomes available.
Resource Class: a group of resources. How such a group is defined depends on the grouping variable, which can either be an organizational unit or a resource role.
Schedule: a component that defines the availability of a resource. It defines whether or not a specific resource is available.

The eight building blocks, as described above, form the base for a simulation model. Each of these blocks interacts with one or more of the other building blocks. Figure 8 provides a comprehensible view of these interactions. This figure can be seen as the meta-model for a BPS-model.

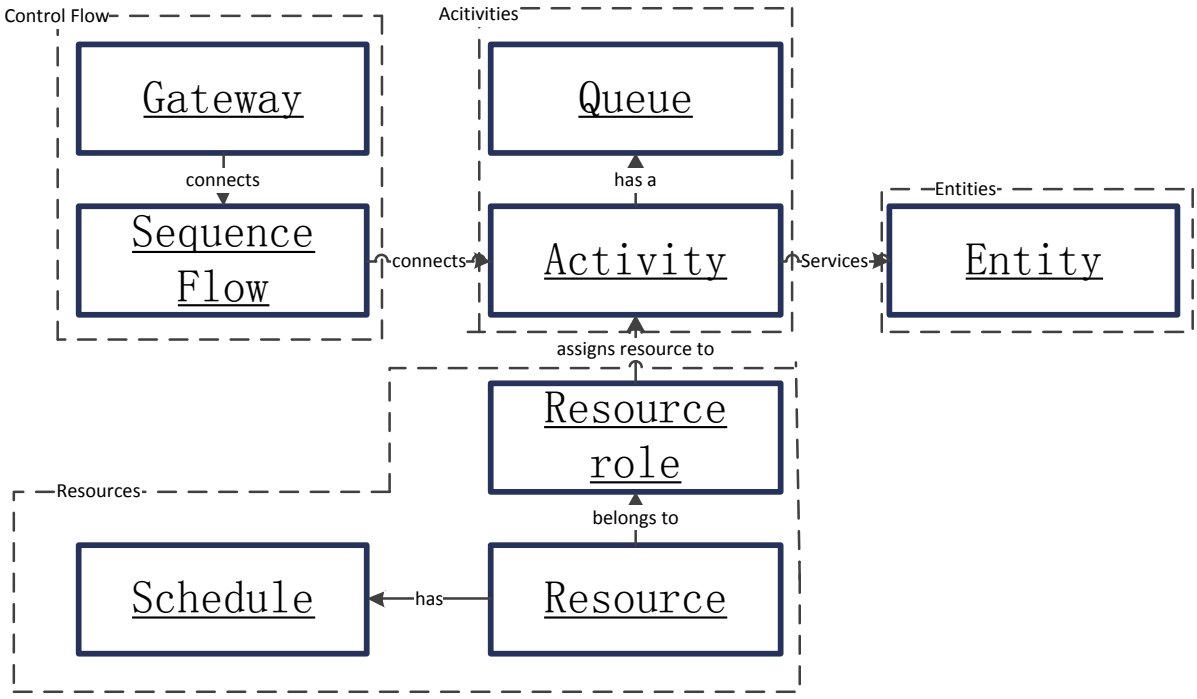


Figure 8 – The various components of a simulation and how they interact

5.3.2 Challenges

As mentioned earlier; only a few attempts[5, 7, 28] were done combining process mining with BPS. Because of the scarce literature, it is expected that some parts of the building blocks as shown in Figure 8 are not, or not fully, researched in such a way that the data directly is applicable for the development BPS-model. Hence, some challenges will have to be tackled in order to retrieve a BPS-model that is as good as possible.

In a recent paper (February 2015) by Martin[29], a list of research challenges is provided. The more of these challenges are tackled, the more extensive our data input will be and the better our simulation model will become. For each component (as shown in Chapter 5.3.1), the biggest challenges are shown:

Entities

- The retrieval of a set of relevant entity attributes, i.e. attributes that influence activity execution such as entity routing and activity durations, is far from trivial.
- Entities can be generalized to larger entity types. An example in our case is by distinguishing mortgages with or without a NHG. Even though event logs do not contain direct entity type information, case and event attributes can be helpful as attribute value convergence suggests entity type existence. However, no research efforts to support entity type modelling using event logs are identified.
- Research interest on the use of timestamp analysis to support entity arrival rate modelling is limited [7, 30, 31].

Activities

- A level of abstraction should be chosen. When the level of the chosen abstraction is higher than that represented by the event log, aggregation of events should take place.
- Activity duration reflects its execution time and can be modelled deterministically, either fixed or conditional on entity attributes, resource attributes, queue length or the system state. Duration observations can be retrieved from an event log, where the observation accuracy depends on the recorded event types. When only either the start or stop time of an event is presented, the actual processing time cannot be determined. However, a proxy can be generated, i.e. the time a previous event ended, or when a resource processed its previous entity. But, consequently, the potential inaccuracy of these estimates should be taken into account.
- Activity duration may be depending on an attribute of a resource our case. For example, some employees may work faster than others. Application of PM for this has only been briefly studied.
- Activity duration may be dependent from the workload of a resource. Some research is already done on this topic[32].

Resources

- Resource assignment rules aim to recommend a single resource for the execution of an activity on a particular entity. To make those efforts applicable in a BPS context, more profound insights are required in resource assignment to an activity when an entity with particular characteristics requests service, potentially taking into account the system state.
- In simulation, resource requirements are often expressed on a resource role level. In that case, the obtained conclusions need to be linked to the allocation of resources to resource roles.
- Mining directly implementable BPS resource schedules is an open research question, where related work is limited to mining resource availability[33, 34].

Queue

- To support queue discipline modelling, log analysis should identify entities that are in the queue at a particular moment, their characteristics, the system state

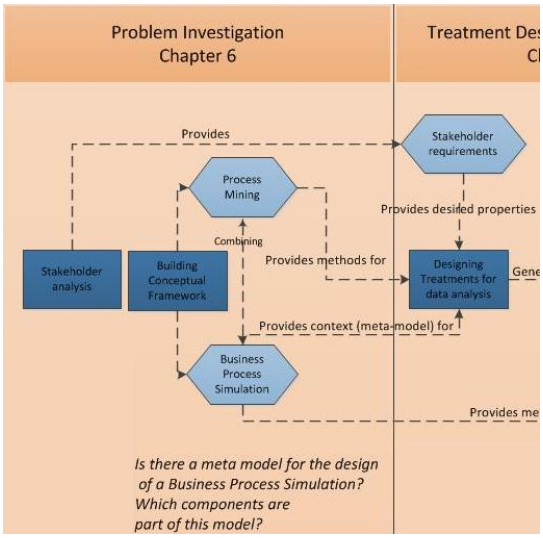
properties, etc. The observed processing order of entities suggests the queue discipline. Literature does not provide clear starting points on this topic.
- A queue abandonment condition can be specified to express conditions under which entities prematurely leave the queue. As a typical event log only registers events related to activity execution, it is not trivial to determine in which queues a case resided before actual processing.

Sequences
- An AND-gateway can be suggested even when not all activity orders are present in the log, implicitly assuming that all interleavings are possible.
- As discovery algorithms tend to be Petri net based, OR-gateways cannot be directly discovered, necessitating further processing.

Gateways
- Routing logic needs to be specified for XOR- and OR gateways. Event logs can support routing logic modelling by analysing activity execution circumstances. Some research has already been done on this topic. This research can be extended by considering non-linear classification rules or by broadening the decision variable scope, e.g., queue length or resource availability instead of only case attributes.

5.4 Conclusion

In this chapter we did a problem investigation: we distinguished the stakeholders, we provided the context about Process Mining and Business Process Simulation, and provided challenges that might occur combining these two.



In order to answer the sub-question: "*Is there a meta model for the design of a Business Process Simulation? Which components are part of this model?*" we derived a meta-model (Figure 9) for a BPS-model, with all components that should be assessed when building a BPS-model. Furthermore, we derived some challenges that should be tackled in order to generate a valid BPS-model.

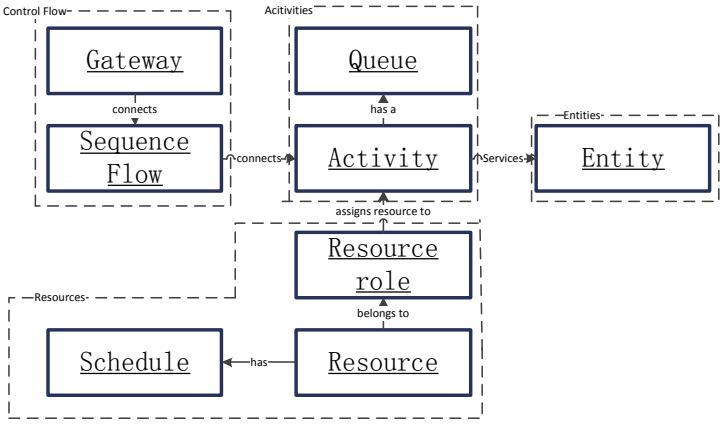
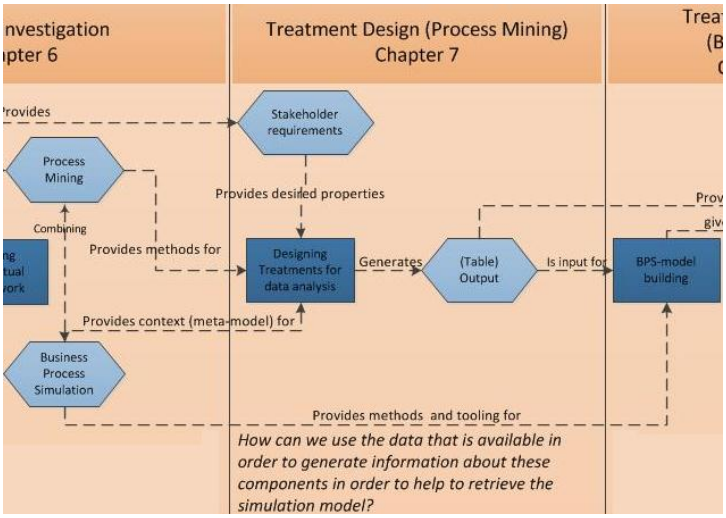


Figure 9 - BPS meta-model

6 Requirements Specification & Artefact Design for simulation input



As mentioned in the previous chapter, several challenges exist while designing this simulation model. Furthermore, in Chapter 5.3.1 we provided a meta-model for a BPS-model, containing several building blocks. In Figure 10 for each of these (or multiple) building blocks a chapter is assigned. In each sub-chapter we design methods to deliver the various attributes of these components.

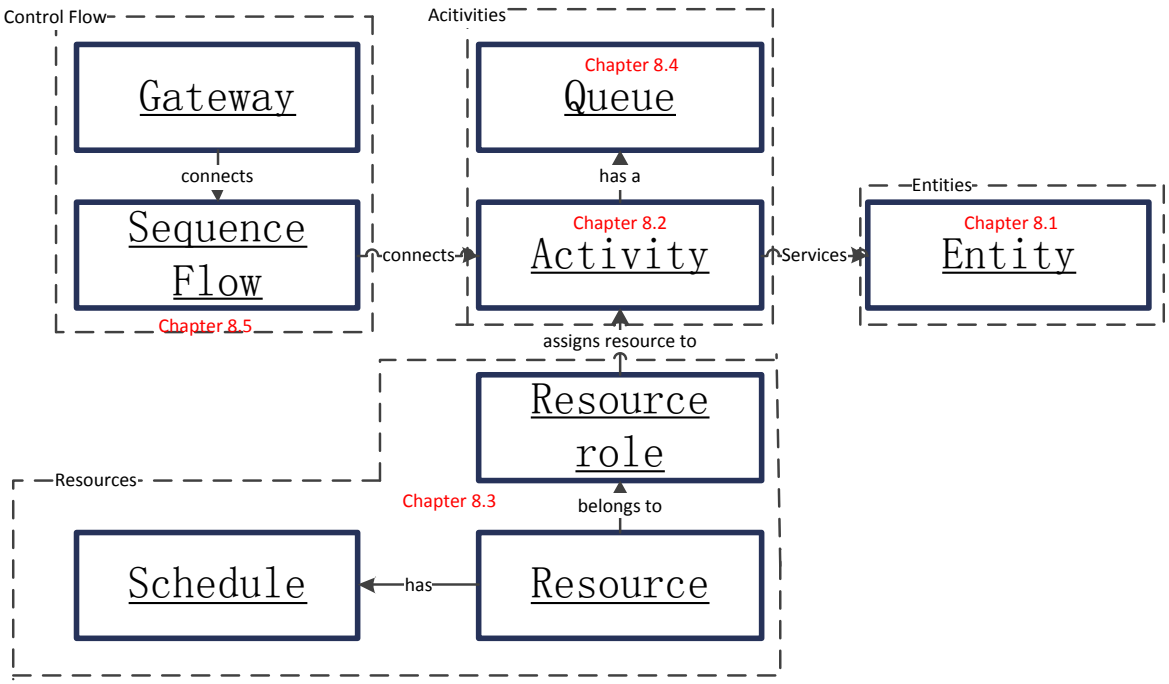


Figure 10 - Division of BPS building blocks into sub-chapters

Per attribute the following process is followed: first we provide treatments that are already available in scientific literature to help us design a solution for an attribute. After that we give requirements an attribute should comply to and give reason(s) why these requirements are there. Finally, combining our requirements with the available treatments will deliver us a final starting point for our own treatment.

6.1 Entities

Entities are the moving parts that require processing. In this sub-chapter we will discuss the arrival rate of the entities.

6.1.1 Arrival rate estimation

The arrival rate is the number of entities that start the process in a given timeframe. It can be assumed that this rate is not constant. For example, for the night and for weekends it can be assumed that the arrival rate is lower is a lot (or possible all) offices of mortgage vendors are closed during these times. Also not every day is the same, so it can be assumed that the arrival rate may differ day-by-day, following some kind of statistical distribution

Scientific literature

For the arrival times two distributions are common: Negative-Exponential and Poisson. The first is most common for inter-arrival times and the latter is used for the number of arrivals within a time zone.

Overdispersion is the presence of greater variability (statistical dispersion) in a data set than would be expected based on a given statistical model. Overdispersion is often encountered when fitting very simple parametric models, such as those based on the Poisson distribution. [35]. The Poisson distribution has one free parameter and does not allow for the variance to be adjusted independently of the mean.[36] If over dispersion is a feature, an alternative model with additional free parameters may provide a better fit. In the case of count data, a Poisson mixture model like the negative binomial distribution can be proposed instead. in this model the mean of the Poisson distribution can itself be thought of as a random variable drawn, in this case, from the gamma distribution thereby introducing an additional free parameter to handle the variance [36].

To validate a distribution a Pearson Chi Square Goodness-of-fit test should be done[37]. This is done with the following steps:

1. Make an hypothesis on the distribution
2. Choose an applicable level of significance
3. Calculate estimators of this distribution based on the historical data
4. Determine the number of bins
5. Calculate the probability and cumulative probability per bin
6. Calculate the bin size of each bin, using the inverse of the expected distribution with the estimators.
7. Determine how many values of the historical data lies within each bin.
8. Calculate the χ^2 and the p-value, using the Pearson Chi-Square test.
9. Conclude whether the hypothetical distribution can be confirmed.

Most of these steps are straightforward, with an exception on determining the number of bins when the data is continuous. With discrete data the number of bins is the same as the amount of unique values. There are several methods for when the number of bins when the data is continuous, for example by using $\sqrt{N}$ (Square root rule) or $1 + \log_2 N$ (Sturges' rule). However, a big disadvantage of both these methods is that they were designed for a normal distribution, where the negative exponential function is heavily skewed. For this reason we choose to use the Doane's formula, where k is the number of bins, N the

number of data points and Skewness a measure of the asymmetry of the probability distribution:

$$k = 1 + \log_2 N + \log_2(1 + \frac{|Skewness|}{\sigma_{Skewness}})$$

$$\text{With: } \sigma_{Skewness} = \sqrt{\frac{6(N-2)}{(N+1)(N+3)}}$$

When the data is discrete, which is the case when investigating the arrival rate per a certain time period; estimating the parameters is quite complex, since no direct estimators are available. For the Negative Binomial distribution we use fitting by Maximum Likelihood Estimation[38] for a Negative Binomial Regression on a set of ones, to obtain a "negative binomial heterogeneity parameter". With this parameter α and the mean μ, we can calculate the *r* and *p* that are necessary for obtaining a Negative Binomial distribution[39]:

$$r = 1/\alpha$$
$$p = 1/(1 + \alpha\mu)\ \mu$$

Requirements

It can be assumed that the arrival rate may differ throughout the week. An example for such differences is because the front-offices are mostly closed during the weekends. Furthermore, the arrival rate may change throughout the day, for example during nights or during lunch break.

These assumptions are supported when looking at the data at first glance. It should be noted that we cannot totally neglect the weekends and the nights, since requests arrive during these moments:

Image deleted for Privacy reasons

Graph 1 - Distribution of the arrival of requests per day

Image deleted for Privacy reasons

Graph 2 - Distribution of the arrival of requests per hour

While looking at the distribution in Graph 1, a hypothesis can be made that the differences between the working days and between the weekend days can be due to chance, where the arrivals are distributed uniformly. However, a Pearson Chi-squared test [3][37] provides us with the following p-values[4]:

---

[3] Pearson's chi-squared test is a statistical test applied to sets of categorical data to evaluate how likely it is that any observed difference between the sets arose by chance *40. Gosall, N.K. and G.S. Gosall,(2012), The doctor's guide to critical appraisal. 2012: PasTest Ltd.*

[4] the p-value is the probability of obtaining a result equal to or "more extreme" than what was actually observed, assuming that the initial hypothesis is true. *41. Biau, D.J., B.M. Jolles, and R. Porcher,(2010), P value and the theory of hypothesis testing: an explanation for new researchers. Clinical Orthopaedics and Related Research®, 2010.* ***468****(3): p. 885-892.*

	Chi-squared	P-value
Mon-Sun	38.966	0.0
Mon-Fri	597	8,44E-112
Sat-Sun	53	3,56E+03

Thus, it can be concluded that we should take in account each day separately, since both the two days in the weekend as the five weekdays are not uniformly distributed. Also per month the arrival rate differs. Especially an increase at the end of the year can be noticed, but also dips during the summer:

Image deleted for Privacy reasons

Graph 3 - Arrivals per month

Final solution

The first step is to check whether a Poisson Distribution seems logical[42]. For this reason we split the dataset is the arrival rate per hour, per working day. After that we calculate the ratio Mean to Variance:

$$Ratio = \frac{Mean}{Variance}$$

With:

$$Mean = \sum_{i=1}^{n} \frac{Arrival\ rate}{n} \text{ and } Variance = \frac{\sum_{i=1}^{n}(Arrival\ rate - Mean)^2}{n}$$

On average this ratio is 0.21; with a minimum of 0.13 and a maximum of 0.35. For a Poisson distribution mean should be the same as the variance and this ratio should be equal to 1. Because this ratio is always <1, we can conclude that the variance is larger than the mean and so larger is than expected. Thus, we can conclude that this data is overdispersed. Hence, we will use a Negative Binomial Distribution. It turns out that this distribution does not fully fit the distribution, but does a decent job as an indicator for the arrival rate, as can be seen in Graph 4.

Image deleted for Privacy reasons

Graph 4 - Actual vs Expected Arrival Rate

Per working day and per hour in the range 07:00 AM – 7.59 PM the P and R-value are calculated. This provides the table in Appendix A.

6.1.2 Conclusion

In the sub-chapter 6.1 we have discussed the arrival rate of the entities.

For the arrival rate we have looked at:

1. Whether the arrival rate differs per month.
2. Whether the arrival rate differs per day of the week.
3. Whether the arrival rate differs per working day.
4. Whether the arrival rate differs per day in the weekend.
5. Whether the arrival rate differs per hour.

Since the arrival rate differs per day of the week and hour of the day, we took the arrival rate per hour and per working day. It was noticed that this rate was over dispersed for a Poisson distribution. For that reason a Negative Binomial Distribution was chosen. For each of these Negative Binomial Distributions the r and *p* value were determined by using Maximum Likelihood estimation.

6.2 Activities

The activities are the services required by the entities. In this chapter we will define a method in order to get an understanding of the processing time of the various activities. For this three methods are used: 1. Start time and duration estimation 2. Data Cleansing and 3. Test for independence.

After this we will discuss the starting state of our activities, hence how many entities exist within the activity and/or the queue at the start of our simulation.

6.2.1 Estimation of processing and queuing time

Scientific Literature

Iacob and Wormbacher have written various papers to obtain the processing time for semi-structured processes [43-45]. They use to following procedure to cleanse the event log , from which we can determine the duration of an activity (Figure 11):

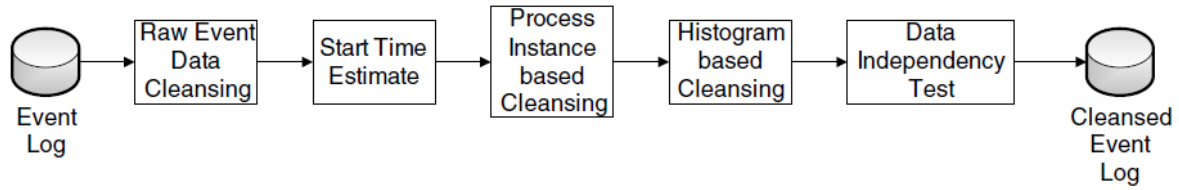


Figure 11- Cleansing of the event log for processing time estimation

Start Time and Duration Time Estimation

Because the start time is not directly available, it should be obtained another way. It is suggested using the end time of the previous task of an employee [43-45]. This can be visualized as followed (Figure 12):

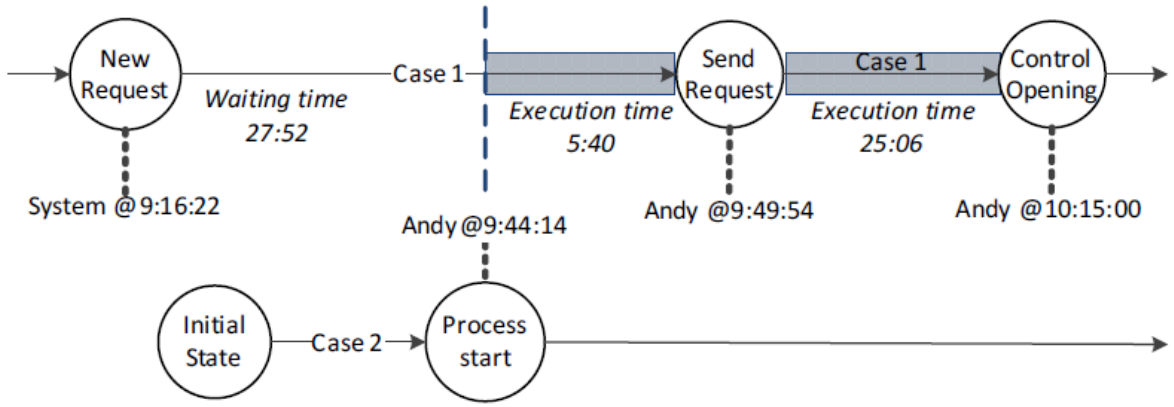


Figure 12 - Estimation logic of processing time [43]

Data Cleaning

One of the most common problems with Big Data is Data Quality. The Data Quality is considered high if "they are fit for their intended uses in operations, decision making and planning"[46].

[43] provides three methods that help to improve the Data Quality in order to obtain an event log of high quality for the purpose of mining the duration of (semi-structured) processes:

Raw Event Data Cleansing:

Raw Data Cleansing consists of two steps: 1. Cleans data that have inconsistencies in the order of the completion time. 2. Eliminate data that is unreliable, for example because servers etc are offline and thus cannot submit data.

Process Instance based Cleansing:

Process Instance based Cleansing cleans special cases that should not be part of the analysis. [43] identified three specific cases:

Test cases: Cases used by a Test user that are used for testing if the program works according to the standards.

Dead lock cases: Cases that in such a state that they are blocked due to some error. Their status has to be changed, disregarding the rules that normally apply.

Livelock state changes: A livelock is similar to a deadlock, except that the process continuously performs state changes but is unable to complete the process due to an infinite loop.

Histogram based Cleansing:

After determining the duration of the various activities, from the set of these durations a histogram can be created. The purpose of this histogram is the detection of outliers. Outliers can occur through various reasons:

Working Hours of Users: Since employees do not work around the clock, there is a time gap between the last action of a day and the first of the next day. Logically, this time gap does not represent the time an employee has worked on a specific case.

Non-visible Activities: In the proposed approach we assume that a user is only working on the system. However, a person also performs other tasks in addition to working in this particular system, such as lunch/restroom breaks, meetings, etc. These activities are not directly represented in the system.

Data Independence Test

There are several aspects that could influence the duration of processing a case:

- Weekday: it is suggested that labour productivity may vary over days of the week for a variety of physical, physiological and compositional reasons relating to lapsed time since the start of a working period [46].
- Iteration: The process flow may contain loops, which results an activity may go through the same activity several times. Since some values may already be altered during the first processing, a hypothesis can be made that it may take a shorter period to process this case a second or more time.
- Workload: the "Yerkes-Dodson Law of Arousal"[47] states that people will take more time to execute an activity if there is less work to do [32].

In order to increase the quality of the simulation model, it should be assessed if one or more of the above factors are of influence.

Requirements

A big disadvantage when using Process Mining on the database, is that it only contains one timestamp for when an activity is ended. Because a timestamp for the start of a process is missing, one cannot directly separate the processing time from the queuing time.
However, the dataset contains a set for an employee-number, which can possibly help to approximate the starting time.

During a preliminary study it was found that the database contains cases of sequence errors [37], because the timestamp is only precise up to seconds. This should be taken into account, because a sequence error can possibly result in a processing time of 0 sec.

Final solution

In the description of the various workflow statuses a Boolean "DoAutomaticStatusChange" was found that indicates whether an activity is either done manually (=0) or automatically (=1). Furthermore a Boolean "Active" was found that indicates shows whether a activity is active (=1) or not (=0). Using these Booleans we can obtain a list of activities that are done manually and active. The next step is selecting the rows that either have a "FromStatus" or a "Tostatus" that occurs in this list. By sorting this output on first the entity number, followed by the timestamp we can calculate the throughput time for a certain activity by looking at the time differences of two rows. For example Figure 13 shows that Entity 68981 took 1 minute and 11 seconds passing through activity "6 – Wijzigen aanvraag".

Image deleted for Privacy reasons

Figure 13 - Example throughput time

As mentioned earlier, the database has sequence errors because multiple events can happen within a second. An example of this can be seen in Figure 14 for the manual process “Wijzigen Aanvraag”. This will lead to a processing time of 0 sec, because the last two rows have the same timestamp.

Image deleted for Privacy reasons

Figure 14 - Sequence error example

To cope with this, the “FromStatus” is sorted by sorting activities that are mentioned in the list of manual activities, if more activities occur in the same timestamp.

Once this data is obtained the calculation and cleaning of the data can be done:

1. Calculate the time differences between the rows
2. Convert this to seconds
3. Delete rows where the “FromStatus” is not a manual activity

The final step is splitting the data into a dataset per activity. To limit the number of unnecessary calculations, for activities that occur at most once every quarter of an hour, only the average throughput time and the variance were calculated. For activities that occur at least once a month, the throughput rate for each activity per quarter of an hour were calculated.

Distribution of processing time

When the processing time is plotted in an histogram, the type of distribution seems to differ among the activities. Some look like an exponential distribution, some look like a lognormal distribution, while others look like mixed Gaussian distribution.

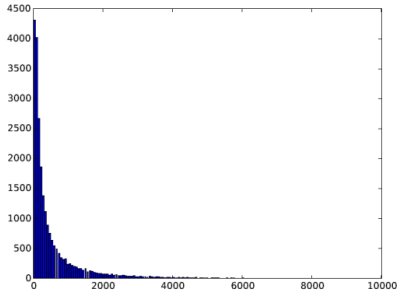


Figure 15 - Example of exponential distribution

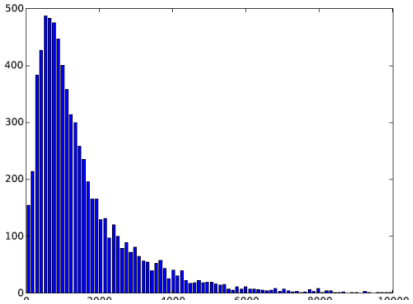


Figure 15 - Example of lognormal distribution

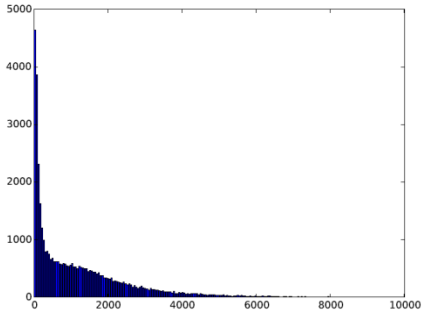


Figure 17 - Example of mixed Gaussian distribution

Most of the processes look like a lognormal distribution, so we use that in order to get an idea about our processing time. This results in graphs such as Figure and Figure :

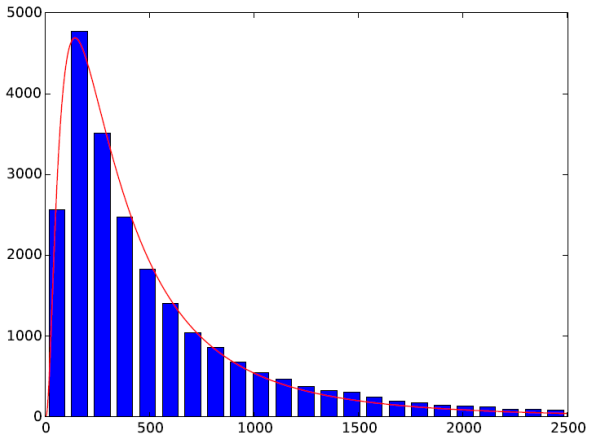


Figure 18

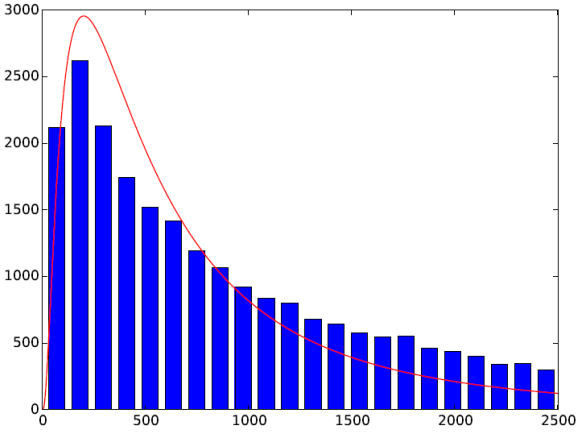


Figure 19

Data independency

In previous paragraph we mentioned three tests for data independency:

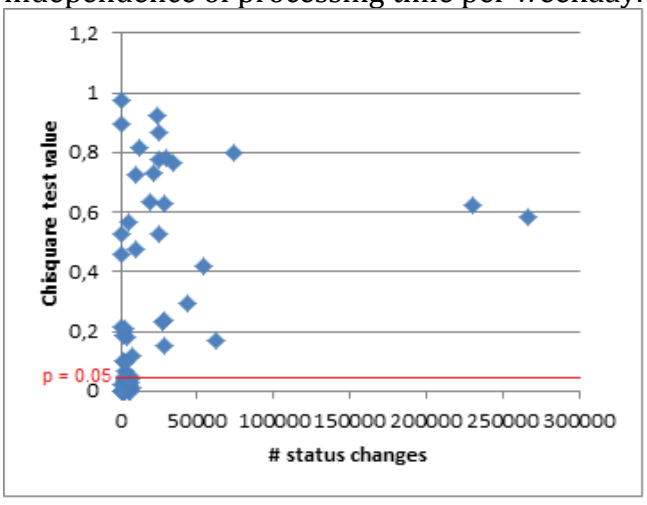
1. Testing for independence of processing time per weekday
2. Testing for independence of processing time per iteration (the number of times an entity arrives at a certain activity
3. Testing for independence of processing time, depending on the workload.

Testing for independence of processing time per weekday

For this test the Pearson Chi square test is used in order to check whether for each activity the average processing time per weekday is the same among all working days.

For most of the processes (80%), there is a difference between days (p>0.05). However, this is mostly for activities with a relatively low number of state transitions. The activities that are done more often, mostly have no distinction between days (Graph 5). We believe this is because for activities that are executed less often, outliers have a greater impact.

When investigating this further, over 90% of the state transitions belong to an activity where there is no distinction between days (p>0.05). For this reason we do assume independence of processing time per weekday.



Graph 5 - Chi square value vs. # state transitions

Testing for independence of processing time per iteration
For this test we add a value to each state transition, # of occurrences: the number of times an entity is processed during a certain activity. So, the first time an entity leaves an activity the # of occurrences is 1. After each loop, when this entity leaves this activity again, this # of occurrences increases with 1.

For each activity we then take the average processing time per groups of # of occurrences. Using Pearson Correlation, per activity, we find whether there is a direct correlation between the # of occurrences and the processing time. As a hypothesis we state that the correlation should be negative, hence the processing time gets smaller when the # of occurrences gets larger. As a threshold for the p-value we take 0.05.

The result is that for most of the activities there is correlation between the iterations and the processing time.

Regression analysis for data independency
We now know there is no correlation between processing time on one hand and day of the week and workload on the other. We also know there is some correlation between the iterations and the processing time. However, it could be that there is some interaction between these independent variables. For that reason we perform a regression on these three variables. With this regression test, we can also derive a formula to implement in our BPS-model.

Since we want to take into account variance and want to derive a formula to implement in our simulation model, we apply linear regression on the values that pass the test as described above. This results in graphs such as Figure 16.

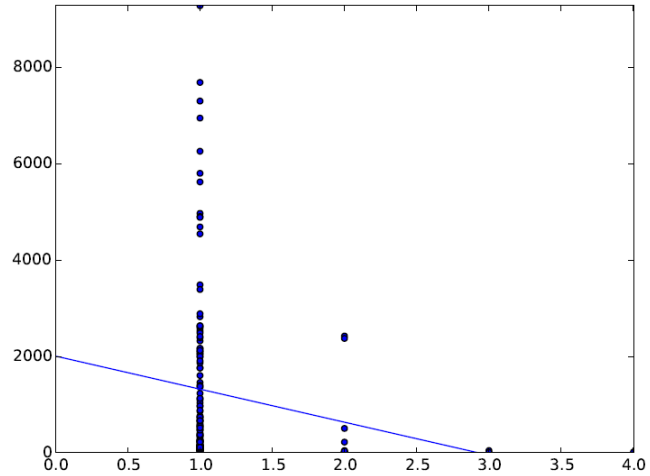


Figure 16 - Relation between number of occurences and processing time

6.2.2 Start number of entities at the simulation

Scientific Literature

When there is a continuous process, such as ours, we cannot start recording our results from the start, since the first entities that arrive always arrive at an empty queue/activity. A so called steady-state has to be obtained in order to provide solid results. Traditionally a warm-up period is calculated to determine steady-state behaviour [6].

But we can also consider this: At any point in time, the workflow process is in a particular state. The current state of each process instance is known and can be used to initialize the simulation model[12].

[28] distinguish five types of data to obtain from a current state for a Petri Net simulation:

- All the running cases of a given workflow and their marking, thus at which queue/activity they are at.
- All the data values associated with each case.
- Information about enabled work items.
- Information about executing work items and the resources used.
- The date and time at which the current state file is generated.

Requirements

During this simulation we assume that no work takes place during the night. This means, if we take a timestamp that lies within the night, we can disregard the work items that are executed and resources that are used. Also our simulation starts on a Monday, thus it seems logic to take a Monday as a current state.

Furthermore we will not make a Petri Net Simulation. For this reason work items are not "enabled", since that is not part of the discrete-event simulation concept. We also do not make a distinction between types of cases/entities, thus the data values associated with each case can also be disregarded.

Final solution

As mentioned in the requirement section, we can disregard information about executing work items and the resources used, information about enabled work items and all the data values associated with each case.

This leaves us with the following two data that we need to obtain a specified current state: The date and time at which the current state file is generated and all the running cases of a given workflow and their marking, thus at which queue/activity they are at.

The current state is heavily dynamic and thus changes rapidly over time. Hence there is not a single "perfect" date from which we can obtain our initial state. We have chosen for 01-09-2014. Furthermore we have chosen 00:00:00 (midnight) as a timestamp, since this is also the start time of our simulation model.

To obtain the state at 01-09-2014 00:00:00, we use the following steps:

1. Filter out events from the event log where the timestamp is after 01-09-2014 00:00:00.
2. Per case, take the event that happened last.
3. Filter out the cases where the last event is at the end of the process. Thus events where the to-status is "Aanvraag afgewezen", "Wachten op getekende offerte", etc.
4. The remaining cases are grouped per activity, where the activity is the last to-status.
5. The size of each Activity-group is the same as the number of cases present at 01-09-2014 00:00:00. These sizes are put in a column which can be added to our Simulation Model.

6.3 Resources

6.3.1 Estimating the number of employees

Scientific Literature

An organizational perspective can be gained using Organization Mining [48]. This can be done with a dataset where each event has a resource attribute. With such a database it is possible to analyse the relation between resources and activities. Using such information, there are techniques to learn more about people, machines, organizational structures (roles and departments), work distribution, and work patterns [8].

There are several organizational perspectives [8]:

- Sociometry refers to methods that present data on interpersonal relationships in graph or matrix form.
- Discovering Organizational Structures tries to determine a profile, based on clustering of the resources. Using this, specific roles, such as managers, expert etc., can be characterized.
- Analysing Resource Behaviour can be used to analyse the behaviour of an organizational entity or resource. For example it can be examined how fast an employee works, by checking how many activities a resource has performed in a certain time slot.

For BPS Organizational Mining can be applied to define the various roles that exist within a process. The behavior of a resource can be characterized by a profile, i.e., a vector indicating how frequent each activity has been executed by the resource. By using such profiles, various clustering techniques can be used to discover similar resources [8].

K-means clustering (also known as the Lloyd's Algorithm) [49] can be used for clustering the various resources, based on some variables. A disadvantage of this algorithm is that the number of clusters has to be determined up front [8, 49]. An alternative can be Agglomerative hierarchical clustering, which produces a dendrogram allowing for a variable number of clusters depending on the desired granularity. Another method is provided by [7], who uses Pearson's correlation coefficient to distinguish various clusters of resources.

However, all methods that were described assume that one employee belongs to one role (i.e. are only in one cluster) and use a "top-down" approach determining the clusters. This can be problematic, since it assumes a very rigid distinction of activities per role. This is quite bald, since in practice this is not always the case. For example, it could be that an employee, such as a manager helps on a different cluster, because this cluster is understaffed. Additionally to this perspective, we suggest a "bottom-up" approach where various activities are combined in a cluster. A method that can support this is Association rule learning.

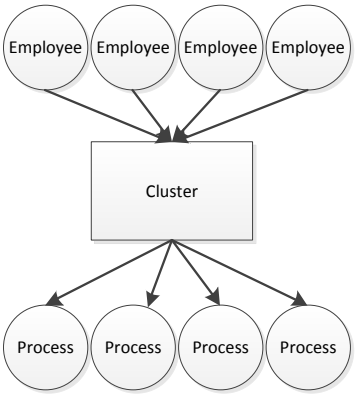


Figure 17 - Bottom-up Approach

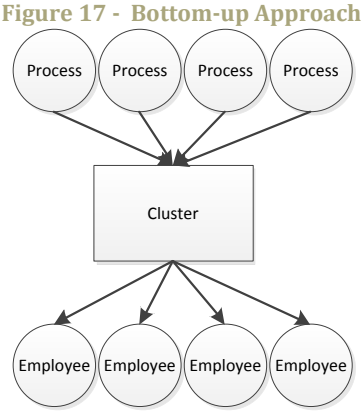


Figure 18 - Bottom-down Approach

Association rule learning is a popular method for discovering interesting relations between variables in large databases [50]. It is most used for marketing. For example, if, it states that if a customer has potatoes and onions in his basket, it will most likely also buy burgers. Using this as an analogy, we can look at individual activities as a "basket-item" and can research, which activities occur often by the resource.

For Association rule learning the Apriori algorithm can be used [51]. This is the most popular and researched algorithm. While looking at the results, three outcomes are important:

- X = The frequency an item or a set of items A occurs
- Y = How many of the X items containing A, also contain item B, this is also known as the support
- Confidence = Can be calculated by Y/ X

Another problem with above methods is that they assume that the number of available employees per day is equal to the total amount of employees and thus does not keep in account vacations and sickness, etc. One can assume this is not the case in real life.

Requirements

Depending on the role a user of the system has, the system provides them with an "inventory" of tasks to be completed. Per role it is different which activities are part of this inventory. This provides a one-to-many relation between the resource and various activities.

So, as already shown in Figure 8 when looking from an activity perspective, each activity is part of a group, or cluster, of activities.

Furthermore, Table 1 shows that the number of employees working on an activity differs with each day. The skewness[5] and kurtosis[6] (near 0) indicate that the number of employees in distributed normally [52].

While assessing the resources, we will make the following assumptions:

- Contrary to the front offices who provide the input for the process, working is only done on working days and not on weekends
- The execution of an activity is not interruptible. Hence, an employee can only leave when he has completed a process.
- The amount of persons working on a cluster of activities can change per day.
- During a day, an employee only has one role.

Final solution

As mentioned earlier, there are two major flaws in the current methods when applying organizational mining for simulation purposes:

1. They assume that every employee only has one role (i.e. Manager, consultant), where the role is related to a certain set op activities. They ignore that, for example, a manager can help with daily (lower-level) activity when it is very busy, which according to described methods cannot be done, because there is a direct relation that states that a manager does not perform these tasks, or that an employee can have multiple roles (i.e. a senior employee), which according to described methods cannot be done, because the relation between employee-role is one-to-many.
2. They assume that an employee is fully available (thus never has vacation and is never sick) and/or are available based on a schedule that either is, or is not, mined from the data.

To cope with above flaws, we change the discrete approach to a more flexible and stochastic approach:

1. Clustering the activities in several clusters
2. Per cluster and per day determining the number of employees that performed at least one activity that is part of this cluster.
3. Determining per working day and per cluster a statistical distribution that describes the number of available employees.

Clustering the activities

A first try was done using association, but failed, because the number of "No" (activity was never performed by an employee) severely outnumbered the number of "Yes" (activity was performed at least once per day on average by an employee). K-Means was also rejected, because we do not know the number of "activity groups", which is a prerequisite for K-Means.

For above reasons we choose to use Agglomerative hierarchical clustering. This method provided us with five clusters (Appendix E: Dendrogram), consisting of one or more

[5] Skewness is a measure of the asymmetry of the probability distribution.
[6] A higher kurtosis means more of the variance is the result of infrequent extreme deviations

activities. Additionally we have the cluster of activities that do not have a user, the automated activities.

Determining the number of employees

The number of employees was determined by:

1. Filtering out the data that have a "From status" that are not part of the cluster of activities.
2. Grouping the state transitions per day
3. Determining the number of unique "Employees"
4. Add the number from Step 3 to an array, depending on the working day. Every working day has its own array

Determining per working day and per cluster a statistical distribution that describes the number of available employees

A hypothesis can be made whether the number per day is distributed normally. Table 1 supports this hypothesis, since the Skewness and Kurtosis are close to zero for most of the days.

	Monday	Tuesday	Wednesd	Thursday	Friday
Meam	144,6182	154,2632	141,6316	149,7222	136
Variance	243,8724	253,2816	374,4783	307,3117	238,6429
Skewness	-0,18969	-0,14645	2,967433	0,084016	-0,41207
Kurtosis	-0,00448	0,120405	16,30252	0,238943	0,189517

Table 1 - Mean and Variance of No of Employees working on Activity 6

To describe a normal distribution, only two parameters are needed: 1. the Mean and 2. the Variance. As a last step these are calculated for the arrays that were obtained in previous step.

6.4 Queue disciplines

Scientific Literature

The queue discipline (or service policy, dispatching rule, queuing policy) determines the priority rule that is applied to the queue, i.e. in which sequence the entities in the queue will be processed. Classical queue disciplines include[10, 53]:

- First-in first-out (FIFO), i.e. entities that are in the queue the longest are processed first.
- Last-in first-out (LIFO), i.e. the last entity that arrived in the queue is processed first.
- Priority rules, where priority rules promote particular entities.

[54] provides additional disciplines:

- Earliest Due Date (EDD), i.e. entities that have the nearest due date are processed first.
- Service In Random Order (SIRO), i.e. entities are processed in random order.
- Shortest Processing time (SPT), i.e. entities that are expected to have the least processing time are processed first.

When it comes to scientific literature on the application of Process Mining on finding Queue disciplines, we can be brief: there is none[10]. Senderovich [55] distinguish three timestamps that are important when assessing the length of the queue for business processes:

- The enqueue time
- The start time
- The stop time

[10] note there are two things important assessing queues: whether the process

Requirements

As found out earlier in this paper, finding the start time of the execution of an activity is far from trivial. It has little impact if we leave this start time out of the scope and assume that the start time is the same as the stop time of the previous activity: only when a request "overtakes" another request, it will affect the method.

Besides that, we assume that per activity, per timestamp at most one state transition takes place. It could be that within a second two or more state transitions can take place, but that chance is negligible.

Furthermore, we will assume that a request cannot leave the queue: it needs a state transition to leave an activity.

Also we leave queue with a length of 1 out of the scope, since a resource does not have any choice picking the next request to process.

The bank has a Service-Level Agreement, where it promises to process a request within three days. For that reason we set the due date at three days after the request has arrived.

Finally, we have to address that it is likely that in our case there is quite some correlation between some of the Queuing Discipline. For example: Before a HDN request has to be replenished, no manual labour has been performed yet on that request. Because the Due date is relatively the same for all requests, there is a 100% correlation between the FIFO and the EDD discipline, because they arrive at this action relatively at the same time (only seconds after the start of the request). Also we assume that there is a correlation between SPT and LIFO: for some requests, some interconnected activities are processed successively by the same resource. We assume that they do this, because the resource has (short-termed) knowledge of the requests, which minimizes the processing time. Because such requests are processed successively, they arrive last, but are processed first, thus have a LIFO service policy. Figure 19 is an example of such a request.

Image deleted for Privacy reasons

Figure 19 - Example of a request that is processed successively

The queue length differs per activity as can be seen in Appendix B. When looking at this graph, for the activities with larger queues a peak can be seen at the end of 2014 and the beginning of 2015. This is most likely caused by the increased arrival rate at the end of 2014.

Final solution

Since no solution is provided by the scientific literature, we have to come up with a method from our own. Therefore we provide the following solution:

Determining the Queue per State transition

The first step is determining the queue before each state transition. To obtain this queue we have to know which request are in a certain status, but did not yet status through to another set. For example, in Figure 20, only Requests 1 & 3 are in the queue, because Requests 2 & 4 have already left the queue and Requests 5&6 are not yet in the queue.

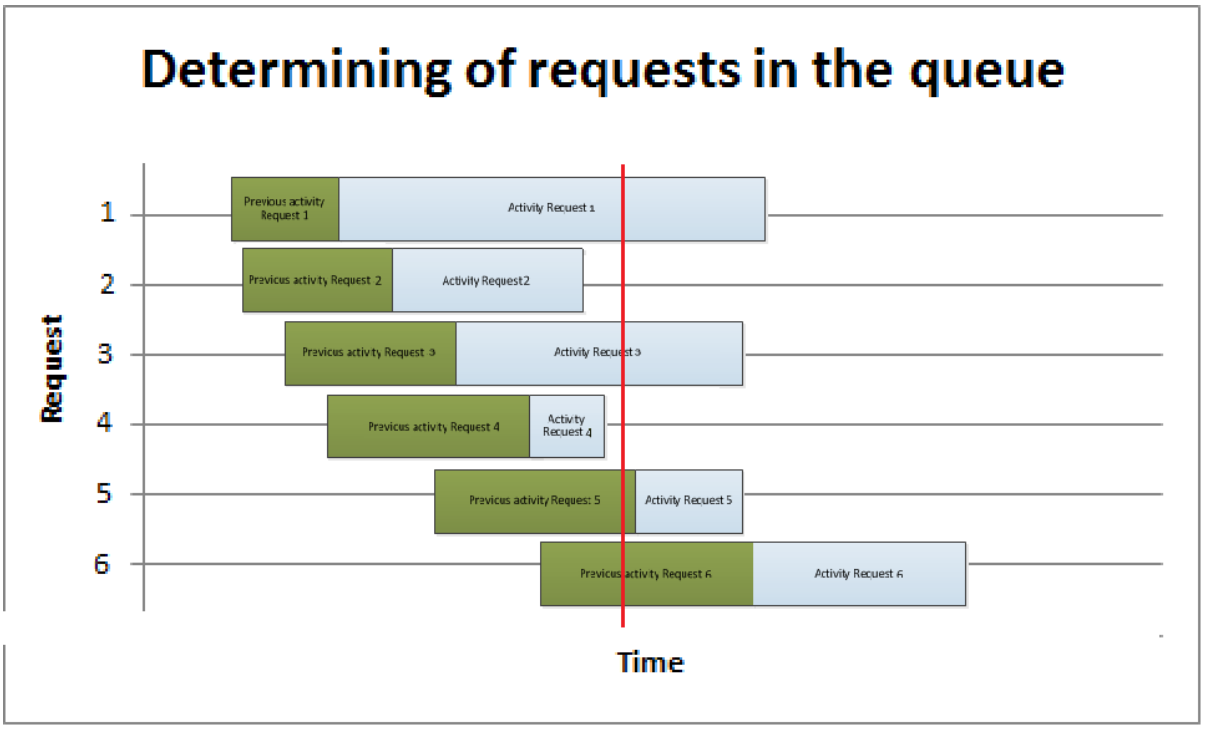


Figure 20 - Obtaining requests in a queue

The method used to determine consists of two parts. First we have to determine which request leaves the queue at a certain time. After that we have to determine what the position of this request was, when sorting all the requests in the queue at that time on the arrival time. For example, if the service policy is a perfect FIFO, the request that leaves the queue is always the one with the oldest arrival time at the activity and has the first position.

Step 1: Determining which request leaves the queue at a certain time:

Let E be a list of events E_i,

$$E =< E_1, E_2, ...E_n >$$

Where:

$$E_i =< N_i, t_{added_i}, t_{removed_i} >$$

with N_i is the activity number of event i, t_{added_i} the timestamp when the event was added to the queue of activity N and $t_{removed_i}$ the timestamp when the event was removed from the queue of activity N. This set is sorted on $t_{removed_i}$. From the database, N_i is the

"FromStatus", $t_{removed_i}$ is the timestamp of the state transition and t_{added_i} is the timestamp of the state transition of the previous state transition on a certain entity.

Then for all activities that exist within E:

$\forall N \in E$:

T is the set of all timestamps $t_{removed_i}$, where t_1 = Min($t_{removed_i}$) and t_m = Max($t_{removed_i}$):

$$T =< t_1, t_2 ...., t_m >$$

Then for every t that exists within *T*:

$\forall t \in T$

We determine all events that are in the queue of Activity *p* at moment *t* as the set:

$$Q(t,p) = \{(N, t_{arrived}), (N, t_{arrived}) \in E \mid t_{arrived} < t \wedge t_{removed} > t \ \wedge N = p\}$$

Furthermore, we determine all events that are in the queue of Activity p at moment t-1 at the set:

$$Q(t-1,p) = \{(N, t_{arrived}), (N, t_{arrived}) \in E \mid t_{arrived} < t-1 \wedge t_{removed} > t-1 \ \wedge N = p\}$$

Then we obtain the set difference $D(t,p)$ between and $Q(t-1,p)$ and $Q(t,p)$:

$$D(t,p) = Q(t-1,p) \backslash Q(t,p)$$

Note that $D(t,p) \neq \emptyset$, only for $t_{removed} \in E$. $D(t,p)$ are all events which are in Q(t-1,p) and are not in Q(t,p), that is all events that left the queue at moment *t-1*.

Using the method described above we can find which requests leaves the queue.

Step 2: Determining the position of the leaving request:
Finally, we obtain the index set I of Q(t-1), where *o* = #Q(t-1) (the cardinality of Q(t-1)).
$I = \{1, 2, ..., o\}$

Furthermore we obtain the indicator function for subset D(t,p) on set Q(t-1,p). This function contains of only zeros, with one exception, the position of the request that leaves the queue is 1:

$$\mathbf{1}_{\mathbf{D(t,p)}}(q(t-1,p)) := \begin{cases} q(t-1,p) = 1 & if \ \ q(t-1,p) \in D(t,p) \\ q(t-1,p) = 0 & if \ \ q(t-1,p) \notin D(t,p) \end{cases}$$

If we then multiply $\mathbf{1}_{\mathbf{D(t,p)}}(q(t-1,p))$ with I, we get an array L consisting of zeros with only one value non-zero, which is equal to the position of the request in the array:

$$L = \mathbf{1}_{\mathbf{D(t,p)}}(q(t-1,p)) \cdot I$$

We do not need all the zeros, so we then take the sum of L, pos:

$$pos = \sum_{i=1}^{o} (l_1, l_2, ..., l_o)$$

We then, per activity, put all *pos* in an array Pos:

$$Pos = [pos_1, pos_2, ..., pos_s]$$

where s is equal to the cardinality of the subset N:

$$s = \#(N \in E)$$

Furthermore we put all the cardinalities of Q(t-1,p) in an array, because we possibly want to assess the position of the leaving request with the size of the queue at that moment:

$$Size = [\#Q(t-1,p)_1, \#Q(t-1,p)_2, ..., \#Q(t-1,p)_s]$$

This results in two arrays, one with the position of the leaving request and one with the queue size.

Finding position in the queue
In order to assess which queue discipline is used, it is important to find what the position is of a certain attribute compared to the others in the queue:

For FIFO and LIFO, it is important to find out what the arrival time of a queue-leaving request is, compared to the others. If it is the request with the earliest arrival time, for most of the times, we can state that FIFO is used. If mostly the request leaves the queue with the latest arrival time at the activity, we can state LIFO is used.
For EDD, it is important to find out what the due date of a queue-leaving request is at an activity compared to the others. If mostly the request leaves the queue with the earliest due date, we can state EDD is used.

In a first attempt we tried to set a benchmark to determine whether a policy is applied. An example of such a benchmark was" "when 90% of the queue-leaving requests are in the 5% of the requests with the earliest due date at the moment of leaving, we assume EDD." However this resulted in only a few results: for the major of the activities no policy was within the benchmark.

A closer look at a process that was within the benchmark, gave us the following plot:

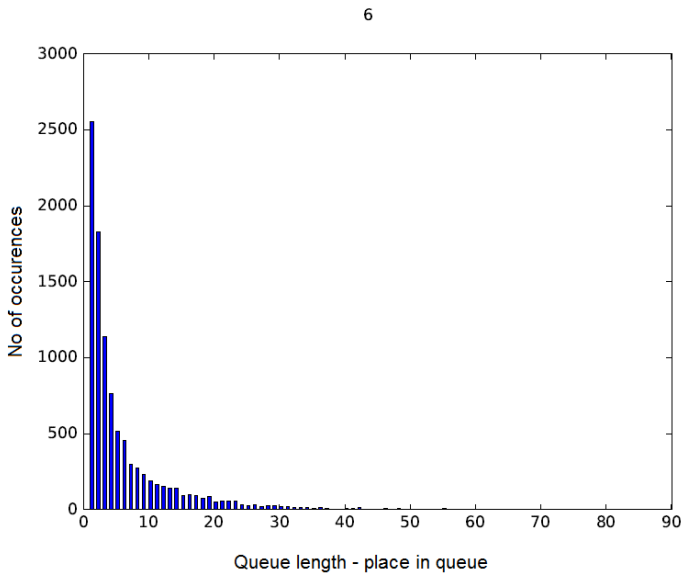


Figure 21 - Plot of a typical LIFO service policy

Note that this graph is heavily skewed to the right. When taken a closer look, a lot of other activities show similar pattern, only less skewed.

For that reason we state the following proposition:
A policy applies when its plot is significantly (p<0.05) skewed to the right

To test for significance, we use the following test [56]:

Z_{Skew} = Skewness/SES where $SES = \sqrt{\frac{6n(n-1)}{(n-2)(n+1)(n+3)}}$

Where SES is the Standard Error of the Skewness, n is the number of total observations and Z_{Skew} is the test statistic.

If $Z_{Skew} > 2$, then the population is very likely skewed positively and thus skewed to the right.

If for multiple policies $Z_{Skew} > 2$ applies, the policy is chosen with the highest Skewness. If there is no pattern is found, we can state that either SIRO (could also be Priority rules) is used.

Policy not taking in account state transitions with a low processing time

In Figure 19 an example was shown of a request that was processed in a short time period. Furthermore, it can be noticed that when using the method as described above the average Skewness of activities marked as "EDD" and "FCFS" is quite lower than those who are marked as "LDD" and "LCFS".

FCFS	2,575528
LCFS	3,090308
EDD	1,517672
LDD	3,374163

Table 2 - The average skewness of the policies

We assume this is because the results are cluttered by a request such as the one in Figure 19. For this reason we chose to eliminate state transitions with a low processing time (under 30 seconds) to see whether this affects our results.

This method only showed little improvement, with one exception only: the activity “X” is clearly affected in the tail, as is shown in Figure 22 and Figure 23.

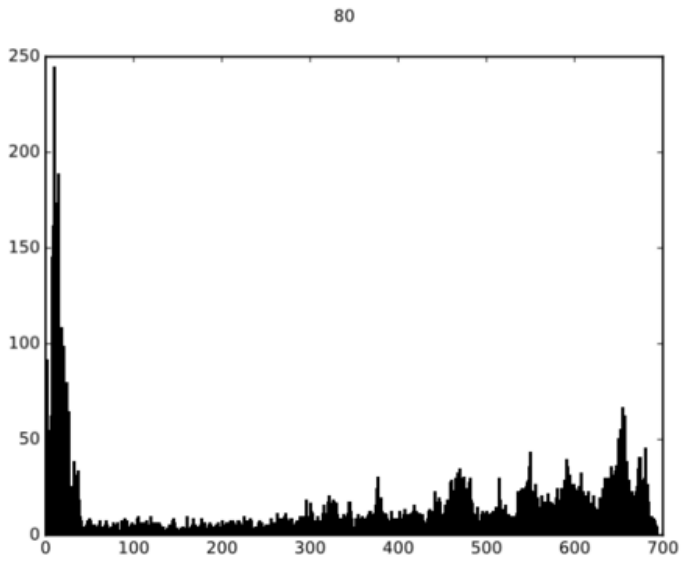


Figure 22 - "X" before Data Cleansing

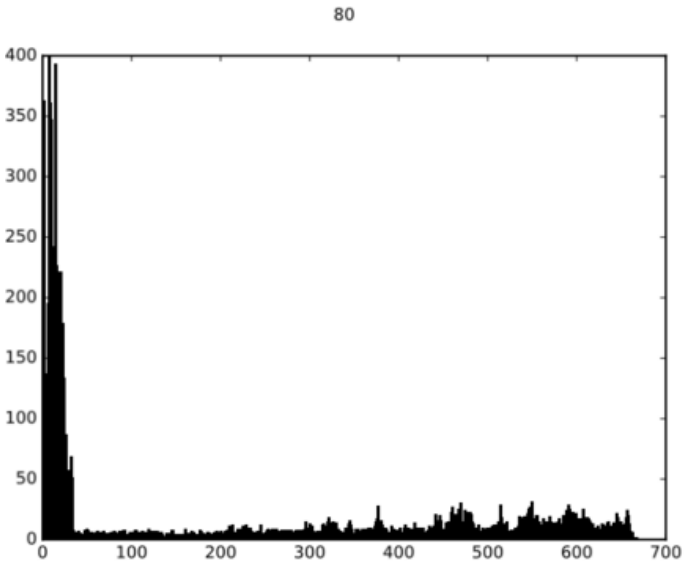


Figure 23 - "X" after Data Cleansing

6.5 Gateways & Sequences

6.5.1 Process discovery

Scientific Literature

Determining in which sequence the activities occur, or Process Discovery, is the base of Process Mining and the most heavily researched topic in this field. [8]distinguished the four most popular Process Discovery algorithms: The α-algorithm, Heuristic Mining, Genetic Mining and Fuzzy Mining.

These algorithms differ when assessed on the following characteristics:

Representational bias: is the class of process models that can be discovered. There are several types possible, such as petri nets, Business Process Model and Notation (BPMN) and Yet Another Workflow Language (YAWL), or Causal nets; a representation tailored toward process mining.

Noise: is the amount of rare and infrequent behaviour or outliers. Noise should not be included in the discovered model:

1. Users typically want to see the mainstream behaviour.
2. It is impossible to infer meaningful information on activities or patterns that are extremely rare.

Noise can be removed by pre-processing the log, or the discovery algorithm can abstract from noise while constructing the model.

The assumption of completeness: is the assumption that all possible behaviour is in the log. The various methods differ in to what extent they assume completeness. This assumption can lead to overfitting the model.

The α-algorithm

The α-algorithm was one of the first process discovery algorithms that could adequately deal with concurrency: activities modelled in parallel but executed in sequence. The α-algorithm should not be seen as a very practical mining technique as it has problems with noise, infrequent/incomplete behaviour, and complex routing constructs[8], but it forms the base of other algorithms.

Heuristic mining algorithms use a representation similar to causal nets, a representation tailored towards process mining. Figure 24 is an example of such a causal net. Dots that are arced together (bindings) are fired or consumed together. C-nets are a more suitable representation for process discovery [8].

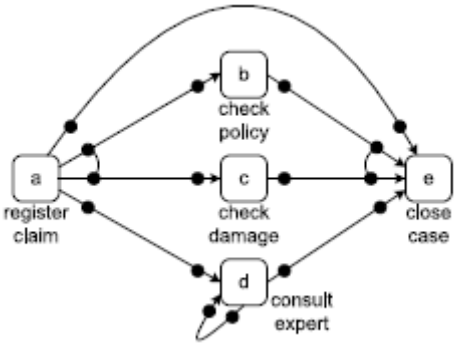


Figure 24 - An example of a Causal net (Van der Aalst, 2011)

Heuristic mining consists of several steps (which is implied in the name "Heuristic"):

1. Determine a frequency table. This is a table which counts the times event B is directly followed by event A.
2. Calculate the determination ratio.
3. Generate a dependency graph can be generated, based on some thresholds.

Genetic Mining

Contrary to Heuristic Mining, Fuzzy Mining and the α-algorithm, which are deterministic, Genetic Mining is evolutionary [8]. An analogy with Darwin's Theory can be made here:

1. Initialization: The first prototype was made (i.e. Adam & Eve or Homo Habilis)
2. Selection: the fitness of each individual is computed. The best (i.e. the alpha males) are selected and moved to the next generation.
3. Reproduction: The selected parent individuals are used to create new offspring (i.e. new birth)
4. Termination: A final form with the best fit is generated (i.e. Homo sapiens sapiens currently).

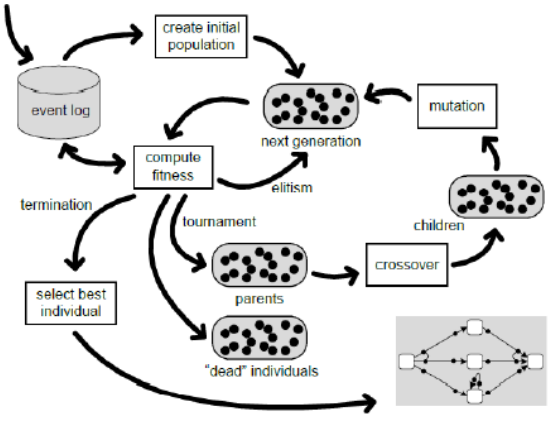


Figure 25 - Genetic Mining Process

Fuzzy Mining

The Heuristic approach is quite generic and can be applied to other representations. An example of this is the Fuzzy Miner [57], which forms the basis for the Disco software. A big advantage of this method is that it can aggregate sub-processes, depending how general you want the view to be.

To do so, Fuzzy Mining uses roadmaps as an analogy [58]:

Aggregation: To limit the number of information items displayed, maps often show coherent clusters of low-level detail information in an aggregated manner. One example are cities in road maps, where particular houses and streets are combined within the city's transitive closure. Fuzzy mining does this by putting multiple process-steps in one step.
Abstraction: Lower-level information which is insignificant in the chosen context is simply omitted from the visualization. Examples are waterways, which are of no interest in a map meant for cars. Fuzzy mining does this by leaving out infrequent paths.
Emphasis: More significant information is highlighted by visual means such as colour, contrast, saturation, and size. For example, maps emphasize more important roads by displaying them as thicker, more colourful and contrasting lines, such as the Dutch Highways are displayed on the maps. Fuzzy mining does this by making basing the width of the arrow on the number of times that path is taken.
Customization: There is no one single map for the world. Maps are specialized on a defined local context, have a specific level of detail (city maps vs. highway maps), and a dedicated purpose (map of the subway-net of London vs. a roadmap of London). Fuzzy mining does this by letting the user decide how much of the paths and activities are shown in an interactive manner.

Requirements

Which process discovery technique to choose, depends on the process. Luckily, some process models already exist within Topicus. At first glance it can be concluded that this process is very complex, with a lot of possible paths, with a lot of splits. From experts within Topicus, some characteristics can be eliminated:

- An application for a mortgage can only have one status, so there is no parallelism.
- The main path has not changed over the years, only extension have been made, thus concurrency can be eliminated.

There are several aspects of a process which can help us assess which algorithm is the most suitable for us which of the characteristics, as mentioned in Appendix B, match with the Topicus FORCE process (Table 3).

Characteristic		**Assesment**
Choice:	30-40	Big influence
Parallelism:	0	No influence
Loop:	20-30	Big influence
Invisible tasks:	0	No influence
Duplicate tasks:	0	No influence
Non-free Choice:	0	No influence
Nested Loop:	5	Little influence
Number of traces:	240	Big influence
Number of distinct traces:	240	Big influence
Number of events:	+ 20 million	Big influence
Minimum trace length:	22	?
Average trace Length:	?	?

Maximum trace length:	?	?
Noise:	Probably small	Little influence
Number of activities:	142	Big influence

Table 3 - Assessment of characteristics of Topicus process & event log

Final solution

From Table 3 - Assessment of characteristics of Topicus process & event log we select all the characteristics that are of some influence (Table 4). In Table 5 the final assessment is done. As can be seen, Genetic Mining scores best, followed by Heuristic Mining and as last the α-algorithm.

Characteristic	Alpha Miner	Alpha Miner+	Alpha Miner++	Heuristics M.	Genetic Miner	DT Genetic M.
Choice:	$S\downarrow$	$P^b\downarrow$ $S\downarrow$	$F^b\uparrow$ $P^b\uparrow$ $F^a\downarrow F^b\downarrow$ $P^a\downarrow P^b\downarrow$	$P^b\uparrow$	$F^a\downarrow$ $P^a\downarrow$	$F^a\downarrow$ $P^b\downarrow$ $G\uparrow$
Loop:	$F^a\downarrow F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $S\uparrow$	$G\downarrow$ $S\uparrow$	$S\uparrow$	$F^a\uparrow$ $P^a\uparrow$	$F^b\uparrow$
Nested loop:	$F^a\downarrow F^b\downarrow$ $P^a\uparrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^a\downarrow$		
Number of traces:				$P^b\downarrow$	$F^a\uparrow$ $P^a\uparrow$	$F^a\uparrow$ $G\downarrow$ $S\downarrow$
Number of distinct traces:			$P^a\uparrow$		$F^a\uparrow$	
Number of events:				$P^b\uparrow$	$F^a\downarrow$ $P^a\downarrow$	$F^a\downarrow F^b\uparrow$ $G\uparrow$ $S\uparrow$
Noise:	$S\uparrow$	$F^b\downarrow$ $P^b\uparrow$	$S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^a\uparrow$	
Number of activities:	$F^a\downarrow F^b\downarrow$ $P^a\uparrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^a\downarrow$ $P^b\downarrow$ $S\uparrow$	$P^a\uparrow$	$F^a\downarrow F^b\downarrow$ $P^a\downarrow P^b\downarrow$ $G\downarrow$

Table 4 - Influence of relevant factors on the various algorithms

Algorithm	Fitness	Generalizability	Precision	Simplicity
α-Algorithm	↓	=	↓	↑
Heuristic Mining	↓	=	=	↑
Genetic Mining	=	↑	↓	↑

Table 5 - assessment of the Topicus process on the various algorithms

Earlier on we mentioned that the α-Algorithm is not practical suitable for process mining. Above table confirms this finding. For these reason we will definitely not use α-Algorithm. Fuzzy mining is not part of above comparison.

Genetic Miner can be used when we need to mine logs with noise, handling of duplicate task names, local and nonlocal non free choice constructs and invisible task [59]. However, it is expected that these characteristics are not part of the Topicus process. Fuzzy Mining is especially applicable to increase the comprehensibility. Because the main purpose of this process is also to get increased insight into the more global process, this will be our technique of choice. As an extra advantage for this choice, is that we can use Disco to run this algorithm, which is easier to use than ProM. A disadvantage is that the discovered model cannot be converted to a Petri net, so we cannot use Token Replay for the Process Conformance. For that reason we choose Genetic Mining as a second choice, if the discovered model and the original model do not conform.

6.5.2 Process Conformance

After discovering the process one can choose to verify whether or not this process is the same as documented. This is called Process Conformance. For this, three methods are commonly used: Delta Analysis, Comparing Footprints and Token Replay.

Delta Analysis

Delta analysis is not a conformance technique like others, because results may be flawed or not representative if the event log is not complete. Besides that, the final result is not quantifiable [60]. However, it serves the same need: comparing the real-life events with the process model. For that reason this method will be discussed in this chapter.
The results can be assessed in three different ways: 1. Visually 2. Inheritance of behaviour and 3. Change regions [60]:

Visually: The visual assessment is quite basic. One puts the discovered model next to the original model. Visually the assessor tries to find differences between the two models.

Inheritance of behaviour: With inheritance, the assessor creates a model using only the processes both the discovery and the original model agree upon.

Change regions: The change region is determined by comparing the two process models and extending the regions that have changed directly by the parts of the process that are also affected by the change of going from one process to the other, i.e., the syntactical affected parts of the processes are extended with the semantically affected parts of the processes to yield change regions. Using this, we can we can highlight the parts of the processes affected by the differences between the predefined and discovered models.

Token replay

In order to assess the conformance of the process model, a measurement is needed. For this reason, fitness is introduced. Fitness can be explained as “the proportion of behaviour in the event log possible according to the model” [8]. This number can be obtained by replaying the log on the process model. This process model should be in the form of a petri net.
This measurement makes use of the mathematical modelling language Petri Nets. Petri nets consist of a process model that makes use of nodes which represent transitions from one state to another and conditions. Places in a Petri net may contain a discrete number of marks called tokens. They model the changing states and locations of objects. Any distribution of tokens over the places will represent a configuration of the net. For more information about Petri Nets, we refer to [61].
Fitness, for a specific case with trace σ within Net *N* can be calculated by:

$$Fitness(\sigma, N) = \frac{1}{2}(1 - \frac{Missing\ tokens}{Consumed\ tokens}) + \frac{1}{2}(1 - \frac{Remaining\ tokens}{Produced\ tokens})$$

A token is *Produced* if a token arrives at a certain condition.
A token is Consumed if a token leaves a certain condition
A token is *Missing* if the next transition is not enabled to fire. A token is added then at the place after that transition.
A token is *Remaining* if, after all transition have taken place and this token has not reached the End-note.
This formula can be extended from one case to multiple cases of paths. And thus, the fitness of a model N for event log L can be assessed by:

$$Fitness(L, N) = \frac{1}{2}(1 - \frac{\sum Missing\ tokens}{\sum Consumed\ tokens}) + \frac{1}{2}(1 - \frac{\sum Remaining\ tokens}{\sum Produced\ tokens})$$

Comparing footprints

From both a process model, and an event log, a footprint can be derived.
For example, Table 6 shows a footprint of a played-out model. If we now use an example event log, and change this into a Footprint (Table 7), we can compare these footprints. As the final result Table 8 shows, there is a discrepancy in the flow between nodes B to E, where it appears that processes C and D are processed sequentially instead of parallel.
The number of correct predicted relationships can act as a KPI:

$$\frac{Number\ of\ correct\ predicted\ relationships}{Total\ number\ of\ relations} * 100\%$$

For this example, that is: 30/36*100=83,33%

	a	b	c	d	e
a	#	→	#	#	#
b	←	#	→	→	#
c	#	←	#	||	→
d	#	←	||	#	→
e	#	#	←	←	#

Table 6 - An example Footprint

	a	b	c	d	e
a	#	→	#	#	#
b	←	#	→	#	#
c	#	←	#	→	#
d	#	#	←	#	→
e	#	#	#	←	#

Table 7 - Alternative Footprint

	a	b	c	d	e
a					
b				→ : #	
c				|| : →	→ : #
d		← : #	|| : ←		
e			← : #		

Table 8 - Comparison of the footprints

6.5.3 Gateways

Scientific Literature

There are two types of gateways: splits and joins. A split is the situation where one activity is followed by more than one activity. A Join is the other way around: it is the situation where multiple activities are followed by one activity.

Scientific literature distinguishes [8, 10] three type of splits and three type of joins.

Split:
An XOR split is a scenario where an activity is followed by at most one of the following activities.
An OR split is a scenario where an activity is followed by one or more of the following processes.
An AND split is a scenario where an activity is followed by all of the following activities.

Joins:
An XOR join is a scenario where an activity is executed after at most one of the preceding activities has happened.
An OR join is a scenario where an activity is executed after one or more of the preceding activities have happened.
An AND join is a scenario where an activity is executed after at all of the preceding activities have happened.

Requirements

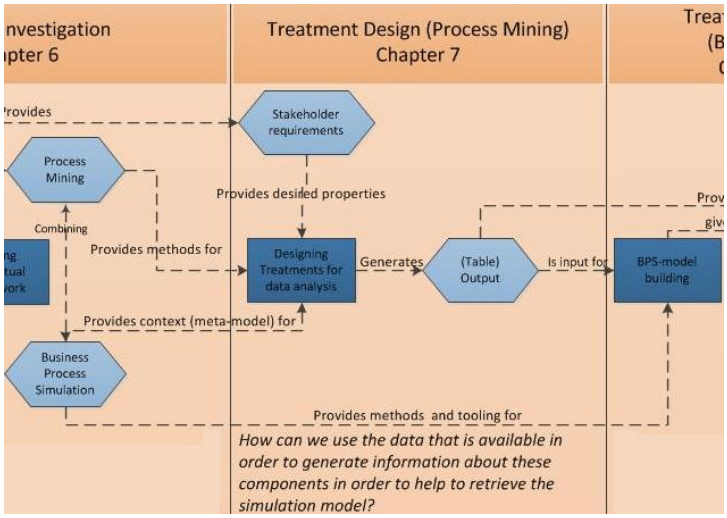
As mentioned earlier, the FORCE engine makes use of states and state transitions, where an entity flows through the process one state at the time.

Final solution

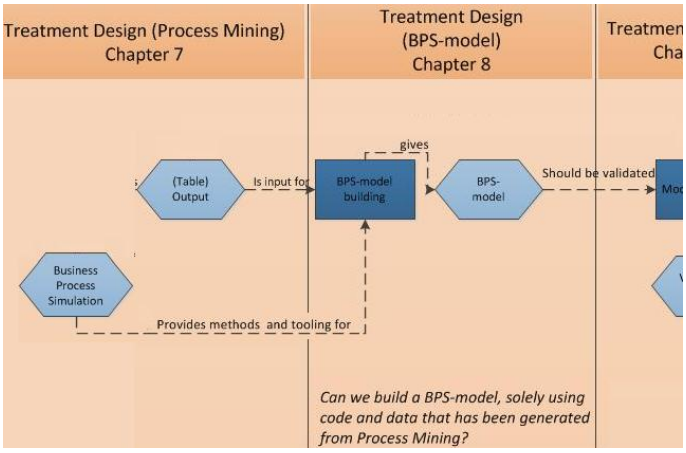
The OR and the AND-join and split require that an entity can be present at multiple activities The FORCE engine does not allow this, since an entity can only have one state. Through the process of elimination we can conclude the all the gateways in the force process are XOR-splits or joins.

6.6 Conclusion

In this chapter we went by every component of the BPS-metamodel and derived one or more characteristics for each of these. For entities, we derived the arrival rate; for activities, we derived the processing time and the start number of entities at each activity; for queues, we derived the queue policy; for the sequence we discovered the flow and for gateways, we concluded that all gateways are XOR.



7 Artefact design: Building the Simulation model



In previous chapter we used Process Mining to generate output for each of the characteristics of the various components in the meta-model. In this chapter we show how we use this output as an input for our BPS-model, by the means of a data scheme. This scheme can also be used, if we want to adapt the process or create a totally different one. Furthermore, in this chapter we will discuss the choice of software for the modelling of the BPS, as well the restrictions and capabilities of the model.

7.1 Choice of software

For the choice of software various tools were evaluated. These were:

- Aris Business Simulator
- SimPy (Package that can be used within Python)
- Bizzdesigner
- Tecnomatix Plant Simulation
- Arena Simulation

These packages were evaluated on the following criteria:

- Usability: For the usability the learning curve as well the ease of use of the software were evaluated.
- Licensing restrictions: Most of the packages, except SimPy, cannot be freely used for commercial purposes. Furthermore, there are restrictions for some of these software packages when it comes to usage of academic purposes.
- Statistical modelling capabilities: In previous chapter some results were obtained that used a more “advanced” statistical distributions, such as Negative Binomial and Lognormal. Some software packages only make use of the basic distributions, such as Normal and Poisson.
- Animation: A model is more likely to be accepted by stakeholders, if there is an animation showing the process. Furthermore, animation can be used for the debugging of the model.[6]

Some of these packages (ARIS, ARENA) were evaluated by Jansen-Vullers and Netjes [19]. This research will be used together with experimentation to examine the various packages. We will score the BPS tools for each of the evaluation criteria ranging from good (++) and neutral (+/−) to bad (− −).

	ARIS	Simpy	Bizzdesigner	Plant Simulation	Arena Simulation
Usability	+	-	+	+/-	+
Licensing restrictions	+/-	++	++	++	--
Statistical modelling capabilities	+	++	+	++	++
Animation	+	--	`+	++	++

Table 9 - Comparison of BPS software packages

First Simpy was dropped as a possibility, because the lack of animation capabilities and limited usability. Furthermore Arena was dropped, because the Model Size Limitations that comes with a student license. Both Aris and Bizzdesigner only support basic statistical distributions and have limited animation capabilities. On the other hand Plant Simulation is harder to learn, because it has more capabilities than needed for our purposes and the programming language that has its own syntax, SIMTALK.

Because Plant Simulation supports us building a generalizable simulation model better, we choose for this package. It will take some additional costs to make the model, because of the complexity of Plant Simulation, but we put up with that.

7.2 Design of the BPS-model

The BPS-model was designed in such a way that the only input needed can be read from tables. This makes it easier to adapt the current process or load another, totally different, process. The model is fully working if all the data in the data scheme (Figure 26) in is loaded. Figure 27 shows how the data in this data scheme is related to the output of our Process Mining methods.

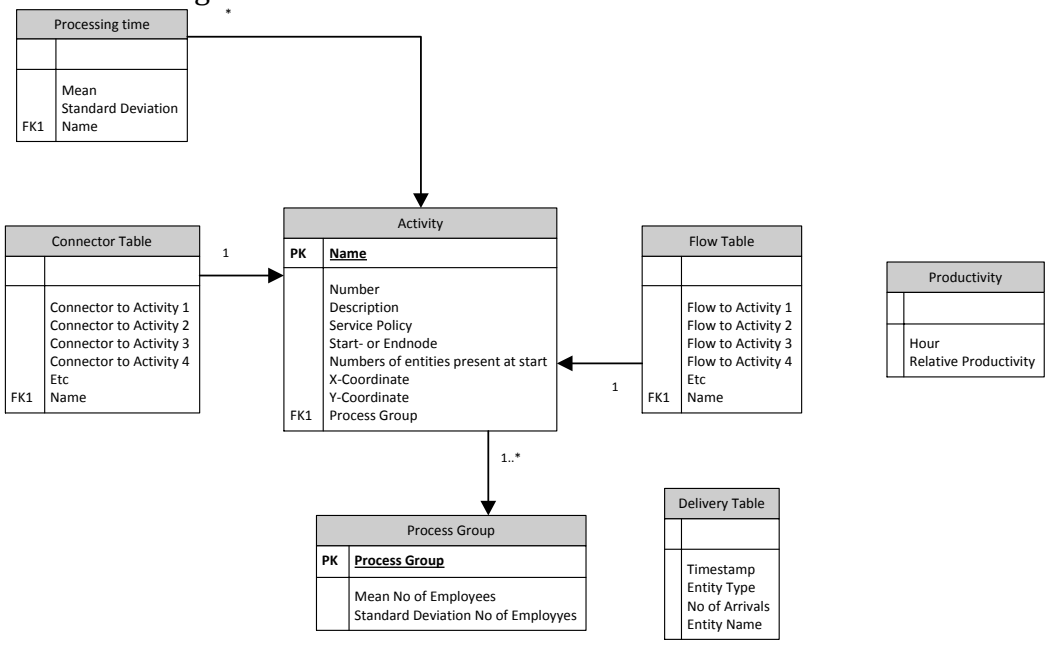


Figure 26 - Data model BPS-model

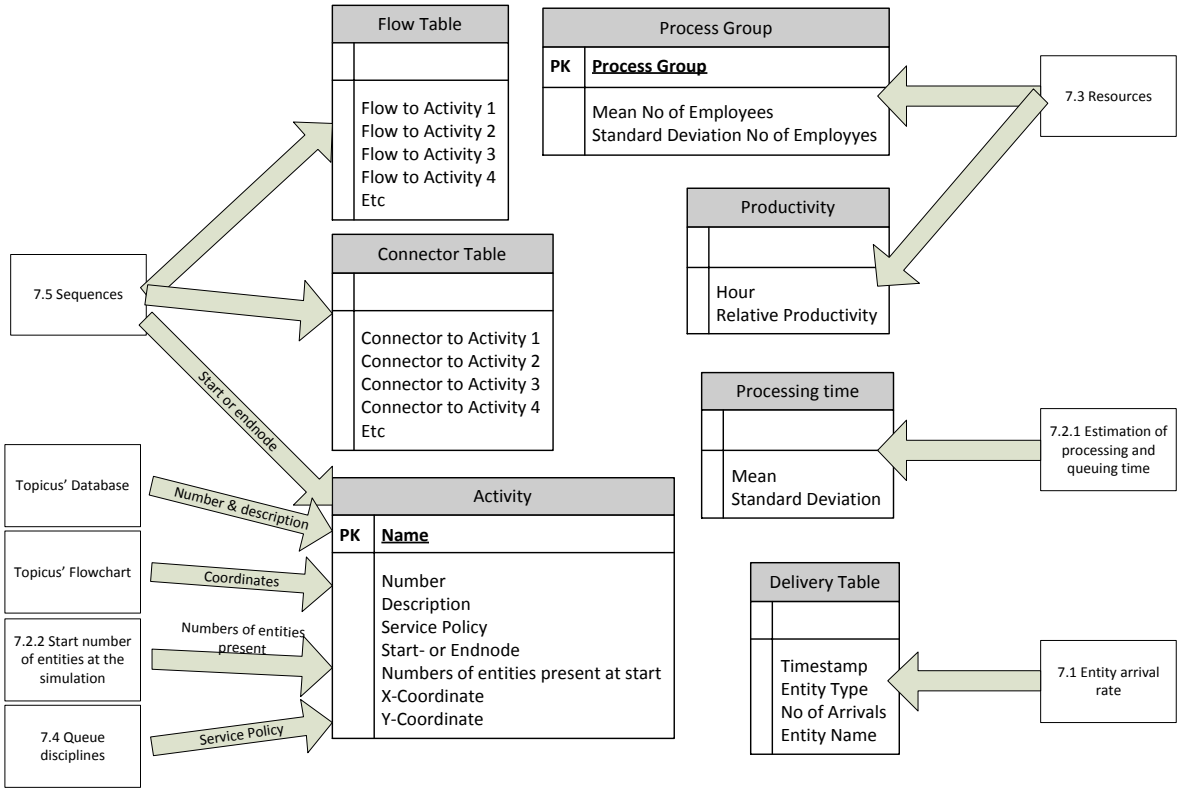


Figure 27 - Relation between PM output and BPS input

7.2.1 Restrictions

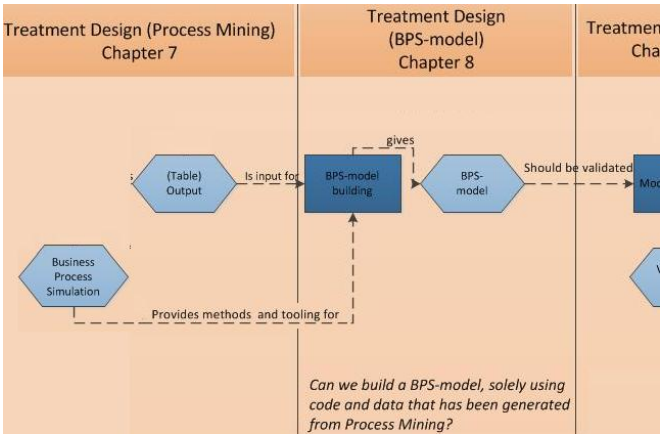
Although we tried to keep the model as general as possible, some assumptions were made:

- Because Plant Simulation does not support a Negative Binomial Distribution, a delivery table has to be used where the number of arrivals in is defined. We have chosen for a Delivery Table that provides deliveries of arriving entities per hour for the duration of one hour.
- During this research, we have done some other generalizations about statistical distributions. These generalizations are added hardcoded (but can easily be changed) to the BPS-model:
    - The numbers of employees per cluster are normally distributed.
    - The processing times per activity are log-normally distributed.

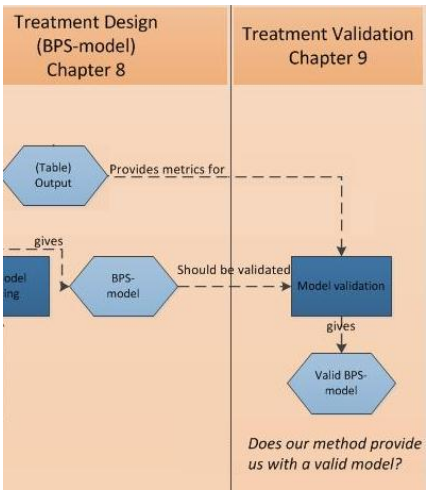
7.3 Conclusion

In this chapter we have substantiated our choice for Plant Simulation: 1. No licensing restrictions, 2. good animation is present and 3. availability of various statistical distributions.

The BPS-model was developed in such a way that is can be applied on every business process, when all the data, as shown the data scheme in Figure 26 is added to the model, with only some small restrictions that can easily be adapted if necessary.



8 Artefact validation



In the previous chapters we have explained the data for and the design of the simulation model. However, this does not guarantee that this simulation represents reality. It is clear that the value of a simulation-based analysis largely depends on the quality and validity of the simulation outcomes. The validity of the representation of the selected key characteristics as discovered in previous chapter is one important aspect that needs to be ensured when approximating a real-life process by a simulation model. Therefore, we want to evaluate how good our simulation model captures the discovered process characteristics.

8.1 Approach

We compare the output of the BPS-model with the results of the PM-model output. This comparison is done for the period 07-01-2014 till 12-31-2014.

First we will validate the functionality of the BPS-model: can the model answer the various business questions that exist within Topicus? For this validation we check how the various relevant scenarios and KPI are incorporated in the model.

After that we will validate various characteristics of the BPS-model. In Chapter 5.3.1 we have identified the components of a simulation model. For the characteristics of these components we need a KPI to assess them in order to conclude whether this component, or set of components is valid or not. We perform this assessment using the output generated from Process Mining tool Disco, together with some data analytics.

In order to assess the simulation as good as possible, we have derived the following KPI's:

Characteristic of model	KPI	Measurement unit
1. Entity arrival rate	Number of arrivals	Total number of arrivals
2. Control Flow	Average number of times of processing per activity	Per Activity: Total times of processing/ Number of Arrivals
3. Throughput time: Processing time, Resources and Queue	Duration of an activity (queue time + processing time); Total time in process	Average Arrival time at Activity I – Departure time at Activity I-1; Average Arrival time at Activity I – Arrival time at Process

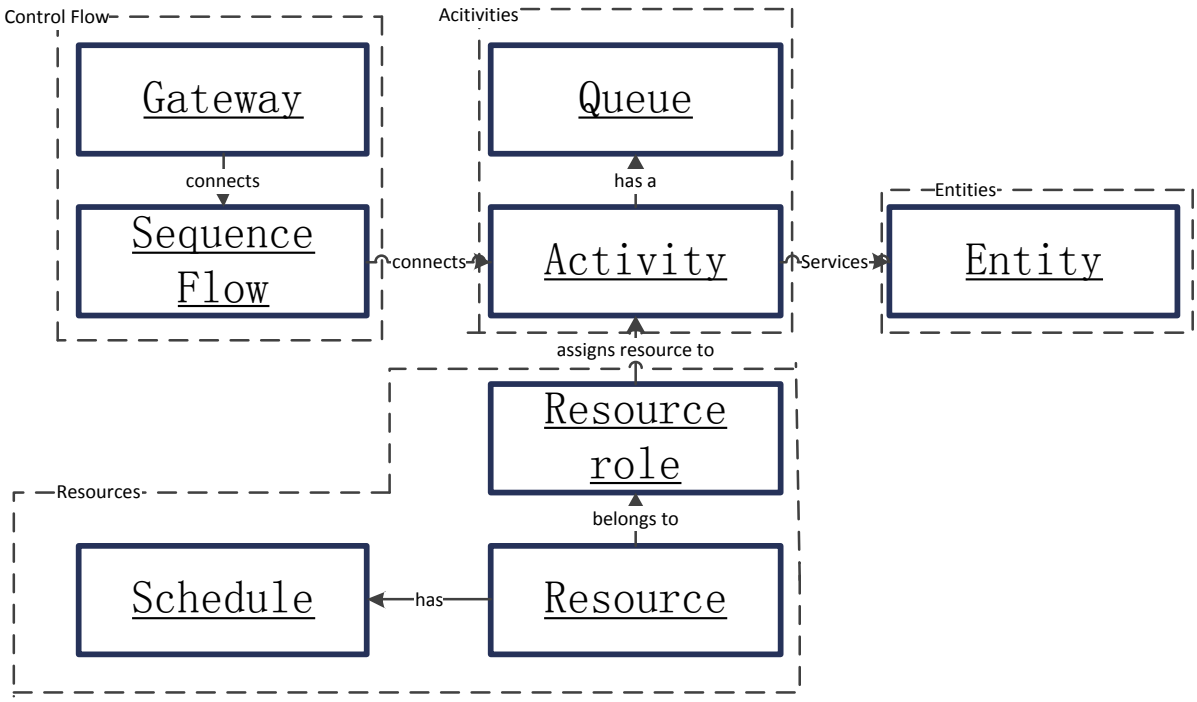


Figure 28 - Components of a simulation model

8.2 Results

8.2.1 Functional Validation

The Business Process Model was created with the purpose to answer various business questions. Each of these business questions relates to some KPI and some scenario. Each of these KPI should be measurable in Plant Simulation and each scenario should applied by changing one (or more) tables in the input.

The business questions are as followed:

1. *What happens with capacity needed for the various departments if external events happen that changes the number of arrivals, in order to keep the same throughput time?*

What?	Which?	In Model?	How?
Scenario	Change in number of arrivals	Yes	Root.DeliveryTable
KPI	Number of employees needed per cluster	Yes	Root.EmployeesDays
KPI	Throughput time	Yes	Entity.TimeinProcess (Custom variable)

2. *What happens with the throughput time if a large change in the process, from conditional to unconditional offering will take place?*

What?	Which?	In Model?	How?
Scenario	Change in process lay-out	Yes	Root.Flowtable; Root.AllConnectors
KPI	Throughput time	Yes	Entity.TimeinProcess (Custom variable)

3. *For most requests, a throughput time of 5 days is promised. However, some requests does not meet this Service Level Agreement (SLA). What happens with the % of requests that meet this SLA if we apply small changes in the process?*

What?	Which?	In Model?	How?
Scenario	Change in process lay-out	Yes	Root.Flowtable; Root.AllConnectors
KPI	Throughput time	Yes	Entity.TimeinProcess (Custom variable)

Conclusion:
Since all the scenarios and KPI's are incorporated in the model, we conclude that the model can answer the various business questions and, thus, the model is functionally valid.

8.2.2 Entity arrival rate

The entity arrival rate heavily depends on the period the analysis is made, as can be seen in Figure 29. Figure 29 shows the moving sum of arrivals for the past half year in the blue line. The horizontal red line shows the total number of arrivals half a year we have generated using a random number generator as an input for the set of negative binominal distributions, which we discussed in Chapter 6.1.1 and stored in our Delivery Table. This total number of arrivals is x, or x per day on average. Figure 29 clearly shows this is a good estimate for the number of arrivals.

On average, the average number of arrivals per day was slightly (7.45%) higher than our calculation: x per day on average, with a standard deviation of x. However, although it comes quite close, there is no statistical difference between these values (p=0.1045).

For the period between 07-01-2014 and 12-31-2014 this number was also slightly higher, x, or x on average per day, with a standard deviation of x. Despite this average number of arrivals is higher than the average all over the year, the test statistic is higher, p=0.1187, which is caused by the lower variance between the number of arrivals in this set.

Image deleted for Privacy reasons
Figure 29 - Actual vs Expected Arrival Rate

Despite the fact that the p-values are just slightly above P=0.10 (thus no statistical difference), we increase our arrival rate with x*100%=9.35% in order to make our simulation model more realistic.

Conclusion:
There is no statistical difference between the number of arrivals we determined using our approach and the actual number of arrivals, for both the average of the entire year, as the average in the period between 07-01-2014 and 12-31-2014. For this reason we can state that our method is valid. However, to increase the accuracy of our model, the number of arrivals was increased.

Control flow

To validate the control flow, we want to check whether if all the actions are performed as many times as expected. Therefor we divide, per activity, the actual number of processed numbers of expected number of processed items, if the actual number of processed

numbers is higher. Otherwise we divide the two the other way around. This ratio ranges from 1 (actual number = expected number) to 2.31 (more than twice the occurrences). However, these numbers can be affected by chance. Figure 30 clearly shows that the ratio drastically declines when the number of processed items increases. For that reason, we choose to take the weighted sum of these ratios, weighted on the number of processed items:

$$\sum_{i=1}^{\#Activities} \frac{\#Processed\ items_i}{\sum_{i=1}^{\#Activities} \#Processed\ items_i} * ratio$$

This weighted sum is 1.008133, which means that our model is 0.8133% off. This number is arbitrarily low, that we assume this number is caused by chance, rounding errors, etc., and can conclude that our flow is valid.

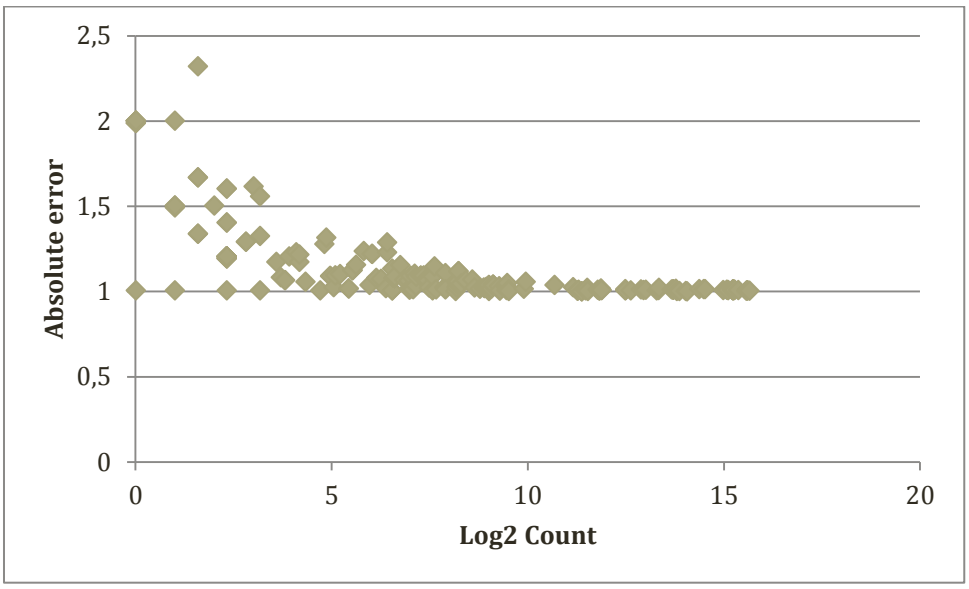


Figure 30 - Absolute error vs. Log of the count

8.2.3 Processing speed

The time it takes for an entity from one activity to another consists of two components: time in the queue en time processing of the processing. Due to the quality of the event log, these cannot be separated using Process Mining, since they are indistinguishable. Thus we have to take the overall time for an activity to process the entity.

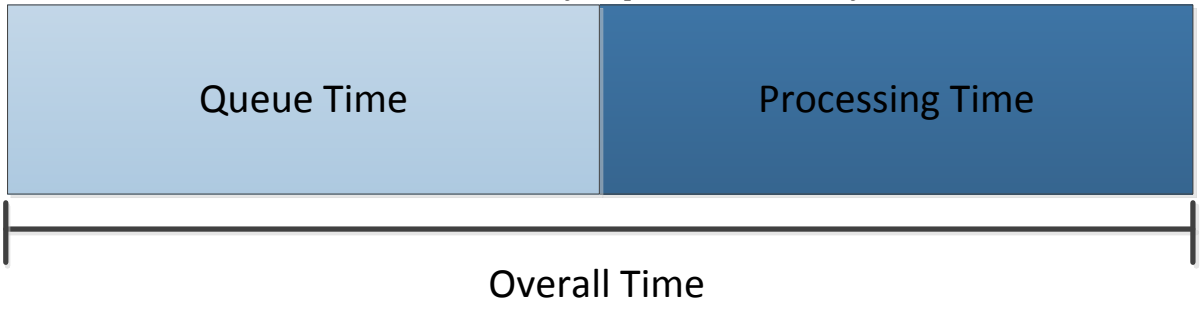


Figure 31 - Time for an activity

Using Process Mining, for each activity in the event log we can calculate the mean and median time of the time difference between when the activity is executed and when its

processor is executed. The same we can do in our BPS-model. We can compare these two with each other to check for validity.

Furthermore, as an extra validation, we have calculated the time difference between when an activity is executed and the starting time of the entity investigated.

The results show that the processing neither of these metrics show real-life behaviour. Changing the number of employees does not affect these results, because the throughput time differs from the real-life throughput time differently per activity.

9 Conclusion, Discussion and Recommendations

9.1 Conclusion

The purpose of this paper is to provide a general method that can be applied on the Topicus/bank case in order to answer the various business questions that exist within Topicus.

During this research we have distinguished the various components of a simulation model: the activity, the resources and their related roles, the queue, the entity and the sequence flow. The better we describe (the characteristics of) these components, using the data, the better the simulation model will be.

We have used various approaches to describe the various characteristics of these components, using existing approaches, adapting existing approaches and coming up with our own approaches.

The table-output of these methods has been used as input for our BPS-model. This model has been created in such a ways, that it can easily be altered by changing the values in the tables.

This had led to a BPS-model that has the potential to answer the various business questions, has a valid arrival rate, a valid flow through the process, but unfortunately an invalid processing speed. Due to this, the current model can answer the business questions about the process that exist within Topicus, but we will not know whether these answer will represent what will happen in the future. It probably will give some hint in which direction they real answer will be, but not more than that.

9.2 Discussion

9.2.1 Generalizability

In order for this paper to contribute to the academic community and designers of BPS-models our methods have to be applicable in other studies and/or BPS-model designs. In this chapter we will discuss the generalizability of the methods we have used.

Generalizability of the entity arrival rate

For the entity arrival rate we have a very statistical distribution that is used rarely, the negative binomial distribution. This distribution is not a very popular distribution, compared to, for example, the Poisson distribution. The negative binomial distribution has some disadvantages compared to the Poisson distribution: 1. The negative binomial distribution is not always part of software packages, like Plant Simulation; 2. The Poisson distribution arrival rate is directly related to the exponential distributed inter-arrival time.

The Poisson distribution is a “child” distribution, since they are the same when the p-value approaches 1. For other studies, if this seems the case, it is recommended to switch to the Poisson distribution.

Generalizability of the processing time estimation
The approach used is largely "borrowed" from [43], which is set up quite general. As the statistical distribution we have used lognormal. This distribution is less often used than the normal distribution, which the most common used distribution, when it comes to processing time. In further studies the author needs to check whether of these distribution, or maybe another distribution, fits their data the best. This can be done by plotting the data, or checking the fitting using the chi-square test.

We also have checked on dependency of other variables on the processing time, such as time, the number iterations and the number of items in the queue, etc. These variables were found in literature, from where it can be assumed that they are general for most of the business processes.

Furthermore, FORCE makes use of automated activities within the process. For these activities the processing time varies between zero seconds and, approximately, one minute. With a median of <1sec. We have generalized these activities, setting a processing time of 1 sec for all of these activities. This approach can only be used for processes that use some kind of automated activities with a similar kind of duration. Such activities cannot involve human interaction or handling of physical material.

Generalizability of the start of entities
There are two types of simulation, terminating and non-terminating. Terminating simulations end with zero entities present at the start and end of a certain time period. For this type of simulation no start of entities. However, most of the simulations are non-terminating. For this type of simulation, the approach used can be useful. The approach used in [12, 28] was designed for Petri Nets, but we have shown it can also be applied on other type of BPS-modelling, such as the notation used in Plant Simulation.

Generalizability of the prediction number of employees
As mentioned earlier, a number of activities within the FORCE process are fully automated. We have assumed that for such activities the only limitation is computer processing availability and have assumed that this availability is unlimited.

For the other activities, instead of mining for schedules of employees we have looked at the total number of available employees and applied a normal distribution for this number. Since the number of employees is always an integer, using this approach is not very applicable on activities/cluster of activities that involve a limited amount of employees, because rounding errors will affect the precision of the model. For example: rounding 5.4 to 5 gives a 8% error, but rounding 500.4 to 500 only gives a 0.08% error. This should be kept in mind while applying this method.

Furthermore we have assumed that an employee is working full-time on the same (cluster of) activities. It allows for some ambiguity, but this method cannot be applied on processes where there are no distinctive clusters of activities.

Generalizability of the queue discipline investigation
The approach used was designed in such a ways that it can be used for every type of process. In this method we only co-operated the most common service policies, such as

FCFS. Other methods where one type of entity is selected first over another can easily be added to the approach by sorting the entities in the queue on the attribute investigated.

Generalizability of the approach for the sequences
As mentioned earlier this aspect is the most heavily researched within Process Mining. The methods and algorithms used for this are very general and can be applied to event logs with the right quality.

The preferred method/algorithm should depend on characteristics of the investigated process. We have shown a decision process to obtain the right algorithm. For other researches a similar process can be used to obtain the right method/algorithm.

Generalizability of the BPS-model
The BPS-model was designed in such a way that the only input needed can be read from tables. This makes it easier to adapt the current process or load another, totally different, process.

Generalizability of the validation of the BPS-model
We have validated various components of the BPS-model. For every other BPS-model where these components play an important role, the method we provided can be applied.

Conclusion
Every process is unique and the process we have investigated is no different. This has led to some assumptions and approaches that cannot be generalized to other processes. However, each of the approaches used, only needs limited adaption in order to make it applicable on other processes, with only one exception: the approach used for the mining of the number of employees per cluster works best when the number of employees gets larger, due to errors in round from the distribution.

9.2.2 Data quality
Regarding data quality there are two major concerns: process mining & STP and the lack of logging of starting times.

Process Mining & STP
The timestamp of the used event log is only precise up to seconds. For most of the times this does not result in any problems, because all the processes may take several seconds due to manual intervention. However, for Straight-Through Processing, some of the status changes are done automatically. This combination of the time precision and these automatic changes results in that some events have the same timestamp. Because these events are not added to the database in a specific order, so are not always sorted in the right order, it may lead to sequence noise[62]. We have mitigated the chance for errors by explicitly adding the order of the activities when they are in the same time-interval. This approach works, but we cannot guarantee that this has led to a 100% correct interpretation of the log.

The lack of logging of starting times
As can been seen in the example of the event log, only the transition from one state to another is logged. Thus, the time between to state transitions of an entity not only involves processing of the entity but also involves the time in the queue of an activity. We

have found a method to make an educated guess about the starting time, but we know for sure that this starting time is not always correct. This has its effect on the quality of the results, which will be elaborated on later on.

9.3 Recommendations

9.3.1 Academic recommendations for further research

In our approach we have clustered activities and counted the number of employees. This is different from other papers. This method should be validated and checked whether it is better than existing methods.

We also have developed a method to find the queue policy. This method should be validated to check whether it shows real-life behaviour.

Furthermore, our approach should be tested on other cases in order to validate our approach. It is particularly recommended to test it on a case where we definitely know of that external factors do not affect the process.

9.3.2 Recommendations for the improvement of our BPS-model

Our biggest concern is the processing speed. It is expected that we have not estimated the processing time well, possibly due that external factors might delay processing the entities. It should be researched if this is the case, if we can distil this delay from the data and how we can incorporate this into our BPS-model.

Furthermore, extra characteristics can be added to the model. One can think, for example, on various information aspects to enrich the information about the entities, such as whether it is a NHG mortgage, if the mortgage applicant has its own company, etc.

9.3.3 Recommendations for Topicus

The biggest recommendation for Topicus is the improvement of the event log. We have used and approximation for the starting times of the various activities. Because it is an approximation, it can be expected that there is some error in this method. Adding starting times to the event log will improve the processing time estimation. Also changing the precision of the timestamps from seconds to milliseconds will improve the event log and makes Process Mining on the data easier.

Despite the fact that the BPS-model was invalid, we still have generated some valuable insight in the mortgage process. This insight can be translated into management information for the team, in the form of extending the operational dashboard or providing consultancy to help them optimize the process.

10 Literature list

Literature List

1. Weske, M., W.M. van der Aalst, and H. Verbeek,(2004), *Advances in business process management.* Data & Knowledge Engineering, 2004. **50**(1): p. 1-8.
2. Khanna, A.,(2010), *Straight through processing for financial services: the complete guide*. 2010: Academic Press.
3. Schabell, E.D. and S. Hoppenbrouwers,(2009), *Empowering Full Scale Straight Through Processing with BPM*, in *Advances in Enterprise Engineering II*. 2009, Springer. p. 18-33.
4. Webopedia,(2016). *API*. 2016; Available from: http://www.webopedia.com/TERM/A/API.html.
5. van der Aalst, W.M.,(2010), *Business process simulation revisited*, in *Enterprise and Organizational Modeling and Simulation*. 2010, Springer. p. 1-14.
6. Kelton, W.D. and A.M. Law,(2000), *Simulation modeling and analysis*. 2000: McGraw Hill Boston.
7. Rozinat, A., et al.,(2009), *Discovering simulation models.* Information Systems, 2009. **34**(3): p. 305-327.
8. Van Der Aalst, W.,(2011), *Process mining: discovery, conformance and enhancement of business processes*. 2011: Springer Science & Business Media.
9. Rozinat, W.M.P.v.d.A.J.N.A. and N. Russell.,(2008), *Business Process Simulation: How to get it right?* BPM Center Report

2008. **BPM-08-07**.

10. Martin, N., B. Depaire, and A. Caris.(2014) *The use of process mining in a business process simulation context: Overview and challenges*. in *Computational Intelligence and Data Mining (CIDM), 2014 IEEE Symposium on*. 2014. IEEE.
11. Landsberger, H.A.,(1957), *Hawthorne Revisited: A Plea for an Open City*. 1957: Cornell University.
12. Rozinat, A., et al.,(2009), *Workflow simulation for operational decision support.* Data & Knowledge Engineering, 2009. **68**(9): p. 834-850.
13. Van Der Aalst, W., et al.,(2012) *Process mining manifesto*. in *Business process management workshops*. 2012. Springer.
14. Wieringa, R.J.,(2014), *Design science methodology for information systems and software engineering*. 2014: Springer.
15. Bhattacherjee, A.,(2012), *Social science research: principles, methods, and practices.* 2012.

16. Han, J., M. Kamber, and J. Pei,(2011), *Data mining: concepts and techniques: concepts and techniques*. 2011: Elsevier.
17. Read, B.J.,(1999), *Data mining and science? Knowledge discovery in science as opposed to business.* 1999.
18. Huang, D.W., B.T. Sherman, and R.A. Lempicki,(2008), *Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources.* Nature protocols, 2008. **4**(1): p. 44-57.
19. Jansen-Vullers, M. and M. Netjes.(2006) *Business process simulation–a tool survey*. in *Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, Aarhus, Denmark*. 2006.
20. Mans, R., et al.,(2008), *Process mining techniques: an application to stroke care.* Studies in health technology and informatics, 2008. **136**: p. 573.
21. Jans, M., et al.,(2011), *A business process mining application for internal transaction fraud mitigation.* Expert Systems with Applications, 2011. **38**(10): p. 13351-13359.
22. van der Aalst, W.M., et al.,(2007), *Business process mining: An industrial application.* Information Systems, 2007. **32**(5): p. 713-732.
23. Aguilar, M., T. Rautert, and A.J. Pater.(1999) *Business process simulation: a fundamental step supporting process centered management*. in *Proceedings of the 31st conference on Winter simulation: Simulation---a bridge to the future-Volume 2*. 1999. ACM.
24. Community, A.,(2015). *Business Process Simulation*. 2015 [cited 2015 15 December].
25. Schriber, T.J., D.T. Brunner, and J.S. Smith.(2013) *Inside discrete-event simulation software: how it works and why it matters*. in *Simulation Conference (WSC), 2013 Winter*. 2013. IEEE.
26. Greasley, A.,(2003), *Using business-process simulation within a business-process reengineering approach.* Business Process Management Journal, 2003. **9**(4): p. 408-420.
27. Mes, M. and M. Bruens.(2012) *A generalized simulation model of an integrated emergency post*. in *Simulation Conference (WSC), Proceedings of the 2012 Winter*. 2012. IEEE.
28. Rozinat, A., et al.,(2008), *Workflow simulation for operational decision support using design, historic and state information*, in *Business process management*. 2008, Springer. p. 196-211.
29. *<Van der Aalst - BPS Survival Guide.pdf>.*
30. Song, M. and W.M. van der Aalst.(2007) *Supporting process mining by showing events at a glance*. in *Proceedings of the 17th Annual Workshop on Information Technologies and Systems (WITS)*. 2007.

31. Martin, N., B. Depaire, and A. Caris.(2015) *Using process mining to model interarrival times: investigating the sensitivity of the ARPRA framework*. in *2015 Winter Simulation Conference (WSC)*. 2015. IEEE.
32. Nakatumba, J., M. Westergaard, and W.M. van der Aalst.(2012) *Generating event logs with workload-dependent speeds from simulation models*. in *Advanced Information Systems Engineering Workshops*. 2012. Springer.
33. Liu, Y., et al.,(2012), *Workflow simulation for operational decision support using event graph through process mining.* Decision Support Systems, 2012. **52**(3): p. 685-697.
34. Van der Aalst, W.M., et al.,(2010), *Business process simulation*, in *Handbook on Business Process Management 1*. 2010, Springer. p. 313-338.
35. Hinde, J. and C.G. Demétrio,(1998), *Overdispersion: models and estimation.* Computational Statistics & Data Analysis, 1998. **27**(2): p. 151-170.
36. Gardner, W., E.P. Mulvey, and E.C. Shaw,(1995), *Regression analyses of counts and rates: Poisson, overdispersed Poisson, and negative binomial models.* Psychological bulletin, 1995. **118**(3): p. 392.
37. Pearson, K.,(1900), *X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling.* The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 1900. **50**(302): p. 157-175.
38. Scholz, F.,(1985), *Maximum likelihood estimation.* Encyclopedia of Statistical Sciences, 1985.
39. Hilbe, J.M.,(2011), *Negative binomial regression*. 2011: Cambridge University Press.
40. Gosall, N.K. and G.S. Gosall,(2012), *The doctor's guide to critical appraisal*. 2012: PasTest Ltd.
41. Biau, D.J., B.M. Jolles, and R. Porcher,(2010), *P value and the theory of hypothesis testing: an explanation for new researchers.* Clinical Orthopaedics and Related Research®, 2010. **468**(3): p. 885-892.
42. Luchak, G.,(1956), *The Solution of the Single-Channel Queuing Equations Characterized by a Time-Dependent Poisson-Distributed Arrival Rate and a General Class of Holding Times.* Operations Research, 1956. **4**(6): p. 711-732.
43. Wombacher, A., M. Iacob, and M. Haitsma.(2011) *Towards a performance estimate in semi-structured processes*. in *Service-Oriented Computing and Applications (SOCA), 2011 IEEE International Conference on*. 2011. IEEE.

44. Wombacher, A. and M. Iacob.(2012) *Estimating the Processing Time of Process Instances in Semi-structured Processes--A Case Study*. in *Services Computing (SCC), 2012 IEEE Ninth International Conference on*. 2012. IEEE.
45. Wombacher, A.,(2013) *Start time and duration distribution estimation in semi-structured processes*. in *Proceedings of the 28th annual ACM symposium on applied computing*. 2013. ACM.
46. Redman, T.C.,(2008), *Data driven: profiting from your most important business asset*. 2008: Harvard Business Press.
47. Yerkes, R.M. and J.D. Dodson,(1908), *The relation of strength of stimulus to rapidity of habit-formation.* Journal of comparative neurology and psychology, 1908. **18**(5): p. 459-482.
48. Song, M. and W.M. Van der Aalst,(2008), *Towards comprehensive support for organizational mining.* Decision Support Systems, 2008. **46**(1): p. 300-317.
49. Lloyd, S.P.,(1982), *Least squares quantization in PCM.* Information Theory, IEEE Transactions on, 1982. **28**(2): p. 129-137.
50. Fayyad, U.M., et al.,(1996), *Advances in knowledge discovery and data mining.* 1996.
51. Zhang, J., et al.,(2010), *Advanced intelligent computing theories and applications*. 2010: Springer.
52. Hazewinkel, M.,(2001), *Normal distribution.* Encyclopedia of Mathematics, 2001. **13**(6): p. 337-342.
53. Van der Aalst, W.M.,(1998), *The application of Petri nets to workflow management.* Journal of circuits, systems, and computers, 1998. **8**(01): p. 21-66.
54. Baggio, G., J. Wainer, and C. Ellis.(2004) *Applying scheduling techniques to minimize the number of late jobs in workflow systems*. in *Proceedings of the 2004 ACM symposium on Applied computing*. 2004. ACM.
55. Senderovich, A., et al.,(2015), *Discovering queues from event logs with varying levels of information.* Lect Notes Bus Inf (forthcoming), 2015.
56. Cramer, D.,(1997), *Basic statistics for social research*. 1997, London: Routledge.
57. Günther, C.W.,(2009), *Process mining in flexible environments*. 2009, Technische Universiteit Eindhoven.
58. Günther, C.W. and W.M. Van Der Aalst,(2007), *Fuzzy mining–adaptive process simplification based on multi-perspective metrics*, in *Business Process Management*. 2007, Springer. p. 328-343.
59. Gupta, A.P.E.P.,(2014), *Process Mining A Comparative Study.* 2014.
60. Van der Aalst, W.M.,(2005), *Business alignment: using process mining as a tool for Delta analysis and conformance testing.* Requirements Engineering, 2005. **10**(3): p. 198-211.

61. Weske, M.,(2012), *Business process management: concepts, languages, architectures*. 2012: Springer Science & Business Media.
62. Van Der Spoel, S., M. Van Keulen, and C. Amrit,(2013), *Process prediction in noisy data sets: a case study in a dutch hospital*, in *Data-Driven Process Discovery and Analysis*. 2013, Springer. p. 60-83.

11 Appendices

11.1 Appendix A

Table deleted for Privacy reasons

11.2 Appendix B

Image deleted for Privacy reasons

11.3 Appendix C – The score of various algorithms on complex routing constructs

Characteristic	Alpha Miner	Alpha Miner+	Alpha Miner++	Heuristics M.	Genetic Miner	DT Genetic M.	DWS Miner	AGNEs Miner	TS Miner	ILP Miner	Causal Miner	Process Tree M.
Choice:	$S\downarrow$	$P^b\downarrow$ $S\downarrow$	$F^b\uparrow$ $P^b\uparrow$	$P^b\uparrow$	$F^a\downarrow$ $P^a\downarrow$	$F^a\downarrow$ $P^b\downarrow$ $G\uparrow$	$F^a\downarrow$ $F^b\uparrow$		$P^a\uparrow$ $S\downarrow$	$F^a\uparrow$ $P^a\uparrow$ $G\uparrow$		
Parallelism:		$P^b\downarrow$	$P^b\uparrow$	$P^b\uparrow$			$F^b\uparrow$					
Loop:	$F^a\downarrow F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $S\uparrow$	$F^a\downarrow F^b\downarrow$ $P^a\downarrow P^b\downarrow$ $G\downarrow$ $S\uparrow$	$S\uparrow$	$F^a\uparrow$ $P^a\uparrow$	$F^b\uparrow$	$F^a\uparrow$ $P^b\downarrow$ $S\uparrow$	$S\uparrow$	$F^a\downarrow F^b\downarrow$ $P^a\uparrow P^b\downarrow$ $S\downarrow$	$F^a\downarrow$ $P^a\downarrow P^b\downarrow$ $G\downarrow$	$S\uparrow$	
Invisible tasks:												
Duplicate tasks:				$F^a\downarrow$ $P^a\downarrow$		$F^a\uparrow F^b\uparrow$ $P^a\uparrow P^b\uparrow$ $G\downarrow$	$F^a\downarrow$ $P^a\downarrow$	$P^b\downarrow$	$F^a\uparrow F^b\uparrow$ $P^b\downarrow$ $S\downarrow$	$F^a\downarrow$ $G\downarrow$		
Non-free choice:	$P^a\downarrow P^b\downarrow$ $G\downarrow$	$P^b\uparrow$ $F^b\uparrow$		$F^a\downarrow F^b\uparrow$	$F^a\uparrow$ $P^a\uparrow$	$S\uparrow$	$F^a\uparrow F^b\uparrow$ $P^b\downarrow$	$F^b\downarrow$ $P^b\downarrow$	$F^b\downarrow$ $P^a\downarrow P^b\downarrow$ $S\uparrow$	$F^a\downarrow$ $P^a\downarrow$ $G\downarrow$	$F^a\downarrow F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	
Nested loop:	$F^a\downarrow F^b\downarrow$ $P^a\uparrow$ $S\uparrow$	$F^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^a\downarrow$			$F^a\uparrow F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $S\uparrow$	$P^b\downarrow$ $S\downarrow$	$F^a\downarrow$ $P^a\downarrow$ $G\downarrow$	$S\uparrow$	
Number of traces:				$P^b\downarrow$	$F^a\uparrow$ $P^a\uparrow$	$F^a\uparrow$ $G\downarrow$ $S\downarrow$		$F^b\uparrow$				$S\uparrow$
Number of distinct traces:			$P^a\uparrow$		$F^a\uparrow$		$F^a\uparrow F^b\downarrow$ $P^b\downarrow$	$P^b\downarrow$ $S\uparrow$	$P^a\uparrow$ $F^a\uparrow$	$G\uparrow$ $S\uparrow$	$F^a\uparrow$	
Number of events:				$P^b\uparrow$	$F^a\downarrow$ $P^a\downarrow$	$F^a\downarrow F^b\uparrow$ $G\uparrow$ $S\uparrow$		$F^b\downarrow$ $P^a\uparrow$				$S\uparrow$
Minimum trace length:	$F^a\downarrow$ $P^a\uparrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$		$F^a\uparrow$ $P^a\uparrow$	$F^a\uparrow F^b\downarrow$ $G\uparrow$	$F^a\uparrow F^b\downarrow$ $P^a\downarrow P^b\downarrow$ $S\uparrow$	$F^b\downarrow$	$F^a\downarrow F^b\downarrow$ $P^a\downarrow P^b\downarrow$ $S\uparrow$	$F^a\downarrow$ $P^a\downarrow$ $G\downarrow$ $S\uparrow$	$F^a\uparrow$	
Average trace length:	$F^a\uparrow F^b\uparrow$ $P^a\downarrow$ $S\downarrow$	$F^b\uparrow$ $P^b\downarrow$ $S\downarrow$	$F^b\uparrow$ $P^b\uparrow$ $S\downarrow$	$F^a\uparrow$	$F^a\downarrow$ $P^a\downarrow$		$F^a\downarrow F^b\uparrow$ $P^b\uparrow$ $S\downarrow$	$F^b\uparrow$		$F^a\uparrow$ $P^a\uparrow$ $G\uparrow$ $S\downarrow$	$F^a\downarrow$	
Maximum trace length:	$F^b\uparrow$ $S\downarrow$	$F^b\uparrow$ $S\downarrow$	$F^b\uparrow$ $S\downarrow$	$F^b\downarrow$		$F^b\downarrow$		$S\downarrow$		$S\uparrow$	$S\downarrow$	
Noise:	$S\uparrow$	$F^b\downarrow$ $P^b\uparrow$	$S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^a\uparrow$		$P^a\downarrow$ $S\uparrow$			$P^b\downarrow$ $S\uparrow$		
Number of activities:	$F^a\downarrow F^b\downarrow$ $P^a\uparrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^a\downarrow$ $P^b\downarrow$ $S\uparrow$	$P^a\uparrow$	$F^a\downarrow F^b\downarrow$ $P^a\downarrow P^b\downarrow$ $G\downarrow$	$F^a\uparrow F^b\downarrow$ $P^b\downarrow$ $S\uparrow$	$F^b\downarrow$ $S\uparrow$	$P^a\downarrow$ $S\uparrow$	$F^a\downarrow$ $P^a\downarrow$ $G\downarrow$ $S\uparrow$	$F^a\uparrow$ $S\uparrow$	

F^a = Alignment Based Fitness
F^b = Behavioral Recall
P^a = One Align Precision
P^b = Behavioral Precision
S = Simplicity
G^a = Alignment Based Probabilistic Generalization

11.4 Appendix D – Characteristics of Genetic, Heuristic and Fuzzy Mining

Miner algo / characteristic	**Heuristic Miner**	**Fuzzy Miner**	**Genetic miner**
Description	Provides a view of scientific workflows by considering long distance dependency	Provides a zoomable view of scientific workflow by controlling significance cutoff to show task at different importance level	Provides a view of frequency for both tasks and succession between both tasks, and discovers all common control-flow structures.
Strategy	Work based on local strategy technique to build a model	Work based on both local and global strategy technique to build a model	Work based on global strategy technique to build a model
Output	Heuristic Net	Fuzzy model	Petri net graph
When to use it	When you have real-life data with not too many different events	When you have complex and unstructured log data, or when you want to simplify the model in an interactive manner	When you need to generate a random population of process models and to find a satisfactory solution
Challenging Problems	Can mine logs which are less sensitive to noise, local and nonfree choice constructs.	Can mine logs with noise, but cannot be converted to petri net.	Can mine logs with noise, handling of duplicate task names, local and nonlocal nonfree choice constructs and invisible task.
Behavior	Heuristic mining algorithms take frequencies into account	Fuzzy miner uses dependency graph representation	Genetic mining algorithms mimic natural evolution

11.5 Appendix E: Dendrogram

Image deleted for Privacy reasons