MASTER

Adding smartness to smart factories

Pepels, E.K.K.

Award date:
2020

Department of Industrial Engineering and Innovation Sciences
Information Systems

Master Thesis

Adding smartness to Smart Factories

Pepels, E.K.K.

Award date: January, 2020

Eindhoven, January 2020

Adding smartness to Smart Factories

Method for the design of networks of Asset Administration Shells in Smart Factories

Eligio Pepels
Student identity number: 0847062

In partial fulfilment of the requirements for the degree of
Master of Science
in Operations Management and Logistics

Keywords: Industrie 4.0 Smart Factory, Cyber-Physical Systems, I4.0 component, Asset Administration Shell

Supervisors:
Dr. M. (Maryam) Razavian, TU/e, IS
Dr. ir. J.J.M. (Jos) Trienekens, TU/e, IS
Dr. B. (Baris) Ozkan, TU/e, IS

MSc. ir. P. (Paul) Bruinen, Atos
Ir. J.L.M.(Hans) Kwaspen, Atos

Abstract

The fourth industrial revolution is becoming an increasingly common concept. It is the fourth paradigm shift, which is characterized by Cyber-Physical Systems, Smart Factories, Internet of Things, and the Internet of Services. Three forms of integration separate the fourth industrial revolution from the third industrial revolution, namely: horizontal, vertical, and end-to-end integration. In this research, one of the most recent innovations in I4.0 is central: the Asset Administration Shell (AAS). The AAS models the cyber (virtual) part of the Cyber-Physical Systems and will cause significant changes in the way business processes are carried out. AAS enable the link between the virtual and physical world, as Cyber-Physical Systems require. The research is focused on the design of systems of AAS to improve Smart Factories to even Smarter Factories. The various elements of the AAS are structured and a method is created to design AAS based on existing method fragments. The method is in the end validated by exemplars, a focus group, and semi-structured interviews.

Preface

This master thesis is the result of an almost 10-month graduation project to fulfil my master's degree in Operations Management & Logistics at the Eindhoven University of Technology (TU/e). I was granted the opportunity to finalize my master's degree within a very warm and friendly company, Atos. I would like to take this opportunity to thank the people that supported me the most during this great educational experience.

First, I would like to thank my mentor Maryam Razavian, thank you for the support and guidance throughout my years as a master's student. Even though I know you were very busy with other work-related activities, I felt like you always wanted to help me where needed. You know my strengths (and weaknesses) and provided me with feedback so I could improve my master thesis a lot. Next, I want to thank Jos Trienekens and Baris Ozkan for stepping in on such short notice at the end of the project, thank you.

Next, I want to thank Paul Bruinen, Hans Kwaspen, Menno Blanken and all the other students within Atos. You allowed me to perform my research in a very nice environment. Thank you, Paul, for freeing up your time week after week. The long discussions with you (about not only academic topics) always gave me reasons to think and criticize the things I read. Thank you for being such a good supervisor. Thank you, Hans, for always being there for everyone. I think all students and employees within Atos can only praise you. Thank you Menno, for giving me another opportunity after our first attempt to cooperate was unfortunately unsuccessful. Thanks to all the other students I met during my time at Atos. You made my time with Atos even more enjoyable and have often helped me when I got stuck with something. A special thank you to Kevin, whom I have worked with all summer long.

In the end, I would like to thank my family, friends, and girlfriend. My family always supports the things I do in life. You help me balance my life between sports at a high level and my education. You always allow me to make my own decisions but are always there for me when I need you. Thanks to my friends, who mostly help me to get my mind a bit off school, but also find time to listen to me when I want to talk about school. In the end, thank you Veerle for being the support I needed especially towards the end of my master thesis. You always helped me find the right motivation and could help me get my mind off of studying when needed.

I know it is hard to express my gratitude towards all of you in words, but at least I tried my best. Thanks all!

Eligio Pepels

Executive summary

0.1 Introduction

This report shows the results of a master thesis research at the I4.0/IoT unit of Atos NL. The unit has stated its interest in one of the most recent advances in Industrie 4.0, the Asset Administration Shell. The Asset Administration Shell (AAS) is the virtual component of the I4.0 Component. The I4.0 Component is a model for Cyber-Physical Systems, which are at the center of Industrie 4.0. The I4.0/IoT unit wants to set up a new service around the AAS concept. Their idea of the service was to help their clients with improving their factories towards smarter factories. This new service must fit within their current I4.0 Consult-Build-Run service strategy. More specifically, this new service must support the Build services offered by the unit. As a result it is decided to create a method to guide the process of designing AAS based on new business requirements. The research objective is subsequently defined as:

'The objective of the research is to develop a method for the design of networks of Asset Administration Shells, the software of I4.0 Components, that are needed to realize new I4.0 related business requirements for the incremental transformation to Smart(er) Factories.'

For the research the Design Science Research Methodology defined by Wieringa (2014) is followed. This methodology describes the following three phases: the problem investigation, artifact design and artifact validation. The Design Science Research Methodology phases are discussed hereafter.

0.2 Problem Investigation

The problem investigation phases is to identify the problem that exists in more detail. Therefore, using a Systematic Literature Review the theoretical research gap is defined. The research on the AAS at this point is still in its infancy. Only 17 documents could be found in five different electronic databases. The AAS research is currently almost exclusively performed by the Plattform I4.0, who have also conceived the concept. Their publications tend to be rather technical and scarcely focus on the business aspects of the AAS. This indicates a gap in the research on the AAS. Also, the research on the AAS is still ongoing and far from being completed. Standardization efforts and improvements to the concept in general are still being made on a regular basis. Additionally, based on the Systematic Literature, the most relevant aspects for the design of networks have been identified.

Besides this, looking at the problem in practice the following can be stated. At this point, Industrie 4.0 technologies are being implemented on a large-scale. However, the implementations of the Asset Administration Shell, which will become at the center of I4.0 in the future, are still far behind. Therefore, the design processes of the currently most advanced implementations are investigated in the problem investigation phase. From the design processes several (mandatory) design activities could be identified. And focusing on the process in general, it is found that their processes lack consistency, are relatively incomplete, and hard to understand.

0.3 Artifact Design

For the design of the artifact, a method for the design of networks of AAS, several requirements can be found. The artifact must be usable by the Atos employees of the unit (usability), it must support the design of networks of AAS (utility), it must be understandable (understandability), and provide guidance of the complete process (completeness). Besides that, the artifact must enable Atos to build a service in the future regarding the AAS and must help raise awareness and understanding of the AAS concept. The design of the artifact is performed in close collaboration with an Business Consultant / Enterprise Architect, which can be seen as the representative of the unit in this research. Together with the representative, a general structure of the method is defined and its fit with Atos' current I4.0 services is determined. The resulting structure of the method is shown below.

Figure 1: The method phasing and fit with current I4.0 services offered by the unit

Thereafter, the design activities and design aspects are fit within the created structure. The result of this is is the artifact: the Asset Administration Shell Design Method (AASDM). The AASDM shows the most important activities to translate new business requirements into a design of a network of AAS. The AASDM is shown below.

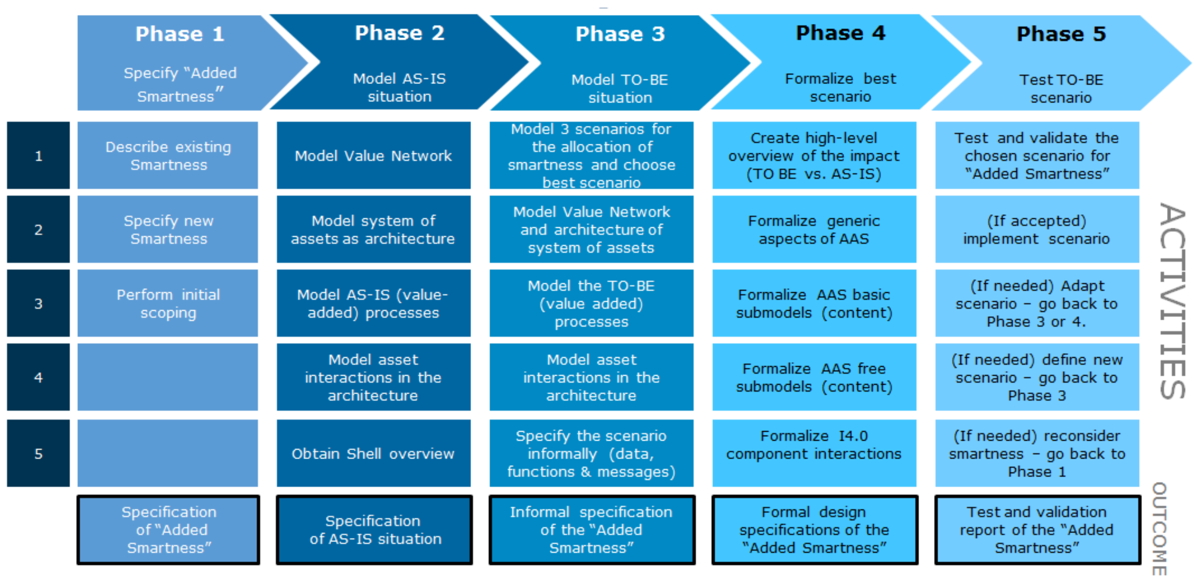

Figure 2: The AASDM

0.4 Artifact Validation

At first, the AASDM is applied to two real-world like case settings, in which the value of the AASDM is tested. Thereafter, to validate the artifact, a focus group was held and semi-structured interviews were conducted with potential users of the AASDM. Together with the manager of the unit, potential users were selected. During the focus group and semi-structured interviews the artifact requirements and stakeholder goals were addressed. In the end, it is found that the artifact meets the artifact requirements and fulfils its stakeholders goals. The potential effect of the AASDM for the unit is identified to see how well the AASDM fulfils the stakeholders goals.

0.5 Conclusion

It can be concluded that the artifact meets its desired outcomes. For the unit the AASDM can be used as a training tool or introduction to the AAS concept. This way the AASDM can be used by the unit to establish a new service around the AAS concept. For the AASDM to be transformed into a service, implementation activities must be undertaken to professionalize the AASDM into a service.

The research conducted contributes to the literature by providing a structured and complete approach to design AAS. Researchers now have the possibility to design and implement AAS more in a more structured manner. Besides that, the AASDM can be used to define new free submodels, which can be at the basis of a standard for a submodel.

The AASDM research can be extended by using it in real-world case studies. This way the value of the AASDM can be tested in practice. Furthermore, the AASDM can be extended to guide the development process of the AASDM as well. The output of the AASDM are design specifications of the network of AAS, but in order to implement the software more activities must be undertaken. Working towards a proof-of-value or proof-of-concept is a good option to prove the business value of the AASDM.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Industrie 4.0, also known as the fourth industrial revolution, is the name given to the current trends in the development of new technologies in the manufacturing domain. In history four paradigm shifts in the manufacturing industry can be recognized, see Figure 1.1. The industry has moved on from digitization to automation toward autonomous systems. Industrie 4.0 is characterized by Cyber-Physical Systems. The term Industrie 4.0 was introduced in Germany in 2011 (Kagermann, Lukas & Wahlster, 2011). Hereafter, the new industrial revolution has gained attention from all around the globe, in both practice and the academic world. However, there is still a gap to bridge from theory to practice for I4.0. As is said Industrie 4.0 is characterized by Cyber-Physical Systems. Now, a standardized model for Cyber-Physical Systems is being developed, the I4.0 Component. The I4.0 component consists of an asset and a virtual representation, called the Asset Administration Shell. The study aims to guide the transition from theory to practice for Cyber-Physical Systems (I4.0 Components) through the design of their virtual representations in the form of Asset Administration Shells. This transition is key to the implementation of I4.0 in Smart Factories. This Master Thesis project is executed in collaboration with the I4.0 group within Atos NL.

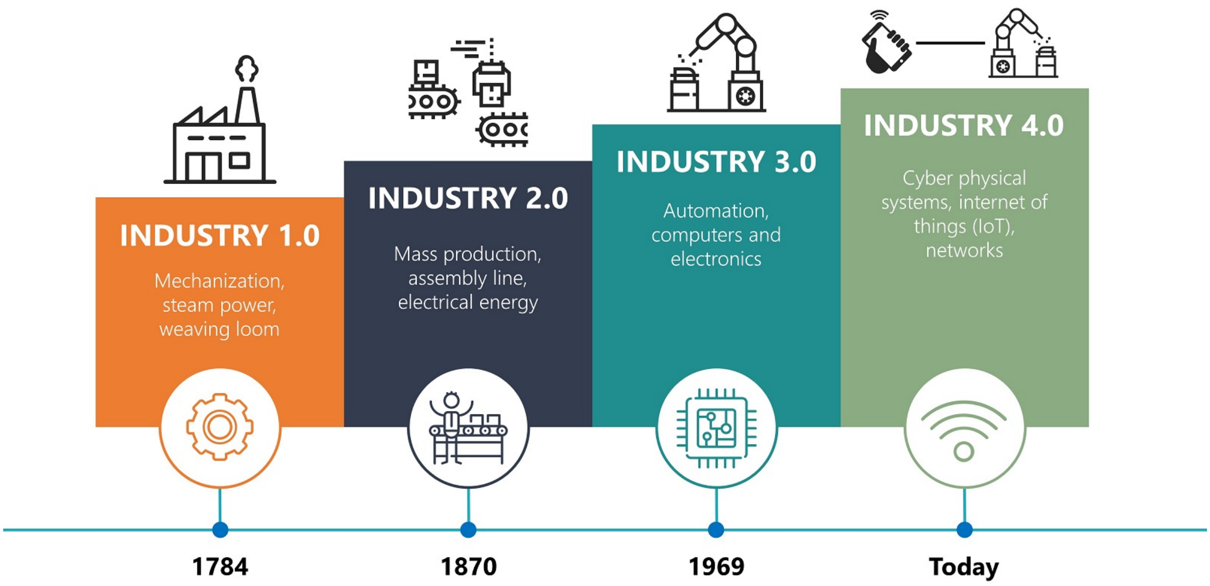

Figure 1.1: The four industrial revolutions

Chapter 1 introduces the project by introducing Atos NL and its I4.0 related services in section 1.1, the problem context and definition in section 1.2 and 1.3, and the research design in section 1.4. In the end, the report structure is shown in section 1.5.

1.1 Company description Atos NL

Atos is a global IT services company located in Europe originated in France. Atos is the global leader in digital transformation with over 122.000 employees (in 2018) in 73 countries and annual revenue of over 12 billion euros. Keywords for Atos are digital transformation, innovation and value creation for Atos' clients and the company itself. Atos is at the top in the cloud, cybersecurity and high-performance computing. Atos supports the digital transformation of its clients across all business sectors and is also responsible for the IT at the Olympic & Paralympic Games [1]. The company operates under the following brands: Atos, Atos Consulting, Atos Healthcare, Atos Worldgrid, Bull, Canopy, Unify, and Worldline.

Services within Atos can be organized into four different divisions: infrastructure & data management, business applications & platform solutions (B&PS), Big Data & cybersecurity, Worldline (e-commerce payments services). Activities within Atos are based on four 'pillars', which are: the cloud, digital workplace, SAP HANA, Big Data Analytics.

Atos I4.0 related services

Atos offers specific solutions and services related to Industrie 4.0 and for the MRT clients (Manufacturing, Retail and Transport). Industrie 4.0 is listed as one of the most recent solutions provided by Atos on their website [2]. Industrie 4.0 is said to be at the heart of digital transformation for every manufacturing enterprise. The goal is to create value from insights generated by the various 'smart and connected' assets throughout the product life cycle (from idea to design, manufacturing and supply chain, sales and services).

The master thesis research project is carried out at the I4.0/IoT Unit of Atos B&PS Digital at Eindhoven. The project is focused on a crucial concept of Industrie 4.0 namely the I4.0 Component, especially on its Asset Administration Shell and the submodels therein. These submodels describe and offer generic and asset-specific functions and related data.

Atos offers Consult-Build-Run services – related to Industrie 4.0 – to its customers. The I4.0 'consult' services include I4.0 Awareness workshops, I4.0 Maturity scans and I4.0 Business Case and roadmap development. The I4.0 'build' services include I4.0 Program/project management (from specifications to implementation), I4.0 Prototyping, I4.0 Business Application configuration/customization, I4.0 Integrations, I4.0 Software development, I4.0 Testing, and I4.0 Training. The I4.0 'run' services include I4.0 Solutions Service management and I4.0 Improvement/upgrade projects.

1.2 Problem Context

I4.0 is characterized by three forms of integration: (1) horizontal integration through value networks to facilitate inter-corporation collaboration, (2) vertical integration of hierarchical subsystems inside a factory to create flexible and reconfigurable manufacturing system, and (3) end-to-end engineering integration across the entire value chain to support product customization (Wang, Wan, Li & Zhang, 2016). Figure 1.2 illustrates the relationships of the three types of integration. The added value of I4.0 lies in among other things: (1) allow for customers to have individual requirements (mass customization), (2) generate dynamic business and engineering processes, (3) facilitate optimized decision-making, and (4) create new business models and forms of value creation (Kagermann, Wahlster & Helbig, 2013).

[1] https://atos.net/en/about-us/company-profile
[2] https://www.atos.net/

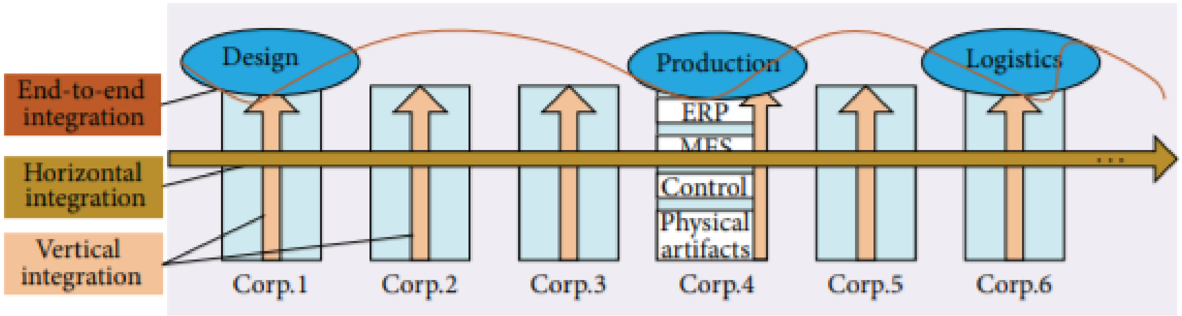

Figure 1.2: Illustration of the three types of integration from (Wang, Wan, Li & Zhang, 2016)

Key to the realization of I4.0 are Smart Factories. The setting of vertical integration is the factory, vertical integration means implementing Smart Factories, which are flexible and reconfigurable (Wang et al., 2016). Smart Factories are realized via Cyber-Physical Systems. According to (Monostori et al., 2016) Cyber-Physical Systems 'are systems of collaborating computational entities which are in intensive connection with the surrounding physical world and its on-going processes, providing and using, at the same time, data-accessing and data-processing services available on the Internet'. (Lee, Bagheri & Kao, 2015) defines Cyber-Physical Systems as technologies for managing interconnected systems between its physical assets and computational capabilities. In (Alcácer & Cruz-Machado, 2019) it is mentioned that Cyber-Physical Systems merge the physical with the virtual world. An easy explanation of Cyber-Physical Systems is that they are smart and connected entities in an I4.0 (production) environment, which have the ability to communicate and act (semi-)autonomously.

As stated before, in the Smart Factories Cyber-Physical Systems are central. Plattform I4.0 has defined a generic model for CPS called the I4.0 component. This model is based on the RAMI4.0 architecture and relies on standards set by the I4.0 community. The I4.0 Component does not only allow for vertical integration but because of its standardized nature results in horizontal and end-to-end integration as well. The I4.0 Component represents production system components (assets) in I4.0 without affecting the basic functionalities of this production system component (Lüder, Schleipen, Schmidt, Pfrommer & Henßen, 2017). The I4.0 Component consists of an asset and an Asset Administration Shell, a data container which contains the data and technical functionalities of the asset (Zezulka, Marcon, Vesely & Sajdl, 2016). The I4.0 components also have the ability to function (semi-)autonomously and communicate with other I4.0 components in the network.

The transition from I3.0 to I4.0 is a radical change (Wang et al., 2016). According to (Radziwon, Bilberg, Bogers & Madsen, 2014) a lot of effort must be put into the I4.0 concept to become true. It would be beneficial to change the radical nature of I4.0 to an incremental nature. Incremental changes should be the focus of I4.0. The transition of I3.0 factories, in which big software packages are key decision-makers, towards the Smart Factories, in which local decision-making by CPSs is key, would become less complicated. These incremental improvements from a barely smart factory towards a full-functioning 'Smart Factory' will be referred to as "Adding Smartness to a Smart Factory" in the remainder of this document.

In this master thesis project, the focus is on adding smartness to Smart Factories using the I4.0 Component concept. The I4.0 Component concept is found to be the most promising way to change from the traditional factory to full-functioning Smart Factory. It enables the three forms of integration from its initial deployment because it is created with the I4.0 forms of integration in mind. The artifact created in this research focuses on the design of I4.0 Components from specific desires to add smartness to a Smart Factory.

1.3 Problem Definition

From a research perspective, the I4.0 component is in general underexposed. The Standardization for I4.0 components is an ongoing process. Implementations of I4.0 components in most cases do not follow the restrictions and standards defined for them based on RAMI4.0. Research is mostly carried out by Plattform i4.0 and focuses on the technical aspects of the I4.0 component such as (Adolphs et al., 2016), (Plattform I4.0, 2018a) and (Platform-I4.0, 2019).

From an application perspective, the I4.0 component is only implemented on small scale projects. The implementation of I4.0 components in functioning factories is yet to be done. The lack of understanding of the I4.0 Component hinders the realization of Smart Factories on a large scale. Therefore, the I4.0 Component concept must become easier to understand and implement. At this point, Smart Factories are realized without the link to standards, without the I4.0 Component concept, which is crucial.

Atos has noticed that the I4.0 Component and its Asset Administration Shell are promising innovations. As a service provider, Atos wants to determine opportunities to provide services concerning the I4.0 component and Asset Administration Shell. Therefore, the artifact must support the ability to be transformed into an Atos service. The preferred outcome is, therefore, a method for the design of Asset Administration Shells for networks of I4.0 components to add smartness to Smart Factories. This method is at the basis of a guide to provide a service called "Adding Smartness to Smart Factories" by Atos. The problem to be solved, in general terms, is formulated as follows:

'The realization of Smart(er) Factories is hindered by the lack of knowledge on and understanding of how to design (networks of) I4.0 components and their Asset Administration Shells correctly.'

,

The method created in this project is aimed to steer employees of Atos NL. However, the method is not company-specific and can be used by other companies as well.

1.4 Research Design

The research addresses the development of a method (at the basis of a new Atos service) to design networks of Asset Administration Shells based on desires for new smartness in a Smart Factory. The project aims to bridge the identified gap between theory and practice on the Asset Administration Shell. For the development of the method, the framework for Design Science by Wieringa (2014) is used. This section defines the research goal, research questions, and methodology 1.4.5 of the Master Thesis project.

1.4.1 Research goal

The goal of the Master Thesis research project is to create a method for the design of Asset Administration Shells for networks of I4.0 components. There is no (standardized) structured approach to design AAS currently. The current research tends to focus on the technical aspects of I4.0 components (and their Asset Administration Shells). The relation of I4.0 Components to business value is often under-explored.

For the design of this Master Thesis research, the Design Science methodology is used. This methodology is focused on the design and validation of an artifact. The artifact - a method for networks of AAS – is developed to provide a solution to a practical business problem following the Design Science methodology. Therefore, it is important to first state the research goal in the form of a Research Objective. The research objective of the Master Thesis project is defined as follows:

'The objective of the research is to develop a method for the design of networks of Asset Administration Shells, the software of I4.0 Components, that are needed to realize new I4.0 related business requirements for the incremental transformation to Smart(er) Factories.'

For clarity, the new I4.0 related business requirements for the incremental transformation to Smart(er) Factories refer to the business requirements for "Adding Smartness" to Smart Factories to create Smarter Factories.

1.4.2 Scope

Based on the research goal, the scope of the project and the outcome is determined. In the research objective, it is stated that a method must be created for the design of networks of Asset Administration Shells, the software of I4.0 Components. One can distract the scope of this project from this statement.

First, the focus is on Asset Administration Shell. The Asset Administration Shell itself is a very specific topic within I4.0, which is the fourth industrial revolution of the manufacturing domain. The scope is thus on software in the manufacturing domain.

Secondly, the focus is on software design, which may not be confused with software development. The difference between the two must, therefore, be made clear. "Software development is a professional activity in which software is developed for business purposes, for inclusion in other devices, or as software products such as information systems and computer-aided design systems" (Sommerville, 2011). Software development for Asset Administration Shells would include the development, implementation, testing, and maintenance of the AAS software. In this master thesis, the focus is rather on software design. Software design is defined here as "a description of the structure of the software to be implemented, the data models and structures used by the system, the interfaces between system components and, sometimes, the algorithms used" (Sommerville, 2011). If we look at the general structure of software development projects often the Software Development Life Cycle is at the basis of this. "The Software Development Life Cycle (SDLC) is the collection of various steps which followed for the systematic development, design and maintenance of the software projects and ensure that all the user requirements are fulfilled with the least amount of resource consumption" (Ragunath, Velmourougan, Davachelvan, Kayalvizhi & Ravimohan, 2010). The SDLC consists of four general stages: requirements, design, implementation, and testing (Ragunath et al., 2010). As can be seen in the SDLC a distinction is made between software design and implementation. This project thus focuses on the first two steps of the SDLC. In Figure 1.3 this is visualized.

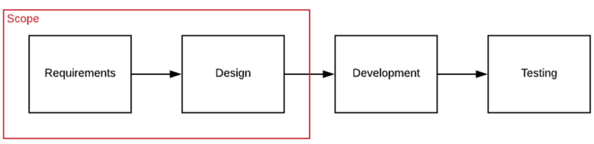

Figure 1.3: Scope in relation to the generic SDLC

The implementation of the software is out of scope for two reasons. Firstly, the implementation of the software requires expert knowledge of the (embedded) software of the assets. The link between the properties of the AAS and the functions of the assets must be established by experts with knowledge of the specific asset. Secondly, the integration of AAS software requires configuration with higher-level IT systems.

Third, it is stated that the research evolves around networks of AAS. To improve the smartness of a Smart Factory not only the design of individual AAS is of importance, but also the interaction between AAS. For the addition of new smartness, both the submodels and the interactions between AAS can change. It is thus important to define how and what I4.0 components interact with each other.

1.4.3 Design Problem

For the design science methodology, it is important to write down the design problem. The design problem can be drafted according to the template of Wieringa (2014).

Problem Context	The lack of knowledge on and understanding of the I4.0 Component and Asset Administration Sell hinder the realization of Smart Factories.
Design Artifact	A method for the design of networks of Asset Administration Shells to add smartness to (semi) Smart Factories.
Artifact Requirements	• Must guide the complete process of translating business requirements into AAS design must be described (completeness). • Must be useful to Atos employees to design networks of AAS (utility). • Must be understandable to Atos employees (understandability). • Must be usable for Atos employees (usability)
Stakeholder goals	• Raise awareness and understanding of the AAS concept in the unit. • Method at the basis of a guide for a new Atos service "Adding Smartness to Smart Factories).

Figure 1.4: The Design Problem

1.4.4 Research Questions

The main research question is based on the research objective and the design problem. The goal of the project is to create the artifact. To do so, the main research question has to be answered. The research goal is to create an artifact – a method for the design of I4.0 AAS software. The main research question is formulated and answered to reach the research objective. This is why the main research question is formulated as follows:

Main Research Question (MRQ): 'How can the (re)design process of Asset Administration Shells be guided to realize new smartness in Smart Factories?'

This research question is leading in creating the artifact, a method for the design of AASs for networks of I4.0 components to add smartness. The main research question must be answered to be able to design the artifact. The main research question is formulated comprehensively and can be broken down into three sub-questions. These sub-questions are based on the Design Science Cycle. An overview of the research questions answered in this research is shown in figure 1.5. The sub-research questions are explained briefly hereafter.

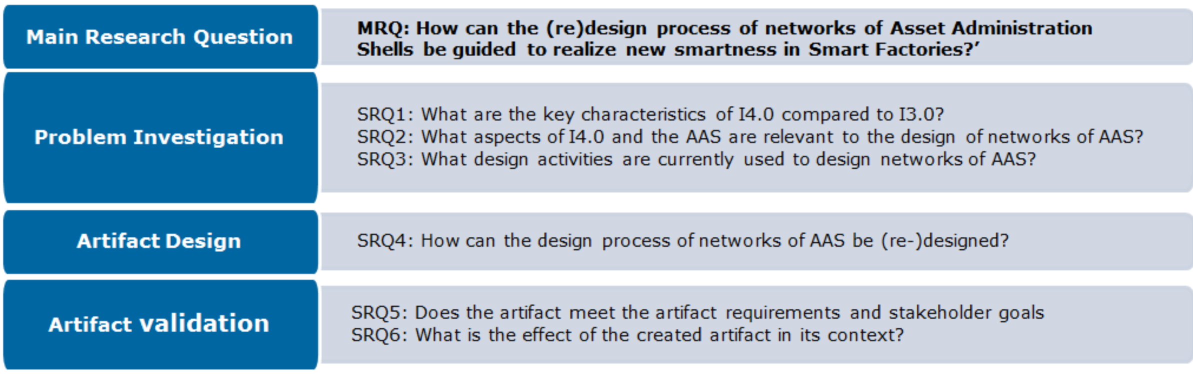

Figure 1.5: Research questions

Sub-Research Question 1 (SRQ1): 'What are the key characteristics of I4.0 (compared to I3.0)?'

Atos' work activities are currently based on more I3.0-like solutions than I4.0 solutions. A common understanding must be established in the Problem Investigation phase on what I4.0 encompasses and what are the main differences between I3.0 and I4.0.

Sub-Research Question 2 (SRQ2):'What aspects of I4.0 and the AAS are relevant to the design of networks of AAS?'

In order to design the artifact, the aspects of the AAS that are relevant for its design must be identified. These aspects will be used to evaluate the current design activities and be on the basis of the design of the artifact. This research question is answered during the Problem Investigation phase.

Sub-Research Question 3 (SRQ3): 'What design activities are currently performed to design networks of AAS?'

I4.0 is still in its infancy and the transition from I3.0 to I4.0 is far from complete. The AAS is one of the most recent advances in I4.0 and is therefore under-explored. Still, some companies and researchers have managed to implement AAS. The design activities undertaken to design (networks) of AAS are analyzed. The current state of AAS design in practice is established as part of the Problem Investigation phase.

Sub-Research Question 4 (SRQ4): 'How can the design process of networks of AAS be re-designed?'

The proposed artifact is a method for the design of networks of AAS from new business requirements. The method must guide the process of designing networks of AAS and therefore this sub-research question must be answered as part of the Artifact Design Phase.

Sub-Research Question 5 (SRQ5): 'Does the artifact meet the artifact requirements and stakeholder goals?'

This sub-research question is focused on the validation of the artifact. It must be validated that the created artifact fulfils the stakeholders' goals and meets the artifact requirements.

Sub-Research Question 6 (SRQ6): 'What is the effect of the artifact in its context?'

The last sub-research question focuses on what effect the created artifact will have for the I4.0/IoT unit of Atos NL. It is part of the validation of the artifact and helps to establish the current use and future use of the artifact.

1.4.5 Research methods

The methodology in this master thesis is based on the Design Science (Research) Methodology defined from Wieringa (2014). A framework is defined for Design Science. It is similar to the framework of Hevner, March, Park and Ram (2004) with some adjustments such as the separation of design and investigation. At the basis of the methodology defined by Wieringa (2014) are the design cycle as part of the engineering cycle and the empirical cycle.

It is generally accepted that Design Science Research is supported by both rigour and relevance. Hevner (2007) describes the relation of the Design Science Cycle with the Rigor Cycle and Relevance Cycle. Hevner (2007) states that the design cycle is supported by a rigour and relevance cycle, in which the contextual environment and knowledge base are linked to Design Science activities. These two cycles, rigour and relevance, appear in the framework for Design Science from Wieringa (2014) as the social context and knowledge base. In this section, the research methods to execute the three phases of the Design Science Methodology are shown in figure 1.6 and explained subsequently.

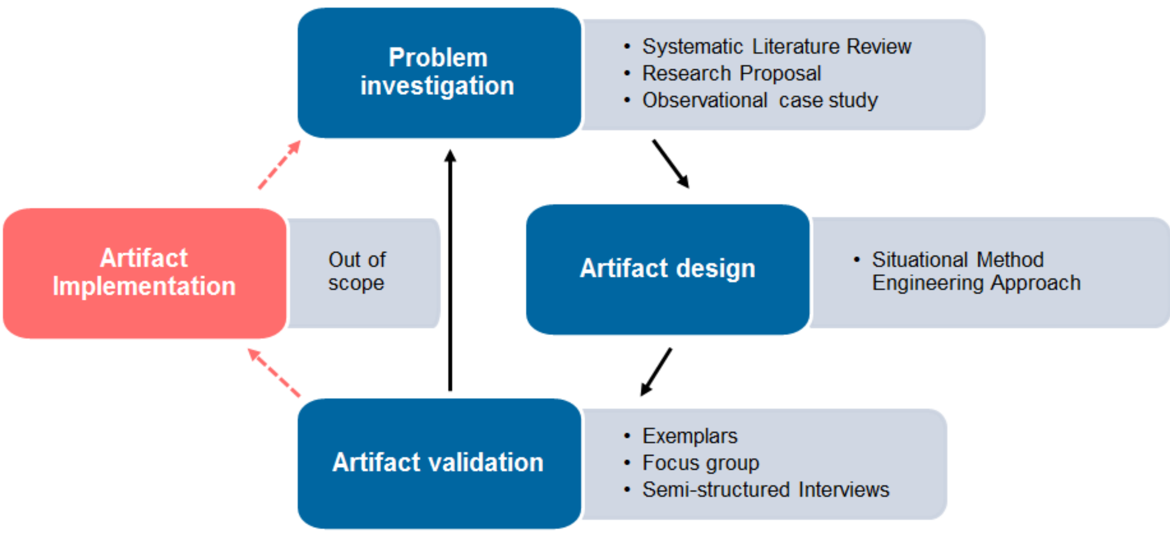

Figure 1.6: The Design Science Methodology

Problem investigation

The problem understanding phase is the first phase of the Design Science Cycle. Sub-Research Question 1 an 2 are to be answered during this phase. A Systematic Literature Review (SLR) is performed to establish the knowledge base required for the Master Thesis Project and answer sub-research questions 1 and 2. The research proposal is performed prior to the Master Thesis project and is used to establish the problem and approach of the project. Thereafter, to answer SRQ3, an observational case study is performed. The Problem Investigation targets the current process and the need for improvement. Based on the requirements and restrictions for the artifact can be determined.

Artifact design

In the artifact design phase, the method for the design of networks of AAS is created to improve the process described in the Problem Investigation phase. A situational method engineering approach is adjusted to establish the method. The conceptual design (method outline) is designed in collaboration with a representative of the unit. Thereafter, method fragments are selected to finalize and operationalize the artifact. This artifact design process is carried out in an iterative manner. This phase will answer the fourth sub-research question: *'How can the design process of networks of AAS be re-designed?'*.

Artifact validation

The final phase of the design science cycle is artifact validation. The artifact validation is carried out by means of exemplars, a focus group and semi-structured interviews. In the validation of the artifact the artifact is validated against its artifact requirements. Besides that, the stakeholder goals and the effect of the artifact in its context are determined. In this phase, the fifth and sixth sub-research questions are answered.

1.5 Report Structure

In figure 1.7 the structure of the report is shown. The report is structured around the Design Science Methodology as can be seen.

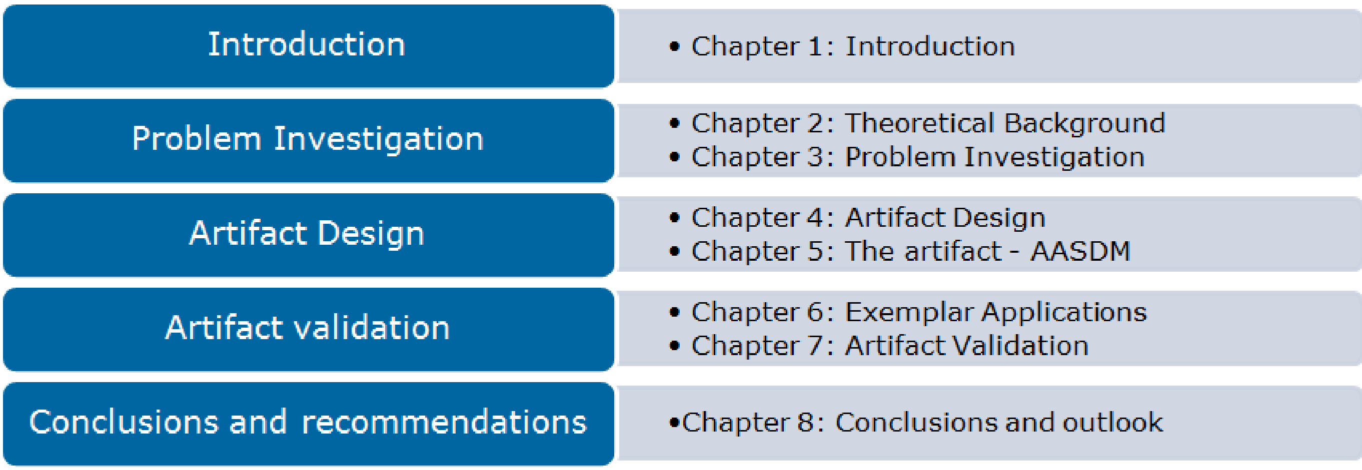

Figure 1.7: Report Structure

Chapter 2

Theoretical background

This chapter contains the theoretical background for this Master Thesis project. The theoretical background is established by a Systematic Literature Review (SLR). The Systematic Literature Review is performed mostly prior to and partly during this Master Thesis Project. The Systematic Literature Review can be split up into two topics: Industrie 4.0 and the Asset Administration Shell. The concept of Industrie 4.0 must be clarified before the Asset Administration Shell can be understood. In the end, the Systematic Literature Review results in a conceptual framework for the artifact.

In section 2.1 the Systematic Literature Review approach is explained in detail. Section 2.2 focuses on the topic of Industrie 4.0 and reports its relevant aspects in relation to the design of the software. Section 2.3 concentrates on the Asset Administration Shell.

2.1 Systematic Literature Review

To form the theoretical background for the Master thesis project a Systematic Literature Review (SLR) is performed. The goal of this SLR is to find all the relevant information about I4.0 (design) and the Asset Administration Shell. To ensure that all relevant information is found, a structured approach is chosen for the SLR according to (Kitchenham & Charters, 2007). An overview of the Systematic Literature Review approach is shown in figure 2.1.

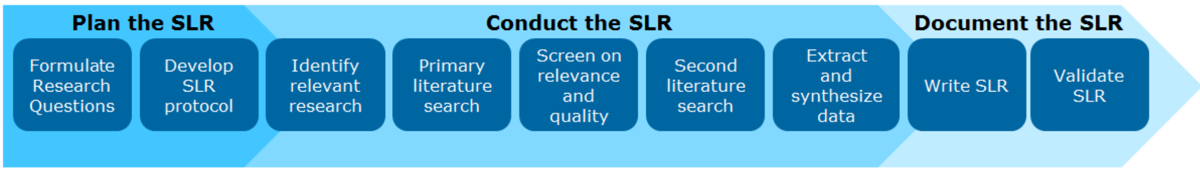

Figure 2.1: Systematic Literature Review approach

With the Systematic Literature Review, a proper knowledge-base is established for the design of the artifact. The topics that are covered in the Systematic Literature Review in relation to Industrie 4.0 are a definition of the concept, the core components of I4.0, characteristics of I4.0, design principles of I4.0, and standardization in I4.0. The topics covered in the Systematic Literature Review with relation to the Asset Administration Shell are: a definition of the concept, the core concepts around the Asset Administration Shell, characteristics of the AAS, and implementations of the AAS.

Research questions
The Systematic Literature Review aims to answer the following two sub-research questions:

- SRQ-1: *What are the key characteristics of I4.0 compared to I3.0?*
- SRQ-2: *'What aspects of I4.0 and the AAS are relevant to the design of networks of AAS?'*

Search strategy
The search strategy is determined to select a literature base of a decent size. The search queries used for the Systematic Literature Review that resulted in a manageable number of documents are:

- Search query 1: ((“Industrie 4.0” OR “Industry 4.0” OR I4.0) AND design) in the meta-data of the papers.
- Search query 2: ((“Industrie 4.0” OR “Industry 4.0” OR I4.0) AND (“Asset Administration Shell” OR AAS)) in the metadata of the papers.

Sources
To find the relevant literature five different electronic databases have been explored. The five used databases all contain literature in the field of information technology. The following list of electronic databases has been used for the literature search: IEEE Xplore Digital Library [1], ScienceDirect [2], Springer [3], Web of Science [4], ACM Digital Library [5].

Primary literature search and screening
The search queries stated in the search strategy have resulted in a long list of documents. The first search query resulted in 121 documents and the second search query resulted in another 17 documents. In total 138 documents were included in the long list. Not all documents obtained from this search strategy are of proper quality and/or of relevance for the theoretical background. Therefore, quality and relevance criteria have been determined (see Appendix A.1). Based on these criteria the long list is reduced to a middle list (after the quality assessment) and eventually to a shortlist (after the relevance assessment). In the end, the long list of 138 documents is reduced to 31 documents (27 from the first query and four from the second query). This is the result of the primary literature search.

Secondary literature search
The secondary literature search is obtained using the primary literature search results. Forward and backward selection are used to extend the document list. However, from the literature selected during the primary search, it has become clear that the Plattform I4.0 and its partners steer the research on the AAS. The Plattform I4.0 has its own database with publications on their advances on the AAS. This database is therefore included in the secondary literature search.

Extract and synthesize data
In the end, the relevant data is subtracted from the documents selected in the primary and secondary literature search (after screening). The results are described in detail in (Pepels, 2019). The relevant topics of the Systematic Literature Review (Pepels, 2019) are selected and explained in the next sections of this chapter.

[1] https://ieeexplore.ieee.org/Xplore/home.jsp
[2] https://www.sciencedirect.com/
[3] https://link.springer.com/
[4] http://apps.webofknowledge.com/
[5] https://dl.acm.org/dl.cfm?coll=portal&dl=ACM/

2.2 Industrie 4.0

The Asset Administration Shell is the central topic in this Master Thesis research project. However, on the basis of the AAS is Industrie 4.0. A common understanding of the concept I4.0 must be established before the AAS is investigated in detail. As stated earlier the focus is on the definition of the concept I4.0 (section 2.2.1), the core components and technologies of I4.0 (section 2.2.2), design principles of I4.0 (section 2.2.4) and standardization in I4.0 (section 2.2.5). In the following (sub-)sections the common ground for I4.0 is built. For this part of the Systematic Literature Review, a total of 121 documents were found after the initial literature search. After applying the relevance and quality criteria 27 documents contained relevant information for establishing the common ground on I4.0.

2.2.1 Definition of I4.0

To establish the common understanding of I4.0 the first step is to define the Industrie 4.0. On this date, there is no generally accepted definition for the term I4.0. Four definitions of I4.0 were subtracted from the literature on I4.0. In this document, the definition of Industrie 4.0 from (Hermann, Pentek & Otto, 2016) is adopted. This definition contains the core components of I4.0 as is determined in section 2.2.2. Other definitions of Industrie 4.0 were subtracted from (Kagermann et al., 2013), (Oztemel & Gursev, 2018), and (Rao & Prasad, 2018). The definition of I4.0 used in this document is thus:

*'**Industrie 4.0**is a collective term for new technologies and concepts of a value chain organization. Within Smart Factories, Cyber-Physical Systems are used to monitor processes and create a virtual copy of the real world, leading to autonomous decision-making in a factory. IoT is used for the communication between CPSs and humans and IoS is used for internal and cross-organizational services offered and utilized by value chain participants.'*
(Hermann et al., 2016)

From this definition, it becomes clear that the term I4.0 is a collection of (sub)concepts and technologies. The concepts Smart Factory, Cyber-Physical Systems, Internet of Things (IoT) and the Internet of Services (IoS) are used to build this definition. Later it is discussed that these concepts are the at the core of I4.0 (see section 2.2.2).

2.2.2 Core concepts and Technologies

From the previous section, it has become clear that the I4.0 concept is a collection of (sub)concepts and technologies. An analysis must be performed to define the core concepts and technologies in I4.0. Eight articles, that contain information on the core concepts and technologies of I4.0, are selected for analysis. From the analysis of (Lasi, Fettke, Kemper, Feld & Hoffmann, 2014), (Hermann et al., 2016), (Saucedo-Martínez, Pérez-Lara, Marmolejo-Saucedo, Salais-Fierro & Vasant, 2018), (Bär, Herbert-Hansen & Khalid, 2018), (Oztemel & Gursev, 2018), and (Strandhagen, Alfnes, Strandhagen & Vallandingham, 2017) it follows that the following concepts are at the core of I4.0: the Smart Factory, Cyber-Physical Systems and the Internet of Things. The I4.0 (enabling) technologies are found and listed as: Machine Learning, Augmented Reality, Big Data, Cloud, Artificial Intelligence, and (intelligent) robotics.

2.2.3 I4.0 characteristics

The I4.0 characteristics that differentiate I4.0 from I3.0 are discussed in this subsection.

Three forms of integration

Industrie 4.0 can best be characterized by its three forms of integration, which represent the main difference between I4.0 and I3.0. In 18 of the 27 documents reviewed on the topic of I4.0, integration is mentioned as a characteristic of the Industrie 4.0. Out of these 17 documents, 14 documents reference to one of the three forms of integration of Industrie 4.0 as defined in (Kagermann et al., 2013) including this document. Eight of these articles have a clear connection to at least two of the three forms of integration. In the end, the forms of integration are described in detail in (Strandhagen et al., 2017), (Pérez-Lara, Saucedo-Martìnez, Salais-Fierro, Marmolejo-Saucedo & Vasant, 2019), (Saucedo-Martínez et al., 2018), and (Kagermann et al., 2013). From the analysis, the following can be stated about the three aspects of integration that characterize Industrie 4.0:

1. **Horizontal Integration** across the entire value creation network. Horizontal integration considers all the links in the value chain and the relationships that are developed, establishing and maintaining networks that create and add value (Saucedo-Martínez et al., 2018). It will facilitate inter-corporation collaboration where material flows fluently among these corporations. Collaborative manufacturing and collaborative development environments emerge through horizontal integration (Strandhagen et al., 2017). In a complementary way, the horizontal flow includes external relations, establishes supplier and customer networks integration, information and management systems (Pérez-Lara et al., 2019). Models, designs and implementations of horizontal integration through value networks should provide an answer to the question: 'How can companies' business strategies, new value networks and new business models be sustain-ably supported and implemented using CPS?' (Kagermann et al., 2013).

2. **Vertical Integration:** the main purpose is to make a factory work intelligently with its products and production processes through Cyber-Physical Systems (Saucedo-Martínez et al., 2018). Vertical integration concerns the integration of various UT systems at different hierarchical levels of a factory (Strandhagen et al., 2017). Strandhagen et al. (2017) refers to Wang et al. (2016) which explain the integration of the automation pyramid, from the sensors and actuators of the shop floor up to the Manufacturing Execution System (MES) level and Enterprise Resource Planning (ERP) level. Pérez-Lara et al. (2019) state that company performance depends on the synergy level of the company. The question to answer for vertical integration is: 'How can CPS be used to create flexible and reconfigurable manufacturing systems?' (Kagermann et al., 2013).

3. **End-to-end Integration** across the entire value chain. It allows to systematically analyze all the data obtained throughout the production process and allows for quick decision making (Saucedo-Martínez et al., 2018). End-to-end integration (engineering) supports the increasing requirements regarding product customization and includes cross-linking of stakeholders, products and equipment along the product life cycle (Strandhagen et al., 2017). In (Kagermann et al., 2013) it is stated that end-to-end digital integration throughout the engineering process so that the real world and digital world are integration along the product entire value chain across different companies while fulfilling customer requirements. The main question to answer according to Kagermann et al. (2013) is: 'How can CPS be used to deliver end-to-end business processes including the engineering workflow?'. Modelling plays a key role in managing the complexity of (new) technological systems.

The three forms of integration are visualized in figure 2.2 (Wang et al., 2016).

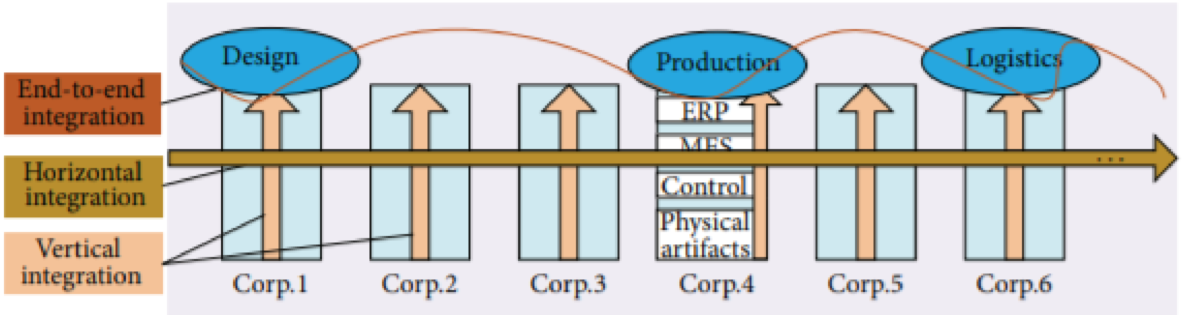

Figure 2.2: Three forms of integration of I4.0

I4.0 networked system architecture
The three forms of integration can be reached by the use of Cyber-Physical Systems as becomes clear from the explanations above. The introduction of Cyber-Physical Systems has had its effects on the way systems are organized. The essence of the three forms of integration can be visualized in the system architecture. The system architecture for I4.0 systems differs from I3.0 systems. I3.0 systems are in general structured hierarchically, whereas I4.0 systems are networked through its CPS. The difference in system architecture is shown in figure

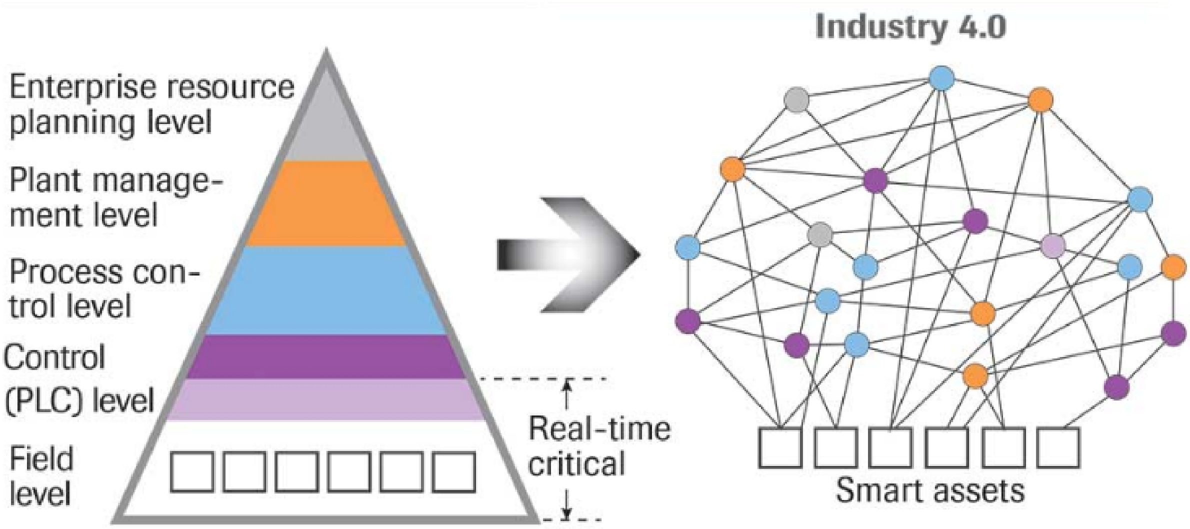

Figure 2.3: Comparison I3.0 (left) and I4.0 system architecture (right)

2.2.4 Design Principles of I4.0

The focus of the Master Thesis project is on the design of Asset Administration Shells. For design in I4.0, the following design principles have been established. The design principles defined by Hermann et al. (2016) are (almost) generally accepted. The design principles from Hermann et al. (2016) are also assumed to be leading in this research. The six design principles for I4.0 are:

1. **Interoperability:** all CPS within the plant are able to communicate with each other 'through open nets and semantic descriptions'. Interoperability is a very important enabler of I4.0 because it connected companies and CPS over the IoT and IoS.
2. **Virtualization:** CPSs are able to monitor physical processes. A virtual copy of the physical world is created. The data obtained via sensors is linked to virtual plants models and simulation models. A virtual copy of the real world is created.
3. **Decentralization:** embedded computers enable CPS to decide on their own. This supports more individualized production because this makes it harder to control systems centrally.
4. **Real-time capability:** data is collected and analyzed in real-time. Information about the processes is obtained at all times and decisions can be made based on this data.
5. **Service orientation:** services of companies, CPSs, and humans are available over the IoS and can be utilized by other participants. Services can be offered within and outside company borders.
6. **Modularity:** modular systems are able to flexibly adapt to changing requirements by replacing or expanding individual modules. This design principle also follows the goal of individualization of I4.0.

2.2.5 I4.0 standardization

Standardization plays a major role in I4.0 since it enables full integration of systems, which is the key characteristic of I4.0. Also, the design principles of I4.0 suggest that standardization efforts are made. For example, the I4.0 design principle 'interoperability' (see the previous section) can only be achieved if CPS can communicate. For this, a key success factor is the standardization of CPS of various manufacturers (Hermann et al., 2016). At the basis of standardization are reference architectures. The first and most widely accepted reference architecture is the Reference Architectural Model Industrie 4.0 (RAMI4.0). This RAMI4.0 is at the basis of the I4.0 Component and Asset Administration Shell and is therefore explained in detail in this section.

RAMI4.0 is a three-dimensional model created to group highly diverse aspects in a common model. Its goal is to achieve a common understanding of what use cases, standards etc. are necessary for I4.0. It represents the three forms of integration that characterize I4.0: horizontal integration, vertical integration and end-to-end integration (engineering) (Peter Adolphs et al., 2015). The model controls (dynamic) cooperation between factories in a common added value network and unites the fundamentals of different application domains. Figure 2.4 shows the RAMI4.0. The remainder of this subsection explains RAMI4.0 in detail.

The three axes of the RAMI4.0 architecture are the RAMI4.0 layers, the hierarchy levels, and the life cycle and value chain axis. The **hierarchy layers of RAMI4.0** describe the functional classification of assets in Smart Factories. It is built based on IEC 62264 and IEC 61512. Most important in I4.0 compared to I3.0 is that the connected world and production layers are added.

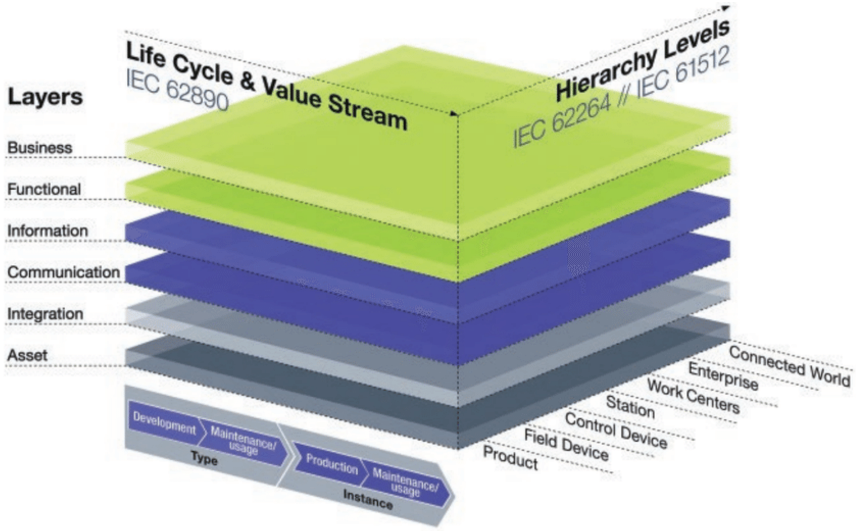

Figure 2.4: RAMI4.0

The RAMI4.0 layers are: business, functional, information, communication, integration and asset. The business layer is responsible for amongst other things mapping the business models, legal and regulatory framework conditions, and the link between business processes. The functional layer provides things such as a formal description of functions and a platform for horizontal integration of functions. The information layer is concerned with the execution of event-related rules, provides a formal description of rules and a run time environment for the processing of events. The communication layer standardizes communication and provides services for control of the integration layer. The integration layer provides information on the assets. The asset layer represents the reality, humans are also included in this layer.

The third axis is the **life cycle and value stream axis**. The life cycle and value stream axis in RAMI4.0 is concerned with the improvement of products, machine and factories during their life cycles. IEC 62890 is a guideline used for the life cycle and value stream. For this, a distinction has to be made between types and instances. A type is created in the development phase of a product. An instance is a single product (for example) of a certain type. The value stream part of this axis of RAMI4.0 offers huge potentials for improvements. The life cycle has to be viewed together with the value-adding processes that it contains. Processes such as purchasing, order planning, assembly, logistics, maintenance, the customers and suppliers need to be linked.

Other reference architectures are created for I4.0. For example the 5C level architecture (Lee et al., 2015). This architecture is focused on the development and deployment of CPSs. The ISA-95 architecture is used for developing an automated interface between enterprise and control systems. ISA-95 is an international standard. Besides that, another 3D architecture is designed called the Industrial Internet Reference Architecture (Lin et al., 2015). The focus in this project is on RAMI4.0 because it is on the basis of the I4.0 component and Asset Administration Shell.

2.3 The Asset Administration Shell

The focus of this Master Thesis research is on the Asset Administration Shell. In the previous section, a common knowledge-base is established for the surrounding concept: Industrie 4.0. Now, the focus is shifted towards the Asset Administration Shell itself.

At first, it might be convenient to formally state the definition of the AAS used in this document. In (almost) all literature on the AAS a definition of the AAS is stated. Nonetheless, the most appropriate definition of the AAS for this research comes from the Plattform I4.0, which have conceived the AAS concept. Their definition of the AAS is as follows:

'a virtual digital and active representation of an I4.0 Component in the I4.0 system'[6].

For more clarity, the definition needs further explanation. The AAS is virtual because it stores data and functions of an asset virtually (as software). The AAS is active because it has the ability to actively communicate and make decisions in an I4.0 system. The AAS is part of the I4.0 Component, a topic addressed in the next subsection. An I4.0 system is defined as: *'a system, consisting of I4.0 Components and components of lower CPS classification which serves a specific purpose, has defined properties, and supports standardized services and states'*.

2.3.1 The role of the AAS in I4.0

The Asset Administration Shell is part of the I4.0 Component concept, which becomes clear from the definition of the AAS. The I4.0 Component is defined by the Plattform I4.0 as:

'a globally unique identifiable participant with communication capability consisting of an administration shell and asset within an I4.0 system which there offer services with defined QoS (quality of service).'[6]

This definition states that the I4.0 Component consists of the AAS and an asset. This can be considered simplistically as a virtual and physical component. An asset is everything that adds value to an organization and does not necessarily have to be physical. Besides this, it is stated that I4.0 Components can communicate and offer services to other I4.0 Components via their AAS. The I4.0 Component is generally visualized as in figure 2.5. This figure shows an asset being covered by an AAS, to allow I4.0 communication.

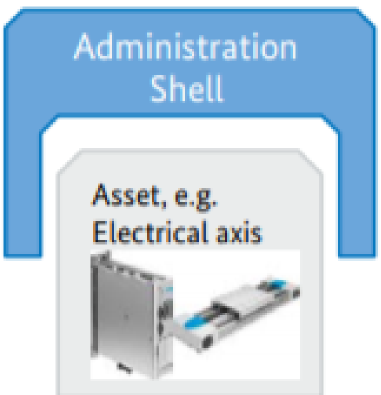

Figure 2.5: The I4.0 component

[6] https://www.plattform-i40.de/PI40/Navigation/EN/Industrie40/Glossary/glossary.html

To understand the relation of the I4.0 Component and the AAS with the manufacturing and I4.0 domain, the following is of great importance. According to Hoffmeister (2015) the I4.0 Component is *'a model for describing in more detail the properties of Cyber-Physical Systems - real-world objects in a production environment networked with virtual objects and processes'.* This highlights the fact that I4.0 Component can be seen as a standardized Cyber-Physical System. Hoffmeister (2015) also state that the I4.0 Component enables Smart Factories to become reality and should be prioritized. Cyber-Physical Systems merge the physical with the virtual world (Alcácer & Cruz-Machado, 2019). The AAS is the virtual part of the CPS and can be seen as the **software in I4.0** that enables the connection of the physical and virtual world. This connection enables the three forms of integration to be achieved.

2.4 The AAS information model

The design of the AAS is based on the information model defined for the AAS. This information model is mainly described in (Plattform I4.0, 2018a) and (Platform-I4.0, 2019). In this subsection, the details of the information model of the AAS are described. In section 2.4.1 the general structure at the basis of the AAS information model is explained. In section 2.4.2 the most important aspects of the information model for the design of AAS are elucidated.

2.4.1 The general structure of the AAS

At the basis of the information model for the AAS is the general structure of the AAS. The Plattform I4.0 is the leading organization in the development of the Asset Administration Shells. In (Adolphs et al., 2016) the requirements for the Asset Administration Shell have been defined. Based on these requirements, the Plattform I4.0 has defined a general structure of the Asset Administration Shell (Adolphs et al., 2016). This general structure is depicted in figure 2.6 and shows a logical overview of the AAS. To verify this general structure of the AAS the structure is discussed in an international paper (Plattform I4.0, Alliance Industrie du Futur & Piano Nationale Impresa I4.0, 2018). No changes are made to the previously defined structure of the AAS in (Adolphs et al., 2016). Other publications from the Plattform I4.0 use the general structure from figure 2.6 as basis for the AAS.

An alternative structure of the AAS is defined by Tantik and Anderl (2017a). This structure consists of six segments: an external interface, authentication and security, data management, functionality, administration, internal interface. The alternative structure from Tantik and Anderl (2017a) is not used throughout this document for the simple reason that the Plattform I4.0 is the leading organization and creator of the AAS, and use their general structure as the basis for almost all of their publications.

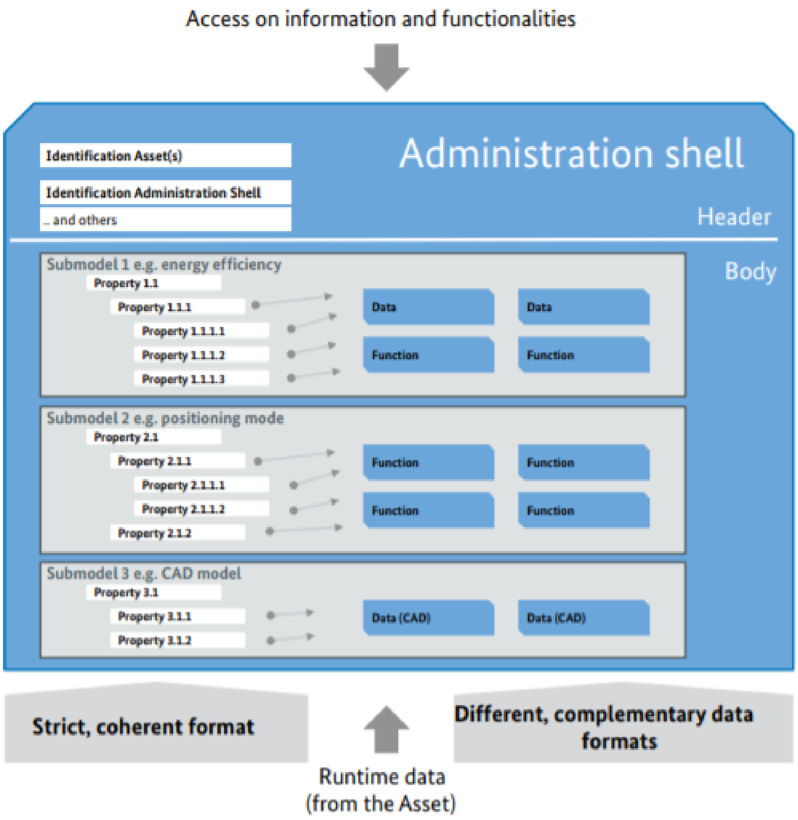

Figure 2.6: General structure of the Asset Administration Shell defined by the Plattform I4.0

2.4.2 Important aspects of the AAS information model

The AAS information model is the information model based on the general structure of the AAS (section 2.4.1). The details of the (meta-)information model of the AAS are mainly discussed in (Plattform I4.0, 2018a) and (Platform-I4.0, 2019). In Appendix A.2 the meta-information model of the AAS is shown. From the meta-information model the AAS aspects relevant to the design of AAS are identified. The AAS aspects and if they are mandatory or optional are shown in table 2.1. Thereafter, the AAS aspects are clarified.

AAS Aspect	Mandatory or optional?
Identification	Mandatory
Security	Mandatory
Views	Optional
Concept dictionaries	optional
Submodels and properties	Mandatory

Table 2.1: AAS aspects

Identification

In (Adolphs et al., 2016) it stated that unique identification is required for different entities in an I4.0 system. Identification is fundamental for the design of AAS (Plattform I4.0, 2018a). In both (Plattform I4.0, 2018a) and (Platform-I4.0, 2019) identification of entities is addressed. Identifiers are needed for the unique identification of entities in I4.0 (Adolphs et al., 2016). Globally unique identification is required for: AAS, assets, submodels, and property definitions. For this two globally unique identifiers can be used: IRDI (ISO 29002-5, ISO IEC 6523 and ISO IEC 11179-6) as identifier scheme for properties (Plattform I4.0, 2018a). For the AAS, assets and (non-standardized) properties URI must be used. Another option is to define custom identifiers. In Table 2 of (Plattform I4.0, 2018a) the identifiers for all the identifiable elements of the I4.0 Component are described, a replica of this table can be found in Appendix A.3.

Security

Security of the AAS is important because information is exchanged between value chain partners. In DETAILS the attribute-based access control is used for the security for the exchange of information. The general principle of attribute-based access control (ABAC) and is visualized in figure 2.7 (Hu et al., 2013).

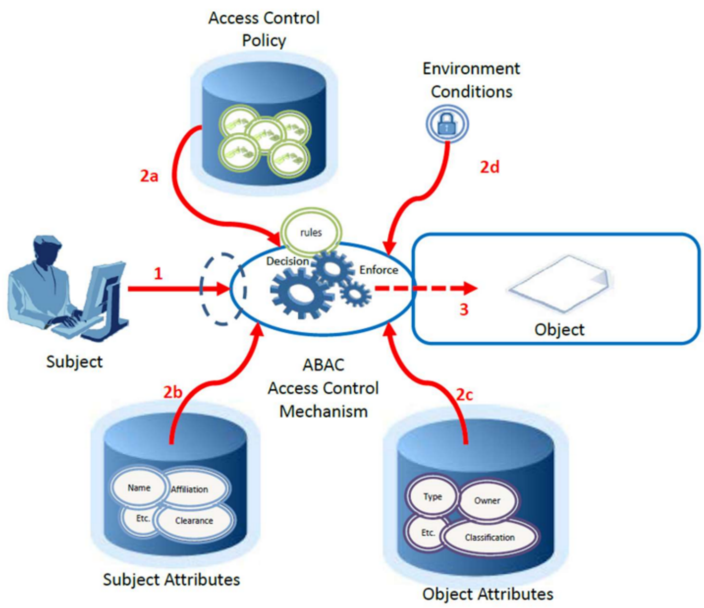

Figure 2.7: Attribute Based Access Control (ABAC) (Hu et al., 2013)

A subject can request a submodel or properties from an asset (object). The access control mechanism evaluates the access permission rules with relation to the subject attributes, object attributes and environment conditions. If the subject is authorized by passing the permission rules, the subject gains access to the object.

Views

AAS contain a large number of properties and not every property of an AAS is relevant to all users or each system. This is why properties can be linked to a view. In DIN SPEC 91345 (RAMI4.0) (Spec, 2016) basic views for the AAS have been defined. The basic views are business, constructive, performance, functional, local, security, network view, life cycle, and human. In appendix A.4 table 2 of (Bedenbender et al., 2018) is duplicated, which shows the best practices/examples of the basic views.

Concept dictionaries

Concept dictionaries are defined as optional in the meta-information model for the AAS (Plattform I4.0, 2018a). Concept dictionaries for AAS contain semantic definitions of its submodel elements. This concept dictionary might contain copies of property definitions of external standards. A concept dictionary contains concept descriptions. A concept dictionary is optional for an AAS.

The concept of submodels
From the general structure (figure 2.6) it can be seen that the body of the AAS consists of submodels. The notion of submodels is important for the understanding of the AAS. Submodels represent different aspects of the relevant asset (Bedenbender et al., 2018). The aim is to standardize one submodel for each individual aspect or technical domain. This way submodels complement each other without interfering. In (Plattform I4.0, 2018a) a submodel is defined as: 'a submodel is used to structure the virtual representation and technical functionality of an Administration Shell into distinguishable parts. Each submodel refers to a well-defined domain or subject matter. Submodels can become standardized and thus become submodel types. Submodels can have different life cycles'.

Submodels are being standardized and therefore we have to distinguish between submodels that have been standardized and that are not. Throughout this document, the term basic submodel is used when speaking of a standardized submodel. The term free submodel is used when this is not the case. In (Platform-I4.0, 2019) a distinction is made between five types of submodels. In this document, the distinction between basic and free submodels will suffice. However, it is worth mentioning the distinction according to (Platform-I4.0, 2019), which is as follows:

- **Basic mandatory submodels** are independent for all AAS and mandatory for all assets.
- **Basic optional submodels** are independent and usable for all AAS, but are optional.
- **Mandatory submodels for an asset class** are submodels that are mandatory for assets of a certain class.
- **Optional submodels for an asset class** are submodels that are optional for assets of a certain class.
- **Free submodels** are optional submodels for an asset and are defined by (an) industry partner(s).

It is explained that submodels describe a specific functionality or aspect of an asset. The submodels describe the data and functions of an asset. Many international standards, consortium specifications and manufacturer specifications are already contributing to the description of submodels (Plattform I4.0 et al., 2018). These standards aim to describe and define the so-called properties of the submodels. Example submodels and their link to standards can be found in Appendix A.5. Hereafter, the concept of properties is described.

The concept of properties
Properties are the building blocks of the submodels, which aggregate information of a specific aspect of an asset. Properties refer to submodel elements which can be data elements, operations, relationship elements, events or submodel element collections (Plattform I4.0, 2018a). Data elements can be further divided into blobs, files, properties and reference elements. The term properties will be used throughout this document to refer to (standardized) submodel elements. Properties in I4.0 must be defined according to IEC 61360. Definitions of properties are documented in repositories such as IEC CDD and eCl@ss and follow a strict, uniform format (IEC 61360). The definition of properties is still an ongoing process.

2.5 AAS communication

I4.0 Components have the ability to communicate with each other. Three types of AAS can be defined each having a different role in the Value Chain and other communication capabilities (section 2.5.1). For the communication of AAS, the Plattform I4.0 and its partners have developed an I4.0 language (section 2.5.2).

2.5.1 Implementation variants of the AAS

The PLattform I4.0 defines three implementation variants of the AAS, which each use different types of communication. These implementation variants of the AAS are (Platform-I4.0, 2019):

- **Passive AAS in file format:** the AAS is described in XML or JSON format. It offers a standardized form to make information about an asset available to authorized users in an I4.0 system.
- **Passive AAS with IP/API access:** contains the same information as the AAS in file format in general but is made available via an interface.
- **Active AAS:** can participate in protocol-based interactions and follows the I4.0 language as defined in (Plattform I4.0, 2018b). This type of AAS enables (semi-)autonomous decision-making.

The difference between the implementations variants can best be visualized by placing them in the RAMI4.0 layers as is shown in figure 2.8 (Belyaev & Diedrich, 2019). As can be seen, the AAS as file format can only make properties of an asset available to its environment. The passive AAS (with IP/API access) has the ability to make functions available to its environment, but cannot make autonomous decisions. The third variant, active AAS, can make their own decisions and initiate active communication.

Besides these implementation variants, which determine how the AAS is used in an I4.0 system, one must define the location of the AAS. Platform-I4.0 (2019) defines three options for the location of the AAS: on the asset itself, on an edge device, or in the cloud. By default, AAS are stored in the cloud. However, it is possible to store parts of the AAS on the asset or an edge device and other parts in the cloud.

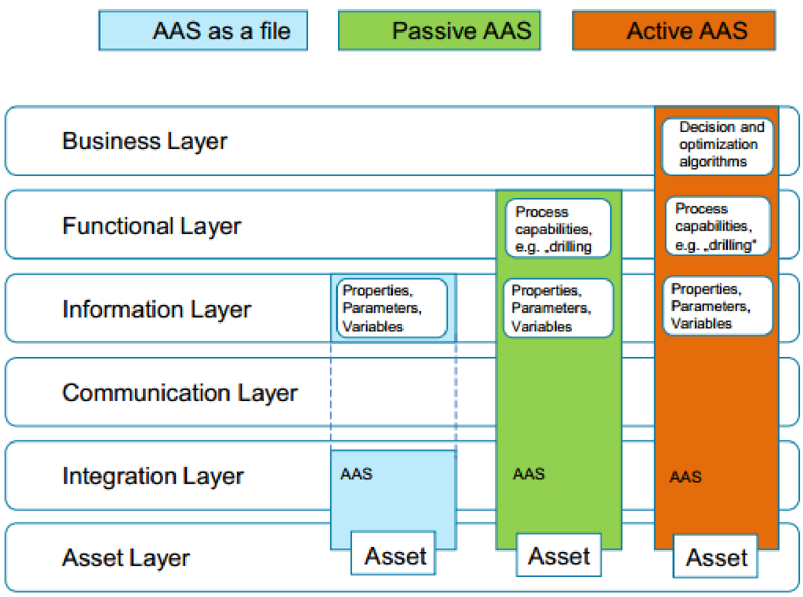

Figure 2.8: The AAS implementation variants mapped on the RAMI4.0 layers.

2.5.2 The I4.0 Language

The communication between I4.0 Components (and thus AAS) is standardized. The Plattform I4.0 has therefore defined an I4.0 language (Plattform I4.0, 2018b). At the basis of this language is the interaction model defined in (Diedrich et al., 2017). The I4.0 language consists of three levels as depicted in figure 2.9: a semantic interaction protocol, structure of messages, and vocabulary for I4.0 language. In VDI/VDE 2193-1 and VDI/VDE 2193-2, the guidelines for the I4.0 language are described. Note that the vocabulary of the I4.0 language is recorded in the submodels (and properties) of the AAS.

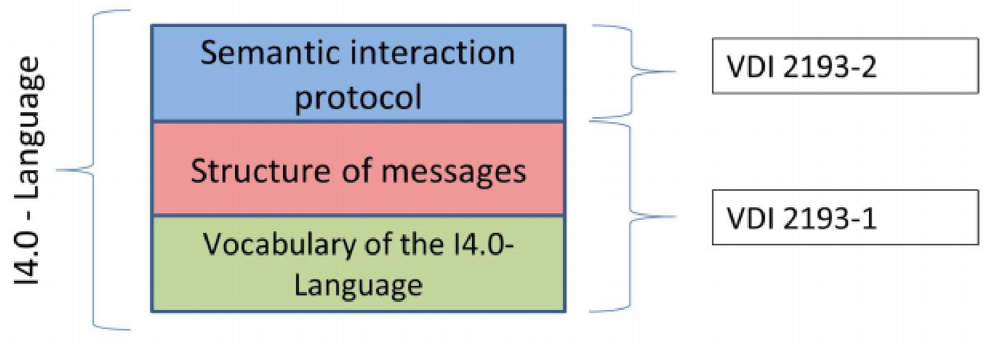

Figure 2.9: The I4.0 language

For the different implementation variants of the AAS, this means that the first two variants (file format and IP/API-access) only require the I4.0 vocabulary to be followed. For active AAS, all three levels of the I4.0 are relevant.

2.6 Theoretical framework

The Systematic Literature results in a theoretical framework to show the relationships between the key concepts in this Master Thesis project. The theoretical framework is visualized in figure 2.10. The relevant aspects of the AAS for the design of networks of AAS are the information model, communication and the system architecture. The underlying concepts of these aspects are the I4.0 design principles and RAMI4.0.

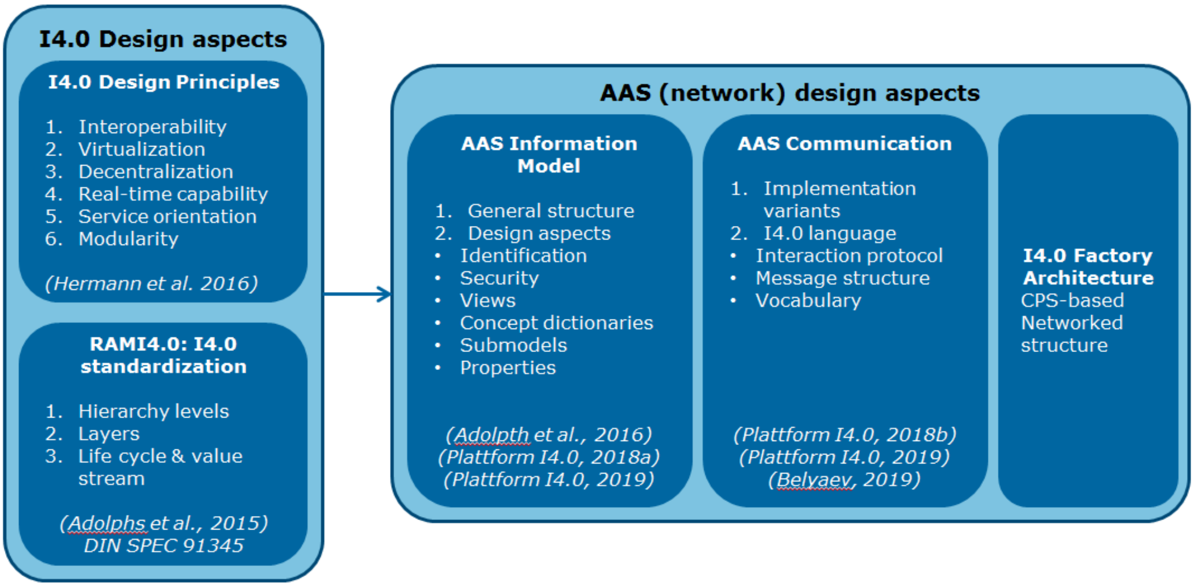

Figure 2.10: The Theoretical Framework

2.7 Conclusions

The knowledge base for this Master Thesis project is established using a Systematic Literature Review. The SLR is focused around two main concepts: I4.0 and the Asset Administration Shell. At first, a definition of I4.0 and its core concepts are determined. Thereafter, the main characteristics of I4.0 in comparison with I3.0 are stated. This roughly comes down to three forms of integration: horizontal, vertical, and end-to-end integration. These forms of integration can be reached through Cyber-Physical Systems. The use of Cyber-Physical Systems causes the system architecture to change drastically. The hierarchic architectures are replaced by network system architectures. The answer to sub-research question 1 *'What are the key characteristics of I4.0 compared to I3.0?'* is herewith given. Besides that, standardization and I4.0 design principles are explained, which are the fundamental aspects of I4.0 design.

Secondly, the AAS concept is analyzed in detail. At first, its role in I4.0 is elucidated. The AAS is the software part of the I4.0 Component concept, which is a way to model CPS. For the design of this software, the following aspects are of relevance: its information model and its communication. The information model for the AAS is based on its general structure and comprehends the following: identification, security, submodels, properties, views and concept dictionaries. The second aspect is the communication between AAS. Different types of communication are possible based on the role of the AAS in the value chain. For the communication between AAS, the I4.0 language is specified in VDI/VDE 2193. The communication requires an interaction protocol, message structure and standard vocabulary. This all answers sub-research question 2: *'What aspects of I4.0 and the AAS are relevant to the design of networks of AAS?'*.

Chapter 3

Problem Investigation

This chapter describes the Problem Investigation phase of the Design Science Methodology. The Problem Investigation phase is a preparation for the Artifact Design. The goal of the Problem Investigation phase is to understand the problem to be solved by the artifact. From the Problem Investigation phase, the requirements and constraints for the artifact must become clear for the artifact design phase. The problem investigation is real-world research and therefore an **observational case study** is chosen as the research method. This chapter aims to answer the question: *'How are AAS being created currently?'*. The result of the Problem Investigation are requirements, constraints and restrictions for the artifact must be stated.

3.1 Problem Investigation Context

Atos is a global IT services company and is the leader in digital transformation as described in section 1.1. Atos works with clients to digitize their firms. It has become clear that I4.0 is at the center of digital transformation for every manufacturing enterprise. Atos wants to help these enterprises with their Consult-Build-Run services (as explained in section 1.1). Currently, Atos is not providing services in the field of the I4.0 component and the Asset Administration Shell. This is mainly caused by the fact that the research in this field is not yet completed. Most Atos employees are therefore not familiar with the terms I4.0 component and Asset Administration Shell. However, most employees are familiar with strongly related technology, the Digital Twin. The Asset Administration Shell can be seen as the I4.0 implementation of an advanced standardized Digital Twin.

Atos has determined that the I4.0 component and Asset Administration Shell offer great opportunities. As a service provider, Atos is looking for a new service in this field. At the basis of a new service for I4.0 components are the awareness of the new concept, the understanding of the I4.0 component, and a proposal for the service to offer. Atos has pronounced its desire for a method to offer a new service concerning the I4.0 component and its Asset Administration Shell for their MRT (Manufacturing, Retail and Transport) clients.

The I4.0/IoT unit of Atos NL has not yet determined an approach for the design of networks of AAS. This is due to the fact that there is no implementation of the AAS or an I4.0 demonstrator, within Atos NL. To investigate the current state of the AAS network design process, cases from other institutions must be observed. The cases are analyzed using the documents published by the institutes.

3.2 Observational case study

For the Problem Investigation phase of the Design Science Methodology, an observational case study is performed. An observational case study is a study of a real-world case without performing an intervention (Wieringa, 2014). On this date, there is no such thing as a method for the design of Asset Administration Shells. This can be explained by the fact that the I4.0

Component and Asset Administration Shells are relatively new and are still a work-in-progress. For the creation of such a method, it is beneficial to investigate the projects in which Asset Administration Shells are implemented.

3.2.1 Objectives and research questions

The objective of the observational case study is to identify what the current process of designing AAS looks like. The Problem Investigation phase of the Design Science Methodology requires an accurate description of the 'current process'. Therefore, the observational case study aims to answer the third sub-research question: *'How are networks of AAS designed currently?'*. This way the shortcomings of the current design processes can be determined. The result is a gap in the current state and the desired artifact.

3.2.2 Selection of cases

The cases selected here are the available projects in which Asset Administration Shells are implemented. No case is available at Atos NL unfortunately, so cases from the literature were selected consequently. The selected cases are all described in the literature obtained during the Systematic Literature Review (Chapter 2). This case base is extended by backward and forward selection of documents in the existing case literature. The result is a long list of potential cases to be analyzed in the observational case study, which can be found in Appendix B.1.

Figure 3.1 shows the three cases and the unit of analysis of these cases. The three selected cases are: the SmartFactory^{KL}, the I4.0 Bottling demonstrator from Siemens and Festo, and the Plattform I4.0 demonstrators are HMI 2018 and 2019. The 2019 HMI demonstrator is an extension of the 2018 demonstrator. These three cases are selected from a long list of eight potential cases. The unit of analysis in the cases is the AAS network design process.

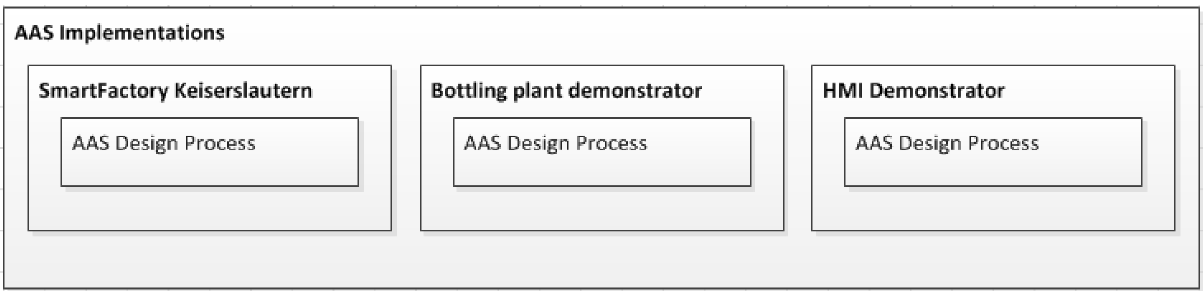

Figure 3.1: Selected cases in their context

3.2.3 Data collection method

The method to gather information on the cases is **document analysis**. This is the only possible data collection method because the Problem Investigation examines cases outside of Atos NL. From the four selected cases, the current state of the design process of (networks of) AAS must be established. This can be done by analyzing the relevant documents of the cases, which are shown in Appendix B.2. The data from the documents are analyzed using a coding scheme. Analysing documents incorporates coding content into themes similar to how focus group or interview transcripts are analysed (Bowen, 2009).

The codes for the analysis are based on the Theoretical Framework derived from the Theoretical Background. The unit of analysis for the cases is the AAS (network) design process. The relevant aspects for the design of networks of AAS have been determined in the Theoretical

Background. The final codes for the document analysis can be found in figure 3.2. The codes are used to analyze the design processes of the specific case.

Category	Codes	Description
RAMI4.0	RAMI4.0 life cycles, layers, hierarchy levels	At the basis of the definition of the content of the AAS is the RAMI4.0 architecture.
I4.0 System Architecture	Set-up, architecture, network	For the design of systems, the system architecture is important
AAS information model	Submodels, properties, identification, security, (views), (concept dictionaries)	The information model (based on RAMI4.0) of the AAS is at the basis of its design.
AAS communication	Interaction protocol, messages	AAS communication is crucial in the design of networks of AAS

Figure 3.2: Codes and categories of document analysis

3.2.4 Case descriptions

For a better understanding of the analysis of the cases, the cases are first described shortly.

The SmartFactory^{KL}

The SmartFactory^{KL} demonstrator might be one of the most well-known demonstrators of I4.0. The SmartFactory^{KL} has created an Industrie 4.0 production plant with its consortium partners. The concepts of I4.0 are advanced in this demonstrator factory. The production plant consists of a modular cell, a Flexible Transport System (FTS), a supply infrastructure, a manual work-station, edge devices, 5G network, integrated IT systems, and clouds. The modular cell consists of the following modules: storage, bottom engraving, clip mounting, force fitting, laser marking, weighing and quality control. The integrated IT systems are an ERP system, engineering system, data analytics system, a modular certification, and an integration layer (in this case: an IoT platform). In figure 3.3 the current set-up of the SmartFactory^{KL} is shown.

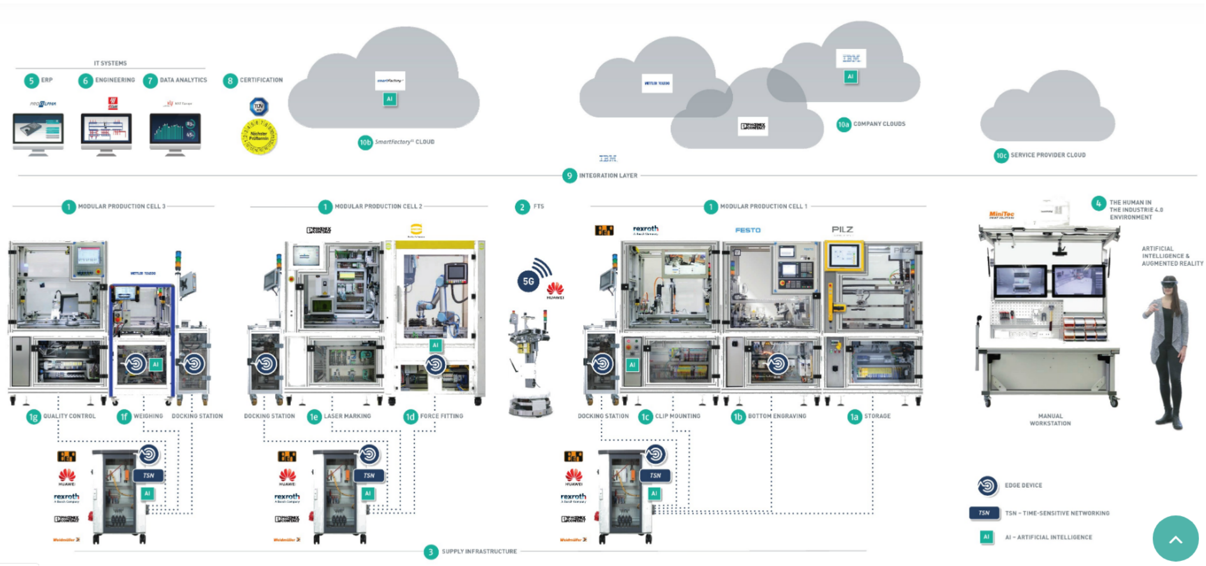

Figure 3.3: Set-up of the SmartFactory^{KL}

Plattform I4.0 Demonstrator at Hannover Messe Industrie
For the Hannover Messe Industrie 2018, the Plattform I4.0 created a demonstrator to show the implementation of AAS with their submodel and submodel elements (properties). The demonstrator showed a simple condition monitoring scenario for a conveyor belt. The AASs were responsible for the visualization, aggregation and calculation of process data. The demonstrator consisted of a mechanical conveyor belt, a positioning system, distance sensors (optic and ultrasound), two deflectors and an IoT edge gateway. The submodels created for this demonstrator were condition monitoring, asset documentation, asset identification, and a technical datasheet. The process data from the AAss were used to monitor the condition of the conveyor belt. The case is described in (Platform-I4.0, 2019). The set up of the demonstrator is shown in figure 3.4.

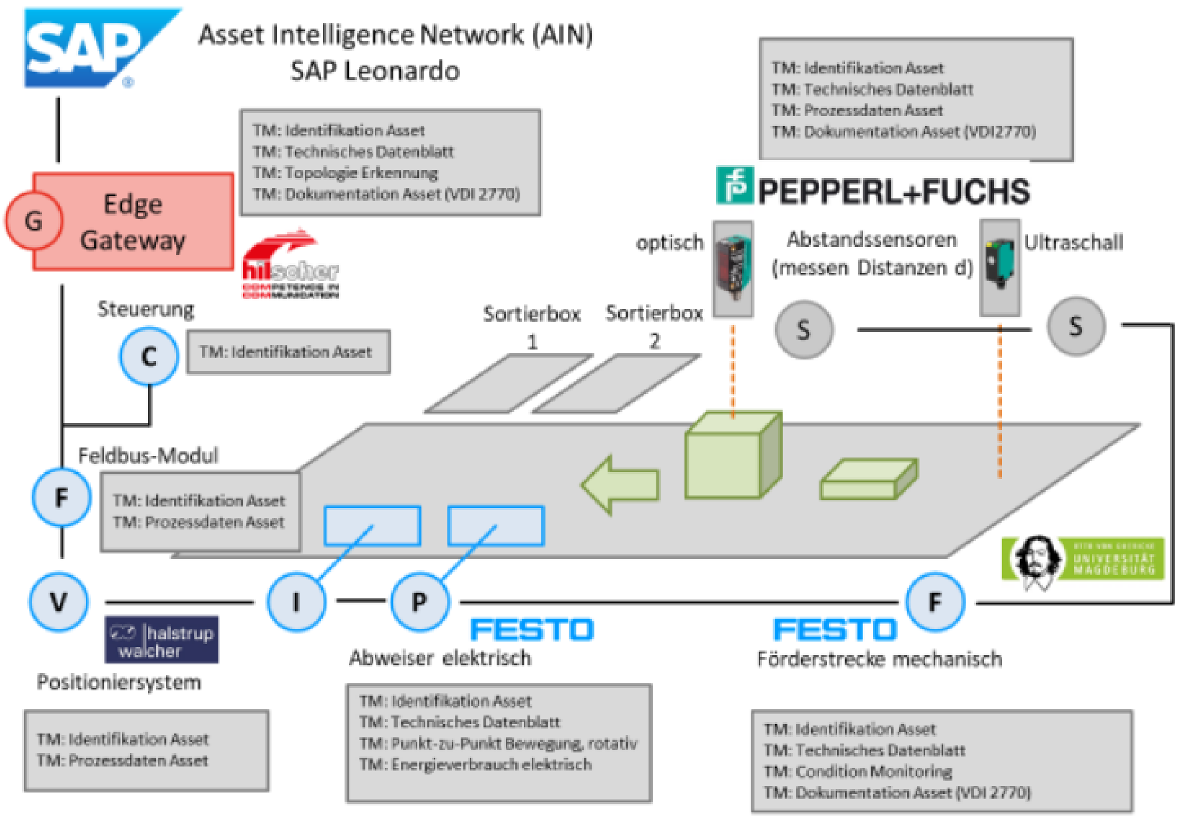

Figure 3.4: HMI 2018 Plattform I4.0 set up

The Plattform I4.0 is later extended for the Hannover Messe Industrie 2019 with AAS interactions. The interaction scenarios are described as 'picking and 'negotiations with the customer'. The demonstrator now deals with the commissioning of orders and is responsible for negotiations with the customers via a bidding process. For this case, the technical aspects of I4.0 communication are described such as the semantic interaction protocol and message structure. This case is as told an extension of the demonstrator at HMI 2018 and is also described in (Platform-I4.0, 2019).

VDI/VDE I4.0 Demonstrator
This I4.0 Demonstrator is described in (Löwen et al., 2016). A flexible transportation system from Siemens, called the multi-carrier system, is created for a bottling manufacturing cell. The demonstrator consists of one production line on which three operations are performed on (shampoo) bottles. The three operations - filling, capping and labelling - are carried out by processing units. The transportation system at the basis of the manufacturing cell consists of motor plates

and carriers. This transportation system is flexible and can move in both directions. The demonstrator is adaptable, monitors energy consumption and movement information of carriers, and is commissioned virtually before its commissioning in reality. In figure 3.5 the set-up of the example is shown.

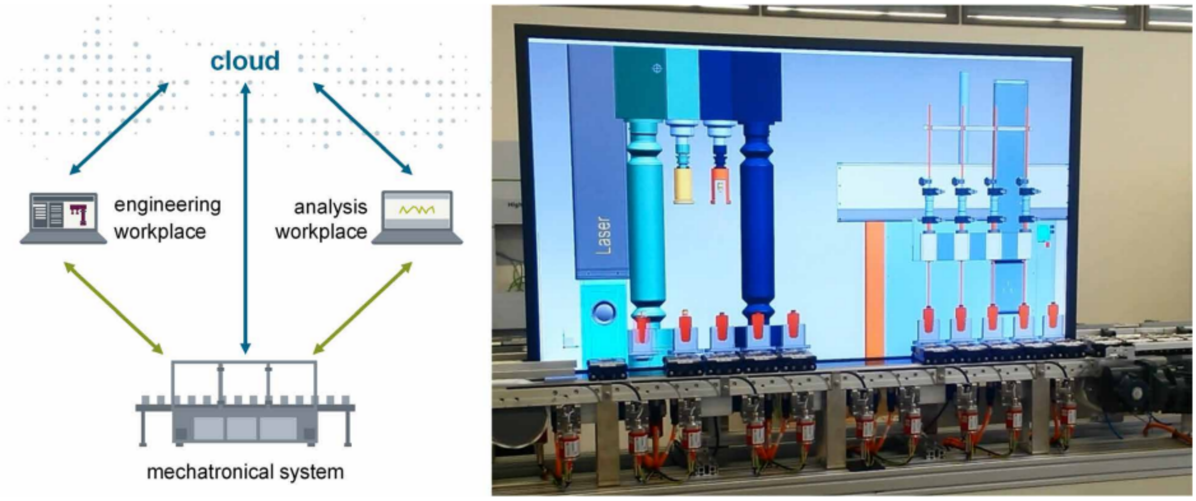

Figure 3.5: The demonstrator as described in (Löwen et al., 2016)

3.3 Results

The results of the observational case study are discussed now. The analysis of the current design processes used to create AAS is based on the design-related aspects of the AAS defined in Chapter 2. First, it is described how the cases address these aspects in their documents. Then, the flow of activities (process) of the design of AAS for the cases are determined. In the end, shortcomings of the current design processes are identified and a generalization of the design process is determined.

3.3.1 Design aspects

The design processes in the cases are analyzed by checking how they address the design relevant aspects of I4.0 and the AAS defined in the theoretical framework. It is examined how the cases address RAMI4.0, the I4.0 system architecture, the AAS information model, and AAS communication in their documents.

RAMI4.0

As is stated in the Theoretical Framework RAMI4.0 is at the basis of the design of the AAS. Therefore it is analyzed how the cases use the RAMI4.0 standards and constraints in their design process. (1) The SmartFactory^{KL} maps their production system to the RAMI4.0 architecture (Marseu, Kolberg & Weyer, 2017). The hierarchy of RAMI4.0 is used to represent the granularity of the system. An example of the mapping of a production module to the RAMI4.0 layers is shown and use the RAMI4.0 life cycles to differentiate the asset types and instances. (2) The bottling plant demonstrator uses the life cycles of RAMI4.0 to determine the submodels of the assets. Places the assets in the RAMI4.0 hierarchy levels and states that the assets have to be placed in the layers of the RAMI4.0 architecture (Löwen et al., 2016). (3) For the Hannover Messe Industrie demonstrators in 2018 and 2019, the mapping of the demonstrator to RAMI4.0 is not explained.

I4.0 System architecture
The system architecture is of importance for the design of networks of AAS. (1) The SmartFactory^{KL} has defined its own system architecture and transfers it to the RAMI4.0 hierarchy layers in (Marseu et al., 2017). (2) The bottling plant demonstrator also defines the set-up of the system and the system set-up in the virtual world based on the system components. (3) The set up of the Hannover Messe Industrie demonstrators is determined. The architecture of the system is thus addressed.

The AAS information model
The internal structure and content of the AAS must be defined according to an AAS information model. (1) The SmartFactory^{KL} uses the general structure defined in (Adolphs et al., 2016) as a basis for the design of their AAS. The submodels are defined based on example SOA queries and properties are defined based on IEC 61360. The property descriptions consist of ID, name, definition, data type, unit of measurement, value list, value, expression semantic, and view. (2) Furthermore, the bottling plant demonstrator does not define the general structure of the AAS for the design of their AAS. The content of the submodels is not explained. (3) The HMI 2018 and 2019 demonstrators both use the general structure of the AAS as defined in (Adolphs et al., 2016). For the HMI demonstrator submodels and properties are defined together with their views. Basic views from DIN SPEC 91345 (Spec, 2016) are used and the submodels are defined based on properties following IEC 61360. A three-step approach is used to define submodels. The property descriptions consist of name, parent, semanticId, idShort, description, value and qualifier.

AAS communication
Communication between AAS allows business value to be created with AAS. (1) The SmartFactory^{KL} does mention communication as a key element of the AAS and states that the AAS must be equipped with interfaces to various communication standards, but does not go into more detail. (2) The bottling plant demonstrator does not consider communication between the AAS in (Löwen et al., 2016). (3) The HMI 2018 demonstrator does not address communication between AAS. The HMI 2019 demonstrator does address communication between AAS (and partners in the Value Network) in detail and specifies interaction protocols and messages structures for the exchange of information between AAS. The HMI 2019 demonstrates thus defines the AAS communication based on the I4.0 language.

3.3.2 The design process

Even though, the design processes of the cases are not explicitly defined. The design activities and the sequencing of the activities to design the AAS are obtained from the cases. In figure 3.6, the design processes of the cases are visualized. A generalized is defined based on the design processes of the cases, which is explained hereafter.

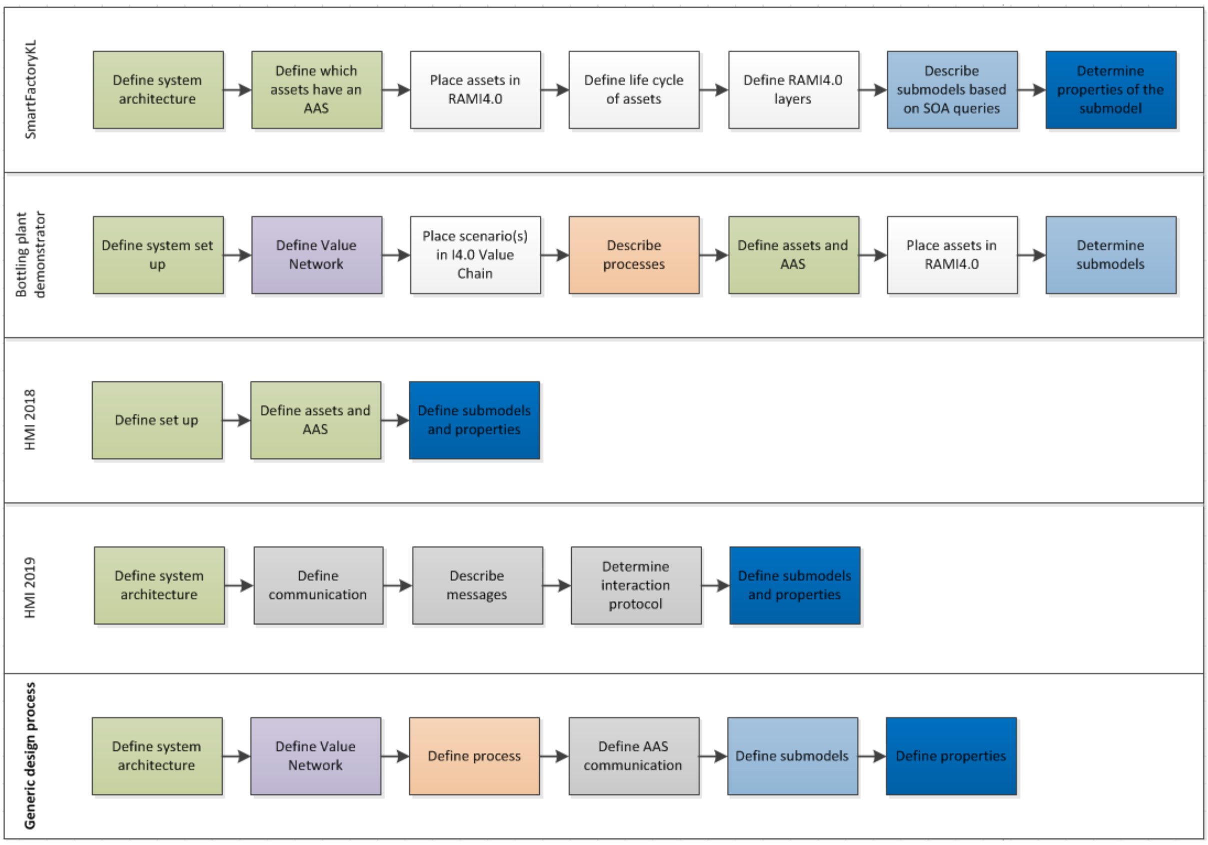

Figure 3.6: Overview of the current processes

Shortcomings of the current design process
As you can see from the analysis of the design processes explained for the three cases there is no standard consistent approach to design AAS networks. The SmartFactory^{KL} shows how they define submodel content, but they do not explain the communication between AAS. The bottling plant demonstrator describes the processes and which submodels have to be incorporated in the AAS but lacks further detail. The HMI demonstrators describe the submodels and properties of the AAS and the AAS communication formally but do not explain the process behind the definition of the submodel and property content. Only the bottling plant demonstrator shows how business requirements can be used to determine submodels, but as is stated does not detail the submodels further into properties. The approaches described for the four cases fail to define the complete design process and do not structure the process appropriately. This implies the need for a structured and complete approach to design networks of AAS. Besides that, the design processes in the documents of the four cases are not easy-to-understand. One needs a thorough understanding of all the concepts around the AAS and therefore there is a desire to make the proposed artifact (more) understandable.

A general design process
The set-up of the system and the system architecture are defined in all three cases. The definition of the value network and the processes are described only in the bottling plant demonstrator. However, these activities are relevant if you think of business requirements. Thereafter, AAS communication according to the I4.0 language is only described in the HMI 2019 demonstrator. The AAS is crucial for the AAS network design and must, therefore, be in the core activities of AAS network design. Defining the submodels and properties is explained by all three cases with the exception that the bottling plant does not in detail describe the properties of the submodels.

The core design process activities defined in the generalized design process must be fit in a structure that suits the work activities. This is taken into account in the definition of the artifact.

3.4 Conclusions

The goal of the observational case study in the Problem Investigation phase is to determine the current way networks of AAS are designed. This is done by analyzing documents from three cases, in which AAS are implemented. The analysis of the cases is based on the relevant design aspects and the sequencing of activities in the design. The relevant design aspects are obtained from the Theoretical Framework defined in 2. The aspects are RAMI4.0, the I4.0 system architecture, the AAS information model, and the AAS communication. Thereafter, the design processes described for the cases are analyzed and shortcomings are determined. In general, the shortcomings of the design processes are that the structure is mostly unclear, incomplete and not easy to understand. Based on the design processes obtained from the cases a general design process is created, the design activities in this process are: define system architecture, define the value network, define the process, define AAS communication, and in the end define the submodels and properties. With this sub-research question 3 is answered.

Chapter 4

Artifact Design

This chapter describes the artifact design process, the resulting artifact is discussed in the next chapter. Together Chapter 4 and 5 form the Artifact Design phase of the Design Science Methodology. The artifact in this master thesis project is a method for the design of networks of Asset Administration Shells called **the AASDM (Asset Administration Shell Design Method)**. The artifact design is closely connected to the other phases in the Design Science Methodology. With the design of the AAS, sub-research question 4 can be answered.

4.1 Artifact goal

The goal of the artifact design phase is to improve the current design process for the AAS. As became obvious in the problem investigation phase, the design processes and activities currently available for the design of networks of AAS lack clarity and completeness. Therefore, requirements for the artifact are that it must be understandable and complete.

Furthermore, the I4.0/IoT unit of Atos NL has stated that it wants to create a new service around the AAS in the future. This desire comes from a unit representative who has identified the lack of understanding and low awareness in the unit concerning the AAS. At the basis for this service, the unit representative has pronounced the desire for 'a method to design networks of AAS' that fits with their current Consult-Build-Run service strategy. From this, the artifact requirements usability and utility follow. Usability in the sense that it can be used by employees of the I4.0/IoT unit and thus must fit with Atos' service strategy. This means that the set up of the service must fit with other I4.0 related services Atos provides. Utility in the sense that the artifact must fulfil its purpose to be useful to design networks of AAS. The artifact requirements and stakeholder goals are summarized in figure 4.1.

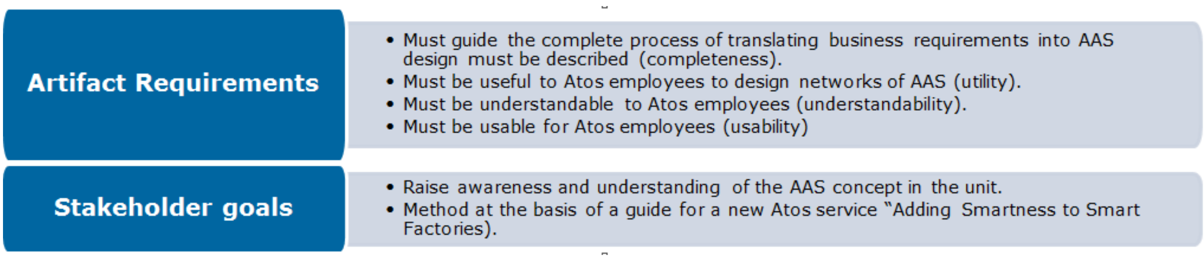

Figure 4.1: Artifact requirements and stakeholder goals

The goal of the artifact design is a method for the design of networks of AAS that is usable for the unit employees, is complete, understandable, and useful. Besides that, it must raise the awareness and understanding of the underlying AAS concept and be at the basis of a new service called "Adding Smartness to Smart Factories". The name given to the artifact is the Asset Administration Shell Design Method, in short, the AASDM. Herewith, sub-research question 4 will be answered.

4.2 Artifact design process

The artifact design process is a method engineering process. Method engineering is defined as 'Method Engineering is the engineering discipline to design, construct and adapt methods, techniques and tools for the development of Information Systems' (Brinkkemper, 1996). More specifically for the design of the artifact, an approach similar to the situational method engineering process is used. Situational Method Engineering is defined as 'a process that aims at harmonisation of methods by providing rules to configure project-specific methods out of fragments from existing (standard) methods' (Harmsen, Brinkkemper & Oei, 1994). In figure 4.2 an overview of the design process is shown based on the process defined by Harmsen et al. (1994).

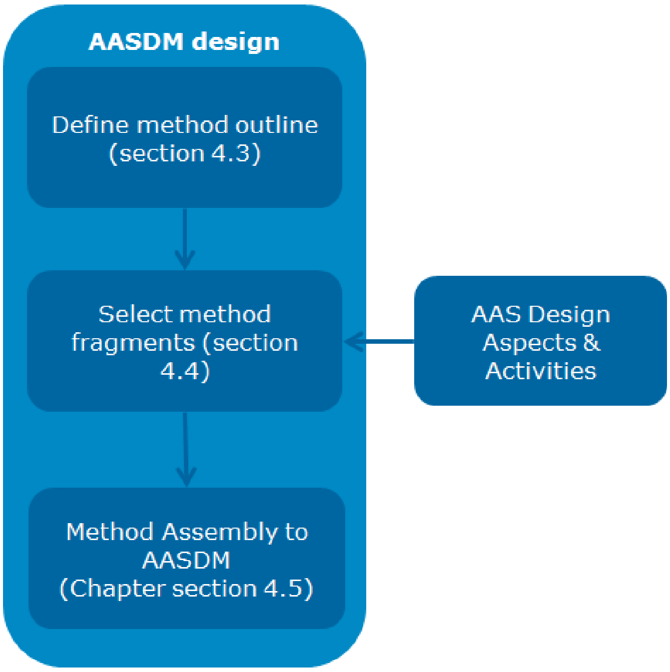

Figure 4.2: AASDM design process

The process from figure 4.2 shows the crucial steps in defining the AASDM. At first, the method outline is determined in close collaboration with a unit representative, then method fragments are selected, and thereafter assembled into the method outline. The method base consists of the AAS desgin aspects and activities derived in chapter 2 and 3.

Before the AASDM is created, the main issues faced in Situational Method Engineering approaches as defined by Harmsen et al. (1994). The first issue is the order of the Method Engineering steps. One can decide to first characterize the project and then select method fragments based on this characterization or one can first select method fragments and then start characterizing the project. The latter approach is used for the artifact design. The second issue is the selection of product-oriented versus process-oriented methods. As stated earlier the process-oriented selection is chosen for the artifact design. The third issue in Situational Method Engineering is the top-down versus bottom-up method assembly. Top-down assembly means the method outline is chosen beforehand. The bottom-up assembly means the block-by-block building of the method based on the outline. A top-down approach is chosen in this Master Thesis project. A conceptual model, method outline, is created first and thereafter the method fragments are placed within this structure.

In the following sections, the selected method engineering process is applied to create the AASDM. Artifact requirements and stakeholder goals have been defined to which the AASDM must comply. Firstly, a method outline is chosen. Then, method fragments are selected to fit within the method structure. In the end, the method is assembled into the AASDM.

4.3 Method outline

In situational Method Engineering approaches method fragments are crucial. It is worth noting that there is a special type of method fragments called the method outline, from which only the main steps are defined leaving room for further product (or process) specific aspects (Harmsen et al., 1994). The method outline, in this case, is the conceptual model of the artifact.

The method outline is created in close collaboration with an I4.0/IoT unit representative. As is stated the artifact must be usable by unit employees, this means the AASDM must be structured to be at the basis of an Atos service. Therefore, the AASDM must fit within Atos' current I4.0 services and thus be structured like a service. The service is concerned with software design and thus must incorporate requirements engineering and design from the general Software Development process from Ragunath et al. (2010). In the end, the method outline must be able to fit AAS design specific activities in the structure.

The method outline is as stated the conceptual model of the AASDM. Together with the I4.0/IoT unit representative, a structure is defined that fits all the criteria stated above. The sessions with the representative are documented in Appendix C, in which whiteboard drawings are shown and explained shortly. Requirements and design activities are based on the outline. To fit with the service structure of Atos its relation to other services is explained as follows: the method must fit with the I4.0 maturity assessment service and therefore an AS-IS and TO-BE state can be used. The method outline must also offer the ability to incorporate the currently used design activities obtained in the Problem Investigation phase.

The resulting method outline is depicted in figure 4.3. The relation with other I4.0 related services is shown in this figure. However, these services will be stated as the preparation of the method. The conceptual AASDM contains the following phases: prepare, specify the smartness, model the AS-IS situation, model the TO-BE situation, formalize the design, and test the added smartness. This structure allows all aspects of the AAS network design to be integrated into a software design (service) structure.

Figure 4.3: The method outline

4.4 Selection of method fragments

The selection of the method fragments is based on the method outline as defined in the previous section. For the AASDM this means that the main activities for each phase have to be determined. Harmsen et al. (1994) states that method fragments can be divided into two types: process and product fragments. Product fragments are the products to be delivered such as milestones, deliverables, models, and diagrams. Process fragments represent stages, activities and tasks to be carried out to produce the product fragment. The AASDM uses both product and process fragments in its description. The AASDM method overview describes the activities to obtain product fragments such as models and diagrams. Below, for each of the phases of the AASDM, the method fragments are described and explained. Method fragments are mainly selected from the method base obtained during the literature review and observational case study from in Chapters 2 and 3.

4.4.1 Preparation

This phase functions as a prerequisite for the use of the AASDM. Its goal is to determine whether the project is feasible or not. To determine the feasibility of the project performing a maturity assessment is chosen as the main activity. Maturity assessments are very common within consultancy firms (such as Atos). From the maturity assessment, the user of the AASDM should investigate whether AAS have been implemented or not. The project is qualified as feasible only if it has implemented some AAS. As is stated in the scope of the artifact, the AASDM can be used to add smartness and not create smartness from scratch, which can be seen as an I4.0 transformation. An I4.0 transformation, in this case, means the transition from an I3.0 factory to an I4.0 factory.

The maturity models to perform the maturity assessment can either be (1) internally created maturity models to measure specific aspects of I4.0 or (2) a standard I4.0 maturity model defined by researchers or consultancy firms. Felch, Asdecker and Sucky (2019) define I4.0 consultancy and scientific maturity models. SIMMI4.0 (Leyh, Bley, Schäffer & Forstenhäusler, 2016), M2DDM (Weber, Königsberger, Kassner & Mitschang, 2017) or the CPS maturity model (Westermann, Dumitrescu et al., 2018) are examples that might be a good fit for the maturity assessment of the AASDM.

4.4.2 Specify the added smartness

The first (official) phase of the method is to specify the smartness to be added. The goal of this phase is to obtain a general idea of the project and its requirements. This phase follows the generic Requirements Engineering practices from (Sommerville, 2011): requirements elicitation and analysis, specification of requirements and validation of requirements. The user of the AASDM is to use any method suitable for the three main activities of the Requirements Engineering approach as described here. The results should be documented in a description of the current smartness, specification of the smartness to be added and the initial scoping. It is important to state these three aspects of the project early on in the project.

4.4.3 Model the AS-IS situation

The next phase is the modelling of the AS-IS situation of the system on which the improvement (addition of smartness) will be applied. This builds further on the current smartness description from the previous phase. The goal of this phase is to model the current situation in such a way that improvements to the system of AAS can be made easily. Relevant aspects of an I4.0 AAS system are: (1) the Value Network, its architecture and asset interactions, the processes at the basis of the I4.0 system, and the current AAS implemented.

The Value Network
The Value Network is important for the design (and development) of AAS because they are the ones that determine the content of the AAS and are responsible for the connection of the AAS to its asset. The Value Network partners know the internal structure of their assets and should thus be involved in the process. To get an overview of the Value Network, the Value Network can be modelled as in (Löwen et al., 2016) (on company level). The resulting activity in the AASDM is to model the Value Network.

The system architecture
The architecture of the system is of great importance for I4.0 systems because it is a key difference between I3.0 and I4.0. A system architecture or systems architecture is the conceptual model that defines the structure, behaviour, and more views of a system (Jaakkola & Thalheim, 2010). The architecture plays a major role in systems analysis, design and development according to Jaakkola and Thalheim (2010). Therefore, the architecture of the system must be modelled. This can be done conceptually using assets and the notation of the Plattform I4.0 for the I4.0 Components (with AAS) as in figure 2.5. Hereafter, the behaviour of the system must be determined. The behaviour of the system is based on business processes, which result in asset interactions. This is why the process and asset interactions must be modelled. The process can be modelled using Business Process Modeling Notation (BPMN), which is a standard for the specification of business processes and is ratified as ISO 19510. Based on the specification of the business process, the asset interactions can be determined and modelled onto the system architecture. The activities of the AASDM thus include: model the system of assets as architecture, model process(es), and model asset interactions onto the architecture.

Current AAS implemented
As is one of the requirements of the AASDM, AAS should have been implemented before using the AASDM. Therefore, the currently implemented AAS must be demonstrated. To demonstrate the AAS currently implemented there are two options: (1) obtain an overview of the formal shell descriptions or (2) obtain an overview of the informal descriptions of the shells. The formal shell descriptions can be obtained using software such as the AASX Package Explorer from (Plattform I4.0, 2018a), in which all relevant aspects of the shell are stated in the formal structure of the shell. The second option is to obtain an informal overview of the shells and submodels in an Excel sheet as is done in (Löwen et al., 2016) or even by stating the data, functions, and messages of the AAS. At this point in the project, it is only necessary to determine the functions, data, and messages of the current system. This can be obtained using both methods. The corresponding AASDM activity: obtain shell overview.

4.4.4 Model TO-BE situation

The third phase of the AASDM is to model the TO-BE situation. The TO-BE situation is the preferred outcome state of the I4.0 system. The goal of this phase is to informally specify the new (TO-BE) state of the system. The process of modelling the TO-BE system is similar to the modelling of the AS-IS situation. However, before the modelling of the TO-BE situation, the options for the system must be outweighed. The allocation of smartness within the system can be done in many different ways. Therefore, a scenario analysis is selected as a method fragment for the modelling of the TO-BE situation. Scenario analysis can provide insights into the future by considering alternative outcomes. It is a frequently-used method to explore what a proposed system is required to do in the early phases of system development (Ferris, Barker & Adcock, 2016). In this article, the importance of not having too many scenarios is emphasized. This is why in the AASDM the optimal number of scenarios is set to three. The best scenario can hereafter be chosen based on for example a cost-benefit analysis for each of the scenarios. The emerged AASDM activities for this phase are: model three scenarios for the allocation of smartness and choose best scenario, model the new Value Network and system of assets architecture of the best scenario, model the TO-BE processes, model assets interactions in the architecture, and specify the scenario informally. The specify scenario informally is similar to the obtain shell overview activity and also requires the data, functions and messages of the AAS to be specified.

4.4.5 Formalize the best scenario

The fourth phase of the AASDM is to formalize the best scenario. Formalize means here to fit the data, functions and messages defined in the previous phase into formal descriptions following the AAS meta-information model and AAS communication requirements. To formalize the best scenario method fragments are mainly selected from (Plattform I4.0, 2018a) and (Platform-I4.0, 2019). At first, one must identify what changes to the system must be made by comparing the AS-IS and TO-BE situation (based on their informal specifications), so no duplicate AAS software is created. From this, it becomes clear what AAS, submodels and properties must be changed or added to the system. This is why the first activity in the fourth phase is: create a high-level overview of the impact (TO BE vs. AS IS). In other words, listing the changes to the current system to reach the TO-Be situation.

Thereafter, the formal specification of the AAS can start. For the formal specification of the AAS, the AAS information model is the basis. From the Systematic Literature Review in chapter 2 it is found that for the design of AAS the identification, views, security, concept dictionaries, submodels and properties must be resolved. For the formalization of the submodel content one must follow the three-step approach from (Platform-I4.0, 2019): define the use and business relevance, define details on possible functionalities and interactions, and define the properties. To formalize the properties of the submodels ISO 61360 must be followed, which is the case for the template of (Platform-I4.0, 2019). This template states that for each property in a submodel the name, parent, semanticId, idShort, description, value and qualifier must be imposed. The resulting AASDM activities are: formalize the generic aspects of the AAS, formalize AAS basic submodels, formalize AAS free submodels.

The next, activity in the AASDM is to formalize the I4.0 Component interactions. For this, the I4.0 language must be adopted as defined in (Plattform I4.0, 2018b). Belyaev and Diedrich (2019) have defined several methods for the definition of the I4.0 Language, which are adopted for the AASDM. For the specification of the interaction protocol, UML sequence diagrams are used. For the messages between the I4.0 Components, the XML and JSON format must be specified. To determine the behaviour of the I4.0 Components, UML activity diagrams are used. In the end, the user of the AASDM should also determine which interaction framework to use. According to the standards for I4.0, the options are: MQTT, AMQP, and HTTP.

4.4.6 Test TO-BE scenario

The TO-BE scenario must be tested once it is created. The goal of this phase is to verify the requirements from phase 1 and 3, and check the created software with the specifications from phase 4. Testing the TO-BE scenario does not require a real method fragment to be determined. Based on the outcomes of the tests, the subsequent activity is selected. The resulting activities are: test the chosen scenario for added smartness, implement scenario, adapt scenario, define the new scenario, and reconsider smartness.

4.5 Method Assembly

For situational Method Engineering consistencies are of great importance during method assembly. For example, the input of phase two is the output of phase one, the input of phase three is the output of phase two etc. Besides that, within each phase, the level of detail is kept consistent. For example, phase four is more detailed than phases two and three, which are more detailed than phase one.

The method outline and method fragments that can be placed in the method outline are at the end structured and assembled to the AASDM. The result is the structure defined in 4.4. In the next chapter, the assembled method is explained in detail.

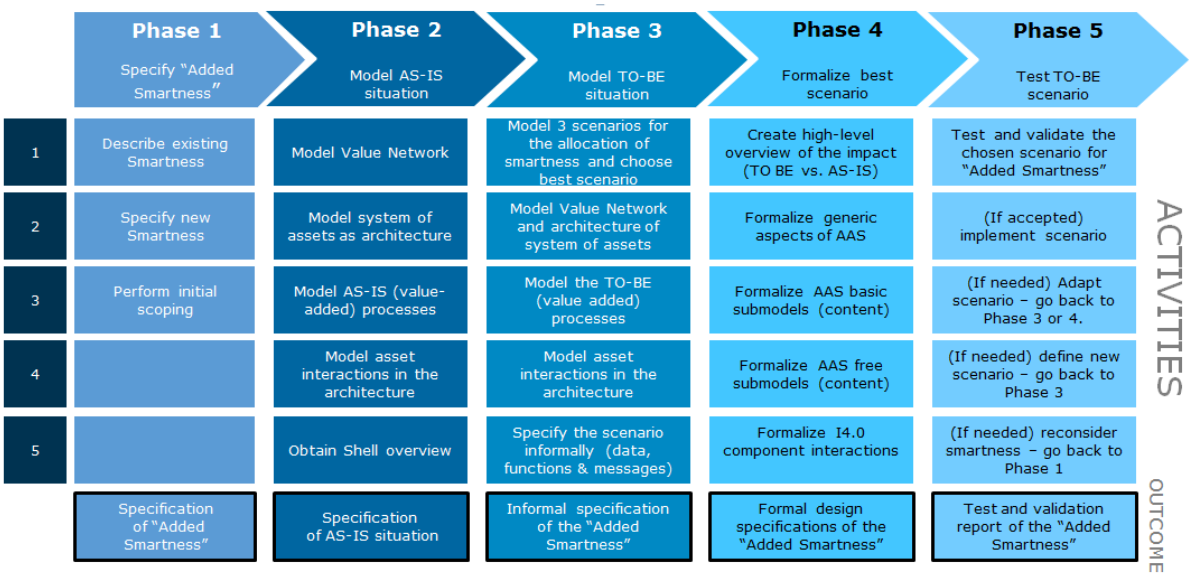

Figure 4.4: Overview of the phases and main activities

Chapter 5

The Artifact: AASDM

In this chapter, the outcome of the artifact design phase is described. The outcome of the artifact design phase is a method for the design of networks of Asset Administration Shells. This method is called the Asset Administration Shell Design Method (AASDM). For Atos, this method is intended to be used as a service called "Adding Smartness to Smart Factories". In this chapter, the details of the AASDM are given. Each of the phases in the AASDM is discussed in detail and the activities to be executed are elaborated upon. A visualization of the entire method with its main activities is given in figure 5.1. Note that the development and testing of the software is left out of scope and is thus not discussed as being part of the AASDM.

The phases of the method are discussed in detail with its activities. Phase 0 functions as a prerequisite for the AASDM. Phase 1 specifies the added smartness (section 5.2), Phase 2 models the AS-IS situation (section 5.3), Phase 3 models the possible TO-BE situations and how it is decided what situation is chosen (section 5.4), Phase 4 formalizes the chosen TO-BE situation with existing standards and turns it into formal design specifications of AAS software (section 5.5), Phase 5 tests whether the outcome of the new smartness matches the desired smartness (section 5.6).

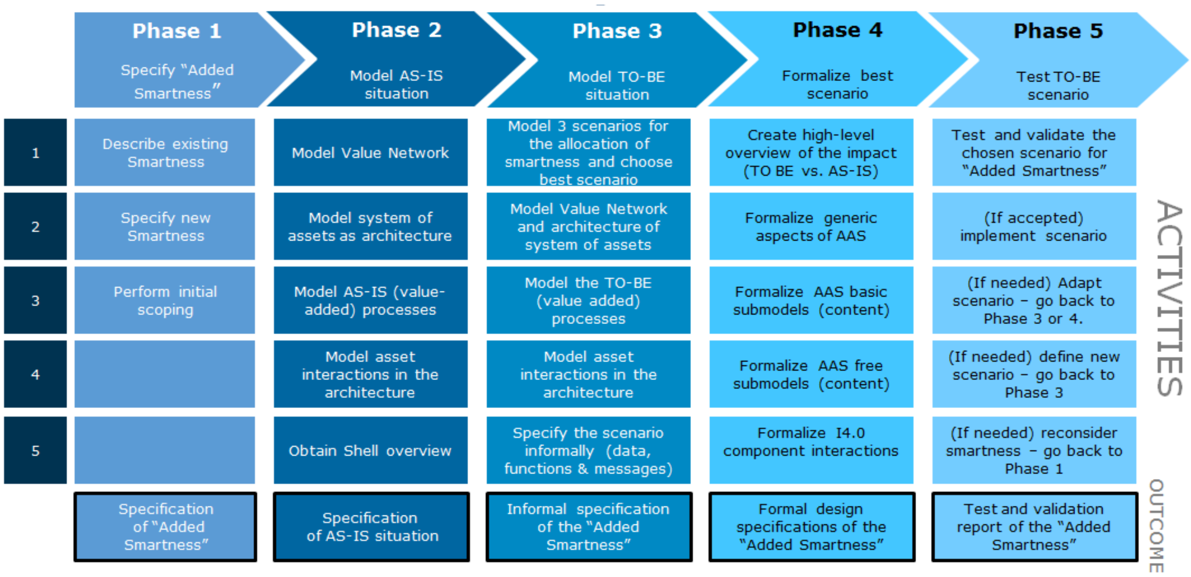

Figure 5.1: Overview of the phases and main activities

5.1 Phase 0: Prepare

The preparation phase of the AASDM consists of an I4.0 maturity assessment and – if necessary – I4.0 workshops. As is stated, a prerequisite for the AASDM to start is that smartness is implemented to some extent. The smartness of a Smart Factory can be determined via I4.0 maturity assessments. However, to measure smartness (in this case: via the Asset Administration Shells) there is no specific maturity assessment available on this date. The user of the AASDM is therefore directed to existing maturity scales such as SIMMI4.0, M2DDM, and the CPS maturity model. These maturity scales tend to measure the I4.0 software-related capabilities of a factory. From the maturity assessment, it should become clear whether AAS have been implemented (correctly) or not.

The maturity assessment functions as a feasibility check for the project. The project is deemed feasible if AAS have been implemented to some extent if not the project is deemed infeasible. If the project is deemed infeasible, I4.0 workshops can be performed to increase awareness of AAS and might initiate the implementation of AAS. Hereafter, if the Smart Factory has increased its smartness by implementing AAS the AASDM might be initiated after this.

5.2 Phase 1: Specify the Added Smartness

The first phase of the AASDM is initiated after the project is deemed feasible in the previous phase (phase 0: prepare). The goal of this phase is to get an understanding of the current smartness, specify the smartness to be added and functions as initial scoping. During this phase, the user works with the problem owner(s) to identify opportunities to add smartness. Thereafter, the focus of the project must become clear through scoping.

Describe current smartness
A high-level overview of the I4.0 capabilities currently achieved by the AAS is sufficient at this stage of the project. The current smartness is defined in collaboration with the factory owner.

Specify the "Added smartness"
The added smartness can be conceived through workshops or on the client's initiative. The document 'Aspects of the Research Roadmap in Application Scenarios' can provide support the consultants to identify opportunities (challenges) for I4.0 applications (Anderl, et al., 2016). The application scenarios focus on the generic value creation processes of manufacturing companies as PLM (product life cycle management), Production Life Cycle Management (PSLM), Supply Chain Management (SCM), and services. The "Added Smartness" must be specified, not only described. Requirements from the problem owner(s), if stated, must be specified as well.

Perform initial scoping
The initial scoping is the second activity in this phase. Scoping must be performed to narrow the solutions down and work more efficiently. Scoping in this phase is mainly focused on identifying the high-level Value Network (stakeholders) and - if applicable - a general understanding of the targeted part of the Smart Factory. A glimpse of the relevant processes can be caught as a basis for the next phase.

5.3 Phase 2: Model AS-IS situation

The goal of this phase is to map the current (AS-IS) situation of the I4.0 components in detail. Five activities are significant to do this in a structured and complete manner.

Model Value network
The first activity is to define the Value Network behind the value-added processes in the current situation. The Value Network does not have to be modelled in detail, the Value Network on company level is sufficient. The Value Network must be defined for two reasons: (1) horizontal integration is important to I4.0 solutions and in a lot of cases, collaboration with partners is required, (2) negotiations on the content of the so-called free submodels (see Appendix A.5) must be held among Value Network partners.

Model the system of assets as architecture
This activity is the most important in this phase. The system of assets of a factory is based on a hierarchical view of the factory. RAMI4.0 has defined the hierarchy levels of I4.0 components: connected world, enterprise, work centers, stations, control devices, field devices and products. Modelling all the assets of the factory in the architecture allows you to see changes to the network of assets more easily. In the architecture, it is clarified what assets are I4.0 components and what assets are not.

Model AS-IS value-added processes
The "Added Smartness" has its influence on the business processes of the factory. The Business Processes on which the added smartness has influence must be modelled. The modelling of business processes can be performed by using Business Process Modeling Notation (BPMN). The process must be described to easily determine the asset interactions (and asset actions).

Model asset interactions in the architecture of the system of assets
Based on section 3.2.3 the interactions between assets (and therefore the I4.0 components) become clear. These interactions can be horizontal interactions between I4.0 components of vertical interactions with higher-level IT systems such as ERP, PLM and MES. These interactions must be visualized on the architecture of the system of assets defined in section 3.2.2. The (physical) actions of the assets (e.g. picking up a product or sending) can also be shown in this architecture.

Obtain a shell overview
A shell overview must be obtained to see all the assets and AAS in one glance. One must also be able to see the details of the assets and AASs (in the form of submodels and properties) in this overview. This way, all the smartness currently implemented is known. In later phases of this guide (section 3.4), this shell overview can be used to get an overview of the impact of the "Added Smartness". A simple Excel sheet might be sufficient to show the shell overview. However, software like AASX Package Explorer is developed to easily create an environment in which the assets and AASs are documented.

5.4 Phase 3: Model the TO-BE situation

This phase describes the TO-BE situation for the “Added Smartness”. During this phase, it is decided how the smartness is added to the current system. Requirements from the partners in the value network are leading here. Five activities are crucial to model the TO-BE situation. The goal of this phase is an informal description of the proposed solution.

Model scenarios for the allocation of smartness as architecture and choose the best scenario
The current architecture of the system of assets and the asset interactions are modelled in the AS-IS situation. This architecture shows the asset information and interactions in one view.

Different scenario descriptions - short process descriptions – can be created for the “Added Smartness”. The allocation of the smartness must be described in these scenarios. It is important to describe what I4.0 component is responsible for which task.

Visualizations of the scenarios can be made similar to the architecture and asset interactions from the previous section. These visualizations are based on the process descriptions of the different scenarios. From these visualizations, it must become clear what I4.0 component is/are responsible for the “Added Smartness”. It must become clear what assets become I4.0 components, what the responsibilities of the I4.0 components are, and what new interactions are established. A guideline for the number of scenarios to be modelled is three scenarios. Three scenarios are found to be sufficient otherwise the process to determine the best scenario becomes too time-consuming.

After the scenarios have been described at a high-level the problem owners are being consulted to determine the best scenario. This best scenario is modelled in more detail in the next activities of this phase. A cost-benefit analysis can be used to determine the best scenario. The best scenario is modelled in a similar way as in the AS-IS situation. The Value Network, architecture, processes, and asset interactions must again be modelled. This is described in the next activities in this phase of the AASDM.

Model new Value Network and new architecture
The best scenario is determined in the previous activity. Changes to the Value Network might occur based on the chosen scenario. Therefore, the Value Network might have to be re-established. The next step for the chosen scenario is to model the system of assets as an architecture (as is done in the previous section). The outcomes of these activities is a new Value Network and the new architecture of the system of assets.

Model the TO-BE value-added processes
Value-added (business) processes change in the new scenario. The next activity is thus the modelling of the new or adjusted value-added processes. The modelling of business processes can again be performed by using Business Process Modeling Notation (BPMN). The process must again be modelled to easily determine the asset interactions (and asset actions). The differences with the AS-IS situation can be highlighted.

Model the asset interactions in the architecture
Again, the process must be described in terms of asset interactions. The outcome shows the AAS interactions and asset actions.

Specify the scenario informally (in terms of data, functions and messages)
In collaboration with the problem owner, the best scenario is chosen (section 3.3.1). For the best scenario, the team must work towards design specifications. At the basis of the design specifications for AASs are the data and functions of I4.0 components and messages exchanged between I4.0 components. The messages exchanged can be seen as the AAS and assets interactions modelled in the preceding activity. The data and functions are what the I4.0 component can do and what it knows. For all the I4.0 components (assets) the data, functions and messages must be described. In the next phase, these informal specifications are translated into formal AAS design specifications.

5.5 Phase 4: Formalize the best scenario

The previous phase has resulted in an informal specification of the AAS. These informal specifications must be translated into formal AAS design specifications. This means, describing the AAS, the submodels and the AAS interactions. The goal of this phase is a formal specification of the AAS design. The activities to undertake to specify the (networks of) AAS formally are described below.

5.5.1 Create a high-level overview of the impact

The TO-BE situation (Phase 3) must be compared to the AS-IS situation (Phase 2). From the new architecture, it becomes clear where to place new shells or what shells to adjust in comparison with the architecture of section 3.2.2. The data and functions from section 3.3.5 compared to the existing overview of 3.2.5 helps identifying the new information or adjusted information to store in the shells. The impact on the existing shell structure can be listed in, for example, an Excel sheet, from which it becomes clear what changes and additions must be made to the current AAS. A high-level overview of the new and changed (and existing) shells is the outcome.

The high-level overview of the impact shows where new AAS are required, where changes or new submodels must be placed, and what AAS interactions must be defined. For the creation of new AAS more activities must be performed and therefore all the activities must be carried out in this phase. If new submodels must be created one can start with the third or fourth activity depending on whether the submodel is free or basic.

5.5.2 Formalize generic aspects of the AAS

The chosen scenario in the previous phase might require new shells to be added to assets (make I4.0 components from the assets). New shells must comply with the rules and standards created by the I4.0 community. The AAS must follow the requirements from meta-data model defined in (Plattform I4.0, 2018a) and (Platform-I4.0, 2019). The key elements of an AAS are defined as:

- Mandatory aspects of an Asset Administration Shell based on the meta-information models from (Plattform I4.0, 2018a) and (Platform-I4.0, 2019).
- Basic submodels (see section 5.5.3).
- Free submodels (see section 5.5.4).

For new shells, the first thing to do is to define the mandatory (generic) aspects of an Asset Administration Shell (for more details on the mandatory aspects of the AAS, see the two documents mentioned earlier). In summary, for the definition of new shells do the following:

1. Define whether the AAS represents an asset type or asset instance. If it is possible, refer to other AAS types.

2. Define identification for all the elements of the AAS based on IRDI, URI or customized identifiers.
3. Define security for the shell. In (Plattform I4.0, 2018a) the focus for security is on attribute-based access control. Other security requirements are met through design.
4. Optionally, link views to the shell. Properties of submodels can be linked to a view. One can use the basic views defined in (Plattform I4.0, 2018a) or define views in collaboration with your Value Network.
5. Optionally, define the concept dictionary. In this concept dictionary, concept descriptions of the elements of the AAS are stored. These concept descriptions might be copies of external standard descriptions.

Besides the generic information of an AAS, the most important information is stored in submodels. The submodels aggregate the information about a specific functionality of an asset. In section 3.4.3 the focus is on basic submodels, which are based on standards and apply to most of the shells (or are asset class-specific). In section 3.4.4 the formalization of the free submodels is discussed. Free submodels are defined based on negotiations with the Value Network for a specific application. "Adding Smartness", the focus of the AASDM, mostly relates to the design of free submodels since only a small number of basic submodels have been determined.

5.5.3 Formalize AAS basic submodels

For the formalization of AAS submodels, several tasks must be performed. In general: the meta-model information of a submodel must be defined, the submodel class must be defined (decide if it is a free or basic submodel), and the three submodel definition steps of (Platform-I4.0, 2019) must be followed.

1. **Define meta-model information of a submodel:** Submodels aggregate information associated with one specific (Plattform I4.0, 2018a) functionality of an asset. For the definition of submodels, the meta-information of a submodel must be defined together with its content. This task focuses on the meta-information of an AAS submodel.

2. **Define submodel class:** For the content of the submodel the first thing to do is to define what class the submodel belongs to. The submodel classes are defined in Appendix A.5. The most relevant distinction to be made is the distinction between basic submodels (this section) and free submodels (section 3.4.4). For basic submodels, there are four different sub-types: basic mandatory, basic optional, asset-specific mandatory and asset-specific optional. The tasks to be performed for these classes of submodels are similar, however.

3. **Define use and business relevance of the submodel:** This is defined as one of the three main steps in the definition of submodels in (Bedenbender, et al., 2019).

4. **Define details on possible functionalities and interactions:** The functionalities correspond to the data, functions and messages defined in Phase 3. Besides this, the bigger context of the properties of a submodel must be defined.

5. **Definition of properties (in this case for basic submodels):** previously it has been determined whether a basic or a free submodel must be created. For basic submodels, standards have been defined for the definition of properties. Generic submodel types have been created for basic submodels from which the content can be copied. The vocabulary (and thus properties) in these submodels are predetermined by linking to existing standards. The properties are defined according to ISO 61360.

5.5.4 Formalize AAS free submodels

Besides the basic submodels, there are free submodels. The free submodels are defined for specific applications. The free submodels are defined in collaboration with the involved partners in the Value Network. Similar tasks must be undertaken as for the basic submodels except for the fifth task, the definition of the properties. For the free submodels, the fifth task must be replaced by the following:

For free submodels, no standards have been defined yet. Therefore, collaboration and negotiation between involved partners of the Value Network must decide on which properties to include in the new free submodel. The following must be described for properties: name, parent, semanticId, idShort, description, value and qualifier. The properties are, of course, defined based on the data and functions from the informal specification established in phase 3.

5.5.5 Formalize I4.0 Component interactions

The general interactions of I4.0 components are defined as messages in the informal specification in phase 3. These messages must be formalized to establish I4.0 communication following the I4.0 language specification in VDI/VDE 2193. The three layers of the I4.0 language must be followed. In the previous activity (definition of submodels) the vocabulary is established for the AASs. Now, the interaction protocols and messages following the I4.0 language structure must be defined. The tasks to be performed in this activity are:

1. **Define interaction protocol:** The I4.0 language suggests the definition of interaction protocols. Interaction protocols define the sequencing of messages between I4.0 components. UML sequence diagrams must be defined for this purpose.
2. **Define formal messages:** The messages between the I4.0 components must be defined according to VDI/VDE 2193. These messages are used to request or provide services for other I4.0 components. Messages must be defined in XML or JSON according to the standards for communication defined in RAMI4.0.
3. **Define the behaviour of the I4.0 components:** The messages between I4.0 components cannot be exchanged if the decision making logic of an I4.0 component is not defined. So far, in VDI/VDE 2193-2 the roles of Service Provider, Service Requester, or Informer can be taken by an I4.0 component. The behaviour of the role of the I4.0 component in the interaction must be defined by UML Activity diagrams.
4. **Define interaction framework:** MQTT is suggested as an interaction framework for the I4.0 language. Other interaction frameworks are AMQP or HTTP. Decide with software developers on what interaction framework can best be used.

5.6 Phase 5: Test the added smartness

The created scenario must be tested. This means, checking if the created scenario complies with the steps taken in the previous phases (phase 1,3 or 4). Possible outcomes are:

1. The chosen scenario fits perfectly with the intended smartness desire. The solution can be implemented.
2. The chosen scenario fits with the intended smartness but needs some adaptations to become exactly what the problem the owner had in mind.
3. The chosen scenario fits with the intended smartness but needs some adaptations to become exactly what the problem the owner had in mind.
4. The "Added Smartness" does completely not comply with the intended smartness desire of the problem owner as defined in Phase 1. Other smartness might be needed to fulfil the problem owner's goals. Specification of new smartness is the next step.

Implement the scenario
The created software, from the software development and testing activities, fits perfectly with the intended smartness desire. Implementation is the next step to take.

Adapt the scenario
The created software does not fully comply with the intention of the problem owner. Minor adjustments can be made to the scenario. Reconsider going back to phase 3 (section 3.3) to adapt the scenario. Also, some issues might have occurred in Phase 4 during the formalization, if this is the case go back to Phase 4 (section 3.4).

Choose new scenario
The scenario does not fit with the desired smartness outcome and the scenario must be reconsidered. This can be due to the infeasibility of the scenario or the problem owner desires another way of allocating the smartness. Infeasibility is mainly due to for example technical infeasibility. A new scenario must be selected from the list of scenarios in Phase 3 or new scenarios must be created in this phase.

Reconsider smartness
The created smartness does not fulfil the desire for new smartness specified in Phase 1. New smartness ideas must be discussed with the problem owner to initiate a new iteration of the service "Adding Smartness".

Chapter 6

Exemplary Applications

In this chapter, the AASDM from the previous chapter is applied to exemplar cases. The exemplars in this chapter contribute to the improvement of the artifact and support the validation of the artifact. The AASDM is tested in a system that corresponds to the real-world. The validation of the artifact is supported by the exemplars because they are leading in the discussions during the artifact validation phase. The importance of the exemplar in the understanding of the artifact may not be neglected. The artifact itself at this point can be hard to understand, the exemplars are developed such that the implementation of the artifact is made clearer. The exemplars show how to carry out certain phases and activities of the AASDM.

The choice of exemplar applications over real-world applications requires further explanation. Section 6.1 describes the rationale behind the decision to use exemplar applications over real-world applications. Hereafter, two exemplary applications of the AASDM are shown. Section 6.2 describes a case in which “mass customization” is desired and section 6.3 describes a case in which preventive maintenance is automated. The cases show how to carry out phase one to three in detail. Phases 4 and 5 require more effort and are thus not documented in this master thesis project.

6.1 Rationale behind the use of exemplary applications

Before the reasoning behind the choice for an exemplary application is explained, it must first be explained why exemplary applications are used by software engineering researchers in general. In (Feather, Fickas, Finkelstein & Van Lamsweerde, 1997) the use of specification exemplar is investigated for writing specifications and for the requirements engineering process. Feather et al. (1997) state that the use of standard exemplar is a widely accepted tool in specification research. These exemplars generally aggregate self-contained, informal descriptions of problems in a specific application domain to serve as input for the specification process. In the broadest sense, exemplars define model specification tasks. Feather et al. (1997) state three primary purposes for exemplars: advancing a single research effort, promoting research and understanding among multiple researchers or research groups, and contributing to the advancement of software development practices. In this exemplar, the focus is on the advancement of software development practices as the goal of the master thesis project was to create a method for (AAS) software design. Software design is a crucial step in the software development process.

The choice for the use of an exemplar over a real-world case study is based on two reasons: (1) the development of Asset Administration Shells is not yet completed and (2) real-world large scale implementations of Smart Factories with Asset Administration Shells do not yet exist or are unavailable. As is said, the first reason is the uncompleted development of the Asset Administration Shell concept. So far the information model of the Asset Administration Shell, an AAS interaction model, and the I4.0 language (semantics) have been defined. For example standardization efforts, the infrastructure and security mechanisms of the Asset Administration Shell have yet to be completed (PlattformI4.0, 2019). The second reason is that Asset Administration Shells are not implemented in large-scale Smart Factories at this point in time. The Asset Administration Shells have not yet been implemented on a large scale, because the development process is still ongoing. One of the assumptions of the method is that smartness must be implemented (via AASs) to some extent. However, if no large scale implementation of AASs exists the method cannot be tested in the real-world. Therefore the choice of an exemplary application is supported.

The creation of Smart Factories on a large scale and Asset Administration Shell applications is in progress. One can find small individual research projects that have implemented Asset Administration Shells. Besides this, three larger-scale projects are created by the Plattform I4.0, SmartFactoryKL and Siemens and Festo. The Plattform I4.0 has created a demonstrator of a parcel sorting system with Asset Administration Shells at the Hannover Messe Industrie 2018 and 2019. The SmartFactory in Kaiserslautern (SmartFactoryKL) have created one of the first Industrie 4.0 production plants. Siemens and Festo have created a flexible multi-carrier system that is used for a bottling demonstrator. These three project all are still work-in-progress, but are at the frontrunners of AAS implementations.

6.2 Example case 1: Mass Customization

The first example case is based on the demonstrator as described by Löwen et al. (2016). This demonstrator is one of the cases examined in chapter 3. This case is slightly adjusted to fit with a larger amount of factories. For example, the machines in the factory called mach-1, mach-2 and mach-3 instead of capping, filling and labelling machine. These generalizations and changes are made to show the essence of the application of the AASDM. More details on the case are defined in the phases 1 and 2 of the AASDM (section 6.2.2 and 6.2.3).

6.2.1 Preparation

For all the example cases described in this chapter, it is assumed that smartness is implemented to some extent. This means the project is deemed feasible based on the outcomes of the maturity assessment performed prior to the AASDM. No additional I4.0 workshops and implementations of smartness are required. In other words, all cases described have implemented AAS.

6.2.2 Phase 1: Specify new smartness

The three main activities in the first phase of the AASDM are: describe the existing smartness, specify the new smartness, and perform initial scoping. The results of these activities for the case are described below.

Describe existing smartness
From the initial talks with the problem owner(s), it has become clear that the factory has adopted "smart machines" and some sort of "smart planning and controlling system" in the current situation. Via AAS the machines, planning and control system are connected. The planning and control are responsible for steering the employees to pick up the products according to the planning. The "smart machines" have the ability to reconfigure themselves for specific products. They communicate with the control system to gain information about a specific product.

Specify new smartness
The talks with the problem owner(s) have shown that there is a desire to achieve mass customization at a high speed by implementing driver-less vehicles in one of the manufacturing cells. These driver-less vehicles must be configured in the existing system via Asset Administration Shells.

Perform initial scoping

The initial scoping has resulted in the following. The case concerns only one manufacturing cell within a factory consisting of three machines, a planning system, control system, and employees that take care of transportation. The process that is targeted is the general production process within this manufacturing cell.

6.2.3 Phase 2: Model the AS-IS situation

Modelling the AS-IS situation involves the following activities: model the Value Network, model the system of assets as an architecture, model the AS-IS process(es), model asset interactions in the architecture, and obtain a shell overview.

Model the Value Network

In figure 6.1 the Value Network is modeled for the first example case. The Value Network shows stakeholders that have contributed to the current smartness of the manufacturing cell. From further talks with the problem owner, it has become clear that the supplier of the sensors, supplier of the machines, implementer of the planning software, implementer of the controller software, suppliers of the raw materials, the integrator of the system, and the operator(s) of the manufacturing cell are the relevant partners in the Value Network.

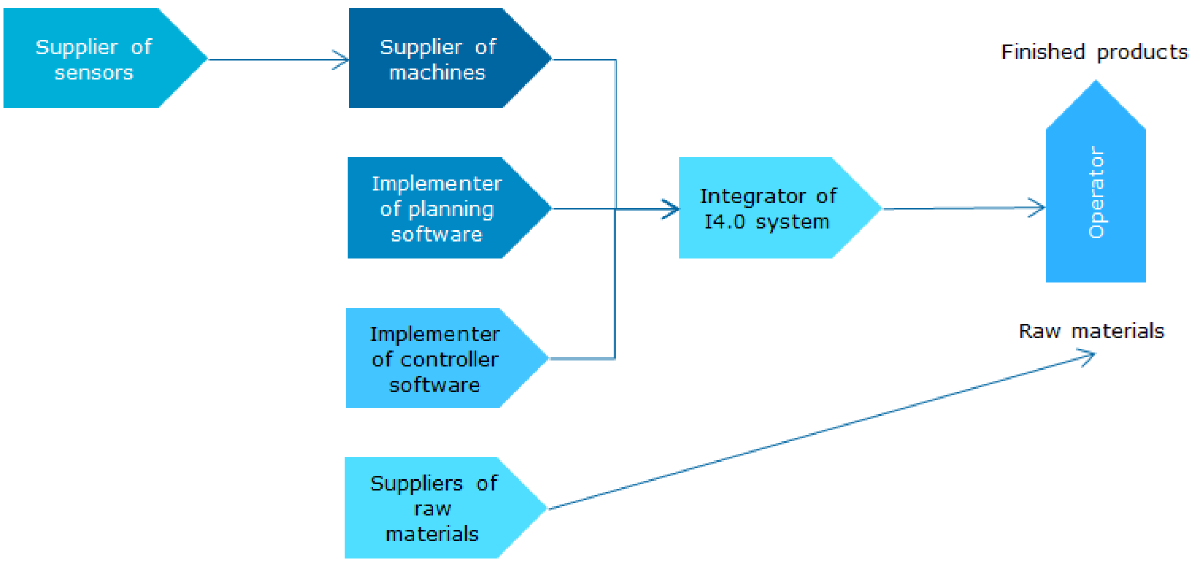

Figure 6.1: Case 1 current Value Network

Model the system of assets as an architecture

Figure 6.2 shows the system of assets as currently implemented. As can be seen the controlling system, planning system and machines are all equipped with an AAS. In the current situation, the transporters are steered by employees.

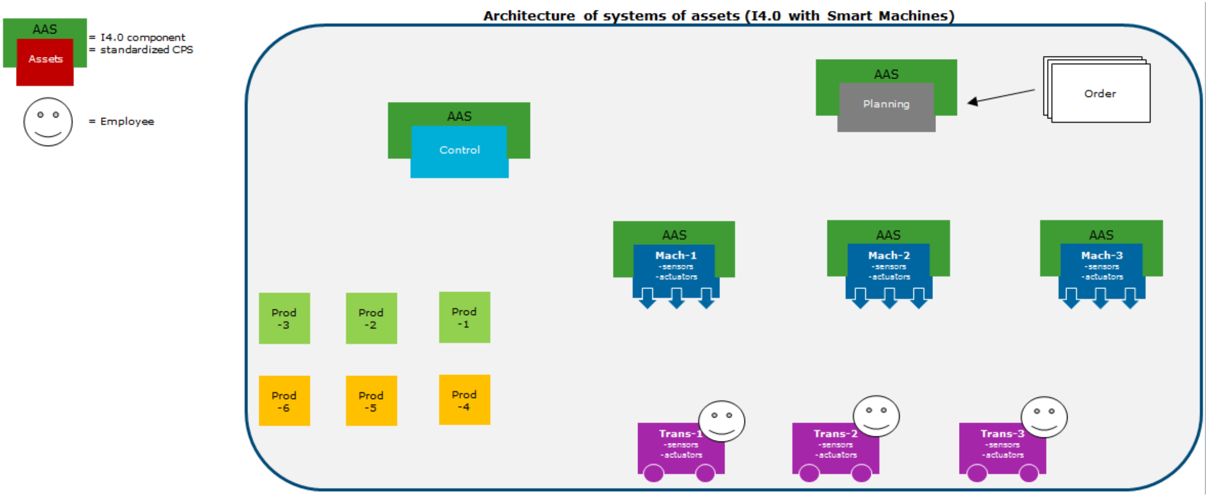

Figure 6.2: Case 1 architecture as system of assets

Model the AS-IS process(es)
As becomes clear the focus of the addition of smartness is in this case on the general production process. In the current situation, the process proceeds as follows: orders arrive, the planning system updates the planning by adding the new orders, the control system then checks the planning to see what products must be finished, then the product is fetched and moved by an employee, the machine re-configures itself - if necessary - and then performs the operation, hereafter the status of the product is updated in the planning system and the planning is again checked by the control system until there are no products left to finish. The process is visualized in figure 6.3.

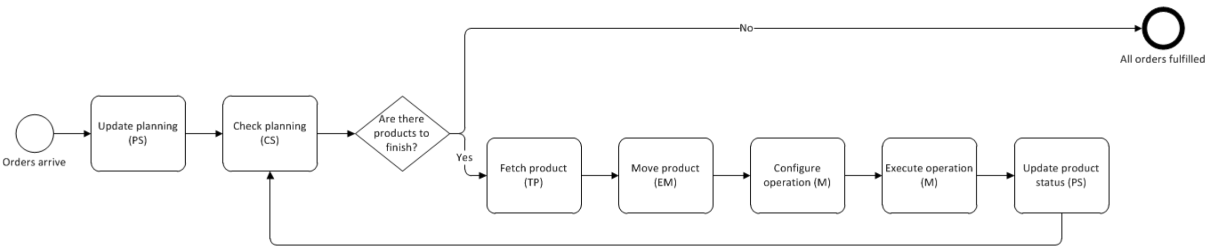

Figure 6.3: Case 1 current process

Model the asset interactions in the architecture
Based on the process described the interactions between the assets can be visualized in the system of assets (architecture). This is shown in figure 6.4. The process from figure 6.3 is described shortly on the left of the figure as well.

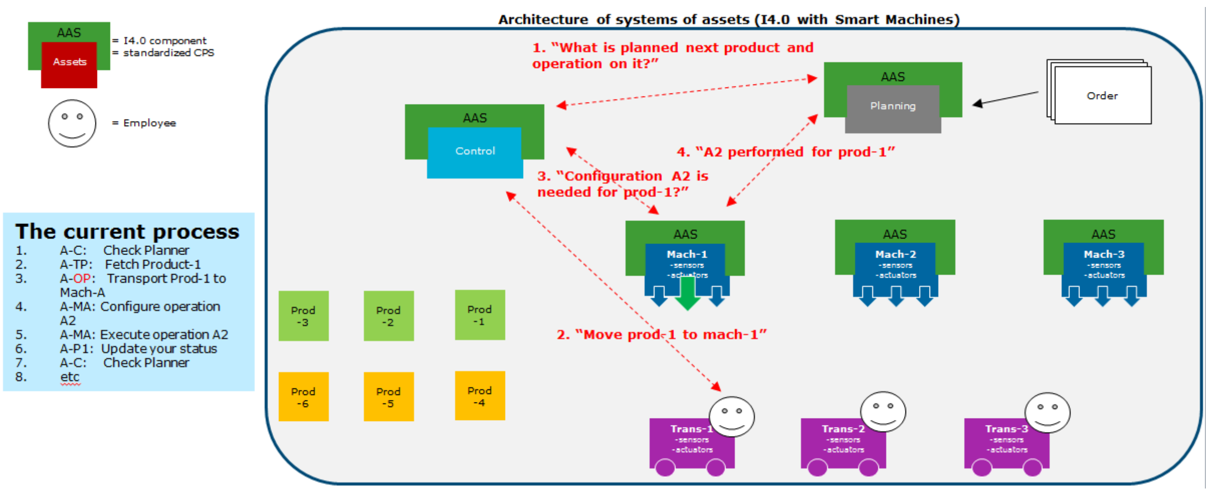

Figure 6.4: Case 1: asset interactions in the asset architecture

Obtain shell overview
The shell overview can, for example, be documented in Excel or in a tool like the AASX Package Explorer [1].

6.2.4 Phase 3: Model the TO-BE situation

Modelling the TO-BE situation means carrying out the following activities: model (three) scenarios for the allocation of smartness and choose the best scenario, model the Value Network and architecture as a system of assets, model the TO-BE process(es), model asset interactions in the architecture, and specify the scenario informally.

Model (three) scenarios for the allocation of smartness and choose the best scenario
From the specification of the smartness (section 6.2.2) several scenarios (possible solutions) can be drawn. In this case, three scenarios are drafted, in which the smartness is allocated differently in each scenario. The difference between the scenarios, in this case, is who is responsible for carrying out the planning. The scenarios for this case are called: smart controller, smart transporters, and smart products. In figure 6.5 the three different scenarios drafted for this case are shown.

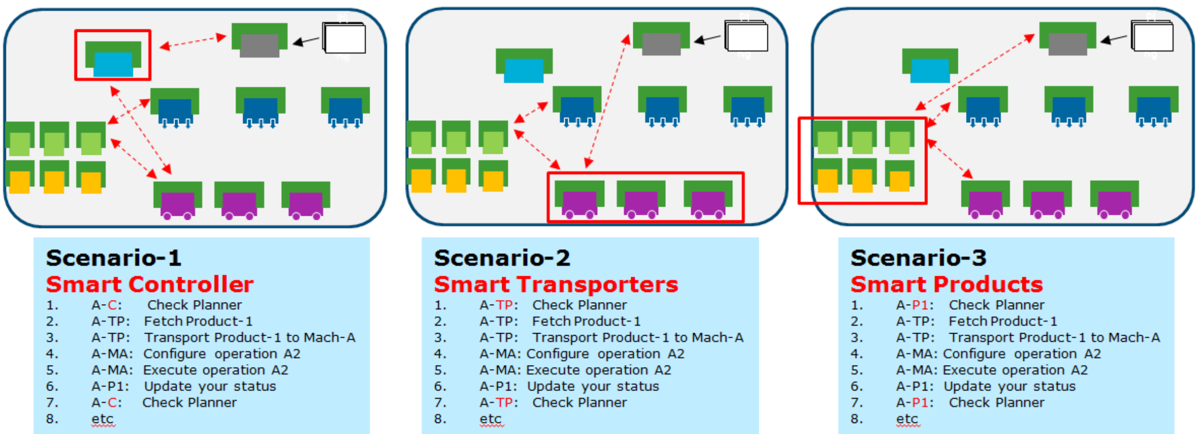

Figure 6.5: Case 1 scenarios

In the first scenario, the smart controller scenario, the controller is responsible for checking the

[1] https://github.com/admin-shell/aasx-package-explorer

planning. This is the scenario that is the most similar to the current situation. In the second scenario, the smart transporters scenario, the driverless vehicles will be connected to the planning system directly via their AAS and will store information about the planning. In the third scenario, the smart products scenario, the products will carry the planning information. Note that in all scenarios driverless vehicles will be implemented to replace the human transporters.

The choice for the best scenario must be made in consultation with the problem owners and other partners in the Value Network. For this case, the scenario that is most similar to the current situation is chosen. This seems logical in practice since it requires the least changes to the current system. Below, the smart controller scenario is worked out further.

Model the Value Network and architecture as a system of assets

For the chosen smart controller scenario the Value Network and TO-BE system of assets as an architecture must be defined. In figure 6.6 the updated Value Network is shown and the architecture is shown in figure 6.7.

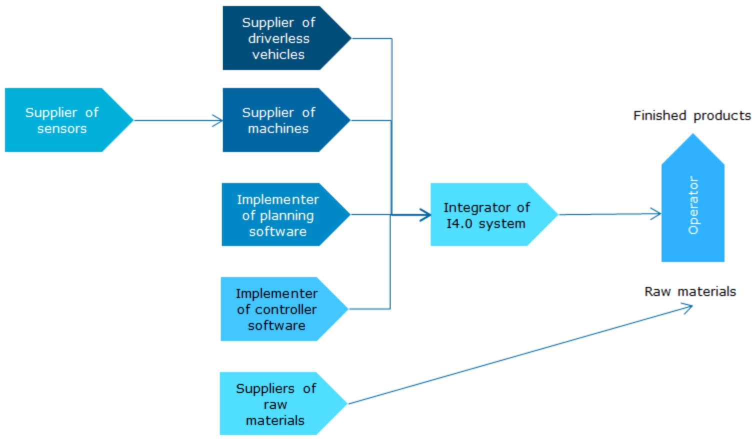

Figure 6.6: Case 1 new Value Network

The only change to the Value Network is the suppliers of the driver-less vehicles. They should be involved in the process from here on.

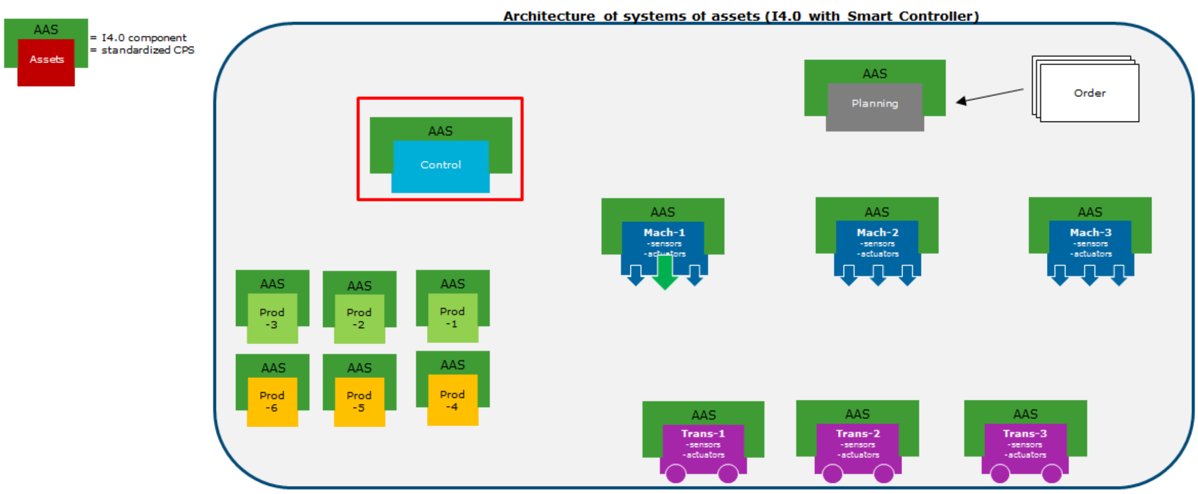

Figure 6.7: Case 1 architecture of the smart controller scenario

Model the TO-BE process(es)

Below the TO-BE process is modelled in figure 6.8. Only one change is made to the current process because the chosen scenario is very similar to the existing situation. In the TO-BE process, the products are fetched and moved by the driver-less vehicles instead of the employees.

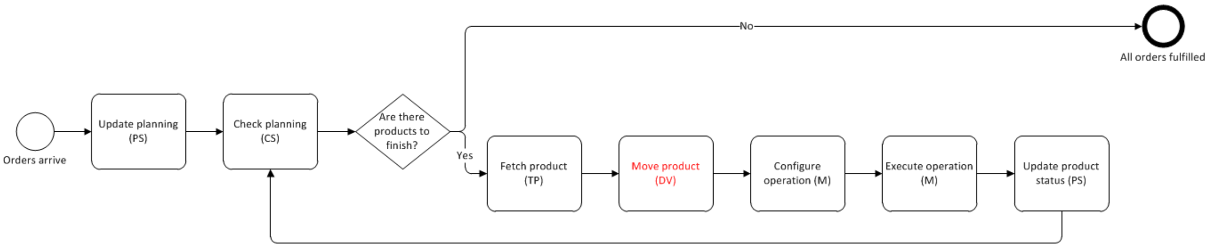

Figure 6.8: Case 1 TO-BE process

Model asset interactions in the architecture

Now that the process is modelled the interactions between the assets might change. In figure 6.9 below the asset interactions are modeled onto the architecture defined earlier.

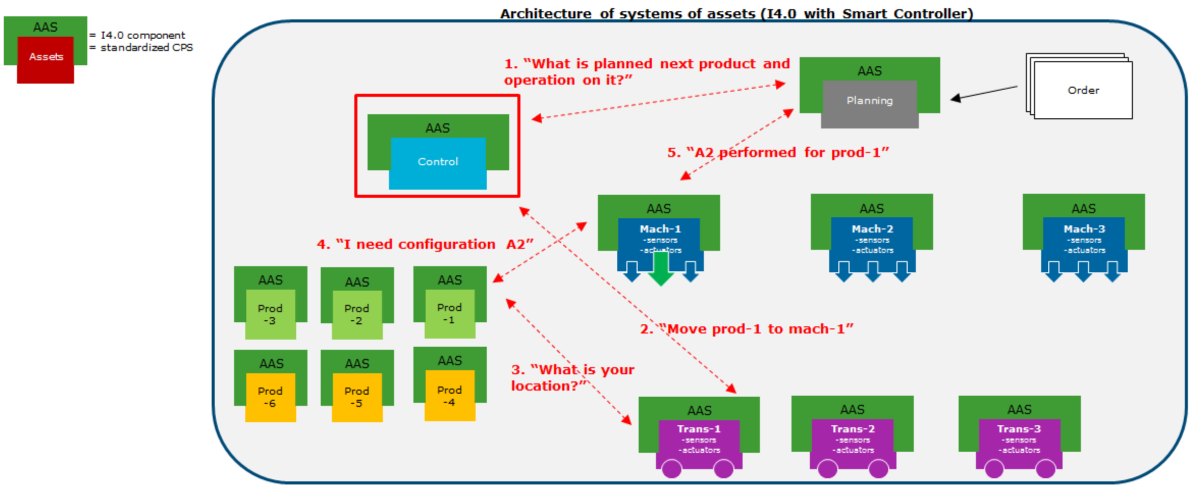

Figure 6.9: Case 1 TO-BE architecture with assets interactions.

Specify the scenario informally

From the TO-BE architecture, the process and asset (and AAS) interactions the scenario can be transitioned to a formal specification. Before that, the content of the AAS submodels must be defined informally to make this transformation easier. The content of the AAS submodels is described as data, functions and messages. The informal specification of the case is shown below in figure 6.10. Data, functions and messages are defined for the AAS in the architecture.

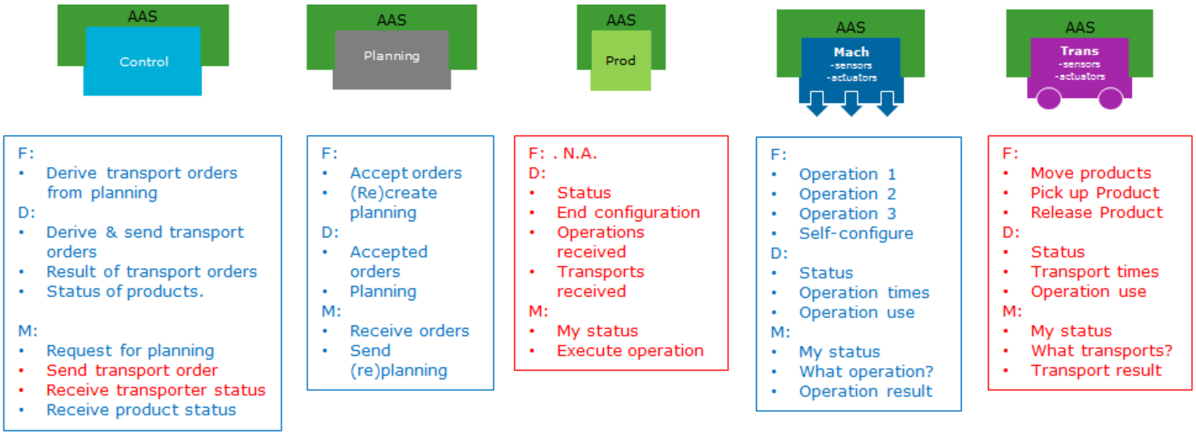

Figure 6.10: Case 1 informal specification

6.3 Example case 2: Preventive Maintenance

The second example case is based on the HMI 2018 demonstrator described in (Platform-I4.0, 2019). This demonstrator describes a parcel sorting system, in which basic submodels and AAS have been defined. This case describes the implementation of preventive maintenance by an outside party. In the current situation, maintenance and replacement are scheduled at predetermined dates. However, a system failure can be prevented by monitoring the condition of the assets in the system. Preventive maintenance can be scheduled as the condition of the asset reaches a threshold value.

6.3.1 Phase 0: Preparation

In the preparation phase of the project, it is determined that the current system has some implemented smartness in the form of AAS.

6.3.2 Phase 1: Specify new smartness

The three main activities in the first phase of the AASDM are: describe the existing smartness, specify the new smartness, and perform initial scoping. The results of these activities for the case are described below.

Describe existing smartness

The current smartness implemented in this case is that all assets are equipped with a technical data sheet, have asset documentation, and asset identification.

Specify new smartness

Talks with the problem owner(s) have shown a desire to implement preventive maintenance. Preventive maintenance, in this case, means automatically calling in maintenance or replacement performed by an outside party. This way the up-time of the system is improved an better quality of sorting can be ensured.

Perform initial scoping

The scope of this system is a parcel sorting system consisting of two sensors, electric reflectors, a mechanical conveyor belt, boxes (products), two sorting boxes, a positioning system and an IoT edge gateway as described in (Platform-I4.0, 2019).

6.3.3 Phase 2: Model the AS-IS situation

Modelling the AS-IS situation involves the following activities: model the Value Network, model the system of assets as an architecture, model the AS-IS process(es), model asset interactions in the architecture, and obtain a shell overview.

Model the Value Network

The Value Network in the current situation can be visualized as in figure 6.11 below. The system integrator integrates the various components of the system into a parcel sorting system that sorts boxes. A maintenance company is responsible for maintenance of the system.

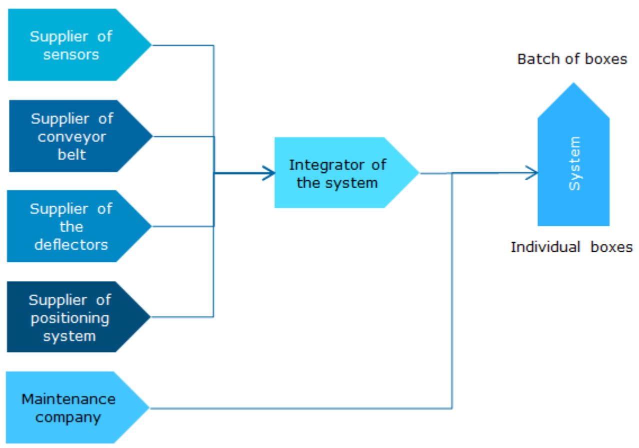

Figure 6.11: Case 2 current Value Network

Model the system of assets as an architecture

The current system architecture is defined in the form of a system of assets. In figure 6.13 the architecture can be seen together with its asset interactions.

Model the AS-IS process(es)

The maintenance process must be described as it is in the current situation. In the current situation, maintenance is scheduled by the maintenance company. Once, the maintenance date is reached the system owner contacts the maintenance company. The maintenance company then performs all maintenance required for the system and schedules a new maintenance date. Hereafter, the process repeats. The current process is shown in figure 6.12.

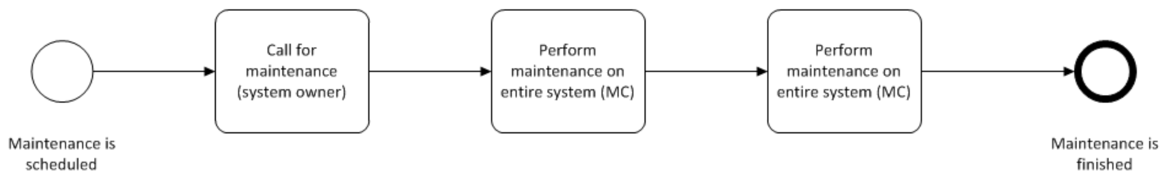

Figure 6.12: Case 2 current process.

Model the asset interactions in the architecture

The only interaction between assets in the system is the system owner requesting a maintenance operation based on the scheduled date. The asset interactions are shown in figure 6.13.

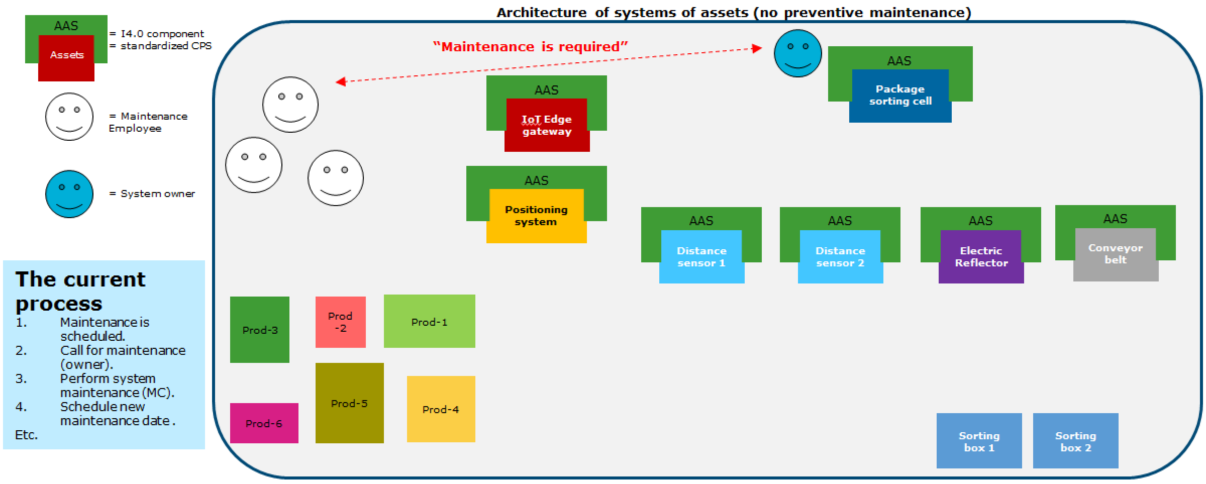

Figure 6.13: Case 2 asset interactions in the architecture

Obtain shell overview
The current data, functions and messages of the system can be modelled in Excel, AASX Explorer Package, or informally in a figure. For this case, the current shell overview is shown in figure 6.14.

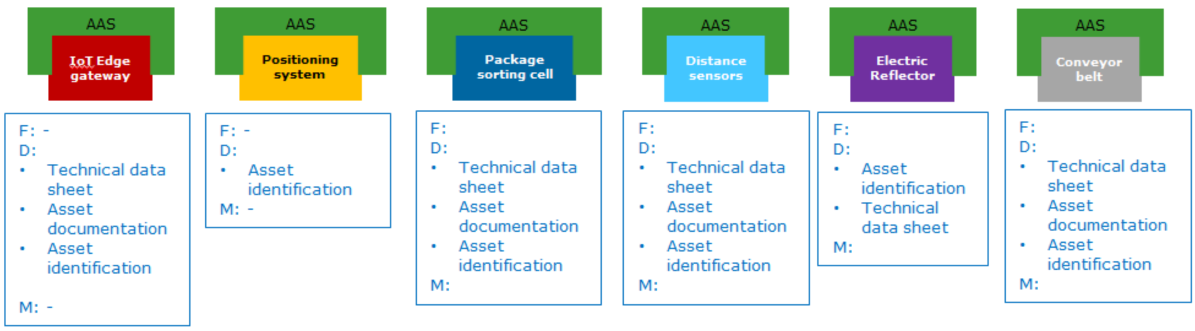

Figure 6.14: Case 2 shell overview.

6.3.4 Phase 3: Model the TO-BE situation

Modelling the TO-BE situation means carrying out the following activities: model (three) scenarios for the allocation of smartness and choose the best scenario, model the Value Network and architecture as a system of assets, model the TO-BE process(es), model asset interactions in the architecture, and specify the scenario informally.

Model scenarios for the allocation of smartness and choose the best scenario
For this case, two different scenarios are created. One in which the AAS of the parcel sorting system requests maintenance. Another, in which the individual components request maintenance. In figure 6.15 the two scenarios are drafted.The scenario that is chosen in for this case is the central request for preventive maintenance (option 1).

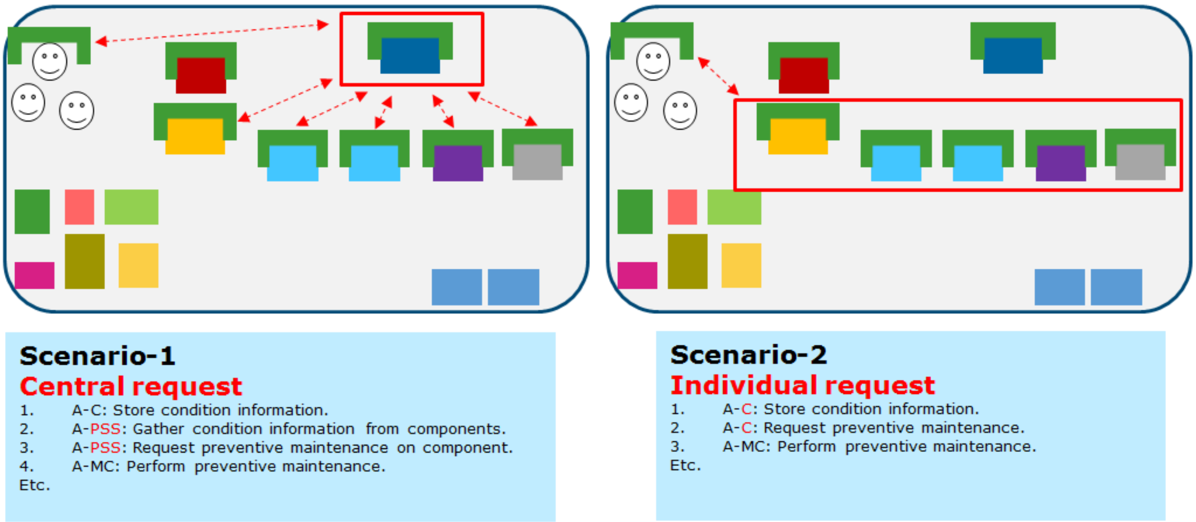

Figure 6.15: Case 2 scenarios

Model the Value Network and architecture as a system of assets

The Value Network does not change for this case. The only difference is that the maintenance company is used in a different manner. The updated system of assets can also be subtracted from figure 6.17. The only difference is that the maintenance employees get an AAS.

Model the TO-BE process(es)

The TO-BE is modeled in figure 6.16. The entire maintenance process is changed. The parcel sorting system AAS monitors the condition of the system components. Once, a threshold is reached for one of the components, maintenance is requested from the maintenance employee (company). Thereafter, the maintenance employee performs preventive maintenance.

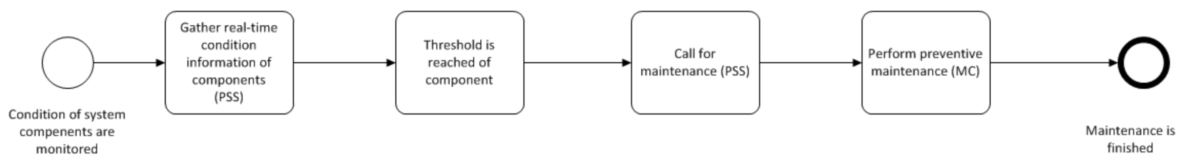

Figure 6.16: Case 2 new process

Model asset interactions in the architecture

Based on the new process, asset interactions are defined. The system components provide the parcel sorting cell with information about the process and their condition. Based on this information the parcel sorting AAS keeps track of the threshold and schedules maintenance if required. Asset interactions are modeled in the new architecture in figure 6.17.

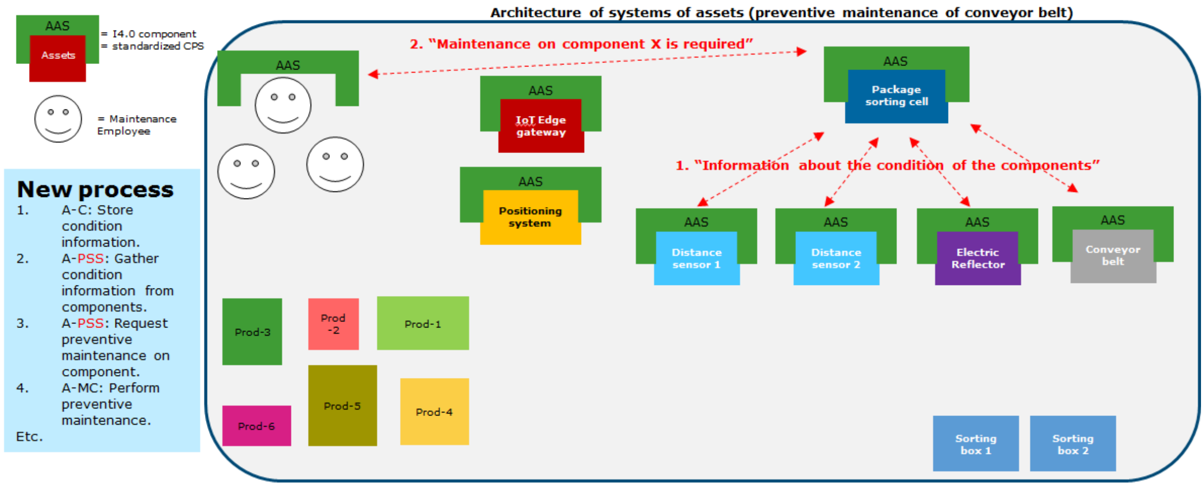

Figure 6.17: Case 2 new asset interactions in architecture

Specify the scenario informally

The data, functions, and messages of the TO-BE situation are specified informally. The content of the submodels can later be based on this information. The data, functions, and messages are visualized in figure 6.18.

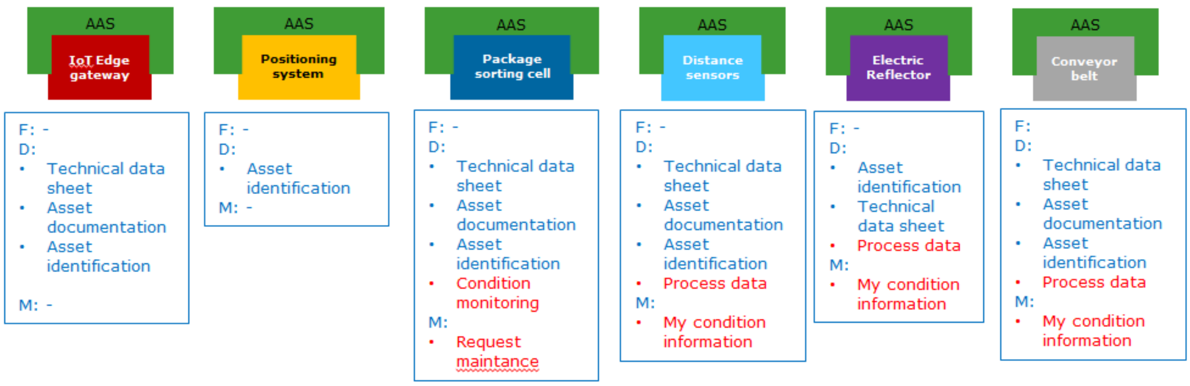

Figure 6.18: Case 2 informal specification

Chapter 7

Artifact Validation

In this chapter, the validation of the artifact is discussed. The final version of the AASDM is used to validate the ability to meet the artifact requirements and to determine its effect in its context. The context, in this case, is related to the stakeholder goals. For the validation of the artifact, two methods are used. The initial validation is carried out using a focus group, in which the AASDM is discussed by employees of the I4.0/IoT unit. First findings for the validity of the artifact are listed. Thereafter, individual semi-structured interviews are held to determine the validity of the artifact in detail.

7.1 Focus Group validation

The goal of the focus group is to validate the artifact. The focus group method is used, because 'a focus group could provide information about a range of ideas and feelings that individuals have about certain issues, as well as illuminating the differences in perspective between (groups of) individuals' (Rabiee, 2004). A focus group for artifact validation is appropriate, because it is flexible, enables direct interaction with users of the design artifact and produces large amounts of rich data for deeper understanding (Stewart & Shamdasani, 2014). Group discussions lead to a deeper understanding of the proposed concepts, which is required before the discussion and validation of the artifact.

7.1.1 Set up of the focus group

For the set up of the focus group, the focus group technique for design research projects from Tremblay, Hevner and Berndt (2010) is used. The focus group from (Tremblay et al., 2010) is based on the technique from (Bloor, 2001), (Morgan, 1996) and earlier versions of (Krueger & Casey, 2014) and (Stewart & Shamdasani, 2014).

The focus group is conducted with three participants and the researcher in a meeting room at the work location of the employees to create a comfortable setting. The focus group took two hours. The participants of the focus group are summarized in table 7.1. Together with the manager of the unit, the potential users of the AASDM were identified. The three participants of the focus group were all potential users of the AASDM. All participants are employees of the I4.0 group of the B&PS unit within Atos NL. Each of the participants has a different view on the AASDM and will be involved in different phases of the AASDM. Together they can cover all phases of the AASDM and are therefore a good sample of the stakeholder perspectives.

Table 7.1: Participants of the focus group

Participant	Function	AASDM Phases
Participant 1	Enterprise Architect / Business Consultant	(1), 2, 3, (5)
Participant 2	I4.0 / PLM Consultant	1, (2), (3), 5
Participant 3	Platform Architect / IoT Developer	(2), (3), 4

Besides the selection of the relevant stakeholders, it is important to properly plan the focus group. To guide the discussions a presentation is held by the moderator. The focus group protocol is described below in table 7.2.

Table 7.2: Set up of the focus group

Part	Moderator tasks
Part 1	Present the goal of the session and introduce the concept. Present background information on concepts around "adding smartness".
Part 2	Show the exemplar "mass customization" to the participants.
Part 3	Explain the AASDM phases and main activities.
Part 4	Discussion of the set-up of the AASDM, usability, utility, completeness, and understandability of the AASDM.

7.2 Semi-structured interviews

The focus group was only attended by three employees of the I4.0 unit. Therefore, the validation of the artifact must be strengthened. Therefore, semi-structured interviews are used for additional validation of the artifact. The combination of a focus group and semi-structured interviews is logical because these data collection methods are similar. Both focus groups and semi-structured interviews are conversational and informal tone (Longhurst, 2003). The focus of the semi-structured interviews is to validate artifact requirements and stakeholder goals as well as defining the effect of the method in its context. The approach for semi-structured interviews from (Newcomer, Hatry & Wholey, 2015) are used to conduct the interviews.

7.2.1 Set up of semi-structured interviews

The goal of the semi-structured interviews is to validate the artifact against its requirements, stakeholder goals and to determine its effect in the context. The nature of the semi-structured interviews are therefore both exploratory and confirmatory.

The set up of the semi-structured interviews is as follows. The employees that were selected for the focus group initially in collaboration with the unit manager have been invited for an interview. This means that again a mix of potential users with different perspectives on the AASDM is selected. The resulting list of respondents is shown in table 7.3.

Table 7.3: Respondents for the semi-structured interviews

Respondent	Function	AASDM phases
Respondent 1	Project Manager	All
Respondent 2	I4.0 / PLM Consultant	1, (2), (3), 5
Respondent 3	Platform Architect / IoT Developer	(2), (3), 4
Respondent 4	Application Consultant / Developer	(2), (3), 4
Respondent 5	Project Manager / Consultant	All
Respondent 6	Enterprise Architect / Business Consultant	(1), 2, 3, (5)

Thereafter, the structure of the semi-structured interviews must be determined. The format of the semi-structured interviews can be found in Appendix D. The semi-structured interview is again guided by a slide presentation both the structure of the discussion and to explain the AASDM in detail. For the interviews approximately one hour was scheduled, in reality, the interviews lasted between 38 and 67 minutes. Prior to the interviews, an e-mail was sent to the respondents with information about the interview to the respondents to establish a small knowledge base and the idea of the AASDM.

7.3 Data collection and analysis

For both the focus group and the artifact the same approach is used for the data collection and analysis. The focus group and semi-structured interviews were conducted, recorded and transcribed. After this, a coding scheme is established to analyze the raw data. The codes are categorized based on the objective of the validation. The code categories are: effect in its context, usability, utility, understandability, and completeness. The coding scheme used for analysis is shown below.

Category	Code	Description
Effect in its context	Goal	Address the goal of the AASDM
	Missing in structure	Missing components in the AASDM
	Change in structure	Possible changes to the AASDM
	Future needs	Needs for future implementation of the AASDM
	(Future) impact	Impact of the AASDM in the future
	Roles	The roles defined for the AASDM
	Relevance	Relevance of use of the AASDM
Usability	Experience	Overall experience of the AASDM, ease-of-use
Utility	Utility perception	The utility of the AASDM
Understandability	Understanding	The ease of understanding the AASDM
Completeness	Completeness of the design process	The completeness of the AASDM

Figure 7.1: Coding scheme for artifact validation focus group

7.4 Results

The results from the focus group and semi-structured interviews for the validation of the artifact are discussed here. The validation of the artifact is both to check whether the AASDM fulfils the stakeholder goals, which is roughly analyzing its effect on its context, and the artifact requirements. The artifact requirements are rated by all participants and respondents. Quotes from the transcript are used to back up the results.

7.4.1 Artifact Requirements

During the interview, the employees were asked to rate the fulfilment of the artifact requirements on a scale of 1 to 5. This gives an indication of how well the artifact requirements are fulfilled by the AASDM. The averages of their ratings are visualized in figure 7.2. The ratings indicate that the AASDM, in general, fulfils its requirements properly. The utility of the AASDM is found to be fulfilled best with a score of 4,33 on average. The usability of the AASDM was found to be the least fulfilled with a score of 3,17 on average.

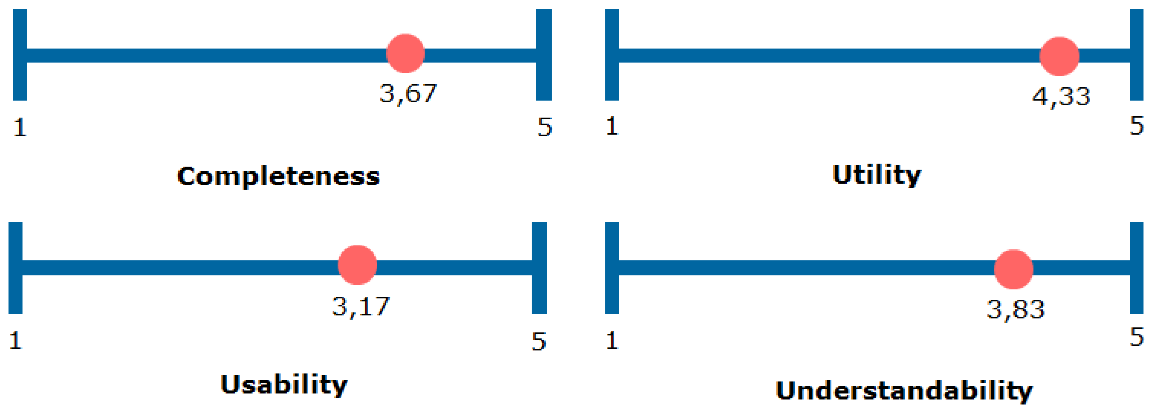

Figure 7.2: Ratings of artifact requirements

Hereafter, the results of the artifact validation in relation to the artifact requirements from both the focus group and the interview are documented. Quotes obtained from the transcripts are used to back up the results.

Artifact Requirement 1: Usability
Usability is the artifact requirement which is perceived to be fulfilled the least with a score of 3,17/5 on average. In general, the respondents found the idea of the AASDM and a service for the design of AAS a good idea. However, for it to be usable in practice today there are some concerns. As Atos is a service provider, the AASDM has to be extended to fit the broader context of a service for it to be really usable for its employees at this moment. The financial aspects, marketing aspects, sales and employee training must all be considered before the AASDM can be used in practice by the unit employees. Some quotes regarding usability are stated in the table below.

Table 7.4: Transcript: usability

Quotes
'For the AASDM to be usable in our daily practices basic maturity of our clients is required.'
'The set up of the AASDM to be used by us is good, but you have to place it in the correct technical opportunity aspect right now. '
'I would like to discuss the concept with a client of mine. Maybe to start a pilot to look into asset intelligence on the shop floor.'
'On this date, I think the knowledge of the unit is not a sufficient level to work on a project involving the AAS, I can imagine us using the AASDM as a basis for a learning course or learning instrument.'
'I think the AASDM might be of use to introduce the AAS concept to our clients.'
'From technology development/vision perspective I think it is very interesting to further develop the AASDM.'
'The AASDM would be usable if we had the clientele to use it. We must find clients that have a basic I4.0 maturity. In my clientele I do not see companies at this point where I can use the AASDM for.'

Artifact Requirement 2: utility
The utility of the AASDM for the design of networks of AAS is the second artifact requirement. The utility of the artifact is scored with a 4,33/5. During the focus group and interviews, respondents were asked how they think the AASDM satisfies this requirement. The respondents found the general structure of the AASDM well established and think that it is useful for the design of networks of AAS. One of the developers stated that tooling might be required for the AAS to be developed based on the design specifications. It is also stated that in order to work with the specifications, understanding of the underlying technologies is required. Note that the employee of the unit might not all have the proper understanding of the concept and thus tend to focus more on the overall phasing of the AASDM. Some quotes regarding utility are stated in the table below.

Table 7.5: Transcript: utility

Quotes
'Talking about the technological foundation of the AASDM I think it is useful.'
'I think you can get really inspired by the concept of the AASDM. It shows ways to use currently available technology building blocks to create a to-be state by considering scenarios.'
'The AASDM covers the most important steps to designing these kinds of systems.'
'I think the current set up steers the user towards design specifications very well.'
'The AASDM follows a standard product development model, which is good for this purpose.'
'The phasing of the AASDM is built on fairly generic steps to develop a solution.'
'For me, as a developer, it might be hard to define the information model myself. I tend to mostly use tooling for these kinds of tasks.'
'From a developers point-of-view, I assume that you must be aware of technologies like OPC UA to work with the specifications.'
'You can use the aasdm model to show the (theoretical) steps to go through to design AAS software. The model is a good reference framework.'

Artifact Requirement 3: understandability
The third requirement for the AASDM is its understandability. The AASDM must be understandable to Atos employees. The understandability of the AASDM was scored with a score of 3,83/5, which is good. One of the respondents stated that the understandability of the concept is its most important aspect. Another respondent stated that the AASDM could be used as a basis for a learning course, which indicates that the understandability of the AASDM is high according to this respondent. Some quotes regarding understandability are stated in the table below.

Table 7.6: Transcript: understandability

Quotes
'From an awareness and understanding point of view, I think the created AASDM can help a lot. This goal of the AASDM is definitely reached.'
'The AASDM looks good and I understand it, the current set up is good'
'Understandability of the idea is the most important aspect of the AASDM.'
'The AASDM is a nice way to introduce someone to the AAS concept.'
'I can imagine us using the AASDM as a basis for a learning course or learning instrument. We must use the concept to life the level of thinking within our unit.'

Artifact Requirement 4: completeness
The fourth and last artifact requirement is its completeness. Completeness of the AASDM is scored with a 3,67/5. The completeness of the AASDM is strongly related to its utility. The utility describes the ability of the AASDM to transform business requirements into AAS network design specifications. Whereas, the completeness focus mores on if the process contains all the steps. Completeness of the artifact from Atos perception is more focused on if all the steps are included in the translation of business requirements to software specifications. The AASDM fulfils this requirement. Some quotes regarding completeness are stated in the table below.

Table 7.7: Transcript: completeness

Quotes
'I think you can get really inspired by the concept of the AASDM. It shows ways to use currently available technology building blocks to create a to-be state by considering scenarios.'
'The AASDM covers the most important steps to designing these kinds of systems.'
'The AASDM looks good and I understand it, the current set up is good.'
'I think the AASDM follows a standard product development model, which is good for this purpose.'
'The phasing of the AASDM is built on fairly generic steps to develop a solution.'
'You have defined the *ist* and *soll*, this is bulletproof. You have defined the phasing as what do you have, what do you want, and what must be created or adjusted.'
'The AASDM shows a great set of minimal steps to undertake to design AAS.'
'You can use the AASDM to show the (theoretical) steps to go through to design AAS software. The model is a good reference framework.'

7.4.2 Effect in its context

The effect in its context is strongly related to the artifact requirements. The effect in its context is used to verify the stakeholder goals, which are: (1) raise awareness and understanding of the underlying AAS concept and (2) be at the basis of a guide for a service "Adding Smartness to Smart Factories". The first stakeholder goal is strongly related to requirement understandability and the second stakeholder goal is strongly related to the usability of the artifact. The effect in the context is therefore subdivided into three categories: awareness and understanding, the basis for a new service, and insights into the future.

Awareness and understanding
The first stakeholder goal is to raise awareness and understanding of the AAS concept with the AASDM. The I4.0/IoT unit and clients of the unit, in general, are not all familiar with the AAS concept. To further develop the practices of the unit, the awareness and understanding of the underlying concept must be improved. This goal corresponds to the understandability of the artifact. Some additional statements about the stakeholder goal 'raising awareness and understanding of the underlying concepts' are shown below. In the end, it can be concluded that this stakeholder goal is met.

Table 7.8: Transcript: awareness and understanding

Quotes
'The shows AASDM states the ability of AAS to share data, this would already be used as an eye-opener for lots of clients.'
'From an awareness and understanding point of view, I think the created AASDM can help a lot. This goal of the AASDM is definitely reached.'
'I can imagine us using the AASDM as a basis for a learning course or learning instrument. We must use the concept to life the level of thinking within our unit.'
'The AASDM is useful to understand the concepts and terminology of the AAS.'

Basis for a new service

The second stakeholder goal is that the AASDM should be on the basis of a new service called "Adding Smartness to Smart Factories". This goal is related to the usability because the AASDM is created for this purpose as well. It can be concluded that the AASDM is a great technological and fundamental basis for the service, but it requires activities concerning the implementation of the service to be executed as well. Financial aspects, business case aspects, marketing aspects, role and profile definition, sales training, standardized documentation, standardized collaterals, standardized documentation, proof-of-values and proofs-of-concept are all relevant when a service must be based on the AASDM. The goal of the AASDM was to be at the basis of a service and this goal is met. It can be concluded that this stakeholder goal is met.

Table 7.9: Transcript: basis for a service

Quotes
'In order for it to become more useful as a service, the service must be professionalized. Standardized documentation and collaterals (that the salespeople can take with them) must be created. Besides that marketing is also important for a service.'
'The objective of the service, marketing, the financial aspects, sales training are of great importance when creating a service. The broader context of a service must be taken into account.'
'I think the high-level start for a service is a good idea. We must take the next steps from there.'
'We need to start projects like this in practice and see what will happen, you will come across new restrictions and constraints. This way the AASDM can be improved and be transformed into a service.'
'In order for it to become more useful as a service, the service must be professionalized. Standardized documentation and collaterals (that the salespeople can take with them) must be created. Besides that marketing is also important for a service.'
'We must determine clients that have a basic I4.0 maturity.'

Additional insights

The main goal of the I4.0/IoT unit is to develop a service around the AAS concept. Additional ideas from the employees on how to improve the AASDM towards a service and its possible effect in the context are shown here. Therefore, most of the additional insights are regarding the AASDM to be implemented as a service. The integration of the IT landscape and the AAS must be clarified, interfaces might have to be developed. Clients must be identified with a basic I4.0 maturity or collaborations with clients must be initiated to start such projects. Privacy and knowledge must be guaranteed. It might be a good idea to start with a domain-specific application of the AASDM. Changes to AAS systems are very complex in practice and should be looked at carefully. The desire for a standard to develop AAS (based on design) is also stated by one of the respondents.

Table 7.10: Transcript: additional insights

Quotes
'You must address the specific IT landscape of the client in the service, so it might be better to leave more room for this in the specification of the architecture. Configuration of the AAS with the current IT landscape is important. Interfaces must be available between the AAS and the IT landscape.'
'You must think of ways to obtain the basic I4.0 maturity for your clients as well. This can be done by actually developing the software.'
'Do not only think of roles, but also profiles for the users of the AASDM. Profiles as in juniors, mediors, and seniors. Determine which mix is right for such projects.'
'Think of how the AASDM can be used for agile/scrum. Normally, you would require a proof-of-value and thereafter a proof-of-concept. With agile methods, you can transform this proof-of-concept into a minimum viable product to reach the business case.'
'In phase 2 of the development of the AASDM you can think of different domains to start implementing the AAS in.'
'How privacy and knowledge of the clients can be guaranteed is of great importance.'
'I hope in the future there will be some sort of standard of template or example of how to develop AAS. Atos must work towards this.'
'It might be a good idea to use the AASDM and fit it for specific domains. I think this is smart because between the sectors there is a great difference in knowledge level.'
'Changes to a system of AAS might be very complex, how to address this in practice is something we must give a lot of thoughts.'

7.5 Conclusions

The validation of the artifact concludes that the artifact requirements and stakeholder goals are met. The usability, utility, completeness, and understandability of the AASDM are validated. The stakeholder goals to raise awareness and understanding of the AAS concept and for the AASDM to be at the basis of a service are validated as well. With the sub-research question 5 is answered. The effect of the artifact in its context is determined as well. For the I4.0/IoT unit of Atos NL, the artifact will function as a learning tool or conceptual idea for a new possible service. It might be used to increase the knowledge of the unit employees and needs further development to be implemented as a service. Herewith, the sixth and last sub-research question is answered.

Chapter 8

Conclusion and outlook

This chapter concludes the main results of the Master Thesis project, describes its contributions, reflects on the project as a whole and defines future research directions.

8.1 Conclusion

This Master Thesis project was focused on the Asset Administration Shell concept. The goal of the research project was to develop a method for the design of new I4.0 component software, Asset Administration Shells, that are needed to realize new I4.0 related business requirements for the incremental transformation to Smart(er) Factories. For this purpose the AASDM is developed and answers the main research question around which the project was built:

Main Research Question (MRQ): 'How can the (re)design process of Asset Administration Shells be guided to realize new smartness in Smart Factories?'

The research project was conducted according to the Design Science Methodology from Wieringa (2014). For each of the three phases of the Design Science Cycle, sub-research questions have been defined to structurally design the artifact (AASDM). Figure 8.1 shows a replica of figure 1.5, in which the main research question and sub-research questions are shown in relation to the Design Science phases.

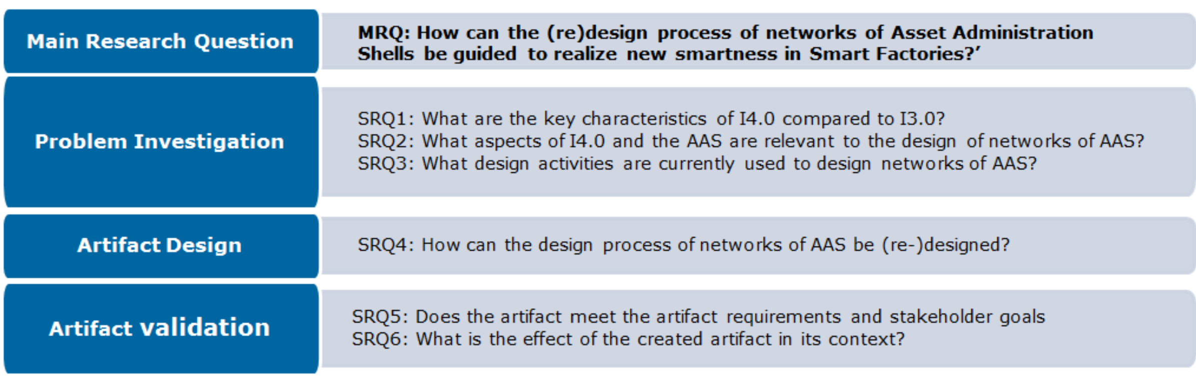

Figure 8.1: Research Questions

Firstly, a Systematic Literature Review is performed to establish the knowledge base for this project. The SLR aims to answer the first and second sub-research questions. The AAS concept and the surrounding I4.0 concept are analyzed. The key differences between I4.0 and I3.0 are highlighted and the design aspects of both I4.0 and the AAS are explained. In the end, an overview of all the relevant design aspects of the AAS is given. The relevant design aspects are the AAS information model, AAS communication, and the I4.0 system architecture. These three aspects are formed based on the underlying I4.0 design principles and RAMI4.0. From the Systematic Literature Review, it can also be concluded that no method for the design of (networks of) AAS exists in the literature.

Secondly, the current design activities and process are analyzed. For this, implementations of AAS were investigated. In practice, there is no full-functioning Smart Factory that has implemented AAS on a large scale. Likewise, Atos NL does not have AAS implementations currently. This justifies the fact that an observational case study is set up to analyze documents from three well-known I4.0 demonstrators (cases) that have implemented AAS. Their design process and activities are analyzed using the design aspects from the theoretical background. Based on this, the third sub-research question is answered. Furthermore, the fourth research question is also answered during the problem investigation phase and its answer follows from discussions with a representative of the I4.0/ IoT unit of Atos NL. The need for a structured, complete, understandable, useful approach to design networks of AAS becomes obvious from the observational case studies. From Atos' point-of-view the method must be used as a basis for a service and help to raise awareness within the I4.0/IoT unit.

The goal of the artifact design phase is to determine a way to guide the process of designing networks of AAS based on business requirements. To guide this process, the idea was to define the AASDM. To develop the AASDM an adjusted situational method engineering approach of (Harmsen et al., 1994) is used. Based on the findings from the Problem Investigation phase the artifact requirements can be defined. Thereafter, in close collaboration with a representative of the unit a conceptual model is created as the method outline. Hereafter, the method is assembled based on method fragments selected from literature and the studied cases. The result is the AASDM, a method for the translation of business requirements into AAS network design specifications. With this, the fourth sub-research question is answered.

Next, exemplary applications of the AASDM are carried out to show the use of the AASDM in detail. Exemplars are used because no real-world case is available. The exemplars are created to model a realistic scenario of the real-world. The first exemplary application of the AAS adds driverless vehicles to reach a form of so-called mass customization. The second exemplar shows a less complex case in which condition monitoring enables preventive maintenance. Both exemplary applications are based on one of the demonstrators as the current state.

In the end, the artifact is validated against its artifact requirements and stakeholder goals. For the artifact validation, a focus group and individual semi-structured interviews are set up. The artifact requirements are usability, utility, completeness, and understandability. The artifact validation shows that these artifact requirements are met by the AASDM. The best rating was given to the utility of the AASDM. The lowest rating was given to the usability of the AASDM, which is logical since its purpose is to function as a basis for a service, not yet a completed service. The stakeholder goals are: raise awareness and understanding of the AAS concept in the unit and form a basis for a new service. To see if the AASDM fulfils the stakeholder goals its effect for the I4.0/IoT unit must be determined. From the focus group and semi-structured it can be concluded that the stakeholder goals are met with the comment that the next step for the AASDM to be usable is to establish the bigger context of the service. The validation of the artifact answers the fifth and sixth sub-research questions.

8.2 Contributions

At first, this Master Thesis research contributes to the literature by designing a structured approach to design Asset Administration Shell. The theoretical background and observational case studies showed that there was a lack of consistency and completeness in the current design processes of the Asset Administration Shell. The design activities and related design aspects of the AAS are combined in the AASDM. With this structured approach, researchers can create and design AAS in a structured manner. Manufacturing companies can start developing AAS implementations or improve their Smart Factories by using the AASDM. Consultancy companies can use the AASDM to create services around the AAS. The AASDM provides the technological background for these type of projects.

Second, the AASDM provides possibilities for new business value within Atos NL (and other consultancy companies). Atos can introduce the AASDM to increase the knowledge of clients of the AAS and provide guidance in designing the new systems or help improve their manufacturing systems. It provides a pragmatic, structured and complete approach to tackle AAS related improvements.

Third, the AASDM can be used to define free submodels for the AAS. Free submodels can, if agreed upon, become basic submodels by defining a standard for it. This contributes to the standardization of the AAS and thus current AAS research.

8.3 Reflection

A research project is (almost) never flawless. Hence, it is good to determine the shortcomings and improvement points of this research project.

The Master Thesis project was not easy to be carried out. Various influences had an effect on the choices made during the research. During problem investigation, no interviews or other data collection method is used to determine the problem from various perspectives within the I4.0/IoT unit. This is justified by the fact that the knowledge on the AAS from the I4.0/IoT unit is very limited currently. As is stated by the unit representative, this research should contribute to improved awareness and understanding of the AAS. Also for the validation of the artifact in the focus group and semi-structured interviews, it must be noted that the knowledge of the interviewees on AAS is mostly limited.

The focus of the research was on the Asset Administration Shell concept. This concept is under-researched in general. As became clear from the Systematic Literature Review only 17 documents could be found in the electronic databases selected. This can be explained due to the fact that the AAS is the idea of the Plattform I4.0 (and its partner companies) and was introduced for the first time in 2016, not even four years ago. The literary basis for this Master Thesis project is thus based on the idea of a small research community.

Another limitation of the research is that the research on the and standardization for the AAS are still ongoing. The developments around the AAS and its communication are still in their infancy. As is stated, the concept is only nearly four years old and has a lot of improvements to undergo. This research is based on the current concept, which is not yet optimized.

Mapping to the RAMI4.0 is mentioned as one of the design activities in one of the cases in the observational case study. RAMI4.0 is also determined as one of the core underlying aspects relevant to the design of networks of AAS. However, in the final version of the AASDM, no reference or mapping to the RAMI4.0 architecture is made. Even though the aspects of RAMI4.0 are incorporated in other activities, it might be a possible point of improvement.

The use of exemplars over real case studies (implementations) can be considered as a weakness of this research. Although, the use of exemplars is justified in this case due to the absence of a real-world factory. And even though, the exemplars are prepared and executed truthfully and in real-world settings, a real case study seems better to get to know the complexity of practice. New limitations and restrictions will arise from using the AASDM in practice.

8.4 Future research

Firstly, the generalizability of the AASDM can be improved by using the AASDM for real case studies. As is stated in the reflection, the AASDM is used to solve exemplary cases. Next, the AASDM should be used in a real-world setting. The next step could be to contact Atos' clients and to set up a pilot program for the implementation of AAS. Another option is to contact the companies that have implemented currently.

Secondly, the focus of the AASDM is on the design of networks of AAS. What is not addressed is the development and implementation of AAS networks. For this software developers must be contacted and proof-of-values and proofs-of-concept of the AAS networks must be created. This can also be done in collaboration with Atos' clients.

Thirdly, Atos should work towards a real service for the AAS. The AASDM can be used as a starting point. To transform the AASDM into a service, the broader context of a service must be investigated. The business case aspect, financial aspect, and marketing aspect etc. of the AASDM must be addressed. A training program and redefinition of roles (and profiles) within the I4.0/IoT unit might be worth investigating. Thereafter, Atos would have to collaborate with its clientele to set-up pilot programs.

Lastly, the research and standardization of the AAS is still an ongoing process. Trends and innovations in the research should be monitored closely. This way possible improvements to the AASDM will become clear. With the use of the AASDM Atos might also be able to contribute to the development of the AAS.

References

Adolphs, P., Auer, S., Bedenbender, H., Billmann, M., Hankel, M., Heidel, R., … Kiele, M. et al. (2016). Structure of the administration shell. continuation of the development of the reference model for the industrie 4.0 component. *ZVEI and VDI, Status Report.*

Adolphs, P. [Peter], Bedenbender, H. [Heinz], Dirzus, D., Ehlich, M., Epple, U., Hankel, M., … Kärcher, B. et al. (2015). Reference architecture model industrie 4.0 (rami4. 0). *ZVEI and VDI, Status report.*

Alcácer, V. & Cruz-Machado, V. (2019). Scanning the industry 4.0: A literature review on technologies for manufacturing systems. *Engineering Science and Technology, an International Journal.*

Bär, K., Herbert-Hansen, Z. N. L. & Khalid, W. (2018). Considering industry 4.0 aspects in the supply chain for an sme. *Production Engineering*, *12*(6), 747–758.

Bedenbender, H., Bentkus, A., Epple, U., Hadlich, T., Hankel, M., Heidel, R., … Woolschlaeger, M. (2018). Relationships between i4.0 components - composite components and smart production. *ZVEI, Working Paper.*

Belyaev, A. & Diedrich, C. (2019). Specification "demonstrator i4.0-language"v3.0. *Technical Report.*

Bloor, M. (2001). *Focus groups in social research.* Sage.

Bowen, G. A. (2009). Document analysis as a qualitative research method. *Qualitative research journal*, *9*(2), 27–40.

Brinkkemper, S. (1996). Method engineering: Engineering of information systems development methods and tools. *Information and software technology*, *38*(4), 275–280.

Diedrich, C. [Christian], Belyaev, A. [Alexander], Bock, J., Gössling, A., Hänisch, R., Kraft, A., … Vollmar, F. et al. (2017). Interaktionsmodell für industrie 4.0 komponenten. *at-Automatisierungstechnik*, *65*(1), 5–18.

Feather, M. S., Fickas, S., Finkelstein, A. & Van Lamsweerde, A. (1997). Requirements and specification exemplars. *Automated Software Engineering*, *4*(4), 419–438.

Felch, V., Asdecker, B. & Sucky, E. (2019). Maturity models in the age of industry 4.0–do the available models correspond to the needs of business practice? In *Proceedings of the 52nd hawaii international conference on system sciences.*

Ferris, T., Barker, S. & Adcock, R. (2016). Scenario selection method for system scenario analysis. INCOSE.

Harmsen, A. F., Brinkkemper, J. N. & Oei, J. H. (1994). *Situational method engineering for information system project approaches.* Citeseer.

Hermann, M., Pentek, T. & Otto, B. (2016). Design principles for industrie 4.0 scenarios. In *2016 49th hawaii international conference on system sciences (hicss)* (pp. 3928–3937). IEEE.

Hevner, A. R. (2007). A three cycle view of design science research. *Scandinavian journal of information systems*, *19*(2), 4.

Hevner, A. R., March, S. T., Park, J. & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, *28*(1), 75–105.

Hoffmeister, M. [Michael]. (2015). The industrie 4.0 component. *ZVEI.*

Hu, V. C., Ferraiolo, D., Kuhn, R., Friedman, A. R., Lang, A. J., Cogdell, M. M., … Scarfone, K. et al. (2013). Guide to attribute based access control (abac) definition and considerations (draft). *NIST special publication*, *800*(162).

Jaakkola, H. & Thalheim, B. (2010). Architecture-driven modelling methodologies. In *Ejc* (pp. 97–116).

Kagermann, H., Lukas, W.-D. & Wahlster, W. (2011). Industrie 4.0: Mit dem internet der dinge auf dem weg zur 4. industriellen revolution. *VDI nachrichten*, *13*(1).

Kagermann, H., Wahlster, W. & Helbig, J. (2013). *Recommendations for implementing the strategic initiative industrie 4.0 – securing the future of german manufacturing industry.* acatech – National Academy of Science and Engineering. München.

Kitchenham, B. & Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering.

Krueger, R. A. & Casey, M. A. (2014). *Focus groups: A practical guide for applied research.* Sage publications.

Lasi, H., Fettke, P., Kemper, H.-G., Feld, T. & Hoffmann, M. (2014). Industry 4.0. *Business & information systems engineering*, *6*(4), 239–242.

Lee, J., Bagheri, B. & Kao, H.-A. (2015). A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing letters*, *3*, 18–23.

Leyh, C., Bley, K., Schäffer, T. & Forstenhäusler, S. (2016). Simmi 4.0-a maturity model for classifying the enterprise-wide it and software landscape focusing on industry 4.0. In *2016 federated conference on computer science and information systems (fedcsis)* (pp. 1297–1302). IEEE.

Lin, S.-W., Miller, B., Durand, J., Joshi, R., Didier, P., Chigani, A., ... Bleakley, G. et al. (2015). Industrial internet reference architecture. *Industrial Internet Consortium (IIC), Tech. Rep.*

Longhurst, R. (2003). Semi-structured interviews and focus groups. *Key methods in geography*, *3*, 143–156.

Löwen, U., Braune, A., Diesner, M., Huettemann, G., Klein, M., Thron, M., ... Okon, M. (2016). Industrie 4.0 components – modeling examples. *ZVEI and VDI, Status report.*

Lüder, A., Schleipen, M., Schmidt, N., Pfrommer, J. & Henßen, R. (2017). One step towards an industry 4.0 component. In *2017 13th ieee conference on automation science and engineering (case)* (pp. 1268–1273). IEEE.

Marcon, P., Diedrich, C., Zezulka, F., Schröder, T., Belyaev, A., Arm, J., ... Vesely, I. (2018). The asset administration shell of operator in the platform of industry 4.0. In *2018 18th international conference on mechatronics-mechatronika (me)* (pp. 1–5). IEEE.

Marseu, E., Kolberg, D. & Weyer, S. (2017). Exemplary transfer of the rami 4.0 administration shell to the smartfactory kl system architecture for industrie 4.0 production systems. *White paper SF-2.1*, *4*.

Monostori, L., Kádár, B., Bauernhansl, T., Kondoh, S., Kumara, S., Reinhart, G., ... Ueda, K. (2016). Cyber-physical systems in manufacturing. *Cirp Annals*, *65*(2), 621–641.

Morgan, D. L. (1996). *Focus groups as qualitative research.* Sage publications.

Newcomer, K. E., Hatry, H. P. & Wholey, J. S. (2015). Conducting semi-structured interviews. *Handbook of practical program evaluation*, *492*.

Oztemel, E. & Gursev, S. (2018). Literature review of industry 4.0 and related technologies. *Journal of Intelligent Manufacturing*, 1–56.

Pepels, E. K. K. (2019). A software design perspective on industrie 4.0 components. Eindhoven University of Technology.

Pérez-Lara, M., Saucedo-Martìnez, J. A., Salais-Fierro, T. E., Marmolejo-Saucedo, J. A. & Vasant, P. (2019). Vertical and horizontal integration systems in industry 4.0. In *Innovative computing trends and applications* (pp. 99–109). Springer.

Pethig, F., Niggemann, O. & Walter, A. (2017). Towards industrie 4.0 compliant configuration of condition monitoring services. In *2017 ieee 15th international conference on industrial informatics (indin)* (pp. 271–276). IEEE.

Platform-I4.0. (2019). Verwaltungsschale in der praxis. *Plattform I4.0, discussion paper.*

Plattform I4.0. (2018a). Details of the asset administration shell, part 1: Part 1- the exchange of information between partners in the value chain of industrie 4.0 (version 1.0). *Plattform I4.0, specification.*

Plattform I4.0. (2018b). I4.0-sprache: Vokabular, nachrichtenstrukture und semantische interaktionsprotokolle der i4.0-sprache. *Plattform I4.0, discussion paper.*

Plattform I4.0, Alliance Industrie du Futur & Piano Nationale Impresa I4.0. (2018). The structure of the asset administration shell: Trilateral perspectives from france, italy and germany. *International Paper.*

PlattformI4.0. (2019). Details of the administration shell - from idea to implementation. Slide presentation. Plattform I4.0.

Prinz, F., Schoeffler, M., Lechler, A. & Verl, A. (2018). Dynamic real-time orchestration of i4. 0 components based on time-sensitive networking. *Procedia CIRP*, *72*, 910–915.

Rabiee, F. (2004). Focus-group interview and data analysis. *Proceedings of the nutrition society*, *63*(4), 655–660.

Radziwon, A., Bilberg, A., Bogers, M. & Madsen, E. S. (2014). The smart factory: Exploring adaptive and flexible manufacturing solutions. *Procedia engineering*, *69*, 1184–1190.

Ragunath, P., Velmourougan, S., Davachelvan, P., Kayalvizhi, S. & Ravimohan, R. (2010). Evolving a new model (sdlc model-2010) for software development life cycle (sdlc). *International Journal of Computer Science and Network Security*, *10*(1), 112–119.

Rao, S. K. & Prasad, R. (2018). Impact of 5g technologies on industry 4.0. *Wireless personal communications*, *100*(1), 145–159.

Saucedo-Martínez, J. A., Pérez-Lara, M., Marmolejo-Saucedo, J. A., Salais-Fierro, T. E. & Vasant, P. (2018). Industry 4.0 framework for management and operations: A review. *Journal of ambient intelligence and humanized computing*, *9*(3), 789–801.

Sommerville, I. (2011). *Software engineering.* Addison-wesley.

Spec, D. (2016). 91345: Reference architecture model industrie 4.0 (rami4. 0). *DIN Std. DIN SPEC*, *91*(345), 04.

Stewart, D. W. & Shamdasani, P. N. (2014). *Focus groups: Theory and practice.* Sage publications.

Strandhagen, J. W., Alfnes, E., Strandhagen, J. O. & Vallandingham, L. R. (2017). The fit of industry 4.0 applications in manufacturing logistics: A multiple case study. *Advances in Manufacturing*, *5*(4), 344–358.

Tantik, E. & Anderl, R. (2017a). Integrated data model and structure for the asset administration shell in industrie 4.0. *Procedia CIRP*, *60*, 86–91.

Tantik, E. & Anderl, R. (2017b). Potentials of the asset administration shell of industrie 4.0 for service-oriented business models. *Procedia CIRP*, *64*, 363–368.

Tremblay, M. C., Hevner, A. R. & Berndt, D. J. (2010). The use of focus groups in design science research. In *Design research in information systems* (pp. 121–143). Springer.

Wang, S., Wan, J., Li, D. & Zhang, C. (2016). Implementing smart factory of industrie 4.0: An outlook. *International Journal of Distributed Sensor Networks*, *12*(1), 3159805.

Weber, C., Königsberger, J., Kassner, L. & Mitschang, B. (2017). M2ddm–a maturity model for data-driven manufacturing. *Procedia CIRP*, *63*, 173–178.

Westermann, T., Dumitrescu, R. et al. (2018). Maturity model-based planning of cyber-physical systems in the machinery and plant engineering industry. In *Ds 92: Proceedings of the design 2018 15th international design conference* (pp. 3041–3052).

Wieringa, R. J. (2014). *Design science methodology for information systems and software engineering.* Springer.

Zezulka, F., Marcon, P., Vesely, I. & Sajdl, O. (2016). Industry 4.0–an introduction in the phenomenon. *IFAC-PapersOnLine*, *49*(25), 8–12.

Appendix A

Chapter 2

A.1 Quality and Relevance criteria SLR

Quality and relevance criteria used to shorten the long list of documents into the short list. The literature on I4.0 and the AAS were split up therefore different quality and relevance criteria have been determined for I4.0 literature and AAS literature. See figures A.1 - A.4.

Relevance inclusion and exclusion criteria for the I4.0 design

1. Does the article have the terms CPS, (I)IoT, smart factory or Industry 4.0 in the keywords or title? (I)
2. Does the article concern the manufacturing industry? (I)
3. Does the article provide a clear overview of Industry 4.0 core concepts, challenges, goal or potential, requirements, frameworks, architecture or a design related topic? (I)
4. The article focuses on a specific application of Industrie 4.0 without providing a decent applicability in general or for the manufacturing industry? (E)
5. The article focuses on technically implementing algorithms/methods of Industry 4.0? (E)

Figure A.1: Relevance criteria I4.0 literature

Relevance inclusion and exclusion criteria for the AAS

1. The article contains the (Asset) Administration Shell in the keywords or title? (I)
2. The topic is focused on the structure or architecture of the AAS.
3. The article focuses on a specific application of the Asset Administration Shell without including general applicability? (E)

Figure A.2: Relevance criteria AAS literature

Quality inclusion and exclusion criteria for I4.0

1. Is the language appropriate? (I)
2. Is the document available in full-text? (I)
3. Is the article a stand-alone article, is it useful without other documents? (I)
4. Addresses at least one section to answer one of the research (sub-) questions? (I)

Figure A.3: Quality criteria I4.0 literature

Quality inclusion and exclusion criteria for AAS

1. Is the full-text version of the document available? (I)
2. Does the article contain at least one section on the structure, architecture or requirements of the AAS? (I)

Figure A.4: Quality criteria AAS literature

A.2 Meta-information model of the AAS

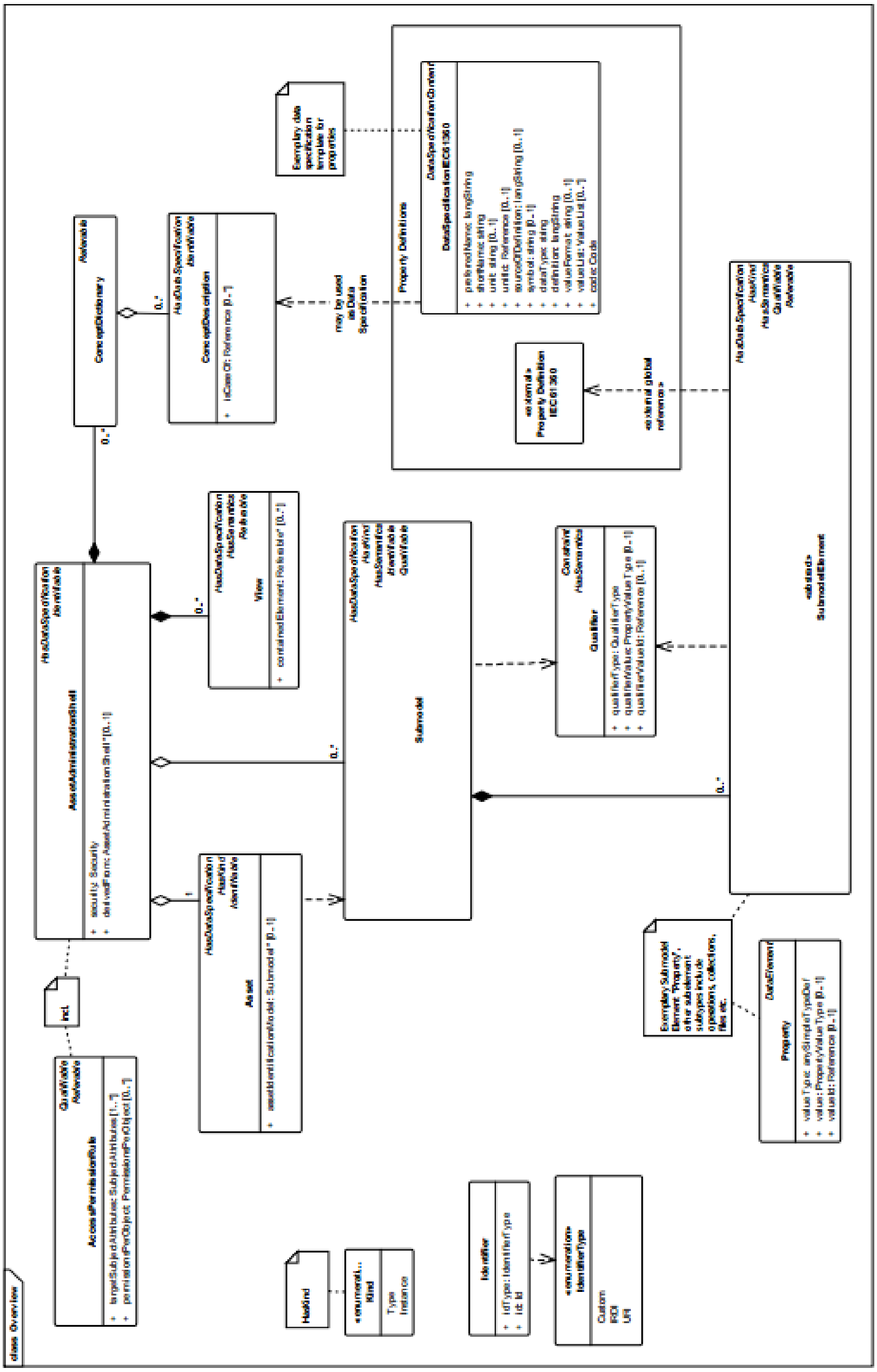

Figure A.5: The meta-information model of the AAS as specified in (Plattform I4.0, 2018a).

A.3 AAS identifiers

Identifiable	Attribute	Allowed identifier	Remarks
AAS	Id	URI	Mandatory
	IdShort	String	n/a
Asset	Id	URI	Mandatory
	IdShort	String	Mandatory
Submodel type	Id	URI	Mandatory
	IdShort	String	Mandatory
	SemanticId	IRDI, URI	Optional
Submodel instance	Id	URI, custom	Mandatory
	IdShort	String	Mandatory
	SemanticId	IRDI, URI	Optional
Submodel element	SemanticId	IRDI, URI, Custom	Mandatory
	IdShort	String	Mandatory
Concept Description	isCaseOf	IRDI, URI	Optional
	SemanticId	n/a	n/a
View Qualifier	SemanticId	IRDI, URI	Optional
	IdShort	String	Mandatory
	SemanticId	IRDI, URI, Internal	Optional

Figure A.6: Identifiables, attributes, allowed identifiers in (Plattform I4.0, 2018a).

A.4 Basic Views DIN SPEC 91345

Basic view	Best practice/example
Business	Data and functions are deposited which allow judging on the business suitability and performance of a component in the life cycle phases Procurement, Design, Operation and Realisation. Examples: prices, terms of delivery, order codes
Constructive	Contains properties relevant for the constructive deployment of the component, thus for selection and building structure. Contains a structure classification system pursuant to EN 81346. Contains numerous properties in respect of physical dimensions and regarding start, processing and output values of the component. Contains a modular view of subcomponents or a device structure. Allows an automation view with inputs and outputs of different signal types.
Performance	Describes performance and behavioural characteristics in order to allow a summary assessment and Virtual Commissioning (V-IBN) of an overall system.
Functional	Makes statements on the function pursuant to EN 81346 and on the function of the subcomponents. Here location of the individual functions of the Technical Functionality also takes place, thus for example so-called “skills”, interpretation, commissioning, calculation or diagnosis functions of the component.
Local	Makes statements on positions and local relationships between the component or its parts or inputs and outputs.
Security	Can identify a property as security-relevant. This property should be taken into account for an assessment of security.
Network view	Makes statements in respect of electrical, fluidic, materials flow-related and logical cross-linking of the component.
Life cycle	Contains data on the current situation and historical utilisation in the life cycle of the component. Examples: allocation to production, maintenance protocols and past applications.
Human	In all views properties, data and functions should appear such that humans can understand individual elements, inter-relationships and causal chains.

Figure A.7: Basic Views according to DIN SPEC 91345 (RAMI4.0)

A.5 Exemplary submodels

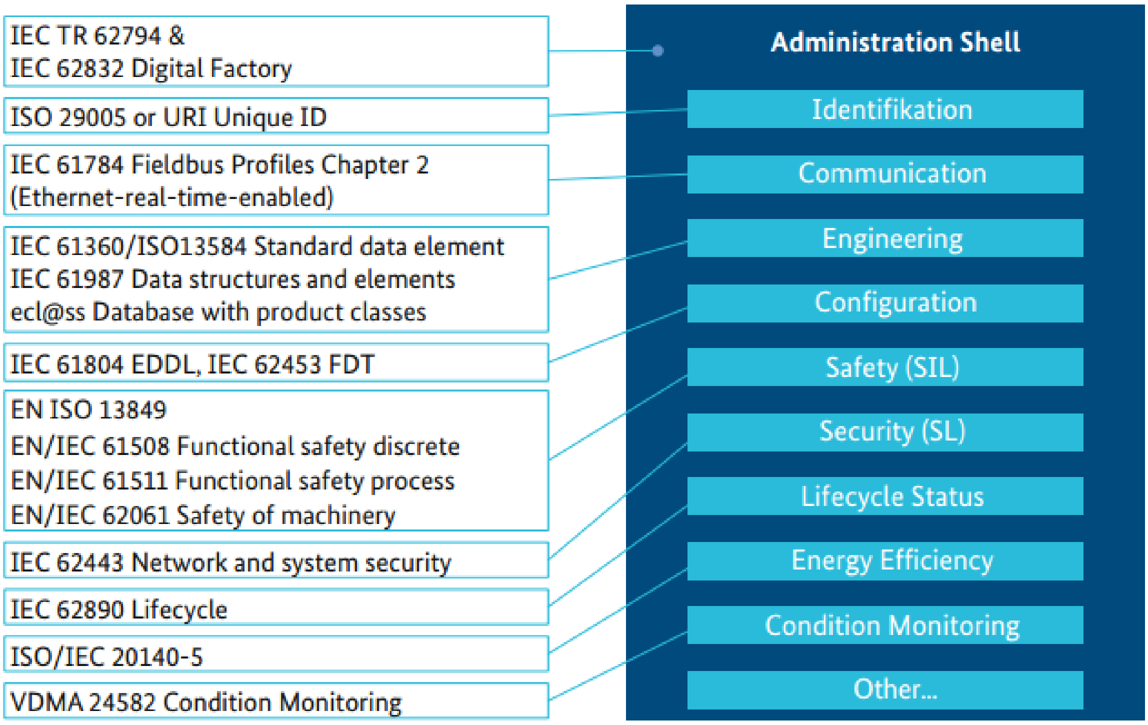

Figure A.8: Exemplary AAS submodels with their link to existing standards.

Appendix B

Chapter 3

B.1 Long list of cases selected from literature

The long list of potential cases for the observational case study obtained from literature.

Table B.1: Case selection

No	Case Name	Case Description	Description source
1	SmartFactory^{KL}	The SmartFactory^{KL} partners consortium consisting of 45 partners in industry and research have created an Industrie 4.0 production plant. Concepts such as flexible and more efficient production and innovative information and communication technologies are tested in practice.	https://smartfactory.de/en/
2	I4.0 Bottling Demonstrator	This I4.0 Demonstrator is created around the multi-carrier system from Siemens and Festo. The demonstrator applies three application scenarios of I4.0: the adaptable factory (AF), value-based service (VBS), and seamless and dynamic plant engineering (SDP).	https://new.siemens.com/global/en/markets/machinebuilding/multi-carrier-system.html AND (Löwen et al., 2016)
3	Plattform I4.0 HMI 2018/ 2019 Demonstrator	In the years 2018 and 2019 the Plattform I4.0 has demonstrated an example implementation of the Asset Administration Shell for a package sorting system.	(Platform-I4.0, 2019) AND https://www.plattform-i40.de/PI40/Navigation/EN/Home/home.html
4	Central Remote Maintenance Platform	A proof-of-concept in the form of a robot arm which is provided with an Asset Administration Shell is created.	(Tantik & Anderl, 2017a)
5	Simple control of a robot arm	A use case is developed in which the control of a robot arm is connected to a Graphical User Interface (GUI) and an X-box controller for the simple control of the robot arm. These are all integrated in an Asset Administration Shell.	(Tantik & Anderl, 2017b)

6	AAS of an operator	This demonstrator shows the implementation of an AAS for an operator presented in a Human-Machine Interface. A smart jacket is attached to this AAS in which air quality information, temperature, and distance to faults in the manufacturing process are obtained through sensors.	(Marcon et al., 2018)
7	Condition monitoring service	A use case scenario is presented for a work cell (pick and place system from Lenze). The Condition Monitoring service is performed for the servo motor in the work cell. Information about the torque and its threshold information are stored in the AAS.	(Pethig, Niggemann & Walter, 2017)
8	Time-sensitive Networking (TSN)	Three use cases are described for two real-time axes based on their defined real-time apable I4.0 framework.	(Prinz, Schoeffler, Lechler & Verl, 2018)

B.1.1 Case selection

Not all cases are examined in this Master Thesis project. From the long list of potential cases to be analyzed a short list is created and analyzed in chapter 3. The selection of cases is based on the following criteria For the selection of the cases the following criteria are handled: (1) the case must describe a real-world like manufacturing system with multiple components and (2) Activities undertaken in the design of the AAS must be documented. Table B.2 shows the results. Cases 1,2, and 3 are the only cases that pass both constraints and are therefore analyzed in the observational case study.

Table B.2: Case selection Criteria

Case No	**C1**	**C2**
1	✓	✓
2	✓	✓
3	✓	✓
4	✓	X
5	X	X
6	X	X
7	✓	X
8	✓	X

B.2 Document Analysis

The documents (and websites) used to investigate the cases are listed below. The documents are categorized in the following categories: (1) I4.0 system architecture, (2) AAS information model, (3) AAS communication, (4) generic information, (5) RAMI4.0.

Case	Document	Category	Type
SmartFactory^{KL}	https://smartfactory.de/en/	4	Website
	White paper SF-1.1	1	Pdf
	White paper SF-1.2	1	Pdf
	White paper SF-2.1	2, 5	Pdf
	Dennis Kolberg presentation	1, 4	Slides
	Wolfgang Wahlster presentation	1, 4	Slides
Bottling plant (Siemens & Festo)	https://new.siemens.com/global/en/markets/machinebuilding/multi-carrier-system.html	4	Website
	Modelling Examples	1, 2, 4, 5	Pdf
HMI Demonstrator	The AAS in practice	1, 2, 3, 4, 5	Pdf

Figure B.1: List of documents

Appendix C

Chapter 4

In this appendix, the results of the meetings with the Senior Business Consultant / Enterprise Architect are documented. Pictures of the white board after the sessions were made. The goal and outcome of the sessions are written below the pictures.

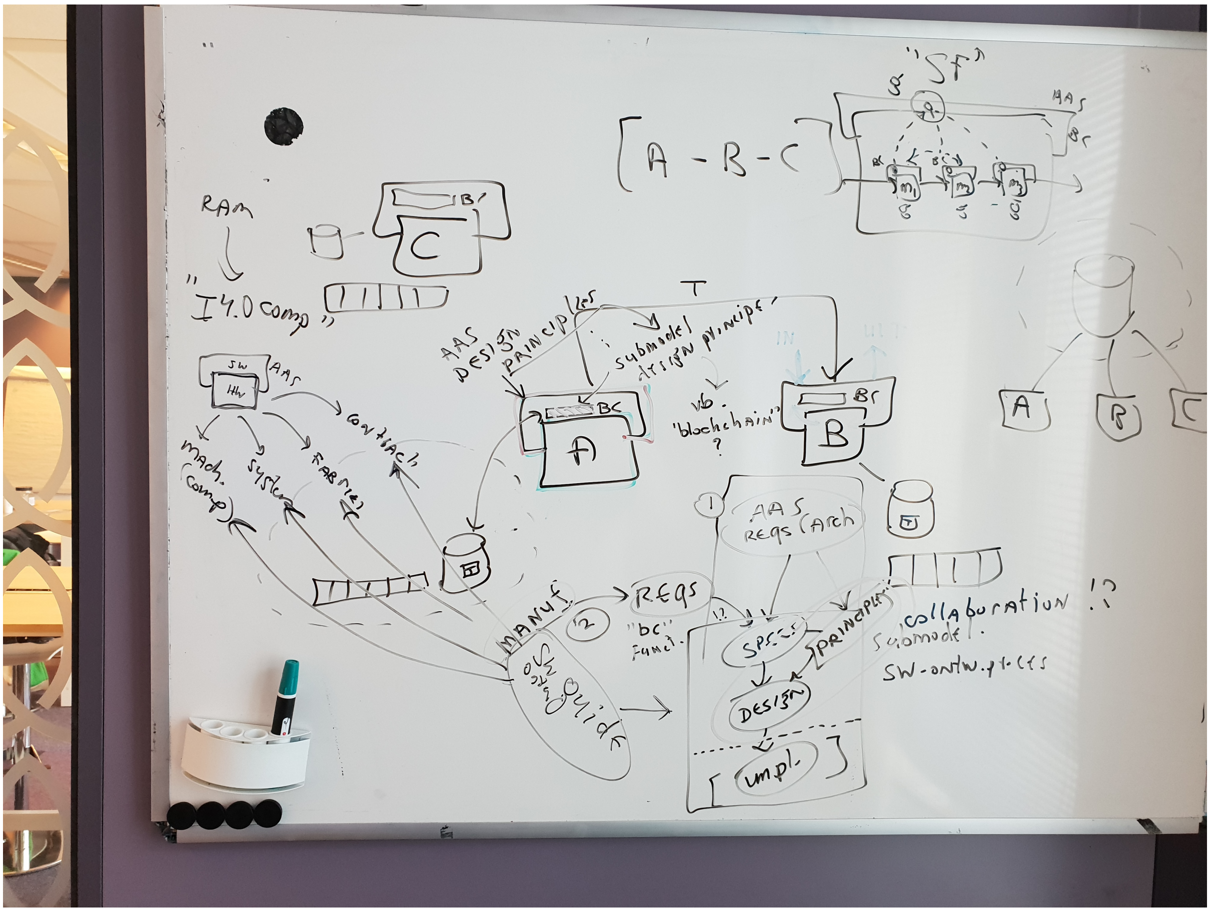

Figure C.1: Discussion 1

Discussion on: design principles, the translation of requirements to design specifications, the architecture as system of assets.

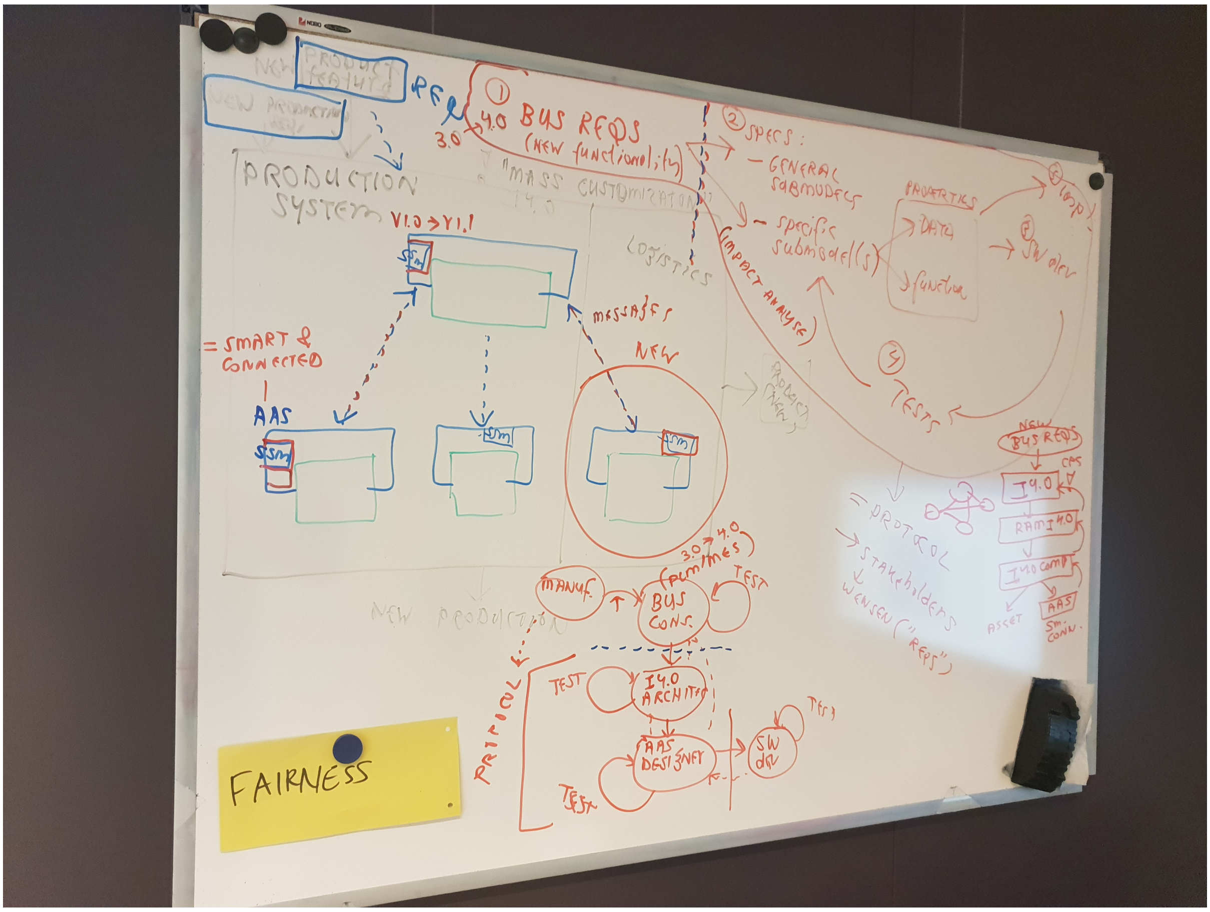

Figure C.2: Discussion 2

Redefining the general structure of the AASDM (version 2). Based on the business requirements and design specifications. Definition of the roles in the AASDM.

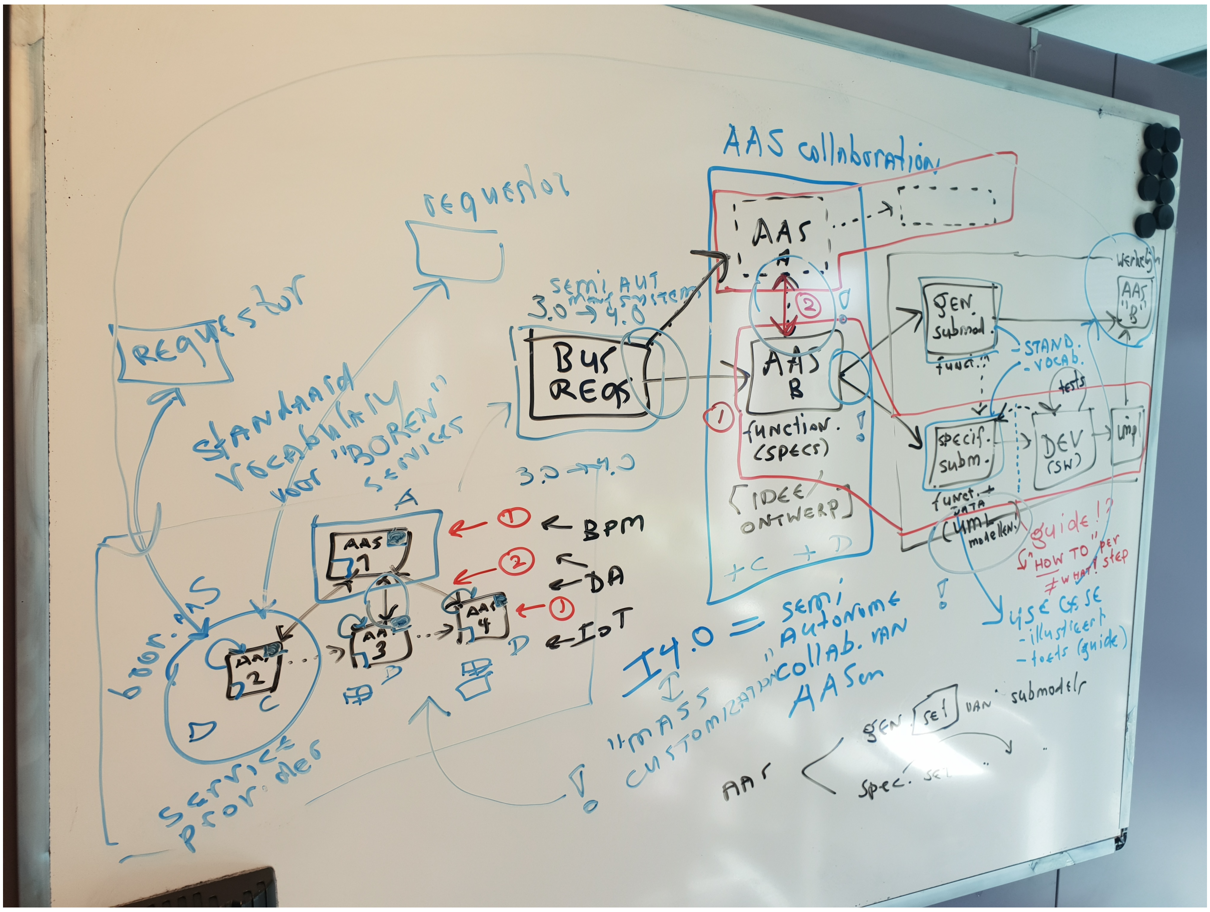

Figure C.3: Discussion 3

Redefining the general structure (version 3). More details are added.

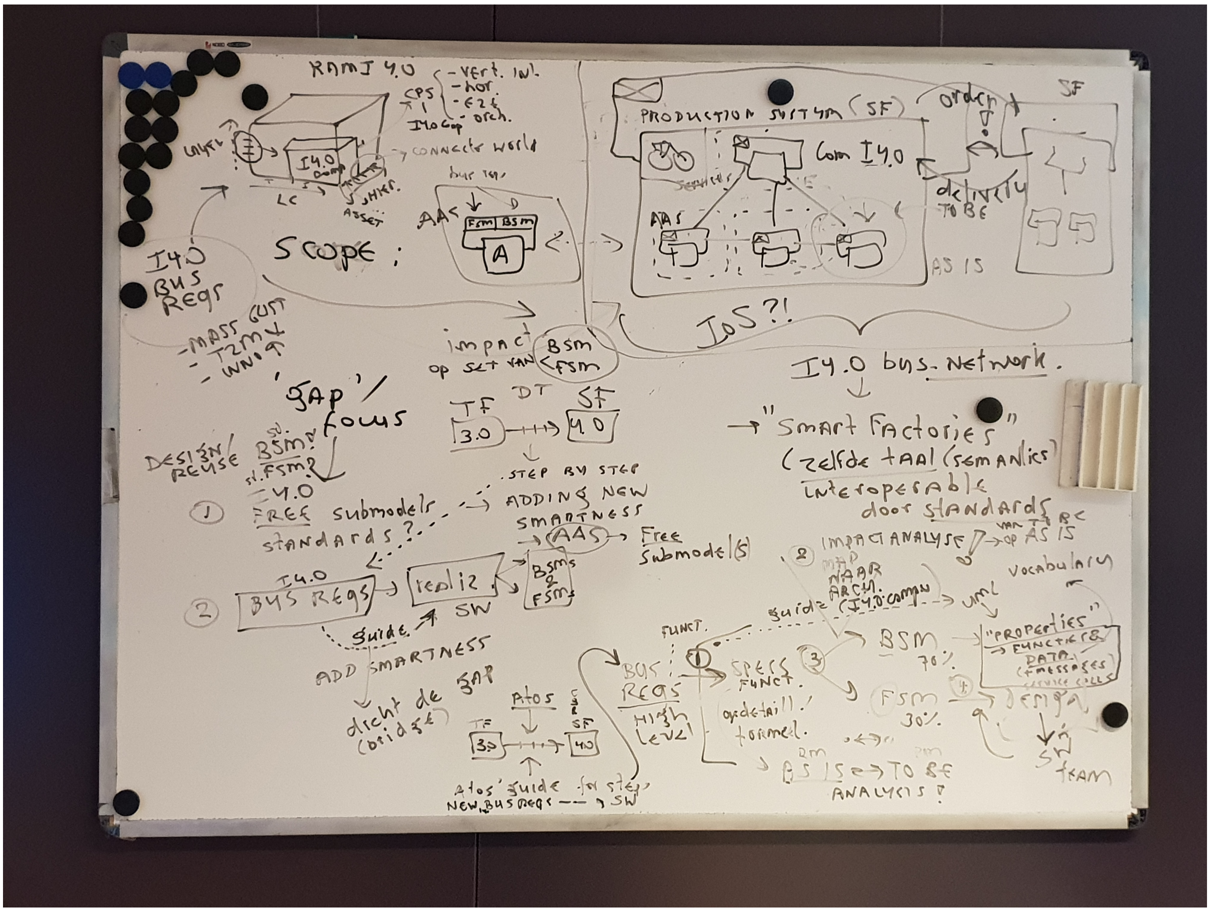

Figure C.4: Discussion 4

Redefining the general structure (version 4). Relation to RAMI4.0 investigated.

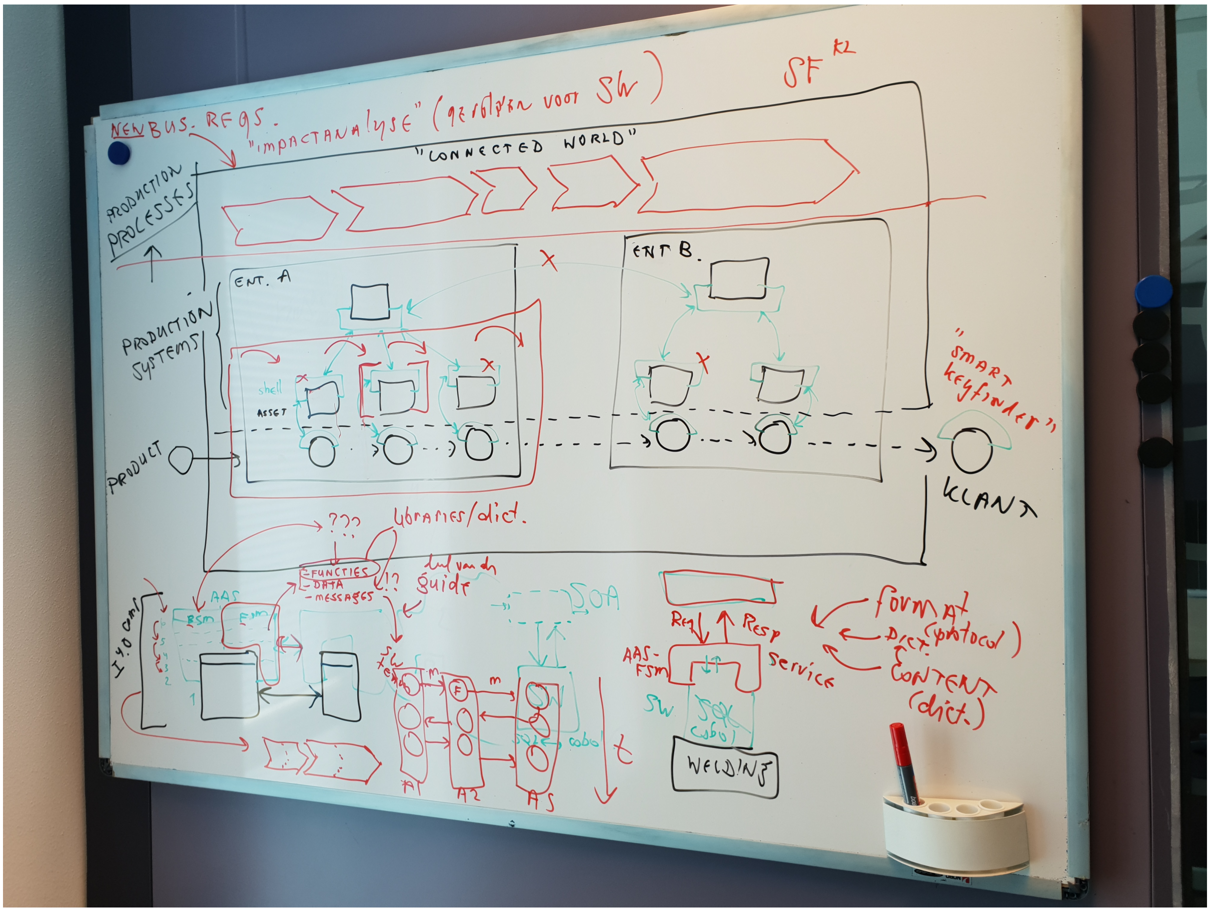

Figure C.5: Discussion 5

Discussion on the relation of the AASDM to production systems.

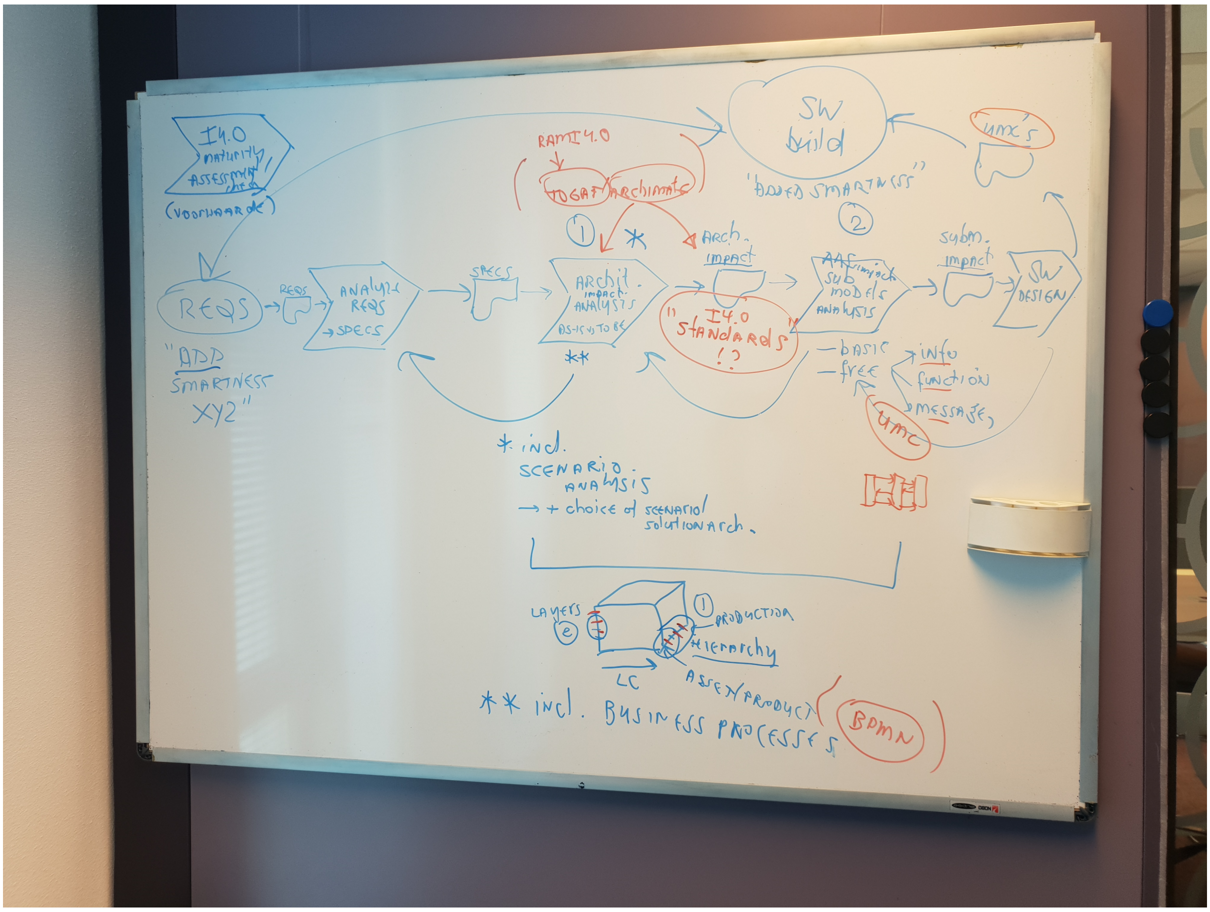

Figure C.6: Discussion 6

First final version of the structure of the AASDM (version 5).

Appendix D

Chapter 7

The semi-structured interviews for the validation of the artifact are structured as follows: at first the AASDM and related concepts are explained, then the artifact requirements are validated. In the end, the effect in its context is determined.

Information prior to the questioning

1. A general introduction to the goal and set-up of the interview.
2. Background information on:
    - The main difference between I3.0 and I4.0.
    - The concept of the AAS.
3. Information on the AASDM emphasizing on:
    - The phasing.
    - Roles of the employees.
    - Main activities of the AASDM.
    - An exemplar application.

Artifact Requirements

1. How would you describe the overall experience of the AASDM (usability)?
    - Score usability on a scale 1 to 5.
2. What is your perception of the utility of the AASDM?
    - Score utility on a scale 1 to 5.
3. Do you think the AASDM is understandable?
    - Score understandability on a scale 1 to 5.
4. Do you think the AASDM guides you through the complete process of designing AAS?
    - Score completeness on a scale 1 to 5.

Effect in its context

1. What do you think of the general structure of the AASDM?
    - What changes can be made to improve the AASDM?
    - What is missing in the current structure of the AASDM?
2. What is needed before the AASDM can be implemented in practice?
3. What do you expect when the AASDM is applied in practice?
4. Are the roles defined adequately?
5. What use do you see for the AASDM within your competences and the unit?
6. What do you think is relevant to its business value?