MASTER

Validation of Flexsim HC simulation models using an integrated view of process mining results

Wijgergangs, G.

Award date:
2014

Eindhoven, February 2014

Validation of Flexsim HC simulation models using an integrated view of process mining results

by

Giel Wijgergangs

BSc Industrial Engineering and Management Sciences (2010)

Student identity number 0629198

in partial fulfilment of the requirements for the degree of

Master of Science

in Operations Management and Logistics

Supervisors:
dr. P.M.E. Van Gorp, TU/e, IS
dr.ir. I.T.P. Vanderfeesten, TU/e, IS
dr. ir. R.S. Mans, TU/e, BIS
ir. R.J.B. Vanwersch, TU/e, IS
dr. F.J.H.M. van den Biggelaar, MUMC+
Prof. dr. R.M.M.A. Nuijts, MUMC+

TUE. School of Industrial Engineering.

Series Master Theses Operations Management and Logistics

Subject headings: process simulation, process mining, health-care

Preface

This master thesis is the result of the project I conducted in partial fulfilment of the degree of Master of Science in Operations Management and Logistics at Eindhoven University of Technology. The last five months have been a very demanding period while I was working full time on my master thesis project and managing my start-up in the evening and weekends. I would like to use this chapter as an opportunity to thank all people who contributed to the project or supported me along the way.

First I would like to thank my TU/e supervisors. Thanks to Pieter Van Gorp for his valuable input and feedback throughout the whole project. His detailed remarks and sharp opinions were well appreciated. Secondly I would like to thank Irene Vanderfeesten for her feedback on the thesis and input for the experiment. Finally I would like to thank Ronny Mans for his input and ideas about process mining.

This project wouldn't be possible without the help and case study provided by the MUMC+. Thanks to Rudy Nuijts for the opportunity to work on a great project at the MUMC+ cataract centre. Thanks to Frank van den Biggelaar who provided me with valuable feedback, all the relevant information and input for the case study. Thanks to Rob Vanwersch who provided me with feedback on the thesis, tool and process simulation case.

The last five months taught me how important focus is to get things done. Without the support of my partners at Cycle Software it wouldn't be possible to keep business running while pursuing my master degree, thank you guys!

Finally I would like to thank all my friends and family for their support, laughter and all the great memories of my student life. Specifically I want to thank my parents, for all the support and being there for me. Last person to thank is my girlfriend Astrid. Your love, support and ability to put things into perspective was all I needed to keep going.

Giel Wijgergangs
February 2014

Executive summary

Healthcare is experiencing increasing pressure to perform effectively and efficient due to an aging population, increasing medical possibilities, more demanding patients, more aware insurers, commercial competition and new health treats (Gorp, 2012; Centraal Bureau voor de Statistiek (CBS), 2013). The healthcare budgets are shrinking which demands efficiency and effective care (Ministerie van Volksgezondheid Welzijn en Sport, 2013).

Flexsim Healthcare (Flexsim HC) is a process simulation tool with very realistic visual 3-dimensional animations. 3D animated simulation models contribute to the credibility of a simulation model (Dat, 2012; Law A. , 2007). Simulation models in Flexsim HC are defined using a graphical user interface where parameters can be set in a long list of activities. This causes the desired system to be programmed inside the graphical user interfaces behind tabs and path editors. Because the desired system is implicitly embedded in the software and no process views are available the detailed model gets lost and validation of the simulation model becomes a time consuming task. The implicit process models and absence of formal process views in Flexsim HC results in poor communication between developers and process owners because the model cannot depict the real process steps in a control-flow diagram and the related resources necessary to complete the activities. This causes model validation to be troublesome and time consuming.

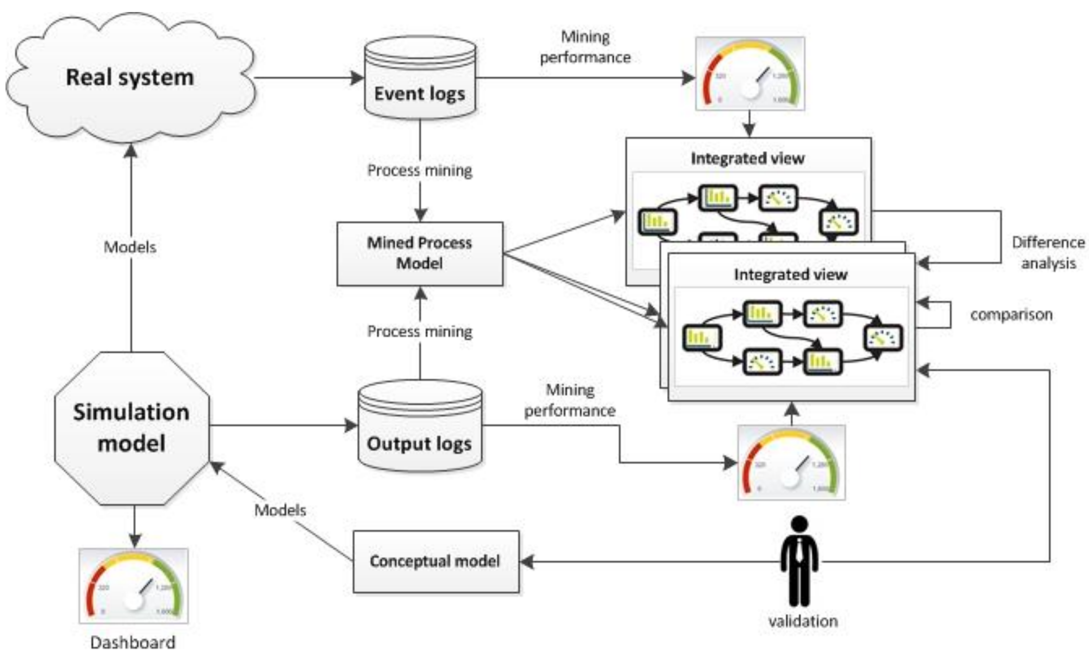


Figure 1: Integrated view in context

To provide a tool to support control-flow validation in addition to visual oriented simulation models process mining was used to discover the simulation model's logic. Mining multiple process mining perspectives into one integrated view creates a model which shows the control-flow of the simulation model and in addition provides performance and resource related information. In Figure 1 it is shown how this view is obtained. The integrated view can be used by the process owners to validate the simulation model with respect to the desired conceptual model and address performance accuracy in one model (Figure 1).

The integrated view was implemented into a tool (IPV Tool). The IPV tool imports a mined model from the Process mining Framework (ProM6). Using a simulation log from Flexsim HC the heuristics miner for mining BPMN models in ProM6 was used to discover the control-flow perspective (process model). The BPMN model was extended with performance data (based on calculations over the output data of Flexsim HC) and related resource information. Figure 2 shows a small part of the integrated view generated by the IPV tool.

The newly created IPV tool (Figure 2) was used in a case study at the MUMC+ in the validation and comparison of simulation models with the process owners. The control-flow was used to determine whether the simulation model correctly reflected the desired system (conceptual model). The performance and resource overlays were used to determine whether the simulation model accurately reflects the performance of the real system. The IPV tool was also used to validate model changes of what-if scenarios compared to the as-is simulation model.

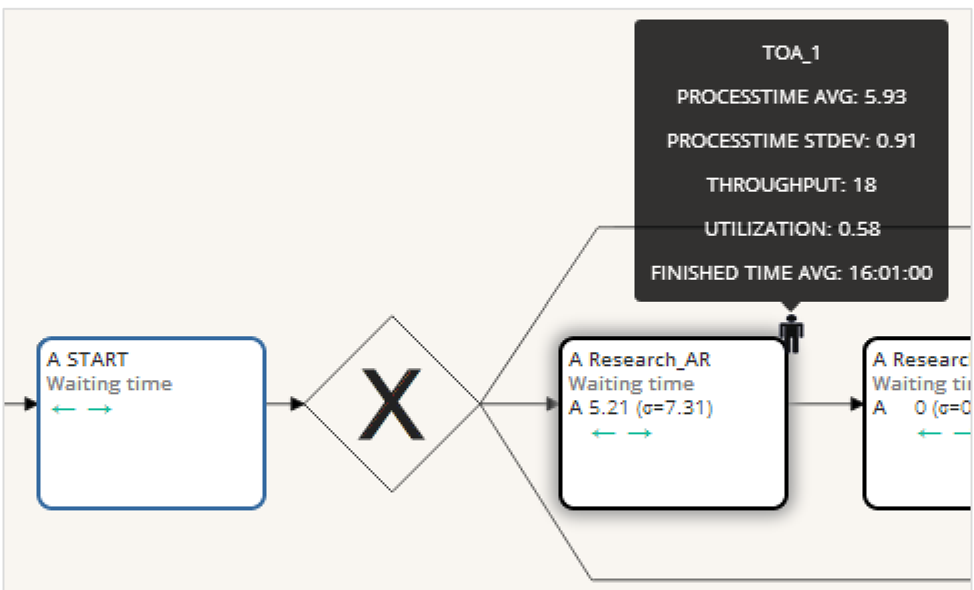


Figure 2: IPV tool view

To validate and justify the solution direction of using an integrated view to validate simulation models a small experiment was conducted. In the experiment two models, a BPMN and a Flexsim HC model were used. In total 8 participants participated in the experiment and answered questions related to the control-flow of a BPMN and Flexsim HC model. Several hypotheses were tested. The results of the experiment supported the hypothesis that answering questions related to the control-flow of BPMN would require less time than Flexsim HC ($p<.05$).

The results also supported the hypothesis that participants prefer having BPMN models over Flexsim HC tracks when verifying a model on correctness ($p<.05$). The hypothesis that BPMN would produce fewer errors in the control-flow related questions was not supported by the data. Because the experiment was very small scale more research is necessary to generalize the claim that having a process view in simulation software package is essential for efficient model validation.

This research contributed a practical solution for Flexsim HC with the IPV tool which can assist in simulation model validation with the process owners (Figure 1). This research also showed a new approach to implement multiple process mining perspectives into one view using a more dynamic approach. In addition the experiment showed that the control-flow views are an important feature in simulation modelling for model validation. In future work the tool could be applied in more generic data models to support other simulation output logs or real life event logs.

Contents

List of tables

List of figures

1. Introduction

Healthcare is experiencing increasing pressure to perform effectively and efficient due to an aging population, increasing medical possibilities, more demanding patients, more aware insurers, commercial competition and new health threats (Gorp, 2012; Centraal Bureau voor de Statistiek (CBS), 2013). The healthcare budgets are shrinking which demands efficiency and effective care (Ministerie van Volksgezondheid Welzijn en Sport, 2013).

The complex challenges healthcare is experiencing require new state of the art methods to evaluate business process performance and techniques to predict future behaviour of processes in order to become more efficient and effective. Discrete event simulation is a method to evaluate the performance and behaviour of future processes. Usage of discrete-event simulation (DES) can help healthcare organisations to effectively allocate resources, improve patient flows while minimizing costs and increase patient satisfaction (Jun, Jacobson, & Swisher, 1999). Although some sources argue that DES is becoming more popular and increasingly more accepted in health-care delivery (Jacobson, Hall, & Swisher, 2006), most studies suggest that healthcare is lagging behind in the application of DES compared to other industries (Jahangirian, et al., 2012; Lowery, Hakes, Keller, Lilegdon, Mabrouk, & McGuire, 1994). Visual oriented 3-dimensional simulation models with animation features can contribute to the acceptance and credibility of simulation projects among non-specialists and thereby increase support of DES in health-care (Dat, 2012; Law A. , 2007).

Flexsim Healthcare (Flexsim HC) is a simulation software package specifically for the healthcare industry with a visual approach using realistic 3-dimensional layouts (Figure 26, Appendix C). Flexsim HC provides dialog boxes in graphical user interfaces where parameters can be set in simplistic path editors to model the process. Unlike popular tools such as Arena and CPN-tools where the simulation logic can be viewed in a process view (Jansen-Vullers & Netjes, 2006), Flexsim HC embeds the control-flow logic of models in simplistic path editors. As a consequence, the inner working of the desired system and process model gets lost in the parameters and user interface of the simulation software (Ryan & Heavey, 2006).

The implicit process models and absence of formal process views in Flexsim HC results in poor communication between developers and process owners because the model cannot depict the real process steps in a control-flow diagram and the related resources necessary to complete the activities. This causes model validation to be troublesome and time consuming. When the simulation model is available in a control-flow view (graphical representation of the model's logic) this requires less time and effort.

Deriving a control-flow view from a simulation model to depict the model's logic can be realized in multiple ways. One way is to manually recreate the control-flow logic based on the information in the graphical user interface by the simulation engineer. A disadvantage of this solution is that it is time consuming and might be more prone to errors than automatic methods. Another solution is to create a model-to-model transformation where the simulation model is automatically transformed into a control-flow model. The disadvantage here is that this would only work for the selected software package and is not broader applicable to other simulation software. With the use of process mining techniques the control-flow view can be discovered based on the simulation output log. With this approach the solution can also be applied in other applications.

The usage of process mining to discover the control-flow of simulation models is shown in Figure 3. The output logs of the simulation models are used to discover the process model and mine performance data. Including the performance data in the mined model creates an integrated view (Figure 3). By including performance data in the integrated view, one model shows the control-flow and in addition implements dashboard data which is normally also available in the software package. This improves the validation process with the process owners because one view can be used to ensure the model correctly reflects the control-flow logic of the desired system (conceptual model) and can also be used to validate the model performance accuracy.

With a tool that automatically generates an integrated view (Figure 3) the simulation project can deliver process owners both strong visual 3D simulation models, but also a process oriented view with performance data for validation purposes and exploration of simulation output results.

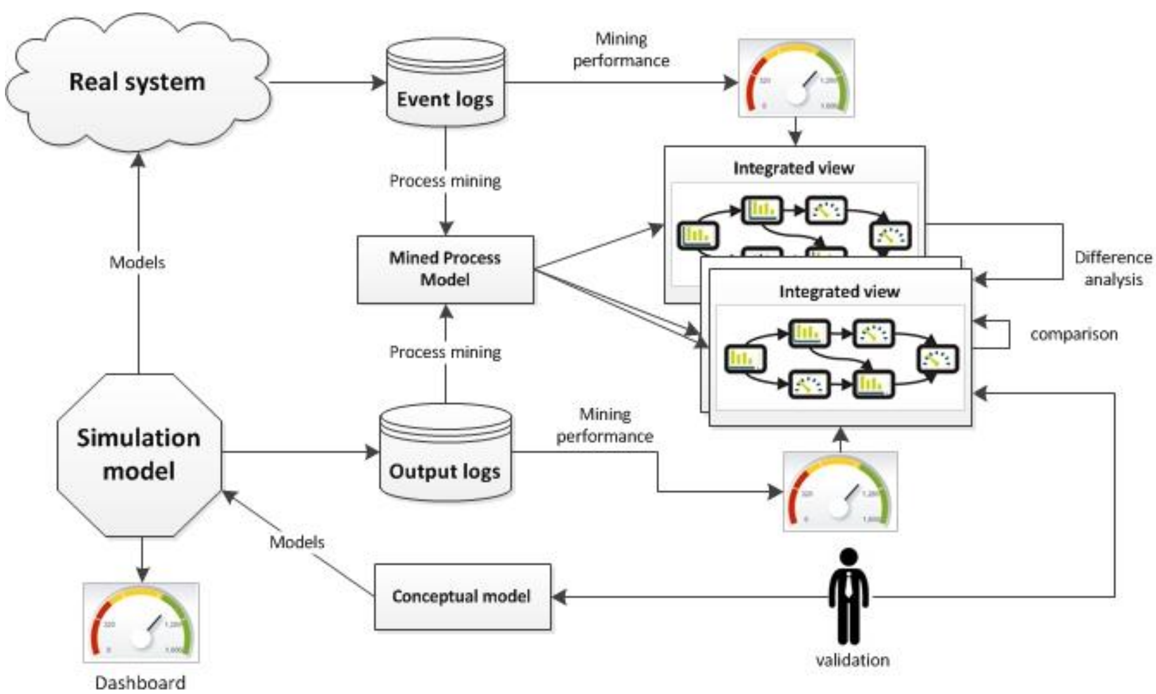


Figure 3: Integrated view in context

Process mining techniques are generally used to generate process models based on data describing what happened in the past. In general four process mining perspectives are considered (Aalst, 2010b). The **control-flow perspective** is used to discover the sequence and frequencies of activity executions. The **time-perspective** is used to generate time-related performance measures such as throughput times. The **organisational perspective** is concerned on the resource involvement, i.e. who performed what task. The **data perspective** focuses on characterization of data properties in cases. The perspectives of process mining are further described in Chapter 2.

In this project process simulation output and process mining techniques are combined in a newly created tool which provides functionality to assess performance of simulation models in a process perspective. By using an approach that generates visual overlays on top of the control-flow perspective a process model can provide more useful insights and performance measures of processes. The IPV (Integrated Process View tool) tool created in the project is used for validation of simulation models in a case study at the Maastricht University Medical Center (MUMC+).

1.1 Motivation

According to Van der Aalst (2010) many organisations attempted to use simulation to predict and analyse performance of business processes at some point. But only few organisations achieved to use it in a structured and effective manner. The reasons given by Van der Aalst (2010) are the lack of training and limitations of existing tools. In the literature review prior to this master thesis project it became clear the existing simulation tools are heavily focused on simulation as a programming task (Ryan & Heavey, 2006). In addition, many tools lack implementation of formal business process modelling languages to support the relation between abstract simulation models and the business process (Jansen-Vullers & Netjes, 2006).

This project addresses the lack of support for validation of control-flow and performance data in Flexsim HC. This is done by delivering an integrated view of performance data, resources involvement and control-flow to support simulation model validation for the end customer (Figure 3).

1.2 Research question and objectives

In the following sections the main problem, research objective and research questions are given.

1.2.1 Main problem

Based on the challenges healthcare is facing and the need for state of the art tools to predict future performance of processes the following main problem is addressed:

Main problem: There is an unsatisfied need for visually oriented simulation models with validation support on control-flow and performance levels

1.2.2 Research objective

This project addresses the following research objective:

Research objective: Provide a method and tool to support validation of simulation models based on simulation output in one view that implements the time-, organisational- and control-flow perspectives in addition to visual oriented simulation models

To meet the research objective this research project provides a method and tool which generate a process view by mining simulation logs to support validation of simulation models. The method is based on the concept of process model extension where multiple process mining perspectives are integrated in one view (Aalst, Process Mining, 2010b). The chosen mining perspectives are useful when validating simulation models because performance measures of interest are often compared to real world data (Sargent, 2005; Robinson, 1997). The organisational perspective here is useful because it links activities with resources. The control-flow perspective is useful because it allows for validation of the process flow (Sargent, 2005; Robinson, 1997). This implements three of the four mining perspectives described by Van der Aalst (Aalst, 2010b).

To address the research problem and meet the objective a case study is performed at the Maastricht University Medical Center (MUMC+). In the case study the processes of the cataract centre are implemented in simulation models to predict performance of various what-if scenarios. In this study the newly created Integrated Process View (IPV) tool is applied in validation of simulation models and the exploration of results presented in the integrated view (Figure 3). In addition a small experiment is conducted to validate and justify the solution direction by identifying whether having access to control-flow diagrams could provide time efficiency in validation of simulation models.

1.2.3 Research questions

Based on the given main problem and research objective in the previous section the following research question is constructed:

Research question: How can simulation models be validated by applying process mining techniques on simulation logs?

The research question can be divided into the following sub-questions:

Q1) What are the problems with model validation in Flexsim HC?

Q2) What is the output of Flexsim HC and what process mining techniques can be applied?

Q3) How can simulation output results be implemented in control-flow diagrams using time- and organisational perspective as overlays?

Q4) How can an integrated process view assist in simulation model validation?

Q5) Are control-flows important for simulation model validation?

1.3 Document structure

The structure of this report is as follows. First an overview of the core concepts in process mining and process simulation are given. In Chapter 3 the research method is outlined. In Chapter 4 the problem is described in detail. In Chapter 5 an analysis is given followed by a solution design in Chapter 6. In Chapter 7 the IPV tool is used in a case study. Chapter 8 provides the results of the experiment and finally in Chapter 9 the conclusions, limitations and further research are given.

2. Preliminaries

In the introduction three solution directions were given to derive a control-flow model from a Flexsim HC simulation model. The solution direction chosen in this project applies process mining on the simulation logs (Figure 3). In the following sections the academic preliminaries for this thesis are covered. In 2.1 the domain of process mining is described and in 2.2 the process simulation domain is covered.

2.1 Process mining

Process mining is a data mining technique for discovery of processes. Van der Aalst (2010b, p. 8) defined process mining as "To discover, monitor, and improve real processes by extracting knowledge from event logs readily available in today's systems". In Figure 4 a model overview of process mining is depicted. This model shows the relation between the real world and the information associated with this real world which is stored in an event log. In this project the information system will be a process simulation tool that stores data in an event log. This data is not derived from the real world, but from an abstraction programmed in a simulation model (Figure 3).

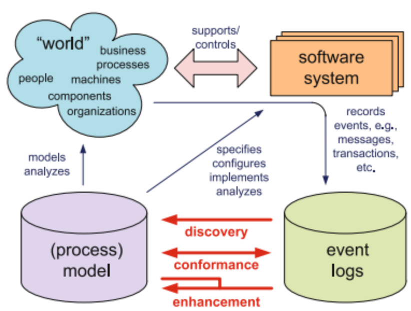


Figure 4: Application of process mining (Aalst, Process Mining, 2010b)

The event log contains records with activity names, timestamps of occurrence and additional information such as resources that executed the activity. Based on this data process mining techniques can be applied such as: process model discovery, conformance checking, performance analysis or use the event log data for enhancement (Aalst, 2010b). Figure 3 shows how process model discovery is combined with performance analysis in this project. The process mining techniques are associated with different perspectives in process mining. In this project three of the four process mining perspectives are applied on the simulation log: the control-flow, time and organisational perspective. In the following sections the perspectives are briefly explained.

2.1.1 Control-flow perspective

The control-flow perspective is focused on the ordering of activities, the goal is to automatically discover a representative process model displaying all possible paths expressed using a notation (e.g. Petri-net, BPMN, EPC). The control-flow perspective is derived by sequences and frequencies of activities in the event log. The control-flow logic of a process model defines start of parallel behaviour, alternative routings (splits) or merging of paths (Figl, Recker, & Mendling, 2013). Multiple algorithms are proposed to discover the control-flow. Ranging from basic algorithms to more complex which can better deal noisy data and complex control-flow structures (Aalst, 2010b).

2.1.2 Time perspective

The time-perspective is concerned with the frequencies and timing of events. Based on timestamps in the event-log the goal the time-perspective might be to find bottlenecks, measure service levels like waiting times, throughput time or resource utilization (Aalst, 2010b).

2.1.3 Organisational perspective

The organisational perspective is focused on the relation between activities and resources. With the organisational perspective the resource involvement in the process is considered. The focus can be on the hand-over of work: who hands over work to whom or simply who executes which task (Aalst, 2010b). The latter is used in this project, when linking the resources to activities it is immediately clear whether the resource executed the right task and it can directly show associated performance data.

2.1.4 Data perspective

The data perspective is focused on data related to a case. Cases can be characterized by values/data related to the specific case which influences the process path or performance (Aalst, 2010b). An example of such a data value is the size of an order.

2.1.5 Combining perspectives

Traditionally in process mining the different perspectives were addressed in an isolated manner. For example the process views for the control-flow, charts for performance data, and social graphs for hand-over of work. Van der Aalst (2011) provides an example of the extension of different perspectives in a process model. By cross-correlation of perspectives the process model can be extended with for example performance data from a time-perspective (Aalst, 2010b). In Figure 5 a Petri Net (describing the control-flow) with the extension of the organisational-, time- and data-perspective as a concept is depicted.

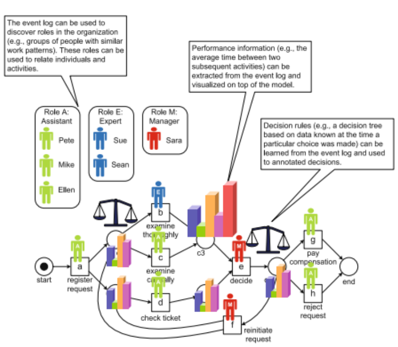


Figure 5: Extension of the control-flow perspective by implementing time-, organisational- and data-perspectives (Aalst, Process Mining, 2010b)

Projecting additional information as an extension in process views was implemented by Rozinat (2010). Rozinat (2010) used extension of process views to project additional information from a data-, organisational- and performance perspective. Several visualization techniques were used in the augmented models: relative line width to frequencies of travelled paths, transparency of edges relative to the time spent on average in that part of the process. In addition to these visualizations a lot of textual overlays were used: decision points (data-perspective) and textual overlays to project the involved resources (organisational-perspective). Figure 6 shows the general model proposed by Rozinat (2010). The extended process view integrates the time-, data-, resource- and control-flow perspective in a Petri Net. For the time perspective the execution time and waiting times are given. Above the activity nodes the resources are displayed. The data perspective is related to the activity nodes in red and probabilities of selection of paths are given on the edges. The proposed model of Rozinat was not implemented in a plugin to automatically generate an extended process model, the model shows an example how multiple perspectives can be integrated.

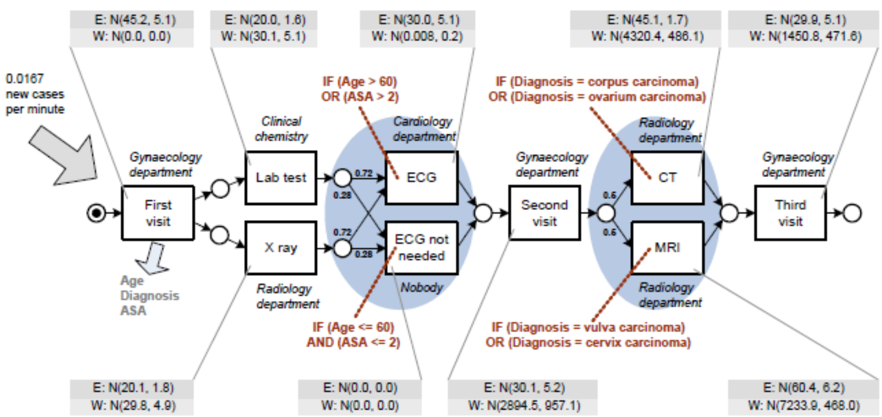


Figure 6: Example process model with integrated data, organisational and performance view (Rozinat, 2010)

An implementation process model extension is described and Bayrakater (2011). Bayraktar (2011) created a BPMN plugin for the process mining framework (ProM6) with performance overlays. These performance overlays provide insufficient performance measures to fully support validation of simulation models and can therefore not be used for that intent. In section 6.4 a comparison between the BPMN plugin and the overlays developed in this project is given to outline the similarities and differences between the BPMN plugin and the approach of this thesis. The integrated view of Bayraktar (2011) was developed to mine event logs from real world data. The relation of mining from real world data to the IPV tool is also depicted in Figure 3.

2.2 Business process simulation

Business process simulation (BPS) is used to predict future performance of processes and is one of the most widely used operations research techniques (Ryan & Heavey, 2006). The simulation model can be the as-is situation or the planned future implementation (to-be) (Kellner, Madachy, & Raffo, 1999). A motivation for using BPS given by Kellner et al. (1999, p. 92): “simulation is an inexpensive way to obtain knowledge about the real system of interest”. Unlike optimization methods, simulation doesn’t provide an optimal solution (Jun, Jacobson, & Swisher, 1999). The output of simulation delivers estimated performance of a model. Since all models are abstract, not all of the aspects of a business process are modelled. The idea is that of all the aspects of the business process which are believed to be relevant are included in the model (Aalst, 2010a; Kellner, Madachy, & Raffo, 1999). By integrating uncertainty of aspects present in the real world the abstract simulation model is used to learn about what will happen in the real world (Paul, Hlupic, & Giaglis, 1998).

The accuracy of a model is an important factor for accessing the validity of a simulation model, i.e. is the performance output consistent with the real system? (Robinson, 1997; Sargent, 2005). Variation and uncertainty can be modelled in conditions such as the arrival rate, service times, availability of resources and probabilities in choices which determine the performance of the simulation model. The overall performance is determined by these conditions and combined effects of other parts in the model. For example the waiting time in the system can be caused by the variation in arrival rate and processing times. When the waiting-times of activities in the simulation model do not accurately reflects the real system this might indicate that the simulation model is incorrectly build or does not support subtle model behaviour which is present in the real system (Aalst, 2010a; Sargent, 2005).

The accuracy can be validated using objective methods such as statistical tests and confidence intervals. Alternatively in a subjective approach the model is validated by comparing the simulation model behaviour to another model or the real system (Sargent, 2005). A simulation model can be validated using black-box and white-box validation (Robinson, 1997). In black-box validation the overall systems performance is assessed. Black-box validation may not lead to total confidence in the model but should help increase confidence. In white-box validation a part of the simulation model is assessed whether it accurately represents the real system (Robinson, 1997). This can, for example, be done by accessing the performance of a single activity. White-box validation can also focus on the correctness of the simulation model compared to the conceptual model (Figure 3) or when this is not available a process description of the desired system. This way it is ensured that the simulation model correctly reflects the desired system.

Since it is not possible to prove that a simulation model is completely correct and accurate, validation methods are important to increase confidence such that models are accepted as valid (Robinson, 1997; Sargent, 2005). Sargent (2005) argues that the relation between the cost, value and model confidence is as depicted in Figure 7. With better tools to support model validation the costs can be decreased while maintaining a high value model. In addition validation methods can increase confidence in the simulation models such that models are accepted and considered credible (Robinson, 1997; Sargent, 2005).

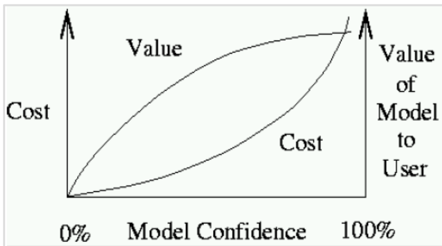


Figure 7: Model confidence vs. the costs and value of a model (Sargent, 2005)

3. Research method

To achieve the research objective the regulative cycle of Van Strien (1997) is used. The regulative cycle has five basic steps: (1) problem definition, (2) analysis and diagnosis, (3) solution design, (4) intervention and (5) evaluation (Van Strien, 1997). The regulative cycle is depicted in Figure 8 and forms the basic structure for this thesis. In the next sections it is described how the regulative cycle is used to structure this thesis.

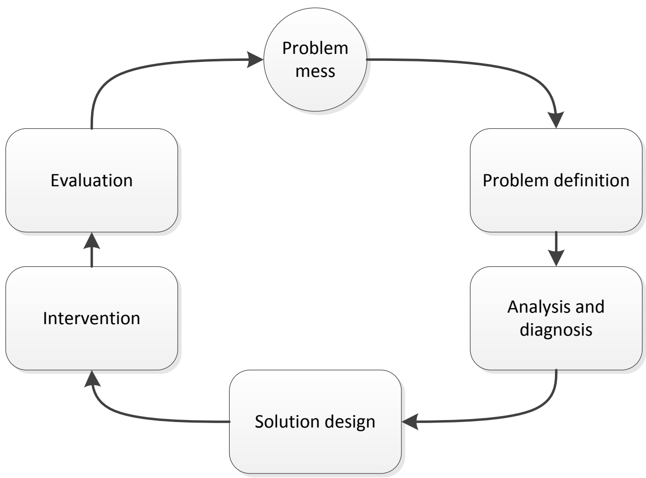


Figure 8: Regulative Cycle, adapted from Van Strien (1997)

1) Problem description

The introduction in Chapter 1 already briefly introduced the problem. In Chapter 4 the problem will be described in more detail using examples and by outlining the shortcomings in Flexsim HC. This provides an answer to research sub-question 1: *What are the problems with model validation in Flexsim HC?*

2) Analysis and diagnosis

In phase two of the regulative cycle an analysis and diagnosis is given. In Chapter 5 the objectives for simulation and key performance indicators of interest for MUMC+ are outlined. Consequently the output of Flexsim HC is analysed and reverse engineered manually into a UML class model. With the detailed understanding of the output of Flexsim HC different process mining algorithms are explored to discover the control-flow perspective and an algorithm is selected for the solution design. The analysis and diagnosis provide an answer to research sub-question 2: *What is the output of simulations and what process mining techniques can be applied?*

3) Solution design

In Chapter 6 the solution design for the described problem is given. The overlays for the integrated view (Figure 3) are developed based on the data available in the output of Flexsim HC and the KPIs of interest for MUMC+. The control-flow perspective is discovered using the ProM6 framework. The control-flow perspective is extended by integrating the KPIs and information from various mining perspectives into the model (time- and organisational perspective). This creates the integrated view as depicted in Figure 3. Figure 9 shows the relation between the simulation software (Flexsim HC), the process mining framework (ProM6) and simulation goals which form the input for the overlays. Chapter 6 provides an answer to research question 3: *How can simulation output results be implemented in control-flow diagrams using time-and organisational perspective as overlays?*

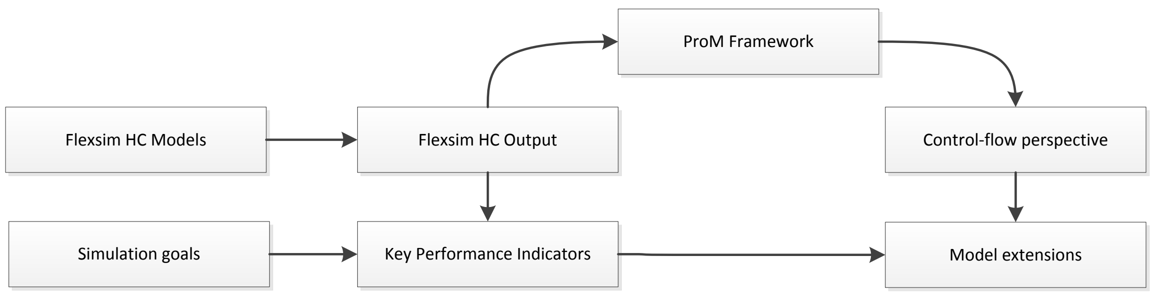


Figure 9: Engineering flow for the process view extensions

4) Intervention

In the intervention phase of the regulative cycle the solution proposed is used in practice. In this project the tool with the integrated process view (Figure 3) is used in a simulation case study at the MUMC+. The IPV tool is used to address the validity of the simulation models but also used to explore and compare simulation models. In Chapter 7 it is shortly describes how the tool was used in this project and what the outcomes of this study are. Chapter 7 thereby provides an answer to research question 4: *How can an integrated process view assist in simulation model validation?*

5) Evaluation

The final phase of the regulative cycle is the evaluation phase (Figure 8). In this project the evaluation phase is separated into two parts. First the created solution is evaluated in the case study. This is done by a description how the integrated process view was used in the case study which is covered under research question 4.

The second part of the evaluation consists of a small experiment. The goal of the experiment is to identify the importance of control-flow logic while assessing the simulation model correctness from a developer's point of view to validate the solution direction of this project. The outcomes of the experiment provide validation and justification of the proposed solution direction. This is done by conducting an experiment where Flexsim HC models and BPMN models are evaluated by the participants. The participants answer questions related to the control-flow elements in the models. The experiment focuses on the understandability (speed and errors) of control-flow views of Flexsim HC models vs. BPMN models. Since the number of Flexsim HC users in the region is very limited the experiment will be carried out using an online survey.

In addition to control-flow related questions the Method Evaluation Model (Moody, 2003) is used to determine the perceived ease of use, usefulness and intention-to-use of having access to control-flow views in simulation software for model verification.

The Method Evaluation Model (ME model) assumes that all methods are designed to improve performance, which can be achieved in two ways (Moody, 2003):

- Efficiency improvement: by reducing effort required to get output
- Effectiveness: improved quality of output

The ME model argues that the following dimensions of success of a IS design method need to be evaluated:

- Actual efficacy: Does the method improve the performance of the task?
- Adoption in practice: Is the method actually used in practice, this regardless of the performance gain.

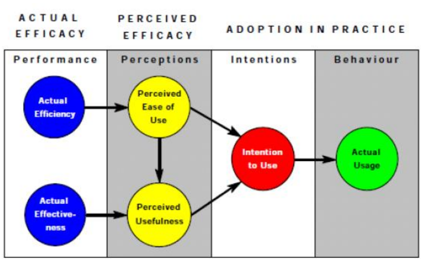


Figure 10: Moody's Method Evaluation Model (2003)

In the experiment the actual efficacy is assessed by measuring the accuracy and speed of questions related to the understandability of Flexsim HC and BPMN models. The adoption in practice is assessed by including the questions related to the constructs perceived ease of use (PEOU), intention to use (ITU) and perceived usefulness (PU). The perceived ease of use is defined as the degree to which a person believes that using a particular method would be free of effort (Moody, 2003). The intention to use is the extent to which a person intends to use a particular method and the perceived usefulness is the degree to which a person believes that a particular method will be effective in achieving its intended objectives (Moody, 2003). Chapter 8 describes the experiment which was conducted to identify the importance of control-flow views for simulation model validation and thereby provides an answer to research question 5: *Are control-flows important for simulation model validation?*

4. Problem description

In this chapter the problem is further analysed. The introduction in Chapter 1 already briefly explained the problem. In this chapter the problem is further outlined using examples and a detailed description of Flexsim HC.

4.1 Process simulation in Flexsim HC

Flexsim HC is an application-oriented simulation software package specifically distributed for the healthcare domain (Dat, 2012; Law A. , 2007). Application oriented simulators were developed towards a certain application and use a graphical user interface. A simulation model is built using dialog boxes and menus where parameters can be set (Law A. , 2007). This causes that process models are implicitly embedded in the software. Given the fact that Flexsim HC cannot provide accurate control-flow process views it is difficult to identify whether the simulation model correctly reflects the desired system. The hierarchical framework of Nikoukaran et al. (1999) for the evaluation of simulation software refers to this as conceptual model generation for testing and efficiency related to software packages. Flexsim HC lacks the conceptual modelling features. A screenshot of a Flexsim 3D model is depicted in Figure 26 of Appendix C. In the following sections the main graphical user interfaces of Flexsim HC are described and the problems associated with the application oriented nature are discussed.

4.1.1 Flow chart view

In Figure 27 of Appendix C the same process is depicted in the Flow Chart view of Flexsim HC. The flowchart depicts all the so called resource destinations, i.e. locations in the process where a resource of some kind is required. In fact the name flowchart is not an accurate name for the view it is representing. The view only represents the flow between different processing and waiting locations. The activities are not depicted in this flowchart. In Flexsim HC the activities trigger a move to another resource destination in this view but are not visible.

4.1.2 Patient Tracks

Processes in Flexsim HC are defined by patient tracks which determine the flow of patients through the system. In the simulation models used in this study each patient track represents a patient type. Each patient track is defined by a series of activities. Activities are identified by an activity number and name. Flexsim HC provides a wide range of activity types. Depending on the activity type a series of parameters can be set. Most important parameters are processing time and the staff resources associated with the activity (Appendix D, Figure 30), as well as a resource destination. For example a room or facility required to execute the activity.

Regularly used activity types in the models are:

- Process: for example performing a measurement or consult
- Patient travel: move patient to another location
- Decision point: XOR split with probabilities for a choice

Difficulties in validation of simulation models in Flexsim HC are due to the simplistic nature of patient tracks and the flow chart views. It is impossible to review the control-flow of a patient track in a single view. For example; to find a XOR split (decision point) one must go through all the activities in a patient track (Figure 29, Appendix D). The routings are implicitly embedded in the model. When there are several more patient tracks it becomes more difficult because comparing patient tracks is also not possible in one graphical user interface. To illustrate this problem an example of a very simple process is given in the BPMN notation and a screenshot of how this process is modelled in a patient track (Appendix D). The process starts with activity Start then either Activity1 or Activity2 is executed finally the activity Finish is executed. It is also possible to define parallel activities in a patient track. An AND split can be defined by specifying multiple predecessors in an activity, this is also illustrated as an example in Figure 33 of Appendix D. This way of defining simulation models causes verifying whether the simulation model correctly reflects the desired system to be time consuming and may also be more prone to errors. In addition, it is often the case that a track is duplicated multiple times with minor differences (for example specific resource assignments per patient track), resulting in a large number of activities and tracks which makes the verification process even more time consuming.

4.1.3 Dashboard

Flexsim HC provides a customizable dashboard view (Appendix C, Figure 28). Here different graphs and key performance indicators can be selected. Because there is no standard dashboard there is some danger for a blinkered view of the results while validating a simulation model. The view does not by default provide measures required for validation. Also the relation between the measures in the dashboard and activities in the simulation model are only known when they are explicitly selected in the dashboard with the danger of missing essential information with regards to specific activities and resources. Currently it is only possible to show the waiting times and service times in the Flexsim HC dashboard in relation to resource locations, not activities specified in the patient track. This information is available in export files but cannot be visualized in the dashboard unless an activity is related in a one-to-one relation with a resource location. This assumption is simply violated when a waiting room is shared for two different activities. This makes it less straightforward to validate the performance accuracy of activities: it is only possible using custom exports or standard output files.

4.1.4 Overview Flexsim HC

In the previous sections the main elements of Flexsim HC were discussed. Flexsim HC lacks possibilities to show control-flow diagrams of the patient tracks. This makes it difficult and time consuming to validate the tracks with process descriptions or a conceptual model, i.e. does the model properly reflect the desired system? Also the dashboard provides insufficient performance measures in relation to a process. The dashboard cannot provide performance measures related to specific activity names in the patient track, only to associated resource locations. Therefore white-box validation, where the performance measure for a single activity of a patient track is compared to the real system performance is not possible within Flexsim HC. It is also not possible to compare two different tracks or models. The only way to do this is by opening the same model in two completely different windows. The IPV tool created in this project is aimed at solving the problems with the lack of process views, insufficient dashboard information and comparison of two models.

4.2 Overview problem

In the previous sections the problems associated with Flexsim HC are described. This provides an answer to research sub-question 1: *What are the problems with model validation in Flexsim HC?*

The main limitations can be summarized as follows:

- Lack of overview global simulation process which makes validation of the model very inefficient
- No possibility to assess specific activity performance
- No support for validation of correct resource allocation to activities in one view (Appendix D, Figure 30)
- No support to compare simulation models to validate model changes

These limitations of Flexsim HC can be overcome by creating a multiple perspective view of both the simulation process (control-flow) and integrating performance and resource data such that validation of models becomes more efficient. In addition by integration of functionality to support comparison of different simulation models it is easier to validate model changes.

5. Analysis & diagnosis

In this chapter the analysis and diagnosis is described. First the objectives and important key performance indicators for MUMC+ are outlined, this forms important input for the solution design. Consequently the output files of Flexsim HC are analysed and reverse engineered into a UML class model. In the last sections of this chapter process mining techniques are applied on event logs generated from the output files of Flexsim HC. The event logs are used to discover the control-flow perspective which can be used to validate the simulation model's logic with respect to the conceptual model as depicted in Figure 3.

5.1 Objectives and key performance indicators

In the following sections the objective of process simulation and the specific key performance indicators (KPIs) of interest are described. The KPIs form an important input for the solution design where the KPIs are integrated into the IPV tool.

5.1.1 Simulation objective

Although the processes at MUMC+ are optimized and standardized process for high volumes of patients, MUMC+ is still figuring out how to setup their protocols to optimize patient throughput, throughput times and resource utilization. The main objective of MUMC+ is to use Discrete Event Simulation to evaluate different what-if scenarios to explore the impacts on key performance indicators of changes to the current system. Specifically the planning heuristics and layout is of interest for MUMC+.

5.1.2 Key performance indicators

In process analysis typically three dimensions of performance are considered: cost, time and quality (Aalst, 2010b). To quantify the cost-dimension several options are available. The process costs can be fixed (Activity Based Costing), dependent on time (Time Based Costing) or based on the utilization of resources (Resource Based Costing) (Aalst, Process Mining, 2010b). In this study the actual costs are not considered but based on time and resource. The utilization of resources and working hours are key performance indicators which can be related to the cost dimension.

For the time dimension different Key Performance Indicators (KPIs) can be defined. Van der Aalst (2010b) defined the following KPIs related to the time-perspective in process mining:

- Flow Time / throughput time: total time that the patient is in the process
- Processing time / Service Time: duration of actual treatment time
- Waiting Time: the time a patient is waiting for resources to become available
- Synchronization time: the time a patient is waiting for a trigger to be able to continue the process, for exampling synchronizing parallel activities

The literature study of Cayirli & Veral (2003) outlines the specific performance measures used to evaluate appointment systems. In addition to the time, and cost-based measures also congestion measures such as average and frequency distribution of patients in queues and systems can be considered. Cayirli & Veral (2003) also distinguished net waiting time and normal waiting time. The net waiting time is calculated by subtracting the greater of {appointment time;

arrival time} from the consultation start time (Cayirli & Veral, 2003). This ensures that the time the patient arrives before the actual consultation time is excluded from the waiting time.

In consultation with process owners of the cataract centre, the main KPIs of interest were selected. The following KPIs are of specific interest when using business process simulation for what-if scenarios at the cataract centre:

- Utilization per resource (cost-dimension)
- Finished time last activity per resource (cost-dimension)
- Throughput time (time-dimension)
- Service time (time-dimension)
- Waiting time (time-dimension)
- Net waiting-time (time-dimension)
- Throughput count (congestion-measure)
- Queue length (congestion-measure)

The quality dimension typically focuses on the product or service that is delivered (Aalst, Process Mining, 2010b). Quality can be measured using different approaches. A common quality dimension in health-care would be the number of successful treatments. In the models of this study it is assumed that all activities deliver equal quality and that the processing times are not dependent on quality. Therefore the quality dimension is not of interest in this study.

5.2 Flexsim HC output data

Flexsim HC provides multiple output options and files. Appendix I documents the output specification options used for the analysis of the data model. The output files are formatted as comma separated values (CSV). In the following sections the different output files will be described and in case of the patient history it is described how the output data can be depicted into a UML class model.

One of the output files given by Flexsim HC is the patient-history log, which is outlined in Appendix E. This file contains a log with on every line an event-trace of a patient. The number of columns can grow very large; for every activity the log contains 25 columns (Appendix E, Table 1, Field 14-39), which results in 500+ columns for the legacy simulation model output. The output stores the waiting time for every activity and every waiting location. By default there is no explicit link between the activities and resources necessary to execute the activity. For example, a doctor is necessary for a consult, the link between doctor and consult cannot be made based on the standard output files (Appendix E). To make this link a piece of custom code is used to store this data in one field using the Javascript Object Notation (JSON). JSON is a syntax for storing and exchanging text information. The code to create the JSON syntax can be found in Appendix J. This notation makes it possible to store a multidimensional data structure in a single text field.

To obtain better insight in the output data, the file structure is manually reverse engineered into a UML class diagram based on analysis of output from the legacy simulation models. Based on the data in Table 14 (Appendix E) five classes are created. The first class is the *Log* class which stores the filename of the output. The class *Case* contains the fields 1 until 13 of the output and stores patient and case related data. The second class created is Activity which stores an activity name. Related to this is de class *ActivityEvent* which contains the information represented in fields 14 until 21 of Table 14 and more generic in Table 17 (Appendix E). *ActivityEvent* is related to one *Activity* instance. The log also stores the waiting-time per activity. According to the format the patient can have multiple waiting locations while waiting for an activity. For this one-to-many relationship another class is constructed; the *WaitingTime* class stores the waiting time and location related to the *ActivityEvent*.

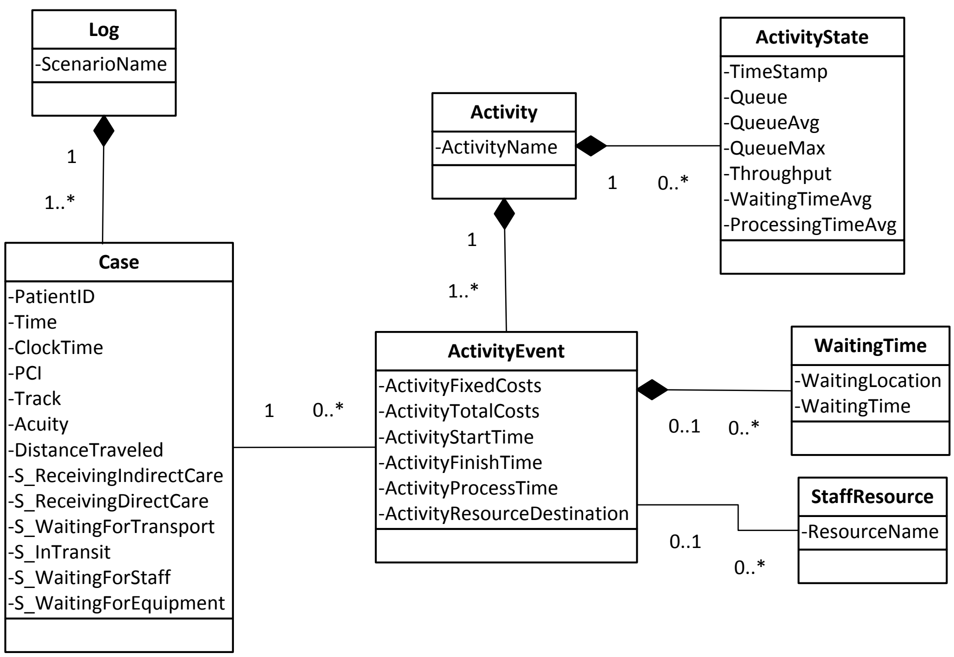
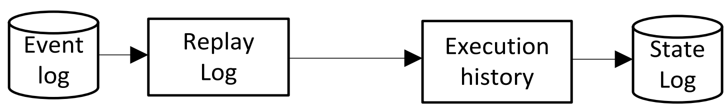


Figure 11: Class diagram for Flexsim HC transformed output

The relationships between the classes *Log*, *Case*, *Activity, ActivityEvent* and *WaitingTime* are depicted in Figure 11. According to Talumis, the distributor of Flexsim HC, the standard Flexsim HC output files cannot provide the explicit link between *StaffResource* and *Activity*. Therefore a custom field in Flexsim HC is used where the relation between *Activity* and *StaffResource* is stored (Appendix J). This field contains a JSON value, which is a programming language independent serialized data object. As Figure 11 shows an event can have none or multiple resources necessary to execute the *ActivityEvent*.

The class model depicted in Figure 11 allows for calculations of waiting times, throughput times, process times and costs for specific patients (*Case.PatientID*) or patient tracks (*Case.Track*). The *Case* table also provides accumulated data for the specific case. Based on the class model an event log for process mining purposes can be generated, this procedure is described in chapter 5.2.

In addition to the patient history log, Flexsim HC provides state history files describing the state of the system every minute. Since the states in these files are not related to activities in the model but to a process or waiting locations it cannot be linked to the *ActivityEvent* class (processing and waiting locations are consumed by an activity) and can therefore not be used to extract activity specific key performance indicators. By replaying the patient history log an execution history of the system can be obtained (Aalst, Process Mining, 2010b). The log describes the states the system went through marked by a timestamp. With this execution log it is possible to assess the state of the system at any point within the event log interval, for example to determine the average waiting time at interval [0,180]. This data is essential to generate time series graphs to depict the evolvement of KPIs over a time interval. Figure 12 shows how the state log is obtained. The replay of the event log marks the state changes, using the state changes the state log describes the state of the system throughout the interval of the event log.



event @time=1
event @time=13
event @time=18

state @time=1
state @time=2
state @time=3
...
state @time=18

Figure 12: Generating a state log describing the system state on every point in interval

The class *ActivityState* in Figure 11 describes the state attributes for an activity with a standard interval of 1 minute. This interval can be adjusted based on the considered time horizon and required level of detail. The state of an activity is described by the queue, throughput, average waiting time and average processing time. In addition to the queue in the execution log the state log also contains aggregate measures such as average queue and maximum queue.

5.3 Control-flow discovery

Control-flow mining is used to discover the process logic embedded in the simulation model. ProM6 is an open source framework which implements a wide range of process mining algorithms and plugins to support process mining techniques. ProM6 can import event-logs describing fact data about activities that are executed. ProM6 supports event log definitions in OpenXES and MXML.

In this study ProM6 is primarily used for the discovery of the control-flow perspective. When this is done appropriately it can provide the control-flow logic of the simulation model and thereby assists in testing and model validation (Nikoukaran, Hlupic, & Paul, 1999).

As already described in chapter 2, the control flow perspective is concerned with the ordering of activities (Weijters, Van der Aalst, & De Medeiros, 2006). The goal of this perspective is to discover a representative characterization of all possible paths in terms of a control flow diagram (Weijters, Van der Aalst, & De Medeiros, 2006). Examples of BPMLs available in ProM6 to mine the control flow are Petri Nets, Event-driven process chains (EPCs), Causal Nets and Business Process Modelling and Notation (BPMN) diagrams.

To explore the different mining algorithms in ProM6 the process as depicted in Figure 13 is used. To start process mining ProM6 requires input in the form an event log. For this project the format OpenXES will be used, a standard for storing event log data (Appendix L). The OpenXES file contains all the process traces based on the data model of Figure 11. For every process trace (*Case*) the log contains a series of records with the activity-name and timestamp of that particular activity. An event is also associated with a lifecycle element. OpenXES supports many lifecycle attributes such as: *start*, *complete*, *scheduled*, *assign*, *withdraw* etc. The serialized log generated for ProM6 contains the elements *start* and *complete* marking the activity start and finished time.

The log is imported into ProM6 where multiple iterations of mining and changing the Flexsim HC models start. This process of mining and altering the Flexsim HC model is depicted in Figure 13. The Flexsim Output Data is generated by a simulation model developed in Flexsim HC. Based on this the OpenXES log is generated which is imported into ProM6.

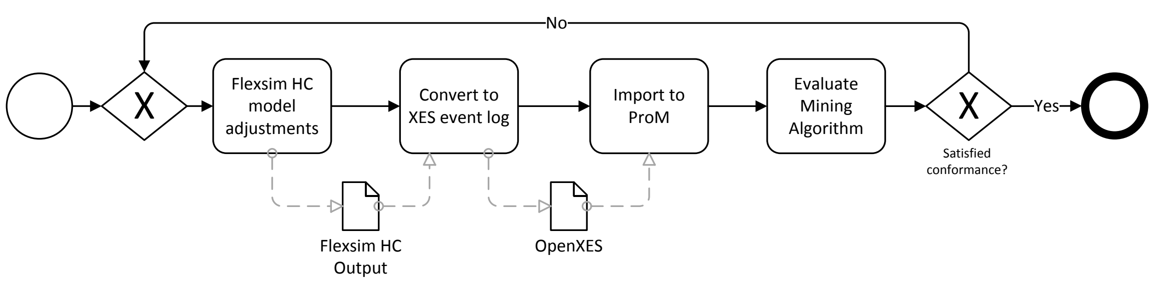


Figure 13: Process mining algorithm evaluation process

5.3.1 Event log generation

The OpenXES event log is serialized using the classes *Cases*, *Activity, ActivityEvent*, and *StaffResource* from the UML class model in Figure 11. Before the log was serialized the following Flexsim HC model adjustments were applied in the patient tracks and output data:

- Naming conventions: use of identical names across patient tracks in Flexsim HC if the process step needed to be the same in the control flow
- Naming conventions: use of different names across patient tracks in Flexsim HC for same activity if they needed to be isolated in the control-flow
- Naming normalization: activities in Flexsim HC are automatically prefixed with an *ActivityID* (see screenshots in Appendix D), the prefix is stripped such that *10_Start* becomes *Start*

These model adjustments were repeated multiple times. The naming of activities across patient tracks has a high influence on the results of the mining algorithm and the conformance of the resulted model. When identical names are used in different patient tracks for essentially different activities in the process this may have a big influence on the mined model.

5.3.2 Control flow mining results

ProM6 provides a wide range of control-flow mining algorithms constructing a control-flow perspective in various BPMLs. The choice for a BPML for the purpose of this study is based on the following aspects: semantics, formalism and readability. The need for control flow semantics like AND, XOR and OR splits are important. Although a patient can only be at one point at a time it is possible that an activity related to the patient is executed without the patient being present. For example an administrative step after the patient already moved on in the process. This is also applicable to the processes in this case study (Appendix A).

Formalism can enhance the re-use and interoperability of models. The implementation of a BPML with a formal specification can leverage this potential. Readability of the model is also an important consideration because the models need to be easy to understand and might be used as a means of communication among process owners with limited knowledge of BPMLs. Based on the given criteria BPMN (Business Process Model and Notation) is selected for the graphical modelling language in the IPV tool. BPMN is formalized in a specification and developed with the intention to create a readily understandable graphical representation of a business process (White, 2003). Also BPMN supports all important control-flow aspects such as exclusive choice and parallel behaviour. Only a small subset of the BPMN2.0 specification is used, this subset is outlined in Appendix M.

ProM6 provides several mining algorithms to generate a BPMN control-flow models. The ProM6 framework provides two plugins to directly mine a BPMN view. The heuristics miner and causal net miner plugins for mining BPMN control-flow views (Bayraktar, 2011). The heuristics miner and causal net miner use an algorithm that mines the control-flow perspective by solely considering the ordering of activities within a case (Weijters, Van der Aalst, & De Medeiros, 2006). The heuristics and causal net miner algorithms deal with noise in the event log and are able to discover AND, XOR and OR splits. These control-flow constructs are important elements for simulation model validation. The ability to deal with noise is less important for simulation logs because they should produce only traces which are defined by the patient tracks.

The heuristics miner in ProM6 with default settings was used to mine control-flow diagrams for the as-is simulation model, the results are depicted in Appendix F. The BPMN model contains the control-flow of the three appointment types: Pre-Operative, post-operative week, and post-operative end. The models depicted in Appendix F show different models obtained after multiple cycles of naming activities. For the appointment types different activity names are used such that the patient tracks can be identified in the process model. This consideration is essential to cross-correlate performance and resource data to original appointment types when the overlays are introduced. This way white-box validation can be applied for specific appointment types (Patient Track) and activities.

Given the fact that Flexsim HC always provide the same order of activities in the output (based on sequence in the patient track, Figure 25 Appendix D), the heuristics miner would not be able to detect the parallel behaviour unless the order of activities in manipulated while generating the event log. Manipulating the order of parallel activities resulted in a model with parallel behaviour but also an OR-join, which is not really desirable. This is caused by multiple patient tracks using different and the same activities in the parallel behaviour part of the process. The problems with poor fitness (6.25% for standard event log and 4.35% for manipulated log) of the model with the event log can be solved by naming the activities for the patient tracks differently. The third model with unique naming is depicted in Appendix F. It shows the BPMN model with 100% path fitness. The model is syntactically not completely correct for the post-operative patients. The parallel paths are joined using an XOR operator. This problem could be solved by adding another extra activity after the parallel activities.

5.4 Overview analysis

In the previous sections the analysis is given. First the main key performance indicators of interest for MUMC+ are given. Consequently by the analysis of the output files of Flexsim HC was described. Based on the data model it is described how the data can provide an event log for the ProM6 framework and how the control-flow can be discovered. This covered research question 2: What is the output of Flexsim HC and what process mining techniques can be applied?

6. Solution design

In section 5.3 the process control-flow mining algorithm was selected: the heuristics miner for mining BPMN models. The mined BPMN models can be used to assess the validity of control-flow of the simulation model compared to the conceptual model (Figure 3). In this chapter the overlays for BPMN process diagram are proposed based on the KPIs of interest for the MUMC+ cataract process as a part of the solution design. Both graphical visualization (graph plots) and data overlays are considered. When the overlays are implemented in the control-flow model an integrated process view is generated which can be used to assess validity of control-flow and performance accuracy in one view (Figure 3).

6.1 Key performance indicators

In Chapter 5 the key performance indicators of interest for MUMC+ were outlined. In the following sections the relation between the data model of Flexsim HC (Figure 11) and the KPIs is given. The KPIs are further defined formally in section 6.1.2.

6.1.1 Key performance indicators based on data model

In the integrated process view (Figure 3) the aim is to cross correlate KPIs and resources to specific activities. In section 4.3.2 the specific KPIs of interest for MUMC+ were given. The calculations of KPIs are dependent on the class model in Figure 11.

In Table 1 an overview is given between the performance indicators and based on what classes they are calculated. For example the average waiting-time for a specific activity is calculated using the class *WaitingTime* in the UML data model in (Figure 11). This measure is also present in the last record of the activity state history for this activity. The variation requires a computation using single values from the *WaitingTime* class. The state history can therefore not be used to calculate the variation. The activity state history is used to provide the evolvement of a KPI over time. For example the average waiting-time time series for a specific activity returns a set of values with the measure calculated on specific points in time, for example every minute. These sets are typically useful to generate time series graph plots.

Table 1: KPIs in relation to UML classes

Description	Type	UML classes
(net) Waiting time	Average, variance	(ActivityState), WaitingTime
	Maximum	ActivityState
(net) Waiting time series	Average, maximum	ActivityState
Service time	Average, variance, maximum	ActivityEvent
Service time series	Average, maximum	ActivityState
Throughput	Cumulative	ActivityState
Throughput time series	Count	ActivityState
Queue length	Average, maximum	ActivityState
Queue length time series	Length, average, maximum	ActivityState
Resource involvement	List of names	StaffResource
Resource Last Finish Time	Value	ActivityEvent. StaffResource
Resource Utilization	Value	ActivityEvent, StaffResource

Table 1 provides an overview of the information and KPIs related to a specific activity. As described earlier, the utilization of resources is also an important KPI for the cost-dimension but this measure is not related to a specific activity. It cannot be assumed that a resource is only involved in one activity at a time; therefore the KPIs such as utilization should be calculated over all activities.

6.1.2 Definitions and formulas

In the previous chapter the main KPIs of interest were determined: Throughput time, (net)-waiting time and utilization of resources. The global KPIs which calculate performance measures for the whole system are defined using aggregation function over values related to an activity, such that the calculation can be drawn as an overlay in the multiple perspective view, e.g. the average waiting time will be calculated by a sum of weighted average waiting times for each activity.

In the following sections the definitions and calculation of various KPIs are given based on the UML class model. The KPIs are separated into **activity specific**, **global system** and **resource related** performance measures and cover the KPIs of interest described in section 5.1.2. Table 2 shows an overview of symbols and the relation to the UML class model (Figure 11).

Table 2: Symbols and relation to the UML class model

Symbol	Description	UML classes (Figure 11)
$\mathbf{T_{c,a}}$	Throughput time for activity a and case c	*Case, Activity, ActivityEvent*
$\mathbf{W_{c,a}}$	Waiting time for activity a and case c	*Case, Activity, ActivityEvent WaitingTime*
$\mathbf{W'_{c,a}}$	Net waiting time for activity a and case c	*Case, Activity, ActivityEvent WaitingTime*
$\overline{W}$	Global system average waiting time	*Case, Activity, ActivityEvent*
$\overline{P}$	Global system average process time	*Case, Activity, ActivityEvent*
$\mathbf{P_{c,a}}$	Processing time for activity a and case c	*Case, Activity, ActivityEvent*
$\mathbf{P_{c,a,r}}$	Processing time for case c, activity a and resource r	*Case, Activity, ActivityEvent*
$\mathbf{E_c}$	Earliness of case c, (time before appointment) time)	*Case*
$\mathbf{E_{c,a}}$	Earliness of activity a for case c	*Case*
$\mathbf{A_c}$	Appointment time for case c	*Case*
$\mathbf{N_a}$	Number of executions for activity a	*Activity, ActivityEvent*
$\mathbf{N_{a,r}}$	Number of executions for activity a by resource r	*Activity, ActivityEvent, StaffReources*
$\mathbf{C}$	Set of all cases c (c=1,2,3…)	*Case*
$\mathbf{A}$	Set of all activities a (a=1,2,3…)	*Activity*
$\mathbf{U_r}$	utilization rate for resource r	*ActivityeEvent, StaffResource*
$\mathbf{H_r}$	working hours for resource r	*ActivityeEvent, StaffResource*
$\mathbf{F_r}$	last activity finished time for resource r	*ActivityeEvent, StaffResource*
$\boldsymbol{\tau}_{\mathbf{r,max}}$	last finish time for resource r	*ActivityeEvent, StaffResource*
$\boldsymbol{\tau}_{\mathbf{r,min}}$	first start time for resource r	*ActivityeEvent, StaffResource*
$\boldsymbol{\sigma}$	Standard deviation	-

Average waiting time for specific activity

The following expressions define the average waiting time for an activity in terms of a sum of the waiting times for the cases.

$$\bar{W}_a = \frac{\sum_{c \in C} W_{c,a}}{N_a}$$

Average processing time for specific activity

$$\bar{P}_a = \frac{\sum_{c \in C} P_{c,a}}{N_a}$$

Average throughput time for specific activity

$$\bar{T}_a = \bar{P}_a + \bar{W}_a$$

Earliness for a case and activity

Earliness is the time that the patient arrives before the first appointment time. The earliness for a specific activity can be calculated as follows:

$$E_{a,c} = max\{0, E_c - \sum_{i=1}^{a-1} (W_{c,i} - P_{c,i})\}$$

Average net waiting time

The net waiting time considers the waiting time minus the average earliness.

$$\bar{W}'_a = \bar{W}_a - \frac{\sum_{c \in S} E_{c,a}}{N_a}$$

Average net throughput time for activity

The net throughput time for an activity is given by the average process times and net waiting time.

$$\bar{T}_a = \bar{P}_a + \bar{W}'_a$$

Global system KPIs

The system waiting time is given by a sum of the activity related waiting times

$$\bar{W} = \sum_{a \in A} \bar{W}_a$$

The net system waiting time is given by the waiting time minus the average earliness of the first activity.

$$\bar{W}'_a = \bar{W} - \frac{\sum_{c \in S} E_{c,1}}{N_1}$$

The system process time is given by a sum of the activity related process times

$$\bar{P} = \sum_{a \in A} \bar{P}_a$$

The throughput of the system is defined as

$\bar{T} = average\ throughput\ time\ of\ the\ system$

$\bar{T} = \bar{P} + \bar{W} = \sum_{a \in A} \bar{P}_a + \bar{W}_a$

$\bar{T}' = average\ net\ throughput\ time\ of\ the\ system$

$$\bar{T}' = \bar{P} + \bar{W}' = \sum_{a \in A} \bar{P}_a + \bar{W}'_a$$

Resource specific key performance indicators

The basic utilization rate is given by the total time a resource is involved working on a case divided by the total time the resource is active in the system. In this case study the total time of resources active in the system is given by the first activity start time of a resource and the latest finish time of an activity. Since the planned working hours for resources are not known based on the class model (Figure 11) the utilization rate provides an approximation. Here it is assumed that the resource is dedicated full time to the process. Because the working hours are not known the interpretation of utilization may give some problems. Therefore the number of hours (H_r) over which the utilization is calculated is also an important measure to include.

$$H_r = \tau_{r,max} - \tau_{r,min}$$

$$U_r = \frac{\sum_{c \in C} \sum_{a \in A} \sum_{r \in R} P_{c,a,r}}{H_r}$$

The average processing time for a specific activity and resource is defined as:

$$\bar{P}_{a.r} = \frac{\sum_{c \in C} P_{c,a,r}}{N_{a,r}}$$

The IPV tool also calculates the standard deviation of the key performance indicators. The standard deviation is very important for validation purposes. If the mean is accurate but the standard deviation is not, this indicates inaccuracy in the model. The standard deviation is calculated according to the following formula:

$$\sigma = \sqrt{\frac{\sum (X - \bar{X})^2}{n - 1}}$$

6.2 Extending the control-flow perspective

The previous chapter provided the control-flow perspective and the previous section defined the KPIs. In the following sections overlays for the control-flow perspective are described. The activity overlays are engineered with the information needs (KPIs) in mind such that the information can be placed in the process context. The general model proposed by Rozinat

(2010) is a static model, i.e. all data is displayed in one picture (Figure 6). In addition, a lot of textual elements are used. Using a more dynamic approach could provide a more comprehensible / cleaner views while still provide all the data. The overlays available in the ProM6 BPMN plugins provide more graphical and dynamic elements such as rollovers (when the mouse pointer is on the activity data is showed). This dynamic approach is further explored in this project such that more information can be addressed in the control-flow perspective. By implementing the information using multiple 'pages' inside the activity node the model provides additional information in a comprehensible way, the structure of the process model is not affected by this overlay. The model doesn't get more edges or nodes; instead the space inside the activity nodes is used to display the information. Minimizing edges and the implementation of pages reduces the amount of relevant information pieces in one view. This lowers the cognitive load of analysing the process model. Research in understandability of process models claim that a reduction of the cognitive load would contribute to the comprehension of process models (Figl, Recker, & Mendling, 2013).

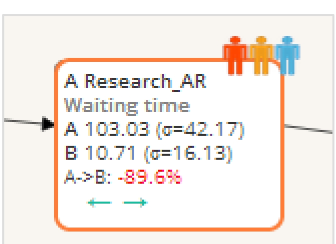
In short the following principles are used to construct the overlays:

- Minimize number of edges in the model
- Provide performance measures cross-correlated to activities
- Provide access to many performance measures without information overload
- Use graphical elements instead of textual where possible
- Possibility to visualize multiple models to show comparison

6.3 Activity overlays

In a control-flow diagram each activity is related to the perspectives time-, data- and resources. For the time perspective it is possible to generate time series plots to depict the evolvement of key performance indicators over time. But in addition to this graphical representation aggregate measure of these KPIs are often very important, especially when white-box validation is performed for a specific activity. Based on Table 2 the following performance indicators which reference to an activity can be included in an activity node:

- (net) waiting-time: average, maximum
- Service-time / processing time: average, maximum
- Throughput: count
- Queue: average, maximum



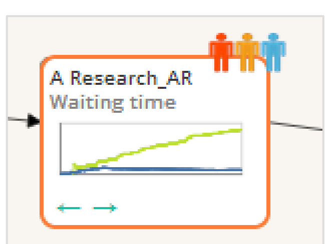


Figure 14: Overlay waiting time, left in terms of values, right in a time series plot

Although the queue length was not specifically a KPI of interest for MUMC+ these congestion measures can be very useful to identify the relationship between waiting time and congestion measures such as the queue length. Note that the given KPIs can be represented both as a measure for the whole simulation run or as a time series plot showing the evolvement of the measure over time. The overlays include time series plots for the waiting time, service time, throughput and queue. Figure 14 shows the overlays for a specific measure and as a time series plot.

Where the general model of Rozinat (2010) uses the whole model surface to link qualitative and quantitative information to activities using edges, the model presented here leverages the surface in the activity node to display relevant KPIs and information associated with the activity. Since the size of the activity node is of great influence on the size of the model a more dynamic model approach is chosen. Instead of providing all the KPIs in the activity node the node consists of several tabs/pages. Each page presents a KPI or chart to present the evolvement of the KPI over time. The resources are displayed as icons in the top right corner of the activity, the colour of the icon relates to the resource and specific activity. Resource information becomes available when the mouse pointer is on the icon (Figure 15). This more dynamic approach of implementing three perspectives provides a model which uses fewer edges. According to literature about understandability of process models, having a model with fewer elements reduces the cognitive load of analysing the model. This reduction of the cognitive load would improve comprehension of the model (Figl, Recker, & Mendling, 2013).

6.3.1 Staff resource allocation and utilization overlays

For the organisational perspective an activity can be associated with one or more resources. This resource perspective is not present in the BPMN specification (OMG, 2011). In BPMN swim lanes are used to group activities to participants or roles (Wohed, Aalst, Dumas, Hofstede, & Russell, 2006). An example of a model with swim lanes can be found in Appendix R. As the example in Appendix R illustrates, a relatively simple and small model easily becomes a spaghetti model. In the model proposed in this project an icon is used to create the link between the activities and resources to provide insight in who performed what activity and at what performance (Figure 15).

The use of icons in process models can improve the understandability of process models because of the direct relationship with their meaning (Recker, Safrudin, & Rosemann, 2012). The icons are coloured according to the resource name. When the mouse-pointed is on the icon it provides KPIs cross-correlated to the specific activity and resource. It provides the throughput of patients for the specific activity and resource, the global resource utilization rate, related working hours, average last activity finished time and processing average and standard deviation for this specific activity and resource.

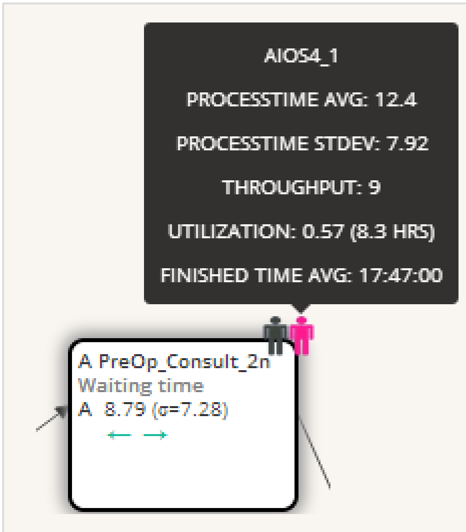


Figure 15: Cross relating to specific activity + resource information

6.3.2 A/B what-if scenario comparison

When simulation results are evaluated or when a model is validated the output is often compared with output from other models or real system data. The IPV tool provides the possibility to compare two simulation logs (Figure 3).

When two result-sets are compared the models are depicted one below the other. The view links two activities with the same border colour and when the mouse pointer is on an activity it will create an edge between the two models to make it easier to find the related activity. Activities with a black border colour are not present in the two models, in Figure 16 this is the case for the activity “Research_New”.

The activity node also shows two measures. This allows for easy comparison of performance measures without searching in two models. For example the waiting time for activity “Research_AR” for log A in Figure 16 is 10.79 and for log B 5.21 which is in fact 51.69% lower. The A/B comparison is particularly helpful in:

- Understanding relations and influences of different what-if scenarios on KPI
- Determining or validation of performance improvement
- Validation of changes in control-flows (changes in models)
- Validation of KPI changes

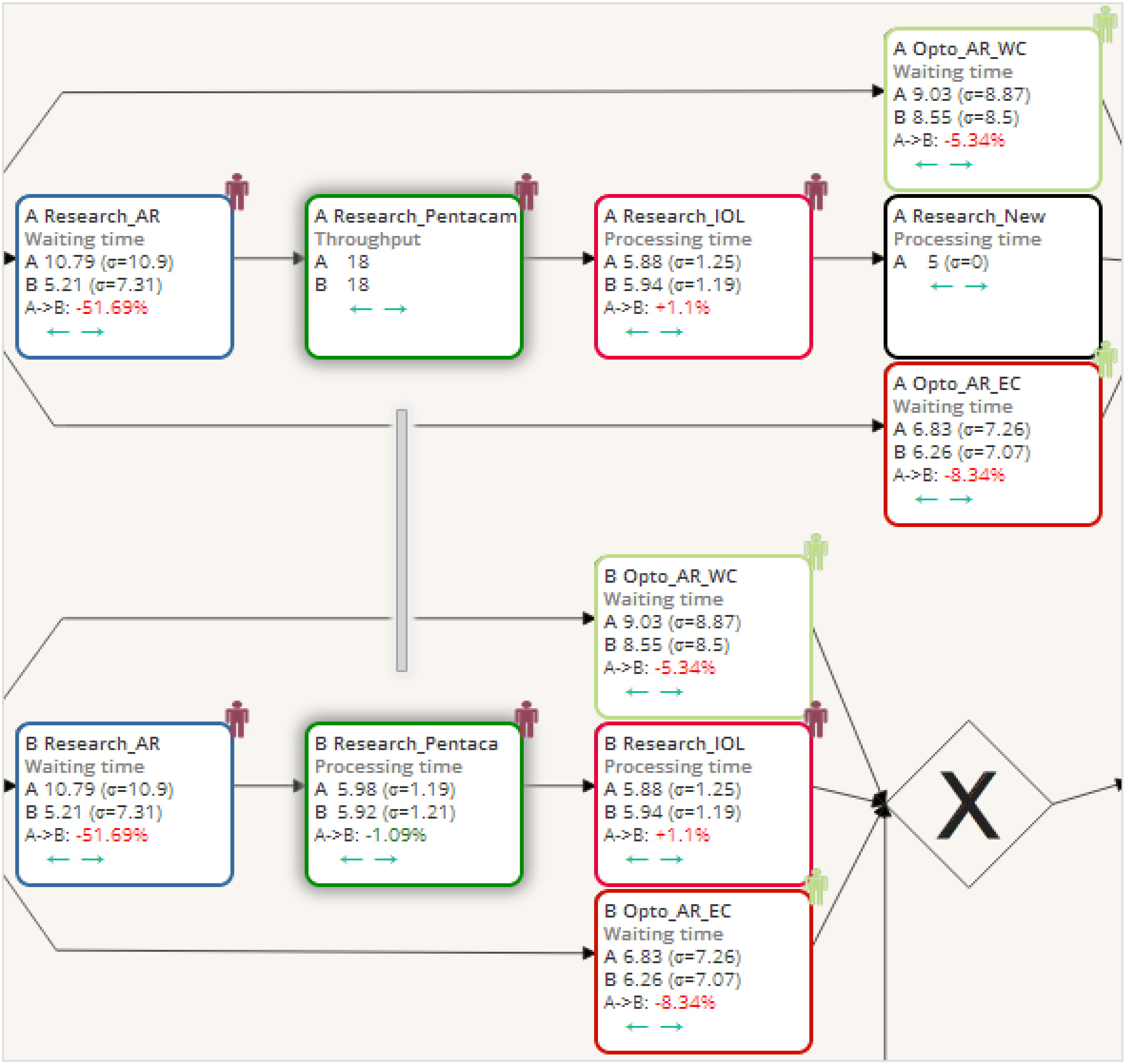


Figure 16: Comparison of two processes by edge links between similar process names

6.3.3 Global KPIs

Using specific activity KPIs is not sufficient when validating overall system performance (black-box validation). In chapter 6.1.2 the definition and formulas for system performance were given. In the IPV tool global KPIs can be selected (global (net)waiting time and processing time) as overlays. Once selected, the activity nodes display activity related values used in the calculation. In Figure 17 an example of the overlay for the global system waiting time is given. The first activity node accounts for 3.74% of all the net waiting time and adding 3.38 minutes to the overall average waiting time in the system, the cloud in Figure 16 represents a series of other activities. The global system waiting time is showed in the end event, which is in the example 90.49 minutes.

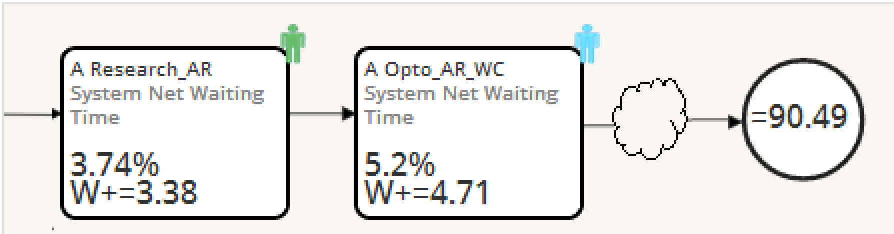


Figure 17: System waiting time contribution of single activity

6.4 Comparison to existing methods

The ProM6 BPMN plugins also provides performance analysis as overlays. This could be considered an alternative to the IPV tool created in this project, since it can also provide the integrated view as shown in Figure 3. The BPMN plugin calculates performance measures based on the event log's timestamps. Table 3 contains the performance measures available as overlays in the BPMN Analysis plugin developed by Bayraktar (2011). In addition to the activity related key performance indicators, the plugin also provides global system measures for throughput time, throughput count and arrival rates. In Figure 18 an example of overlays in ProM6 for the BPMN plugin is given. Appendix R provides a legend with more detailed explanation of the overlay. In Table 3 an overview of the performance data available in ProM6 is given. Since this integrated view available in ProM6 can also be used for validation of simulation model accuracy a comparison between the BPMN ProM6 performance overlays and the integrated view of this project is given in the following sections.

Table 3: ProM6 BPMNAnalysis plugin performance overlays

Description	Type
Waiting time	Average, Minimum, Maximum
Throughput time	Average, Minimum, Maximum
Synchronisation time	Average, Minimum, Maximum
Throughput	Count, Percentage

6.4.1 Comparison BPMN plugin and IPV tool

To mine the control-flow perspective the BPMN plugin in ProM6 was used. The BPMN plugin in ProM6 provides overlays for the throughput (count, percentage of choice), synchronisation time, waiting time and service times (Table 4, Figure 18). This data might be enough for validation purposes but there are some shortcomings. To start, the BPMN plugin does not provide standard deviations and variance of measures. These measures are important for validating the white-box measures. For example the real life processing time for an activity needs to be validated. The mean might be accurate, but when the variance is not taken into account this might lead to a false conclusion about the accuracy. When the variance is way off compared to the real system this may indicate that the used distribution or data in the simulation model is wrong, therefore the access to the variance specifically for an activity is important. In Table 4 a comparison between KPIs available in the BPMN plugin and the IPV tool is given.

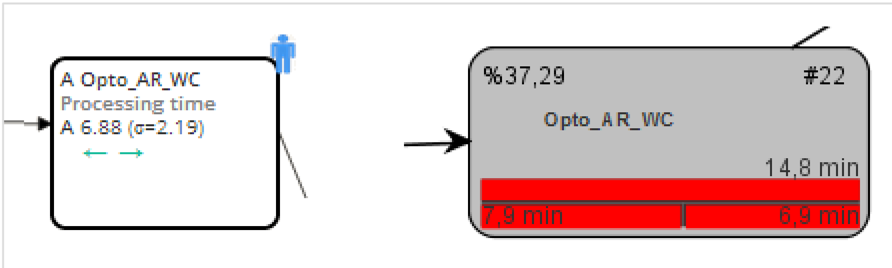


Figure 18: Overlay in IPV tool and ProM6 BPMN plugin

The BPMN plugin does not support all lifecycle elements for the OpenXES standard. For example lifecycle *scheduled* is not supported in the BPMN plugin (Bayraktar, 2011). Therefore a calculation of net waiting time is not possible. The waiting time and throughput times are calculated using the time between subsequent activities; the time is calculated as the difference between the last activity *lifecycle::completed* and next activity *lifecycle::start*.

Figure 18 and Table 5 show the overlays and performance data for an activity from the same log. The overlays show the same performance data for the waiting time and processing time (Figure 18) Also the global system measures provided by the BPMN plugin are equal to the one measures of the IPV tool (Table 5).

Table 4: Comparison of activity specific KPIs in BPMN plugin and IPV tool

	Available in BPMN overlays	Available in Tool
Net waiting time avg	No	Yes
Net waiting time stdev	No	**Yes**
Waiting time avg	**Yes**	**Yes**
Waiting time stdev	No	**Yes**
Waiting time min/max	**Yes**	No
Processing time avg	**Yes**	**Yes**
Processing time stdev	No	**Yes**
Synchronisation time avg	**Yes**	No
Synchronisation time min/max	**Yes**	No
Queue length avg	No	**Yes**
Queue length max	No	**Yes**
Throughput count	**Yes**	**Yes**
Throughput percentage	**Yes**	No
Resource utilization	No	**Yes**
Resource specific processing times	No	**Yes**
Resource number of patients	No	**Yes**

Table 5: Overall system performance BPMN vs IPV tool

	System throughput	Standard Deviation
BPMN plugin	1.36 hrs = 81.6 min	39.78 min
Tool	81.59 min	39.62 min

6.4.2 Conformance

The conformance of a model is addressed by measuring how many traces in the event logs can be executed using the discovered process model (Aalst, 2010b). The IPV tool assumes that all the activities are present in the imported control flow. The calculation performance indicators in the IPV are independent on the control flow. When a control-flow view with a low conformance is imported in the tool the measures for the specific activities will be equal to a view with high conformance, but the context of the activities might be inaccurate. To ensure the correct representation of the simulation output the conformance should be addressed before exporting the model to the IPV tool. The ProM6 BPMN plugin provides conformance measures. Here the percentage of conformant cases can be addressed. As already discussed, conformance is an important measure because it reflects how well the mined model reflects the log data.

6.4.3 Overview BPMN Plugin

Where the ProM6 BPMN plugin provides a small variety of KPIs, it cannot depict the net waiting time and net throughput time. Also time series graph plots and the congestion measures such as the queue length are not provided in ProM6. The control-flow view also does not provide performance measures related to the resources, such as utilization or specific resource performance measure such as average processing time. Some of the measures simply cannot be calculated in ProM6 because the IPV tool relies on a richer data model. The most important shortcoming identified in the BPMN plugin is the absence of the variation in measures. Having access to the standard deviation or variance is important for white-box validation because this shows how accurate the uncertainty is in comparison with the real system (Robinson, 1997). Therefore the BPMN plugin cannot be used in this case as the integrated view to provide the information to validate the simulation models

6.5 IPV - Tool

The Integrated Process View (IPV) tool developed during this project implements the overlays described in the previous chapter. The tool is created on top of web development frameworks and open source projects. In the following sections the tool functionalities will be described shortly, followed by an overview of the technologies used in the tool. Screenshots of the IPV tool are included in Appendix V.

6.5.1 Tool functionalities

As already mentioned, the IPV tool is primarily aimed at providing a view to validate Flexsim HC simulation models. Because the control-flow and performance data is available in one view it is easier to perform white-box, black-box validation and determine whether the control-flow is a correct representation of the desired system. The IPV supports a wide range of model changes to be visible in the tool. When the control-flow is changed this is visible in the IPV tool. Also resource assignments are very well supported, because the view shows the relation between activities and resources. Other changes in the model which affect the performance are notable in the performance overlays. Screenshots of the functionalities can be found in Appendix V.

The IPV tool provides the following functionalities:

- Import Flexsim HC output (max 41 replications)
- Export OpenXES event log for ProM6 (Appendix L)
- Import XPDL process model from ProM6
- Generate integrated view of control-flow, performance and resources (Figure 16)
- Show global system performance measures as overlays in BPMN (Figure 17)
- Performance measures related to activities: (net)waiting time, process time, queue length, throughput counts, graph plots
- Compare two scenarios on performance measures and control-flow (Figure 16)

6.5.2 Tool technologies

The tool created as part of this project is built upon several technologies. The IPV tool runs on an Apache webserver using PHP5.4-backend and a HTML5 front-end generated with JavaScript. The data was stored in a MySQL database. The open source BPMN Javascript engine of the Camunda project (Camunda.org) was used to generate a Scalable Vector Graphic (SVG) based on a BPMN2.0 file. The BPMN model specification file is the result of a transformed XPDL file exported from the ProM6 framework. The XPDL to BPMN model transformation is an implementation of the mapping described by White (2003). The model-to-model transformation supports events, tasks and decisions (AND, XOR, OR), the supported BPMN elements be found in Appendix M. Figure 19 shows the relation between the different software packages (Flexsim HC, ProM6), file formats (XPDL, BPMN) and the IPV tool (SVG, JSON).

The extensions are generated by the IPV tool developed in this project. The plugin relates the activities in the BPMN file to the data model in Figure 11 using serialized JSON object such that the overlays are linked to the activities in the SVG.

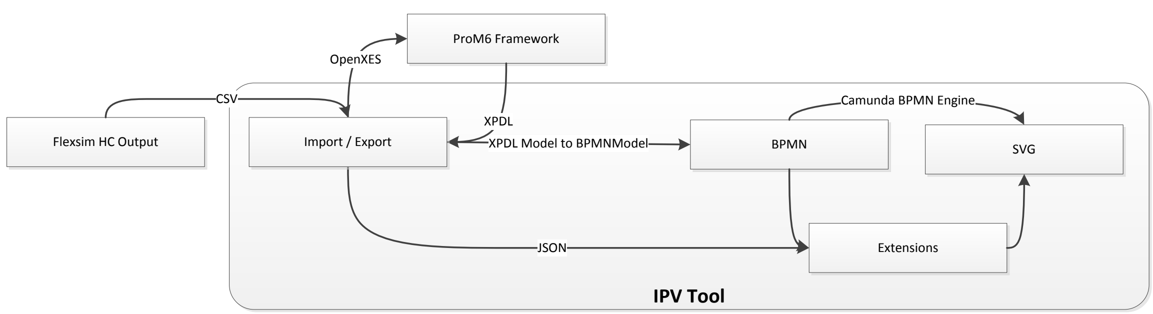


Figure 19: IPV tool context for generation of 3-perspective view

7. Simulation case study

In the previous chapter the IPV tool was introduced. This tool can automatically generate an integrated view of control-flow, performance and resource data. In this chapter the case study is described. This chapter first describes the processes and gives an overview of what-if simulation scenarios of interest for the MUMC+. Finally this chapter provides the simulation results and describes how the IPV tool was used as a validation instrument and how the case study benefits from having such a tool in addition to visual animation models.

7.1 MUMC+

MUMC+ is a hospital employing circa 5,200 employees facilitating around 715 beds for patients (MUMC+, 2012). MUMC+ is an academic hospital; this implies their goal is not only facilitating care but also research and education in the medical domain are important goals. Moreover, MUMC+ is not only striving to improve care but also exploring new methods to improve their operational processes (MUMC+, 2012). The case study for this project is conducted at the cataract centre in MUMC+; an outpatient clinic specialized in the treatment of cataract in a high volume of patients.

7.2 MUMC+ processes

The cataract centre is located at floor level 2 of the ophthalmology tower of MUMC+ (Appendix H). At this level the pre-operation and post-operation consults are performed. Floor levels 3 and 4 of the ophthalmology tower are used for the surgeries. The MUMC+ cataract centre at level 2 operates on Mondays and the afternoons of Wednesday and Thursday. When the term cataract centre is used throughout this report it refers to the processes at floor level 2. In the following sections the clinical processes for the as-is situation are described.

7.2.1 Cataract centre process (as-is)

In the following sections three different appointment types for the cataract centre are described followed by a short description of the scheduling layout and human resources working on the patient process.

Diagnosis

Cataracts are cloudy areas in the lens inside the eye. Where a normal clear lens allows light to pass through an eye with cataracts will result in blurry vision. This light-scattering disorder is typically acquired with age (>50 years) (Shiels, Bennett, & Hejtmancik, 2010). Cataracts can develop in one or both eyes. In most cases cataracts develop in both eyes which require two separate surgeries.

A patient is normally referred to the MUMC+ by a general practitioner (GP), eye physician or optician. When a patient is referred to the MUMC+ for visual acuity (VA) complaints it receives a general consult. After the general consult the patient is either dismissed because the patient has another disease not related to the eyes or does not want an operation to cure the VA complaints or continues and is planned for a consult in the cataract centre, where the cataract process really starts.

Pre-operative appointment: New patient

When a patient is diagnosed with cataract in a general consultation and he or she wishes to get treated an appointment is scheduled for a pre-examination. First the AR-, Pentacam-, IOL-measurements are taken. The measurements are taken by a technical ophthalmology assistant (TOA) or in rare cases the optometrist. Subsequently the results of the measurements are discussed in consultation with the doctor and the patient receives eye-drops to widen the eye. The process to widen the eyes takes about 20 minutes, the patient is in the waiting room during this period. When the eyes are wide enough the second consult is performed. After the second consultation the doctor refers the patient to the planning office at floor level 3 of the ophthalmology tower to schedule the operation while the doctor performs some administrative actions. A process model for pre-operative patients can be found in Appendix A.

Post-operative week control appointment

During the post-operation control consult an AR-measurement is taken before the consult, this measurement is taken by an optometrist. In addition, the optometrists perform some tests to measure the visibility. After the tests the patient moves to the waiting room. Subsequently the patient receives a consultation with the doctor about the results of the optometrist. When the patient leaves some administrative actions are taken. The post-operative process model can be found in Appendix A.

Post-operative end control appointment

Four weeks after the operation the patient is scheduled for a post-operative appointment with the optometrist and with the doctor. The process steps are identical to the post-operative week control. But the tests take some more time and a prescription for glasses is given by the optometrist.

Scheduling

The volume of patients which is distributed over two days every week: Monday, and the afternoons of Wednesday and Thursday with a total volume of 101 patients. The planning layout used for the cataract centre is included in Appendix G. In the scheme there are three different appointments. The pre-operative patient (np01), post-operation week control (wc), end-post-operation control (ec). The post-operation control is a check one week after surgery. The end-post-control is 4 weeks after the surgery.

Resources

In the as-is situation the following resources are required at the cataract centre:

- 2 doctors
- 1 technical ophthalmic assistant (TOA)
- 1 optometrist
- 1 nurse (located at floor level 3)

7.2.2 Process tasks

In Table 6 the process steps in the cataract-centre are outlined. The times in the table are based on the day schedule layout which can be found in Appendix G. A very abstract overview of the processes for pre-operative and week / end post-operative patients can be found in Appendix A.

Table 6: Process steps and target process time set by MUMC+

Task	Target process time	Resource(s)
Pre-Operative measurements	20 minutes	Technical Ophthalmic Assistant
Pre-Operative Consult 1/2	20 minutes	Doctor
Pre-Operative Consult 2/2	10 minutes	Doctor
Post-Operative Week measurements	10 minutes	Optometrist
Post-Operative consult	5 minutes	Doctor
Post-Operative end measurements	10 minutes	Optometrist
Post-Operative end consult	5 minutes	Doctor

7.3 Simulation models

Figure 20 shows the relations between the real system, the legacy simulation models developed in the master thesis project of Van Balkom (2013), the as-is simulation model and the what-if simulation models. The picture also shows how validation relates to the simulation models and actual system. The AS-IS simulation model and what-if simulation models are used in the case study. To create the models, time measurements were performed in the cataract centre to get the average and variation in the processing times. Also the arrival times were monitored during the observations.

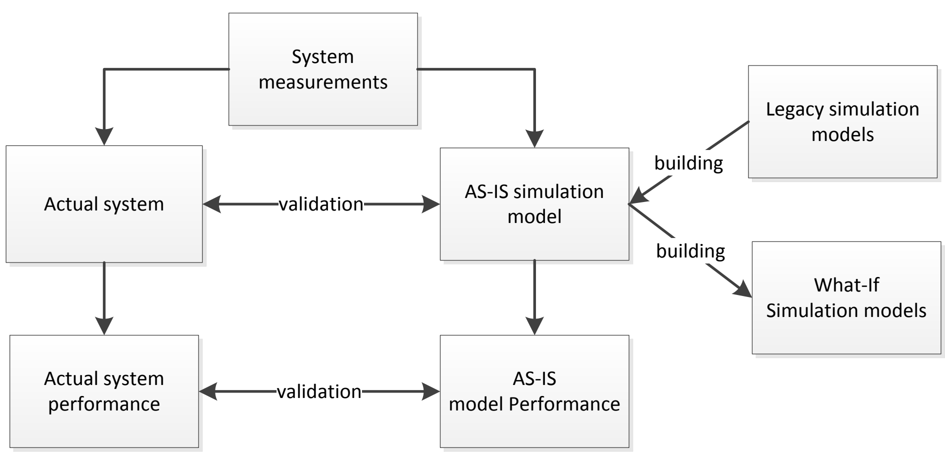


Figure 20: Relation between actual system and models

7.3.1 AS-IS simulation model

Since the processing times used in the legacy models no longer apply new measurements had to be taken. In total approximately 30 process measurements over 3.5 operating days were taken for each of the process steps: Pre-Operative measurements, optometrist measurements, Pre-Operative consult, 2nd Pre-Operative consult, week and end post-operative consults and administrative activities. The measurements were fit on a distribution using ExpertFit, a software package distributed with Flexsim HC. The results of the measurements and fitted distributions can be found in Appendix N. The fitted distributions were all significant ($p<0.05$) and were implemented into the simulation model. The coefficient of variation for the service times for the activities varies between 0.20 and 0.59. This amount of variation is fairly low. The extensive literature review of Cayirli & Veral (2003) found service time values for the coefficient of variation varying between 0.35 and 0.85.

7.3.2 What-if scenarios

With the process owners of the cataract centre what-if scenarios were constructed. Because the scheduling system is an important factor for the MUMC+ cataract centre first some important scheduling constructs are defined followed by an overview of the what-if scenarios.

What-if #1: Adding a measurement to pre-operative track

MUMC+ recently acquired a new eye measurement computer. This computer is an addition to the AR-, IOL- and Pentacam measurements. In the what-if scenario the additional measurement is taken by the TOA. To simulate the additional task an assumption for the increase in process time was made. Because the processing time for this measurement was not obtained during observations the distribution is estimated using the beta PERT approximation technique (Farnum & Stanton, 1987). In this approximation the minimum, maximum and median of the processing time (based on estimation) is used to estimate the distribution. The calculation of beta PERT can be found in Appendix N. This what if scenario will be referred to as “as-is plus measurement”. The IPV tool for should show differences in control-flow (extra activities) and additional KPIs and influences on other performance measures.

What-if #2: Influence lateness of doctors

According to the management of the cataract centre it happens that sometimes the consultations of the doctors start late. This may be due to an overtime morning schedule which occupied the rooms but also due to unpunctuality of doctors. In this scenario the influence of lateness will be addressed by starting 15 or 30 minutes late to address the influence of this unpunctuality on patient throughput times. The following two variants are simulated:

- What-if #2.A: AIO 15 minutes late
- What-if #2.B: AIO 30 minutes late

These scenarios will not affect the control-flow. Process should be similar to the as-is model. The scenarios should influence the performance measures for throughput time and waiting times.

What-if #3: Use individual block / fixed interval

The as-is planning layout uses variable interval between appointments (Appendix G). This what-if scenario can provide insights in the consequences of switching to an individual block (1 patient per time-slot), fixed interval scheduling system. The theoretical background on appointment systems can be found in Appendix U. This would only apply to the appointments for the optometrist and the TOA because they handle the first appointments of the patients. This scenario with fixed block scheduling is further referred to as “fixed-block” scenario. This scenario has no influence on the control-flow of the model. Changes should be noticeable in waiting times.

What-if #4: Schedule week control first or end control first

In the legacy process scheduling the planning was arranged such that week control patients were scheduled first, followed by all the end control patients. To predict the performance of this scheduling rule, two what-if scenarios were constructed where the week control-patients are scheduled first or end control patients first:

- What-if #4.A: Week control first, then end control
- What-if #4.B: End control first, then week control

This scenario has no influence on the control-flow of the model. Changes should be noticeable in KPIs, since the week post-operative control patients require less processing time this should influence the throughput time.

What if #5: Eliminate scheduled gap between measurements and consults

In the current schedule layout there is a gap between the planned first consult and the second consult (Appendix G). In this scenario the planning layout is altered such that the gap is 0 minutes. This could potentially benefit from a reduced waiting time. This scenario is referred to as “schedule 5 min delay”. This scenario has no influence on the control-flow of the model. Changes should be noticeable in KPIs.

What-if #6: Adding extra resources and increase number of patients

In the legacy process for the cataract centre scheduled all patients on one day. In the as-is the cataract centre operates two full days. In this what-if scenario the capacity is increased by one optometrist and one doctor. The planning for this scenario is based on the time slots of the as-is cataract centre (Appendix G). Since the TOA will become the bottleneck it’s important to also assign pre-operative measurements to the added optometrist. The layout of the system is constructed using the existing layout and meetings with the process owners. This scenario will be referred to as “3pci”. This scenario should show changes in resource aspect of the model. The added optometrist executes activities normally only executed by the TOA in addition to the activities also executed by the optometrists. Next to this, an AIO is added for activities executed by a doctor. Activity KPIs will be influenced since the TOA shares the measurement equipment with the added optometrist.

What-if #7: Planning scenarios

The planning layout currently used at MUMC+ has a break during the afternoon (Appendix G, Figure 37). In this what-if scenario alternative layouts are tested. In eight alternative planning layouts (Appendix G, Figure 38) the influence of the break is tested in an attempt to discover a better time slot to plan breaks.

Overview simulation scenarios

The previous sections described the what-if simulation models. In Table 7 an overview is given of the expected changes of the what-if scenario to be observed in the integrated view generated by the IPV tool. The what-if scenarios both provide noticeable changes in the control-flow view and performance measures in the model.

Table 7: Expected results

Scenario	Changes control-flow	Expected changes in KPIs
What-if #1	Added activity	Higher throughput, longer waiting times, longer service times, higher TOA utilization
What-if #2-3	-	Longer waiting times
What-if #3	-	Unknown
What-if #4	-	Unknown
What-if #5	-	Unknown
What-if #6	Resource allocation (who performs what task?)	Influence new track on KPIs, additional resources, Longer throughput times because shared resources for TOA and Optometrist
What-if #7	-	Lower throughput times, lower utilization, longer working hours

7.4 Simulation results

With access to an integrated view to support model validation and result exploration the case study is used to conduct a simulation study. First the model validation for the as-is situation is addressed. Consequently an overview of the simulation results for the what-if scenarios is outlined in section 4.5.2. The validation was conducted with the process owners of the cataract centre.

7.4.1 Model validation

As the picture in Figure 3 suggests the integrated view in the tool was created to determine whether the simulation model correctly reflects the conceptual model of the cataract centre. Since there were no really conceptual models except for the process models in Appendix A the mined models in the integrated views were used in discussion to determine whether the simulation model correctly reflected the desired system. This was done by analysing the control-flow in the IPV tool and discussing whether this model correctly reflected the desired system. The discussion helped at determining whether the correct paths were modelled into the simulation model.

To validate the accuracy of the developed as-is model both white-box and black-box validation was conducted. During an evaluation session with the process owners the IPV tool was used to assess the KPIs of specific activities and explore the relationships of KPIs among various activities. In this evaluation the waiting times of specific activities are of particular interest. Because the process times are fitted on statistical distributions they are normally not the cause of inaccuracy (unless there is an error in the simulation model). When the waiting-times of activities do not accurately reflects the real system this might indicate that the simulation model is incorrectly build or does not support subtle model behaviour which is present in the real system.

In the case study the as-is model was inaccurate (Figure 21, model A). The integrated view suggested that there was something wrong with the as-is simulation model because the waiting time for pre-operative appointments for the second consultation were too high. This led to the discovery of an error in the priority rule for this activity (*PreOp_Consult_2n*, Figure 21). After this error was fixed the performance of the model reflected the system more accurately (Figure 21, model B). When only black-box validation was applied, this error may not have been discovered, because white-box validation can provide more detailed insights of specific activity related accuracy and relation of KPIs between activities as shown in Figure 21.

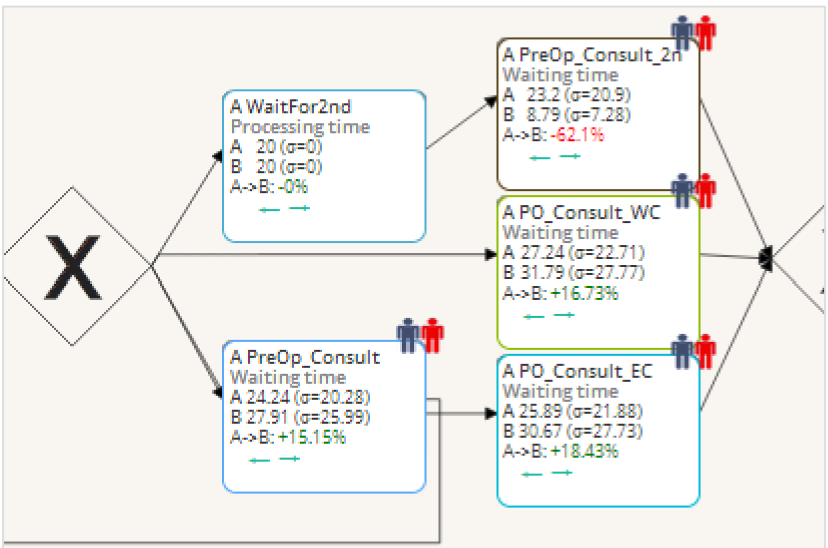


Figure 21: Example of comparison wrong model (A) and correct model (B)

In addition to the as-is simulation model also the simulation models where the control-flow changed were validated and compared to the as-is simulation model. In what-if scenario 1 and 6 (described in section 7.3.2) the control-flow showed the model changes. For scenario 1 it was observed that there was an extra activity (*Research_new*, Figure 16). For what-if scenario 6 it was validated that there were new resources (extra optometrist and doctor) related to the activities (Appendix P, Figure 43).

In Table 23 and Table 24 (Appendix P) an overview of the comparison is between the real system's KPIs and the simulation model is given (the simulation measures were extracted from the integrated view). Based on this comparison the as-is simulation model was accepted as accurate by the process owners (after fixing the priority error).

7.4.2 Simulation results

The simulation output results were obtained by an average of 41 simulation runs. The model was accepted as valid given the acceptable differences in throughput times. Often, simulation studies require a warm-up length such that systematic error caused by cases entering an empty system can be reduced (Fishman, 2001). Since the system under consideration is finite and only have a small run-time period (about 9 hrs. for circa 60 cases) no warm-up length was defined.

In Figure 22 the net throughput times with 95% confidence intervals are given. The results are based on 41 simulation runs. A higher number of runs were not possible because of the memory limitations in the IPV tool. Although more replications may be desirable, 41 independent replications are enough to assess and compare the performance. As the results show, the what-if scenario with alternative planning layouts provide the lowest average throughput time. Planning schema 8 and 7 (Appendix G, Figure 38) are significantly different than the as-is scenario. The significantly worse performing scenarios, based on 95% confidence interval using Student t-distribution, are the scenarios where the AIOs start 15 or 30 minutes late.

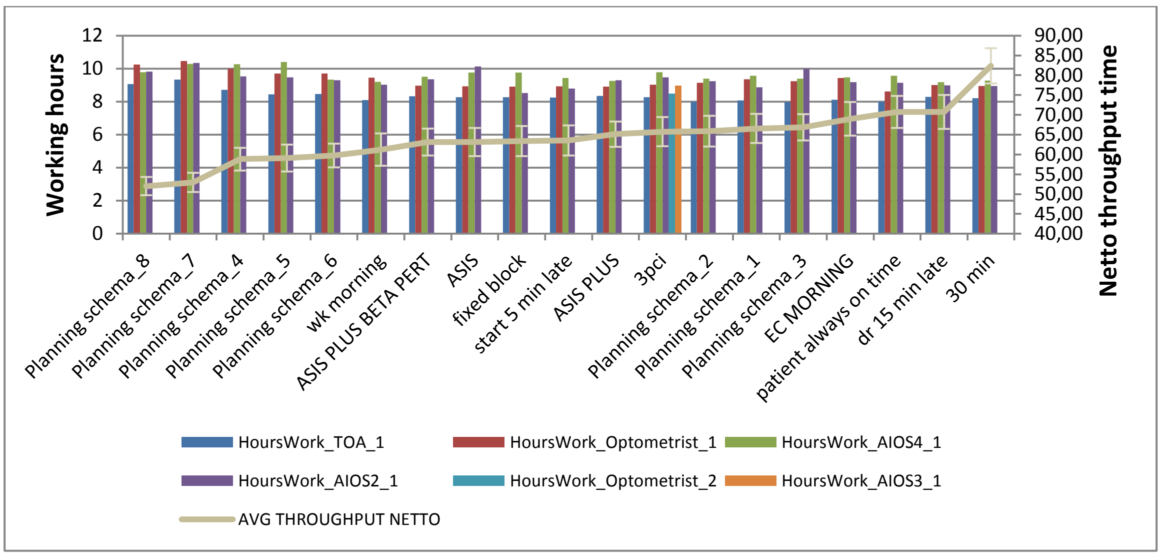


Figure 22: Simulation net throughput times with 95% confidence intervals

The overall results show that the as-is planning layout provides overall sufficient performance in relation to the working hours of staff resources (Figure 22). Planning schema 8 and 7 (Appendix G) perform significantly better on throughput time but also yield longer working days. Planning schema 6 might also be an alternative to the *ASIS* planning layout. Regarding adding an extra activity (*ASIS PLUS BETA PERT*) the results show that this is not significantly influencing the performance (Figure 22). Also the scenario with extra resources (*3pci*) is worth considering because the performance decrease is acceptable. The full simulation results can be found in Appendix Q.

8. Experiment

In the previous chapters a tool was created to support validation of simulation models. The goal of the experiment is to provide support and rationale for the solution direction chosen in this project: providing control-flow views (BPMN model) for validation of Flexsim HC models. The experiment compares Flexsim HC models with BPMN models to discover the importance of having access to control-flow views in simulation model validation.

8.1 Experiment setup

In the experiment two models were provided among two groups (Table 8). The participants were asked questions related to the control-flow of a simulation or BPMN model. The questions were related to concurrency, exclusivity, order and repetition used to measure the understandability of the model. According to Lau & Gadatsch (2011) these types of questions have been used in numerous other studies which focused on measuring the understandability of business process models.

Table 8: Experiment setup

	First model	Second model
Group 1	Flexsim HC	BPMN
Group 2	BPMN	Flexsim HC

For the experiment consultants and students familiar with Flexsim HC were asked to participate. According to Talumis, the distributor of Flexsim HC in the Netherlands there are only three active licenses. Former students with experience in Flexsim HC were also invited as well as some employees of Talumis. Because the participants are widely spread across the country a remote experiment was setup using an online accessible questionnaire with access to the Flexsim HC models. The experiment was also published on a forum for Flexsim HC users, which unfortunately did not yield any response.

8.2 Hypotheses

The results of the experiment were tested against three hypotheses:

H1: Response time of control-flow questions related to Flexsim HC models are longer than BPMN models

It is expected that the time required to answer a question related to a Flexsim HC model takes considerably longer than a BPMN model. When using Flexsim HC a respondent needs to click various tabs to check for control-flow structures in the patient tracks, while for BPMN this is visible in one view.

H2: Number of correct answers to control-flow questions of Flexsim HC models are lower than BPMN models

This hypothesis states that the answers related to the Flexsim HC models produce more errors than the answers related to the BPMN model.

H3: BPMN is likely to be adopted over Flexsim HC as a method to verify control-flow aspects

The Moody Method Evaluation model was used to determine whether the Flexsim HC users prefer using BPMN to answer questions related to control-flow aspects over Flexsim HC tracks. The actual efficacy of using BPMN over Flexsim HC is determined by hypotheses 1 and 2. The Moody constructs (2003) are used to determine whether BPMN would likely to be adopted in practice when available as a feature in Flexsim HC.

8.3 Experiment design

To investigate the hypotheses an online experiment was setup specifically for this project. In Appendix S a screenshot of the experiment is given. Participants were guided through 4 steps:

1) Demographics: age, education, familiarity with process models and Flexsim HC Models
2) Questions related to Flexsim HC about repetition, concurrency, splits and sequencing
3) Questions related to BPMN model about repetition, concurrency, splits and sequencing
4) Questions from the Method Evaluation model of Moody (2003)
    a. Perceived Ease Of Use (PEOU)
    b. Perceived Usefulness (PU)
    c. Intention To Use (ITU)

For group 1 the second part consisted of a Flexsim HC track and for the third part a BPMN model. For group 2 this was vice versa (Table 8). The participants were assigned a group when starting the experiment such that the results would balance the number of participants in each group.

8.4 Experiment results

In total 8 participants participated in the study. The average age in the result set was 27.6 (σ=3.5) of which 87.5% male. All participants were high educated (university degree). The average working year experience with Flexsim HC was 2.5 (σ=1.69). Respondents scored themselves on 3.75/5 on familiarity with process modelling and 3.86/5 on Flexsim HC.

First the data set was explored on outliers and missing values. No missing values were found in the data set. Respondents could abort the experiment at any time by closing the browser. Only the results of complete questionnaires were used. The response times (time necessary to answer a question) are plotted in a boxplot to identify outliers (Appendix T). Although there seem to be some large response times for the Flexsim HC, these are not really unrealistic and therefore not deleted. Table 9 shows a summary of the results. The rating for difficulty of the model seems to depend on the kind of model (BPMN vs. Flexsim HC). Both groups rate the Flexsim HC track systematically as more difficult. The results also show that there are more errors for the Flexsim HC track than the BPMN models, 13 errors in Flexsim HC vs. 8 in BPMN. Figure 23 and Figure 24 show the results of the response time and errors graphically. The average response time for BPMN is always lower except for question 4 of model 1 (M1Q4). The figures also show the cumulative number of errors for each question.

Table 9: Summary results

		BPMN	Flexsim HC
Group 1	Response time	10.18 (σ=6.61)	43.27 (σ=55.33)
	Errors	4	8
	Rating difficulty	2 (σ=0)	3.5 (σ=.98)
Group 2	Response time	16.22 (σ=15.67)	64.31 (σ=79.07)
	Errors	4	5
	Rating difficulty	2.75 (σ=.75)	4 (σ=0)
Total	Response time	13.20 (σ=12.29)	53.69 (σ=68.20)
	Errors	8	13
	Rating difficulty	2.37 (σ=.87)	3.75 (σ=.85)

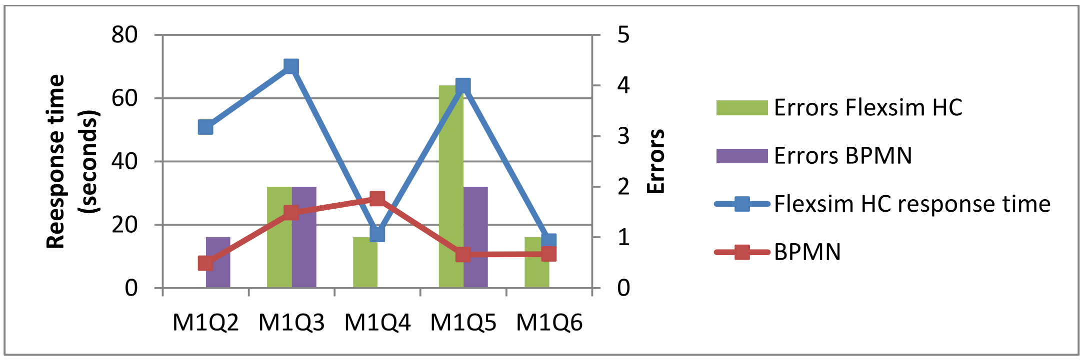


Figure 23: Results for Model 1

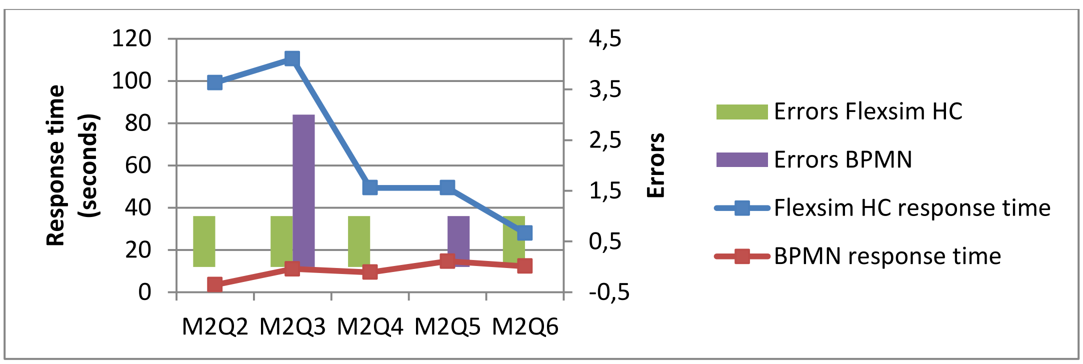


Figure 24: Results for Model 2

8.4.1 Response times

In order to test the hypothesis whether Flexsim HC response times are higher, the time necessary to respond to a question compared to BPMN is tested. There are a number of different statistic tests to compare two samples. Although the independent t-test seems to be the right statistic test, not all the assumptions of the t-test are met:

1) Independence of observation
2) Normality: the t-test assumes that the samples are normally distributed

3) Homogeneity of variance: variances of samples assumed to be equal

The first assumption is met, given the fact that the respondents only review one BPMN model and one Flexsim HC track. The normality assumption is violated (Appendix T, Table 26). The Kolmogorov-Smirnova test is significant ($p<.05$) which indicates that the response times are not normally distributed. Also the assumption of homogeneity of variance is violated, given the fact that the Levene's test is significant ($p<.05$, Appendix T Table 27). Since both the normality and homogeneity of variance assumptions are violated a non-parametric should be used. The Mann-Whitney U test is a widely used statistic test to compare two samples for non-parametric data. Since the Mann-Whitney U test is significant ($p<.05$) the null hypothesis that the means are equal is rejected (Appendix T, Table 30). The data positively supports the hypothesis that Flexsim HC users spent more time on verifying statements about Flexsim HC models than BPMN. The output of statistics tests using SPSS can be found in Appendix T.

8.4.2 Understanding accuracy

The null hypothesis that Flexsim HC models are more prone to errors in answers relating to control-flow behaviour is rejected. The results show that 14 of 48 incorrect answers were given related to the Flexsim HC tracks and 9 of 48 incorrect answers to the BPMN model (Table 9). To test whether there is a statistical difference in errors the sum of incorrect answers per question per model is compared. Since the data is non-parametric the Mann-Whitney U test is used to test whether the Flexsim HC models produce significantly more errors. The Mann-Whitney U test is insignificant (Appendix T, Table 33). The test shows no significant difference in errors between Flexsim HC and BPMN models. Again the results of the statistics tests using SPSS are included in Appendix T.

8.4.3 Likelihood of adoption

To determine the likelihood of adoption of BPMN over Flexsim HC, the constructs of Moody (perceived-ease-of-use, usefulness, intention-to-use) are tested on significant difference compared to the baseline value of 3 (a 5-point Likert scale was used). If significantly different it will likely lead to the adoption of using BPMN if available for simulation model validation against a conceptual model (Figure 3). The following section first assesses the reliability of responses before comparing the responses in a one sample test.

Reliability

Reliability is checked to ensure consistency among responses for the three constructs. According to literature values for Cronbach's alpha above .7 are generally accepted as reliable (Moody, 2003; Field, 2005). Because the inter-item correlation of responses PEOU6, PU1 and PU2 related to their constructs were low they were excluded (Appendix N, Table 34, Table 35, Table 36). This resulted in satisfactory values for the Cronbach alpha as a measure for the consistency among the items (Table 10).

Table 10: Reliability of constructs

Construct	Cronbach Alpha (before deletion)	Cronbach Alpha
Perceived ease of Use	0.59	0.76
Perceived usefulness	0.147	0.70
Intention to use	0.86	0.86

Constructs

To test whether the constructs are significantly different to the baseline score of 3 a one sample test is applied. For parametric data the one sample t-test can be applied if the data is normally distributed, if not a non-parametric test should be used. The results show that for all the constructs the responses are not normally distributed ($p<.05$). Although the one-sample t-test is pretty robust against non-normal data a non-parametric test will also be used to determine whether the responses are significantly different compared to the baseline score of 3 (5-point Likert scale). All constructs showed significantly different means for the t-test ($p<.05$). The non-parametric test also showed that there is evidence to support hypothesis 3, because all the tests were significant ($p<.05$). The data supports hypothesis 3.

Table 11: Significance of one sample tests Moody constructs

Construct	Mean - StDev	One sample t-test	Significance non-parametric
Perceived ease of Use	3.73 (stdev=0.98)	.000	.000
Perceived usefulness	3.89 (stdev=0.64)	.000	.000
Intention to use	3.93 (stdev=0.99)	.002	.008

8.5 Overview experiment

The outcomes of the experiment showed promising results. Although the number of participants was very low (N=8) a significant difference in response times was found. The results also showed that questions related to the control-flow aspects (repetition, parallel behaviour, splits) yielded more errors when Flexsim HC models were used, but this statement was not supported by a significant difference, therefore hypothesis 2 was rejected. The questions based on the Method Evaluation Model of Moody (2003) showed that the participants prefer BPMN over Flexsim HC when they are asked about repetition, parallel behaviour and splits. The average response on the difficulty of the models showed that the participants found the Flexsim HC models more difficult. This indicates that having control-flow aspects in a patient track of Flexsim HC increases the need for control-flow views. The outcomes provide some promising results regarding the usage of process views in Flexsim HC, but given the low number of participants further research is necessary to justify generalizing the conclusions.

9. Conclusion, Discussion and future research

The main research of this master thesis project was:

> **Research question:**
> How can simulation models be validated by applying process mining techniques on simulation logs?

To provide an answer to the research question this document was setup with the structure and steps of the regulative cycle of Van Strien (1997). In the first phase the problem was outlined which provided an answer to research question 1: *What are the problems with model validation in Flexsim HC?* Flexsim HC does not provide a process view of the simulation model. Therefore there is no relation to the conceptual model of the desired system (Figure 3) which makes validation time consuming. In addition Flexsim HC's performance dashboard provide insufficient functionality to assess specific activity performance accuracy.

In the second phase of the regulative cycle the analysis was performed. In the analysis the output of Flexsim HC was analysed and reverse engineered into a UML class model. In addition, the KPIs of interest and objectives for the MUMC+ were outlined The in Chapter 5.1.2. In the last part of the analysis/diagnosis phase different process mining approaches were explored and evaluated to answer research question 2: *What is the output of Flexsim HC and what process mining techniques can be applied?* The output class model is depicted in Figure 11. For the control-flow perspective BPMN was chosen as the graphical notation given its easy to understand graphical representation (White, 2003), support of control-flow elements (Appendix M) and formal specification (OMG, 2011). To mine the BPMN models the heuristics miner in ProM6 was used.

In the solution design phase the solution for the problem was proposed and thereby provided an answer to research question 4: *How can simulation output results be implemented in control-flow diagrams using time-and organisational perspective as overlays?* The solution for the problem was engineered into the newly created IPV tool which makes it possible to generate an integrated process model with extensions of performance and resource data (Figure 16). This multiple perspective view implements the desired KPIs for MUMC+ and thereby provides a solution to the problem described in Chapter 4.

In the intervention phase of the regulative cycle a solution is used in practice. This was done by applying the IPV tool in a simulation case study at the cataract centre of the MUMC+. The integrated view was used to validate the simulation models but also to compare different what-if scenarios. The IPV tool assisted in the discovery of errors in the model (Figure 21). This answered research question 5: *How can an integrated process view assist in simulation model validation?*

In the last phase of the regulative cycle an evaluation was conducted to answer research question 6: *Are control-flows important for simulation model validation?* The evaluation in this study was a small experiment to identify the importance of control-flow in the understandability of simulation models. BPMN and Flexsim HC were compared on the speed of responses and incorrect answers given to control-flow questions about repetition, sequences, concurrency and splits. The results of the experiment supported the hypothesis that responses related to Flexsim HC tracks are longer than those to BPMN ($p<.05$, Appendix T Table 30). The experiment also supported the hypothesis that BPMN models are likely to be adopted when available as a method for verification of models when available ($p<.05$, Appendix T, Figure 51). No support was found for the hypothesis that Flexsim HC tracks would produce more incorrect answers (Appendix T, Table 33). Given the small scale of the experiment more research is necessary to draw justified conclusions. The small experiment provided an answer to the last research question.

9.1 Practical and theoretical contributions

This research project provided some practical and scientific contributions. To start with, the IPV tool for validation of simulation models in Flexsim HC provides a practical solution for the lack conceptual modelling in Flexsim HC. It provides an integrated process view which supports white-box, black-box validation of Flexsim HC simulation models by implementing time- and resource- overlays in a BPMN process model. The IPV tool is useful for the process owner to assess the validity of the simulation model on both control-flow logic and performance accuracy.

This research also showed a new approach to implement multiple process mining perspectives into one view using a more dynamic approach with mouse rollovers and tabs. The thesis provides a very detailed description of how the multiple process mining perspectives are integrated into one view. This can be seen as a valuable theoretical contribution.

Another theoretical contribution is the experiment which showed that the control-flow perspective is an important feature in simulation modelling when models are verified on correctness. A BPMN model is preferred over dialog boxes and graphical user interfaces with tabs where the process is defined behind this tabs. In addition the results showed that having a BPMN model decreases the response time of questions about the understandability of control-flow compared to Flexsim HC models.

9.2 Limitations

The most important limitation of this research project is the fact that the tool supports Flexsim HC data only. The overlays are developed based on the data model of Flexsim HC. Applying the IPV tool on other data models requires programming another import plugin which at least imports the activities with process times, associated resources and waiting times to support the majority in current set of KPIs. In addition the case study was limited to patient oriented processes where Flexsim HC also supports goods flow processes.

For validation purposes the tool would be better if it also supports implementation of real life data in addition to comparison two sets of Flexsim HC output. This could be done using two ways. When aggregate measures are known for the real system this could be implemented by manually registering KPI's to activities or by mining a real event logs (Figure 3). It is then possible to address the accuracy compared to the real system in one view which is really important for end-users to gain credibility in the simulation model. Another possibility is to mine from real system event logs and integrate these results in the IPV tool. Currently the IPV tool only support comparison of two simulation logs (Figure 3).

Another important limitation of this project is the small scale of the experiment. Since the number of participants was really low it is not justified to generalize the conclusions of the results. A larger sample would be required to enhance the validity of the conclusions. In addition, the experiment was conducted using an online survey. It was therefore impossible to check whether the participants did not interrupt the experiment which would result in invalid measurements of response times. The experiment did also not include the integrated view developed in this project. This was deliberately chosen such that the overlays did not influence the responses on understandability of control-flow aspects. Including the integrated view in the experiment requires more questions, an in depth interviews or open questions to assess the understandability of the overlays and how well the integrated visual presentation is understood.

9.3 Future research

Performance and organisational overlays in process mining is still an area that requires more research. This research shows an approach in integrating and extending a control-flow model with multiple process mining perspectives. This model is more dynamic and provides more information, interactively with the user. Concerning the visual approach more research is necessary on how to incorporate the information in a way such that the process model is still very compact and understandable and how to implement this dynamic approach in other BPMLs.

Another future research topic could focus on how to apply the integrated view on real life data. For example: how to integrate real system data in the models to provide a comparison between simulation results and the real system. When future research focuses on extracting a richer data set from event logs it could extract more KPIs, for example the net-waiting time when more lifecycle elements are supported such as the lifecycle scheduled. Future research could further define a method to define the extraction of KPIs on a higher level such that there's no need for low-level programming to calculate the KPIs from event logs. A combination of richer event-logs and high level definition of KPIs calculation would provide the possibility to use the solution on other data models.

The experiment in this project was solely focused on the importance of control-flow views for simulation model validation. Another experiment setup could investigate the influence of the overlays as proposed in this project on the understandability of analysing control-flow models, i.e. do the overlays influence the understandability of the control-flow? Since the experiment was also very small scale, more experiments are necessary to justify the conclusions of the importance of control-flow for simulation model validation. Adding other software packages to the experiment could also provide a better basis to generalize the results.

References

Aalst, W. v. (2010a). Business process simulation revisited. *Enterprise and Organizational Modeling and Simulation*, 1-14.

Aalst, W. v. (2010b). *Process Mining.* Eindhoven: Springer.

Balkom, H. v. (2013). *Health Care Business Process Redesign: A method for the structured generation of redesign scenarios.* Technical University Eindhoven - Series Master Theses Operations Management and Logistics.

Bayraktar, I. (2011). *The Business Value of Process Mining.* Eindhoven: Technical University Eindhoven.

Camunda.org. (n.d.). *Open Source BPM and Workflow with BPMN 2.0*. Retrieved 1 3, 2014, from http://www.camunda.org/

Cayirli, T., & Veral, E. (2003). Outpatient scheduling in health care. *Production and operations management*, 519-549.

Centraal Bureau voor de Statistiek (CBS). (2013). *Gezondheid en zorg in cijfers.* CBS.

Christian W. Günther, E. V. (n.d.). *XES standard definition.* Retrieved from http://www.xes-standard.org/

Dat, P. (2012). *Business process simulation in healthcare: the applicability of general-purpose and application-oriented simulation software.* Technical University Eindhoven.

Farnum, L., & Stanton, W. (1987). Some Results Concerning the Estimation of Beta Distribution Parameters in PERT. *he Journal of the Operational Research Society*, 278-290.

Field, A. (2005). *Discovering Statistics Using SPSS - 2nd edition.* Sage.

Figl, K., Recker, J., & Mendling, J. (2013). Study on the Effects of Routing Symbol Design on Process Model Comprehension. *Support Systems, vol. 54*, 1104-1118.

Fishman, G. (2001). *Discrete-Event Simulation: Modeling, Programming, and Analysis.* Springer.

Gorp, P. v. (2012). *Healthcare Business Networks Lecture Slides.* Technical University Eindhoven.

Huang, X. (1994). Patient Attitude Towards Waiting in an Outpatient Clinic and its Applications. *Health services Management Research vol.7*, 2-8.

Jacobs, M. (2013). An experimental study into the structure of Business Process Redesign techniques for the Healthcare domain. *Technical University Eindhoven - Series Master Theses Operations Management and Logistics*.

Jacobson, S., Hall, S., & Swisher, J. (2006). Discrete-event simulation of health care systems. In *Patient flow: reducing delay in healthcare delivery* (pp. 211-252). Springer US.

Jahangirian, M., Naseer, A., Stergioulas, L., Young, T., Eldabi, T., Brailsford, S., et al. (2012). Simulation in health-care: lessons from other sectors. *Operational Research vol12 issue 1* , 45-55.

Jansen-Vullers, M., & Netjes, M. (2006). Business process simulation–a tool survey. *Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools.* Aarhus, Denmark.

Jun, J., Jacobson, S., & Swisher, J. (1999). Application of Discrete-Event Simulation in Health Care Clinics: A Survey. *The Journal of the Operational Research Society vol. 50 issue 2*, 109-123.

Kellner, M., Madachy, R., & Raffo, D. (1999). Software process simulation modeling: Why? What? How? *The Journal of Systems and Software vol 46*, 91-105.

Laue, R., & Gadatsch, A. (2011). Measuring the Understandability of Business Process Models - Are We Asking the Right Questions? *Business Process Management Workshops*, (pp. 37-48).

Law, A. (2007). *Simulation Modeling and Analysis. 4th edition.*

Law, A. (2008). How to build valid and credible simulation models. *40th Conference on Winter Simulation. Winter Simulation Conference*, (pp. 39-47).

Lowery, J., Hakes, B., Keller, L., Lilegdon, W., Mabrouk, K., & McGuire, F. (1994). Barriers to implementing simulation in health care. *Society for Computer Simulation International*.

Martens, A. (2012). *Onderzoek naar de processen in het staarcentrum van het Oogziekenhuis Maastricht UMC.* Technical University Eindhoven - Bachelor Thesis Project.

Melcher, J., Mendling, J., Reijers, H., & Seese, D. (2010). On Measuring the Understandability of Process Models. *Business Process Management Workshops*, (pp. 465-476).

Ministerie van Volksgezondheid Welzijn en Sport. (2013). *Rijksbegroting 2014 XVI Volksgezondheid, Welzijn en Sport.*

Moody, D. (2003). The Method Evaluation Model: A Theoretical Model for Validating Information Systems Design Methods. *ECIS*.

MUMC+. (2012). *Jaarverslag.* Maastricht.

Nikoukaran, J., Hlupic, V., & Paul, R. (1999). A hierarchical framework for evaluating simulation software. *Simulation Practice and Theory 7.3*, 219-231.

OMG. (2011). *Business Process Model and Notation, version 2.0.* Object Management Group.

Overduin, M. (2013). Exploration of the link between the execution of a clinical process and its effectiveness using process mining techniques. *Technical University Eindhoven - Series Master Theses Operations Management and Logistics*.

Paul, R., Hlupic, V., & Giaglis, G. (1998). Simulation Modelling of Business Processes. *3rd UK Academy of Information Systems Conference* (pp. 311-320). McGraw-Hill.

Recker, J., Safrudin, N., & Rosemann, M. (2012). How novices design business processes. *Information Systems*, 557-573.

Reijers, H., & Mansar, S. (2006). Best practices in business process redesign:an overview and qualitative evaluation of successful redesign heuristics. *Omega 33*, 283-306.

Robinson, S. (1997). Simulation model verification and validation: increasing the users' confidence. *Proceedings of the 1997 Winter Simulation Conference.*, (pp. 53-59).

Rozinat, A. (2010). *Process Mining: Conformance and Extension.* Technical University Eindhoven.

Ryan, J., & Heavey, C. (2006). Process modeling for simulation. *Computers in Industry*, 437-450.

Sargent, R. (2005). Verification and validation of simulation models. *Proceedings of the 2005 Winter Simulation Conference*, (pp. 130-143).

Shiels, A., Bennett, T., & Hejtmancik, F. (2010). Cat-Map: putting cataract on the map. *Molecular Vision vol. 16*, 2007-2015.

Van Strien, P. (1997). Towards methodology of psychological practice. *Theory and Psychology*, 683-700.

Weijters, A., Van der Aalst, W., & De Medeiros, A. (2006). Process Mining with the Heuristics Miner. *Technische Universiteit Eindhoven, Tech. Rep. WP 16*.

White, S. (2003). *XPDL and BPMN.* The Object Management Group - http://www.omg.org/bpmn/Documents/XPDL_BPMN.pdf.

White, S. (2005). *Using BPMN to Model a BPEL Process.* BPTrends.

Wohed, P., Aalst, W. v., Dumas, M., Hofstede, A. t., & Russell, N. (2006). On the Suitability of BPMN for Business Process Modelling. *4th International Conference, BPM 2006* (pp. 161-176). Vienna, Austria: Springer Berlin Heidelberg.

Abbreviations

BPM: Business Process Management

BPML: Business process modelling language

BPMN: Business Process Model and Notation

BPS: Business Process Simulation

DES: Discrete Event Simulation

EPC: Event driven Process Chains

Flexsim HC: Flexsim Healthcare

ITU: Intention To Use

IPV-tool: Integrated Process View tool

JSON: JavaScript Object Notation

KPI: Key Performance Indicator

MUMC+: Maastricht University Medical Center

PEOU: Perceived Ease Of Use

PU: Perceived usefulness

ProM6: Process Mining framework version 6

SVG: Scalable Vector Graphix

UML: Unified Modelling Language

Appendix A: Process models

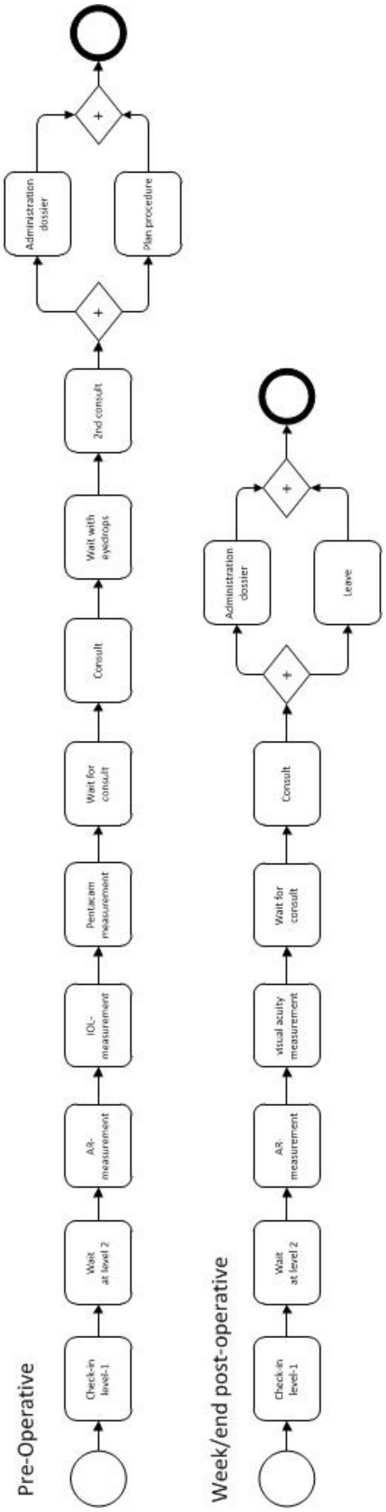


Figure 25: Processes for pre-operative and post-operative patients

Appendix B: Legacy process description

The cataract centre in the legacy situation only operated on Monday, with a throughput of circa 90 patients. Three different types of appointments are scheduled: **new patients**, **post-operative week control** and **post-operative end control**. Note that all the activities described in the following sections take place on the same day.

Pre-operative: New patient

First the patient is required to fill out an anaesthesia form before they enter the appointment. While patients fill out their forms a physician prepares for some measurements. The first measurement the patient undertakes is the Auto Refractor (AR) measurement with the objective to measure the patients' refractive error. This measurement is taken by technical ophthalmic assistant (TOA). When the measurement is taken the patient enters the waiting room and waits for two other measurements. The second and third are the Pentacam and the intraocular lens (IOL) measurements, which can be executed in any order and in the legacy process only by the optometrist. After the measurements the patient waits for the doctor to receive a consult. During this consult the patient receives eye drops to widen the eyes. Since this takes about 15 minutes the patient waits in the waiting room while the doctor performs another consult. After the 15 minutes the doctor is able to look better into the eye. After the second consult the patient can schedule for the surgery and the post-operative controls.

Post-operative week control

The post-operative week control is scheduled the week after the cataract surgery. First the patient receives an AR-measurement and the vision is registered. After the measurements the patient waits for the consult with the doctor. After the consult the patient leaves the clinic or receives eye drops to widen the eyes and waits for a second consult when the patient requires surgery on the other eye.

Post-operative end control

The last control is 4 weeks after the operation. The structure of this visit is the same as the week control: the patient's vision is registered and receives a consult.

Scheduling

During the Monday at cataract centre in the legacy process circa 30 new patients were planned and 60 post-operative controls.

Resources

During the cataract centre the following resources are present:

- 4 doctors
- 2 technical ophthalmic assistant (TOA)
- 1 optometrist
- 2 nurses

Based on the process descriptions of the legacy processes, the following business process redesigns were identified and implemented into the simulation models originally developed by Van Balkom (Balkom, 2013):

- Task elimination: Patients no longer go to the desk on level 2 of the cataract centre, post-operative patients no longer take IOL- and Pentacam measurements.
- Task composition: the technical eye assistant performs all the standard measurements whereas in the legacy model the optometrist was required for IOL- and Pentacam measurements.
- Planning heuristics: Where the legacy scheduled circa 30 new patients and 60 post-operative control patients, the as-is planning is spread across 2 days (one full day, two afternoons). The as-is appointment system is also changed. In the legacy system the time slots were grouped by appointment type, e.g. first al week control patients followed by all pre-operative patients. In the as-is model the appointment system uses rotational design (see Appendix G)
- Resource allocation: because the planning changed also fewer resources were required.

Appendix C: Simulation models

Flexsim HC version: 4.02

In Figure 26 a screenshot of a 3-dimensional Flexsim HC simulation model is depicted.

Figure 26: Flexsim 3D view of cataract legacy process

In Figure 27 the flow-chart view of Flexsim HC. This view only depicts the flow between different processing and waiting locations in the model. Activities are not visible.

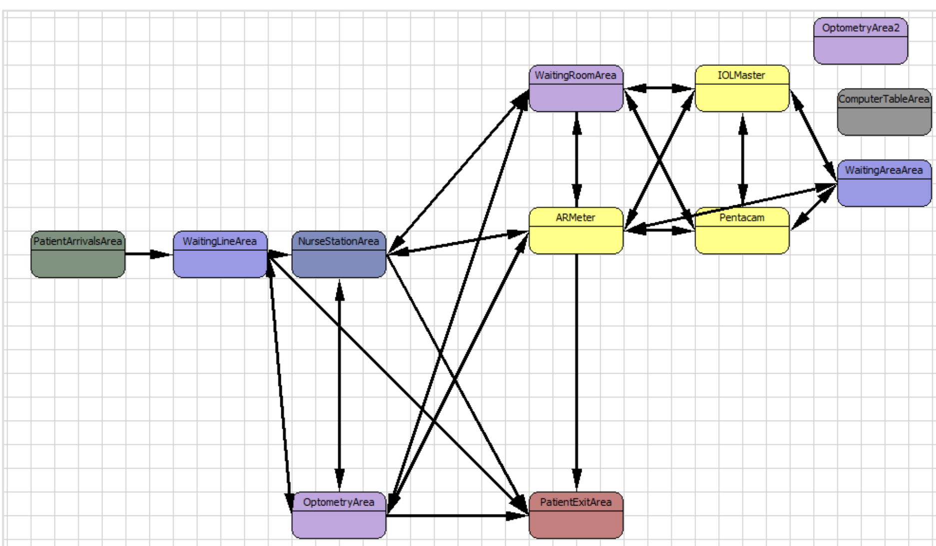


Figure 27: Flowchart view of cataract legacy process, displaying patient processing locations

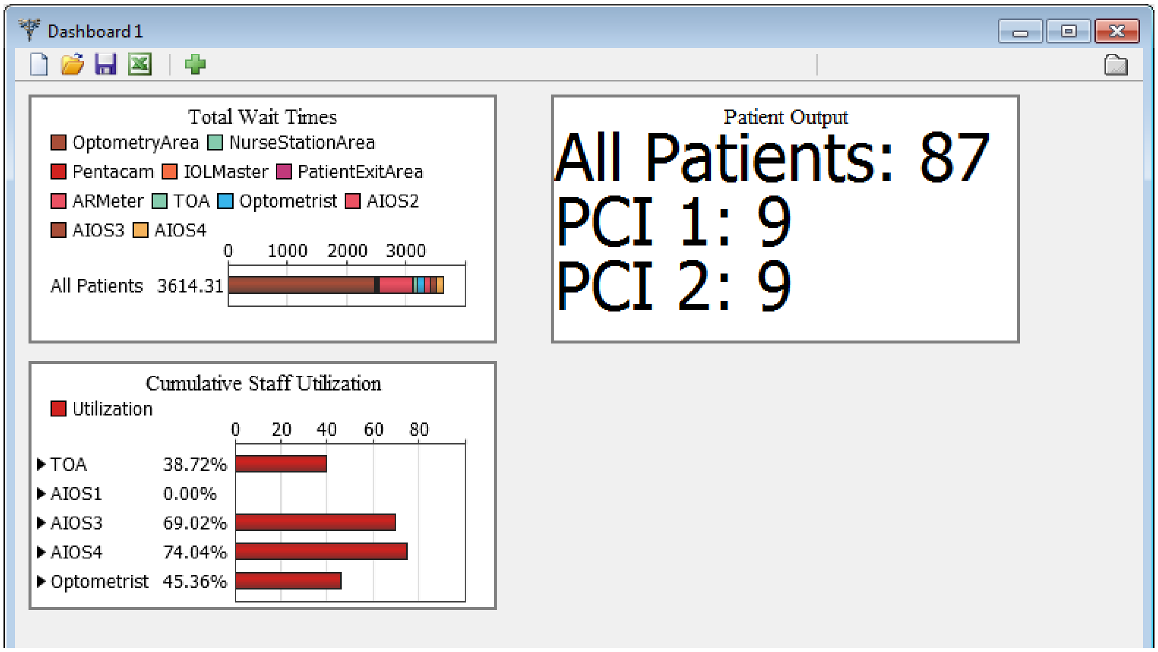


Figure 28: Dashboard view Flexsim HC

Appendix D: Flexsim HC Patient Tracks

In Figure 29 a patient track is depicted for pre-operation consult for the legacy process. The patient flow is determined by parameters set in this graphical user interface. Figure 30 shows how resources are allocated to an activity.

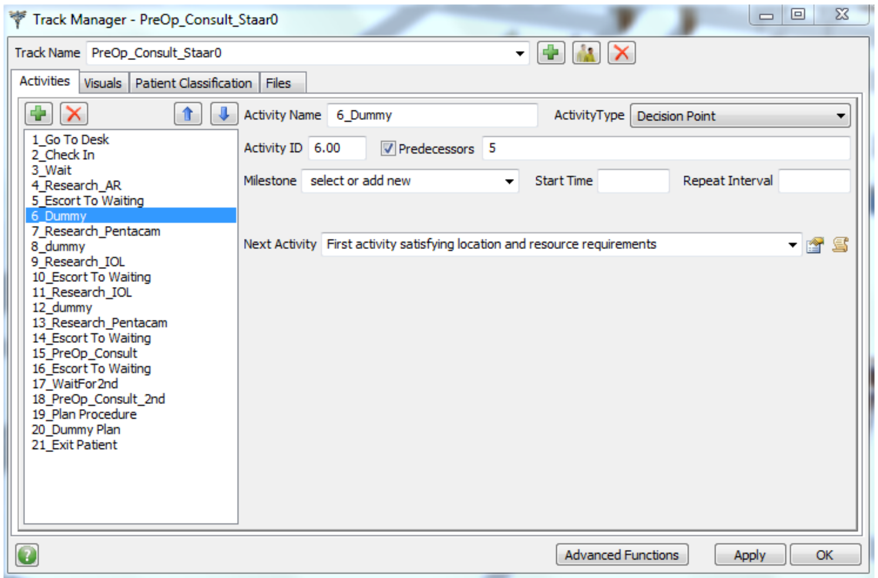


Figure 29: Patient track

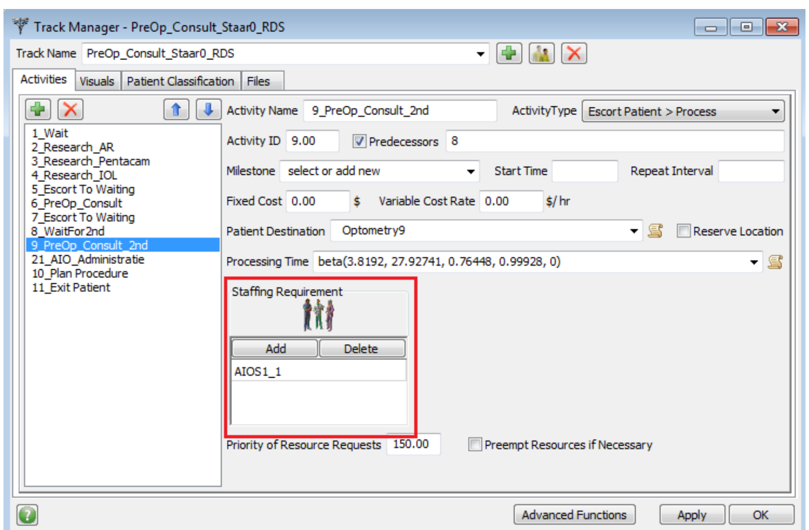


Figure 30: Allocation of resources to activity

In Figure 31 an exclusive OR split is defined in an example patient track. The process model in Figure 32 is a BPMN representation of the patient track of Figure 31. An example of an AND split in Flexsim HC is defined by using multiple predecessors, which is shown in Figure 33 and Figure 34.

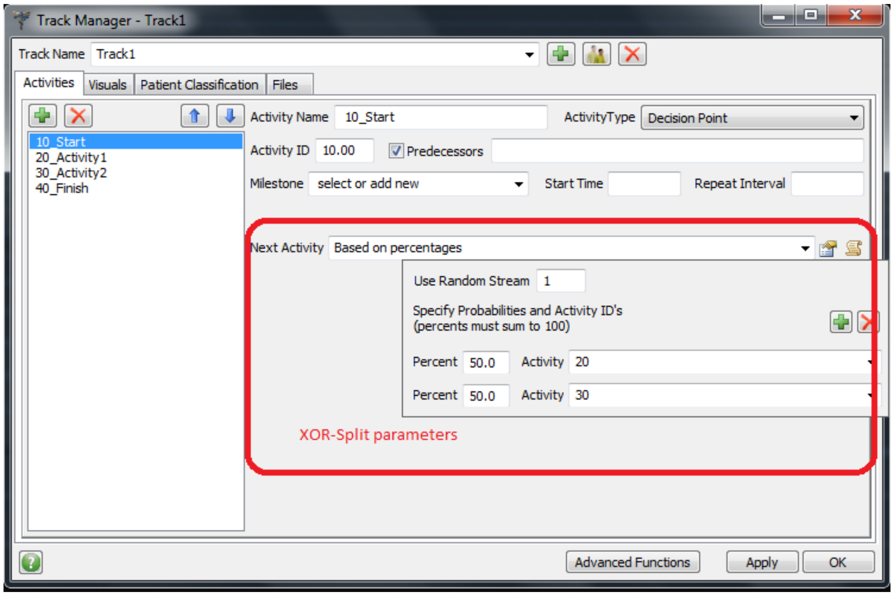


Figure 31: XOR Split in Flexsim HC Patient Track

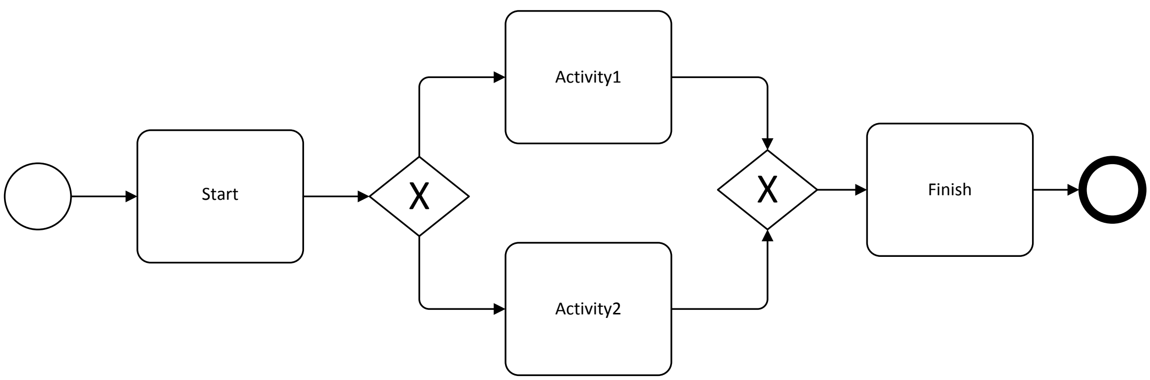


Figure 32: XOR split in BPMN representing patient track given in Figure 31

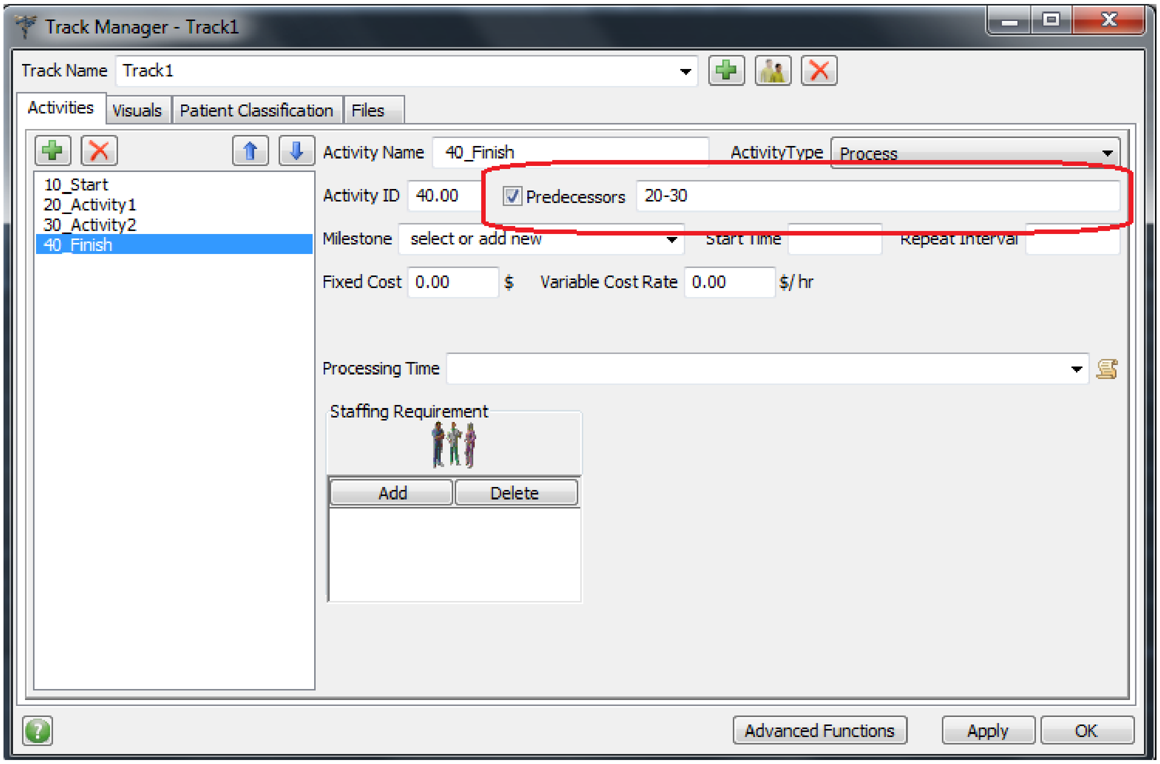


Figure 33: Parallel activities are specified by using multiple predecessors

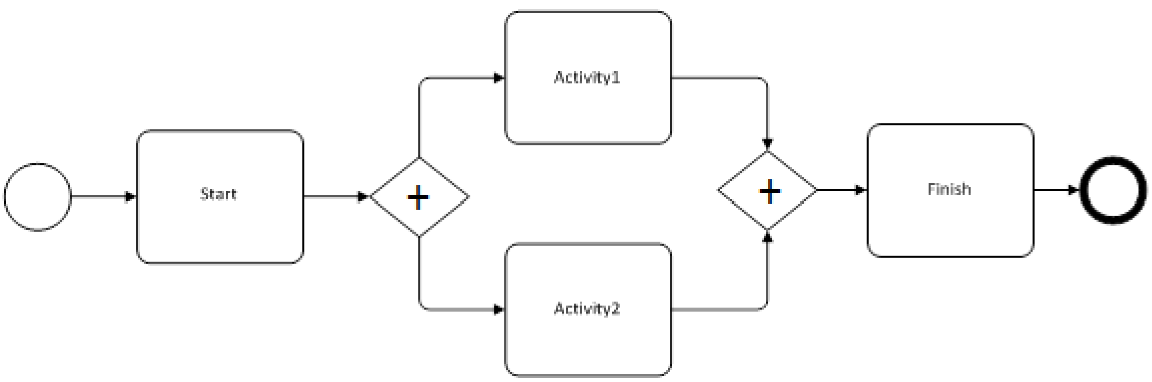


Figure 34: AND split in BPMN representing patient track given in Figure 33

Appendix E: Flexsim HC output files

The following table represents data from the main output file in Flexsim HC. The fields numbers 14 until 39 are repeated for each activity that belongs to the PatientID.

Staff State History

The staff state history file contains for every minute the state for a staff member. In the settings for the output (Appendix I) the time between every state can be set. The records are unrelated to other data files, i.e. the state records are not related to a patient or location. The records only represent the state for each staff member. Based on this data the utilization of staff resources can be calculated. The utilization can be calculated at any moment recorded in the state history.

Table 12: KPI for staff state history

Performance indicator	Dimension	Description
WaitingForTaskTime	Cumulative	Evolvement of WaitingTime over time
ProcessingTime	Cumulative	Evolvement of ProcessingTime over time
InTransitTime	Cumulative	Evolvement of InTransitTime over time. Represents the time
Distance traveled	cumulative	Travel distance for staff over time

Patient Processing Locations State History

The patient processing locations state history contains for every minute and every processing location a record with the queue length, throughput at that moment, processing time and other location related information. The patient processing location is related to a *ResourceDestination* in the patient history log.

Table 13: KPI for Patient Processing locations

Performance indicator	Dimension	Description
WaitingTime	Average,Maximum	Evolvement of WaitingTime over time
ProcessingTime	Average,Maximum	Evolvement of ProcessingTime over time
Throughput	cumulative	Patient throughput cumulative over time
Queue	Current, average, maximum	Queue length information over time

Patient Queuing Locations State History

Queuing locations are also related to the *ResourceDestinations* of the patient history log, and to the names in the flowchart depicted in Appendix I. For the queuing locations the data recorded in the logs are very similar to the processing locations state history (Table 15), except for some measures like *HoldingForPatientTime* and maintenance related times because they do not apply to a queuing location.

Table 14: Flexsim output data: Patient history log

Field No.	Field	Description
1	ClockTime	Time of entry in process
2	PatientID	Unique identifier for process instance
3	PCI	Patient track id
4	Track	Patient tracks
5	Acuity	
6	DistanceTraveled	Total distance travelled in run?
7	S_ReceivingIndirectCare	Cumulative time u
8	S_ReceivingDirectCare	Sum time until receiving direct care?
9	S_WaitingForTransport	Cumulative time waiting for transport
10	S_WaitingForRoom	Cumulative time waiting for room
11	S_InTransit	Cumulative time in transit
12	S_WaitingForStaff	Cumulative time waiting for staff
13	S_WaitingForEquipment	Cumulative time waiting for equipement
14	ActivityName_1	Activity name of first activity
15	ActivityType_1	Activity type identification of first activity
16	ActivityFixedCosts_1	Fixed costs for first activity
17	ActivityTotalCosts_1	Total costs for first activity
18	ActivityStartTime_1	Start time of the first activity
19	ActivityFinishTime_1	End time of the first activity
20	ActivityProcessTime_1	Processing time of the first activity
21	ActivityResourceDestination_1	Resource destination for activity 1, relates to a Patient
22	ActivityWaitTime_WaitingRoomArea	Waiting time in WaitingRoomArea for activity 1
23	ActivityWaitTime_OptometryArea	Waiting time in OptometryArea for activity 1
24	ActivityWaitTime_NurseStationArea	Waiting time in NurseStationArea for activity 1
25	ActivityWaitTime_PatientArrivalsArea	Waiting time in PatientArrivalsArea for activity 1
26	ActivityWaitTime_OptometryArea2	Waiting time in OptometryArea2for activity 1
27	ActivityWaitTime_Pentacam	Waiting time in Pentacam for activity 1
28	ActivityWaitTime_IOLMaster	Waiting time in IOLMaster for activity 1
29	ActivityWaitTime_ComputerTableArea	Waiting time in ComputerTableArea for activity 1
30	ActivityWaitTime_PatientExitArea	Waiting time in PatientExitArea for activity 1
31	ActivityWaitTime_ARMeter	Waiting time in ARMeter for activity 1
32	ActivityWaitTime_WaitingLineArea	Waiting time in WaitingLineArea for activity 1
33	ActivityWaitTime_SecondResearch	Waiting time in SecondResearch for activity 1
34	ActivityWaitTime_FirstResearch	Waiting time in FirstResearch for activity 1
35	ActivityWaitTime_Secretary	Waiting time in Secretary for activity 1
36	ActivityWaitTime_AIOS1	Waiting time in AIOS1for activity 1
37	ActivityWaitTime_AIOS2	Waiting time in AIOS2for activity 1
38	ActivityWaitTime_AIOS3	Waiting time in AIOS3for activity 1
39	ActivityWaitTime_AIOS4	Waiting time in AIOS4for activity 1
40	ActivityName_2	Activity name of the second activity
41	ActivityType_2	Activity type identification of second activity
..	…	

..	ActivityWaitTime_AIOS4	Waiting time in AIOS4 for activity 2
	ActivityName_3	Activity name for activity 3
..	..	

Table 15: Flexim output data: Patient Processing Locations

Field No.	Field	Description
1	Time	Time of recording
2	ClockTime	Clock time at recording
3	Name	Name of processing location, refers to
4	Area	Processing location is a member of this area
5	Queue	Queue length at *Time*
6	QueueAverage	Queue average for interval [0,*Time*]
7	QueueMax	Queue max for interval [0,*Time*]
8	StayTimeAvg	Average processing time for interval [0,*Time*]
9	StayTimeMax	Max processing time for interval [0,*Time*]
10	Throughput	Cumulative throughput at *Time*
11	OffScheduleTime	Cumulative off schedule time for interval [0,*Time*]
12	InterruptedTime	Cumulative interrupted time for interval [0,*Time*]
13	OccupiedTime	Cumulative time occupied for interval [0,*Time*]
14	VacantTime	Cumulative vacant time for interval [0,*Time*]
15	HoldingForPatientTime	Cumulative holding for patient time for interval [0,*Time*]
16	MaintenanceTime	Cumulative maintenance time for interval [0,*Time*]
17	WaitingForMaintenanceTime	Cumulative waiting for maintenance time for interval [0,*Time*]

Table 16: Cases

Original field	Original field
1	ClockTime
2	PatientID
3	PCI
4	Track
5	Acuity
6	DistanceTraveled
7	S_ReceivingIndirectCare
8	S_ReceivingDirectCare
9	S_WaitingForTransport
10	S_WaitingForRoom
11	S_InTransit
12	S_WaitingForStaff
13	S_WaitingForEquipment

Table 17: Activities

Original field	Original field	New field name
-	-	ActivityID
2	PatientID	PatientID
14, 40, ..	ActivityName_{X}	ActivityName
15, 41, ..	ActivityType_{X}	ActivityType
16, 42, ..	ActivityFixedCosts_{X}	ActivityFixedCosts
17, 43, ..	ActivityTotalCosts_{X}	ActivityTotalCosts
18, 44, ..	ActivityStartTime_{X}	ActivityStartTime
19, 45, ..	ActivityFinishTime_{X}	ActivityFinishTime
20, 46, ..	ActivityProcessTime_{X}	ActivityProcessTime
21, 47, ..	ActivityResourceDestination _{X}	ActivityResourceDestination

Table 18: WaitingTimes

Original field no	Original field	New field name
-	**-**	WaitingID
-	-	ActivityID
2	PatientID	PatientID
22, 39, ..	Last part of field name	WaitingLocation
22, 39, ..		WaitingTime

Table 19: Patient Queuing Locations

Field No.	Field	Description
1	Time	Time of entry in process
2	ClockTime	Unique identifier for process instance
3	Name	Time of recording
4	Area	Clock time at recording
5	Queue	Queue at *Time*
6	QueueAverage	Average queue at interval [0,*Time*]
7	QueueMax	Max queue at interval [0,*Time*]
8	StayTimeAvg	Average queuing time at interval [0,*Time*]
9	StayTimeMax	Max queuing time at interval [0,*Time*]
10	Throughput	Cumulative throughput at *Time*
11	OffScheduleTime	Cumulative time off schedule at interval [0,*Time*]
12	InterruptedTime	Cumulative interrupted time at interval [0,*Time*]
13	OccupiedTime	Cumulative occupied time at interval [0,*Time*]
14	VacantTime	Cumulative vacant time at interval [0,*Time*]

Table 20: Staff state history

Field No.	Field	Description
1	Clock	Time of entry in process
2	ClockTime	Unique identifier for process instance
3	Name	Staff resource name
4	Group	Staff group name
5	SkillLevel	Skill level of the resource
6	Payrate	Payrate of the resource
7	DistanceTraveled	Cumulative distance travelled at interval [0,*Time*]
8	OffScheduleTime	Cumulative off schedule time at interval [0,*Time*]
9	InterruptedTime	Cumulative interrupted time at interval [0,*Time*]
10	PerformingTaskTime	Cumulative performing task time at interval [0,*Time*]
11	WaitingForTaskTime	Cumulative waiting for task time at interval [0,*Time*]
12	InTransitTime	Cumulative in transit time at interval [0,*Time*]
13	WaitingForStaffTime	Cumulative waiting for staff time at interval [0,*Time*]
14	WaitingForEquipementTime	Cumulative waiting for equipment time at interval [0,*Time*]
15	Break1Time	Cumulative break 1 time at interval [0,*Time*]
16	Break2Time	Cumulative break 2 time at interval [0,*Time*]
17	LunchTime	Cumulative lunch break time at interval [0,*Time*]

Appendix F: Process mining results

In fig x the mined model for the as-is simulation model is given. To mine the model the BPMN heuristics miner available in ProM6 was used, using its default settings.

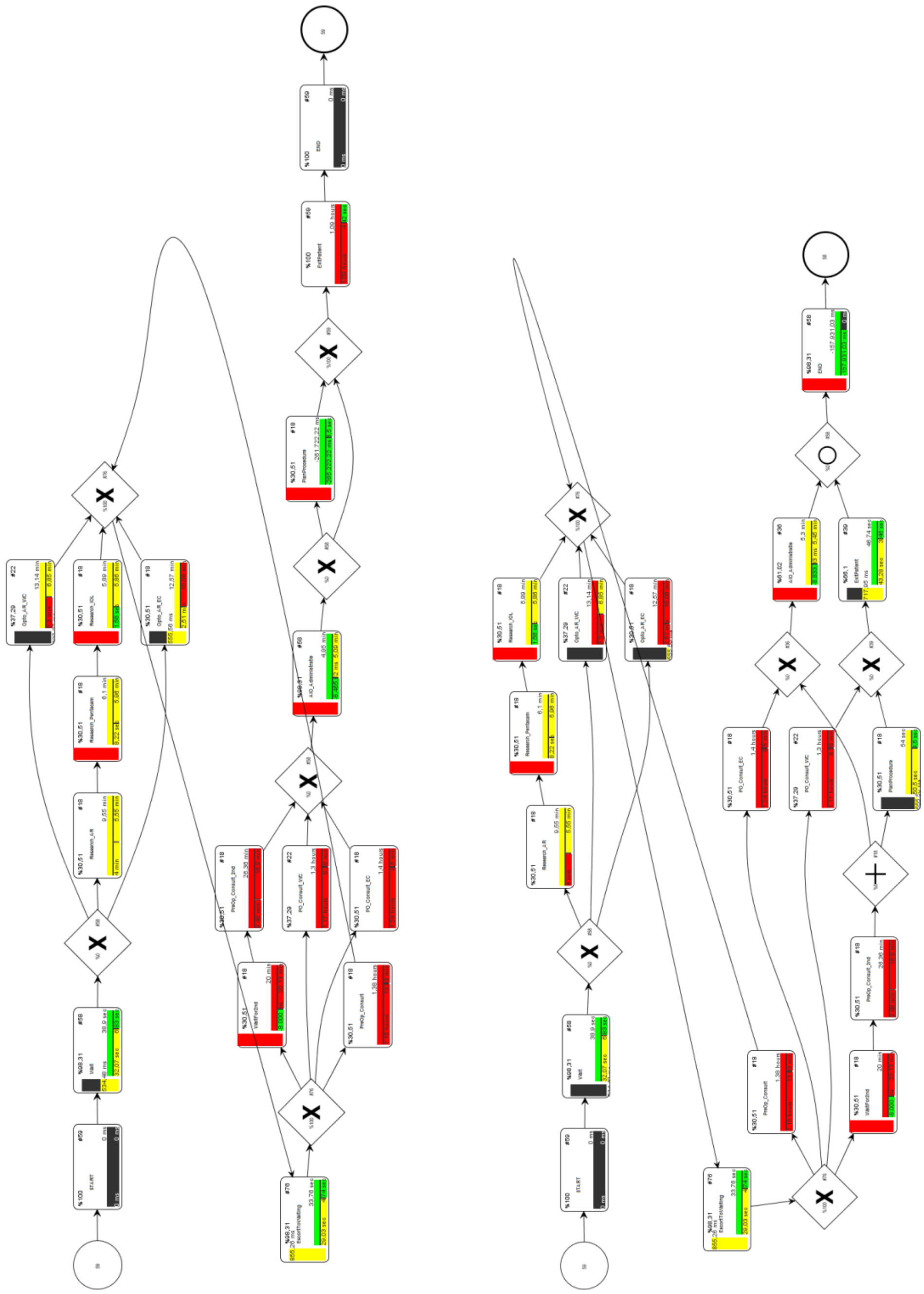


Figure 35: Mined as-is simulation log without order manipulation (left) and with order manipulation (right)

Model with further naming distinguishing tracks to get higher conformance:

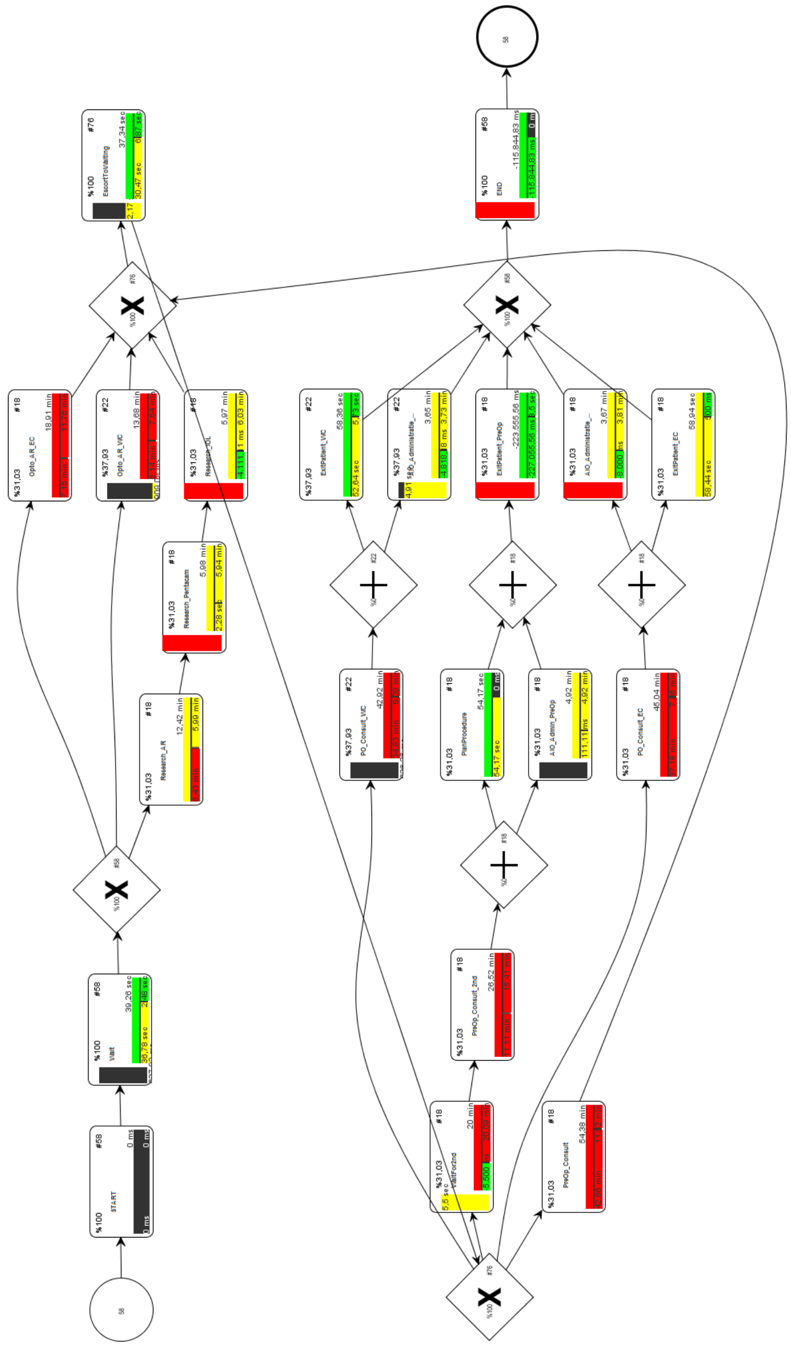


Figure 36: Mined model with highest conformance

Appendix G: Planning example

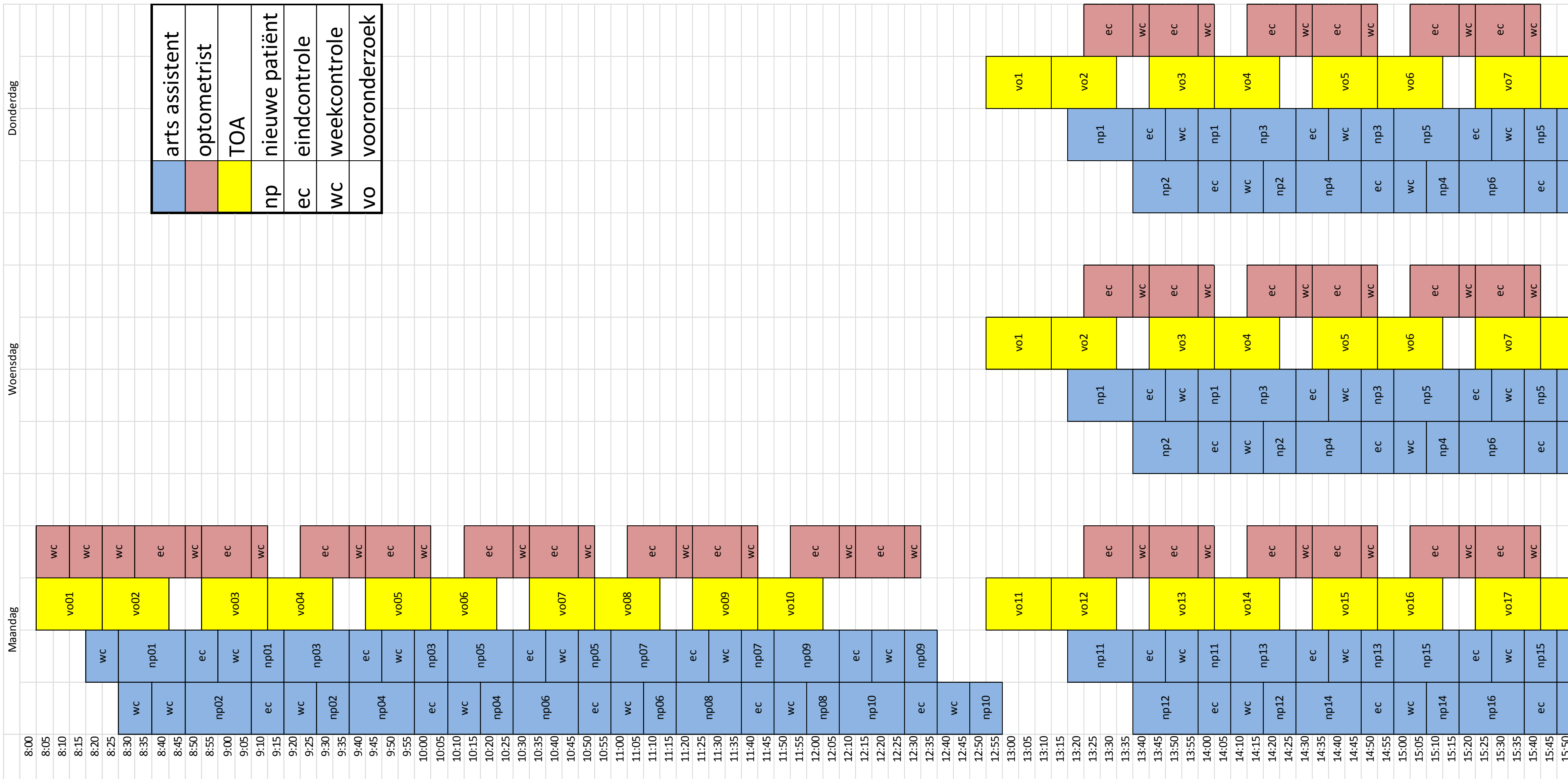


Figure 37: Example planning as-is cataract level 2

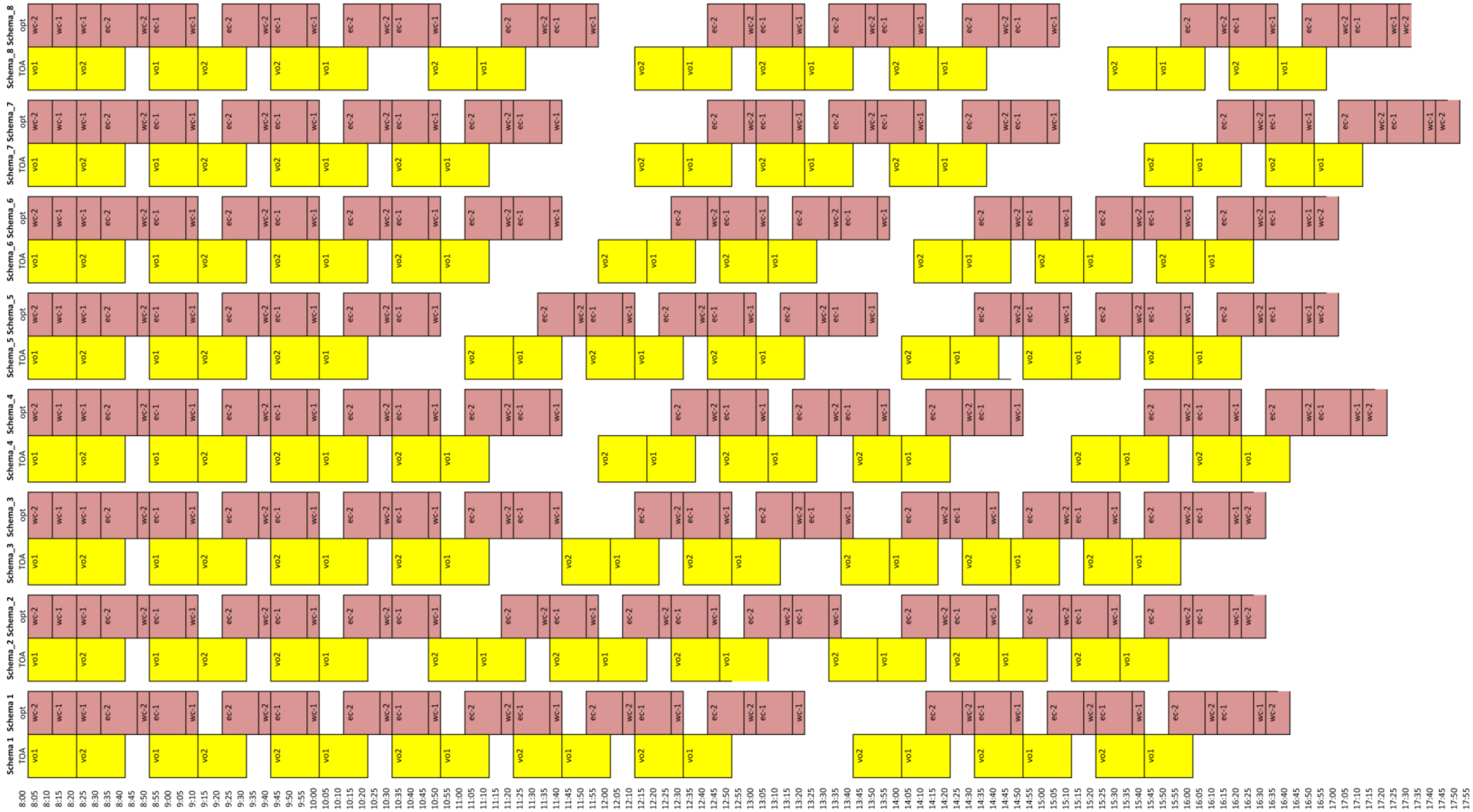


Figure 38: Alternative planning schemas

Appendix H: Floor plan cataract centre

In Figure 39 the floor plan of the cataract centre is depicted. Van Balkom (2013) identified the following sections in the floor plan, which are applicable to both the legacy and as-is situation of the cataract centre:

1) Entrance
2) Pre-consult measurement room
3) Measurement room (AR-, Pentacam-, IOL-measurement)
4) Consult room for doctors or optometrists
5) Waiting room

Van Balkom (2013) based the system layout of the cataract centre on the floor plan given in Figure 39 and the process descriptions given by Martens (2012).

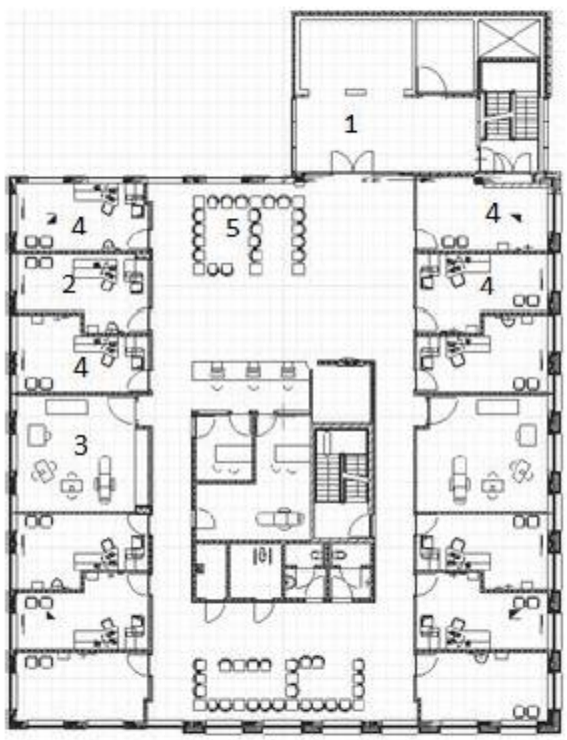


Figure 39: Floor plan cataract centre

Appendix I: Flexsim HC output settings

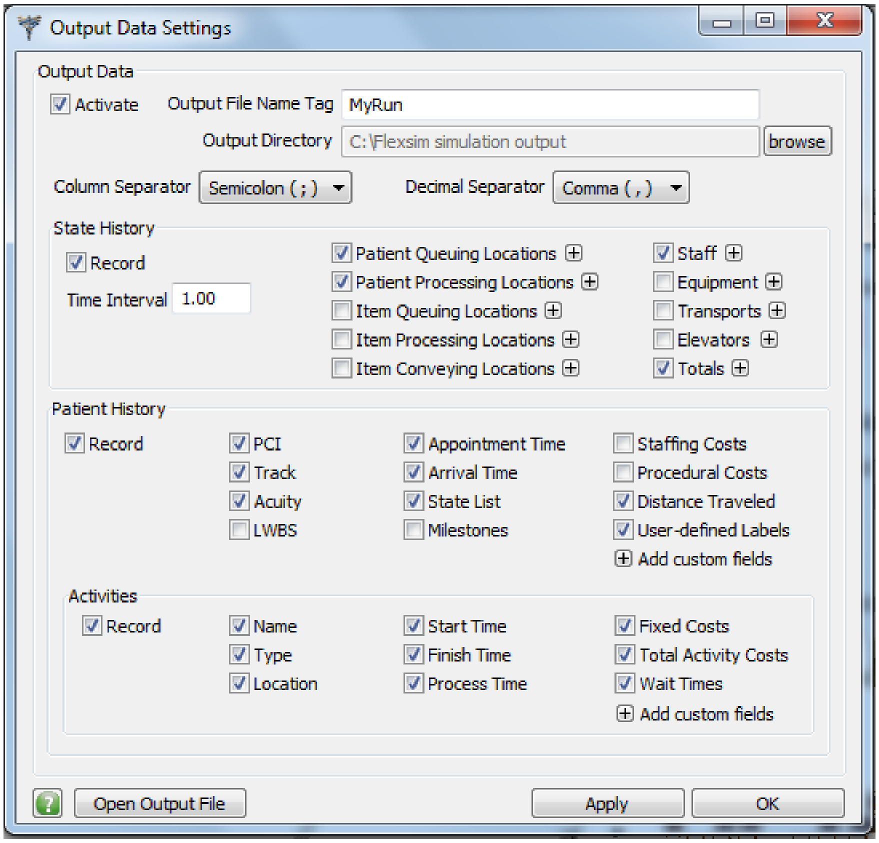


Figure 40: Flexsim output settings

Appendix J: Explicit resource links in Flexim HC

The code below makes explicit resource links in a JSON string. This code needs to be implemented as advanced function for Activity Finished Trigger for every activity. By default there is no explicit link between the activities and resources necessary to execute the activity. This piece of code generates a JavaScript Object (JSON) with a link between activities and resources. Using a JSON syntax allows to use one label to store this multidimensional data type.

```
string Thislabel="ResourceJSON";
string currentlabel=getlabelstr(patient,Thislabel);
if(comparetext(currentlabel,"")){
	currentlabel=concat("{\"",numtostring(activityrow),"\":[");
}else{
	currentlabel=stringcopy(currentlabel,1,stringlen(currentlabel)-1);
	currentlabel=concat(currentlabel,",\"",numtostring(activityrow),"\":[
");
}
int first_el=1;
treenode activitynode = rank( activitytable, activityrow );
for(  int  resource  =  1;  resource  <=  content(rank(activitynode,17));
resource++ )
{
	treenode Resourcenode   = rank(  rank( activitynode, 17 ), resource
);
	treenode NameMode       = rank( Resourcenode, 2 );
	string  ActualResource      =  concat(getname(  tonode(  getnodenum(
NameMode ) ) ),gettablestr(activitynode,activityrow,COL_ActivityType));
	if(first_el==1){
		currentlabel=concat(currentlabel,"\"",ActualResource,"\"");
	}else{
		first_el=0;
		currentlabel=concat(currentlabel,",\"",ActualResource,"\"");
	}

}
currentlabel=concat(currentlabel,"]}");
setlabelstr( patient, Thislabel,  currentlabel );
```

Appendix K: Log fragment OpenXES


```xml
<?xml version="1.0" encoding="UTF-8" ?>
<log xes.version="1.0" xes.features="nested-attributes" openxes.version="1.0RC7" xmlns="http://www.xes-standard.org/">
	<extension name="Lifecycle" prefix="lifecycle" uri="http://www.xes-standard.org/lifecycle.xesext"/>
	<extension name="Organizational" prefix="org" uri="http://www.xes-standard.org/org.xesext"/>
	<extension name="Time" prefix="time" uri="http://www.xes-standard.org/time.xesext"/>
	<extension name="Concept" prefix="concept" uri="http://www.xes-standard.org/concept.xesext"/>
	<extension name="Semantic" prefix="semantic" uri="http://www.xes-standard.org/semantic.xesext"/>
	<global scope="trace">
		<string key="concept:name" value="__INVALID__"/>
	</global>
	<global scope="event">
		<string key="concept:name" value="__INVALID__"/>
		<string key="lifecycle:transition" value="complete"/>
	</global>
	<classifier name="MXML Legacy Classifier" keys="concept:name lifecycle:transition"/>
	<classifier name="Event Name" keys="concept:name"/>
	<classifier name="Resource" keys="org:resource"/>
	<string key="source" value="SIM-PromEventLogGenerator-20131010134839"/>
	<string key="concept:name" value="SIM-PromEventLogGenerator-20131010134839"/>
	<string key="lifecycle:model" value="standard"/>
	<trace>
		<string key="concept:name" value="patient_4"/>
		<event>
			<string key="org:resource" value="UNDEFINED"/>
			<date key="time:timestamp" value="2013-01-01T09:15:00.000+01:00"/>
			<string key="concept:name" value="GoToDesk"/>
			<string key="lifecycle:transition" value="Start"/>
		</event>
		<event>
			<string key="org:resource" value="UNDEFINED"/>
			<date key="time:timestamp" value="2013-01-01T09:16:00.000+01:00"/>
			<string key="concept:name" value="GoToDesk"/>
			<string key="lifecycle:transition" value="complete"/>
		</event>
		<event>
			<string key="org:resource" value="FirstResearch_1"/>
			<date key="time:timestamp" value="2013-01-01T09:19:00.000+01:00"/>
			<string key="concept:name" value="FirstResearch"/>
			<string key="lifecycle:transition" value="complete"/>
		</event>
		<event>
			...
		</event>
		</trace>
		<trace>
			...
		</trace>
</log>
```

Appendix L: OpenXES metamodel

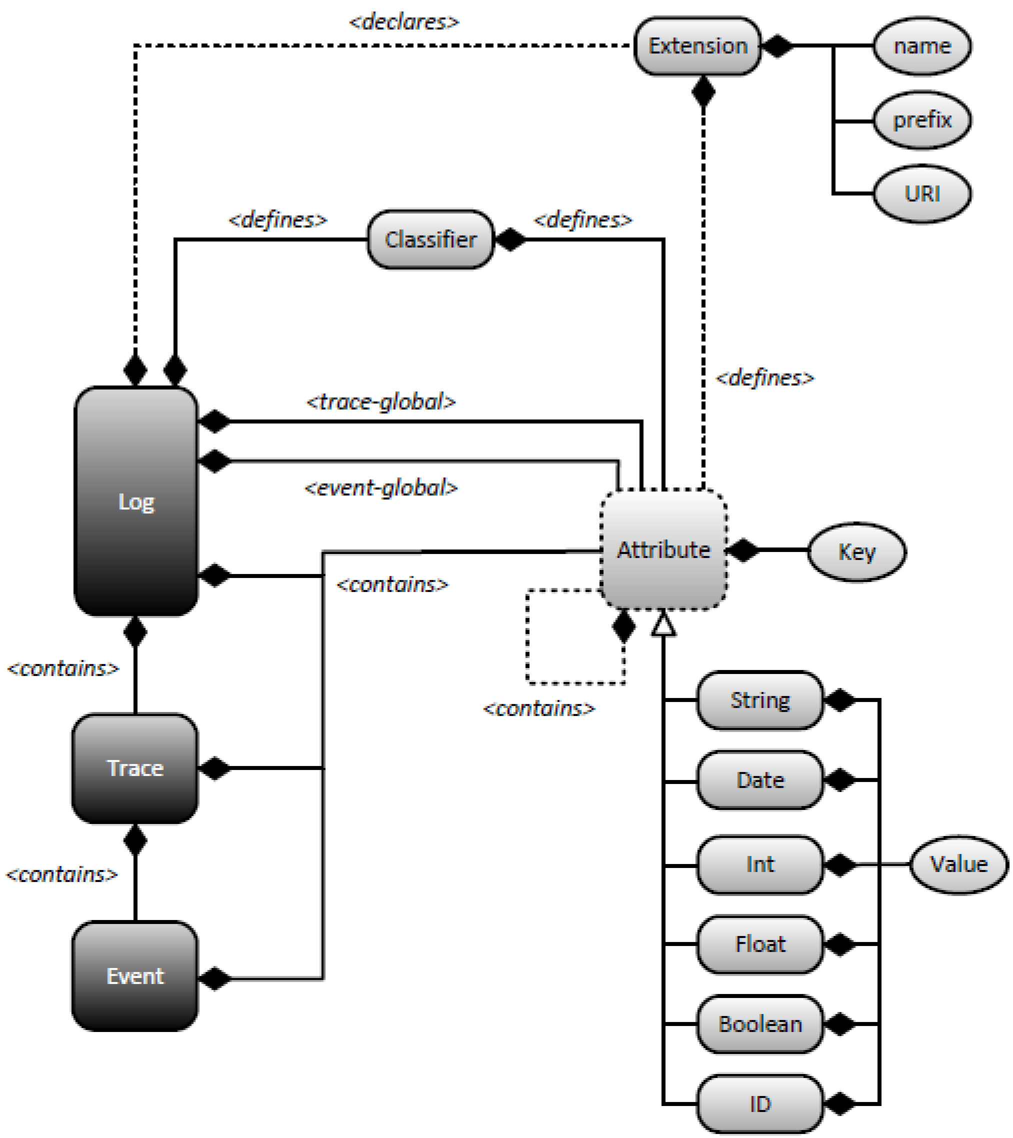
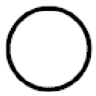
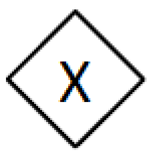
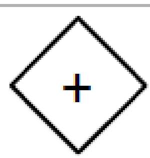
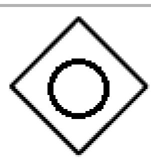


Figure 41: OpenXES metamodel (XES-Standard.org)

Appendix M: BPMN operators

The following table shows a subnet of modelling elements from the BPMN2.0 specification. The following subset is used in this project.

Symbol	Description
	Event: An event is used to model that something has happened. In this project only start and end events are used to mark the start and end of the process.
	Activity: Rectangles refer to activities which are executed.
	Exclusive-or (XOR): This gateway models an exclusive or split or exclusive or join. The abbreviation most commonly used is XOR
	Parallel (AND): This gateway models parallel execution paths. Also referred to as AND operator.
	Inclusive-or (OR): The OR-splits allows splitting and merging of one or more paths. All combination of patch may be chosen when using the inclusive-or.
	Sequence flow: used to show the order of activities

Appendix N: Process time measurements

The following times were measured during operational days at the cataract centre. The data was fitted using ExpertFit, all distributions were significant on 0.05 level.

Task	Target time	Count	Average; Stdev	Distribution
Pre-Operative measurements	20 min	47	17.77; 3.62	JohnsonSB(10.72526;32.77452;1.03279;1.18174)
Pre-Operative Consult 1/2	10 min	47	11.69; 6.86	JohnsonSB(2.30516;28.09044;.51169;0.63059)
Pre-Operative Consult 2/2	5 min	29	14.35; 7.17	Beta(3.81952;27.92741;0.76448;0.99928)
Post-Operative Week measurements	10 min	53	7.10; 1.76	JohnsonSB(2.08684;7.09182;1.12192; 1.48485)
Post-Operative week consult	5 min	7	7.75; 4.42	PearsonType6(2.51645; 36.61112;1.68839;12.79235);
Post-Operative end measurements	10 min	40	11.18; 3.16	JohnsonSB(2.08684;17.09182;1.12192;1.48485)
Post-Operative end consult	5 min	29	8.85; 3.73	Beta(2.10699; 34.77780;1.58714;6.53699)

Table 21: Measurements

Beta PERT process time approximation for extra measurement

The service time for the extra measurement in what-if scenario AS-IS-PLUS is estimated using the beta PERT approximation (Farnum & Stanton, 1987). Based on the expected minimum, median and maximum, the mean is estimated and the parameters for the beta distribution calculated.

$$mean = \frac{min + 4 * median + max}{6}, \qquad stdev = \frac{max - min}{6}$$

Parameters for the beta distribution are calculated according to:

$$v = \left(\frac{mean - min}{max - min}\right)\left(\frac{(mean - min)(max - mean)}{stdev^2} - 1\right)$$

$$w = \left(\frac{max - mean}{mean - min}\right) v$$

The following estimations were used:

$$mean = \frac{4 + 4 * 5 + 10}{6} 6.33 =, \qquad stdev = \frac{10 - 4}{6} = 1$$

$$v = \left(\frac{6.33 - 4}{10 - 4}\right)\left(\frac{(6.33 - 4)(10 - 6.33)}{1^2} - 1\right) = 2.938$$

$$w = \left(\frac{10 - 6.33}{6.33 - 4}\right) 2.938 = 4.628$$

The beta distribution's parameters are:

$$beta(min, \mathrm{max}, v, w) = \mathrm{beta}(4{,}10, 2.939{,}4.628)$$

Appendix O: Activity overlays

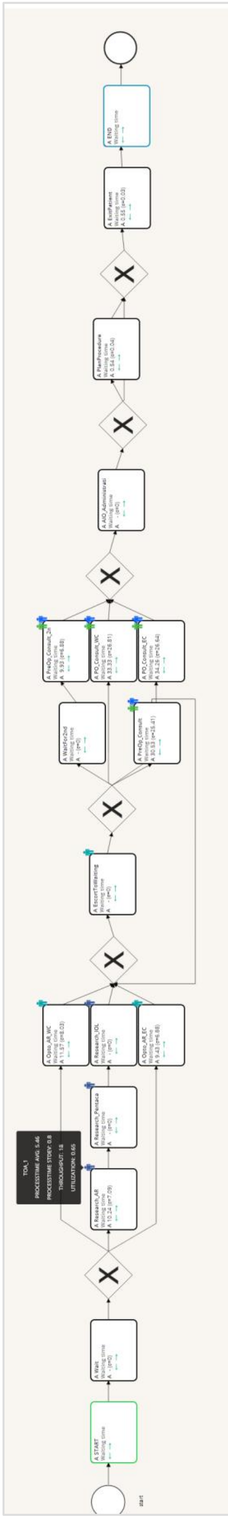


Figure 42: BPMN with overlays

Appendix P: Model validation

In Figure 43 it is demonstrated how to two simulation models are compared. This comparison shows the different resource allocation for what-if scenario #6 (see chapter 7.3.2). In Table 22 the overall throughput times are given. It is not possible to further aggregate the systems performance when validating the system, since the measurements for the whole system would not yield an accurate measure. This is due to the number of measurements for each track. Table 23 and Table 24 provides white-box validation measures for the service / consultation times and the waiting times.

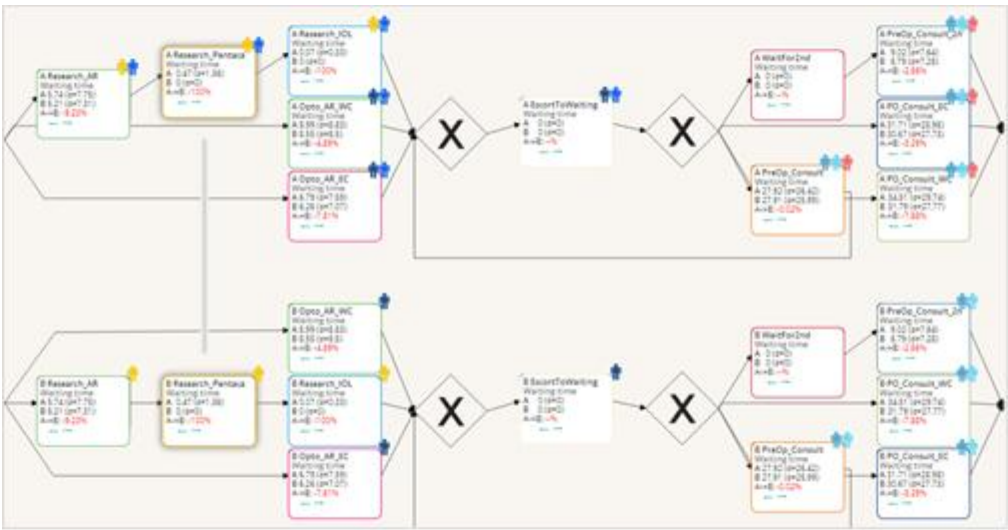


Figure 43: comparison of two simulation models (What-if#6 with as-is)

Table 22: Throughput times (black-box validation)

Appointment type	Real net throughput time	Simulation net throughput time
Pre-Operative	92.97 (σ=24.02)	98.86 (σ=30.85)
Post-Operative Week	41.73 (σ=24.55)	46.31 (σ=30.01)
Post-Operative End	47.29 (σ=20.34)	48.09 (σ=30.11)

Table 23: service times (white-box validation)

Task	Target process time	Real process time	Simulation process time
Pre-Operative measurements	20 minutes	17.77 (σ=3.62)	17.82 (σ=3.09)
Pre-Operative Consult 1/2	15 minutes	11.69 (σ=6.86)	11.61 (σ=6.7)
Pre-Operative Consult 2/2	5 minutes	14.35 (σ=7.17)	14.18 (σ=7.22)
Post-Operative WK optometrist	10 minutes	7.10 (σ=1.76)	7.02 (σ=2.06)
Post-Operative WK consult	5 minutes	7.75 (σ=4.42)	7.78 (σ=4.67)
Post-Operative end optometrist	10 minutes	11.18 (σ=3.16)	11.88 (σ=3.75)
Post-Operative end consult	5 minutes	8.85 (σ=3.73)	8.42 (σ=4.21)

Table 24: Waiting times (white-box validation)

Task	Real waiting time	Simulation waiting time
Pre-Operative measurements	2.72 (σ=7.92)	1.85 (σ=5.53)
Pre-Operative Consult 1/2	29.45 (σ=22.67)	27.91 (σ=25.99)
Pre-Operative Consult 2/2	24.59 (σ=11.49)	28.79 (σ=7.28)
Post-Operative wk measurements	10.52 (σ=10.71)	8.55 (σ=8.5)
Post-Operative wk consult	21.64 (σ=10.97)	31.79 (σ=27.77
Post-Operative end measurements	9.85 (σ=10.92)	6.26 (σ=7.07)
Post-Operative end consult	23.02 (σ=15.21)	30.67 (σ=27.73)

Appendix Q: Simulation results

In the following pictures the results of the simulation runs are depicted with a 95% confidence interval calculated using the Student-t distribution. In addition in Figure 44 the finish time for the resources is depicted.

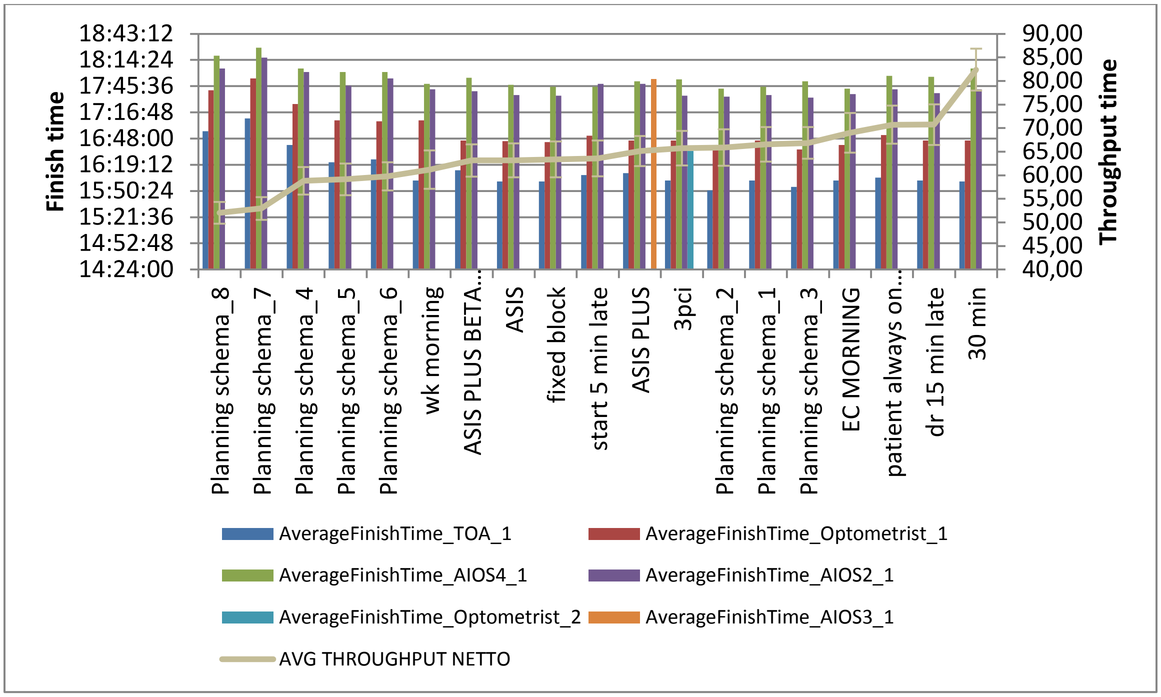
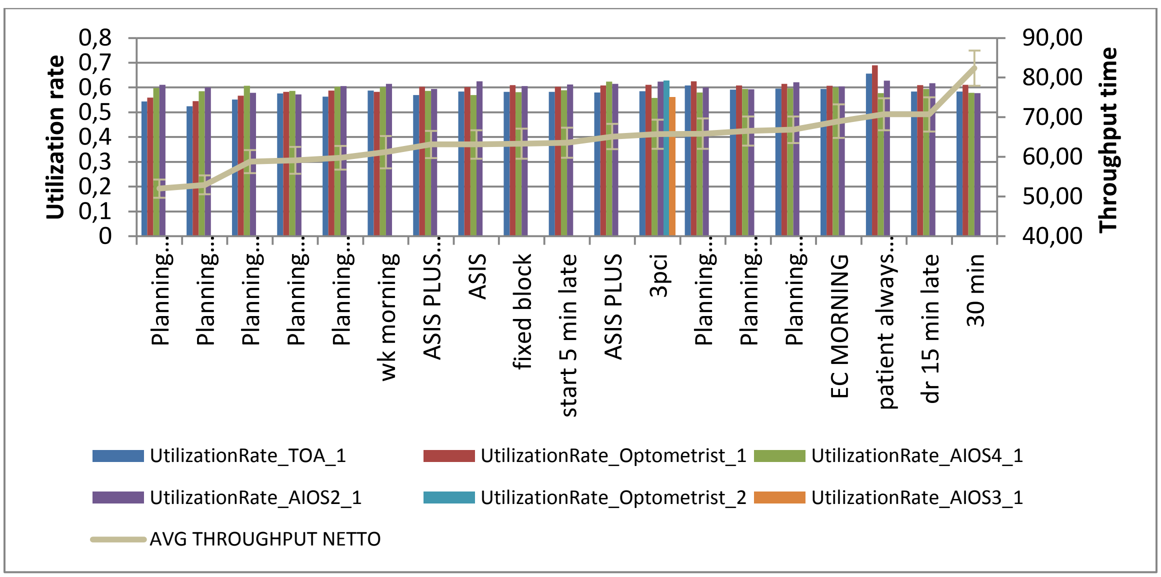


Figure 44: Net throughput time system and finish time per resource



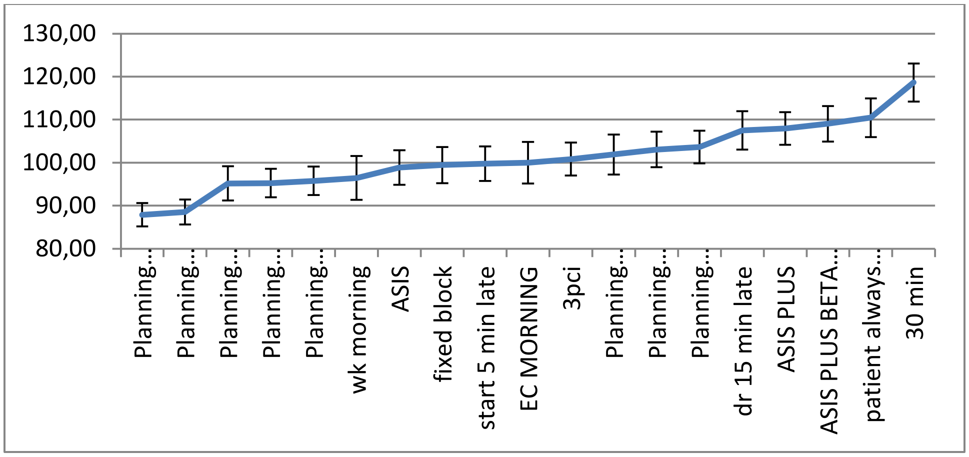


Figure 45: Net throughput time PreOperative patients

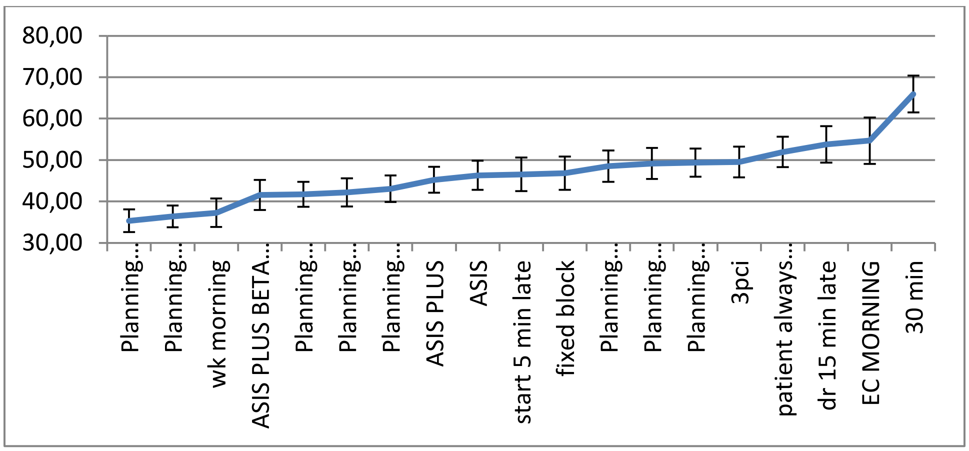


Figure 46: Net throughput time post-operative week control patients

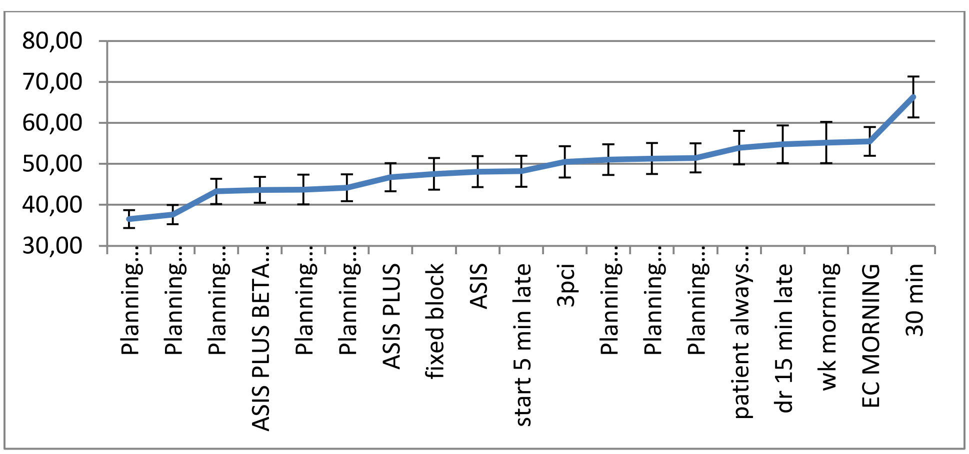


Figure 47: Net throughput time post-operative end control patients

Appendix R: BPMN Plugin ProM6

ProM version: 6.3 (rev. 10095)
BPMNAnalysis Plugin version: 6.3.61
BPMN Plugin: 6.4.109

The BPMN model in Figure 48 depicts the control-flow of the process for the cataract centre at level two using swim lanes.

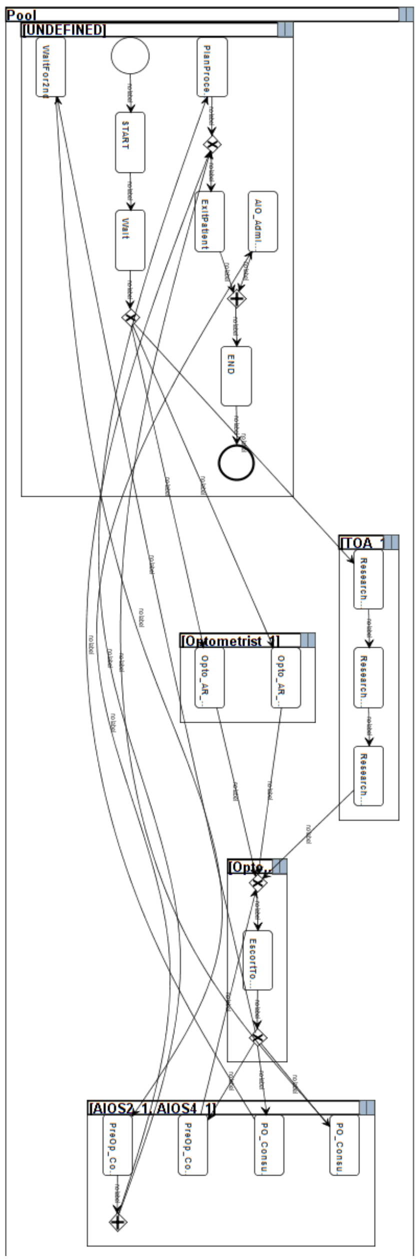


Figure 48: BPMN model for cataract centre with swimlanes

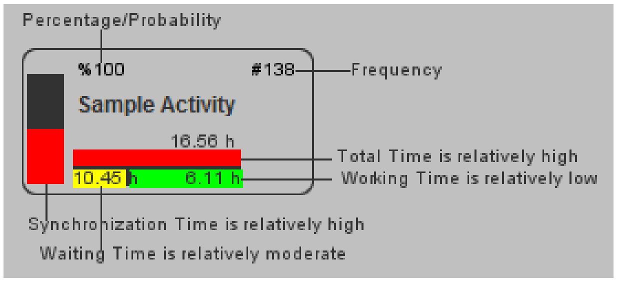


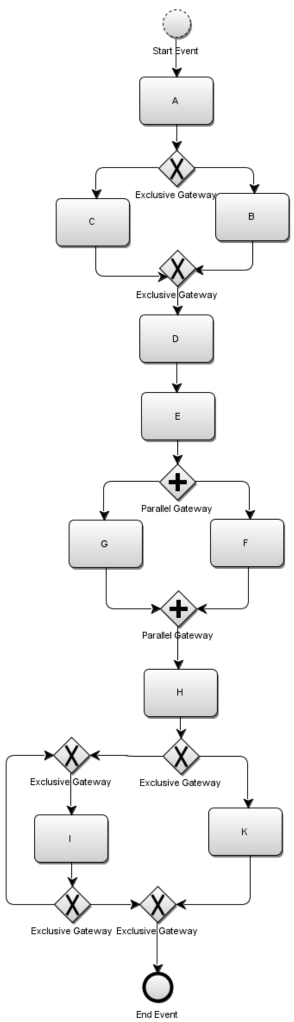
Figure 49: Legend of overlays in BPMN plugin

Appendix S: Experiment Questionnaire

In this appendix the questions from the experiment are given. Also the BPMN model of the first model in the experiment is given.

Age	
Gender	◯ Male ◯ Female
Education	
How many years are you working with Flexsim HC?	

	Strongly disagree				Strongly agree
I am familiar with process models	Strongly disagree ◯	◯	◯	◯	◯ Strongly agree
I am familiar with Flexsim HC	Strongly disagree ◯	◯	◯	◯	◯ Strongly agree



C is executed after B?	Yes	No	I don't know
G is always executed after F?	Yes	No	I don't know
Either I or K is executed once?	Yes	No	I don't know
I can be executed multiple times?	Yes	No	I don't know
E is always executed directly before F?	Yes	No	I don't know
The last activity in the process is K?	Yes	No	I don't know

	Very easy				Very difficult
Please rate the difficulty of model 1					

<<Model2 Flexsim HC or BPMN>>

Can the process finish without executing H?	Yes	No	I don't know
Can H be executed twice?	Yes	No	I don't know
Is D always directly followed by F?	Yes	No	I don't know
C is always preceded by B	Yes	No	I don't know
After C, either D or E is executed, not both	Yes	No	I don't know
Can I be executed after directly after G?	Yes	No	I don't know

	Very easy				Very difficult
Please rate the difficulty of model 2					

Questions based on (Moody, 2003)

I found the BPMN process model complex and difficult to follow	Strongly disagree				Strongly agree
Overall, I found the BPMN process models difficult to use	Strongly disagree				Strongly agree
I found the BPMN models easy to read and learn	Strongly disagree				Strongly agree
I found it difficult to answer the questions related to the track in Flexsim	Strongly disagree				Strongly agree
I found the semantics of BPMN easy to understand	Strongly disagree				Strongly agree
I am not confident that I am now competent to use BPMN in practice	Strongly disagree				Strongly agree
I believe that using BPMN models would reduce the effort required to validate large simulation models	Strongly disagree				Strongly agree
Large simulation models represented using this method would be more difficult for users to understand	Strongly disagree				Strongly agree
This method would make it easier for users to verify whether simulation models are valid	Strongly disagree				Strongly agree
Overall, I found using the process models to be useful	Strongly disagree				Strongly agree
Using this method would make it more difficult to maintain large simulation models	Strongly disagree				Strongly agree
Overall, I think this method does not provide an effective solution to the problem of evaluating simulation models	Strongly disagree				Strongly agree
Overall, I think this method is an improvement to the standard Flexsim HC patient tracks	Strongly disagree				Strongly agree
Using this method would make it easier to communicate simulation models to process owners	Strongly disagree				Strongly agree
I would definitely not use this method to evaluate simulation models	Strongly disagree				Strongly agree
I intend to use BPMN model in preference to the patient tracks for validation	Strongly disagree				Strongly agree

Appendix T: SPSS Output

In this Appendix the output of SPSS can be found. First an overview of output for the response times is given followed by an overview of the errors. The last section covers the output for the constructs of the Method Evaluation model.

Response times

The following SPS output is related to the response times of questions related to Flexsim HC tracks and BPMN models.

Table 25: Descriptive statistics response times

Descriptives

			Statistic	Std. Error
Times Flexsim HC	Mean		53,7935	10,78373
	95% Confidence Interval for Mean	Lower Bound	31,9813	
		Upper Bound	75,6057	
	5% Trimmed Mean		44,6194	
	Median		20,8200	
	Variance		4651,556	
	Std. Deviation		68,20232	
	Minimum		,78	
	Maximum		329,34	
	Range		328,56	
	Interquartile Range		60,95	
	Skewness		2,403	,374
	Kurtosis		6,549	,733
Times BPMN	Mean		13,2023	1,94246
	95% Confidence Interval for Mean	Lower Bound	9,2733	
		Upper Bound	17,1312	
	5% Trimmed Mean		11,7658	
	Median		8,8800	
	Variance		150,926	
	Std. Deviation		12,28520	
	Minimum		1,88	
	Maximum		58,56	
	Range		56,68	
	Interquartile Range		7,70	

	Skewness	2,196	,374
	Kurtosis	4,744	,733

Table 26: Response times normality test

Tests of Normality

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
Times Flexsim HC	,235	40	,000	,701	40	,000
Times BPMN	,249	40	,000	,715	40	,000

a. Lilliefors Significance Correction

Table 27: Response times results of Independent Samples Test and Levene's test

Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
time	Equal variances assumed	25,705	,000	3,704	78	,000	40,59125	10,95728	18,77697	62,40553
	Equal variances not assumed			3,704	41,528	,001	40,59125	10,95728	18,47111	62,71139

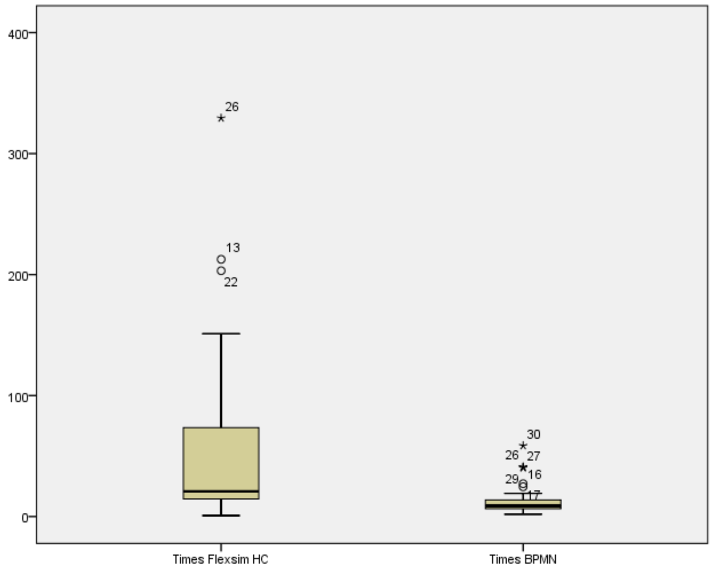


Figure 50: Box-plot response times

Table 28: Response times descriptives non-parametric test

Descriptive Statistics

	N	Mean	Std. Deviation	Minimum	Maximum	Percentiles 25th	50th (Median)	75th
time	80	33,4979	52,80125	,78	329,34	7,4600	14,2700	34,3350
group\	80	1,50	,503	1	2	1,00	1,50	2,00

Table 29: Response times Ranks

Ranks

	group\	N	Mean Rank	Sum of Ranks
time	1	40	52,45	2098,00
	2	40	28,55	1142,00
	Total	80		

Table 30: Response times result of non-parametric Mann-Whitney U test

Test Statistics[a]

	time
Mann-Whitney U	322,000
Wilcoxon W	1142,000

Z	-4,600
Asymp. Sig. (2-tailed)	,000

a. Grouping Variable: group\

Errors

The following SPSS output tables are related to testing the number of errors related to Flexsim HC tracks and BPMN models.

Table 31: Descriptive statistics errors Flexsim HC vs BPMN

Descriptives

	Group			Statistic	Std. Error
errors	Flexsim HC	Mean		1,17	,322
		95% Confidence Interval for Mean	Lower Bound	,46	
			Upper Bound	1,87	
		5% Trimmed Mean		1,07	
		Median		1,00	
		Variance		1,242	
		Std. Deviation		1,115	
		Minimum		0	
		Maximum		4	
		Range		4	
		Interquartile Range		2	
		Skewness		1,505	,637
		Kurtosis		3,212	1,232
	BPMN	Mean		,75	,305
		95% Confidence Interval for Mean	Lower Bound	,08	
			Upper Bound	1,42	
		5% Trimmed Mean		,67	
		Median		,00	
		Variance		1,114	
		Std. Deviation		1,055	
		Minimum		0	
		Maximum		3	
		Range		3	
		Interquartile Range		2	
		Skewness		1,149	,637

Kurtosis	,126	1,232

Table 32: Test for normality of Errors

Tests of Normality

	Group	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
		Statistic	df	Sig.	Statistic	df	Sig.
errors	Flexsim HC	,309	12	,002	,808	12	,012
	BPMN	,345	12	,000	,748	12	,003

a. Lilliefors Significance Correction

Table 33: Ranks and test statistics of non-parametric independent samples test

Ranks

	Group	N	Mean Rank	Sum of Ranks
errors	Flexsim HC	12	14,04	168,50
	BPMN	12	10,96	131,50
	Total	24		

Test Statistics[a]

	errors
Mann-Whitney U	53,500
Wilcoxon W	131,500
Z	-1,134
Asymp. Sig. (2-tailed)	,257
Exact Sig. [2*(1-tailed Sig.)]	,291[b]

a. Grouping Variable: groepid

b. Not corrected for ties.

Moody Constructs

The following SPSS output is related to the constructs of Moody (2003).

Table 34: Perceived Ease of Use Cronbach Alpha when deleting PEOU6

Item-Total Statistics PEOU

	Scale Mean if Item Deleted	Scale Variance if Item Deleted	Corrected Item-Total Correlation	Squared Multiple Correlation	Cronbach's Alpha if Item Deleted
PEOU1	18,38	6,839	,723	,935	,398

PEOU2	18,25	7,643	,625	,812	,461
PEOU3	18,38	7,696	,334	,913	,539
PEOU4	18,88	6,411	,501	,907	,453
PEOU5	18,50	8,286	,268	,938	,566
PEOU6	19,50	9,429	-,093	,281	,762

Table 35: Inter Item correlations for Perceived Usefulness

Inter-Item Correlation Matrix PU

	PU1	PU2	PU3	PU5	PU6	PU7	PU8
PU1	1,000	,301	-,552	-,120	,361	-,417	,000
PU2	,301	1,000	-,061	,092	-,277	-,320	-,641
PU3	-,552	-,061	1,000	,218	-,655	,000	,000
PU5	-,120	,092	,218	1,000	,333	,577	,577
PU6	,361	-,277	-,655	,333	1,000	,577	,577
PU7	-,417	-,320	,000	,577	,577	1,000	,500
PU8	,000	-,641	,000	,577	,577	,500	1,000

Table 36: Results when PU1 en PU2 deleted

Item-Total Statistics PU

	Scale Mean if Item Deleted	Scale Variance if Item Deleted	Corrected Item-Total Correlation	Squared Multiple Correlation	Cronbach's Alpha if Item Deleted
PU3	15,50	2,571	-,126	,786	,815
PU5	15,88	1,554	,681	,500	,552
PU6	15,88	1,839	,398	,875	,673
PU7	15,63	1,411	,675	,625	,540
PU8	15,63	1,411	,675	,625	,540

Table 37: Inter-Item correlation matrix for Intention To Use (ITU)

Inter-Item Correlation Matrix ITU

	ITU1	ITU2
ITU1	1,000	,808
ITU2	,808	1,000

Descriptives

			Statistic	Std. Error
PEOU	Mean		4,06	,170
	95% Confidence Interval for Mean	Lower Bound	3,70	
		Upper Bound	4,42	
	5% Trimmed Mean		4,07	
	Median		4,00	
	Variance		,463	
	Std. Deviation		,680	
	Minimum		3	
	Maximum		5	
	Range		2	
	Interquartile Range		1	
	Skewness		-,074	,564
	Kurtosis		-,489	1,091
ITU	Mean		3,94	,249
	95% Confidence Interval for Mean	Lower Bound	3,41	
		Upper Bound	4,47	
	5% Trimmed Mean		4,04	
	Median		4,00	
	Variance		,996	
	Std. Deviation		,998	
	Minimum		1	
	Maximum		5	
	Range		4	
	Interquartile Range		1	
	Skewness		-1,702	,564
	Kurtosis		4,439	1,091
PU	Mean		4,00	,183
	95% Confidence Interval for Mean	Lower Bound	3,61	
		Upper Bound	4,39	
	5% Trimmed Mean		4,00	
	Median		4,00	
	Variance		,533	
	Std. Deviation		,730	

	Minimum	3	
	Maximum	5	
	Range	2	
	Interquartile Range	2	
	Skewness	,000	,564
	Kurtosis	-,907	1,091

Table 38: Results one-sample T-Test

One-Sample Test

	Test Value = 3					
					95% Confidence Interval of the Difference	
	t	df	Sig. (2-tailed)	Mean Difference	Lower	Upper
PEOU	5,136	47	,000	,729	,44	1,01
ITU	3,758	15	,002	,938	,41	1,47
PU	11,041	63	,000	,891	,73	1,05

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
1	The median of PEOU equals 3,000.	One-Sample Wilcoxon Signed Rank Test	,000	Reject the null hypothesis.
2	The median of PU equals 3,000.	One-Sample Wilcoxon Signed Rank Test	,000	Reject the null hypothesis.
3	The median of ITU equals 3,000.	One-Sample Wilcoxon Signed Rank Test	,008	Reject the null hypothesis.

Asymptotic significances are displayed. The significance level is ,05.

Figure 51: Results non-parametric one sample test

Appendix U: Appointment scheduling system

This appendix provides some theoretical background for defining appointment scheduling systems.

Appointment scheduling schemas

Scheduling is an important factor to effectively match the demand and capacity to better utilize resources and minimize patient waiting times. Cayirli & Veral (2003) break an appointment system down into three decision categories: (1) appointment rules, (2) patient classification and (3) adjustments to account for disruptive interruptions. The last category is not very applicable in the MUMC+ models since the cataract centre does not handle walk-in patients and emergencies. Also the number of no shows is fairly low; during observation sessions of the cataract centre 11 no shows over 180 patients were counted. The appointment rules are according to Cayirli & Veral (2003) described by three variables:

1) Block size: number of patients scheduled in a block (a session is divided into *i* blocks)
2) Begin-block: number of patients given an identical appointment time at the start of a session
3) Appointment interval: the interval between two successive appointment times

Any combination of the three variables described above determines a possible appointment rule. The current appointment rule for the cataract centre is an individual-block with a variable interval. This appointment rule is applicable to the technical ophthalmic assistant, optometrist and the doctor. The doctor’s schedule is dependent on the schedule of the TOA and optometrist but handles the patient according to the layout given in Appendix G.

The cataract appointment system assigns pre-marked slots for a patient classification (pre-operative, post-operative week/end). This works fairly well because the number of patients with a different patient classification is very well distributed. The number of patients for pre-operative and post-operative is distributed as 1:2. Also the interval and slot-times are dependent on the classification for post-operative week and end control.

Appendix V: Tool screen shots

Figure 52 shows the import screen. Here the output logs of Flexsim HC are imported into the tool. In Figure 53 the overview of different imports is given. It is possible to upload the XPDL, download the OpenXES event log, download the BPMN file or download the JSON object with KPIs. In this screen two logs can be selected to generate the process view which is depicted in Figure 54. Here the global system KPIs can also be chosen in addition to the default view.

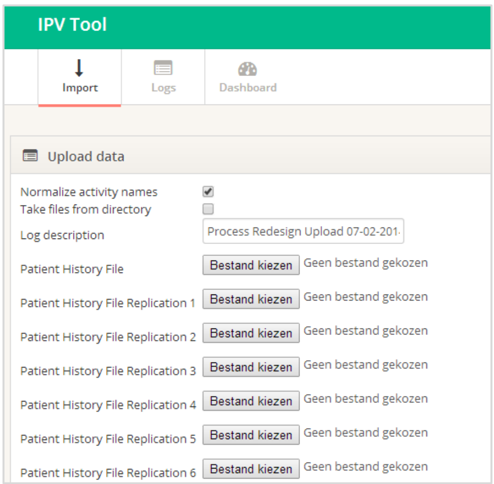


Figure 52: Import screen of the IPV tool

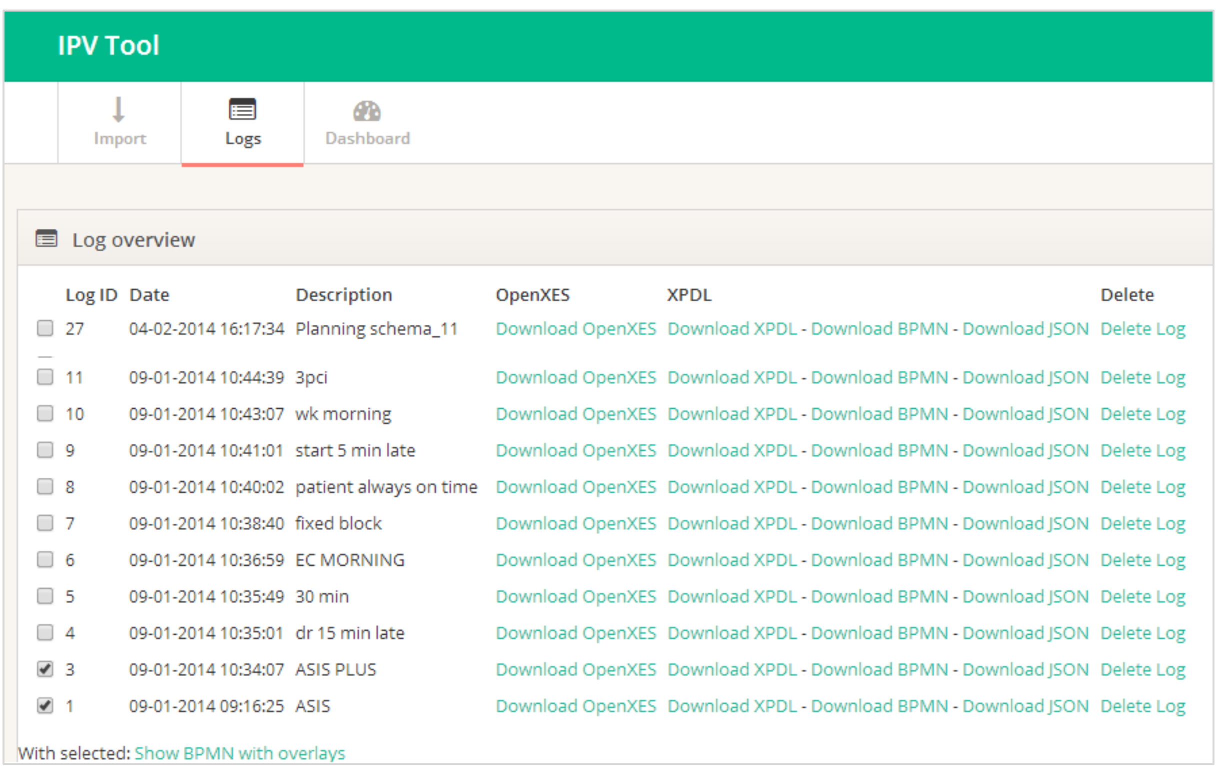


Figure 53: Overview of simulation logs in IPV Tool

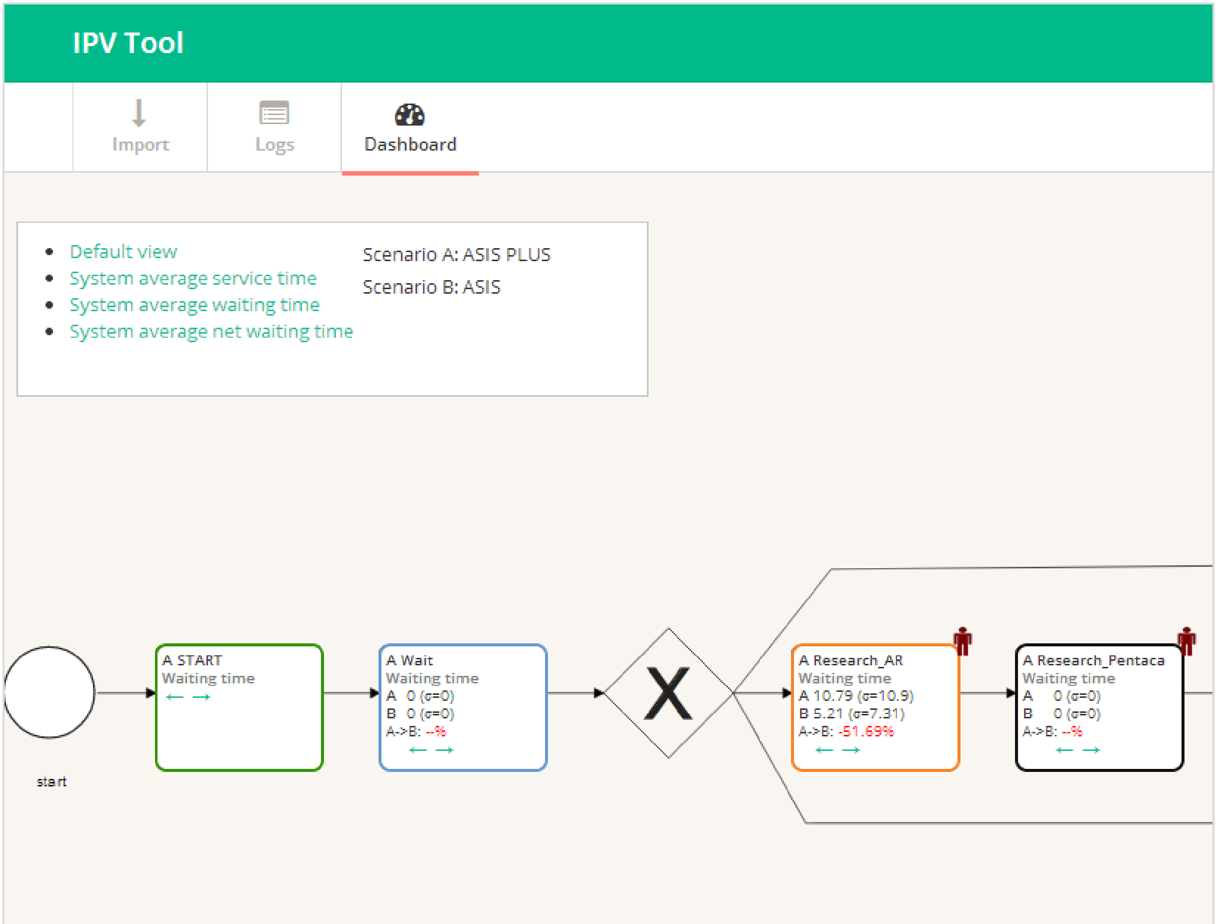


Figure 54: Integrated Process View of the tool