MASTER

A product-oriented design of an IS architecture, underlying a standard outsourcing process case study within the large-scale, high-tech, manufacturing industry

van Zutphen, A.J.A.N.

Award date:
2018

Eindhoven, March 2018

A Product-Oriented Design of an IS Architecture, underlying a Standard Outsourcing Process

case study within the large-scale, high-tech, manufacturing industry

By A.J.A.N. (Arjan) van Zutphen

Student identity number 0977847

In partial fulfillment of the requirements for the degree of

Master of Science

in Operations Management and Logistics

Supervisors:
prof. dr. ir. P.W.P.J. (Paul) Grefen, TU/e, IS
dr. ir. I.T.P. (Irene) Vanderfeesten, TU/e, IS
dr. ir. R.M. (Remco) Dijkman, TU/e, IS

TU/e, School of Industrial Engineering

Series Master Theses Operations Management and Logistics

Key words: outsourcing, system architecture, information systems, high-tech manufacturing industry, three-dimensional cube.

Management Summary

Due to a shift in manufacturing from mass production to mass customization, an increase in customized products exist. Besides, the economic growth requires production growth to meet the increase in customer demands. Together, these factors result in the need for manufacturing companies to invest in their core-business and assets or to outsource part of their production processes to external companies. Since investments in assets are very expensive, and inflexible, manufacturing companies choose for outsourcing to be able to meet their customer demands and to maintain their flexibility. However, due to an increase in production and an increase in customization outsourcing becomes more complex.

A firm that outsources part of its operations for these reasons is TC, an international high-tech manufacturing company with both manufacturing plants and assembly plants. Besides, it has multiple supporting divisions, including an IT department. The research described in this report was conducted at the logistics department of one of the manufacturing plants of TC in combination with the IT department. Within this manufacturing plant outsourcing is used when production requirements exceed the capacity of its machines. Also TC is facing the increasing complexity of outsourcing. In order to manage the outsourcing process, there is no standard outsourcing process defined. In order to support outsourcing multiple self-built applications were built. This so called shadow IT is not monitored and controlled by the IT-department. In order to sufficiently support the outsourcing process, TC requires information systems (ISs), appropriately structured into an IS architecture. Additionally, based on a literature review, the need for such an architecture in high-tech manufacturing exists.

Therefore, this study used several models, which were positioned and structured by using the three-dimensional cube of Grefen (2016). In order to structure the research the problem solving cycle of Van Aken, Berends, & Van der Bij (2012) was used. This resulted in four phases, namely: (1) AS-IS, (2) Process Analysis, (3) TO-BE, and (4) Reference Architecture.

First, the current situation (AS-IS) was explored and modeled in a business process diagram and architecture model. In this way potential threats could be defined, arising when outsourcing becomes more complex and occurs more frequently. The most important threats were the increase in occurrence and frequency of manual interfaces, the error-proneness, and the need for safety buffers due to a loss of insights. A TO-BE model had to be developed in order to solve these potential threats and thus regain insights in the outsourcing process.

Therefore, this study designed an IS architecture, including ISs that support outsourcing processes for the manufacturing plant of TC. First, a standard outsourcing process is based on the outsourcing circle of Perunovic (2006). In this cycle five phases are identified, by taking into account several studies on outsourcing: (1) preparation, (2) vendor(s) selection, (3) transition, (4) managing relationship, and (5) reconsideration. In each of the phases certain decisions have to be made regarding the outsourcing: whether to outsource, what to outsource, when to outsource, to whom to outsource, how to manage the outsourcing, etc. For each decision during the outsourcing process information is required.

Second, in the Process Analysis it is defined what information is required during the outsourcing process. This need for information differs between manufacturing environments, which are distinguished by the positions of their customer order decoupling points. In the literature four manufacturing environments exist: (1) Make-to-Stock, (2) Make-to-Assembly, (3) Make-to-Order, and (4) Engineer-to-Order. For each of the manufacturing environments a company in high-tech manufacturing was approached in order to retrieve what information it requires for each decision in

the outsourcing process. Additionally, best practices of Oracle (2014) were considered. For each kind of information it was then determined whether it is applicable to TC or not.

Third, when applicable to TC, it was found which ISs existing within TC provide this kind of information or what ISs should be expanded. This resulted in a TO-BE model, consisting of the standard outsourcing process and the ISs required to support each phase in the standard outsourcing process. This was captured into a product-oriented architecture design, by using ArchiMate. The design was evaluated on both relevance and rigor. Based on this evaluation several conclusions were drawn and recommendations were given. First, TC regained insights in its outsourcing process. In this way outsourcing activities become interchangeable between planners of the logistics department, and it is known what information is required for what decision in the outsourcing process. Second, the TO-BE model replaces the shadow IT, including its manual interfaces. This was one of the most significant potential threats for TC, when outsourcing occurs more frequently and becomes more complex. Especially, due to its scalability the TO-BE model is able to manage the increase in outsourcing as it eliminates the manual interfaces, decreases the error-proneness, and reduces the need for safety buffers.

Finally, the proposed architecture design was generalized to a reference architecture, for high-tech manufacturing companies in general. In other words, the TC architecture design was used as a case study. First, the TO-BE model was expanded by adding extra ISs since extra information is required for the reference architecture. Second, based on the evaluation of the TO-BE model, that is limited due to its legacy restrictions, the model was improved. In this way the reference architecture was an improved and extended version of the TO-BE model. First, it contains a service bus, allowing for flexible direct synchronous coupling. This is an important factor in outsourcing as manufacturing planning is constantly subject to change. Second, the reference architecture contains a business process management system, resulting in a workload reduction and the ability to constantly monitor the outsourcing process.

Preface

This report is part of the master thesis project in completion of the Master Operations Management & Logistics (OML) at Eindhoven, University of Technology (TU/e). Besides, it is the final deliverable of almost seven years of studying, started by obtaining a BSc degree in Business Administration at Tilburg University. The master thesis project was conducted during an internship at a high-tech manufacturing company, from September 2017 till March 2018. In this preface I would like to express my gratitude to multiple persons for their help and support. Together with an exchange period in Stockholm, this project was one of the most challenging periods in my relatively short career, especially in personal development.

First of all I would like to thank my 1st supervisor Paul Grefen, who stimulated me to continuously learn about my work and about myself. The metaphors he used for explaining complex concepts, worked as an inspiration for me, and gave me new insights.

Second, I am very grateful to Annette, my supervisor at the company, who guided me through the entire project. Just graduated and immediately starting as a company's thesis supervisor is definitely not for everyone! Every meeting I found it very insightful to receive her feedback.

Third, I would like to thank my 2nd supervisor Irene Vanderfeesten and the students participating in the HORSE-project. During the presentation sessions with the HORSE-team, I got valuable feedback on how to present and explain the content of my project.

Next, I would like to thank the company's employees Edwin, Martijn, Iris, and Paul for the pleasant work environment and for helping me to appropriately gather information in order to succeed my master thesis project. A special thanks to Jan, not only for his immense knowledge on the company's IT systems, but also for giving me a daily positive boost by his enthusiastic morning greetings.

Finally, I would like to thank my family, friends and fellow students for their support and the nice times we had during my study.

Arjan van Zutphen

March 2018

Table of Contents

List of Figures

List of Tables

1. Introduction

This thesis is part of the Master Operations Management & Logistics at the Eindhoven University of Technology. The project described in this report was executed at a high-tech manufacturing company, referred to as TC in this report due to confidentiality. This section describes consecutively background information of the problem (1.1), the research problem (1.2), and the research method (1.3). The research method consists of the research questions, the research framework, the phases and deliverables, the scope, and finally the structure of the report.

1.1. Problem Description

TC is a high-tech manufacturing company. It has several plants for both manufacturing and assembly processes. First, raw materials are manufactured to components. Second, the components are, together with procured components, assembled to a final product. The project was conducted at the logistics department of one of the manufacturing plants of TC.

TC applies lean management that is defined as 'an approach to management focusing on reducing or eliminating waste in all facets of the system' (APICS, 2011). This is conform with TC's just-in-time objective to strive for no inventory, meaning that a product is delivered exactly at the time the customer demands it (APICS, 2011). In order to be lean, TC applies three basic principles known in logistics: (1) Make-To-Order (MTO), meaning that products are made entirely after a customer has ordered (APICS, 2011), (2) Best Point of Use, meaning that waste of the operator is minimized by serving the operator at the right place on the right moment and, (3) Mixed Model Assembly, meaning that all final products are assembled on the same assembly line.

These principles affect the planning of operations on machines. This is an activity of the logistics department of the manufacturing plant, making use of the Enterprise Resource Planning (ERP) system. At this department multiple material planners are each responsible for the output of a number of machines within the plant. Machines might be dependent on other machines, when a product needs multiple operations on multiple machines. Other machines are dependent on the procurement of raw materials. In the long term the planners determine whether the machines are able to satisfy the forecasted needs. If this is not the case the planners request for outsourcing. There is no standard identified for the planners to manage outsourcing and it is not sufficiently supported by the ERP system. Hence, manual handling is required.

Momme (2001) defined outsourcing as 'the process of establishing and managing a contractual relationship with an external supplier for the provision of capacity that has previously been provided in-house'. More recent studies eliminate the part of this definition that states that capacity was first provided in-house, like Yang & Li (2015), which is the definition that will be used during this study. They state that outsourcing refers to 'that in order to concentrate resources and energy on the core business, and enhance core competitiveness of the enterprise, entrust some or all of the businesses which a company is not good at for operation in way of contract' (Yang & Li, 2015).

Outsourcing at the manufacturing plant of TC happens for three reasons: (1) TC is not capable to execute the operation, (2) TC plans to outsource operations beforehand due to limited capacity, or (3) outsourcing happens last-minute and unplanned.

Since multiple factors are playing a role in outsourcing TC perceives that outsourcing is becoming more complex. These factors are: (1) an increase in production, resulting in an increase of the number of outsourcing partners and the need for faster throughput times, and (2) an increase in customized products, resulting in an increase in variants of semi-finished products. TC expects that complex outsourcing will occur more often, will become even more complex, and consequently in the long term may lead to difficulties. Therefore, TC requires ISs (Information Systems) that support complex outsourcing processes, structured into an appropriate IS architecture.

In order to conduct this project, both the IT department as well the manufacturing plant within TC were involved as stakeholders.

1.2. Research Problem

After providing an introduction about the project, it can be summarized in a problem statement as follows:

"The current ISs are not able to effectively support the increasingly more complex outsourcing processes within a manufacturing plant of TC"

The problem and its context are visualized in Figure 1 in a problem mess. Due to a planned increase in production and an increase in customization outsourcing becomes more complex. Concurrently, there is no standard outsourcing process defined and there is a lack of a supporting IT structure. Conjointly, this leads to a loss of insights within the outsourcing process. Hence, the IT landscape and outsourcing processes are not future proof. This might cause that future demand cannot be met, without creating safety stocks, resulting in a loss of revenue. The focus of this study is highlighted in Figure 1.

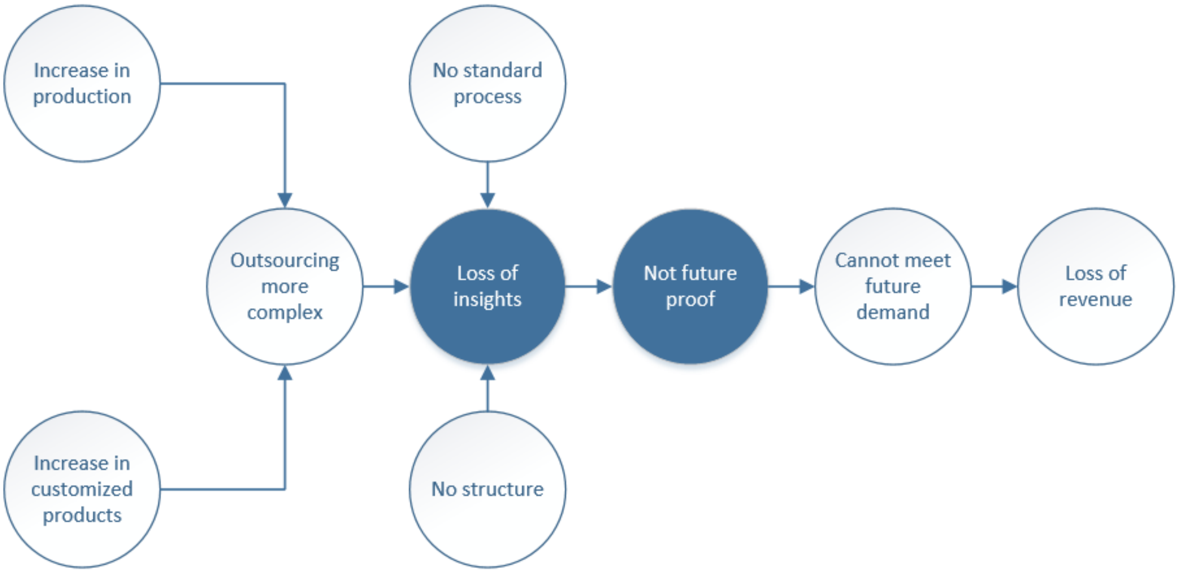


Figure 1 Problem mess

1.3. Research Method

In this section the research method is described. First the main research question and goals were set. Second, the used research framework is given followed by the sub questions. Third, the phases and deliverables were identified. Fourth, the scope of this study was set. Finally the structure of the report is given.

1.3.1. Main Research Question

Based on the problem definition the main research question was defined.

The main research question is as follows:

"How should a future IS architecture design for TC look like in order to effectively support its processes in complex outsourcing?"

Answering the research question resulted in an architecture design. According to Grefen (2016) an architecture design can be interpreted in two ways: the product-oriented face and the process-oriented face. First, the product-oriented face of architectures focuses on architectures as sets of structural blueprints for the realization of ISs. Second, the process-oriented face focuses on procedural prescriptions for the realization of ISs. The solution design of this study, i.e. the proposed architecture design, is product-oriented.

1.3.2. Research Goals

Based on meetings with TC's management, of both the manufacturing plant as well the IT department, the research goals were determined. The goals are in abstract terms briefly summarized for each stakeholder in Figure 2. These objectives are numbered from A to F for later purposes. Based on a literature review a third stakeholder (i.e. research) was defined. In order to support outsourcing processes it was concluded that there is a need for a reference architecture.

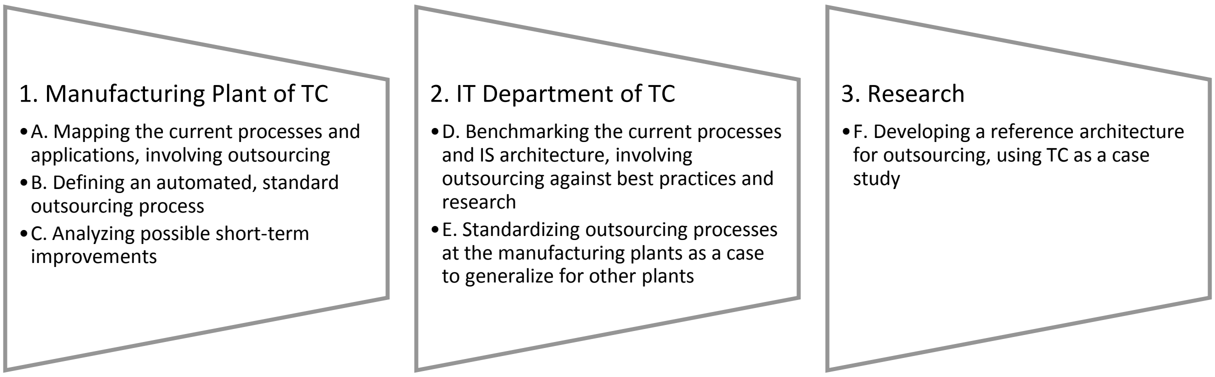


Figure 2 Research goals for each stakeholder

1.3.3. Research Framework

In order to structure the research method two conceptual frameworks are now discussed. First the problem solving cycle of Van Aken et al. (2012) is briefly described, followed by the three-dimensional cube of Grefen (2016).

1.3.3.1. Problem Solving Cycle

This thesis is built on the problem solving cycle by Van Aken et al. (2012). The cycle is shown in Figure 3 and consists of the following five steps: (1) Problem Definition, (2) Analysis & Diagnosis, (3) Solution Design, (4) Implementation of Solution, and (5) Evaluation.

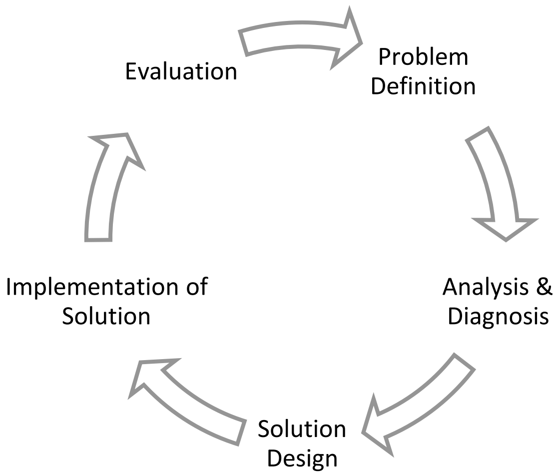


Figure 3 Problem solving cycle (Van Aken et al., 2012)

The 'Problem Definition' step contains intake and orientation (Van Aken et al., 2012). In this step the problem context, problem statement and deliverables are defined. In the second step (Analysis & Diagnosis) the business problem is analyzed in this project consisting of an empirical and theoretical analysis. Third a redesign of the current state is suggested, which is implemented in the 4th step. Finally evaluation of the solution takes place in the 5th step.

1.3.3.2. Grefen's Three-dimensional Cube

As the second element of the research framework the three-dimensional cube of Grefen (2016) was used. In this thesis several models were developed, e.g. architecture models. The cube was used to structure these models. It captures four dimensions: aspect, aggregation, abstraction, and realization. The aspect dimension describes a number of aspects from which we can view an architecture, the aggregation describes an architecture from completely undetailed (black box) to very detailed, the abstraction dimension ranges from very abstract to very concrete, and the realization dimension from very business-oriented to very IT-oriented (Grefen, 2016). For the realization dimension it uses the levels of the BOAT framework (**B**usiness, **O**rganization, **A**rchitecture, and **T**echnology). More elaboration on the cube is given in Chapter 2.

1.3.4. Research Sub Questions

Based on TC's research goals the research sub questions were developed. Each research sub question corresponds to a research goal (discussed in Section 1.3.1), to a level on the realization dimension in Grefen's cube (discussed in Section 1.3.3), and to a phase or multiple phases of the problem solving cycle of Van Aken et al. (discussed in Section 1.3.3). This is visualized in Table 1.

Table 1 Research sub questions related to research goals and framework

	Research Goal(s)	Realization Dimension (Grefen, 2016)	Problem Solving Cycle (Van Aken et al., 2012)
1: What is the current situation at the manufacturing plant of TC?			
1a: What are the current processes for complex outsourcing at the manufacturing plant of TC?	A	O	Diagnosis & Analysis (TC)
1b: How are the current processes in complex outsourcing supported by IT at the manufacturing plant of TC?		A	
2: How are the current processes and supporting IT compared to best practices and theoretical references?			
2a: How are the current processes and supporting IT compared to best practices?	D	O & A	Diagnosis & Analysis (TC + General)
2b. How are the current processes and supporting IT compared to theoretical references?			
3: What is the future situation for the manufacturing plant of TC?			
3a: How should a standard complex outsourcing process for the manufacturing plant of TC look like?	B, C & E	O	Solution Design, Implementation of Solution & Evaluation (TC)
3b: How should a future IS architecture design for TC look like?		A	
4: How should a general IS architecture for outsourcing in high tech manufacturing companies look like?	F	A	Solution Design (General)

Answering these questions resulted in a general architecture design to support outsourcing processes for high-tech manufacturing companies. This was done by designing and generalizing a product-oriented architecture design for the manufacturing plant of TC. In other words, the manufacturing plant of TC is used as a case to develop an architecture design for outsourcing in high-tech manufacturing companies.

1.3.5. Phases & Deliverables

Based on the research sub questions and the problem solving cycle, five research phases were identified in this section. These phases are visualized in Figure 4.

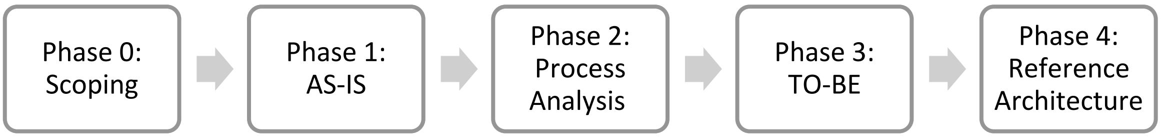


Figure 4 Phases of this study

Each phase corresponds to one or multiple research sub question(s) as visualized in Figure 5. Phase 0 does not correspond to a research question as it is a preliminary phase. The deliverables of each phase are also shown in Figure 5 and are now briefly described.

First, in the AS-IS phase the current processes were modeled in a business process diagram, and the supporting ISs were modeled in an architecture model. In this phase potential threats for TC, arising when outsourcing occurs more frequently and becomes more complex, were identified. Second, in the Process Analysis phase a framework was created that includes for each phase in the outsourcing process what information is required to make a decision. This was based on an analysis of both theory and practice, and formed the input for Phase 3 and Phase 4. Third, in the TO-BE phase a standard outsourcing process was created taking into account results of the second phase. Also an architecture design was developed for TC to support this standard outsourcing process and to solve the potential threats identified in the first phase. Finally, in the fourth phase the architecture design was generalized for high-tech manufacturing companies, taking into account the second phase and evaluation on the TO-BE phase, resulting in a reference architecture.

Phase	Research sub questions	Deliverables
AS-IS	1a, 1b	• Business process diagram of current process • Architecture model of current IT supporting outsourcing process • Potential threats
Process Analysis	2a, 2b	• Analysis framework containing information required for outsourcing, applicable to both TO-BE and Reference Architecture
TO-BE	3a, 3b	• Standard outsourcing process for TC • Architecture design TC specific, supporting outsourcing, solving potential threats
Reference Architecture	4	• Reference architecture for outsourcing

Figure 5 Research sub questions and deliverables for each phase

Each phase can be projected on the problem solving cycle by Van Aken et al. (2012). This study was done on two abstraction levels: case-specific for TC and more general for high-tech manufacturing companies. In other words, a distinction is made between practice (TC) and theory (literature). Whereas the problem statement and main research question involve the case-specific situation, the result is in Phase 4 generalized for high-tech manufacturing companies. Therefore for both situations certain phases of the problem solving cycle were passed. For the case-specific situation the cycle was fully completed whereas the cycle terminated for the general reference architecture at the solution design phase. No evaluation on the reference architecture was done since there were not enough cases for an appropriate evaluation. Evaluation on the reference architecture is interesting for future research. How each phase for both the case-specific as well the general situation suits the phases of the problem solving cycle is visualized in Figure 6.

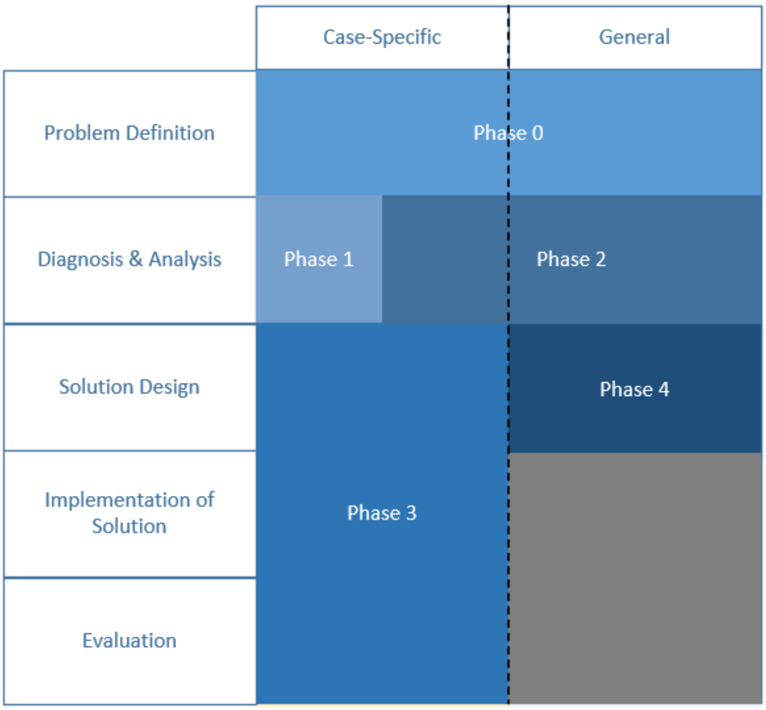


Figure 6 Phases projected on the problem solving cycle of Van Aken et al. (2012)

1.3.6. Scope

In the scoping phase (Phase 0) several scoping decisions were taken.

First, only tactical and operational processes of outsourcing were included in this project comprising part of the activities before and during outsourcing. The reason for this is that the logistics department of the manufacturing plant of TC is not able to take decisions on a strategic level. The department is only able to take operational and tactical decisions: where to outsource, what to outsource, etcetera. Thus, decision-making on a strategic level was out of scope, involving the decision to acquire machines. Additionally, the processes occurring after outsourcing, e.g. review of outsourcing, were also out of scope.

Second this study was done in order to provide a reference architecture for outsourcing for high-tech manufacturing companies and so involved the outsourcing of manufacturing processes, i.e. operations. This indicates that the outsourcing of other services and business functions, as defined in Porter's value chain (1985) and shown in Figure 7, was out of scope of the project.

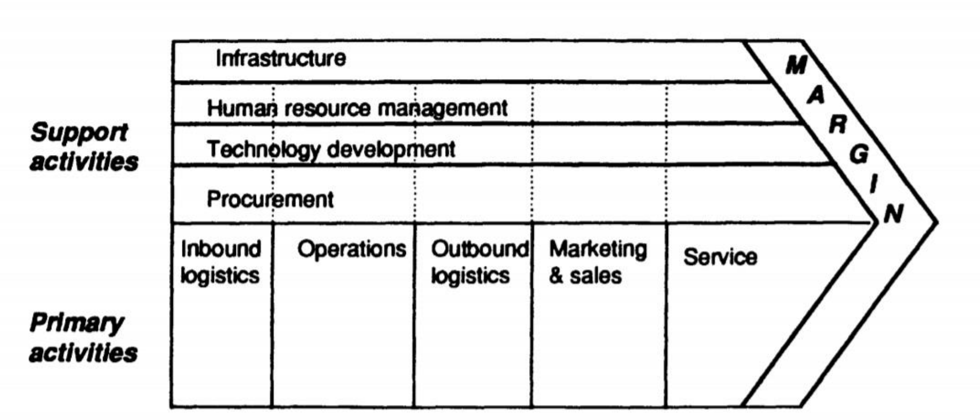


Figure 7 Porter's value chain (Porter, 1985)

Third it was chosen to analyze the outsourcing processes and its underlying ISs, corresponding to the O and the A level of the three-dimensional cube (Grefen, 2016), leaving the (B)usiness level and (T)echnology level out of scope. This scoping decision was needed due to limited time and resources during the project. Besides, these O and the A level were considered as most relevant as they represent the link between business and IT within an organization.

1.3.7. Structure of the Report

The structure of the report largely follows the identified phases. First, theoretical background is provided to the reader in Chapter 2. Then in Chapter 3 to Chapter 6 for each phase the method, results, and conclusions are given. Additionally, an evaluation of the solution design takes place in Chapter 5. In Chapter 7 discussion takes place, drawing main conclusions, providing recommendations, discovering limitations, and suggesting future research. Finally Chapter 8 covers the references list, followed by the appendices. The structure is visualized in Figure 8.

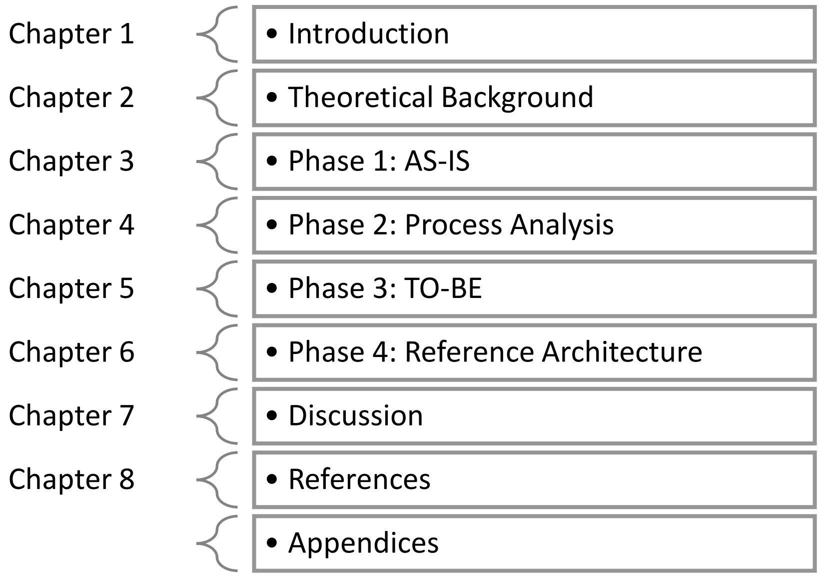


Figure 8 Structure of the report

2. Theoretical Background

This section provides the reader with theoretical background. First the main conclusions of the literature review, completed prior to this study, are given. Then this chapter elaborates in more detail on what is written in literature about outsourcing. Third, the three-dimensional cube is explained which is part of the research framework, as given in Section 1.3.3. Finally, conclusions are given.

2.1. Summary Structured Literature Review

Prior to this thesis a structured literature review was conducted (Van Zutphen, 2017). Starting from a broad scope the research area was explored to gain insights in academic literature on the topic of outsourcing business processes in manufacturing industry and its underlying IS architecture. In order to execute this structured literature review the method of Kitchenham (2006) was used. In this paragraph the main conclusions are briefly discussed.

In the structured literature review it was found that there is a gap in the literature concerning the underlying IS architecture of outsourcing processes. In other words: there is no structure defined for ISs that provide information required for the decisions to be made during the outsourcing process. This makes among others information sharing in this stage of the supply chain complex and unclear, while it was found that information sharing is one of the most important factors in outsourcing (Araz, Ozfirat, & Ozkarahan, 2007). Additionally, it was concluded that customer order decoupling points might influence the IS architecture. The customer order decoupling is the point where the product is linked to a specific customer order with each its own specification.

Thereby it was noticed that the literature is fragmented on the given subject: from multiple points of view the lack of an IS architecture is approached and multiple researchers do face the problem (Bui, Muralikrishnan, & Raja, 2005; Gunasekaran & Ngai, 2007; Hentza et al., 2013; Herrmann, Rogers, Gebhard, & Hartmann, 2015; Khoei et al., 2011; Robson, Watanabe, & Numao, 2007; Tóth, Döbrössy, & Mánik, 2006; Yantao Wang, 2008; Yanyan Wang, Wu, Liu, & Tang, 2007). However, a concrete solution is not given. This indicates the complexity of the subject.

The conclusion drawn in the structured literature review is that there is a need for an IS architecture, underlying the outsourcing process on a tactical and operational level, when (parts of) production processes are outsourced.

2.2. Outsourcing

While the structured literature review focused on what is written in literature about IS architectures underlying outsourcing, this section elaborates on outsourcing processes and the decisions during these processes. It is described why companies use outsourcing, what steps occur in the outsourcing process, and what decisions are made in each step.

As shown in the problem mess (Figure 1), the cause of this study's research problem is an increase in production and in customized products, i.e. high volume and high variety, termed mass customization (APICS, 2011). Kotha (1995) and Sanchez (1997) already addressed the shift in industrial sectors to mass customization (Momme & Hvolby, 2002). As a result companies must adapt their operations while products become more complex. Hence, processes in organizations must be flexible. This is where outsourcing comes in, that is used by companies to be flexible, to lower costs, and to maintain the company's focus on its core operations (Harris, Giunipero, & Hult, 1998).

Outsourcing is a dynamic process since decisions and actions related to outsourcing must continuously be adapted to changes, e.g. as a result of the gap between supply and demand. Additionally, outsourcing is a recurring process: at the end of the contract period a decision must be made whether to prolong the relationship, find another supplier, or insource the products (Momme & Hvolby, 2002).

2.2.1. The Outsourcing Circle

Several researchers identified phases of an outsourcing process. Perunovic (2006) found in his research a number of frameworks describing the stages of the overall outsourcing process, including strategic, tactical, and operational processes (Click & Duening, 2005; Corbett, 2004; Cullen & Willcocks, 2003; Franceschini, Galetto, Pignatelli, & Varetto, 2003; Greaver, 1999; McIvor, 2005; Momme & Hvolby, 2002). Most of these frameworks focus on the strategic phase of outsourcing where it is decided whether to outsource or not (Perunović, 2007). Perunovic (2006) aligned the frameworks and grouped the phases into the following sequence: (1) preparation, (2) vendor(s) selection, (3) transition, (4) managing relationship, and (5) reconsideration. The grouping of the phases of each framework is visualized in one figure by Perunovic (2006), shown in Figure 9.

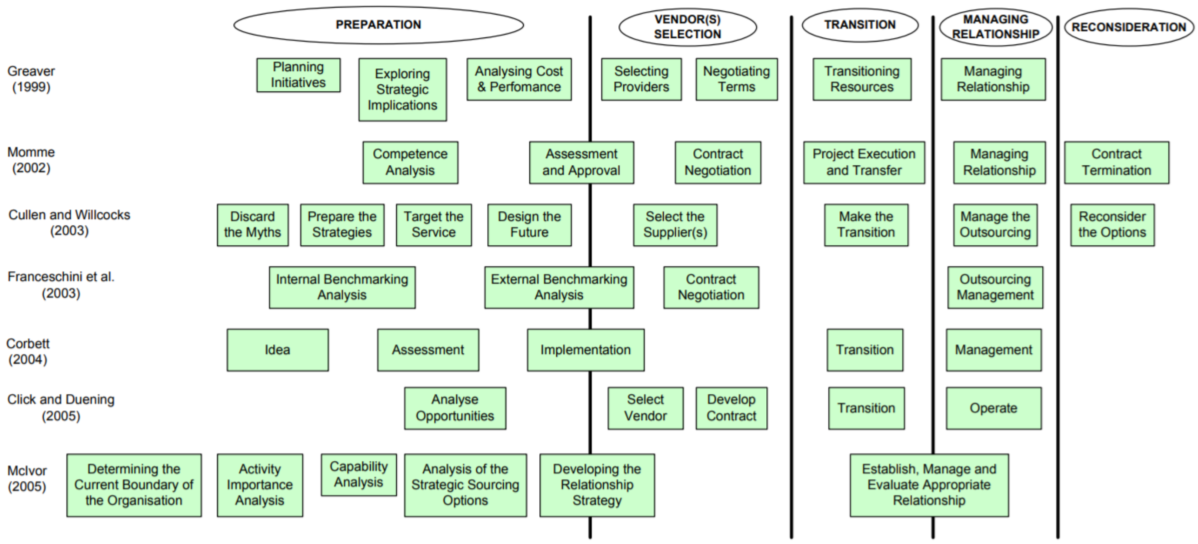


Figure 9 Phases of all frameworks grouped, by Perunovic (2006)

In each phase decisions are made regarding the outsourcing process: whether to outsource, what to outsource, where to outsource, when to outsource, how to outsource, to whom to outsource, how to manage the outsourcing, and what after the outsourcing (Perunović, 2006). This is visualized by Perunovic in the 'outsourcing circle' as shown in Figure 10, in which the decisions that are out of scope are indicated. In each phase activities occur that are now described.

2.2.1.1. Preparation

The major task in the preparation phase is to explore outsourcing options. The main debate is that the company should keep its core activities in house while those not important may be outsourced, which is part of the strategic outsourcing process and thus out of scope during this study. Then the outsourcing approach should be set, determining the basic shapes of outsourcing agreements. The next step is to determine the configurational agreements which relate to a high-level description of the set of choices the organization makes. One of these choices is the duration of the outsourcing agreement (Perunović, 2006).

2.2.1.2. Vendor(s) Selection

The key activities within the vendor(s) selection phase are: writing a request for proposal, determining evaluation criteria, evaluating and selecting vendor, negotiating and finalizing the contract (Perunović, 2006).

2.2.1.3. Transition

The key activity during the transition phase is the initiation of outsourcing to the selected vendor. This is typically focused around transferring assets, people, contracts, hardware and software, information and projects that the vendor will have responsibility for in the future (Perunović, 2006).

2.2.1.4. *Managing Relationship*

In this phase the outsourcer has to establish communication, information and knowledge sharing, and monitoring systems in order to secure a successful outsourcing relationship (Perunović, 2006).

2.2.1.5. *Reconsideration*

Outsourcers should use the reconsideration phase as time to put together and see whether they have achieved success or they have failed and to decide what to do now. There are possible options: continue with the same supplier, contract another supplier, or insource (Perunović, 2006). Note that this phase is out of scope.

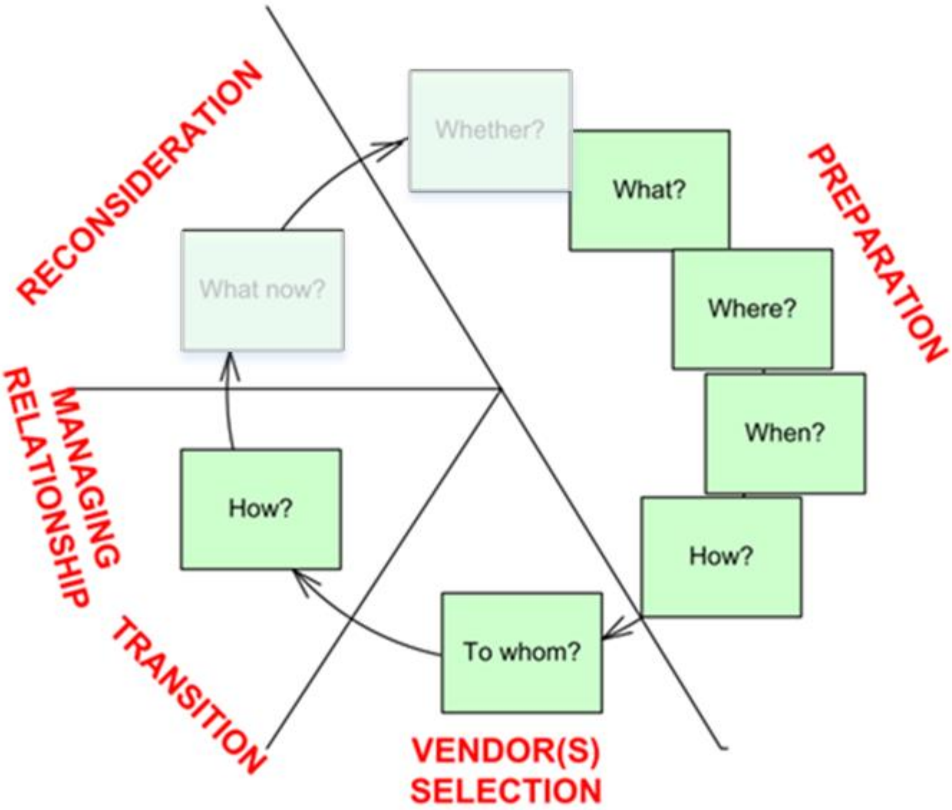


Figure 10 The outsourcing circle *(Perunović, 2006)*

2.3. Grefen's Three-dimensional Cube

One of the objectives of this thesis involves mapping outsourcing processes and underlying architectures. In order to organize architecture descriptions, multiple dimensions can be distinguished. These dimensions help in describing architectures. Four dimensions are distinguished, according to Grefen (2016), which are each described briefly in the following subparagraphs. In the last subparagraph it is shown how these dimensions can be visualized in a cube. As mentioned in Section 1.3.3 the three-dimensional cube is part of the research framework in order to structure and transform different models.

2.3.1. The Aggregation Dimension

In the aggregation dimension it is determined how many details an architecture description should include. The less detailed, the more aggregated the architecture is. The less aggregated the more components the architecture contains. Usually aggregation levels are used to describe the architecture from an overall picture to a detailed picture. Aggregation level 0 indicates a black box, showing only one component. When moving one aggregation level down, components at the upper level are exploded, and so on. The levels used during this study are given in Table 2.

Table 2 Levels on the aggregation dimension (Grefen, 2016)

0	The black-box.
1	The black-box is decomposed to the main ISs.
2	The main ISs are decomposed to their subsystems.
3	The subsystems are decomposed to their subsystem components or system tasks.

2.3.2. The Abstraction Dimension

In the abstraction dimension it is determined how abstract/general or concrete an architecture description needs to be. This dimension varies from more abstract (no choices are being made yet) to more concrete (all choices are being made). As an example Grefen (2016) uses the levels given in Table 3 which are also being used during this study.

Table 3 Levels on the abstraction dimension (Grefen, 2016)

1	**Class type components**	Components are described in terms of general software system classes, indicating their functionality.
2	**System type components**	Components are described in terms of general software system types.
3	**Vendor type components**	Components are described in terms of specific software system series from specific vendors.
4	**Vendor version components**	Components are described in terms of specific software systems from specific vendors including their version.

2.3.3. The Realization Dimension

In the realization dimension it is determined whether the description is more business-oriented or more technology-oriented. Grefen (2016) uses the BOAT-framework consisting of the four levels that are described in Table 4.

Table 4 Description of BOAT-levels

1	**B**usiness	Describes the business goals of an information system.
2	**O**rganization	Describes how organizations are structured to achieve the goals defined at the Business level.
3	**A**rchitecture	Covers the conceptual software structure of an information system required to make the Organization level work.
4	**T**echnology	Describes the technological realization of the system of which the architecture is specified at the Architecture level.

2.3.4. The Aspect Dimension

'An aspect of an IS architecture is a specific way to look at that architecture by focusing on specific characteristics of that architecture only' (Grefen, 2016). In other words, an architecture can be viewed from multiple aspects that is chosen in the aspect dimension.

For the aspect dimension a modernized version of Truijens' framework is used (Truijens, Oosterhaven, Maes, Jägers, & Van Iersel, 1990). Based on this framework, Grefen (2016) modernized and adapted Truijens' aspects (1990) to the aspects as given in Table 5.

Table 5 Aspects of Truijens et al. (1990)*, modernized by Grefen* (2016)

Data	Describes the organization of the data in an information system, typically in terms of data structure diagrams or specifications.
Process	Describes the organization of the business process managed by or executed in an IS, in terms of business process models.
Software	Describes the organization of the software of an information system in terms of its modules and connection between them.
Platform	Describes the organization of the software and hardware underlying an information system.
Organization	Describes how the information system is embedded into an organization for its design, implementation, and maintenance.

2.3.5. The Design Cube

The four dimensions can be visualized in a design cube, shown in Figure 11. In this cube the aspect dimension is made implicit. This means that this dimension is hidden to keep the cube readable. Each cell in this cube is characterized by a certain level of abstraction, aggregation, and realization. During this study the cube's visualization of Tummers (2017) was used since the visualization of Grefen (2016) assumes a start and end-position for a complete design process. During this thesis specific cells of the cube were used as the start and end-position and thus Grefen's original visualization does not correspond to the situation discussed in this thesis.

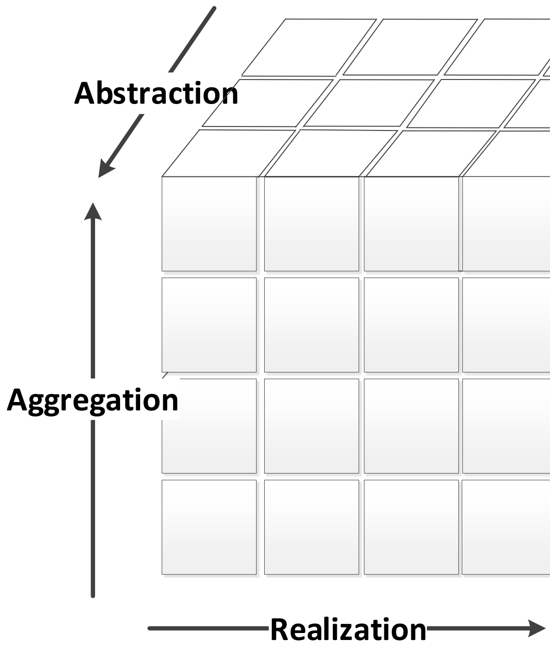


Figure 11 Three-dimensional cube (Grefen, 2016; Tummers, 2017)

To make visualization easier and more understandable the 3D cube is transformed to a 2D model using cross-sections of the cube. The cross-section is done for both the Organization as well the Architecture level of the realization dimension (respectively level 2 and 3) as defined in the scope. It includes the abstraction level on the horizontal axis and aggregation level on the vertical axis. Figure 12 gives the cross-sections of the cube that are used for describing each model during each phase. This figure is used to show the position of each model in the cube during this project.

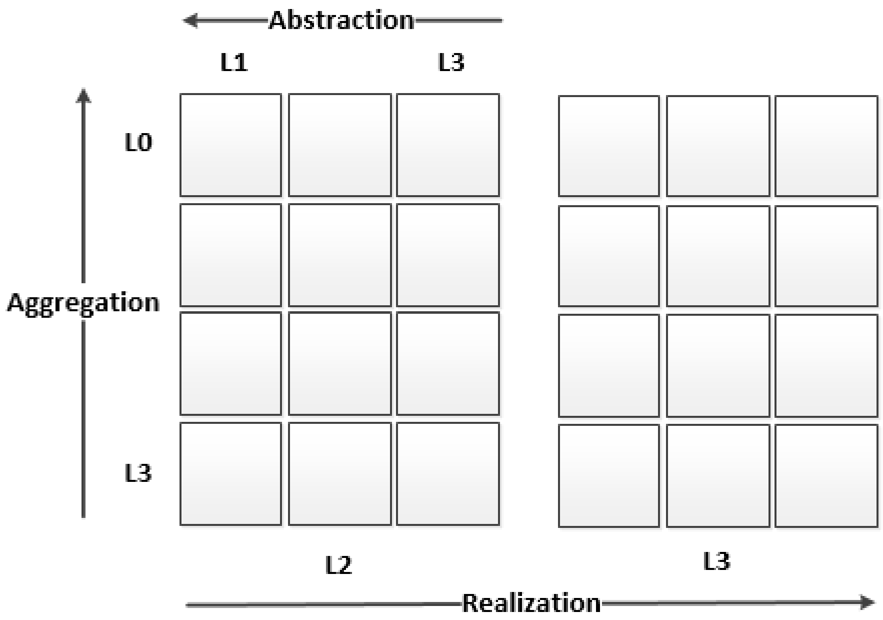


Figure 12 Cross-section of the cube

2.4. Conclusions

In this chapter the results of the literature review, executed prior to this thesis, were discussed. The most important outcome of this review is the need for an IS architecture containing ISs supporting the outsourcing processes: what information is needed from what system for which decision? It was also found that this might depend on the positioning of the customer order decoupling point.

Further it was discussed that several studies elaborated on the phases that are passed during the outsourcing process. Perunovic (2006) grouped these researches in a single framework. In this way he identified five phases in the outsourcing process: preparation, vendor(s) selection, transition, managing the relationship, and reconsideration. For each phase it is described what decisions need to be made. The outsourcing circle (Figure 10) combines the phases with its corresponding decisions in one visualization.

Finally the three-dimensional cube (Figure 11) of Grefen (2016) has been explained. This cube describes architectures from several dimensions: abstraction, aggregation, realization, and aspects. The cube is part of the research framework of this thesis and is used to structure and transform models to other levels of the given dimensions.

3. AS-IS

This chapter covers Phase 1: the AS-IS phase. An answer to the first research question (as given in Table 1) is provided by capturing the current situation at the manufacturing plant of TC and its potential threats. First, the method to explore the current situation is described, followed by the results, and finally partial conclusions. This chapter is part of the diagnosis and analysis phase of the problem solving cycle of Van Aken et al. (2012) for the case-specific situation as given in Table 1 and Figure 6.

3.1. Method

In this section the method for the AS-IS phase is described. First, it is illustrated how the three-dimensional of Grefen (2016) was used. Second, it is given what research techniques were used to gather information serving as input for the models. Finally, this section describes how the processes and IS architectures were modeled to visualize them.

3.1.1. Grefen's Three-dimensional Cube

In order to model the current processes and underlying IS architecture the dimensions of Grefen's (2016) cube were used. The cube and its dimensions are discussed in Paragraph 2.3. In this section for each dimension it is described what levels were used.

3.1.1.1. Aggregation

First, the aggregation level was set. The modeling of processes on the aggregation dimension starts on a highly aggregated level (level 0): the black box. For the processes this level contains outsourcing as the main subject. Consequently, the model covers each level up to level 3, on which the activities are described for the company-specific situation, as shown in Figure 13. For the underlying IS architecture also level 3 was used, describing each component/task separately. How the levels of the aggregation dimension were defined for the architecture is described in Section 2.3.1 and given in Table 2.

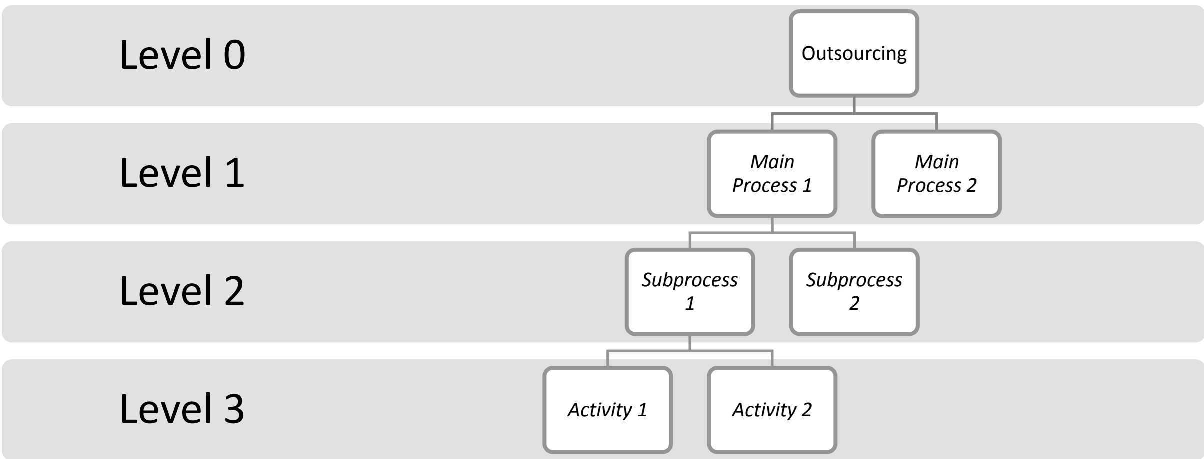


Figure 13 Aggregation levels for the processes

3.1.1.2. Abstraction

Second, the abstraction level was determined. Although the AS-IS situation is company-specific, it was modeled on level 2 of the abstraction dimension. Grefen (2016) gives an example of the values the abstraction dimension can consist of for the underlying ISs. This is given in section 2.3.2 and was used during this study. On abstraction level 2 the components were described in terms of general software system types. Usually, using general terms implies that design choices are not made yet. However, in this case general terms were used due to confidentiality. Also the processes were described in general terms.

3.1.1.3. Realization

Third, the realization level was determined based on scoping, as set in Section 1.3.6. The AS-IS situation covers the current processes and underlying ISs. First, in order to capture the processes the O(rganization) level was used. Second, for the underlying ISs the A(rchitecture) level was used.

3.1.1.4. Aspects

Fourth, the aspects for both realization levels were determined, making use of Grefen's modernized version (2016) of Truijens' Framework (1990), as described in paragraph 2.3.4. For the processes in the O-level the 'Process' aspect and 'Organization' aspect were chosen. These aspects cover the processes, often making use of business process models, and the people (in the organization) executing those processes. For the underlying IS in the A-level, the 'Software' aspect was used. This aspect covers the applications that are supporting the processes described in the 'Process' aspect.

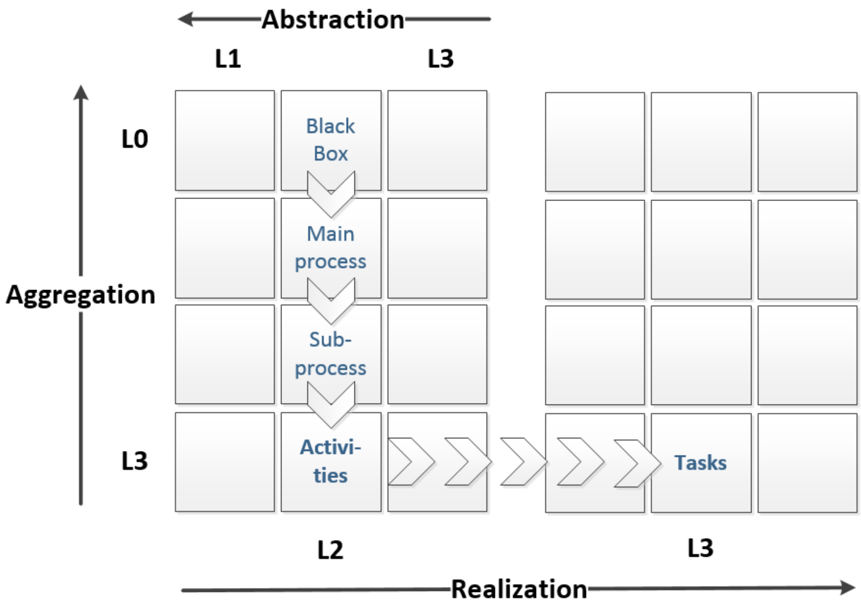


Figure 14 Design path in the AS-IS phase

3.1.1.5. Visualization

The use of these dimensions is visualized in the cube of Grefen (2016), where the aspect level is made implicit. The design path for both processes (L2) and architecture (L3) is shown in Figure 14. First, the organization level (L2) on the realization dimension was modeled from aggregation level 0 to level 3. Then on level 3 of the aggregation dimension the corresponding system tasks were modeled on the architecture level (L3). This resulted in two models, which are positioned in the cube, as visualized in Figure 15. Figure 14 and Figure 15 make use of cross-sections of the cube (Tummers, 2017), for both the organization (L2) and the architecture level (L3) of the realization dimension.

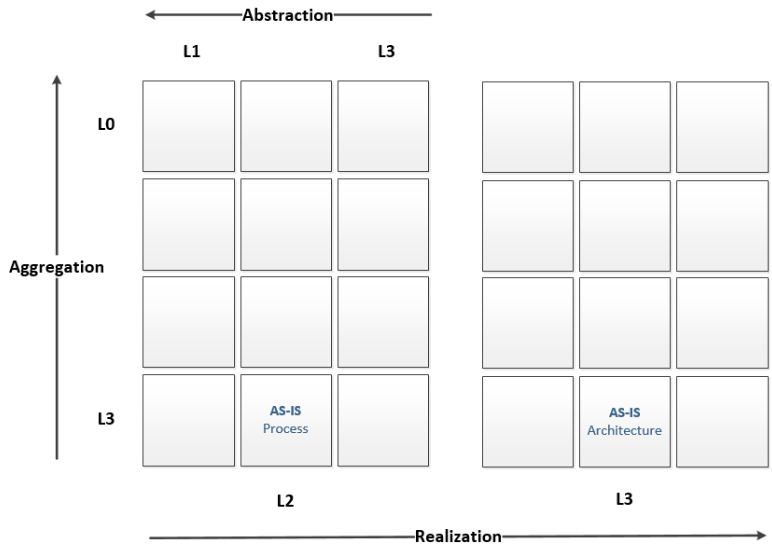


Figure 15 Dimensions of the AS-IS

For both the processes as well the architecture, it is summarized for each dimension what levels were used. This is given in Table 6.

Table 6 Levels of dimensions used for the AS-IS model

Dimension	Processes	Architecture
Aggregation	Level 3: activities separately described	Level 3: each task/component/application separately described
Abstraction	Level 2: specific situation described in general terms	Level 2: specific situation described in general terms
Realization	Level 2: Organization	Level 3: Architecture
Aspect	Process and Organization	Software

3.1.2. Information Gathering

The method to gather information during this phase was qualitative research. The most common qualitative research method is a semi-structured interview (Alvesson & Deetz, 2000). Thereby it is often most effective in gathering information, according to Kvale & Brinkmann (2008). Semi-structured interviews are conversations in which a set of questions is created and the interviewer has a good idea of what topics will be covered (Fylan, 2005). It involves 'prepared questioning guided by identified themes in a consistent and systematic way interposed with probes designed to elicit more elaborate answers' (Qu & Dumay, 2011). A format for interviewing when exploring the AS-IS phase is added in Appendix A.

Each actor involved in the outsourcing process of the TC manufacturing plant was questioned via the interviewing format. During interviewing additional questions were developed to gain deeper understanding. Then the activities executed by those actors were for each actor modeled in process diagrams. Finally, the models were validated for appropriateness and merged into one general model. The method for modeling is described next.

3.1.3. Modeling

For both the outsourcing processes as well the ISs supporting these processes, the method for modeling is now explained.

3.1.3.1. Processes

The outsourcing processes were modeled in business process diagrams. Business Process Modeling Notation (BPMN) was used since BPMN is the global standard for process modeling. Further, the use of BPMN is an highly important factor in the alignment of business and IT (Camunda.org, 2017).

Reading a model in BPMN can be complex. Therefore, this study made use of the basic and most popular BPMN vocabulary to keep the model understandable for the stakeholders (Chinosi & Trombetta, 2012; Muehlen & Recker, 2008).

The outsourcing process was modeled in swim lanes. Note that, in this way the organization aspect of the Grefen's modernized framework (2016) of Truijens (1990) was captured, as described in Table 5. The swim lanes horizontally correspond to the functions or departments involved in the process. The vertical partition corresponds on a higher level to the outsourcing process on aggregation level 1 consisting of the following levels: tactical, operational, and review. Level 2 on the aggregation level is not included in the modeling format since processes on this level might be widely spread over the different functions, lowering the readability of the model. The format used for modeling the outsourcing process is shown in Figure 16.

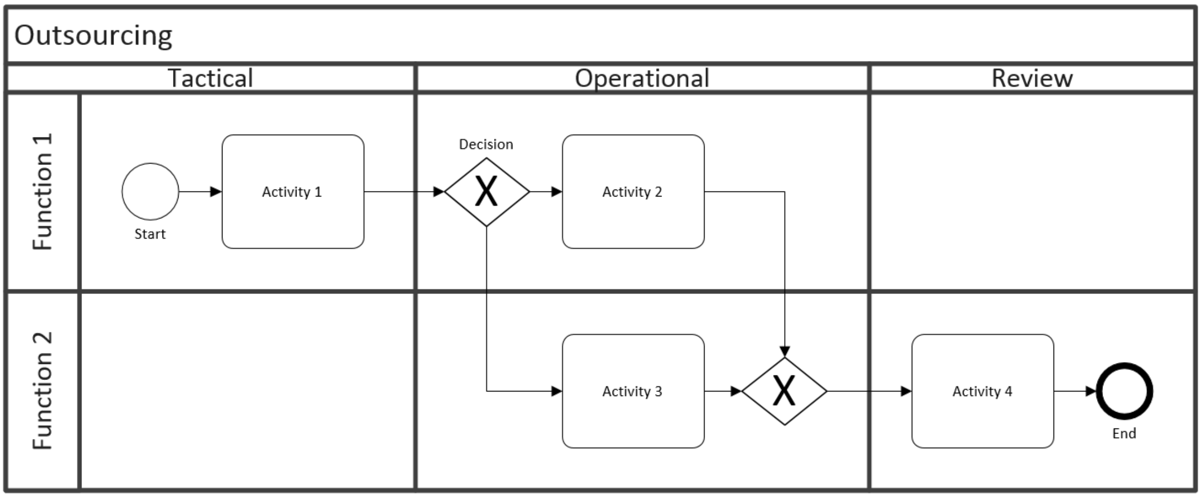


Figure 16 Modeling the outsourcing process in BPMN

3.1.3.2. Architecture

In order to map the current ISs supporting outsourcing an architecture model was created. To structure architecture models one can use styles and patterns (Grefen, 2016). The architecture model used a columned style. More elaboration on the different styles defined by Grefen (2016) is given in Appendix B.

The applications were grouped in the columns by considering whether these applications are supported by the IT department, or are self-built plant-specific. For the applications supported by the IT department a distinction is also made between ERP-applications and other supporting applications.

Simultaneously, the columns also distinguish the functionalities of the applications. The supporting applications have a communication function, the ERP-applications are for material handling, and the self-built plant-specific applications serve the users for managing and monitoring the outsourcing process.

Modeling was done by using ArchiMate, a modeling language. In Chapter 5 more elaboration on this language is provided. For modeling purposes Enterprise Architect (EA) was used. This tool is used by TC as a standard modeling tool and it supports the ArchiMate language notation.

The format for the modeling of the underlying applications is given in Figure 17. Solid lines between applications denote automated interfaces. Dashed lines denote that there is no automated interface between the applications and that transfer of information between these applications is done manually.

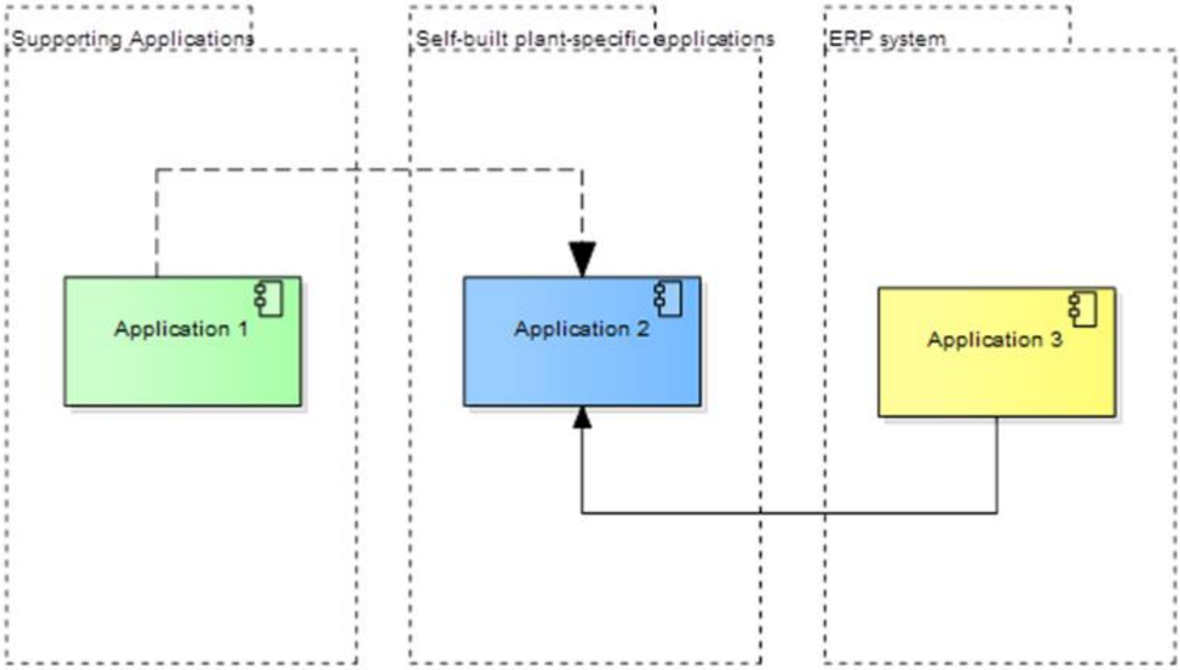


Figure 17 Modeling the underlying applications using a columned style

Finally, with the given that outsourcing is becoming more complex, and occurring more frequently, potential threats were defined by assessing each interface. These potential threats need to be solved by the solution design, i.e. the TO-BE model.

3.2. Results

In this section the results of the AS-IS are described following the method defined in Section 3.1. First, the outsourcing processes are explained. Second, the supporting ISs are given.

3.2.1. Processes

Outsourcing processes within the manufacturing plant are triggered for three reasons, as briefly described in the introduction of this report. First, outsourcing occurs because of capacity reasons. If an increase in production occurs the manufacturing plant might not have enough machines or personnel to meet the requirements. The second reason to outsource is that the manufacturing plant does not have the appropriate machines for a certain production process. This type of outsourcing is structural until strategic management decides to acquire the appropriate machine or the products produced by that machine is terminated. Third, a machine crash might trigger the need for a fast unplanned outsourcing decision.

Then, an outsourcing decision has to be made. The outsourcing decision includes what operations should be outsourced, how many products, for how long, and to whom those operations should be outsourced. This is part of the tactical outsourcing process (preparation and vendor selection). In this decision several departments are involved, e.g. the planning department and the production engineering department, who may have diverging interests, sometimes resulting in conflicts.

When the outsourcing decision is made, a request for outsourcing is asked for. After the request is approved by several departments the outsourcing decision is approved.

Then the operational outsourcing processes were identified. In order to map the operational outsourcing processes, first the outsourcing activities executed for each planner of the planning department were modeled. The process differs between planners, but also contain some similarities. Thereby an outsourcing coordinator overviews some of the processes. All possible ways in the different processes were then modeled in one general process diagram for the logistics department of the manufacturing plant of TC, resulting in multiple paths in the process diagram. The format can be seen in Figure 16. In an abstract way it is now explained what path was taken to obtain the business process diagram. Remember, the last phase of the outsourcing process, comprising review, is out of scope.

To model the outsourcing processes first a black box was created (aggregation level 0), see Figure 18. Then the black box was divided into the main phases of the outsourcing process: tactical, operational, and review (aggregation level 1), as shown in Figure 19. Note that Figure 19 is an explosion of Figure 18. Note also that the review phase is out of scope en thus colored grey.

For TC there are no well-structured sub-processes defined on aggregation level 2. Contrary, activities on aggregation level 3 exist that were allocated to one of the main process phases on aggregation level 1: tactical, operational, or review. The final business process diagram is TC-specific and is therefore not provided for confidential reasons, although it is modeled on a higher abstraction level.

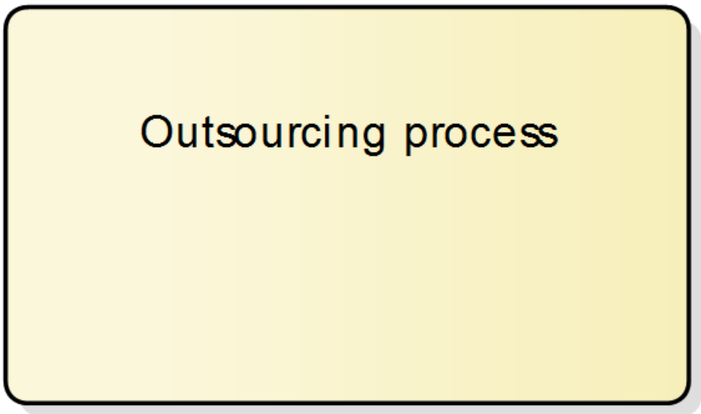


Figure 18 Outsourcing process on aggregation level 0 (black box)

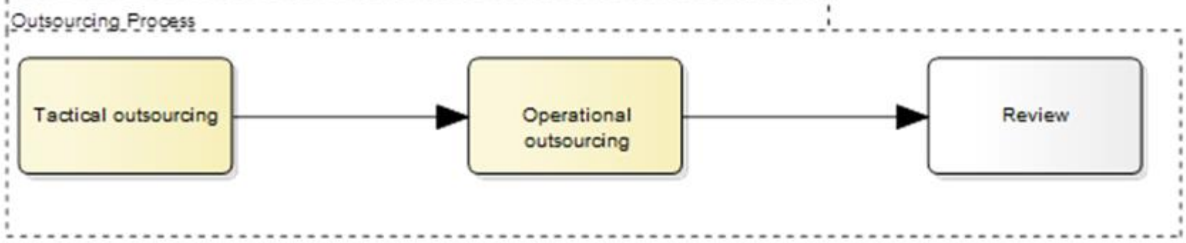


Figure 19 Outsourcing process on aggregation level 1

3.2.2. Architecture

An abstract version of the architecture model is provided in Figure 20. It includes the most critical applications supporting the outsourcing process and its most critical interfaces, numbered from i1 to i11. Since outlook is widely used as a mean to inform and to communicate to other departments (e.g. for financial purposes or for preparation of materials to be sent) it is explicitly mentioned in the figure. Note that there are no outgoing arrows from the outlook component since the model focuses on the logistics department of TC's manufacturing plant and thus e-mails to external parties are not included in the model. Also noticeable is the existence of self-built plant-specific applications and their interfaces, so called shadow IT, which is a significant threat to TC.

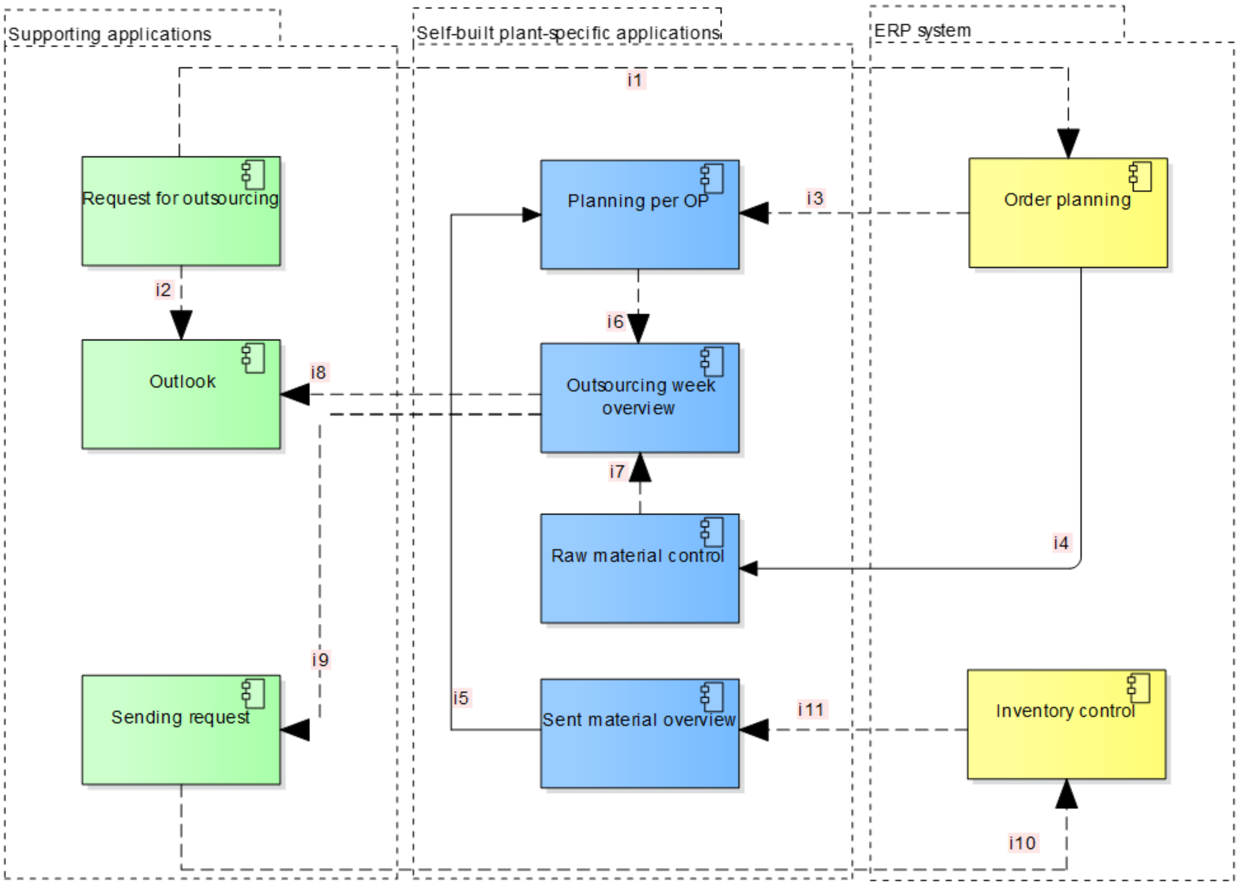


Figure 20 Underlying applications supporting the outsourcing process

In Table 7 for each interface a brief description is given, followed by the occurrence of the interface and whether the interface is executed manually or automatically. Then, it was determined what potential threats arise for these interfaces when outsourcing occurs more often and is becoming more complex. As a result TC gained insights on whether its ISs supporting outsourcing are future proof. The most critical threats are (1) an increase in occurrence and frequency of the manual interfaces, resulting in a higher workload, (2) error-proneness, resulting in e.g. incorrect production requirements for outsourcing partners (OP's), and (3) safety buffers due to a loss of insights, and thus costs, in order to cover for insufficiencies during the manufacturing process. The identified potential threats were solved by the TO-BE model that is proposed in Chapter 5.

Table 7 Potential threats per interface

Inter-face	Description	Configuration	Occurrence	Potential threat:
i1	Based on information defined in the Request for Outsourcing, orders are created and configured in the ERP system (Order Control).	Manually	Event-Based, for each order	Increase in occurrence.
i2	The Request for Outsourcing is communicated via outlook to other departments for information sharing and approvals.	Manually	Event-Based, for each order	Increase in time-delays and occurrence. Information flow is not aligned with the process.
i3	Based on production requirements for components in the ERP system, a planning for the required components to be outsourced per Outsourcing Partner (OP) is created.	Manually	Weekly, for each component	Time-consuming, error-prone. Increase in frequency. High safety buffers.
i4	Based on the needs for materials, a self-built application gives the total need for the requested material for an indicated period.	Automated	Daily updated	Self-Built, not supported by IT department. Potential compatibility problems.
i5	The requirements for components are compared to the components in the pipeline. It is determined whether the components in the pipeline satisfy the requirements.	Automated	Event-based	Needs to be built iteratively for each new order and monitored, increase in occurrence.
i6	Based on the planning per OP, a weekly outsourcing overview is made. This includes what materials needs to be sent to all OP's in order to meet production requirements for components.	Manually	Weekly, and event-based updated	Time-consuming to determine for each component and OP. Increase in occurrence.
i7	In the outsourcing week overview it is determined whether there is sufficient raw material on hand for internal use in the coming period.	Manually	Event-based, for each potential sending	No decision rules defined, task based on experience. Safety-buffers resulting in increasing inventory. Increase in occurrence.
i8	The outsourcing week overview is communicated to other departments via outlook.	Manually	Weekly, and event-based updated	Time-consuming, non-standard, increase in frequency and complexity of product specification.
i9	For each sending in the outsourcing week overview transportation is requested via a sending request.	Manually	Event-based for each OP	Time-consuming, error-prone, increase in occurrence.
i10	Based on the sending request, inventory of the given material is mutated in the ERP system.	Manually	Event-based, for each sending	Time-consuming, error-prone, increase in occurrence.
i11	When inventory is mutated in the ERP system, also the inventory in the sent material overview is adapted.	Manually	Event-based, for each sending	Time-consuming, error-prone, increase in occurrence.

3.3. Conclusion

This section answered the first research question capturing the current situation at TC. In this way TC gained insights on how its current outsourcing processes are supported by ISs, and what potential threats arise when outsourcing occurs more frequently and becomes more complex.

Multiple conclusions can be drawn. First, the existence of multiple paths in the process diagram leads to the need for a standard in the outsourcing process. Second, there is no standard in when and how information is exchanged with the outsourcing partner. Third, internal information exchange is done manually, leading to error-proneness, time-delays, and safety buffers, as shown in Table 7. Finally, multiple self-built applications exists, i.e. shadow IT, causing a potential threat for TC since they are not supported by the IT department and so are uncontrolled.

4. Process Analysis

In Phase 2 the 'diagnosis and analysis' phase of Van Aken et al. (2012) was conducted, during the Process Analysis phase. In this phase it was researched how the AS-IS situation can be compared to both practice and theory, captured by research questions 2a and 2b (Figure 5).

The objective of this section is to provide input for the consecutive phases: Phase 3 (Chapter 5) and Phase 4 (Chapter 6). Note again, that Phase 3 is case-specific for TC's manufacturing plant and Phase 4 applies to high-tech manufacturing companies in general. Hence, the reference architecture also applies to other plants of TC. The input for these phases is caught into a single framework. This framework captures for each decision in the outsourcing process what information is needed, for both the TO-BE situation for TC as well the generalized situation. First, the method to obtain this framework is described, followed by results and conclusions.

4.1. Method

In this section the method for the Process Analysis is described. First, it is described how theoretical references were researched, followed by practical references. Finally, it is given how the framework was developed.

4.1.1. Theory

First, based on a paper of Oracle (2014), commonly observed best practices of outsourcing were taken into consideration. These best practices cover certain decisions made in outsourcing that require certain functionalities supported by ISs. For each decision other information is required. Therefore, it was determined for each decision whether it is applicable to TC or not. In this way a distinction was made between what input is needed for the TO-BE model (Phase 3) and what input is needed for the reference architecture (Phase 4). In this way it can be judged for the TO-BE model in Phase 3 whether the current supporting ISs are sufficient, should be expanded, or replaced.

Second, according to the structured literature review conducted prior to this study, the customer order decoupling point plays an significant role in outsourcing (Van Zutphen, 2017), as described in Section 2.1. In the literature different manufacturing environments are distinguished, based on the positioning of their customer order decoupling point (Robert Jacobs & Chase, 2017). This distinction was used to retrieve practical references, which is now described.

4.1.2. Practice

In order to provide an appropriate general solution design all existing manufacturing environments should be taken into account. Therefore, by using semi-structured interviewing techniques (3.1.2), other plants of TC were included in the analysis. Also an external company in the field of manufacturing was approached to cover all existing manufacturing environments, comprising an Engineer-to-Order environment. A format for interviewing, and information on the interviews (who and when) is given in Appendix C. These plants and the external company were categorized in the different manufacturing environments, depending on the positioning of their customer order decoupling point. The deliverable of these interviews is a list of information that is needed during the outsourcing process. For each kind of information it was again determined whether it is applicable to TC or not. In this way again a distinction was made between what input is required for the TO-BE phase and what input is required for the reference architecture phase.

4.1.3. Framework

By retrieving information from these theoretical and practical references a translation can be made from processes to IT. The initial step in this translation is done by creating a framework that consists of the information that is required for each phase of the outsourcing process. Note that, the completion of this translation is executed in Chapter 5 and Chapter 6, where the relevant information is linked to systems for the solution designs in respectively the TO-BE model and reference architecture.

Therefore, for each kind of information obtained from the references it was decided for what decision it is needed during the outsourcing process. For the outsourcing process the outsourcing circle of Perunovic (2006) forms the basis on aggregation level 2 (Figure 21). In the AS-IS phase no well-structured sub processes were defined yet on this level. The outsourcing circle is given in Figure 10 and consists of the following phases: preparation, vendor(s) selection, transition, managing relationship, and reconsideration. The reconsideration phase is colored grey since it is out of scope.

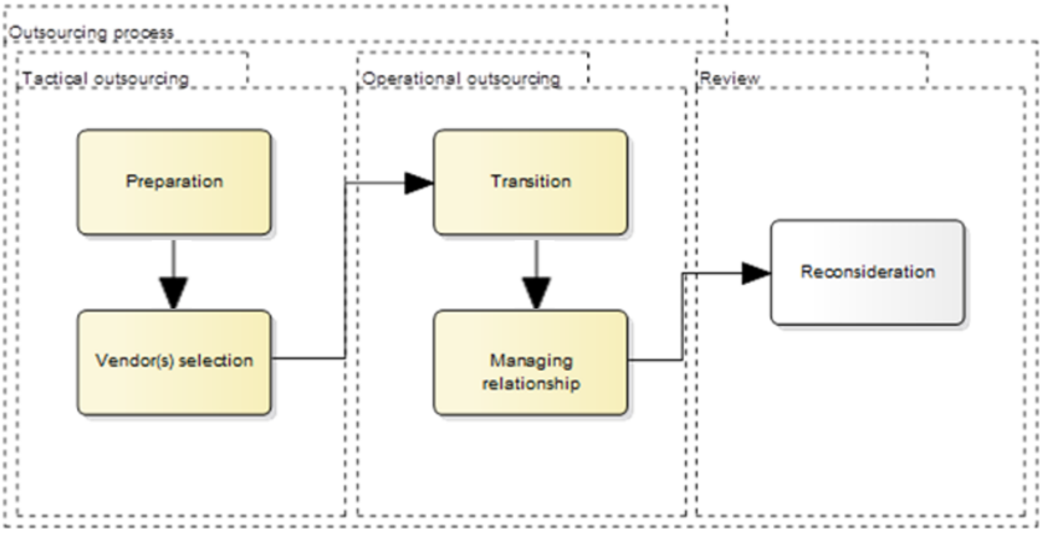


Figure 21 Outsourcing process on aggregation level 2

In this way the analysis framework was developed. The framework consists of five columns. The first column diverges between operational and tactical outsourcing on aggregation level 1 (Figure 19). The second column explodes this level of aggregation to level 2 (Figure 21), consisting of the phases of Perunovic (2006). The third column consists of the decisions to be made in each phase, according to Perunovic (2006). These decisions are discussed in Section 2.2. Finally, this framework diverges between information that is needed in the TO-BE phase (fourth column) and additional information needed during the reference architecture phase (fifth column). A format for the framework is given in Table 8.

Note that the reconsideration phase of Perunovic (2006) is excluded since only tactical and operational processes are in the scope of this study, as mentioned in Section 1.3.6. Note also that the transition phase and managing relationship phase are combined, since they comprise similar decision-making.

Table 8 Format for the analysis framework

Phase (level 1)	Phase (level 2)	Decision	Information for TO-BE phase	Additional information for reference architecture phase
Tactical	**Preparation**	What? When? Etc.		
	Vendor(s) Selection	To Whom?		
Operational	**Transition**	How?		
	Managing Relationship			

4.2. Results

In this section the results of the process analysis are discussed following the method described in the previous section. First the AS-IS situation was compared to theory. Second, based on theory, practical references were explored. These practical references were categorized into several manufacturing environments. Finally, the analysis framework is given.

4.2.1. Theory

The paper of Oracle (2014) defined best practices relevant for the outsourcing process. As mentioned earlier, outsourcing occurs for several reasons that might result in benefits for the company. However, according to Oracle (2014): 'to derive these benefits of outsourcing, companies must deploy appropriate IT systems to deal with these challenges'. In other words these outsourcing best practices should be appropriately supported by ISs. Several factors play a role for companies and their ISs when outsourcing (parts of) their production processes to an outsourcing partner (OP). According to Oracle (2014) the following factors play a role:

(1) Extent of Outsourcing: the extent to which the manufacturing process is being outsourced.
(2) Supply of Components: by whom the supply of components is controlled.
(3) Ownership of Components: by whom the components are owned and/or managed.
(4) Shipment of Finished Goods: the way finished goods are supplied to customers.

For each factor there are several possibilities relevant for outsourcing shown in Table 9. It is also given whether these factors apply to the TO-BE situation of the manufacturing plant of TC. Each possibility results in different relevant information and so in different requirements for the ISs, supporting outsourcing.

Table 9 Outsourcing best practices (Oracle, 2014)

(1) Extent of Outsourcing	**To what extent is the manufacturing process being outsourced?**	**Applicable to TO-BE?**
a) Completely outsourced	Manufacturer outsources complete manufacturing to OP by raising a purchase order for supply of the product.	Yes
b) Partly outsourced	Manufacturer handles a portion of its operations by itself and outsources the rest to one or multiple OP's. The OP gets paid for the value added services.	Yes
(2) Supply of Components	**Who controls the supply of components?**	
a) The Outsourcing Partner	The OP either manufactures or procures the necessary components, which' costs are factored into the price.	Yes
b) The Manufacturer	Manufacturer control the supply of components by shipping the components directly to the OP or by engaging a supplier to drop these components to the OP's facility.	Yes
(3) Ownership of Components	**Who owns and manages the components?**	
a) The Manufacturer	Manufacturer owns and manages component inventory at the OP facility and periodically replenishes stock based on consumption.	Yes
b) The Supplier	Supplier consigns inventory at the OP facility.	No
c) The Outsourcing Partner	Manufacturer sells the components to OP resulting in a complete transfer of ownership.	No*
d) Chargeable subcontracting	The manufacturer ships and makes a provisional sale of components used to build the product at OP's facility. Ownership of components still lies with the manufacturer and inventory is reported under manufacturers' inventory valuation. The manufacturer periodically nets payable and receivable invoices and makes payment to the OP only for the value addition.	No
4) Shipment of Finished Goods	**How are the finished goods supplied to the customer?**	
a) Via manufacturer	Manufacturer ships finished products to customer upon receipt from OP.	Yes
b) Directly from OP	OP directly ships finished products to customer.	No

*) Components are owned by the manufacturer (TC), however there is one exception at the manufacturing plant of TC that sales components and subsequently TC repurchases processed components.

Note that, for both 'extent of outsourcing' and 'supply of components' all possible ways apply to TC. Therefore, the ISs of TC should be able to provide information relevant for these possibilities. In contradiction, a number of possibilities in ownership of components and shipment of finished goods are not relevant to TC. Hence, these possibilities were taken into consideration for the reference architecture phase.

4.2.2. Practice

Another factor playing a role in outsourcing and influencing the supporting ISs is the customer order decoupling point. The customer order decoupling point 'determines where inventory is positioned to allow processes or entities in the supply chain to operate differently' (Robert Jacobs & Chase, 2017). The positioning of the customer order decoupling point distinguishes four manufacturing environments, which are described in Table 10.

Table 10 Manufacturing environments (Robert Jacobs & Chase, 2017)

Make-to-stock	'A production environment where the customer is served 'on demand' from finished goods inventory'
Assemble-to-order	'A production environment where pre-assembled components, subassemblies and modules are put together in response to a specific customer order'
Make-to-order	'A production environment where the product is built directly from raw materials and components in response to a specific customer order'
Engineer-to-order	'A production environment where the firm works with the customer to design the product, which is then made from purchased material, parts and components'

These manufacturing environments, and the position of its customer order decoupling points (denoted by 'DP'), are abstractly visualized in Figure 22. This figure is based on a study of Emmet and Crocker (2016).

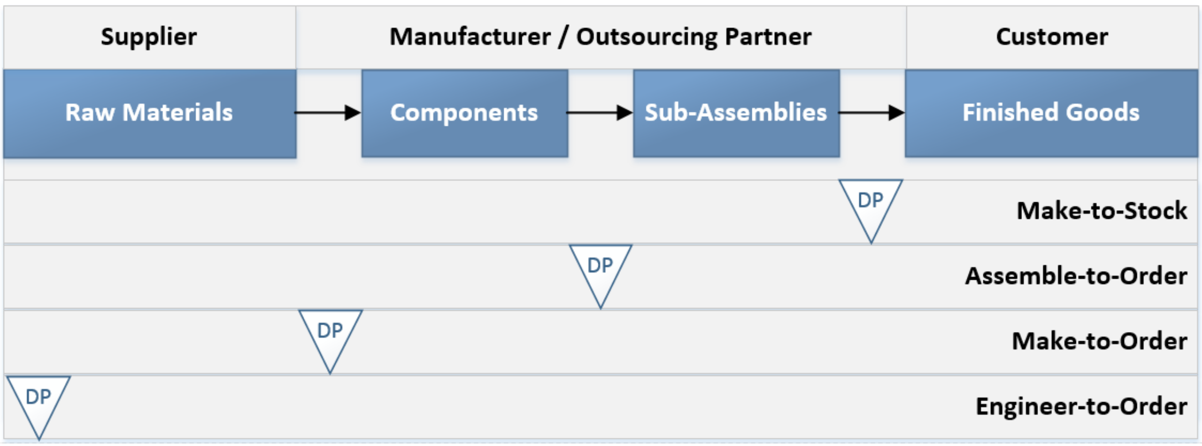


Figure 22 Positioning the customer order decoupling point (based on Emmet and Crocker, 2016)

For the firm's ISs that support outsourcing, the customer order decoupling point is important, since different information is needed at different stages of production. Hence, it matters at what point the order becomes customer specific. The order can become specific before outsourcing, after outsourcing, or during outsourcing. Based on the descriptions (Robert Jacobs & Chase, 2017), the positions of the customer order decoupling points for each manufacturing environment in the context of outsourcing are given in Figure 23. In this abstract visualization the manufacturing partner and assembly partner serve as a possible outsourcing partner to whom the manufacturing company outsources its activities. An assembly partner assembles components to (sub-)assemblies, and a manufacturing partner manufacture components from raw materials or preliminary components.

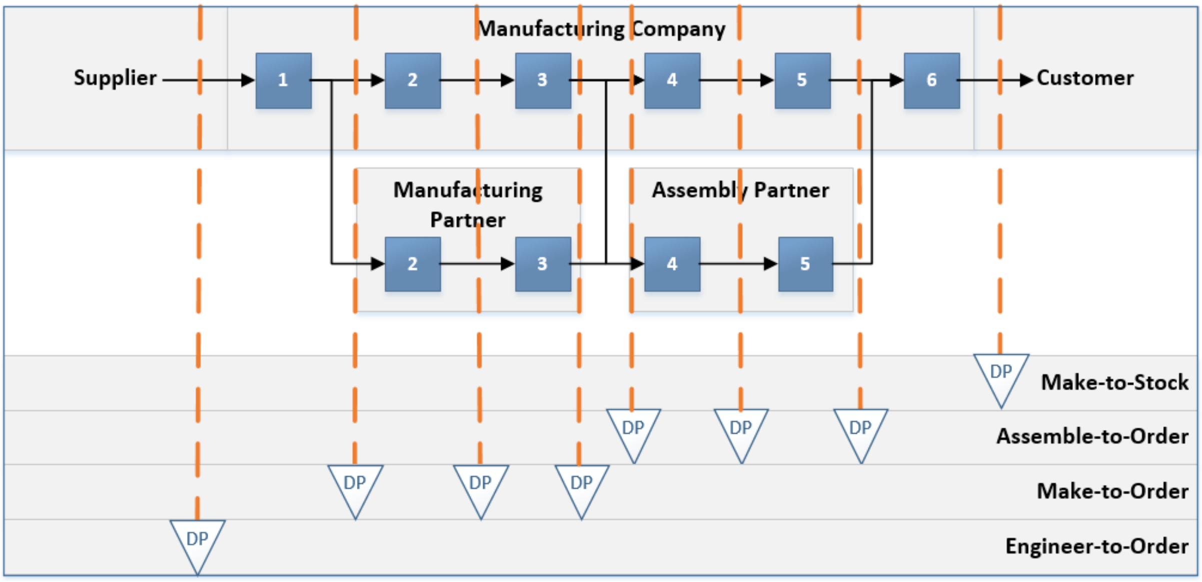


Figure 23 Positioning of the customer order decoupling point in the context of outsourcing

In order to obtain a sufficient framework, each manufacturing environment is captured in the analysis. It is now described for each manufacturing environment what practical references were approached and what information these references noticed as relevant during the outsourcing process.

4.2.2.1. Make-to-Stock

First, in the Make-to-Stock environment the AS-IS situation was categorized. For the manufacturing plant of TC in the AS-IS situation the orders are based on forecast orders and sales orders. However, the components manufactured are not customer specific yet for most cases. Therefore, the customer order decoupling point is in the process positioned after the AS-IS manufacturing plant. Thus, in the context of outsourcing, the customer order decoupling point is positioned after outsourcing. Therefore the manufacturing plant is, according to the definitions described, categorized as a Make-to-Stock environment.

The most significant factors in the outsourcing process for this Make-to-Stock environment are the capacity, the inventory, and the planning. In other words, the comparison between how many components are on hand, how many should be manufactured, and how many can be manufactured.

4.2.2.2. Assemble-to-Order

Within TC there is an assembly plant that assembles manufactured components together with procured components into final products. This plant can be categorized as an Assemble-to-Order plant. The final product is customer specific and possesses an unique identifying number for a specific customer. In this manufacturing environment the customer order decoupling point lies before, after, or while outsourcing.

The most significant factors for this Assemble-to-Order environment are the expected deliveries of the customer specific components that include specific product options.

4.2.2.3. Make-to-Order

TC also owns plants that manufacture customer specific components from raw materials and components manufactured in the AS-IS plant. These customer specific components possess an unique identifying number for a specific customer. In this manufacturing environment the customer order decoupling point lies before, after, or while outsourcing.

The most significant factors for this Make-to-Order environment are the lead times and production status at the outsourcing partner, and whether the outsourcing partner is running on schedule.

4.2.2.4. Engineer-to-Order

In order to complete the categorization, additionally a project-based company was analyzed during the process analysis. This company manufactures for each customer a unique product, which it calls a project. Each new customer involves a new product and hence a new product design. Based on the product design, raw materials are purchased. In this environment the customer order decoupling point is positioned at the start of the supply chain. In this way the customer order decoupling point is always located before the outsourcing.

The most significant factor for this Engineer-to-Order environment is the allocation of production over the outsourcing partners by considering their capacity and expected delivery times.

4.2.2.5. Information in Outsourcing

Based on semi-structured interviews with these practical references, several kinds of information were defined. For each kind of information it was determined whether it is applicable for the TO-BE situation of the manufacturing plant of TC. The result is given in Table 11.

Table 11 Information gathered from each manufacturing environment

Manufacturing Environment	Reference	Information	Applicable to TO-BE?
Make-to-Stock	TC: plant AS-IS	1) Material/Component definition	Yes
		2) Current capacity	Yes
		3) Current inventory	Yes
		4) Forecasted / Sales order definition	Yes
		5) Period to outsource	Yes
		6) Machines to outsource	Yes
		7) Raw materials for internal use	Yes
		8) Type of packaging	Yes
		9) Tooling needed	Yes
		10) Lead time at OP	Yes
		11) Quality requirements	Yes
		12) OP definition	Yes
		13) Costing price of operation to outsource	Yes
Assemble-to-Order	TC: assembly plant	14) Bill of materials (BOM)	No
		15) Bill of processes (BOP)	No
		16) Product/Material options	No
		17) Customer-Identifying-Number	No
		18) Expected Delivery Date (EDD)	No
Make-to-Order	TC: other manufacturing plants	19) Product/Material options	No
		20) Rejections of components	Yes
		21) Forecasted / Sales order definition	Yes
		22) Inventory at OP	Yes
		23) Lead time at OP	Yes
Engineer-to-Order	Project-based company	24) Currency rates	Yes
		25) Capacity at OP	Yes
		26) Inventory at OP	Yes
		27) Expected Delivery Date (EDD)	No
		28) Customer-Identifying-Number	No
		29) Price of operation at OP	Yes

4.2.3. Framework

Finally, based on Table 9 and Table 11, for both TO-BE and reference architecture relevant information is given in this section. Each kind of information is categorized, based on its relevance for the decision of each phase of the outsourcing circle of Perunovic (2006). Note again that the 5th phase (reconsideration) is not included due to scoping. The framework is shown in Table 12. Information applicable to the TO-BE model is automatically applicable to the reference architecture.

Table 12 Information applicable to TO-BE and reference architecture

Phase (level 1)	Phase (level 2)	Decision	TO-BE model	Additional for reference architecture
Tactical	**Preparation**	What? When? Etc.	- Completely/partly outsourced - Material/Component definition - Current capacity - Current inventory - Forecasted/Sales order definition - Period to outsource - Machines to outsource - Tooling needed - Quality requirements - Costing price of operation to outsource	- Customer-Identifying Number - Bill of Materials - Bill of Processes
	Vendor(s) Selection	To Whom?	- Lead time at OP - OP definition - Capacity at OP - Currency rates - Price of operation at OP	
Operational	**Transition** **Managing Relationship**	How?	- Supply of Components via OP or manufacturer - Ownership of Components by OP or Manufacturer - Shipment of Finished Goods via manufacturer - Forecasted/Sales order definition - Raw material for internal use - Type of packaging - Rejections of components - Inventory at OP - Tooling needed	- Ownership of Components by Supplier or chargeable subcontracting - Shipment of Finished Goods directly from OP - Product/material options - Expected Delivery Date (EDD) - Customer-Identifying Number

4.3. Conclusion

From this section several conclusions can be drawn.

This section provided input for both the TO-BE model as well the reference architecture. For each manufacturing environment, including TC's, it was determined what information is relevant for each decision in the outsourcing process. The manufacturing environments were distinguished based on the position of their customer order decoupling point, resulting in different kinds of required information. The most important difference is the addition of customer specific options on products, making each outsourced operation unique.

Based on the required information (Table 12), ISs that provide these types of information can be identified in the next phases. Therefore, this section provided a significant input in the translation from business (processes) to IT (systems).

5. TO-BE

This section describes Phase 3, comprising the case-specific TO-BE situation. This section covers the third research question, as mentioned in Figure 5. The objective was to obtain a standard outsourcing process and to structure supporting ISs for the manufacturing plant of TC. The latter objective also answers the main research question of this thesis, as given in Section 1.3.1. It comprises the solution design, implementation, and evaluation of the problem solving cycle of Van Aken et al. (2012) for the case-specific situation as described in Section 1.3.5.

First, the method is discussed to obtain the TO-BE model and to evaluate it. Second, results are given consisting of a standard outsourcing process for TC and an IS architecture. Third, an evaluation on the solution design is given. Finally, conclusions are drawn.

5.1. Method

In this section the method is explained to obtain the TO-BE model, which visualizes both the processes as well as the ISs supporting them. First, for each dimension of the three-dimensional cube (Grefen, 2016) the levels were determined. Second, the method to gather information is given. Third, it is discussed how the standard outsourcing processes and the IS architecture were modeled. Finally, a method to evaluate the TO-BE model is given.

5.1.1. Grefen's Three-dimensional Cube

The TO-BE model makes use of the three-dimensional cube of Grefen (2016). This is done in a similar way to the method used for obtaining the AS-IS situation, as described in Section 3.1.1. For each dimension the levels were determined on which the TO-BE situation is modeled. Consecutively, the aggregation, abstraction, realization, and aspect levels were set.

5.1.1.1. Aggregation

First, the aggregation dimension was chosen. Contrary to the AS-IS model, the TO-BE model uses level 2 of the aggregation dimension since less detail is needed. As defined in Section 3.1.1.1 level 2 comprises the sub-processes. For this level the phases of Perunovic (2006) were used, as shown in Figure 21: preparation, vendor(s) selection, transition, managing the relationship, and reconsideration. For the architecture this level uses subsystems, unlike the AS-IS model that consists of detailed tasks executed by these subsystems.

5.1.1.2. Abstraction

Second, the abstraction dimension was set. As for the AS-IS model, level 2 was used to model the situation in general terms due to confidentiality, as described in Section 3.1.1.2.

5.1.1.3. Realization

Third, the realization dimension was determined, based on the scoping set in Section 1.3.6. Similar to the AS-IS situation the processes and architecture were modeled. Respectively, the O-level and the A-level were used, as explained in Section 3.1.1.3.

5.1.1.4. Aspects

Fourth, the aspects captured in the model were chosen. For the O-level again the process aspect was used. The organization aspect was not modeled for the TO-BE since the processes are on the given aggregation level not assigned to certain departments or functions. For the A-level again the software aspect was used. These levels are discussed in Section 2.3.4.

5.1.1.5. Visualization

Finally, each dimension was visualized in the cube, as for the AS-IS situation (Figure 15). As shown in Figure 24 the transformation from the AS-IS to the TO-BE model contains a shift of one level up on the aggregation dimension. This results in a less detailed model. Note that the aspect dimension is again made implicit in this visualization, so it is not visible that the organization aspect was excluded.

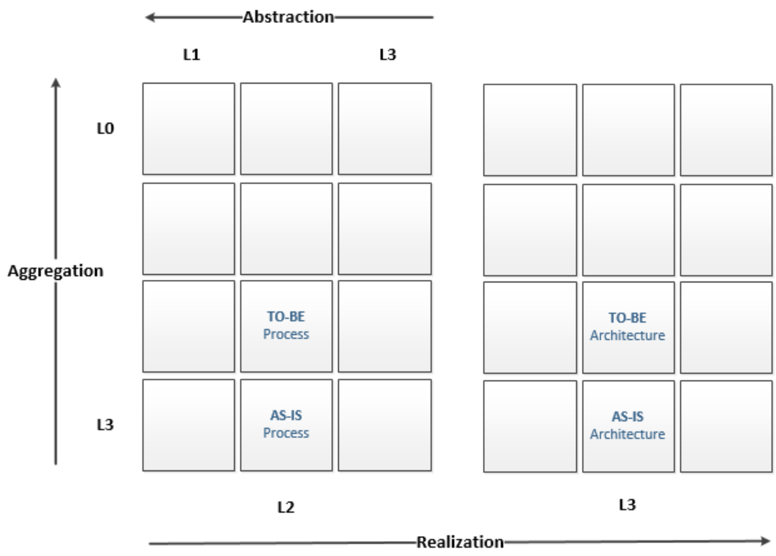


Figure 24 Dimensions of TO-BE model

In Table 13 it is summarized for each dimension of the cube what levels were used for both the standard outsourcing process as well the IS architecture.

Table 13 Levels of dimensions used for the TO-BE model

Dimension	Processes	Architecture
Aggregation	Level 2: phases of Perunovic (2006)	Level 2: each subsystem described
Abstraction	Level 2: specific situation described in general terms	Level 2: specific situation described in general terms
Realization	Level 2: Organization	Level 3: Architecture
Aspect	Process	Software

5.1.2. Information Gathering

In order to gather information the TO-BE model made use of both the case-specific AS-IS situation (Chapter 3) as well the Process Analysis (Chapter 4).

In order to design a standard outsourcing process the outsourcing circle of Perunovic (2006) was used as a foundation.

For the underlying ISs it was determined what information is relevant for each decision in the outsourcing circle. This is the result of the AS-IS situation and the Process Analysis. Then by interviewing experts from the IT-department of TC it was found from what ISs this information is retrievable. Also a former IS architect of TC was questioned. In this way it is found what ISs capture the information that is needed for each decision in the outsourcing process or what ISs should be expanded. When the required information is not captured by one of these ISs it is marked as a required system.

5.1.3. Modeling in ArchiMate

For modeling the TO-BE situation the ArchiMate language was used. This language makes it possible to cover both business processes and its underlying applications in one model (Lankhorst, Proper, & Jonkers, 2010). This is also useful in this study: through using ArchiMate it can easily be visualized how each phase of the outsourcing process is supported by what ISs. First, an introduction on ArchiMate is given. Second, design choices, using ArchiMate are explained.

5.1.3.1. Introduction to ArchiMate

ArchiMate distinguishes between three layers: the Business layer, the Application layer, and the Technology layer.

- The Business layer offers products and services to external customers, which are realized in the organization by business processes performed by business actors and roles (ArchiMate, 2018).
- The Application layer supports the business layer with application services which are realized by (software) application components (ArchiMate, 2018).
- The Technology layer offers infrastructural services needed to run applications, realized by computer and communication hardware, and system software (ArchiMate, 2018).

The most important concepts of ArchiMate are shown in Figure 25. In this figure the components of each layer and their relationships are given. For elaboration on these components the reader is referred to the study of Lankhorst, Proper & Jonkers (2010) and to Appendix D.

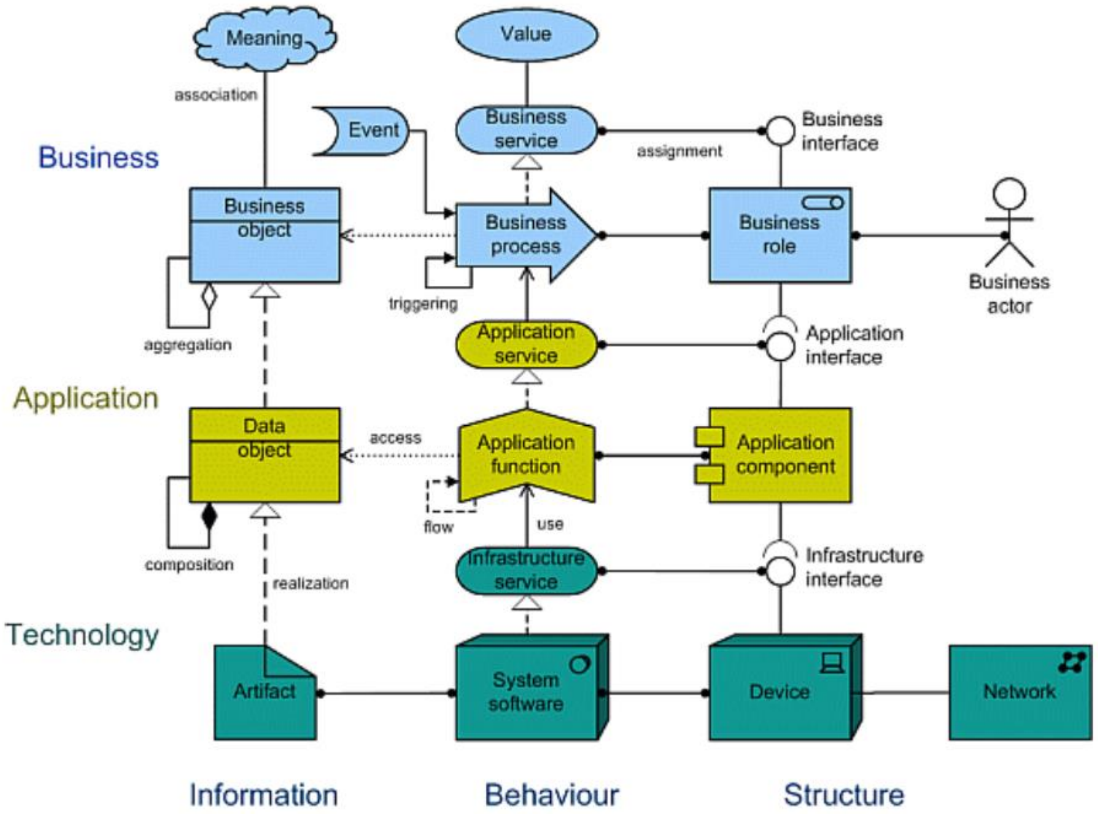


Figure 25 The most important concepts of ArchiMate (ArchiMate, 2018)

The layers of ArchiMate can be mapped to the BOAT-levels that are used for the realization level of Grefen's (2016) three-dimensional cube as given in Table 4.

The BOAT framework consists of Business, Organization, Architecture and Technology, as described in Section 2.3.3. Due to scoping, as discussed in Section 1.3.6, only the Organization and Architecture level on the realization dimension are relevant. Note that, the Business layer can be mapped to the B(usiness) and O(rganization) level on the realization dimension, the Application layer to the A(rchitecture) level, and the Technology layer to the T(echnology) level. Thus, the Technology layer is out of scope.

5.1.3.2. Design Choices

In order to use ArchiMate for the IS architecture in this study design choices were made. These choices are now described.

First, the Business layer was used for modeling the outsourcing processes. The outsourcing processes were modeled by using the 'Business Process'-component in ArchiMate. Second, the Application layer was used for modeling the underlying ISs. The underlying ISs were modeled in ArchiMate's 'Application Components'. These components were grouped into an 'Application Function' (Vermeulen, 2013).

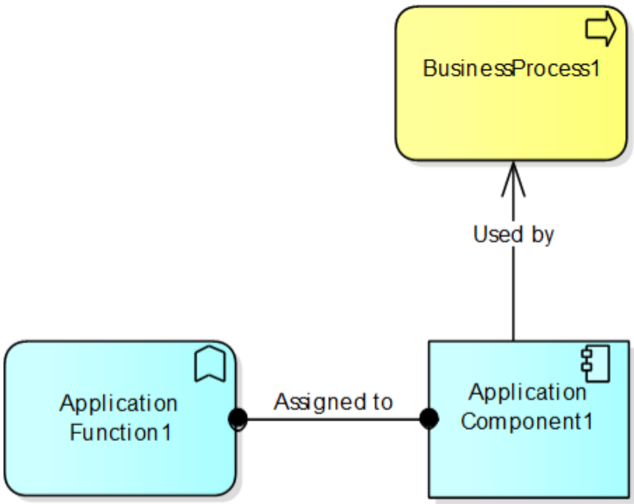


Figure 26 Relationships between concepts

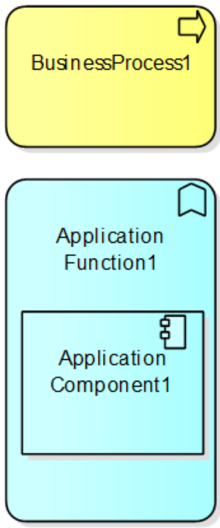


Figure 27 Relationships between concepts made implicit

In ArchiMate several relationships between concepts exist. It is now described what relationships exist for the chosen ArchiMate concepts. First, the 'Application Components' are used by a 'Business Process' (Vermeulen, 2013). Second, an 'Application Component' is assigned to an 'Application Function'. The relationships between the used concepts of ArchiMate are visualized in Figure 26.

In Figure 27 these relationships are implicit to make visualization of the concepts easier. In this way it can easily be seen which application components, used by which business processes, are assigned to which application function. The relationship can now be described as: The 'Business Process' uses the 'Application Components' that are assigned to an 'Application Function'. For modeling purposes Enterprise Architect (EA) was used. Remember, this tool is used by TC as a standard modeling tool and it supports the ArchiMate language notation.

It is now discussed how this visualization is integrated in the context of this study, using the design choices made. First, for the outsourcing process the 'Business Process'-notation is used. In the outsourcing process each phase of the outsourcing circle of Perunovic (2006) is included as a part of the whole outsourcing process. Second, each phase of the process is associated with an 'Application Function' that contains a similar name. Third, assigned to an 'Application Function' are the 'Application Components' that consist of the subsystems as described on aggregation level 2 (Section 5.1.1.1). These subsystems provide information for a certain phase of the outsourcing process, i.e. in terms of ArchiMate: these 'Application Components' are used by a certain 'Business Process'. These choices resulted in a format for a TO-BE model in ArchiMate. The format, with the MRP component as an example, is shown in Figure 28. This format is used to model the TO-BE situation, including its standard outsourcing process and underlying IS architecture. The reconsideration phase is colored grey since it is out of scope.

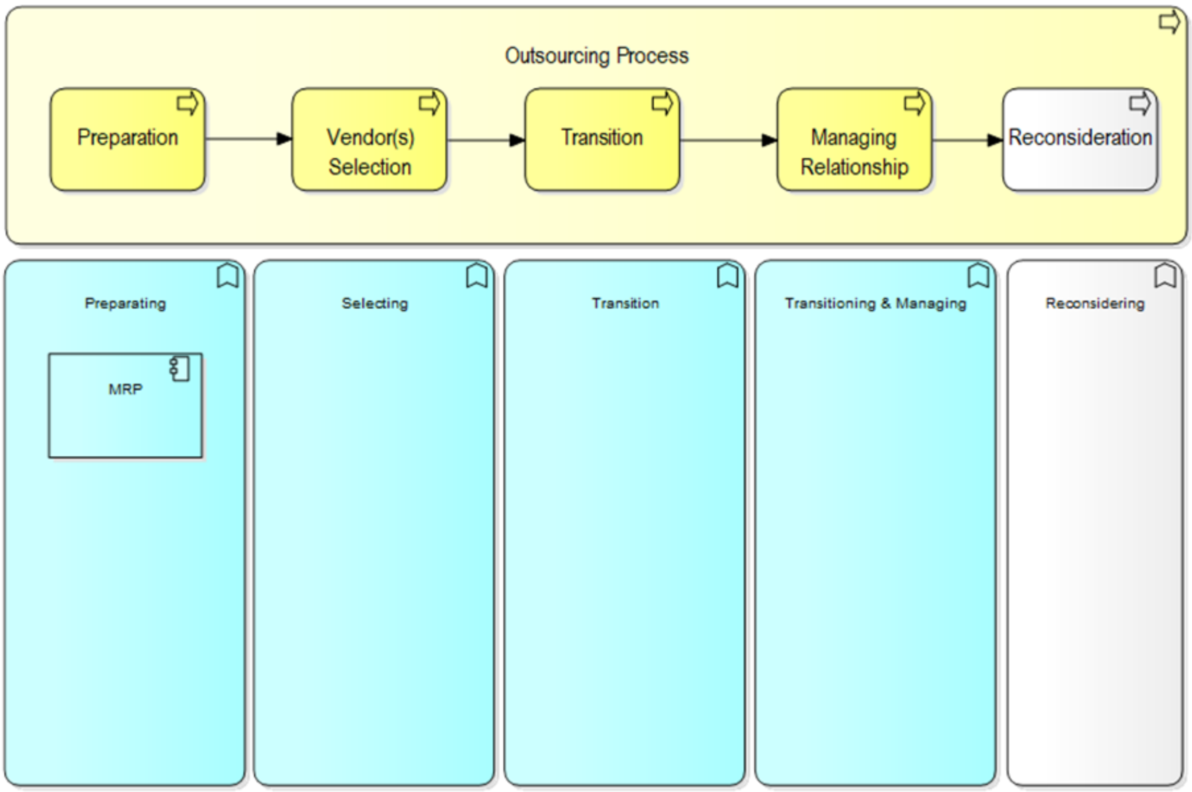


Figure 28 Format for modeling TO-BE in ArchiMate

5.1.4. Evaluation

In this section the method that was used for evaluation of the TO-BE model is described. This comprises the final phase of the problem solving cycle of Van Aken et al. (2012) for the case-specific situation, as described in Section 1.3.5. The evaluation was done on both relevance as well on rigor.

First, for the relevance an expert panel of TC was used consisting of the managers of both the logistics department of the manufacturing plant as well the manager of the corresponding IT domain. Here, the criteria of Shrivastava (1987) were used consisting of: (1) meaningfulness, (2) goal relevance, (3) operational validity, (4) innovativeness, and (5) cost of implementation. More information on these criteria can be found in Appendix E.

Second, for evaluation on rigor a paper-based evaluation took place. Here, the architecture design principles of Greefhorst & Proper (2011) were used. They define an architecture principle as 'a declarative statement that normatively prescribes a property of the design of an artifact, which is necessary to ensure that the artifact meets its essential requirements' (Greefhorst & Proper, 2011). These architecture design principles were useful as they mainly assess the product-oriented face of architectures, corresponding to the solution design of this study, as described in Section 1.3.1. Each principle is driven by the following quality attributes: functionality, reliability, usability, efficiency, maintainability, and portability (Greefhorst & Proper, 2011). In the evaluation it was given what quality attributes are affected by the principles that the model does or does not adhere to. Elaboration on these attributes is given in Appendix F.

5.2. Results

In this part the results are discussed. The structure is similar to the results described in Section 3.2. For the TO-BE situation first, the standard outsourcing process was defined. Second, a future IS architecture underlying this standard outsourcing process was designed. Hence, the TO-BE model visualizes both the processes as well the ISs supporting them.

5.2.1. TO-BE Standard Outsourcing Process

For the TO-BE standard outsourcing process the outsourcing circle of Perunovic (2006) forms the foundation. This cycle was discussed in Section 2.2.1. In the TO-BE situation each phase should follow this cycle, with its corresponding decisions. The phases and decisions were also given in Section 2.2.1.

As a result, one standard is developed for all material planners within the logistics department. In this way outsourcing activities become interchangeable between planners, and the number of individual errors decrease.

5.2.2. TO-BE IS Architecture

In the Process Analysis in Table 12 it was determined what information is relevant for outsourcing in the TO-BE framework. Based on this table and following the described method the TO-BE architecture design was developed.

The resulting TO-BE architecture design is given in Figure 29. Note that the reconsideration phase is colored grey, because it is out of scope. In Appendix G it is given what information is retrieved from what subsystems and in Appendix H a description for each subsystem is given. It is now described how the TO-BE architecture is structured.

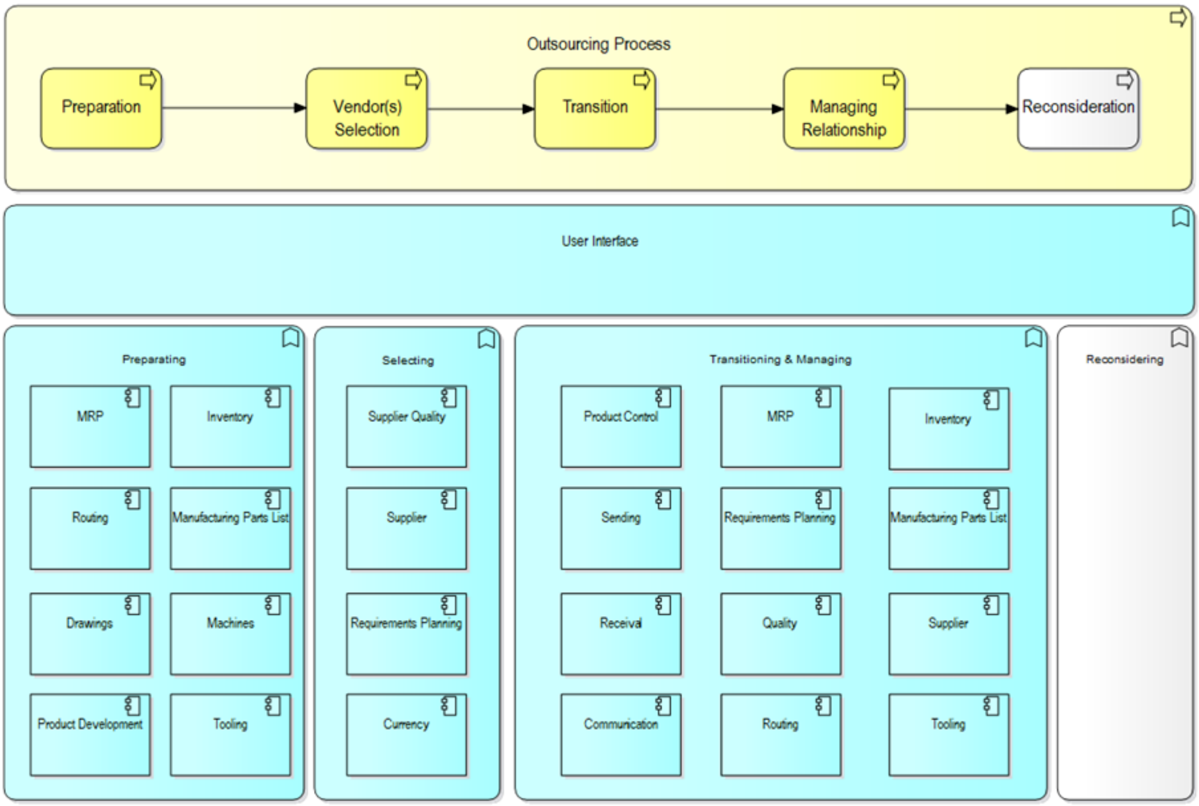


Figure 29 TO-BE IS architecture design

First, the TO-BE architecture design is structured by a clear columned style, based on functionalities. Due to the columned style it is clear what subsystems are required for decisions to be made in each phase of the outsourcing process. Thus, the function of subsystems in each column is to support the process above.

Second, the TO-BE architecture design has a clear layered style. Through the use of ArchiMate the outsourcing process (business layer) and underlying subsystems (application layer) are distinguished in layers by nature. Further, the application Layer is divided into two layers: the user interface layer and the information layer. The user interface layer connects the user of the processes with what is used: the information layer. In other words, the user interface layer connects each component in the information layer to the users of the processes that have to make a decision within the outsourcing process. These connections are hidden to keep the figure readable.

Third, the components in the TO-BE architecture design are connected by a shared database, which is one of the patterns of Grefen (2016) that are described in Appendix B. A shared database has flexible asynchronous coupling between the components and requires availability of transaction management (Grefen, 2016). This requirement is satisfied since the ERP system is a transactional system. In the TO-BE model the ERP system serves as a shared database, since the required information is retrieved from the subsystems in the ERP system by multiple phases in the process, through an user interface.

Implementing the TO-BE IS architecture solves the problems defined in the AS-IS situation as it replaces or defines a system or task in the main ERP system. In this way the self-built plant-specific applications become redundant, and thus occurrence of manual interfaces disappears, resulting in less error-proneness, and less need for safety buffers. Also the supporting applications become redundant, since the communication function is captured within the TO-BE model, resulting in less error-proneness as communication is no longer done manually. The replacement by the TO-BE model is abstractly visualized in Figure 30.

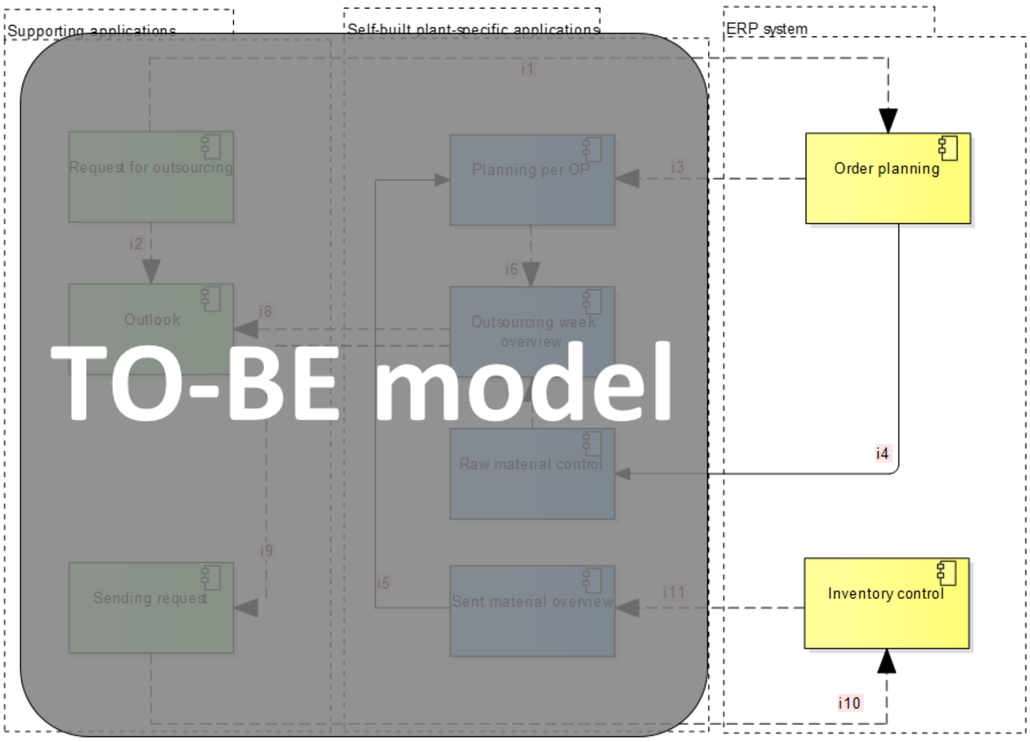


Figure 30 The TO-BE model replacing the AS-IS model

5.3. Evaluation

The TO-BE model was evaluated on both relevance as well rigor as described in the method (Section 5.1.4).

5.3.1. Evaluation on Relevance

Evaluation on relevance took place by using the criteria of Shrivastava (1987). The criteria and details are given in Appendix I.

Based on an expert panel the TO-BE model was considered as relevant, insightful, and understandable to TC. This captured the meaningfulness and goal relevance criteria of Shrivastava (1987). Since the model is highly conceptual, for operational validity it needs more specification, especially for the user interface.

Further the model was considered as innovative for TC since it replaces the self-built plant-specific applications. The last criterion, capturing costs of implementation, was not clear.

In general the model was perceived as positive, especially the standard defined in the outsourcing process and the replacement of self-built plant-specific applications. Also a preliminary implementation of the solution design, executed as a first step to the standard, was received positively by TC.

5.3.2. Evaluation on Rigor

Evaluation on rigor took place by using the architecture design principles of Greefhorst & Proper (2011) as described in the method. A detailed evaluation is given in Appendix J, where it is indicated for each design principle whether the TO-BE model adheres to it. The main results are now described, additionally indicating what quality attribute (reliability, functionality, usability, efficiency, maintainability, or portability) is affected by adhering or not adhering to a design principle (Greefhorst & Proper, 2011). These quality attributes are described in Appendix F. First, the principles where the TO-BE model adheres to, are described. Then, the principles that the TO-BE model does not adhere to, are described.

First, as a result of the standard outsourcing process proposed in the TO-BE model, design principles 4 and 5 are adhered to: ‘processes are straight through’ and ‘processes are standardized’, increasing each quality attribute (as given in Appendix F), except functionality. Second, due to this standard, for each decision in each phase of the outsourcing process an outcome is required, adhering to design principle 7: ‘tasks are designed around outcome’, increasing the reliability, usability and efficiency. Third, most frequently occurring manual tasks are replaced by the TO-BE model, satisfying design principle 8: ‘routine tasks are automized’, improving the reliability and efficiency. Fourth, due to the TO-BE model less safety buffers are required to anticipate on incidents, e.g. design principle 3: ‘Stock is kept at a minimum’. Fifth, by providing data via user interfaces, the TO-BE model adheres to design principles 18 and 25, increasing the usability and maintainability: ‘content and presentation are separated’ and ‘applications have a common look-and-feel’. Sixth, the TO-BE model is adhered to design principles 35 and 36: ‘components have a clear owner’ and ‘IT systems are standardized and reused throughout the organization’. It replaces the self-built plant-specific applications that supports outsourcing since it is centrally supported for each plant in the organization by the IT department. In this way the maintainability and reliability increase. Finally, the TO-BE model adheres to principle 33: ‘IT systems are scalable’, increasing the efficiency of the model and ability to manage more frequent outsourcing.

Now, the design principles the TO-BE model did not adhere to are described. First, it is not adhered to design principle 16: ‘Data are captured once’. In the TO-BE model subsystems exist multiple times, e.g. MRP, decreasing the efficiency of the model. Second, it is not adhered to design principle 21: ‘data are exchanged in real-time’. The main ERP system is integrated in the TO-BE model and does not provide real-time data, which is a legacy problem, and is decreasing the usability and efficiency. Third, due to a columned style the applications are not modular, and thus the model does not adhere to design principles 9 and 28: ‘Primary business processes are not disturbed by implementation of changes’ and ‘applications are modular’, decreasing the reliability, maintainability, and portability. Fourth, there is no direct communication possible between the outsourcing partner and the customer, and thus the model does not adhere to design principle 13: ‘the status of customer requests is readily available inside and outside the organization’, decreasing the usability. Finally, the model does not adhere to design principle 41: ‘business processes are supported by a business process management system’, which is a possible extension in the general model.

5.4. Conclusion

Based on the TO-BE model and the evaluation of the model conclusions can be drawn.

First, TC regained insights in its outsourcing process by adapting to a standard outsourcing process, defined by Perunovic (2006). In this way outsourcing activities become interchangeable between planners of the logistics department, and it is known what information is required for what decision in the outsourcing process.

Second, the TO-BE model replaces the shadow IT, including its manual interfaces. This was one of the most significant potential threats for TC, when outsourcing occurs more frequently and becomes more complex. Especially, due to its scalability the TO-BE model is able to manage the increase in outsourcing as it eliminates the manual interfaces, decreases the error-proneness, and reduces the need for safety buffers.

Based on an evaluation, especially the efficiency and the usability were increased by the TO-BE model. However, there is still room for improvement, as the evaluation indicated. Therefore, the design principles that are not adhered to (mainly due to a legacy situation), are taken into account for the next phase: creating a reference architecture.

6. Reference Architecture

In this section the last phase is discussed: Phase 4. It contains the general reference architecture, answering the fourth research question (see Figure 5). It comprises the solution design phase of the problem solving cycle of Van Aken et al. (2012), as described in Section 1.3.5. First, the method is described. Second, the results are discussed, followed by conclusions.

6.1. Method

In the final phase the reference architecture was developed. Based on the TO-BE model which is TC-specific and the Process Analysis, additional subsystems were included. Further the TO-BE model is improved since it is not limited to the legacy-situation existing at TC's manufacturing plant. First, the levels of each dimension of Grefen's cube (2016) were set. Second, the steps that are taken to abstract the TO-BE model to a reference architecture are described.

6.1.1. Grefen's Three-dimensional Cube

In order to create the reference architecture, again the three-dimensional cube of Grefen (2016) was used. The cube was used to transform the TO-BE model to a reference architecture. For each dimension of the cube, as described in Section 2.3, the levels were set.

6.1.1.1. Aggregation

First, the aggregation dimension was determined. Similar to the TO-BE model (see Section 5.1.1.1), this dimension was set on level 2. This level consists for the processes of the phases of Perunovic (2006) as shown in Figure 21: preparation, vendor(s) selection, transition, managing the relationship, and reconsideration. For the architecture this level uses subsystems, like the TO-BE model, but unlike the AS-IS model that consists of detailed tasks executed by these subsystems.

6.1.1.2. Abstraction

Second, the abstraction dimension was set. Unlike the AS-IS and the TO-BE model this level was set on level 1, as defined in Section 2.3.2. The architecture components are described in terms of general software system classes, indicating their functionality. In this way the model transforms one level up in the abstraction dimension.

6.1.1.3. Realization

Third, the realization level was determined, based on scoping in Section 1.3.6. Similar to the AS-IS and TO-BE model the processes and architecture are modeled. Hence, the O-level and the A-level were used respectively, as explained in Section 3.1.1.3.

6.1.1.4. Aspects

Fourth, the aspect level was chosen. Similar to the TO-BE model the process aspect was used for the processes (O-level) and the software aspect was used for the architecture (A-level).

6.1.1.5. Visualization

The positioning of the model in the cube is visualized in Figure 31. It can be seen how the reference architecture (RA) diverges from the AS-IS and TO-BE model. In Table 14 the levels of each dimension are summarized.

Table 14 Levels of dimensions used for the reference Architecture

Dimension	**Processes**	**Architecture**
Aggregation	Level 2: phases of Perunovic (2006)	Level 2: each subsystem described
Abstraction	Level 1: general terms, indicating functionality	Level 1: general terms, indicating functionality
Realization	Level 2: Organization	Level 3: Architecture
Aspect	Process	Software

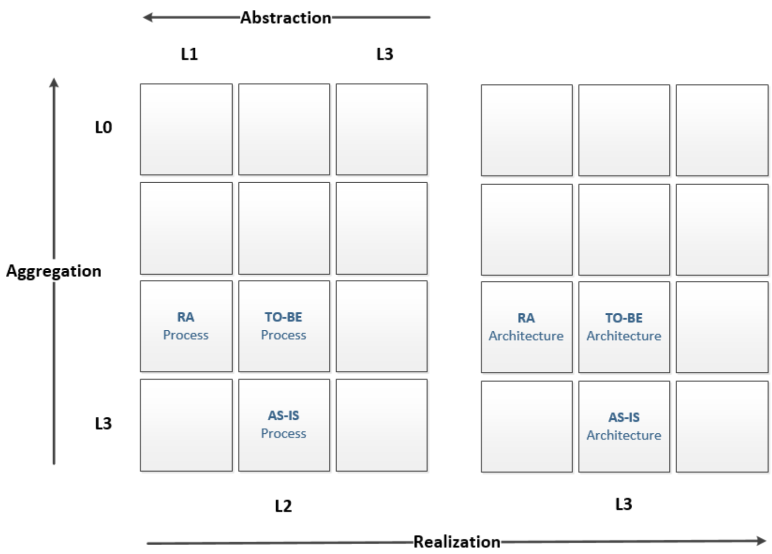


Figure 31 Dimensions of reference architecture

6.1.2. Designing a Reference Architecture

As mentioned before, the objective of this phase is to design a reference architecture for outsourcing. A reference architecture is 'a general design (abstract blueprint) of a structure for a specific class of information systems' (Grefen, 2016). Reference architectures often focus on the software aspect (as defined in Section 2.3.4), but may include elements of other aspects. Like the TO-BE model, the proposed reference architecture also includes the process aspect (as defined in Section 2.3.4).

According to Grefen (2016) reference architectures can be descriptive or prescriptive. A descriptive reference architecture describes a standard in a specific context based on the existing state of the art of that context. An example is a reference architecture that is built from best practice cases. A descriptive reference architecture can be used in analysis of concrete architectures or as inspiration for the design of new architectures. A prescriptive reference architecture describes a standard that is obligatory to fit into a specific context. In this thesis the reference architecture is based on the TO-BE situation for TC. Hence, it is classified as a descriptive reference architecture.

The reference architecture was developed by expanding and improving the TO-BE model. Where the TO-BE model is limited due to a legacy situation, the reference architecture is able to eliminate the limitations of the legacy situation. This was done by considering two steps, which are now described.

6.1.2.1. Step 1: Expanding the TO-BE Model

In the first step, subsystems were added to the model. This was based on the Process Analysis in Chapter 4 in which information was identified that is required during the outsourcing process. This is shown in Table 12. Based on TC's manufacturing environment, certain information was not required in the TO-BE model. However, the reference architecture should apply to the other manufacturing environments as well. In order to provide the additional required information, additional subsystems were included in the reference model, or subsystems were expanded.

6.1.2.2. Step 2: Integrating Architecture Principles

In the second step, the model was improved by integrating architecture principles. This is based on the evaluation on rigor as described in Section 5.3.2, which makes use of the architecture design principles catalog of Greefhorst & Proper (2011). Important drivers for these architecture principles to improve the TO-BE model are the following quality attributes: functionality, reliability, usability, efficiency, maintainability, and portability (Greefhorst & Proper, 2011). Elaboration on these attributes is given in Appendix F.

6.2. Results

Following the method described in the previous section, this section discusses the results.

6.2.1. Step 1: Expanding the TO-BE model

First, the TO-BE model was expanded by taking into consideration the Process Analysis in Chapter 4. For the identified additional information required it is given in Appendix K from what subsystems this information can be retrieved or how a subsystem should be expanded. Also a description of these subsystems is given in Appendix K.

The most signification addition was the order management system that comprises information on the specific customer orders. Contrary to the TO-BE situation of TC which is not customer specific, this information is required for other manufacturing environments other than TC's manufacturing plant to satisfy customer's specific needs.

6.2.2. Step 2: Integrating Architecture Principles

Next, architecture design principles were applied to give recommendations to improve the model. Besides, it is given what quality criteria, as described in Appendix F, were improved. The results are now described.

First, the TO-BE model does not adhere to design principle 21: 'data are exchanged in real-time', due to a legacy situation. In contradiction, the reference architecture should be able to exchange data in real time to increase the efficiency and usability.

Second, in the TO-BE model an interface exists for each application component, which implies the model does not adhere to design principle 16: 'data are captured once'. Some interfaces (e.g. for the inventory component) are existing twice (e.g. for both the preparation function and the transition & managing relationship function). Therefore, subsystems that are existing twice can be eliminated once, increasing the efficiency. These subsystems are: 'MRP', 'Routing', 'Inventory', 'Manufacturing Parts List', 'Supplier', 'Tooling', and 'Requirements Planning'. This can be done through the development of an enterprise service bus which allows direct flexible synchronous coupling (Grefen, 2016), adhering to design principle 58: 'all messages are exchanged through the enterprise service bus'. In this way less interfaces are required, improving the maintainability and portability of the architecture. An enterprise service bus is a communication broker that connects services (Grefen, 2016). It is one of the patterns described by Grefen (2016) and given in Appendix B. In order to achieve these benefits the reference architecture includes an enterprise service bus that is designated as communication system. However, this is not specified since the technology level of the BOAT-framework is out of scope. The communication system automatically replaces the 'Sending', 'Receival', and 'Communication' subsystems of the TO-BE model. An extension to this in the reference architecture is the use of an user interface that serves both internally within the company and externally to the outsourcing partner. In this way the reference architecture allows that e.g. change in production requirements are directly communicated to the outsourcing partner via a portal. This results in that the reference architecture also adheres to design principle 13: 'the status of customer requests is readily available inside and outside the organization'.

Third, a business process management system is added to the architecture, adhering to design principle 41: 'processes are supported by a business process management system', and improving the efficiency and maintainability. Business process management systems are 'generic software systems that are driven by explicit process representations to coordinate the enactment of business processes' (Weske, 2012).

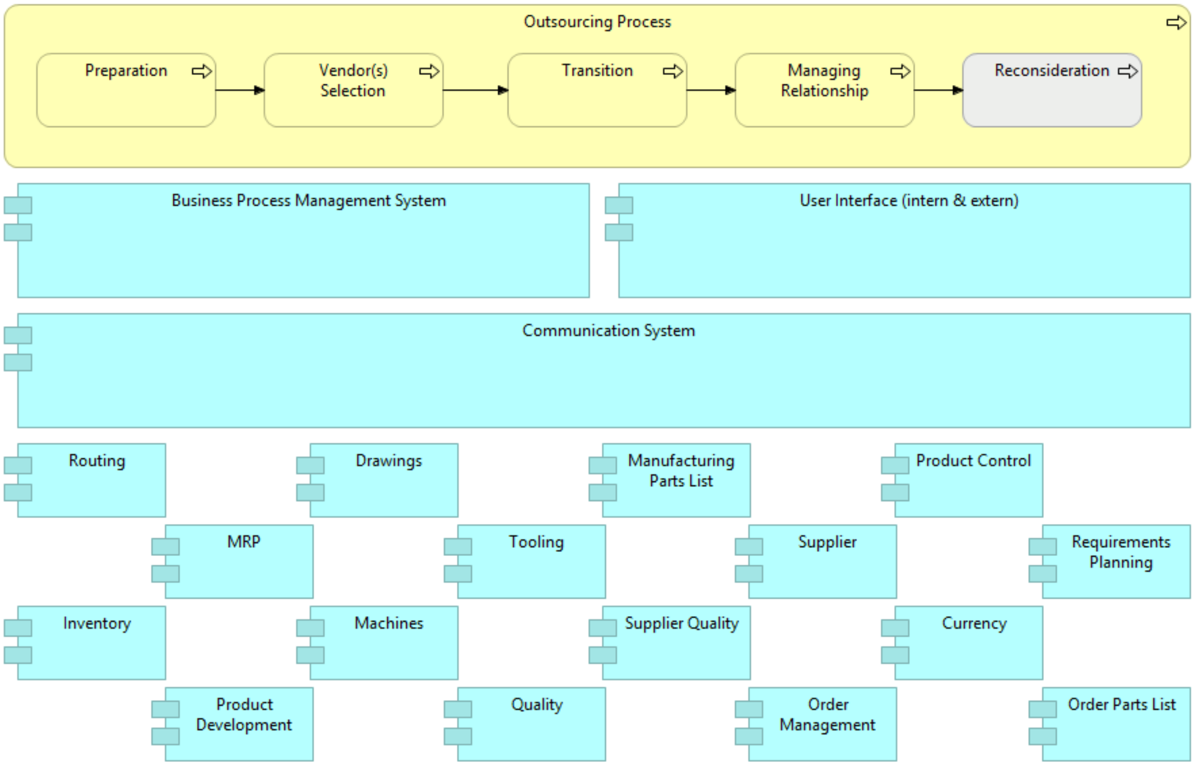


Figure 32 Reference architecture design

Similar to the TO-BE model, the reference architecture, as shown in Figure 32, has a clear layered structure, based on its function. First, the upper layer comprises the process layer. In this layer the outsourcing process is defined, as given by Perunovic (2006). As the reconsideration phase is out of scope, it is colored gray. Second, the next layer is the user interface layer. In this layer the user interface and business process management system are positioned. Third, the communication system belongs to the communication layer, comprising the shared bus. Finally, the fourth layer is the information layer, existing of the subsystems that provide the required information. Since these subsystems are modular, the reference architecture also adheres to design principles 9 and 28: 'primary business processes are not disturbed by implementation of changes' and 'applications are modular', increasing the reliability, maintainability, and portability of the architecture.

6.3. Conclusion

Based on the results conclusions can be drawn, including the most important extensions and improvements of generalizing the TO-BE model.

First, in contradiction to the TO-BE model that contains a shared database, the shared bus (communication system) proposed in the reference architecture allows direct flexible synchronous coupling. This implies that change of information in one subsystem is directly modified and updated in the other subsystems. In terms of outsourcing, this is required as manufacturing schedules are constantly subject to changes, e.g. a machine crash might trigger the need for a fast outsourcing decision, as described in Section 3.2.1.

Second, the business process management system allows to indicate what activities should be executed when the customer order decoupling point is positioned differently. As shown in the Process Analysis (Chapter 4) different information is required at certain stages during the outsourcing process. The business process management system is able to monitor the processes, resulting in a workload reduction, flexible integration, and transparency in execution of tasks in the processes (Dumas, La Rosa, Mendling, & Reijers, 2013).

7. Discussion

This chapter consists of the following sections: conclusions and recommendations, reflection on the project, limitations of the solution design, and finally suggestions for future research.

7.1. Conclusions and Recommendations

This study proposed an IS architecture, underlying a standard outsourcing process, for both TC as well high-tech manufacturing companies in general. These architectures were designed by determining what information from what systems is required to support decisions during the outsourcing process. In this section main conclusions are drawn and corresponding recommendations are given.

First, this study proposed a standard outsourcing process, based on the outsourcing circle of Perunovic (2006). It is recommended for TC to adapt its processes to this standard, as a first step to become future proof, in order to manage the increase in complexity and frequency of outsourcing. For both the tactical as well the operational level, standardization requires guidelines for communication and tasks to be executed. In this way TC ensures that each department involved in the process becomes aware of the standard process. Implementing the standard results in the ability to interchange outsourcing activities between material planners and results in a decrease of error-proneness.

Second, the TO-BE model comprises an IS architecture. In this architecture ISs are structured to efficiently support each decision in the outsourcing process. TC can use the architecture as a blueprint to implement decision support. For each decision to be made in the outsourcing process the required information and subsystems were defined. In this way, the shadow IT with manual interfaces, becomes redundant. Due to its scalability, the TO-BE model is able to manage the increase in frequency of outsourcing and the increase in complexity. In order to implement the TO-BE model it is further recommended for TC to specify the user interfaces in the legacy system, in order to increase the operational validity. Based on the evaluation on relevance it is also recommended to conduct research on the costs of implementation of the TO-BE model.

Third, the reference architecture is an improved and extended version of the TO-BE model, since it is not restricted to TC's legacy situation. As it contains a shared bus pattern it allows direct flexible synchronous coupling, updating data directly in each involved system. In this way both the firm as well the outsourcing partner keep insights in the outsourcing process, which is a significant factor as planning of production is constantly subject to changes. Besides, a business process management system allows to monitor execution of tasks. In terms of outsourcing, insights in status of the outsourcing and different kinds of outsourcing constructions, for both manufacturer as well outsourcing partner can be achieved.

7.2. Reflection

A few words on the progress during this study are given in this section.

First, in the problem definition phase, the problem was presented by TC as a broad unstructured question, resulting in challenges when setting the scope. This was solved by taking an academic point of view, based on what was concluded from the literature review: the need for an IS architecture underlying outsourcing processes.

Second, since there were multiple stakeholders involved in the project (both IT department as well logistics department), objectives needed to be aligned. Where the logistics department advocated for fast short term solutions, the IT department prioritized durability of a proposed solution. Therefore, during the project, I was present at both departments a number of days per week. Both were satisfied by implementing a segment of the standard outsourcing process and proposing the TO-BE model.

7.3. Limitations

In this section limitations of the project are discussed.

First, in order to find practical references in the process analysis this study depended heavily on possibilities of the company and the limited time during the project. In this way, the analysis framework might not be complete and is subject to expansion. In order to design an encompassing reference architecture more case studies are required to be analyzed. Also evaluation of the reference architecture was out of scope, leaving room for improvement. Note that evaluation of the reference architecture requires much more analysis, which was not feasible during this study.

Second, due to the legacy situation, some design principles by Greefhorst & Proper (2011) could not be adhered to. Thereby, expert evaluation was only done inside the company, so may be biased.

Third, this study was limited to the O(rganization) and A(rchitecture) level of the BOAT-framework, resulting in its conceptual character. As the B(usiness) level was out of scope, business goals may be missing, describing what should be reached by the models. Especially, as the T(echnology) level was out of scope, specification on the architecture design is required before implementation is possible. Note that, the proposed architecture is product-oriented and thus does not prescribe the process for realization of the model.

7.4. Future Research

For future research several suggestions were done.

First, evaluation of the reference architecture might be subject for future research. In this way the architecture can be validated.

Second, this study made use of a bottom-up method for designing the reference architecture: from an existing situation the reference architecture is abstracted. A future research direction is to create the reference architecture by using a top-down method: starting from a greenfield situation, and gradually add more detail (Grefen, 2016). A method that can be used and is considered as a standard in architecture design methods is TOGAF (Grefen, 2016).

Third, also for the process analysis another method might be insightful. As there was no method defined for the analysis in literature, an alternative academic methodology could be designed.

8. References

Alvesson, M., & Deetz, S. (2000). *Doing Critical Management Research*. Los Angeles: Sage.

APICS. (2011). *APICS: Operations Management Body of Knowledge Framework* (3rd ed.). Chicago: Apics. Retrieved from http://www.apics.org/sites/apics-supply-chain-council/research-and-publications/ombok

Araz, C., Ozfirat, M., & Ozkarahan, I. (2007). An integrated multicriteria decision-making methodology for outsourcing management. *Computers & Operations Research*, *34*, 3738–3756. https://doi.org/10.1016/j.cor.2006.01.014

ArchiMate. (2018). What is ArchiMate? Retrieved January 31, 2018, from http://www.archimate.nl/en/about_archimate/what_is_archimate.html

Bui, S. H., Muralikrishnan, B., & Raja, J. (2005). A framework for Internet-based surface texture analysis and information system. *Precision Engineering*, *29*(3), 298–306. https://doi.org/10.1016/j.precisioneng.2004.11.006

Camunda.org. (2017). BPMN 2.0 Tutorial. Retrieved from https://camunda.org/bpmn/tutorial/

Chinosi, M., & Trombetta, A. (2012). BPMN: An introduction to the standard. *Computer Standards and Interfaces*, *34*(1), 124–134. https://doi.org/10.1016/j.csi.2011.06.002

Click, R. L., & Duening, T. N. (2005). *Business Process Outsourcing: The Competitive Advantage*. Hoboken: John Wiley & Sons.

Corbett, M. F. (2004). *The Outsourcing Revolution: Why it makes sense and how to do it right*. Chicago: Dearborn Trade Publishing.

Cullen, S., & Willcocks, L. (2003). *Intelligent IT Outsourcing: Eight Building Blocks to Success*. Oxford: Butterworth-Heinemann.

Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2013). *Fundamentals of Business Process Management*. Berlin: Springer-Verlag.

Emmet, S., & Crocker, B. (2016). *The Relationship-Driven Supply Chain: creating a culture of collaboration throughout the chain*. CRC Press. Retrieved from https://books.google.nl/books?hl=nl&lr=&id=8PifCwAAQBAJ&oi=fnd&pg=PP1&dq=the+relationship-driven+supply+chain&ots=huooPO0SXe&sig=w2mFPN5AvBhexEzBYM_uSO539e4#v=onepage&q=the relationship-driven supply chain&f=false

Franceschini, F., Galetto, M., Pignatelli, A., & Varetto, M. (2003). Outsourcing: guidelines for a structured approach. *Benchmarking: An International Journal*, *10*(3), 246–260. https://doi.org/10.1108/14635770310477771

Fylan, F. (2005). *Semi-Structured Interviewing. A handbook of research methods for clinical and health psychology*.

Greaver, M. F. (1999). *Strategic outsourcing: a structured approach to outsourcing decision and initiatives*. New York: AMACOM.

Greefhorst, D., & Proper, E. (2011). *Architecture Principles The Cornerstones of Enterprise Architecture*. *Architecture Principles The Cornerstones of Enterprise Architecture* (Vol. 6). Berlin: Springer-Verlag. https://doi.org/10.1007/978-3-642-20279-7

Grefen, P. (2016). *Business Information System Architecture* (Fall 2016). Eindhoven: Eindhoven

University of Technology.

Gunasekaran, A., & Ngai, E. W. T. (2007). Knowledge management in 21st century manufacturing. *International Journal of Production Research*, *45*(11), 2391–2418. https://doi.org/10.1080/00207540601020429

Harris, A., Giunipero, L. C., & Hult, G. T. M. (1998). Impact of organizational and contract flexibility on outsourcing contracts. *Industrial Marketing Management*, *27*(5), 373–384.

Hentza, J. B., Nguyen, V. K., Maederb, W., Panaresec, D., Gunninkd, J. W., Gontarze, A., … Hascoëth, J. Y. (2013). An enabling digital foundation towards smart machining. In *Procedia CIRP* (Vol. 12, pp. 240–245). Elsevier B.V. https://doi.org/10.1016/j.procir.2013.09.042

Herrmann, S., Rogers, H., Gebhard, M., & Hartmann, E. (2015). Co-creating value in the automotive supply chain: an RFID application for processing finished vehicles. *Production Planning & Control*, *26*(12), 981–993. https://doi.org/10.1080/09537287.2014.1002023

Khoei, M. R., Kashefi, M., Ghadimi, P., Azadnia, A. H., Rani, M. R. A., & Lalmazloumian, M. (2011). Order Processing in Supply Chain Management with Developing an Information System Model: An Automotive Manufacturing Case Study. *Informatics Engineering and Information Science, Pt Iv*, *254*, 65–76. https://doi.org/10.1007/978-3-642-25483-3_6

Kitchenham, B. (2006). *Performing systematic literature reviews in software engineering*. *Proceeding of the 28th international conference on Software engineering - ICSE '06* (Vol. 45). Keele. https://doi.org/10.1145/1134285.1134500

Kotha, S. (1995). Mass Customization: implementing the emerging paradigm for competitive advantage. *Strategic Management Journal*, *16*, 21–42.

Kvale, S., & Brinkmann, S. (2008). *Interviews: Learning the craft of qualitative research interviewing*. Los Angeles: Sage.

Lankhorst, M. M., Proper, H. A., & Jonkers, H. (2010). The anatomy of the ArchiMate language. *International Journal of Information System Modeling and Design*, *1*(1), 1–32. https://doi.org/10.1007/978-3-642-01862-6_30

McIvor, R. (2005). *The Outsourcing Process: Strategies for Evaluation and Management*. Cambridge: Cambridge University Press.

Momme, J. (2001). *Outsourcing Manufacturing to Suppliers*. Aalborg University.

Momme, J., & Hvolby, H. H. (2002). An outsourcing framework: Action research in the heavy industry sector. *European Journal of Purchasing and Supply Management*, *8*(4), 185–196. https://doi.org/10.1016/S0969-7012(02)00003-5

Muehlen, M. Zur, & Recker, J. (2008). How much language is enough? Theoretical and practical use of the business process modeling notation. *Advanced Information Systems Engineering - Proceedings 20th International Conference on Advanced Information Systems Engineering*, 465–479. https://doi.org/10.1007/978-3-540-69534-9_35

Oracle. (2014). *Driving Outsourced Manufacturing Best Practices with Oracle*.

Perunović, Z. (2006). Utilisation of Information and Communication Technology across the Outsourcing Process - The Vendor ' s Perspective. In *Proceedings of the 15th international conference of management of technology IAMOT '06*. Beijing. Retrieved from http://orbit.dtu.dk/fedora/objects/orbit:82664/datastreams/file_5053207/content

Perunović, Z. (2007). Outsourcing Process and Theories. *POMS 18th Annual Conference*, *8*(5), 35.

Retrieved from http://www.poms.org/conferences/poms2007/cdprogram/topics/full_length_papers_files/007-0003.pdf

Porter, M. E. (1985). Competitive Advantage: Creating and Sustaining Superior Performance. *Competitive Advantage: Creating and Sustaining Superior Performance*. New York: The Free Press. https://doi.org/10.1182/blood-2005-11-4354

Qu, S. Q., & Dumay, J. (2011). *The qualitative research interview*. *Qualitative Research in Accounting & Management* (Vol. 8). https://doi.org/10.1108/11766091111162070

Robert Jacobs, F., & Chase, R. B. (2017). *Operations and Supply Chain Management* (15th ed.). New York: McGraw-Hill.

Robson, C., Watanabe, Y., & Numao, M. (2007). Parts traceability for manufacturers. In *Proceedings - International Conference on Data Engineering* (pp. 1212–1221). https://doi.org/10.1109/ICDE.2007.368980

Sanchez, R. (1997). Preparing for an uncertain future: managing organizations for strategic flexibility. *International Studies of Management and Organization*, *27*(2), 71–94.

Shrivastava, P. (1987). Rigor and Practical Usefulness of Research in Strategic. *Strategic Management Journal*, *8*(1), 77–92.

Tóth, L., Döbrössy, P., & Mánik, D. (2006). RFId application in production process of an automotive industry supplier. In *INES 2006: 10th International Conference on Intelligent Engineering Systems 2006* (pp. 45–48). https://doi.org/10.1109/INES.2006.1689339

Truijens, J., Oosterhaven, A., Maes, R., Jägers, H., & Van Iersel, F. (1990). *Informatie-infrastructuur: een Instrument voor het Management*. Kluwer Bedrijfs-wetenschappen.

Tummers, A. (2017). *Strategic Business & IT Alignment through a Standard IS Architecture Design*. Eindhoven University of Technology.

Van Aken, J. E., Berends, H., & Van der Bij, H. (2012). *Problem Solving in Organizations: A Methodological Handbook for Business and Management Students* (2nd ed.). New York: Cambridge University Press.

Van Zeist, B., Hendriks, P., & Paulussen, R. (1996). *Kwaliteit van softwareprodukten: Praktijkervaringen met een kwaliteitsmodel*. Sdu.

Van Zutphen, A. (2017). *Structured Literature Review: Outsourcing production processes and its corresponding IS architecture in (truck) manufacturing*.

Vermeulen, R. J. M. (2013). *Design of an Enterprise Architecture for information requirements identification : An exploratory study within the high-tech industry*. Eindhoven University of Technology.

Wang, Y. (2008). Study on the Manufacturing Base Oriented Collaborative Serving System Development and Key Technology. In *Electronic Engineering* (pp. 1078–1082).

Wang, Y., Wu, Y., Liu, Y., & Tang, A. (2007). The application of Radio frequency identification technology on tires tracking. In *Proceedings of the IEEE International Conference on Automation and Logistics, ICAL 2007* (pp. 2927–2930). https://doi.org/10.1109/ICAL.2007.4339082

Weske, M. (2012). *Business Process Management* (2nd ed.). Berlin: Springer-Verlag. https://doi.org/10.1007/978-3-642-28616-2

Yang, Z., & Li, H. (2015). Study of Manufacturing Logistics Outsourcing Decision. *Proceedings of the 2015 International Conference on Management, Education, Information and Control*, *125*, 1181–1185.

Appendices

Appendix A: Semi-Structured Interviewing Format AS-IS Situation

In exploring the AS-IS situation semi-structured interviews were used, with the following questions:

1) Tactical Outsourcing

a) What operations are being outsourced?
b) How do you determine what to outsource?
c) What are the reasons to outsource?
d) What information do you need to decide what to outsource?
e) What systems/functionalities/applications do you use to decide?
f) What decisions are further taken?
g) How do you document data and where is it being saved?

2) Operational Outsourcing

a) How are products being ordered at outsourcing partner?
b) What functionalities do you use of what applications?
c) How are products being sent to the outsourcing partner?
d) How do you decide what to send?
e) What happens when processed products are received?
f) What information do you need to manage the outsourcing?

Appendix B: Styles and Patterns of IS Architectures by Grefen (2016)

In this Appendix styles and patterns that are used for the product-oriented face of architectures are described.

B.1 Styles

Grefen (2016) defined an architecture style as 'a generally recognized structure class describing the overall structure of an architecture at a high level of abstraction (and indirectly the process of architecting)'. Hence, an architecture style primarily defines how the main structure of architectures appears. Architecture styles can be applied to all architecture aspects (see Section 2.3.4). However, they are most commonly used to structure the software aspect (as defined in Table 5).

Grefen (2016) formed a simple catalogue for these architecture styles, by distinguishing four basic styles:

1) Monolithic: 'the monolithic style uses a black-box approach: all functionality is included in one monolithic and hence there is a complete absence of explicit structure'.
2) Layered: 'the layered style defines structure by organizing functionality into several layers of functional abstraction'.
3) Columned: 'the columned style defines structure by organizing functionality into several functional sub-areas at the same level of functional abstraction'.
4) Component-Oriented: 'the component-oriented style defines structure by grouping coherent application functionality into components with explicit interfaces'.

These styles are visualized in Figure 33.

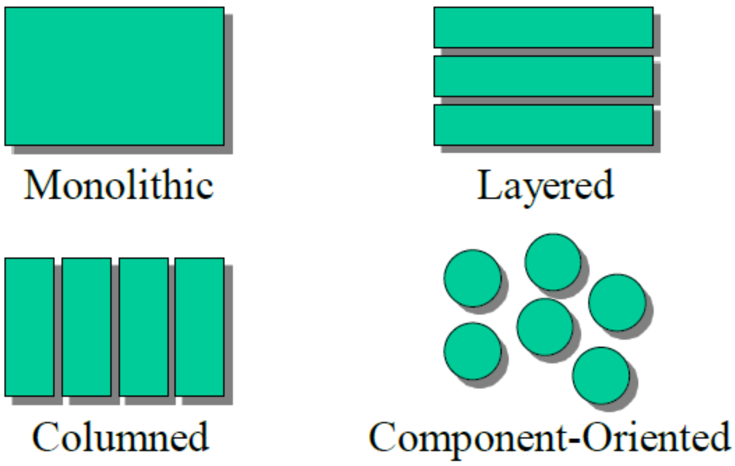


Figure 33 Basic IS architecture styles (Grefen, 2016)

B.2. Patterns

Grefen (2016) defined an architecture pattern as 'a generally recognized recurring (sub)structure that is used to describe part of the overall structure of an architecture'.

Grefen (2016) formed a simple catalogue for architecture patterns, by distinguishing four basic patterns to connect architecture components, which are now briefly described and visualized in Figure 34:

1) Direct invocation: 'remote procedure invocation between modules'.
2) File transfer: 'file transfer between modules'.
3) Shared database: 'data transfer via shared database'.
4) Shared bus: 'data transfer via bus'.

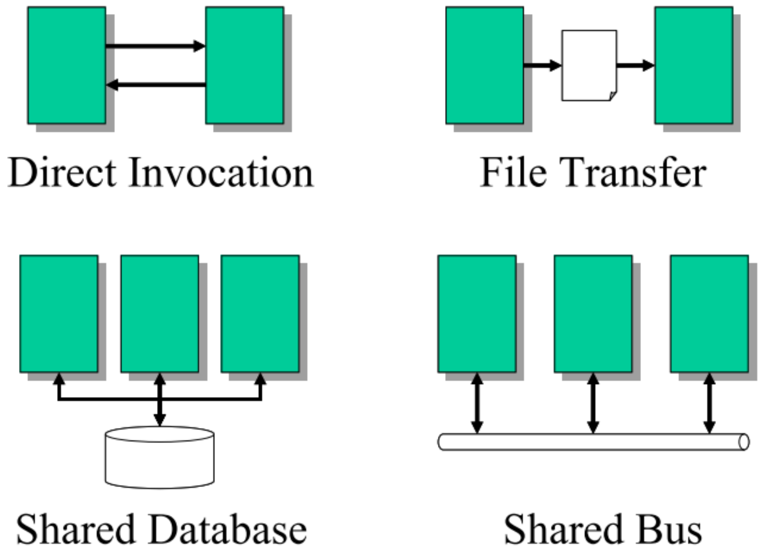


Figure 34 Basic IS architecture patterns (Grefen, 2016)

Appendix C: Semi-Structured Interviewing during the Process Analysis

A. The Outsourcing Process

1. Tactical processes: How is it determined **what** processes are being outsourced?
   a. Who is involved?
   b. How is outsourcing requested? Are there standard documents?
   c. What information is needed further?

2. Operational processes: **How** is communication arranged between outsourcing partner and you?
   a. How are products being 'ordered'?
   b. How do you know which and how many (finished) products are in the pipeline and when it's going to be delivered?
   c. What information is needed further?

B. List of functions interviewed

In Table 15 a list is provided including the references interviewed during the process analysis. Included are the manufacturing environment to which they belong, the reference in general terms, the function(s) of the interviewee(s) and the corresponding date.

Table 15 References interviewed during Process Analysis

MANUFACTURING ENVIRONMENT:	REFERENCE:	FUNCTION(S):	DATE:
ASSEMBLE-TO-ORDER	TC Assembly Plant	Material Planner & EDI-Coordinator	9-11-2017
ASSEMBLE-TO-ORDER	TC Assembly Plant	System Architect	14-11-2017
MAKE-TO-ORDER	TC Manufacturing Plant A	Material Planner	8-11-2017
MAKE-TO-ORDER	TC Manufacturing Plant B	Material Planner	8-11-2017
MAKE-TO-ORDER	TC Manufacturing Plant B	Manager Logistics	8-11-2017
ENGINEER-TO-ORDER	Project-Based Company	Two IT-Specialists	17-11-2017

Appendix D: ArchiMate Notation Language

The components used in the ArchiMate language are described for the business, application and technology layer, respectively in Table 16, Table 17, and Table 18 (Lankhorst et al., 2010).

Table 16 Business layer components

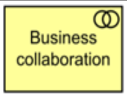
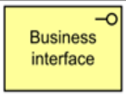
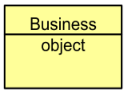
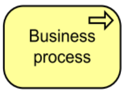
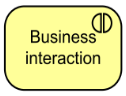
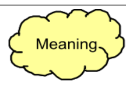
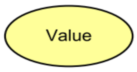
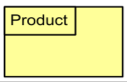
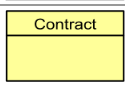
Component	Description	Notation
Business Actor	An organizational entity that is capable of performing behavior.	
Business Role	A named specific behavior of a business actor participating in a particular context	
Business Collaboration	A (temporary) configuration of two or more business roles resulting in specific collective behavior in a particular context.	
Business Interface	Declares how a business role can connect with its environment.	
Business Object	A unit of information that has relevance from a business perspective.	
Business Process	A unit of internal behavior or collection of causally related units of internal behavior intended to produce a defined set of products and services.	
Business Function	A unit of internal behavior that groups behavior according to, for example, required skills, knowledge resources, etc., and is performed by a single role within the organization.	
Business Interaction	A unit of behavior performed as a collaboration of two or more business roles.	
Business Event	Something that happens (internally or externally) and influences behavior. Business	
Business Service	An externally visible unit of functionality, which is meaningful to the environment and is provided by a business role.	
Representation	The perceptible form of the information carried by a business object.	
Meaning	The knowledge or expertise present in the representation of a business object, given a particular context.	
Value	That which makes some party appreciate a service or product, possibly in relation to providing it, but more typically to acquiring it.	
Product	A coherent collection of services, accompanied by a contract/set of agreements, which is offered as a whole to (internal or external) customers.	
Contract	A formal or informal specification of agreement that specifies the rights and obligations associated with a product.	

Table 17 Application layer components

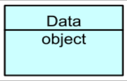
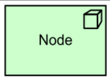
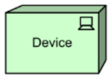
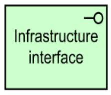
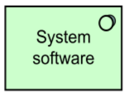
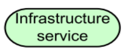
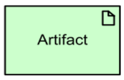
Component	Description	Notation
Application Component	A modular, deployable, and replaceable part of a system that encapsulates its contents and exposes its functionality through a set of interfaces.	
Application Collaboration	An application collaboration defines a (temporary) configuration of two or more components that co-operate to jointly perform application interactions.	
Application Interface	An application interface declares how a component can connect with its environment.	
Data Object	A coherent, self-contained piece of information suitable for automated processing.	
Application Function	A coherent group of internal behavior of a component.	
Application Interaction	A unit of behavior jointly performed by two or more collaborating components.	
Application Service	An externally visible unit of functionality, provided by one or more components, exposed through well-defined interfaces, and meaningful to the environment.	

Table 18 Technology layer components

Component	Description	Notation
Node	A computational resource upon which artifacts may be deployed for execution.	
Device	A physical computational resource upon which artifacts may be deployed for execution.	
Network	A physical communication medium between two or more devices.	
Communication Path	A link between two or more nodes, through which these nodes can exchange information.	
Infrastructure Interface	A point of access where the functionality offered by a node can be accessed by other nodes and application components.	
System Software	A software environment for specific types of components and objects that are deployed on it in the form of artifacts.	
Infrastructure Service	An externally visible unit of functionality, provided by one or more nodes, exposed through well-defined interfaces, and meaningful to the environment.	
Artifact	A physical piece of information that is used or produced in a software development process, or by deployment and operation of a system.	

Appendix E: Evaluation Criteria of Shrivastava (1987)

The TO-BE model was evaluated on relevance, by using the criteria of Shrivastava (1987). These criteria are described in Table 19.

Table 19 Shrivastava's (1987) evaluation criteria

Criterion:	Description:
Meaningfulness	The TO-BE model is meaningful, understandable and adequately adheres the strategic problems faced by TC.
Goal relevance	The TO-BE model adheres to the performance indicators which are relevant to TC's management goals.
Operational validity	The TO-BE model provides clear action implications.
Innovativeness	The TO-BE model surpasses 'common-sense' solutions and provides non-obvious insights into practical problems.
Cost of implementation	The TO-BE model is feasible in terms of time and costs.

Appendix F: Quality attributes, used by Greefhorst & Proper (2011)

In order to classify architecture design principles, Greefhorst & Proper (2011) use six main characteristics, representing quality attributes. These attributes are based on the Extended ISO Model (Van Zeist, Hendriks, & Paulussen, 1996) as given in Figure 35, and are described in Table 20.

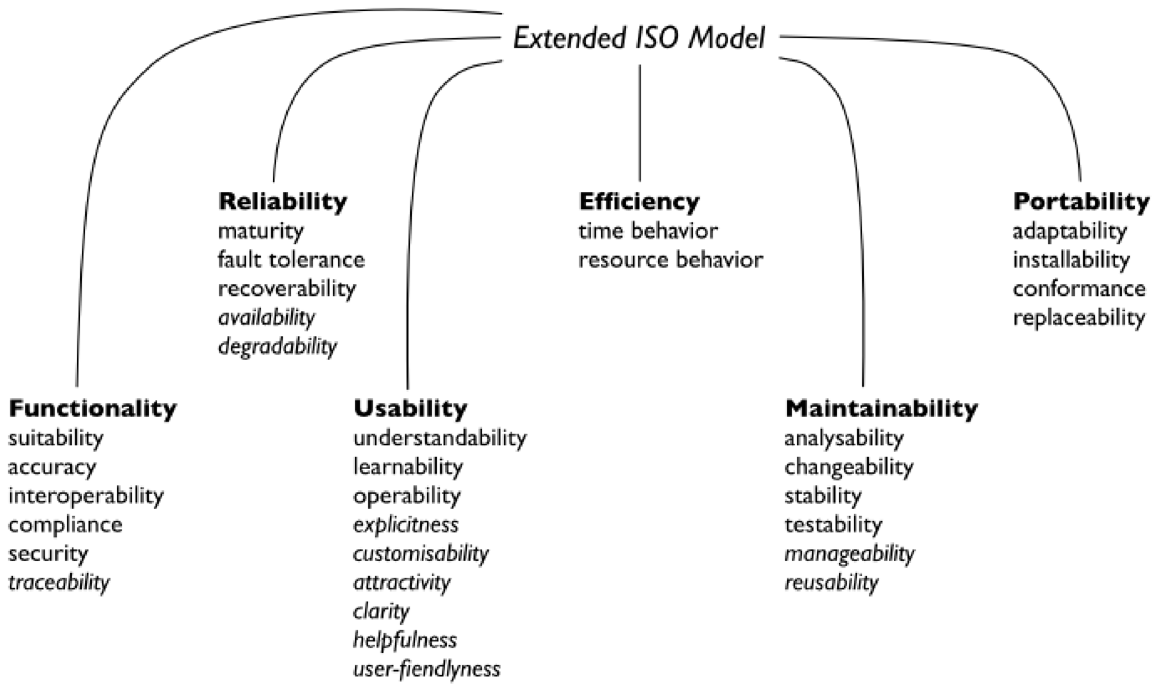


Figure 35 Extended ISO Model (Van Zeist et al., 1996)

Table 20 Main characteristics of quality attributes (Greefhorst & Proper, 2011)

Main characteristic:	Description:
Functionality	A set of attributes that bear on the existence of a set of functions and their specified properties. The functions are those that satisfy stated or implied needs.
Reliability	A set of attributes that bear on the capability of software to maintain its level of performance under stated conditions for a stated period of time.
Usability	A set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by a stated or implied set of users.
Efficiency	A set of attributes that bear on the relationship between the level of performance of the software and the amount of resources used, under stated conditions.
Maintainability	A set of attributes that bear on the effort needed to make specified modifications.
Portability	A set of attributes that bear on the ability of software to be transferred from one environment to another.

Appendix G: Retrieval of Information per System (TO-BE Model)

In Table 21 it is given for each type of information from what system of the TO-BE model it can be retrieved. Furthermore it is indicated whether a new system or an expansion of the current subsystems is needed in order to provide the required information.

Table 21 Retrieval of information per system (TO-BE model)

Phase (level 2)	Information	Subsystem	Expansion/New?
Preparation	- Completely/partly outsourced	Routing Overview	
	- Material/Component definition	Manufacturing Parts List, Drawings	
	- Current capacity	X	Expansion of Machines and Routing Overview
	- Current inventory	Inventory System	
	- Forecasted/Sales order definition	MRP	
	- Period to outsource	X	Expansion of Machines and Routing Overview
	- Machines to outsource	Machines, Routing Overview	
	- Tooling needed	Tooling	
	- Quality requirements	Drawings	
	- Costing price of operation to outsource	Product Development	
Vendor(s) Selection	- Lead time at OP	X	Expansion of Supplier
	- OP definition	Supplier, Supplier Quality	
	- Capacity at OP	X	Expansion of Supplier
	- Currency rates	X	New (Currency)
	- Price of operation at OP	Requirements Planning System	
Transition & Managing Relationship	- Supply of Components via OP or manufacturer	Product Control	
	- Ownership of Components by OP or Manufacturer	X	Expansion of Inventory
	- Shipment of Finished Goods via manufacturer	Sending, Receival, Communication	
	- Forecasted/Sales order definition	MRP	
	- Raw material for internal use	Manufacturing Parts List, Requirements Planning System	
	- Type of packaging	Supplier, Routing Overview	
	- Rejections of components	Quality Information	
	- Inventory at OP	Inventory	
	- Tooling needed	Routing Overview, Tooling	

Appendix H: Subsystems Descriptions

The subsystems described in the TO-BE architecture design are described in general terms. Table 22 gives for each system a description on the content of the system.

Table 22 Subsystem descriptions (TO-BE)

Subsystem:	Description:
MRP	Calculates requirements for materials, based on data and gives recommendations to release replenishment orders
Inventory	In this system the flow of material is registered, by executing stock mutations. It also includes the inventory locations.
Routing	Registers the step-by-step operation to manufacture components, including what material, packaging, etc. is required for each step.
Manufacturing Parts List	Includes the bill of materials for the each component.
Drawings	Registers drawings for each component including quality requirements.
Machines	Registers available machines and the operations executed per machine.
Product Development	Registers all product information of components, including the costing price.
Tooling	Registers the tooling required for each operation
Supplier	Information on the supplier is defined and updated in this system.
Supplier Quality	Registers information on quality indications of suppliers
Requirements Planning	Generates calls for orders and calculates the requirements.
Product Control	Registers the characteristics of each component to be produced.
Sending	In this system the sending of materials and components is registered.
Receival	In this system the receival of material and components is registered.
Communication	This system takes care of communication to external suppliers and outsourcing partners.
Quality	Registers data on rejections of produced components.

Appendix I: Evaluation on relevance

It was determined, based on an expert panel, to what extent the TO-BE model satisfies each criterion of Shrivastava (1987). This is given in Table 23.

Table 23 Evaluation of TO-BE model on relevance

Criterion:	Description:
Meaningfulness	The TO-BE model was considered as understandable and helpful for future purposes of TC.
Goal relevance	The TO-BE model was considered relevant to managerial goals
Operational validity	The TO-BE model was considered operationalizable, but requires more specification since the model is conceptual.
Innovativeness	The TO-BE model was considered as new and innovative, especially the replacement of self-built applications.
Cost of implementation	More research is required on the costs and time of implementation of the TO-BE model.

Appendix J: Evaluation on rigor

In order to assess the TO-BE IS architecture design a paper-based evaluation was used: the architecture design principles of Greefhorst & Proper (2011). For each principle the TO-BE model is assessed on a 3-point scale, ranging from not adhered to (-), to irrelevant, out of scope, or not known (+/-), to adhered to (+). This is based on the study of Tummers (2017). The results of this evaluation are given in Table 24, including an explanation if needed. Note that the many of the last principles are assessed as not known since they comprise design principles representing technology (remember the BOAT-framework in Section 2.3.3). This was out of scope for the model.

Table 24 Evaluation TO-BE model on rigor

	Design principle	**Assessment**	**Explanation (if required)**
1	Business units are kept autonomous	+/-	
2	Customers have a single point of contact	+/-	
3	Stock is kept to a minimum	+	Less use of safety-buffers
4	Processes are straight through	+	Standard outsourcing process
5	Processes are standardized	+	Standard outsourcing process
6	Management layers are minimized	+/-	
7	Tasks are designed around outcome	+	Standard outsourcing process: outcome required per decision
8	Routine tasks are automized	+	Manual tasks automated
9	Primary business processes are not disturbed by implementation of changes	-	Subsystems are not modular
10	Components are centralized	+/-	
11	Front-office processes are separated from back-office processes	+	Already in place
12	Channel-specific is separated from channel-independent	+/-	
13	The status of customer requests is readily available inside and outside the organization	-	No communication between outsourcing partner and customer
14	Data are provided by the source	+	TO-BE acquire data from the source applications
15	Data are maintained in the source application	+	
16	Data are captured once	-	Subsystems exist multiple times in model
17	Data are consistent through all channels	+	Updates occur in source applications
18	Content and presentation are separated	+	Using an user interface
19	Data are stored and exchanged electronically	+	
20	Data that are exchanged adhere to a canonical data model	+/-	
21	Data are exchanged in real-time	-	Event-based
22	Bulk data exchanges rely on ETL tools	+	Already in place

23	Documents are stored in the document management system	+/-	
24	Reporting and analytical applications do not use the operational environment	-	
25	Applications have a common look-and-feel	+	Using an user interface
26	Applications do not cross business function boundaries	+/-	
27	Applications respect logical units of work	+	Already in place
28	Applications are modular	-	
29	Application functionality is available through an enterprise portal	+/-	
30	Applications rely on one technology stack	+/-	
31	Applications interfaces are explicitly defined	+	Event-driven: interface used when required
32	Proven solutions are preferred	+/-	
33	IT systems are scalable	+	Able to support more frequent outsourcing
34	Only in response to business needs are changes to IT systems made	+	
35	Components have a clear owner	+	Self-built applications that are now redundant did not
36	IT systems are standardized and reused throughout the organization	+	Applicable to other plants of TC
37	IT systems adhere to open standards	+/-	
38	IT systems are preferably open source	+/-	
39	IT systems are available at any time on any location	+/-	
40	IT systems are sustainable	+/-	
41	Processes are supported by a business process management system	-	Possible extension in general model
42	Presentation logic, process logic, and business logic are separated	+	Layered style
43	IT systems communicate through services	+	Already in place
44	Reuse is preferable to buy, which is preferable to make	+/-	
45	IT systems support 24/7 availability	+/-	
46	IT systems are selected based on a best-of-suite approach	+/-	
47	Sensitive data are exchanged securely	+/-	
48	IT systems may under no circumstances revert to insecure mode	+/-	
49	Management of IT systems is automated as much as possible	+/-	
50	End-to-end security must be provided using multiple defensive strategies	+/-	

51	Access rights must be granted at the lowest level necessary for performing the required operation	+/-
52	Authorizations are role-based	+/-
53	The identity management environment is leading for all authentications and authorizations	+/-
54	Security is defined declaratively	+/-
55	Access to IT systems is authenticated and authorized	+/-
56	Integration with external IT systems is localized in dedicated IT components	+/-
57	Application development is standardized	+/-
58	All messages are exchanged through the enterprise service bus	-
59	Rules that are complex or apt to change are managed in a business rule engine	+/-

Appendix K: Retrieval of Information per System (Reference Architecture)

In Table 25 it is given for each type of information, based on the Process Analysis, from what system of the reference architecture it can be retrieved.

Table 25 Retrieval of information per system (reference architecture)

Phase (level 2)	Information	Subsystem
Preparation	- Customer Identifying Number	Order Management
	- Bill of Materials	Order Parts List
	- Bill of Processes	Order Parts List
Transition & Managing Relationship	- Ownership of Components by Supplier or chargeable subcontracting	Inventory
	- Shipment of Finished Goods directly from OP	Requirements Planning
	- Product/material options	Order Management
	- Expected Delivery Date (EDD)	Requirements Planning
	- Customer-Identifying Number	Order Management

In Table 26 for each subsystem added a description is given.

Table 26 Subsystems described (reference architecture)

Subsystem:	Description:
Order Management	In this system each customer order, including its customer specific options, is registered.
Order Parts List	In this system the bill of materials and bill of processes is registered. For each finished good it is set what materials/components are needed and how each material/component is being processed to manufacture the finished good. It corresponds to a hierarchical structure.
Inventory	In this system the flow of material is registered, by executing stock mutations. It also includes the inventory locations. It should be included by who the components at an inventory location are owned.
Requirements Planning	Generates calls for orders and calculates the requirements. For each outsourcing order it should be added what the EDD is and how finished goods are shipped.