Faculty of Business Economics

Master of Management

Masterthesis

A comparative study of business process simulation tools

Kiarash Diba

Thesis presented in fulfillment of the requirements for the degree of Master of Management, specialization Business Process Management

SUPERVISOR :

dr. Niels MARTIN

2017
2018

www.uhasselt.be
Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek

Faculty of Business Economics

Master of Management

Masterthesis

A comparative study of business process simulation tools

Kiarash Diba

Thesis presented in fulfillment of the requirements for the degree of Master of Management, specialization Business Process Management

SUPERVISOR :

dr. Niels MARTIN

Preface

This thesis is written and submitted as partial requirement for a Master's degree in Business Process Management (BPM) at University of Hasselt, Belgium. A comparative study of business process simulation tools has been conducted in order to evaluate different categories of simulation tools on their suitability for business process simulation. This thesis attempts to contribute to the field of business process simulation for both BPM users and BPM research community.

I would like to thank Dr. Niels Martin for his help and valuable feedbacks in the role of supervisor. His supervision not only contributed greatly to the quality of this thesis but also has grown my academic knowledge and made me ready for the next step in my academic career.

Summary

Business process management (BPM) is a discipline focusing on processes inside an organization with the aim of improving business processes and eventually the organization’s performance (Weske, 2012). One way to achieve these improvements is to use process simulation (van der Aalst et al. 2016). Business process simulation (BPS) is experimenting with the process model with the aim of enabling and enhancing change and improvement in the current processes. Simulation model is a computer model which mimics the behavior of a real life process and attempts to identify bottlenecks and inefficiencies in the process (Martin et al. 2016). It then assists finding a solution to tackle these problems and to improve the process by experimenting on the model rather than the real life. To perform simulation on business processes, variety of simulation tools with different features and requiring different levels of expertise exist. The selection of an appropriate simulation tool, therefore, is a difficult task and requires careful assessments of alternatives. The selection should be based on project goals, available resources and expertise. Therefore, the simulation tool selection procedure should be structured and decision makers need to be guided. To this end this thesis provides a guideline and uses a structured decision making approach called Analytical Hierarchy Process (AHP). This approach assists the decision making by structuring the decision making process into hierarchy levels, considering different criteria (Saaty 1994). This thesis considers three categories of simulation tools and compares them for their suitability on simulating business processes. These three categories are:

i. Dedicated simulation tools
ii. Business process management tools with simulation capabilities
iii. Simulation packages in programming languages

Dedicated simulation tools are software packages developed exclusively for simulation modeling. A few instances of this group of tools are Arena, Simul8, AnyLogic, Bimp and Scylla. In addition, some business process management software packages have deployed simulation into their platform and enabled business process simulation. A few examples are, Signavio, Bizagi and Aris. Some simulation packages have been developed for programming languages such as Python, Java and R. SimPy, Desmo-J and Simmer are a few of this type of simulation tools.

This thesis aims at comparing these three categories of simulation tools for the purpose of BPS and to assist users in the selection procedure by providing a structured approach for comparison and selection of BPS tools. Furthermore, the main limitations and areas for future developments of simulation tools in the field of business process management (BPM) are identified.

Five simulation tools have been selected from the above-mentioned categories for the comparison. These five tools are Arena and Scylla from the dedicated simulation tools, Signavio and Bizagi from business process management tools and SimPy from the simulation packages in programming languages.

In order to evaluate and compare simulation tools, a set of criteria and desired features should be identified and the tools need be compared on these criteria. The criteria in this thesis have been selected by analyzing the literature on simulation software evaluation and selection. In addition, a new group of criteria regarding simulation modeling of business processes has been identified which have not been mentioned in previous literature. This group of criteria is the support of different modeling tasks involved in simulation of business processes. The five groups of criteria used in this thesis are as follows:

i. Criteria regarding coding aspects
ii. Modeling and simulation capabilities
iii. Input and output functionalities
iv. Criteria concerning users and simulation vendor
v. Supported modeling tasks

The evaluation and comparison has been performed using a template developed, as part of this thesis, in excel. This template implements AHP, the evaluation method used, and can be used as a guideline for the comparison and selection of simulation tools. Furthermore, the template can be updated and adjusted for new comparisons. Different criteria can be added, the weights of criteria can be adjusted based on project requirements and resources and new tools can be added to be compared.

Keywords: Business process simulation, Simulation software evaluation, Simulation tool comparison

Table of Contents

List of tables

List of figures

1. Introduction

1.1. Introduction

A Business process is a set of tasks and activities performed to realize a specific business goal and create or deliver value to customers (Weske, 2012). Processes ranging from production and manufacturing processes to sales and customer support may exist in organizations and can be managed by Business Process Management (BPM). BPM is a discipline focusing on designing, managing, analyzing and improving business processes. As defined by Weske (2012) "Business process management includes concepts, methods and techniques to support the design, administration, configuration, enactment, and analysis of business processes". Van der Aalst (2013) defines BPM as a "discipline that combines knowledge from information technology and knowledge from management sciences and applies this to operational business processes". The aim is to manage, design and improve processes in a way that leads to performance and productivity improvements, cost reduction and eventually achieving strategic objectives. BPM is based on Workflow Management (WFM) which focuses on automation of processes. However, BPM includes not only process automation but also process analysis and improvement, enactment, operations management and organization of work (van der Aalst 2013; Weske 2012).

A business process model is a graphical representation of a process inside an organization and is the foundation of BPM. A business process model displays activities, events and the control flow (i.e. the logical ordering of activities) involved in a business process. It is important for the design and analysis of process-aware systems, re-engineering and change management (Indulska et al. 2009; Weske 2012).

Over the last few decades, the use of business process modelling has increased significantly among various industries. Business process modelling can increase efficiency, clarity and facilitates communication between people and departments in a company. To ensure the efficient execution of business processes there is a need for continuous improvement and evaluation of processes as the original goal of BPM is to change and improve business processes (van der Aalst 2013). Continuous improvement requires effective techniques. Analytical approaches, in which mathematical models and formulas are used to describe the system, such as queueing theory can be used to analyze and model business processes using mathematical and statistical techniques. Queueing theory is a branch of operations research and deals with problems involving queuing or waiting (e.g. in banks, supermarkets, call centers) (Sundarapandian 2009). However, for complex systems it is difficult or sometimes impossible to use analytical techniques as they often fail to consider randomness and variability of the system. Examples of variability in processes are customer arrival rate or duration of activities (White and Ingalls 2015). Analytical approaches can only provide reliable and valid result when specific assumptions, on which the model is formulated are held. However, in real-life systems these assumptions are often violated. Therefore, a simplified model of the system need to be considered which affects the reliability of the result (Greasley 2003).

As a result of the limitations of analytical approaches the need for a more reliable method for process improvement arises. One way to approach process improvement projects is to first model processes and then simulate them and analyze them by measuring key performance indicators (KPIs). Instances of the KPIs which determine the performance of the process are cycle times, waiting times, cost, etc. KPIs defined for a certain project need to be aligned with objectives of the project and the strategic objectives of the organization. The aim here would be to find a process design which yields to improved KPIs and more efficient use of resources (van der Aalst et al. 2016). Simulation is imitating the behavior of a system or a process using a computer model of a real system. One of the advantages of simulation is incorporating the variability of the system using statistical distributions as opposed to fixed numbers (e.g. defining probability distributions for arrival rate instead of fixed intervals) (Greasley 2003).

Business process simulation (BPS) “mimics the behavior of business processes” (Melão and Pidd 2003). It explores and clarifies the effects of changes in the process prior to implementation (Melão and Pidd 2003). The objective is to enable clear analysis and improvement of business processes (Rozinat et al. 2008b). Simulation is a cost-effective option to analyze and compare alternative designs as experimenting in real life will be costly and time consuming and often impossible. It enables the ability of performing scenario based analysis and what-if analysis. Simulation is not only used in the change process and in the context of process re-engineering but also to understand and analyze the behavior of the current version of the process. It provides insight into the current version of the model to measure variables of the system (e.g. waiting times, cycle times, resources availability, etc.) and to analyze the performance of the system (Greasley 2003).

Kellner et al. (1999) categorizes the reasons to use process simulation into six groups: (i) strategic management, (ii) planning, (iii) control and operational management, (iv) process improvement and technology adoption, (v) understanding, (vi) and training and learning. Simulation can help managers with strategic decisions and answering high-level strategic question such as the decision to outsource a certain task or to perform the job in house. Furthermore, it can assist in planning and forecasting variables such as cost, time, staffing and resources needed. It also supports control and operational management by facilitating comparison of planned values versus actual values of KPIs which helps managers take corrective measures where needed. By forecasting the impact of change and evaluating alternative designs prior to implementation, simulation supports the process improvement projects. In the same way, by evaluating the impact of adoption of new technologies into a business process, it assists the decision making of technology adoption. Simulation facilitates better understanding of the processes (process flows, activities, resources involved, etc.) and therefore identification of process issues and bottlenecks in the process (Kellner et al. 1999).

An important step in a simulation project is designing and mapping the process model and assignment of resources to the model. Then variables of interest and performance measures (e.g. cycle time, activity cost, etc.) are selected and simulation is run on the model. The result of the simulation is then examined by analyzing the output of the performance measures and variables. Statistical data analysis can be used on the simulation output to gain useful insights (Tumay, 1996).

There are different types of simulation based on the implementation methods. Continuous system simulation, agent based, discrete-event simulation (DES), etc. (White and Ingalls 2015). In business process models it is assumed that the state of the system only changes at discrete points in time. Thus, for the purpose of BPS the most relevant approach is the discrete-event simulation. (Greasley 2003).

One important decision is the choice of a suitable simulation tool. Due to the discrete-event characteristic of business processes mentioned in the above paragraph, the choice of simulation tools is limited to simulation tools with discrete-event simulation functionalities. The focus of this thesis, therefore, will be on discrete-event simulation. The tool can be a programming language, dedicated simulation software or process modeling software with simulation functionalities. The decision should be based on the project objective and requirements, complexity of the model and skills and resources available. Depending on the objective of the project different simulation features will be of higher importance. For example, when process improvement and alternative evaluation is the objective of the project, simulation capabilities, statistical and experimentation facilities will be important (Bosilj-Vuksic et al., 2007). There are several simulation tools available each having different characteristics which makes it difficult for firms to choose the best tool based on their needs and expertise.

1.2 Research Objective

The objective of this thesis is twofold:

1. to guide the users in evaluation, comparison and selection of simulation tools for business processes and,
2. to identify limitations and areas for further developments of simulation tools for business processes

These are addressed by drawing a comparison between different business process simulation tools with regard to their level of strength in modeling and simulation capabilities, statistical facilities, visual aspects, flexibility, ease of modeling, supported simulation modelling tasks, etc. (Bosilj-Vuksic et al. 2007). This thesis points out strong aspects and limitations of every option and compares tools on different criteria in order to guide the decision makers in selecting a suitable tool for their simulation needs based on the project objectives, available resources and the expertise regarding the business process simulation. This thesis also sheds light on key areas for potential developments of BPS tools and identifies the limitations of current tools. This will assist BPM researchers, developers and software vendors in developing more comprehensive tools for BPS in the future. We focus on three categories of simulation tools relevant for BPS. These three categories are as follows:

1. dedicated simulation tools: software packages developed for the sole purpose of simulation.
2. process modelling tools with simulation functionalities: These are software packages for modelling business processes. They usually support one or more business modeling notations such as BPMN or Petri net and the simulation provided by these tools are tailored toward these modeling notations.
3. simulation packages in Python/R/Java: In some programming languages simulation libraries and packages have been developed to facilitate simulation. These libraries can be imported to the associated programming language and used to build simulation models.

Every tool in each category requires different expertise and resources and they have different strengths and limitations.

1.3 Research Question

This thesis focuses on answering the following questions:

Main question:

1. What are the main differences between business process simulation tools with respect to modeling capabilities, simulation capabilities, ease of use, etc. and how can companies be supported during the selection of an appropriate simulation tool?

Sub-questions:

1.1. Which of the tools are the most suitable for BPS?

1.2. What are the strong points and limitations of tools belonging to each of the three categories of dedicated simulation tools, process modeling tools with simulation capabilities and simulation packages in programming languages such as Python?

1.3. What are the areas for future developments?

1.4. How can decision makers in companies select a suitable tool for their requirements?

1.4 Research Methodology

To answer the research questions outlined above a comparative study of business process simulation tools is designed. The desired features of simulation tools and the criteria for evaluation and comparison of simulation tools are discussed and five simulation tools are selected to perform the final comparison. A flexible comparison and selection template for business process simulation tools is provided.

We start with analyzing the existing literature on evaluation criteria of simulation tools. For collecting the literature, Google scholar and Hasselt university library have been consulted. The search terms used to extract relevant literature are 'Business Process', 'Simulation', 'Business process simulation', 'simulation tools', 'BPS tools', 'Simulation tool comparison', 'Simulation software evaluation', etc. Afterwards, we select important criteria for the comparison of BPS tools from previous studies and existing literature regarding this subject. Regarding the comparison methodology, we explore literature on comparison of computer software and simulation tools. Based on this analysis, Analytical Hierarchy Process (AHP), a multi-criteria decision making method developed by Saaty (1994), is chosen as the comparison methodology of this thesis. First, the criteria are compared based on their importance over each other in a pairwise way. Then, the tools are compared in a pairwise way against each criterion. In this way every criterion is assigned a weight of importance and each tool is assigned a score on its performance on every criterion. After aggregating the result, the tools are ranked from the highest score to the lowest. A more detailed explanation of the comparison methodology and AHP is given in chapter 3 of this thesis.

Five simulation tools are selected for performing the comparison with regard to the three following categories of simulation tools: dedicated simulation tools, process modelling tools with simulation functionalities, and simulation packages in Python/R/Java. The selected tools are installed (for the commercially available tools, free student versions with unlimited simulation functionalities has been installed). These tools are checked for the existence of desired features by checking their websites, reading available documentation and watching tutorials. For the quality check of features, sample models have been designed in each tool and simulation has been performed on them.

Research approach of this thesis is a combination of context analysis and empirical research. Context analysis is used by reviewing existing literature on the evaluation of BPS tools and to gather important features of simulation tools and criteria for comparison. Empirical research is done by working with a selection of simulation tools from each category outlined in this thesis (Bosilj-Vuksic et al. 2007).

1.5. Thesis outlook

The remaining of this thesis is structured as follows:

Chapter 2 outlines the conducted literature review and is divided to three parts. A comprehensive list of a number of available simulation tools is provided in section 2.1. Section 2.2 lists the simulation software evaluation criteria mentioned in previous literature and analyzes the findings and the limitations of these works. In section 2.3, simulation software evaluation is studied from a methodological point of view. The different methods used in previous literature for evaluation and comparison of simulation tools are outlined and analyzed.

Chapter 3 sets the scene for the evaluation and comparison conducted in this thesis. In section 3.1 the selected tools, from the list in section 2.1, for performing the comparison are mentioned. In section 3.2 the evaluation criteria used are listed and a brief explanation of each criterion is given. Section 3.3 defines and explains AHP, the comparison methodology selected for the comparative study in this thesis. Section 3.4 provides an overview of the spreadsheet developed in this thesis to be used for comparison of business process simulation tools.

In chapter 4 we outline the result of the comparison, identifying strong points and limitations of every category of tools. A discussion has been made on important areas for further development of simulation tools for business processes.

Finally, chapter 5 concludes this thesis and provides recommendation for further research on this subject.

2. Literature review

In this chapter, an overview of the literature on the comparison and selection of simulation tools is provided. In section 2.1 a list of a number of available tools for business process simulation is provided. A comprehensive review of existing works on evaluation criteria and desirable features of simulation tools is drawn up in section 2.2 and previous works attempting to provide guidelines and methodologies for the selection of a suitable tool are discussed in part 2.3.

2.1. List of simulation tools

A list of a number of available simulation tools suitable for business process simulation is provided in table 2.1. The tools are divided into three categories in accordance with the comparison objective of this thesis. These three categories are:

1. Dedicated simulation tools
2. Business process modelling tools with simulation functionalities
3. Simulation packages in Python/R/Java:

Furthermore, the group of dedicated simulation tools, is divided in two sub-categories:

1.1. General purpose simulation tools: simulation software packages that use their specific modelling notation.

1.2. Simulation tools using BPMN notation: simulation tools developed for the purpose of simulating BPMN models.

Simulation software	Description
1. Dedicated simulation tools	
1.1. General purpose simulation tools	
Arena	A general purpose and discrete-event simulation software by Systems Modeling and acquired by Rockwell Automation in 2000. It is built on the SIMAN simulation language (Hammann and Markovitch 1995).
AnyLogic	A simulation modeling software developed by the AnyLogic Company. It can be used for agent-based, discrete-event, and system dynamics simulations (Yang et al. 2014).
Simul8	SIMUL8 is a simulation tool by SIMUL8 Corporation. It is a discrete-event simulation tool (Hauge and Paige 2004).
Simio	Simio Simulation and Scheduling Software is simulation tool developed by Simio LLC. Simio is an object-oriented simulation software with a 3D modeling environment (Kelton et al. 2011).
SIMPROCCES	SIMPROCESS is a process simulation tool focusing on process modeling and analysis. SIMPROCESS supports Process mapping, hierarchical event-driven simulation, and activity-based costing (ABC) (Swegles 1997).
ProModel	A process simulator developed by ProModel Corporation. It is a free DES tool which simulates Microsoft Office Visio flowcharts, Value Stream Maps and workflow diagrams. Process Simulator installs as an add-in to Visio, allowing the user to create and run simulation models inside Visio (Harrell et al. 2004).
ExtendSim	A simulation tool for modeling of discrete event, continuous, agent-based, and discrete rate processes. There are four ExtendSim packages: CP (continuous processes). OR (operations research) which adds discrete event. AT (advanced technology) which adds discrete rate, a number of advanced modeling features, and for statistical distribution fitting. Suite which adds 3D animation (Krahl 2009).

Table 2.1. List of simulation tools

Simulation software	Description
Enterprise Dynamics	Enterprise Dynamics is a discrete event simulation software by INCONTROL Simulation Solutions. It includes different libraries for different business areas, like manufacturing, logistics, Crowd Simulation, etc. ("Enterprise Dynamics," n.d.).
FlexSim	A discrete-event simulation tool by FlexSim Software Products, Inc. Its product family consists of a general purpose discrete-event simulation software and a simulation tool specifically for Healthcare, FlexSim Healthcare (Nordgren 2003).
GPSS World	A simulation tool by Minuteman Software based on the General Purpose Simulation System, GPSS, which is a programming language for simulation (Cox 1991).
CPN Tools	A tool developed by the CPN Group at Aarhus University for editing, simulating, and analyzing Petri net models. The simulation can support basic Petri nets, timed Petri nets and Colored Petri nets. In 2010, CPN Tools was transferred to Eindhoven University of Technology, The Netherlands (Jansen-Vullers and Netjes 2006).
1.2. Simulation tools using BPMN notation	
Bimp	Bimp is a free business process simulator supported by University of Tartu and the Estonian Research Council. Models created using BPMN standards can be uploaded to the software and simulation scenarios should be defined, then simulation can be executed. (Freitas and Pereira 2015).
Scylla	Scylla is an open-source, java-based and extensible BPMN process simulator developed in Hasso Plattner Institute (HPI) in Potsdam, Germany. It is based on the java-based discrete event simulator DESMO-J. (Pufahl et al. 2017).

Table 2.1. List of simulation tools (continued)

Simulation software	Description
BP simulator	BP Simulator is a free business process simulator. It offers Event-driven Process Chains (EPC), BPMN and Visio modeling, discrete-event simulation, and task-oriented business analysis (Lazzari and Crosslin 1996).
L-Sim	A simulation software aimed at simulation of business processes, developed by Lanner group. L-Sim is a simulation engine that can be embedded within BPM software platforms. L-Sim supports BPMN 2.0 format and enables simulation of BPMN based models ("L-Sim BPMN Model Simulation Engine," n.d.).
2. Business process management tools with simulation	
Signavio	Signavio Process Manager is an application for modeling business processes and decisions. It enables the creation of process diagrams using Business Process Model and Notation (BPMN) and decision diagrams using Decision Model and Notation (DMN). As part of the Signavio Business Transformation Suite, Process Manager also provide simulation of business processes ("Process Simulation | Signavio," n.d.).
Bizagi	Bizagi BPM Suite consists of three products, Bizagi BPMN Modeler which is an application to diagram, document and simulate processes in BPMN format. Bizagi Studio is a BPM solution which allows organizations to automate business processes and workflows. Bizagi Engine then, executes previously modeled & automated processes (Freitas and Pereira 2015).
Engage Process	A BPM software from a software vendor in the Netherlands. Engage Modeler enables design and evaluation of business processes. Engage Modeler has a built-in event driven simulator("Easily optimise your business processes," n.d.).

Table 2.1. List of simulation tools (continued)

Simulation software	Description
ARIS Express	A free business process modeling tool developed by IDS Scheer and later bought by Software AG. It supports different business modeling notations such as BPMN 2.0 and EPC. The tool which include simulation is provided as freeware on the ARIS Community webpage (Cimino and Vaglini 2014).
IGrafx	iGrafx is a web-based BPM platform. iGrafx Process is a process analysis and simulation tool which provides discrete-event simulation. (Cimino and Vaglini 2014).
Visual Paradigm	Visual Paradigm is a UML CASE Tool supporting UML 2 and Business Process Modeling Notation (BPMN). It also contains simulation functionality. (Cimino and Vaglini 2014).
ProcessModel	A process engineering tool for improving business processes. It supports modeling and simulation of processes. ("Simulation software," n.d.)
Sparx Systems Enterprise Architect	Sparx Systems Enterprise Architect is a modeling tool based on UML. It also supports BPMN standard for modeling business processes. Simulation is only possible by purchasing MDG BPSim Execution Engine add-inn separately (Cimino and Vaglini 2014).
3. Simulation packages in programming languages	
SimPy	A simulation framework for Python originally developed by Klaus Müller and Tony Vignaux. It is a process-based discrete-event simulation framework based on standard Python. SimPy has been re-implemented in other programming languages; SimSharp (for C#), SimJulia (for Julia), and Simmer (for R) are all re-implementations of SimPy (Muller and Vignaux 2003).

Table 2.1. List of simulation tools (continued)

Simulation software	Description
Desmo-J	A discrete-event simulation library in Java developed at University of Hamburg. It supports event-oriented and process-oriented simulation. It has been integrated into some business process modelling tools to support simulation (Göbel et al. 2013).
SystemC	SystemC is a set of C++ classes and macros which provide an event-driven simulation interface (Black et al. 2011).
SIM.JS	SIM.JS is JavaScript simulation library for modeling and simulation of discrete-event systems. (“SIM.JS,” 2011)

Table 2.1. List of simulation tools (continued)

2.2. Evaluation Criteria and simulation software features

In order to evaluate and compare tools a set of evaluation criteria is needed. There are many works focused on evaluation criteria and desirable features of discrete-event-simulation (DES) tools. Pidd (1992) describes three categories of DES software selection criteria. One category focuses on the computer programming aspect (the programming language used for the tool, access to the code, machine code, assembly languages, compilers, etc.) Another group focuses on analyzing different simulation approaches (statistical distributions, random number generation and report generation). The third group focuses on factors to be considered for the assessment of software packages, such as the type of application (discrete-event, continuous, etc.), knowledge and user support.

Davis and Williams (1994) provide a framework for the selection of simulation tools. Their defined criteria and sub-criteria were:

- cost of purchasing the software,
- comprehensiveness of the system (flexibility, statistical facility, graphical capability),
- integration with other systems,
- documentation (instruction manual, reference manual),
- availability of training by vendor,
- ease of use (expertise required, time needed for a new user, expert or end-user to build a model),
- hardware and installation,
- confidence related issues (support, further development).

Bradley et al. (1995) suggests a framework for the comparison of Business process re-engineering (BPR) tools which provide modeling and simulation capabilities, based on the following criteria:

- Tool capabilities (including a rough indication of modelling, simulation and analysis capabilities).
- Tool hardware and software (the type of platform, languages, external links and system performance).
- Tool documentation (the availability of several guides and online help).
- User features (user friendliness, level of expertise required, and existence of a graphical user interface).
- Modelling capabilities (Model integrity analysis (i.e. the ability to check the syntax of the model and find syntactical errors), model flexibility (i.e. the ability to model processes from different domains such as manufacturing, insurance, etc.), and level of detail, etc.).
- Simulation capabilities (handling of time and cost aspects and statistical distributions).
- Output analysis capabilities.

Oakshott (1997) outlines modeling flexibility, ease of use, animation, general simulation functions, statistical functions, interface with other software, product help and support, price and expandability as main features of simulation tools.

Hlupic, Paul, & Irani (1999) listed more than 310 criteria for the evaluation of simulation software. The criteria are divided into 13 groups: general modelling features, visual aspects, coding aspects, efficiency, modelling assistance, testability, software compatibility, model input/output, experimentation facilities (multiple runs, warm-up period), statistical facilities, user support, financial and technical features and pedigree (i.e. the history and reputation of the software and vendor which could be an indication of its reliability).

Nikoukaran et al. (1999) argues that features related to vendor and user also need to be examined beside software features. They classified features into seven groups:

- Vendor: pedigree, documentation, support and the possibility of demo or free trial.
- User: simulation type (discrete-event, continuous, etc.), hardware and operating system, required experience, etc.
- Model and input: model building, reusable models, statistical distributions, input, program generator (Generates the code of the built model in order to facilitate modification).
- Execution: multiple runs, warm-up period, etc.
- Animation
- Testing and efficiency: tracing, validation and verification.
- Output: integration with other software, reports, analysis, graphics.

According to Pidd & Carvalho (2006) the main capabilities provided by simulation packages are modeling tools, simulation execution, experimentation support, and links to other software. Modeling features consist of graphical modeling environment, simulation objects, visual controls, etc. simulation execution refers to running the model, animated graphics, and user interaction. Analysis tools and result interpretation and presentation are important aspects of experimentation support. The last feature is the links to other software like spreadsheets, databases and ERP systems.

Law and Kelton (2007) mentioned the following features and criteria for the analysis of simulation software: general capabilities (e.g. modelling flexibility and ease of use), hardware and software considerations, animation, statistical capabilities (including random number generator, probability distributions, and warm-up period), output reports (including reports for the estimated performance measures), customer support and documentation.

Jansen-Vullers and Netjes (2006) developed a framework for the evaluation and comparison of BPS tools. Their framework is based on the framework suggested by Bradley et al. (1995) and it consists of three groups of evaluation criteria:

- modeling capabilities (ease of model building, verification of correctness, level of detail, transparency and suitability for communication, etc.)
- simulation capabilities (performance dimensions (quality and flexibility), distributions, animation, scenarios)
- output analysis capabilities (statistics, format, what-if analysis, conclusion-making support)

Bosilj-vuksic et al. (2007) proposes a guideline for the evaluation of BPS software. Their guideline is based on the comprehensive framework for the evaluation of simulation tools developed by Hlupic et al. (1999) which was focused on manufacturing simulation. They adapted the framework to the domain of business process simulation. The author proposes to evaluate BPS software based on four main categories: hardware and software features, modeling capabilities, simulation capabilities, and input/output capabilities. Each category is subdivided to subcategories.

Regarding hardware and software features following aspects are considered:

- Coding aspects (programming flexibility, access to source code, global variables, built-in functions, and support of programming concepts).

- Software capability (integration with spreadsheet packages, integration with statistical packages, integration with DBMS, integration with legacy, applications, ERP, and integration with WFM systems)
- User support (documentation and tutorial, consultancy, training course, package maintenance, and demo models, and libraries)
- Financial and technical features (pricing and total cost of ownership, frequency and comprehensiveness of update, and portability)
- Pedigree (age, spread, reputation of supplier, and availability of references)

Aspects regarding modeling capabilities are as follows:

- General features (experience and education required for software use, ease of learning, user friendliness, formal logic, simulation modelling approach (process based, activity based, etc.)
- Modeling assistance (documentation notes, on-line help, modularity, model and data separation)

Features included in the guideline concerning simulation capabilities are:

- Visual aspects (animation, type of animation, animation with visual clock, expressiveness and quality of graphics, graphic library)
- Efficiency (robustness, level of detail, model reusability, model reliability, time scale for model building, model chaining: linking outputs from different models, Queuing policies)
- Testability (logic checks, error messages, ease of debugging, trace files, step function (event to event jumping), dynamic display of elements (capacity, events, state), display of the workflow path
- Experimentation facilities (warm-up period, breakpoints, speed adjustment, automatic determination of run length, automatic batch run)

- Statistical facilities (theoretical statistical distributions, user-defined distributions, random number streams, output data analysis, quality of data analysis facility, distribution fitting, confidence intervals)

Regarding input/output functionalities the following sub-categories are considered:

- Input and output capabilities (input data reading from files, quality and understandability of output reports, user-defined output, periodic output of simulation results)

Analysis capabilities (what-if-analysis, conclusion-making support, optimization)

Table 2.2. outlines a summary of all criteria and sub-criteria mentioned in the literature. As can be seen, some criteria are mentioned frequently in most papers, but some other criteria are only mentioned in few papers and are neglected by others. This is mostly due to the differences in the research objectives and goals. Two different types of literature with two different objectives have been recognized in this part. First are those which attempt to provide a comprehensive framework of criteria and desired features. These works have suggested large sets of criteria and sub-criteria. Second are those works focused on the most important criteria and desired features. They kept their framework narrower in order to evaluate and compare tools on these criteria. In addition, a few of these works only focused on functionalities of the software itself and failed to consider criteria focusing on users or the software vendor (e.g. documentations and user helps). Furthermore, some works focused on less technical aspects and ignored criteria such as programming and coding capabilities.

Criteria regarding statistical facilities, input and output capabilities and integration with other software packages have gained a lot of attention and have been mentioned in most papers. The attention on these criteria along with graphical capabilities and animation indicates an emphasis on features regarding the simulation capabilities and result interpretation facilities. Other frequently mentioned criteria such as user support and existence of proper documentation as well as ease of use indicates the importance of features concerning the user. Modeling flexibility and coding aspects have also been mentioned regularly in the literature and their importance has mostly been specified in modeling complex systems.

As outlined in a questionnaire carried out by Mackulak et al. (1994) to a group of qualified simulation practitioners, the most important simulation features were user-friendliness, input data capabilities, and existence of an interactive debugger for error checking. The survey conducted by Melao and Pidd (2003) among practitioners engaged in modelling activities in business process improvement programs outlines the most desired features of BPS tools. These features are easiness of use, flexibility to use in different application areas and projects with different purposes. Hlupic (2000) conducted a survey on the use of simulation software tools among members of the Simulation Study Group of the Operational Research Group of Great Britain, both from industrial and academic institutions. In this survey the main features of simulation tools as stated by both academic and industry users were ease of model development and visual capabilities. The main limitation of tools for industrial users were stated as lack of flexibility, the lack of links with other packages and the lack of interfaces for data input. Perera & Liyanage (2001) identified finding the balance between ease of use, flexibility and data input capabilities as an important problem. As process modeling is mostly in a context of business rather than a technical context, tools need to be easy-to-use for non-technical users but at the same time provide sufficient modeling capabilities.

These surveys on desired features connects with the findings from table 2.2 by the emphasis on user related criteria, such as user friendliness and ease of use, input and output related criteria and modeling criteria. This shows the importance of these groups of criteria for evaluation of simulation tools. These observations from the literature and the surveys conducted on the desired features, outline user related features among with flexibility, input capability, output analysis and modeling capabilities as the important features to consider for simulation tools evaluation.

In addition, one important group of criteria which has not been considered by previous literature on evaluation of simulation packages is the modeling tasks which are supported by the software. As outlined by Martin et al. (2016), there are different modeling task that need to be modeled while simulating business processes. These modeling tasks such as queue discipline and resource schedule, represent important characteristics of the system and if modeled properly the simulation result will be more precise and realistic.

		Pidd (1992)	Davis & Williams (1994)	Bradley et al. (1995)	Oakshott (1997)	Hlupic et al. (1999)	Nikoukaran et al. (1999)	Pidd & Carvalho (2006)	Law and Kelton (2007)	Jansen-Vullers & Netjes (2006)	Bosilj-vuksic et al. (2007)
Coding aspects	Access to source code	x				x					x
	Programming flexibility					x					x
	Expandability				x						
	Programming language	x		x		x					
	Compiler	x									
	Program generator					x	x				
Input	Integration with other software		x	x	x	x	x	x			x
Statistics	Statistical distribution	x	x	x	x	x	x		x	x	x
	Random number generation	x				x			x		x
Output	Output analysis			x		x	x	x	x	x	x
	Graphics		x			x	x	x			
	Reports	x				x	x	x	x		x
	Links to other software		x	x	x	x	x	x			x

Table 2.2. Simulation software evaluation criteria mentioned in the literature

		Pidd (1992)	Davis & Williams (1994)	Bradley et al. (1995)	Oakshott (1997)	Hlupic et al. (1999)	Nikoukaran et al. (1999)	Pidd & Carvalho (2006)	Law and Kelton (2007)	Jansen-Vullers & Netjes (2006)	Bosilj-vuksic et al. (2007)
	What if analysis									x	x
Vendor	Cost		x		x	x					x
	Pedigree					x	x				x
	Demo or free trial					x	x				x
	User support	x	x		x	x	x		x		x
	Further development		x			x					x
	Training		x			x					x
	Documentation		x	x		x	x		x		x
	Online help			x		x					x
	Instruction manual		x	x		x					
	Hardware and installation		x			x	x		x		
User	Ease of use		x		x	x			x	x	
	Expertise required		x	x			x				x
	User friendliness			x		x					x

Table 2.2. Simulation software evaluation criteria mentioned in the literature (continued)

		Pidd (1992)	Davis & Williams (1994)	Bradley et al. (1995)	Oakshott (1997)	Hlupic et al. (1999)	Nikoukaran et al. (1999)	Pidd & Carvalho (2006)	Law and Kelton (2007)	Jansen-Vullers & Netjes (2006)	Bosilj-vuksic et al. (2007)
	Ease of learning					x					x
	Time needed to build a model		x			x					
	Graphical user interface			x		x					
	Type of application	x				x	x				
Modeling	Model integrity			x							
	Model flexibility		x	x	x				x	x	
	Modeling assistance					x					
	Model building					x	x			x	
	Graphical modeling environment							x			
	Handling of time and cost			x		x					
Simulation	Simulation objects					x		x			
	Visual controls					x		x			
	animation				x	x	x	x	x	x	x

Table 2.2. Simulation software evaluation criteria mentioned in the literature (continued)

		Pidd (1992)	Davis & Williams (1994)	Bradley et al. (1995)	Oakshott (1997)	Hlupic et al. (1999)	Nikoukaran et al. (1999)	Pidd & Carvalho (2006)	Law and Kelton (2007)	Jansen-Vullers & Netjes (2006)	Bosilj-vuksic et al. (2007)
Experimentation facilities	Multiple runs					x	x	x			
	Warm-Up period					x	x		x		x
Efficiency	Reusable models					x	x				x
	Level of details					x				x	x
Testability	Tracing					x	x				x
	Validation and verification					x	x			x	x
	Debugger					x					x

Table 2.2. Simulation software evaluation criteria mentioned in the literature (continued)

2.3. Methodologies and guidelines for the evaluation, comparison and selection of simulation tools

For the selection of the suitable tool, the objective of the project should be considered as different projects have different requirements and therefore, different simulation features are needed. To address this, features can be assigned different weights of importance regarding the project in hand. For example, if process improvement and alternative evaluation is the objective of the project, simulation capabilities, testability, statistical and experimentation facilities will be important (Bosilj-vuksic et al. 2007). For rapid modeling, which means that the model should be developed quickly and without too many details, ease of learning the tool, ease of use, visual features and animation are the most important features. However, for detailed and comprehensive modeling of complex systems the most relevant feature are those regarding flexibility, programming and coding possibility, and links to the databases for integrating large amount of data. (Hlupic et al. 1999)

Banks et al. (1991) suggests a scheme for the evaluation of simulation packages by scoring them based on features and criteria. First, assign a score from 0 to 10 to each feature, then, software is scored a number between 0 and 1 on each feature. Williams and Trauth (1991) used a similar weighing approach for the ranking of 30 manufacturing software packages. Giving every criterion a weight from 0.1 to 1 and every feature of each software a score from 1 to 10. Then, they used Analytic Hierarchy Process (AHP) to identify the best of the top three software indicated by previous evaluation. Davis and Williams (1994) also used Analytic Hierarchy Process (AHP) for the evaluation and selection of simulation software. In their AHP approach, they compared all the criteria pairwise and assigned them a number on their level of importance. Then alternative packages were also compared pairwise against each criterion. A total score was assigned to each tool by multiplying weight of each package on each criterion by weights of the associated criterion. Finally, the alternative with the highest score was assumed as the most suitable. Gupta et al. (2010) also used a similar AHP approach for the comparison of four manufacturing simulation software, NX-IDEAS, StarCD, Micro Saint Sharp and ProModel. The same approach has been used by Otamendi et al. (2008) to select a simulation tool to be used for the daily allocation of parking spots in an airport. They compared Arena, Wittness and a specific tool developed in Java and chose the specific tool as the most suitable based on the result of their AHP framework. Ereeş et al. (2013) used AHP to rank and determine the necessary criteria for selecting simulation software for educational purpose. The result indicated the ability of programming, reporting and modelling as the most important criteria to consider when selecting a simulation tool for the aim of education. Hlupic and Paul (1995) also rated simulator's quality by scoring groups of criteria from 1 being poor quality to 10, excellent quality.

SimSelect is a system developed by Hlupic & Mann (1995) to assist users in the selection of suitable simulation software. It includes a database and a user interface. The database stores information on evaluation criteria of some simulation tools. It suggests a suitable software and a recommendation of alternatives based on the requirements specified by the user. The criteria used by SimSelect consist of following groups: General features, visual aspects, coding aspects, efficiency, modeling assistance, testability, input/output, software compatibility, experimental facilities, statistical facilities, and financial and technical features. In order to remain objective, they only indicate if the simulation software offers a certain feature or not, rather than rating them by quality of each feature. The user selects the desired requirements and indicates a "priority rating" to the selected features ranging from high priority to medium and low priority. A query is then generated and is divided in two parts. First the General Features, including type of simulation and cost are checked and those tools satisfying these requirements are selected. For example, if the user has a budget constraint and is searching for a DES tool, those tools offering DES and costing less than the user budget constraint are chosen for the next round. Then the second half of the query related to the remaining required features are executed and tools matching these criteria are selected by checking if the tools include the desired features. The system makes sure that the criteria marked as 'High Priority' are offered by the suggested tool.

Smart Sim Selector is a software providing support for users when selecting simulation software. Smart Sim Selector consists of a database which is linked to an interface. At the moment, the database holds information related to the evaluation details of 11 packages. The system queries the database and finds a suitable simulation package based on user requirements. First, user should outline desired criteria and indicate their level of importance ranging from very high to low. Then the system generates a query. For the selection of the most suitable simulation tool, Smart Sim Selector uses AHP and TOPSIS (Technique of Order Preference by Similarity to Ideal Solution). TOPSIS is a multi-criteria decision analysis method in which two alternatives are considered. A positive ideal alternative (the one which has the best level for all attributes considered) and a Negative ideal alternative (the one which has the worst attribute values). Then the alternative that is the closest to the positive ideal solution and farthest from the negative ideal alternative is selected as the best option (Gupta 2014).

Tewoldeberhan et al. (2002) introduces an evaluation and selection methodology for discrete-event simulation tools consisting of two phase. The first phase is a 'feature check' phase. In this phase a list of required features for the specific project is made by looking at previous projects and literature. A long list of tools is checked for the existence of these features. The required features are then weighed based on level of importance by using questionnaires answered by team members. Tools are scored on each criteria to 1 if they provide the feature and 0 if they do not. Then the top 10 highest scoring tools are selected for thorough evaluation by multiplying the weights and scores. The outcome of this list is a short list of potentially suitable tools. In the phase two called 'quality check', a thorough evaluation of the top 10 tools is performed. A case study should be created and packages should be tested on this case study. Based on the outcome, simulation packages are ranked and the most appropriate package is selected.

Jansen-Vullers and Netjes (2006) compared a number of tools relevant for the purpose of BPS and evaluated them based on their framework. They scored each tool on each of the evaluation criteria ranging from good, neutral to, bad and compared them based on their score. The tools were selected from three different categories: Business Process Modeling tools, Business Process Management tools and General Purpose Simulation tools. The compared tools were Arena and CPN Tools as general purpose simulation tools, Protos and Aris as business process modeling tools and FLOWer and FileNet as BPM tools. In their conclusion they stated that both selected business process management tools fell short on their simulation capabilities. On the other hand, both general purpose simulation tools and one of the process modeling tools, Aris, were suitable for BPS as they provided relevant and adequate modeling and simulation features.

Verma et al. (2008) present a guideline, a comprehensive framework and a methodology for the selection of simulation software. They divide 205 evaluation criteria in 4 main groups: hardware and software features, modeling capabilities, simulation capabilities and input/output features.

Their tool selection methodology consists of six stages to assist the tool selection process for users. These stages are as follows: Identifying the need for purchasing simulation software, initial software survey, evaluation, software selection, negotiating software contract, and software purchase. First, the intended purpose of simulation, available resources, time and budget and information about the modeler should be established. According to the determined initial elements a short list of suitable tools should be made. For example, if discrete-event simulation is required and the budget is limited, a short list of discrete-event simulation tools in the budget constraint is made. Based on the output of this stage, evaluation of candidate tools is performed using the suggested framework. The criteria should be ranked on importance regarding the project in hand and the tools should be compared based on these criteria. Then an appropriate software is selected and negotiation for a contract with the vendor takes place and finally the software is purchased.

Many authors have used scoring and weighting methods to compare simulation tools. From these different weighing approaches AHP has been used in a considerable number of conducted comparisons. Most of these comparisons lack clarity and transparency. They do not provide a simple guideline for re-conducting the comparison or reproducing the same result. To overcome this, a template for performing the comparison is provided in this thesis. This will clarify the comparison method used and will also provide a guideline for users for performing the comparison based on their project requirements.

3. Criteria and methodology

This chapter introduces the required elements for conducting the comparison of BPS tools. These elements are:

- The alternative simulation tools to be compared,
- A set of evaluation criteria on which the alternative tools are compared,
- The methodology to compare the alternatives on the criteria,
- A template developed for performing the comparison.

In section 3.1 the selected simulation tools for performing the comparison and a justification of their selection are outlined. Section 3.2 outlines the selected evaluation criteria to be used for simulation tools comparison. Section 3.3 explains AHP, the methodology used for comparison in this thesis. Finally, in section 3.4 the developed spreadsheet for comparison of BPS tools is explained.

3.1. Selected simulation tools for comparison

Given the large number of simulation tools, of which a non-exhaustive overview is presented in Section 2.1, a comparison of all tools is beyond the scope of this dissertation. Consequently, this subsection outlines the simulation tools that will be included in the comparison. According to the objective of these thesis on comparison of three groups of simulation tools for BPS purposes, following tools are selected from these three groups:

1. Dedicated simulation tools
   1.1. General purpose simulation tools
      - Arena
   1.2. Business process simulation tools using BPMN
      - Scylla

2. Business process management tools with simulation functionalities
    - Signavio
    - Bizagi
3. Simulation packages in programming languages
    - Simpy

Dedicated simulation tools:

Two tools are selected from the group of dedicated simulation tools. According to the further division in this category, one tool, Arena, is selected as a general purpose simulation tool while Scylla is chosen as BPMN simulation tool. Arena is one of the most known simulation software packages. There is an important number of articles and scientific papers focused on Arena and a variety of use cases exist. The selection of Scylla is due to its focus on BPS and specially using BPMN. In addition, it is a new tool and uses a plug-in approach to simulation which offers expandability and suggest prospect in further development. Furthermore, the selection has been as such that one tool, Arena is commercially available while Scylla is an open source tool.

Business process management tools with simulation functionalities:

Two tools are chosen as business process management tools which provide simulation capabilities: Signavio and Bizagi. These are two of the main BPM software vendors and have gained considerable reputation in recent years. Signavio is commercially available while Bizagi Modeler which is a part of Bizagi Suit and contains simulation is available free of charge.

Simulation packages in programming languages:

From the last group, simulation packages in programming languages, we chose Simpy as one of the most important and powerful tools in this category. A number of other simulation packages for other programming languages has been developed by re-implementing SimPy and its libraries.

3.2. Comparison criteria

Based on literature on evaluation and comparison criteria of simulation tools mentioned in the previous chapter, we draw our set of criteria for the purpose of BPS tools comparison. In this thesis we consider the important features of every aspect of simulation tools in order to keep the set of criteria simple while covering a comprehensive range of features.

The criteria are grouped in five main categories each containing sub-criteria. The five main categories are features regarding coding and programming capabilities, simulation and modeling capabilities, input and output considerations, user and software vendor considerations, and supported modeling tasks. Some of the criteria mentioned in table 2.2 are grouped together and some are placed inside other categories as sub-criteria. For example, statistics and experimentation facilities are place inside the simulation and modeling capabilities as they can be considered as part of the simulation functionalities that are provided by simulation tools to run and experiment with the simulation model. Our criteria set is as follows:

- Coding aspects
    - Access to source code
    - Program generator
    - Expandability
- Simulation and Modeling aspects
    - Animation
    - Statistical facilities
    - Modeling notation
    - Warm-up period

- Input/Output
  - Integration with other software
    - Input models
    - Input data
    - Output result
  - Output analysis
    - Graphical representations
    - What-if analysis
- User/Vendor
  - User friendliness
    - Graphical user interface
    - Ease of use
    - Ease of model building
  - Expertise required
  - Ease of learning
    - Documentation
    - Tutorials
    - Online help
    - Training
  - Price
- Supported modeling tasks

3.2.1. Coding Aspects

The possibility of additional coding and programming is an important aspect of simulation tools. It defines the flexibility and strength of the tool to model complex systems in different application areas and domains. In this category we consider the possibility to access the source code of the tool, program generator and expandability as sub-criteria.

Access to source code identifies the possibility to access and modify the source code of the tool. This will be useful in order to add functionalities or adjust the tool to specific requirements. In addition, the possibility to access the source code allows further development of the tool to be made not only by the software vendor itself but also by expert users and researchers. In this way, the community of users can contribute to further development of the tool (Nikoukaran et al. 1999).

program generator provides the underlying code of the created simulation model and enables modification and further development of the model in order to have a more tailor made model of the system and its specifications (Nikoukaran et al. 1999).

Expandability defines the ability to expand the existing version of the tool. This can be useful for future developments, adding missing functionalities, and extending the tool toward specific requirements (Hlupic et al. 1999; Nikoukaran et al. 1999).

3.2.2. Simulation and modeling aspects:

In this group of criteria, the criteria regarding building the simulation model, running the simulation, and experimenting with the created model are considered. The existence and quality of animation, statistical facilities and probability distributions, the supported modeling notations, and the ability to define warm-up periods are considered.

Animation is the graphical presentation of the instances and their run through the system. It enables users to watch a simulation being executed step by step and track the instances from the moment they enter the system, when they wait in a queue, up until they leave the system. This enables better and easier understanding of the system and communication; (Nikoukaran et al. 1999).

***Statistical distributions*:** Due to the randomness that is present in the majority of simulation models (e.g. in the customer arrival rate), the tool should provide standard statistical distributions to be assigned to the variables of the model. The number of statistical distributions (e.g. normal, exponential, triangular, etc.) available in the tool is considered as a criterion for comparison. In addition, the ability to fit the input data into a distribution automatically by the tool and the ability to add user-defined distributions are considered (Hlupic et al. 1999; Nikoukaran et al. 1999).

The modeling notation that the tool uses for building the model and supports to input the models to the tool is important while simulating business processes. There are different modeling notation and there are different preferences by users and experts toward these notations. Regarding simulation on business processes, it is important to consider whether or not the tool supports one of the business process modeling standard notations such as BPMN, as this will make the model building easier and faster for BPMN users familiar with these notations.

Warm-up period is the time that the simulation will run without collecting statistics until it reaches a steady state. Defining warm-up periods allows the system and its queues to reach the real life conditions. This means that rather than considering the system empty at starting point the simulation will start from a real life state in the system when queues and accumulated work from the past might already exist. (Hlupic et al. 1999; Nikoukaran et al. 1999).

3.2.3. Input and output capabilities

Integration with other software: The simulation software can integrate with other software packages such as database management systems (DBMS), spreadsheets, legacy systems, ERP, statistical packages, etc. to import and export models and data. Here we consider the ability to import models and data and export the simulation result (Bosilj-vuksic et al. 2007).

Input models: The ability to import previously built models to the system from other software packages and integrate the model for the simulation run can ease the simulation process.

Input data**:** Simulation data can be retrieved from database, spreadsheets or other software packages to the simulation tool to be used for model building. The existing data can be used, for example, to define the probability distributions of different variables in the system. For example, from data gathered on customer arrival rate, appropriate distribution can be determined to be assigned to this variable. The ability to import and handle large amount of data can assist the model building of complex systems (Nikoukaran et al. 1999).

Output result*:* The output data can be stored in databases and exported to other tools such as spreadsheets and statistical packages to be further analyzed (Nikoukaran et al. 1999).

Output analysis: Simulation tools can possess the ability to analyze the result and assist the interpretation of the output data. Statistical functionalities can be used to analyze the output data on performance measures (Hlupic et al. 1999).

Graphical representation: The ability to present simulation results graphically with histograms and charts, can assist result interpretation and conclusion making (Nikoukaran et al. 1999).

what-if analysis to compare the output of different scenarios and alternatives and examine the effect of different scenarios on performance measures facilitates the interpretation and conclusion making of the result (Hlupic et al. 1999; Nikoukaran et al. 1999; Bosilj-vuksic et al. 2007).

3.2.4. User/Vendor considerations

In this category we focus on features concerning users and available support and assistance by the software vendor.

User friendliness is generally considered as an important aspect of every software package and simulation tools are no exception. The ease of use, the graphical user interface provided for the user to simplify the interaction of user with the system can be considered as essential aspects of user friendliness. We evaluate user friendliness of software packages by assessing its ease of use, the quality of the interface, the ease of building the model and importing data and the general feel of the use of the tool. Furthermore, the level of expertise and prior experience in simulation needed to build the model and the time needed for new users to learn the package and build their first model is considered.

Graphical user interface (GUI) eases the interaction with the software by providing predefined icons and visuals instead of text or command interaction (Hlupic et al. 1999).

Ease of use refers to the general use and interaction of the user with the tool to perform different functions in the tool environment.

Ease of model building specifically assesses the modeling environment of the tool and the ease of creating different elements of the model, connecting them and assigning attributes and simulation data to the elements.

The ease of learning is assessed by looking at the quality of ***documentation, tutorials, online-help,*** user support and availability of ***training*** by the vendor company.

price of the simulation tool is important to be considered as there are variety of tools available with variety of prices ranging from free tools to highly priced commercial tools. (Hlupic et al. 1999; Nikoukaran et al. 1999; Bosilj-vuksic et al. 2007).

3.2.5. Supported modeling tasks

We argue that one important criterion that needs to be taken into account when selecting an appropriate BPS tool, in addition to the mentioned criteria by previous literature, is the modeling tasks that are supported by the tool. This criterion has not been considered in the prior literature on evaluation and comparison of BPS tools. Modeling these tasks will make the simulation result more reliable and similar to the real system as it incorporates important characteristics and conditions of the system. Different projects might require considering different modeling tasks. Decision makers can quickly identify tools supporting their required modeling tasks and rule out the tools in which the required task is not supported.

When simulating business processes there are different components of the process which need to be considered while building the model. The eight main BPS model building blocks according to Martin et al. (2016) are entities, activities, resources, queues, sequence flows, gateways, resource roles and schedules. According to the relation between these components, Martin et al. (2016) combine them into four groups:

- ***Entities:*** Entities are instances that enter the system, activities are performed on them and finally leave the system, e.g. a received order from the customer.
- ***Activities (activity, queue):*** Activities are services and tasks performed to serve the entities, e.g. availability check of an order.
- ***Resources (resource, resource role, schedule):*** Resources are required elements to perform the activities and provide the service to entities, e.g. a sales clerk.

- ***Control flow (sequence flow and gateway):*** Defines the relationship and ordering of activities in a process. It is the logical flow that the entities follow to be served.

Martin et al. (2016) identify fifteen different modeling tasks for the aggregated BPS building blocks for the purpose of simulation modeling of business processes. These modeling tasks with a brief explanation are outlined in table 3.1.

Modeling Component	Modeling Tasks	Definition
Entities	Entity attribute	Characteristics of entities are known as entity attribute, e.g. the number of items in a received order (Kelton et al. 2010).
	Entity type	An entity type is used to group entities with similar attributes, e.g. domestic orders and international orders. Modeling entity types simplifies model building by only specifying behavior of each entity type rather than each attribute. This allows calculation of performance measures on the entity type level (Kelton et al. 2010; Martin et al. 2016).
	Entity arrival rate	The time interval between arrival of entities to the system. It can be defined by a fixed number or a probability distribution. The entity arrival rate to the system has a significant impact on process performance measures such as the average waiting time and queue length (Martin et al. 2016).

Table 3.1. BPS modelling tasks

Modeling Component	Modeling Tasks	Definition
Activities	Activity definition	Defining the activities in the system, e.g. availability check of an order (Martin et al. 2016).
	Duration	The execution time of activities. It can be modeled fixed or conditional on attributes. For example, the duration of the availability check activity is affected by the number of items in an order (Martin et al. 2016).
	Resource requirements	The required resources and their quantity for an activity to be executed (Martin et al. 2016).
	Queue discipline	In case that the required resource is occupied by another entity, the entities have to wait in a queue for the resource to be available. The structure of these queues should be modeled. The typical queue disciplines are first-in-first-out (FIFO), last-in-first-out (LIFO) and priorities (Kelton et al. 2010; Martin et al. 2016).

Table 3.1. BPS modelling tasks (continued)

Modeling Component	Modeling Tasks	Definition
	Queue abandonment condition	The condition under which the entities leave the queue before being served either to join another queue or to completely leave the process. (Martin et al. 2016).
	Interruptibility and Unexpected interruption	The interruptibility indicates if it is possible to interrupt an activity during execution. For example, when a break starts during picking an order. In addition, the occurrence frequency and duration of unexpected interruptions (e.g. breakdown of a machine), need to be modeled (Martin et al. 2016).
Control flow	Control flow definition	Sequence flows and gateways are used to specify the movement of entities in the process (Martin et al. 2016).
	Gateway routing logic	The branching rules in gateways. The branching probability needs to be modeled (Martin et al. 2016).

Table 3.1. BPS modelling tasks (continued)

Modeling Component	Modeling Tasks	Definition
Resources	Resource roles	Resources that can execute similar activities can be grouped in a resource role (Martin et al. 2016).
	Resource schedule	The time tables of resources during which they are active and can perform activities on the entities. Two types of schedules, one for current resources and one for new resources based on resource roles, need to be specified (Martin et al. 2016).
	Unavailability handling procedure	The unavailability handling procedure is used to define how periods during which required resource is unavailable are processed (Martin et al. 2016).
	Entity handling procedure	The number of entities that a resource can handle at the same time in case of accumulation of similar works (Martin et al. 2016).

Table 3.1. BPS modelling tasks (continued)

3.3. Comparison methodology

Selecting an appropriate business process simulation tool is a multi-criteria decision making problem which involves the assessment of many different criteria. In addition, comparing software packages can be influenced by personal judgment and preferences. Therefore, a multi-criteria decision making technique with the ability to control the human judgment is needed. We use the Analytic Hierarchy Process (AHP) for the comparison and selection of BPS tools. AHP is a multi-criteria decision making approach developed by Thomas L. Saaty (1994). It Takes the human judgement into consideration and attempts to decrease the inconsistencies in human judgement (Gupta et al. 2010). It divides the problem into sub-problems, and finally aggregates the solution of sub-problems into final conclusion (Saaty 1994; Gupta et al. 2010). It was originally used in socio-economic context, but it has been used in a variety of domains such as economics and planning (Emshoff and Saaty 1982), distribution of resources (Gholamnezhad and Saaty 1982), education (Saaty and Rogers 1976) as well as software selection and simulation software selection by Williams and Trauth (1991), Davis and Williams (1994), Otamendi et al. (2008), Gupta et al. (2010), Ereeş et al. (2013). This shows the applicability and value of using AHP in simulation software selection and comparison.

In comparing simulation tools with AHP, first all criteria are compared pairwise against each other on their relative importance. These comparisons are numericalized using Saaty's intensities of importance (Saaty and Roger 1976) shown in Table 3.2, from 1 meaning equally important to 9 indicating absolute importance. These values fill up the upper triangle of the criteria comparison matrix with 1 as diagonal element. The lower triangular will then be filled with the reciprocal entries. Once the matrix has been filled, the next step is to normalize the values by dividing them by the corresponding sum of the column. Then, the weight of importance for each criteria is found by calculating the average of each row.

The next step is to compare alternative simulation tools pairwise with regard to each and every criterion, using the same 1 to 9 scale from table 3.2. (1 meaning that the two alternatives perform equally on a specific criterion to 9 meaning one absolutely outperforms the other alternative). These values are then normalized in the same way and the weights of each alternative tool on each criterion is calculated.

Finally, the weights of each criterion is multiplied by the weight of each alternative on that criterion and the overall score for each alternative is calculated by summing up the result for every criterion (Saaty 1994; Gupta et al. 2010).

Intensity of importance	Definition
1	Equal importance
3	Weak importance
5	Strong importance
7	Very strong importance
9	Absolute importance

Table 3.2. Saaty’s intensities of importance

3.4. Comparison spreadsheet

For the purpose of comparing simulation tools, we have developed a template in Excel. We perform our comparison using this template. This template can also be used by other users to perform their own comparisons. The current version of the template uses the criteria mentioned in the section 3.2 of this thesis and compares the tools mentioned in section 3.1. However, it can be updated with new criteria and tools. The AHP comparison matrices of criteria and alternative tools are placed in different tabs. The first tab of the template as shown in figure 3.1 calculates the final result and creates the ranking of alternatives. In addition, additional input, the number of criteria used (26 in our comparison) and the number of alternative tools to compare (5 in this case), are mentioned in this tab. (The user needs to update these values in case of adjusting criteria or tools).

The second tab which is called “Criteria matrix” contains the pairwise comparison matrix of criteria. The user needs to fill in the upper triangle of this matrix by assigning the relative importance of one criterion against another by using the values in table 3.2, intensities of importance. These values should be assigned either as integer numbers or fractions (such as 1/3, 1/9). Fractions means that the considered criterion has lower importance than the other criterion. The lower triangle then will be filled as reciprocal entries. Figure 3.2 shows a fraction of this matrix.

The third tab which can be considered as the backend calculations, normalizes the values of the criteria matrix and calculated the total weight for each criterion.

Each of the following tabs (26 in this case for the 26 criteria) are dedicated to comparison matrices of alternative tools against one criterion. The numeric and fractional values need to be input to the matrices the same way as in criteria matrix. However, the meaning of these values is slightly different in a way that instead of importance, the performance of alternative tools on one criterion compared to the other alternative is considered. An instance of these matrices are demonstrated in figure 3.3.

Once all the tables are filled the template calculates the total weights of each alternative tool. The alternative with the highest weight is suggested as the most suitable tool based on user defined values for the criteria comparison matrix.

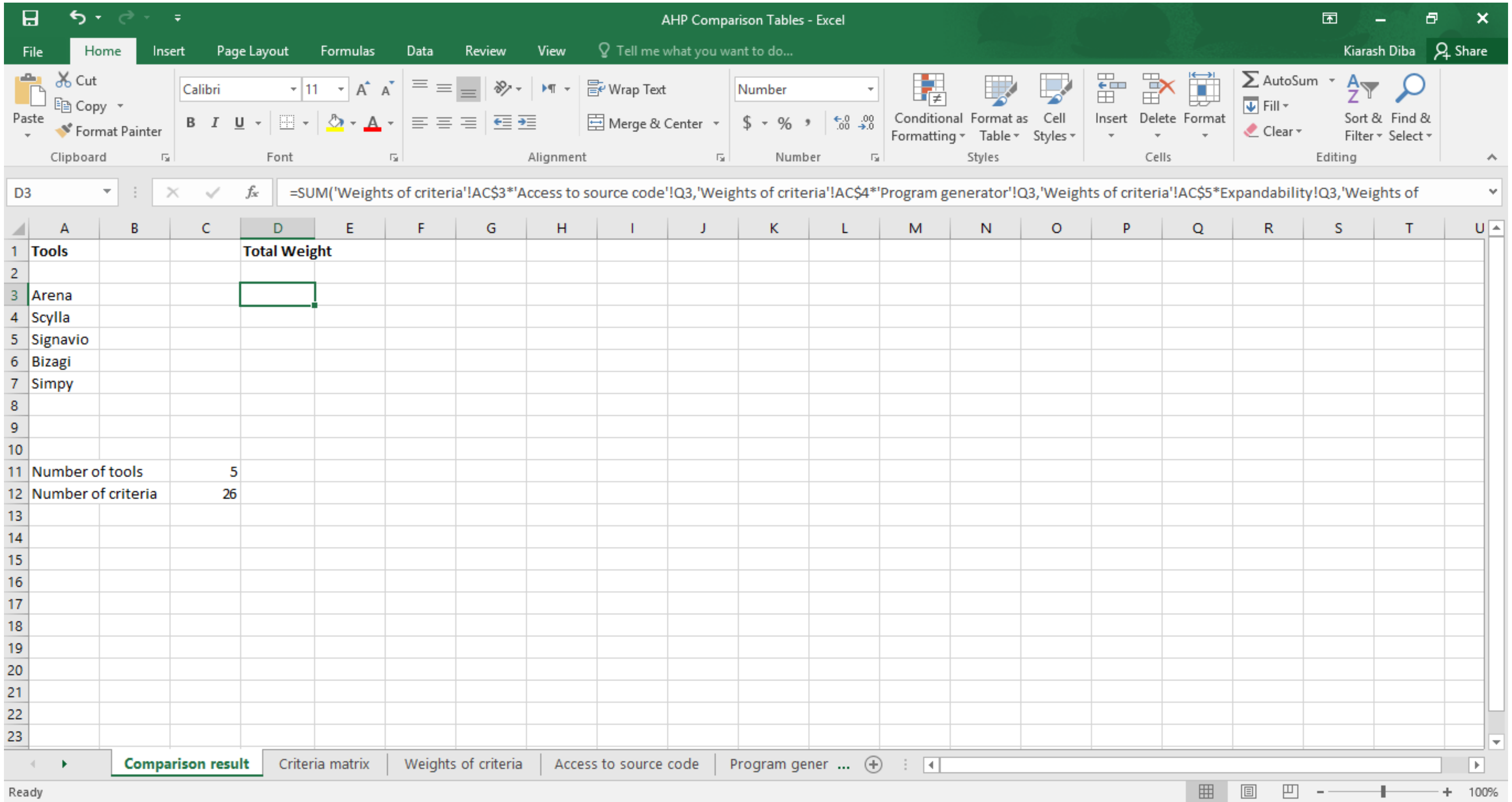
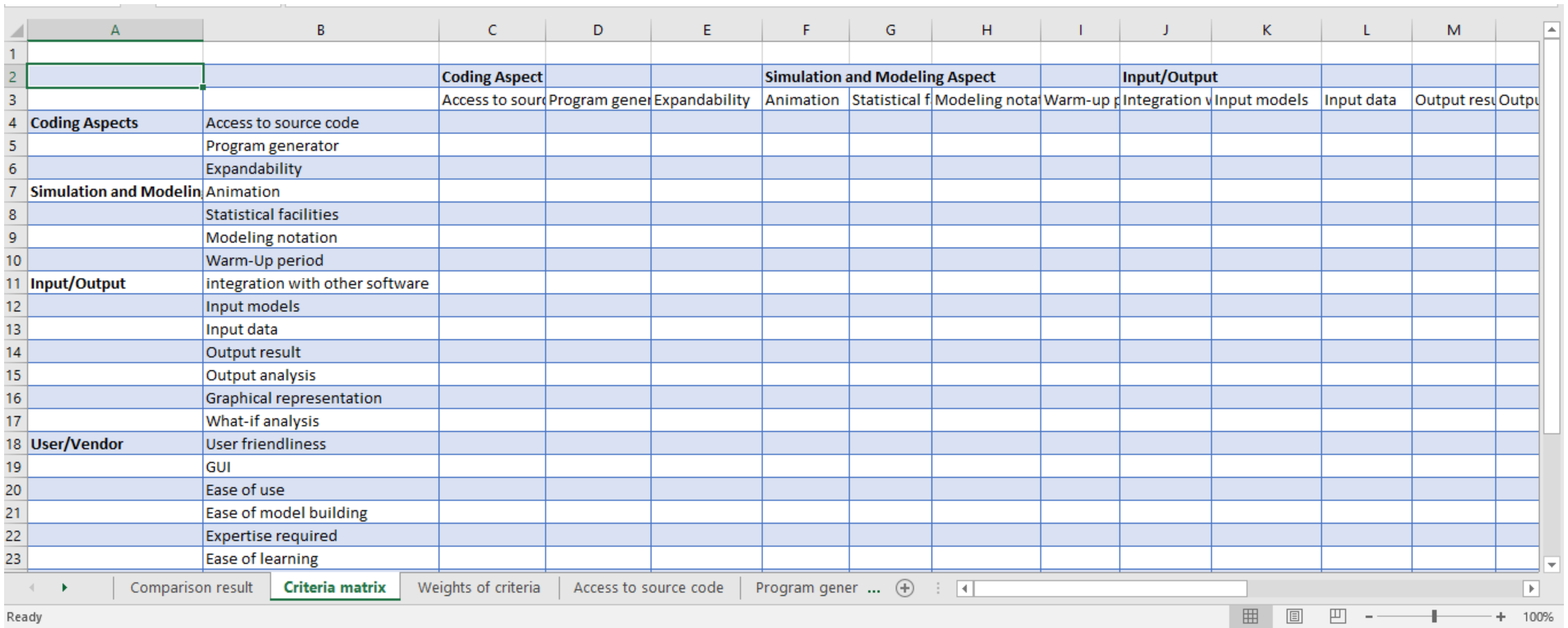


Figure 3.1: The comparison template - result tab



	A	B	C	D	E	F	G	H	I	J	K	L	M	
1														
2			Coding Aspect			Simulation and Modeling Aspect				Input/Output				
3			Access to sourc	Program gener	Expandability	Animation	Statistical f	Modeling nota	Warm-up p	Integration v	Input models	Input data	Output resu	Outpu
4	**Coding Aspects**	Access to source code												
5		Program generator												
6		Expandability												
7	**Simulation and Modelin**	Animation												
8		Statistical facilities												
9		Modeling notation												
10		Warm-Up period												
11	**Input/Output**	integration with other software												
12		Input models												
13		Input data												
14		Output result												
15		Output analysis												
16		Graphical representation												
17		What-if analysis												
18	**User/Vendor**	User friendliness												
19		GUI												
20		Ease of use												
21		Ease of model building												
22		Expertise required												
23		Ease of learning												

Figure 3.2: The comparison template – A fraction of the criteria matrix

	A	B	C	D	E	F	G
1	Access to source code						
2		Arena	Scylla	Signavio	Bizagi	Simpy	
3	Arena	1					
4	Scylla		1				
5	Signavio			1			
6	Bizagi				1		
7	Simpy					1	
8	**Total**	**1**	**1**	**1**	**1**	**1**	
9							

Figure 3.3: The comparison template - alternatives comparison

The spreadsheet developed and used in this thesis has the following advantages:

1. It is clear, simple and easy to use
2. It allows various assignments of weights to criteria based on the project requirements, available resources or the target user.
3. It can simply be update and extended with new tools and criteria.

In the next part, a guidance on using the template and how to update and adjust the criteria and alternative tools in the template is provided.

using the template:

To use the template and compare the five alternative tools considered in this thesis, user needs to compare the criteria and fill in the "Criteria matrix" based on the project requirements, preferences and available expertise. Based on these values the template calculates the most suitable tool.

Updating the template:

The template can be updated by modifying the set of criteria or the alternative tools. New criteria and tools can be added and the existing ones can be removed.

Adding new criteria:

In order to add new criteria, the following steps should be performed:

1. The new criteria need to be added in both "Criteria matrix" and the table in the third tab of the template "Weights of criteria".
2. The number of criteria (located in the first tab) should be adjusted.
3. For each of the added criterion a new matrix (preferably in a new tab) needs to be created and be filled with the values of pairwise comparison of tools against that criterion.
4. Finally, the formula in the result tab should be adjusted to include the multiplication of the total weight of the criterion by the weight of alternatives on that criterion.

Adding new tools:

In order to add new tools, the steps needed are as follows:

1. Each of the tool comparison tables in each of the tabs for each criterion should be adjusted with the new tools and the pairwise comparison with other tools should be performed.
2. The number of tools (located in the first tab) needs to be adjusted.
3. The name of the tool needs to be added in the first tab and the formula calculating the total weight needs to be extended for it.

*Deleting criteria or tools***:**

In order to delete criteria or tools similar steps to the above mentioned steps need to be taken only instead of adding rows and columns to the tables, the respecting rows and columns should be deleted from the tables.

4. Comparison result

This chapter evaluates and compares the performance of the selected tools with respect to each of the criterion outlined in Section 3.2.

4.1. Coding Aspects:

Access to source code: Among the chosen simulation tools in this thesis Scylla and SimPy are open source tools which allow access to their source codes. Arena, Signavio and Bizagi do not provide access to the source code. Therefore, Scylla and SimPy will be assigned a higher weight than the others in this sub-criterion.

Program Generator: In Arena the simulation code in the form of a simulation language, SIMAN (the underlying simulation language in Arena), is available to view and edit. None of the other tools support this functionality.

Expandability: In this category Scylla, being an open source application and using a plug-in approach, is assigned the highest weight in our comparison. Custom plug-ins can be written in JAVA and added to the software to be used for simulation for a specific purpose. SimPy is also expandable by further programming in Python. In Arena it is possible to write code in Visual Basic (VB) language. A VB coding environment is provided inside Arena and can be used for applications. Signavio and Bizagi do not support any functionality regarding expandability.

Overall, regarding the coding aspect which can be considered a good measure for flexibility of the tool, Scylla and SimPy score the highest and are considered as the most flexible tools. SimPy is a programming package and Scylla with its plug-in architecture allows for customization and expandability. Arena is also a flexible tool due to the nature of its modeling architecture which allows for flexible modelling, and access to the SIMAN simulation language and the VB programming possibility. However, Signavio and Bizagi, the two BPM tools, do not provide any programming possibility and expandability. Therefore, they are not considered as flexible regarding their simulation functionality.

4.2. Simulation and Modeling aspects:

Animation: Animation in Arena is the most comprehensive and has the best quality. It offers customizable 2D and 3D animation. It is possible to animate entities, resources and queues. Different types of entities and resources can be distinguished by custom animated objects. It is also possible to integrate some special animation packages to improve animation. None of the other tools considered are comparable in their animation capability to Arena. In general, it can be said that animation is best provided by the dedicated general purpose simulation tools. Signavio provides a basic walk-through animation, showing instances in different states of the process. It shows waiting instances, running instances and executed instances next to each state of the model. Bizagi, however, only shows the executed instances along with waiting times and executed times related to each activity in the model. Figures 4.1 to 4.4 illustrate simple examples of the different animations in Arena (2D and 3D), Bizagi and Signavio. Scylla does not offer any animation capability at the moment. SimPy does not offer animation but the simulation code written in SimPy can be used in computer graphic applications such as Maya, to create animations. This requires advanced knowledge of both graphics and programming. To sum up, Arena dominates this category.

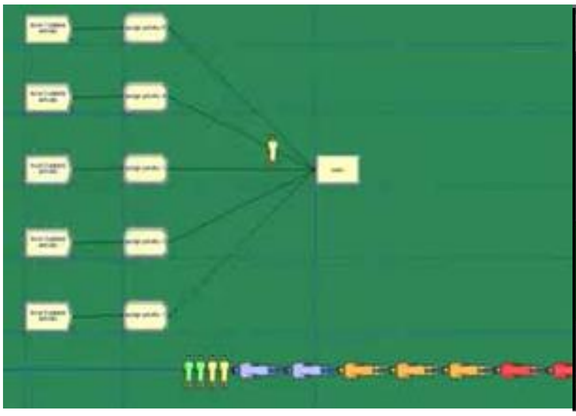


Figure 4.1. Animation in Arena (2D)

Figure 4.2. Animation in Arena (3D)

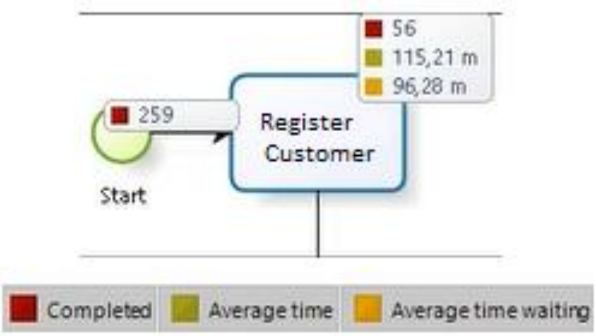


Figure 4.3. Animation in Bizagi

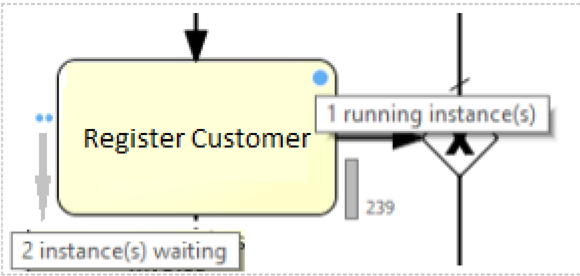


Figure 4.4. Animation in Signavio

Statistical facilities: Arena has a comprehensive set of statistics and distribution functions and also allows user-defined distributions. Bizagi has a sufficient predefined set of distribution functions but Signavio only supports normal and uniform distribution functions. In Scylla there are also sufficient predefined distribution functions to be used. SimPy does not contain statistical facilities in itself but there are plenty of Python packages such as NumPy to supplement SimPy. This category is again dominated by Arena.

Modeling notation: Both BPM tools support modeling based on the BPMN modeling notation. Signavio provides modeling of many BPM modeling standards but simulation is only possible on BPMN models. It is possible to transform some of these models like EPC (event-driven process chain) to a BPMN model in Signavio. Bizagi also uses BPMN standard for modeling. Scylla is also a BPMN simulation tool which works with BPMN compatible models. However, Arena uses its own tool-specific modeling notation. SimPy does not have an explicit modeling notation as it is a package in a programming language without explicit user interface used for model development. As a consequence, BPMN constructs such as gateways need to be converted to code statements. Therefore, in this category Signavio is the highest scored tool due to its BPMN compatibility and the possibility of transforming some standards to BPMN. Bizagi and Scylla scored high as well.

Warm-up period: Warm-up periods can be defined and implemented in Arena and SimPy. Scylla, Signavio and Bizagi do not provide this functionality at the moment.

Overall, despite the sub-criterion modeling notation, Arena performs the best in this group of functionalities possessing the most complete and powerful simulation and modeling functionalities.

4.3. Input/Output:

Integration with other software: The following sub-criteria are assessed as a measure for the general software integration of the simulation tools.

Input models: In Arena it is possible to import Visio flowcharts. Bizagi also supports importing Visio flowcharts along with BPMN models and XPDL (The XML Process Definition Language) which is a standard format for business processes to be used in different modeling tools. In Scylla BPMN files can be imported. In Signavio BPMN and XPDL files can be imported. SimPy obviously does not support any input model.

Input data: Only Arena provides the possibility of importing data from spreadsheets and databases. In all other tools simulation data should be assigned manually to the model.

Output result: In Arena, Bizagi, Signavio it is possible to export the simulation result (output measures) to Excel. Scylla only generates an XML-file as output containing the output logs and measures. In SimPy it is possible to export the output using other Python packages.

Output Analysis: Regarding output analysis, Arena provides capabilities to analyze the output data and to perform statistical analysis on the data. The output of the simulation in Signavio and Bizagi can be further analyzed after being exported. In SimPy the analysis can be done using other packages in python. The output of Scylla should also be analyzed using other software packages as there is no analysis capability provided.

Graphics: Arena provides very good graphics, charts, plots, etc. to visualize the simulation result and recorded KPI-values (e.g. average waiting times, cost, etc.). Signavio automatically creates charts of the simulation output exported to an excel file. Bizagi provides basic visualization on the output but it is possible to expand it in the excel output. No predefined graphical output is provided in Scylla. In SimPy, there are other Python packages such as Matplotlib to create histograms and charts.

What-if analysis: Defining different scenarios and comparing the simulation results based on different scenarios and performing a what-if analysis is supported in Arena, Bizagi and Signavio. In Scylla and SimPy the functionality to define what-if analysis is not provided. The user needs to create separate models and perform separate simulation runs and then compare the result manually.

Overall, Arena as the representative of dedicated general purpose simulation tools provides the most comprehensive and powerful functionalities regarding input and output capabilities. The quality of reporting, the ability to visualize and analyze the simulation output along with integration with databases to input large amount of data make Arena the dominant tool in this group of criteria. Following Arena, the two BPM tools provide more functionalities compared to Scylla and SimPy.

4.4. User/Vendor

User friendliness: We measured the user friendliness of the tools by considering the graphical user interface, ease of use and ease of model building. Arena, Bizagi and Signavio are identified as the user friendliest, followed by Scylla due to the lack of model building capability. SimPy is considered as the least user friendly tool due to the absence of user interface and the required coding for model building.

Graphical User Interface (GUI): All the four software tools, Arena, Bizagi, Signavio and Scylla provide a simple and intuitive GUI. but SimPy is a programming package in python and the interface is the python editor and no graphical interface is available. However, it can be integrated in a GUI. Within the GUI of Arena, the model can be build, with simple drag-and-drop of predefined elements from the left hand side project bar and simulation specific data and different attributes can be created and assigned. In Signavio and Bizagi, building a model is done in a similar way to Arena. The simulation editor opens in another view to define simulation data and create scenarios. The interface in Scylla only supports the definition of simulation configuration and information, but the construction of a model is not possible and models should be imported to the tool.

Ease of use: All the four software are easy to use in general, but in SimPy writing code is required.

Ease of model building: apart from the ease of model building concerning the modeling notations and the BPMN notation, in general building the model in Arena, Signavio and Bizagi is easy and is done by simple drag and drop. Scylla does not provide process modeling ability but simulation specification can be modeled rather easily. In SimPy coding is required.

Expertise required: For building the model in Arena user needs to be familiar with the modeling notation and different modeling modules in Arena. Modeling complex systems needs an intermediate to high level of expertise in simulation and can be difficult for a non-simulation expert. Modeling in Signavio and Bizagi requires knowledge of BPMN and defining the simulation specifications is rather easy. These tools are mostly aimed at business and process management users and do not require high level of simulation expertise. Working with predefined plug-ins and the currently available version of Scylla does not require expertise but if more functionalities are needed, creating plug-ins and adding to the tool requires programming knowledge. SimPy requires familiarity with Python programming along with high level of simulation expertise as a thorough understanding of simulation is required to be able to build a model with programming.

Ease of learning: Learning to perform business process simulation in BPM tools, Signavio, Bizagi and simulation tools using BPMN, Scylla is easy. In Arena, more effort is needed to learn the tool specific modeling. SimPy is identified as the most difficult to learn. SimPy users should be familiar with coding and specifically python language.

Documentation, Tutorial, Online help and training: The most comprehensive set of documentation, tutorials and online helps are available for Arena and SimPy. As there are online forums and communities of user to communicate, solve their problems and in case of SimPy play a role in the further development of the tool. For Bizagi and Signavio, sufficient and good quality documentation is available and learning the simulation functionalities is rather easy following these documentations. Bizagi performs slightly better due to availability of some video content and an active e-learning platform. For scylla there are not many documentation and tutorials available. In addition, Arena and Signavio provide further on-site trainings if desired by the company.

Price: SimPy and Scylla are the two open source and free tools. Bizagi Modeler, one part of Bizagi platform, which contains the simulation functionality is also available for free. However, Arena and signavio are commercial software tools which provide free trials. They also provide free access to students for educational purposes.

Overall, on one hand Arena and SimPy, score better in documentations and learning material. On the other hand, they require more simulation expertise and they can be more difficult to learn and use for users familiar with BPMN.

4.5. Supported modeling tasks

Table 4.1 indicates the modeling tasks supported by each simulation tool. As can be seen in the table 4.1 Arena supports modeling of all the modeling tasks involved in business process simulation. The two BPM tools, Signavio and Bizagi, and the BPMN simulation tool, Scylla, support the same tasks. They support the basic tasks required for the simulation and neglect the more complex ones needed for more detailed modeling, such as the queue discipline. Neglecting these modelling tasks and complexities of the system can influence the accuracy of the simulation model and the reliability of simulation result. This can be considered as one of the important shortcomings of BPMN tools, both BPM tools and simulation tools using BPMN. However, Scylla enables the possibility to expand and to include more modelling tasks by creating plug-ins. Plug-ins for each of the modeling tasks can be programmed and added to the tool. In SimPy, although only a small number of modeling tasks are directly supported and are predefined to be used in the models, all the modeling tasks can be modeled with programming efforts and writing code. However, this will require high level of expertise in programming. For evaluating SimPy on these criteria we consider the modeling tasks that are predefined in the package or there are documentations and instructions available for modeling them. The more complex tasks which are not mentioned in documentations are considered to be not supported for modeling.

Modeling Component	Modeling Tasks	Arena	Scylla	Signavio	Bizagi	SimPy
Entities	Entity attribute	X	-	-	-	X
	Entity type	X	-	-	-	X
	Entity arrival rate	X	X	X	X	X
Activities	Activity definition	X	X	X	X	X
	Duration	X	X	X	X	X
	Resource requirements	X	X	X	X	X
	Queue discipline	X	-	-	-	X
	Queue abandonment condition	X	-	-	-	X
	Interruptibility and Unexpected interruption	X	-	-	-	X
Control flow	Control flow definition	X	X	X	X	X
	Gateway rooting logic	X	X	X	X	-
Resources	Resource roles	X	X	X	X	X
	Resource schedule	X	X	X	X	-
	Unavailability handling procedure	X	-	-	-	-
	Entity handling procedure	X	-	-	-	-

Table 4.1. Supported modeling tasks

4.6. Result and discussion

The pairwise comparison of criteria in this thesis has been influenced by the conducted surveys on important criteria and desired features of simulation tools, mentioned in the literature and outlined in section 2.2 of this thesis (page 22). This is an attempt to reduce the effect of individual's judgment. In the pairwise comparisons of criteria performed in this dissertation, and with regard to the result of the conducted surveys mentioned in the literature review, ease of use and ease of model building and output analysis along with the supported modelling tasks have obtained the highest weights. This is in accordance with the target audience for the performed comparison which is business users with little programming knowledge. Figure 4.1 illustrates the calculated weights of criteria.

The result of the AHP analysis is shown in figure 4.1. The result indicates that Arena, the representative of the dedicated simulation tools, obtains the highest aggregated weight and outperforms the other evaluated tools by a high margin for simulating business processes. The two BPM tools, Signavio and Bizagi obtain similar scores. They are followed closely by Scylla which obtains a higher weight than SimPy. Based on the evaluation carried out and the relative weights of criteria assigned to the selected set of criteria using the Analytical Hierarchy Process (AHP), Arena is identified as the most powerful for simulating business processes. The group of BPM tools are in the second place followed by the simulation packages in programming languages in the third spot. It should be noted that this evaluation focused on the BPM users and users with limited simulation and technical expertise. This affects the result of the AHP analysis. Conducting similar evaluation for different user groups require re-adjusting the matrix of pairwise comparison for the set of criteria. For example, in this thesis, using the BPMN notation is considered more important than access to the source code of the software for BPM users with limited knowledge of computer programming. However, changing the audience to more technical users and familiar with programming will change this pairwise comparison and subsequently the final result. The screenshots of the excel tables of pairwise comparisons used for comparison in this thesis can be found in the appendix at the end of this thesis.

	A	B	C	D	E
1	**Tools**			**Total Weight**	
2					
3	Arena			0.318575	
4	Scylla			0.173846	
5	Signavio			0.183623	
6	Bizagi			0.185980	
7	Simpy			0.137702	
8					

Figure 4.5. The total weights of alternative tools

Table 4.2 summarizes the criteria analysis of the selected simulation tools and identifies the availability of features in each simulation tool. As can be deduced from the analysis in section 4.1 and the table, SimPy and Scylla perform the best regarding the criteria group of coding aspects. With regards to simulation and modeling capabilities Arena outperforms the other tools possessing the most comprehensive and powerful simulation functionalities. In addition, regarding input and output capabilities, Arena again possesses the most complete set of features. The criteria related to user and vendor is dominated by Arena and SimPy, having the most comprehensive set of tutorials and support and an active community of users. Although using BPMN notation and a low level of simulation expertise required for BPM software tools make the simulation modeling easier and faster for BPM users, the lack of simulation functionalities and limited support of different modeling tasks make them unsuitable for modeling complex systems. These are the biggest shortcomings of these types of tools. BPM users can use BPMN software tools for simple modellings and at times when a simplified model of the real system is sufficient for their simulation purposes. However, when more detailed and realistic models are needed, they will be forced to turn to more powerful dedicated simulation tools or even programming packages. Therefore, further developments are required with regard to the simulation functionalities of BPM tools. Considering these observations, Scylla presents good prospects in tackling these problems regarding simulation for BPMN users due to its plug-in architecture and expandability. However, the expandability in Scylla requires expertise and knowledge of computer programming. To provide a simulation tool for BPMN users with limited expertise in these aspects, the tool needs further development. The intended audience for the Scylla project according to project initiators in Hasso Plattner Institutes is the BPM research community. A level of technical expertise with respect to programming can be expected in this target audience. Therefore, this community can contribute to creating a powerful BPM simulation tool aimed for BPM users without programming knowledge. Further developments and an extensive set of plug-ins can make fundamental contributions to this field. These observations indicate a prospect for Scylla to provide a powerful simulation tool to BPM users. Important areas to focus for further developments are simulation functionalities, supported modeling tasks and input and output capabilities.

Criteria	Sub-criteria	Arena	Scylla	Signavio	Bizagi	SimPy
Coding Aspects	**Access to source code**	NP	P	NP	NP	P
	Program generator	P	NP	NP	NP	NP
	Expandability	P	P	NP	NP	P
Simulation and Modeling Aspects	**Animation**	P	NP	L	L	NP
	Statistical facilities	P	P	L	P	E
	Modeling notation	Tool specific	BPMN	BPMN	BPMN	Tool specific
	Warm-up period	P	NP	NP	NP	P
Input/output	**Input models**	P	P	P	P	NP
	Input data	P	NP	NP	NP	NP
	Output result	P	P	P	P	P
	Graphical representation	P	NP	L	L	E
	What-if analysis	P	NP	P	P	NP
User/Vendor	**GUI**	P	P	P	P	NP
	Ease of use	Easy	Easy	Easy	Easy	Difficult
	Ease of model building	Easy	Easy	Very easy	Very easy	Difficult
	Expertise required	intermed iate	Low	Low	Low	High
	Ease of learning	Easy	Very easy	Very easy	Very easy	Difficult
	Documentation	P	NP	P	P	P
	Video Tutorials	P	L	L	L	E
	Online help	P	NP	P	P	P
	Training	P	NP	P	P	NP
	Price	Priced	Free	Priced	Free	Free

Table 4.2. Criteria check of the evaluated tools

(P: Provided or Possible; L: Limited;

E: externally possible; NP: not provided or possible)

5. Conclusion

Simulating business processes can be performed in a variety of software packages. This thesis identifies three group of simulation tools that can be used for business process simulation. These three groups are: dedicated simulation tools, business process management tools, and simulation packages in programming languages. Five simulation tools, Arena and Scylla as dedicated simulation tools, Signavio and Bizagi from BPM tools and SimPy as a simulation package of Python have been selected to be evaluated and compared based on a selected set of simulation software evaluation criteria. The selected criteria are identified from five different groups, each containing a number of sub-criteria. These five groups are criteria related to: (i) the computer programming functionalities, (ii) the modeling and simulation functionalities, (iii) the input and output considerations, (iv) the user and the support by the software vendor, and (v) the different business process modeling tasks supported by the tools.

Using the selected tools and criteria, a comparison is performed using the Analytical Hierarchy Process (AHP), a decision making method developed by Saaty (1994). The comparison focuses on BPM users with limited simulation and computer programming knowledge rather than simulation experts. For performing the comparison, a template has been developed in excel spreadsheets. This template can be used as a structured guideline for comparison and selection of simulation tools. It can be updated with new criteria and tools and adjusted for different users and projects. The result of the comparison identifies Arena, from the group of dedicated simulation tools as the most suitable option. Arena possesses the most comprehensive set of functionalities, supported modeling tasks and powerful simulation capabilities. However, a higher level of simulation expertise is required and none of the business process modeling notations such as BPMN is supported which makes the tool less compatible with BPM. Arena is followed by BPM software packages using BPMN, Signavio and Bizagi which are sufficient for simple models but lack functionalities for the development of more complex systems. Scylla follows the two BPM tools in the ranking. The lack of tutorials and documentation for users and lack of input and output functionalities are the important shortcomings of this tool. However, the expandability and its plug-in architecture are its strong points. SimPy from the group of simulation packages in programming languages is in the last spot due to the complexity in the model development and the amount of coding required to build simulation models. It can be concluded from the result that further development in simulation functionalities of BPM tools is required for the BPM users to satisfy their business process simulation needs. To this end, open source and extensible BPM simulation tool, Scylla, demonstrate potential for further development in this field. The BPM research community and software vendors can play an important role in development of a powerful and flexible business process simulation tool.

One of the important limitations of the performed comparison in this thesis is the small number of evaluated tools. This issue prevented making a general conclusion on comparison of the three different groups of tools and their suitability for business process simulation. In addition, although AHP, the methodology used for comparison in this thesis controls and considers the human judgment, the comparison performed and the weights of criteria are still to some extent subjective and influenced by judgment.

Further research can build on the comparison carried out in this thesis to compare large number of simulation tools from each of the three groups of simulation tools identified in this thesis. This will be necessary in order to come to a general conclusion regarding the three groups of business process simulation tools. Further research can also expand the evaluation by re-adjusting the comparison matrices for different user groups (Business users, Researchers and academics, Simulation experts, etc.). The matrices provided in the excel file attached to this thesis can simply be manipulated to perform further evaluations. Furthermore, in order to obtain a more accurate and less judgmental weighing of criteria, further research can conduct surveys and questionnaires among different user groups to identify the important criteria. The result of these surveys can assist the pairwise comparison of criteria and reduce the effect of individual's judgment and preferences.

References

Banks, J., Aviles, E., McLaughlin, J. R., & Yuan, R. C. (1991). The simulator: new member of the simulation family. *Interfaces*, *21*(2), 76-86.

Black, D. C., Donovan, J., Bunton, B., & Keist, A. (2011). *SystemC: From the ground up* (Vol. 71). Springer Science & Business Media

Bosilj-Vuksic, V., Ceric, V., & Hlupic, V. (2007). Criteria for the evaluation of business process simulation tools. *Interdisciplinary Journal of Information, Knowledge, and Management,* 2, 73-88.

Bradley, P., Browne, J., Jackson, S., & Jagdev, H. (1995). Business process re-engineering (BPR)—a study of the software tools currently available. *Computers in Industry*, *25*(3), 309-330.

Cimino, M. G., & Vaglini, G. (2014). An interval-valued approach to business process simulation based on genetic algorithms and the BPMN. *Information*, *5*(2), 319-356.

Cox, S. W. (1991). GPSS World: a brief preview. In *Simulation Conference, 1991. Proceedings., Winter* (pp. 59-61). IEEE.

Davis, L., & Williams, G. (1994). Evaluating and selecting simulation software using the analytic hierarchy process. *Integrated manufacturing systems*, *5*(1), 23-32.

Easily optimise your business processes | Engage Process. (n.d.). Retrieved from https://www.engageprocess.com/en-euro/

Emshoff, J. R., & Saaty, T. L. (1982). Applications of the analytic hierarchy process to long range planning processes. *European Journal of Operational Research*, *10*(2), 131-143.

Enterprise Dynamics - INCONTROL Simulation Solutions. (n.d.). Retrieved from https://www.incontrolsim.com/product/enterprise-dynamics/

Ereeş, S., Kuruoğlu, E., & Moralı, N. (2013). An application of Analytical Hierarchy Process for simulation software selection in education area. *Frontiers in Science*, *3*(2), 66-70.

Freitas, A. P., & Pereira, J. L. M. (2015). Process simulation support in BPM tools: The case of BPMN.

Göbel, J., Joschko, P., Koors, A., & Page, B. (2013, May). The Discrete Event Simulation Framework DESMO-J: Review, Comparison to Other Frameworks and Latest Development. In *ECMS* (pp. 100-109).

Gholamnezhad, A. H., & Saaty, T. L. (1982). A desired energy mix for the United States in the year 2000: An analytic hierarchy approach. *International Journal of Policy Analysis and Information Systems*, *6*(1), 47-64.

Greasley, A. (2003). Using business-process simulation within a business-process reengineering approach. *Business Process Management Journal*, *9*(4), 408-420.

Gupta, A. (2014). How to Select a Simulation Software. *International Journal of Engineering Research and Development*, *10*(3), 35-41.

Gupta, A., Singh, K., & Verma, R. (2010). A critical study and comparison of manufacturing simulation softwares using analytic hierarchy process. *Journal of Engineering science and technology*, *5*(1), 108-129.

Hammann, J. E., & Markovitch, N. A. (1995, December). Introduction to Arena [simulation software]. In *Simulation Conference Proceedings, 1995. Winter* (pp. 519-523). IEEE.

Harrell, Ghosh, & Bowden. (2004). *Simulation using promodel*. McGraw-Hill/Higher Education.

Hauge, J. W., & Paige, K. N. (2004). *Learning SIMUL8: the complete guide*. PlainVu.

Hlupic, V. (2000). Simulation software: a survey of academic and industrial users. *International Journal of Simulation*, *1*(1-2), 1-12.

Hlupic, V., Irani, Z., & Paul, R. J. (1999). Evaluation framework for simulation software. *The International Journal of Advanced Manufacturing Technology*, *15*(5), 366-382.

Hlupic, V., & Mann, A. S. (1995, December). SimSelect: a system for simulation software selection. In *Simulation Conference Proceedings, 1995. Winter* (pp. 720-727). IEEE.

Hlupic, V., & Paul, R. J. (1995). A critical evaluation of four manufacturing simulators. *International Journal of Production Research*, *33*(10), 2757-2766.

Indulska, M., Recker, J., Rosemann, M., & Green, P. (2009, June). Business process modeling: Current issues and future challenges. In *International Conference on Advanced Information Systems Engineering* (pp. 501-514). Springer, Berlin, Heidelberg.

Jansen-Vullers, M., & Netjes, M. (2006, October). Business process simulation–a tool survey. In *Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, Aarhus, Denmark* (Vol. 38).

Kellner, M. I., Madachy, R. J., & Raffo, D. M. (1999). Software process simulation modeling: why? what? how? *Journal of Systems and Software*, *46*(2-3), 91-105.

Kelton WD, Sadowski P, Swets NB (2010) *Simulation with Arena*. McGraw-Hill, New York.

Kelton, W. D., Smith, J. S., & Sturrock, D. T. (2011). *Simio & simulation: Modeling, analysis, applications*. Learning Solutions.

Krahl, D. (2009, December). ExtendSim advanced techology: discrete rate simulation. In *Winter Simulation Conference* (pp. 333-338). Winter Simulation Conference.

Law, A. M., & Kelton, W. D. (2007). *Simulation modeling and analysis* (Vol. 3). New York: McGraw-Hill.

Lazzari, D., & Crosslin, R. (1996). Introduction to work flow modeling with BPSimulator. In *Simulation Conference, 1996. Proceedings. Winter* (pp. 429-431). IEEE.

L-Sim BPMN Model Simulation Engine | Lanner. (n.d.). Retrieved from https://www.lanner.com/en-us/technology/l-sim-bpmn-simulation-engine.html

Mackulak, G. T., Cochran, J. K., & Savory, P. A. (1994). Ascertaining important features for industrial simulation environments. *Simulation*, *63*(4), 211-221.

Martin, N., Depaire, B., & Caris, A. (2016). The use of process mining in business process simulation model construction. *Business & Information Systems Engineering*, *58*(1), 73-87.

Melão, N., & Pidd, M. (2003). Use of business process simulation: A survey of practitioners. *Journal of the operational Research Society*, *54*(1), 2-10.

Muller, K., & Vignaux, T. (2003). Simpy: Simulating systems in python. *ONLamp. com Python Devcenter*.

Nikoukaran, J., Hlupic, V., & Paul, R. J. (1999). A hierarchical framework for evaluating simulation software. *Simulation Practice and Theory*, *7*(3), 219-231.

Nordgren, W. B. (2003, December). Flexible simulation (Flexsim) software: Flexsim simulation environment. In *Proceedings of the 35th conference on Winter simulation: driving innovation* (pp. 197-200). Winter Simulation Conference.

Oakshott, L. A. (1997). *Business modelling and simulation*. Pitman.

Otamendi, J., Pastor, J. M., & Garcı, A. (2008). Selection of the simulation software for the management of the operations at an international airport. *Simulation Modelling Practice and Theory*, *16*(8), 1103-1112.

Perera, T., & Liyanage, K. (2001). IDEF based methodology for rapid data collection. *Integrated manufacturing systems*, *12*(3), 187-194.

Pidd, M. (1992). Guidelines for the design of data driven generic simulators for specific domains. *Simulation*, *59*(4), 237-243.

Pidd, M., & Carvalho, A. (2006). Simulation software: not the same yesterday, today or forever. *Journal of Simulation*, *1*(1), 7-20.

Process Simulation | Signavio. (n.d.). Retrieved from https://www.signavio.com/process-simulation/

Pufahl, L., Wong, T. Y., & Weske, M. (2017, September). Design of an Extensible BPMN Process Simulator. In *International Conference on Business Process Management* (pp. 782-795). Springer, Cham.

Rozinat, A., & Van der Aalst, W. M. (2008). Conformance checking of processes based on monitoring real behavior. *Information Systems*, *33*(1), 64-95.

Saaty, T. L. (1994). How to make a decision: the analytic hierarchy process. *Interfaces*, *24*(6), 19-43.

Saaty, T.L., and Rogers, P.C. (1976). Higher education in the united states (1985-2000). Scenario construction using a hierarchical framework with eigenvector weighting. *Socio-economic Planning Sciences*, 10(6), 251-263.

SIM.JS | Discrete Event Simulation in JavaScript. (2011). Retrieved from http://simjs.com/

Simulation software for process improvement. (n.d.). Retrieved from https://www.processmodel.com/

Sundarapandian, V. (2009). *Probability, statistics and queuing theory*. PHI Learning Pvt. Ltd.

Swegles, S. (1997, December). Business process modeling with SIMPROCESS. In *Winter Simulation Conference* (pp. 606-610).

Tewoldeberhan, T. W., Verbraeck, A., Valentin, E., & Bardonnet, G. (2002, December). Software evaluation and selection: an evaluation and selection methodology for discrete-event simulation software. In *Proceedings of the 34th conference on Winter simulation: exploring new frontiers* (pp. 67-75). Winter Simulation Conference.

Tumay, K. (1996, November). Business process simulation. In *Proceedings of the 28th conference on Winter simulation* (pp. 93-98). IEEE Computer Society.

Van der Aalst, W. M. (2013). Business process management: a comprehensive survey. *ISRN Software Engineering*, *2013*.

Van der Aalst, Wil; La Rosa, Marcello; and Santoro, Flavia Maria 999813618 (2016) "Business Process Management - Don’t Forget to Improve the Process!," *Business & Information Systems Engineering*: Vol. 58: Iss. 1, 1-6.

Verma, R., Gupta, A., & Singh, K. (2008). Simulation software evaluation and selection: a comprehensive framework. *J. Automation & Systems Engineering*, *2*, 221-234.

Weske, M. (2012). Business process management architectures. In *Business Process Management*. Springer, Berlin, Heidelberg.

White, K. P., & Ingalls, R. G. (2015, December). Introduction to simulation. In *Winter Simulation Conference (WSC), 2015* (pp. 1741-1755). IEEE.

Williams, G. B., & Trauth, J. (1991). Evaluation and selection of manufacturing software for a small business. In *Advances in Manufacturing Technology VI, Proceedings of the Seventh National Conference on Production Research* (pp. 102-6).

Yang, Y., Li, J., & Zhao, Q. (2014). Study on passenger flow simulation in urban subway station based on anylogic. *Journal of Software*, *9*(1), 140-146

Appendix: The comparison matrices of criteria and tools

		Coding Aspect			Simulation and Modeling Aspect			
		Access to sour	Program gener	Expandability	Animation	Statistical f	Modeling nota	Warm-up p
Coding Aspects	Access to source code	1	5	1/5	1/3	1/5	1/3	5
	Program generator	1/5	1	1/9	1/7	1/9	1/7	1
	Expandability	5	9	1	3	1	3	9
Simulation and Modelin	Animation	3	7	1/3	1	1/5	1/3	7
	Statistical facilities	5	9	1	5	1	3	9
	Modeling notation	3	7	1/3	3	1/3	1	7
	Warm-Up period	1/5	1	1/9	1/7	1/9	1/7	1
Input/Output	integration with other software	3	7	1/3	3	1/3	1	5
	Input models	1	3	1/5	1	1/5	1/3	3
	Input data	3	5	1/3	3	1/3	1	5
	Output result	3	5	1/3	3	1/3	1	5
	Output analysis	5	9	1	5	1	3	9
	Graphical representation	1	5	1/5	1/3	1/5	1/3	3
	What-if analysis	1/3	3	1/5	1/5	1/5	1/5	3
User/Vendor	User friendliness	5	9	1/3	3	1/3	1	7
	GUI	3	5	1/5	1/3	1/5	1/3	3
	Ease of use	7	9	1	5	1	3	9
	Ease of model building	5	7	1	5	1	3	7
	Expertise required	3	5	1/3	3	1/3	1	5
	Ease of learning	5	7	1/3	3	1/3	1	7
	Documentation	3	5	1/3	3	1/3	1	7
	Tutorials	3	5	1/3	3	1/3	1	7
	Online-help	1	3	1/5	1/3	1/5	1/3	3
	Training	1/3	1	1/9	1/7	1/9	1/5	1
	Price	1/3	3	1/5	1/3	1/5	1/3	3
Supported Modeling Tas	Supported modeling tasks	5	9	1	5	3	5	9
Total		**74.40**	**144.00**	**11.07**	**59.30**	**12.93**	**32.02**	**140.00**

Figure A.1. A fraction of criteria comparison matrix

2			Input/Output						
3			Integration v	Input models	Input data	Output resu	Output analysis	Graphical repres	What-if ana
4	**Coding Aspects**	Access to source code	1/3	1	1/3	1/3	1/5	1	3
5		Program generator	1/7	1/3	1/5	1/5	1/9	1/5	1/3
6		Expandability	3	5	3	3	1	5	5
7	**Simulation and Modelin**	Animation	1/3	1	1/3	1/3	1/5	3	5
8		Statistical facilities	3	5	3	3	1	5	5
9		Modeling notation	1	3	1	1	1/3	3	5
10		Warm-Up period	1/5	1/3	1/5	1/5	1/9	1/3	1/3
11	**Input/Output**	integration with other software	1	3	1	1	1/3	3	5
12		Input models	1/3	1	1/3	1/3	1/7	1	3
13		Input data	1	3	1	1	1/5	3	3
14		Output result	1	3	1	1	1/5	3	3
15		Output analysis	3	7	5	5	1	5	5
16		Graphical representation	1/3	1	1/3	1/3	1/5	1	1
17		What-if analysis	1/5	1/3	1/3	1/3	1/5	1	1
18	**User/Vendor**	User friendliness	1	5	5	5	1	5	5
19		GUI	1/5	1	1/3	1/3	1/5	1	1
20		Ease of use	3	5	5	5	3	5	5
21		Ease of model building	3	5	5	5	3	5	5
22		Expertise required	1	3	3	3	1	5	5
23		Ease of learning	3	3	3	3	1	5	5
24		Documentation	1	3	1	1	1	5	5
25		Tutorials	1	3	1	1	1	5	5
26		Online-help	1/5	1/3	1/3	1/3	1/5	3	3
27		Training	1/7	1/5	1/7	1/7	1/9	1/3	1/3
28		Price	1/3	1/3	1/5	1/5	1/5	3	3
29	**Supported Modeling Tas**	Supported modeling tasks	3	5	3	3	1	5	5
30	**Total**		**31.75**	**67.87**	**44.08**	**44.08**	**17.94**	**81.87**	**92.00**

Figure A.2. A fraction of criteria comparison matrix (continued)

		User/Vendor					
		User friendlinss	GUI	Ease of use	Ease of mode	Expertise requi	Ease of learni
Coding Aspects	Access to source code	1/5	1/3	1/7	1/5	1/3	1/5
	Program generator	1/9	1/5	1/9	1/7	1/5	1/7
	Expandability	3	5	1	1	3	3
Simulation and Modelin	Animation	1/3	3	1/5	1/5	1/3	1/3
	Statistical facilities	3	5	1	1	3	3
	Modeling notation	1	3	1/3	1/3	1	1
	Warm-Up period	1/7	1/3	1/9	1/7	1/5	1/7
Input/Output	integration with other software	1	5	1/3	1/3	1	1/3
	Input models	1/5	1	1/5	1/5	1/3	1/3
	Input data	1/5	3	1/5	1/5	1/3	1/3
	Output result	1/5	3	1/5	1/5	1/3	1/3
	Output analysis	1	5	1/3	1/3	1	1
	Graphical representation	1/5	1	1/5	1/5	1/5	1/5
	What-if analysis	1/5	1	1/5	1/5	1/5	1/5
User/Vendor	User friendliness	1	5	1/3	1/3	1/3	1
	GUI	1/5	1	1/5	1/5	1/5	1/5
	Ease of use	3	5	1	1	3	3
	Ease of model building	3	5	1	1	3	1
	Expertise required	3	5	1/3	1/3	1	1/3
	Ease of learning	1	5	1/3	1	3	1
	Documentation	1	5	1/3	1	3	1
	Tutorials	1	5	1/3	1	3	1
	Online-help	1/3	3	1/5	1/3	1/3	1/5
	Training	1/7	1/3	1/9	1/7	1/7	1/7
	Price	1/3	3	1/3	1/3	1/3	1/3
Supported Modeling Tas	Supported modeling tasks	3	5	1	1	3	3
Total		**27.80**	**83.20**	**10.08**	**12.36**	**31.81**	**22.76**

Figure A.3. A fraction of criteria comparison matrix (continued)

							Supportec
		Documentati	Tutorials	Online-he	Training	Price	Supportec
Coding Aspects	Access to source code	1/3	1/3	1	3	3	1/5
	Program generator	1/5	1/5	1/3	1	1/3	1/9
	Expandability	3	3	5	9	5	1
Simulation and Modelin	Animation	1/3	1/3	3	7	3	1/5
	Statistical facilities	3	3	5	9	5	1/3
	Modeling notation	1	1	3	5	3	1/5
	Warm-Up period	1/7	1/7	1/3	1	1/3	1/9
Input/Output	integration with other software	1	1	5	7	3	1/3
	Input models	1/3	1/3	3	5	3	1/5
	Input data	1	1	3	7	5	1/3
	Output result	1	1	3	7	5	1/3
	Output analysis	1	1	5	9	5	1
	Graphical representation	1/5	1/5	1/3	3	1/3	1/5
	What-if analysis	1/5	1/5	1/3	3	1/3	1/5
User/Vendor	User friendliness	1	1	3	7	3	1/3
	GUI	1/5	1/5	1/3	3	1/3	1/5
	Ease of use	3	3	5	9	3	1
	Ease of model building	1	1	3	7	3	1
	Expertise required	1/3	1/3	3	7	3	1/3
	Ease of learning	1	1	5	7	3	1/3
	Documentation	1	1	3	5	3	1/3
	Tutorials	1	1	3	5	3	1/3
	Online-help	1/3	1/3	1	3	1	1/5
	Training	1/5	1/5	1/3	1	1/3	1/9
	Price	1/3	1/3	1	3	1	1/5
Supported Modeling Tas	Supported modeling tasks	3	3	5	9	5	1
Total		**25.14**	**25.14**	**70.00**	**142.00**	**70.00**	**10.13**

Figure A.4. A fraction of criteria comparison matrix (continued)

Access to source code					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1/9	1	1	1/9
Scylla	9	1	9	9	1
Signavio	1	1/9	1	1	1/9
Bizagi	1	1/9	1	1	1/9
Simpy	9	1	9	9	1

Program generator					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	7	7	7	7
Scylla	1/7	1	1	1	1
Signavio	1/7	1	1	1	1
Bizagi	1/7	1	1	1	1
Simpy	1/7	1	1	1	1

Expandability					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1/5	5	5	1/3
Scylla	5	1	9	9	3
Signavio	1/5	1/9	1	1	1/7
Bizagi	1/5	1/9	1	1	1/7
Simpy	3	1/3	7	7	1

Figure A.5. Comparison matrices of tools on 'Coding aspect'

Animation					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	9	5	5	9
Scylla	1/9	1	1/5	1/5	1
Signavio	1/5	5	1	1	5
Bizagi	1/5	5	1	1	5
Simpy	1/9	1	1/5	1/5	1

Statistical Facilities					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	3	5	3	1
Scylla	1/3	1	3	1	1/3
Signavio	1/5	1/3	1	1/3	1/5
Bizagi	1/3	1	3	1	1/3
Simpy	1	3	5	3	1

Modeling notation					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1/7	1/7	1/7	5
Scylla	7	1	1/3	1	9
Signavio	7	3	1	3	9
Bizagi	7	1	1/3	1	9
Simpy	1/5	1/9	1/9	1/9	1

Warm-up period					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	5	5	5	1
Scylla	1/5	1	1	1	1/5
Signavio	1/5	1	1	1	1/5
Bizagi	1/5	1	1	1	1/5
Simpy	1	5	5	5	1

Figure A.6. Comparison matrices of tools on 'Simulation and modeling'

Integration with other software					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	7	5	5	7
Scylla	1/7	1	1/3	1/3	1
Signavio	1/5	3	1	1	3
Bizagi	1/5	3	1	1	3
Simpy	1/7	1	1/3	1/3	1

Input Models					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1	1	1	9
Scylla	1	1	1	1	9
Signavio	1	1	1	1	9
Bizagi	1	1	1	1	9
Simpy	1/9	1/9	1/9	1/9	1

Input data					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	9	9	9	9
Scylla	1/9	1	1	1	1
Signavio	1/9	1	1	1	1
Bizagi	1/9	1	1	1	1
Simpy	1/9	1	1	1	1

Output result					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	5	1	1	5
Scylla	1/5	1	1/5	1/5	1
Signavio	1	5	1	1	5
Bizagi	1	5	1	1	5
Simpy	1/5	1	1/5	1/5	1

Output analysis					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	7	3	3	7
Scylla	1/7	1	1/5	1/5	1
Signavio	1/3	5	1	1	5
Bizagi	1/3	5	1	1	5
Simpy	1/7	1	1/5	1/5	1

Graphical presentation					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	9	5	5	7
Scylla	1/9	1	1/5	1/5	1/3
Signavio	1/5	5	1	1	5
Bizagi	1/5	5	1	1	5
Simpy	1/7	3	1/5	1/5	1

What-if analysis					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	5	1	1	5
Scylla	1/5	1	1/5	1/5	1
Signavio	1	5	1	1	5
Bizagi	1	5	1	1	5
Simpy	1/5	1	1/5	1/5	1

Figure A.7. Comparison matrices of tools on 'Input/output'

User friendliness	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	3	1	1	7
Scylla	1/3	1	1/3	1/3	5
Signavio	1	3	1	1	7
Bizagi	1	3	1	1	7
Simpy	1/7	1/5	1/7	1/7	1

GUI	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1	1	1	9
Scylla	1	1	1	1	9
Signavio	1	1	1	1	9
Bizagi	1	1	1	1	9
Simpy	1/9	1/9	1/9	1/9	1

Ease of use	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1	1	1	9
Scylla	1	1	1	1	9
Signavio	1	1	1	1	9
Bizagi	1	1	1	1	9
Simpy	1/9	1/9	1/9	1/9	1

Ease of model building	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	7	1	1	9
Scylla	1/7	1	1/7	1/7	3
Signavio	1	7	1	1	9
Bizagi	1	7	1	1	9
Simpy	1/9	1/3	1/9	1/9	1

Expertise required	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1/5	1/5	1/5	5
Scylla	5	1	1	1	9
Signavio	5	1	1	1	9
Bizagi	5	1	1	1	9
Simpy	1/5	1/9	1/9	1/9	1

Ease of learning	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1/5	1/5	1/5	5
Scylla	5	1	1	1	9
Signavio	5	1	1	1	9
Bizagi	5	1	1	1	9
Simpy	1/5	1/9	1/9	1/9	1

Documentation	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	9	3	3	1
Scylla	1/9	1	1/5	1/5	1/9
Signavio	1/3	5	1	1	1/3
Bizagi	1/3	5	1	1	1/3
Simpy	1	9	3	3	1

Tutorials	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	9	5	5	3
Scylla	1/9	1	1/5	1/5	1/7
Signavio	1/5	5	1	1	1/3
Bizagi	1/5	5	1	1	1/3
Simpy	1/3	7	3	3	1

Online help	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	9	5	3	1
Scylla	1/9	1	1/3	1/5	1/9
Signavio	1/5	3	1	1/3	1/5
Bizagi	1/3	5	3	1	1/3
Simpy	1	9	5	3	1

Training	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	3	1	3	3
Scylla	1/3	1	1/3	1	1
Signavio	1	3	1	3	3
Bizagi	1/3	1	1/3	1	1
Simpy	1/3	1	1/3	1	1

Price	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	1/9	1	1/9	1/9
Scylla	9	1	9	1	1
Signavio	1	1/9	1	1/9	1/9
Bizagi	9	1	9	1	1
Simpy	9	1	9	1	1

Figure A.8. Comparison matrices of tools on 'User/Vendor'

Supported modeling tasks					
	Arena	Scylla	Signavio	Bizagi	Simpy
Arena	1	3	5	5	3
Scylla	1/3	1	3	3	1
Signavio	1/5	1/3	1	1	1/3
Bizagi	1/5	1/3	1	1	1/3
Simpy	1/3	1	3	3	1

Figure A.9. Comparison matrices of tools on 'Supported modeling tasks'

	Access to	Program	Expandab	Animatior	Statistical	Modeling	Warm-up	Integratio	Input moc	Input data	Output re	Output an	Graphical	What-if ar
Arena	0.047619	0.63636	0.145451	0.564987	0.342358	0.065064	0.384615	0.56618	0.243243	0.692308	0.294118	0.47203	0.545816	0.294118
Scylla	0.428571	0.09091	0.506726	0.043557	0.129806	0.228452	0.076923	0.061007	0.243243	0.076923	0.058824	0.050502	0.036579	0.058824
Signavio	0.047619	0.09091	0.04049	0.17395	0.055672	0.45014	0.076923	0.155903	0.243243	0.076923	0.294118	0.213484	0.176262	0.294118
Bizagi	0.047619	0.09091	0.04049	0.17395	0.129806	0.228452	0.076923	0.155903	0.243243	0.076923	0.294118	0.213484	0.176262	0.294118
Simpy	0.428571	0.09091	0.266843	0.043557	0.342358	0.027892	0.384615	0.061007	0.027027	0.076923	0.058824	0.050502	0.065081	0.058824

Figure A.10. Calculated weights of tools on criteria

	User frien	GUI	Ease of us	Ease of m	Expertise	Ease of le	Document	Tutorials	Online-he	Training	Price	Supportec
Arena	0.283278	0.243243	0.243243	0.305141	0.07889	0.07889	0.35241	0.495931	0.367341	0.333333	0.034483	0.462385
Scylla	0.114179	0.243243	0.243243	0.054652	0.297482	0.297482	0.032653	0.033239	0.034175	0.111111	0.310345	0.195154
Signavio	0.283278	0.243243	0.243243	0.305141	0.297482	0.297482	0.131263	0.111801	0.075277	0.333333	0.034483	0.073653
Bizagi	0.283278	0.243243	0.243243	0.305141	0.297482	0.297482	0.131263	0.111801	0.155866	0.111111	0.310345	0.073653
Simpy	0.035987	0.027027	0.027027	0.029924	0.028664	0.028664	0.35241	0.247228	0.367341	0.111111	0.310345	0.195154

Figure A.11. Calculated weights of tools on criteria (continued)

	Weights
Access to source code	0.0159661145
Program generator	0.0060773167
Expandability	0.0807394579
Animation	0.0247981930
Statistical facilities	0.0795063828
Modeling notation	0.0359597780
Warm-Up period	0.0061399124
integration with other software	0.0373550809
Input models	0.0175846311
Input data	0.0319376130
Output data	0.0319376130
Output analysis	0.0673578191
Graphical representation	0.0127062041
What-if analysis	0.0110414107
User friendliness	0.0491771543
GUI	0.0135786102
Ease of use	0.0897493320
Ease of model building	0.0764928213
Expertise required	0.0423088095
Ease of learning	0.0528390738
Documentation	0.0437173087
Tutorials	0.0437173087
Online-help	0.0162340563
Training	0.0061387056
Price	0.0162777431
Supported modeling tasks	0.0903868239

Figure A.12. Calculated weights of criteria

Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:
A comparative study of business process simulation tools

Richting: **Master of Management-Business Process Management**
Jaar: **2018**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

Diba, Kiarash

Datum: **31/05/2018**