MASTER

How to improve and automate decisions with Decision Mining and Decision Modeling at Fokker Services

Jonkers, Marco

Award date:
2023

How to improve and automate decisions with Decision Mining and Decision Modeling at Fokker Services

Master Thesis

Student:

M. Jonkers 0909483

Supervisors TU/e:

dr. H. Eshuis

dr. W.L. van Jaarsveld

Supervisor Fokker Services:

W. van Dis

Eindhoven University of Technology

Abstract

In this research it was investigated how decisions within Fokker Services could be improved or automated. To achieve this, a Decision Mining technique was implemented to try to obtain useful decision logic for improving the decisions. During this implementation there was a special focus for the effect of data quality on decision-making. Based on the results improved decision logic for the decisions was obtained that were modeled according to the DMN standard. These improvements should help Fokker Services make better decisions and improve the performance of their processes.

Executive Summary

Fokker Services is an aerospace service provider for regional, commercial, and military aircraft. They provide a wide range of services to customers in daily operations. These operations consist of many processes and tasks and require many operational decisions. However, Fokker Services faces the problem that these operational decisions are made very inefficiently which negatively influences the performance of the processes. Besides improving these operational decision Fokker Services also aims to automate them in the future. However, they face the problem that they lack the knowledge of how to improve the decisions within the organization. Therefore, the main objective in this master thesis project will be to help Fokker Services solve the issue by investigating the possibilities for improving the operational decisions. Since Fokker Services provide many services that require many different operational decisions. Including all of these in the research and trying to improve them is not possible as there are simply too many decisions. Therefore, the scope of this research was limited to the operational decisions within the services of CMRO and Exchange Programs. For the investigation of possibilities to improve the decisions, the topics of Decision Mining and Decision Modeling with the DMN standard will have a central role in this research. For this objective, the following research question was developed:

How can decision mining techniques help Fokker Services improve their decisions within CMRO and Exchange Programs?

Both Decision Mining and Decision Modeling with the DMN standard are topics focused on improving decisions within a process. Decision Mining is referred to as decision point analysis and aims at the detection of data dependencies which are rules that explain which circumstances lead to a certain activity and explain when an activity is selected over another activity. Decision Modeling expresses how a decision should be made as a rigorous, verifiable model. It formalizes decision-making so it can be clearly and widely understood, managed, and used effectively. Decision Modeling and Notation (DMN) is a notation standard specifically developed for Decision Modeling that became the industry standard.

The first step was to identify and select the decisions from the processes of CMRO and Exchange Programs that will be considered for improvement in this research. Three decisions were selected for this research for improvement based on a set of criteria for Suitable Decisions for Automation: 'Decide on Service', 'Create Price Quote', and 'Decide on In-house or Outsource'.

The objective of this research is to try to improve decisions. Without knowing the current situation it is not possible to determine whether the improved decisions are better. To determine this the performance of the processes will be calculated by using the performance measures of the total time to complete orders and the service levels of orders. In the current situation the service levels turned out to be relatively low with 66% for the orders of the service CMRO and 49% for the orders of the service Exchange Programs. These are much lower than the target of 90% set by Fokker Services.

Data quality plays an important role in Decision Mining and should be taken into account when improving decisions. The next step was to investigate which data is required for the decisions and test the data quality of this data. The data quality was checked with the help of a Data Quality Assessment. The results of this quality assessment were quite good as there were not many quality issues identified. Some of the issues could be solved with data cleaning and the resulting data sets had a relatively high data quality.

The last thing remaining before the implementation of the Decision Mining Technique was to determine whether the decisions could be fully automated. In the case of the decisions 'Decide on Service' and 'Decide on In-house or Outsource' there are no limitations to fully automate these decisions. However, for the decision of 'Create Price Quote' there is one limitation. Due to the high variety of components within the process, it is not possible to create general business rules for this decision. Therefore, this decision needs to remain dependent partially on human knowledge and input.

For the implementation, a relatively simple Decision Mining technique was selected. This is to prevent the implementation from becoming too complex. Besides that, it was chosen to test the technique on one decision. 'Decide on Service' was the selected decision for this as it was considered the most important decision in the process.

During the implementation some interesting findings were done related to the effect of data quality. It turned out that the data set was extremely imbalanced, a quality issue that was not included in any quality assessment. Besides that, much of the relevant data according to the quality assessment, turns out to be not relevant at all for the implementation of Decision Mining. With further data cleaning, it was possible to solve these issues. However, it was an important finding that some specific data quality issues occur after the completion of the quality assessment.

The result of the implementation provided a quite extensive decision tree. However, the achieved decision logic does not make much sense from a logical perspective. The best example is shown in figure 1 below, which is a snapshot of the decision tree. In the first node of the decision tree, the tree tried to group different departments for decision logic. The issue with this is that the departments mostly operate as separate shops and can have many differences. That also becomes clear from the fact that the next node in the decision tree is different for both groups of departments.

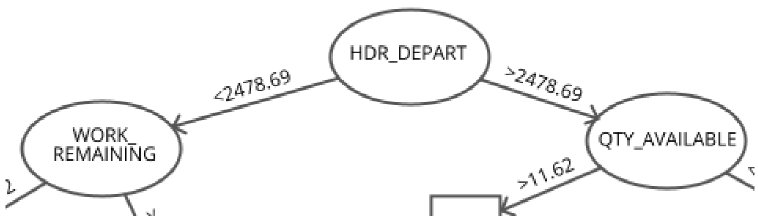

Figure 1: First Node Decision Tree

It is also more logical to treat the departments as completely separate shops as that is how the shops operate. Therefore, it was tried to apply the technique again with sub-datasets for each department. For the departments, 2400, 2500, 2520, 2540, and 2550 a decision tree was created, but for the other departments, it was not even possible to create decision trees. This is not very surprising since the data set only includes data about all orders and the organization in general and does not include any data for specific departments. For example, the data such as the Work Remaining and Orders in Progress is determined for all departments together, not individually for each department. Even though decision trees could be created for some departments, the decision logic that is obtained is based on incorrect data, which makes it impossible to make trustworthy conclusions from it.

The penultimate part of this research was to identify the required changes to improve the decisions. For each of the three selected decisions, the ideal situation for this decision was created. As explained, the only useful finding from the implementation was related to the consideration of separate departments. Despite that, the ideal situation for the decisions was modeled by investigating the issues in the current situation and coming up with solutions for these issues to improve these step-by-step.

When comparing the ideal situation with the initial situation, it becomes clear that quite some changes are necessary to improve the decisions. The decision of 'Decide on In-house or Outsource' is the decision that requires the biggest changes. In short, the decision is not considered an active decision, while in the ideal situation, it should be considered as an active decision again. For the 'Decide on Service' important information as the capacity and how busy it is in the shops need to be considered in the decision. For the last decision of 'Create Price Quote', it is important to consider information and data from orders in the past when they are available.

Besides these changes to the specific decision, it is probably more important that Fokker Services focuses on collecting the correct data that is required. The ideal situation for the decisions contains quite some input that is not available at the moment within the organization. However, once this data is available, Decision Mining should provide a lot of useful decision logic that can be used for improving decisions.

The final part of this research was to investigate the impact of improving the decisions. It was determined that the total time of the orders and the service level were the most important performance measures. The decisions in the ideal situation should prevent the overload of work from occurring. As a result of that fewer delays should occur as the waiting time of orders decrease. It was analyzed what the total time of the orders would be without this waiting time and new service levels were calculated for the ideal situation. A comparison of the service levels between the current and the ideal situation is given in table 1 below, which show quite some improvements.

	Current Situation	Ideal Situation	Improvement
Customer Orders	66%	81%	15%
Exchange Orders	49%	85%	36%

Table 1: Comparison Service Level

The only thing remaining is the answer to the main research question, which was: *How can decision mining techniques help Fokker Services improve their decisions within CMRO and Exchange Programs?* Decision Mining techniques were not very helpful for the specific decision in this research as they provide not many useful results. To implement the ideal situation more specific data should be collected first, otherwise no useful results will be obtained. However, provided that this data can be collected, Decision Mining can still be very useful in the future to improve decisions within Fokker Services. The final recommendation is related to the problem of the lack of knowledge about improving decisions. The Decision Management Approach is a complete approach that provides a solid guideline as it provides every step required to improve the decisions. Even though Decision Mining is not part of this approach it can be used very well as a complementary step to determine decision logic for improved decisions.

Contents

List of Figures

List of Tables

1 Introduction

Fokker Services is an organization that performs service for customers in daily operations. These operations are part of many different processes and require many important operational decisions. However, these decisions are not made very efficiency and allow for many improvements. Besides that, Fokker Services does not have a lot of knowledge on how good decision are made or how decisions could be improved. As a result of this, it is difficult for Fokker Services to solve this problem.

1.1 Business Context

Fokker Services is an aerospace service provider for regional, commercial, and military aircraft. They provide a wide range of services to customers, which could be certificate holder-related product support services, flight hour-based component availability and repair programs, spare parts, engineering, modifications, and documentation support. These services are generally divided into seven different groups of services: Aircraft MRO, Component Maintenance Repair and Overhaul (CMRO), Parts Availability, Exchange Programs, Engineering Services, Aircraft Modifications, and Defense. An explanation of the services is included in Appendix A.

These services differ a lot from each other and all contain quite complex processes. As an aerospace service provider Fokker Services provides service all across the world and has customers in many different countries. To be able to provide the worldwide service, Fokker Services has several locations across the world. The locations in Hoofddorp and Schiphol in The Netherlands are mainly focused on the European customers. The location in Lagrange in the United States serves the different customers across the entirety of America. Finally, the location in Singapore focuses on the customers in Asia. All these services across the different locations consist of many different processes. Managing all these different services across the world is quite complex and also lead to several difficulties.

1.2 Problem Definition

Fokker Services encounters the problem that operational decisions are not made very efficiently. Almost all of these decisions are made by employees and depend heavily on their knowledge and experience. Due to this dependence on the employees, many decisions are very time-consuming as well. Besides that, not always the best possible outcomes are chosen. One of the main reasons for this is that the available data does not provide the required information or conclusions to make the decisions. In these cases, the employees still have to investigate some data or make their conclusions based on the data. Therefore, the outcome of the decision depends partly on how the employees interpret the available data and information. That results in the possibility that different employees make different decisions based on the same information and data.

The services provided by Fokker Services to the customers consists of many processes that include many different operational decisions. Having inefficient or sometimes even wrong decisions influences the processes negatively and processes become inefficient as well. As a result customer orders encounter delays and are not delivered on time anymore. With most of the customers Fokker Services maintains contracts with predetermined service levels. However, in many cases these service levels are not met. Another result of this is that the amount of work in the repair shop is piling up and they are not able to process all incoming orders. Even though there may be several other factors that lead to these consequences, the inefficient decisions is considered to be the main issue.

Not only are the decisions made inefficiently, but all operational decisions also have to be made manually by the employees. Even though there are quite some relatively simple decisions within the organization, they still demand a lot of time from the employees as the number of decisions is very high. In the current situation basically, none of the operational decisions are automated in some way. Within Fokker Services there is a general idea that there should be possibilities to improve decision-making by automating the operational decisions. Especially since there is a lot of data available within the organization. However, Fokker Services is lacking knowledge on how the decisions within the organization can be made more efficiently. This is also one of the reasons why Fokker Services has not actively tried to improve or automate the decisions within the organization. Based on this problem the following problem statement was defined:

Problem Statement
Operational decisions within the organization are made inefficiently and can be improved, but Fokker Services is lacking knowledge on how to improve their decisions.

1.3 Scope

Improving every operational decision within the organization of Fokker Services is not really possible within the scope and time frame of this research. There are simply too many operational decisions within the organization to improve them all. Besides that, as described in the first part of this chapter, Fokker Services performs many different type of services. Due to the variety of services provided to customers not all services can be considered within this research. These services are completely different from each other and all consist of different processes. Therefore, the scope of this research will be limited quite a lot. The first limitation is regarding the type of services that will be included. In this research the services of Component Maintenance Repair and Overhaul (CMRO) and Exchange Programs will be the main focus point. The other services provided to customers will be kept outside of the scope during this research. CMRO and Exchange Programs are both separate services, with each its own decisions included in the process. Even for only these two services improving all decisions will be too much within the time frame of this research and a selection is required. Initially all decisions within the process of these two services are considered for improvement, but the most important ones within the process will be selected for improvement during the research. Which decisions will be selected and why they are selected is the part of this research and will be discussed in a later chapter. All decisions that are not selected during this part of the research will be kept outside of the scope during the remainder of this research.

1.4 Research Objective

The objective of this research is to investigate the possibilities of how Fokker Services can improve its decisions within the organization. Fokker Services aims to improve their decisions within the organization and try to make them less dependent on the employees' knowledge and experience. Improving the decisions, should not only take less time to make them but also prevent that sub-optimal options are chosen. If the chosen options appear to not be the most optimal, delays or issues will arise with the provided services. In the end one of the main goals of Fokker Services is to provide the services on time to their customers. So the total time to handle the customers' requests is the main criterion throughout this research and the time measures will be used to measure the improvement of decisions. However, there is one important constraint for this time measure criterion. Which is that required service levels for the customers' requests are still matched. This is to prevent scenarios where the optimal decision is to do nothing and reject all the customers' requests, as that will lead to the lowest time to handle the customers' requests.

As described in the previous part certain decisions will be selected for improvements. However, just improving these selected decisions is not the only objective of this research. Fokker Services aims to further improve other decisions that are outside of the scope of this research in the future as well. Another objective is to investigate how decisions can be improved in general, so they have the knowledge to improve other decisions in the future. A central topic for improvement of the decisions is the use of decision mining. During this research, it will be investigated how decision mining techniques can be applied to improve the decisions and investigate the impact of these techniques. As described earlier in this section the scope is limited to the decisions within the services of CMRO and Exchange Programs.

Besides the main objective for Fokker Services, there is also a side objective that should deliver the contribution literature. An important topic of decision mining is the data quality. According to Rozinat and Van Der Aalst, 2006 it is one of the biggest challenges. The success rate and effect of the decision mining techniques depend heavily on the data quality. Without having data of high quality it is not possible to fully utilize the decision mining techniques. There is a lot of literature available on how to find issues with the data quality and how these can be solved. Not specifically for decision mining, but there is a lot available for data mining and process mining. That requires similar high quality data and can be used well for decision mining. However, there is almost no available literature about the effect of data quality on the decision-making process. In this project the goal is to improve decisions with the available data within the organization. So, the data available within Fokker Services is everything that can be used throughout this research. As far as possible data quality will be improved, but there is no guarantee that all the data will be perfect, without any shortcomings or available. The effect of the data quality on the possible improvements of decision-making will be a big topic in this project. Since there it not much literature about this available, this topic will be the main contribution to the literature.

Based on this research objective the following research question was formulated:

1.5 Research Questions

Main Research Question
How can decision mining techniques help Fokker Services improve their decisions within CMRO and Exchange Programs?

To help answer the main research question, several sub-questions were formulated:

- Q1: What are the important decisions within the process that could be improved?
- Q2: Which data is required for the decisions, and what is the quality of this data?
- Q3: Which of the decisions to be improved can be fully automated, and which decisions remain (partly) dependent on human knowledge?
- Q4: Which changes in the process are required to improve the decisions, and can these be achieved with decision mining techniques?
- Q5: What is the impact of the improved decisions?

An overview of the deliverables for each sub-question is given below:

- Q1: An overview of the identified decisions within the process and their influence on the process.
- Q2: A decision model of the current decisions within the process and the results of the data quality assessment of the required data in the decision model.
- Q3: An overview of the decisions whether they are fully automated or not including the reasons for the possibility to automate them. The reasons should always include the effect of the data quality.
- Q4: A decision model of the ideal situation and an overview of the required changes to achieve this situation.
- Q5: A comparison of the total time of order request and service levels of the current and ideal situation.

1.6 Thesis Structure

In the last part of the introduction, an overview of the structure of the thesis report will be given. Chapter 2 consists of a literature review that covers the main topics of this research: Decision Mining and Decision Modeling with the DMN standard. In Chapter 3 the methodology of this research will be explained, with a more detailed explanation of how the research questions will be answered. Chapter 4 provides information on how decisions can be automated and when decisions are suitable for this. This information will help investigate the possibilities to automate the decisions in a later chapter. Chapter 5 describes the relevant processes with the different decisions. In this chapter, the sub-question Q1 will be answered. Chapter 6 focuses on identifying the required data for the decisions and checks the data quality for this data. This chapter will provide answers to the sub-question Q2. Chapter 7 describes the implementation of the Decision Mining technique and provides the results of this implementation. With this chapter part of sub-question Q4 will be answered. In the first part of Chapter 8, the ideal situation for the decisions will be provided. This includes a description of to which extent they will be automated. Based on this sub-questions Q3 and Q4 will be answered. In the last part of this chapter, a comparison is made between the current and ideal situation to answer the last sub-question Q5. Chapter 9 is the final chapter and provides the conclusion of this research.

2 Literature Review

In this chapter the relevant topics from literature for this research will be discussed. In this research the focus is on trying to improve decisions within a process. Topics that are aiming to achieve this are Decision Mining and Decision Modeling with Decision Model and Notation (DMN) standard. First, it will be discussed what the topics are about. Next, it will be discussed how the Decision Mining and Decision Modeling can be useful. Finally, several challenges for the topics will be included.

2.1 Decision Mining

Decision Mining is a term that exist for quite a while in literature. The research of Rozinat and Van Der Aalst, 2006 was the first to introduce the term of Decision Mining back in 2006. However, it took several years until other researchers performed more research on the topic of Decision Mining. Meanwhile everything has changed as Decision Mining has became a prevalent term in the area of business process management (De Smedt, vanden Broucke, et al., 2017). Besides that, decision mining was applied in different contexts by different researchers (De Smedt, vanden Broucke, et al., 2017).

2.1.1 Initial Decision Mining Approach

Decision Mining is also referred to as decision point analysis and aims at the detection of data dependencies that affect the routing of a case (Rozinat & Van Der Aalst, 2006). These data dependencies are so called rules, that explain which circumstances lead to a certain activity and explain when an activity is selected over another activity (Rozinat & Van Der Aalst, 2006). To discover these rules Rozinat and Van Der Aalst, 2006 developed an approach that use event log data from a Process-aware Information System (PAIS). A complete overview of this approach is given in figure 2.

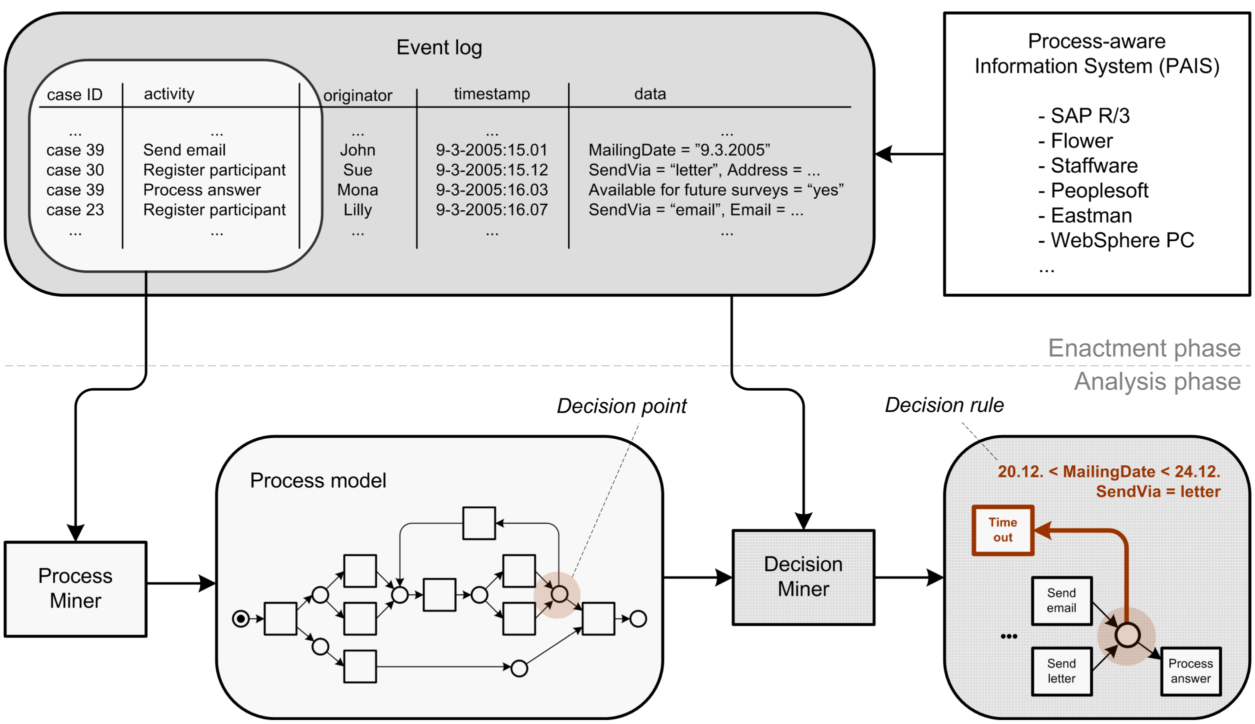

Figure 2: Overview Decision Mining Approach (Rozinat & Van Der Aalst, 2006)

The first step is that a PAIS records data in an event log. This event log is used to discover the process and create a process model by using process mining algorithms. Several algorithms can be used, but most important is that decision points will be identified in the process (Rozinat & Van Der Aalst, 2006). Next, the decision point analysis need to be performed for these identified decision points. The main step in this approach is to turn every decision point into a separate learning problem that can be solved with machine learning algorithm. More specifically, the decision points are converted into a classification problem which can be solved by using decision trees. Several decision tree algorithms can be used to solve the classification problem, but Rozinat and Van Der Aalst, 2006 selected the C4.5 algorithm as it is one of the most popular algorithms and has some practical applications. The last step in the approach is to transform the decision trees into decision rules. This is done by converting the decision tree to logical expressions.

2.1.2 Improvements and Extensions

Several similar decision mining approaches were developed where the problem is converted into a classification problem that was solved with decision tree algorithms. However, these approaches have one big shortcoming. The all rely on the strong assumption that the rules attached to the activity choice needs to be mutually exclusive, which does not hold often in reality (De Leoni et al., 2016). In most cases business rules are non-deterministic and decision are influenced by contextual information that is not included in the event logs. So similar conditions may lead to the selection of a different activities and rules will have overlap. To handle this issue, De Leoni et al., 2016 developed a technique that is able to discover these overlapping rules and use trade-offs in precision and fitness. Similar to other approaches, an initial decision tree is created with event log data. Afterwards the wrongly classified instances for each tree are used to create another decision tree. Based on this tree another set of decision rules are developed which are used in disjunction with the initially created decision rules. As a result a set of overlapping rules is obtained (De Leoni et al., 2016).

Another shortcoming of the approach of Rozinat and Van Der Aalst, 2006 is that it only focuses on the retrieval of control flow decisions but neglect data decisions and dependencies included in the logged data. Therefore, Bazhenova et al., 2016 made an extension to this approach that includes the identification of data decisions and dependencies between them. Besides that, they created an algorithm that detect the dependencies between discovered control flow and data decisions, and provide a complete DMN decision model. Another extension to handle this issue was developed by De Smedt, Hasic, et al., 2017. They developed an approach that also includes the different type of activities in the process model and determine in which way they contribute to the decision layer of the model.

Besides these, many other decision mining algorithms were developed throughout the years. Most of them were improvements or additions to the approach of Rozinat and Van Der Aalst, 2006. However, some used different approaches to solve the classification problem. According to Bazhenova and Weske, 2016 there are three prominent approaches for solving the classification problems. The first one is by Baesens et al., 2003 that solved the problem by constructing a neural network. The second one is by Lovell and Walder, 2007 that solved the problem with the help of support vector machines. Both of these have very accurate classifications as output, much higher than the decision trees. However, the both have a big downside as well. The output of a neural network are very complex which is very difficult to transform into decision rules or a decision model. For the support vector there are even no existing solutions to transform the output to decision rules. This is not an issue with the use of decision trees like the research of Rozinat and Van Der Aalst, 2006, which was the last prominent approach mentioned by Bazhenova and Weske, 2016. Transforming the collected business rules from the decision tree to decision models is not an issue and very easy according the the approach of Bazhenova and Weske, 2016, however, it is at the cost of accuracy of the results as decision trees is worse than of the other two approaches (Bazhenova & Weske, 2016).

2.1.3 Comparison of Approaches

As there are many differences between them it is very difficult to compare them and determine the best algorithm. The more algorithms that exist, the more important evaluation of these algorithm becomes (Jouck et al., 2019). Therefore, a framework was developed by Jouck et al., 2019 that evaluate and compare the the different decision mining techniques. The frameworks allows evaluation of techniques that discover both decision logic and decision models from event log data. In the paper of Jouck et al., 2019 results of the comparison between a mutually exclusive technique and an overlapping technique was successfully compared with the framework. However, more extensive evaluation is still required to include more techniques for comparison. For example, developed techniques from Bazhenova et al., 2016 and De Smedt, Hasic, et al., 2017 are not yet included in the framework (Jouck et al., 2019).

2.1.4 Decision Mining Frameworks

As Decision Mining became more popular and more techniques have been developed, the term Decision Mining has been used in a more diverse set of contexts (De Smedt, vanden Broucke, et al., 2017). To asses the definition of Decision Mining and distinguish the different techniques a framework named the Decision Mining Quadrant was developed by De Smedt, vanden Broucke, et al., 2017. The framework is shown in figure 3. The framework distinguishes the control flow and data model dimension first and afterwards reviews the input types and existing techniques. In the Quadrant the decision control flow is shown along the vertical dimension and the decision model maturity is shown along the horizontal dimension. Regarding the decision control flow, data mining techniques on one side are not aware of any dynamic aspects of the data, while process mining techniques on the other side derive a control flow of activities (De Smedt, vanden Broucke, et al., 2017). Regarding the decision model maturity, on one side no decision models are used, while on the other side they are (De Smedt, vanden Broucke, et al., 2017). The differences regarding the inputs for the mining approaches are the following. In the approaches of Q1 and Q2 the inputs focus solely on data attributes of single instances or on the sequential and concurrency aspect of the instances (De Smedt, vanden Broucke, et al., 2017). The approached in Q3 and Q4 use a mix of event-based and instance-based data instead of the simpler forms of event logs (De Smedt, vanden Broucke, et al., 2017). The difference between Q3 and Q4 is made by which aspect is prioritized. Approaches in Q3 use control flow data to determine the overall structure first, while approaches in Q4 prioritize the data perspective (De Smedt, vanden Broucke, et al., 2017).

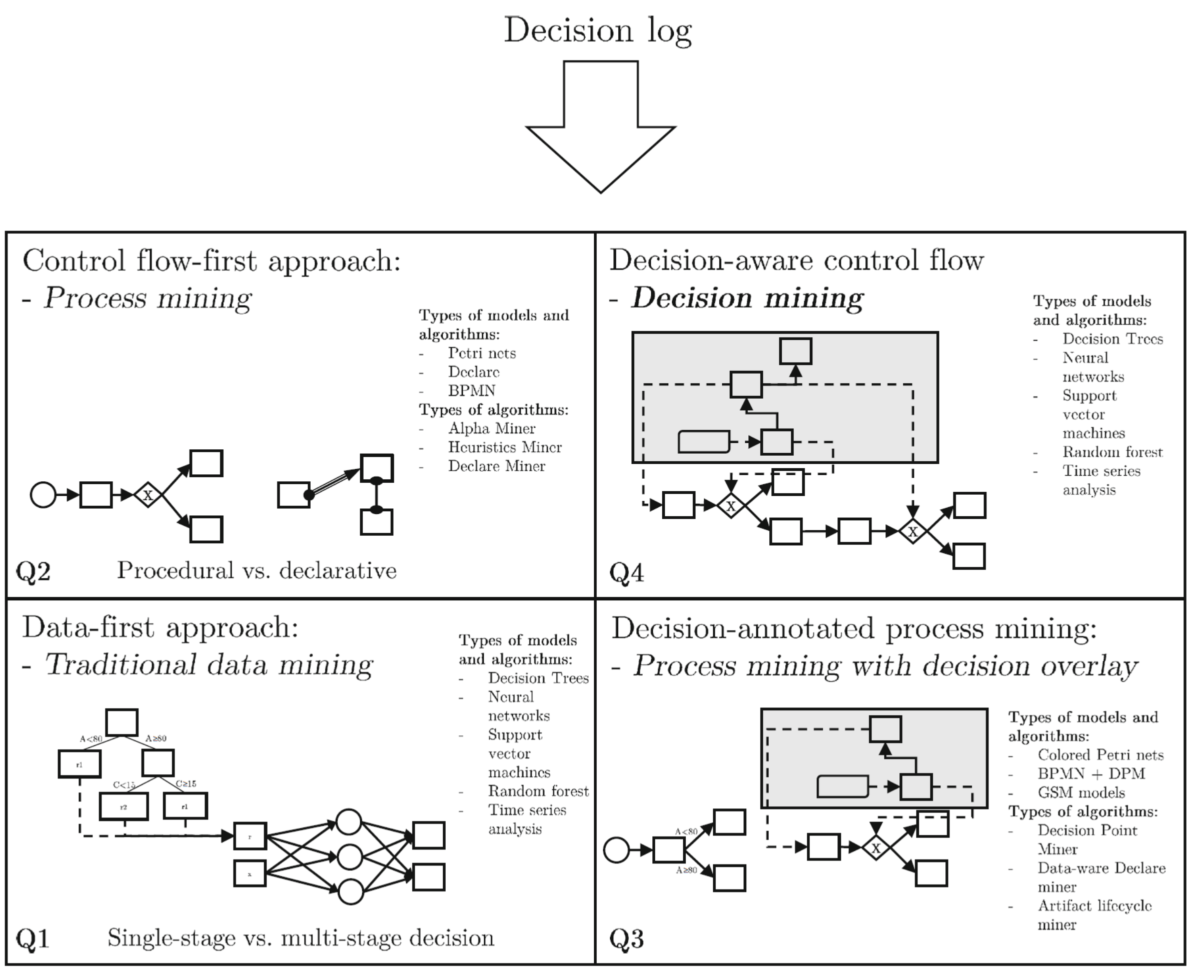

Figure 3: Decision Mining Quadrant (De Smedt, vanden Broucke, et al., 2017)

The research of Leewis et al., 2020 performed a literature study on the current state and the context of Decision Mining. An important part of the research is the research on the relations between Data Mining, Process Mining and Decision Mining. Decision Mining has two main influences. Decision Mining focused on mining Decision Points from business processes (decision-annotated) and a Decision Mining approach where more implicit data involved in the decision-making process (decision-aware) is taken into account Leewis et al., 2020. Both of these directions of Decision Mining have overlap and have a main focus for Process Mining techniques that utilize Data Mining techniques. These relations are shown in figure 4. To summarize the current state of Decision Mining a conceptual framework was created which is shown in figure 5. It includes Data Mining, Process Mining and Decision Mining all as elements of Business Intelligence (BI). In this case BI is described as a set of models and analysis methodologies that utilize data to generate information and knowledge for the support of decision-making processes Leewis et al., 2020. The big difference between Data, Process and Decision Mining is that Data Mining is the mining of aggregation patters, Process Mining is the mining of sequencing patterns and Decision Mining is utilizing annotated decisions Leewis et al., 2020. Each of these all use different algorithms and modelling standards to identify patterns based on statistical analysis.

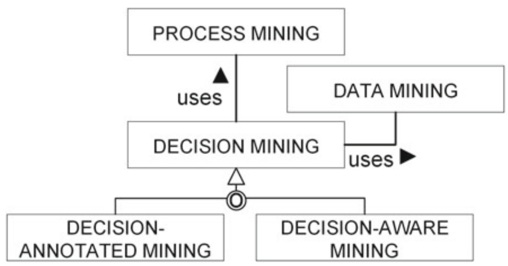

Figure 4: Decision Mining Relations (Leewis et al., 2020)

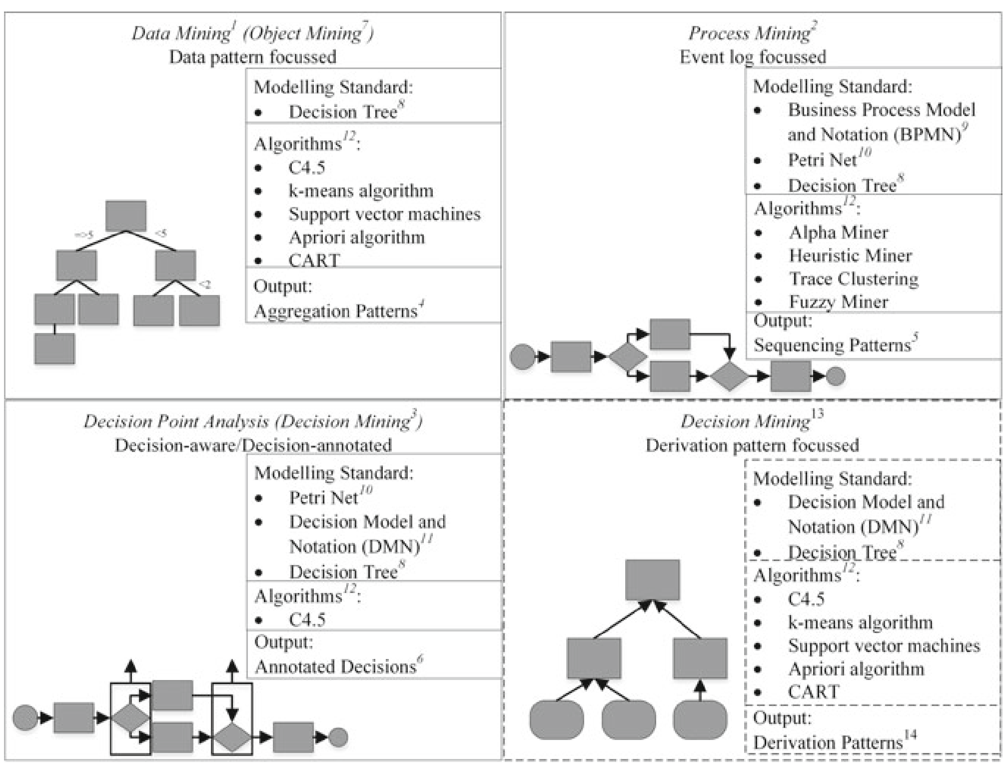

Figure 5: Conceptual Framework (Leewis et al., 2020)

2.1.5 Challenges of Decision Mining

To be able to implement Decision Mining in real-life processes two important challenges needs to taken into account for it to be successful (Rozinat & Van Der Aalst, 2006).

The first challenge is related to the quality of the data and the correct interpretation of their semantics. Event logs typically include a missing or incorrectly logged events which create noise in the data. The mining algorithms need to be sufficiently robust to be able to handle the noisy data, otherwise the results of the Decision Mining algorithm will be very poor. Therefore, data quality is seen as one of the biggest challenges for Decision Mining.

The second challenge is related to the correct interpretation of the control flow semantics of the process models when classifying the decision that have been made. In control flow semantics of real-life processes several scenarios may occur that provide problems. These problems are invisible activities, duplicate activities and loops.

- In the ideal situation all activity names correspond to their log event label and no activities have the same label. Besides that, there are no activities without a log event associated. However, for real-life event logs this might not always be the case as there can be activities that have no correspondence in the event log. For example, activities that are only added for routing purposes. These kind of activities are called invisible activities. The problem when these activities cannot be identified in the event log is that the occurrence of the next activity is not always sufficient in order to classify the options for the decision point.
- Another problem when trying to classify the options for a decision point is duplicate activities. In most of the real-life event logs, multiple activities have the same event log event associated. Therefore, it is not possible to distinguish their occurrences in the event log. As a result it is not possible to use these activities to make choices for a decision point.
- Loops in the process model are another problem for classifying the options for a decision point. Loops can occur in three different way to decision points. First decision points can be contained in a loop. This means that the decision point may occur multiple times in a process instance. This results in having more than one training examples for the decision tree algorithm. Decision point can also contain a loop. In this case activities may occur multiple times in the process. However, only the first occurrence is used to classify the decision point. Lastly, decision points can also be a loop. In this case the choice represents a post-test loop and every occurrence of the activities except the first ones are related to the decision point.

2.2 Decision Modeling

Decisions are important to organizations and understanding, modeling, managing and automating them became more important for many organization. Decision modeling therefore became crucial for organizations to improve their business outcomes (Taylor & Purchase, 2016). What exactly is decision modeling? Taylor and Purchase, 2016 came up with the following definition for decision modeling:

- *Decision modeling expresses how a decision should be made as a rigorous, verifiable model. It formalizes decision-making so it can be clearly and widely under-stood, managed and used effectively. Decision modeling supports the documentation of an organization's decisions such that they can be made consistently, improved over time and automated where appropriate.*

An important note on the terminology is the difference between Decision Requirements Modeling and Decision Modeling. A decision model consist of Decision Requirements Diagrams and Decision Logic such as Decision Tables. The term Decision Requirements Modeling refers to the models and diagrams while Decision Modeling refers to the overall approach (Taylor, 2016).

There are many reasons why the modeling of decisions can be useful. One of the most important reasons is that there is an emerging consensus that a Decision Requirements Model is the best way to specify decision-making Taylor, 2016. However, there are many reasons why decision modeling is useful:

- Decision Modeling allows decisions to be modeled separately from the process. Therefore, they do not clutter up the process anymore and allows easier changes to the business process. A separate but linked model allows for clarity in context (Taylor, 2016). It breaks down the decision into manageable building blocks that define required data for decision-making. Besides that, it lets you capture all policies, regulations and analytic insights that affect each part of the decision (Taylor, 2021).

- Specifying a Decision Requirements Model provides a repeatable and scalable approach to scoping and managing decision-making requirements for business rules efforts (Taylor, 2016). It provides a standardized approach to decision-making that can be used for both automated and manual decisions (Taylor, 2021).
- Decision Requirements Modeling is a technique that developes a richer and more complete business understanding earlier. It provides results in a clear business target, an understanding of how these results will be used and by whom they are used (Taylor, 2016) (Taylor, 2021). The graphical view of the decision-making helps making it easy to understand the results across different teams across the organization, such as business, analytics and IT teams (Taylor, 2021).
- Decision Requirements Modeling can be used very well to guide and shape analytics projects. It will decrease the reliance on specialists resources by improving information gathering, help teams asking the key questions, and enables much more effective collaboration throughout the organization by bringing analytics, IT and business professional together (Taylor, 2016).
- Decision Requirement Modeling can also be used to document analytic project requirements. That enables the organization to compare multiple projects on prioritization, reuse knowledge by creating an increasingly detailed and accurate view of decision-making and the role of analytics, and value information sources and analytics in terms of business impact (Taylor, 2016). Analytic teams usually try to integrate, clean and understand all available data, which is very difficult when organizations have large diverse data sets. Decision requirements models will answer the question which data might be useful and help the analytic teams to focus on the data that matters for the decisions (Taylor, 2021).
- Decision Requirements Models have several other benefits for the use of analytics. It ties the analytics and algorithms to the decisions they influence, the KPIs that needs to be improved, processes and systems that are impacted and the organizations that the decision point impacts. As a result of these ties, a solid business case for the analytics can be created (Taylor, 2021). Besides that, decision requirements models can also help identify the potential for analytics insights and machine learning techniques (Taylor, 2021). In the end the link between the analytic insights and the decisions are the most important gain of the decision requirements models. Instead of building the analytics and hope they make a difference, the role that analytics play in changing decision-making can be clearly identified (Taylor, 2021).
- Decision Requirements Models also play a role in the CRISP-DM framework. It is part of the business understanding phase of the framework as the models helps understanding and identifying the involved data to get the project off to an effective start (Taylor, 2021). Besides that, the model also is part of the evaluation loop of the framework, as it provides critical information to support deployment, such as organizations impacted and which processes and systems the new analytics-based decision-making must be injected to (Taylor, 2021).
- Decision Requirements Modeling is ideal for capturing business requirements for a dashboard and for driving the design, implementation and maintenance processes. Besides that, it also allows for dashboard implementation by tying data and knowledge requirements to presentation element (Taylor, 2016).
- Decision Modeling provides a framework that can be used by teams across the organization that works for both business professionals, IT professionals and analytic teams. As decisions can be tied easier to performance measures and the business goals of the project, it is easier for the teams to focus where they will have the most impact and to measure the results (Taylor, 2016).

2.2.1 Iterative Approach

Decision Modeling is a technique that consist of four steps that are performed iteratively: Identify Decisions, Describe Decisions, Specify Decision Requirements, and Decompose and Refine (Taylor, 2016). The cycle of the iterative approach is shown in figure 6. There is not a pre-defined number of iterations that is required for creating a decision model. The process is repeated until the decisions in the decision model are completely specified and everyone has a clear sense of how the decisions are made. Once this point is reached, a requirements document can be generated to act as a specification for the business rules implementation work or development of predictive analytics (Taylor, 2016).

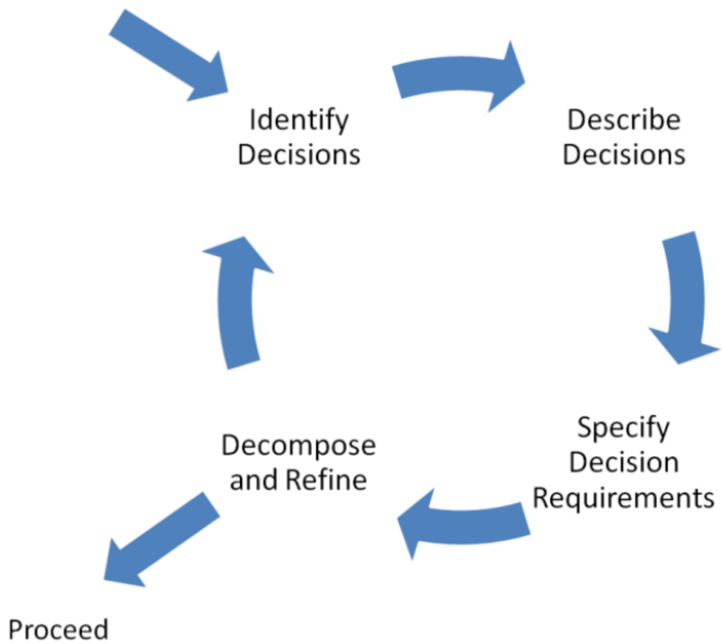

Figure 6: Iterative Decision Modeling Cycle (Taylor, 2016)

2.2.2 Suitable Decisions for Decision Modeling

Decision Requirements Modeling can be applied for any decision, however, it costs a lot of time and energy to create the model. According to Taylor, 2016 organizations do not make them for all decisions, but only when the following things are true for the decisions:

- Action oriented: Describing a suitable decision as action-oriented may seem redundant, as the essence of decision-making is to select from an array of possible actions, pick one and take it. However, some decision are more about getting answers than taking actions. The top-level or target decision in a project will generally be action-oriented.
- Value in defining: There needs to be value in defining how the decision will be made in advance. For decision that are made many times this is true as it improves consistency and makes it easier to share best practices. For decisions that are made once, it will be the case if they are very complex, are a point of contention and need to be transparent.
- Non-trivial: If decisions are trivial there is no value in creating decision models for them. However, for the decisions where many policies or regulations apply, a wide range of options is available and a lot of data needs to be considered, the decisions are likely to be non-trivial Also, when the way a decision is made must change often or is very dynamic, the decision will be considered as non-trivial. For these non-trival decision it could be worth modeling them.
- Measurable: The value of decisions must be measurable and it should be definable in advance. It should

be possible to identify the KPIs and metrics that are improved by a better decision or weakened by a poor decision. Defining this value in advance allows to create a baseline for the current performance of the decision and shows the value of the investment in improving the decision. Besides that, it can also be critical for understanding how a good decision looks like.

2.3 Decision Modeling and Notation

Decision Modeling and Notation (DMN) is a notation standard developed by the Object Management Group to create decision models. The DMN standard has been widely used and developed into the industry standard for Decision Modeling (Taylor, 2016). The aim of DMN standard has been to provide a notation that can be used to model decisions, so that decision-making can be readily depicted in diagrams, accurately defined by business analysts and automated (Omg, 2015). In existing modeling standards decision-making is addressed from the perspectives of Business Process Models and Decision Logic. The intention of DMN is to provide a third perspective of Decision Requirements Diagram, that forms a bridge between Business Process Models and Decision Logic (Omg, 2015). The relationship between the three aspects are shown in figure 7. Decision Requirements Diagrams and Decision Logic provide a complete decision model that complements the Business Process Model by specifying in detail the decision-making in the process tasks Omg, 2015. The use of graphical models in the DMN standard is a big advantage. Since the models are not technical and are made in a common notation, they are easy to understand for all business users (Chiheb et al., 2019). This allows easier communication and collaboration on the requirements and outcomes (Taylor, 2021). This easier communication and collaboration is one of the basic elements that support the decision-making and help making smarter and faster decisions (Chiheb et al., 2019).

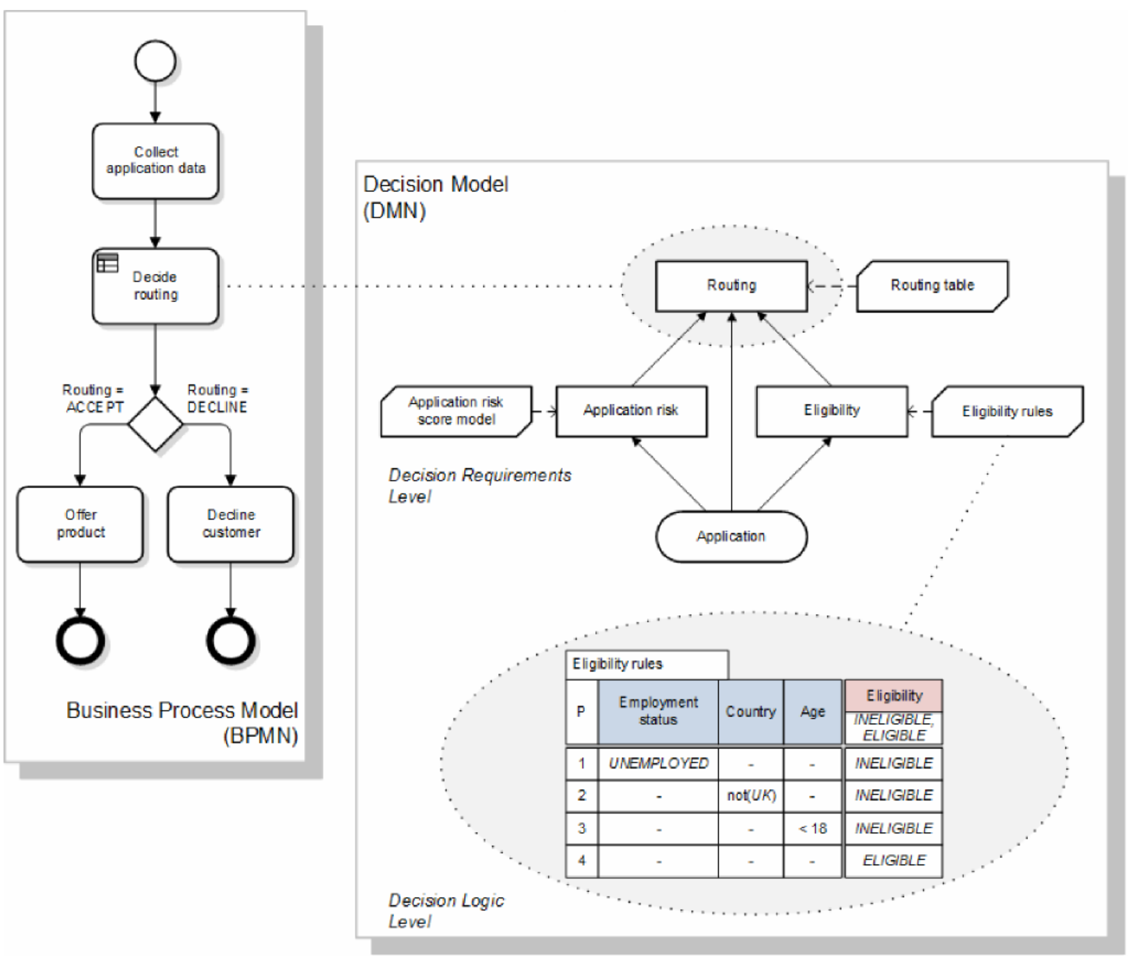

Figure 7: Relationship DMN (Omg, 2015)

Decision modeling is carried out by business analysts in order to understand and define the decision used in a organization, these decisions are operational decisions most of the time rather than strategic decision-making. In this context three uses of DMN can be discerned: For modeling human decision-making, for modeling requirements for automated decision-making and for implementing automated decision-making (Omg, 2015). These contexts are not mutually exclusive as they are generally combined in large process automation projects. According to Omg, 2015 the combination of these contexts goes in the following way. First, the decision-making within the existing process is modeled. Based on this a 'as-is' situation is created that provides the baseline for process improvement. Next, the process is redesigned to make the most effective use of both automated and human decision-making. This model provides the 'to-be' specification of the required process and decision-making it coordinated. The 'as-is' and 'to-be' models can be compared to indicate the requirements not just for automation technology, but also for change management. Finally, the 'to-be' model will be implemented as executable system software.

3 Methodology

In this chapter, it will be explained how this research will be performed and how the different research questions are answered. It will also include how the knowledge from the literature review in the previous chapter will be used. The bigger picture of this research will follow the steps of the Problem Solving Cycle of van Aken et al., 2012. The Problem Solving Cycle is very useful for solving typical business problems. Since the problem of Fokker Services that will be investigated with this research is a typical business problem, this cycle is suitable as an approach for this research.

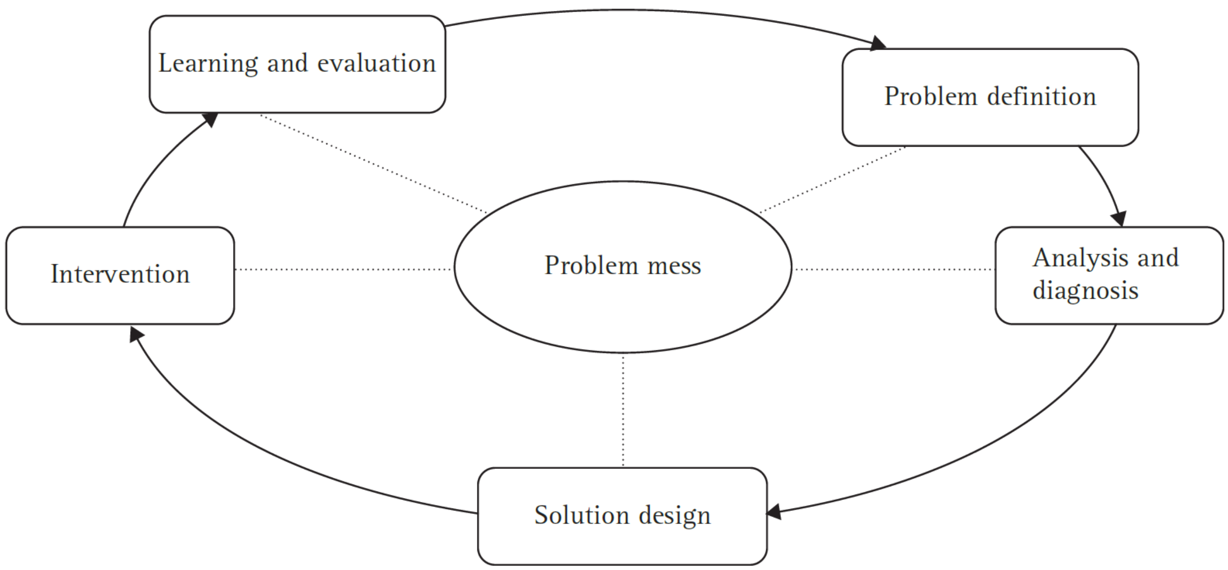

Figure 8: Problem Solving Cycle (van Aken et al., 2012)

The problem-solving cycle consists of 5 steps as can be seen in figure 8. The first step is Problem Definition. In this step, it is the goal to identify and structure the business problem. The problem for this research has already been described in first chapter of this report. Therefore, this step has already been completed and does not require much more tasks in the future of this research. Besides that, a literature review has been performed on the topics of Decision Mining and Decision Modeling with the DMN standard. The literature review it is explained what the topics include, which advantages they have, how they can be used, and which challenges must be considered. Therefore, the literature review provides a lot of useful information that needs to be considered when Decision Mining and Decision Modeling are used. As described in the first chapter the research objective is to improve the operational decisions within Fokker Services and investigate how Decision Mining and Decision Modeling can be used for that. The literature review provides a lot information that needs to be considered and is therefore important for the remaining steps of this research. The remaining four steps in the problem-solving cycle will be performed by answering the different sub-questions formulated in the first chapter. In the next part for each sub-question will be described to which part of the cycle it belongs and how the question will be answered.

- What are the important decisions within the process that could be improved?

The first sub-question of this research is part of the analysis and diagnosis step. The main goal of this sub-question is to identify the different decisions within the services of CMRO and Exchange Programs. As described in the previous chapter, it will not be possible to improve all decisions within these services. Therefore, a selection must be made for the decisions that are going to be improved. Fokker Services did not try to actively improve the decision in a way they would like to do with the help of this research. So, it is expected that almost all decisions could be improved in some way. The decisions that will be selected for this research will be based on how much influence the decisions have on the process. Besides that, Fokker Services aims not only to improve the decisions, but also automate them in the future. Therefore, with the final selection of the decisions it will be taken into account whether the decisions are suitable for improving and automation. Before, the final decisions that will be considered for improvement, the important decisions need to be identified first. To find out which decision will be the most important within the process, information about the process and decisions that were collected in a previous assignment will be used as a baseline. Additionally, an interview will be performed with the Manager Program Management will be performed. This interview will mostly be a semi-structured interview. The goal is to get more detailed information about the important decisions within the process and semi-structured interviews allow to ask more in-depth questions about the important information. The Manager Program Management is the person with the most knowledge of the processes within the organization and how decisions are used within the processes. Therefore, he is the most ideal person to identify the important decisions and provide information about them.

- Which data is required for the decisions and what is the quality of this data?

The second sub-question is also part of the analysis and diagnosis step. When it is known which decisions are important and need to be improved it is important to investigate which data is required to make the decisions. For this required data the quality of the data needs to be checked. According to Rozinat and Van Der Aalst, 2006 the data quality is one of the most important challenges when using decision mining. Since this research aims to improve the decisions with the help of decision mining, it is important to know that the data is of enough quality. Martin, 2021 provides an introduction to data quality specifically for process mining. According to Martin, 2021 checking the data quality and solving potential issues consist of three phases. First, the different data quality taxonomies are defined which describe the potential data quality issues. The next step is to perform the data quality assessment to detect the data quality issues. Finally, several heuristics are introduced that can be used for data cleaning to solve data quality issues. To answer this sub-question these phases will be performed to ensure the data is of sufficient quality.

- Which of the decisions to be improved can be fully automated, and which decisions remain (partly) dependent on human knowledge?

This sub-question will be the last part of the analysis and diagnosis step and provide a starting point for the solution design. Before a solution design can be made the last analysis needs to be performed. In the current process of Fokker Services, none of the decisions are automated and depend heavily on employees' knowledge. Therefore, automating the decisions could be a very useful option for the solution design. The only downside is that there could be several limitations or constraints that make it not possible to make all decisions fully automated. This sub-question will investigate when decisions could be automated and when not. It will include the knowledge about the quality of the required data that was gathered in the second sub-question. The main reason for this is that the data quality has a big influence on the possibility to automate the decisions with decision mining (Rozinat & Van Der Aalst, 2006). Not only is it the goal to investigate when a decision can be fully automated and when not, but also to conclude this for the selected decision from the first sub-question. Once it is concluded for the selected decisions to which extent it is possible to automate them, a solution design can be developed to improve the decisions, which is the main goal of this project. Lastly, it is important to mention that as described in the research objective, improving decisions means that decisions are made faster and sub-optimal decisions are not possible.

- Which changes in the process are required to improve the decisions, and can these be achieved with decision mining techniques?

Once it is clear to which extend the decisions could be automated, the solution design can be developed. The main goal of this sub-question is to create an ideal situation for the decisions of the process that can be compared with the current situation. By answering this sub-question the side objective of this research should be achieved as well. That objective was to investigate the effect of data quality on the possible improvements. Fokker Services only has a certain amount of data that can be used for this research and there is no guarantee that this data will be perfect in terms of data quality. Therefore, it will be tested to what extend the ideal situation can be created based on decision mining techniques. This way this sub-question helps achieving that side objective of this research. When the ideal situation is completed, it is important to indicate the required changes from the current situation as this will be necessary to answer the last sub-question. Finally, it should be determined how these required changes can be achieved as it can help Fokker Services with future improvements on other decisions.

- What is the impact of the improved decisions?

The last sub-question is related to the last step of the cycle, which is the evaluation of the improvements. With the answers of the previous sub-question it is known how the improved decision looks like with the required changes from the current situation. However, in the end the main goal to improve the decisions for Fokker Services is to achieve better performance of their processes. More specifically achieve a lower time to handle the customer orders and higher service levels. Therefore, the time for order requests and service levels needs to be evaluated again with the improved decisions that will be compared with the initial situation to find the impact of the improved decisions.

4 Automation of Decisions

This chapter will investigate the requirements for decisions to be able to automate them. However, before it is possible to determine this, it needs to be investigated how automated decisions can be achieved and which possibilities there are. In the first part of this chapter, an approach specifically for automating and improving decisions will be discussed. After that, the second part of this chapter will describe the different criteria and requirements that are required for decisions to use them for the approach. This information will be used in a later chapter for the decisions to determine to which extent they could be automated and answer sub-question Q3.

4.1 Link between DMN and automating decisions

As described in the literature review of this research, Decision Modeling with the DMN standard helps model the requirements for automated decision and the implementation of automated decision-making. However, besides the decision models, there is more needed to be able to fully automate the decision-making. The use of DMN for modeling the requirements of automated decision-making prescribes the decision logic of decisions (Omg, 2015). The decision logic must be complete for full automation of decisions. This means that the decision logic can provide a result for any set of values of the input data (Omg, 2015). However, requirements can also be modeled for partly automated decisions. In this case, some decision-making remains the preserve of employees and which is much more common (Omg, 2015). In this case, also the interactions between human and automated decision-makers are modeled. This makes it possible to separate the tasks in the business process between human decision-makers and automated decision-making. For example, automated business rule tasks may forward some of the cases to the human reviewer. Then the decision logic needs to be specified completely for the automated tasks, while it could be left unspecified for the reviewers' task (Omg, 2015). However, it is important to note that the decision requirements model only provides a specification of the requirements for automated decision-making. It doesn't automate the decision in any way by itself. Therefore, the decision models still need to be executed (Omg, 2015).

4.2 Decision Services

One way to implement these is by using "decision services" deployed from a Business Rules Management System (BRMS) that is called a Business Process Management System (BPMS). A decision service encapsulates the decision logic supporting a Decision Requirements Diagram (DRD) that is part of the decision model (Omg, 2015). So when the decision services receive a set of input data, it will evaluate the decision it needs to make and provide the results of the option that was selected. According to Taylor, 2017 Decision Services is the implementation of a decision. Besides finding the best decision for a case, a decision service also makes the decision reusable and widely available (Taylor, 2017). Decision Services are essentially business services in a Service Oriented Architecture (SOA) that deliver an answer to a specific question (Taylor, 2017). Questions could for example be "How should we handle this claim" or "What is the right discount for this order?". Generally, these services do not update information and since they do not make permanent changes, they can be used to answer these questions whenever it is needed (Taylor, 2017). As mentioned before decision services are deployed from a BRMS. In many of the cases only a BRMS is required for decision services, they also support the integration of predictive analytics (Taylor, 2017). An overview of how these elements come together to provide Decision Services for an application portfolio is shown in figure 9.

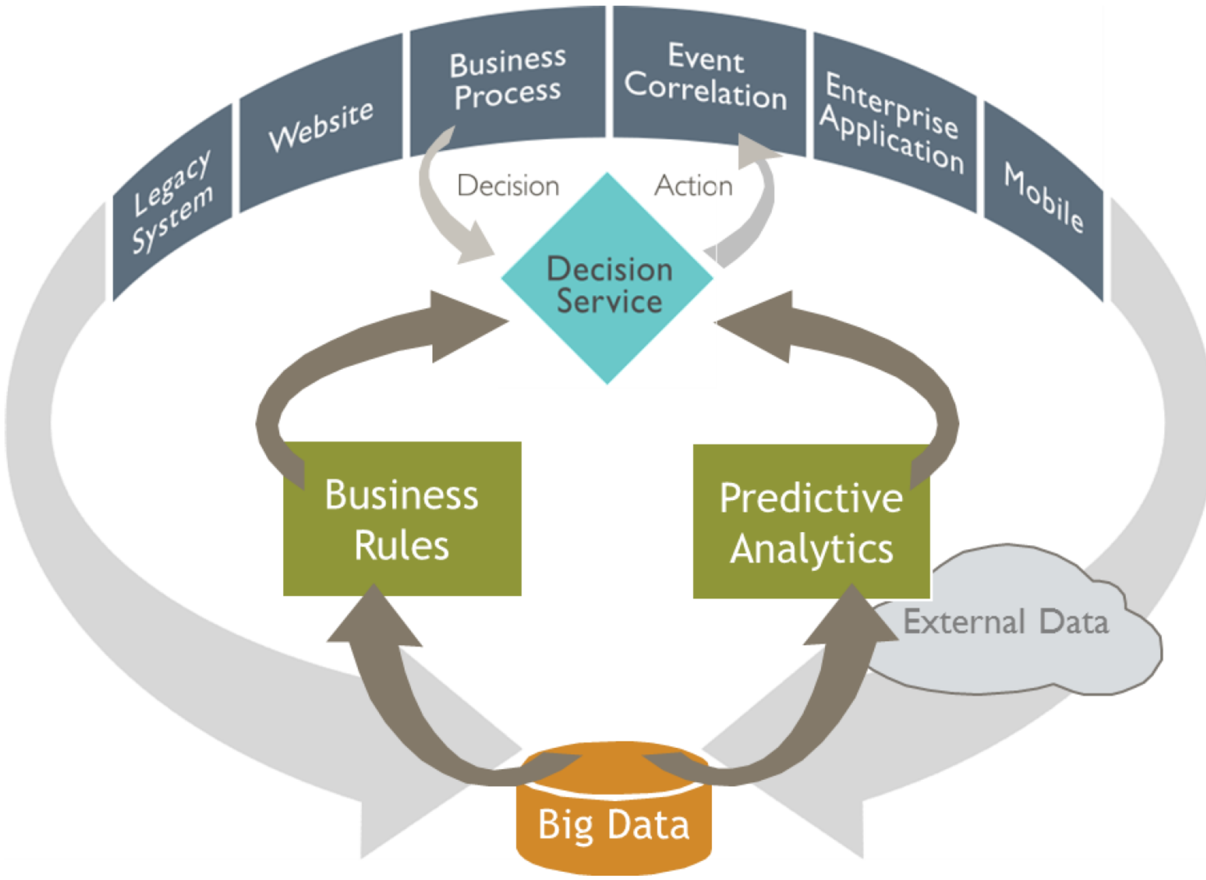

Figure 9: Decision Services (Taylor, 2017)

4.3 Decision Management Approach

The creation of decision models and implementation of decision services to try to automate and improve decisions within organizations is part of an approach named Decision Management. Taylor, 2014 defined Decision Management in the following way:

- *Decision Management is an approach that improves day to day business operations. It increases an organizations business agility and adaptability by making its system easier to monitor and change. It puts data to work improving the effectiveness and profitability of every action. It is a proven framework for applying AI technologies such as business rules, machine learning, predictive analytics and optimization.*

Decision Management focuses on the decision that creates value in the business, recognizes these decisions as reusable assets, and makes them widely available via an SOA. (Taylor, 2017). Decision Management consists of the following three phases: Decision Discovery and Modeling, Decision Service Definition and Implementation, and Decision Measurement and Improvement and is shown in figure 10 (Taylor, 2017). The first phase aims to find the decisions that matter for the business and model them with the DMN standard. The second step is about the implementation of the decision services, which are built with BRMS and can be enhanced with results of data mining and predictive analytics. The last step should be about monitoring and consistently improving the decision-making to deliver increasing value over time.

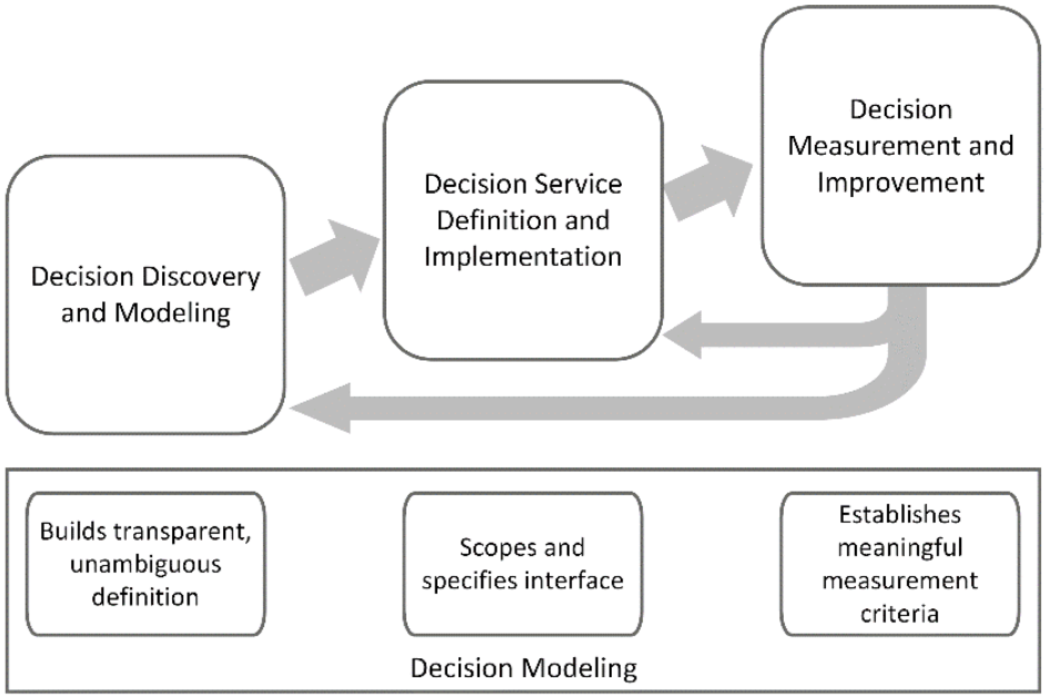

Figure 10: Three Phases of Decision Management (Taylor, 2017)

What is also included in the overview of the Decision Management Approach in 10 is the role of Decision Modeling. As described earlier in this section there is a strong role for Decision Modeling with DMN as it is used to prescribe the decision logic of automated decision-making. However, Decision Modeling has a role in each phase of the Decision Management approach. In the first phase Decision Modeling builds a transparent definition of the decision at the beginning of the project before rules are gathered, providing a decisions first, top-down view (Taylor, 2017). In the second phase Decision Modeling scopes and specifies the interface and content of Decision Services. The decision model structures the internals of the decision service and provides a business-focused structure for the rules in the service (Taylor, 2017). In the last phase Decision Modeling establishes meaningful and business-centric measurement criteria and structures the monitoring and improvement activities, closing the loop back to the original decision (Taylor, 2017).

According to Taylor, 2014 four aspects of digital decisioning drive organizations to adopt new Decision Management specific technologies. These capabilities can be used to completely automate a decision in every circumstance. However, more commonly they are focused on just some of the elements of a decision, using a decision model as structure, and delivering both decision support and decision management capabilities.

- Managing decision logic for transparency and agility. Organizations typically adopt a Business Rules Management System to manage decision logic more effectively.
- Embedding machine learning for analytic decision-making. Organizations use machine learning algorithms and predictive analytic workbenches, packaged analytic models, and other techniques to turn their historical data into usable analytic insight for decision-making.
- Selecting the best alternative given real-world trade-offs and simulating results. Some organizations adopt constraint-based optimization tools while others use trade-off matrices, genetic algorithms, and other analysis techniques.
- Monitoring and improving decision-making over time. Decision Management technologies offer simulation, impact analysis, integration with performance management, and more to support ongoing decision analysis.

4.4 Business Rules

BRMS plays an important role within the implementation phase of Decision Management as business rules are the primary element of the decision in Decision Services. Business rules present the expertise, tribal knowledge, regulations, and policies that drive a business (Taylor, 2017). Business rules create a language that both business and IT employees can understand, which is essential for effective Decision Management. They are maintained in a repository or catalog that is updated by both business and technical users (Taylor, 2017). Business rules have several other usages as they are also used in workflow, to manage data quality or control the user interface. However, these effective uses are different than the ones that are built into a Decision Service. The rules used for Decision Services are truly about the business and how it should act. So they are independent of a company's current databases, systems, or processes (Taylor, 2017).

There is a strong link between Decision Models and business rules. As described in the previous subsection decision models scope and specify the interface and content of Decision Services. The decision model acts as a framework for capturing and defining the business rules, keeping each set of business rules focused on a particular sub-decision (Taylor, 2017). Knowledge sources in the decision models show where the business rules can be found and input data in the decision models show how the data is used by these business rules to make the decision. This all is managed in the BRMS (Taylor, 2017).

According to Taylor, 2017 using a BRMS to encode business rules that has several advantages that are critical for managing decisions, even when equivalent software code would be simple:

- The syntax is clearer to a non-programmer so that decisions built with them can be managed by non-programmers
- The business rules can easily be reused across multiple systems that use the decisions
- The rules are independent. No sequence is implied making it possible to edit them, and thus change decisions, to respond to changing business circumstances, without unintended consequences.

4.5 Suitable Decisions

There is not much information available on the requirements of whether decisions are suitable for automation in general. In the subsection about Decision Modeling in the literature review of this research, a set of requirements was given for decisions to be suitable for Decision Modeling. However, with only Decision Modeling decisions cannot be automated. Therefore, the Decision Management approach is needed, as described in the previous subsection. Decision Management Systems are focused on improving and automation specific decisions and Taylor, 2012 provides an extensive list of characteristics that are required for decisions to be suitable to be implemented in Decision Management Systems.

4.5.1 Repeatability

Repeatability is by far the most important criterion for a suitable decision according to Taylor, 2012. If a decision is not repeatable, there is no value to develop a Decision Management System for that decision to automate it. For a decision to be repeatable Taylor, 2012 describes the four following tests that the decision must pass:

- It is possible to say when the decision will need to be made. An organization should know when a repeatable decision must be made. For example, the decision could be needed for a process to be complete or move to the next step. But many other decisions could be made. Another example is the possibility that decisions are made on regular basis such as hourly, daily, or weekly. However, for these decisions, it is known in advance that they are needed. Of course, it is also possible that it is known decisions must be made several times during the year without knowing when exactly they need to be made. These decisions may still be repeatable if the organization can define the circumstances for when the decision needs to be made.
- Each time the decision is made, the same information is available, considered, and analyzed. The Decision Management Systems need to work with a consistent set of information, so every time a decision is made a consistent set of information must be available. It does not mean that an identical set of information is presented to the Decision Management Systems. However, it does mean that a set

of information can be defined that is a superset of the information that will be presented each time.

- The set of possible actions remains consistent between decisions. Every time a decision is made it selects from a set of possible actions. Even though different actions can be selected for different occurrences of the decision, the set of possible actions should remain consistent.
- How the success of these actions is measured in terms of business outcomes remains consistent. If it is not possible to say when a decision was good or bad, decision-making cannot be improved. Therefore, it is essential to understand how the outcomes of the decisions can be measured. This is true for all decisions, even when they are just one-off ad-hoc decisions. Also, it is very important that a consistent measurement is used to measure the success of a decision.

4.5.2 Non trivial

The next criterion for suitable decisions is that the decision must be nontrivial, which is similar to the requirement for Decision Modeling. If Decision Management Systems are used for decisions it requires an investment of both time and money. To see a return on this investment, the decision to be automated must have a degree of complexity. Typical drivers of complexity are policies and regulations, the need for domain knowledge, the need to analyze large amounts of complex data, the need to select from many different possible outcomes and the need to trade-off competing objectives. If none of these drive the complexity of the decision, the decision-making approach needs to be updated so often that it should be considered non-trivial on that basis. Of course, it is not needed to test the decision for all of these drivers. Having just very complex regulations or very difficult trade-off can already be enough for a decision to be non-trivial. Also if the individual drivers are not complex, a combination of all these drivers can create enough complexity for a decision to be non-trivial.

- Policies and Regulations are the most common drivers of complexity. Policies and regulations can constrain decision-making quite a lot. Organizations usually have several policies to ensure consistency and avoid known problems or pitfalls. Besides that, organizations need to deal with regulations that are imposed by governments. Making sure that all decisions are always compliant with the policies and regulations can be quite a challenge that makes the decision-making very complex.
- Domain Knowledge and Expertise can make decisions very complex. Making decisions for the first time is usually much harder than when you made it many times before. More experience makes it easier to make the decision quicker with higher chances that good decisions were made. Decisions like this require domain knowledge and expertise as they require a deep understanding of a particular domain and are therefore nontrivial.
- Analysis is required for many decisions. If a decision involves making the "best" or "most appropriate" selection, some judgment or analysis is required. Only after the analysis is performed decisions can be made. If the analyses are repeatable and well-defined, they can be part of the decision without it requiring manual decision-making. Analysis can be both current information, for example all the data in an order, or historical information. Any need for analysis will tend to make the decision nontrivial.
- Large amounts of data in decisions make it very likely for the decision to be nontrivial. Some decision needs to consider many different data elements of data attributes to choose the final action. For example, if hundreds of elements need to be checked, the decision can become very complex.
- Large amount of actions in decisions makes the decision very complex. Generally, decisions with a small set of possible actions are much simpler than decisions with a large set of actions. It is often said that people have difficulties with a list of more than seven items. So if a decision has a set with more possible actions, it is very likely to be a suitable decision. Of course, decisions with a small set of possible actions can also be very complex. Decisions that have only two possible actions yes and no are not always easy decisions.
- Trade-offs that must be made can make decisions very challenging. In that case, the possible options are all not perfect and have different positive and negative effects. These must be balanced against their value which is usually very complex. The need for trade-offs does mean that in most cases the decision is nontrivial.

4.5.3 Measurable Business Impact

Another important criterion is the possibility to measure the impact the decision will have on the business. As described earlier, building a Decision Management System for a decision requires an investment. If there is no return on this investment, there is no point in doing it. Repeatable and nontrivial decisions both are likely to show a return, however, this is not sufficient to be suitable. For a suitable decision, it must be possible to see the cost of the bad decisions and the value of the good decisions. Besides that, the organization must be able to see the impact of a decision related to the measurement framework of the organization. In some cases, it may be difficult to see the impact of the decisions as the effect is not apparent immediately. For some decisions, it can take months or even years to see the impact of improving how a decision is made. Besides that, it could also be difficult to measure the impact of a single decision. For example, it is not reasonable to look at the improvement of a single decision, but rather at a cumulative improvement over time. In some cases, it is possible to see that a decision is very important and what impact is has on the organization, without being able to measure the exact impact. Even for these cases, it is best to first create an infrastructure that makes it possible to measure the exact impact, before the Decision Management System is built to improve and automate the decision.

4.5.4 Candidates for Automation

There is a last criterion for suitable decisions if the decisions are repeatable, nontrivial, and has a measurable business impact. As long as the organization does not accept the decision as a candidate for automation, it will never be a suitable decision. If the organization believes that the decision requires human judgment, it will not be suitable to build a Decision Management System. It will be a waste of investment if the built Decision Management System will be not be used. Most decisions depend on other decisions and can therefore be compared into multiple smaller and simpler decisions. When top-level decisions may not be suitable candidates as the organization is unwilling to have them automated, the low-level decisions still can be. The reason for this is that low-level decisions usually have a more limited scope that makes it easier for the organization to be comfortable with the automation of the decision. Besides that, if decisions are not a suitable candidate for automation, does not mean they will never be suitable. Decision-making within organizations consistently evolves and the enthusiasm for automating decision-making can change over time a lot.

5 Current Situation & Decisions Identification

In this chapter, the current situation for the relevant processes of this research will be described. Next, the decisions in these processes will be identified and finally, a selection of decisions will be made. These selected decisions are the ones that will be improved throughout the research. With this decision identification sub-question Q1 of this research will be answered. The overview of the current situation of the processes and including decisions form a baseline and starting point for this research. At the end of the research, the performance of this current situation will be used to investigate the effect of the improvements.

5.1 Process Overview

As mentioned in the first chapter of this research, the BPMN models creates in the previous internship assignment are used as a base for this research. Besides that, the scope of this research will be limited to CMRO Service and Exchange Programs. Other processes within the organization are not included in this research. The process of CMRO Service and Exchange programs were both identified by creating an information flow model based on BPMN for each process. BPMN provides a rich and generally broad graphical notation for modeling business processes (Loshin, 2013). So, it fits perfectly for the required overview of the processes. Besides that, BPMN is considered the modeling standard for business processes by both academia and industry (Van Der Aalst, Mylopoulos, et al., 2012). Therefore, BPMN will be used throughout this research as well and no completely new models were required. However, that does not mean that no changes at all are required. There are some important remarks on processes of CMRO Service and Exchange programs that are important for this research. They will be discussed next.

5.1.1 Updated to BPMN models

One important remark on the processes of the previous work is that the scope is limited to the repair process within the organization of Fokker Services. This is only a part of the services of CMRO and Exchange Programs. As the name already states, CMRO service includes not only the repair of components but also the maintenance and overhaul of components. This research will not be limited to only repair orders and will include all types of orders within the CMRO service.

Even though, the model was created specifically for a repair order, including the maintenance and overhaul orders as well does not make the model obsolete. There are several reasons why this is not the case. First, the detailed information about the task where the actual repair has been performed has been kept outside of the scope. The process includes that the repair action is performed in the repair shop. However, an entire sub-process could be created for all tasks and steps that are performed, and the focus of this project as well it to look at the bigger picture of the process. Of course, different actions may be performed with an overhaul compared to a repair. However, the other tasks in the process are the same and need to be performed regardless of whether it is a maintenance, repair, or overhaul request. Secondly, the repair shop where the actions are performed is the same for all activities. Fokker Services has only one shop in Schiphol that needs to perform all activities for the organization. Within the shop, there are no separate departments for the different types of activities. So it is very well possible that certain types of requests have to wait for other types of requests. So there is a certain dependency between the different types of requests within the CMRO service. Lastly, it is not always the case that only one type of activity is performed. For example, when a component is sent to the repair shop for repair, it is possible that during the repair several other issues were detected that require extra work. In these cases, an overhaul is required for the component as well. So while it has been a repair request throughout the process, overhaul activities were performed in the shop. These requests where other activities need to be performed in the shop than initially planned are no exceptions in the process and happen quite often. Therefore, it is not an issue to use the process for all orders in the CMRO service.

Another important remark is that the CMRO service has a little overlap with the service of Modifications. Modifications are not only made to actual aircraft, but also single components. These modifications are also performed in the same shop as the repairs. In many cases, modifications are combined with a repair or overhaul. Therefore, the modifications of components also become part of the CMRO service and undergo the same process as components that require normal, maintenance, repair, or overhaul. These orders will also be included in this research for similar reasons that were mentioned in the previous paragraph. Other than that the service of Modifications will be kept outside of the scope.

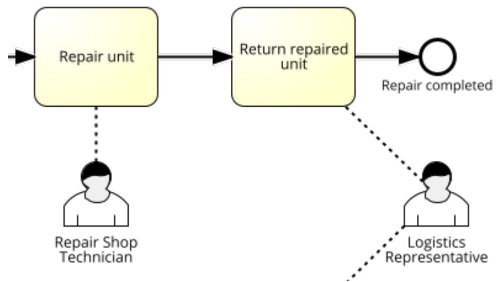

Figure 11: Original BPMN Terms

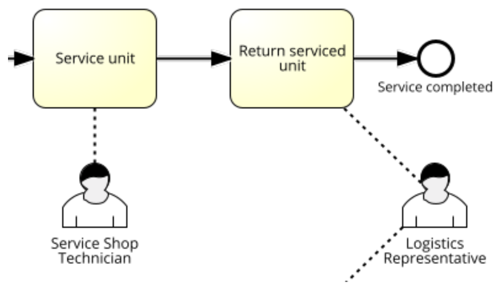

Figure 12: Updated BPMN Terms

As the created models for the repair process are not obsolete when all services are included, some minor changes are necessary. The changes that need to be made to the model are all related to the naming of the tasks, events and resources. An example of how required changes is shown between figure 11 and figure 12, which shows a small part of the BPMN model. In the initial model of figure 11 the term repair was used in the tasks and event and to indicate the resource. Since all other types of service are included in this research, a more general term of service was used in the BPMN model as can be seen in 12. For this research all terms that needs to be changed were updated. However, it didn't have any effect on the routing or flow of the model, that remained the same. Both the model or the CMRO Service and the Exchange Programs were updated. The complete and updated BPMN models are included in the Appendix of the report. Appendix B includes the BPMN model for CMRO Service, Appendix C includes the model for Exchange Programs and all sub-processes are included in Appendix D.

5.2 Decision Identification

The process contains several decisions and some of them we already modeled in the previous work. Similar to the BPMN model, these decision models can be used as a base for this research. However, as described in the previous sub-section, some small changes were made to the process as in this research. So similar to the BPMN model, the decision models must be updated accordingly as well. The four identified decisions are 'Check Requirements Quote', 'Create Price Quote', 'Decide Alternative Options' and 'Decide on Service'.

The decision 'Check Requirements Quote' is a relatively simple decision that checks whether a price quote is required for the request. This decision will be made in both CMRO Service and Exchange Programs. However, the outcome can be different as it is not always required for requests of an Exchange Program. In that case the contract conditions needs to be checked by the employee. Since this information is not generally available this decision depends a lot on the knowledge of the employees. An overview of the updated decision model is shown in figure 13.

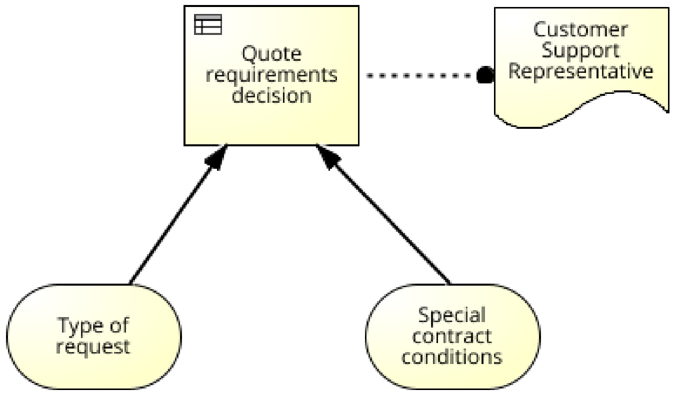

Figure 13: Check Requirements Quote

The next decision of 'Create Price Quote' is one of the most complex decisions as it requires a lot of information. In this decision the exact price for a quote needs to be determined. The fact that prices differ a lot between different components, or even differ over time for the same components, makes it even more difficult to determine the best price. This is another decision that depends heavily on employees and is very time consuming. Besides that, the decision outcome has a big influence the profitability of an order, which makes it a very important decision. The updated decision model is shown in figure 25.

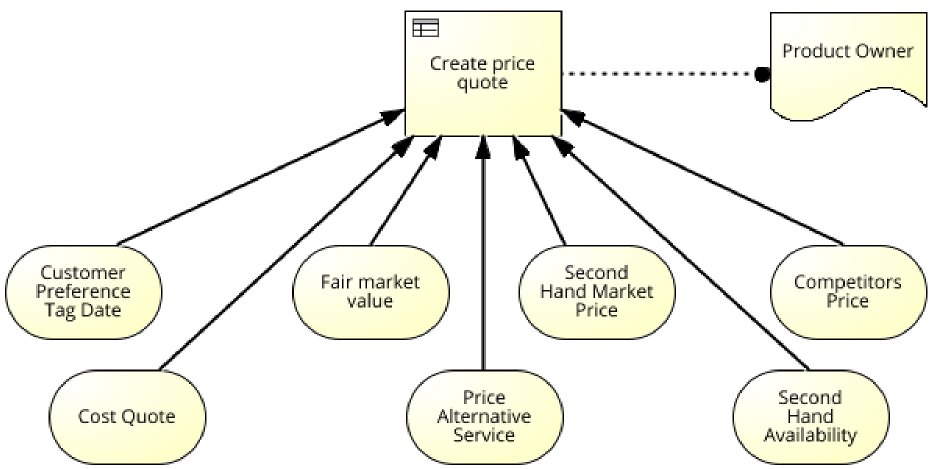

Figure 14: Create Price Quote

The decision 'Decide Alternative Options' only occurs within the process of the CMRO service. Depending on the outcome of the previous decisions it may not be necessary at all. Only when the price quote that was created in the previous decision did not pass the sanity check, this decision is required. Then it simply checks whether an alternative option is available, which is a relatively simple decision. In two sub-decisions it will be decided if alternatives are available. If an alternative is available it will always be selected as an alternative option. The updated decision model is shown in figure 15.

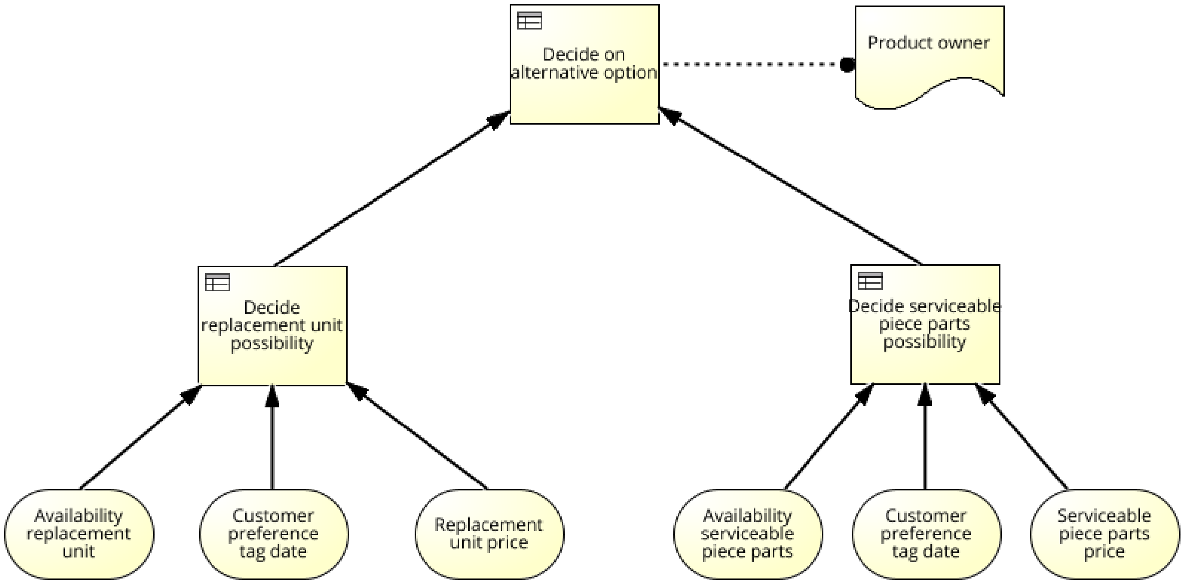

Figure 15: Decide Alternative Option

The last of the already identified decisions is 'Decide on Service'. This is a decision that only occurs in the process of Exchange Programs, but it is the most important one in this process. In this decision Fokker Services decide what they are doing with the unserviced or failed components they receive back from customers. Simply servicing all components is not possible and not servicing any will cause issues for future stock levels. Therefore, several important trade-off needs to be made that require a lot information. That makes it very time consuming again for the employees. An overview of the updated decision model is shown in 23.

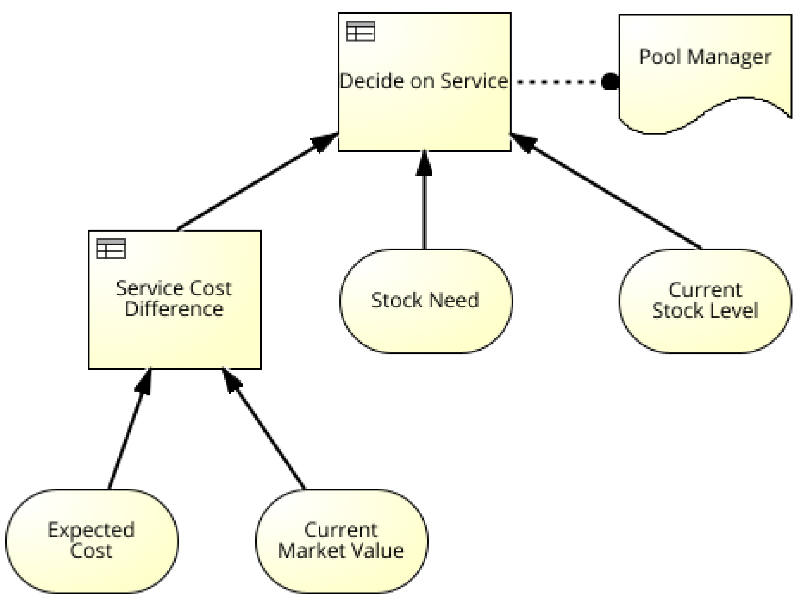

Figure 16: Decide on Service

Besides the four decisions that were modeled, there are two other decisions in the process of Exchange Programs that were identified. The decisions 'Decide on option' and 'Decide on alternative option' were kept outside of the scope of the research. In these decisions, Fokker Services decides which action they will take with the received order for an exchange unit. The main goal of the process is to deliver the components on time to the customer. However, the components do not always need to be sent immediately to the customers. There are agreements with customers on the delivery time, service level, and many other factors that influence the decision. For example, there are certain cases where it is beneficial to not send the component immediately to the customer, even if risks exceed the agreed delivery time. The biggest issue regarding the required information for the decision is that Fokker Services does not have many general agreements with customers. Many different customers have different contract conditions which they agreed on with Fokker Services. As a result of this, the decisions become very complex as the employees need all this information to make the best decisions possible. However, the biggest issue is that information regarding the contract conditions is not widely available. To collect this information all contract needs to be investigated which takes a lot of time and does not fit well within the scope and the goals of this research. Therefore, this decision will not be included in the remaining of this research and not be selected as one of the decisions that will be improved.

Lastly, there is another decision that was not identified as a decision. In the sub-process 'Estimate cost' the first task is 'Check predefined repair option'. This check is performed for every incoming component and with this check, it is decided whether the component will be repaired in-house or outsourced. This is another decision with the only difference being that it has predefined outcomes for every type of component. In a certain way, this is an automated task that is not being considered by anyone or anything. Fokker Services can perform repairs and maintenance to a wide variety of components. However, there are some components that they are not able to repair as they lack the equipment to perform the repairs. In these cases, they do not have any other possibility than to outsource it to another organization. For all components that Fokker Services can repair in-house, they will send them to their repair shop. Besides that, there are no criteria to split in-house or outsource repairs. The outcome of this decision has a direct effect on how high the workload in the internal repair shop will be. Since there are some issues regarding to that as mentioned in the problem statement of this research, it is important to include it in this research. Similar to the other decisions a decision model was created as a base that is shown in figure 17.

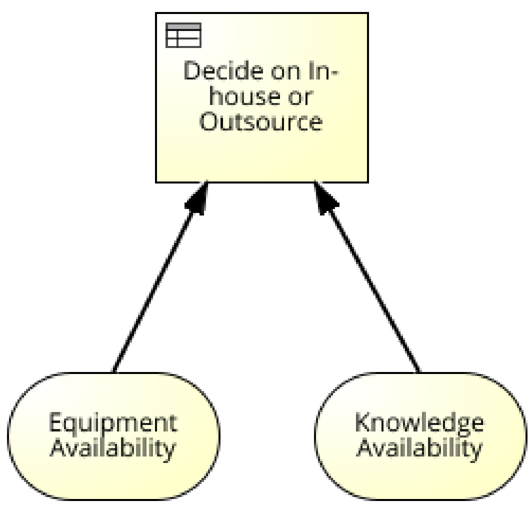

Figure 17: Decide on Inhouse or Outsource

5.3 Decision Selection

In the previous subsection, all decisions in the process were identified, but there is still too much to improve each of them. Therefore, a selection has to be made. Only for both 'Decide on option' and 'Decide on alternative option' it was already determined that these will not be selected. All other identified decisions are still a candidate. The main goal of this research is to improve the decisions. However, since Fokker Services is aiming to automate decisions in the future, it will be useful to focus on the decisions that have a high potential to be automated in the future. To determine which decisions have the most potential for automating, the identified decisions were tested to the criteria of Taylor, 2012 that were described in the section 4.5 about Suitable Decisions for Automation. An overview of the results for the tested decisions can be seen in figure 18. T1-R4 and D1-D6 behind Repeatable and Non-trivial indicate the different tests and drivers in the same orders as they were described in the previous chapter. In the figure a the green boxes indicates whether if fulfill to the criteria, while a red one indicates that it did not.

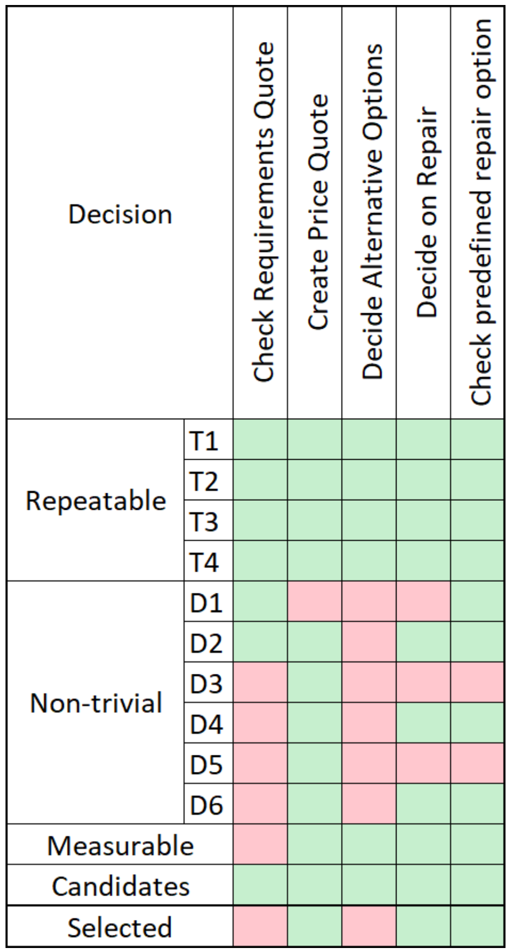

Figure 18: Decision Selection

The first decision is 'Check Requirements Quote'. It is known when the decision needs to be made as it a fixed task in the process that will be performed for every incoming order. For all of the orders it will always consider the same possible options, whether it is, or it not required to make a quote for the order. The information considered for this decision is mainly the contract information, as the decision may depend on the exact conditions. Lastly, there is consistent measurement of whether the decision was good or not, as it always becomes clear if the wrong decision was made when it needs to be corrected. Therefore, this fulfills all tests of the criteria repeatability. However, this is different for the criteria of non-trivial. Since it is such a simply decision, there are not many drivers of complexity in this decision. The decision is basically made on a simple policy for the type of order which is complemented with information for the contract conditions. As this information is not widely available and the contract conditions are quite different from each other, it does depend a lot on the experience of employees. However, this is the only driver of complexity that really occurs. Therefore, it does not fulfil the criteria of non-trivial and this decision will not be selected.

The second decision of 'Create Price Quote' also fulfills to the criteria of repeatability. It is known when this decision is needed and considers similar information to determine a price for all orders. Because the price has direct influence on the profitability of the order is very easy and consistent to measure the success of the decision based on the profitability. This decision is also non-trivial as there are many drivers of complexity within this decision. Experience of the employees is very important in this decisions as requires quite a lot knowledge about several factors. Besides that, a lot of data and information is required to be able to select the best option. In many cases it includes the use of some analysis as well. Even though a price needs to be determined for every quote, the range of prices can differ a lot between different orders as are for different components. Finally, there a very important trade-off needs to be made for the price. If the price is too high, customers may decline, however, lowering the price will be at the cost of profit. Therefore, this is one of the most complex decisions within the process. It is very well possible to measure the business impact of the decision as it has direct influence of the profitability of the orders in the process. Therefore, it also fulfills this criteria. The last criteria is whether the organization accepts the decision to be improved with the help of automation which is also true for this decision. Fokker Services aim to automate decisions, so there is a lot of acceptance for this within the organization. For this specific decisions the employees involved in the decision-making also think that it can help them spending less time on the decision-making. so this decisions fulfills all criteria and will therefore be selected as one of the decisions.

The third decision is 'Decide Alternative Options'. This decision fulfills the criteria of repeatability, as it uses similar information to select from the same options and it is known when this decision needs to be made. However, also for this decision it does not fulfill the criteria of non-trivial. It is a very straightforward decision again, that does not need any domain knowledge, does not use much data or any analysis. There simply is not any driver of complexity for this decision that makes it non-trivial. Therefore, this decision was not selected as well.

The fourth decision is 'Decide on Repair' which also fulfills the criteria of repeatable like the other decisions. However, for non-trivial it does not have all drivers the can cause complexity. There are no real policies or regulations that influence this decision. Besides that, there are no analysis needed for the decision and it has only two possible options which does not make the decision very complex. However, it is a decision that requires a lot of domain knowledge and experience due to the high amount of components that can occur. Even though no analysis are needed, the decision require quite some data. What makes this decision quite complex is the availability of trade-offs. Each of the two possible outcomes cam have several advantages and disadvantages. These can be different for different type of components as well. Therefore, this decision can become very complex that makes is non-trivial. It is very well possible to measure the business impact of this decision. It can have effect on the throughput time of several orders and therefore influence the profitability of the process. Lastly, this decision is considered as one of the most important in the process by Fokker Services. Therefore, they are open for this decision to be improved. So this decision will be the second one to be selected for this research.

The last decision considered is 'Check predefined repair option'. Like all other decisions this one also fulfills the tests of the repeatable criteria. Besides that, it also has a lot of drivers for complexity, even though the decision is very simple ans straight forward. In the current situation this decision is more a policy than a decision as Fokker Services aims to handle as much orders as possible in-house. However, as a result of this policy the repair show faces an extremely high work load that caused many problems for the organization. Therefore, considering this as an active decision again can be very useful. If that is the case, the complexity of the decision is very high, as it requires several analysis and a lot of data for this decision to be made. There will also be several difficult trade-offs that needs to be considered. Therefore, this decision has more than enough complexity to consider it as a non-trivial decision. The measurable impact of this decision is also very straightforward, as it has effect on the throughput time of orders and may solve the current problems of high work load. That last part is also what makes it why Fokker Services would be open to consider this decisions as a possible candidate. So, this decision will be the last one that is selected for improvement in this research.

5.4 Performance of Current Situation

The decisions to be improved are now selected. The next step is to determine how the possible improvements of the selected decisions will be measured throughout this research. To improve decisions, monitoring performance is critical in the process of measuring and improving decisions (Taylor & Purchase, 2016). A general approach in decision modeling is to link the decisions to the performance metrics of the business. However, to increase business performance not only decision performance needs to be monitored, throughput and basic statistics are also required (Taylor & Purchase, 2016).

In the problem statement, it was described that the decisions are inefficient or wrong. However, Fokker Services does not aim to improve the decisions just to have better decisions. The reason they are considering improving decisions is that they would like to improve the performance of the processes. Therefore monitoring the performance of the entire process is a good way to check the effect of the decisions and whether they have improved. Measuring the performance of processes can be done in many different ways based on the performance indicators of the process. So it is necessary to make a selection of important performance indicators for the relevant processes at Fokker Services.

The research of Hon, 2005 investigated the different performance measures in manufacturing systems. In total 442 performance measures have been collected and are included in a framework for Performance Measurement. As these were too much to include, Hon, 2005 made a selection which is shown in table 2.

Time Measures	**Cost Measures**	**Quality Measures**	**Flexibility Measures**	**Productivity Measures**
average batch processing time	overhead cost	average outgoing quality limit	component reusability	assembly line effectiveness
average lead time	scrap cost	incoming quality	delivery flexibility	direct labour productivity
changeover time	setup cost	MTTF	machine flexibility	machine effectiveness
cycle time	tooling cost	not right first time	number of different parts	network effectiveness
machine downtime	total quality cost	process capability index	process flexibility	overall equipment effectiveness
mean flow time	unit labour cost	return tate	process similarity	return on assets
on-time delivery	unit manifacturing cost	rework %	routing flexibility	stock turn
setup time	unit material cost	scrap %	supply chain flexibility	throughput efficiency
Takt time	work in progress	vendor quality rate	total system flexibility	total productive maintenance
throughput time		warranty claim %	volume flexibility	value-added per employee

Table 2: Performance Measures (Hon, 2005)

For these selected performance measures Hon, 2005 performed research among organizations in the Aerospace Industry in the North West of England to identify which are important measures and how they are used by within the Aerospace Industry. Within Fokker Services one of the main goals is to finish the orders on time and deliver the components on time to the customers. Therefore, the time performance measures can be very useful for this research. This goal of Fokker Services matches perfectly with the results from the research of Hon, 2005, as 'On Time Delivery' and 'Overall Lead Time' are by far the most essential time performance measures. An overview of the results of importance of the different time performance measures is shown in figure 19.

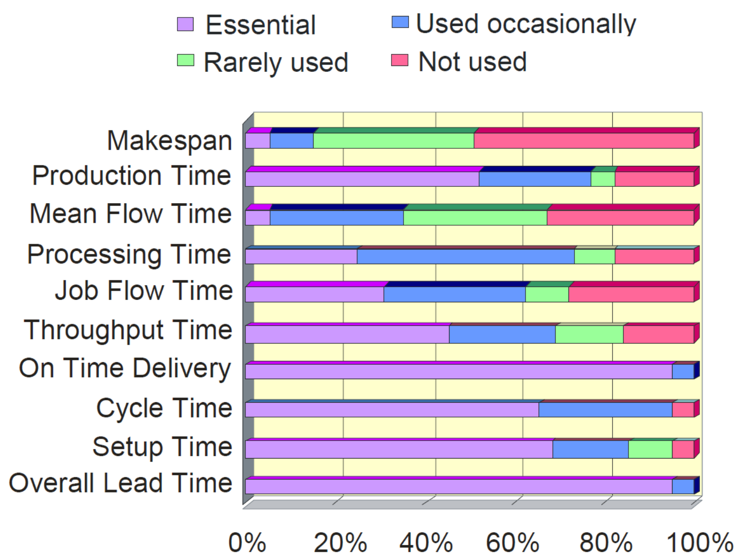

Figure 19: Time Performance Measures (Hon, 2005)

Due to a lack of data about the orders of Fokker Services determining the Overall Lead Time is very difficult. However, the available data allows the calculation throughput time. In the research of Hon, 2005 it was a measure that is slightly less essential than overall lead time and on time delivery. However, from the measures that could be used based on the available data, throughput time is the measure which was the most essential in the research of Hon, 2005. Throughput time would therefore be a suitable performance indicator for the process to reduce the throughput time of the process. Besides that, for Fokker Services the service level is very important as they use it as one of their main performance measures. This service level is basically the on time delivery which was one of the most essential measures in the research of Hon, 2005. Therefore, this should be used as one of the main performance measures.

Since Fokker Services maintains contracts with customers where service levels are included, it is most important to meet these determined service levels. Lowering the throughput time for all orders is not the goal. Many orders are finished in time, so further reducing the throughput time of these orders does not directly affect the service levels. Instead, the focus within Fokker Services is to increase the service levels, especially since they know most of the required service levels are not met. Throughput time will be measured and monitored as it is required to determine the service rate. However, in this research, the goal for improving decisions is to achieve higher service levels for the processes.

5.4.1 Performance of Current Processes

Before improving the decisions to increase the service levels, it is important to have a baseline of the current performance. Since Fokker Services does not have actual data or information on service levels, they were calculated with the help of the throughput time of the orders. The throughput time was calculated for all orders from 2018 onwards. In total Fokker Services completed around 45000 orders in the last five years. An overview of the total number of orders per year is given in figure 20. In the figure, a distinction was made between Customer Orders of the CMRO Service and Exchange Orders of the Exchange Programs Service. The first thing that stands out is that the number of exchange orders is very low compared to the number of customer orders. Only 15% of all orders in the last five years were exchange orders. Even though the service of Exchange Programs is relatively important to Fokker Services, it only provides a small portion of the orders. Another remarkable thing the figure shows is that the total number of orders decreased significantly by 40% in 2020. The main reason for this is the situation in the world regarding the Covid pandemic from 2020 onwards. The aerospace industry in general was hit hard by this pandemic. Aircraft were used less frequently which made a lot of the maintenance in the industry redundant. As a maintenance provider, Fokker Services was hit by this as well and had to downsize its operations. The final important remark regarding the number of orders in 2022, is that it does not include the orders for the entire year. This data was collected and calculated in October 2022. Therefore, only orders until the first of October were included and there is another significant decrease in the number of orders in 2022. The expected number of orders in 2022 is expected to be similar to 2021.

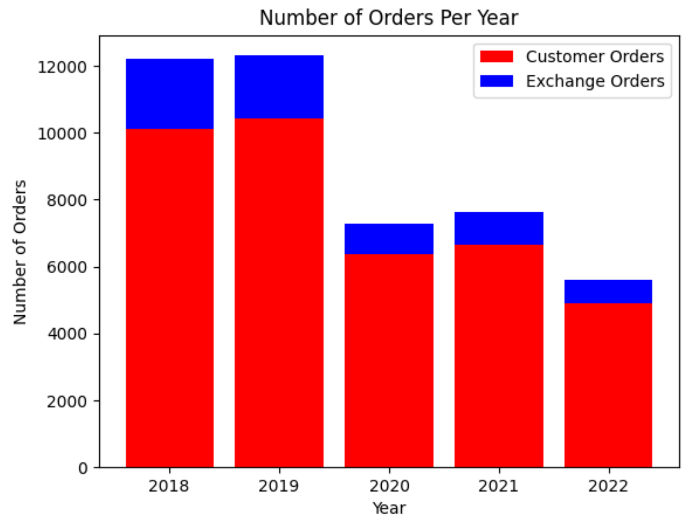

Figure 20: Number of Orders

An overview of the calculated throughput time of the orders is given in figure 21 for the customer orders and figure 22 for the exchange orders. Both types of orders show a similar pattern in that most orders are completed in a relatively short amount of time. The mode for the throughput time of the orders appears at 7 days for both processes. However, some orders took many months to be completed. A few even took over a year to complete. These were kept outside of the set limits of the figures to keep the figures somewhat readable. As mentioned before, just the throughput time of orders is not sufficient to use as a performance indicator. To determine the achieved service levels it is important to know the time allowed to complete the orders. Fokker Services does not register these times, therefore, it is quite difficult to get 100% correct for every order. However, a solid estimation can be made based on general rules within Fokker Services and the agreements they have with customers. Fokker Services maintains a general time of 21 days per order. That means that the customer must receive the fixed or replaced component within 21 days after they requested their order. Of course, there will be some exceptions to this rule as there could be some requests that require emergency and might have a shorter required time. However, to take these into account, for every order the contract conditions need to be investigated as the information is not stored anywhere else. This would make everything much more complex. Most of these orders are exceptions anyways and would not have a too big influence on the results. Therefore, an assumption was made that all orders need to be finished within 21 days.

With this information, the figures show that many orders will take longer than the allowed time to complete an order. The achieved service level for Customer Orders turned out to be only 66% throughout the last five years, while the service level was even lower for Exchange Orders with only 49%. These service levels are quite low, especially compared to Fokker Services' target of 90% for every service they provide. Therefore, it can be concluded that the performance of both processes is far from ideal and have a lot of room for improvement.

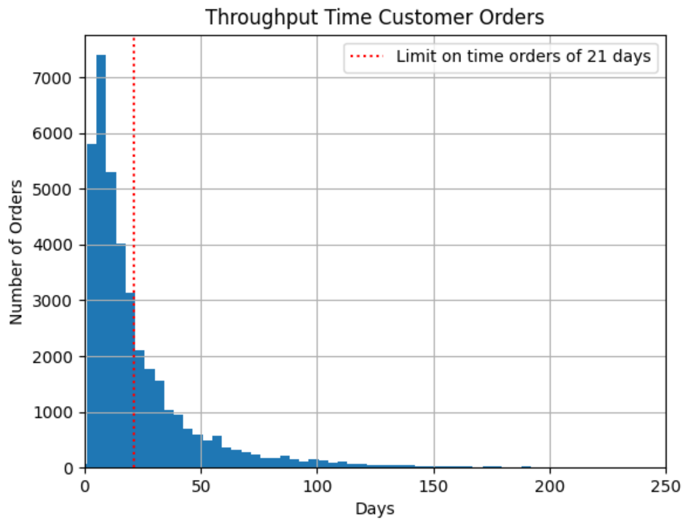

Figure 21: Throughput Customer Orders

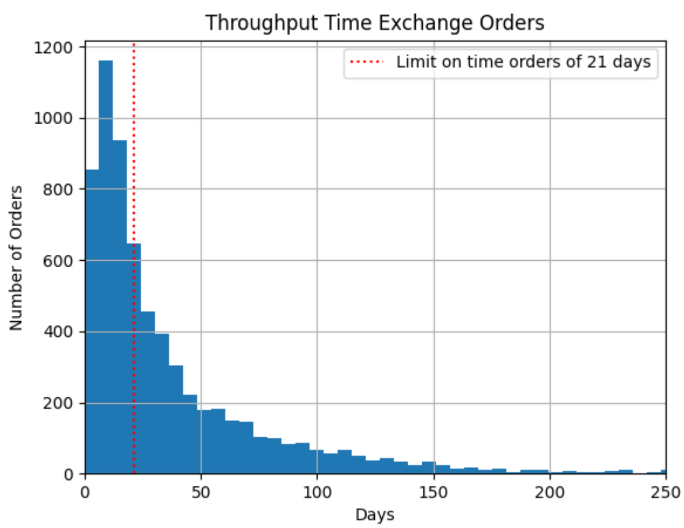

Figure 22: Throughput Exchange Orders

These results for the throughput time and service level of the orders in the current situation form a baseline for the remaining part of this research. The selected decisions in the first part of this chapter will be improved in later parts of this research. At the end of this research a comparison will be made to investigate the effect of the improved decisions where the results from this research will be compared with the results of the improved situation.

6 Data Needed & Data Quality

Before the decisions will be improved and Decision Mining algorithms can be implemented, it is necessary to identify the data required for the decisions and check for data quality issues. This will be done in this chapter. The current situation for the decisions was already modeled by creating Decision Requirement Diagrams (DRD) and Decision Tables. These models can be used as a base to identify the input data for the decisions. Once it is clear which data can be used as input data, the quality of the data will be checked to find potential data quality issues. For the identified quality issues, further data cleaning will be performed to improve the data quality and solve the issues. With the decision models and the data quality assessment in this chapter, sub-question Q2 will be answered.

6.1 Data Needed

'Decide on Service', 'Create Price Quote' and 'Decide on In-house or Outsource' were the three selected decisions for improvement in this research. for all three of these decisions, a decision model was already made. 'Decide on Service' and 'Decide on In-house or Outsource' includes a DRD and a Decision Table while 'Create Price Quote' only includes a DRD.

Figure 23 shows the DRD of the decision 'Decide on Service'. In total 4 different inputs are required to make this decision. Two of the inputs first determine a sub-decision that is required for the final decision together with the other two inputs. 'Expected Cost' and 'Current Market Value' are both inputs for the sub-decision. the only data that is needed for this sub-decision is the price for the expected cost and the price of the current market value. In this sub-decision, the difference between the prices will be calculated by subtracting the values from each other. The output of this sub-decision can be positive and negative, and based on that outcome it will be used as input for the main decision. The remaining input that is required for the decision is 'Stock Need' and 'Current Stock Level'. Exact calculations for stock level and stock need were kept outside of the scope of the internship assignment. Instead, the exact data was simplified to two options for both inputs which are enough to make this decision. Whether there was a stock need or not and whether the current stock level is sufficient or not. In Figure 24 the decision table for 'Decide on Service' is shown provide a clear overview of which input combinations result in which decision outcomes.

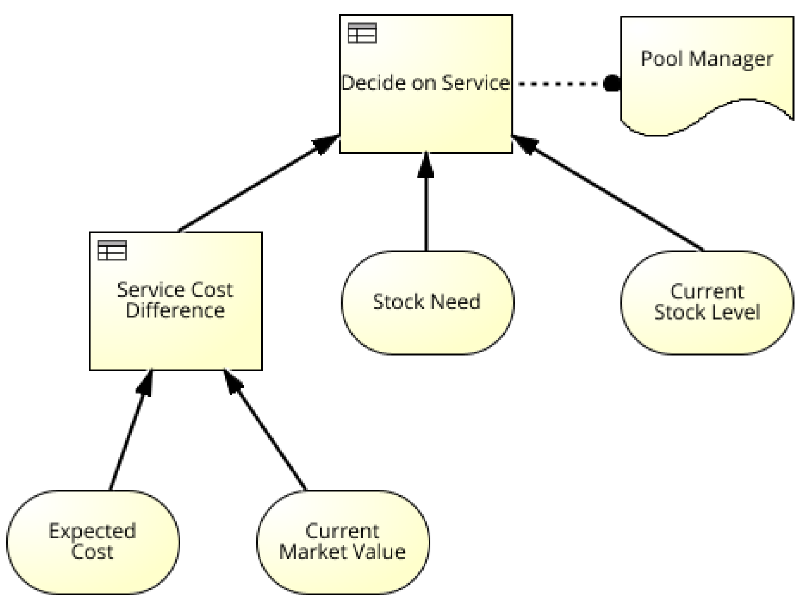

Figure 23: Decide on Service

A	Inputs						Outputs
	Service Cost Difference		Stock Need		Current Stock Level		Service decision
	Number		Boolean		{Sufficient,Unsufficient}		Boolean
1	≥	0		-		-	true
2	<	0	=	true	=	Unsufficient	true
3	<	0	=	false		-	false
4	<	0		-	=	Sufficient	false

Figure 24: Decide on Service table

'Create Price Quote' is a decision that requires the most input and data of all. As Figure 25 shows seven different inputs are required for this decision, therefore, a lot of data is needed. Most data that is needed is about prices and value of the component. 'Cost Quote' is the amount of cost that will be made to perform the service. 'Price Alternative Service' is the amount of cost that will be made if alternative options are used instead of the normal service. In this decision the exact value for these options is required, therefore no simplification of data can be used. 'Fair Market Value', 'Second Hand Market Price', and 'Competitors Price' are all data about the components' prices outside of Fokker Services. Similar to the previous two inputs, the data of the exact prices are required to be able to make the decision. For 'Second Hand Availability' data is required for the number of components that is available on the market. The exact amount of available components is preferred to have, although in most cases estimations are already sufficient for this decision. For example, if there is a large number of components available, it does not matter too much whether there is one more available. The last input of the decision is 'Customer Preference Tag Date', which is data about the preferred age of the components for every customer.

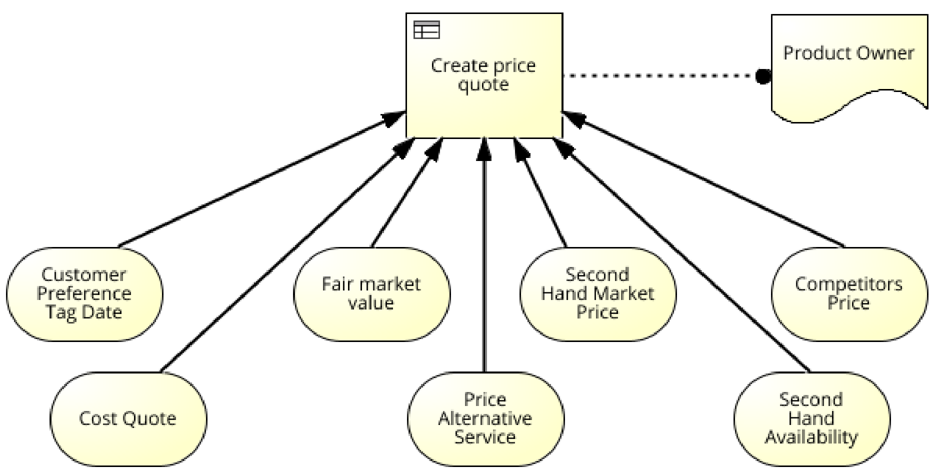

Figure 25: Create Price Quote

In Jonkers, 2022 the decision 'Decide on In-house or Outsource' was not considered as an active decision. In the process, it was included as a single task that checks for the component if it must be serviced in-house or outsourced. The criteria for this check are straightforward. If a component can be serviced in-house, Fokker Services will always send it to their repair shop. Only if it is not possible to service the component in-house, for example, if they do not have the required knowledge or equipment for the service, it will be outsourced. However, this check has been performed for every component already so the outcome is pre-determined. In reality, the check is not to see what the possibilities are for the specific order, but just to check the outcome that was determined for the component.

If the check will be considered as an active decision, the pre-determined outcome for a component does not have to be fixed. The criteria used for the initial check are still required as input for the decision. To be able to make the decision it is always necessary to know if Fokker Services can perform the service in-house. Therefore, the two inputs ’Equipment Availability’ and ’Knowledge Availability’ are still required. It is possible to make a simple decision model based on this. The DRD of this decision is shown in Figure 26 and the decision table is shown in Figure 27. As the decision table shows the decision would be very straightforward, both the knowledge and equipment need to be available for Fokker Services to be able to perform the service in-house. If any of the two is not available it means Fokker Services cannot repair the component themselves and need to outsource it. Besides that, no other data is required for this decision. However, the aim of considering this decision for improvement is to try to solve the current problem that the amount of work in the repair shop is piling up. One way is to make this decision on much more data as input, for example, current conditions in the repair shop. Which inputs and data will exactly be useful for this decision is something to investigate. Therefore, the relevant available data for the process and orders needs to be considered.

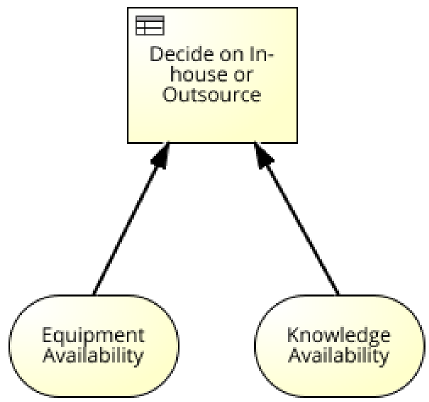

Figure 26: Decide on In-house or Outsource

A	Inputs				Outputs
	Knowledge Availability		Equipment Availability		In-house or Outsource
	Boolean		Boolean		{In-house,Outsource}
1	=	true	=	true	In-house
2	=	false		-	Outsource
3		-	=	false	Outsource

Figure 27: Decide on In-house or Outsource Table

6.2 Data Understanding

In the previous subsection, it was described which data is required to make the decisions in the current situation. Most of the data that is required is about the orders, the components, or the status of the repair shop. Fokker Services has several data sources. The one which provides the most data about the orders and components is Pentagon, which is the ERP system of Fokker Services. However, Fokker Services collect a lot of data from different sources as well. A big example is the information about the availability of components in the market and the associated prices of components on the market. This data and information are stored in separate data sets that were collected in a data warehouse. Fokker Services used to have a data warehouse that collects most of the data within the organization. Even data from Pentagon has been processed and converted to data sets in the data warehouse. Before the data was stored in the data warehouse, data scientists within Fokker Services processed and cleaned the original data from Pentagon. Therefore, most of the data within the data warehouse was fully cleaned and should have high data quality. Very recently Fokker Services switched towards a data fabric to further integrate the data of the entire organization and be able to better manage their data. All data sets that will be useful and required for this research are gathered from this new set-up data fabric.

To collect the data that is needed for the decisions as described in the previous sub-section, a total of four data sets are required:

- 'Work_Order': This data set contains all the data corresponding to the orders that Fokker Services received and processed. It contains the most information about the orders is will be the most useful for this research. It includes for example most basic information about orders as part numbers of components, the type of maintenance action that was required, receival and completion dates, and incurred costs of the orders.
- 'Work_Order_Labour': This is an additional data set of 'Work_Order' that includes more detailed information about the labour that was performed in the repair shop. In 'Work_Order' the only data included about labour is labour cost. More detailed information, for example, which employees performed the tasks for an order and how much time they spent working on these orders.
- 'Work_Order_Status': This is another additional data set of 'Work_Order' that includes more detailed information on the status of the orders. This data set has a lot of similarities with an event log of the process as it includes the time stamps when a new status of an order has started and is completed.
- 'Stock': This data set contains all data about the current stock of Fokker Services' warehouse. Every part that is available in stock includes data such as the amount of the part, the condition of the part, and the duration it has been available since its last repair.

Fokker Services collected data about their orders and processes throughout their organization for many years. As a result, all data sets contain an enormous amount of data. Using the data sets with all historical data is not suitable for a few reasons. The biggest reason is that very old data will not be very representative of the current processes within the organization. For example, components that were repaired regularly about ten years ago could not be requested anymore since they are not being used in active aircraft anymore. On the other hand, many components are repaired in recent years that were not yet used in the past. In the ideal situation, only the most recent data should be used. However, by limiting the data to only the current year 2022, the size of the data set will not be very large, as there were only about 6000 orders up till this point in 2022.

Determining what is the best size for the data set is very difficult. In the literature, there are no clear ideal ranges for data set size in the topics of decision and process mining. The best indication can be taken from the literature on decision trees. For the topic of decision trees, there is more information available on how the data set size influences the performance. The reason decision trees are relevant for decision mining is that it is a very common algorithm for decision mining. Within the topic of decision mining, the research of Rozinat and Van Der Aalst, 2006 has been seen as a baseline, and the algorithms used by Rozinat and Van Der Aalst, 2006 were decision trees. Many other different decision mining techniques have been developed throughout the recent years (Jouck et al., 2019) (De Smedt, vanden Broucke, et al., 2017). However, decision trees are still considered one of the traditional techniques of decision mining (De Leoni et al., 2016).

Morgan et al., 2003 performed research on how the modeling accuracy is influenced by the size of the data. The results of this study show that the accuracy of the decision tree models reached a plateau around the size of 16000. Besides that, in 80% of the cases, the accuracy reached a level within 0.5% of the optimum with a size of 10000 already. Therefore, having a data set with a size of at least 10000 to 16000 can be set as a minimum bound. The performance will be increasing when the size of the data set increases, however, it is important to set an upper bound for the data set size as well. Sug, 2009 performed another research on how the size of the data set influences the accuracy of decision tree methods and concluded that overfitting becomes an issue when using larger data sets. Sug, 2009 made comparisons for several decision tree algorithms and compared a data set of size 40000 with a data set of size 64000. For the algorithm of

C4.5, one of the most popular decision tree algorithms, a data set of 64000 has a slightly higher accuracy of 0.06%. However, due to the higher risk of overfitting, it is not always preferred over the data set with a size of 40000. Therefore, the range of 40000 to 64000 is a decent upper bound for the size.

As mentioned before Fokker Services only had about 6000 orders in 2022 which is not enough. Therefore, data from previous years need to be included as well to create a larger data set. When including all orders from the previous five years, so from the start of 2018, the data set will include about 45000 orders. Compared to the limits set in the literature on decision trees, a data set with a size of 45000 orders should be able to perform well with decision tree algorithms. In the next subsection, the data quality will be checked for the entire data set, so the size of the data set might slightly decrease if there are some erratic orders included. Since Fokker Services already has cleaned most of its data, the number of expected removals is limited and the data for the last five years is more than enough. Using data from the previous five years should also not lead to the issues of not having relevant data as explained earlier in this subsection.

6.3 Data Quality

Even though, Fokker Services have a lot of clean data that can be used for this research, the data quality of the described data sets will be checked. Within the decision-mining community, data quality plays an important role. The success rate of decision mining algorithms depends heavily on the quality of the data (Rozinat & Van Der Aalst, 2006). According to Rozinat and Van Der Aalst, 2006 data quality is so important for decision mining that the techniques cannot be utilized completely without having data of the highest quality. Therefore, data quality is seen as one of the main challenges for decision mining (Rozinat & Van Der Aalst, 2006). Even though it is clear that data quality is important for decision mining, there is barely any literature and information available on the topic of data quality for decision mining. On the other side, there is a lot of available literature on the topic of data quality for data mining and process mining that can be used as a baseline for decision mining.

For data mining and process mining much more research has been performed on the topic of data quality. Within the data mining community, the importance of data quality is well recognized and considered as crucial (Goel et al., 2022). As a result, many data quality metrics have been developed in the past to solve data quality issues. Also within the process mining community, the importance of data quality has been well recognized (Goel et al., 2022). According to Van Der Aalst, Adriansyah, et al., 2012, the quality of process mining results depends heavily on the input as high-quality logs are essential for getting the best possible results. This is similar to decision mining where high-quality data is essential as well. In another research, Jagadeesh Chandra Bose et al., 2013 mentioned that more attention should be paid to the data quality of the event logs before the process mining algorithms are applied. Most of the process mining algorithms do not take data quality into account at all (Goel et al., 2022). As a result, erroneous results will be obtained by the process mining algorithm that will eventually lead to inaccurate or misleading conclusions (Goel et al., 2022). The researchers of Jagadeesh Chandra Bose et al., 2013 have a lot of practical experience with process mining techniques within organizations. They experienced that a lot of logs are far from ideal and include many quality issues. For example, incomplete, noisy, and imprecise data are some of the issues with the logs used for process mining (Jagadeesh Chandra Bose et al., 2013).

To be able to solve the quality issues it is important to investigate what the potential issues may be. In literature, a lot of research has been performed and several frameworks and approaches are provided. The research of Martin, 2021 provides an introduction to the main topics of data quality for process mining. In this research hospital data in a healthcare environment was used. However, this is only relevant for the examples of the data quality assessment results, which is only a small part of the research. The approaches and topics discussed about data quality are a general approach for process mining. Even though these examples of the assessment were applied to a healthcare log, there is no limitation to use it for different logs in a different background. For example, 'Data Quality Taxonomies', which is one of the topics of data quality in this research include several taxonomies and none of these were specially developed for logs in healthcare logs.

Three main topics in the research of Martin, 2021 are 'Data Quality Taxonomies', 'Data Quality Assessment', and 'Data Cleaning'. These three topics are all required and can be seen as three phases to improve the data quality of the event logs. Firstly, data taxonomies are used to identify the potential data quality issues of the log. Once these potential issues are identified, a data quality assessment is required to detect which of the quality issues occur in the event log. Finally, when it is known which quality issues need to be solved, data cleaning is required, where heuristics are used to solve the data quality issues of the event log (Martin, 2021).

It is important to note that the data used for decision mining in this research will not consist of event logs but uses data sets instead that was described in the previous subsection. Even though the data in this research will consist of a completely different structure as described in the literature, the approaches to solve the data quality issues can be used well for this research. Not everything described in the topics of Martin, 2021 could be used exactly due to the specific requirements of an event log. However, for each of the topics, several approaches were included in the research of Martin, 2021. This can help well if one of them turned out to not be very useful for decision mining. In the end, the main goal in this part of the research is to improve the data quality of the specified data sets of Fokker Services. To be able to do that, first, the quality issues need to be identified. Then it also needs to be assessed how big these issues are before it is possible to improve the data sets by further data cleaning. Since these steps match well with the topics described in (Martin, 2021), it provides a good baseline for improving the data quality.

6.3.1 Data Taxonomies

In the part of the research of Martin, 2021 that described the topic of data quality taxonomies, not only specific data quality taxonomies for process mining are mentioned. It also includes general data quality taxonomies, such as the widely applicable taxonomies of Wang and Strong, 1996. In the research of Wang and Strong, 1996, four types of data quality are distinguished and each includes several data quality dimensions.

- Intrinsic data quality: The most evident dimensions are accuracy and objectivity of data, however, it also includes believability and reputation.
- Contextual data quality: Completeness and timeliness are the well-known dimensions within this type. Other dimensions within this type are relevancy, value-added and appropriate amount of data.
- Representational data quality: This type includes the dimensions related to the meaning of data, interpretability, and easy of understanding, but also to the dimensions related to the format of data, concise and consistent representation.
- Accessibility data quality: The last type naturally includes the dimension for how accessible the data is, but also includes access security of the data.

This taxonomy provides some important quality dimensions that could be used for any kind of data. Therefore, these included quality dimensions are good candidates to start with and use further in this research. Besides, this general taxonomy, Martin, 2021 provides two taxonomies specific to event log used for process mining. The first one is the taxonomy of Jagadeesh Chandra Bose et al., 2013. This taxonomy distinguishes four categories that contain a total of 27 quality issues.

- Missing Data: This is about the scenario where mandatory information is missing from the log. It could be cases, events, relationships, case attributes, positions, activity names, timestamps, resources or event attributes that are missing from the event log.
- Incorrect Data: This is about the scenario where the data in the log is not logged correctly based on context information. It could be cases, events, relationships, case attributes, positions, activity names, timestamps, resources or event attributes that are logged incorrectly in the event log.
- Imprecise Data: This is about the scenario where the logged data is too coarse, which results in not being able to perform certain analyses or unreliable results of these analyses. It could be relationships, case attributes, position, activity names, timestamps, resources or event attributes that are not precise enough for further utilization.
- Irrelevant Data: This is about the scenario where the data in the event log is not useful, while other relevant data for the analysis might still need to be obtained. It could be only cases and events that are not relevant to the preferred analysis.

These quality issues are very specific for events logs and are therefore completely different from the general quality dimensions from the previous taxonomy. Since the data of Fokker Services is not an event log, these specific quality issues for event logs will not be extremely useful. However, the other specific taxonomy for event logs provided by Martin, 2021 is different. The taxonomy of Verhulst, 2016 consists of a framework that is based on both general data quality literature and specific data quality issues for event logs. Verhulst, 2016 used general quality dimensions from literature as a baseline, and checked for all of these whether they could be used as an event data quality dimension. The ones that are useful for events are included in a developed framework for data quality aspects. Also for the dimensions that were not included in the framework, the reasons why they are not useful for the event log were explained. Therefore, it provides a lot of clarity about quality dimensions for event logs. An overview of the suitable and unsuitable dimensions is shown in table 3

Suitable Event Quality Dimensions	**Unsuitable Event Quality Dimensions**
Completeness	Institutional/Business Environment
Uniqueness/duplicates	Linkability
Timeliness	Usability
Validity	Interpretability/Understandability
Accuracy/correctness	Reputation
Consistenct	Value-Added
Believability/credibility	Objectivity
Relevancy	Reliability
Security/confidentiality	Verifiability
Complexity	Data Decay/Periodicity
Coherence	Ease of Use/Maintainability/Accessibility
Representation/format	Disaggregation

Table 3: Suitable and Unsuitable Quality Dimensions

The quality dimensions considered by Verhulst, 2016 are similar to the ones described by Wang and Strong, 1996. Most of the suitable event quality dimensions are widely used dimensions in general data taxonomies. Another research of Cichy and Rass, 2019 compared many data quality frameworks in literature and investigated the data quality dimensions in these frameworks. Most of the quality dimensions included in the taxonomies of both Wang and Strong, 1996 and Verhulst, 2016 are very commonly used throughout quality assessments. According to the research of Cichy and Rass, 2019 the five most common quality dimensions are: Completeness, Accuracy, Timeliness, Consistency, and Accessibility.

Of the previously described taxonomies, the taxonomy of Verhulst, 2016 provides the most usable base for usable quality dimensions. These quality dimensions can be widely used for all kinds of data and most of them could therefore be considered the most important quality dimensions. Another reason why this taxonomy would be preferred for the next part of this research is that the framework based on the taxonomy is ideal to use in the next step of the actual data quality assessment. Verhulst, 2016 provide a very clear description of the methods for the quality assessment where the taxonomy is used and can be applied very easily to the different data sets of Fokker Services. The taxonomy includes the following data quality dimensions with the definition according to Verhulst, 2016:

- Completeness is the dimension that determines how complete the data is and if it includes as much necessary information as possible.
- Uniqueness/duplicates is the dimension that measures the number of values for an attribute that occurs only once.
- Timeliness is the dimension that measures how old the data is and whether it fits in the expected time frame.
- Validity is the dimension that checks whether the data is valid and it conforms to the syntax.

- Accuracy/correctness is the dimension that determines whether the data correctly estimates or describes the quantities or characters it is supposed to measure.
- Consistency is the dimension that checks how much absence or difference there is between the data.
- Believability/credibility is the dimension that measures the trust and objectivity of the data.
- Relevancy is the dimension that checks how important the data is and which data could not be relevant.
- Security/confidentiality is the dimension that measures how well the data and information assets are protected.
- Complexity is the dimension that measures the structuredness of the process data.
- Coherence is the dimension where the interconnection between attributes in the data will be checked.
- Representation/format is the dimension that checks to what extent data is compactly represented and whether it is represented in the same format.

6.3.2 Data Quality Assessment

The quality dimensions in the taxonomy describe potential quality issues that can occur. However, with only these dimensions it is not possible to determine the quality of the data and whether the issues occur. Therefore, the quality dimensions in the taxonomy were included in a special data quality framework developed by Verhulst, 2016. The framework provides a measurement method, scale, domain, scoring system, and additional information if it is required. With this framework, a data quality assessment can be performed to identify the quality issues and determine how severe these issues are. For many quality dimensions in the framework, multiple measurement methods are described as there are several possibilities to measure the quality of these dimensions. Accordingly, there are different scoring systems for the different measurement methods. Since some of the measurements of the dimensions are specifically used for event logs, they are not very applicable to the data sets of Fokker Services. Therefore they will not be performed. An explanation for all measurements that will be used in the assessment of the data is included in Appendix E.

Among the data sets some quality issues were found with the assessment. The biggest issue that was not limited to one data set was the completeness of data. Especially the data sets of Work_Order and Stock have Stock have a lot of missing data. If data attributes contain missing values, the percentage of missing values is usually very high. It affects the percentage missing data on the entire data set a lot and makes it score quite low in the quality. On the other hand most of the attributes do not have missing data at all. Therefore, it is not the case that the entire data set is bad considering completeness. Another issues that became clear was related to the Accuracy/correctness of the data. There were some timestamps that were incorrectly logged. However, this only applied to a very small number of data entries. Besides that, no big quality issues were identified and the data quality of the data sets was relatively high. Results of the full Data Quality Assessment can be found in Appendix F.

6.3.3 Data Cleaning

The total number of quality issues that were identified in the quality assessment of the four data sets was not very high. Since Fokker Services has already undertaken several steps in the past to clean its data this result is not a big surprise. However, there were still some issues detected that could potentially be solved by cleaning the data. Missing data is by far the biggest quality issue that was identified as three of the four data sets include missing data. Other identified quality issues were incorrect data, inconsistent data, and irrelevant data.

Missing data is a frequent issue for many years in data analysis and a lot of research on this topic has already been performed (Brown & Kros, 2003). As explained earlier in this research, the research specific to decision mining is very limited. The research of Rozinat and Van Der Aalst, 2006 described that one of the reasons to select decision trees as a decision mining algorithm is that they can deal with missing attribute values. Since the decision trees will be used in this research as well to try to improve the decisions within Fokker Services, the missing data may not be a very big problem. On the topic of data mining, a lot of methods to solve missing data are available. In general, methods to deal with missing data can be divided into four categories: use of complete data only, deleting selected cases or variables, data imputation, and model-based approaches (Brown & Kros, 2003). Due to a large amount of missing data in some of the attributes, removing the entries with missing data and using only the complete data is not a proper solution. Then the data sets become way too small to use for further use. Imputation methods will also be difficult to apply to the data sets of Fokker Services. Fokker Services maintains a wide variety of components and parts in the process that cannot be compared to each other. In many cases, there is not enough data available to determine reliable replacement values. With the model-based approaches, decision trees are also included. Similar to the reasons of Rozinat and Van Der Aalst, 2006, decision trees are a good methodology to deal with the missing data (Brown & Kros, 2003). Even though the other category of deleting selected cases or variables may be useful in some cases for the data sets of Fokker Services, this will not be used beforehand. During or after the creation of the decision tree it could still be useful to handle the missing data by making use of pruning of the tree Brown and Kros, 2003.

Incorrect data in the data sets could be cleaned relatively easily. In the data set of Work_Order, the incorrect data is less than 0.01% of the entire data set. All of the incorrect data was regarding the start and/or completion date that result in a negative throughput time for the orders. Within Fokker Services no reasons for these incorrect data could be determined and it was assumed that they were just mistakes or incorrectly logged data. Therefore, it was decided to remove the few orders that include incorrect data from the data set. In the data set of Work_Order_Status, the reason for incorrect data was known. The incorrect timestamps were caused by the use of different time zones. The exact log error because of the different time zones could not be determined, as it is not known which time zone was logged. It is known that these events finished within one day as the time zone differences are at most 8 hours and none of the orders had a throughput time of fewer than -7 hours. Therefore, these values were corrected by adding 8 hours to each event with a negative throughput time and can still be included in the data set.

Finally, the irrelevant data would not be used in the remaining part of this research. Therefore, the irrelevant memo data attributes in Work_Order were simply removed from the data set.

6.3.4 Outliers

In the last part of the section on data quality outliers in the data set will be considered. The main reason for this is that some potential outliers were spotted in the data set when the throughput for the current situation was analyzed. Figure 21 and 22 provide an overview of the throughput time of most orders. However, the range for the x-axis of the number of days has been limited to a minimum of 0 days and a maximum of 250 days to be able to make the plot better readable. Therefore, some of the orders are not included in this plot as a few orders took more than a year to be completed. Figure 28 and 29 show an overview of all orders. These orders look like typical outliers as they take much longer than other orders. In the quality assessment of Verhulst, 2016 the outliers check is one of the checks that was not implemented in the framework. Outliers can be very tricky to identify. Even when they stand out in the data set and look like clear outliers, they are supposed to be like that in the data set. Having domain knowledge plays an important role in the evaluation of these outliers (Verhulst, 2016). In machine learning outliers are generally removed from the data set to improve the results of the model, however, for process mining, it may be better to keep them in the data set. A general rule of thumb is that outliers should only be removed if they are caused by data quality issues and should be kept if they truly happened (Rozinat, 2020). The orders that have a long throughput time include events in the data set Work_Order_Status that took a very long time and explain the extremely high throughput times. Therefore, there are no quality issues for these orders. Besides that, cases with an unusually long duration that took that long are the situations where outliers should not be removed (Rozinat, 2020). An example of outliers that would need to be removed were the orders with a negative throughput time. These were caused by data quality issues. However, these orders were already removed in the data cleaning part after the issues were identified in the quality assessment.

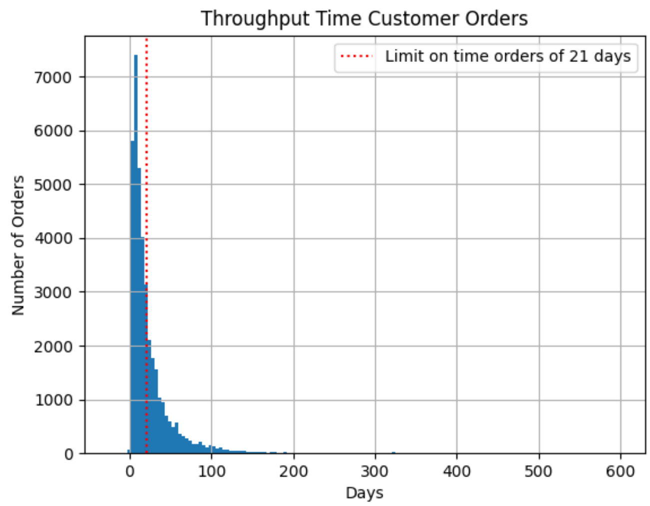

Figure 28: Throughput Customer Orders

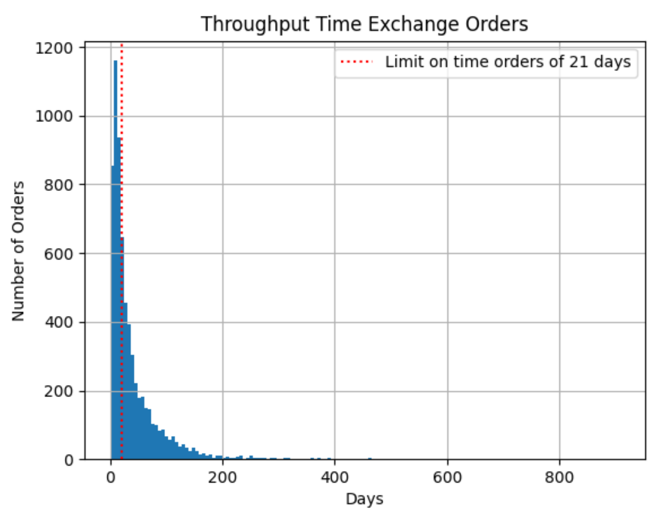

Figure 29: Throughput Exchange Orders

With the removal of the outliers the data cleaning part was completed, which also concluded the data quality assessment. Now the biggest quality issues are resolved and the data sets should be ready for the implementation of the decision mining algorithms in following chapters.

7 Implementation of Decision Mining Algorithm

In this chapter, the implementation of the Decision Mining algorithm and the results of this implementation will be discussed. Thereby this chapter will answer the part of the sub-question Q4 to which extent the improvements can be achieved with decision mining. As explained in the part of the literature review about Decision Mining, the main goal of the Decision Mining approach is to detect rules from the data and convert these into usable logical expressions. One of the main goals of this research is to identify whether it is possible to apply Decision Mining algorithms on the data set of Fokker Services and see whether these results can be used to improve the decisions. To prevent the implementation from becoming too technical and too much focused on different Decision Mining approaches, a relatively simple approach was selected. The approach selected was similar to the original Decision Mining approach of Rozinat and Van Der Aalst, 2006. So the identified decision was converted into a classification problem that was solved by using decision trees. For the decision tree algorithm, C4.5 was selected as it was relatively easy to implement due to the popularity of the algorithm. The results of the decision tree will then be used as business rules to create the ideal situation. Besides that, the decision mining approach was implemented for one decision to prevent the implementation from becoming too time-consuming. The decision that was selected for this implementation is 'Decide on In-house or Outsourced'. It is one of the two decisions that is considered for full automation. Out of these two decisions it is considered to have the most impact on solving the issue of overload of work in the internal shop.

7.1 Final Data Preparation

7.1.1 Data Integration

The data preparation of the different data sets was already performed and described in the chapter Data Quality. However, to use the data for the decision mining approach, and more specifically the decision tree algorithm C4.5, one data set that includes all data is required. So the different data sets need to be integrated. The data sets of Work_Order_Labour and Stock were integrated with the main data set of Work_Order. For the data set of Work_Order_Labour this provided not many problems as it contained the same list of order numbers as Work_Order which allowed merging the data based on this number. However, the data set of Stock provides some issues. First, the Stock data set does not include any order numbers as the stock is not linked to the orders in any way. But most importantly, the stock data set is like a snapshot of the stock level at the time the data set is created. Besides that, these data sets have not been stored at the time of order arrival. Therefore, it was not possible to have historical data on the stock levels for the relevant components at the time. The only way the data of stock can be integrated is by using a recent overview of the stock levels for all components at that time and merging these with the historical data of the orders. The big downside of this is that the stock levels may not be correct for all orders. For example, the stock level for a component may have been different at the time the order took place in the past. However, it was decided to still merge the stock data set with the data set of Work_Order. The main goal of the implementation of the Decision Mining algorithm is to detect potential useful decision logic. By not including Stock data, it is impossible to find any decision logic for the stock levels. If it is included, then it is likely that some interesting rules could be detected. Besides that, another goal of this research is to investigate the effect data quality has on the results and how it will have an effect on improving the decisions. However, it is important to take into account this potential quality issue when making conclusions based on the results.

7.1.2 Relevant Data

As a result of the integration of the data sets, one large data set was created which contained almost 100 columns as possible data features. However, from a logical perspective, many of these do not provide any added value. The decision for which the decision mining algorithm will be implemented is 'Decide on In-house or Outsourced'. The implementation aims to be able to classify the orders and whether they should be serviced in-house or outsourced at an other company. To be able to classify the orders, most of the data features will not influence the outcome. For example, if a component has a different serial number does not change the outcome of the classification. Another example is stock data such as the responsible person in stock or codes related to the location of the stock, a different name or number is not going to make a difference in the classification outcome. The last example is for features that may seem very relevant but provide no added value, namely the order number and part number. In the first test of the implementation, these were included. However, the result of the tree provided a split based on whether the order number is larger or smaller than a certain number. Meanwhile, the order number is just a number that identifies a certain order that increases for every new order that is created. so from a logical point of view, this is not useful at all and does not provide any added value.

Besides these examples, many other features will be useless in this implementation. Including these in the data set that is used in the C4.5 algorithm is not ideal. Even though pruning the decision tree in the C4.5 algorithm may help reduce the tree, the decision tree is very likely to become very large and complex. Due to the high number of irrelevant features, some will end up in the tree and make the decision tree unnecessarily large and complex. Besides that, having smaller data sets will save quite a lot of computation time during the implementation of the algorithm. Therefore, it was decided to remove these irrelevant data from the data set. As a result, the following data relevant data remained in the data set and was used in the implementation of the decision tree algorithm:

- PART_OWNER: This tells who is the owner of the component and is used to differentiate normal service orders from exchange orders.
- HDR_DEPART: This indicates in which department of the shop the order has been performed.
- INV_DOC_TOTAL: This shows the total cost that has been made by Fokker Services to complete the order.
- QTY_AVAILABLE: This shows the number of components from the same type of component that is in stock.
- ORDERS_IN_PROGRESS: This is the total number of orders that are in progress in the internal shop at the time the new order was created.
- WORK_REMAINING: This is the total number of hours of work remaining in the internal shop at the time of arrival of the order.
- OUTSOURCED: This indicates whether the component of the order has been outsourced to another company or not and will be used as a classification target.

7.1.3 Issue of Imbalanced Data

After the irrelevant data has been removed the data one last issue remained for the final data set, which is the fact that the data turned out to be extremely imbalanced. The old policy of Fokker Services was to perform as many services in-house as possible. As a result of that about 99% of the orders were completed in-house and about 1% of orders were outsourced to another company. There are many different techniques to handle this imbalanced data. However, undersampling and oversampling are the two main approaches for resolving imbalanced learning problems as both balance the classes in different ways (Shelke et al., 2017). Both are the complete opposite of each other, as undersampling reduces the majority class samples while oversampling increased the minority class samples (Shelke et al., 2017). As this data set is extremely imbalanced, undersampling is not ideal. To balance the data set it needs to remove a very large number of samples from the data set. This will result in a data set that is too small to be useful to gain results from the decision trees. Therefore, oversampling was used, and more specifically SMOTE: Synthetic Minority Over-sampling Technique. SMOTE creates new synthetic data points by considering its k nearest neighbors of a sample from the data set (Shelke et al., 2017). With this technique, it was possible to balance the data while remaining a decent size for the dataset. The biggest downside of the implementation of SMOTE was that it was not able to deal with missing data in the data set. The data was kept within the data set as the C4.5 algorithm can deal with the missing data. However, as the balancing of the data is necessary, the missing data had to be removed. As a result of this, the data set was reduced from about 45000 orders to about 32000 orders. This is still relatively fine compared to the bounds for the size of the data set that was determined earlier in this research.

7.2 Results of Implementation

The implementation of the C4.5 algorithm returned the results in the form of code that consist of expressions for the logic of the decision tree. As this code is not extremely clear and easily readable it was included in

Appendix G. Based on these expressions a visualization of the decision tree was made that is shown in figure 30. This visualization is much easier to read and understand.

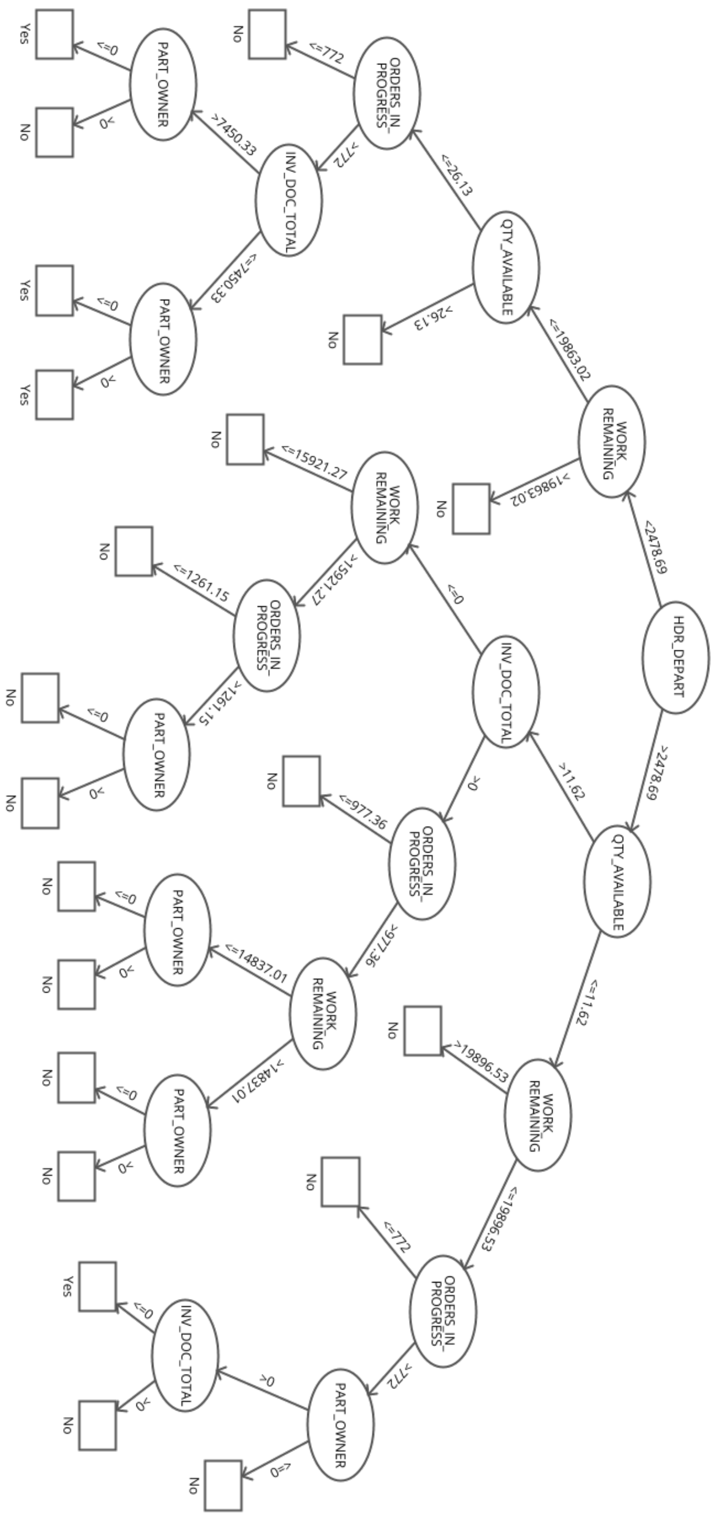

Figure 30: Decision Tree

However, the decision tree shows some remarkable and interesting results. The first interesting result is that the first split in the tree is based on the different departments. Departments are split into two groups and after this split, the remaining part of the decision tree looks completely different for both groups. For example, the next split for both groups is based on different features. Also, the exact values that determine the split different in most parts of the decision tree are different. This means that there will be several differences in the decision logic for the different departments. Something very remarkable about the first split of the department is, that it is split into only two groups and not in more separate departments.

Another interesting result of the decision tree is that there are several places where splits are made that do not lead to different classifications. The biggest example is after the split of QTY_AVAILABLE right of the first split. For the instances in the group of QTY_AVAILABLE > 11.62, several splits are made afterward. However, they all end up with the classification 'No'. So, these splits are redundant for the classification of these instances and indicate that further pruning of the decision tree is necessary. After using some post-pruning to the decision tree, these redundant splits were removed from the decision tree. The updated decision tree is less complex and shown in figure 31.

7.2.1 Results Implementation Separate Departments

From a logical perspective it does not make sense to group the different departments, or take them all together. The main reason for this is that they basically work as a independent shop and do not have much influence to each other. In the ideal situation the departments will need to be considered separately and the model needs to be implemented for separate data sets. Even tough, these separate data sets do not occur within Fokker Services it is possible to split the main data set into several sub-sets. It was decided to split the data set into several sub-sets for each department and implement the algorithm for the separate departments again and see to which extent useful results could be provided. The first interesting finding was that three of the departments, '2510', '2530', and '4230' did not outsource a single order. Therefore, all orders in the sub-set had the same classification of 'No' for the target variable. As a result of that it was not possible to create separate decision trees for these departments. For the other departments, it was possible to create them. The output of the algorithm is included in Appendix G and the visualization of these decision trees is included in Appendix H. There are many differences between the decision trees of the different departments, which support the need for a separate approach for the departments. Besides that, the additional decision trees do not provide any useful results. This is not very surprising as this is split for the departments is already some kind of a 'to be' situation and needs specific department data accordingly to be completely useful. Finally there is another interesting finding with the use of the sub-sets which is related to the size of the data sets. Some of the sub-sets turned out to be quite small. For example, the department '2550' only included about 3000 orders. This number is way below the minimum bound for a data set that was determined earlier in this research. This is not stimulating the performance of the algorithm.

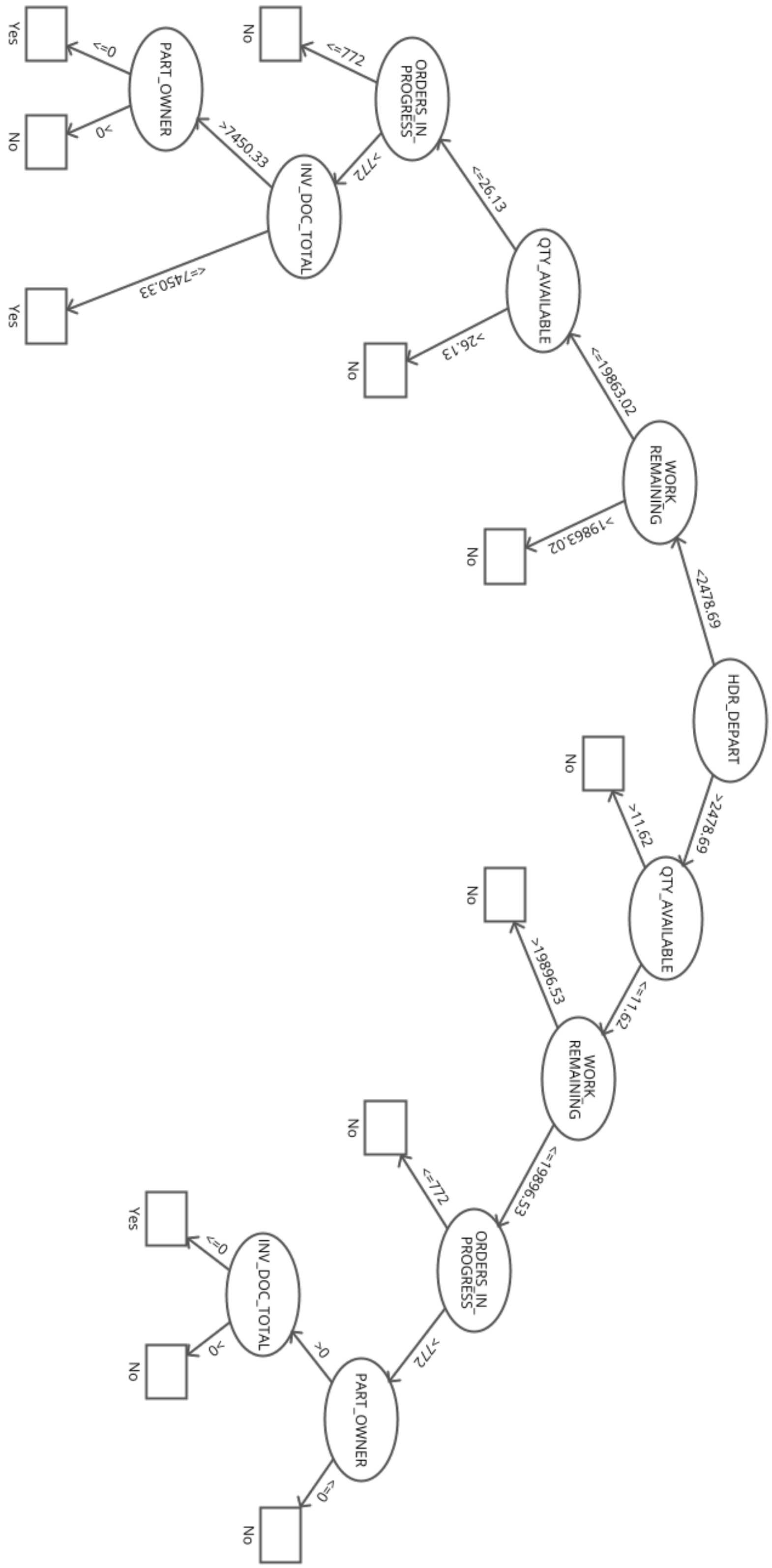

Figure 31: Decision Tree After Pruning

7.2.2 Overview Performance Evaluation Decision Tree

In this last part of the chapter, the performance of the implemented decision trees will be evaluated. Evaluating the performance of decision trees is very important as it tells a lot about whether the results of the decision tree are correct and reliable. Since the aim is to use the results to determine the ideal situation for the decisions, it is important to know whether these results are correct. If it turns out the results of the decision trees are not reliable they can lead to many wrong choices. That is something that needs to be prevented since these wrong choices will not help improve the decisions. To evaluate the decision trees several evaluation measures will be used.

Besides the output of the decision tree, the implemented C4.5 algorithm provided a Confusion Matrix with the following evaluation measure: Accuracy, Precision, Recall, and F1 Score. Accuracy measures the ratio of correct predictions over the total number of instances evaluated (Hossin & Sulaiman, 2015). Accuracy is one of the easiest metrics to apply and understand. However, it does have some main limitations. Accuracy produces less distinctive and less discriminable values (Hossin & Sulaiman, 2015). Besides that, it leads to less discriminating power to accuracy in selecting and determining the optimal classifier. Finally, the accuracy is also powerless in terms of informativeness and less favor towards minority class instances (Hossin & Sulaiman, 2015). Therefore, the F-Measure (F1 Score) or Geometric-mean are better metrics to use as they are a good discriminator and performed better than accuracy in optimizing the classifier for binary classification problems (Hossin & Sulaiman, 2015). The F-Measure is a metric that represents the harmonic mean between recall and precision values (Hossin & Sulaiman, 2015). Precision is used to measure the positive patterns that are correctly predicted from the total predicted patterns in a positive class and Recall is used to measure the fraction of positive patterns that are correctly classified (Hossin & Sulaiman, 2015). In this classification problem, correct prediction of both classes 'Yes' and 'No' are important. Therefore, accuracy seems to be more useful than precision and recall. However, as described before the F1 score is a better metric for classification problems. So this will be the most important metric to evaluate the performance of the results. An overview of the scores of the different decision trees is shown in 4.

	Accuracy	Precision	Recall	F1 Score
Complete Data	72,77%	71,42%	75,95%	73,61%
Department 2400	70,71%	63,63%	96,70%	76,75%
Department 2500	65,54%	76,38%	44,99%	56,63%
Department 2520	85,06%	81,52%	90,68%	85,86%
Department 2540	65,87%	94,98%	33,51%	49,54%
Department 2550	80,10%	100%	60,19%	75,15%

Table 4: Overview Performance Evaluation Decision Tree

The first row of the table shows that the performance of the decision tree for the entire data set is good. All evaluation measures have a score above 70%, but most importantly the F1 Score is 73.61%, which is not too bad. For the decision trees of the separate departments, the performance is much more variable. However, the fact that the size of sub-sets was below the determined minimum bound does not become clear in the scores. The decision tree for the departments 2500 and 2540 do have very poor scores for Recall and F1, with a lower accuracy at 65%. These lower scores are very likely to be caused by the smaller size of the data set. On the other hand, the scores of the other departments are not worse than the score of the total data set. Department 2520 has score that are all above 80% with an F1-Score of 85.56%, which is pretty decent. In the end, it seems that the issue of the smaller size does not influence the performance of the decision tree as much as expected, which is quite surprising. The size of the sub-sets are quite a bit below the minimum bound determined earlier and it was expected to heavily impact the results. However, this turned out to not be the case.

8 Ideal Situation and Improved Decisions

In this chapter, the ideal situation for the three selected decisions will be described. Each of these will have a newly created decision model that indicates how the decisions would need to be made in the ideal situation. It will include the level of automation for each decision. Based on this ideal situation it can be identified which changes are required to achieve it, sub-questions Q3 and Q4 can be answered. The figures of the DRD and Decision Tables shown in this chapter are different compared to the ones of the current situation in Chapters 5 and 6, but there are also some similarities. To identify the changes from the original situation new and different inputs and sub-decisions are grouped in a box in the DRDs. For the decision tables an extra annotation column is added that specifies when rules have a different output from the original situation. In the last part of this chapter the comparison of Total Order Time and Service Level is made between the current and ideal situation to answer the sub-question Q5.

8.1 Decide on Service

The first decision that was modeled is 'Decide on Service'. In this decision, Fokker Services determines what they are going to do with the unserviced exchange components they received from the customer. The decision is a bit more complex than the decision to service the component or cancel the service of the component. The service can also be delayed or an alternative for the service can be selected. To make this decision and determine which option to select, several input data are required. Three of the required input are small sub-decisions that requires other input. The DRD of the decision and its input is shown in figure 32. The first sub-decision of 'Service Cost Difference' is a comparison of the expected cost with the current market value of the component that uses the following expression to calculate the difference: *'Current Market Value' - 'Expected Cost'*. The sub-decision of 'Cost Difference Alternative Component' is similar and compares the expected cost of the component with the current price of the alternative based on the following expression *'Current Market Price' - 'Expected Cost'*. The third sub-decision 'Determine Required Capacity Repair Shop' is used as an indication of how busy the internal shop is. In this sub-decision, the expression *'Work Remaining in Shop' / 'Employee Hours Available' * 100* is used to calculate the percentage of the capacity that is required to perform all the remaining work in the internal shop. Besides these three sub-decisions the main decision also requires input for the 'Stock Need' and the 'Current Stock Level', which is information on the demand prediction and stock level of the relevant component.

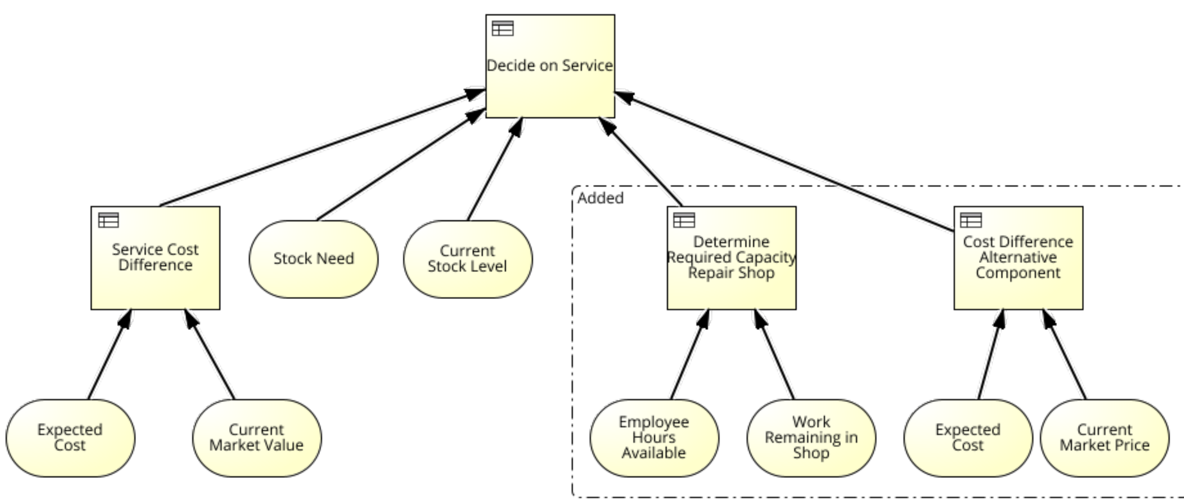

Figure 32: Ideal Situation Decide on Service

As the decision depends on several inputs, there are many possible combinations of factors that lead to a certain outcome. A set of decision rules was created that determines how the input will lead to a certain outcome of the decision. The decision table with all rules is shown in 33. As the table shows some inputs have more impact on the end decision than others. The most important one is probably the 'Service Cost Difference', which determines whether it will be profitable for the organization to service the component. If it is profitable, the expected cost for the service is lower than the current market value of the component after the service, and the 'Service Cost Difference' is larger than 0. If this is the case, the component will always be serviced.

However, the other criteria will influence when this is done. The main reason for this is that sending all components for service to the internal shop immediately can create an overload of work. To determine whether that would be the case the input of 'Determine Required Capacity Repair Shop' is required. This is a percentage of the current capacity that is required to perform all remaining work in the internal shop. The percentage indicates how busy the current situation in the internal shop is. A low number means that there are a lot of employees available to perform the service while a very high number indicates that there is a lot of work present in the internal shop already. In the ideal situation you want to prevent the overload of work in the internal shop, as it causes delays and longer waiting times. Since other components from other processes can also enter the same internal shop some margin with the capacity will be applied. In the end, these components that require service are not from customers and have no immediate deadline in terms of when it needs to be finished. This is different for the components from other processes in the internal shop. They need to be serviced directly for the customers and therefore need to be finished in time to not exceed the deadline. Since one of the main goals of Fokker Services is to match the predefined service levels for the customer, these orders that are directly influencing the service level should have priority in the internal shop. Therefore, if the required capacity is very high, the components should not be serviced immediately.

If the capacity usage is below 85%, the margin is considered large enough to not cause issues regarding an overload of work and the component will be serviced. However, if the percentage is above 85% it does not automatically delay the service for a later moment. If there is predicted demand for that component or the current stock level is very low, the component still needs to be serviced quickly. Not being able to deliver it on demand for other future orders is something you want to prevent in the ideal situation as well. So as rules 2 and 3 in the table show, if the 'Stock Need' = 'true' or the 'Current Stock Level' = 'insufficient', the component will still be serviced. So the service of the components will only be delayed if there is no predicted demand, and there is still sufficient stock available.

	Inputs					Outputs	Annotations
A	Service Cost Difference	Determine Required Capacity Repair Sh...	Stock Need	Current Stock Level	Cost Difference Alternative Component...	Service decision	Changes
	Number	*Number*	*Boolean*	*{Sufficient,Insufficient}*	*Number*	*{Service,Delay Service,Cancel Service,Select...*	***Informational***
1	≥ 0	< 85	-	-	-	Service	
2	≥ 0	≥ 85	= true	-	-	Service	
3	≥ 0	≥ 85	-	= Insufficient	-	Service	
4	≥ 0	≥ 85	= false	= Sufficient	-	Delay Service	New/Different Output
5	< 0	-	= true	-	< 0	Select Alternative	New/Different Output
6	< 0	-	-	= Insufficient	< 0	Select Alternative	New/Different Output
7	< 0	-	= false	= Sufficient	< 0	Cancel Service	
8	< 0	-	-	-	≥ 0	Cancel Service	

Figure 33: Ideal Situation Decide on Service Table

On the other side, if the 'Service Cost Difference' is smaller than 0 it does not automatically mean the service is not performed. In these cases the cost to service the components are higher than the value of the component afterwards and it is not profitable to service the component. A very important input for these cases is the sub-decision 'Cost Difference Alternative Component'. This compares the expected service cost with the price for alternatives of the same component. For example, it compares the price of similar serviced components on the market. If the price of these alternatives is lower than the expected cost, it is more profitable to choose this alternative over servicing the components. However, also the 'Stock Need' and 'Current Stock Level' are influencing the outcome. As rules 5 and 6 in the table show, the 'Stock Need' needs to be 'true' or the 'Current Stock Level' needs to be 'insufficient' to select the alternative. If any of these two is not different, servicing the component or arranging the alternative for the component does not provide value and the service will be canceled. Besides that, there is no suitable alternative available, so all of them are more expensive than the expected service cost, it is not profitable to do anything for this component and the service will be canceled.

8.1.1 Overview Changes Decide on Service

For the decision 'Decide on Service' the biggest difference is to include the required capacity of the internal shop. Based on this information it is possible to delay some of the services if the workload in the shop is too high and they are not necessary immediately. Another important change is to include the cost difference between alternatives. Instead of regular service of the component, there are many possible alternatives available such as purchasing serviced components for the market. If the service of the components was required soon because of expected demand, the components were serviced. Even if the service of the components was not profitable. By considering these alternatives, it is possible to not need to service these components that are not profitable. Since the data sets used in this research did not contain information about prices of orders, it was not possible to determine the number of orders affected by these changes.

8.2 Create Price Quote

The second decision for which the ideal situation was modeled is 'Create Price Quote'. This is another very complex decision that requires a lot of information as input. A DRD of the decision with all its input is shown in figure 34. For this decision, it is not possible to create a decision table. The reason for this is that many different types of components can occur within the process. These different types of components have a different range for the price. It would only be possible to create a decision table for a single type of component. Since Fokker Services maintains over 1000 different components, creating this amount of decision tables is simply impossible.

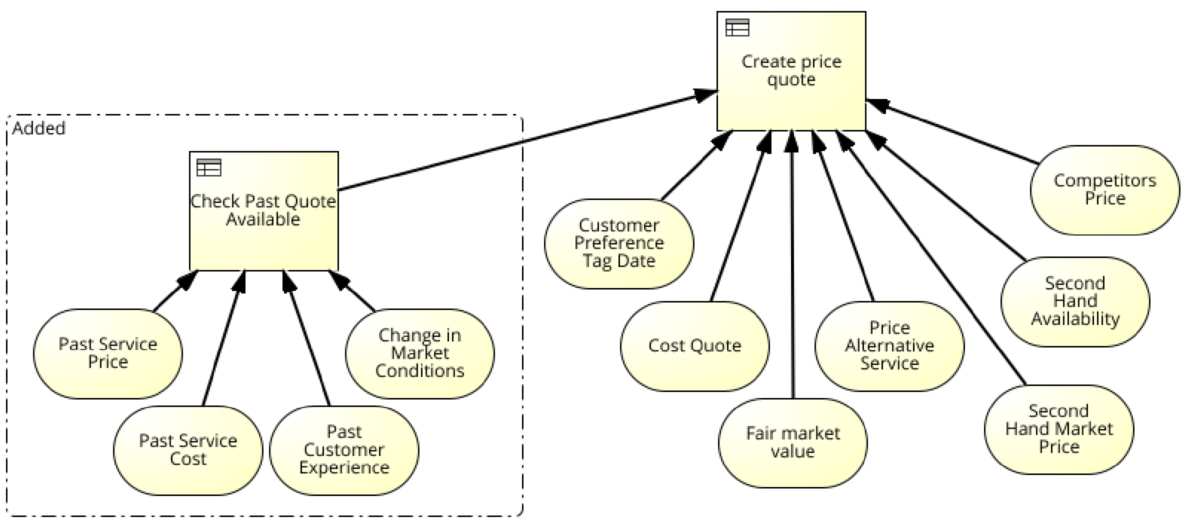

Figure 34: Ideal Situation Create Price Quote

As the DRD shows there is a sub-decision 'Check Past Quote Available'. For this sub-decision, a decision table can be created, which can be found in figure 35. The outcome of this sub-decision is one of the most important inputs of the main decision. The reason for this is that the output determines to which extent other inputs are required in the main decision. In the ideal situation, information and data from past quotes are used in the determination of future quotes. This might not always be possible, for example, when Fokker Services need to service a component they never serviced before, there is of course no past data available.

A	Inputs								Outputs
	Past Service Price		Past Service Cost		Past Customer Experience		Change in Market Conditions		Past Quote Experience
	Boolean		Boolean		{Positive,Negative}		Boolean		{Available and Usable,Available and Chang...
1	=	true	=	true	=	Positive	=	false	Available and Usable
2	=	true	=	true	=	Positive	=	true	Available and Change
3	=	true	=	true	=	Negative		-	Available and Change
4	=	false		-		-		-	Not Available
5		-	=	false		-		-	Not Available

Figure 35: Ideal Situation Sub-Decision Table

The sub-decision needs input on the availability of past data, whether customers were pleased with the past quote, and whether there are changes in the market conditions. Based on this input there are three possibilities for the sub-decision. The information is available and usable for the new price quote, the information is available, but it requires several changes before it can be used or the information is not available at all. To be able to use the past data, both information on the Past Service Price and Past Service Cost need to be available. If any of the two is missing, it will automatically result in the outcome 'Not Available' and no past data can be used in the main decision. However, if all the data is available it does not automatically lead to the outcome that the past data is usable. It then also depends on how the customer experienced the offer and whether the market conditions are changed. If the customer experienced the received price quote as positive, it means they thought the price they received for the service was fair. Then it is also safe to assume that they would accept the same offer again if the conditions are the same. When the experiences of the customer were negative it is a strong indication that they would not accept the same offer again. To be able to make them accept it, changes to the price quote are required and the same price cannot be used again. A similar situation occurs when the market conditions have changed. For example, a huge shortage of components arises over the past weeks which caused a severe increase in the price. Then offering the same price quote again could not be beneficial again and you want to change the prices according to the market changes. In that case, the past data needs to be changed and cannot be used again directly. Only when both the customer experience is positive and there are no changes in market conditions the outcome will be 'Available and Usable'.

As mentioned before the outcome of the sub-decision will have a lot of influence on the main decision. The decision is very complex as it depends on a lot of input. However, when past data is available not all of the input is required to make the decision, which makes the decision less complex. For example, if a similar component was serviced a few weeks ago with service, there is already past data available for this component. If the customer was satisfied with that offer and the market conditions did not change, a similar price quote can be created again for the new price quote. It will not be useful to consider all inputs again as that was done a few weeks ago already by creating the past price quote. Doing it again would lead to a similar outcome, since conditions have not changed and would therefore be a waste of time. Even in cases where market conditions have changed, making a new decision from scratch again would be not very efficient. Past data is available and forms a decent base for the new price quote. Instead of considering all inputs again, the focus should be on why the conditions were changed and what effect it will have on a price quote. Based on these finding the past price quote can be adjusted to finalize the decision. A similar situation happens when the customer experience turns out to be negative. Then the focus of the decision should be finding the reasons for that and changing the past price quote accordingly to prevent that for the new price quote. Only when there is no past data available, because it is a new component for example, all other inputs should be considered in the decision.

8.2.1 Overview Changes Create Price Quote

For the decision of 'Create Price Quote' there is one big change that is required. In this decision, the data from past decisions should be available. In the current situation, these are not widely available for these decisions, which causes the decision to be made from scratch all the time. Since there is no data available whether past orders were useful for a new decision, it is also impossible to determine exactly how many orders are affected by the changes. Especially for the orders that falls into the group of rules 1, 2 or 3 in figure 35. For these orders it is not possible to determine when the data form past orders would be useful or when it requires changes. However, it is possible to determine whether orders should have past data available and when it is not available. In total there were only 2609 unique component types while there were many more orders. For each component type, there has been a first time that Fokker Services is receiving it. In these cases there is no past data available. However, every other order is from a type that has been serviced before, which means that past data should be available. This means that it is possible to say how many orders would be in the group of rules 1, 2 or 3 and which are in the group of rule 4 and 5. An overview of the exact numbers and percentage is shown in 5. So, if the past data is available and it is still relevant, it can be used to create the price quote for similar components. By doing this a lot of time is saved in the decision-making. Even though it is not possible to say how many have relevant data, for 94% of the orders can potentially made less time-consuming.

	Output	Number of Orders affected by Change
Rule 1	Available and Usable	43035 (94%)
Rule 2	Available and Change	
Rule 3	Available and Change	
Rule 4	Not Available	2609 (6%)
Rule 5	Not Available	

Table 5: Overview of Orders Affected by Changes

8.3 Decide on In-house or Outsource

The final decision where the ideal situation was modeled is 'Decide on In-house or Outsource'. As the name of the decision shows, it determines when a component should be serviced in-house or outsourced. Besides these two options, there are no other possible outcomes for the decision. The decision includes two sub-decisions with each its input and the inputs 'Type of Request' and 'Expected Waiting Time'. An overview of this is shown in the DRD in figure 36.

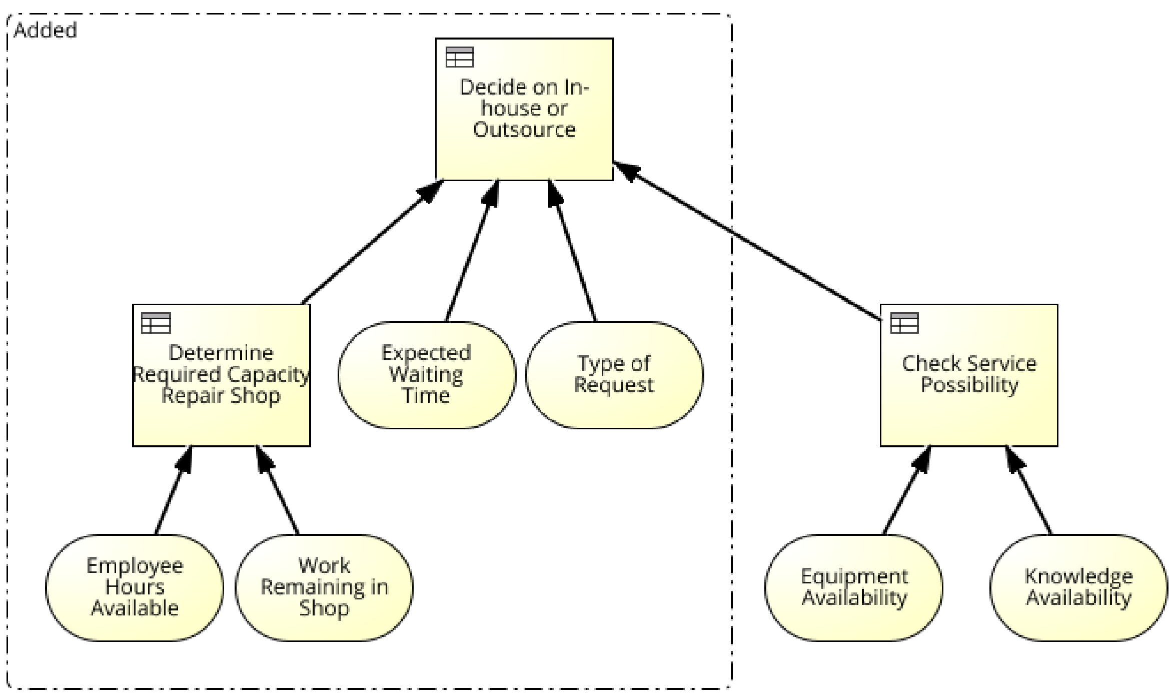

Figure 36: Ideal Situation Decide on In-house or Outsource

The sub-decision 'Determine Required Capacity Repair Shop' is the same as in the decision 'Decide on Service' that was modeled earlier in this chapter. So it provides a percentage for the required capacity based on the expression: *'Work Remaining in Shop' / 'Employee Hours Available' * 100*. The sub-decision 'Check Service Possibility' is used to see whether it is technically possible for Fokker Services to service the component. For that to be possible, Fokker Services must have both the Equipment and Knowledge available within the internal shop. If any of the two is not available, the component cannot be serviced in-house. These rules can be included in a simply decision table which is shown in figure 37. The output of this sub-decision is then used as input in the main decision.

A	Inputs: Knowledge Availability	Inputs: Equipment Availability	Outputs: In-house Service Possible
	Boolean	*Boolean*	*Boolean*
1	= true	= true	true
2	= false	-	false
3	-	= false	false

Figure 37: Sub-Decision Table Check Service Possibility

The sub-decision is one of the main inputs in the decision. If Fokker Services cannot service the component in-house, there is simply no other option than to outsource the component. In these cases, the other inputs are not relevant anymore. When it is possible to perform the service in-house the other inputs will be the deciding factor in the decision. The type of request does also have a big influence on the decision. If the component is part of an exchange request, the outcome of the decision will also be to outsource it. The main reason for this is that Fokker Services need to manage the amount of work in the internal shop. Since the components of the exchange requests are the property of Fokker Services they can easily decide to outsource the components. For normal Service Requests, many customers specifically choose Fokker Services as a service provider for their components. So outsourcing these may lead to discontent among the customers. Therefore, based on this criteria you would like to minimize the number of requests that are outsourced.

However, also for normal service requests some cases need to be outsourced. This is mainly to prevent an overload of work in the internal shop. To determine the amount of work in the shop two inputs will be used, 'Expected Waiting Time' and 'Required Capacity Repair Shop'. If there is a quite big overload of work in the shop such that the expected waiting time of an order will be more than 21 days, then the component needs to be outsourced. The 21 days are the allowed throughput time for the orders. So if it takes more than 21 days before they start working on the order, it will never be finished in time. Therefore, it is better to outsource it immediately. For the cases where the expected waiting time is less than 21 days, the required capacity for the shop will be decisive. If the required capacity to complete all orders is above 100%, there is an overload of work in the shop. Even though the expected waiting time may be below 21 days, there is a very low chance that the service will be finished on time. Therefore, it is better to also outsource the components in these situations. Only when the required capacity is below 100%, it will be no issue for the shop to have the service completed in-house. A summary of all conditions when the service should be performed in-house or not is shown in the decision table in figure 38.

A	Inputs: Expected Waiting Time	Inputs: Determine Required Capacity Repair Sh...	Inputs: Type of Request	Inputs: Check Service Possibility	Outputs: In-house or Outsource	Annotations: Changes
	Number	*Number*	*{Service Request,Exchange Request}*	*Boolean*	*{In-house,Outsource}*	***Informational***
1	-	-	-	= false	Outsource	
2	-	-	= Exchange Request	= true	Outsource	Different Output
3	≥ 21	-	= Service Request	= true	Outsource	Different Output
4	< 21	≥ 100	= Service Request	= true	Outsource	Different Output
5	< 21	< 100	= Service Request	= true	In-house	

Figure 38: Ideal Situation Decide on In-house or Outsource Table

A last final remark for the ideal situation is that instead of considering all data and orders together on one big pile, they should be considered separately for the different departments. In total Fokker Services has 8 departments across three different locations. All perform the service of components in a similar way. However, the big difference is that not all departments can perform service to all components. Departments are specialized in several types of components and do not perform service to components that other departments perform. Considering the departments as separate internal shops will be much better. For example, it would be better to consider the capacity of the departments separately. Some shops may have much more incoming orders than other shops. Therefore, the required capacity to perform all work and the expected waiting time of the different departments can be completely different. As these are important inputs for the decision, sub-optimal decisions are likely made for orders if the departments are not considered separately. Since the departments operate similarly on a high level, there are no differences in the process and the same decisions need to be made. Therefore, the decision model and decision table of 36 and 38 are applied for each department separately in the ideal situation.

8.3.1 Overview Changes Decide on In-house or Outsource

The decision of 'Decide on In-house or Outsource' is the one that requires the biggest changes. The main reason for this is that it is not considered as a decision in the current situation. Instead of a decision Fokker Services maintained a policy for this. Which is that all components were sent to the internal shop if they can perform the service. By considering this as an active decision again, several components can be outsourced, which should help reduce the overload of work in the internal shop. To be able to make this decision, there are several inputs and data needed. Two of the inputs are data about the required capacity for the shop and the expected waiting time of orders, which are two very important indications of whether the internal shop has a high workload or not. In the end the aim of considering this as an active decision again is to help solving the issue of the high workload in the internal shop.

If they consider this as a decision again, it will have effect on many orders as more will be outsourced. The rules 2, 3 and 4 in figure 38 all result in the option 'Outsource' while they result in 'In-house' in the current situation. For the exchange orders, it could easily be determined how many orders would be affected by the changes. However, for the other two different rules it was more difficult. The data sets did not include data about expected waiting time and required capacity. Instead, estimations were made for both these options. For expected waiting time, the actual waiting time of the orders was uses. So all orders with a total waiting time of more than 21 days were included in the group of rule 3. For the required capacity, the total hours of work remaining was used. This was compared to an estimated capacity that was based on the number of resources and the available working hours. The orders that made the total hours of work remaining go above that estimated capacity were included in the group of rule 4. An overview of the total number of orders and the percentage affected by these changes is shown in figure 5.

	Output	Number of Orders affected by Change
Rule 2	Outsource	6684 (15%)
Rule 3	Outsource	312 (1%)
Rule 4	Outsource	944 (2%)

Table 6: Orders Affected by Changes

8.4 Level of Automation

The three decisions modeled in this chapter were selected based on several criteria that indicate whether they are suitable for automation. Decisions that fulfill these criteria have a very high potential for automation in the future. In the ideal situation, you would therefore assume that all of these decisions are fully automated. As described earlier in this research, automation of decisions can be achieved with the Decision Management Approach and the Decision Models prescribe the decision logic for automated decision-making. Therefore, these created decision models provide a decent base for the first phase of the Decision Management Approach already. In the second phase of the implementation, the Business Rules and the management of them play a very important role. However, for the second modeled decision 'Create Price Quote', it was not possible to create general business rules that could be applied to all orders. Without the created Business Rules, it is impossible to fully automate these decisions with the Decision Management Approach. However, the sub-decision does have complete decision logic that can be implemented and managed. So, this part can be automated as well, which helps the efficiency of the decision-making a lot. Therefore, the decision 'Create Price Quote' will be partially automated in the ideal situation and human knowledge remains part of the decision. The other two decision 'Decide on Service' and 'Decide on In-house or Outsource' does not have the limitation of missing decision logic and can be automated completely.

8.5 General Changes to All Decisions

The ideal situation described for the decisions is different from the decisions in the current situation. Some of the changes apply to all decisions while others are very specific for the decisions. The previous part describes the level of automation for the decision in the ideal situation. This level of automation is one of the required changes to the decisions. In the current situation, none of the decisions are automated. Meanwhile, in the ideal situation the decisions 'Decide on In-house or Outsource' and 'Decide on Service' are fully automated and the decision 'Create Price Quote' is partially automated. Another general change that is relevant to the decisions is to include different data and information in the decision-making. Especially the data and information that indicate the level of workload and other information on the current situation of the internal repair shops. To be able to solve the problem of the high workloads it is a must to know the exact situation in the shop. Besides that, the focus should be on collecting and using data specifically required for the decisions instead of using a lot of data that might not all be relevant. The last very important general change is that decisions between different departments should be treated as separate shops and not together as one shop. Results from the implementation in the previous chapter show that it is very useful. Besides that, it also makes the most sense from a logical perspective. Even though the process is the same on a high level, many differences within the departments can cause differences in decision-making. For example, the same decision in another department can have a different ideal outcome if the current conditions are not the same. To a certain extent the departments operate as separate shops and to have optimal decision outcomes the departments should be treated like that.

8.6 Comparison of Total Order Time and Service Level

The last part remaining is to investigate the effect of the improved decisions on the process. The ideal situation for the decisions described in this chapter should lead to quicker and easier decision-making, but most importantly to better decision outcomes that should have a positive effect on the processes. Earlier this research it was determined that total order time and service level are the important performance measure for this process that are aimed to improve. How much effect improving the decision can have on the process will be investigated by comparing the total order time and service level. The data set of Work_Order_Status consists of the different statuses all orders had throughout the processes. This data set is used to identify how the orders would have been affected by the improved decisions and to predict the total order time and service level for this ideal situation. Finally, this total order time and service level for the ideal situation is compared with the ones from the current situation to answer the final research question.

8.6.1 Service Level Outsourced Orders

One of the biggest issues of the process of Fokker Services is the overload of work in the internal shops that cause delays for many orders. In the ideal situation, these delays do not occur as orders will be outsourced instead of serviced internally if the capacity of the internal shops is reached. Besides that, as figure 38 showed, there are many scenario's when orders will be outsourced. For orders of an Exchange Request, it even means the orders will always be outsourced. By outsourcing these orders it is not possible to simply assume all orders will be returned on time and result in a 100% service level for Exchange Orders. These other organizations can have issues and delays as well. It is therefore difficult to determine what the exact service level will be for these orders. However, according to the knowledge within Fokker Services, a service level of around 85% is assumed to be the average within the industry. Based on this the assumption is made that the outsourced orders will have an average service level of 85%. This will then lead to a service level of 85% for Exchange Orders in the ideal situation.

8.6.2 Service Level In-house Orders

To determine the service level for the orders that are serviced internally the data set of Work_Order_Status is used. With this data set it was possible to identify the waiting time and delays of the orders. It is important to distinguish the several reasons for the waiting time of the orders. The goal of the ideal situation is to not have any delays caused by the overload of work. Therefore, not all statuses of orders related to waiting time should be included. A good example of waiting time that should not be included is 'Awaiting Piece Parts'. In these cases, the required piece parts for the service are not available within the organization anymore and the mechanics are waiting for these parts to arrive before they can continue the service of the component. Situations like these may still occur when the issue of overload of work is solved. Therefore, only the statuses that are caused by the overload of work should be considered. The data set contains the following statuses that indicate waiting time that is caused by the overload of work: 'Awaiting Other Work Center', 'Awaiting Resources', 'Awaiting Test', 'Awaiting Test Systems', 'Ready for Inspection', and 'Ready to Build'.

These statuses together are a significant part of the waiting time of the orders that affect the throughput time and service level of the orders. In the ideal situation decisions are made to prevent the overload of work. If the decisions are made perfectly, these waiting times should not occur and orders should take less time to complete. The throughput time has been calculated again, but this time the waiting time mentioned in the previous paragraph have been excluded. Similar to the current situation orders that are finished within 21 days are considered on time. This results in a service level of 80% for the orders that are completed in-house.

8.6.3 Comparison Service Level

As described earlier in this chapter, the Exchange Orders are all outsourced in the ideal situation. This makes the service level for Exchange Orders the same as the service level for the Outsourced Orders with 85%. However, Customer Orders are both outsourced and serviced in-house in the ideal situation. In the ideal situation there are more situations that will result in orders to be outsourced. Based on this logic from the ideal situation, about 12% of the customer orders will be outsourced. Since the in-house orders have a service level of 80% and the outsourced orders have a service level of 85% again, combining these for all Customer Orders will lead to a service level of 81%. Table 7 below shows the service levels compared with the initial service levels of the current situation.

	Current Situation	Ideal Situation	Improvement
Customer Orders	66%	81%	15%
Exchange Orders	49%	85%	36%

Table 7: Comparison Service Level

9 Conclusion

The main objective of this research was to investigate the possibilities of how Fokker Services can improve its decisions within the organization. To investigate the possibility of the improvement of the decisions, the focus was put on the topics of Decision Mining and Decision Modeling with the DMN standard throughout this research. There was another side objective in this research that should contribute to the literature. Data quality is very important for Decision Mining and this research, investigated the effect of data quality on possible improvements in decision-making. To help achieve these research objectives the following research question was formulated: *How can decision mining techniques help Fokker Services improve their decisions within CMRO and Exchange Programs?*. In the remaining part of this chapter, this research question will be answered.

For the selected decisions in the processes of CMRO and Exchange Programs in this research the Decision Mining techniques did not provide much added value. One of the few interesting and useful findings was that the achieved decision tree showed that there are clear differences between the different departments. This suggests that the departments should indeed be considered separately and that for each department separate decision logic should be developed. Additionally, it was already tried to split the data set into sub-sets for the departments and use decision mining to find decision logic for each department. However, no useful decision logic was obtained from this. In the end, this is not surprising as Fokker Services does not have the data for separate departments.

The fact that Decision Mining did not provide much added value for the selected decisions in this research does not mean it will be completely useless at all for Fokker Services. As described before, it did indicate that there are clear differences between departments. If Fokker Services can collect data for separate departments, it should be possible to implement decision mining again and decision trees for each department. There have not been any big issues regarding the results of the decision tree. Even the smaller size of the sub-sets was not as big of an issue as expected. Therefore. this should not limit Fokker Services from implementing the Decision Mining technique for separate departments. Then the achieved decision logic will likely be very useful and form a very good base for better decision-making.

Even though the implementation of Decision Mining did not provide much added value for the improvement of the decisions, it was still possible to create an ideal situation for the decisions. If the selected decisions will be changed according to this ideal situation it will lead to a significant improvement in the performance of the process. More specifically, it will lead to much higher service levels of the orders in both the process of CMRO and Exchange Programs. These improvements do not immediately lead to achieving the target service level of 90%. However, it will not be too far away from that target anymore. Therefore, these improvements to the decisions will be very useful for Fokker Services and it is worth it to try to achieve this.

Besides the main objective, there was another objective to investigate the effect of data quality on decision-making. During this research, some interesting findings were obtained, especially with the implementation of the Decision Mining technique. It is well known that data quality is a big challenge for Decision Mining. Therefore, a quality assessment was performed for the data sets of Fokker Services. In this assessment it turned out that the data quality was relatively high and after a few steps of data cleaning the data set should be ready for the implementation. However, this was not the case as additional issues were identified that required some final data preparation.

The biggest issue was related to the way the data set has been imbalanced. It turned out that the data was extremely unbalanced which cause many issues for the creation of decision trees that are used within Decision Mining. There are many possible ways to balance the data set and solve this issue. However, this is a quality issue that was not included in any of the quality assessments. Since it is impossible to create good and reliable decision trees with imbalanced data, this is a critical issue for Decision Mining that should always be considered when checking the data quality. Besides that, the quality dimension of Relevancy is much more important for Decision Mining as well. In the quality assessment, the relevant data included much more than was used in the implementation. For example, order numbers and part numbers are considered as a very relevant data feature. However, decision trees use all the data included to create decision logic, even if it does not make sense from a logical perspective. Besides that, decision trees tend to become very large very quickly. Therefore, it is much more important to select only the relevant data for decision logic and not relevant data in general. So, if Decision Mining is used to try to improve decision-making, it can be concluded that the effect of data quality is very important to be able to obtain reliable and useful results.

9.1 Limitations

The biggest limitation was related to the investigation of the effect of the changed decisions. The ideal situation described for each decision includes many changes compared to the current situation. By implementing these changes many orders would have been affected. However, it was not possible to exactly determine which orders were affected by all the changes. For most of the changes in decision logic a estimation could be made for the number of orders that are affected by these changes. However, one of the biggest general changes is the consideration of separate departments. Fokker Services only have data available that considers all orders together. The only relevant thing that is known is in which department the component has been serviced. Besides that, all data included in the data sets were not related to the departments. A good example of this was the number of orders and amount of remaining work within the organizations. This was only available for all orders in the entire organization together and not for orders within departments. Therefore, it was not possible to predict the effect of this change.

9.2 Future Research & Recommendations

In this research, it was one of the objectives to investigate the effect of data quality on decision-making. It became clear that the quality issue of imbalanced data is such important that it prevents the results to be any useful. This was the case even when the quality issues in the data quality assessment were not significant at all. In general, the data set that was used can be considered of quite high quality. However, it was not sufficient for the decision mining technique to be able to provide useful and reliable results. So, additional data preparation was still required before the implementation. Therefore, the most interesting topic for future research would be to develop a data quality assessment specifically for decision mining. Even though it was already known that data quality is one of the biggest challenges of decision mining, there is still no information and literature available about the requirements for data quality. As this research showed, there are some very important quality dimensions for the implementation of Decision Mining.

Another potentially interesting topic for future research could be to investigate if different decision mining techniques work perform better with certain quality issues. Many different techniques have been developed throughout the years and some frameworks have been developed to compare the techniques on performance. However, for this comparison, it is assumed that data quality issues are not a limiting factor for the implementation. In practice this is usually completely different as data quality issues are the biggest limitations of achieving good results with decision mining techniques. In this research one of the original decision mining techniques was implemented but many others have been developed in the years afterwards. Therefore, these newer and more recent techniques may have fewer issues with some specific quality issues. Knowing if certain techniques handle certain quality issues in a better way, could be very useful to select the best technique for a specific case. Therefore, it can be another very interesting topic for future research.

To conclude this research some recommendations are done on how Fokker Services can continue improving their decisions in the future. In this research, only three decisions were selected, but there are many more decisions that Fokker Services like to improve. The first recommendation is to collect data more specifically for decision-making. There is a lot of data available within the organization, but the collection and use of data throughout the organization are not focused on using the data for decision-making. It became clear during this research that the lack of correct and available data prevented the results of the decision logic to be useful. A good example of this was the incorrect data for the separate departments. For most of this data, it should not be impossible to collect it within the organization if they put the focus on it. When this data is available and correct the decision mining implementation should be able to provide Fokker Services with useful decision logic to help improve their decisions.

The only remaining problem for Fokker Services is then how to determine which data they need to collect and use. The recommendation for this goes together with Fokker Services' issue that they need the knowledge for a general approach on how to improve and automate their decisions. To achieve this the three phases of the Decision Management Approach are very useful for Fokker Services. This is a complete approach that starts with the identification of the decisions and ends with the improvement of decisions. The approach includes every single step that is required to improve and automate decisions. Besides that, the steps taken in this research have many similarities with the approach. The biggest difference is that decision mining is not a specific step of this approach. However, decision mining can be used very well as a complementary step to the Decision Management Approach. Decision Logic in the form of Business Rules plays an important role in the approach. If Fokker Services can solve the data quality issues, it is very well possible to achieve this Decision Logic with Decision Mining and use it to further improve their decisions.

References

Baesens, B., Setiono, R., Mues, C., & Vanthienen, J. (2003). Using neural network rule extraction and decision tables for credit-risk evaluation. *Management Science*, *49*(3), 312–329. https://doi.org/10.1287/mnsc.49.3.312.12739

Bazhenova, E., Buelow, S., & Weske, M. (2016). Discovering Decision Models from Event Logs. https://doi.org/10.1007/978-3-319-39426-8

Bazhenova, E., & Weske, M. (2016). Deriving decision models from process models by enhanced decision mining. *Lecture Notes in Business Information Processing*, *256*, 444–457. https://doi.org/10.1007/978-3-319-42887-1{_}36

Brown, M. L., & Kros, J. F. (2003). Data mining and the impact of missing data. *Industrial Management and Data Systems*, *103*(8-9), 611–621. https://doi.org/10.1108/02635570310497657

Chiheb, F., Boumahdi, F., & Bouarfa, H. (2019). A new model for integrating big data into phases of decision-making process. *Procedia Computer Science*, *151*, 636–642. https://doi.org/10.1016/j.procs.2019.04.085

Cichy, C., & Rass, S. (2019). An overview of data quality frameworks. *IEEE Access*, *7*, 24634–24648. https://doi.org/10.1109/ACCESS.2019.2899751

De Leoni, M., Reijers, H. A., & Van Der Aalst, W. M. P. (2016). *Decision Mining Revisited-Discovering Overlapping Rules* (tech. rep.). www.tue.nl/taverne

De Smedt, J., Hasic, F., vanden Broucke, S., & Vanthienen, J. (2017). *Towards a Holistic Discovery of Decisions in Process-Aware Information Systems* (J. Carmona, G. Engels, & A. Kumar, Eds.; Vol. 10445). Springer International Publishing. https://doi.org/10.1007/978-3-319-65000-5

De Smedt, J., vanden Broucke, S., Obregon, J., Kim, A., Jung, J.-Y., & Vanthienen, J. (2017). *Decision Mining in a Broader Context: An Overview of the Current Landscapeand Future Directions* (M. Dumas & M. Fantinato, Eds.; Vol. 281). Springer International Publishing. https://doi.org/10.1007/978-3-319-58457-7

Goel, K., Leemans, S. J., Martin, N., & Wynn, M. T. (2022). Quality-Informed Process Mining: A Case for Standardised Data Quality Annotations. *ACM Transactions on Knowledge Discovery from Data*, *16*(5). https://doi.org/10.1145/3511707

Hon, K. K. B. (2005). *Performance and Evaluation of Manufacturing Systems* (tech. rep.).

Hossin, M., & Sulaiman, M. (2015). A Review on Evaluation Metrics for Data Classification Evaluations. *International Journal of Data Mining & Knowledge Management Process*, *5*(2), 01–11. https://doi.org/10.5121/ijdkp.2015.5201

Jagadeesh Chandra Bose, R., Mans, R. S., & van der Aalst, W. (2013). *2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM).* IEEE.

Jonkers, M. (2022). *MSE15 Intership Identifying and Analyzing Decisions in CMRO at Fokker Services* (tech. rep.).

Jouck, T., de Leoni, M., & Depaire, B. (2019). A Framework to Evaluate and Compare Decision-Mining Techniques. *Lecture Notes in Business Information Processing*, *342*, 482–493. https://doi.org/10.1007/978-3-030-11641-5{_}38

Leewis, S., Smit, K., & Zoet, M. (2020). Putting Decision Mining into Context: A Literature Study. *Lecture Notes in Information Systems and Organisation*, *38*, 31–46. https://doi.org/10.1007/978-3-030-47355-6{_}3

Loshin, D. (2013). Business Processes and Information Flow. In *Business intelligence* (pp. 77–90). Elsevier. https://doi.org/10.1016/b978-0-12-385889-4.00006-5

Lovell, B. C., & Walder, C. J. (2007). Support vector machines for business applications. In *Mathematical methods for knowledge discovery and data mining* (pp. 82–100). IGI Global. https://doi.org/10.4018/978-1-59904-528-3.ch005

Martin, N. (2021). Data Quality in Process Mining. https://doi.org/10.1007/978-3-030-53993-1{_}5

Morgan, J., Dougherty, R., Hilchie, A., & Carey, B. (2003). *Sample Size and Modeling Accuracy with Decision Tree Based Data Mining Tools SAMPLE SIZE AND MODELING ACCURACY WITH DECISION-TREE BASED DATA MINING TOOLS* (tech. rep.).

Omg. (2015). *An OMG ® Decision Model and Notation TM Publication Decision Model and Notation* (tech. rep.). https://www.omg.org/spec/DMN20191111/DMN13.xsd

Rozinat, A., & Van Der Aalst, W. M. P. (2006). *Decision Mining in Business Processes* (tech. rep.). www.processmining.org.

Rozinat, A. (2020). Do I Need to Remove Outliers for My Process Mining Analysis?

Schroten, R. (2020). Determining the work scope of work orders using historic aircraft component maintenance data at Fokker Services B.V.

Shelke, M. S., Deshmukh, P. R., & Shandilya, V. K. (2017). A Review on Imbalanced Data Handling Using Undersampling and Oversampling Technique. *International Journal of Recent Trends in Engineering and Research*, *3*(4), 444–449. https://doi.org/10.23883/ijrter.2017.3168.0uwxm

Sug, H. (2009). *An Effective Sampling Method for Decision Trees Considering Comprehensibility and Accuracy* (tech. rep.). http://kowon.dongseo.ac.kr/~sht

Taylor, J. (2012). *Decision management systems: a practical guide to using business rules and predictive analytics*. IBM Press/Pearson Education.

Taylor, J. (2014). *The Decision Management Manifesto An Introduction* (tech. rep.).

Taylor, J. (2016). *Decision Requirements Modeling with DMN* (tech. rep.).

Taylor, J. (2017). *Maximizing the Value of Business Rules* (tech. rep.).

Taylor, J. (2021). *Framing Analytic Requirements Achieve analytical, data-driven decisions with decision modeling and the Decision Model and Notation (DMN) standard Dedicated to DecisionsFirst TM* (tech. rep.). www.decisionmanagementsolutions.com

Taylor, J., & Purchase, J. (2016). *Real-World Decision Modeling with DMN*.

Van Der Aalst, W., Adriansyah, A., Alves De Medeiros, A. K., Arcieri, F., Baier, T., Blickle, T., Chandra Bose, J., Van Den Brand, P., Brandtjen, R., Buijs, J., Burattin, A., Carmona, J., Castellanos, M., Claes, J., Cook, J., Costantini, N., Curbera, F., Damiani, E., De Leoni, M., . . . Wynn, M. (2012). *Process Mining Manifesto* (tech. rep.).

Van Der Aalst, W., Mylopoulos, J., Rosemann, M., Shaw, M. J., & Szyperski, C. (2012). *Lecture Notes in Business Information Processing 125 Series Editors* (tech. rep.).

van Aken, J., Berends, H., & van der Bij, H. (2012). *Problem Solving in Organizations* (2nd). Cambridge University Press.

Verhulst, R. (2016). *Evaluating Quality of Event Data within Event Logs: An Extensible Framework (Master Thesis)*. Eindhoven University of Technology.

Wang, R. Y., & Strong, D. M. (1996). *Beyond Accuracy: What Data Quality Means to Data Consumers* (tech. rep. No. 4).

A Appendix A - Description Services

- **Aircraft MRO** is the service where the heavy maintenance and repairs on the aircraft are performed. The maintenance and repairs are always performed by qualified technicians of Fokker Services and occur in the aircraft hangar.
- **Component Maintenance Repair and Overhaul** (CMRO) is the service where Fokker Services performs the maintenance or repair of aircraft components. The main difference compared to Aircraft MRO is that the failed components are removed from the aircraft and repaired in repair shops. In these cases, usually, the components are sent to the repair shops by the customers and sent back once the parts are repaired.
- **Parts Availability** is a logistical operation and therefore a completely different service compared to the MRO services. Fokker Services operates and manages warehouses with spare parts for customers around the world. Fokker Services manages the stock of a wide range of spare parts that are delivered on time when the customer requests them.
- **Exchange Programs** is an extension of the Parts Availability service. Not only the spare parts will be delivered from stock, but Fokker Services also receives the failed components in return. If there is a demand for a certain part, it gets delivered immediately from stock. The failed part is sent back to the repair shop where it will be repaired or disposed of. In the case it is disposed it does not have any value anymore. If the part has been repaired, it will be replenished to the stock again and can be delivered to customers in the future. It may also be possible that repaired parts are sold on the second-hand market, however, this is a very rare occasion and basically never happens. Fokker Services maintains strict contracts with their customers that include very high service levels for the parts. An alternative way of replenishing the stock is by purchasing the same component on the second-hand market. To match the requirements of the service levels, a combination of both replenishment options is used.
- **Engineering Services** is another completely different service provided by Fokker Services. It consists of providing the customers with certain technical or operational expertise for their concerns. Engineers of Fokker Services provide their engineering expertise to help solve the customers' concerns.
- **Aircraft Modifications** is a service where Fokker Services uses its knowledge and engineering expertise to modify aircraft according to the customers' wishes. A wide range of modifications is performed. Common examples of modifications are an upgrade or installation of the newest equipment and systems.
- **Defense** is a type of service which is created for serving military customers. The services provided to military customers are not much different from the ones previously mentioned. Defense service include maintenance, modifications many other services. However, military customers is a special group of customers that has a lot of special requirements. For example, a lot of work needs to be strictly confidential. This special requirement is what really distinguishes the Defence service from the other services.

B Appendix B - BPMN Repair Unit

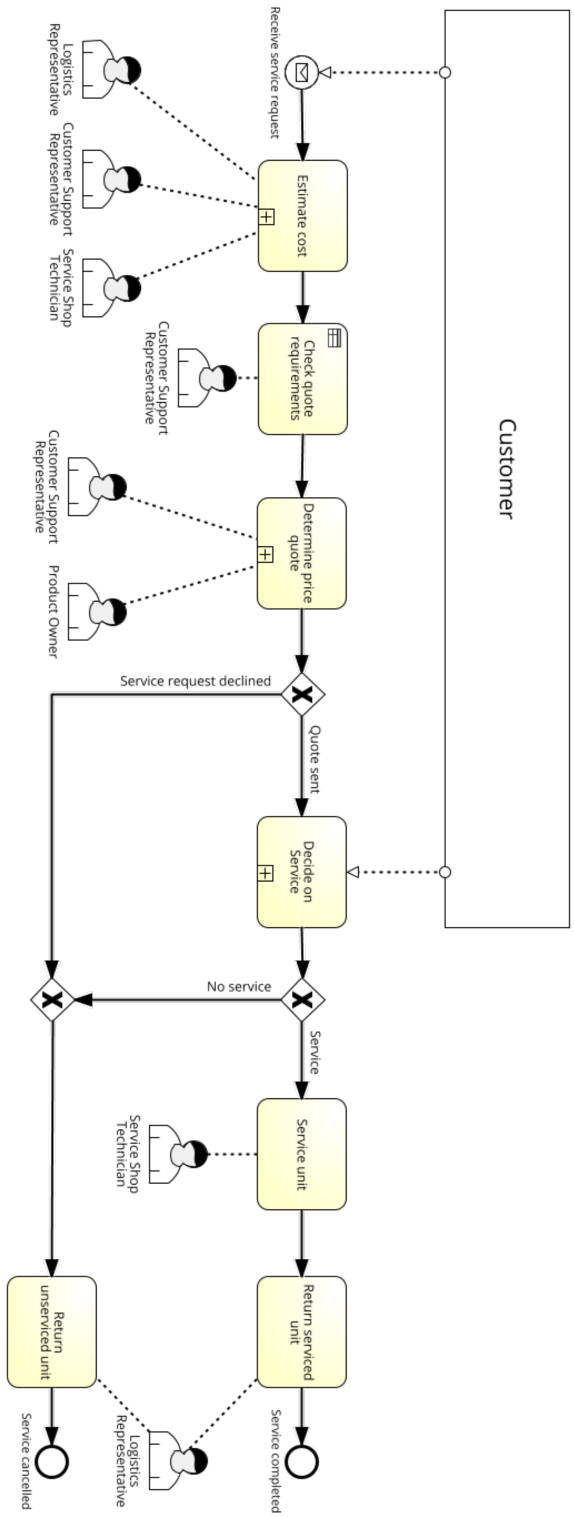

Figure 39: BPMN Normal Service

C Appendix C - BPMN Exchange Unit

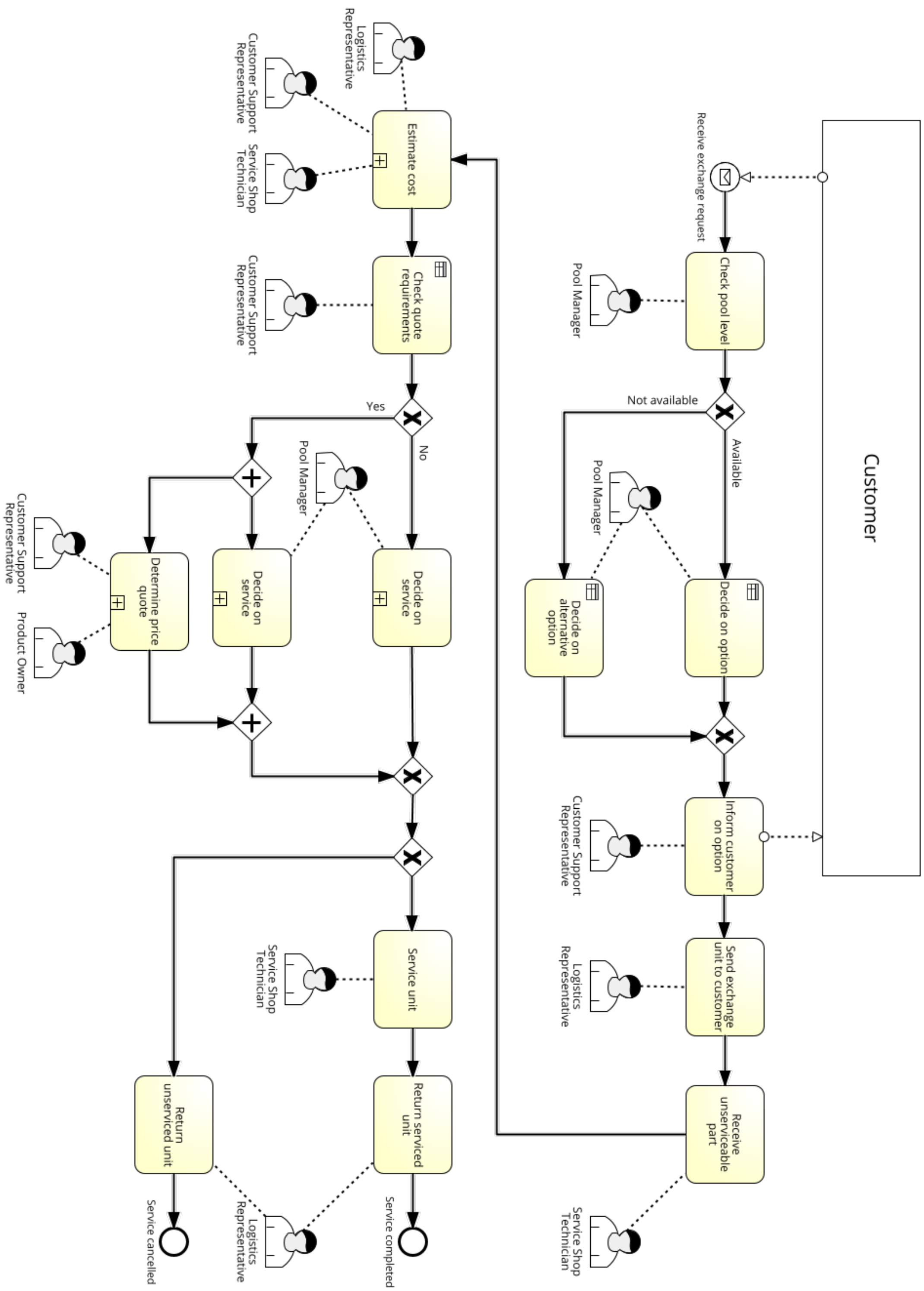

Figure 40: BPMN Exchange Unit

D Appendix D - Sub-Processes

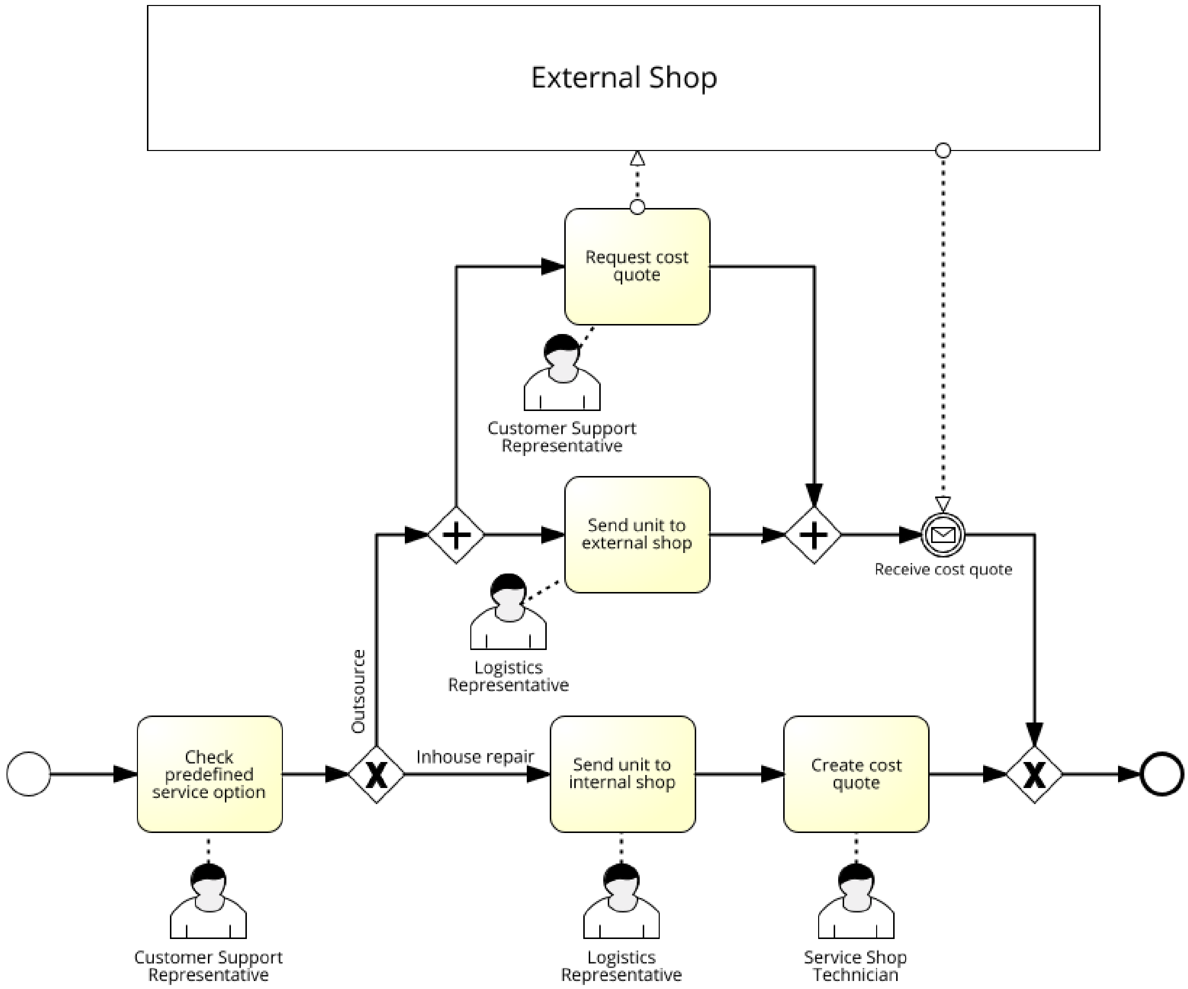

Figure 41: Sub-process Estimate Cost

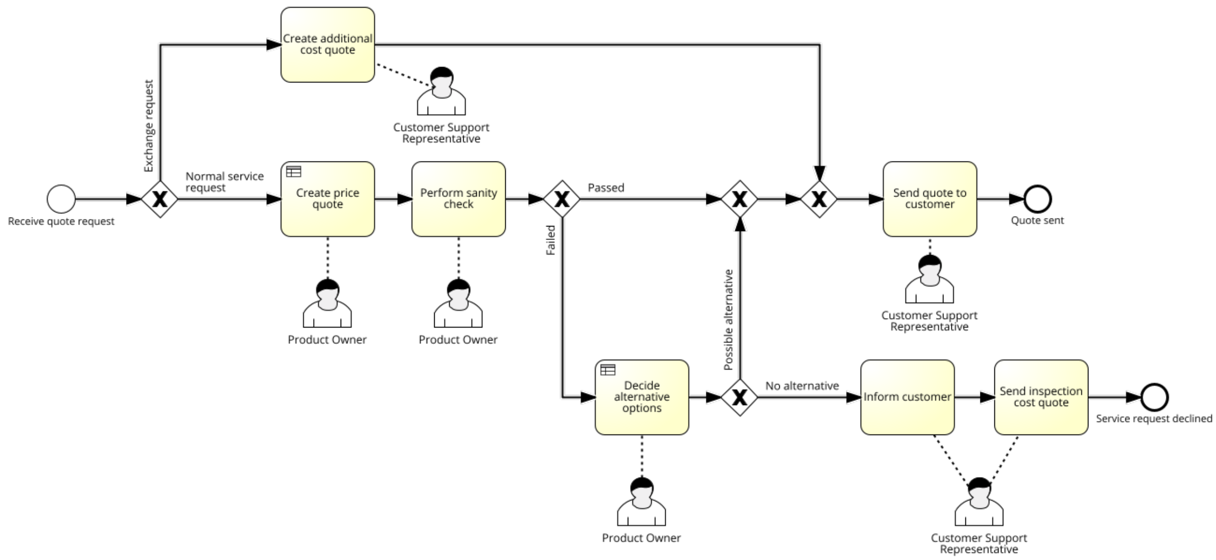

Figure 42: Sub-process Determine Price Quote

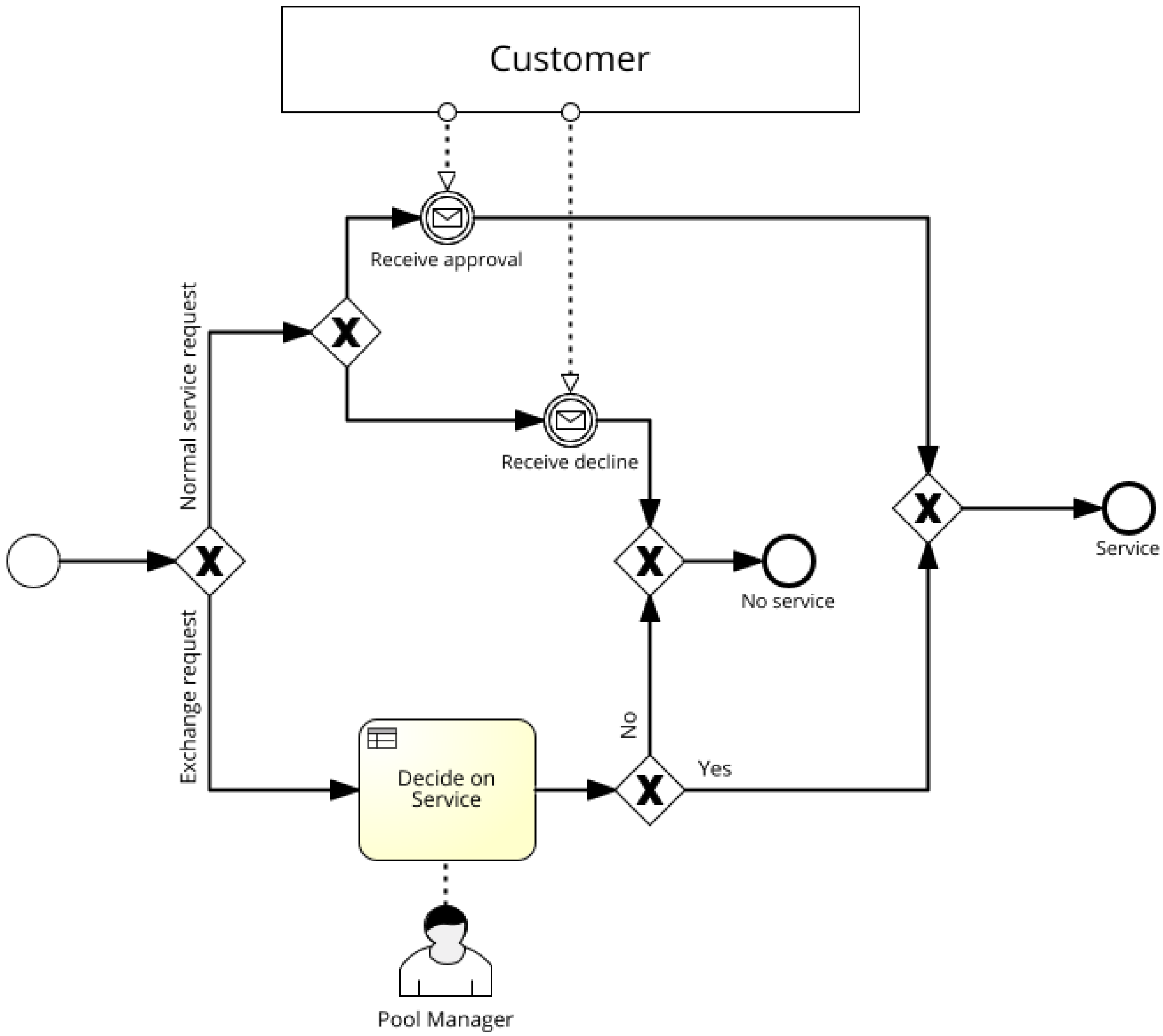

Figure 43: Sub-process Decide on Service

E Appendix E - Data Quality Assessment Measurement Method

- Completeness will be measured by calculating the percentage of data entries that are missing in the data set. Data entries that contain values such as "empty" or 'NaN' will also be considered as missing data. A score between the value of 1-10 will be given for completeness. For each percentage missing data, 1 point of the score will be deducted. For example, it the data set contains 2% missing entries, the score for completeness will be 8.
- Uniqueness will be measured by counting the number of unique values in the data set per attribute. It is not possible to apply a score to this dimension. Therefore, the total number of unique values will be the output for this dimension. Measuring duplicates and giving a score for this measure is not applicable in this case. For event logs, they indicate unnecessary information, which is therefore seen as an issue. However with the data sets of Fokker Services, this is not the case as it may provide important information for decisions. Therefore, duplicates will not be used as a quality dimension.
- Timeliness will be measured by checking if the data falls within a specified time frame. This time frame is chosen to select only relevant data based on how old the data is. For example, if only data from last 3 years is required, all that is older will fall outside of the time frame. Applying a score for timeliness is not possible, as the data can be either within the time frame (true) or not (false). So if all data is within the time frame the outcome will be true, while it will be false if not all data is within the time frame.
- Validity will be measured by calculating the percentage of valid entries for each attribute in the data set. A data entry is considered valid when the format and syntax of the data attribute is correct and the same as other entries in the attribute. If this is not the case the data is considered invalid. For every percentage of invalid data 1 point will be deducted from the score that range between 1-10. For example, it the data set contains 3% invalid data, the score for completeness will be 7.
- Accuracy/correctness is measured by checking whether the dates of time stamps in the data have occurred in the past and correctly logged. If the data is from the past and is logged correctly it is considered as correct data. If any of the two is not the case it will be considered as incorrect. To determine a score for correctness the percentage of incorrect data in the data set will be calculated. If the percentage of incorrect data is 5% or more, a score of 2 will be given. If the percentage of incorrect data is 5% or less, a score of 4 will be given. If the percentage of incorrect data is 3% or less, a score of 6 will be given. If the percentage of incorrect data is 1% or less, a score of 8 will be given. Finally if all data is correct a score of 10 will be given.
- Consistency is determined by comparing the length of the values of a data entry with the average length of all entries of the respective attribute. A value is considered inconsistent when the standard deviation diverges more than 2 in length from the average. For the scoring there are a few possible scenarios. If there is inconsistency in length together with a mix of only string, only digit and string/digit values a score of 2 will be given. If there is inconsistency in length together with a mix of two out of three possible composition possibilities a score of 4 will be given. If there is consistency in length together with a mix of only string, only digit and string/digit values it will result in a score of 6. Inconsistency in length together with only one specific composition or consistency in length together with a mix of two out of three possible composition possibilities will both result in a score of 8. Finally, Consistency in length together with only one specific composition will result in a score of 10.
- Believability/credibility cannot be measured based on the data only. The idea proposed in the framework is to note strange values and let these be open for interpretation by the users. Therefore, no scores will be given for this dimension
- Relevancy checks whether there are attributes that occur within specific events only. Data is considered irrelevant if it occurs in less than 5% of the events. Even though this has a specific use for event logs, the check can be performed in a similar way for the data sets of Fokker Services to find irrelevant data. To determine the score the number of irrelevant data attributes will be counted. For each irrelevant attribute 1 point will be deducted from the maximum score of 10. For example, if there are 4 irrelevant attributes in the data set, the score for relevancy will be 6.

- Security/confidentiality will check for all attributes whether personal data has been anonymized. Applying scores for this dimension is not possible as the anonymization is done or not. So this dimension will be true if all data that requires anonymization is anonymized and false is this is not done.
- Complexity is a measurement specifically for process mining as it aims to detect certain structures within the event log of the process. Performing this measurement for the data sets of Fokker Services is not possible as these are structured in a completely different way than event logs. Therefore, it is not applicable in this case and no scores will be given for this dimension.
- Coherence can only be measured by checking for logical interconnections between data attributes. It goes as far as using an analysis based approach to set up decision rules. Therefore, it is not applicable as part of the data quality assessment of this research and no scores will be given.
- Representation/format measures how precisely the time stamps in the data are. The more precise the data is, the higher the score will be. Based on the precision of the time stamps different scores can be given. Empty timestamps will result in a score of 1. Timestamps that only include the years will result in a score of 3. If the month is logged as well in the timestamp a score of 5 will be given. For timestamps that include the days a score of 6 will be given. If the hours are logged in the timestamp the score will be a 7. If it includes minutes as well the score will be a 8. Timestamps that are logged up to the second will result in a score of 9. Finally, if the timestamps include miliseconds the score will be a 10.

F Appendix F - Data Quality Assessment Results

F.1 Data Quality Assessment Work_Order

The results of the quality assessment for the data set 'Work_Order' are given in Table 8. In the assessment, several quality issues were indicated for the data set. The biggest of these is regarding the completeness of the data as there is quite a bit of missing data in the data set. The missing data is not spread out across all data attributes as there are 40 attributes in the data set without missing data. Besides that, another 6 attributes have less than 1% missing data. The last 16 attributes include almost all missing data whereas most of the attributes have more than 70% missing data. 9 of these are attributes that provide data about the utilization of components. For this data, Fokker Services is mainly dependent on the data the users of the components provide to them. If they are not providing their data to Fokker Services it is almost impossible to get this data. Another 3 attributes with a lot missing are dates of a status of an order. In many cases, Fokker Services determines a first and last promise date for the customers' order, which is a range of when the customer can expect their order to be finished. However, this is not a standard part of the process and is not done for all orders, therefore, this data does not exist for every order. Another quality issue is regarding the correctness of the data. For some of the orders, either the start time or the completion time is not correct. It is not possible to track down which of the two is incorrect and why they are incorrect. Therefore, it is assumed that these timestamps are just incorrectly logged. The last quality issue is regarding the relevancy of the data. The data set includes 11 memo attributes. These attributes contain all kinds of information about the component or the order. This memo data is collected at different places and moments throughout the process and is there for an employee to make important notes. The data in these memo attributes include many more quality issues than the ones mentioned before. In earlier research of Schroten, 2020 similar data sets were investigated, and it was concluded that the memo attributes may provide important information. However, to extract this important information text data mining would be needed to extract this information from the data. Since one of the goals is to investigate the possibilities of the current data and how quality issues may affect the results, it was decided to not use the memos in the remaining part of this research. As a result, this data was considered irrelevant and removed from the data set. These attributes were therefore also excluded from the quality assessment of the other quality dimensions.

Quality Dimension	Score	Comments
Completeness	1	In total 16.57% of the entries in the data set are missing. However, the missing data occurs in 22 of 62 attributes. In 3 of these there are only a few entries missing which is less than 0.01%, and another 3 have around 1% missing entries. The other 16 attributes have a lot of missing data, with some of them over 70% missing data.
Uniqueness/duplicates	-	
Timeliness	TRUE	All orders are from last 5 years and contains dates within this specified timeframe.
Validity	10	No invalid data in data set.
Accuracy/correctness	8	There are a few orders with a negative throughput time, which is of course not possible. Either the start or completion data of these orders is not correct. Since it was only the case for 0.01% of the orders it's not a very big issue. Besides that, there is no indication any of the data is not close to the true value.
Consistency	10	Several attributes contain quite lot inconsistent data. However, this is not considered as a quality issue for the data sets. The attributes that have inconsistent data are attributes like partnumbers, serial numbers and part description. These may have names or numbers with a completely different length. Other attributes where consistency does matter are 100% consistent.
Believability/credibility	-	
Relevancy	9	The data set contains several memos. These include a lot of random information collected throughout the process and is considered not to be relevant for the remaining part of this research.
Security/confidentiality	TRUE	All data that requires anonymization had been anonymized.
Complexity	-	
Coherence	-	
Representation/format	9	All timestamps are logged up to the exact second.

Table 8: Results Data Quality Assessment Work_Order

F.2 Data Quality Assessment Work_Order_Labour

The results of the quality assessment for the data set 'Work_Order' are given in Table 9. The assessment of this data set also showed some quality issues. Missing data is an issue in this data set as well, even though it is not as bad as in the data set of Work_Labour. There is only one attribute that includes missing data. In the attribute 'WORK_CENTER about 35% of the data is missing. This data set should include all information about when and where the labour is performed. So it is a quite big issue if it is unknown where the work exactly has been performed. The biggest quality issue in this data set is regarding the consistency of the data. The attributes 'DESCRIPTION' and 'ROUTING_DESCRIPTION' consist of inconsistent data. Even though a lot of different labour can be performed in the repair shop, the number of descriptions should be limited. Otherwise, it will be very difficult to determine which work has been performed in the shop. So it is a big issue if the description of these jobs is very inconsistent. Not only are there large differences in the length and structure of the descriptions, but both English and Dutch descriptions are also used interchangeably. The last issue is about the format of the duration of the work performed. There are no exact time stamps included, only the duration of the performed work in the number of hours. The values included in the data are up to two decimals, so it is still relatively accurate. This issue is not extremely big, as in most cases the number of hours up to two decimals provides enough information already.

Quality Dimension	**Score**	**Comments**
Completeness	7	In total 3.21% of the data is missing, however, only the attribute WORK_CENTER contains missing data. The other attributes have no missing data.
Uniqueness/duplicates	-	
Timeliness	TRUE	No dates are included in the data set. All orders are from last 5 years.
Validity	10	No invalid data in the data set.
Accuracy/correctness	10	No indication that any of the data is not close to the true value.
Consistency	2	In total 14.82% of the data is inconsistent. The inconsistent data occurs in the attributes 'DESCRIPTION' and 'ROUTING_DESCRIPTION'. The first issue is that Dutch and English terms are mixed together in the same attribute. Another issue is the complexity of the descriptions, some have a simple description of only one key word, while others have a long and extremely detailed description. This also leads to a high number of unique values within these attributes.
Believability/credibility	-	
Relevancy	10	All attributes in the data set include potential important information and is therefore considered as relevant.
Security/confidentiality	TRUE	This is only relevant for the employee code, which are anonymized instead of using full names.
Complexity	-	
Coherence	-	
Representation/format	8	No dates are present, but other values as cost are accurate. For the duration only the number of hours are specified, however, they include up to 2 decimals, which makes them a bit more accurate.

Table 9: Results Data Quality Assessment Work_Order_Labour

F.3 Data Quality Assessment Work_Order_Status

The results of the quality assessment for the data set 'Work_Order_Status' are given in Table 10. This data set does not have many quality issues. The biggest issue in this data set is regarding the correctness of the data. There are 486 events in the data set that have a completion date that is earlier than the starting date. This is not possible as that event took a negative amount of time to be completed. This issue is the result of the system allowing employees to log the times of different time zones. There is no standard time zone applied to the timestamps. What could happen for example is that the start date is logged in Europe with the local time zone, while the completion date is logged in the U.S. with the local time zone. If the event then takes a few minutes to complete but the completion date is logged in a timezone 7 hours earlier, the issue that the completion date is logged earlier is created. Another potential issue is regarding the consistency of the attribute 'NAME'. This attribute contains a lot of inconsistent data. However, the number of unique names is relatively low and the names used in the data set are used correctly. Using different names would therefore not be possible. So the inconsistency of the status names is not considered a quality issue for this data set.

Quality Dimension	**Score**	**Comments**
Completeness	10	There is no missing data in this data set.
Uniqueness/duplicates	-	
Timeliness	TRUE	All data in the data set is within specified time.
Validity	10	No invalid data in the data set.
Accuracy/correctness	8	Timestamps are all in the past, however, 486 instances have an earlier completion date than start date, which does not seem correct. It is only 0.1% of the total number of instances in data set. The issue in this case is that the system does not require an universal time zone for logging of the data. So data can be logged for different time zones in different countries and the system will accept this data.
Consistency	10	The attribute 'NAME' consist of 66.37% inconsistent data. However, this is not considered as a quality issue as the number of unique names is relatively low and the status names may differ that much in length.
Believability/credibility	-	
Relevancy	10	All attributes in the data set include potential important information and is therefore considered relevant.
Security/confidentiality	TRUE	There is no data included that needs further anonymization.
Complexity	-	
Coherence	-	
Representation/format	9	All timestamps are logged up to the exact second.

Table 10: Results Data Quality Assessment Work_Order_Status

F.4 Data Quality Assessment Stock

The results of the quality assessment for the data set 'Stock' are given in Table 11. This data set also does not have many quality issues. Missing data is the biggest quality issue in this data set. In total 7 of the 27 attributes have missing data, of which some have around 90% missing data. The main reason for this is that the data in this data set can be very old. Many components currently in stock are there for a very long time, some even over 25 years, and not all information about these components has been collected and stored in the past. Another potential quality issue would be the inconsistency of the attribute 'PARTNUMBER'. However, since components can be stored for a very long time, the difference in part numbers becomes bigger over time. Also did the structure of part numbers slightly change over the years. These two things result in a lot of inconsistent data. However, all part numbers are correct and as they are supposed to be. Changing them is also not possible. Therefore, the inconsistency of the part numbers is not considered a real quality issue of this data set. The last quality issue is regarding the confidentiality of the data. In two of the attributes, the names of real employees are used and stored as they are not anonymized. However, this is more of a general issue that Fokker Services will need to solve in the future, as having anonymized data will not affect the results of this research.

Quality Dimension	**Score**	**Comments**
Completeness	1	In total 16.07% of the data is missing. All of this missing data is included in of the 7 attributes. 3 of these 7 have around 90% missing data. The other 20 have 100% complete data. So it is not the case that the entire data set is bad regarding completeness.
Uniqueness/duplicates	-	
Timeliness	TRUE	The entire data set consists of recent data. Time frame is not relevant in this case.
Validity	10	No invalid data in data set.
Accuracy/correctness	10	No indication any of the data is not close to the true value.
Consistency	10	The attribute 'PARTNUMBER' contains 29.73% inconsistent data. However, this is not considered as a quality issue. The structure of partnumbers was different in the past which caused a difference in structure of the partnumbers. Besides that, all data are indeed partnumbers as they are supposed to be.
Believability/credibility	-	
Relevancy	10	All attributes in the data set include potential important information and is therefore considered relevant.
Security/confidentiality	FALSE	2 attributes with responsibles still include employee names and are not anonymized.
Complexity	-	
Coherence	-	
Representation/format	9	All timestamps are logged up to the exact second.

Table 11: Results Data Quality Assessment Stock

G Appendix G - Decision Tree Outputs

G.1 Output Total Data Set

```
def findDecision(obj): #obj[0]: PART_OWNER, obj[1]: HDR_DEPART, obj[2]: INV_DOC_TOTAL, obj[3
                                              ]: QTY_AVAILABLE, obj[4]: ORDERS_IN_PROGRESS,
                                              obj[5]: WORK_REMAINING
  # {"feature": "HDR_DEPART", "instances": 63418, "metric_value": 1.0, "depth": 1}
  if obj[1]>2478.6950550316947:
    # {"feature": "QTY_AVAILABLE", "instances": 39013, "metric_value": 0.9159, "depth": 2}
    if obj[3]<=11.61543501127593:
      # {"feature": "WORK_REMAINING", "instances": 34637, "metric_value": 0.9492, "depth": 3
                                              }
      if obj[5]<=19896.52884311756:
        # {"feature": "ORDERS_IN_PROGRESS", "instances": 34567, "metric_value": 0.9498, "
                                              depth": 4}
        if obj[4]>772:
          # {"feature": "PART_OWNER", "instances": 34566, "metric_value": 0.9498, "depth": 5
                                              }
          if obj[0]<=0:
            # {"feature": "INV_DOC_TOTAL", "instances": 27586, "metric_value": 0.9239, "
                                              depth": 6}
            if obj[2]>0.0:
              return 'No'
            elif obj[2]<=0.0:
              return 'No'
            else: return 'No'
          elif obj[0]>0:
            # {"feature": "INV_DOC_TOTAL", "instances": 6980, "metric_value": 0.9995, "depth
                                              ": 6}
            if obj[2]<=0.0:
              return 'Yes'
            elif obj[2]>0.0:
              return 'No'
            else: return 'No'
          else: return 'No'
        elif obj[4]<=772:
          return 'No'
        else: return 'No'
      elif obj[5]>19896.52884311756:
        return 'No'
      else: return 'No'
    elif obj[3]>11.61543501127593:
      # {"feature": "INV_DOC_TOTAL", "instances": 4376, "metric_value": 0.2263, "depth": 3}
      if obj[2]>0.0:
        # {"feature": "ORDERS_IN_PROGRESS", "instances": 3408, "metric_value": 0.1461, "
                                              depth": 4}
        if obj[4]>977.360309782433:
          # {"feature": "WORK_REMAINING", "instances": 2751, "metric_value": 0.1729, "depth
                                              ": 5}
          if obj[5]>14837.013700543303:
            # {"feature": "PART_OWNER", "instances": 2297, "metric_value": 0.095, "depth": 6
                                              }
            if obj[0]<=0:
              return 'No'
            elif obj[0]>0:
              return 'No'
            else: return 'No'
          elif obj[5]<=14837.013700543303:
            # {"feature": "PART_OWNER", "instances": 454, "metric_value": 0.452, "depth": 6}
            if obj[0]<=0:
              return 'No'
            elif obj[0]>0:
              return 'No'
            else: return 'No'
          else: return 'No'
        elif obj[4]<=977.360309782433:
          return 'No'
```

```
        else: return 'No'
      elif obj[2]<=0.0:
        # {"feature": "WORK_REMAINING", "instances": 968, "metric_value": 0.4429, "depth": 4
                                            }
        if obj[5]>15921.26918392388:
          # {"feature": "ORDERS_IN_PROGRESS", "instances": 531, "metric_value": 0.6522, "
                                            depth": 5}
          if obj[4]>1261.153052708061:
            # {"feature": "PART_OWNER", "instances": 453, "metric_value": 0.7148, "depth": 6
                                            }
            if obj[0]>0:
              return 'No'
            elif obj[0]<=0:
              return 'No'
            else: return 'No'
          elif obj[4]<=1261.153052708061:
            return 'No'
          else: return 'No'
        elif obj[5]<=15921.26918392388:
          return 'No'
        else: return 'No'
      else: return 'No'
    else: return 'No'
  elif obj[1]<=2478.6950550316947:
    # {"feature": "WORK_REMAINING", "instances": 24405, "metric_value": 0.7775, "depth": 2}
    if obj[5]<=19863.024626495902:
      # {"feature": "QTY_AVAILABLE", "instances": 24374, "metric_value": 0.7758, "depth": 3}
      if obj[3]<=26.133262218608472:
        # {"feature": "ORDERS_IN_PROGRESS", "instances": 24363, "metric_value": 0.7752, "
                                            depth": 4}
        if obj[4]>772:
          # {"feature": "INV_DOC_TOTAL", "instances": 24362, "metric_value": 0.7751, "depth
                                            ": 5}
          if obj[2]<=7450.3278367449175:
            # {"feature": "PART_OWNER", "instances": 22362, "metric_value": 0.7429, "depth":
                                             6}
            if obj[0]<=0:
              return 'Yes'
            elif obj[0]>0:
              return 'Yes'
            else: return 'Yes'
          elif obj[2]>7450.3278367449175:
            # {"feature": "PART_OWNER", "instances": 2000, "metric_value": 0.9837, "depth":
                                            6}
            if obj[0]<=0:
              return 'Yes'
            elif obj[0]>0:
              return 'No'
            else: return 'No'
          else: return 'Yes'
        elif obj[4]<=772:
          return 'No'
        else: return 'No'
      elif obj[3]>26.133262218608472:
        return 'No'
      else: return 'No'
    elif obj[5]>19863.024626495902:
      return 'No'
    else: return 'No'
  else: return 'Yes'
```

G.2 Output Department 2400

```
    def findDecision(obj): #obj[0]: PART_OWNER, obj[1]: HDR_DEPART, obj[2]: INV_DOC_TOTAL,
                                              obj[3]: QTY_AVAILABLE, obj[4]:
                                              ORDERS_IN_PROGRESS, obj[5]: WORK_REMAINING
  # {"feature": "QTY_AVAILABLE", "instances": 11212, "metric_value": 1.0, "depth": 1}
  if obj[3]<=16.54332026652556 6:
```

```
    # {"feature": "ORDERS_IN_PROGRESS", "instances": 8497, "metric_value": 0.947, "depth": 2}
    if obj[4]>862.1817817492554:
      # {"feature": "WORK_REMAINING", "instances": 8238, "metric_value": 0.9548, "depth": 3}
      if obj[5]<=19843.164004593218:
        # {"feature": "INV_DOC_TOTAL", "instances": 8207, "metric_value": 0.9558, "depth": 4}
        if obj[2]<=8588.623202249808:
          # {"feature": "PART_OWNER", "instances": 7324, "metric_value": 0.9644, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 5055, "metric_value": 0.9719, "depth": 6}
            if obj[1]<=2400:
              return 'No'
            else: return 'No'
          elif obj[0]>0:
            # {"feature": "HDR_DEPART", "instances": 2269, "metric_value": 0.9446, "depth": 6}
            if obj[1]<=2400:
              return 'No'
            else: return 'No'
          else: return 'No'
        elif obj[2]>8588.623202249808:
          # {"feature": "PART_OWNER", "instances": 883, "metric_value": 0.8456, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 794, "metric_value": 0.8856, "depth": 6}
            if obj[1]<=2400:
              return 'No'
            else: return 'No'
          elif obj[0]>0:
            return 'No'
          else: return 'No'
        else: return 'No'
      elif obj[5]>19843.164004593218:
        return 'No'
      else: return 'No'
    elif obj[4]<=862.1817817492554:
      # {"feature": "INV_DOC_TOTAL", "instances": 259, "metric_value": 0.2535, "depth": 3}
      if obj[2]<=17402.784047216563:
        # {"feature": "PART_OWNER", "instances": 252, "metric_value": 0.1623, "depth": 4}
        if obj[0]<=0:
          # {"feature": "WORK_REMAINING", "instances": 192, "metric_value": 0.2006, "depth": 5}
          if obj[5]<=11933.288887459019:
            # {"feature": "HDR_DEPART", "instances": 184, "metric_value": 0.2073, "depth": 6}
            if obj[1]<=2400:
              return 'No'
            else: return 'No'
          elif obj[5]>11933.288887459019:
            return 'No'
          else: return 'No'
        elif obj[0]>0:
          return 'No'
        else: return 'No'
      elif obj[2]>17402.784047216563:
        # {"feature": "WORK_REMAINING", "instances": 7, "metric_value": 0.8631, "depth": 4}
        if obj[5]>10827.49:
          return 'Yes'
        elif obj[5]<=10827.49:
          return 'No'
        else: return 'No'
      else: return 'Yes'
    else: return 'No'
  elif obj[3]>16.54332026652556:
    # {"feature": "INV_DOC_TOTAL", "instances": 2715, "metric_value": 0.3967, "depth": 2}
    if obj[2]<=6410.163846891858:
```

```
      # {"feature": "ORDERS_IN_PROGRESS", "instances": 2693, "metric_value": 0.3694, "depth": 3}
      if obj[4]>792:
        # {"feature": "WORK_REMAINING", "instances": 2692, "metric_value": 0.3681, "depth": 4}
        if obj[5]>13208.29146805262:
          # {"feature": "PART_OWNER", "instances": 2643, "metric_value": 0.3473, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 1753, "metric_value": 0.3515, "depth": 6}
            if obj[1]<=2400:
              return 'Yes'
            else: return 'Yes'
          elif obj[0]>0:
            # {"feature": "HDR_DEPART", "instances": 890, "metric_value": 0.3389, "depth": 6}
            if obj[1]<=2400:
              return 'Yes'
            else: return 'Yes'
          else: return 'Yes'
        elif obj[5]<=13208.29146805262:
          # {"feature": "PART_OWNER", "instances": 49, "metric_value": 0.9486, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 44, "metric_value": 0.8757, "depth": 6}
            if obj[1]<=2400:
              return 'Yes'
            else: return 'Yes'
          elif obj[0]>0:
            return 'No'
          else: return 'No'
        else: return 'Yes'
      elif obj[4]<=792:
        return 'No'
      else: return 'No'
    elif obj[2]>6410.163846891858:
      return 'No'
    else: return 'No'
  else: return 'Yes'
```

G.3 Output Department 2500

```
def findDecision(obj): #obj[0]: PART_OWNER, obj[1]: HDR_DEPART, obj[2]: INV_DOC_TOTAL, obj[3]: QTY_AVAILABLE, obj[4]: ORDERS_IN_PROGRESS, obj[5]: WORK_REMAINING
  # {"feature": "QTY_AVAILABLE", "instances": 27398, "metric_value": 1.0, "depth": 1}
  if obj[3]<=14.041136985933278:
    # {"feature": "ORDERS_IN_PROGRESS", "instances": 23998, "metric_value": 0.9891, "depth": 2}
    if obj[4]<=1454.1248893385957:
      # {"feature": "WORK_REMAINING", "instances": 23849, "metric_value": 0.9878, "depth": 3}
      if obj[5]<=19245.56659334468:
        # {"feature": "INV_DOC_TOTAL", "instances": 23720, "metric_value": 0.9866, "depth": 4}
        if obj[2]<=9804.427019581562:
          # {"feature": "PART_OWNER", "instances": 23417, "metric_value": 0.9847, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 19328, "metric_value": 0.9647, "depth": 6}
            if obj[1]<=2500:
              return 'Yes'
            else: return 'Yes'
          elif obj[0]>0:
            # {"feature": "HDR_DEPART", "instances": 4089, "metric_value": 0.9684, "depth": 6}
            if obj[1]<=2500:
              return 'No'
```

```
            else: return 'No'
          else: return 'No'
        elif obj[2]>9804.427019581562:
          # {"feature": "PART_OWNER", "instances": 303, "metric_value": 0.7375, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 277, "metric_value": 0.7735, "depth": 6
                                              }
            if obj[1]<=2500:
              return 'No'
            else: return 'No'
          elif obj[0]>0:
            return 'No'
          else: return 'No'
        else: return 'No'
      elif obj[5]>19245.56659334468:
        return 'No'
      else: return 'No'
    elif obj[4]>1454.1248893385957:
      return 'No'
    else: return 'No'
  elif obj[3]>14.041136985933278:
    # {"feature": "ORDERS_IN_PROGRESS", "instances": 3400, "metric_value": 0.3515, "depth":
                                              2}
    if obj[4]>1178.904117647059:
      # {"feature": "INV_DOC_TOTAL", "instances": 1805, "metric_value": 0.5426, "depth": 3}
      if obj[2]>0.0:
        # {"feature": "WORK_REMAINING", "instances": 1296, "metric_value": 0.3095, "depth":
                                              4}
        if obj[5]>17313.740687595335:
          # {"feature": "PART_OWNER", "instances": 707, "metric_value": 0.4748, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 675, "metric_value": 0.4898, "depth": 6
                                              }
            if obj[1]<=2500:
              return 'No'
            else: return 'No'
          elif obj[0]>0:
            return 'No'
          else: return 'No'
        elif obj[5]<=17313.740687595335:
          return 'No'
        else: return 'No'
      elif obj[2]<=0.0:
        # {"feature": "WORK_REMAINING", "instances": 509, "metric_value": 0.882, "depth": 4}
        if obj[5]>16462.81374344425:
          # {"feature": "PART_OWNER", "instances": 422, "metric_value": 0.9448, "depth": 5}
          if obj[0]>0:
            # {"feature": "HDR_DEPART", "instances": 389, "metric_value": 0.9669, "depth": 6
                                              }
            if obj[1]<=2500:
              return 'No'
            else: return 'No'
          elif obj[0]<=0:
            return 'No'
          else: return 'No'
        elif obj[5]<=16462.81374344425:
          return 'No'
        else: return 'No'
      else: return 'No'
    elif obj[4]<=1178.904117647059:
      return 'No'
    else: return 'No'
  else: return 'No'
```

G.4 Output Department 2520

```
def findDecision(obj): #obj[0]: PART_OWNER, obj[1]: HDR_DEPART, obj[2]: INV_DOC_TOTAL, obj[3
                                              ]: QTY_AVAILABLE, obj[4]: ORDERS_IN_PROGRESS,
```

```
                                                obj[5]: WORK_REMAINING
# {"feature": "QTY_AVAILABLE", "instances": 11434, "metric_value": 1.0, "depth": 1}
if obj[3]<=4.730092565191176:
  # {"feature": "PART_OWNER", "instances": 10020, "metric_value": 0.9856, "depth": 2}
  if obj[0]<=0:
    # {"feature": "INV_DOC_TOTAL", "instances": 6688, "metric_value": 0.9668, "depth": 3}
    if obj[2]<=6603.537852392868:
      # {"feature": "ORDERS_IN_PROGRESS", "instances": 4522, "metric_value": 0.8265, "
                                                depth": 4}
      if obj[4]<=1338.8645925947112:
        # {"feature": "WORK_REMAINING", "instances": 3496, "metric_value": 0.906, "depth":
                                                 5}
        if obj[5]>11416.212028779177:
          # {"feature": "HDR_DEPART", "instances": 3275, "metric_value": 0.9244, "depth":
                                                6}
          if obj[1]<=2520:
            return 'No'
          else: return 'No'
        elif obj[5]<=11416.212028779177:
          # {"feature": "HDR_DEPART", "instances": 221, "metric_value": 0.3044, "depth": 6
                                                }
          if obj[1]<=2520:
            return 'No'
          else: return 'No'
        else: return 'No'
      elif obj[4]>1338.8645925947112:
        # {"feature": "WORK_REMAINING", "instances": 1026, "metric_value": 0.2852, "depth
                                                ": 5}
        if obj[5]<=18807.048168621943:
          # {"feature": "HDR_DEPART", "instances": 880, "metric_value": 0.3193, "depth": 6
                                                }
          if obj[1]<=2520:
            return 'No'
          else: return 'No'
        elif obj[5]>18807.048168621943:
          return 'No'
        else: return 'No'
      else: return 'No'
    elif obj[2]>6603.537852392868:
      # {"feature": "ORDERS_IN_PROGRESS", "instances": 2166, "metric_value": 0.9136, "
                                                depth": 4}
      if obj[4]>1066.7444272172156:
        # {"feature": "WORK_REMAINING", "instances": 1977, "metric_value": 0.8335, "depth
                                                ": 5}
        if obj[5]<=18168.675202404214:
          # {"feature": "HDR_DEPART", "instances": 1914, "metric_value": 0.7956, "depth":
                                                6}
          if obj[1]<=2520:
            return 'Yes'
          else: return 'Yes'
        elif obj[5]>18168.675202404214:
          return 'No'
        else: return 'No'
      elif obj[4]<=1066.7444272172156:
        return 'No'
      else: return 'No'
    else: return 'Yes'
  elif obj[0]>0:
    # {"feature": "INV_DOC_TOTAL", "instances": 3332, "metric_value": 0.3779, "depth": 3}
    if obj[2]<=0.0:
      # {"feature": "ORDERS_IN_PROGRESS", "instances": 3163, "metric_value": 0.1618, "
                                                depth": 4}
      if obj[4]>789:
        # {"feature": "WORK_REMAINING", "instances": 3162, "metric_value": 0.1601, "depth
                                                ": 5}
        if obj[5]>10324.77:
          # {"feature": "HDR_DEPART", "instances": 3161, "metric_value": 0.1585, "depth":
                                                6}
```

```
            if obj[1]<=2520:
              return 'Yes'
            else: return 'Yes'
          elif obj[5]<=10324.77:
            return 'No'
          else: return 'No'
        elif obj[4]<=789:
          return 'No'
        else: return 'No'
      elif obj[2]>0.0:
        return 'No'
      else: return 'No'
    else: return 'Yes'
  elif obj[3]>4.730092565191176:
    return 'No'
  else: return 'No'
```

G.5 Output Department 2540

```
def findDecision(obj): #obj[0]: PART_OWNER, obj[1]: HDR_DEPART, obj[2]: INV_DOC_TOTAL, obj[3
                                              ]: QTY_AVAILABLE, obj[4]: ORDERS_IN_PROGRESS,
                                              obj[5]: WORK_REMAINING
  # {"feature": "WORK_REMAINING", "instances": 5198, "metric_value": 1.0, "depth": 1}
  if obj[5]>11456.403993484131:
    # {"feature": "QTY_AVAILABLE", "instances": 4962, "metric_value": 0.9984, "depth": 2}
    if obj[3]<=5.883057487252508:
      # {"feature": "ORDERS_IN_PROGRESS", "instances": 4837, "metric_value": 0.996, "depth":
                                               3}
      if obj[4]>880.7982689605151:
        # {"feature": "INV_DOC_TOTAL", "instances": 4770, "metric_value": 0.9942, "depth": 4
                                              }
        if obj[2]>0.0:
          # {"feature": "PART_OWNER", "instances": 4334, "metric_value": 0.977, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 4281, "metric_value": 0.973, "depth": 6
                                              }
            if obj[1]<=2540:
              return 'Yes'
            else: return 'Yes'
          elif obj[0]>0:
            return 'No'
          else: return 'No'
        elif obj[2]<=0.0:
          # {"feature": "PART_OWNER", "instances": 436, "metric_value": 0.4862, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 428, "metric_value": 0.4851, "depth": 6
                                              }
            if obj[1]<=2540:
              return 'No'
            else: return 'No'
          elif obj[0]>0:
            # {"feature": "HDR_DEPART", "instances": 8, "metric_value": 0.5436, "depth": 6}
            if obj[1]<=2540:
              return 'No'
            else: return 'No'
          else: return 'No'
        else: return 'No'
      elif obj[4]<=880.7982689605151:
        return 'No'
      else: return 'No'
    elif obj[3]>5.883057487252508:
      return 'No'
    else: return 'No'
  elif obj[5]<=11456.403993484131:
    return 'No'
  else: return 'No'
```

G.6 Output Department 2550

```
def findDecision(obj): #obj[0]: PART_OWNER, obj[1]: HDR_DEPART, obj[2]: INV_DOC_TOTAL, obj[3]: QTY_AVAILABLE, obj[4]: ORDERS_IN_PROGRESS, obj[5]: WORK_REMAINING
  # {"feature": "QTY_AVAILABLE", "instances": 3688, "metric_value": 1.0, "depth": 1}
  if obj[3]<=3.5830525440414167:
    # {"feature": "ORDERS_IN_PROGRESS", "instances": 3168, "metric_value": 0.9805, "depth": 2}
    if obj[4]>1084.2850756440873:
      # {"feature": "INV_DOC_TOTAL", "instances": 2771, "metric_value": 0.9195, "depth": 3}
      if obj[2]<=35963.06871692356:
        # {"feature": "WORK_REMAINING", "instances": 2654, "metric_value": 0.8876, "depth": 4}
        if obj[5]<=18673.630675738506:
          # {"feature": "PART_OWNER", "instances": 2578, "metric_value": 0.8618, "depth": 5}
          if obj[0]<=0:
            # {"feature": "HDR_DEPART", "instances": 2468, "metric_value": 0.8504, "depth": 6}
            if obj[1]<=2550:
              return 'Yes'
            else: return 'Yes'
          elif obj[0]>0:
            # {"feature": "HDR_DEPART", "instances": 110, "metric_value": 0.9979, "depth": 6}
            if obj[1]<=2550:
              return 'Yes'
            else: return 'Yes'
          else: return 'Yes'
        elif obj[5]>18673.630675738506:
          return 'No'
        else: return 'No'
      elif obj[2]>35963.06871692356:
        return 'No'
      else: return 'No'
    elif obj[4]<=1084.2850756440873:
      return 'No'
    else: return 'No'
  elif obj[3]>3.5830525440414167:
    return 'No'
  else: return 'No'
```

H Appendix H - Visualization Decision Trees Departments

H.1 Decision Tree Department 2400

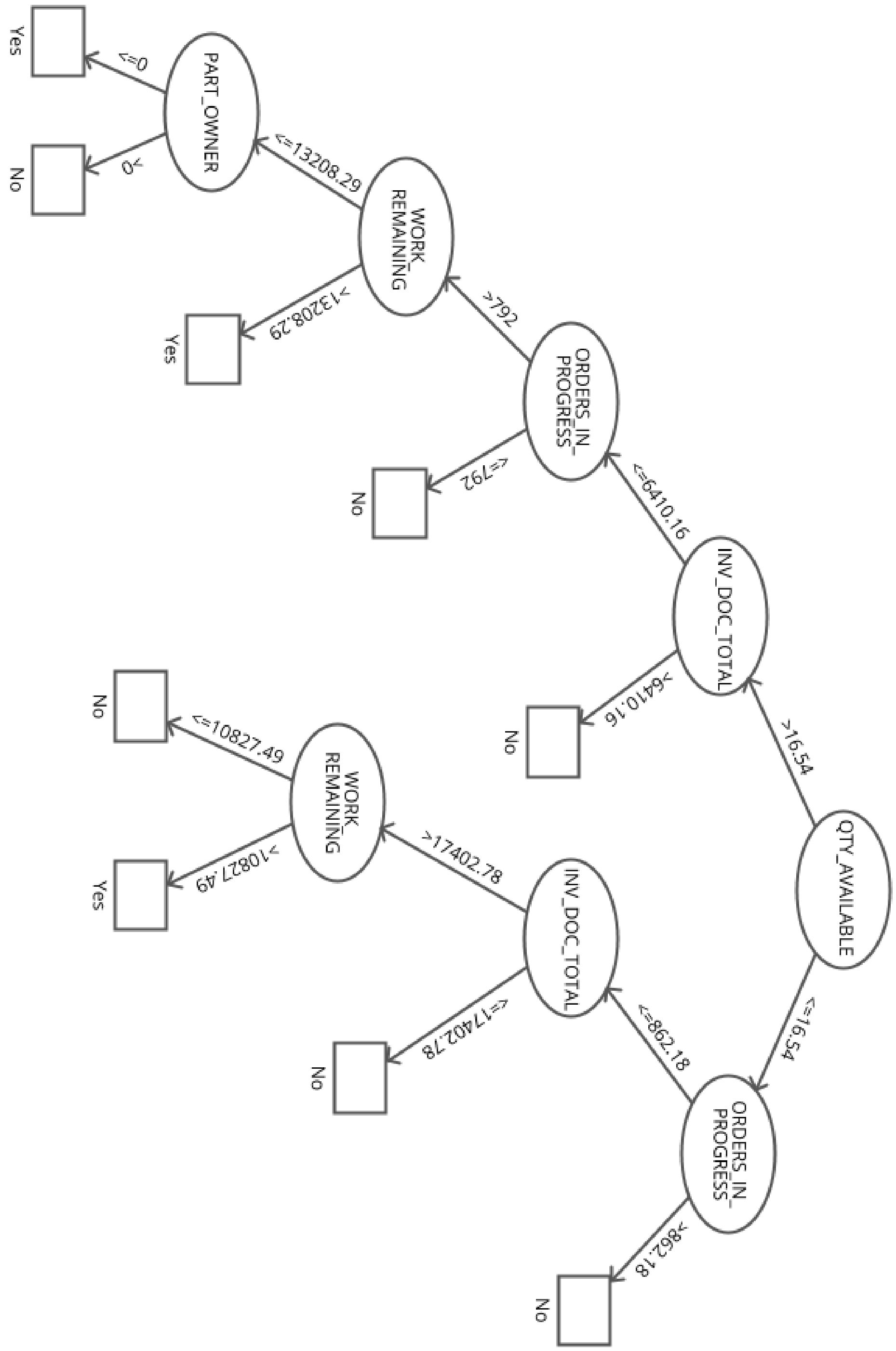

Figure 44: Decision Tree Department 2400

H.2 Decision Tree Department 2500

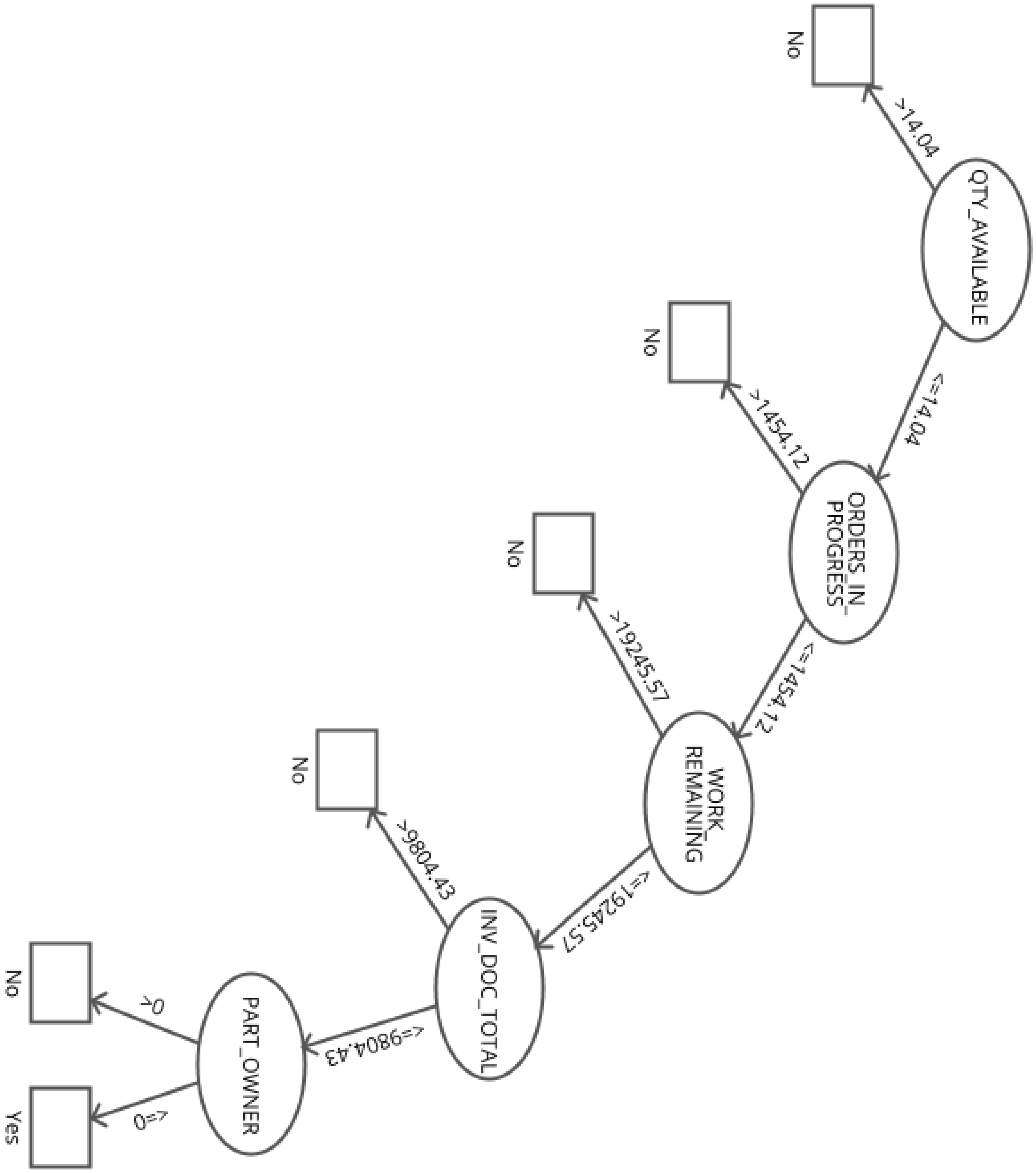

Figure 45: Decision Tree Department 2500

H.3 Decision Tree Department 2520

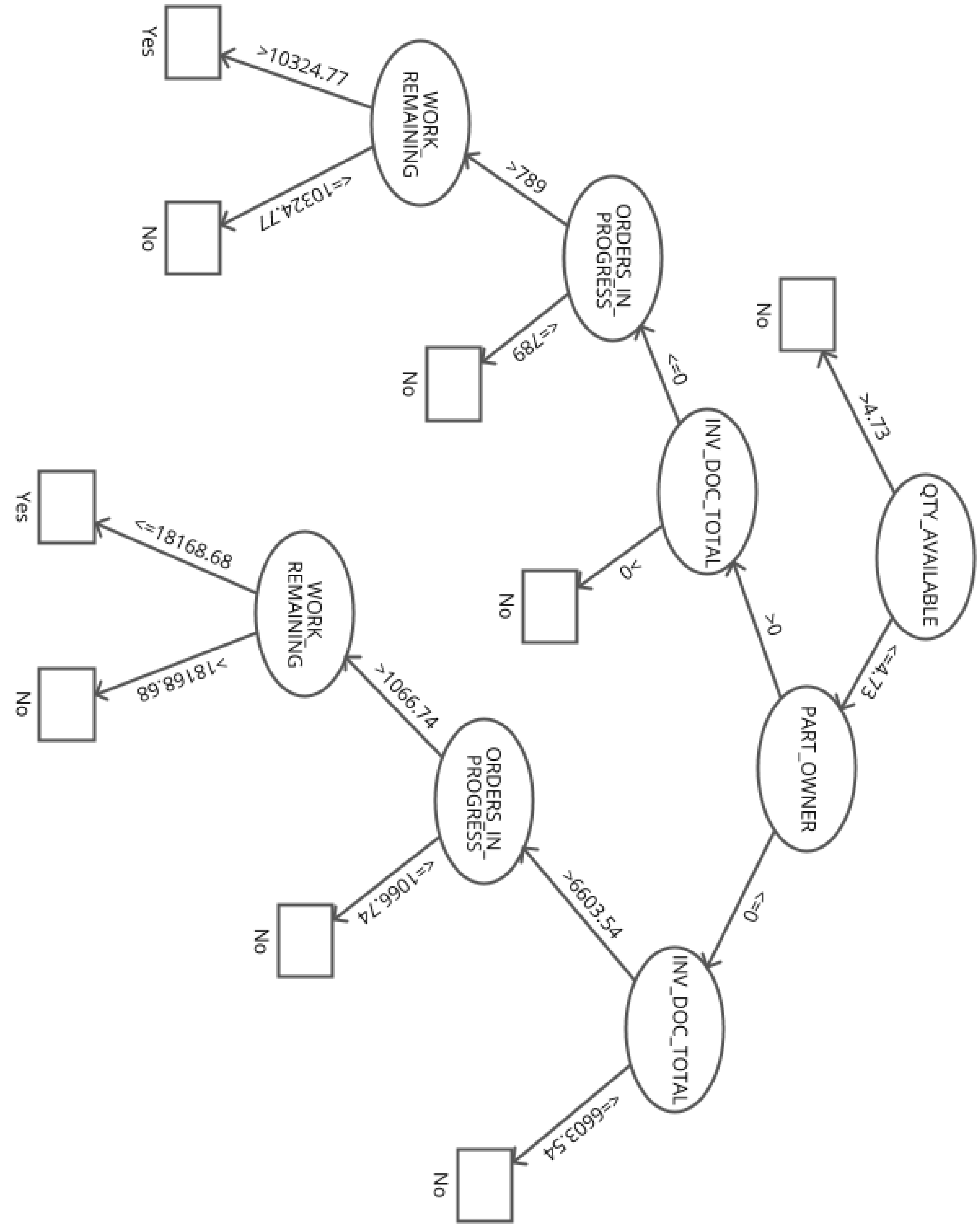

Figure 46: Decision Tree Department 2520

H.4 Decision Tree Department 2540

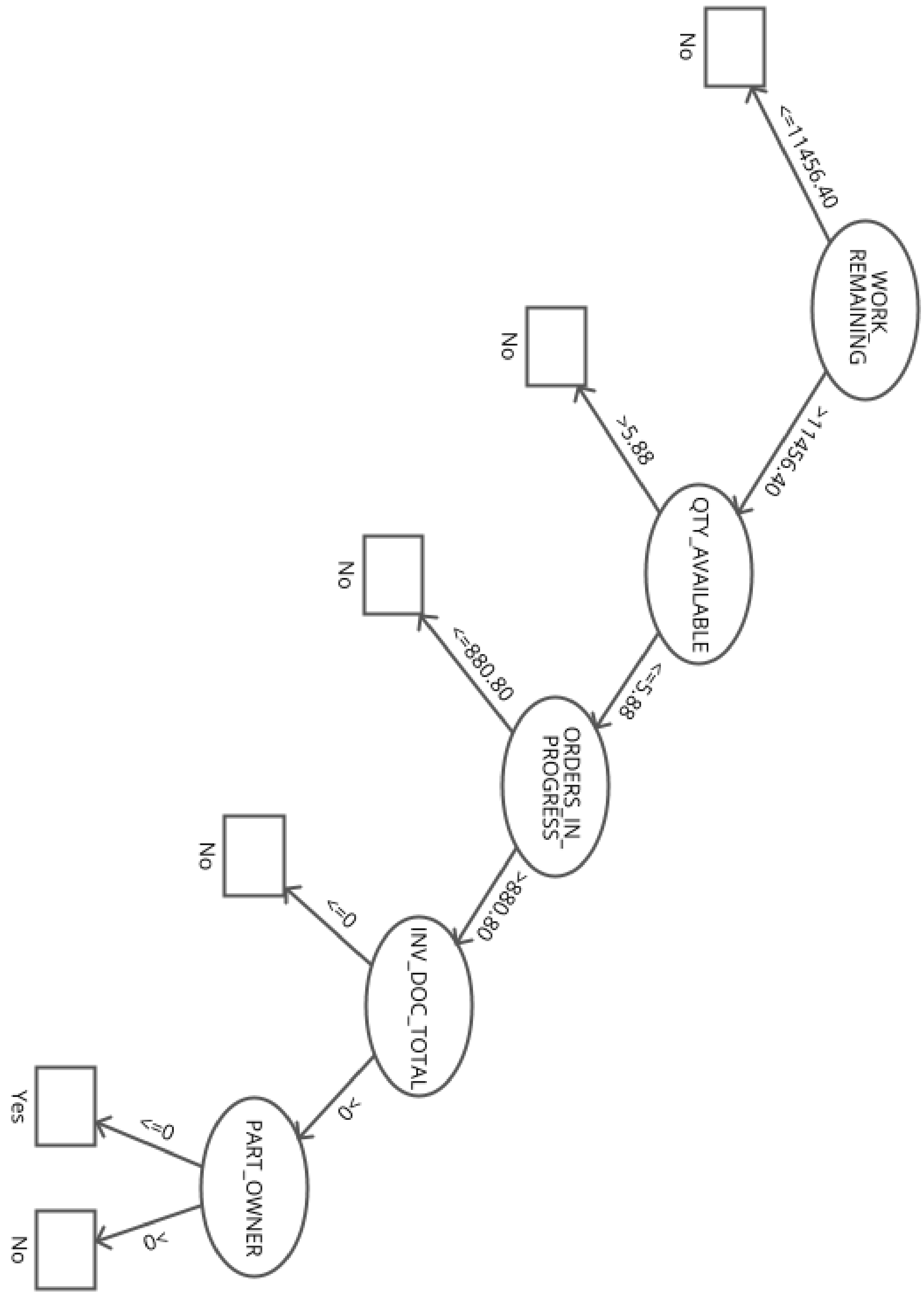

Figure 47: Decision Tree Department 2540

H.5 Decision Tree Department 2550

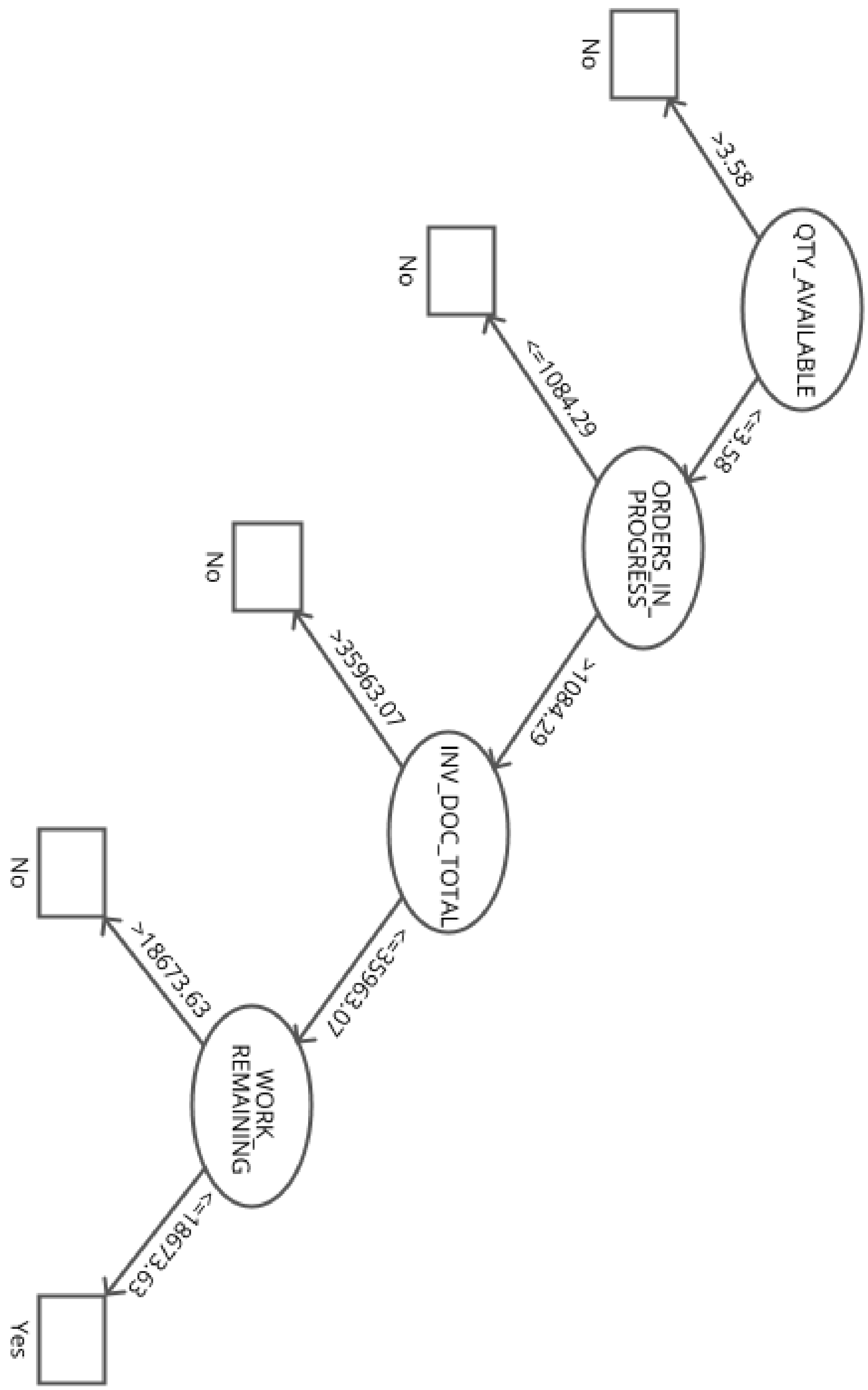

Figure 48: Decision Tree Department 2550