CAPTURING AND MODELLING COMPLEX MANUFACTURING SYSTEMS – EXTENDING AND EMBEDDING BPMN IN DES

Basma Elsharef Elmabrouk

PhD

2021

CAPTURING AND MODELLING COMPLEX MANUFACTURING SYSTEMS – EXTENDING AND EMBEDDING BPMN IN DES

By

Basma Elsharef Elmabrouk, B.Sc., M.Sc.

This Thesis is submitted in accordance with the requirements of Dublin City University for the award of the degree of

Doctor of Philosophy

Supervisors:

Dr. Paul Young

School of Mechanical & Manufacturing Engineering

Dublin City University, Ireland

Prof. Khaled S. El-Kilany

Department of Industrial and Management Engineering

Arab Academy for Science, Technology and Maritime Transport

Alexandria, Egypt.

2021

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Ph.D. is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Basma Elsharef Elmabrouk

Signed:

ID: 12212379

Monday, 11 January 2021

ACKNOWLEDGEMENT

I thank God for granting me patience and perseverance to get through all the difficult times I faced; thank God for giving me the will and effort to fulfil this work.

I would like to express my deepest gratitude and appreciation to Dr. Paul Young for his guidance and encouragement. I greatly appreciate the time he has generously given me in resolving any issues I encountered. He has been very helpful and has given me an infinite amount of technical support and advice which have been invaluable during my research period.

It gives me the greatest pleasure and delight to express my sincere gratitude to Prof. Khaled S. El-Kilany (Head of Industrial and Management Engineering Department – AASTMT) from all my heart and soul. His wide knowledge and logical way of thinking have been of great value to me. His understanding, encouragement, personal guidance, and continuous supervision were the cornerstones that made this work come to light. Not only has he helped with my work, but also, he helped with almost every single problem I faced.

I would also like to thank Prof. Amr Ali Hassan (Dean of Engineering and Technology college –AASTMT), for his care and support in every possible way.

I would like to acknowledge that the models developed for this work are built using the ExtendSimTM Suite v9.2 simulation environment from Imagine That, Inc. The tool has been offered to the Department of Industrial and Management Engineering, AASTMT, as a grant for teaching and research purposes as part of the ExtendSim Adopter Program.

Very special thanks to all my family especially my Mum and Dad, for their endless love, encouragement, and providing me with continual support and help. Thank you for your patience and serenity, without my family I couldn't have accomplished this work.

I am deeply grateful to my beloved husband, Ahmed El-Shiekh who supported me and encouraged me all the way showing how much he believes in me. Also, I would like to thank my daughter Noura who joined us when I was struggling with my thesis, she made me feel better and stronger, and she gave me unlimited happiness and pleasure.

At last, but never at least, I wish to express my warm and sincere thanks to all my friends, especially Ms Noha Mansour who supported me and encouraged me all the way.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

BP	Business Process
BPEL	Business Process Execution Language
BPM	Business Process Management
BPMI	Business Process Management Initiative
BPMN	Business Process Model and Notation
BPSim	Business Process Simulation Standard
CT	Cycle Time
DSPML	Domain-Specific Process Modelling Languages
EM	Emergency Maintenance
IDEF	The Integrated DEFinition
IoT	Internet of Things
Mini-fab	Intel Five Machine Six Step Mini-Fab
MPMN	Manufacturing Process Model and Notation
MTBA	Mean Time Between Arrivals
MTBF	Mean Time Between Failures
MTTF	Mean Time To Failure
MTTR	Mean Time To Repair
OMG	Object Management Group
PM	Preventive Maintenance
PT	Processing Time
PyBPMN	Performability-enabled BPMN
SBPMN	Simple Business Process Modelling
SysML	Systems Modelling Language

TBA	Time Between Arrivals
TH	Throughput Rate
TTF	Time to Failure
TTR	Time to Repair
uBPMN	ubiquitous BPMN
UML	Unified Modelling Language
UMLAD	Unified modelling Language Activity Diagram
YAWL	Yet Another Workflow Language

ABSTRACT

Basma Elsharef Elmabrouk
CAPTURING AND MODELLING COMPLEX MANUFACTURING SYSTEMS - EXTENDING AND EMBEDDING BPMN IN DES

Business process models are developed for the purpose of understanding the system behaviour and identifying the possible areas for performance improvement. Among existing process modelling languages, Business Process Model and Notation (BPMN) was selected to be assessed and extended, as it is the leading standard for business process modelling. The BPMN is currently gaining great attention in various business practices; it is an easy and flexible way to construct business process models, and thus it was hard to be overlooked by authors who are concerned with improving manufacturing processes.

The introduction of BPMN to the manufacturing domain potentially allows all stakeholders to take advantage of the simplicity of this language in gaining full understandings for manufacturing processes through simple representations of the process models. Only limited work can be found addressing the use of BMPN in the modelling of manufacturing systems, and it is still not clear how powerful BPMN is in realizing performance improvements. This work proposes Manufacturing Process Model and Notation (MPMN) as an extension to BPMN; it offers a set of new and adapted notations that represent manufacturing processes. In order to add the ability to optimise or improve the system under study, MPMN is further extended and integrated with a full discrete-event simulation package to be able to easily build and simulate models of manufacturing systems.

MPMN simulator is created to be a generic and reusable tool. It has a drag and drop library for non- simulation experts, to model and simulate MPMN models in the ExtendSim environment. This integration is considered a substitution to BPSim, the Business Process Simulation Standard that integrates BPMN and Simulation to model and execute business processes. The MPMN simulator combines both, models that are understood by all stakeholders and a simulation tool that is expressive enough to handle the varying levels of complexity in the manufacturing domain. This will strengthen the analysis and the evaluation of the current and future status of a system under study. The effectiveness of the developed system is demonstrated by application to two case studies covering different industry sectors.

1. INTRODUCTION

The vast change in today's business environment made the survival and competition of companies a tough challenge. Most modern companies are adopting a business process management (BPM) strategy, which ensures a continuous improvement of business processes (BPs) and their adaptation to change. BPM includes several activities such as automation, execution, and monitoring of BPs, but the modelling activity remains the most crucial one. Indeed, it allows specifying the way of carrying out BPs, which has a direct influence on the quality of a company's deliverables and thus on customer satisfaction [1]. Thus, business and manufacturing processes must be modelled using simple and understandable notations to be flexible enough to respond promptly to the fast market changes and to meet market needs. Business processes are usually expressed in graphical notations; these notations are modelled for the purpose of understanding the behaviour of these processes and identifying the possible areas for improvement. There are several modelling languages that define the basic elements for constructing business process models and can be adopted to model manufacturing systems, among which the Business Process Model and Notation (BPMN) has gained great attention in various business practices. BPMN is a large and feature-rich modelling language serving a wide array of different purposes [2], and it is believed that it will standardise business interactions in the future and can possibly make other modelling languages used in that field obsolete.

Different work can be found in literature about modelling business processes using BPMN from pure business processes documentation [3] to higher level of risk management, supply chain design [4, 5], and workflows definition [6, 7]. This may be attributed to the fact that it is primarily used to provide a graphical representation of how the system works and does not provide any information about the dynamics of that system. Due to the increasing usage and acceptance of BPMN for business process design, and because it is gaining momentum in the business domain more and more extensions to BPMN are proposed [8]. BPMN as a standard language has the potential to revolutionize the collaboration of business process design and process implementation. However, BPMN was hardly overlooked by authors who were continuously concerned about improving manufacturing processes; since its ability to improve the performance of a manufacturing system is still under study. Yet, there is minor work in literature that focus on modelling manufacturing systems using BPMN. BPMN has to be extended or to be coupled with another language to model complex interactions in manufacturing [9]. One of these extensions that are offered to BPMN to enhance manufacturing systems modelling was found in the work of S. Zor et al [10], who extended BPMN by developing new notations that were customized to include some missing aspects required to model manufacturing processes and systems.

This work addresses BPMN's deficiencies in representing manufacturing systems and proposes an extension to BPMN to accommodate manufacturing systems modelling. This work further extends the BPMN notations by building on Zor's work aiming to provide a set of notations that can represent and effectively model manufacturing systems of different levels of complexity. First, BPMN modelling capabilities are assessed using two real-life case studies. Modelling the case studies shows that BPMN falls behind in representing the main features of a manufacturing domain. Based on the modelling outcome, an extension to BPMN is proposed; Manufacturing Process Model and Notation (or, in short, MPMN) is proposed to overcome the shortfalls of BPMN to describe important aspects of manufacturing systems. It involves modifications to the existing BPMN notations and the development of new notations that are able to model manufacturing processes using core modelling notations that better represent the system.

However, it is not possible to overlook that manufacturing processes require more than a mere process representation. MPMN has to deal with the characteristics of manufacturing systems; it has to be able to handle its growing scale and complexity and have the capability to support improvements and decision-making for a given manufacturing system under study. Thus, capturing the dynamic, stochastic, and behavioural tendencies of manufacturing systems is crucial; simulation is one of the most commonly used techniques for analysing and understanding the dynamics of manufacturing systems, it has played a significant role in evaluating the design and operational performance of manufacturing systems. Moreover, it has successful applications in many practical real-world problems and it has proved its effectiveness in approaching various problems in the manufacturing sector [11].

Consequently, simulation of the developed models for the two case studies is advised; it was claimed that BPMN and BPSim are powerful standards and have the ability to create models for business processes and help to promote the use of business process simulation and to build tools for modelling simulation models independently from simulation tools [12]. Accordingly, the Bizagi modeler is selected to model the case studies; it uses BPMN to build models, and BPSim to run discrete-event simulation in the same software environment. But then again, in fact, both BPMN and BPSim suffer from limited capabilities to represent some important features of complex flows [4], and both standards show that they are incapable to handle the complexity of a manufacturing system. To overcome this issue, MPMN is further extended, and it is integrated with a full discrete event simulation software, as a substitution to BPSim, resulting in development of the MPMN Simulator. Employing component-based modelling concepts and making use of the full capabilities of a discrete-event simulator such as ExtendSim, the MPMN Simulator promises the development of MPMN models that will go straight from a static MPMN diagram to a dynamic MPMN model, without interacting with the simulation model directly or doing the programming behind.

Thus, three different modelling and simulation tools are used in this work. The first is a full discrete event simulation software, which is ExtendSim and will be used to develop based models for the presented case studies. The second tool is a commercially available tool that integrates BPMN modelling and simulation in one software, which is the Bizagi Modeler. The third tool is the one developed specifically for the manufacturing domain and promises the capability of capturing complex manufacturing aspects using its set of notations and at the same time be able to simulate these notations, which is the MPMN Simulator. Sets of experiments are then conducted using the presented case studies to compare the modelling tools and to assess their modelling capabilities. For each of the case studies, the three tools are used for modelling and simulation, models developed are verified and validated in relation to effectiveness, efficiency, and usability by analysing models' performance.

Building on the previous paragraphs, the main contribution of this work is proposing MPMN; MPMN extends BPMN with new elements to accurately model manufacturing processes via a set of more meaningful core modelling elements that better describe the manufacturing system. Also, this contribution is evolved and MPMN is integrated with simulation, this integration presents the MPMN Simulator, which is considered a substitution to BPSim, the Business Process Simulation Standard that integrates BPMN and simulation to model and execute business processes. The MPMN Simulator offers new simulation capabilities to model manufacturing processes, it is proposed to give the non-simulation experts the advantage of building models using understandable notations and simulate it to retrieve reliable outputs to make decisions upon.

1.1 MOTIVATION OF THIS WORK

There is a large and growing number of BPMN extensions that are proposed in the literature targeting various objectives; BPMN extensions are either proposed to be able to represent the processes of a specific domain, or extensions are introduced to improve the BPMN language itself. Also, an expressive and understandable modelling language is needed to model manufacturing processes. BPMN notations lack the ability to represent some important aspects when modelling manufacturing processes. Consequently, in the present work, BPMN notations are extended to model the specific domain of manufacturing and offer a set of new notations that include missing aspects that are needed to model a manufacturing system.

Moreover, manufacturing process modelling has to evolve, and static models have to be translated into a suitable discrete event simulation environment to be able to analyse, experiment, and improve the system under study. Thus, the proposed set of notations is integrated with simulation to be able to encompass various manufacturing complexities by building and simulating manufacturing processes. This extension contributes to business process management (BPM) activities that aim to improve the BPMN language. Moreover, this extension can be considered as an extension that is independent of a specific domain.

1.2 AIM

This work aims to develop an understandable set of notations that is able to model processes specific to the manufacturing domain. These notations should also be able to easily produce quantitative data that are useful in describing manufacturing systems' behaviour.

1.3 OBJECTIVES

The following objectives have been identified to track the progress of the research and ensure that the aim is achieved:

- To identify whether BPMN 2.0 is capable to model manufacturing processes or not.
- To assess the capability of simulation BPMN models in quantifying the performance of manufacturing systems.
- To extend the BPMN set of notations to model manufacturing processes.
- To incorporate full simulation capabilities in the extended BPMN set of notations.

1.4 THESIS OUTLINE

The thesis consists of eight chapters and two appendices.

- Chapter two covers a review of the literature offering background and related work to this research.
- Chapter three explains the research methodology and how this work has been developed.
- Chapter four presents the contribution of this work; it introduces the development of MPMN modelling and simulation aligned with a full analysis of the proposed approach.

- Chapter five includes modelling and simulation of a simple flow shop manufacturing system, which will introduce new processes to the BPMN like cutting, batching, setups, rework, etc.
- Chapter six includes modelling and simulation of a complex manufacturing system. This case study is selected to assess BPMN and the developed tools' capabilities to model more complex operations.
- Chapter seven presents the discussion.
- Chapter eight sums-up the conclusions and offers recommendations for future work.

2. LITERATURE REVIEW

This chapter provides a review of the previous research works regarding Business Process Model and Notation (BPMN). The chapter first starts with an overview and background of BPMN, and the importance of its existence. The research approach and the objective of the literature review are provided in the form of questions.

Afterward, a more comprehensive review on previous works related to BPMN is presented. This includes a review on BPMN applications, BPMN and manufacturing, detailed discussion on BPMN extensions, and presentation of how BPMN and Simulation are integrated in the previous publications. An analysis of the literature review is then conducted to present and critically evaluate the state of the art in this research area. Finally, identifying the gaps that should be bridged in this research area based on the review that took place in the research presented.

2.1 OVERVIEW AND BACKGROUND

In today's dynamic and competitive world of business, systems need to be prompt and adaptable to quick changes. Recently most companies are adopting a business process management (BPM) strategy, which ensures continuous improvement of business processes (BPs) and agility. As mentioned earlier, modelling is one of the most crucial activities of BPM, and the initial phase in the BP lifecycle. Business process modelling is an essential part of understanding and redesigning the activities a typical enterprise uses to achieve its business goals [13]. To decrease the cost of corrections and to early identify the errors and defects during the modelling phase, all relevant stakeholders who understand the whole system must be involved. Among all existing modelling languages, Business Process Model and Notation (BPMN) has become the de-facto standard for BP modelling [16], and it has evolved to become one of the most commonly applied modelling languages in the business process management discipline [17]. It provides businesses with the capability of understanding their internal business procedures in a graphical notation and can give organizations the ability to communicate these procedures in a standard manner [18].

2.1.1 Business Process Model and Notation

The Business Process Model and Notation (BPMN) has gained wide acceptance both practically and academically; it has recently emerged as the industry-standard notation for modelling business processes [20]. BPMN has been developed by Stephen A. White in 2002 [21], within the scope of Business Process Management Initiative (BPMI), and is currently maintained by the Object Management Group (OMG) since the two organizations merged in 2005 [22]. "It aims to provide a common language for modelling business processes, to replace the multiple competing standards that currently exist" [23] as stated in its latest release.

BPMN's main goal is to "provide a notation that is readily understandable by all business users, ranging from the business analysts creating initial drafts of the processes, to technical developers responsible for implementing the technologies performing those processes, and finally, to the business people managing and monitoring those processes" [18, 19].

BPMN is created using a set of graphic elements Flow Objects, Connecting Objects, Swimlanes, and Artifacts [20, 24]. These elements enable the easy development of simple diagrams that will look familiar to most business analysts because it is based on flowcharting technique; the elements were selected to be distinguishable from each other and to utilize shapes that are familiar to most modellers. For example, activities are represented as rectangles and decisions as diamonds [25].

The four main pillars of BPMN are:

1. Flow Objects, which fall into three categories: events, activities, and gateways. These are the most basic elements used to create BPMN models.

2. Connecting Objects are the lines connecting the flow objects through three different types of connectors; Sequence Flow, Message Flow, and Association.

3. Swimlanes are used to organize processes in a BPMN diagram. Activities are grouped into separate categories for different functional capabilities or responsibilities.

4. Artifacts in BPMN are data objects, Groups, and text annotation, and they are used to add additional data or notes to describe the process [26, 27].

2.1.2 Approach to Literature Review

The present study is motivated by findings of a preliminary selective review of literature showing that the body of literature on BPMN applied to manufacturing is unexpectedly limited and fragmented, while recognised as a vital modelling language at the core of business interactions.

Thus, in the current work, the approach of literature starts with a review for prior efforts on BPMN appearance in the domain of business and manufacturing published between 2002 and 2020. The main objective of the present literature review is to provide a structuring overview of BPMN, particularly in regard to application purposes, problems tackled, extensions proposed and related approaches to simulate models. This is done to identify research gaps and suggested potential paths for improvements.

2.1.3 Objective of Literature Review

A literature review is conducted to answer the following questions:

- Why BPMN is selected among all the available standards?
- Where has BPMN been applied, and how does modelling using BPMN affect the area of application?
- Does BPMN go under revisions for an extension? Why?
- Has BPMN been applied in Manufacturing? And how BPMN is extended to suits manufacturing domain?
- Why BPMN and BPSim are integrated?

In order to make sure that high-quality and relevant material has been selected during the literature review, certain search terms were used and examined. Over and above, this review is conducted using several databases and search engines, these include, ScienceDirect, Emerald Management Xtra, JSTOR, Google Scholar, and IEEE Xplore.

Also, this review targets all the work published in journals, conferences, workshop proceedings, and book chapters. Key authors that have significant work in the field of BPMN are also mentioned; all the work that is cited by them is encountered and some of the work that cites their work is included. For this, title and abstract were used for initial screening, and papers were filtered according to the inclusion and exclusion criteria presented in the following table:

Table 2-1: Inclusion and exclusion criteria.

Criteria	Reason for Inclusion/ exclusion
Abstracts that contain one or more of the following keywords: BPMN, BPMN and applications, BPMN and extensions, BPMN and simulation, and BPSim are encountered.	To ensure that relevant articles applicable to the research field are gathered and observed.
Articles in all sciences, but the English language are viewed	To check and cover how BPMN was applied in a wide range of topics
A time frame of articles from 2002 till the present day was examined	To gather all important articles since the development of BPMN in 2002, however, the recent ones may be of greater significance
Extensions and improvement to BPMN are encountered	To ensure that the papers are relevant to the problem definition.
BPMN integration with other languages and with simulation was studied	To experience the different approaches and solution techniques that could be used in such an area
All outcomes and results related to BPMN in the field of manufacturing and business are included.	To highlight the opportunities for development
Papers that extend other modelling languages or execution languages like BPEL are excluded	Those kinds of papers are out of the research scope.

There have been several papers in the literature on modelling manufacturing systems [28–31]. The literature review reports the academic publications about modelling and its application in the manufacturing and business since 2000. Papers were examined to find how manufacturing and business systems have been modelled, which of the modelling languages were commonly applied to manufacturing, and how does modelling affect real-world application?

The literature search was conducted using the keywords combination ("BPMN" OR "Modelling Languages" AND "Manufacturing" OR "Business"), ("BPMN" AND "applications" OR "extensions"), ("BPMN" AND "simulation"), and ("BPMN" AND/OR "BPSim"). Papers with irrelevant titles were excluded, further filtering, based firstly on abstract reviewing and secondly on full-text reading, resulted in a set of 117 relevant papers.

2.2 BUSINESS PROCESS MODEL AND NOTATION BPMN

The next section will discuss why BPMN was selected among several available techniques and will also state its applications and determine the current state of the art of BPMN extensions, this to identify the gaps that should be filled in this research area.

2.2.1 BPMN and Other Modelling Languages

The evaluation and comparison of BP 'Business Process' languages that have achieved widespread adoption in industry practice have been addressed in literature [32–34]. Formally defined Business Process Modelling Languages can be compared in terms of their expressive power [35].

It is worth mentioning that BPMN was not developed to compete with existing BP languages like (Unified modelling Language Activity Diagram (UMLAD)), Business Process Execution Language (BPEL), Yet Another Workflow Language (YAWL), and XML Process Definition Language (XPDL), but rather for complementing them [18]. UML and Systems Modelling Language (SysML) are object-oriented languages, whilst BPMN is based on a process-oriented approach that is more suitable within the analysed domain or in other words [36]. BPMN has been intended to be used as an alternative language by business analysts without technical knowledge. On the other side, different authors claimed that BPMN is better than many other languages; firstly it emerged and showed proven success over existing notations, and secondly because it gives a full description of its graphical notations and it includes a large number of symbols that can be combined to represent several applications [19]. Besides, BPMN was selected as a modelling language for discussion and extension for evaluation and analysis of business processes because it is believed that BPMN is more expressive as compared with other modelling languages [37].

A study was conducted to assess the comprehension of a selected business process modelling notation, among the three notations that were subjected to diagnosis Business Process Management BPM practitioners, perceive BPMN as easy-comprehensible notation [22, 38]. Next, BPMN applications and extensions will be presented.

2.2.2 BPMN Applications and Extensions

In the following section, some of the BPMN applications in different domains will be tackled, and some of the efforts to extend BPMN will be shown. Building on that, the steps needed to extend BPMN is enumerated.

Applications

BPMN is the de-facto standard for representing in a very expressive graphical way the processes occurring in virtually every kind of organization one can think of, from cuisine recipes to the Nobel Prize assignment process, incident management, e-mail voting systems, travel booking procedures, to name a few [2]. BPMN is specified as General Purpose Language, which means that the modelling concepts are generic elements in the context of business processes, without any domain characteristics. Thus, the BPMN is not restricted to a single domain [17]. For example, BPMN has been used in the service management sector to analyse processes for better communication [39], or to document workflows and to identify areas for improvement [3]. It has been used in the supply chain sector to model actors and processes [40] and to represent complex flows [41].

Despite BPMN being widely used, when it comes to describing several domains this business process modelling language is lacking; BPMN is either needed to be extended or integrated with other modelling language to serve a specific purpose. In order to overcome this shortcoming and expand the use cases of BPMN, the OMG (2013) has introduced an extension mechanism allowing users to integrate new elements and provide valid BPMN extensions [18].

Some domains such as healthcare and Internet of Things (IoT) [42] are focused to propose extensions to BPMN to suit their needs. In the coming section, some of the previous extensions of BPMN inside and outside the scope of manufacturing are covered, in order to present its state of the art and emphasize existing shortcomings.

Related work on BPMN Extensions

BPMN has evolved to one of the most applied modelling General Purpose Languages in the Business Process Management discipline, and due to its application in different domains, there is a frequent necessity to extend the BPMN either for dealing with processes of specific domains or for improving the language itself [17]. The BPMN language is widely used for its expressiveness, simplicity, and semantic richness, still, some authors believe that domain-specific process modelling languages (DSPML) can have significant advantages over BPMN. However, domain-specific language development is hard, requiring both domain knowledge and language development expertise [43]. Business process designers choose to extend BPMN and reuse its kernel, to take advantage of its benefits (e.g. standardization, tool support) instead of developing a domain-specific modelling language from the scratch, which is very costly and time-consuming [48].

BPMN provides the possibility to adjust a general-purpose process modelling language for domain-specific elements. In that way, domain-specific modelling approaches can make use of the strengths of the BPMN (e.g., standardized stable core elements; popularity; expressible elements) and add their specific concepts. Thus, the level of reuse is very high, which could lead to a foreshortening of the development time or the reduction of design faults. It seems to be more promising to focus on the development of language extensions instead of building dedicated languages, which need to contain all basic process modelling aspects (e.g., nodes, control flows, decision points) [17].

Since the introduction of the BPMN extension mechanism in 2011, BPMN has been extended by numerous contributions to describe different process characteristics across distinct domains [9]. Accordingly, a large and growing number of BPMN extensions are proposed in the literature targeting various objectives such as the representation of domain-specific BPs (e.g. healthcare, IoT, and manufacturing) or the improvement of the BPMN language itself (e.g. flexibility, complexity, and variability).

Only two existing literature review papers focus on work extending BPMN; Braun and Esswein [8] in 2014, and Zarour et al [1] in 2019. In this work, the systematic literature review conducted by Zarour et al [1] is considered a foundational brick to determine the current state of the art of BPMN extensions and identify the gaps that should be filled in this research area. The authors have classified 52 extensions published between 2014 and 2018, and they have answered important questions regarding the targeted area; for example, what are the areas and goals targeted by BPMN extensions these last years? What are the formats used for the representation of BPMN extensions? And how are BPMN extensions demonstrated, implemented, and evaluated?

It is worth mentioning that Zarour et al [1] stated that more than half of the extensions proposed to BPMN do not conform to BPMN extension mechanism. However, Zor et al [10] reported that in order to create a BPMN-compliant extension, there are some rules which have to be followed: the extensions must not contradict the semantics of any element that is defined in the specification, the graphical elements should be easy to understand by any viewer of the process diagram, and the extension elements should have the ”look and feel“ of BPMN.

In the next paragraphs, some of the extension efforts mentioned in the paper will be discussed, in addition to the most cited extension efforts to let the reader build basic background knowledge on BPMN extension and position the suggested proposal.

Extension areas are diversified, and different domains are targeted, and among these domains’ healthcare is the most targeted one. The interest in healthcare is justified by the fact that it is very broad, very sensitive and highly regulated [1]. For example, some authors were interested in extending BPMN and adding time constraint in the modelling of healthcare processes related data [45–47], others offered an extension to BPMN and apply it in the healthcare domain to show its applicability [48, 49].

However, numerous extensions are proposed to BPMN, some of them to represent or handle a particular domain and other extensions are proposed to improve the BPMN language and can be used in any domain. These extensions are given an 'extension name' and examples of the most cited references will follow. The Simple Business Process Modelling "SBPMN" [50], a simplified business process modelling notation BPMN-MUSIM [51], and (BPMN + tailoring) "BPMNt" [52]. These extensions were proposed to enable quick modelling of processes through simple and clear notations to improve BPMN expressiveness and flexibility, or extensions like "PyBPMN" Performability-enabled BPMN (PyBPMN) to improve business process performance [53]. Another extension of BPMN is offered to include time constraints on an activity and to model precedence dependencies between two activities is the "Time-BPMN" [14, 54]. Other extensions are proposed to serve specific domain like ubiquitous BPMN "uBPMN", an extension for modelling ubiquitous business processes [55] and extending BPMN with reliability information "relyBPMN" to serve healthcare domain [56, 57].

Building on the previous paragraphs, some authors [22, 38] confirmed that BPMN is an easy language to work with, and an appropriate tool for business process modelling; they believe it will standardize business interactions, and it was adopted to model several processes representing different domains. Others [21, 34, 58] questioned its understandability; they found BPMN is incapable to rely on it only, and it has to be integrated or complemented with other languages; BPMN has gone through several extensions and enhancements to embrace several processes representing different sectors.

In the following section modelling in manufacturing and precisely the application of BPMN in manufacturing systems will be discussed.

2.3 MANUFACTURING AND MODELLING

2.3.1 Manufacturing Process Modelling

In the last decade, and up till now, manufacturing process management focuses on the management of process data, and before starting to apply any approach of improvement, it is a must to have a graphical representation of a logical sequence of workflow or steps constituting our manufacturing process, from raw material to the finished good. The essence of the concern is to streamline and control the flow through value-adding processes and to eliminate superfluous processes [9].

The different aspects of manufacturing process information and the relationship between them are complex; it is indispensable to have a clear and accurate description of manufacturing process information modelling, which has been a shortcoming of the existing literature [28].

To start modelling the manufacturing systems domain, one of the first steps is the selection of a precise and proper language [59]. The Integrated DEFinition (IDEF) series (IDEF0, IDEF1, IDEF1X, IDEF-TD, IDEF2, IDEF3) appear to be the most favoured in manufacturing [60]. However, modelling languages are always selected to serve a specific purpose; there are four main requirements to ensure the proper selection of a modelling language to model the manufacturing domain [59, 61]:

1. Must be general enough and adaptable to describe different production systems and allow data storage
2. A graphical notation has to be provided to offer easy use and maintenance
3. They must support interoperability; machine readability, bridging to another model, representation of universal and domain categories, and a proper link to web standards.
4. Should support the automatic reasoning; finite time reasoning, time-efficient reasoning, formal semantics, and inheritance.

In the next section, there will be a discussion about matching the above requirements with the suggested solutions.

2.3.2 BPMN and Manufacturing

Evidence from the literature shows the interest of various researchers in modelling different manufacturing processes and systems using BPMN. However, most of the work was focused on using the BPMN standard notations, without introducing any extensions to these set of notations, to develop their models.

BPMN was used in literature to model the functions of a manufacturing execution system such as production scheduling, dispatching of orders, resource acquisition and process planning. It was mentioned that BPMN allows the definition of workflows in a simple and flexible way, as well as simplifying understanding among all participants [62, 63]. In these cases, BPMN was used without offering any extension. However, theses models addressed the management of the production system rather than the system itself. This is not the same focus of this work that targets the extension of BPMN to model the actual manufacturing processes.

Estruch and Álvaro [64] developed BPMN models for simple manufacturing scenarios with a focus on event-driven modelling such as quality and maintenance events and claimed that it was possible to use the widely extended BPMN notations to handle complex events and to express, at the same time, the main manufacturing process activities. Also, Lodhi et al [37] used the original BPMN notations to model a manufacturing example; however, they extended BPMN in the development of meta-models for post-execution analysis and improvement of business processes.

BPMN was also compared to established modelling tools for manufacturing processes such as the IDEF3 Process Description Capture Method and Value Stream Mapping (VSM) in an attempt to position BPMN among the existing notations for modelling manufacturing systems and their processes. The three methods were used to model a tobacco manufacturing process and it was concluded that BPMN cannot model the existing objects and their transitions like IDEF3 object schematics can; additionally, VSM is a much simpler notation than BPMN that focus on the material and information flows rather than the exact sequence of operations [65]. For this reason, VSM complements BPMN and is also probably the reason why others have looked into integrating VSM and BPMN to model manufacturing processes such as the work of Zor at al [66].

Even researchers who rely on BPMN solely to model the processes of products manufactured by machine tools had to provide an extension to BPMN elements, and to complement it with other modelling languages to represent modelling the physical object flow, the continuity of object flows, and object semantics [67]. Michalik et al [68] modelled the function of maintenance management in a production line and confirmed, based on their modelling experience, that BPMN is not fully sufficient to model complex systems. They also used a scoring method to compare modelling the maintenance function using BPMN and UML and concluded that the BPMN standard is better than the UML Activity Diagram on all criteria selected for comparison.

In 2019, Abouzid and Saidi [9] stressed the need for a proper modelling approach for business processes in the manufacturing domain and presented some preliminary work that extends BPMN notations and their application to the manufacturing domain. Specifically, they extended BPMN by presenting two new notations to BPMN set of notations: one for tasks scheduling and another one for inventory.

Recently [1] there are two extensions made to BPMN to enhance manufacturing process modelling, and these two extensions are classified as extensions offered to the manufacturing domain specifically. The first one is proposing a BPMN extension that enables a process-oriented human physical risk assessment in the manufacturing domain [69]. The second one is proposing notations of tasks scheduling and inventory in which allows putting much more information into the process model [9]. Zor et al [10] and Braun and Esswein [70], are the only researchers who proposed customized notations to explicitly represent some crucial concepts of the manufacturing domain; specifically, manufacturing processes, resources, and material flow.

Since BPMN notations lack the ability to represent some of the manufacturing aspects and handle its growing complexity, the following section will discuss the previous efforts to extend BPMN by integrating it with simulation, and how this can support manufacturing process modelling.

2.4 BPMN AND SIMULATION

It was only beginning in the 1990s that simulation approaches began to appear more frequently in major management journals, and researchers in operations management are among those who showed major interest in a computer simulation. The simulation-based approach has also been a powerful tool in the study of BPs and operational performance. Process simulation plays an important role in the context of business process management (BPM). The purpose of the simulation is to give business analysts insight on what the performance and the cost of processes could be according to the actual process design and to provide hints on the corrective actions to take in order to improve the overall process performance [71].

BP simulation is a widely used approach in business process management (BPM); BPM aims to analyse BPs and their interactions, identifying potential improvements as support to decision-makers, in a manner that addresses the multidisciplinary nature and the complexity of the true processes [72]. Simulation techniques are successfully applied to analyse and validate the performance of a business process (BP) since the early phases of its lifecycle when the BP representation is commonly specified in BPMN. The BP simulation model is first to be built from the BPMN model, then implemented and finally executed to yield the performance indices of interest [73].

2.4.1 Simulation of Business Processes

Combining both modelling and simulation of business processes can assist in first visualising process behaviour, then measuring process operational performance, and finally examining the different scenarios for improvement [72]. Wagner [74] explored adding simulation to BPMN, he claimed that using BPMN as a basis for developing a process simulation modelling approach is the best choice considering other modelling languages, like UML. This is due to its great expressivity and wide adoption as a modelling language. However, there are no guidelines and no best practices on how to use BPMN for simulation modelling, because it is not specifically designed for this purpose, accordingly, BPMN has to be modified and extended to serve simulation modelling. Wagner [75] stated that several authors proposed different approaches to use BPMN for DES modelling, but still there is no agreement on how to define activities in DES. Wagner first proposes DES approach, to define activities as complex events having a start event, an end even. Then, he proposed 'The Discrete Event Process Modelling Notation' (DPMN) which extends and modifies the language of BPMN Process Diagrams to make event rule design models and process design models [76].

BPMN is seen as the right candidate if compared to more complex and general-purpose system modelling languages, such as UML and SysML to be used by non-simulation expert [36, 77, 78]. The expressiveness of BPMN has contributed to make it the preferable choice to help non-specialists of modelling and simulation to define in a unified and standardized way the execution process of the simulation [76].

However, to date, it is still vague when it comes to optimizing the use of BPMN, and how to adapt its syntax and semantics, for simulation modelling, but simulation researchers still believe that BPMN is the best option in comparisons to other standards available for linking with simulation [79, 80].

2.4.2 Business Process Simulation (BPSim 2.0) Standard

BPSim 2.0 is the integration of BPMN and Simulation and is considered an extension to the BPMN graphical representation. This standard defines the parameterization and interchange of process analysis data allowing structural and capacity analysis of process models (www.BPSim.org) [81]. It is developed by the Workflow Management Coalition, and it allows business process models to be augmented with simulation-specific parameters such as task durations, probabilities, resources, costs, and other information that supports the implementation of simulation modelling for the purpose of analysing and improving the system under study before implementing it in real life [40, 41].

Currently, BPSim 2.0 is the only standard specifically designed to support the simulation of a BPMN diagram. BPSim 2.0 groups the information needed to simulate a BPMN model into six categories (time, control, resource, cost, property and priority), and it has promising capabilities, but BPSim 2.0 also has some major limitations [84].

As mentioned earlier, the most targeted domain that uses BPMN is healthcare, thus integrating BPMN and BPSim 2.0 to improve healthcare domain was of some authors' interest. Bizagi modeler was used to manage modelling using BPMN and simulation using BPSim 2.0. This integration showed the effectiveness of BPMN to describe process model and BPSim 2.0 was able to complement BPMN for execution and analysis [72]. However, for more complex flows BPMN and BPSim 2.0 integration in Bizagi failed to represent some complex flows. BPSim 2.0 is not to be used for the simulation but instead, it is recommended to use full discrete-event simulation software. So, the structure of the simulation model built as a process map in BPMN was imported into a discrete event simulation software. BPMN is a standard modelling language that is non-software dependent. Hence, a model represented using BPMN can be used by any simulation software that supports the standard. This provides flexibility for users to choose a simulation package based on their existing knowledge or value-for-money and reduces training costs because once users are familiar with BPMN, the knowledge is transferrable to any simulation software that supports this standard [85].

To sum up the previous paragraphs, BPMN provided a quick and easy to use process mapping tool producing straightforward understandable visualisations to stakeholders. It provides detailed modelling through the hierarchy of processes and sub-process, and it can capture and represent complexities like resource responsibilities through pools and lanes. BPMN in Bizagi helped overcome the problem of simulation modelling being a time-consuming activity, due to the lack of simulation awareness and poor coding capabilities. However, integrating BPMN and BPSim 2.0 to represent complex flows appeared to be insufficient; BPSim 2.0 reads a model written in BPMN and adds the extra information that is needed to simulate the BPMN model. The main advantage of this extension is that it leaves BPMN as it is. The main disadvantage is that the capability of BPSim 2.0 is limited by the limitations of BPMN [84]. Thus BPMN and BPSim 2.0 are not yet mature for representing many important flow features [85].

2.5 CONCLUSIONS

The growing amount of complexity and enterprise data creates a need for novel BP analysis methods to assess the process optimization opportunities [86]. Therefore, modelling has been at the core of organizational and information systems design, and business process models are fundamental to all the phases of the BPM lifecycle and cover different aspects depending on the final modelling purpose [87]. It is desirable that a Business Process Model can be understood by the various stakeholders involved in the most straightforward manner possible. This could be best achieved through the use of graphical representations [35].

Integration of complex engineering projects draws the attention to the need for flexible and adaptable standards to model the whole system, among all of the available languages; BPMN has drawn the attention of enterprise engineers, since it allows the definition of workflows in a simple and flexible way, and also facilitates understanding among all participants [62]. Since research in the field of BPMN extensions is very active, several extensions have been proposed in the last few years, as some authors believed that BPMN 2.0 is not sufficient to model complex systems [85]. Thus, BPMN is extended to improve the language itself to better describe the system, or it is integrated with one of the modelling languages to enhance modelling of complex systems [66].

Few works have addressed the application of BPMN to model manufacturing systems [9, 65, 66]. This is because BPMN set of notations lacks important manufacturing aspects such as the flow of physical material, buffers, and queues [88]. In fact, BPMN might be lacking the required notations to model complex flows in the manufacturing domain [62], and there is a need to develop an extension to BPMN that is capable of representing the activities and missing elements in the manufacturing environment [9, 10].

Proposing extension to BPMN could enable the easy of manufacturing models representation, yet, data analysis and validation of BP performance are still needed [85]. Thus, it might be a promising approach to investigate the potential for integrating BPMN and a full DES software in manufacturing.

Consequently, as others did in different domains [9, 55, 71, 89–91], BPMN's appealing features were the drivers behind selecting BPMN as the specification on which to ground the research's modelling needs. To extend BPMN notations and link it with full DES to deal with the manufacturing complexities.

The following chapter will discuss the research methodologies and methods, how this work has been developed, and a detailed explanation of the process of the research development will be declared with the aid of a BPMN diagram.

3. RESEARCH METHODOLOGY

This chapter defines the procedures and techniques used to reach the research objectives. The chapter starts with presenting some of the most commonly used research methodologies that are applied to BPMN in general. Then, insights on the research methodology proposed in this work are presented in detail, the process involved to develop this research and the tools selected for use in this work.

3.1 PREVIOUS RESEARCH METHODOLOGIES APPLIED FOR BPMN

Methods refer to techniques used to acquire and analyse data to create knowledge. There are several research methods and the choice of which method to employ is dependent upon the nature of the research problem [92]. Three main research methods used in conjunction with BPMN were reported in the literature: case-based, model-based, and survey-based research methods.

3.1.1 Case-Based Research Method

The case-based research method appeared in several previous works as one of the research methods used in conjunction with BPMN modelling. Case-based research was criticised by some as an approach that lacks scientific rigour and reliability; also, as one that does not address the issue of generalizability. Still, there are some strengths of case-based research method; it enables the researcher to gain a holistic view of a certain phenomenon or a series of events and can provide a full picture when many sources of evidence were used. Moreover, using different case studies to examine a certain phenomenon enhances the accuracy, validity, and reliability of the results; the development of consistent findings over multiple case studies can then be considered a very robust finding [92].

BPMN was used to model processes in different case studies; for example, BPMN was used to model a diagram of hotel service processes to assess its suitability [93]; also, BPMN was used to model the interactions of roles, activities, and artifacts in a software company [94]. While, other case studies used BPMN to confirm or to evaluate a given approach of modelling with BPMN or to assess an extension offered; for example, Brown, Recker, and West [95] suggested a three-dimensional BPMN modelling environment that adds to the collaboration development of process models and used a case study to evaluate the suggested approach.

3.1.2 Model-Based Research Method

A model is a representation similar to, but simpler than, one or more processes and systems in the real world. Modelling is used to explore theoretically possible scenarios within a set of system states and boundary conditions. Simulation of a model allows behaviour and performance study of real-world systems that would otherwise be too complex, expensive, dangerous or time-consuming to undertake. Furthermore, simulation of a model can be used in evaluating designs and plans, to reduce system failure, to meet specifications, to avoid possible bottlenecks, and, in general, to optimize system performance [96]. BPMN graphical models are used to represent resources, this is to propose virtual process simulation as a technique for identifying and analysing uncertainty in processes [97]. BPMN was also used to model workflows of material supplies within construction projects using the Signavio BPM Academic Initiative platform [98].

3.1.3 Survey Based Method

The essence of the survey method can be explained as "questioning individuals on a topic or topics and then describing their responses". Survey method can be used in both, quantitative, as well as, qualitative studies. There were several papers identified in the literature that used the survey method to question BPMN understandability, to picture the familiarity with the BPMN elements, and to identify the preferred and most used BPMN tools [2, 99–101]; others conducted a comprehensive series of interviews on the use of BPMN [102, 103].

This work integrated two methods: model-based research method and case-based method. Combining different techniques for eliciting data is said to strengthen and confirm results.

3.2 PROCESS OF RESEARCH DEVELOPMENT

3.2.1 Research Methodologies Used

First, the case base method is selected to test the proposed extension and to understand its applicability and usefulness to model manufacturing processes and systems. Two case studies are used to assess BPMN capabilities to model the specific domain of manufacturing. These two case studies cannot be considered as multiple, but they reflect the proposed BPMN extension suitability at different levels of complexities. The first case study is a real-life example; it was observed that almost all the authors that provided examples to demonstrate their extensions are derived from real-world scenarios [1]. The second case study is of a semiconductor wafer fabrication facility, which is one of the most complex manufacturing environments [104, 105], due to the phenomenal and complex nature of the process of microchips manufacturing [106]. The Mini-fab model is selected for this work as a representative model of the semiconductor wafer fabrication facilities. Though the Intel Mini-fab model is small, yet it represents the challenges of a semiconductor fab with the inclusion of re-entrant flow, operators, batching, machine failures, setups, loading and unloading [107], multi-product processing, preventive maintenance, and transportation times, among others [108].

Secondly, a model-based approach using discrete-event simulation models is used in this work to compare the capabilities of the proposed approach to existing tools in assessing the true performance of the case studies and to enrich the static models with quantitative data useful for studying and analysing a systems performance.

Finally, the survey-based method was excluded, as all the papers questioning the BPMN understandability and suitability through questionnaires and surveys and this is out of this research purpose.

3.2.2 Main Steps of The Work

This work sheds new light on the issues of using BPMN to model complex manufacturing systems and these issues are being addressed using static and dynamic models. Figure

3-1, shows the main steps of this work and how this research was developed. The model describes the whole work through a series of linked processes.

Three main parts are easily identified for the research methodology; the first part is the literature review, which is a review of the existing literature on BPMN as shown in the previous chapter.

The second part is the static part of the thesis, where different case studies are modelled graphically using BPMN to assess BPMN manufacturing processes modelling capabilities. As a result of BPMN assessment the main contribution of this work emerged; the MPMN approach is developed, where modifications are done on existing BPMN notations and new notations are created to accommodate manufacturing aspects.

In the third part, the proposed extension is integrated with simulation. MPMN Simulator is proposed to model case studies, and these models were experimented and analysed to discuss issues and report results. The process of the thesis development will be described in the following part.

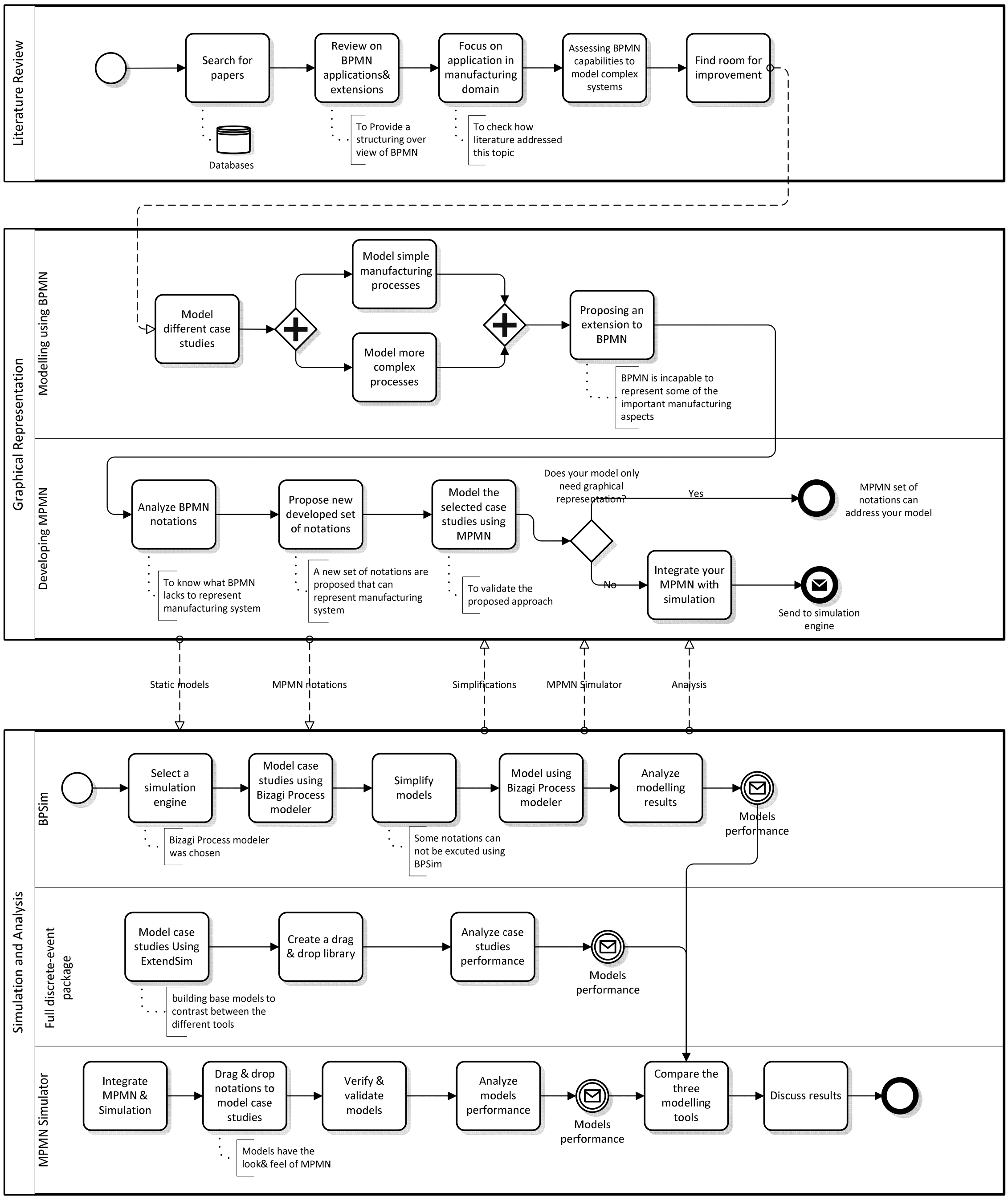

Figure 3-1: Research methodology.

3.2.3 Graphical Representation

The graphical representation pool has two lanes; one lane is for modelling case studies using BPMN and the other lane for MPMN (Manufacturing Process model and notation) the proposed extension to BPMN notations. Two manufacturing systems with different complexities are selected as case studies to assess the capabilities of BPMN and recognize how effective BPMN is to describe each system. The first case study is a simple manufacturing process at the shop floor level, and the second is a representative model of semiconductor wafer fabrication facilities that include re-entrancy, batching, and changeovers. These two case studies are selected to examine the ability of BPMN language to model manufacturing systems of different levels of complexity.

Modelling of the two systems showed that BPMN current notations lack the ability to describe some of the aspects of manufacturing systems. While, in other instances, BPMN is able to describe manufacturing-specific characteristics; however, in a more complex way.

Accordingly, modelling the two case studies is followed by suggesting a more manufacturing-specific set of notations that can support modelling complex manufacturing systems' characteristics. This has led to developing the Manufacturing Process Model and Notation (MPMN) as an essential modification to BPMN in order to support the modelling of manufacturing systems. An approach involves adding some modification to the existing BPMN notations and developing new notations that can accommodate manufacturing characteristics.

The two case studies were modelled using the newly developed MPMN notations to validate the approach capabilities; the simple newly added notations enhance the representation of the exact picture of the manufacturing processes, but it still lacks the ability to improve or assess the system under study. Thus, to analyse, optimize or improve models, a simulation engine is needed to execute these models.

3.2.4 Simulation and Analysis

First, Bizagi modeler is used to model the two case studies. Bizagi modeler is selected to build and simulate the BPMN models, where, parameters can be added to the BPMN elements and execute simulation models. Bizagi is used to model the two case studies in order to assess the standard ability to represent and simulate the cases. It is selected because it is a free tool and supports both BPMN and BPSim 2.0.

The ExtendSim simulation is selected to model the two case studies, since, full discrete event package capabilities are needed. Models developed in ExtendSim environment will act as base models to analyse other models' performance. This work is intended to substitute BPMN and BPSim 2.0 by integrating MPMN and a full discrete event simulation software resulting in the manufacturing domain-specific MPMN simulator. This simulator is supposed to improve the illustration of complex manufacturing systems and to elaborate on the formulation of simulation models. The proposal is to create an MPMN simulator, which has the capabilities of building models that have the same look and feel of MPMN, along with powerful simulation capabilities. Models created using MPMN simulator will be validated and verified and results will be compared with the output of the ExtendSim base models.

Chapters 5 and 6 will describe the two selected case studies; each case study will be modelled using BPMN and MPMN. Each model will be explained, and the challenges and opportunities will be pointed out. Developed models are also simulated using the three mentioned tools.

3.3 RESEARCH TOOLS

3.3.1 Standards

First, the *BPMN 2.0* is used as it is considered the de facto standard for graphical process modelling and it is widely supported by both free and commercial process modelling tools. The second one is *BPSim 2.0*, which is the integration of BPMN and simulation. It is considered an extension to BPMN, where parameters can be added to BPMN elements and formulate simulation models.

3.3.2 Modelling and Simulation Tools

Graphical Modelling

Microsoft Visio is used to build BPMN models because Visio includes a template that contains the graphical elements described by the BPMN 2.0 specification. *Bizagi Modeler* is the second tool used. It is selected since it is one of the most representative commercial tools at the time of writing, and it claimed to have a very user-friendly interface, it is used to build simulation models using BPMN and BPSim 2.0 notations.

Models Simulation

Bizagi Modeler is selected to model case studies as it integrates BPMN and BPSim 2.0 to build simulation models. Another one is used to simulate models is the *ExtendSim*. It is one of the powerful full discrete-event simulation (DES) environments, besides it was not only the first “drag and drop” simulation program, but it was also the first graphical simulation tool to embody the concept of modelling components as objects.

4. DEVELOPMENT OF MPMN MODELLING AND SIMULATION

As mentioned earlier, the Business Process Model and Notation (BPMN) has seen a huge uptake in both academia and industry over the past years. It is seen by many as the de facto standard for business process modelling and has become very popular with business analysts, tool vendors, and end-users.

This chapter begins with a complete analysis of BPMN notations, the primary goal of this analysis is to assess BPMN notation's capabilities by providing a base of knowledge of the BPMN notations and defining their purpose and how it can be used in a manufacturing setting. This will lead to a clear understanding of each notation and will be the guide to develop the MPMN that includes the missing modelling notations that are needed to accommodate manufacturing process modelling. Analysing and extending notations will be followed by proposing full DES capabilities to MPMN for data analysis and improvements.

It should be noted that based on the research methodology presented in the previous chapter, identifying the need for new MPMN notations and integration with the full DES software was partly based on the case studies; yet, the case studies will be detailed later in Chapter 5 and Chapter 6.

4.1 BPMN ANALYSIS

Figure 4-1 represents an overview of BPMN graphical notations; as of version 2.0, the BPMN contains a comprehensive set of concepts and notational elements; BPMN 2.0 specification spans more than 500 pages and the definitions of these elements are distributed across various sections. Thus, the difficulty of notations understanding and lack of models' validation issues may arise [109].

There are four BPMN notations categories: flow objects, connecting objects, swimlanes, and artifacts, as shown in Figure 4-2 below. Under each of the four notation categories, there are different notations. The analysis conducted completely analyse the BPMN notations; for each notation, its name, shape, and a brief description are represented and how it can be mapped to represent the manufacturing system if possible.

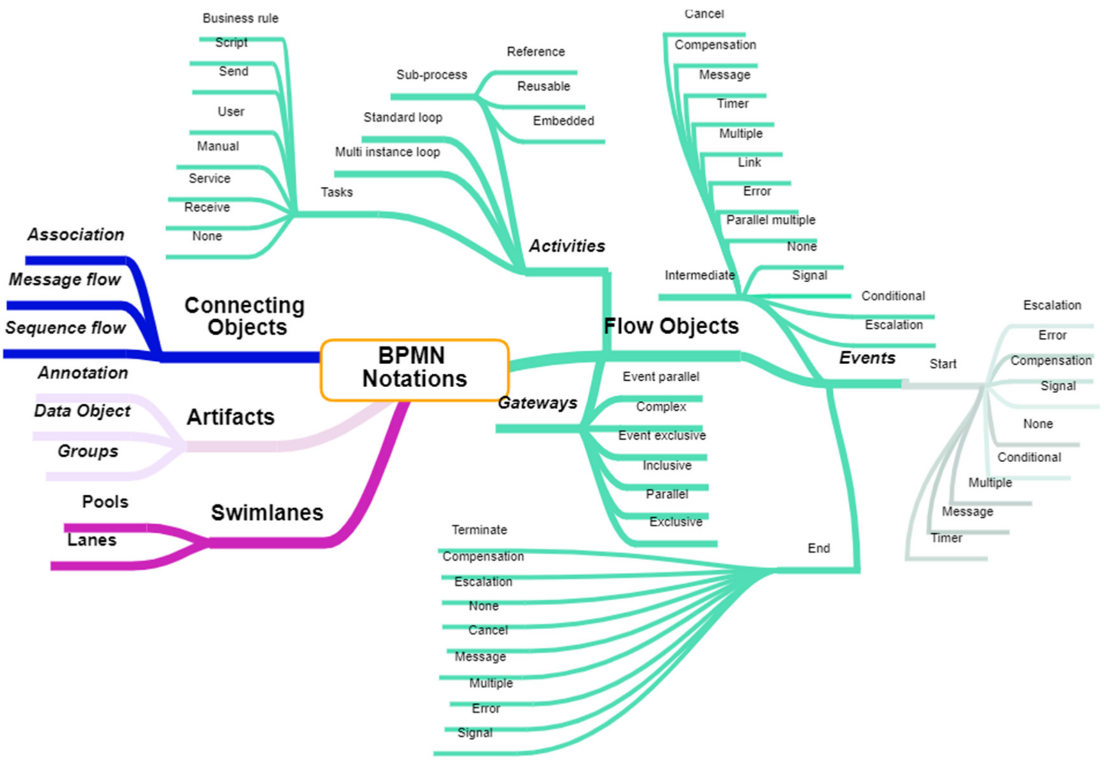

Figure 4-1: An Overview of BPMN Graphical Notations.

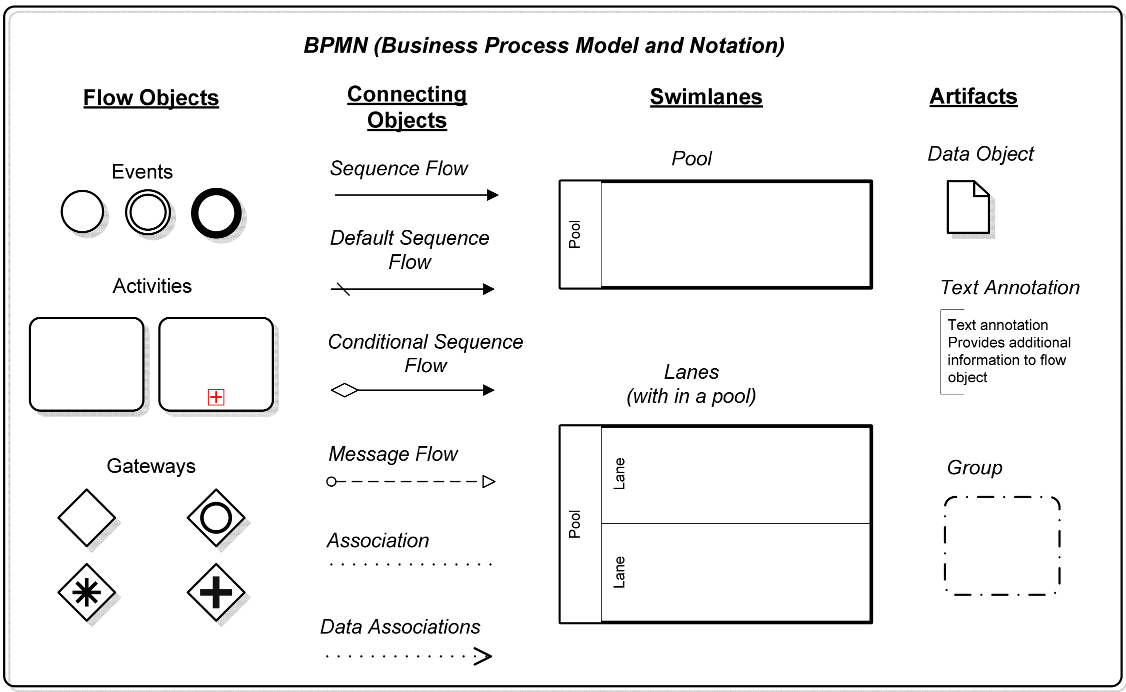

Figure 4-2: BPMN Elements.

4.1.1 Flow Objects

ACTIVITIES

An activity is represented with a rounded-corner rectangle and named according to the type of work that is required.

- *Task Activity*: a *task* in manufacturing can graphically represent any process, like cutting, packing, and other manufacturing processes.
- *Send Task*: in manufacturing, *Send Task* could be the activity of sending a re-order request when you reach a certain inventory level or sending a job order to the shop floor.
- *Receive Task*: in manufacturing, *Receive Task* can represent receiving a job order request or receiving customer demand.
- *User Task*: when relating this task to manufacturing, this activity will not only concern human performance it may include any resource that is used in manufacturing.
- *Loop Task*: in manufacturing represents activities that behave like a loop, for example, repeatedly heat the material, applying a certain coat repeatedly and in a sequence, or for any process needed to be repeated sequentially.
- *Manual Task*: in manufacturing would include manual inspection, assembly, moving parts along the shop floor.
- *Sub-process*: BPMN sub-process can be used to represent a manufacturing sub-process, which can be decomposed to represent parent activity and to include pools and lanes.

EVENTS

Start Events

Each process has to begin with an initiating event which is the start event.

- *Start Event*: in manufacturing, can be used to tell where the process starts, such as the beginning of a production line.
- *Message Start Event*: a process is started on receipt of a message, in manufacturing the message could be the arrival of raw material, reaching a reorder point, or any message arriving from a participant and triggers the start of the process.
- *Timer Start Event*: when a specific time or date triggers the start of a process and the same goes with manufacturing, for example releasing lots according to a production schedule.
- *Conditional Start Event:* a process is started based on changed manufacturing conditions or matching rules.
- *Signal Start Event:* a process starts when a signal from another process is captured.
- *Multiple Start Event:* it indicates that there are many ways to start the process and that one, out of the set of events, will be required to start the process.
- *Error Start Event*: when an error triggers the start of a process, such as when a failure of a machine triggers a repair process.

Intermediate Events

An intermediate event is any event that occurs between a start and an end event. These can be used within the sequence flow or attached to the boundary of activity to indicate an exception flow.

- *Intermediate Message Event:* indicates that a message can be sent or received, in manufacturing messages are sent to and from operators on a production line either to inform them that a batch is received, an order is completed, or send a message to advise the manager that an operator is free.

- *Intermediate Timer Event:* indicates waiting time within a process, in manufacturing it can be used to represent wait for a batch size to be completed, and a machine to be available. If the timer is attached to the boundary of activity it can be used as a timer for machine maintenance, an operator scheduled training; indicating that after a certain time an exceptional flow must be followed.

- *Intermediate Conditional Event:* indicating that a part or a lot should wait until a manufacturing condition has been fulfilled; for example, a new batch is needed an operator is free to do a specific job.

- *Intermediate Error Event:* this event needs to be attached to the boundary of activity to catch errors and handle them. In manufacturing, it could be attached to the machine task boundary to represent that the machine is subjected to emergency maintenance.

- *Intermediate Signal Event:* in manufacturing an intermediate signal event could represent catching a signal for process execution or throwing a signal to continue process execution.

<u>End Events</u>

End events are the final events, indicating that a process is completed.

- *Non-End Event:* it is the standard end of a process, where a process can only finish when all the routes of the flow arrive at an end.

- *Message End Event:* used when a message is sent at the end of a process, for example, send finished goods to the warehouse.

- *Error End Event:* an error is generated at the end of a process.

- *Signal End Event:* indicates that a signal is generated when a process end.

- *Multiple End Event:* This means that there are multiple consequences of ending the process, all of them will occur.
- *Terminate End Event:* it triggers the immediate termination of a process.
- *Cancel End Event:* it indicates that the entire process is cancelled.

GATEWAYS

Gateways are used to control how Sequence Flows interact as they converge and diverge within a process. There are different types of gateways as follows:

- *Exclusive Gateway:* it represents decisions in the manufacturing process. For example, items are either accepted or discarded.
- *Inclusive Gateway*: can be used to create an alternative but also parallel paths within a process flow, each path is considered to be independent, so, all combinations of the paths may be taken or at least one of them.
- *Parallel Gateway:* it may represent merging the output of several machines to be processed in another machine.
- *Complex Gateway:* It could represent complex flows in the manufacturing process.
- *Event Exclusive Gateway:* is similar to the exclusive gateway; both involve one path in the flow. An event-based gateway is routing the sequence flow upon evaluating which event has occurred first.
- *Event parallel Gateway*: a parallel event gateway is used to indicate that the process is depending on two events; for example, there has to be a free operator and an available machine to start a job.

4.1.2 Connecting Objects

Connecting objects are lines that connect BPMN flow objects. Various connectors can be used in BPMN; these include:

- *Sequence flow:* it can be used to show the sequence flow of processes in any manufacturing model.
- *Default Sequence Flow:* the default path is activated when a condition exists on the flow out of an activity or a gateway. For example, raw materials will be loaded to machine X by default.
- *Conditional Sequence Flow:* it is associated with a conditional rule that is evaluated at runtime to determine whether or not the sequence flow will continue.
- *Message flow:* it shows the flow of messages between two participants or two processes, but it cannot connect to objects within the same pool.
- *Association:* it is represented with a dotted line, and it is used to link artifacts with other BPMN elements. For example, it can be used to connect text annotation to tasks.
- *Data Associations:* used to move data between data objects, data store and inputs and outputs of activities. They have no direct effect on the flow of a process.

4.1.3 Swimlanes

Pools and Lanes represent responsibilities for activities in a process. A pool or a lane can be an organization, a role, or a system. Swimlanes do not provide any information about their performance, skills, workload, and working time [37]. They could illustrate active actors in any manufacturing model.

4.1.4 Artifacts

Artifacts are used to visually represent objects outside of the actual process. It can represent data or notes that describe the process or they can be used to organize tasks or processes. The different types of artifacts that can be used in BPMN are:

- *Data object:* in manufacturing tasks are often data-dependent and data objects could represent those data requirements of the process like the processing time waiting time, arrival time, or the data object may represent the result of a process in the form of data like the cycle time and the throughput.

- *Text Annotation:* in manufacturing for example it can be added to tell the reader how to perform a specific task, or when to stop a certain process.

- *Group:* in manufacturing, it could be used to group elements of the diagram to highlight processes of the same category or to group processes that need to be executed by a specific machine.

4.2 MPMN DEVELOPMENT

4.2.1 Main Manufacturing Aspects

After analysing BPMN notations, this section will address the important aspects of the manufacturing domain and linking these aspects to BPMN elements. Through this mapping, the missing aspects of standard BPMN to the manufacturing domain will be detected and thus the extension concepts will be identified. In Figure 4-3 an upper ontology for the manufacturing domain is presented to describe the domain as a whole, the present ontology is based on MAnufacturing's Semantics ONtology (MASON) ontology proposed by Lemaingnan et al [110].

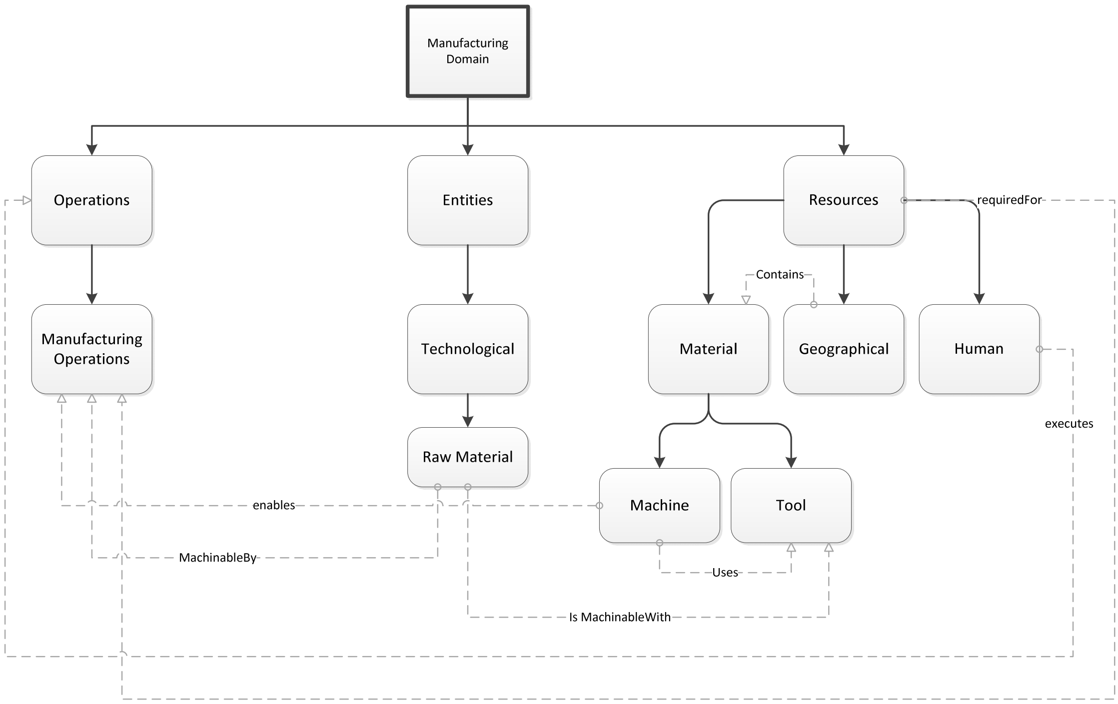

Figure 4-3: An ontology for the manufacturing domain [110]**.**

There are two main subcategories to Manufacturing Operations that emerged through the literature as shown in Figure 4-4: manufacturing processes and material handling. The manufacturing processes involve the conversion of raw materials into finished products with specific characteristics. The materials handling includes all industrial activities that do not add to the value of the product but are essential for the proper running of the factory [111].

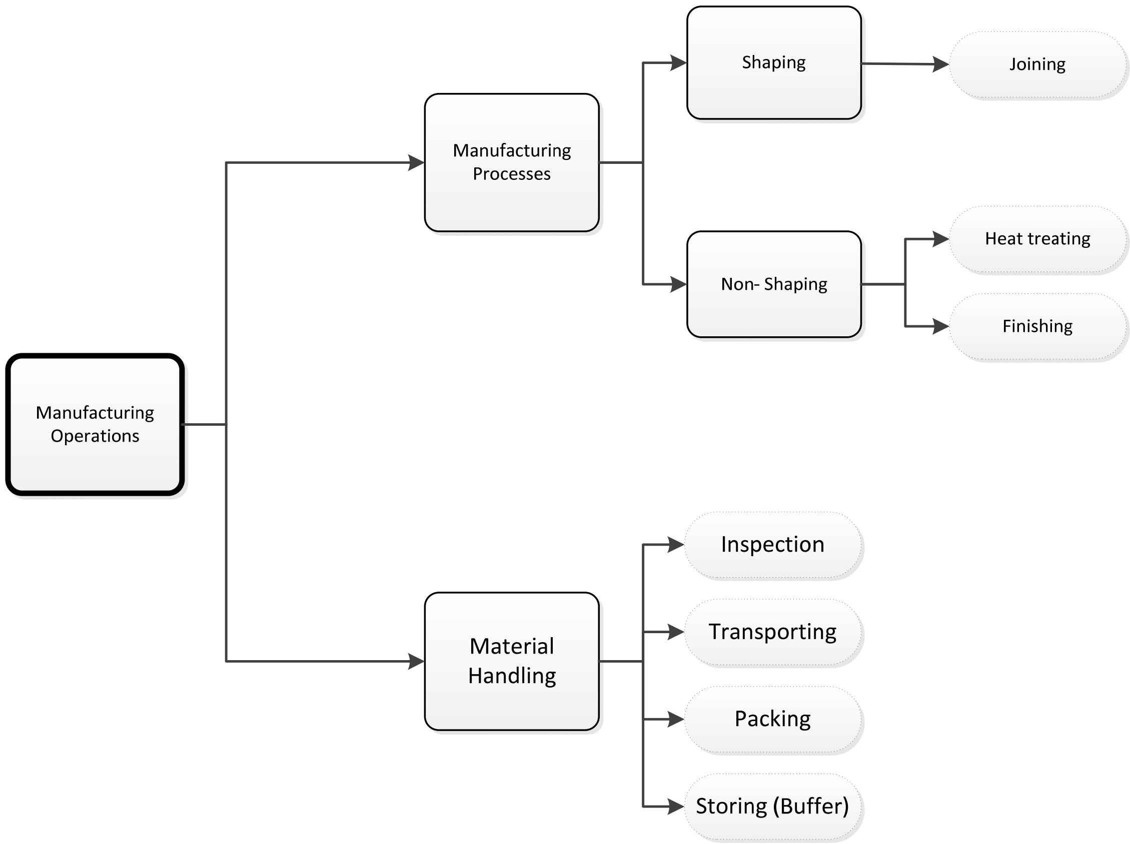

Figure 4-4: Classification of operations in the manufacturing domain.

4.2.2 Proposing MPMN

Identifying the key manufacturing aspects in the previous section and analysing BPMN notations in section 4.1, showed that BPMN notations lack some important aspects needed to model manufacturing systems, such as, batching, queuing, resources, tasks subjected to failure, etc. However, as mentioned in the literature, it is necessary to implement domain-specific concepts to enrich the modelling language by typical domain concepts, which foster better communication with domain experts and help to describe the domain adequately [17]. Thus, the need to extend BPMN notations to build models that are specific to the manufacturing domain evolved. This extension offers modifications to the existing BPMN notations and the development of new notations that encompass important missing aspects to represent a manufacturing system.

There are suggested rules for BPMN extension [9]. First proposing an extension to BPMN must not contradict the semantics of any element that is defined in the BPMN specification; the shapes defined in the specification must not be changed, and the shapes of extension elements must not conflict with the shapes defined in the specification. Second, the graphical elements should be easy to understand by any viewer of the process diagram. Also, the extension elements should have the "look-and-feel" of BPMN. The main aim of these rules is that the requirements of different domains can be satisfied while maintaining a "valid BPMN core", that can be easily understood by business experts. Adding to this, Zor et al [10] mentioned that BPMN extensions permit additional attributes to be added to the elements defined in the specification. Additional markers and indicators to be added to graphical elements that are already defined in the specification and additional graphical elements representing any kind of artifact may be added.

When using BPMN in modelling the different problems found in the case studies, presented in the next chapters, some of the current BPMN notations were used in a different manner to model specific manufacturing-related processes/operations. Although BPMN notations were adopted to fit the needs of modelling the manufacturing process, notations that truly represent manufacturing aspects are still needed.

MPMN (Manufacturing Process Model and Notation) is an extension to BPMN. MPMN involves modifications to the existing BPMN notations and the development of new notations that are able to model manufacturing processes using core modelling notations that is more suitable to represent a manufacturing system.

It should be emphasized that one of the drivers for extending BPMN and the development of MPMN is to create a simple and understandable tool for modelling manufacturing processes, while at the same time being able to handle the complexity inherent to manufacturing processes. The approach taken to handle these two conflicting requirements was to organize a new and extended set of the graphical aspects of the notation into the same specific categories of BPMN.

As mentioned earlier, Zor et al [10] extended BPMN by developing new notations that were customized to include some missing aspects required to model manufacturing systems, yet, their extension efforts still lack some important aspects to model the manufacturing system. Accordingly, this work further extends the BPMN notations and builds on Zor's work aiming to provide a set of notations that can represent and effectively model manufacturing systems of different levels of complexity. Some of the proposed notations were typically utilized on the (as-is) basis, others were adopted to act in a different manner and introduced to new representations, in addition to the new notations developed in this work to embrace other important manufacturing aspects.

Specifically, Zor et al proposed four different gateways, in order to control the interaction of converging and diverging of material flows [10]. Material select and material route gateways are proposed by (Zor et al) and utilized here with the same icon shape and function. The batch and un-batch gateways which are proposed also by Zor et al were adopted in the MPMN set of notations with the same icon shape but with different names.

As shown in Figure 4-5, MPMN notations are summarized in a way that is similar to the BPMN notations, so that the readers who are familiar with BPMN can easily recognize the basic types of elements and understand the diagram. All modified and newly developed notations fall under the four BPMN categories (flow objects, connecting objects, swimlanes, and artifacts) and the main goal when developing this notation was to keep the basic look and feel of the BPMN standard. Furthermore, the flow objects, connecting objects, and artifacts shown in Figure 4-5 are the new proposed notations to accommodate manufacturing processes modelling.

It should be noted that the notations shown in Figure 4-5 and described in the next section are MPMN specific and the full MPMN includes the notations mentioned in Section 4.1 that already existed in the BPMN set of notations but were used here in a different manner.

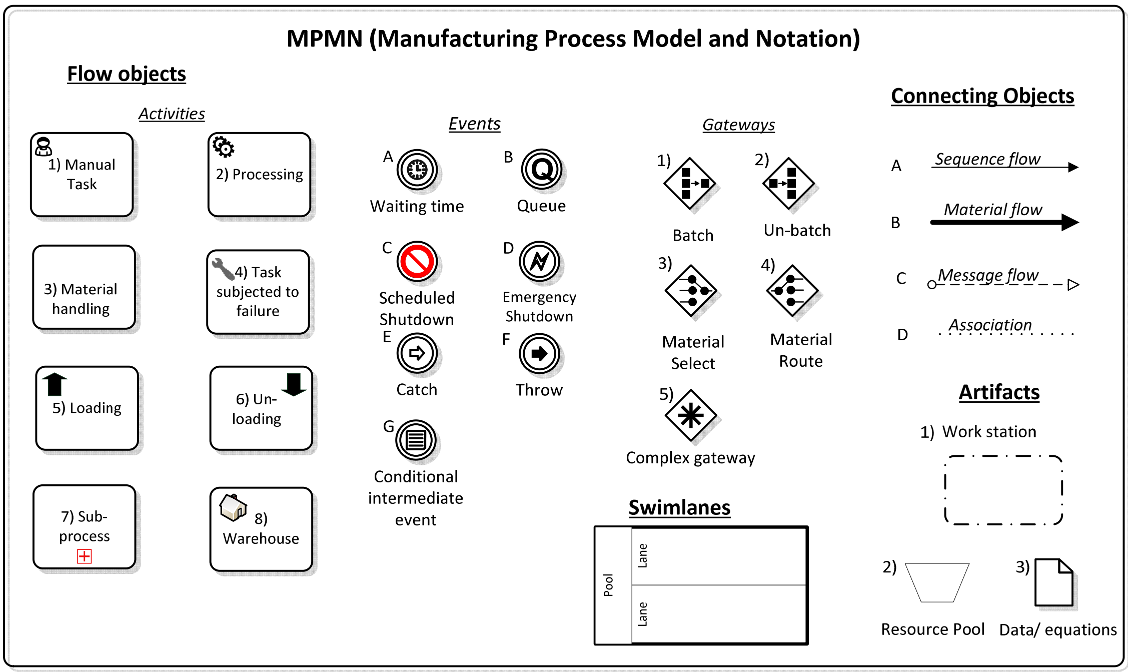

Figure 4-5: MPMN Notations.

4.2.3 Flow Objects

ACTIVITIES

- *Manual Task:* modifications are done to the user task, the same notation is used to represent the manual task, an icon of a man on the top left corner indicating it is a manual task or any task is performed manually i.e. involve the use of a personal

- *Processing*: it includes any process that transforms raw material to finished goods and it may involve processes like turning, cutting, or any process performed by a machine. It is added to specify the processing time and the number of delays.

- *Material handling*: A newly developed notation: it represents any equipment used for moving goods from place to place throughout the process. It has been added to specify the capacity of the equipment speed and to define how far the items moved.

- *Task subjected to failure*: a newly developed task, it represents any task that subjected to schedule or emergency maintenance. Border events could be attached to such tasks to specify the meantime to failure and the time to repair.

- *Loading:* a newly proposed activity, it represents the action of loading and the time that materials consume to be loaded for processing should be specified.

- *Un-loading*: a newly proposed activity, it represents the action of un-loading and the time that materials consume to be un-loaded after processing should be specified.

- *Sub-Process*: The Sub-process is a decomposable activity, i.e. a process that is included within another process; an expanded call activity represents the Sub-Process in MPMN as it has the same shape of a subprocess but with a double thicker line. In BPMN a sub-process cannot have pools or lanes and are not a true hierarchal representation of the parent activity; the BPMN sub-process is different than the manufacturing sub-process.

- *Warehouse:* a newly developed notation, it is the task with a house in the upper left corner, indicating that goods are stored.

EVENTS

- *Waiting time*: timer event is modified and used to represent the waiting time

- *Queue*: a new proposed event; it represents the sequence of items awaiting to proceed. It is used to release items in a specified order or can be used as a resource pool queue, which releases items when their required resources are allocated.

- *Scheduled shutdown*: a new proposed event, added as a time to a failure event, to specify the time to repair. It can be used in manufacturing as an interrupting border event to tasks subjected to failure.

- *Emergency shutdown*: modification is done to error event to be used to represent the unusual stoppage of the production line.

- *Catch*: Link events in BPMN are special cases; they have no special execution semantics but serve as "GoTo" to another point in the same process model. Link events are used in MPMN to represent catch; it is added to the MPMN set of notations and is used to catch items.

- *Throw:* throw event is used to throw items, two matching links will be used to sequence items processing; throwing event will act as the exit point, and catching event is the receiving point.

- *Conditional intermediate event*: is like a wait until a condition is set to items passing through the block. For example, using the MPMN simulator conditions like attributes, the priority value on an item, and the quantity of the item can be set.

GATEWAYS

- *Batch:* batching items into a single item, you can specify the quantity needed for each batch.

- *Un-batch*: creates parallel flows by dividing material into multiple constituent parts. The condition specifies how the material has to be split up.

- *Material Select Gateway*: it has multiple incoming flows and one outgoing flow. For example, one out of multiple materials will be selected to be routed through the production process.

- *Material Route Gateway*: A diverging material route gateway is used when alternative paths within a process could be taken. This new gateway can model a decision that can be thought of as a question that is asked at a particular point in the production process.

- *Complex Gateway*: Complex gateway is one of the BPMN gateways and is used for the most complex flows in the business process, but it represents complex flows graphically only and is not supported by common execution engines. Complex gateway is added to MPMN set of notations, as it models complex decisions but in a different manner than that of the BPMN. It will be used graphically and in simulation models to display attributes on items, and then passes the items through. The attribute value is shown in the dialog and output at the value output connector.

4.2.4 Connecting Objects

- *Sequence flow connector* will be used to connect elements in a model to show the sequence of processes.

- *Material Flow Connector*: it is represented by a thick black line with an arrowhead. It is proposed to indicate the sequence of "material flow" in a model, and it is used to connect manufacturing tasks, events, or gateways with each other.

- Two connectors are utilized here with no modifications and illustrated in the new proposed set of notations to show how important they are to be among the specified connectors. The first one is *the Message flow connector* that represents the flow of data between two pools or different processes, and the second one is an *association*: it is the dotted line connector, used to link existing and newly developed artifacts with other elements.

4.2.5 Swimlanes

The pool represents the process participants, and Lanes are pool subdivisions; they remain the same with no modification in the new approach. They will illustrate active actors in any manufacturing model.

4.2.6 Artifacts

- *WorkStation*: the basic notation "Group" is changed to represent workstation; in manufacturing workstations group processes that belong together. This kind of notation will be only used graphically and won't have any effect on the execution of the model.

- *Resource Pool*: stores the resource units to be used in the model and should be used with the resource pool queue element. These units limit the capacity of a section of a model.
- *Data object*: as mentioned earlier data object does not affect the flow of the process. However, in MPMN new set of notations data object notation has a direct effect on the process; it is used to add equations, rules, and calculations to the model.

To sum up the previous sections; the BPMN set of easy and understandable notations is used to model case studies to create a common language that represents the exact picture of manufacturing processes require; then, enhancing these notations to fit some manufacturing aspects. Afterward, MPMN is developed as an extension to BPMN in terms of adding some elements to describe missing essential manufacturing aspects, such as queuing, batching and un-batching, resources, and others. However, MPMN is still a static technique that cannot quantitatively describe, analyse, and/or improve the behaviour and performance of a system under study. For example, queue length or waiting time cannot be determined; resource pool is unable to provide the number of free operators or their utilisation, and so on. Thus, it is concluded that both modelling and simulation should be combined to be able to visualize, measure and improve the performance of a system under study.

4.3 SIMULATION OF PROCESS MODELS AND NOTATION

Advanced modelling simulation knowledge and skills are always required [112]; hence, a modelling language needs to be designed to facilitate the simulation models development, which can be used to describe the relationships among the components of a manufacturing system (resources, entities, etc.). Also, when the communication involves different stakeholders, a standard representation that can be understood by all stakeholders is essential. The fact that communication between stakeholders is important for the success of a simulation model makes the need for a good conceptual model representation become even more essential [113]. BPMN is a process modelling standard that is readily understandable by non-specialists and can help the different groups involved to work together. The link from non-software-specific conceptual maps to DES is a very promising direction for engaging stakeholders and closing the model coding gap as simulation modelling should engage stakeholders throughout its lifecycle [85].

Hence, manufacturing process modelling has to evolve, and static models have to be translated into a suitable discrete event simulation environment to be able to analyse, experiment, and improve the system under study. Thus, an integration that depicts manufacturing complexity is proposed; this is to combine both models that are understood by all stakeholders while being an expressive simulation tool that is enough to handle the various levels of complexities in the manufacturing domain. This will strengthen the analysis and the evaluation of the current and future status of a system under study.

4.3.1 Existing BPMN Modelling and Simulation Tools

BPSim 2.0 is the integration of BPMN and Simulation. BPSim 2.0 standard complements BPMN; allowing process simulation. This standard aims to make it possible to exchange simulation models between different modelling and simulation tools [82]. Several tools exist that allow verifying quality aspects of BP models. The most widely used among those tools are Signavio, No Magic MagicDraw, Bizagi modeler, and Camunda Community. All these tools provide BP model editing capabilities besides BP model verification capabilities [114].

As mentioned in Chapter 3, the Bizagi modeler is selected as it is one of the most representative commercial tools at the time of developing this work. Bizagi was selected because Bizagi is claimed to provide powerful simulation capabilities that enable organizations to make better decisions by visualizing the impact of proposed ideas and changes prior to implementation in a real-world setting[1]. Bizagi uses BPMN to build process maps and BPSim 2.0 to run a discrete-event simulation (DES) in the same software environment [85]. Bizagi supports both BPMN 2.0 and BPSim 2.0. Bizagi is a

[1] www.bizagi.com

freeware tool, where modellers can use Bizagi for building and simulating BPMN diagrams without licence fees requirements, and because it is popular among editors for its ease of use and extensive validation [2, 72]. BPSim 2.0 will be represented throughout this work using the Bizagi modeler[2].

4.3.2 Discrete-Event Simulation (ExtendSim)

ExtendSim (or Extend as it was first known) is the second simulation tool used in this work and it represents a full Discrete-Event Simulation Environment. It was not only the first “drag and drop” simulation program, but it was also the first graphical simulation tool to embody the concept of modelling components as objects [115]. Its user interface, architecture, and extensibility set a standard that has been followed by numerous other simulation programs. The combination of an inexpensive, easy-to-use and powerful simulation program has led to the adoption of ExtendSim in several domains [116].

Hence, ExtendSim is one of the powerful full discrete-event simulation environments, and it allows the modeller to create customized Blocks from scratch (user-programmed Blocks) using development tools that ExtendSim provides [117], which can be achieved without sacrificing the capability to model large and complex systems [116].

It is worth mentioning that ExtendSim has all the building blocks that support the modelling of a complex manufacturing system, but as mentioned previously, developing simulation models using ExtendSim only requires experienced modellers and will not allow all participants to take part in making decisions and improving the system. As mentioned before, this work aims to model and simulate complex manufacturing processes using simple and understandable notations. Thus, the integration of MPMN and ExtendSim is proposed to model manufacturing systems, as using ExtendSim solely will sacrifice the understandability of the developed models and will consequently lose the concept of involving all the stakeholders in making decisions and improving the system.

---

[2] Referred to as Bizagi throughout the remaining text

ExtendSim will serve two purposes in this work: first, it will be used to develop the MPMN Simulator library; also, it will be used to develop reference models for the case studies presented in the next chapters, which will be used to validate the results of Bizagi and the developed MPMN Simulator tools.

4.4 MPMN SIMULATOR

MPMN is intended to provide a means to rapidly build, modify, and validate processes in a user-friendly and straightforward environment, and the simulator will give the user the advantages of simulation where they can analyse experiment, and improve the system without interacting with simulation or doing the programming behind.

4.4.1 Component-based Modelling

The development of the MPMN Simulator is based on the component-based modelling approach, which relies on the concept of having pre-built models or model components that can be plugged together to form a model of the desired system. The idea is that we simply select these components from a library and use them directly. Where each component or in this case it is called notation is built once, then this notation is verified, and if it works as intended it will be added to the library to be used in many different applications [118].

Component libraries are being developed to allow rapid prototyping of new designs, which provides a means of creating a rich depiction of systems from reusable models rather than starting from scratch each time Component-based modelling allows users to quickly and efficiently create high fidelity simulation models by linking independent model objects. The result of linking these models is a model network that can be used to evaluate the aggregate performance of the system as well as investigate the interactions and performance of the individual component models. Model users should be able to assemble the model component parts in a plug-in manner, thus minimizing the time, cost, and expertise required to construct comprehensive models within the context of their organization [119].

4.4.2 Developing the MPMN Simulator

To start developing a model using MPMN Simulator in ExtendSim a library is created that includes all the developed MPMN notations that are needed to build a manufacturing model. First, an ExtendSim block is created to establish the discrete event simulation capability and made hierarchical. The graphical appearance of each hierarchical block is then developed to match the MPMN notation. All developed notations are then saved in a library to allow easy reuse. This library has a drag and drops features, where created notations are dragged and dropped into a new model to create an MPMN representation as shown in Figure 4-6. These notations look exactly like MPMN and have the simulation capabilities to match.

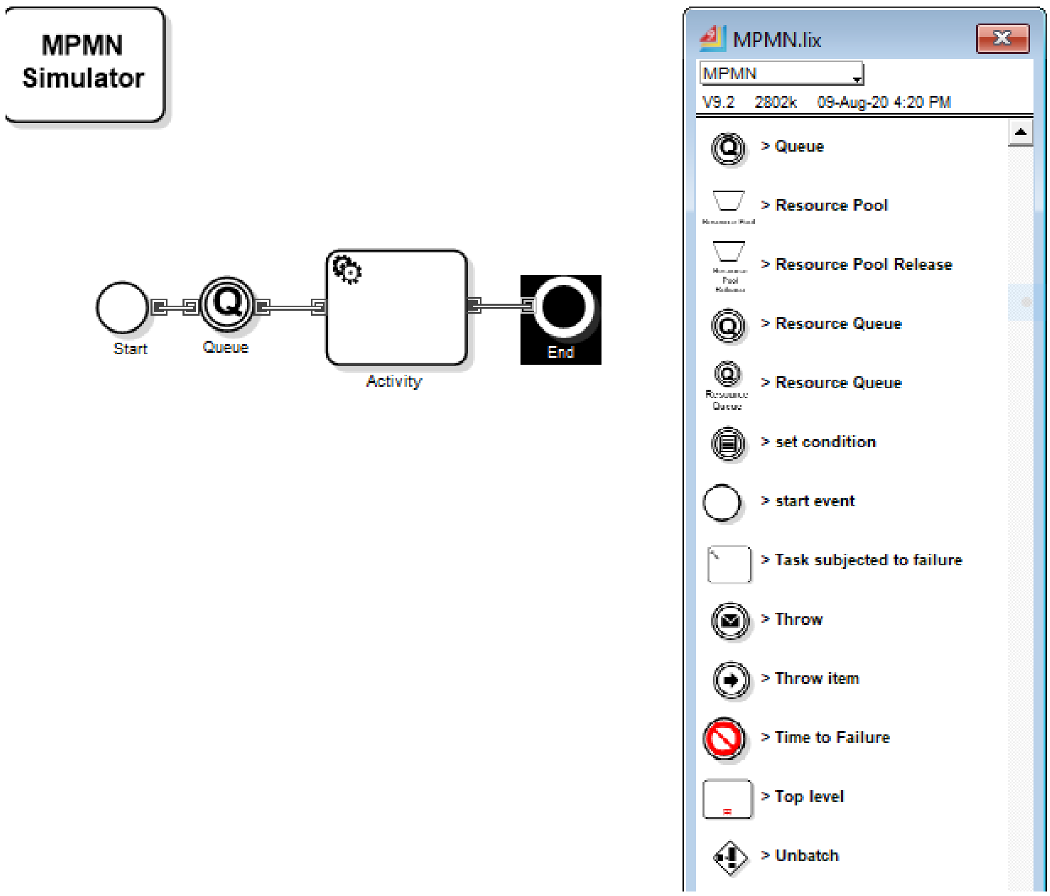

Figure 4-6: Representing a flow using MPMN Simulator library.

First, the MPMN Simulator task must be placed to the left of all other notations in all models. Then, start modelling using the new MPMN notations, drag and drop the notations needed to build the model, and for every single notation's input values and defining distributions (if any) can be easily put in, as shown in Figure 4-7. Although some of the original BPMN notations are used here in a different manner to better represent the system.

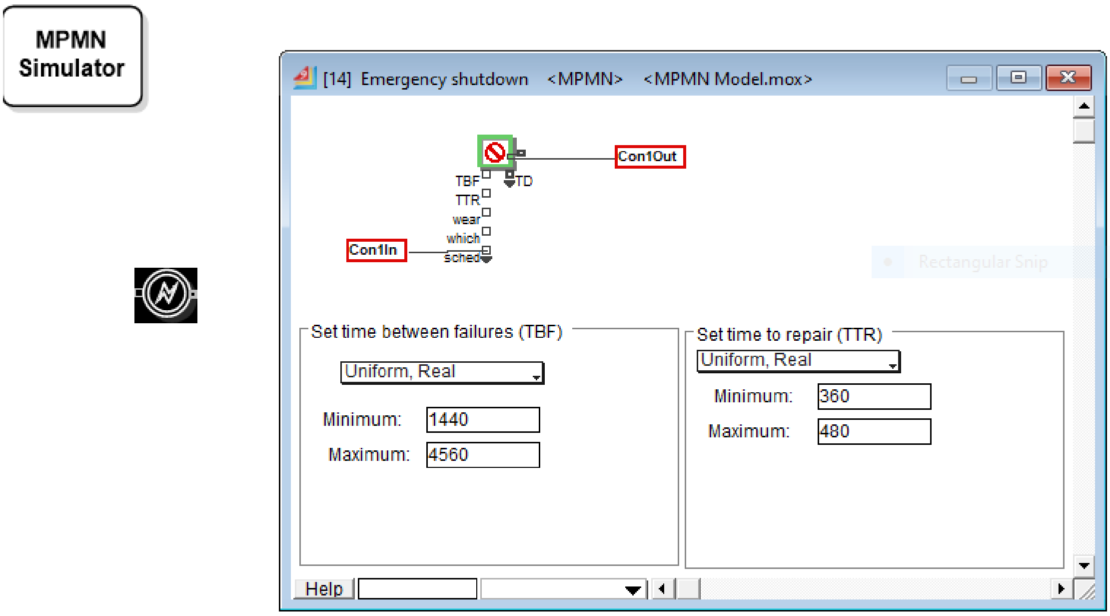

Figure 4-7: Emergency shutdown modeled in MPMN Simulator.

It is worth mentioning that some authors [36][84] use an engineering design approach introduced by OMG called Model Driven Architecture (MDA) to help the design of their BPMN extension that is suitable for simulation. MDA provides a set of guidelines for applying model-driven engineering principles. For example, Imen et al [120] used a translation approach to transform from a BPMN model to an XML model.

Here, the MDA rules and translation approaches are not needed as the library is already in the ExtendSim environment and the conceptual models built in ExtendSim are automatically complete simulation models without the need for any transformation.

5. SIMULATION OF A SIMPLE MANUFACTURING SYSTEM

This chapter presents the first case study used in this work to investigate the suitability of using BMPN 2.0 notations to model a simple flow shop manufacturing system. The case study is of a real-life automotive parts manufacturer in Egypt. The chapter starts with representing the system using a BPMN model to assess BPMN capability to model such a system. The models developed are critically analysed at the notation-level to determine if, when applied to manufacturing, a BPMN notation can be used without any modification or not. The models are then re-developed using the proposed MPMN to illustrate the need for a manufacturing-focused notation that should be added as an extension to BPMN.

Next, a set of performance measures and simulation parameters are defined. A simulation model using ExtendSim, as a full DES software, is developed and is used to accurately evaluate the performance of the system under study. Afterward, both models, BPMN and MPMN, are simulated using Bizagi and the ExtendSim MPMN Simulator to assess their capabilities to simulate the developed models. Furthermore, the results reported from the simulation models are compared to the ExtendSim simulation model results to determine the accuracy of the developed models in quantifying the performance of the system.

5.1 CASE STUDY DESCRIPTION

5.1.1 Background

The case study presented in this chapter is of a real-life automotive parts manufacturer and supplier of parts used by automotive manufacturers in Egypt. The manufacturer first founded the company in 1987 and had only one manufacturing facility. Afterward, two additional manufacturing facilities were established in Egypt. This case study focuses on one facility only and also focuses on one of the parts that are manufactured in that facility. The part selected is the U-bolt; in 2014, the company considered U-bolts as their primary product and at that time they became the sole suppliers of U-bolts for the automotive industry in Egypt.

This case study is selected to assess the capabilities of BPMN to represent some of the challenges which may be present in manufacturing systems but are not normally seen in business operations. Specifically, the case study is a flow shop that is labour intensive, and most processes require both labour and machines as resources.

This manufacturing system has a simple flow of consecutive production stages till the final product is completed. The manufacturing processes of the U-bolt are mostly conventional with little reliance on advanced manufacturing technologies. The models developed will present the flow of material from the beginning of the line, through the sequence of processes, until the U-bolt is completed and sent to shipping. The models will introduce new processes to the BPMN like cutting, batching, setups, rework, etc.

5.1.2 Data Collection

The first step in collecting data is to determine the data required for building the U-bolt case study model. Data collection is categorised into three main types: structural, operational, and numerical.

Structural Data

Structural data involve all of the objects in the system to be modelled, which is a facility that produces U-bolts. This includes such elements as entities (U-bolts), resources (workers and machines), and locations (inspection areas). The U-bolt fabrication process structural data are listed in Table 5-1.

Operational Data

Operational data explain how the U-bolt facility operates that is, when, where, and how events and activities take place. Steel rods, which is the raw material needed for the U-bolts, are shipped from Korea to the factory premises. Upon the arrival of the rods, a preliminary inspection takes place and, once accepted, is stored in the plant's warehouse to be used in the U-bolt fabrication process. The routing of the activities needed to produce a U-bolt starts by releasing a batch of rods from the warehouse and is broken down into the steps shown in Figure 5-1.

Table 5-1: U-bolt fabrication structural data.

Process	Number of Machines	Number of Workers
Cut	1	1
Adjust	3	3
Inspect A	1	1
Thread Rolling	2	2
Heat and Bend	5	5
Inspect B	2	2
Heat Treatment	5	5
Inspect C	2	2
Galvanize	4	4
Final Inspection	2	2

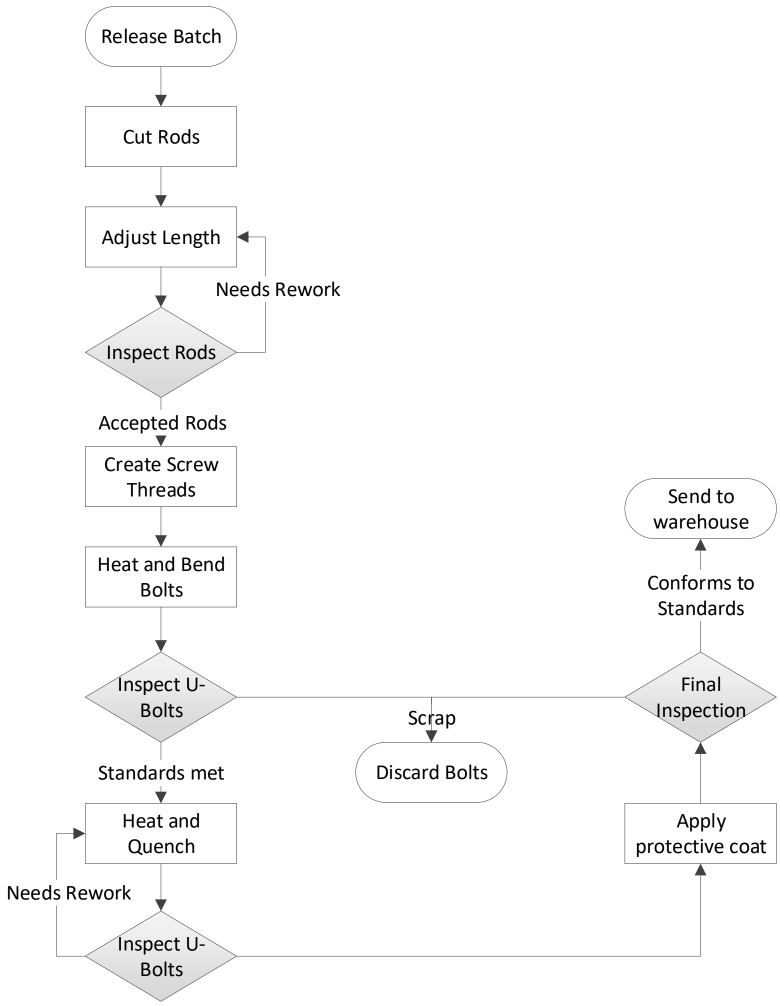

Figure 5-1: U-bolt Fabrication flowchart.

A brief description of each process is presented as follows:

- *Release Batch*: rods are released from the warehouse in batches. Usually, a batch of 50 rods is released to the shop floor every 3 hours on average.
- *Cutting Rods:* the rods which are 4 meters long, are cut to match the U-bolts length size which is 50-70 cm long.
- *Length Adjustment*: after the cutting process, the rod lengths are adjusted to the desired length needed for the U-bolts.
- *First Inspection*: U-bolts are inspected in this stage to make sure that they meet the specified length and tips standard.
- *Thread Rolling*: screw threads are created on both ends of a U-bolt.
- *Heating and Bending*: heating machines are used to heat the bolts to high temperatures, and immediately after heating the bolts, rods are easily bended by an automatic hot bending machine where it bends the straight rod into the required U shape of the bolt.
- *Second Inspection*: inspection takes place at this stage to make sure geometrical and dimensional requirements of the U-bolts are accurately met.
- *Heat Treatment (Heating and Quenching)*: bolts are introduced to a heat treatment furnace and heated for 3 hours. Afterward, the bolts are quenched for 2 hours.
- *Third Inspection (Shaping/Rework)*: inspect the shape of the U-bolt that might be changed due to the heat treatment process. It has been found that about 5-10% of each batch needs rework.
- *Galvanizing*: is a protective method, where three coats of Zinc are applied to the U-bolt in order to prevent rusting and corrosion.
- *Final Inspection*: the final product is inspected to make sure that it conforms to the U-bolt standards.

- *Warehousing*: inspected U-bolts are sent back to the warehouse in batches to be stored.

Numerical Data

The numerical data is the data that provide quantitative information about the U-bolt facility. Examples of numerical data include the values used for arrival rates, activity times, setup times... etc. These values can either be deterministic (constant) or stochastic (probabilistic). For stochastic data, the probability distributions and their parameters should be determined as well. Numerical data needed for modelling and simulation of the U-bolt manufacturing process are presented as follows:

Deterministic Data

The facility has 7 effective production hours per shift and works 6 days per week. The facility outputs on average 700 U-bolts per shift. The processing time of the processes with their loading and unloading times are as shown in Table 5-2.

Table 5-2: Summary of processes timings.

Process	Processing time (min/Part)	Loading (min/Part)	Un-loading (min/Part)	Setup (min/batch)
Cutting	0.25			
Adjust	0.80	0.2	0.2	5
Inspect A	0.40			7
Thread Rolling	0.90			
Heating	1.20	0.2	0.2	5
Bending	0.45			
Inspect B	0.40			
Heat Treatment	1.40	0.2	0.2	30
Inspect C	0.40			
Galvanizing	1.30	0.2	0.2	15
Final inspection	0.80			

Stochastic Data

Although the processing times are assumed constant, the developed model of the U-bolt case study is stochastic due to the randomness of arrival rate and outcome of inspection processes.

The time between arrivals (TBA) is assumed constant, and a batch is introduced to the facility every three hours; yet, the number of units per arrival is stochastic. It is given that a batch consists of 50 rods and each rod is 400 cm long. Nevertheless, the length of a U-bolt ranges between 50 to 70 cm and follows a uniform distribution. Thus, the number of U-bolts released to the shop floor every 3 hours varies from approximately 286 to 400 units and follows a discrete uniform distribution.

The second source of randomness in the model is the outcome of the inspection processes. It is given that on average 5% of the bolts inspected need rework. Rework takes place in this case study after Inspection A and Inspection C. Bolts needing rework after Inspection A is sent back to the adjusting process, and U-bolts that need rework after Inspection C is sent back to the heat-treatment process to get back in shape. Additionally, it is given that an average of 5% of the U-bolts is discarded. If the U-bolt doesn't meet the geometrical and dimensional requirements after Inspection B and if the final product doesn't comply with the standard after the final inspection takes place, these U-bolts are considered as scrap and are discarded.

5.2 CONCEPTUALIZATION

The next section will include the conceptual model of the U-bolt fabrication case study that is developed first using BPMN then with MPMN, these two graphical models are the foundation of the model analysis and will describe the inputs, outputs, assumptions, and simplifications of the model.

5.2.1 Modelling with BPMN

U-bolt fabrication case study is first modelled using BPMN. Modelling with BPMN will show BPMN capabilities and inadequacy to model some of the challenges associated with modelling the U-bolt manufacturing process. Separate pools and lanes are used to represent the different actors that are active in the U-bolt case study, and to define which one is in control of which process. As shown in Figure 5-2, the model represents both material flow and message flow, the message flow is shown as the dashed lines between the two actors, and the material flow is the solid line within the swimlanes. The model is illustrated using a collaboration diagram to show the interaction between the customer and the U-bolt fabrication process. Two pools are used, one for the customer and the other for the U-bolt fabrication, which is subdivided into four lanes, showing the different stages responsible for the U-bolt manufacturing.

The process starts with a conditional start event, which means that the process will only be triggered by a customer placing an order. The first process used here is a manual task which means that the activity must be executed by a human. The order is sent from the customer to the warehouse through the message flow. The warehouse starts its process by receiving the raw materials, and then the raw materials are inspected manually. Gateways are used to control how the process flows; the accepted rods will be stored, and others will be discarded.

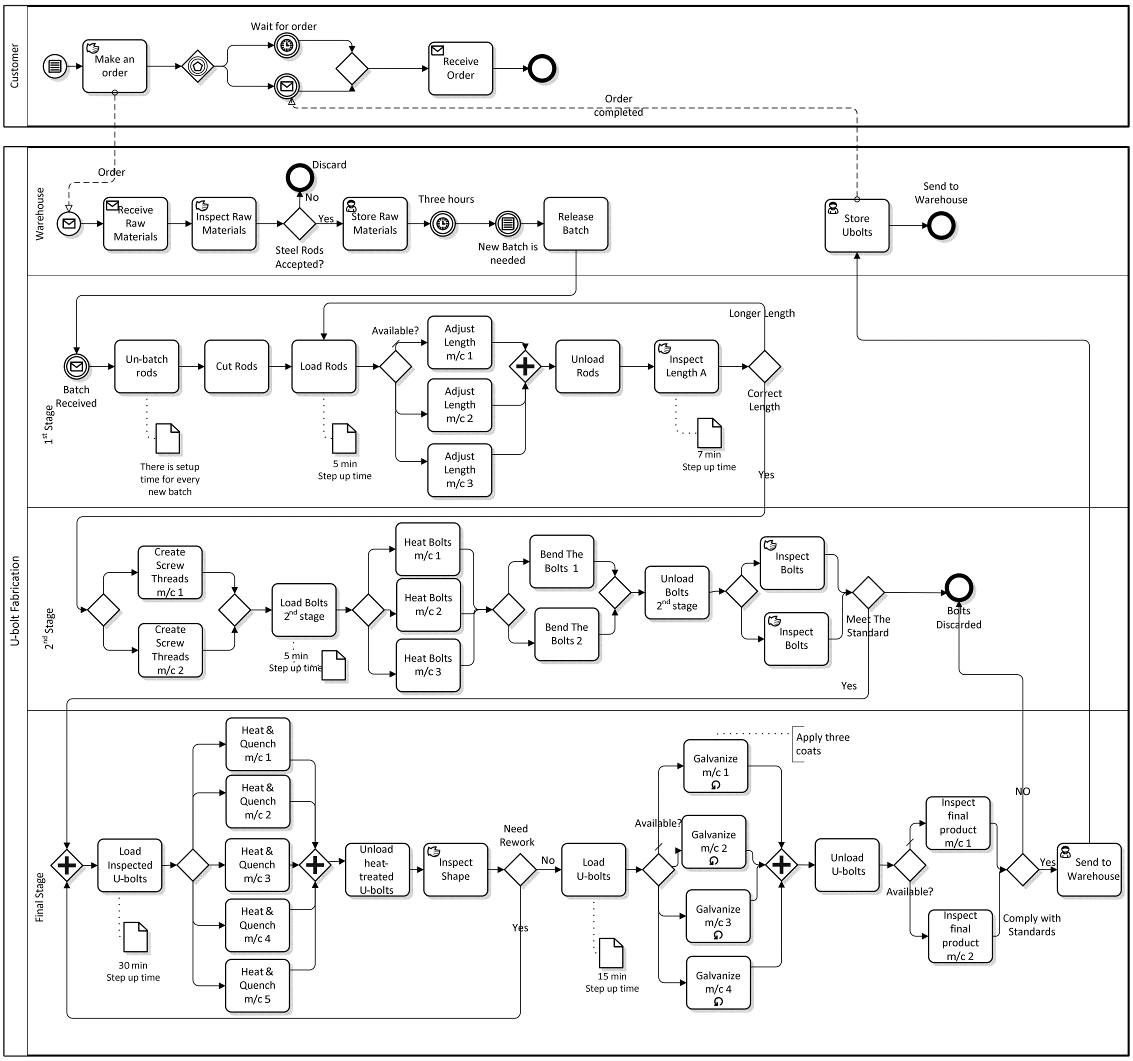

Figure 5-2: BPMN model for the U-bolt fabrication process.

Every three hours a new batch is released from the warehouse to the shop floor to start the fabrication process. The U-bolt fabrication process is divided into three stages, each stage is represented in a separate lane, and each lane has several tasks of processing in a sequential flow. The first stage is for rod cutting and length adjustment, the second stage is for rod heating and bending, and the final stage is for U-bolt galvanization and final inspection. The un-batching of the 50 rods before starting the process was the first challenge in this model, as there is no notation in the BPMN that specifically represents un-batching. Hence, the un-batch is represented as a normal activity that has to precede rods cutting.

Some of the U-bolts need rework after Inspection A and Inspection C; they are sent back to the previous stage for correction. Others are discarded after passing through Inspection B and the final inspection. As mentioned before, the percentage of U-bolts that need rework is 5% and scrap is also equal to 5% on average; yet, at this stage, the BPMN is merely a static representation of the case study; decisions or probabilities cannot be defined.

Each station has more than one machine, which is modelled using several tasks to indicate that more than one machine is working in parallel. The setup times for adjusting, Inspection A, heating and bending, heat treatment, and galvanization take place when a new batch is introduced to the shop floor. The BPMN lacks the notation to define setup time. To overcome this issue, setup is modelled as input data to the un-batch task to tell the reader that with every new batch a setup time has to be considered. It should be noted that setup could have been modelled using a complex gateway and a task representing setup whenever a new batch is received, however, the model might look more complicated adding more notations.

Challenges and Opportunities

In the U-bolt case study, there were so many challenges in representing customer orders, the equipment, the material flow, the jobs assignment, and the flow of messages between the different departments. Model validation using BPMN standard is not supported, it is totally up to the purpose of the model and therefore a decision the modeller has to make, whether a collaboration diagram with different pools is useful, or sticking to one pool with different lanes to represent the interaction between the departments.

The use of pools and lanes helped in better describing the process, and the model appeared to be understandable as pools and lanes helped to organize the whole process. The pools show the active participants in the model and subdividing pools into lanes shows how responsibilities are divided among different departments. Also, BPMN has a definite starting event showing where the process starts and has an ending event indicating where the process finished. Therefore, it is easy to track the start and the end of a sequence of processes.

On the other hand, BPMN demonstrated limitations in terms of representing some aspects of the U-bolt manufacturing process. As mentioned above the challenges to model rods un-batching, and factoring step time was facilitated by BPMN notations that are used in a different manner and helped to represent these challenges. Furthermore, as stated by Zor et al [10], BPMN lacks the type of connector which indicates a transfer of product material from source to destination. This was one of the difficulties faced when modelling the U-bolt manufacturing process. BPMN might have the capabilities to represent all the difficulties faced, but it loses the concept of model simplicity and understandability as more and more notations have to be used to represent one operation [114]. Ultimately, a simple case study would be demonstrated by a complex BPMN model, with a large number of notations.

At this stage of model development, analysis of the model is not available as BPMN only offers a static representation of the whole system. It lacks the capability of data input; hence, useful performance measures such as throughput rate, cycle time, and others cannot be estimated. For that, it is not clear how powerful BPMN is to analyse and improve a system under study.

5.2.2 Modelling with MPMN

The newly developed MPMN is used to model the manufacturing process of the U-bolt to evaluate the capability of the extended BPMN notations to model the exact processes of the manufacturing system under study. MPMN diagrams should still have the basic look-and-feel of BPMN so that a diagram by any modeller should be easily understood by any viewer of the diagram.

Modelling with MPMN showed some changes with respect to the BPMN model as shown in Figure 5-3. The use of the three types of connectors all together in one model, make it clear to who is reading the model to differentiate between the three flows. The material flow for example is the bold black line that is connecting the tasks representing the manufacturing processes of the U-bolt. Sequence flow, which is the solid line, indicates the sequence of processes for a customer to make an order. While, message flow, represented by the dashed lines, involves sending and receiving orders messages between the two pools.

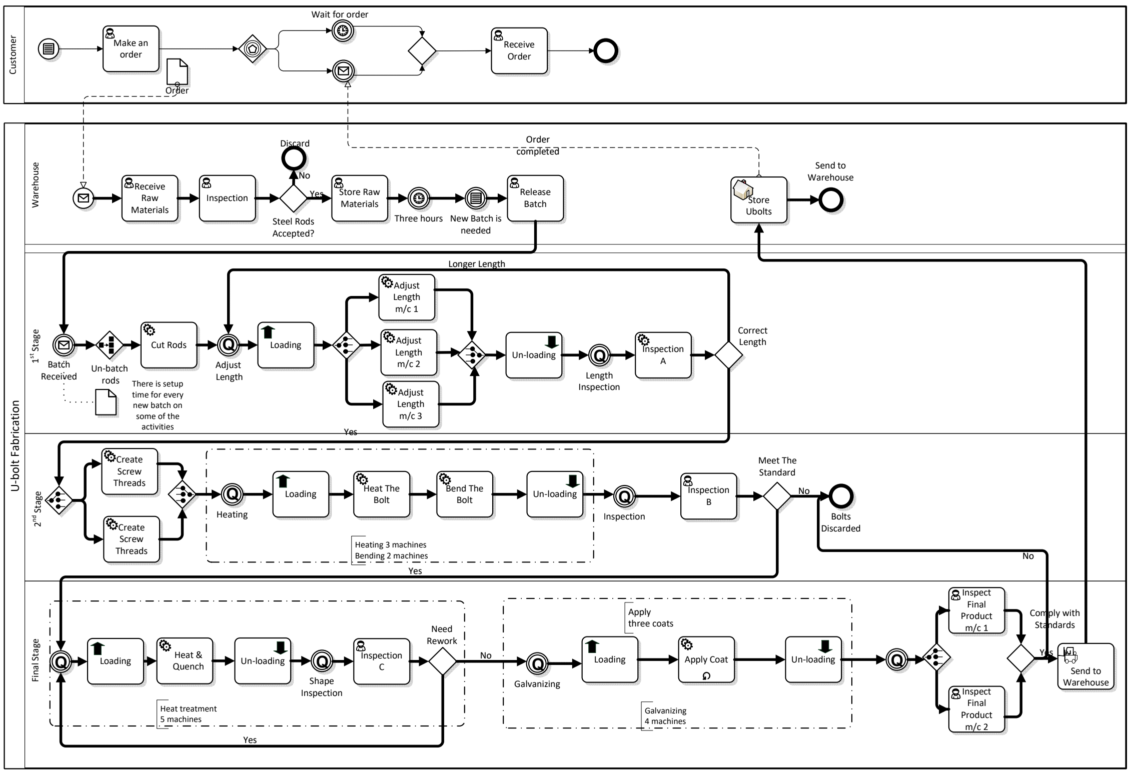

Figure 5-3: MPMN model for the U-bolt fabrication process.

The manual task notation shows that ordering and storing raw materials are the parts of the process that have to be executed manually. There are tasks showing the loading and unloading of U-bolts. The loading and unloading activities indicate the time consumed by the parts before and after processing. Any activity that includes machines in this case study is modelled using the processing task; for example, there are two types of inspection in the U-bolt fabrication process, manual inspection which is modelled using the manual task, and automated inspection which is modelled using the processing task as the machine is used to inspect the U-bolt. Another task appeared here, which is the material handling task, and is used to specify that moving the U-bolts to the warehouse need material handling equipment.

Workstation is added to the MPMN set of notations to represent the manufacturing work area that is set to perform specific tasks. It has no effect on the execution of the process, but graphically it organizes the model and showed a fewer number of machines. It uses the task annotation to write down the number of machines needed for each activity. For example, the heating and bending workstation has three machines for heating the bolts and two machines for bending.

A queue is an event added to show that parts joined a queue and started waiting to be processed; if a resource needed to load the parts and the machine is busy, U-bolts wait in a queue until an operator is free.

The use of material select gateway shows that parts are selected to be processed one by one. Besides, the material route that is used to diverge that path of the U-bolt if there is more than one machine in the station that is doing the same job, where the U-bolts go to the first available machine. The un-batch gateway shows that the rods have to be unbatched before the cutting process; in MPMN un-batching is not just a process, it is an independent notation that represents this aspect that is commonly used in manufacturing.

Also, there is a setup time with every new batch for specific processes as mentioned before; setup is modelled using the complex gateway, where every new batch the U-bolt will be directed to setup activity.

The artifact 'Data' that changed in MPMN to include rules, calculations, and equations, shows that there is an order requested from the customer and is received by the warehouse.

5.3 TRANSLATION

The translation is concerned with transforming the models' conceptualisation, presented in the previous step, into a computer recognizable form. The U-bolt case study will be modelled using three different tools; the first model will be built using ExtendSim. This full DES model will act as the base model, which will later be used to compare the output of the other two tools to its output. The second tool is the Bizagi modeler and the third tool is the newly developed MPMN Simulator.

5.3.1 Modelling with ExtendSim

Full discrete event simulation is particularly effective in the analysis and prediction of the behaviour of complex dynamic systems. As mentioned in Chapter 3, ExtendSim 9.2 simulation environment is the full DES software selected to model the U-bolt manufacturing processes and act as a base model. Figure 5-4 shows a snapshot of the U-bolt fabrication process developed in ExtendSim. Where the outputs of both Bizagi and the MPMN Simulator will be compared.

Using the create block in ExtendSim, a batch is introduced to the shop floor with a constant time between arrivals of three hours. Upon the arrival of a new batch, an attribute is defined to identify the batch number to be used later in determining whether the setup is required before specific processes or not. The un-batch block is used to split the batches released to the shop floor into U-bolts, where, the units per arrival are uniformly distributed, with a minimum of 286 U-bolts and a maximum of 400 U-bolts.

The attribute defining the batch number is used in different parts of the model to identify whether the setup is needed before certain operations or not. Once a new batch is received, the change in the batch number attribute is detected, and setup time will be added to these operations.

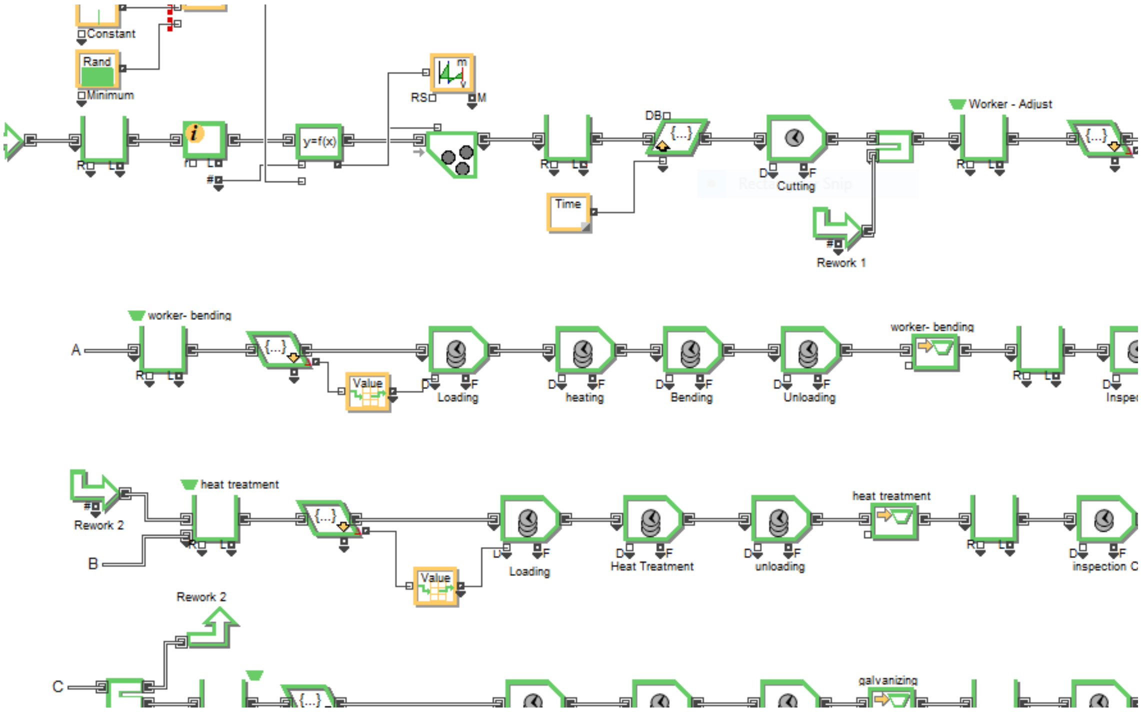

Figure 5-4: Snapshot of U-bolt case study modelled using ExtendSim.

ExtendSim can generate any custom report of any number of performance measures based on the need of the system modeller, like the average waiting time in queue for a bottleneck activity, mean cycle time, mean throughput rate, and resource utilizations; also, the modeller can clone any measure of interest and add it the custom report. All these can be reported in separate reports or in one report; either during a run, at the end of the run, or even at the end of a number of replications.

5.3.2 Modelling with Bizagi Modeler

Modelling using Bizagi involves different stages for developing a validated model that can be simulated and produce output results. As shown in Figure 5-5, this model is the first model developed using Bizagi, and it consumes a lot of effort to translate the conceptual model developed using BPMN into a computer model using the Bizagi modeler.

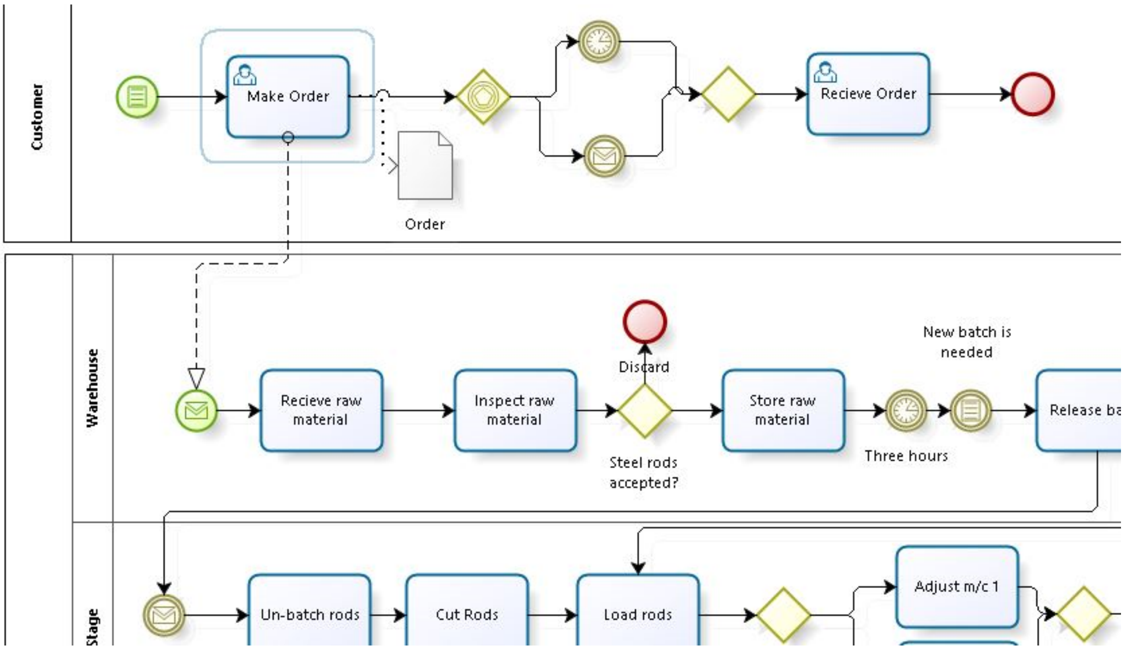

Figure 5-5: BPSim for U-bolt fabrication process (as-is model).

The second model is modified based on feedback from Bizagi's validation tool; the model is re-built to follow Bizagi's validation rules; for example, using one gateway to merge then diverge flows cannot be executed; the same gateway is either used for merging or diverging cannot do both.

Moreover, running multiple pools is not supported by Bizagi, thus the whole process is represented in one pool as shown in Figure 5-6. Besides, the use of artifacts does not affect the simulation or the analysis of the process, thus it has been deleted.

Furthermore, other BPMN elements are not supported by Bizagi's simulation engine:

- Multiple events: Start, Intermediate, and End.
- Complex gateways.
- Multiple instance tasks.
- Multiple instance Sub-Processes.

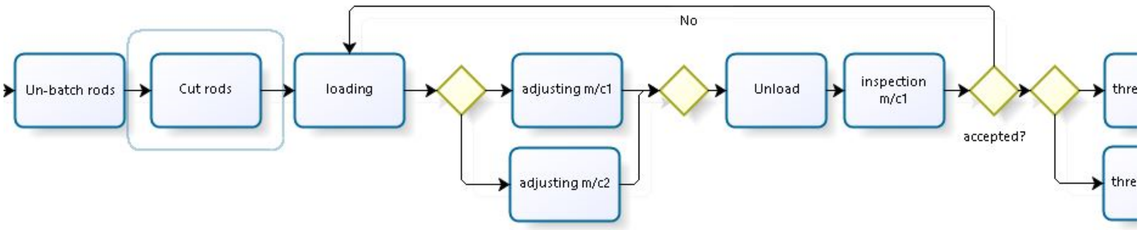

Figure 5-6: Validated BPSim model for the U-bolt fabrication process.

Assumptions Introduced

When simulating the second model developed in Bizagi, two issues were evident, which are representing the arrival of batches and how these batches are split into a specific number of U-bolts and the second issue is modelling the loading, unloading, and setup activities. For that two assumptions are introduced to resolve these issues:

The first assumption introduced to the model is for the definition of the mean time between arrivals of U-bolts. As mentioned earlier in the data collection section, a batch of 50 rods is released to the shop floor every 3 hours; where each rod is 400 cm long. The length of a U-bolt ranges from 50 cm to 70 cm; i.e., 60 cm on average. This translates into $50 \times 400/60 = 333$ U-bolts released to the shop floor every 3 hours, which in turn translates into 1.85 bolts per minute and a time between arrivals of 0.54 minutes per bolt. Furthermore, an exponential distribution is assumed to define the arrival process of the U-bolts.

The second assumption made to the model is to define processing times of activities after including loading, unloading, and setup times. The effective processing time (T_e) that is defined for the different tasks in Bizagi's is calculated using Equation 5-1. Where, the effective processing time is equal to the sum of raw processing time (t_0), loading time (L), and unloading time (UL). In addition, setup time (t_s) that is divided by the number of U-bolts per batch (N_s). Table 5-3 lists the processing times that are defined in Bizagi for all tasks.

$$T_e = t_0 + L + UL + (\frac{t_s}{N_s}) \quad \text{Equation 5-1}$$

Table 5-3: Summary of activities processing time.

Process	Processing time (min/part)	Load (min)	Unload (min)	Setup time (min/batch)	Effective PT (min/part)
Cutting	0.25				0.25
Adjust	0.80	0.2	0.2	5	1.21
Inspect A	0.40			7	0.41
Thread Rolling	0.90				0.9
Heating	1.20	0.2	0.2	5	1.61
Bending	0.45				0.45
Inspect B	0.40				0.4
Heat Treatment	1.40	0.2	0.2	30	1.86
Inspect C	0.40				0.4
Galvanizing	1.30	0.2	0.2	15	1.73
Final inspection	0.80				0.8

Thus, the third model (Figure 5-7) is modified to resolve the issue with factoring setup; setup notation is now removed and setup is now included in the tasks processing time. Also, the issue encountered when defining resources, allocating both labours and machines to some of the processes that caused a blockage in the line; thus, loading and unloading activities are deleted and the activity processing time is the effective processing time which is defined in Table 5-3. Finally, the time between arrivals of U-bolts is defined using an exponential distribution with a mean of 0.54 min/bolt to resolve the un-batching of rods issue.

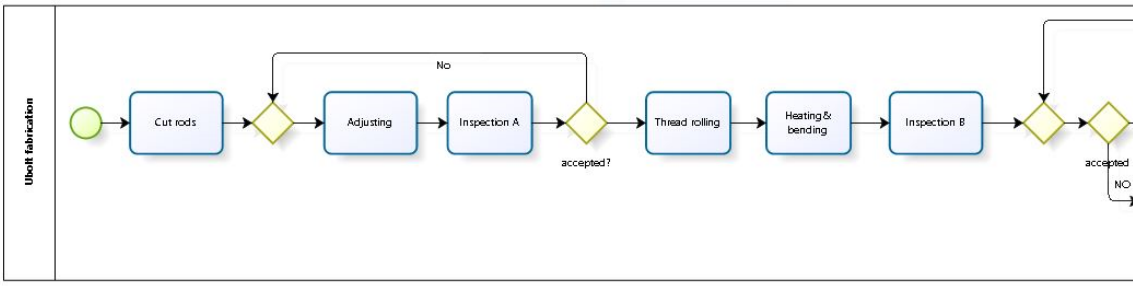

Figure 5-7: Final version of BPSim model for the U-bolt fabrication process.

5.3.3 Modelling with MPMN Simulator

As mentioned before MPMN uses ExtendSim software to develop the MPMN Simulator. As shown in Figure 5-8, the simulation model created would have the same look and feel of MPMN, and the conceptual model developed in Section 5.2.2 remains mostly intact.

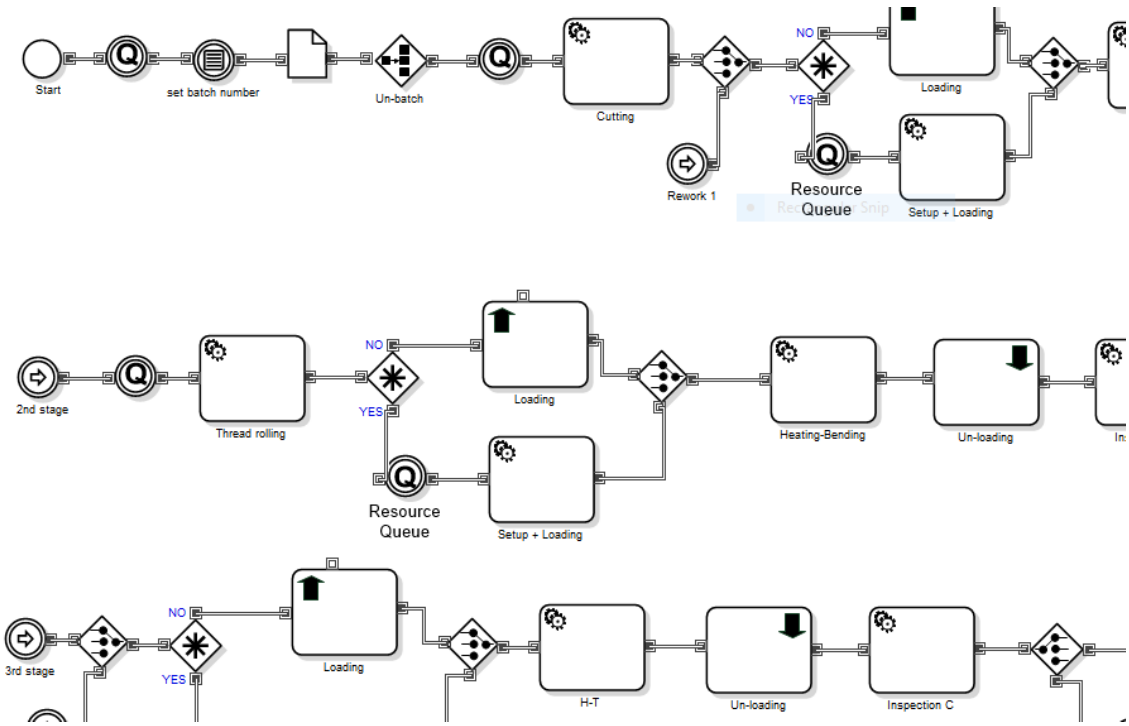

Figure 5-8: Snapshot of U-bolt case study model modelled using MPMN Simulator

MPMN Simulator uses the start event to specify the time between arrivals; a batch is specified to be released every three hours with a constant distribution. The arrival process was modelled easily due to the presence of a newly introduced Un-batch notation that splits an incoming batch into the desired number of U-bolts.

Setup was easily factored as well due to the presence of a notation that defines the batch number as an attribute. Furthermore, the presence of a gateway that can be activated based on a decision rather than a percentage only helped to identify whether a new batch is introduced, by retrieving the batch number attribute value, and; hence, whether the setup is required before the process or not.

Catch and throw events are now used to help organize the model to look like the conceptual model; the U-bolt manufacturing is divided into three stages represented above each other. Furthermore, loading/unloading tasks appeared again in the model, they were removed because Bizagi blocked the line when defining labours and machines for the same process, in MPMN Simulator there is flexibility in defining resources.

Additionally, MPMN Simulator was able to define resources using different MPMN notations; one of them is the resource pool queue, which releases U-bolts when their required resources are allocated. It is connected to a resource pool that holds a number of resources, and to a release block to release resources. Queues are now visible to show when and where a U-bolt would wait if the required resources were unavailable.

Finally, customised reporting of results for completed U-bolts only can be included in any form suitable for the user's needs.

5.4 MODEL VERIFICATION AND VALIDATION

This section will focus on validating and verifying the ExtendSim model, as it is considered the reference model; against this model, the other two models will be validated.

5.4.1 Model Verification and Validation of Simulation Models

Verification and validation are often an integrated process that occurs over the life cycle of the simulation.

Verification

Process model verification can be defined as checking whether a model shows the desired behaviour. Verification concerns the operational model; it is the process of determining that the computer programming and the model implementation represent the developer's conceptual description and specifications [121].

Validation

Validation is the process of determining that the conceptual model is an accurate representation of the real system. The best way to test a simulation model's validity is to compare its output to that of the real system [122].

Three steps of validation can be followed: face validation, validation of model assumptions, and input/output transformations validation [123, 124].

1) Develop a model with a high face value; face validation: the first goal of a simulation modeler is to develop a model that is approved on its face to model users and others who are knowledgeable about the real system under study.

2) Validation of model assumptions: testing the assumptions of the model empirically; for example, use sensitivity analysis to determine how much the model will vary with a small change in an input parameter. Model assumptions fall into two general classes: structural assumptions and data assumptions.

- Structural assumptions involve questions of how a system operates and usually involve simplifications and abstractions of reality. It is concerned with the validation of the resources (stations, machines…etc.).

- Data assumptions should be based on the collection of reliable data and correct statistical analysis of the data. It is done by confirming the input model variables that are generated randomly represent the actual variables. Making use of input distribution tests to determine proper goodness-to-fit is an important part of the validation of data assumptions.

- Validating Input/output transformations: is to determine how representative the simulation output data are. This validation is done by comparing output data from the simulation to that of the real system.

5.4.2 Model Verification

This step was carried out by using the animation capabilities of the model building software and by reporting the results of the different building blocks of the model to ensure that the model was working as intended. Also, the animation capabilities of ExtendSim were used to verify that the process flow matches the flow specified in 5.1.2

5.4.3 Model Validation

Complete validation implies that the developed simulation model is behaving just like the real-world system. The U-bolt fabrication process is a real system, so the best thing is to conduct a full validation against data from the real factory. However, validating the model based on the three types of validation in this case study was impossible.

Face validation was the first goal of this simulation model, where, the constructed model appeared to be reasonable on its face. Although it is one of the easiest types of validation consulting one of the business stakeholders or representatives was impossible in the time of writing as they went out of business. This type of validation was carried out using the animation capabilities of the simulation model, to confirm that the U-bolts were being processed in the sequence instructed in by the U-bolt fabrication process manual.

One of the model assumptions validation is the structural assumptions, and it took place by validating the number of machines for each process. For example, the cutting process has one machine; the thread rolling process uses two machines, and so on.

Input/output transformation validation was done by comparing the output utilization and number of U-bolts per shift of the ExtendSim model with respect to utilisations, and the number of U-bolts produced per shift in the basic capacity analysis presented below in Table 5-4.

Basic Capacity Analysis

To calculate the utilization of the different resources (U) presented in Table 5-4, the arrival rate of U-bolts is divided by the effective production rate of each activity. The number of resources defined for each process (m) is divided by the effective processing time (T_e) of these processes, which has been calculated previously, to determine the effective production rate (r_e).

Furthermore, as mentioned previously, a batch is introduced to the facility every three hours, then a 0.33 batch is received every hour. A batch has on average 340 bolts, this is equivalent to 113.33 bolts per hour. Thus, the arrival rate (r_a) is equal to 1.89 bolts per minute. For the adjustment and heat treatment stations, the arrival rate is equal to the original r_a in addition to the 5% rework added to processes (i.e. it is equal to $1.05 * r_a$), which is equal to 1.98 bolts/ minute.

Table 5-4: Basic capacity analysis.

Activity	T_e (min)	m	r_e	r_a	U
Cutting	0.25	1	4	1.89	47%
Adjustment	1.21	3	2.48	1.98	80%
Inspect A	0.41	1	2.42	1.98	82%
Thread Rolling	0.9	2	2.22	1.89	85%
Heating	1.61	5	3.11	1.89	61%
Bending	0.45	5	11.11	1.89	17%
Inspect B	0.4	2	5	1.89	38%
Heat Treat	1.86	5	2.69	1.98	74%
Inspect C	0.4	2	5	1.98	40%
Galvanize	1.73	4	2.31	1.89	82%
Final Inspection	0.8	2	2.5	1.89	76%

Validating input/output transformations is conducted by comparing the reported machines utilisation from ExtendSim model "Reported U" to the "Calculated U" presented in Table 5-4. It is clear that the utilizations of all the machines reported from the model are close to the values of the calculated utilization.

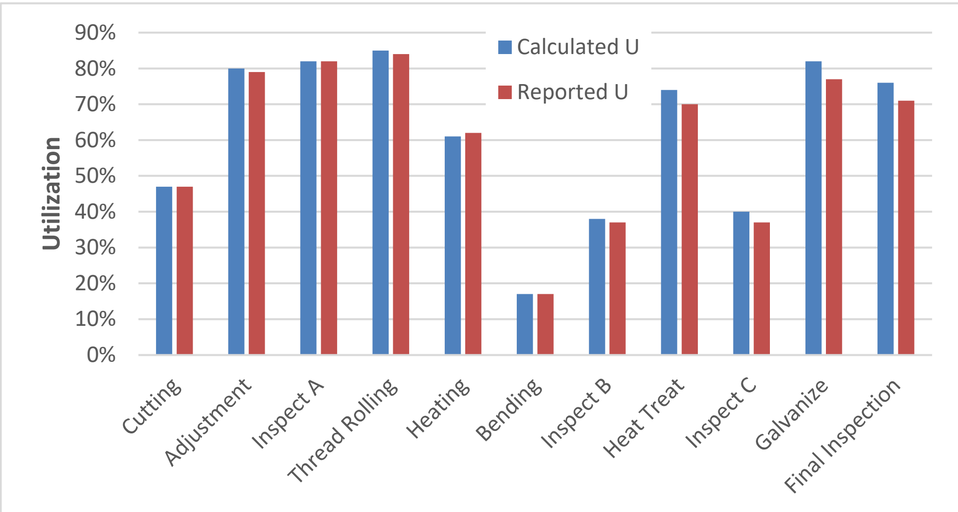

Figure 5-9: Processes Utilizations U-bolt fabrication case-study.

Also, the output TH of the ExtendSim model is verified as it matches the calculated output TH mentioned in the case study description which is almost 700 U-bolts/day.

After the conceptual model has been translated, implemented in ExtendSim verified, and validated to the extent possible; different experiments were carried out to compare between the Bizagi model and the MPMN Simulator model against the ExtendSim model.

5.5 EXPERIMENTATION AND ANALYSIS

5.5.1 Performance measures

The facility operates one 8 hours shift with a total of 1-hour breaks during the shift, resulting in a total of 7 working hours per shift. The performance measures that will be evaluated in this study are:

- *Cycle time* (minutes), is the time taken for the raw material (rod) to be a finished product (U-Bolt). This is reported as the mean of the cycle time reported for each U-bolt over the run averaged based on the outcomes of a number of replications ($\overline{\overline{CT}}$).

- *Throughput rate*, which is the number of finished U-bolts per day. This is reported as the average of daily throughput rate means reported from a number of replications $\overline{\overline{(TH)}}$.

- *Utilisation* of the resources (machines and workers), which is the percentage of time these resources are busy. This is reported as the mean of the weekly instantaneous monitored utilisation over the whole run averaged based on the outcomes of the number of replications ($\overline{\overline{U}}$).

5.5.2 Simulation Parameters

The simulation parameters that must be defined for any simulation experiment are the length of a simulation run, warmup period, and the number of replications. For each of the tools, the simulation parameters have to be defined but in this case study, these parameters will be based on the ExtendSim model.

Length of the Simulation Run

To ensure that enough output data have been obtained from the simulation, a single long run was performed which is equal to 126,000 minutes (50 weeks). However, as shown in Figure 5-10, both CT and TH are fairly steady; thus, it is decided to set the simulation runtime with 63,000 minutes (25 weeks), which is fairly sufficient.

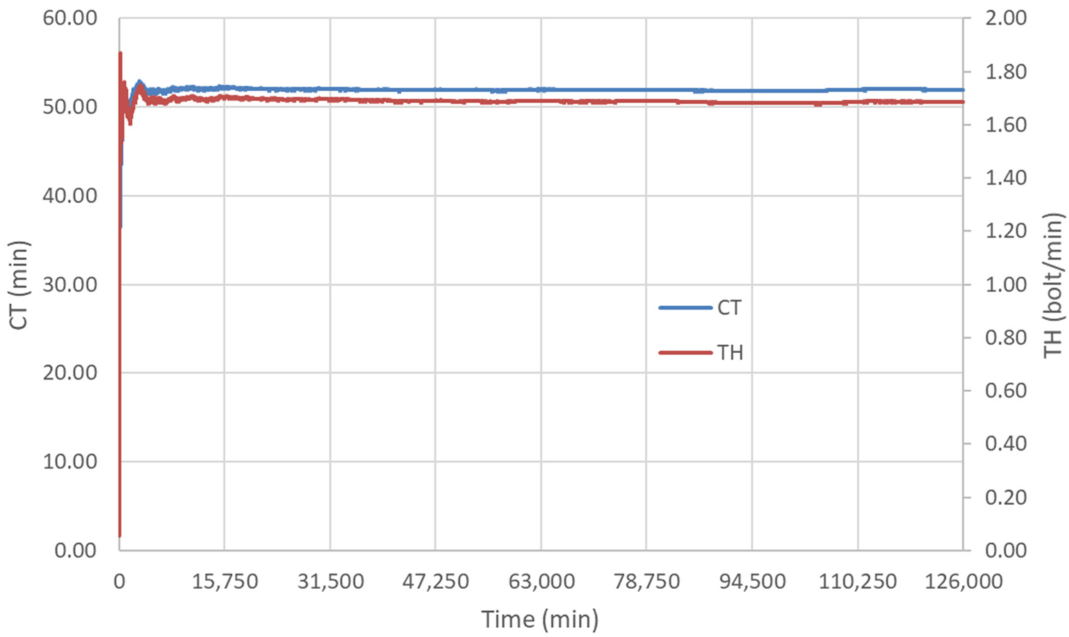

Figure 5-10: Mean TH and CT for the U-bolt manufacturing process.

Warm-up Period

To determine the warmup period, the graphical method is carried out, the warmup period is selected when the plot of both the throughput rate and cycle time are becoming fairly smooth as shown in Figure 5-11.

It should be noted that the Bizagi modeler doesn't have the capability to define warmup period, Although BPSim 2.0 specification defines warm-up period parameter, and some vendors have extended their implementation by this parameter [82], but Bizagi the most commonly used tool to build BPMN and BPSim 2.0 models still not extended to accommodate this feature.

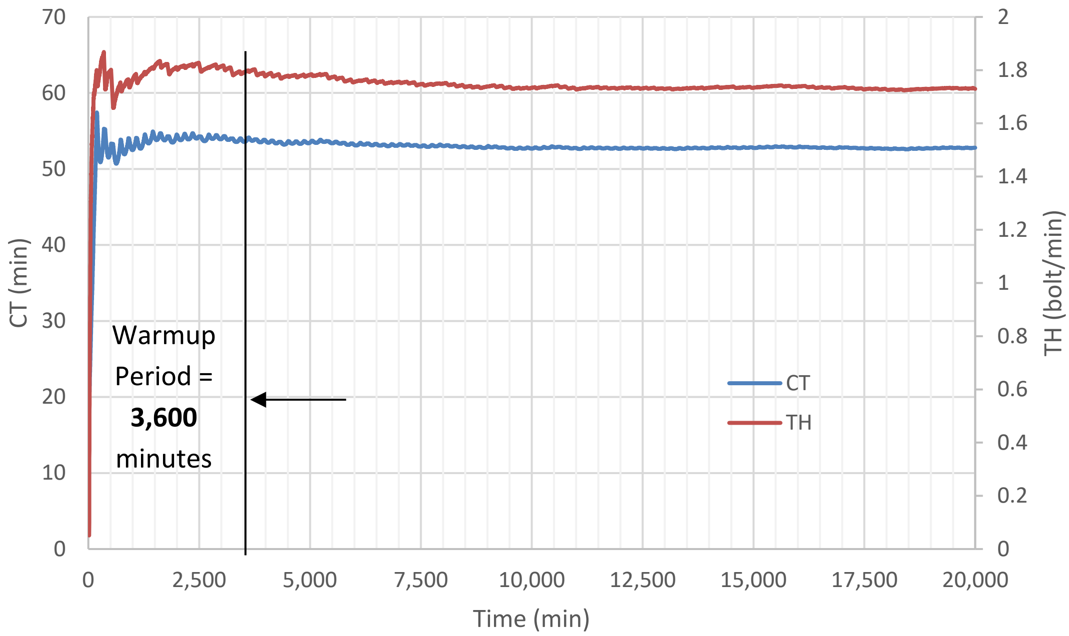

Figure 5-11: Warmup period.

Number of Replications

The confidence interval method is developed to confirm the selected number of replications for all models. The number of replications is selected at the point where the interval reaches and remains below the desired level of deviation.

Cycle time is used to determine the number of replications needed for each run. Using the simulation results of the mean CT for 20 different replications, the cumulative mean of the average cycle time is calculated.

Based on the cumulative mean, standard deviation, and the number of replications. The lower and upper confidence intervals are calculated and reported in Table 5-5. Based on the percentage deviation and the plot of the cycle time intervals shown in Figure 5-12, 20 replications are enough.

Table 5-5: Number of replications and confidence interval for ExtendSim model.

Replication	CT (min)	Cum. mean average	Standard deviation	Lower Confidence interval	Upper Confidence interval	% deviation
1	52.1	52.1	n/a	n/a	n/a	n/a
2	52.3	52.2	0.1	51.1	53.3	2.12%
3	52.1	52.2	0.1	51.9	52.4	0.43%
4	52.2	52.2	0.1	52.0	52.3	0.23%
5	52.1	52.1	0.1	52.0	52.2	0.18%
6	52.1	52.1	0.1	52.1	52.2	0.14%
7	51.6	52.1	0.2	51.9	52.2	0.37%
8	52.0	52.1	0.2	51.9	52.2	0.31%
9	51.6	52.0	0.2	51.8	52.2	0.34%
10	52.0	52.0	0.2	51.9	52.2	0.30%
11	52.1	52.0	0.2	51.9	52.2	0.27%
12	51.9	52.0	0.2	51.9	52.1	0.24%
13	51.9	52.0	0.2	51.9	52.1	0.23%
14	51.7	52.0	0.2	51.9	52.1	0.23%
15	52.1	52.0	0.2	51.9	52.1	0.21%
16	52.2	52.0	0.2	51.9	52.1	0.20%
17	51.6	52.0	0.2	51.9	52.1	0.21%
18	52.2	52.0	0.2	51.9	52.1	0.20%
19	52.2	52.0	0.2	51.9	52.1	0.20%
20	51.6	52.0	0.2	51.9	52.1	0.21%

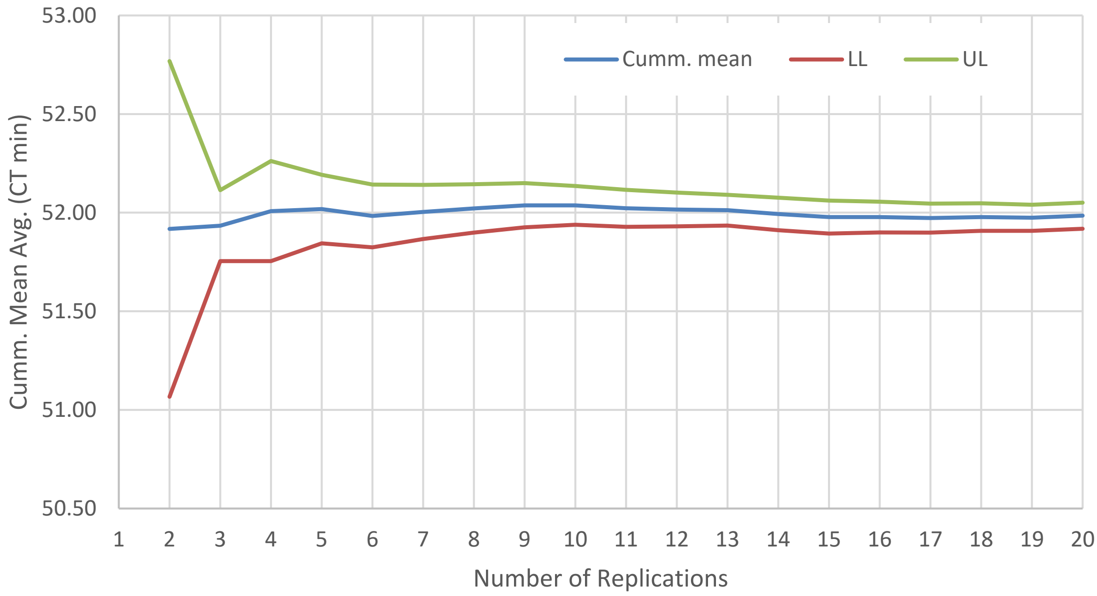

Figure 5-12: Plot of Cumulative Mean and 95% Confidence Intervals (ExtendSim).

Thus, in conclusion, it is decided that 20 replications are needed, each replication covers a simulation run time of 25 weeks (63,000 minutes), for the three tools and with a warmup period of 3,600 minutes for ExtendSim model, and the MPMN Simulator model.

5.5.3 Comparison of Results

The outcome of the simulation runs of the three tools are shown in Table 5-6, CT and TH are directly reported from the block *stats* in ExtendSim for the ExtendSim model and MPMN Simulator model, while in Bizagi Process modeler throughput is not reported directly; the average number of complete items are divided by the length of the simulation run (63,000 minutes) and then multiplied by 420 minutes (1 day) to result in the daily throughput rate.

As shown in Table 5-6, both MPMN Simulator and Bizagi reported the same bottlenecks and similar utilizations to that are reported by ExtendSim; yet, Bizagi reported shorter cycle time and higher throughput rates than the other two models.

Table 5-6: The results of the simulation runs of the three modelling tools.

		ExtendSim	MPMN	Bizagi
Mean of means CT (min)		52.0	46.3	11.6
Mean TH (U-bolts/day)		707	704	794
Utilization	**Activity**			
	Cutting	47%	47%	47%
	Adjust	79%	79%	81%
	Inspect A	82%	78%	83%
	Heat and bend	77%	77%	78%
	Inspect B	37%	37%	38%
	Thread Rolling	84%	84%	85%
	Heat treatment	70%	70%	73%
	Inspect C	37%	37%	37%
	Galvanize	77%	76%	78%
	Final Inspection	71%	71%	72%

5.5.4 Analysis of Results

It is clear from the presented results in the previous section, that the outcomes of both ExtendSim and MPMN Simulator are quite similar; MPMN Simulator is performing better than Bizagi and it delivers closer results to ExtendSim, this is because the same engine is used to simulate the models. Also, the MPMN Simulator has the same ExtendSim simulation capabilities to model some of the challenges faced, MPMN Simulator has the ability to un-batch rods using an independent notation, with no assumptions or calculations. Raw processing times were the input to the activities; Setup time is factored using the complex gateway which is capable of routing decisions based on the defined attributes. Finally, MPMN Simulator has the ability to define warmup and collecting results from the model after this point.

After analysis of the results reported by Bizagi. Specifically, ones for the cycle time and the number of completed items, it was concluded that Bizagi reports scrapped items as complete successful manufacturing process; CT averaged the time of scrapped items which is very short, and report scrap as a completed U-bolt, this affects the number of completed units (scrapped items are included in the completed instances), that is why Bizagi has higher TH than the other two models. Moreover, the time of one unit to be scrapped is considered the minimum CT in Bizagi and it is included in the average CT and this time is very short; explaining why Bizagi reported the shortest CT.

5.5.5 Discussion

BPMN is extended by researchers, either for dealing with processes of specific domains or for improving the language itself [1]. In this work, BPMN is extended to deal with the manufacturing domain, and BPMN itself was integrated with a full DES to improve data analysis, for evaluating and improving such a system.

It is possible to model most of the challenges faced with BPMN, however, different notations will be used in different manners to represent some of the manufacturing aspects. Moreover, more notations will be used to describe one aspect like for example setup. Also, using more and more notations may affect the understandability of the model. The proposal of MPMN showed a true representation of the real system and it captured the exact picture of the dynamics of the U-bolt. MPMN solved issues like representing un-batching and identifies the flow of materials. Also, where the items wait to be processed are now shown using the queue event, the use of workstations organises the model. The ‘material select gateway’ and the ‘material route gateway’ showed the diverging the routing of the U-bolts. The diagram itself Figure 5-3 has the look of BPMN and the feel of a manufacturing model.

However, simulation has to be added to support system improvements and decision making. The Bizagi was selected to model U-bolt and it has shown inadequacy to model such a simple system. In terms of lacking the required notations and the simulation capabilities to produce proper results. For example, it lacks the notations to represent setup, un-batch. Moreover, Bizagi doesn't report the throughput rate. The report only includes the number of completed instances over an entire run; which includes both finished products and scrap items. For example, to calculate the number of U-bolts (TH/day), the completed instances are divided by the run length (63,000 minutes) and multiplied by 420 (one day) to get the TH/day. This number is equal to all U-bolts including the ones that are discarded; Bizagi considers the scrapped items as finished U-bolt that successfully completed the manufacturing process.

U-bolt demonstrated that MPMN is a useful tool to graphically model manufacturing systems, thus it is integrated with a full DES to capture the full picture. MPMN Simulator added important elements to model and simulate a manufacturing domain. In the U-bolt case study, the MPMN simulator was able to use the same visual structure used in the MPMN model. Also, the MPMN Simulator was able to factor setup, define attributes, define resources, identify warmup period and generate customized reports.

6. SIMULATION OF A COMPLEX MANUFACTURING SYSTEM

This chapter will follow the same structure of Chapter 5 and will present a more challenging case study of a manufacturing system with higher complexity than the one presented before. Specifically, the second case study is one of the most popular models used by researchers in the semiconductor industry, which is the Mini-fab model. Where the developed MPMN notations and MPMN Simulator are challenged to model the Mini-fab.

This case study is selected to assess BPMN and the developed tools' capabilities to model more complex operations. It allows the evaluation of the capacity of BPMN and the developed tools to deal with the process complexities involved in scheduling re-entrant wafer fabrication facilities. Significant developments in the use of BPMN to address particular manufacturing system issues like breakdowns, changeover time, and batching have been developed and evaluated.

6.1 CASE STUDY DESCRIPTION

6.1.1 Background

The semiconductor industry is characterized by complex and expensive manufacturing processes, due to hundreds of process steps, re-entrant process flows, a variety of products, batch processing machines, and rapid and continuous technological changes as well as new technological challenges [29, 125].

The Mini-fab model, addressed in this chapter, is a model of a wafer fabrication facility that features six processing steps and five machines distributed across three stations. Although this model is of relatively small size, it captures most of the challenges found in semiconductor wafer fabrication facilities. Specifically, Mini-fab includes batching and un-batching operations, complex routing decisions, preventive maintenance, and unexpected equipment breakdowns. Most operations included requiring both time and resources to complete. Each station comprises two identical machines with exception of Station Three, which is a single machine station. Lots have to revisit each station twice to go through the six steps. Every station challenges the modelling process differently. For instance, the issue with Station One is parallel batching, for Station Two both machines are subject to emergency maintenance, and Station Three undergoes serial batching; hence, a setup time is required to changeover between the different types of lots.[3]

6.1.2 Data Collection

Structural Data

The structural data are the three stations (locations) with their five machines (resources), where:

- Station 1 has two machines, machine A and machine B.
- Station 2 has two machines, machine C and machine D.
- Station 3 has only one machine, machine E.

Resources also include the production operators; where, two production operators (PO1 and PO2) are available for 540 minutes each per shift, each gets two 60-minute breaks and one meeting per shift.

These machines are used to process three types of lots which are the standard product (Pa), standard product (Pb), and testing product (TW).

Operational Data

The operational data is the data related to:

<u>Routing of Lots</u>

As seen in Figure 6-1, to produce a lot, all production follows the sequence of:

Lots In → Step 1 → Step 2 → Step 3 → Step 4 → Step 5 → Step 6 → Lots Out.

---

[3] Further details can be found at http://aar.faculty.asu.edu/research/intel/papers/fabspec.html.

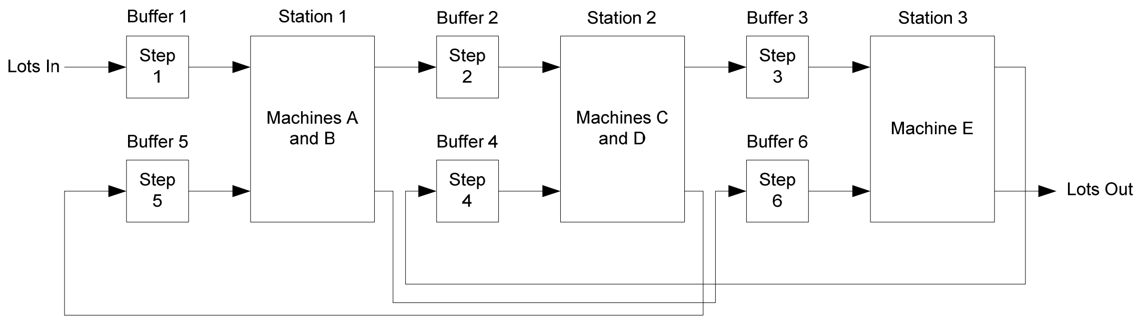

Figure 6-1: Mini-fab block diagram [126]**.**

The Preventive and Emergency Maintenance

Two types of maintenance operations affect the machines; emergency maintenance (EM) and preventive maintenance (PM). Emergency maintenance (EM) requests are surprises, the EM specified here applies to machines C and D only and happens randomly, and follows a uniform distribution with a minimum of 1,440 minutes and a maximum of 4,560 minutes. The repair, once started, takes between 6 and 8 hours; again, uniformly distributed.

Preventive maintenance (PM) requests are expected. There is a known window in which they must be started. After the window opens, and a machine is in any stage of its running cycle when the window opens, the technician cannot start the PM until the whole cycle is finished. Once the PM is started, it takes a known time to complete [115].

Batching Operations

The batching of the three lots at a time in Station One. The rule that applies to batching is that mixing lots waiting for S1 and lots waiting for S5 into the same batch is never acceptable.

Setup Time

The setup times of machine E in Station Three. In one possibility, the lot type stays the same and the step changes (S3 to S6 or S6 to S3), in another possibility, the step stays the same and the lot type changes (among Pa, Pb, and TW), in the third possibility, both the step and the lot type change (for example, going from Pa on S6 to Pb on S3).

Stations Characteristics

Station 1: serves steps S1 and S5. The issue with station 1 is *parallel batching*; where the machine can run if three appropriate lots are available to load at the same time and lots in a batch must have the same step number.

Station 2: serves steps S2 and S4. The issue with machines in this station is that both machines are subject to *emergency maintenance*.

Station 3: serves steps S3 and S6. The issue with this station is *serial batching* and; hence, setup time is required to changeover between the different types of lots.

Numerical Data

As mentioned before numerical data is the data concerning the input values and they can either be deterministic (constant) or stochastic (probabilistic), these are presented as follows:

Deterministic Data

The processing time of the steps with their loading and unloading times are shown below in Table 6-1.

Table 6-1: Summary of loading, unloading, and processing times.

Step Number	**1**	**2**	**3**	**4**	**5**	**6**
Processing time (minutes)	225	30	55	50	255	10
Loading (minutes)	20	15	10	15	20	10
Unloading (minutes)	40	15	10	15	40	10

In Station 1 there are 75 minutes of scheduled maintenance every 24 hours (2 shifts) for each machine (machine A and machine B), in Station 2 there is a 120 minutes scheduled maintenance every 12 hours shift for each machine (machine C and machine D), in Station 3 there are 30 minutes scheduled maintenance every 12 hours shift as summarized in Table 6-2.

Table 6-2: Summary of scheduled maintenance.

Station	Machine	Up Time (minutes)	Down Time (minutes)	Frequency (minutes)
1	A/B	1,365	75	1,440
2	C/D	600	120	720
3	E	690	30	720

As mentioned earlier, due to the existence of only one machine at Station 3, product changeovers result in different setup times. When the lot type stays the same and the step changes (S3 to S6 or S6 to S3), this takes 10 minutes, when the step stays the same and the lot type changes (among Pa, Pb, and TW) this takes 5 minutes when both the step and the lot type change (for example, going from Pa on S6 to Pb on S3) this takes 12 minutes.

Stochastic Data

Different stochastic inputs are identified in the Mini-fab model, which are summarized in Table 6-3. The inter-arrival of items is exponentially distributed with a mean of 120 minutes, which is equivalent to an average of 84 lots per week. Also, the three different product types produced Pa, Pb, and TW follow an empirical distribution with probabilities 0.61, 0.36, and 0.03; respectively. Finally, the unscheduled breakdowns for machines C and D at Station 2 are uniformly distributed with a minimum of 24 hours and a maximum of 76 hours. The repair time is also uniformly distributed with a minimum of 6 hours and a maximum of 8 hours.

Table 6-3: Summary of stochastic data.

<table>
<tr><th>Input Data</th><th>Distribution</th><th colspan="2">Parameters</th></tr>
<tr><td>Lot generation
(Inter arrival time)</td><td>Exponential</td><td colspan="2">Mean = 120 minutes/lot</td></tr>
<tr><td rowspan="4">Product type;
Pa = 1
Pb = 2
TW = 3</td><td rowspan="4">Empirical</td><td>Value</td><td>Probability</td></tr>
<tr><td>1</td><td>0.61</td></tr>
<tr><td>2</td><td>0.36</td></tr>
<tr><td>3</td><td>0.03</td></tr>
<tr><td>Unscheduled breakdown
(machines C and D)</td><td>Uniform</td><td colspan="2">Min = 1,440 minutes
Max = 4,560 minutes</td></tr>
<tr><td>Repair time
(machines C and D)</td><td>Uniform</td><td colspan="2">Min = 360 minutes
Max = 480 minutes</td></tr>
</table>

6.2 CONCEPTUALIZATION

Model conceptualization is carried out using both BPMN and MPMN. First, the BPMN is introduced and how it is challenged to model more complex case-study. Then MPMN is used to model the Mini-fab case-study to evaluate its capability to capture important aspects that are commonly presented in manufacturing systems.

6.2.1 Modelling with BPMN

Four BPMN models are developed for the Mini-fab; the first model is the top-level model showing the three stations Figure 6-2. This top-level model represents both the material and message flow along the entire production line.

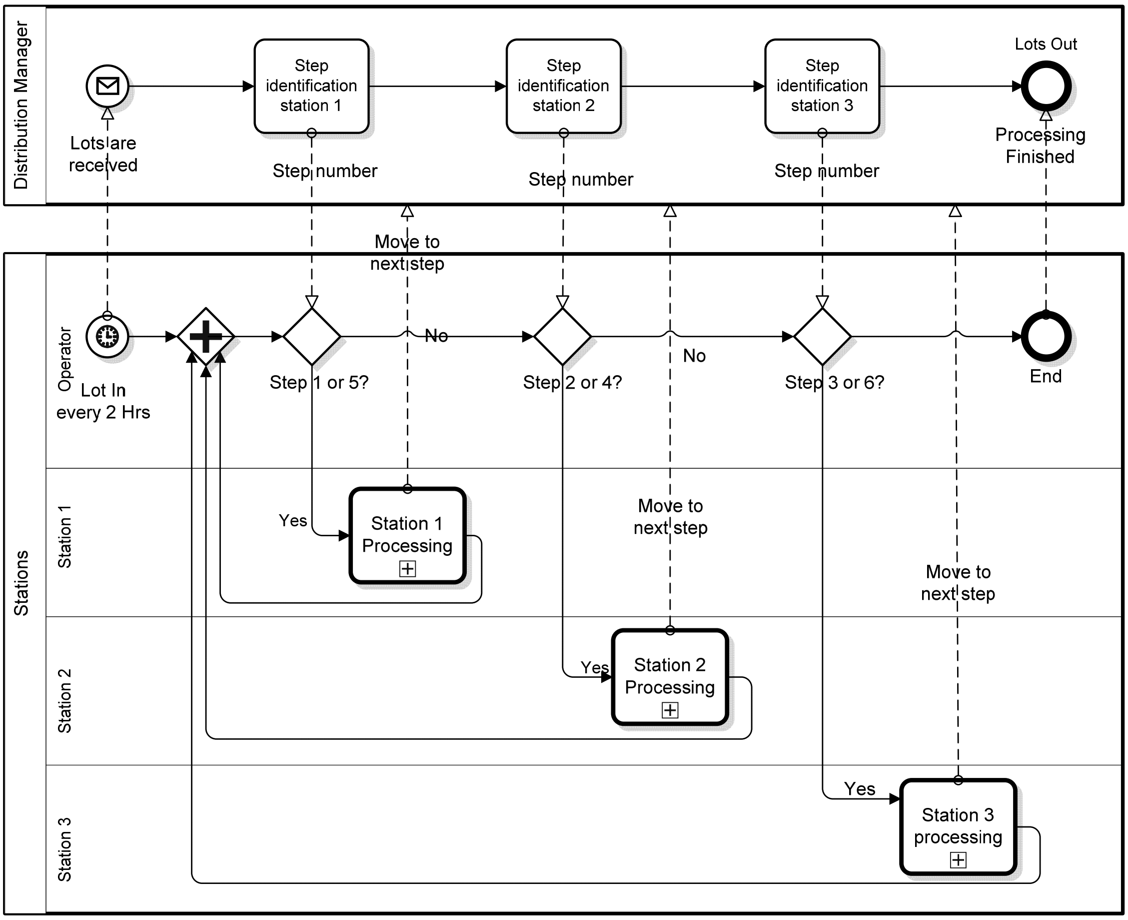

Figure 6-2: BPMN Model for the Mini-fab Process.

The Mini-fab is represented by two Pools; the "Distribution Manager" pool and the "Stations" pool, which in turn is divided into four lanes; one lane for each of the three stations and a lane for the operators. Interaction between the distribution manager and the different stations on the shop floor is modelled as a collaboration diagram. The distribution manager coordinates all operations in the Mini-fab; it sends and receives operator status, monitors step number at each station, and sends messages to move lots to the next step. As previously mentioned, lots have to revisit each station twice to go through the six steps. The steps served by each station are quite clear in the collaboration diagram.

A call activity that is "thickly bordered" is usually used to represent a repeated business activity; yet, it has been employed here to represent sub-process in manufacturing. Stations are modelled as call activities their collapsed sub-processes are presented in the following models. The model represents both material flow and information flow, the information is the dashed lines between the different pools and the material flow is the solid line between the swimlanes. The process itself begins when the lots are received by the shop floor, the operator is responsible for notifying the distribution manager that the lots have arrived, and the distribution manager is responsible for keeping track of lots progress and informing the operator of the next processing step.

In addition to the re-entrant flow, every station presents a different challenge for the modelling process, the issue with Station One is parallel batching, in Station Two both machines are subjected to emergency maintenance and the issue with Station Three is serial batching, hence a setup time is required to changeover between the different types of lots. To show the different issues of each station in more detail, each station is then expanded in a separate model. It is must be noted that in the following models for each station; the distribution manager is represented by a "black-box pool", it doesn't show any processes, or in other words it is hidden; this helps to focus on the main processes performed at each station.

Figure 6-3 shows the collapsed sub-process of Station One. A parallel event gateway is used to indicate that the process is depending on two events; there has to be a free operator and an available machine to start the job. The maintenance technician was added in a separate pool to Station 1 and other stations, to show that he is a different participant. The maintenance technician's job is to repair the machine and to attend 30 minutes meeting each shift.

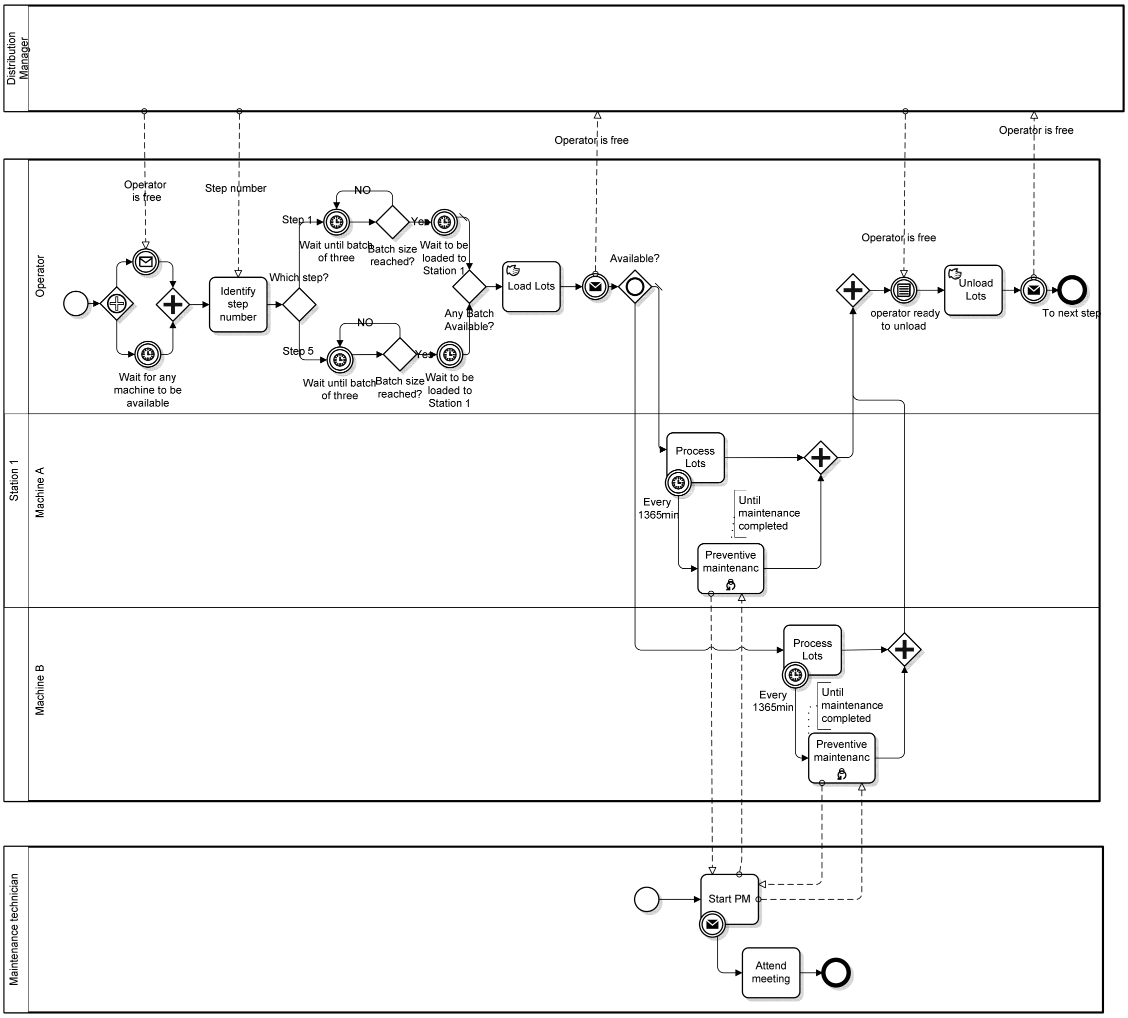

Figure 6-3: BPMN Model for Station 1.

The issue with Station 1 is parallel batching. Figure 6-4 shows how the batching operation is modelled using BPMN. An exclusive gateway is first used to split lots based on the step number and then an intermediate timer event represents the waiting of the arrived lots until the required batch is formed. Once batching is complete and the machine is available, it is loaded to the machine and the distribution manager is informed of the free operator. After processing at any machine, lots are unloaded given that an operator is free. Again, the distribution manager is informed of the free operator after the unload operation is complete.

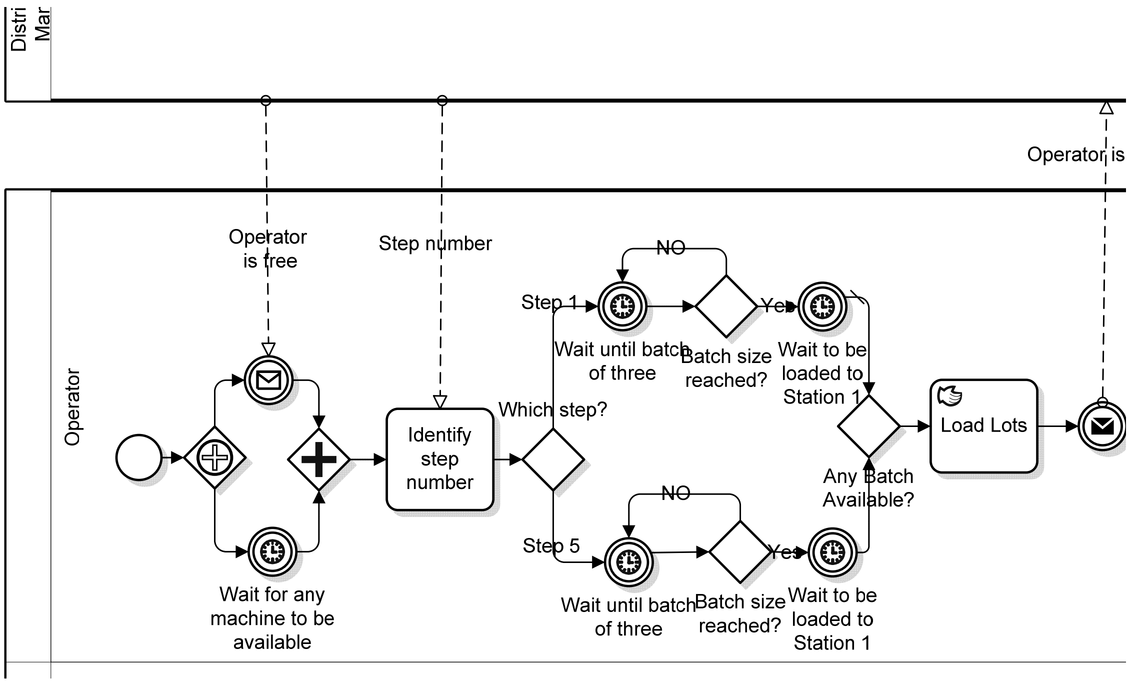

Figure 6-4: Modelling of Batching for Station 1 using BPMN.

The collapsed sub-process of Station Two, which serves Step 2 and Step 4, is shown in Figure 6-5. If a machine is available, a free operator will start the job, identifies the step number, and loads the lots to the station.

The issue with machines in this station is that both machines are subjected to two types of maintenance: preventive maintenance and, unlike the other stations, emergency maintenance. Figure 6-6 shows how these two types of stoppages were introduced to the machines. The two solid circles attached to each machine is called “border events” they are interrupting events. Meaning that any lots processing being performed will immediately be cancelled and will follow the path defined by the boundary events.

For preventive maintenance, a “timer boundary event” is used because the machine will take a certain known time to open the window for repair. On the other hand, “error boundary event” was selected for the emergency maintenance because this type of breakdown happens in an unplanned manner. The maintenance technician is responsible for starting the preventive maintenance and repairing the emergency maintenance.

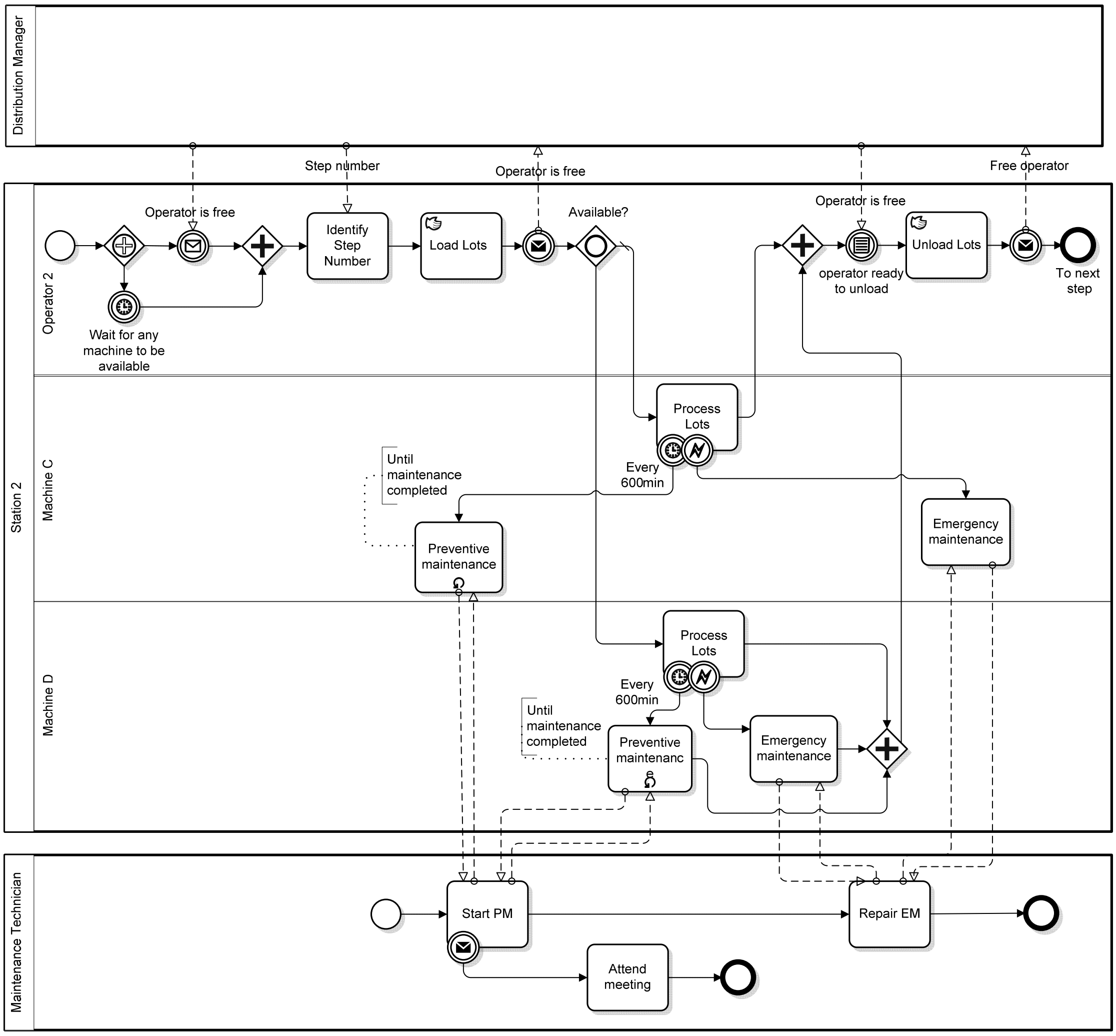

Figure 6-5: BPMN Model for Station 2.

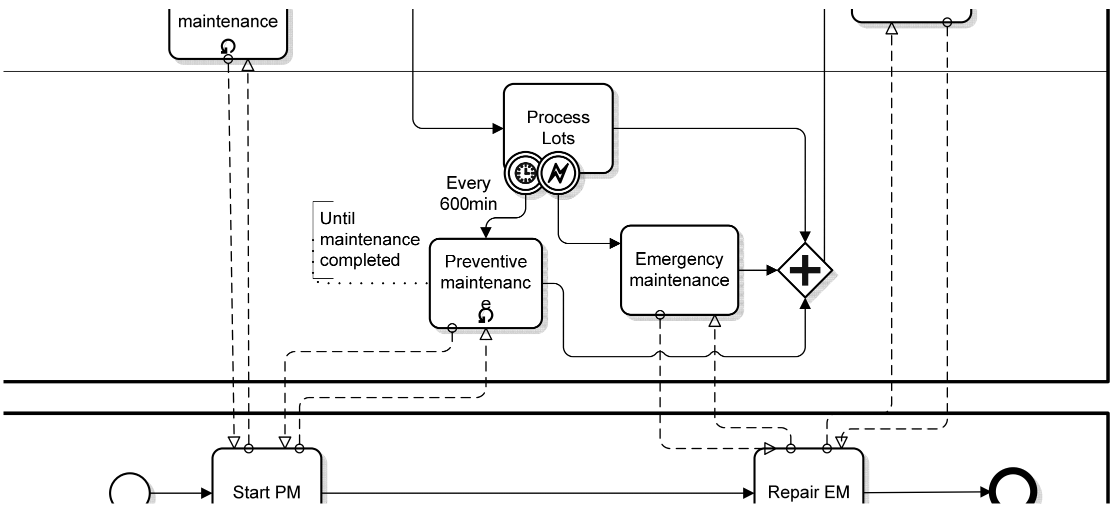

Figure 6-6: Modelling maintenance operations for Station 2 using BPMN.

Figure 6-7 represents Station 3; this station has one machine only and is subjected to one type of maintenance. The process starts with a conditional start event, indicating that the process will not start until an operator is free to load the lots, and when the operator is ready, he first checks the machine availability to start processing. Also, Step 6 is performed at this station, which is the final step needed to complete a lot. Thus, a message boundary event is attached to the unloading lots task, to detect if the unloaded lot performed the final processing step and to move the completed lots out of the Mini-fab or not.

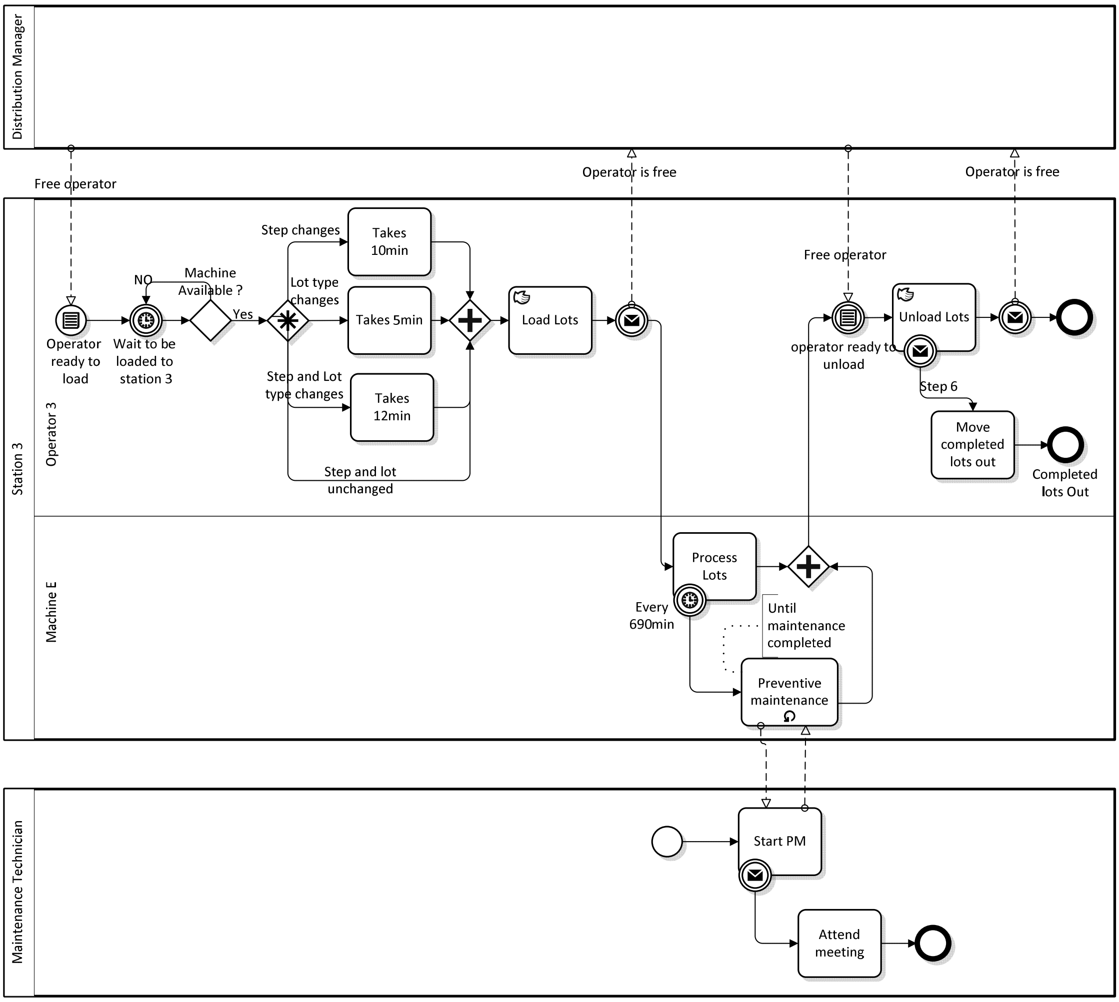

Figure 6-7: BPMN Model for Station 3.

Station 3 is the only station with one machine; hence, the issue with this station is serial batching, and setup time is required for changeover between batches. A complex gateway is used here because three types of lots changes can occur: if only the step changes this will take 10 minutes, if only lot type changes this will take 5 minutes, and if both step and lot type change this will take 12 minutes. Of course, if both step and lot type remain unchanged, loading of lots should start immediately, as shown in Figure 6-8.

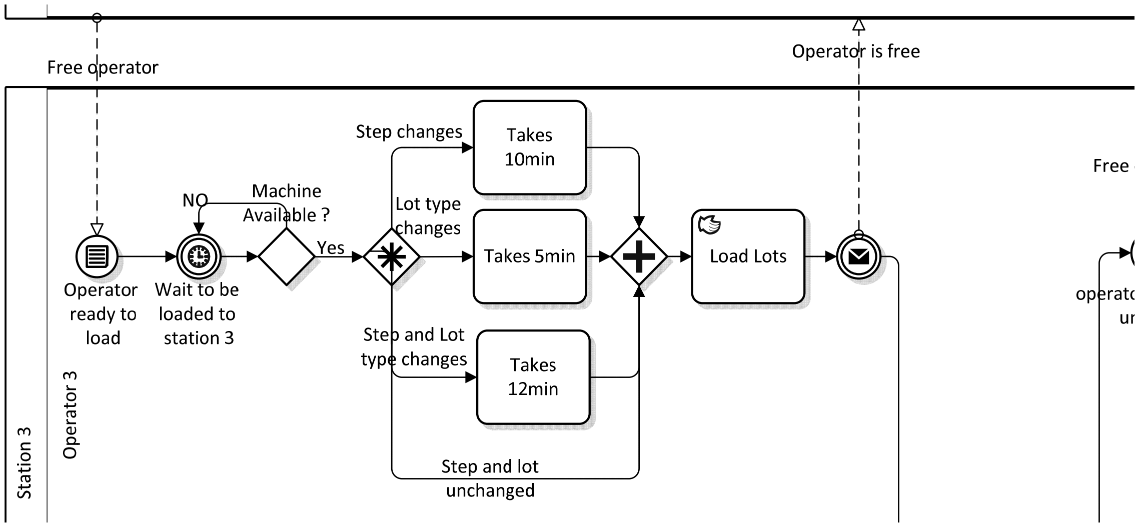

Figure 6-8: Modelling of serial batching for Station 3 using BPMN.

Challenges and Opportunities

This case study captures most of the challenges involved in scheduling re-entrant wafer fabrication facilities. The development of the BPMN model has revealed the strengths and weaknesses of the BPMN notation while representing manufacturing systems, where the accumulation of product and the timing of activities are critical to performance.

Separated pools and swimlanes helped in representing active actors in the model, no difficulties were faced in modelling the re-entrant flow of lots, the model reader can follow the sequence of steps easily, where the five machines were illustrated clearly and the flow of lots through the processes is obviously shown. With the aid of gateways and events, a challenge like parallel batching was represented using the BPMN notations. Tasks with border events helped to differentiate between the two types of maintenance and the resources needed for repair were represented in this challenge flawlessly. Complex gateway was used to allow the representation of the setup times rules involved in serial batching.

Despite the proven success of BPMN in illustrating some of the projected challenges, BPMN failed to represent some of the manufacturing aspects from the modeller perspective:

- Each station was supposed to be modeled as a sub-process but sub-processes cannot have pools or lanes and are not a true hierarchal representation of the parent activity; the BPMN sub-process is different than the manufacturing sub-process, and therefore each station is modeled in an expanded “call activity” where the reusable sub-processes is called “call activities”.
- Operator’s shifts schedules and training times are not shown in this model it was difficult to represent them.
- It was impossible to show how parts are moved between the different stages because there are no clear notations of any of the material handling equipment.

BPMN is a static representation of the whole system; it is not clear how powerful it is to improve the system under study; WIP, throughput, cycle time… etc. cannot be calculated. For example, no buffer or queue to report the number of parts waiting, in emergency maintenance the time to repair could not be specified, and also no reports of the machine status whether it is up or down.

Although BPMN might have the capabilities to represent all the difficulties faced, but it will lose the concept of the model, more and more notations have to be used, thus common operations of manufacturing will be demonstrated by a complex BPMN model.

6.2.2 Modelling with MPMN

The newly developed MPMN is now used to model the Mini-fab fabrication process. Sub-processes are used here as decomposable parent activities; these expand to show the child's activities and can include pools and lanes. To ease the understanding of the model, the top-level diagram was modified to include the three types of MPMN connectors, as shown in Figure 6-9. Where, sequence flow indicates the sequence of processes, material flow shows the flow of lots throughout the Mini-fab, and message flow indicates the flow of information between different actors.

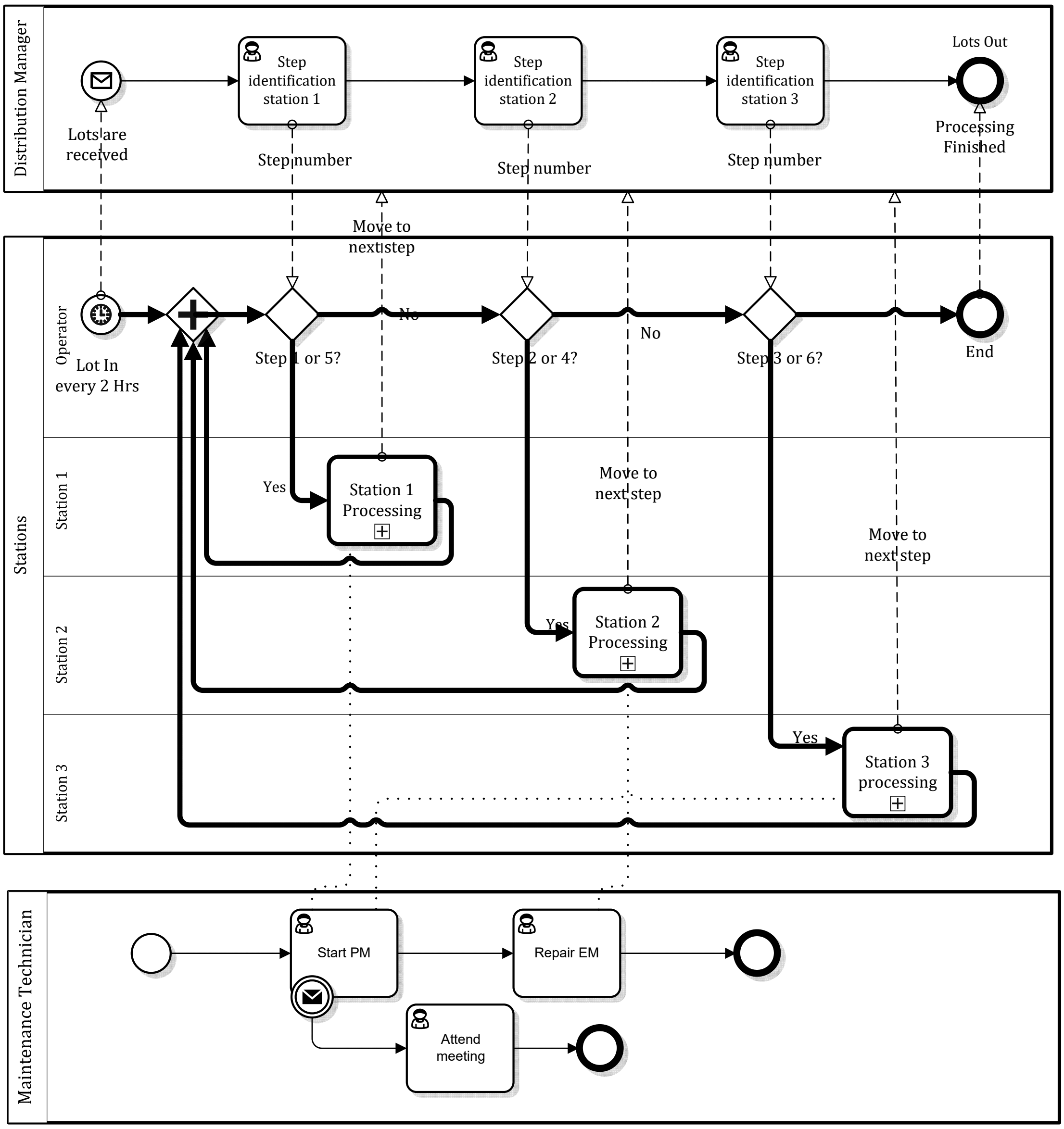

Figure 6-9: MPMN model for Mini-fab process.

Figure 6-10, represents Station 1, the issue with Station 1 is parallel batching, the process starts when the lot arrives; catch and throw events which are added to the MPMN set of notations, are used here to route lots from the last completed manufacturing step to the next one. The distribution manager sends the step number to the operator to be identified manually.

The newly proposed batching notation is used here to show that the lot has to be a batch of three having the same step number to be loaded to the machine to start processing.

Any task that is subject to failure is represented by a round corner rectangle with a wrench on the top left corner. Furthermore, a border event is attached to these tasks to represent the scheduled maintenance operation.

Also, the queue event is added before any machine, to indicate where parts should start waiting for a machine to be free. The resource pool, which is a newly proposed artifact is used to seize the resources used in the model; when an operator is required a resource is seized from the pool, and when the machine is unloaded the resource is released to the pool. The other resource pool that is shown is the maintenance technician pool; where, an operator is seized to handle scheduled maintenance.

Furthermore, the new notations for material route and material select are used here to route the lots to one of the available machines to be processed and after processing, the material select gateway is used to select the lot to be unloaded.

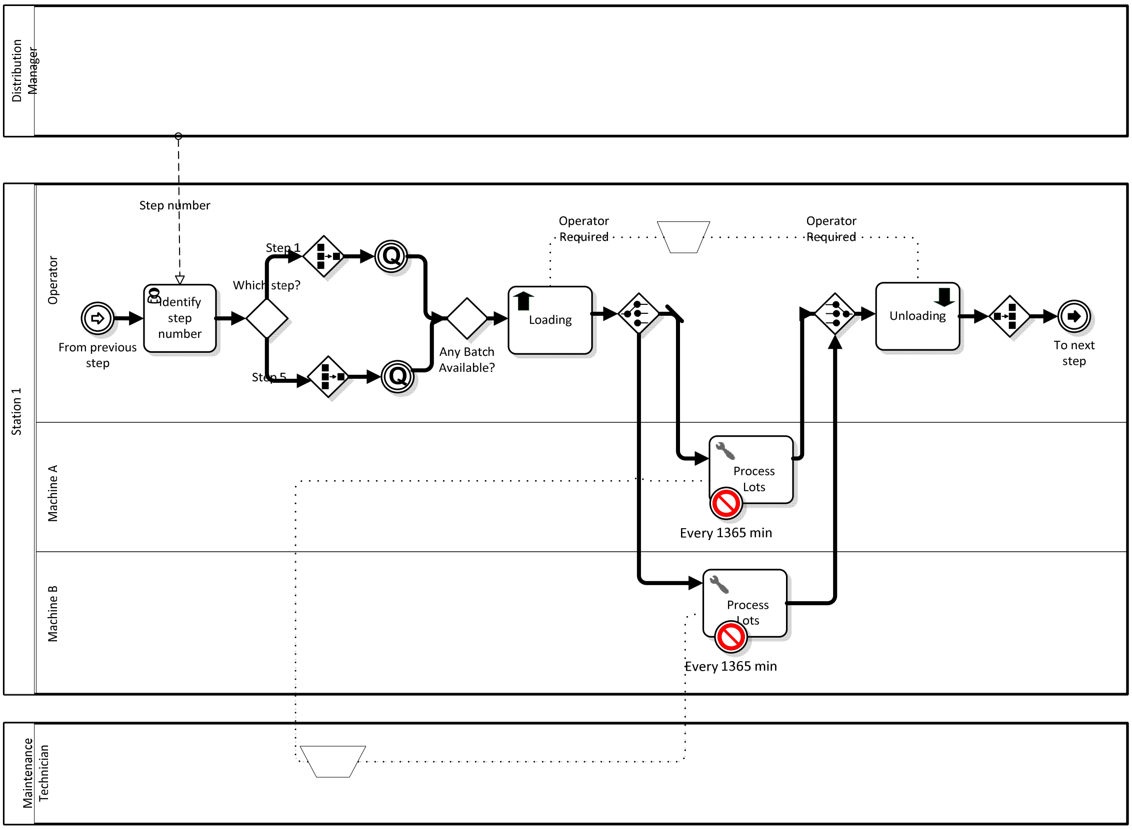

Figure 6-10: MPMN model for Station 1.

Station 2 MPMN model is shown in Figure 6-11. As mentioned before, the issue with the machines in this station is that both machines are subject to two types of maintenance.

The border events are likely to be attached to such a task; the timer border event is responsible for representing the meantime to failure, and the time to failure event is responsible for representing the time to repair. Both failures are done by the maintenance technician.

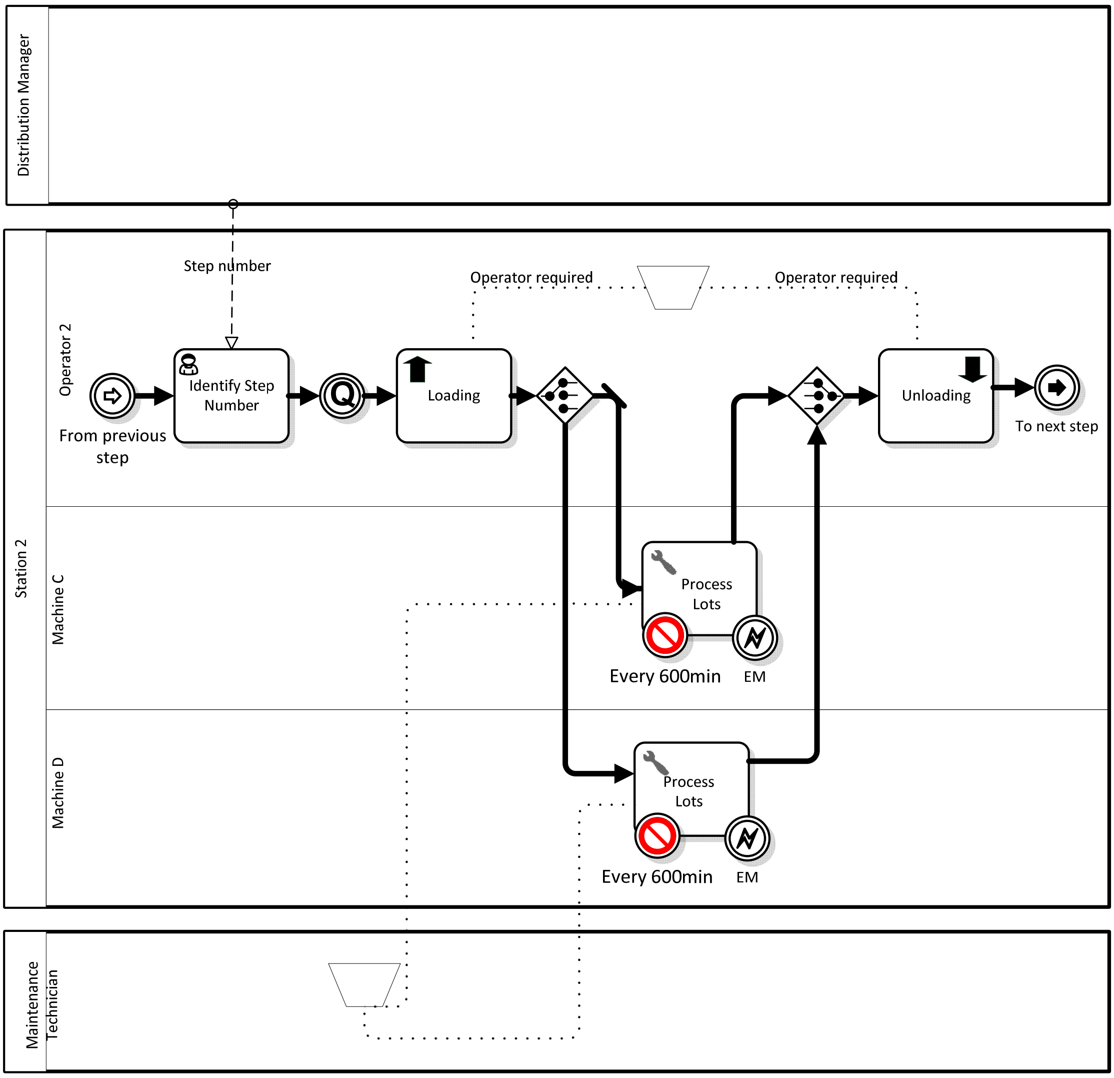

Figure 6-11: MPMN model for Station 2.

An MPMN model for Station 3 is shown in Figure 6-12, the issue with this station is serial batching and the setup time for the changeover between batches. Complex gateway is added to MPMN set of notations, to model complex decisions based on attributes, and used here to model setup time. A material select gateway is used before the complex gateway to select lots in before changes occur. After the process is finished the lots are either moved to the next step or moved out and this route is directed using the throw event.

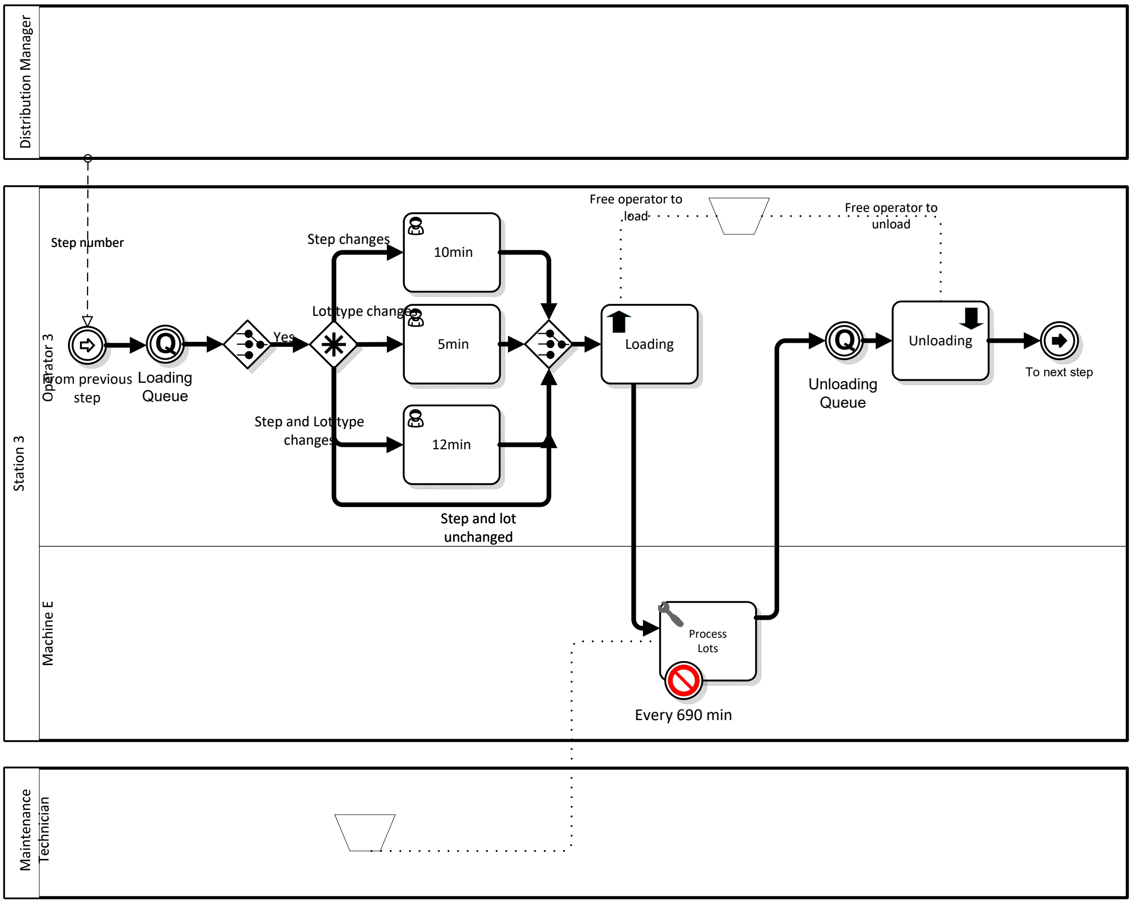

Figure 6-12: MPMN model for Station 3.

6.3 TRANSLATION

Following the same structure of the previous chapter, conceptual models developed using BPMN and MPMN are translated into computer recognizable format using the three simulation tools mentioned in Chapter 3 and Chapter 4; ExtendSim, Bizagi, and MPMN Simulator, this to assess their simulation and analysis capabilities.

To simplify the simulation models developed in this section; the following assumptions are introduced:

- Batching rules were simplified.
- Random failure and random repair times are exponentially distributed.

- Maintenance technicians were not modelled.
- No rework is needed.
- This model does not include travel times.
- Tool processing times are deterministic.

6.3.1 Modelling with ExtendSim

The steps undertaken for the previous case study to compare between tools' capabilities are followed here. First, the simulation models for the Mini-fab fabrication process are developed in the ExtendSim environment as shown in Figure 6-13.

The ExtendSim has all the capabilities to model all the issues concerned with the different stations; yet, the conceptual models developed using BPMN or MPMN are completely changed; the ExtendSim blocks replace the MPMN notations developed earlier. However, it was possible to model all the challenges faced in this case study using ExtendSim.

It has the ability to define attributes for entities, which helped in defining different lot types based on the probability distribution presented in section 6.1.2, which can be used later in defining changeover between different lots. Moreover, another attribute which is the step number can be defined for incoming lots which is a key for all routing decisions needed for defining re-entrancy batching and setup time. This can be done using the set block which is used to set the step number, and it increments the step number by one whenever a manufacturing step is completed. Based on the new step number, the lot is directed to the specific station based on the routing defined in the data section 6.1.2. Furthermore, using the Equation block, for every step the processing time is calculated and updated to match the step number.

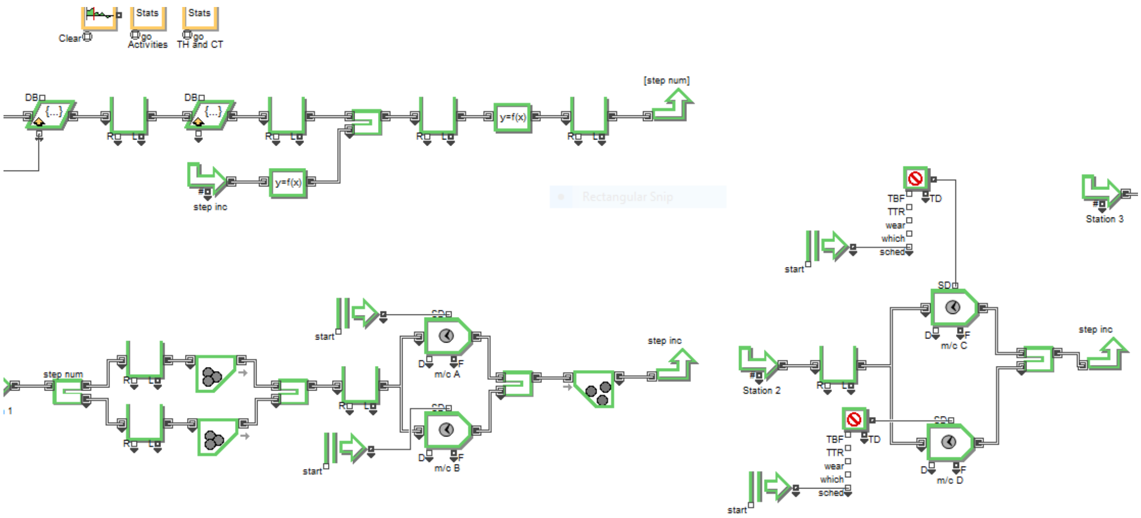

Figure 6-13: Snapshot of Mini-fab case study modelled using ExtendSim.

For Station 1 modelling, ExtendSim has blocks that specify batching and un-batching of items this block specify batching activity in addition to the wait-to-batch-time. For Station 2, unpredicted failures are modelled using the Shutdown block which generates "up" and "down" signals over time and sends it to machines C and D. Also, time to failure (TTF) and time to repair (TTR) can be set with the specified distribution in the data collection section. Station 3 issue was setup time for machine E, which is based on the step number and the product type. Get blocks are used to identify whether the step number and lot type attribute values are changed or not and this is used to calculate the setup time required. A schedule to model preventive maintenance are created for all machines in all stations, this create block are connected to the activity, and maintenance frequency can be set.

This is in addition to the customized reports that ExtendSim is capable to create for every scenario and for any selected block. In this case study, customized reports for average CT and TH are generated also reporting of machine and resources utilization which can be reported continuously during a run and summarized statistically after completing a run for every single replication and for all replications.

6.3.2 Modelling with Bizagi Modeler

As mentioned in the previous chapter, simulation in Bizagi Modeler starts with an as-is transfer of the BPMN conceptual models presented in Section 6.2.1 into Bizagi. Four models are used to represent the Mini-fab fabrication process: the first one is the top-level activity shown in Figure 6-14, it represents the three stations and graphically represents the re-entrant flow between the stations. However, these as-is models cannot be simulated. Bizagi Modeler does not support the modelling of message flows to/from the internal elements of a sub-process. The Bizagi doesn't understand sub-process as a true hierarchal representation of the parent, because in BPMN a sub-process cannot include pools or lanes as shown in the figure.

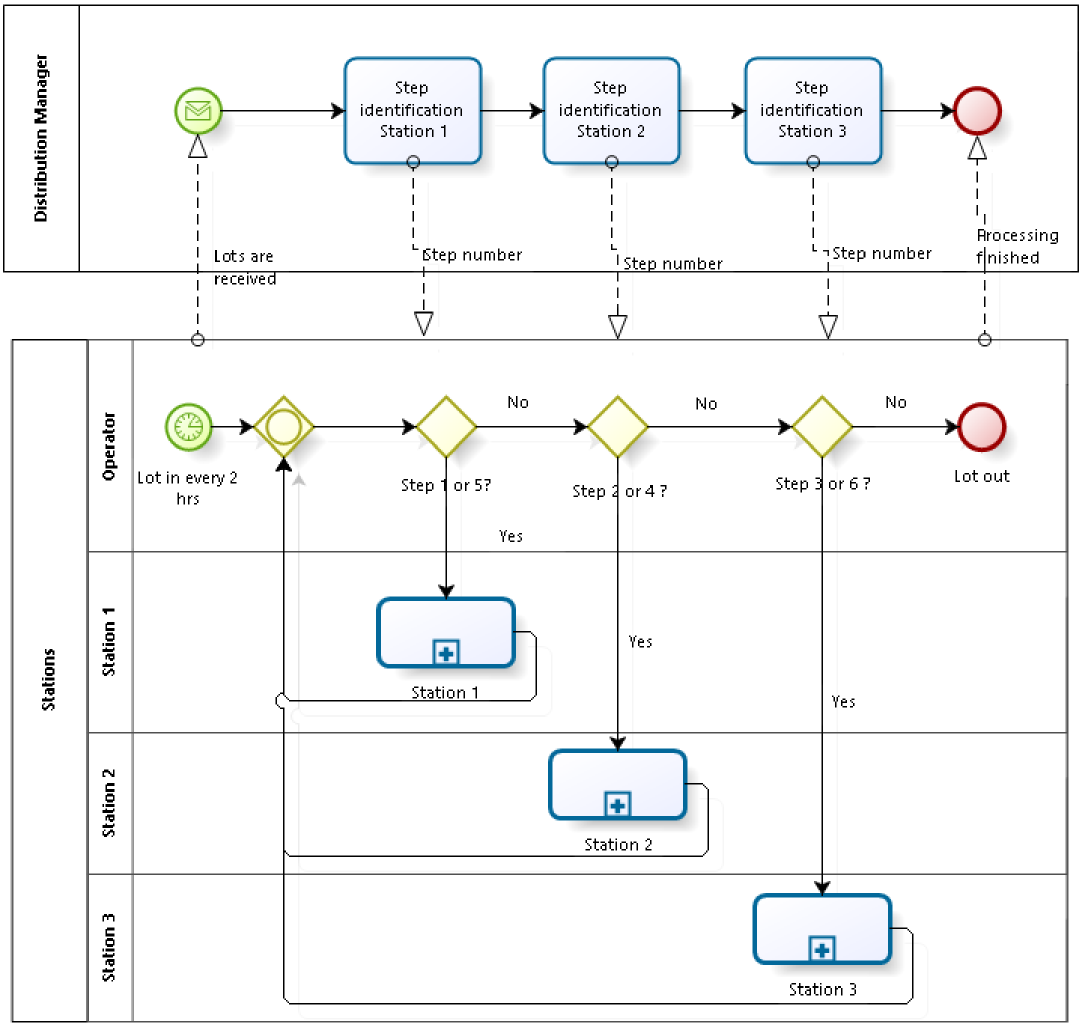

Figure 6-14: Mini-fab processes in Bizagi modeler (as-is model).

Station 1, Station 2, and Station 3 are each modelled in a separate window without decomposition as shown in Figure 6-15, Figure 6-16, and Figure 6-17; respectively. However, Bizagi simulated each window separately; it was mentioned before that running sub-processes with multiple pools are not supported by Bizagi.

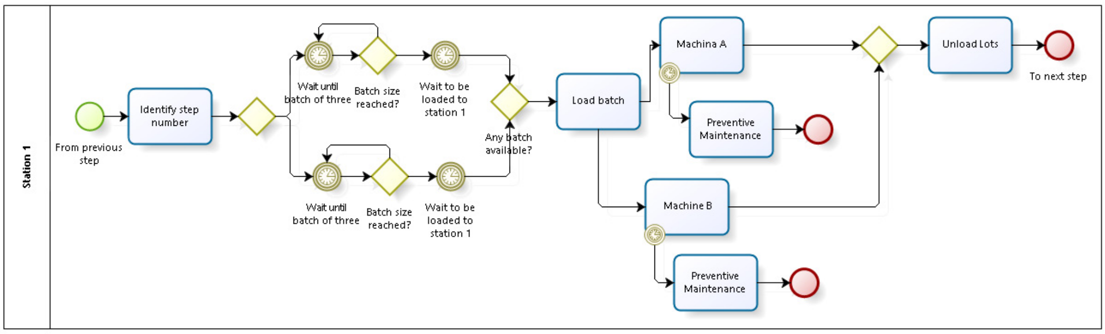

Figure 6-15: Station 1 modelled in Bizagi modeler.

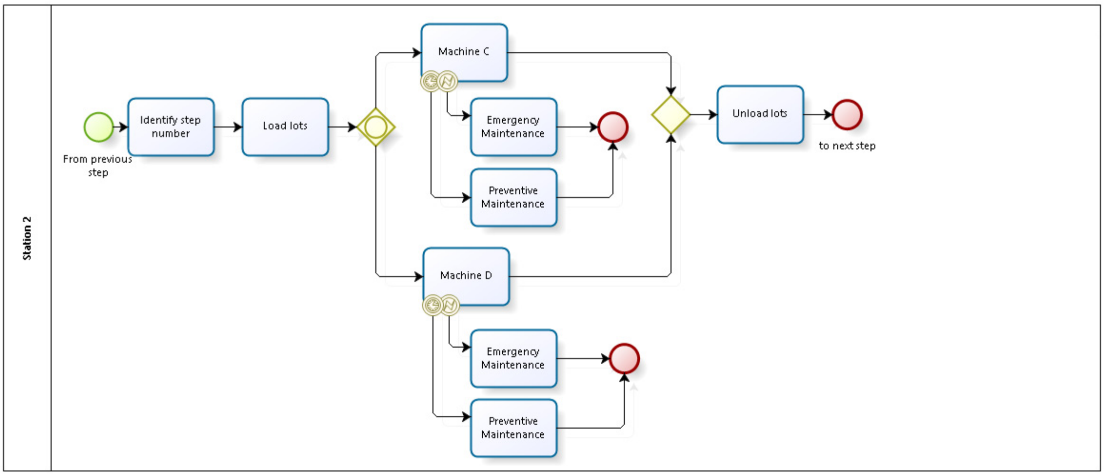

Figure 6-16: Station 2 modelled in Bizagi modeler.

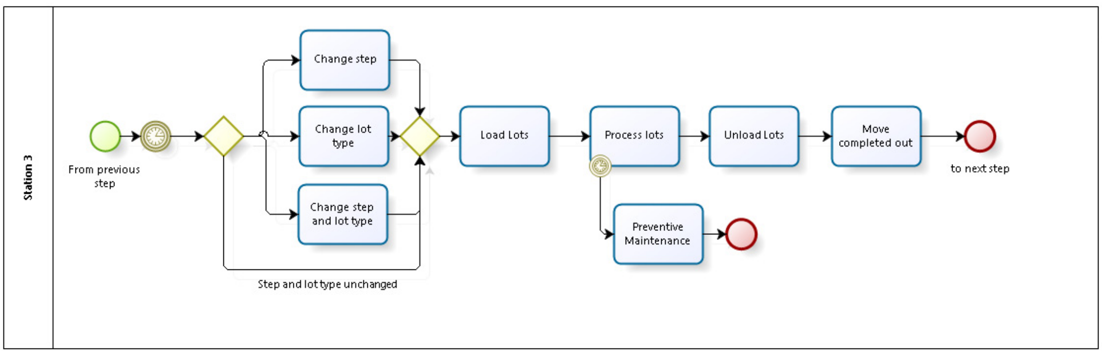

Figure 6-17: Station 3 modelled in Bizagi modeler.

Challenges and Opportunities

Although the top-level model (Figure 6-14) shows the re-entrant flow of lots, when simulating this model, the arrow going back to be processed in the next step is considered as a new part. The simulation engine doesn't have the ability to identify the step number or to recognize that the part is going through a new step; it is unable to identify routing decisions.

When modelling preventive maintenance for each station, interrupting border events that are attached to the machines in all stations, do not interrupt the process when running the model and it does not have any effect neither on the process nor the reported results. Bizagi doesn't support this feature and thus the process will never be interrupted for maintenance.

When using gateways in all models as mentioned in the previous chapter, the gateway should be used either to split a part flow or to merge parts' flows. Exclusive and inclusive gateways are the only two types of gateways that can be simulated. Using gateways to define a route for the part can be done based on a probability from 0 to 100% only. Hence, the diversion and conversion of paths or any form of routing, which is critical when modelling the Mini-fab, cannot be based on equations or calculations.

Assumptions and Modification of the Conceptual Model

In order to simulate the Mini-fab model using Bizagi, the conceptual model developed in 6.2.1 has to been simplified. This necessitates introducing several modifications to the model developed in Bizagi and additional assumptions to that already introduced to the Mini-fab model in the first place.

The first assumption that had a significant effect on the modelling of the Mini-fab model is that the re-entrant flow will not be considered. The model is represented using a set of consecutive activities instead, rather than three stations with five machines, and each station is visited twice by a lot with a total of 6 manufacturing steps. The model is represented as 6 consecutive manufacturing steps as shown in Figure 6-18. Where each step needs one machine that is modelled as a shared resource with another step.

Machines A, B, C, D, and E are defined as resources with a quantity of 1. Machine A and Machine B are shared between Step 1 and Step 5; Machine C or Machine D are shared between Step 2 and Step 4, and Machine E is shared between Step 3 and Step 6.

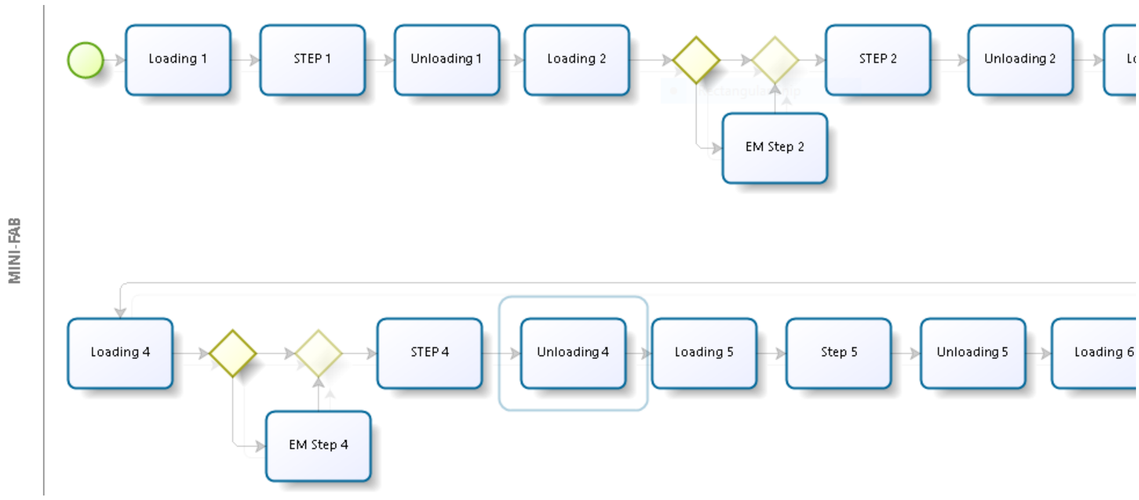

Figure 6-18: Mini-fab model in Bizagi process modeller.

Additional simplifications and assumptions made to the model to enable its representation using Bizagi modeler include modifying the processing times of all manufacturing steps to include the effect of preventive maintenance that applies to all machines, and to include the parallel batching effect on the capacity of Station 1.

Availability of all machines is determined first using Equation 6-1 based on the uptime and downtime defined for each machine in Section 6.1.2. Based on calculated availabilities, the effective processing time of each step is determined using Equation 6-2. As for the batching effect, since no specific notation to specify batching and un-batching exists in Bizagi; thus, the effective processing time of Step 1 and Step 5 is determined by dividing the batch processing time by the batch size. A summary of the calculations of the effective processing time defined for each manufacturing step is given in Table 6-4.

$$Availability = (1 - \frac{Down\ time}{Total\ time}) \times 100$$ Equation 6-1

$$Effective\ Time = \frac{Nominal\ Time}{Availability}$$ Equation 6-2

Table 6-4: Effective process time for each step.

Step	Downtime (min)	Total time (min)	Availability	Nominal Processing Time (min)	Effective Time (min)	Units	Batch Size	Time Defined in Bizagi (min/lot)
1	75	1,440	95%	225	237.4	min/batch	3	79.1
2	120	720	83%	30	36	min/lot	1	36
3	30	720	96%	55	57.4	min/lot	1	57.4
4	120	720	83%	50	60	min/lot	1	60
5	75	1,440	95%	255	269	min/batch	3	89.7
6	30	720	96%	10	10.4	min/lot	1	10.4

It is stated that the emergency maintenance (EM) applies to machines C and D only and happens randomly following a uniform distribution with a minimum of 1,440 minutes and a maximum of 4,560 minutes as shown in Figure 6-19. The repair, once started, takes between 6 and 8 hours, uniformly distributed. Gateways are used to direct the path for lots going through EM is based on percentages: 4% of the time, the lots will go to EM, where the processing time of this task is the meantime to repair of the machine. Bizagi can define repair times using distributions, although, there is no specific notation to represent emergency maintenance.

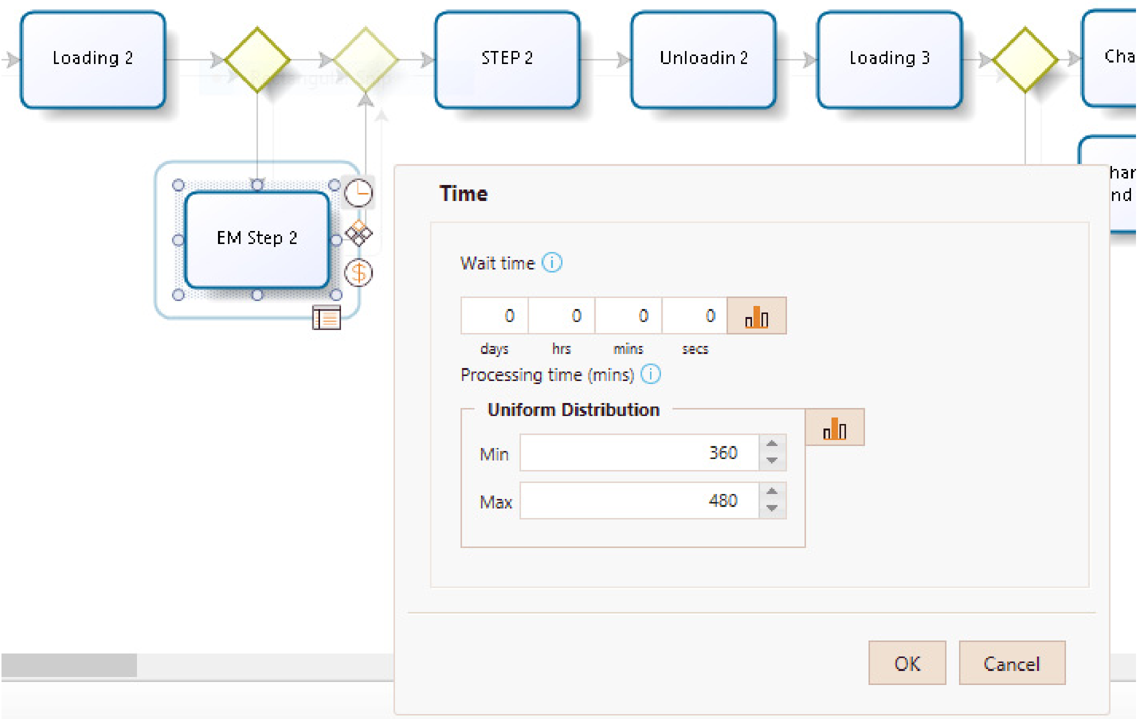

Figure 6-19: Snapshot of Emergency maintenance modelled in Bizagi.

Setup time: gateways were used to activate one path for the lots to go through the setup process, these paths are based on percentages and not an attribute. As shown in Figure 6-20, the probability of no setup cannot be added, and the probability of setup due to changing the lot type only cannot be added as well; this is because the model is a series of the sequential process, meaning that every time the step number is changed, thus, remaining probabilities are 50- 50, either step number only changed or step number and lot type.

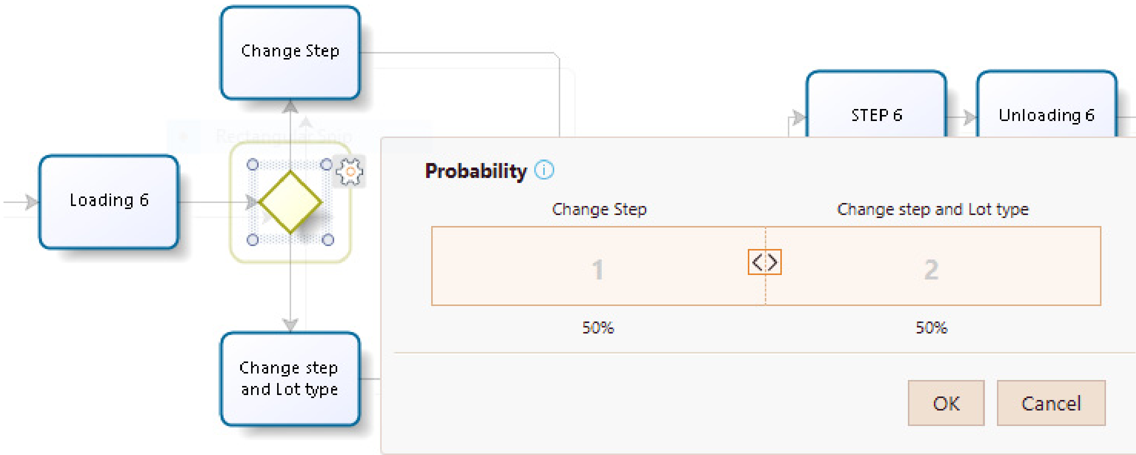

Figure 6-20: Snapshot of Station 3 setup modelled in Bizagi.

6.3.3 Modelling with MPMN Simulator

As mentioned previously, the Mini-fab model is of a relatively small size. Yet, it captures most of the challenges involved in scheduling re-entrant wafer fabrication facilities.

Preliminary Model with No Re-entrancy

To guarantee a fair comparison, the modelling of Mini-fab using MPMN Simulator first follows the same approach used in the Bizagi modeler; the model represents Mini-fab as six sequential processes and no re-entrancy is applied as shown in Figure 6-21. This means that the Mini-fab is modelled differently than the conceptual MPMN model due to the modifications and simplifications done to the model built in the Bizagi modeler. However, in addition to using the existing BPMN notations, the newly developed MPMN notations are used as well.

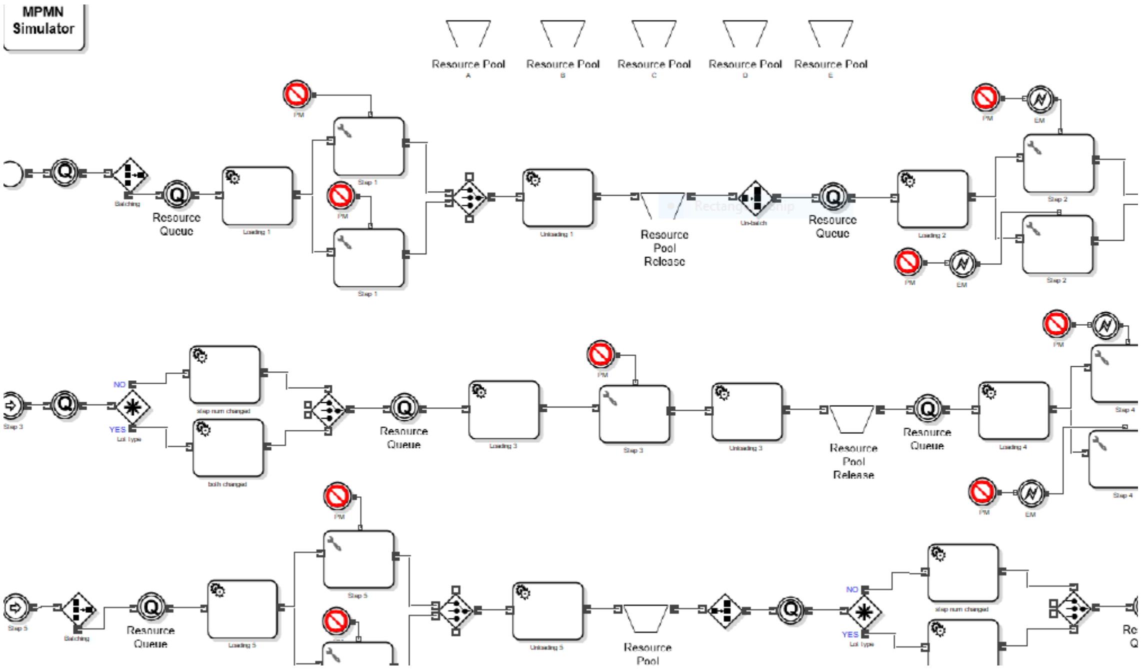

Figure 6-21: Snapshot of Mini-fab modelled using MPMN Simulator.

MPMN Simulator was able to define resources using three different MPMN notations. The first one is the resource pool queue, it releases the lot and/or batch when their required resources are allocated, hence, it is now shown where items wait to be processed. The second one is the Resource Pool which is used to hold a number of resources used in the simulation. It allocates a resource to an item and it works with the third notation which is the Resource Pool Release block to release the resource. For this behaviour, passing items trigger the immediate release of resources to return back to the pool specified in the block dialog.

Scheduled and emergency maintenance: both types of maintenance are simply dragged and dropped from the MPMN simulator library and connected to a task subjected to failure. Emergency shutdown event is the new notation to represent the unusual stoppage of the production line (EM), where the time between failure (TBF) and time to repair (TTR) can be set according to a specified distribution. Scheduled shutdown event: the new proposed event, added to represent the scheduled shutdown of the machines. It is used to represent PM, using this notation PM schedule and frequency are set.

Batching and un-batching operations are represented using batching and un-batching gateways, these gateways are used to batch and un-batch the specified quantities.

MPMN Simulator has the ability to define attributes for entities that helped in defining different lot types based on the probability distribution presented in section 6.1.2, which can be used later in defining changeover between different lots. Moreover, another attribute which is the step number, can be defined for incoming lots to make all routing decisions needed for defining re-entrancy batching and setup time. This can be done using the 'Conditional start event' which is used to set the step number, and the 'Data' artifacts increment the step number by one whenever a manufacturing step is completed. Based on the new step number, the lot is directed to the specific station based on the routing defined in the data section 6.1.2. Furthermore, using the 'Data' artifacts is used again to calculate the processing time for every step is and update it to match the step number.

This is in addition to the customized reports that MPMN Simulator is capable to create for every scenario and for any selected block. It has the same ExtendSim capabilities of reporting continuous reports during a run and summarized statistically after completing a run for every single replication and for all replications.

Material select and Material route: both gateways already exist in the MPMN set of notations, but as the model gets more complex and has several processes, the number of incoming and outgoing flow increases. The number of connectors was increased to match this.

Complex gateway: complex gateway is one of the BPMN gateways and is used for the most complex flows in the business process. Due to the complexity of the Mini-fab case study, modeling requires more specific notations to model combined flows. Complex gateway is added to MPMN set of notations, to model complex decisions but in a different manner than that of the BPMN. A complex gateway is used to display properties on items, and then passes the items through. The property value is shown in the dialog and output at the value output connector. It is used to model setup time where complex gateway displays the property on the passing item to identify whether the step number and lot type attribute values are changed or not.

Sub Process: in BPMN the sub-process helps with the expanding/collapsing view. A sub-process describes a detailed sequence, but it takes no more space in the diagram of the parent process than does a task. The sub-process is added to the MPMN set of notations for graphical representation and to mimic the MPMN static modelling using Visio, but it does not affect the flow of the process.

Mini-Fab Model with Re-entrancy

This section discusses the development of the MPMN Simulator model that considers re-entrancy, also includes calculations and equations to represents setup times. The complex case study of the Mini-fab process is remodelled again using the MPMN Simulator. This time MPMN simulator capabilities emerged to consider the re-entrant flow of the model, to include equations and calculations to manage setup times and batch and un-batching rules, and result in the model shown in Figure 6-22.

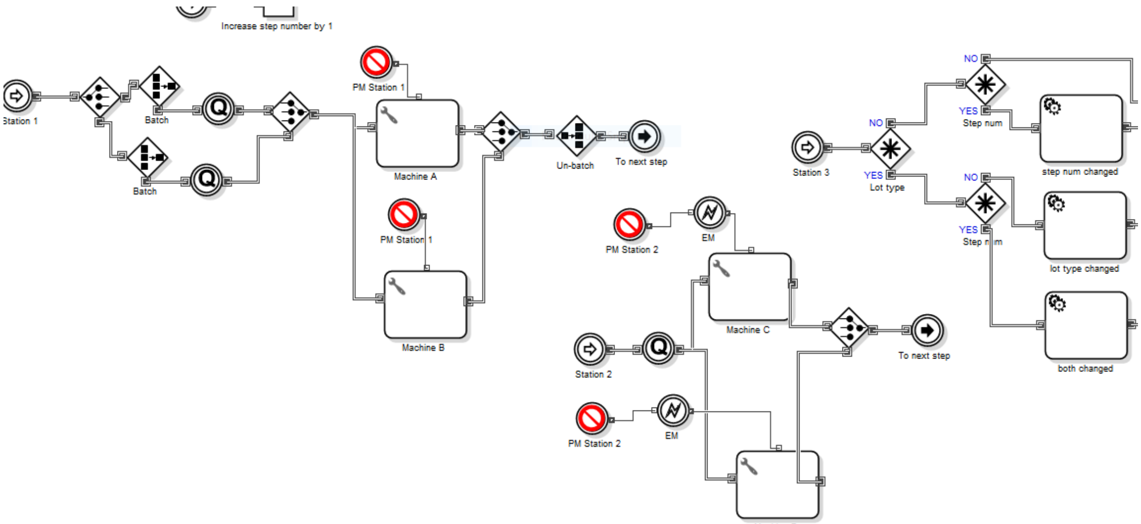

Figure 6-22: Snapshot of Mini-fab modelled in MPMN Simulator (reentrancy is considered).

In addition to the MPMN notations mentioned in the previous sub-section, a conditional start event: is used in the model to represent that receiving a lot will trigger the start of the process. Conditional intermediate event: as mentioned in Section 4.2.3, it is like a wait until a condition is set to items passing through the block. For example, in the Mini-fab model, a conditional intermediate event is used to set the step number for every lot passing through the different stations.

Catch and throw events which is the key events employed here to model the re-entrant flow of the Mini-fab.

Conceptual model: the model has the same look of the MPMN diagram, although it has been modelled on one page rather than being modelled using different pools and lanes.

Re-entrancy: is modelled in a facilitated way using conditional start and conditional intermediate events.

Batching and un-batching: are now shown and rules are applied, rather than using calculations to represent batching.

Failures: preventive and emergency maintenance are now visible using the aid of the error event, also machine status now can be identified whether it's up or down.

Setup: complex gateway simplifies setup time modelling, where lot type and step number affect changeover based on the decision rather than a percentage.

Warmup period: MPMN Simulator provides a facility for specifying the warmup period.

6.4 MODEL VERIFICATION AND VALIDATION

6.4.1 Model Verification

ExtendSim

As mention in the previous chapter, this section will focus on validating and verifying the ExtendSim model. Model verification was carried out using ExtendSim's reporting and animation capabilities of the different building blocks of the model to ensure that the model is working as intended.

The animation capabilities of ExtendSim were used to verify that the process flow matches the flow specified in 6.1.2. Also, the animation was used to show up and down signals on the activities to verify maintenance. Further details about duration and numbers of downtimes can be retrieved using ExtendSim's reporting capabilities. Which further verifies the calculated availability.

6.4.2 Model Validation

Face validation is firstly used to confirm that constructed model appeared to be reasonable on its face and lots are being processed in the correct sequence flow, and each station behaves as intended. This was approved by consulting one of the professors who had studied and researched this case study extensively.

Validation of model assumptions falls into two general classes: structural assumptions and data assumptions.

- The structural assumptions involve validating the number of stations with their machines. In Station One, there are two machines A and B. In Station Two, there are two machines C and D. In Station Three, there is only one machine, machine E.

- Finally, input/output transformation validation was done by comparing the output of the constructed model with respect to utilisations of the stations to those published at the Mini-fab website mentioned before. As shown in Table 6-5, the presented data about station utilization in the capacity analysis "calculated U" of the Mini-fab model is compared to the station utilization reported from the developed ExtendSim model "Reported U".

Table 6-5: Stations Utilization.

Station	Calculated U	Reported U
1	87.9%	85%
2	77.8%	80%
3	91.3%	90%

6.5 EXPERIMENTATIONS AND ANALYSIS

The following sections will discuss in detail the sets of experiments used to compare the output of the different simulation tools. As shown in the previous section, four simulation models are developed to show the capabilities of the three modelling tools. The first simulation model is developed using the ExtendSim simulation environment and making use of the full capabilities of a DES to develop the base model. The second simulation model is developed using Bizagi. While the third and fourth models are developed using the MPMN simulator to develop the Mini-fab model with re-entrancy and without re-entrancy.

6.5.1 Performance Measures

The performance measures that will be evaluated are:

- *Cycle time*, which is the time spent in minutes to produce one lot starting from entering the fab, to begin with, Step 1 and ending with leaving the fab after finishing Step 6. This is reported as the mean of the cycle time reported for each lot over a simulation run and averaged again based on the outcomes of a number of replications $(\overline{\overline{CT}})$.
- Throughput rate, which is the number of finished lots per week. This is reported as the average weekly throughput rate in a single simulation run and averaged again based on the means reported from a number of replications $\overline{\overline{(TH)}}$.
- *Utilisation* of the resources (stations), which is the percentage of time the resources are busy. This is reported as the mean of the weekly instantaneous monitored utilisation over the whole run averaged based on the outcomes of the number of replications $(\overline{\overline{U}})$.

6.5.2 Simulation Parameters

Setting The Length of Simulation Run

The Mini-fab case study belongs to non-terminating simulations, then there is no definitive way of picking the simulation run time. Thus, it is decided to set the simulation runtime to 2 years; 104 weeks (1,048,320 minutes).

Determining the Warmup Period

To determine the warmup period the graphical method was used. The plot of both average cycle time and average throughput rate becomes almost smooth after 70,000 minutes, as shown in Figure 6-23. This number is approximated to the nearest week;

thus, the warmup period is decided to be 7 weeks, which is equivalent to 70,560 minutes.

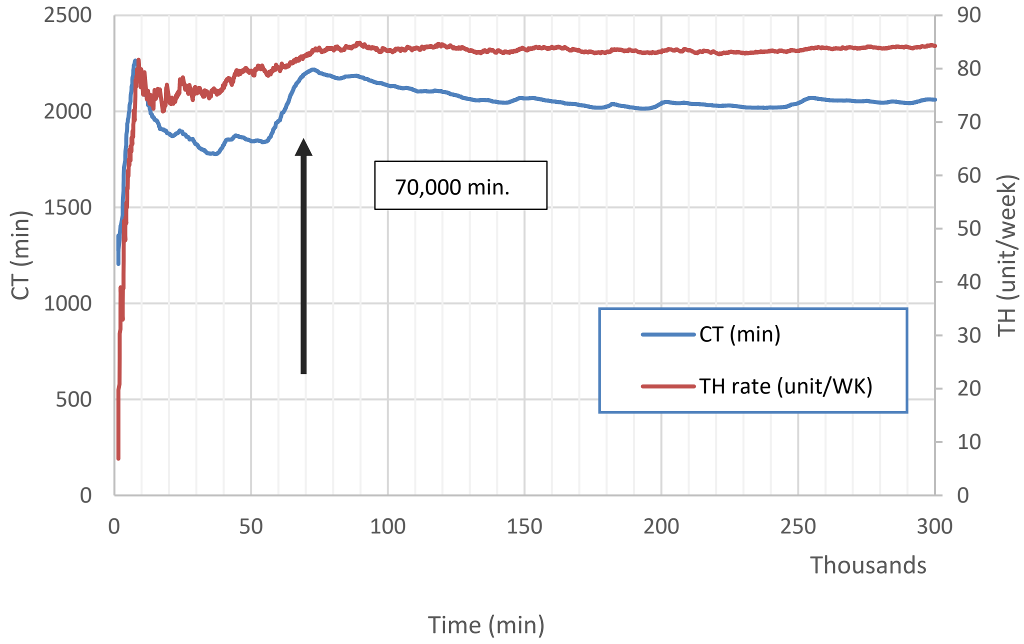

Figure 6-23: Warmup period.

Setting the Number of Replications

As mentioned before in Chapter 5, the confidence interval method is used to determine the number of replications needed for each of the modelling tools. The number of replications is selected at the point where the interval reaches and remains below the desired level of deviation.

Cycle time was used to determine the number of replications needed for each run. Using the simulation results of the mean CT for 20 different replications, the cumulative mean of the average cycle time is calculated. Based on the cumulative mean, standard deviation, and the number of replications; the lower and upper confidence intervals are calculated and reported in Table 6-6. Based on the percentage deviation and the plot of the cycle time intervals shown in Figure 6-24, 20 replications would suffice.

Table 6-6: Number of replications and confidence interval for ExtendSim model.

Rep.	CT (min)	Cum. mean average	Standard deviation	Lower Confidence interval	Upper Confidence interval	% deviation
1	2,147	2,137	n/a	n/a	n/a	n/a
2	2,118	2,086	73	1,432	2,740	31.34%
3	2,052	2,080	52	1,950	2,210	6.27%
4	2,232	2,106	68	1,998	2,214	5.11%
5	2,130	2,080	83	1,976	2,183	4.97%
6	1,917	2,076	75	1,998	2,155	3.79%
7	2,163	2,075	68	2,012	2,139	3.05%
8	1,906	2,069	66	2,014	2,124	2.66%
9	2,044	2,076	65	2,026	2,125	2.40%
10	1,972	2,068	66	2,021	2,115	2.28%
11	2,157	2,058	70	2,011	2,105	2.29%
12	2,001	2,057	67	2,014	2,099	2.07%
13	1,885	2,063	68	2,022	2,105	2.00%
14	2,085	2,059	68	2,020	2,098	1.90%
15	2,146	2,059	65	2,023	2,095	1.76%
16	2,071	2,072	81	2,028	2,115	2.08%
17	1,997	2,075	80	2,034	2,116	1.97%
18	2,268	2,070	81	2,029	2,110	1.94%
19	2,027	2,063	83	2,023	2,103	1.95%
20	1,999	2,055	89	2,014	2,097	2.02%

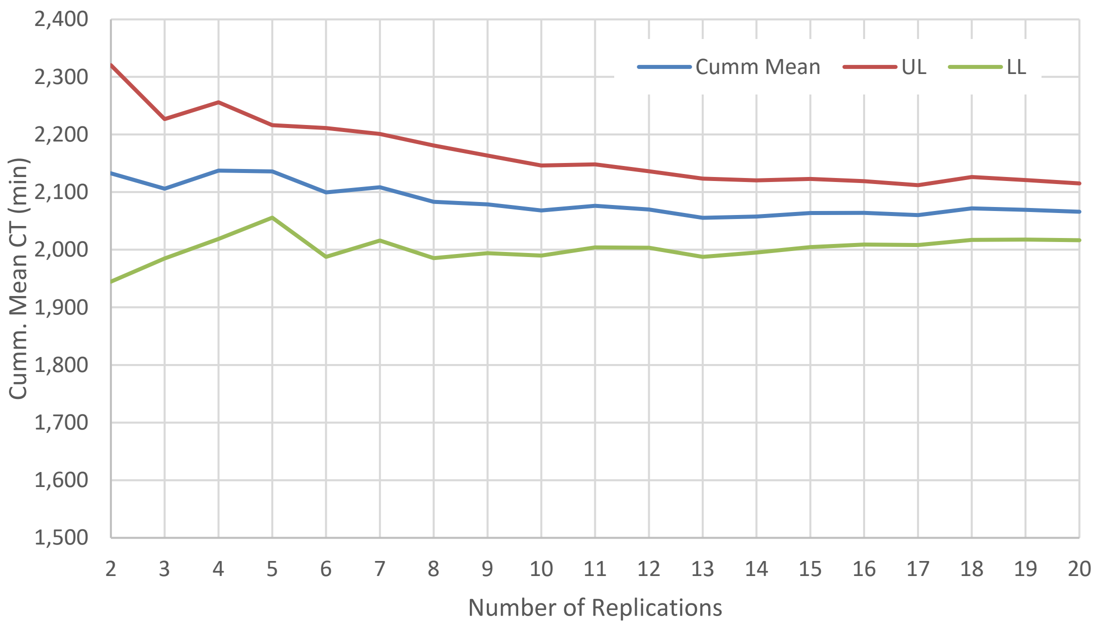

Figure 6-24: Plot of Cumulative Mean and 95% Confidence Intervals.

In conclusion, the simulation parameters used for all simulation models of the Mini-fab are a simulation run time of 104 weeks (2 years) with a warmup period of 7 weeks and

20 replications. Simulation parameters are also counted in the new Mini-fab model; warmup period and the number of replications are also based on ExtendSim simulation parameters a warmup period of 7 weeks and 20 replications are needed for this model, also, the run length remains the same 104 weeks (1,048,320 minutes).

6.5.3 Comparison of Results

As shown in Table 6-7, Mean CT, TH, and station utilization are based on the outcomes of 20 replications for every mean time between arrivals tested for the four models using the three tools.

Table 6-7: Results of the three modelling tools.

MTBA 120 min		ExtendSim (Re-entrant Flow)	Bizagi (No Re-entrancy)	MPMN (No Re-entrancy)	MPMN (Re-entrant Flow)
Mean of means CT (min)		2,066	2,113	2,044	2,115
Mean TH (lots/week)		84	84	84	84
Utilization	Station 1	87%	83%	84%	85%
	Station 2	78%	77%	80%	79%
	Station 3	89%	88%	89%	89%

The output results reported from Bizagi and MPMN simulator models are close to the output of the ExtendSim model. The mean of means CT are within the same range, the four models reported the same TH, and they all report that machine three is the bottleneck and that what matches the output results of the ExtendSim model.

The presented results are the output of two different models. The issue in ExtendSim is that two models must be exactly the same graphically for Common Random Number (CRN) to work. The removal or exchange of a block will alter the running of the simulation, breaking CRN settings. The baseline ExtendSim model is not replicated exactly by the MPMN model but has a completely different structure. Comparative running of the two representations will therefore contain different random seeds and so cannot be expected to deliver identical results. The differences seen here are small enough to be considered as representative of the same underlying results.

6.5.4 Confirmation of Results

Paired t approach is used to compare the output of the three tools, results are generated, and then a judgment can be made as to whether the difference in the results is significant. In this work the spreadsheets available on the website (www.wileyeurope.com/go/robinson) are used to compare the three modelling tools; the comparison is based on the CT.

Using the mentioned spreadsheets, the common random number is ignored, as the three models use two different simulation software and different number generators will be available for each tool. Paired t comparison is used to compare the mean of CT as presented in Table 6-8, Table 6-9, and Table 6-10, this is to determine if the mean CT reported from the different tools are significantly different or not; where:

- $S1$ = results from Scenario 1 for each of the 20 replications (it will represent ExtendSim output results).
- $S2$ = results from Scenario 2 for each of the 20 replications (it will represent Bizagi modeler results in Table 6-8, MPMN Simulator with no re-entrancy results in Table 6-9, and MPMN Simulator with re-entrancy results in Table 6-10).
- Difference = difference between Scenario 1 results and Scenario 2 results for each replication *j*.
- Cum. mean difference= cumulative mean.
- *SD* = is the standard deviation of the differences.
- Lower and Upper Interval = are the developed confidence intervals for the difference between the results of the two scenarios calculated using the *t*-distribution at 95% significance level.

The conclusion of the comparison is based on the analysis of the upper and lower intervals reported in the last row of each table (highlighted row). The outcome of the analysis is either Scenario 1 is greater than Scenario 2, or Scenario 2 is greater than Scenario 1, or no significant difference between the two scenarios.

The comparison showed that, since for the 3 comparisons conducted, the lower interval is negative and the upper interval is positive, then there is no significant difference between S1 and S2. Hence, there is no difference between the mean cycle times reported by the ExtendSim model and the other three models.

Table 6-8: Paired-t Confidence Interval for ExtendSim and Bizagi.

Rep.	S1 EXTENDSIM	S2 BIZAGI	Difference	Cum. mean difference	SD	Lower interval	Upper interval
1	2,147	1,938	209.2	209.2	n/a	n/a	n/a
2	2,118	2,169	-51.6	78.8	184.4	-1577.9	1735.6
3	2,052	2,232	-179.8	-7.4	198.2	-499.9	485.1
4	2,232	1,887	344.7	80.6	239.1	-299.9	461.1
5	2,130	2,180	-49.3	54.6	215.1	-212.4	321.7
6	1,917	1,840	76.6	58.3	192.6	-143.8	260.4
7	2,163	2,440	-277.7	10.3	216.9	-190.3	210.9
8	1,906	1,862	43.9	14.5	201.2	-153.7	182.6
9	2,044	2,201	-157.3	-4.6	196.7	-155.8	146.6
10	1,972	2,040	-68.3	-11.0	186.5	-144.4	122.5
11	2,157	2,282	-125.5	-21.4	180.3	-142.5	99.7
12	2,001	2,014	-13.0	-20.7	171.9	-129.9	88.5
13	1,885	2,137	-252.0	-38.5	176.7	-145.2	68.3
14	2,085	2,394	-309.1	-57.8	184.5	-164.3	48.7
15	2,146	2,055	90.7	-47.9	181.9	-148.6	52.8
16	2,071	2,068	3.3	-44.7	176.2	-138.6	49.2
17	1,997	2,243	-245.7	-56.5	177.4	-147.7	34.7
18	2,268	2,170	98.3	-47.9	175.9	-135.4	39.6
19	2,027	1,988	39.5	-43.3	172.2	-126.3	39.7
20	**1,999**	**2,124**	**-125.0**	**-47.4**	**168.6**	**-126.3**	**31.5**

Table 6-9: Paired-t Confidence Interval for ExtendSim and MPMN with no re-entrancy.

Rep.	S1 EXTENDSIM	S2 MPMN no reentrancy	Difference	Cum. mean difference	SD	Lower interval	Upper interval
1	2,147	1,920	227.2	227.2	n/a	n/a	n/a
2	2,118	2,076	41.4	134.3	131.4	-1046.4	1315.0
3	2,052	1,989	63.4	110.7	101.5	-141.6	362.9
4	2,232	1,925	307.4	159.8	128.6	-44.8	364.5
5	2,130	2,107	23.9	132.6	126.9	-24.9	290.2
6	1,917	2,151	-234.4	71.5	188.0	-125.8	268.8
7	2,163	2,171	-8.8	60.0	174.3	-101.2	221.2
8	1,906	1,940	-33.7	48.3	164.7	-89.4	186.0
9	2,044	1,960	84.0	52.3	154.5	-66.5	171.0
10	1,972	2,022	-50.1	42.0	149.3	-64.8	148.8
11	2,157	1,870	286.8	64.3	159.7	-43.0	171.5
12	2,001	1,989	12.4	59.9	153.0	-37.2	157.1
13	1,885	1,922	-37.5	52.5	148.9	-37.5	142.5
14	2,085	1,938	147.1	59.2	145.3	-24.7	143.1
15	2,146	1,901	244.8	71.6	148.0	-10.4	153.5
16	2,071	2,138	-66.9	62.9	147.1	-15.5	141.3
17	1,997	2,137	-140.1	51.0	150.7	-26.5	128.5
18	2,268	2,123	144.9	56.2	147.9	-17.3	129.7
19	2,027	1,877	150.0	61.1	145.3	-8.9	131.2
20	**1,999**	**1,998**	**1.2**	**58.1**	**142.1**	**-8.3**	**124.6**

Table 6-10: Paired-t Confidence Interval for ExtendSim and MPMN with re-entrancy.

Rep.	S1 EXTENDSIM	S2 MPMN reentrancy	Difference	Cum. mean difference	SD	Lower interval	Upper interval
1	2,147	1,986	160.9	160.9	n/a	n/a	n/a
2	2,118	2,209	-90.8	35.0	178.0	-1564.5	1634.5
3	2,052	2,134	-82.0	-4.0	142.9	-358.9	350.9
4	2,232	2,145	86.7	18.7	125.2	-180.4	217.8
5	2,130	2,324	-193.8	-23.8	144.1	-202.8	155.2
6	1,917	2,193	-276.5	-65.9	165.1	-239.2	107.4
7	2,163	1,927	235.5	-22.8	188.9	-197.6	151.9
8	1,906	2,041	-134.7	-36.8	179.3	-186.8	113.1
9	2,044	2,281	-237.0	-59.1	180.5	-197.8	79.7
10	1,972	2,372	-400.8	-93.2	201.6	-237.5	51.0
11	2,157	2,036	120.1	-73.8	201.8	-209.4	61.7
12	2,001	1,941	60.3	-62.7	196.3	-187.4	62.0
13	1,885	2,012	-127.9	-67.7	188.8	-181.8	46.4
14	2,085	2,124	-39.1	-65.6	181.5	-170.5	39.2
15	2,146	2,033	113.4	-53.7	180.9	-153.9	46.5
16	2,071	2,013	58.0	-46.7	177.0	-141.0	47.6
17	1,997	2,254	-256.3	-59.1	178.8	-151.0	32.9
18	2,268	2,008	260.3	-41.3	189.1	-135.3	52.7
19	2,027	2,155	-127.4	-45.8	184.8	-134.9	43.2
20	**1,999**	**2,117**	**-117.4**	**-49.4**	**180.6**	**-133.9**	**35.1**

6.5.5 Mean Time Between Arrivals

Several simulations are carried out with different mean time between arrivals (MTBA) distributed exponentially, as described before, a lot is introduced to the Mini-fab with a mean of an exponential distribution of 120 minutes. Starting with a mean of 120 minutes and decreasing 1 minute for every simulation run until reaching 105 minutes for all the models. Three performance measures are evaluated for the three modelling tools; machine E (bottleneck) utilization, mean of means CT (minutes), and TH rate (lots /week). Machine utilization, the mean for the TH, and CT averaged based on the outcomes of the 20 replications for every mean time between arrivals tested for all the models as presented below.

All models reported almost the same utilization for station E for all the simulation runs as shown in Figure 6-25. As for the Cycle time, it was expected to increase rapidly as the MTBA decreases, but as shown in Figure 6-26, the Bizagi model CT at 105 MTBA reported a CT which is almost more than double of the CT for the other models. The throughput rate for the ExtendSim model and MPMN Simulator models are almost the same as shown in Figure 6-27, but for the Bizagi model, the TH rate doesn't reach 93 lots per week in all the runs, where all the models exceed this number and it starts decreasing at 107.

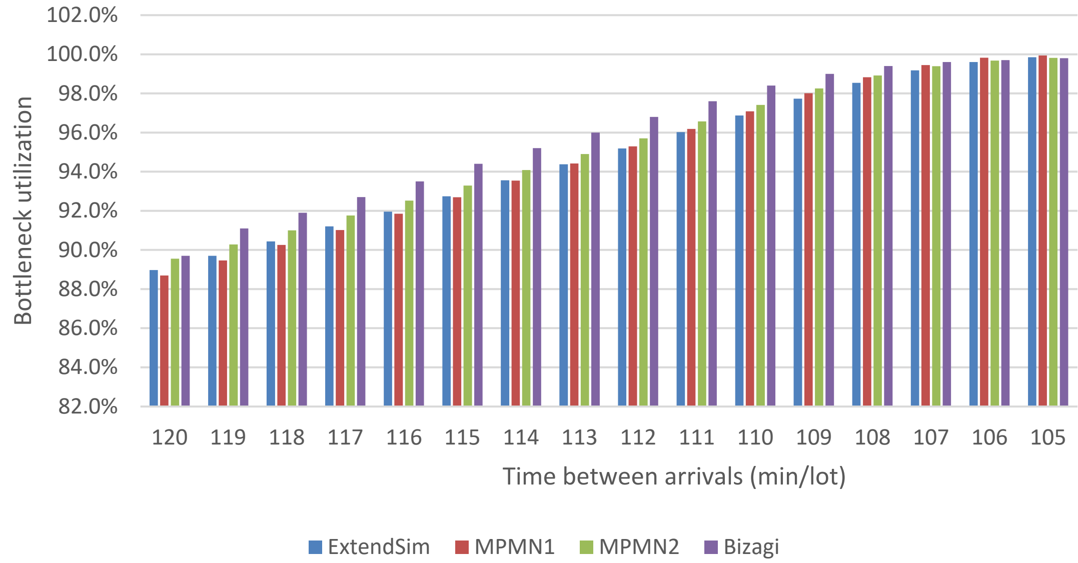

Figure 6-25: Machine E utilization using different MTBA.

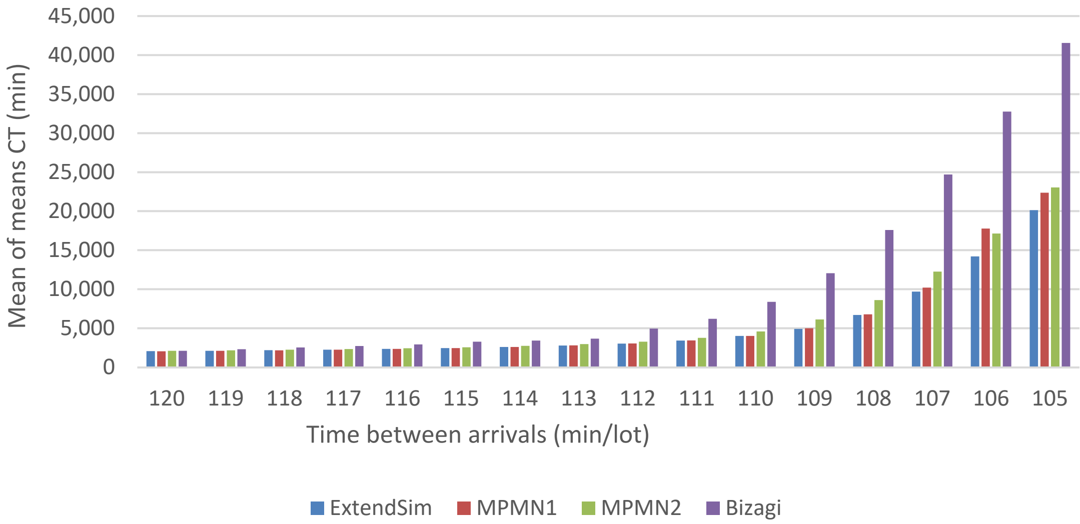

Figure 6-26: Mean CT for different MTBA for the three modelling tools.

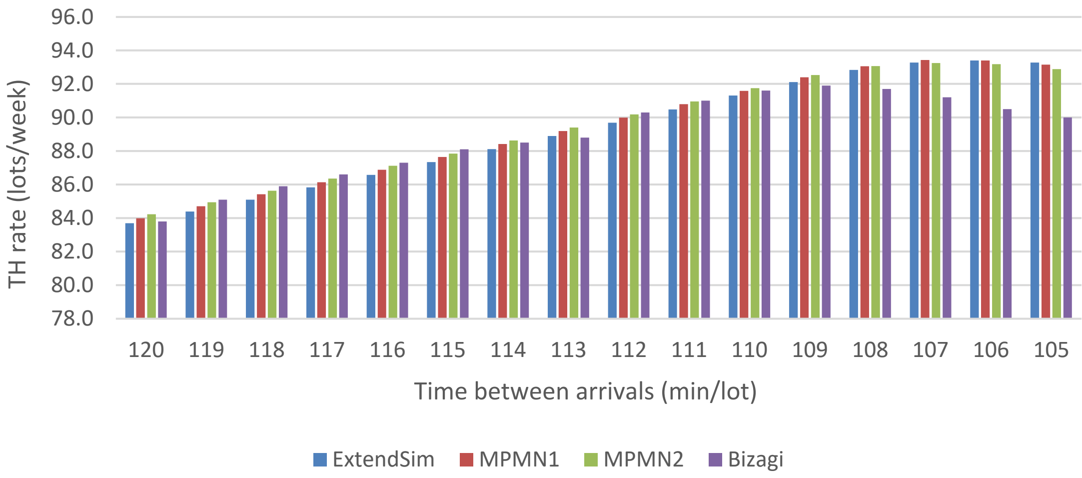

Figure 6-27: Throughput rate for different MTBA for the three modelling tools.

6.5.6 Analysis of Results

MPMN Simulator models match the ExtendSim model at all the MTBA level utilization of resources. As mentioned in 5.5.4, MPMN Simulator has the same ExtendSim simulation capabilities to model the complexity of the Mini-fab model, where fewer assumptions and calculations are used to simulate the model.

Although the paired *t* approach showed no differences at all between all the models, using different MTBA to compare the output of the three tools showed significant differences between Bizagi and the other tools. Clear difference starting from TBA 108, where Bizagi reached the maximum capacity as the number of units increases. At TBA 111 the resource utilization increases Figure 6-25, this increases the waiting time in the queue in the bottleneck station, which is reflected in the higher CT which almost double of the other models CT Figure 6-26, and lower TH Figure 6-27. It is worth mentioning that the use of simulation in any given case study is to experiment with the system under study by using different scenarios, when this applied to Bizagi it failed to produce reliable data, thus, the output of Bizagi is not enough for decision making. This is to confirm that MPMN Simulator is performing better than Bizagi and it delivers closer results to ExtendSim.

6.5.7 Discussion

The objective of selecting the Mini-fab case study is to evaluate and address the opportunities and limitations of modelling a more complex case study using the developed notations and the MPMN simulator compared to the traditional BPMN and Bizagi modeller. Although not shown in the thesis; numerous versions of BPMN models were developed before arriving at the final versions reported in this work.

Modelling a more complex case study confirms that BPMN and BPSim 2.0 suffer from limited technical capabilities to represent some important features of complex flows. It has insufficient notations to represent some important yet basic manufacturing aspects. Also, Bizagi failed to represent some manufacturing aspects in a straightforward manner. Despite enhancing Bizagi with new capabilities to model complex manufacturing systems through calculations, assumptions, and simplifications, these kinds of calculations to simplify the model cannot be executed by any user; an expert who has the background to compute variability analysis would be needed to produce proper results. Moreover, Bizagi is supposed to have validation capabilities, but it doesn't validate the logic used in the models; it for example checks that gateways are only used for merging or diverging and not for both at the same time, or it checks if any of the arrows are not connected properly to the notations.

Modelling using MPMN and MPMN simulator reports better statistical and numerical results. Data such as capacities, arrival rates, activity times, and time between failures were able to be defined using the new approach. MPMN Simulator is capable of factoring Setup time, determine routing decisions, calculate batching and un-batching.

7. DISCUSSION

The demands of modern manufacturing require data-driven decision making. However, a manufacturing system is rarely a stable fixed-configuration and so modelling is required to support the prediction of the system response to changes in inputs. Developing a representative model is time-consuming and expensive, so improved methods to provide useful models are required.

Rather than invent a whole new approach, the application of solutions shown to be successful in other domains seems sensible. To this end, BPMN provides a solid base on which to build. Aimed mainly at business processes this work first evaluates its direct application to manufacturing, identifies the shortcomings, builds on BPMN to extend it in meaningful ways to capture the essential elements for manufacturing. Of course, a notation by itself only describes the system and so simulation must be added to enable the generation of decision supporting data. For the business community, a tool like Bizagi has coupled BPMN with BPSim 2.0 to create such data. Here the representation of manufacturing systems using Bizagi is investigated and evaluated to determine its strengths and weaknesses. Finally, in order to create a tool to allow the rapid use of the expanded notation to rapidly represent a manufacturing system and enable enhanced simulation of the system a library for a full DES package has been developed.

In this chapter, the improvements achieved are discussed based on the assessment of the developed approach on two manufacturing systems. Many barriers to modelling are discussed and new capabilities that are offered to simulate models are addressed. MPMN Simulator was a result of BPMN evolvement through three development stages which will also be discussed in this chapter. This work also demonstrated several limitations and is presented at the end of this chapter.

7.1 INTRODUCTION

BPMN is considered the leading standard to model business processes since its main purpose to be understandable by all business users to improve efficiency and gain a competitive advantage. Taking advantage of such a standard and bringing it to model manufacturing processes was the aim of this work; this work aims to develop an understandable set of notations that can model processes specific to the manufacturing domain, and these notations should also be able to easily produce quantitative data that are useful in describing manufacturing process behaviour.

In this work, BPMN was introduced to the manufacturing domain to allow all stakeholders to take advantage of the simplicity of this language in gaining full understandings of manufacturing processes through simple representations of the process models. However, BPMN was unable to represent many basic aspects of the manufacturing system and incapable to represent complex flows in this study.

The BPMN 2.0 standard is a detailed one, presented in a more than 500-page document [109], including an explanation of each notation and provides examples of BPMN models; yet the description of several notations is ambiguous, the use of some notations is not clear, identifying the model suitable notation to use wasn't always easy, and more importantly, the standard still doesn't present clear instructions of how to verify and validate the developed static models. One ambiguous BPMN construct that clearly comes forward from the literature is the Inclusive OR-join gateway [129]. However, this provides the modeller flexibility in developing the models to fit the situation being modelled, yet, one can face difficulties in judging whether the models developed are correct or not.

Using BPMN to build the conceptual model was one of the most challenging parts of modelling the manufacturing systems studied, hence, it required considerable time and effort. Although BPMN is capable of illustrating some of the faced challenges, it cannot represent some of the manufacturing aspects from the modeler perspective, as the standard already lacks them. It might have the capabilities to represent all the difficulties faced, but it will lose the concept of the model; to model manufacturing system, BPMN needs to be extended or combined with another language or tool to be able to embrace various manufacturing aspects and depicts manufacturing complexities.

7.1.1 Extending BPMN For Manufacturing Systems Modelling

The first objective of this work is to assess BPMN capabilities to model manufacturing systems, the highly adopted and the worldwide accepted modelling language. BPMN has the capability of modelling the manufacturing process but it cannot reflect the real dynamics of the system; it was difficult to use business-related notations to model complex manufacturing processes. Many basic aspects of manufacturing modelling are illustrated using several notations to make the model understandable, for example, un-batch and setup which are demonstrated using the BPMN diagram presented in 5.2.1 on page 59, and batching which is represented using a group of notations rather than one customized notation in the BPMN diagram presented in 6.2.1 on page 87. To better represent manufacturing processes, BPMN notations were completely analyzed, to understand each notation, and this analysis was the guide to develop the new MPMN set of notations with all the missing elements needed to model manufacturing processes. This was done by either extending or modifying existing BPMN notations. These new and or modified notations have the same look-and-feel of BPMN. The challenge here was to apply a notation that is a business process base to design, model, and experiment with different complexities of manufacturing systems. However, extending BPMN notations and introducing a new MPMN set of notations, enhances BPMN notations with more elements that can better describe the system, and it confirms that BPMN can describe the information-intensive activities which support the manufacturing process and can describe repetitive manufacturing processes with few variations [9].

7.1.2 New Simulation Capabilities

Moving from static representation to capturing the dynamics of the system, BPMN and BPSim 2.0 were integrated to model case studies. Bizagi was selected as a model simulator to provide numerical models to test the appropriateness of the prevailing concept.

It was possible to model some of the challenges faced using Bizagi. However, to be able to provide proper data, modelling manufacturing systems using Bizagi encompasses assumptions and simplifications. In both case studies, BPMN conceptual models can be rapidly 'translated' to a DES with the same visual structure using Bizagi. However, as mentioned in the previous section BPMN models still lack some important flow features and were inadequate to represent some detailed data, which Bizagi proved to lack. Despite the contribution made to allow the Bizagi model the challenges faced in the conceptual model stages, like for example the possibility to factor setup and model batching and un-batching. Still, the capability of Bizagi is limited by the limitations of BPMN. Many of these issues were found to be resolved through the use of MPMN.

7.1.3 Developed Approach to Build Simulation Models

Despite enhancing Bizagi with new simulation capabilities, Bizagi showed a lot of shortcomings in representing manufacturing systems, and many challenges were faced to model basic manufacturing processes. Thus, this led to extending MPMN and integrating it with a full DES software to dynamically capture the state of a manufacturing system. This simulator is proposed to create manufacturing models and help to promote the use of simulation tools, where users need to have some domain knowledge and only require a minimal introduction to simulation, with no full understanding of the tool. It has the advantage of allowing the non- simulation expert to build and simulate high fidelity manufacturing models with easy and understandable notations. The new MPMN notations must also be realised in simulation form as well as symbols. These notations were built in a drag and drop library to develop a tool that has the capacity to model and simulate manufacturing systems using simple and understandable notations. The domain expert will be able to use MPMN to define a DES model and execute it under full DES control to enable a full set of analysis results comparable with those provided by a simulation expert, on the potential to provide better data to decision-makers.

7.2 Improved Notation Capabilities

Introducing BPMN to model a simple manufacturing case study showed that it lacks the notations that represent basic manufacturing processes like cutting, batching, setups, and rework. Also, to cover more of manufacturing processes modelling requirements

BPMN was challenged to address particular manufacturing system issues like breakdowns, changeover time, and batching using a more complex case study that is presented in Chapter 6. In both case studies, notations use was optimized to represent different manufacturing processes, several iterations, and several versions were developed to build the conceptual models. That is because it was not clear how to model a manufacturing system using BPMN. Actually, there is no clear semantics that shows how to use BPMN to model a validated business process.

Based on the BPMN application to model these two manufacturing systems, new tailored notations were needed to provide a true representation of manufacturing systems processes. Extending BPMN by introducing MPMN which is a key contribution of this work provides a true representation to the manufacturing systems. The MPMN set of notations was proposed after conducting a complete analysis of BPMN notations. This analysis was crucial, to clearly understand the standard and to find the appropriate modelling notations to be used for building models conveying the right meaning of the process flow. Despite the significant extensions to BPMN, only Zor et al [10] extended BPMN by developing new notations that were customized to include some missing aspects required to model manufacturing systems. Yet, as mentioned in 4.2, on page 39 their extension efforts still lack some important aspects to model manufacturing system. Thus, building on Zor's work, Manufacturing Process Model and Notation (MPMN) has been developed to overcome the shortfalls of BPMN for describing important aspects of manufacturing systems. It involves modifications to the existing BPMN notations and the development of new notations that are able to model manufacturing processes using core modelling notations that better represent the system. MPMN enhances important yet basic manufacturing aspects like resources, queues, even different types of flows that were difficult to represent using the original BPMN. Nevertheless, this extension follows the rules presented in 4.2 to extend BPMN; MPMN does not contradict the semantics of any element that is defined in the BPMN specification. Moreover, MPMN graphical notations are easy to understand by any viewer of the process diagram as they have the "look-and-feel" of BPMN.

As a part of MPMN development facilitated model representation is provided to overcome the barriers faced; in both case studies MPMN provided a quick and easy-to-use process mapping tool producing straightforward visualizations that the stakeholders could understand intuitively. It proposed a more expressive set of notations, common aspects like queue is now shown, complexities like routing decision, batching and un-batching can be captured. It is obvious that the diagram Figure 5-3, on page 63 differentiates between different types of flows and explicitly shows the flow of U-bolt throughout the production process. Also, the use of workstation notation illustrates a more organized diagram, while, the presence of queues which is a basic, yet common notations in the manufacturing system can now be clearly demonstrated. The development of customized notation to represent batching and un-batching in Figure 6-10, on page 96, predictive and preventive maintenance Figure 6-11, on page 97, and the illustration of the complex gateway which is proposed to factor setup Figure 6-12, on page 98, make the diagram easy to read and the conceptual model is turned to be a true representation of the system.

7.3 Improvements to Simulation of MPMN Notation

Although the conceptual modelling stage was achieved in facilitated mode using new proposed MPMN notations, a barrier to continuing this through model coding was encountered in building the DES model structure in Bizagi. This arose from the complexity of the models' requirements to represent the manufacturing systems under study being greater than the current capability of the modelling standards (BPSim 2.0) and software (Bizagi). Modelling manufacturing processes using Bizagi contributes to confirming that BPMN and BPSim 2.0, are insufficient to model and simulate manufacturing systems [4][85]. For example, to ensure that the models developed in Bizagi were good enough to represent the system, their performance was validated by comparing it to the results of the ExtendSim models. Although Bizagi has validation capabilities it validates the models by ensuring that the process passes through all the sequence flows, and behaves as expected, but it doesn't validate the notations or the logic used in the diagrams, and it only ensures that the model is an error-free model.

Moreover, when reporting results, only results on each task after completion of simulation can be viewed. However, continuous reporting of results from any activity while running can't be recorded. Upon completion of a simulation run, Bizagi generates the “output sheet” which is basically an un-customisable report of the simulation output; however, it can be exported to MS Excel. Results are reported for the entire model, each task, each event, each gateway, and each resource. Results include the number of instances started and the number of instances completed by the model and all tasks, events, and gateways. Also, the minimum, maximum, average, and total time of processing for the entire model and for each task. If a resource is defined for a specific task, the minimum, maximum, average, standard deviation, and total waiting time for that resource is reported as well.

BPSim 2.0 and BPMN (as implemented in Bizagi) are insufficient to model complex systems. It was possible to model some of the challenges faced in the U-bolt case study, it is given in the data collection section 5.1.2 that a batch is received every three hours, in Bizagi due to the stochastic arrival rate of units, calculation replaced simulation, and the mean time between arrivals is assumed to be 0.54 min/bolts.

7.4 MPMN Library in ExtendSim

There are well-known differences in the way experts and novices create and use models. However, the MPMN library is suitable for both; users that are familiar with the system under analysis can create simulation models using simple and understandable notations; notations are built in a drag and drop library and the approach is based on component-based modelling, where model components can be linked together to form a model of a manufacturing system. This proved MPMN Simulator’s capability to keep conceptual models intact and to embrace notations that can actually represent the manufacturing system. Moreover, the library that was built in the ExtendSim environment has the ability to set simulation parameters and measure model performances. The challenge in using software like the ExtendSim was manifested in the time and effort to understand the software, each block developed and changed to the recognizable MPMN notations, as well as developing the block behaviour and characteristics so it could simulate what it looked like.

MPMN Simulator is able to define resources, model scheduled and emergency maintenance, batch, and un-batch items as shown in Figure 6-21 on page 107. MPMN

Simulator has the ability to define attributes, which helped in defining different lot types based on the probability distribution presented in section 6.1.2, and to define an attribute for step number to make all routing decisions needed for defining re-entrancy batching and setup time. MPMN models can now include rules and calculations; 'Data' artifacts were used to increment the step number by one whenever a manufacturing step is completed, and to calculate the processing time for every step is and update it to match the step number 6.3.3.

Also, the complex gateway which is used to model complex flows has been used to display properties on items, and then passes the items through, and used to model setup in both case studies.

This is in addition to the customized reports that MPMN Simulator is capable to create for every scenario and for any selected block. It has the same ExtendSim capabilities of reporting continuous reports during a run and summarized statistically after completing a run for every single replication and for all replications. MPMN Simulator has the ability to report waiting times, bottlenecks, average queue length and it gives a detailed report about the whole process and for a specific block if needed. Moreover, one of the main advantages of the MPMN Simulator is the identification of the steady-state of any of the performance measures; the ability to determine the warmup period and to discard all the statistics before that period from every run.

Finally, there are two viewpoints from which the performance of the MPMN simulator library is evaluated,

From a graphical representation perspective: this work is based on a widely adopted standard. BPMN has a set of understandable notations that helps the modeller to statically describe a system and show the flow of processes, thus the MPMN Simulator library has a glossary of notations that serves manufacturing process representation. MPMN is proposed as an extension to BPMN that adopts new notations that truly represent manufacturing aspects. MPMN showed capabilities to model case studies; it shows the different flows in the system, differentiates between manual tasks, and tasks that are operated by machines. In addition to the initial guide that has been developed to aid the modeler to create graphical and dynamic models using MPMN and MPMN Simulator.

From an execution perspective: the MPMN Simulator uses a well-established discrete event simulation software to build the drag and drop library that has all the necessary notations to model a given manufacturing system. First, it combines the ease of description that the business management analysts want, with the necessary dynamics needed by manufacturing engineering analysts. MPMN Simulator has been developed to model processes in a user-friendly and straightforward environment, and this modelling is complemented with the advantages of simulation where they can analyse experiment, and improve the system without interacting with simulation or doing the programming behind.

7.5 Wider Application of The MPMN to Manufacturing Processes

Using a case study like the Mini-fab to assess the proposed extension, shows that this proposal can be used to represent a very complex manufacturing process. The inclusion of re-entrant flow, operators, batching, machine failures, setups, loading and unloading multi-product processing, preventive maintenance, indicates that the proposed approach can be used to model broader case studies and to represent other manufacturing processes.

As the MPMN Simulator is implemented in ExtendSim there is no translation from notation to simulation. Further, the ease with which the MPMN library can be extended to add more notations and new features makes dealing with any missing representations a quick task. The MPMN Simulator has successfully provided the technical solution envisaged by Proudlove [85]. This approach succeeds in managing the two conflicting objectives of modelling manufacturing systems using easily understood notations while having powerful simulation capabilities. It gives non-simulation experts the ability to build and simulate high fidelity manufacturing models.

7.6 Research Limitations

It is important to point out that proposing new notations to the BPMN comprehensive set of notations can affect novice modelers performance on model understanding, since, there are well-known differences in the way experts and novices create and use diagrams. However, when the purpose of a diagram is to reach a shared high-level understanding of a process among an audience of novices (non-technical stakeholders as end-users, business experts), then it is particularly important to have a language with a few symbols that are understandable and quickly distinguishable [130]. Yet, on the contrary, if a language is to support complex systems, and it is to be fed into a simulation engine it is more important to have a comprehensive set of symbols, that better describe the system and be able to represent different levels of complexities. Moreover, the usefulness of the proposal makes up for the lost simplicity. Modelling using MPMN Simulator allows the users to understand the main concepts behind each block without the need for an additional document.

Another obvious limitation of this research is the application of the newly proposed MPMN on two case studies only, leaving little chance to experiment with its capabilities on broader functions of manufacturing systems. Applying MPMN on various manufacturing systems will give space to the evolvement of future extensions that will offer an even better user experience.

8. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

This work showed how BPMN went through several extensions to develop the MPMN Simulator that is capable to model and simulate manufacturing systems.

This chapter reports the most important findings and conclusions of this work accompanied by recommendations and directions for future work.

8.1 CONCLUSIONS

The main conclusions drawn from this work are summarised as follows:

BPMN was not suitable for modelling manufacturing systems; taking BPMN notations to the shop floor to create a common language that represents the exact picture of manufacturing processes required the enhancement of BPMN notations to fit some manufacturing aspects. Thus, an extension to BPMN has been developed and presented here. MPMN has been proposed to create simple and understandable models, while at the same time it is fully sufficient to model complex systems. In the two case studies, the use of *MPMN appeared to be an effective tool to illustrate some manufacturing aspects with understandable notations*, adding some elements to describe missing manufacturing notations, like queue, batching and un-batching gateways, resource pool, setup, and others.

Static representation was further extended to enhance data analysis and improvements for a given system under study. *Modelling the manufacturing process using Bizagi contributes to confirming that BPMN and BPSim 2.0, (which are both integrated in the Bizagi modeler), are not suitable for decision making*. It was possible to use Bizagi to simulate BPMN notations and to represent error-free models. However, Bizagi failed to represent some manufacturing aspects in a straightforward manner. Calculations, assumptions, and simplifications were always the way to represent and simulate models. Although it was possible to model some of the challenges faced using Bizagi, these kinds of calculations to simplify the model cannot be executed by any user; an expert who has the background to compute variability analysis would be needed, and this is not the purpose of this work. Besides, data analysis using Bizagi is unsatisfactory; it lacks the ability to calculate downtimes, rework, report TH, unable to represent batching and un-batching, and lots of basic manufacturing system aspects.

However, *the developed fully facilitated DES proved its capability to substitute BPSim 2.0*. It has the advantage of allowing a non- simulation expert to build and simulate high fidelity manufacturing models with easy and understandable notations. MPMN Simulator is capable to keep conceptual models intact and to embrace notations that can actually represent the manufacturing system. Moreover, building models was facilitated since notations are built in a drag and drop library and the approach is based on component-based modelling. Furthermore, it is built in the ExtendSim environment; as a result, the MPMN Simulator has the ability to calculate proper cycle time, throughput and utilization, and have verification and validation capabilities. It can generate customized output reports and give information about the machines and whether they are up or down bottlenecks, and the number of waiting units in a queue.

8.2 RECOMMENDATIONS FOR FUTURE WORK

It is recommended that the proposed MPMN Simulator be applied to more cases to enhance broader functions of manufacturing systems, such as planning and inventory management. Thus, MPMN notations will evolve and there will be a need for future extensions that will offer an even better user experience.

Although the proposed extension has been designed with manufacturing system modelling in mind, BPMN notations extension and the fully facilitated DES could have the capability to model and simulate other applications in different domains, like healthcare. Hence, testing the ability of the proposed approach on other complex case studies from other domains will strengthen the credibility, and will broaden the applications of the proposed approach.

At the present time, this simulator is a proof-of-concept to show the possibilities of the extension. As a further step, the MPMN Simulator can serve as a commercial software solution to solve modelling issues in various manufacturing industries and some other domains. This simulator has an extensible library, that offers a drag and drops feature for non- simulation experts, to model and simulate models, also, this may spread the use and exploit the advantages of the proposal. The proposal can allow business owners to observe the potential changes and verify the positive impact they would have on the company before beginning to make operational changes in the organization, which can result in offering solutions to improve companies' profit, customer satisfaction, costs reduction, and allow companies to respond better to competitive pressures.

REFERENCES

[1] K. Zarour, D. Benmerzoug, N. Guermouche, and K. Drira, “A systematic literature review on BPMN extensions,” *Bus. Process Manag. J.*, vol. 26, no. 6, pp. 1473–1503, 2019.

[2] M. Chinosi and A. Trombetta, “BPMN: An introduction to the standard,” *Comput. Stand. Interfaces*, vol. 34, no. 1, pp. 124–134, Jan. 2012.

[3] M. Moreno, “Streamlining interlibrary loan and document delivery workflows : tools , techniques , and outcomes,” *Interlend. Doc. Supply*, vol. 40, no. 1, pp. 31–36, 2012.

[4] B. A. Mousavi, R. Azzouz, C. Heavey, and H. Ehm, “A survey of model-based system engineering methods to analyse complex supply chains: A case study in semiconductor supply chain,” *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 1254–1259, 2019.

[5] T. Djatna and R. Luthfiyanti, “An Analysis and Design of Responsive Supply Chain for Pineapple Multi Products SME Based on Digital Business Ecosystem (DBE),” *Procedia Manuf.*, vol. 4, pp. 155–162, 2015.

[6] Z. El Akkaoui and E. Zimányi, “Defining ETL Workflows using BPMN and BPEL,” in *Proceedings of the ACM twelfth international workshop on Data warehousing and OLAP*, 2009, pp. 41–48.

[7] R. Rosalina Patrícia, N. Lopes, and G. Santos, “Improvement of the food hygiene and safety production process of a Not-for-profit organization using Business Process Model and Notation (BPMN),” *Procedia Manuf.*, vol. 41, no. 2018, pp. 351–358, 2019.

[8] R. Braun, W. Esswein, R. Braun, W. Esswein, D. B. Extensions, and U. Frank, “Classification of Domain-Specific BPMN Extensions,” in *7th IFIP Working Conference on The Practice of Enterprise Modeling (PoEM)*, 2014, pp. 42–57.

[9] I. Abouzid and R. Saidi, “Proposal of BPMN extensions for modelling manufacturing processes,” in *Proceedings of the 5th International Conference on Optimization and Applications*, 2019, pp. 1–6.

[10] S. Zor, D. Schumm, and F. Leymann, “A Proposal of BPMN Extensions for the Manufacturing Domain,” *Proc. 44th CIRP Conf. Manuf. Syst. (ICMS 2011); Madison, Wisconsin, June 1-3, 2011.*, pp. 1–6, 2011.

[11] A. Negahban and J. S. Smith, “Simulation for manufacturing system design and operation: Literature review and analysis,” *J. Manuf. Syst.*, vol. 33, no. 2, pp. 241–261, Apr. 2014.

[12] R. Laue and C. Mueller, “The Business Process Simulation Standard (BPSIM): Chances And Limits,” in *ECMS*, 2016, pp. 413–418.

[13] I. Moreno-Montes De Oca, M. Snoeck, H. A. Reijers, and A. Rodríguez-Morffi, “A systematic literature review of studies on business process modeling quality,” *Inf. Softw. Technol.*, vol. 58, pp. 187–205, 2015.

[14] C. Arevalo, M. J. Escalona, I. Ramos, and M. Domínguez-Muñoz, “A metamodel to integrate business processes time perspective in BPMN 2.0,” *Inf. Softw. Technol.*, vol. 77, pp. 17–33, 2016.

[15] A. Krishna, P. Poizat, G. Salaün, A. Krishna, P. Poizat, G. Salaün, V. Automated, and B. Processes, “VBPMN : Automated Verification of BPMN Processes (tool paper),” in *International Conference on Integrated Formal Methods*, 2017, pp. 323–331.

[16] M. Chinosi and A. Trombetta, “BPMN: An introduction to the standard,” *Comput. Stand. Interfaces*, vol. 34, no. 1, pp. 124–134, Jan. 2012.

[17] R. Braun and H. Schlieter, “Requirements-based development of BPMN extensions: The case of clinical pathways,” in *2014 IEEE 1st International Workshop on the Interrelations Between Requirements Engineering and Business Process Management, REBPM 2014 - Proceedings*, 2014, pp. 39–44.

[18] OMG (2013), “Object Management Group, Business process modeling notation (BPMN) version 2.0.2,” *Object Management Group*. available at: www.omg.org/spec/BPMN/2.0.2/.

[19] C. S. Chao and S. J. H. Yang, “A novel three-tiered visualization approach for firewall rule validation,” *J. Vis. Lang. Comput.*, vol. 22, no. 6, pp. 401–414, 2011.

[20] P. Bera, “Does Cognitive Overload Matter in Understanding BPMN Models?,” *J. Comput. Inf. Syst.*, vol. 4417, pp. 59–70, 2012.

[21] P. Wohed, W. M. P. Van Der Aalst, M. Dumas, A. H. M. Hofstede, and N. Russell, “On the Suitability of BPMN for Business Process Modelling,” in *International conference on business process management.*, 2006, pp. 161–176.

[22] R. Gabryelczyk and A. Jurczuk, “Does Experience Matter? Factors Affecting the Understandability of the Business Process Modelling Notation,” *Procedia Eng.*, vol. 182, pp. 198–205, 2017.

[23] O. M. G. OMG, “Business Process Model and Notation,” no. January, 2015.

[24] T. Takemura, “Formal Semantics and Verification of BPMN Transaction and Compensation,” *2008 IEEE Asia-Pacific Serv. Comput. Conf.*, pp. 284–290, Dec. 2008.

[25] S. A. White, “Introduction to BPMN,” *BPTrends*, pp. 1–11, 2004.

[26] M. Zur Muehlen and J. Recker, “How Much Language is Enough ? Theoretical and Practical Use of the Business Process Modeling Notation .,” in *IN Advanced Information Systems Engineering*, 2008, pp. 465–479.

[27] V. S. W. Lam, “A Precise Execution Semantics for BPMN,” *IAENG Int. J. Comput. Sci.*, vol. 39, no. 1, pp. 20–33, 2012.

[28] B. Yang, L. Qiao, Z. Zhu, and M. Wulan, “A Metamodel for the Manufacturing Process Information Modeling,” *Procedia CIRP*, vol. 56, pp. 332–337, 2016.

[29] R. Sadeghi, S. Dauzere-Pérès, and C. Yugma, “A Multi-Method Simulation

Modelling for Semiconductor Manufacturing," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 727–732, 2016.

[30] Ö. Başak and Y. E. Albayrak, "Petri net based decision system modeling in real-time scheduling and control of flexible automotive manufacturing systems," *Comput. Ind. Eng.*, vol. 86, pp. 116–126, 2015.

[31] J. C. Hernandez-Matias, A. Vizan, J. Perez-Garcia, and J. Rios, "An integrated modelling framework to support manufacturing system diagnosis for continuous improvement," *Robot. Comput. Integr. Manuf.*, vol. 24, no. 2, pp. 187–199, 2008.

[32] S. A. White, "Process Modeling Notations and Workflow Patterns," *Work. Handb.*, pp. 265–294, 2004.

[33] J. Recker, M. zur Muehlen, K. Siau, J. Erickson, and M. Indulska, "Measuring Method Complexity : UML versus BPMN," *Assoc. Inf. Syst.*, 2009.

[34] D. Q. Birkmeier, S. Klöckner, and S. Overhage, "An Empirical Study on the Usability of BPMN and UML Activity Diagrams for Business Users," in *ECIS*, 2010, vol. 2010, pp. 119–134.

[35] W. M. P. Van Der Aalst, A. H. M. Hofstede, and M. Weske, "Business Process Management : A Survey," in *Proceedings of the 1st International Conference on Business Process Management*, 2003, pp. 1–12.

[36] A. Garro, A. Falcone, A. D'Ambrogio, and A. Giglio, "A model-driven method to enable the distributed simulation of BPMN models," in *Proceedings - 2018 IEEE 27th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE 2018*, 2018, pp. 121–126.

[37] A. Lodhi, V. Küppen, and G. Saake, "An Extension of BPMN Meta-model for Evaluation of Business Processes," *Sci. J. Riga Tech. Univ. Comput. Sci.*, vol. 43, no. 1, pp. 27–34, Jan. 2011.

[38] R. Gabryelczyk and A. Jurczuk, "The diagnosis of information potential of

selected business process modelling notations," *Inf. Syst. Manag.*, vol. 4, pp. 26–38, 2015.

[39] M. Zur Muehlen and D. T. Ho, "Service Process Innovation: A Case Study of BPMN in Practice," in *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*, 2008.

[40] T. Pizzuti and G. Mirabelli, "The Global Track & Trace System for food : General framework and functioning principles," *J. Food Eng.*, vol. 159, pp. 16–35, 2015.

[41] J. C. P. Cheng, K. H. Law, H. Bjornsson, A. Jones, and R. D. Sriram, "Advanced Engineering Informatics Modeling and monitoring of construction supply chains," *Adv. Eng. Informatics*, vol. 24, no. 4, pp. 435–455, 2010.

[42] F. Martins and D. Domingos, "Modelling IoT behaviour within BPMN Business Processes," *Procedia Comput. Sci.*, vol. 121, pp. 1014–1022, 2017.

[43] N. Lagos, A. Mos, and M. Cortes-cornax, "Towards semantically-aided domain specific business process modeling," *Data Technol. Appl.*, vol. 52, no. 4, pp. 463–481, 2018.

[44] R. Braun, W. Esswein, R. Braun, W. Esswein, D. B. Extensions, and U. Frank, "Classification of Domain-Specific BPMN Extensions," pp. 42–57, 2014.

[45] F. Zerbato, B. Oliboni, C. Combi, M. Campos, and J. M. Juarez, "BPMN-based representation and comparison of clinical pathways for catheter-related bloodstream infections," in *Proceedings - 2015 IEEE International Conference on Healthcare Informatics, ICHI 2015*, 2015, pp. 346–355.

[46] C. Combi, B. Oliboni, and F. Zerbato, "Modeling and handling duration constraints in BPMN 2.0," in *Proceedings of the Symposium on Applied Computing*, 2017, pp. 727–734.

[47] C. Combi, B. Oliboni, and F. Zerbato, "A modular approach to the specification and management of time duration constraints in BPMN," *Inf. Syst.*, vol. 84, pp. 111–144, 2019.

[48] E. Bazhenova, F. Zerbato, B. Oliboni, and M. Weske, “From BPMN process models to DMN decision models,” *Inf. Syst.*, vol. 83, pp. 69–88, 2019.

[49] G. Decker and F. Puhlmann, “Extending BPMN for modeling complex choreographies,” in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems*, 2007.

[50] H. F. Fernández, E. Palacios-González, V. García-Díaz, B. C. Pelayo G-Bustelo, O. Sanjuán Martínez, and J. M. Cueva Lovelle, “SBPMN — An easier business process modeling notation for business users,” *Comput. Stand. Interfaces*, vol. 32, no. 1–2, pp. 18–28, Jan. 2010.

[51] J. Solís-martínez, J. P. Espada, B. C. P. G-bustelo, J. Manuel, and C. Lovelle, “BPMN MUSIM : Approach to improve the domain expert ’ s efficiency in business processes modeling for the generation of specific software applications,” *Expert Syst. Appl.*, vol. 41, pp. 1864–1874, 2014.

[52] R. M. Pillat, T. C. Oliveira, P. S. C. C. Alencar, and D. D. Cowan, “BPMNt: A BPMN extension for specifying software process tailoring,” *Inf. Softw. Technol.*, vol. 57, pp. 95–115, 2015.

[53] P. Bocciarelli and A. D’Ambrogio, “A BPMN extension for modeling non functional properties of business processes,” in *Proceedings of the 2011 Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium*, 2011, pp. 160–168.

[54] D. Gagne and A. Trudel, “Time-BPMN,” *2009 IEEE Conf. Commer. Enterp. Comput.*, pp. 361–367, Jul. 2009.

[55] A. Yousfi, C. Bauer, R. Saidi, and A. K. Dey, “UBPMN: A BPMN extension for modeling ubiquitous business processes,” *Inf. Softw. Technol.*, vol. 74, pp. 55–68, 2016.

[56] A. Respício and D. Domingos, “Reliability of BPMN Business Processes,” *Procedia Comput. Sci.*, vol. 64, pp. 643–650, 2015.

[57] D. Domingos, A. Respício, and R. Martinho, "Using Resource Reliability in BPMN Processes," *Procedia Comput. Sci.*, vol. 100, pp. 1280–1288, 2016.

[58] M. Jankovic, M. Ljubicic, N. Anicic, and Z. Marjanovic, "Enhancing BPMN 2.0 informational perspective to support interoperability for cross-organizational business processes," *Comput. Sci. Inf. Syst.*, vol. 12, no. 3, pp. 1101–1120, 2015.

[59] E. Negri, L. Fumagalli, M. Garetti, and L. Tanca, "Requirements and languages for the semantic representation of manufacturing systems," *Comput. Ind.*, vol. 81, pp. 55–66, 2016.

[60] J. B. B. Barber, K. D., Dewhurst, F. W., Burns, R. L. D. H., & Rogers, "Business process modelling and simulation for manufacturing management: A practical way forward," *Bus. Process Manag. J.*, 2003.

[61] E. Negri, L. Fumagalli, M. Garetti, and L. Tanca, "A review of semantic languages for the conceptual modelling of the manufacturing domain," *Proc. Summer Sch. Fr. Turco*, 2014.

[62] L. Prades, F. Romero, A. Estruch, A. García-dominguez, and J. Serrano, "Defining a Methodology to Design and Implement Business Process Models in BPMN according to the Standard ANSI / ISA-95 in a Manufacturing Enterprise," *Procedia Eng.*, vol. 63, no. 2004, pp. 115–122, 2013.

[63] L. Shah, A. Etienne, A. Siadat, and F. B. Vernadat, *(Value , Risk) -based Performance Evaluation of Manufacturing Processes*, vol. 45, no. 6. IFAC, 2012.

[64] A. Estruch, J. Antonio, and H. Álvaro, "Event-Driven Manufacturing Process Management Approach," in *International Conference on Business Process Management*, 2012, pp. 120–133.

[65] A. Garcia-Dominguez, M. Marcos, and I. Medina, "A comparison of BPMN 2.0 with other notations for manufacturing processes," in *Proceedings AIP conference, New South Wales, Australia*, 2012, vol. 1431, no. 1, pp. 593–600.

[66] S. Zor, K. Görlach, and F. Leymann, "Using BPMN for Modeling Manufacturing

Processes," in *43rd CIRP Conference on Manufacturing Systems*, 2010, pp. 1–8.

[67] M. R. Khabbazi, M. K. Hasan, R. Sulaiman, and A. Shapi, "Business Process Modelling in Production Logistics : Complementary Use of BPMN and UML," *Middle East J. Sci. Res.*, vol. 15, no. 4, pp. 516–529, 2013.

[68] P. Michalik, J. Štofa, and I. Zolotová, "The Use of Bpmn for Modelling The Mes Level in Information and Control Systems," *Qual. Innov. Prosper.*, vol. 17, no. 1, pp. 39–47, 2013.

[69] M. Polderdijk, V. I., J. Erasmus, K. Traganos, T. Bosch, G. Van Rhijn, and D. Fahland, "A Visualization of Human Physical Risks in Manufacturing Processes Using BPMN," in *International Conference on Business Process Management*, 2017, no. September, pp. 732–743.

[70] R. Braun and W. Esswein, "Extending BPMN for modeling resource aspects in the domain of machine tools," *WIT Trans. Eng. Sci.*, vol. 87, no. December, pp. 450–458, 2014.

[71] V. Cartelli, G. Di Modica, and O. Tomarchio, "Extending the BPMN specification to support cost-centric simulations of business processes," *Commun. Comput. Inf. Sci.*, vol. 631, pp. 492–514, 2016.

[72] S. Bisogno, A. Calabrese, M. Gastaldi, and N. L. Ghiron, "Combining modelling and simulation approaches: How to measure performance of business processes," *Bus. Process Manag. J.*, vol. 22, no. 1, pp. 56–74, 2016.

[73] A. D'Ambrogio, E. Paglia, P. Bocciarelli, and A. Giglio, "Towards performance-oriented perfective evolution of BPMN models," *Proc. 2016 Spring Simul. Multiconference - TMS/DEVS Symp. Theory Model. Simulation, TMS/DEVS 2016*, pp. 1–8, 2016.

[74] G. Wagner, "Tutorial: Information and Process Modeling for simulation," in *Proceedings of the Winter Simulation Conference 2014*, 2014, vol. 2015-Janua, pp. 103–117.

[75] G. Wagner, "Information and Process Modeling for Simulation – Part I: Objects and Events," *J. Simul. Eng.*, vol. 1, no. 1, 2018.

[76] G. Wagner, "Discrete Event Process Modeling Notation (DPMN)," vol. 1, no. of 25, pp. 1–25, 2018.

[77] P. Bocciarelli, A. D'Ambrogio, E. Paglia, and A. Giglio, "An HLA-based BPMN extension for the specification of business process collaborations," in *2017 IEEE/ACM 21st International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, 2017, pp. 1–8.

[78] S. Gorecki, Y. Bouanan, G. Zacharewicz, J. Ribault, and N. Perry, *Integrating HLA-Based Distributed Simulation for Management Science and BPMN*, vol. 51, no. 11. Elsevier B.V., 2018, pp. 655–660.

[79] A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, "Tutorial: information and process modeling for simulation," in *Proceedings of the 2014 Winter Simulation Conference - , WSC 2014*, 2014, pp. 103–117.

[80] Bhakti S. S. Onggo, "BPMN PATTERN FOR AGENT-BASED SIMULATION MODEL REPRESENTATION," in *Proceedings of the 2012 Winter Simulation Conference*, 2012, no. Zeigler 1976.

[81] M. Yi and K. Filippidis, "BPSIM Standard," *Work. Manag. Coalit.*, no. 0828102, pp. 1–41, 2013.

[82] R. Laue and C. Mueller, "The Business Process Simulation Standard (BPSIM): Chances And Limits.," *Ecms*, vol. 2, no. Cd, 2016.

[83] N. Afifi, A. Awad, and H. M. Abdelsalam, "RBPSim: A Resource-aware Extension of BPSim Using Workflow Resource Patterns," in *ZEUS*, 2018, pp. 32–39.

[84] B. S. Onggo, N. C. Proudlove, S. A. D'Ambrogio, A. Calabrese, S. Bisogno, and N. Levialdi Ghiron, "A BPMN extension to support discrete-event simulation for healthcare applications: an explicit representation of queues, attributes and data-driven decision points," *J. Oper. Res. Soc.*, vol. 69, no. 5, pp. 788–802, 2018.

[85] N. C. Proudlove, S. Bisogno, B. S. S. Onggo, A. Calabrese, and N. Levialdi Ghiron, “Towards fully-facilitated discrete event simulation modelling: Addressing the model coding stage,” *Eur. J. Oper. Res.*, vol. 263, no. 2, pp. 583–595, 2017.

[86] N. Rizun, A. Revina, and V. Meister, “Method of Decision-Making Logic Discovery in the Business Process Textual Data,” in *International Conference on Business Information Systems*, 2019, pp. 70–84.

[87] F. Zerbato, “Tackling Different Business Process Perspectives.".”

[88] J. Erasmus, I. Vanderfeesten, K. Traganos, and P. Grefen, “Using business process models for the specification of manufacturing operations,” *Comput. Ind.*, vol. 123, p. 103297, Dec. 2020.

[89] R. Braun, H. Schlieter, M. Burwitz, and W. Esswein, “BPMN4CP revised - Extending BPMN for multi-perspective modeling of clinical pathways,” in *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2016, pp. 3249–3258.

[90] H. H. Chiu and M. S. Wang, “Extending event elements of business process model for internet of things,” in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, 2015, pp. 783–788.

[91] W. Khlif, H. Ben-Abdallah, and N. E. AyedBen, “A methodology for the semantic and structural restructuring of BPMN models,” *Bus. Process Manag. J.*, 2017.

[92] K. B. M. Noor, “Case study: A strategic research methodology,” *Am. J. Appl. Sci.*, vol. 5, no. 11, pp. 1602–1604, 2008.

[93] S. K. Milton and L. W. Johnson, “Service blueprinting and BPMN: A comparison,” *Manag. Serv. Qual.*, vol. 22, no. 6, pp. 606–621, 2012.

[94] L. P. Lopez-Arredondo, C. B. Perez, J. Villavicencio-Navarro, K. E. Mercado, M. Encinas, and P. Inzunza-Mejia, “Reengineering of the software development

process in a technology services company," *Bus. Process Manag. J.*, vol. 26, no. 2, pp. 655–674, 2019.

[95] R. Brown, J. Recker, and S. West, "Using virtual worlds for collaborative business process modeling," *Bus. Process Manag. J.*, vol. 17, no. 3, pp. 546–564, 2011.

[96] R. Pelorosso, "Modeling and urban planning: A systematic review of performance-based approaches," *Sustain. Cities Soc.*, vol. 52, p. 101867, 2019.

[97] J. H. Helquist, A. Deokar, J. J. Cox, and A. Walker, "Analyzing process uncertainty through virtual process simulation," *Bus. Process Manag. J.*, vol. 18, no. 1, pp. 4–19, 2012.

[98] Q. Chen, B. T. Adey, C. Haas, and D. M. Hall, "Using look-ahead plans to improve material flow processes on construction projects when using BIM and RFID technologies," *Constr. Innov.*, pp. 471–508, 2020.

[99] J. Recker, "BPMN Modeling – Who , Where , How and Why," in *BPTrends*, 2008, vol. 2, no. May, pp. 1–8.

[100] W. M. P. Van Der Aalst, "Don ' t go with the flow : Web services composition standards exposed," 2003.

[101] N. Lohmann, E. Verbeek, and R. Dijkman, "Petri Net Transformations for Business Processes – A Survey," in *Transactions on petri nets and other models of concurrency II*, 2009, pp. 46–63.

[102] J. Recker, M. Indulska, M. Rosemann, and P. Green, "How good is BPMN really? Insights from theory and practice.," IT University of Gotteborg., 2006.

[103] J. Recker, "Opportunities and constraints: The current struggle with BPMN," *Bus. Process Manag. J.*, vol. 16, no. 1, pp. 181–201, 2010.

[104] A. Chouichi, J. Blue, C. Yugma, and F. Pasqualini, "Chamber-to-Chamber Discrepancy Detection in Semiconductor Manufacturing," *IEEE Trans. Semicond. Manuf.*, vol. 33, no. 1, pp. 86–95, Feb. 2020.

[105] M. A. Chik, A. B. Rahim, A. Z. M. Rejab, K. Ibrahim, and U. Hashim, “Discrete event simulation modeling for semiconductor fabrication operation,” *IEEE Int. Conf. Semicond. Electron. Proceedings, ICSE*, pp. 325–328, 2014.

[106] H. Geng, *Semiconductor Manufacturing Handbook*, Second Edi. McGraw-Hill Education, 2018.

[107] R. Singh and M. Mathirajan, “A New Stage-Wise Control Release Policy for Semiconductor Wafer Fabrication Systems,” *IEEE Trans. Semicond. Manuf.*, pp. 1–1, 2020.

[108] J. A. Ramirez-Hernandez and E. Fernandez, “A simulation-based Approximate Dynamic Programming approach for the control of the Intel Mini-Fab benchmark model,” in *Proceedings of the 2009 Winter Simulation Conference (WSC)*, 2009, pp. 1634–1645.

[109] C. Natschläger, “Towards a BPMN 2.0 Ontology,” in *International Workshop on Business Process Modeling Notation*, 2011, pp. 1–15.

[110] S. Lemaignan, A. Siadat, J. Y. Dantan, and A. Semenenko, “MASON: A proposal for an ontology of manufacturing domain,” in *Proceedings - DIS 2006: IEEE Workshop on Distributed Intelligent Systems - Collective Intelligence and Its Applications*, 2006, pp. 195–200.

[111] D. This, “Extending BPMN for modeling manufacturing processes Master of Science,” 2017.

[112] C. Battista, G. Dello Stritto, F. Giordano, R. Iannone, and M. M. Schiraldi, “Manufacturing systems modelling and simulation software design: A reference model,” *Ann. DAAAM Proc. Int. DAAAM Symp.*, no. May 2014, pp. 1107–1108, 2011.

[113] B. S. S. Onggo, “Towards a unified conceptual model representation: A case study in healthcare,” *J. Simul.*, vol. 3, no. 1, pp. 40–49, 2009.

[114] F. Corradini, A. Ferrari, F. Fornari, S. Gnesi, A. Polini, B. Re, and . O. Spagnolo, “A

guidelines framework for understandable BPMN models,” *Data Knowl. Eng.*, vol. 113, pp. 129–154, 2018.

[115] D. Krahl, “ExtendSim advanced techology: Discrete rate simulation,” in *Proceedings - Winter Simulation Conference*, 2009, pp. 333–338.

[116] D. Krahl, “ExtendSim: a history of innovation,” in *Proceedings of the Winter Simulation Conference*, 2012, pp. 1–8.

[117] T. J. Schriber, D. T. Brunner, and J. S. Smith, “Inside discrete-event simulation software: How it works and why it matters,” in *Proceedings of the 2013 Winter Simulation Conference WSC 2013*, 2013.

[118] M. A. Shararah, K. S. El-kilany, and A. E. El-sayed, “Component Based Modeling and Simulation of Value Stream Mapping for Lean Production Systems.”

[119] K. S. El-Kilany, “Reusable Modelling and Simulation of Flexible Manufacturing for Next Generation Semiconductor Manufacturing Facilities,” 2003.

[120] J. Imen, M. Khemaja, and M. Abed, “Towards MDA implementation based on a novel BPMN metamodel and ATL transformation rules,” *2011 4th Int. Conf. Logist.*, pp. 62–67, May 2011.

[121] O. Allani and S. A. Ghannouch, “Verification of BPMN 2.0 process models: an event log-based approach,” in *Procedia Computer Science*, 2016, vol. 100, pp. 1064–1070.

[122] A. M. Law, “How to build valid and credible simulation models,” *Proc. - Winter Simul. Conf.*, pp. 24–33, 2009.

[123] B. L. Nelson, D. M. Nicol, J. Banks, and J. S. Carson, *Discrete Event system Simulation*, Fifth Edit. Pearson, 2010.

[124] J. Strickland, *Discrete Event Simulation Using ExtendSim 8*, First Edit. Lulu, 2013.

[125] A. Mili, A. Siadat, S. Bassetto, S. Hubac, and M. Tollenaere, “Unified process for action plan management: Case study in a research and production

semiconductor factory," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 6, pp. 4–8, 2010.

[126] K. Kempf, "Intel Five-Machine Six Step Mini-Fab Description," 1994.

[127] K. S. El-Kilany, *INTEL FIVE-MACHINE SIX STEP MINI-FAB*. .

[128] I. A. El-khouly, "Investigation of Wip Management for Control of Semiconductor Manufacturing Segments Investigation of Wip Management for Control of Semiconductor," DCU, 2015.

[129] H. Bart, *Addressing Ambiguous BPMN Patterns: A Case Study at The BrainCap Group*, no. 10254846. 2014, pp. 1–71.

[130] N. Genon, D. Amyot, and P. Heymans, "Analysing the cognitive effectiveness of the BPMN 2.0 visual notation," in *International Conference on Software Language Engineering.*, 2010, pp. 377–396.

APPENDIX A

A PUBLICATION RELATED TO THIS WORK

This Appendix includes the publication related to the topics discussed.

B. El-Sharef, K. S. El-Kilany, and P. Young, "Using Bpmn To Capture and Model Complex Manufacturing Systems," in *IIE Annual conference and Expo*, 2016.

APPENDIX B

B MPMN SIMULATOR GUIDE

This appendix provides a preliminary guide to create models using MPMN and MPMN Simulator

An initial guide has been developed to aid the modeler to create graphical and dynamic models using MPMN and MPMN Simulator. Where some of the notations are adopted as-is from the BPMN set of notations, others are modified BPMN notations to fit in the new MPMN set of notations, and the remaining ones are newly developed or extended notations. These notations can be either used for graphical representation only or can be used for graphical representation in addition to having simulation capabilities. Table B-1 summarizes these notations: where, (M) is Modified notation, (A) is Adopted As-Is BPMN notation, (N) is a New or extended notation, (G) is for Graphical representation, (S) indicates that a notation can be Simulated, and (GS) is for Graphical and Simulation modelling.

Table B-1: Preliminary guide to create models using MPMN and MPMN Simulator.

Notation Name	BPMN	MPMN	Bizagi	MPMN Simulator
Flow Objects				
ACTIVITIES				
Task Activity	A	A	GS	GS
Send Task	A	A	GS	G
Receive Task	A	A	GS	G
User Task	A	M	GS	GS
Loop Task	A	A	GS	G
Manual Task	A	M	GS	GS
Sub-process	A	M	GS	GS
Manual Task	A	M	GS	GS
Processing		N		GS
Material handling		N		GS
Task subjected to failure		N		GS
Loading		N		GS
Un-loading		N		GS
Sub-Process		N		GS
Warehouse		N		GS
EVENTS				

Notation Name	BPMN	MPMN	Bizagi	MPMN Simulator
Start Events				
Start Event	A	A	GS	GS
Message Start Event	A	A	GS	GS
Timer Start Event	A	A	GS	GS
Conditional Start Event	A	M	G	GS
Signal Start Event	A	A	G	G
Multiple Start Event	A	A	G	G
Error Start Event	A	M	G	GS
Intermediate Events				
Intermediate Message Event	A	A	GS	GS
Intermediate Timer Event	A	A	GS	GS
Intermediate Conditional Event	A	M	GS	GS
Intermediate Error Event	A	M	GS	GS
Intermediate Signal Event	A	A	GS	GS
Waiting time		N		GS
Queue		N		GS
Scheduled shutdown		N		GS
Emergency shutdown		N		GS
Catch		N		GS
Throw		N		GS
Conditional intermediate event		N		GS
End Events				
Non-End Event	A	A	GS	GS
Message End Event	A	M	GS	GS
Error End Event	A	A	GS	GS
Signal End Event	A	A	GS	GS
Multiple End Event	A	A	GS	GS
Terminate End Event	A	A	GS	GS
Cancel End Event	A	A	GS	GS
GATEWAYS				
Exclusive Gateway	A	A	GS	GS
Inclusive Gateway	A	A	GS	GS
Parallel Gateway	A	A	GS	GS
Complex Gateway	A	M	G	GS
Event Exclusive Gateway	A	A	G	G
Event parallel Gateway	A	A	G	G
Batch		N		GS
Un-batch		N		GS
Material Select Gateway		N		GS
Material Route Gateway		N		GS
Complex Gateway		N		GS
Connecting Objects				
Sequence Flow	A	A	GS	GS
Default Sequence Flow	A	A	GS	GS
Conditional Sequence Flow	A	A	GS	GS
Message Flow	A	A	G	G
Association	A	A	G	G
Data Associations	A	A	G	G
Material Flow Connector		N		G

Notation Name	BPMN	MPMN	Bizagi	MPMN Simulator
Swimlanes				
Pools	A	A	GS	G
Lanes	A	A	GS	G
Artifacts				
Data object	A	M	G	GS
Text Annotation	A	A	G	G
Group	A	M	G	G
WorkStation		N		G
Resource Pool		N		GS