David Frühstück, BSc

Model-based Systems Engineering supported Powertrain Architecture Specification with focus on Functional Modeling

MASTER'S THESIS

to achieve the university degree of
Diplom-Ingenieur
Master's degree programme: Mechanical Engineering and Business Economics

submitted to

Graz University of Technology

Supervisor
Univ.-Prof. Dipl.-Ing. Dr.techn. Hannes Hick
Institute of Machine Components and Methods of Development

Graz, February 2020

"If I had an hour to solve a problem I'd spend 55 minutes to thinking about the problem and five minutes thinking about the solution"[1]
Albert Einstein (1879-1955)

---

[1] Einstein 2010, p.1.

Acknowledgements

I would like to thank all those who have accompanied and supported me throughout my studies, especially those who have contributed to the success of this master's thesis.

First, I'd like to thank Mr. *Univ.-Prof. Dipl.-Ing. Dr.-techn. Hannes Hick* for the opportunity to study and complete my master's thesis at the Institute for Machine Components and Development Methodology. Without the tireless guidance, perseverance and technical edits of Mr. *Dipl.-Ing. Matthias Bajzek BSc.*, this thesis would not exist. For this, I will be eternally grateful to him.

Beyond all, I thank my spouse *Tamara Frühstück* for her great patience and understanding, which she has shown me through all the long and many hours of learning. I also thank my parents *Ulrike* and *Herwig Frühstück*, who made my studies possible and have supported me in all matters. Furthermore, I would like to thank my extended family, friends, and companions. Without them, all this would not have been possible.

Eidesstattliche Erklärung
Affidavit

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master‘s thesis.

Graz, 18.02.2020

Ort, Datum

David Fuhstuck

Unterschrift

Abstract

In today's technological world, technical solutions are becoming continuously more and more complex as the requirements increase. In particular, the automotive industry pursues greater environmental compatibility and efficiency. These facts lead to an increase of system complexity and to a variety of different approaches to find a appropriate solution. The higher system complexity leads to higher demands for system development processes. Therefore, development processes are supported by methods and models.

Model-based systems engineering enables a model-based development of complex systems and interrelationships of systems. One method of model-based systems engineering is the functional modeling approach. This approach is used to understand, describe and model the functions of a system. *Eigner et al.* describe the interdisciplinary system development process as a sequence of requirements, functions, logical system architecture and physical system architecture (RFLP-approach). This system development process is analyzed in two steps in order to describe the added value of functional modeling. For this purpose, the method is applied to the example *powertrain*. First, selected powertrain architectures are analyzed in reverse order. Therefore, selected physical architectures are considered to identify the generic logical system architecture and the generic functions of the system. Second, the development process is applied in the described order (RFLP) to consider the system specification with the help of functional modeling in a *powertrain* case study.

The findings from both steps are discussed and summarized to strengths and weaknesses of functional modeling. Finally, the added value of functional modeling for the development of complex systems is described.

Kurzfassung

In der heutigen Welt werden technische Lösungen mit steigenden Anforderungen zunehmend komplexer. Insbesondere in der Automobilindustrie führt das Streben nach mehr Umweltverträglichkeit und nach höherer Effizienz zu zunehmender Systemkomplexität und Vielfalt an Lösungsansätzen. Eine höhere Systemkomplexität führt zu höheren Anforderungen an die Entwicklung von Systemen. Daher werden Entwicklungsprozesse durch Methoden und Modelle unterstützt.

Model-based Systems Engineering ermöglicht eine modellhafte Beschreibung von Systemen und komplexen Zusammenhängen. Eine Methode des *Model-based Systems Engineering* ist die funktionale Modellierung. Diese wird als Zwischenschritt in der Entwicklung eingesetzt, um die Funktionen eines Systems verstehen, beschreiben und modellieren zu können. *Eigner et al.* beschreiben in seinem Vorgehensmodell die interdisziplinäre Systementwicklung als eine Abfolge von Anforderungen, Funktionen, logischer und physikalischer Systemarchitektur. Um herauszufinden, welchen Mehrwert die funktionale Modellierung zur Systementwicklung beiträgt, wird der Entwicklungsprozess in zwei Schritten analysiert. Für diesen Zweck wird die Methode an dem Beispiel *Antriebsstrang* angewendet. Als ersten Schritt werden ausgewählte Antriebsstrang Architekturen in umgekehrter Reihenfolge analysiert. Das bedeutet, dass ausgewählte physikalische Architekturen zunächst hinsichtlich ihrer generischen logischen Systemarchitektur und anschließend hinsichtlich ihrer generischen Funktionen untersucht werden. In einem zweiten Schritt wird der Entwicklungsprozess in der zuvor beschrieben Reihenfolge nach *Eigner et al.* angewendet, um die Systemspezifikationen mit Hilfe der funktionalen Modellierung am Beispiel einer *Antriebsstrang* Fallstudie zu betrachten.

Die Erkenntnisse aus beiden Analyseschritten werden diskutiert und zu Vor- bzw. Nachteilen der funktionalen Modellierung zusammengefasst. Abschließend wird der Mehrwert der funktionalen Modellierung für die Entwicklung komplexer Systeme beschrieben.

List of Abbreviations

BDD	Block definition diagram
CO_2	Carbon dioxide
ECU	Engine control unit
H_2	Hydrogen
IBD	Internal block diagram
INCOSE	International council on systems engineering
ISO	International organization for standardization
MBSE	Model-based systems engineering
MVPE	Modellbasierte virtuelle Produktenwicklung
NASA	National aeronautics and space administration
NVH	Noise vibration harshness
NO_x	Nitrogen oxide
OEM	Original equipment manufacture
OEM	Object management grouping
PCU	Powertrain control unit
PM	Particle matter
RFLP	Requirements, function, logical architecture and physical architecture
SE	Systems engineering
SoS	Systems-of-systems
SysML	Systems modeling language
UML	Unified modeling language
V/V	Verification/Validation
VDI	Verein Deutscher Ingenieure
VW	Volkswagen group

Contents

1 Introduction

Many aspects of the world of today are highly technological.[1] The technological world consists of specialized products that are designed to meet the needs of the common person. Examples of the technological world include driving with cars over a vast road network, communicating over the worldwide internet, or improving the daily living conditions. Research enables the achievement of these technological advances and develop sophisticated products by using the laws of nature.

The expanding trends of a technological world lead to an increasingly higher demand for products. Especially the increasing number of functionalities of a product with a massive number of interactions and influences between products which are not predictable make a product or a system complex. *Lindemann* describes that complex products lead to complex product development processes. The aim is to handle both types, complexity of products and complexity of processes.[2]

The following described facts show that products and their development processes are becoming more complex. Products are not solely mechanical systems anymore; instead, they are mechatronic systems. Therefore, the development of these systems is very interdisciplinary, involving the disciplines mechanics, electrics and electronics as well as software. Another fact is that every new product has to be designed for the whole product life-cycle, from the idea to the disposal, to be able to satisfy every stakeholder. Lastly, the market is especially demanding for innovative products, a wide variety, shorter development times, better quality and lower prices.

In total, the increasing complexity of products exceeds the performance of the development in industrial practice causing an integration gap, as shown in figure 1.1. In order to handle this integration gap between the complexity of the products and the performance of the development in industrial practice, companies and universities are working on models and methods to narrow the gap.[3]

This thesis deals mainly with one methodological approach out of the system development pool. The method is called functional modeling, which is an approach out of the subject area of model-based systems engineering. This method should support the development process of complex systems. In order to be able to deal with the methodology in detail, the method is applied to a case study. The primary case study of this thesis is the automotive powertrain. The automotive powertrain is the main assembly

---

[1] Ropohl 2009, p.15.

[2] Lindemann 2009, p.8.

[3] Bajzek 2018, Slide 9.

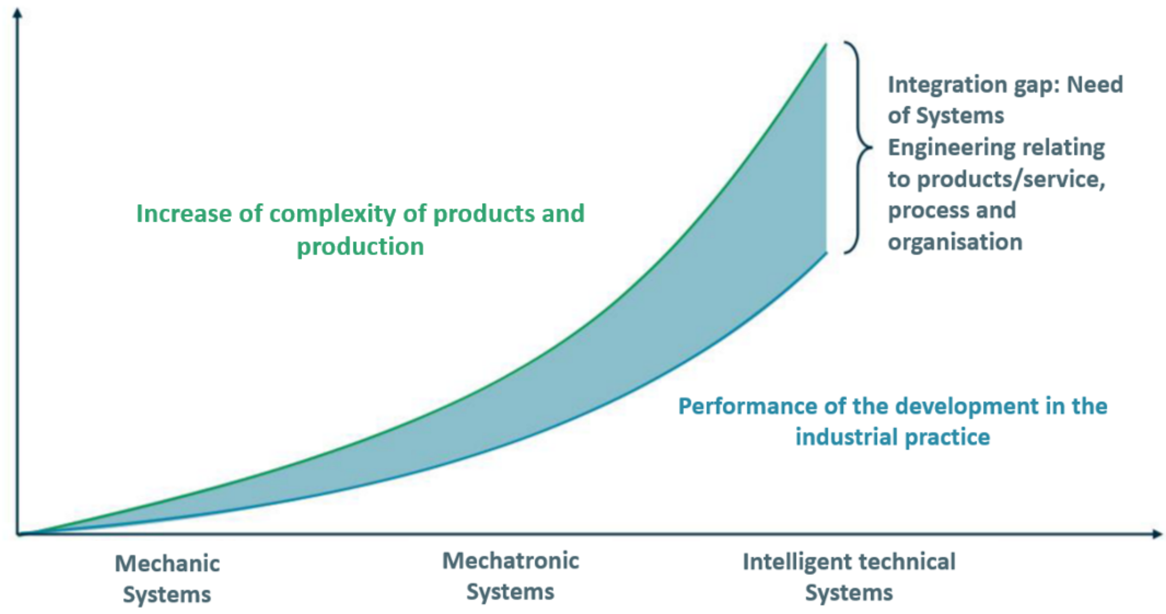


Figure 1.1: Challenges in development[4]

of a vehicle that generates power and delivers it to the wheels. The following subsection explains the initial situation of the vehicle according to the automotive powertrain. Furthermore, this chapter explains the target, the motivation, and the structure of the thesis.

1.1 Initial situation

As described before, the products of today are more complex than ever before. One example of a complex product is the vehicle. The requirements are different for all stakeholder groups, as shown in the following discussed trends.

The influential trends in the automotive industry are urbanization, demographic change, autonomous driving, and reduction of emissions. *Urbanization* leads to growing cities, which cause increasing traffic density and consequently changed demands on infrastructure, vehicle technologies, connected mobility, electrification, digitalization, etc. *Demographic change* means that the drivers and passengers are getting older, which leads to new demands on ergonomics, safety, and comfort functions. Original Equipment Manufacturers (OEMs) pursue the *autonomous driving* concept as the next big step in the history of vehicles, which leads to requirements for continuously increasing automated driving functionalities. The continuously strengthened exhaust *emission*

---

[4] Lindemann 2016, translated from German.

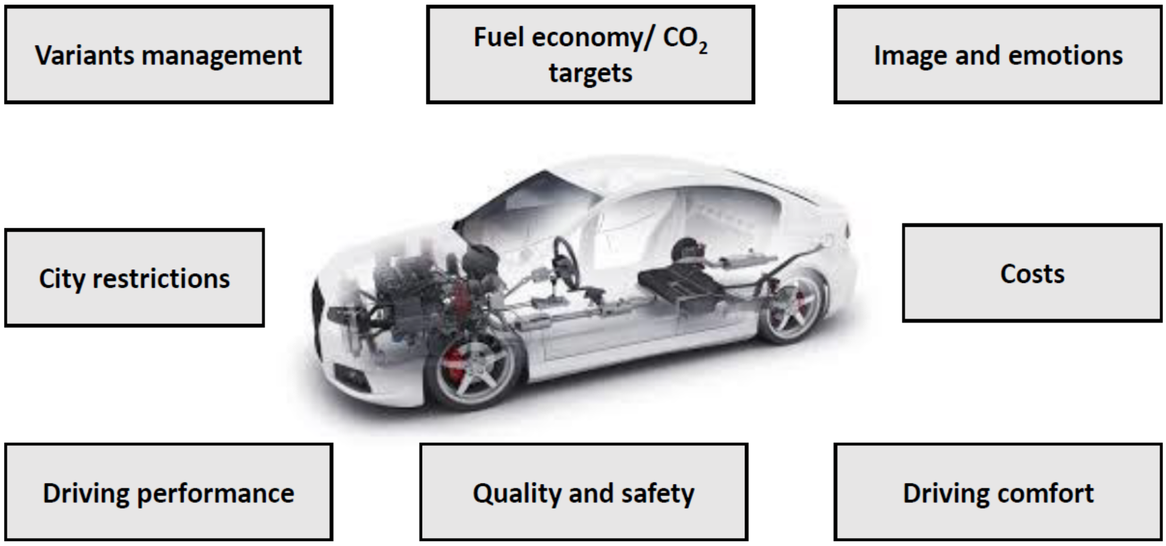


Figure 1.2: Market drivers for future powertrains[6]

legislation implies an increasing demand for highly efficient technologies, alternative energy supply for mobility, and the consideration of the entire life-cycle of technologies - including production, in-use phases, and disposal.[5]

Additionally to the trends in the automotive industry, figure 1.2 shows the main market drivers for the future powertrain. The changing trend and market drivers lead to changing requirements which a future powertrain has to fullfill. The state of the art of powertrain systems has a high diversity, which is defined by the variation of powertrain components such as combustion engines, electric motors, fuel cells, batteries, and many others. The technical possibilities to fulfill the requirements and meet the market drivers of future powertrains are endless because of this diversity of possibilities.[7]

The specification of the powertrain architecture based on requirements that are either given or to be determined is becoming increasingly difficult due to many different influences (on the powertrain) and dependencies (between the components of the powertrain). In order to find the ideal architecture, the early consideration of the possible alternatives as well as a comprehensive understanding of the system are required. The definition of the powertrain architecture becomes also complicated by the fact that optimal components do not necessarily lead to the optimization of the overall system.

Functional modeling methods are used to consider the powertrain in a solution independent way to find the optimal architecture. This method allows a solution-neutral view of the problem and focuses on the actual tasks of development. After the func-

---

[5] P. D. M. Hirz 2018, Slide 7 ff.

[6] Siemens 2020.

[7] Ellinger and Schöffmann 2019, p. 9.

tional model is applied, components are selected to contribute the functional fulfillment. Furthermore, functional models are advantageous due to versatility and applicability over a more extended amount of time. One of the main challenges is to define the functions that simplify the powertrain specification process and balance the additional cost of creating a function model with regards to the value of the solution. It is indisputable that there is a need in the industry to find the most cost-effective process for product development. Functional modeling can be the answer, but companies and researchers still have many questions:

- What added value does functional modeling create?
- How can the added value be measured?
- How can the functional model support the decision-making process?
- Is the step functional modeling justified concerning the work to develop those models?

1.2 Target and motivation of the thesis

The motivation of the thesis is to examine the applied functional modeling methods on the case study on *automotive powertrains*.

The focus of the thesis is the analysis of the functional modeling approach, the impact of the method on the case study, and the upcoming challenges with functional modeling application, including inconsistencies.

The following research question encompasses if functional modeling is valuable and effective for overall product development.

"What added value does functional modeling provide for finding the optimal architecture, in consideration of possible alternatives?“

1.3 Structure of the thesis

This thesis contains five main chapters (additionally to this introduction chapter): theoretical background, analysis of the powertrain system architectures, powertrain system specification, discussion as well as conclusion and outlook.

The *second chapter* introduces the theoretical background of product development. Furthermore, it describes the topics systems engineering (SE) and model-based systems engineering (MBSE). Based on the theoretical background of MBSE, it describes the theory of functional modeling in detail.

The *third chapter* shows the different powertrain architectures common in today's vehicles. A development process of a powertrain starts with the definition of requirements and develop the powertrain piece by piece until the final powertrain is created and ready for production. The development process from the requirements to the design of a single physical part follows the top-down approach. The counterpart, the bottom-up approach, is used to analyze the development process in sequences in the reverse order. A range of powertrain architectures is analyzed by following the bottom-up approach. Based on the analysis of three specific powertrains, a generic logical model is created. This generic logical model is used to create a generic functional model. Both model types are generated by using the software language *SysML* and the program *Eclipse Papyrus*.

The *fourth chapter* analyzes the development process by following the top-down approach. To show how a system can be specified by using functional modeling, the analysis starts with a simple case study on the development of a light bulb. The case study starts by defining the requirements. Afterwards, the development process is shown by following the rules of functional modeling. The case study's aim is to show the supporting effects of the method of functional modeling on a *light bulb*. After the simplified case study, the same procedure should be shown on the case study on the *automotive powertrain*. The bottom-up approach from *chapter three* and the top-down approach described in this chapter are the basis to answer the research question.

The *fifth chapter* discusses advantages and disadvantages as well as strengths and weaknesses of the functional modeling approach. Furthermore, the added value of functional modeling for the development of complex systems is described.

The *sixth chapter* describes the conclusion of the thesis and the outlook of the functional modeling approach.

2 Theoretical Background

This chapter introduces the fundamentals of product development and discusses why a model-based development approach is essential for today's and future product development. Furthermore, the basics of SE are explained, together with the fundamentals of the systems theory and the main procedure models. Moreover, this chapter introduces the topic of model-based systems engineering as an essential part of product development. Finally, the fundamentals behind the method of functional modeling are emphasized in detail.

2.1 Product development

"Every once in a while a revolutionary product comes along that changes everything."
Steve Jobs[1]

The title of the topic *product development* consists of two words with there own specific definitions. The term *product* is defined as the result of the production, which the company touts at the market. From the customers' perspective, a product is something that meets their needs. Generally, two categories of products are distinguished: material goods and services.[2] The term *development* can be described in the context of research. Research means the acquisition of new knowledge, and development means to apply that knowledge into practical implementation.[3] The definition of product development is a derivation out of these two definitions.

Eigner et al. define product development as an integrated, multidisciplinary approach that includes all activities and disciplines to describe the product, its production, the operation and disposal, the necessary environment over the product-life-cycle, all involved activities and disciplines, supply chains to describe and to apply this knowledge into practice. One result of the product development process is a corresponding product description that consists of all documents and configurations.[4]

Lührig defines product development as a transformation of an idea into a combination of goods or services for commercial purposes carried out by one or more persons or

---

[1] Isaacson 2011, p.473.

[2] Gabler 2013, p.351.

[3] Gabler 2013, p.157.

[4] Eigner 2014, p.7.

organizations. It is assumed that the idea is technically feasible, and that there are sufficient marketing opportunities for the product.[5]

The definitions of *Eigner et al.* and *Lührig* have in common that the goal of product development is to bring new products on the market. The success of product development has a significant effect on companies' success.[6]

The product itself influences the product development process. Products differ in requirements for the product development process, and are therefore changing the process. For example, new products have to be designed with the whole life-cycle in mind, and therefore also the development process has to include a holistic view. Furthermore, the aim of many companies to have a shorter time to market, in order to meet customers' demands, leads to a simultaneous running of the stages from the development process instead of a sequential process. The changing product structure has a significant effect on the development process. The interacting mix of mechanics, electronics and electrics as well as software grows continuously. That requires interdisciplinary teamwork from different fields of expertise in the development process. A further issue is that companies develop products more and more in value networks.[7]

In order to handle those changes, the product development process gets support from new approaches. One common approach is to support the development process with models, which is called the model-based development approach.

Figure 2.1 shows the principle difference between a document-based and a model-based approach. The humans or employees communicate in document-based systems or companies by transferring their information based on specific documents. Those documents contain only specific pieces of information, and there is no connection to other information. For example, a specific design document contains information about the structure of a product. It does not contain any information about the material of each part and has neither any connection to other information. In the model-based approach, the different models are connected to others, which results in a network of information and inter-connected models. Specific documents are part of the information network and can be derived from the model. This approach represents as a whole a computer-aided virtual product that captures the whole existing knowledge of the product.

The three main properties which characterize a model are the figure, the reduction, and the pragmatic property. The *figure property* means that a model is always an illustration of the real part. The *reduction property* describes the fact that a model does not contain the full information of the original product but contains only the information that is relevant for the user. The *pragmatic property* names the value of the model including the effect on their insertion function for particular subjects, within specific time intervals and certain mental or actual operations.[8]

---

[5] Lührig 2006.

[6] Schömann and Ringlstetter 2011, p.61.

[7] Lindemann 2016, p.3.

[8] Stachowiak 1973, p.131.

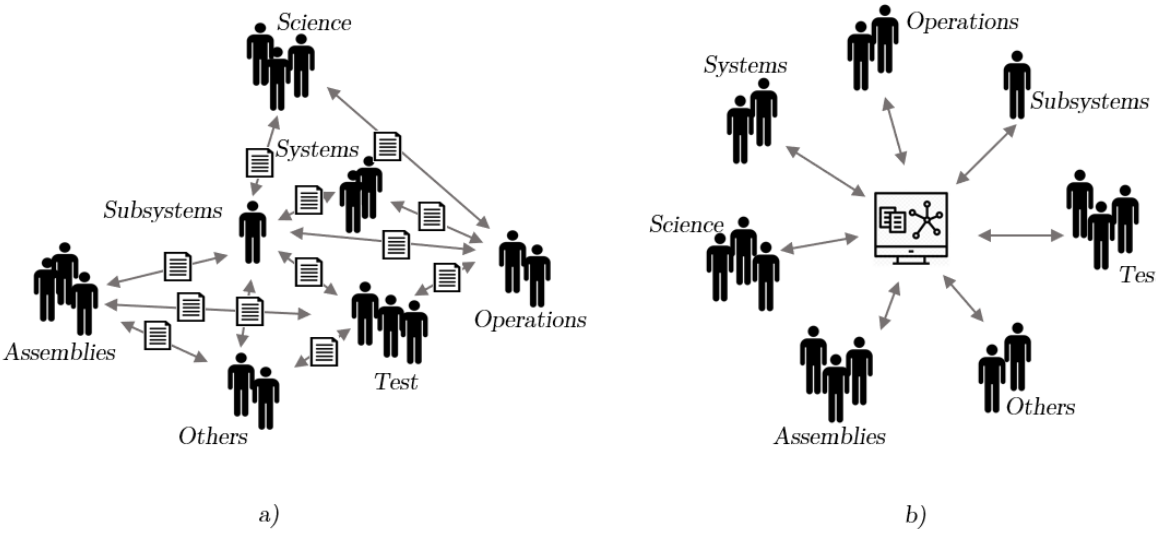


Figure 2.1: a) Document-based view
b) Model-based view[10]

In combination with product development, *Eigner et al.* define model-based product development as the end-to-end, computer-assisted, formal and semi-formal modeling and documentation of all development-relevant phases of the product life-cycle to transfer the model to the next development phase and the further use of these models for simulation, verification, and validation. The goal of model-based development is the development of product and production knowledge in the initial phases of the life-cycle in order to optimize the product properties as early as possible. The result is a holistic optimization of the entire product life-cycle and the drastic reduction of physical prototypes.[9]

Models are used for many purposes. It is not only a topic for the development of new products, it is also used for existing products. Many existing products are poorly documented. Therefore, modeling the existing products provides a concise way to capture the existing product architecture and design. Another essential purpose of models is the concept of formulation and evaluation of products. That means that models are early applied in the product definition process to synthesize and evaluate alternative concepts. This includes a clear and unambiguous definition of the value that the product is expected to deliver to the beneficiaries. Another purpose is to support the product architecture design and requirements flow-down. Models can be used to support architecting product solutions as well as to flow down product requirements to product components. Finally, there are many other purposes of models. For example, models also support the integration and verification of products and are a beneficial

---

[9] Eigner 2014, p.9.

[10] Louie 2014, p.1.

tool for capturing knowledge. The described purposes emphasize the importance of modeling in today's development process.[11]

2.2 Systems engineering

Systems engineering is a concept to transfer problems into solutions. *Haberfellner et al.* describe a problem as the difference between the current state of something and the target state of something. *Haberfellner et al.* add that a problem can be an unclear current state as well as an unclear target state.[12] Independent from the problem itself, systems engineering is a specific way of thinking to turn the problem into a solution.

Figure 2.2 shows the SE concept by *Haberfellner et al.* The SE components are arranged in the illustrated to they form of a human. At the beginning, the SE philosophy is the human's head. It consists of systems theory, which includes the fundamentals of SE, and the procedure model, which explains generic ways to come from the problem to the solution. The problem-solving process is illustrated by the human's body and connects the right hand (problem) with the left hand (solution). The process of systems design is typically embedded in a project, whereby project management leads the problem-solving process over a specific time in an organization. The project management part of the systems engineering approach is not further discussed in this thesis. Furthermore, the problem-solving process is supported by some best practice techniques.

The user of the systems engineering approach can use this structure in combination with the knowledge of the system theory and the procedure models to solve problems supported by best practise techniques. In this chapter, the fundamentals of SE, procedure models, and best practice techniques are described.[13]

Besides the introduced view of *Haberfellner et al.* there are many other possible definitions for define systems engineering. The word *system* is commonly described as a set of interrelated components working together toward some common objective. The word *engineering* can be equated with the terms of *generating something* or of *developing something.*[14] The *National Aeronautics and Space Administration (NASA)* characterizes systems engineering as a methodical, multidisciplinary approach for design, realization, technical management operations, and retirement for systems.[15] A common description is defined by the *International Council on Systems Engineering (INCOSE)* as follows:

---

[11] Walden et al. 2015, p.181 f.

[12] Haberfellner et al. 2015, p.24.

[13] Haberfellner et al. 2015, p.25f.

[14] Gabler 2013.

[15] NASA 2016, p.3.

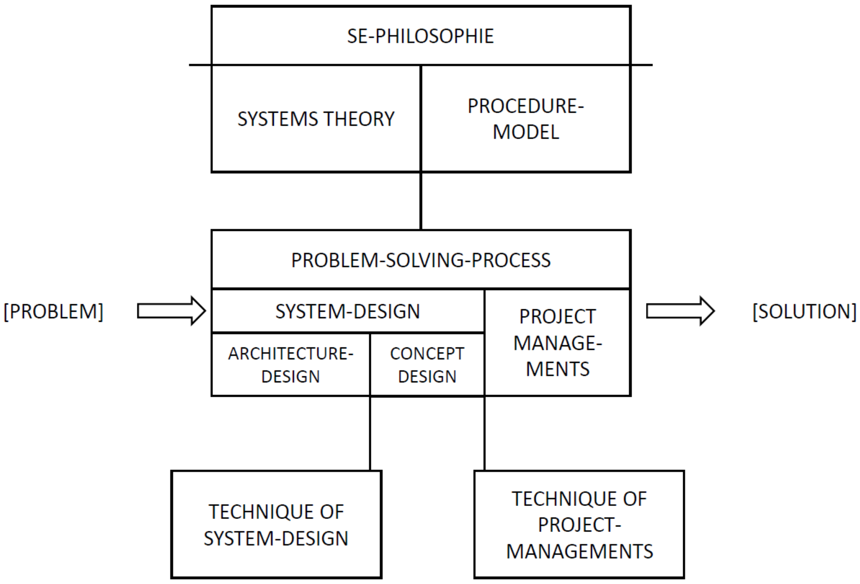


Figure 2.2: The SE-concept[16]

> *"Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem. [...] Systems Engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation. Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs."*[17]

To sum up, systems engineering seeks a holistic and balanced way of thinking to solve problems in the face of opposing interests and multiple, conflicting constraints. It is important to emphasize that systems engineering (SE) is an approach to successfully develop complex systems or products. It is a way of looking at the big big picture and a logical way of thinking.[18] Overall, *Kossiakoff* summarizes systems engineering as a powerful discipline, requiring a multidisciplinary knowledge, integrating diverse system elements.[19]

---

[16] Haberfellner et al. 2015, p.26 translated from German.

[17] Walden et al. 2015, p.9 translated from German.

[18] NASA 2016, p.3.

[19] Kossiakoff et al. 2011, p.3.

2.2.1 Systems theory

The origin of the systems approach can be attributed to the Greek philosopher *Aristotle*. He already stated that the whole is greater than just the sum of its parts.[20] This statement becomes clear when considering the example of a powertrain. The powertrain consists of many parts, but only as a whole, added value can be created. The behavior of the individual interacting parts differs from the behavior of the whole. The engine as a single part transforms chemical energy to mechanical energy. The transmission as a single part transports the mechanical energy from the engine to the wheels. Only the powertrain as a whole can create the obtained added value, namely changing the kinetic energy of the vehicle by using the chemical energy.

Ropohl described a system as a *model of a whole entity that shows the relationships between attributes (inputs, outputs, states, etc.), that consists of linked parts or subsystems and that is distinct from its environment or a superordinate system.*[21]

The *International Organization for Standardization (ISO)* defines a system in the context of a life-cycle and technical viewpoint as follows:

> *"[...] a system is sometimes considered as a product or as the services it provides. [...] in practice, the interpretation of its meaning is frequently clarified by the use of an associative noun, e.g., aircraft system. Alternatively, the word "system" is substituted simply by a context-dependent synonym, e.g., aircraft, though this potentially obscures a system principles perspective. [...] a complete system includes all of the associated equipment, facilities, material, computer programs, firmware, technical documentation, services and personnel required for operations and support to the degree necessary for self-sufficient use in its intended environment."*[22]

In principle, the system fulfills a determined purpose for which the elements of the system have to interact. The description of distinguishable systems provides an overview. Therefore, primarily socio-technical systems are described as follows:

- **Social system** means in general a group of humans including interrelations, for example at sports, in companies, etc.[23]
- **Socio-technical system** means the interrelation between humans and maschines. Maschines are built from and for humans, and it is important to incorporate both in development.[24]

---

[20] Aristoteles 1960.
[21] Ropohl 2009, p.77.
[22] ISO15288 2018.
[23] Berghaus 2011, p.61.
[24] Walden et al. 2015.

- **Technical system** *"[...] are artificially produced geometric-material structures that fulfil a particular purpose, hence effect operations (physical, chemical, biological processes)."*[25]

Fundamental system terminologies

Systems engineering uses approved terms to describe systems. Figure 2.3 illustrates a system that contains the commonly used terms. The term *system*, as mentioned before, is a byword for the connection and interaction of a specific number of elements. Components are physical parts which are represented as *elements* in a system. A system contains elements and is determined by *systems boundaries*. The connections between the elements are called *relationships* and describe transportation lines for matter, energy, and information. The illustrated topology is the *structure* of a system. A system can include a system which in turn contains elements and relationships. That system in a system is called a *sub-systems* of the system. The environment of a system shows the connection between the system and other elements or systems outside of the boundaries. All systems and elements outside of the system which are relevant to the system are called the *system context*.[26]

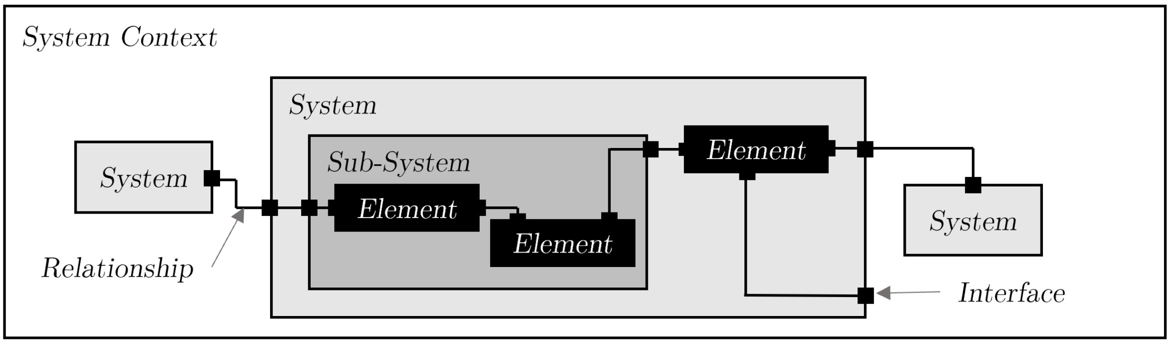


Figure 2.3: Fundamental system terminologies inspired by *Haberfellner et al.*

Concepts of systems-theory

Ropohl uses different system aspects to define three system concepts, namely the functional, the structural, and the hierarchical concept, as illustrated in figure 2.4.

The standard concept is the *structural concept*. This concept represents the holistic view of *Aristotle*. The focus of this concept is on the structure of a system that consists of elements and relationships. The *functional concept* illustrates the system as a *black box*. That means that the system is only characterized by connections with

---

[25] Ehrlenspiel and Meerkamm 2017, p.27.

[26] Haberfellner et al. 2015, p.32ff.

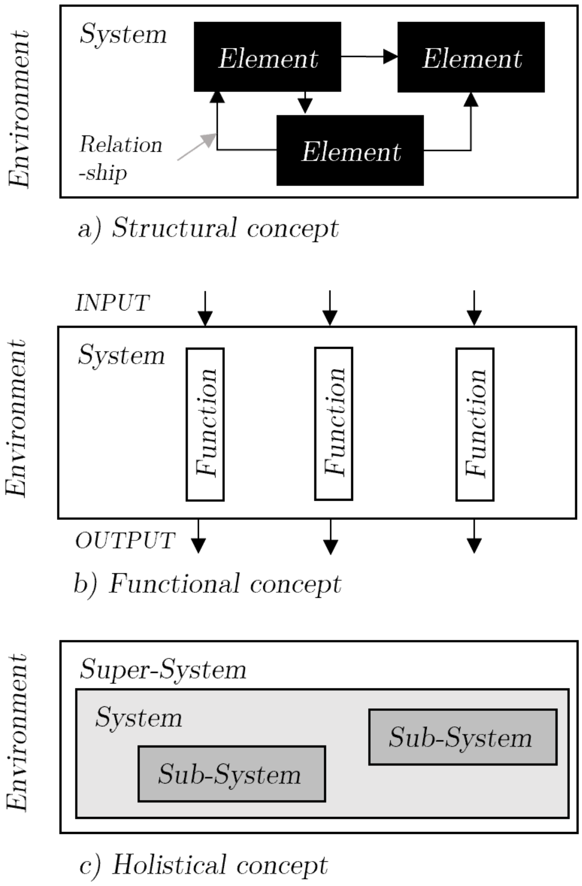


Figure 2.4: Basic concept of systems[28]

the environment and the function of the system. The connections can be divided into input and output, and the function is the *formula* which links the input to the output. The last concept is the *hierarchical concept*. This concept emphasizes the fact, that a system can consist of a sub-systems or can also be part of an upper system. *Ropohl* supplements that a system is only fully described when each of the three concepts is described.[27]

System hierarchy vs. systems of systems

Depending on how systems work together, two different definitions are derived. As described before, a system can be part of a system which is then called a sub-system. The system itself can also be a part of an upper system. When this is the case, the upper system is then called a super system. For example, the system powertrain is part

---

[27] Ropohl 2009, p.75ff.

[28] Ropohl 2009, p.75 translated from German.

of the super system vehicle on the other hand the system powertrain is a sub-system of the system vehicle. Besides of this definition, the term *system of systems* has to be differentiated. System of systems (SoS) have two characteristics, as follows:[29,30]

- Each system of the SoS is able to **act independently**.
- Each system of the SoS is **independently developed**.

If both apply, it is called a system of systems and no super system. For example, the integration of the cell phone in the vehicle is an SoS because each system acts independently, and each system is independently developed. An SoS should have per definition a higher value than the systems alone. For example, a cell phone and the vehicle together have a higher value than the systems alone because they have more functionalities together.

Black-box, grey-box and white-box

As described above, the black-box is an illustration of the system. This view focuses on the interfaces like input or output and the functionalities of the system. The topology of the system is hidden to keep the focus on this information and protect against focusing on detail in early development phases. Contrary to the black-box view, the description of the topology of the system is significant for the white-box view. Therefore, the white-box illustrates the elements and relationships of the system in detail. That view is relevant for detailed considerations, like system simulations. The grey-box view is a rough view of the structure of the system. In that view, only the main generic elements of the system are illustrated. It is a useful instrument to focus on an area of a complex system.[31]

Definition of complex systems

The definition of systems engineering contains the word *complex* and states that systems engineering is an approach to handle complex systems. First of all, complex and complicated systems have to be distinguished. Complicated systems are hard understandable systems, which relates only to the personal understanding limits. A complex system, on the order hand, is a system that is unpredictable, dynamic and cannot be described unambiguously, when every element and their interactions are known. Therefore, the definition of complex is the following: *“A system is complex if it cannot be described unambiguously, even if we have complete knowledge of its elements and their interaction.”*[32]

---

[29] Haberfellner et al. 2015, p.35.

[30] Ropohl 2009, p.75ff.

[31] Haberfellner et al. 2015, p.36.

[32] Härtl 2008.

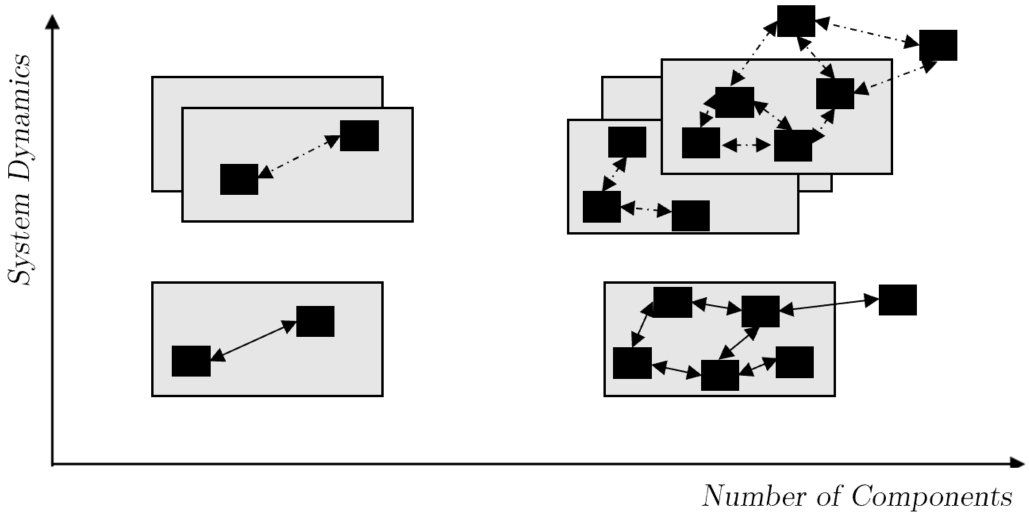


Figure 2.5: Principal classification of systems and meaning of complexity[33]

Figure 2.5 illustrates the classification of systems regarding dynamics and structure, which shows the difference between a complicated and complex system even more apparent. The dynamics of a system, which means the change over time, can be divided into predictable dynamic (or static) or unpredictable dynamic behavior. The structure of a system can be visible because of low numbers of elements and relations. Still, the structure can also be very invisible with a vast amount of elements and relations.

Different types of systems can be classified with this diagram. A *simple system* consists of few elements which are statically connected, and there are only few numbers of relations between the elements of the system. A *complicated system* is statically too but boasts many elements and relations, which leads to the fact that those systems are hard to understand. A dynamic system is a system where the connections between the elements are changing over time. A dynamic system is not called complex when the system only has a few number of elements and of relationships. A complex system has a vast amount of elements and relations, which are frequently changing over time.[34]

The approach of systems engineering is to support the understanding and development of complex systems and help to deal with complexity in an optimum way.

---

[33] Ulrich and Probst 1988, p.89.
[34] Ulrich and Probst 1988, p.89.

2.2.2 Procedure models

Besides the systems theory described in the previous section, procedure models are another essential part of the SE philosophy. The procedure models recommend the best way from a problem to a solution that has proven in practice.[35] The most established procedure models of mechanical engineering are based on the *product development process* (PDP) which consists of five main phases:[36]

- Requirements specification and planning
- Concept and design
- Detailing
- Realization and integration
- Verification and validation

Over time, many different procedure models have been developed. The list spreads over standard models, for example the *VDI2221*, up to newer models, such as the *agile spiral models*. In this section, only one model which is relevant for this thesis is described. It is a variation of the well known *V-model*. Before the V-model is described, four fundamental ideas of SE need to be reviewed. According to *Haberfellner et al.* these four fundamental ideas are the essential factors for an appropriate procedure and include:[37]

- *Top-down approach*: The top-down approach means to avoid dealing with detailed problems in the development process instead of working on the whole task. In order to recognize every possible solution, it is essential to start with a broad view of the problem and go step by step into a more detailed stage. The explained black-box/white-box approach is the basis of this principle. The approach starts with a black-box, where only the input and the output are known. Step by step, the system can be developed into a white-box where the relations are visible. Besides the top-down approach, the bottom-up approach is commonly used to analyze systems in reverse order.
- *Thinking in variants*: Another very relevant basic idea is the creation of variations. For each problem, variations of solutions are possible. On the way to the best solution during each stage in the top-down approach, it is essential to look which variations of solutions are possible, and to decide which one is the best to go further.

---

[35] Haberfellner et al. 2015, p.55.
[36] Eigner 2014, p.9 translated from German.
[37] Haberfellner et al. 2015, p.57.

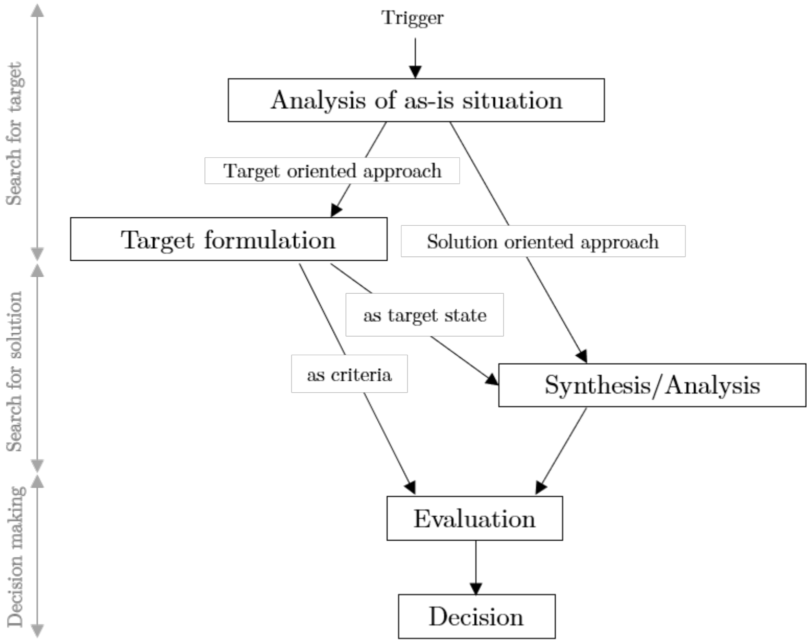


Figure 2.6: Problem-solving cycle to handle arising problems[38]

- *Structuring the procedure in phases*: The process from a problem to a solution should be split into stages. This supports the top-down approach and the thinking of variants in each stage.
- *Problem-solving cycle*: Figure 2.6 shows an excellent approach to handle arising problems in a development approach and to make the right decisions.

V-model

The described model is specially modified for the needs of the model-based product development approach and is called the *MVPE-procedure model.* Figure 2.7 illustrates the model. The left side of the V represents the system definition and the right side represents the verification and validation of the realized system. This left side of the V-model represents the top-down approach. The top-down approach is complemented by the RFLP (Requirements-Functions-Logic-Physical) approach. The process starts with the specification of requirements based on the customer needs. The requirements are a list of conditions of properties and functions that the product should fulfill. After that, a functional model is created based on the defined system requirements. The functional model is further developed into a logical model. Simulations support the process. Physical elements are selected to execute the principal previously defined

---

[38] Haberfellner et al. 2015, p.261 translated from German.

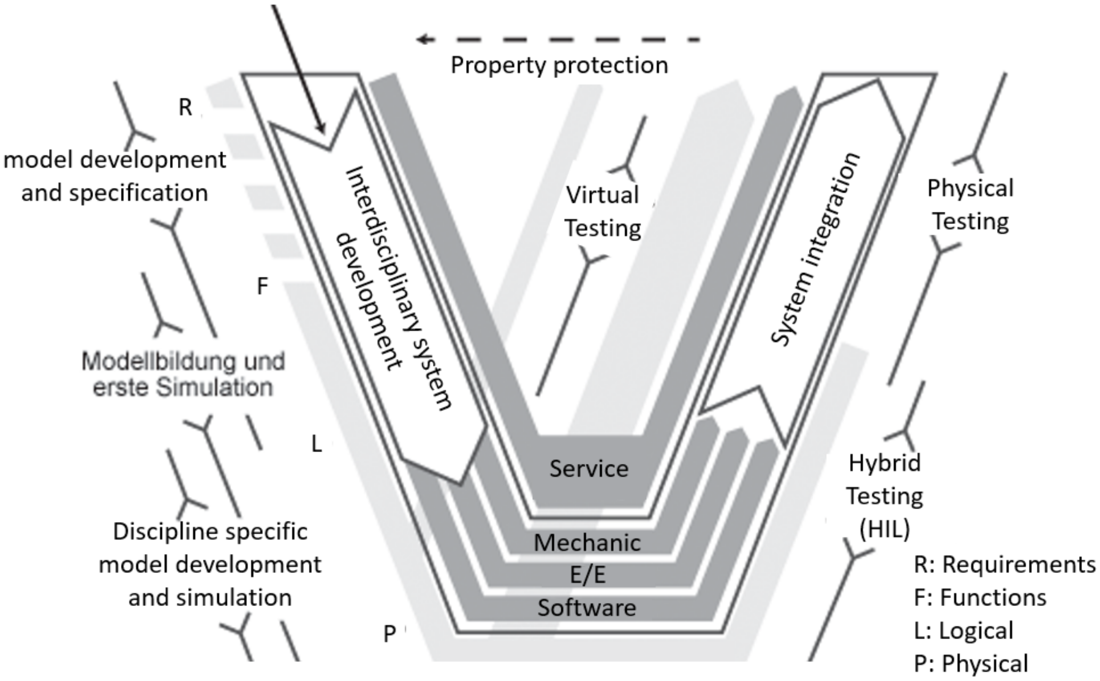


Figure 2.7: MVPE-Procedure model[41]

logical model. The *V*-model emphasizes the importance of horizontal information transfer, from the ever-more advanced models on the left side to the system verification and validation tasks on the right side, in order to enhance the idea of early assurance of requirements fulfillment. The *V*-model includes the life-cycle relevant information, which should be integrated from the beginning of the development.[39,40]

2.2.3 Model-based systems engineering

Model-based systems engineering (MBSE) is the combination of the basic idea of developing a product based on models as described in section 2.1 and the principles of SE. Therefore, the model represents the system. *INCOSE Systems Engineering* define model-based systems engineering as the:

---

[39] Lindemann 2016, p.428.

[40] Eigner 2014, p.86 ff.

[41] Eigner 2014, p.89 translated form German.

> *"formalized application of modeling to support systems requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life-cycles phases"*[42]

In contrast to the documented-based, the model-based approach leads to the possibility to capture, analyze, communicate, and to manage the information which is used for product specification. The advantages of MBSE are better ...[43]

- ... stakeholder communication in the development process.
- ... control of the system complexity.
- ... quality of the product based on the completeness and consistency of the model.
- ... capture of knowledge and informations.
- ... understanding of the SE principles on the basis of the clearness of the model.

System model

Model-based systems engineering focuses on the generation and use of the system model as a primary source of information. Therefore, the system model supports the development process in order to center the information and provide the process as a basis for communication and interaction of the several disciplines. The model-based approach does not prohibit the documents. Instead, the model-based approach supplements the documents as a primary source of advanced knowledge. The model represents the object or product as an abstraction of reality and is a network of connected information.[44] The following properties characterize a system model:[45]

- A system model can consist of many repositories, but it has to concise and represent one single model
- A system model incorporates different points of view (e.g. interdisciplinary views or structural view and functional view)
- A system model supports the realization of a digital twin, as it supports the storage, and processing of system-related information

---

[42] INCOSE 2007, p.291 translated from German.
[43] INCOSE 2007, p.291.
[44] Rambo and Weber 2017, p.173.
[45] Weilkiens 2014, p.22.

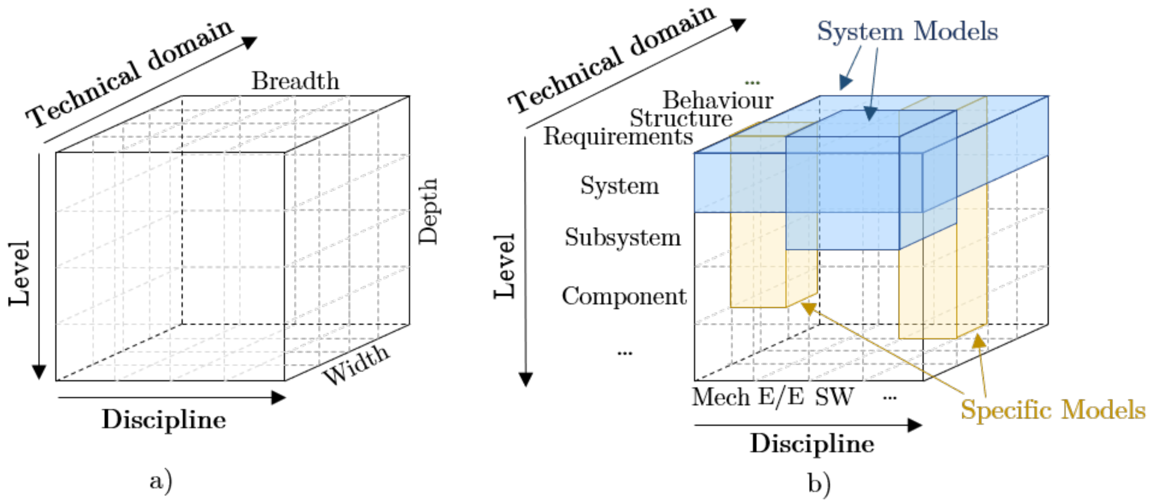


Figure 2.8: a) General illustration of classification scheme in form of multidimensional system cube b) Exemplary models (system models and specific models) positioned within the system cube[47]

Figure 2.8 illustrates a principle presentation of a system model to get an idea of what a system model can be or preferably, which information a system model can contain. Furthermore, it is distinguished between a specific model and a system model. First of all, the general illustrations of the principle model frame is shown, which describes the classification scheme in the form of a multidimensional gird. The depth of the cube describes the hierarchical concept of a system. For example, the depth describes if the focus is on the whole vehicle, on the powertrain or only on the engine. The width describes the different technical domains of a system; for example the requirements, the structure, or the behavior of a system. Moreover, the breadth describes the different disciplines such as mechanics, electronics and electrics as well as software, which in turn also generates different points of view. The cube can classify a system model, which means that a model is a system model when it includes more than one technical domain or/and discipline. On the opposite side, a specific model is characterized by one technical and discipline. Both models are needed for the successful use of the MBD approach.[46]

---

[46] Hick et al. 2019.

[47] Hick et al. 2019.

Modeling language SysML

The system model can be defined by using a machine readable language. Therefore, computers require a explicit and clear modeling language. A modeling language consists of clearly defined elements and connection roles. A semi-formal or formal model is created based on a model language and is machine readable. Modeling languages are defined as follows:

> *"The modeling language is an artificial set of rules consisting of individual elements of prescribed meaning (semantic) and rules for their interlinking (syntax). Modeling languages are used to describe models (defined as the original image or image of an original) with the primary purpose of providing clear interpret ability of the content described. "*[48]

In software engineering, *unified modeling language* (UML) has established as a standard modeling language. In 2001, *INCOSE* decided to use UML and adopted the language for the needs of systems engineering. The language is widely used and can be adapted to the specific needs of SE. The adapted version of UML for the needs of SE is called *systems modeling language* (SysML) and is mostly used for systems engineering.[49] The C*onsortium Object Management Grouping* (OMG) adapted UML and further expanded it to SysML. SysML is a graphical modeling language, and the illustrations are called the SysML diagrams. The graphical elements and connections are standardized. In SysML, four diagram classes are defined which are addressing the concept of MBSE:[50]

- System requirements
- System structure
- System behavior
- System parameter

Each diagram class defines one or more diagram types. The diagrams are the representation of the system model. The system model includes the complete information, which contains the elements and the connections between the elements and between the diagrams. The models which are shown in this thesis are developed with the SysML language. The diagram types are described in detail in the following chapters when they are applied.

---

[48] Eigner 2014, p.89 translated from German.

[49] Weilkiens 2014, p.16.

[50] Eigner 2014, p.90 translated from German.

2.3 Functional modeling

Functional modeling can be considered as a part of MBSE. In this section, the term *function* and the topic *functional modeling* are described.

The main approach of functional modeling is to describe the functional topology of a system. Therefore, the function is a specific requirement, action, or activity, which the system has to fulfill. The target of functional modeling is to design the functional topology. The created functional model should be the basis for the following physical system architecture specification. Functional modeling describes *what* the system should do and not *how* the system is realized in order to fulfill the requirements.[51]

As described before, the functional concept is one of the three basic concepts of the system theory of *Ropohl*. The concept uses the black-box view for visualization. The functions can be interpreted as the purpose of the system. Therefore, the functions describe the relationship between the input parameters and the output parameters. The functions of a product can also be understood as the behavior of the system which is expected from the user. The function modeling does not describe the elements. It describes the behavior and provides a description of what a system does and not what is it made of.[52]

The realized functions of a product are also essential for the success of a product. The better the function of the product matches the function of the customers' expectations, the better the product will be marketable. Therefore, it is essential to translate the requirements into functions in a first step and not directly into physical architectures.[53]

2.3.1 Definition of the term function

Like in mathematics also in product development, a function is defined as the connection between the input parameters and the output parameters as illustrated in figure 2.9. In the context of SE, a function is defined as follows: *"Function is the general and intended connection between input and output of a system to fulfill a task."*[54] In the first step, it is not relevant how the transformation is realized. This fact is one of the essential properties of functions because they should be formulated as solution-neutral as possible. Therefore, it has to be ensured that the formulation of the function is based on *what* and not *how*.[55] The input and output parameter can be energy (mechanical, thermal, electrical, chemical, nuclear, etc.), matter or material (gas, fluid, solid, etc.) and signal (parameter, information, impulse, etc.). In general, functions are named with a combination of verbs and nouns, for example *transform energy*.[56]

---

[51] INCOSE 2007, p.292 ff.

[52] Ropohl 2009, p.76 ff.

[53] Lindemann 2016, p.691.

[54] Lindemann 2016, p.691.

[55] Lindemann 2016, p.691.

[56] Feldhusen and Karl-Heinrich Grote 2013, p.346.

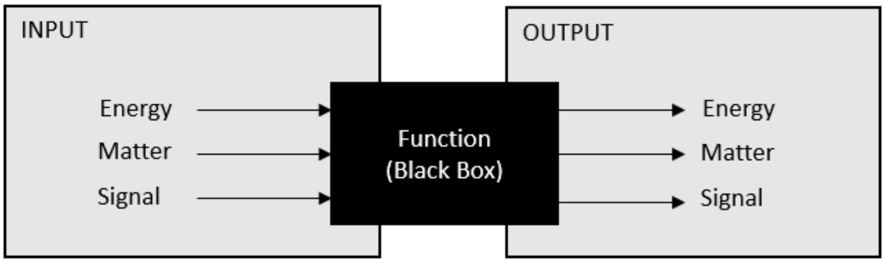


Figure 2.9: Functional connection between input and output[57]

2.3.2 Types of functions

Functions are classified in different types. The following definitions are established in the technical area:[58]

- **Overall function**: describes the model as a whole
- **Sub-function**: describes a subtask of a system
- **Main function**: is a sub-function which is directly derived from the overall function
- **Auxiliary function**: is a sub-function which supports the main function and cannot be directly derived from the overall function
- **Elementary function**: describes a function which is not further divisible and is generally applicable

The specification of function types, as described above is based on the assumption that functions can be divided.

Furthermore, the different function types are used in the development process as follows: The functional description of a product starts with the designation of the overall function, which describes the purpose of the product. The overall function can only be one or a sum of functionalities of the product. Afterwards, the overall function can be divided into sub-functions. Sub-functions are classified into main and auxiliary functions.

---

[57] Feldhusen and Karl-Heinrich Grote 2013.

[58] Lindemann 2016, p.692.

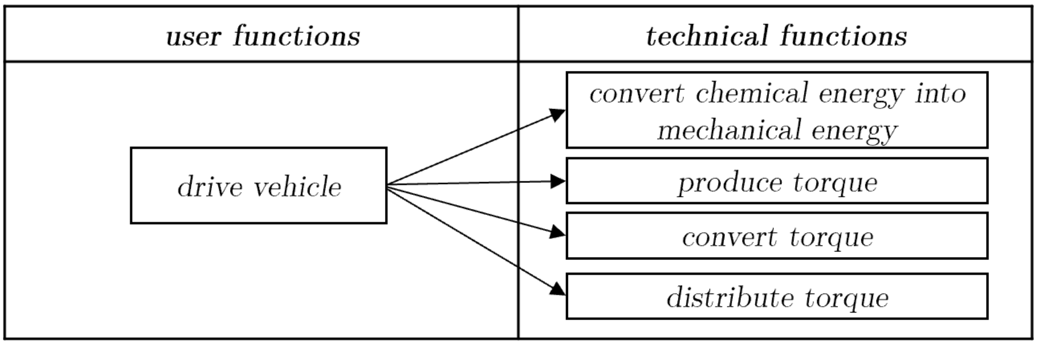


Figure 2.10: User functions vs. technical functions[60]

Additionally, elementary functions are not dividable. *Koller et al.* describe the elementary functions in detail. For example, the elementary functions for energy transformations are: convert, increase/decrease, change direction, conduct, isolate, sum, divide, mix, and separate.[59]

Besides, functions contain information about different points of view. Figure 2.10 shows the powertrain described from two different points of view. The left side shows the view of a user, which describes the function of a powertrain *to drive the vehicle*. On the opposite side, the technical view describes the function as a sequence of energy transformation processes.

2.3.3 Structure of functions

The starting point of a functional description is the overall function. The overall function is described by the input and output relations of the overall system. As described before, the overall function can be divided into connected sub-functions. The visualization of the connection between the overall functions and sub-functions is called the structure of functions. The method of *structure functions* is used to recognize the problems of products and understand the connections between the functionalities of the elements. In general, two structural views are required to describe the system. Both are shown in figure 2.11. The hierarchical view is to brake down the complex overall function in a structural way. Therefore, the overall functions are broken down into sub-functions. The other view is the in/output-view, which is the *mathematics view* on a problem. The illustration shows the input and output parameters connected by a function.[61]

The described procedure from the overall function to the sub-functions is commonly

---

[59] Koller and Kastrup 1998, p.8ff.

[60] Bajzek et al. 2019, S.20.

[61] Lindemann 2016, p.692.

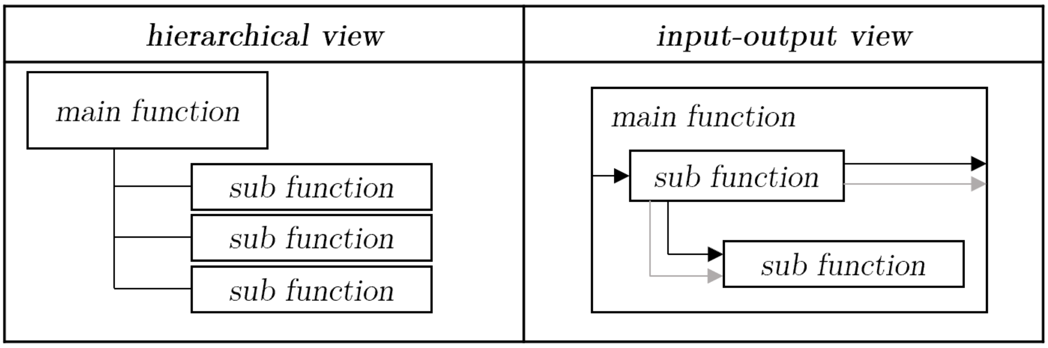


Figure 2.11: Hierarchical view vs. input/output-view[62]

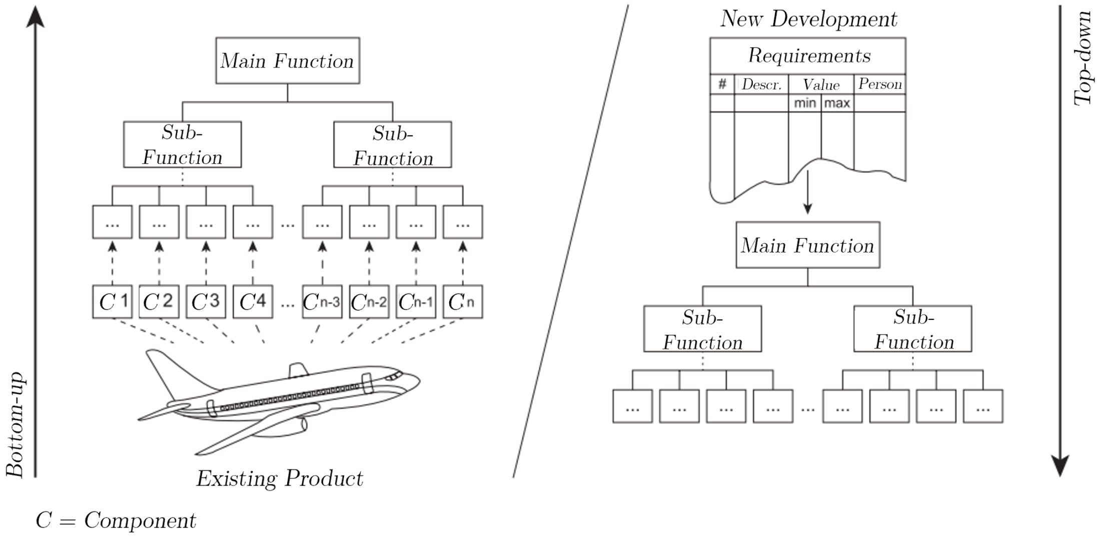


Figure 2.12: Top-down vs. bottom-up approach[63]

used to describe products and follows the top-down approach (figure 2.12). This way is used to structure and describe the product as a solution neutral description which should support the further development. The bottom-up approach follows the procedure in reverse order and is mainly used to analyze the existing products. The output from the bottom-up approach is an understanding of the system functionalities or the identification of potential for improvement.

---

[62] Lindemann 2016, translated from German.

[63] Lindemann 2016, translated from German.

2.3.4 Functional modeling

The functional model is, in general, the illustration of the structure of functions from a system or in further consequence from a product. That means that modeling of functions operates hand in hand with structuring systems because connections between functions are shown and documented.[64] For modeling the following approaches are significantly important:

- **Abstraction**: The functional model allows a powerful solution neutral view. That should be used to get a neutral view of the model instead of a detailed solution at early phases.
- **Dissecting**: The overall function will be divided into sub-functions, which also can be decomposed into sub-functions.
- **Projection**: Projection means that it is possible to consider the function model out of different perspectives.
- **Concentration**: It is also possible to concentrate on some details in order to set their the system boundaries.[65]

There are different types of functional model structures possible. Three types are illustrated in figure 2.13. The simplest structure is a list of functions. It could be a first brainstorming about which functions a product should have. It could also be a list of the functions from a part of the product. The next level of granularity is the function tree. There are the functions shown as overall functions or sub-functions, and they are connected between the levels. The modeling of this type can be top-down or bottom-up. The function model structure with the highest level of detail is the network model. Therefore, each function is connected in a complex system with the other functions. The higher the level of granularity, the higher the effort and work to create the model but also the more detailed the model description of complex systems is.

2.3.5 Approach of functional modeling

In this subsection, the expectations on functional modeling and different approaches or points of view on the area of functional modeling are described. Figure 2.14 illustrates a barrier between a problem and a solution. Every problem, in the context of this thesis a product development problem, is separated from the solution and the barriers are an illustration for thinking in known solution patterns that prevent to get a new innovative solution or even to find some new possibilities. The expectation of functional modeling

---

[64] Lindemann 2016, p.698.

[65] Lindemann 2016, p.702.

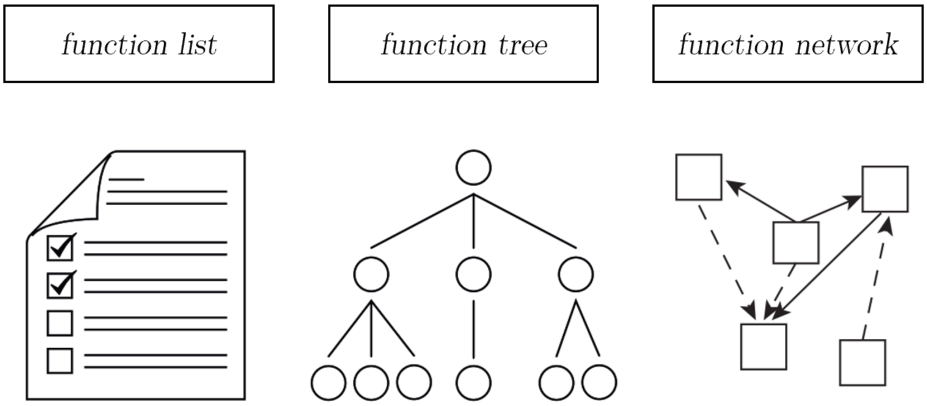


Figure 2.13: The three different types of functional modeling structures[66]

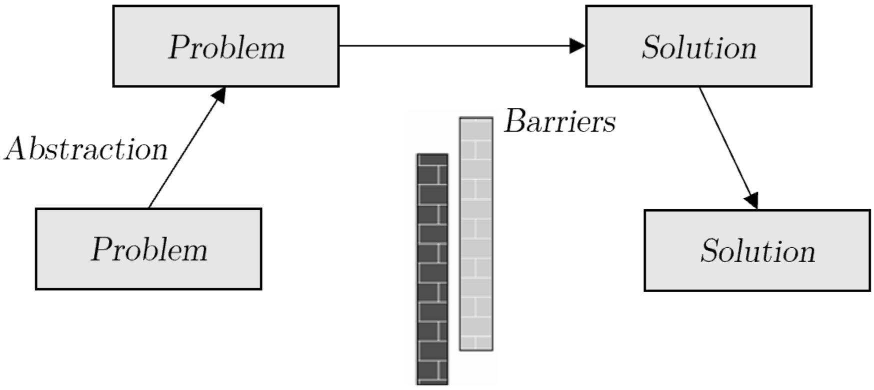


Figure 2.14: Barriers between the problem and the solution[67]

is to abstract the problem. For the functional modeling approach is the abstract version of the development problem, the functional model. The functional model should lead to an ideal abstract solution which can be further developed into a specific solution. This process should help overcome the barriers.

The next two approaches are procedure models with the context of functional modeling. The primary procedure model on which the master's thesis will be obtained is the *V*-model from *Eigner et al.*, which was described in section 2.2.2. The description of the following two models should provide a better overview and is the basis for the discussion at the end.

---

[66] Lindemann 2016.
[67] Lindemann 2005.

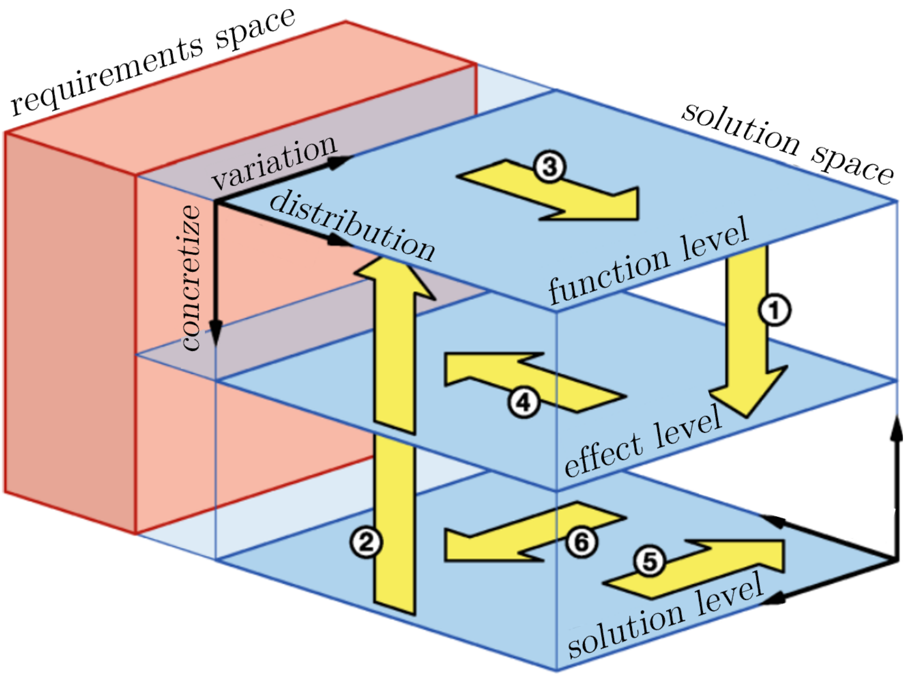


Figure 2.15: Model space of creation[69]

Münchner Poduktkonkretisierungsmodell

The *Münchner Produktkonkretisierungsmodell* is illustrated by a solution space and a related approach. The three dimensions are concretion, variation, and distribution. The requirements space is next to the solution space and illustrates the conditions of the model. The solution space is divided into three levels. The level at the top is the functional level, where the functional model is created. The second level is the effect level, where the functional model is translated into physical effects. For the solution level, the previously defined structure of this physical effect is the basis to find an appropriate physical solution. According to this model, the design process is understood as an ascertainment from functions to the final solution.[68]

---

[68] Ponn and Lindemann 2008, p.32ff.

[69] Ponn and Lindemann 2008, p.32ff translated from German.

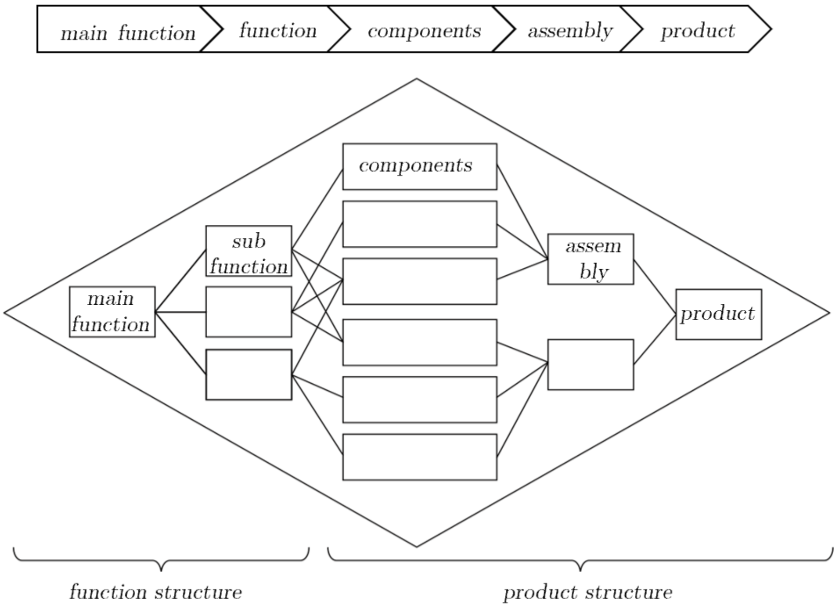


Figure 2.16: Connection between functions and product structure[70]

METUS Raute

Pahl and *Beitz* describe the process for product development as a sequence between a functional structure and a product structure. Therefore, the overall function can be divided into sub-functions. Based on that function tree and specific on the last sub-functions level, the components of the product are selected. One component can support more than one sub-function. The product is built up with a bottom-up approach. This approach should lead to a modular structure of the product and should help to select the components in a solution neutral way.[71]

---

[70] Feldhusen and Karl-Heinrich Grote 2013, p.257ff translated from German.

[71] Feldhusen and Karl-Heinrich Grote 2013, p.257ff.

3 Analysis of the Powertrain System Architectures

This chapter analyzes the powertrain system architecture by following the RFLP approach in reverse order (figure 2.7).

In the beginning of the chapter the powertrain and the boundaries of the powertrain are defined. Afterwards, the basics of a vehicle development process and an overview of actual powertrain architectures are shown. A selection of the actual powertrain range is chosen. This selection of actual powertrain architectures is analyzed according to the bottom-up approach and the procedure model of *Eigner et al.* The result of this chapter is a generic logical model of the powertrain and an overview of the powertrain functions.

3.1 Powertrain architecture

The complete vehicle development process has a duration of several years. The figure 3.1 illustrates stages in a typical state of the art full-vehicle development process. Five phases characterize the process. The process starts with the definition phase, which is an initial characteristic estimation and evaluation of the car model. Therefore, the phase is supported by approaches, for example, market research or trend prognoses. The definition phase also considers the overall product strategy of the manufacturer and boundary conditions such as economic and financial situations. The outcome of the phase should be a list of requirements that describe the product characteristics, for example, the car classification, the main dimensions, and the driving behavior. The target of the following concept phase is the detailed description of the vehicle itself, which includes rough approximation of styling, packaging, and functional details. Based on the concepts, the pre-development phase establishes the vehicle layout by using simulations, testing loops etc. The packaging study also includes an assembly and placement definition and functional optimization. The following series development phase deals with the realization of the concept, with the aid of process and production engineering. The level of details is significantly higher in this phase than in the previous phases and includes the production of final prototypes. The last phase is the pre-series production phase, where the pre-series and series production ramp-up happens. Therefore, the series tools are used, but the process is not entirely in tune. This phase

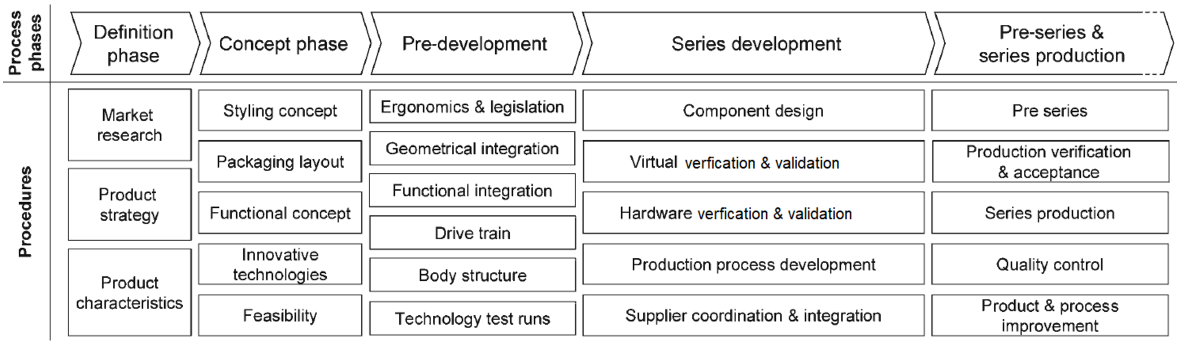


Figure 3.1: Stages in a typical state-of-the-art full-vehicle development process[2]

is used to test the tools and assembly procedures and to evaluate and improve them.[1]

The powertrain plays an essential role in the development process of the car. The powertrain is the heart of the car. Multiple powertrain concepts have been established in the history. The following subsections give an overview of the current state of powertrain architectures.

3.1.1 Definition in the context of the master's thesis

For SE, it is vital to start with the determination of the system boundaries, because that specifies the context of the system approach. To be able to determine the system boundary, it has to be clear what a powertrain is. In a broad sense, a powertrain transforms stored energy in kinetic energy. The powertrain is in the course of this thesis defined as follows:

> "The powertrain contains the main components to generate propulsion power and deliver it to the wheels."

The powertrain includes the energy storage system, the energy transformation system and the energy distribution system. Fundamentally, the brake system is a component of the chassis. This master's thesis considers the brake system as a component of the powertrain instead of the chassis because the brake system is relevant for the functional description of the powertrain.

3.1.2 State of the art of powertrain systems

Combustion engine and electric engine drive the common powertrains. Various primary energy types power different engines. An overview of the connection between

---

[1] M. Hirz et al. 2013, p.12ff.

[2] M. Hirz et al. 2013, p.12.

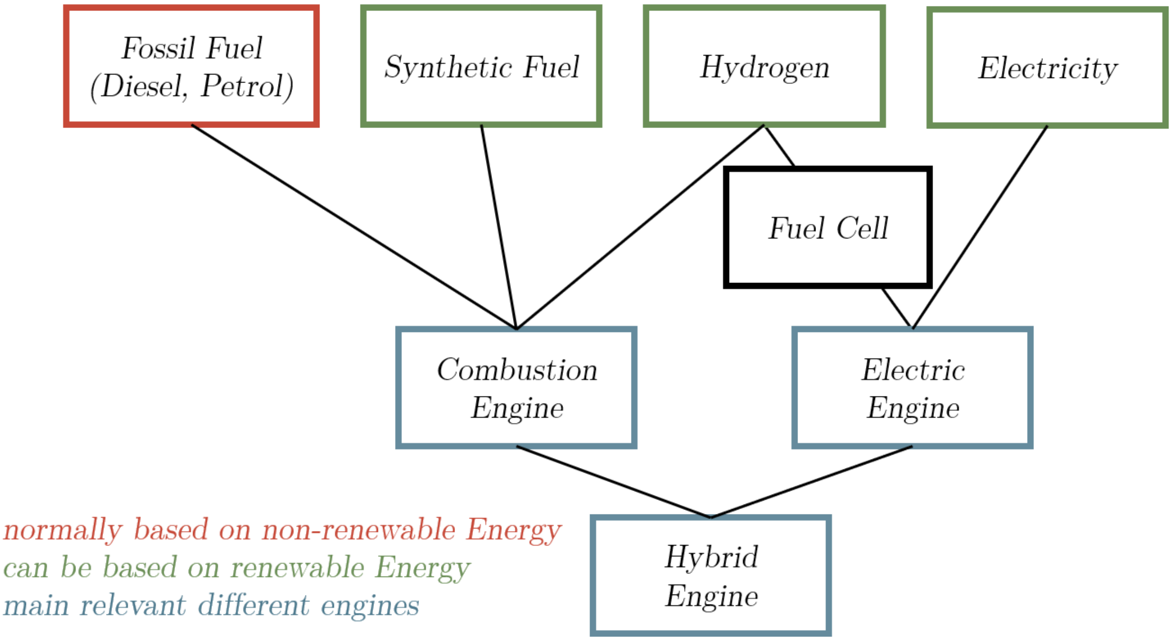
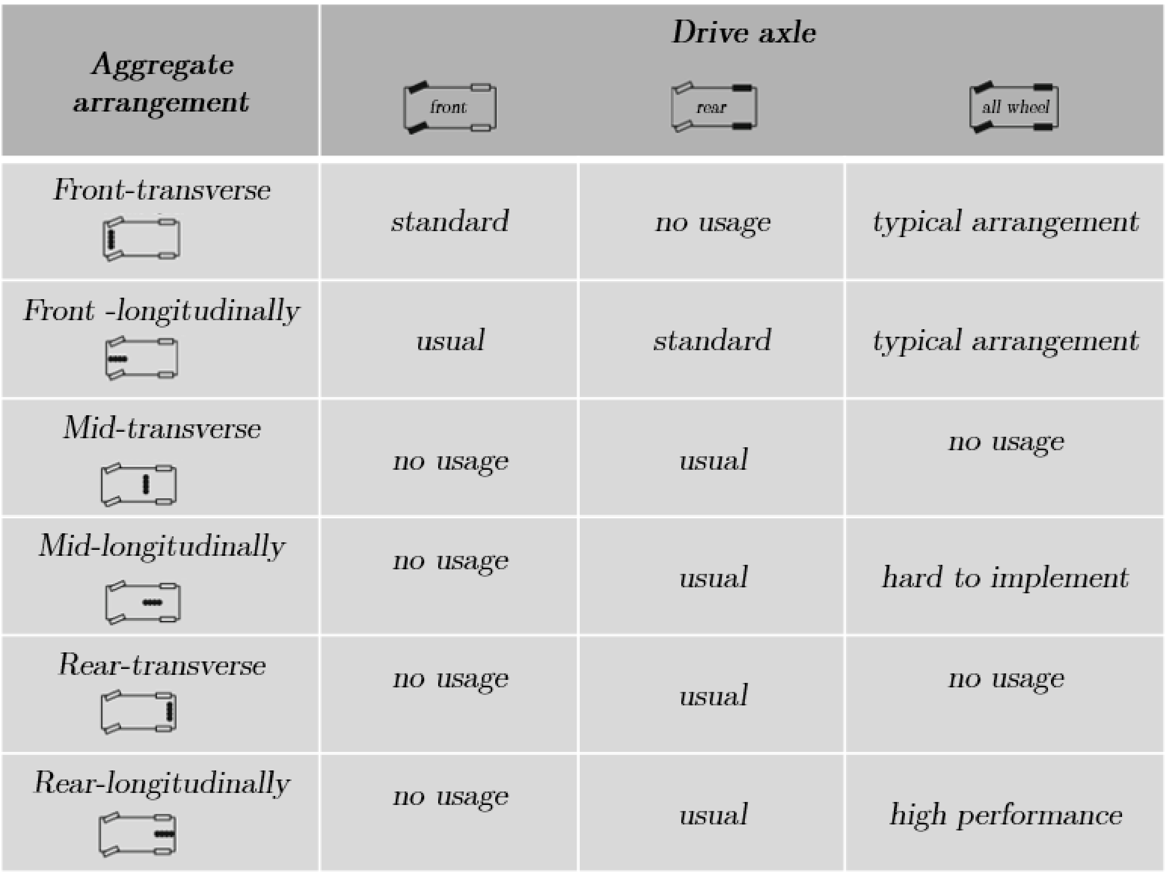


Figure 3.2: Overview of Concepts

different primary energy types and different powertrain engine technologies are shown in figure 3.2. The most traditional established powertrain technology uses the combustion engine, which is usually powered with none renewable fossil energy, for example diesel, or gasoline. Renewable energy sources for the combustion engine are synthetic fuel or hydrogen. The alternative to the traditional combustion engine is the electric engine. This engine is powered by electric energy, which has to be stored in batteries or produced by a fuel cell that transforms the chemical energy of H_2 into electric energy. The hybrid powertrain system consists of both engine technologies. The combustion engine, the electric engine, or a combination of both can be considered as the state of the art of powertrain concepts. Furthermore, the next section explaines the three concepts separately and shows examples.

Combustion engine powertrain architectures

The combustion engine driven powertrain architectures are the most established technology and can be produced by low costs. The system has the advantage of quick refilling, vast driving distances, and has the potential for further improvements. Nevertheless, the system also has disadvantages like thermodynamically bad efficiency, local exhaust emissions, and actually direct dependency on crude oil. However, the existing range of architecture is enormous because it is the most established technology. Figure 3.3 shows a matrix which describes possible aggregate arrangements regarding to defined driven axles. It can be distinguished between a front, back, or all axle



Aggregate arrangement	*Drive axle* *front*	 *rear*	 *all wheel*
Front-transverse	*standard*	*no usage*	*typical arrangement*
Front -longitudinally	*usual*	*standard*	*typical arrangement*
Mid-transverse	*no usage*	*usual*	*no usage*
Mid-longitudinally	*no usage*	*usual*	*hard to implement*
Rear-transverse	*no usage*	*usual*	*no usage*
Rear-longitudinally	*no usage*	*usual*	*high performance*

Figure 3.3: Power unit arrangement to driven axle[4]

driven vehicle. The range of power unit arrangements starts with the standard front length or cross arrangement up to middle and bag length or cross arrangements.[3] Many architectures are conceivable for each crossing point of the matrix. Figure 3.4 shows examples of front power unit arrangements. The main parts of the shown architecture are the combustion engine, transmission (G), and differential (D). Each of the many combination possibilities has advantages and disadvantages.

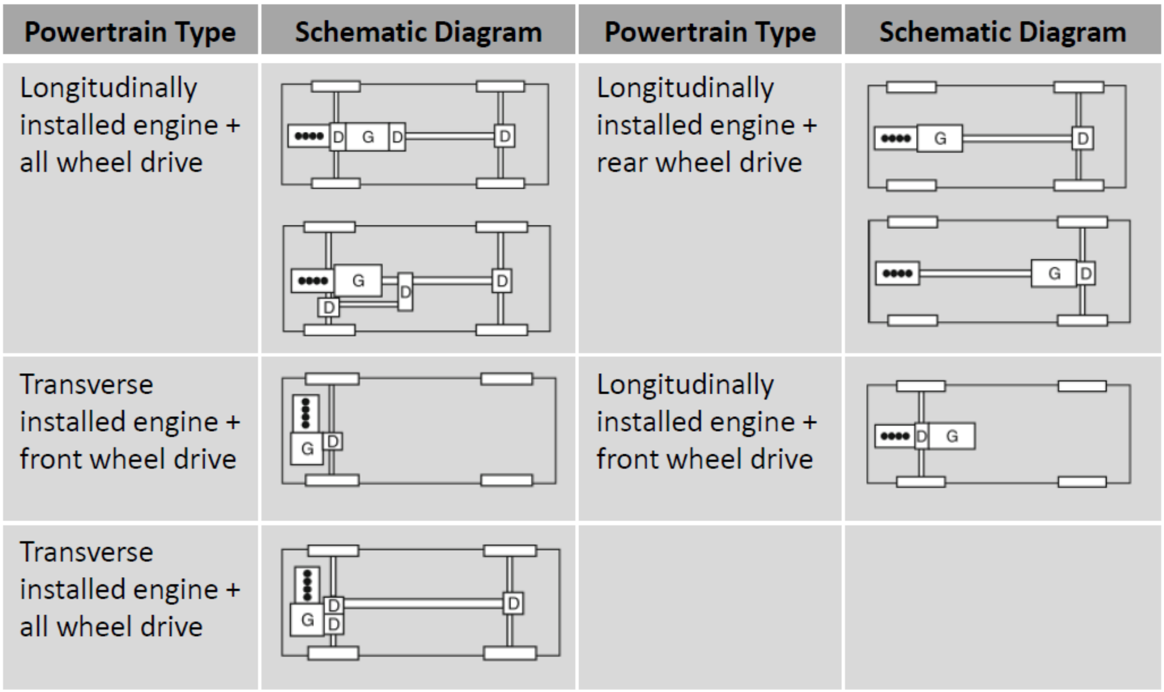
Electric engine powertrain architectures

The electric engine driven powertrain primary consists of a battery, in which the electric energy is stored and an electric engine that transforms the electric energy into mechanical energy. The energy can also be stored with hydrogen, whereby a fuel cell transforms the hydrogen into electric energy. The fuel cell technology will not be further discussed

---

[3] P. D. M. Hirz 2018, Slide:27.

[4] Braess 2013, p.144.



Powertrain Type	Schematic Diagram	Powertrain Type	Schematic Diagram
Longitudinally installed engine + all wheel drive		Longitudinally installed engine + rear wheel drive	
Transverse installed engine + front wheel drive		Longitudinally installed engine + front wheel drive	
Transverse installed engine + all wheel drive			

Figure 3.4: Example of different architectures[5]

in this master's thesis.

The electric driven powertrain with a battery storage system is the most efficient propulsion technology (depending from energy source) and emits no local emissions. This powertrain is outstanding for a quiet technology with high driving comfort. The main disadvantage of those systems is the expansive and complex battery systems. The battery system also has the problem of driving distance capacity and charging times. The purpose of an environment-friendliness depends on the technology for electric power generation. An additional problem is the extraction of some batteries materials.[6]

However, there are diverse possibilities of electric driven powertrain architectures. Figure 3.5 illustrates three different possibilities. Figure 3.5 a) shows a powertrain architecture example, which can be realized as a front-wheel-drive or back-wheel-drive. There are many other architectures conceivable. Figure 3.5 b) shows a tandem driven powertrain and figure 3.5 c) shows wheel hub driven powertrain. The electrically driven powertrain concepts are not as established as combustion driven powertrain concepts. Therefore, there are still many conceivable variants that have not yet been implemented, especially with the electric engine driven powertrain.

---

[5] Braess 2013, p.146f.

[6] P. D. M. Hirz 2018, Slide:27.

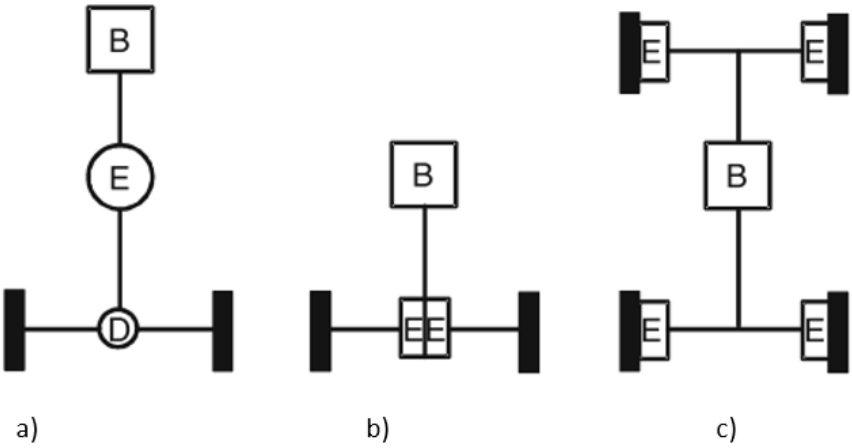


Figure 3.5: Example of different architectures[7]

Hybrid powertrain architectures

A hybrid system is a combination of a combustion engine and an electric engine in one powertrain concept. By combining them, it is possible to use the advantages out of both. Therefore, hybrid systems have excellent efficiency and have a long driving ranges. Nevertheless, the disadvantage is the system complexity because of the integration of two propulsion systems. That demands tremendous resources in production and is expansive. In figure 3.6, three different basic concepts are shown. At the serial concept, only the electric engine drives the car, and the combustion engine produces electricity with a generator. The second concept shows a parallel concept where both engines can drive the car. At last, a mix of serial and parallels is also conceivable. Also, the possibilities of hybrid architecture seem endless.

3.2 The modeling language SysML

This section explains the basics of the modeling language SysML already presented in section 2.2.3, before the bottom-up approach is explained in detail. Figure 3.7 shows different diagram classes. Four basic classes namely the requirements, the structure, the behavior, and the parameter classify the different diagrams. This thesis focus on the *block definition diagram*, *internal block diagram* and the *activity diagram* for the functional modeling of the system structure. For modeling the powertrain structure of

---

[7] Braess 2013, p.162.

[8] Braess 2013, p.188.

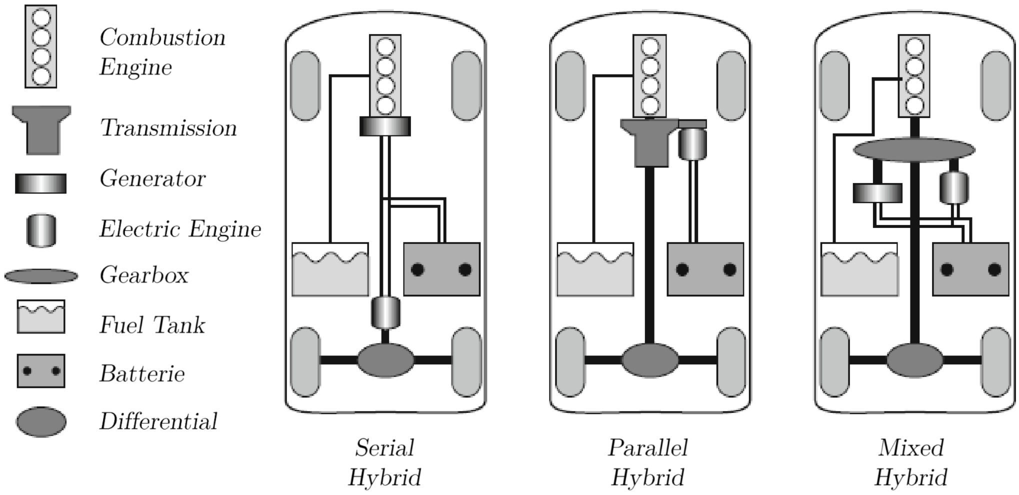


Figure 3.6: Basic hybrid concept[8]

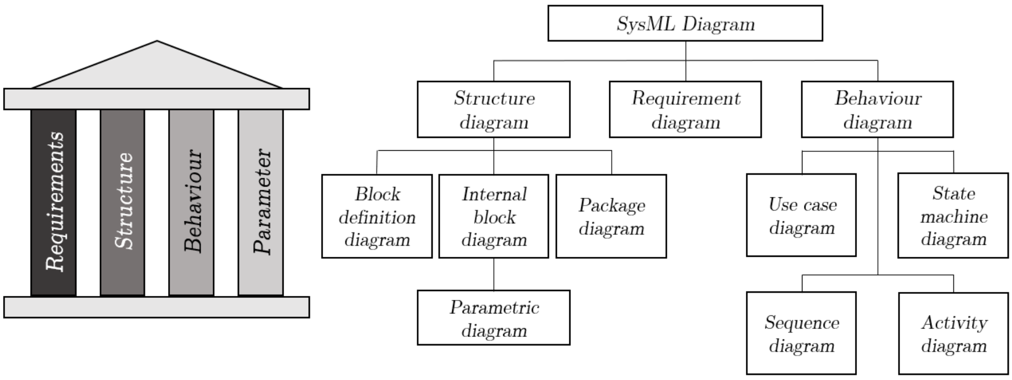


Figure 3.7: SysML diagrams[9]

systems the *block definition diagram (BDD)* and the *internal block diagram (IBD)* are relevant. The BDD diagram shows the elements or sub-systems of the system manual as blocks and can further visualize the defined properties of each element or sub-system. The IBD diagram defines the topology of a system. The diagram shows the internal structure by connecting different blocks in the context of the upper system. The black-/grey-/white-box approach can be implemented by connecting blocks and creating inputs and outputs. Finally, the activity diagram models the functional processes. Activity diagrams are modeled by connecting functions in flow charts to show processes hierarchically.

---

[9] Eigner 2014, p.90.

3.3 Bottom-up approach

As described before, the vehicle development process starts with the definition phase. This phase translates the needs of the stakeholders into requirements. Along the process, these requirements are successively developed further over the different phases to a final vehicle. The functional modeling approach is part of the presented modified V-model (chapter 2.2.2) as well as from the phases development description. The master's thesis analyzes the process in reverse order, from physical solutions to the requirements, with a focus on the system functionalities and the modeling approach. Figure 3.8 shows this approach. In the first step, three representative state of the art powertrain architecture are selected. These architectures are modeled with the *block definition diagram* and *internal block diagram*, which represent the structure of each powertrain. These different powertrain-specific diagrams are combined into a single generic BDD and a single generic IBD, which are representative for a high number of different powertrain architectures. Finally, the functions and further the functional model should be derived from this generic structure diagrams. This approach ensures the definition of functions in a scientifically substantiated way.

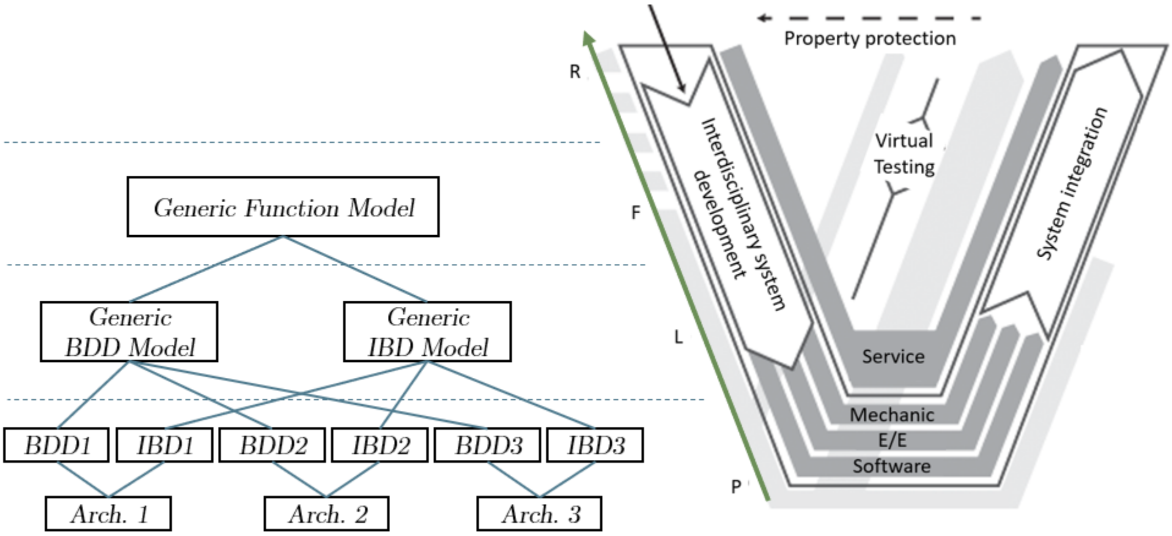


Figure 3.8: Bottom-up approach

3.3.1 Powertrain examples

Figure 3.9 shows the three different examples of powertrain architecture. All three architectures are examples from the *Volkswagen* (VW) group in order to reduce the impact of different competitors. The *first architecture*, a VW Touareg, represents the typical combustion engine driven powertrain architectures. The *second architecture*, a VW Passat GTE, represents the group of hybrid driven powertrain architectures. The

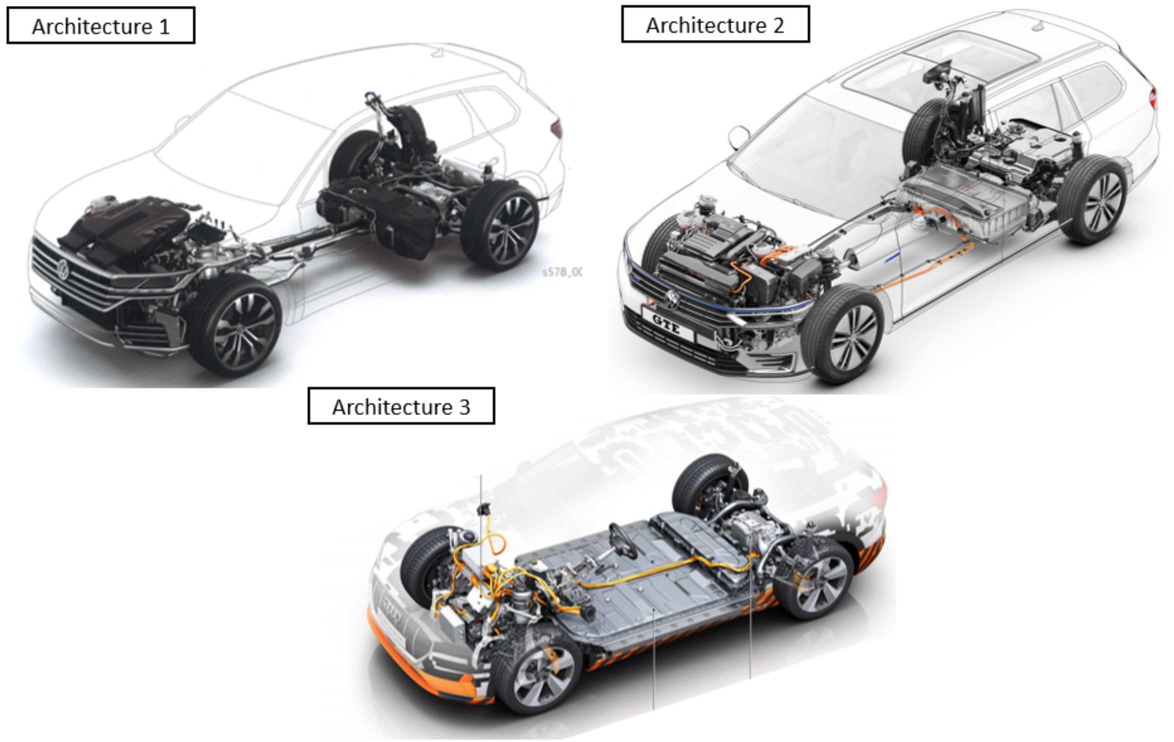


Figure 3.9: The three different powertrain architecture examples[10]

third architecture, an Audi e-tron, represents the group of electric driven powertrains. These three powertrain architectures are the representation of the actual state of the art of powertrains.

In the first step, the powertain architectures are illustrated in a schematic representation. In the second step, the specific BDD and IBD models of each powertrain architecture are created based on the schematic representation. The procedure is explained based on *architecture 3*, the Audi e-tron. The procedure of the *architecture 1* and *2* are similar, and the associated diagrams are shown in the appendix.

Figure 3.10 shows the schematic representation of the powertrain *architecture 3*, the Audi e-tron. Two electric engines drive the vehicle. Both engines are asynchronous motors with a regular power of 70kW at the front wheels and 95kW at the rear wheels. The front-engine is connected to the front wheels via a one-gear-transmission with integrated differential by using a parallel concept. The rear-engine is connected to the rear wheels via a one-gear-transmission with integrated differential by using a coaxial concept. The battery stores the electric energy and has a capacity of 83,6kWh, a charging capacity of 150kW, and a weight of 700kg. Those are the main components of the Audi e-tron. The schematic representation does not include all necessary additional

---

[10] Volkswagen 2019, Homepage.

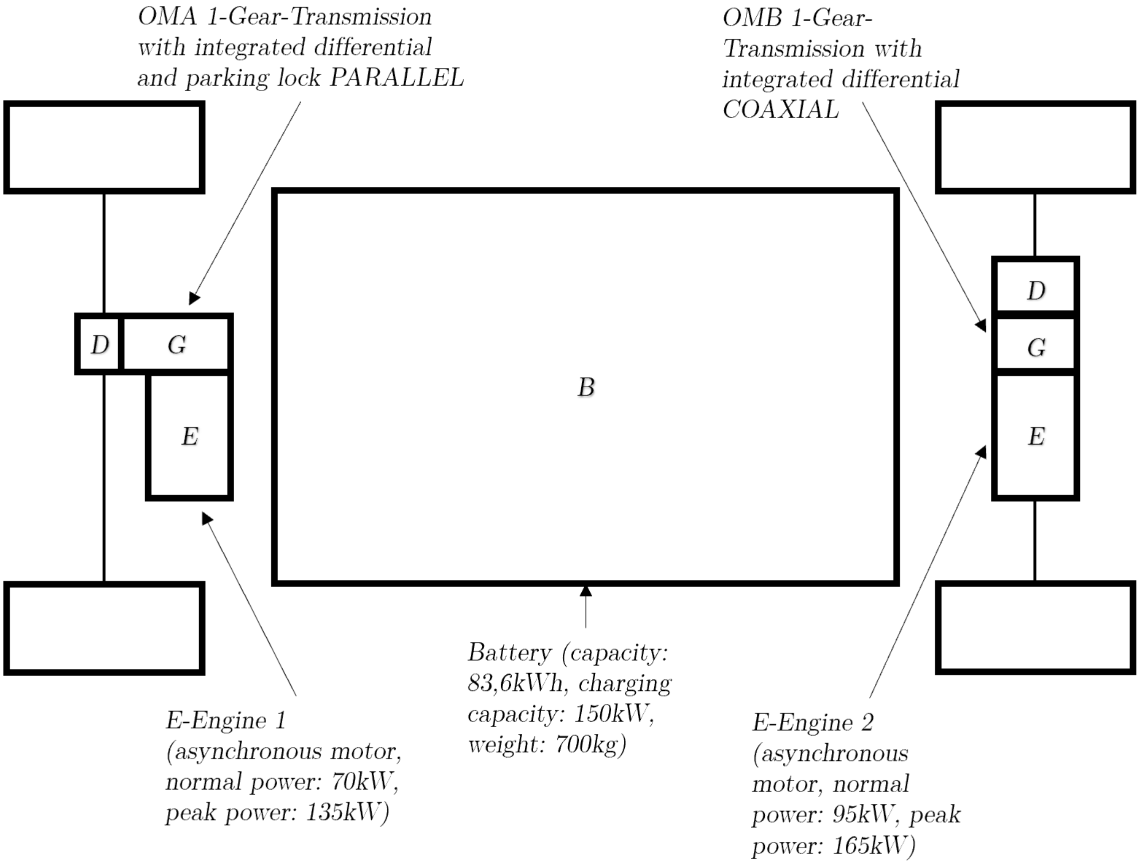


Figure 3.10: Schematic representation of the powertrain *architecture 3*, the Audi e-tron

components.

In the next step, the specific BDD and IBD are created based on this schematic representation. Both diagrams are created with Microsoft Power Point. The generic models of the logical and functional level are modeled with the explained SysML software.

The figure 3.11 illustrates the BDD of the powertrain *architecture 3*. Each block in the diagram represents one ore more components of the real example, and the properties are listed below each block. The diagram contains the explained parts, the engines, transmissions with integrated differential and the battery. Furthermore, the diagram contains additional components such as the braking system, cooling/heating systems, battery charger, voltage converter and the grouped block of the control units such as ECU, PCU, etc. The BDD contains the information about the components and the properties of the components of a system.

The structure of the powertrain example is modeled in an IBD diagram. The figure 3.12 illustrates the associate IBD of the exemplary powertrain *architecture 3*. The whole box represents the powertrain. The defined input and output connectors join the powertrain system and other systems/elements in the environment. The IBD can

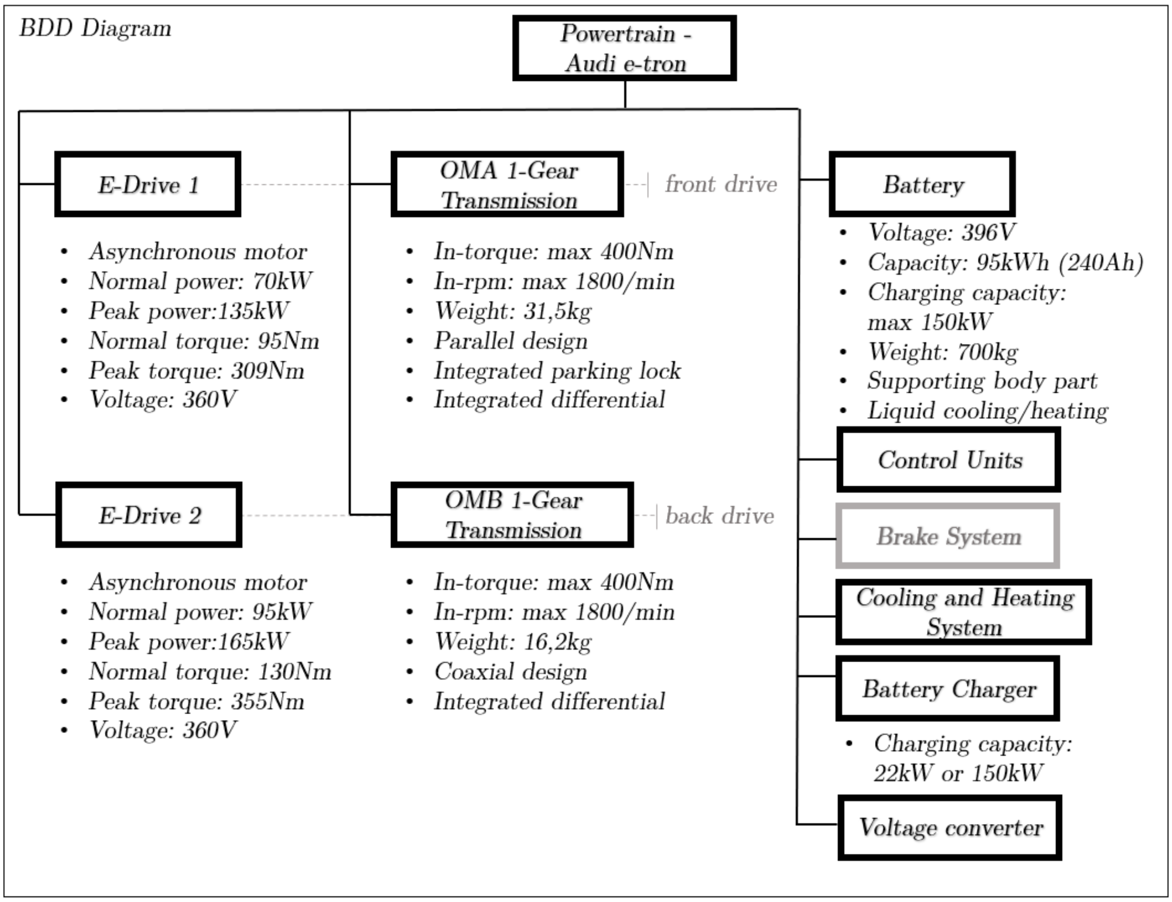


Figure 3.11: BDD of the powertrain *architecture 3*, the Audi e-tron

contain every block of the associated BDD diagram. The diagram shows the principal structure of the powertrain by connecting all inputs and outputs of the sub-systems and elements. The diagram represents the structure of the specific powertrain architecture example Audi e-tron. The appendix contains the BDD and IBD of the other powertrain architecture examples. These three BDD and three IBD give an overview of existing powertrain architectures. The BDD contains a hierarchical view of the used components, and the IBD contains the associated topology of the example. The next step is to create a generic BDD and a generic IBD model based on the specific BDD's and IBD's.

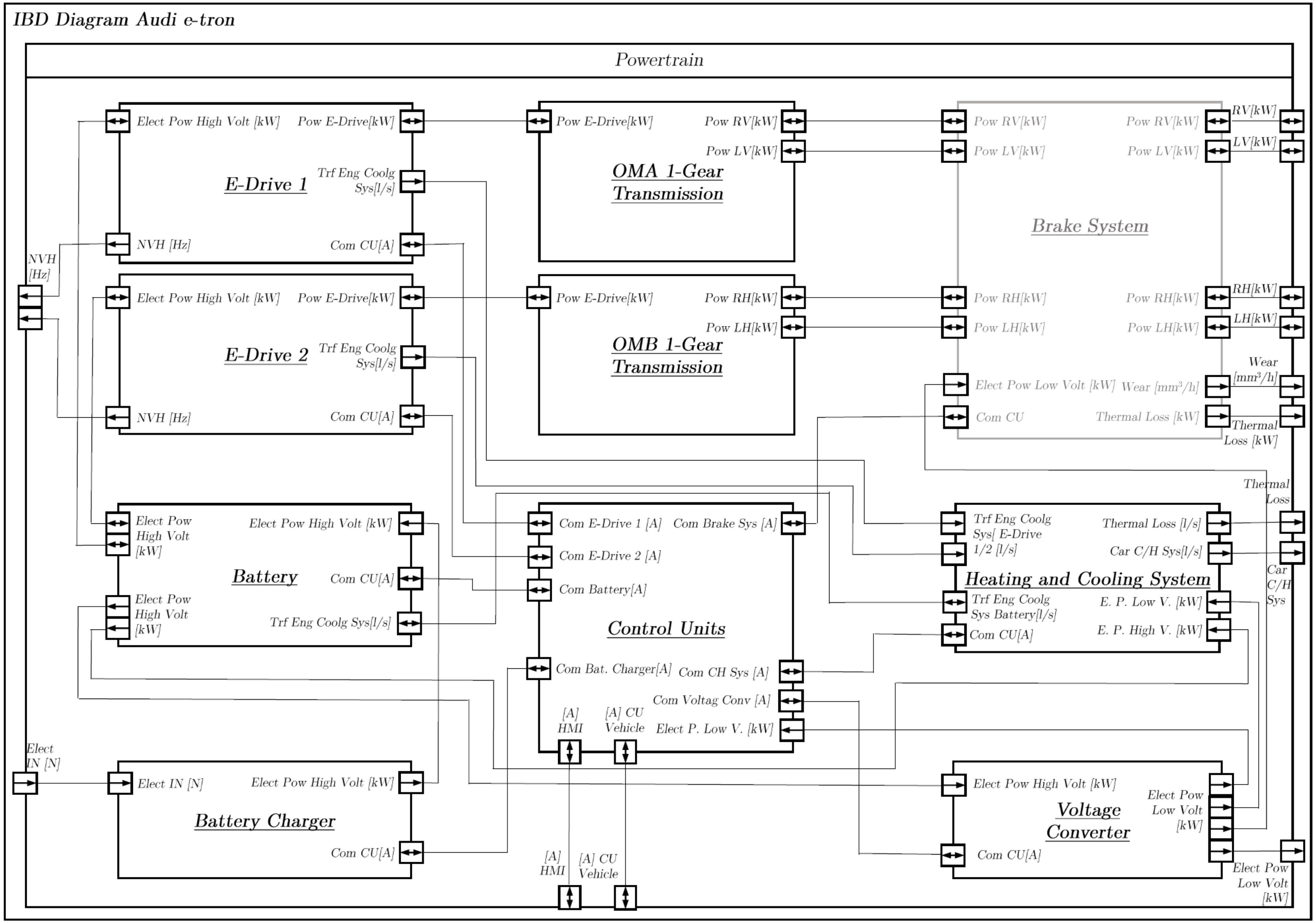


Figure 3.12: IBD of the powertrain *architecture 3*, the Audi e-tron

3.3.2 Logical description

This section explains the creation of a generic model, based on a bottom-up approach, starting with on the specific models of the powertrain architecture examples.

The creation process starts with the definition of a generic BDD. In the beginning, all defined parts/blocks of the specific BDD's are collected together to a pool of blocks. The powertrain-specific blocks out of the pool are sorted to groups based on defined functionalities and properties. Those blocks/elements are assigned to a generic sub-system, which in turn creates the generic system. By considering the previously defined systems hierarchy are those generic sub-systems an intermediate layer between the powertrain system and the element level. The figure 3.13 illustrates the associated modified BDD diagram with the different layers. The following list describes each generic sub-system shortly:

- **Low/high voltage systems**: This system includes all supporting electric elements in the powertrain system. It bases on the powertrain-specific elements, the low voltage batterie supporting combustion engine, the LV/HV voltage converter and the HV battery charging system of the electric powertrain, and the overall wiring system of the powertrain. Those elements have the electronic support of the powertrain in common.

- **Energy storage and support systems**: This system represents the storage of the primary energy, which is used to power the vehicle. The word *support* is regarding the fact that the primary energy may not always be used to drive the vehicle. This virtual system contains the fuel tank of a combustion engine driven powertrain, and the main battery of an electric enigne driven powertrain.

- **Energy transformation systems**: The function of this system is to transform energy into different types of energy. Therefore, the energy transformation system converts the primary stored energy into mechanical energy, which drives the wheels in order accelerate the vehicle. Various primary energy courses require different transformation systems such as the combustion engine to transform chemical energy into mechanic energy or the electric engine to convert electrical energy into mechanic energy.

- **Energy transportation and distribution systems**: The transportation and distribution system connects the transformed kinetic energy from the transformation system to the wheels. The word *distribution* accords the fact that the transformed primary kinetic energy has to be distributed to the different wheels of the vehicle. This virtual system contains for example the transmission, the transfer case, or the differential.

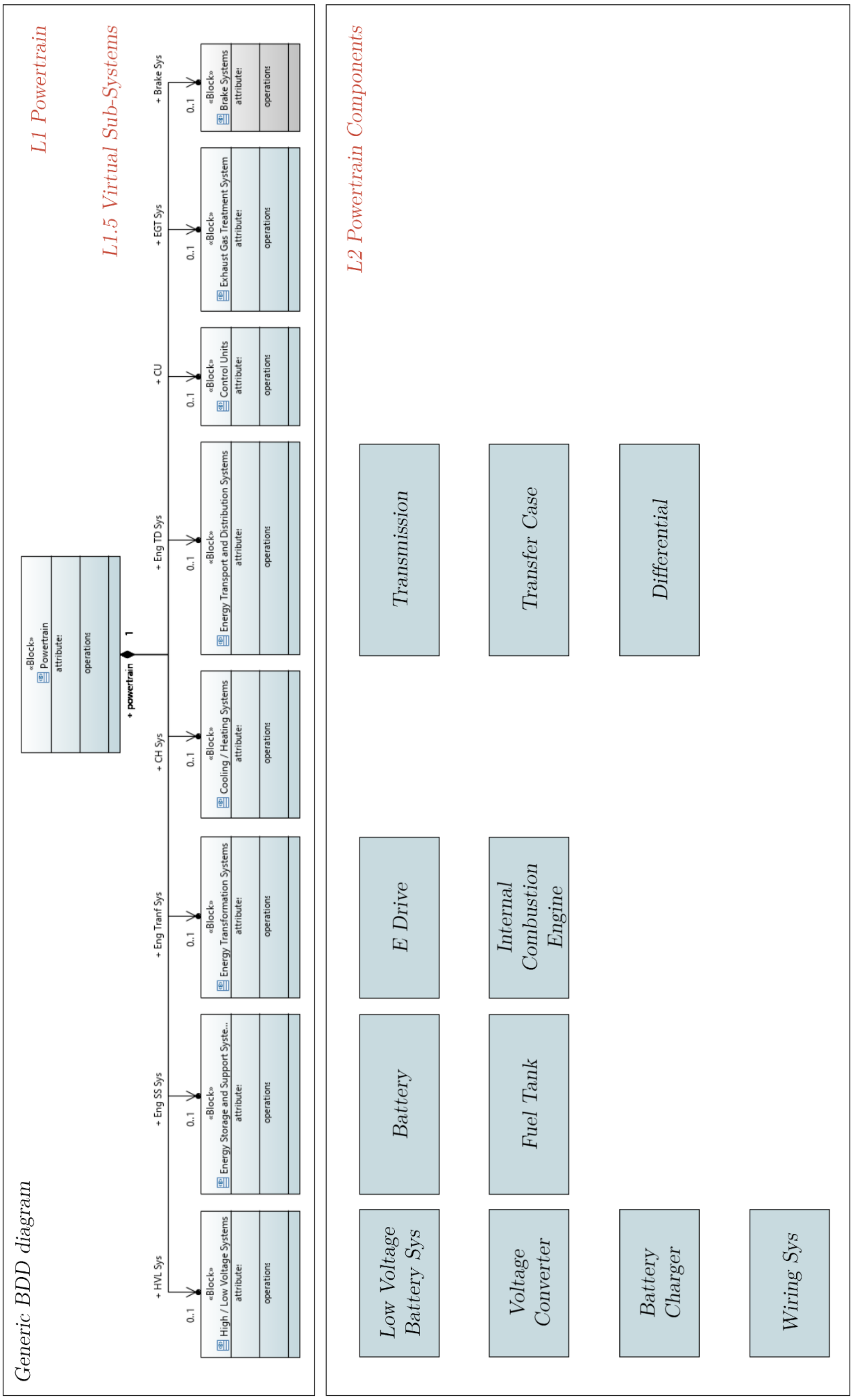


Figure 3.13: The modified generic BDD diagram

- **Cooling/Heating Systems**: The cooling/heating system includes the whole thermal management of the powertrain. The solutions regarding these thermal systems are very diverse. For example, the cooling/heating system of a combustion engine driven vehicle have to manage the thermal situation of the engine and transport the waste heat to places where it is needed or to derive it from the powertrain. Another example is the electric driven powertrain, where the system has to manage the temperature of the battery.
- **Control Units**: This system contains all control units of the powertrain. It is responsible for the collection, processing, and distribution of information. The control units are the communication platform of the powertrain and also communicate with other systems in the system environment.
- **Exhaust Gas Treatment Systems**: The main goal of this system is to reduce the environmental impact of the exhaust of the vehicle. Therefore it depends on the used propulsion technology if a exhaust gas treatment system is required or not.
- **Brake Systems**: As described in the section 3.1.1, the brake system is considered as a subsystem of the powertrain. By definition, the brake system reduces the kinetic energy of the vehicle.

After defining the generic sub-systems of the powertrain, where each sub-system represents a couple of elements with corresponding functionalities and properties, the boundaries of those systems have to be defined to be able to create an according IBD. Figure 3.14 shows the approach for the definition of the generic sub-systems based on the example of the transformation system. Additionally, the other systems are listed in the appendix. The orange block represents the virtual sub-system *energy transformation system.* The red and green block represent the specific components, the e-drive (electric engine), and the internal combustion engine. The boundaries of each components are specified by defining inputs and outputs. *Ports* are used in SysML to represent these in/outputs. The arrow of each port indicates an input, output, or input and output combined. The color of the port indicates different types of energy, matter, or signals:

- **Grey**: Mechanical Energy
- **Red**: Electric Energy
- **Blue**: Thermal Energy
- **Green**: Matter
- **Yellow**: Communication Signal

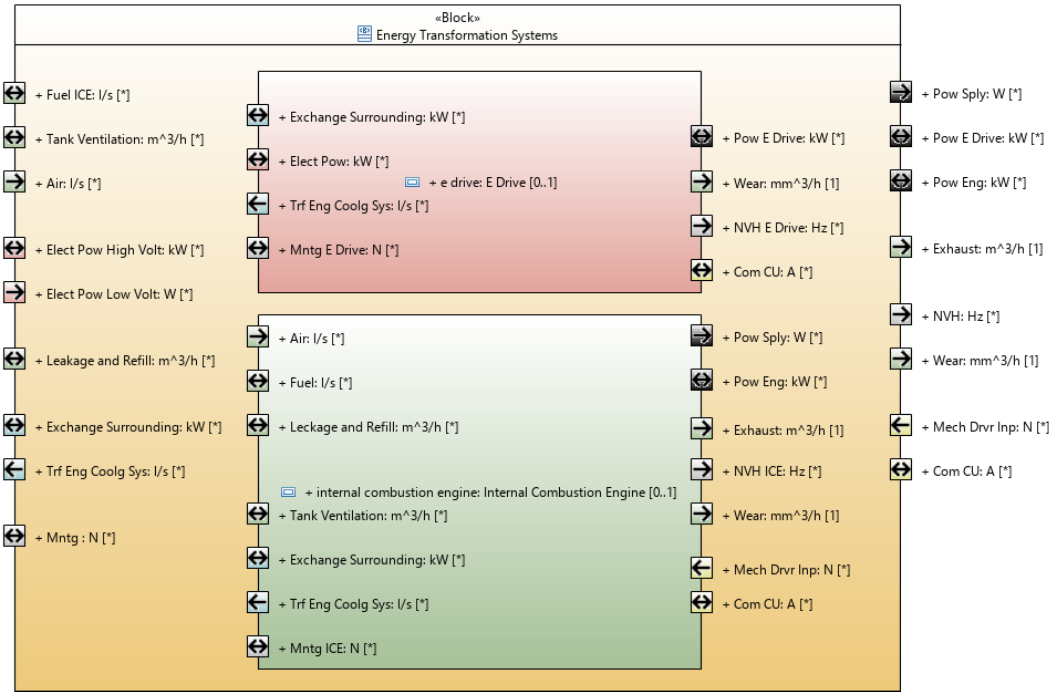


Figure 3.14: Defintion of boundaries for the energy transformation system

The generic energy transformation system represents all the containing systems. Therefore, the generic system must have all the boundaries of the specific systems. So, the boundaries of the generic systems is a sum of the represented specific components. The more specific components are included, the better is the description of the generic block. Table 7.1 and 7.2 in the appendix describe the exact definition of each port in detail.

Figure 3.15 illustrates the IBD of the powertrain (a large version is appended at the end). The IBD shows the generic structure of the powertrain based on the generic sub-systems. It consists of all generic sub-systems of the associated generic BDD. The creation process of the IBD Diagram starts with the generic sub-system blocks and their defined ports. A logical connection of the ports creates relationships. There are two types of relationships. The first type is a relationship between two ports of the sub-systems. The second type is a connection between a port of a sub-system and a port on the powertrain boundaries. The creation process is finished when every defined port of a sub-system is connected with another one. After this process, also the ports of the powertrain system are defined based on the generic sub-systems and the process of connection.

The IBD represents the generic structure of all three powertrain architecture examples - this means that all three specific architectures can be derived from the generic structure diagram.

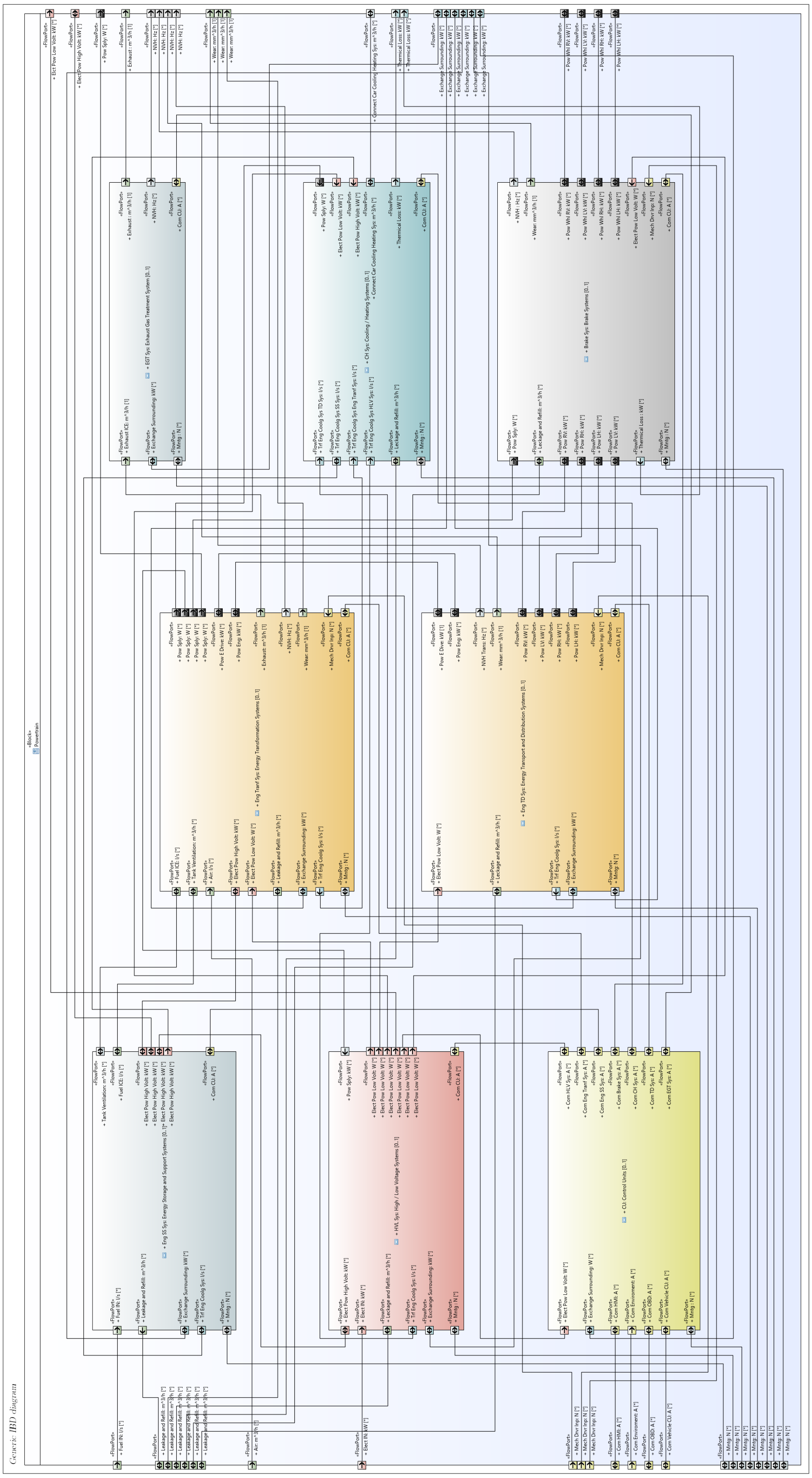


Figure 3.15: Generic IBD diagram

3.3.3 Functional description

As defined in chapter 2.3.1, a function is a general and intended connection between input and output of a system to fulfill a task. Accordingly, the functions of a system can be defined based on ports. Furthermore, it should be emphasized that a function is described by verbs and nouns and should be as solution-neutral as possible.

The approach is shown for the example of the sub-system *energy transformation system* (figure 3.16). The main functions of that sub-system are to *transform stored energy into kinetic energy* or to *transform kinetic energy into storable energy*. This function connects the ports *air*, *tank ventilation*, *fuel*, *elect pow high volt* with the ports *pow e-drive*, *pow ICE*. It consists of a verb and a noun, and it is a solution-neutral description. Besides, many auxiliary functions are defined in the same way, for example, to *supply auxiliary units with mechanic power* or that the system has to be *mount at the chassis*. The system can also have secondary functions, for example, the function to *emit noise vibration harness and exhaust*. These functions are marked with a black triangle. With this approach, functions for every block can be defined. Furthermore, the main powertrain functions are described, but the appendix includes the functions of every generic sub-system.

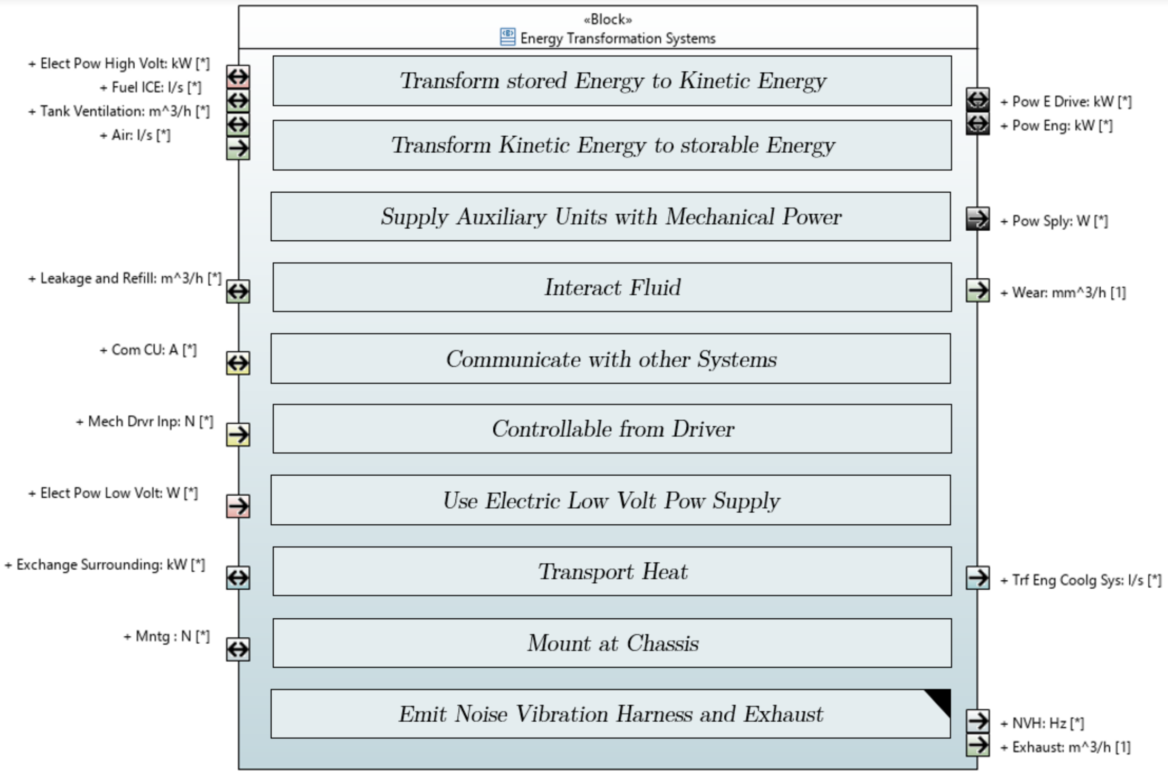


Figure 3.16: Definition of function of the transformation system

The figure 3.17 shows the function of the powertrain block. The main function of the powertrain is to *change the kinetic energy of the vehicle*. This function connects the input port's *air*, *fuel*, and *electric energy* with the output *mechanic power*, which drives the vehicle. Many auxiliary functions are besides the main function defined. For example, one of the auxiliary functions is *controllable by the driver*. The communication can take place via a mechanical input or via an electronic interface. The powertrain also has to *communicate with other control systems*, for example, with the environment, the vehicle CU or OBD interfaces. The powertrain also has the function to *supply the vehicle with energy* such as high/low voltage energy or mechanic energy. The function *mount on chassis* addresses the fact that the powertrain is an essential part of the car packaging. Furthermore, the function *interact fluid* addresses every changeable or not changeable liquid in the powertrain, which has to be re-fill or may expire. The function *exchange thermal energy* describes the thermal loss and the general thermal exchange with the surroundings and the connection with the vehicle cooling/heating system. The function *emit emission*, includes emissions such as NVH, PM, or exhaust (CO_2, NO_x, etc.).

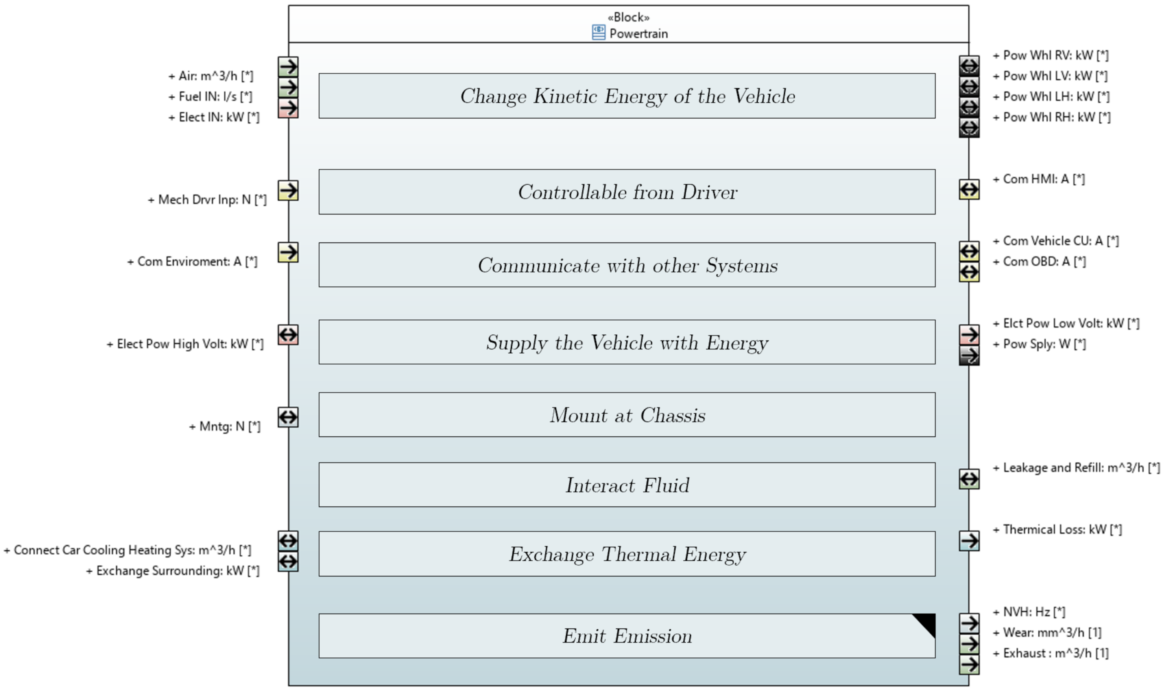


Figure 3.17: Definition of functions of the powertrain system

The activity diagrams are used to describe the behavior of a system. The behavior can be allocated to the functions of the upper level. Figure 3.18 shows an activity diagram of the function *change the kinetic energy of the vehicle* for the case that the driver makes a *positive power request*. Swimmlines represent different sub-systems.

The sub-systems contain the defined functions. The activity connects the necessary functions of the various sub-systems. For the simplified example, this means that energy, such as electricity or fuel, is first buffered with the function *store energy*. The stored energy is afterward converted into mechanical power and transferred to the wheels. In this simplified example, the functions of the control unit were neglected and directly specified via requests.

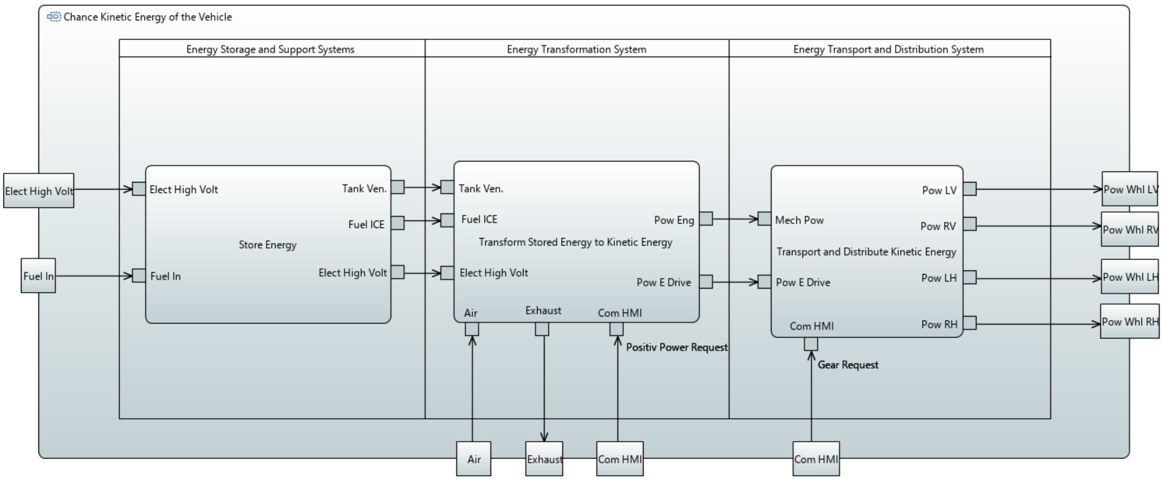


Figure 3.18: Activity diagram for the function *change kinetic energy of the vehicle* for the case *positive power request*

4 Powertrain System Specification

This chapter describes the powertrain system specification process. The first section introduces the concept and procedure of the applied top-down approach. The second section shows the described method used in a powertrain case study.

4.1 System specification approach

This section describes the approach and procedure to specify a functional model. It is essential for this approach to define the meaning of technical functions. Therefore, the subsection *different engineering meanings of technical functions* describes three archetypical meanings of functions. Afterwards, the subsection *conceptual approach and procedure* describes the conceptual idea and procedure of the used approach. Additionally, the last subsection *example* applies the described approach to a simple example the light bulb, before the approach is applied to the powertrain case study in the following section.

4.1.1 Different engineering meanings of technical functions

Engineers use the term function for different meanings. *Vermaas'* paper reviews different meanings of the term function and how a simplified account of functions is beneficial. *Vermaas'* explanation focuses on the description of technical devices.[1]

The starting point of his description of various meanings of the term function is *Brown* and *Blessing's* five-key-concepts for devices. This concept describes the specification process of a device or system as a sequence of goals, actions, functions, behavior, and structure.[2] The table 4.1 describes the five terms in detail. This sequence is very comparable to the sequence of *Eigner et al.* procedure model (table 4.2), which is shown in chapter 2.2.2 and describes the process as a sequence of requirements, functions, logical an physical.[3]

The five-key-concept is explained by using a simple example of a light bulb as given in the table 4.3. First, the goal is formulated: *illumination of a room.* Second, the actions that have to be carried out with the device are planned: *put the lamp in the room and connect it to the lamp.* Third, the functions of the device have to be defined

---

[1] Vermaas 2010, p.183.

[2] Brown and Blessing 2005.

[3] Eigner 2014, p.89.

Table 4.1: Description of terms

Term	**Description**
Goal	A state of affairs desired to be realised by an agent with the device
Action	A deliberate manipulation the agent is to carry out with the device in order to achieve the device's goal
Function	An effect the behavior of the device has to have for letting the device's actions be successful
Behavior	The evolution of the physical state of the device
Structure	The physical configuration of the device in its enviroment

Table 4.2: Procedure model of *Eigner et al.* vs. procedure model of *Brown and Blessing*

Model	**Description**
Eigner et al.	Requirements – Functions – Logical – Physical
Brown and Blessing	Goal – Action – Function – Behavior – Structure

Table 4.3: Examples for five-key-concept

Term	**Light Bulb**	**Powertrain**
Goal	Illumination in a Room	Drive the Vehicle
Action	Put the Lamp in the Room and connect it to the Mains	Give the Powertrain in each moment Information about Power request and Gear selection
Function	Convert Electricity into Light	Change Kinetic Energy of the Vehicle
Behavior	Convert Electricity into Light and other, Thermal, Radiation	Change Kinetic Energy of the Vehicle by using Chemical Energy and producing Emissions (e.g. CO_2, NO_2, PM, NVH, etc.)
Structure	Old fashioned Glass Blub containing a tungsten Wire in vacuum	Internal Combustion Engine driven Powertrain with a manual Transmission

by describing the effects of the behavior of the device: *convert electricity into light.* Fourth, the behavior should be based on the functions of the device: *convert electricity into light and other, thermal, radiation.* Finally, the structure should be defined that it can exhibit the identified behavior: *Old fashioned glass blub containing a tungsten wire in vacuum.* The table 4.3 also shows the five-key-concept applied to the powertrain.

The term function at the five-key-concept can be seen as the desired effect of the behavior. Based on the five-key-concept and other concepts examined, *Vermaas* defined three different meanings of functions as follows:[4]

Functions of devices as ...
1. ... the purposes for which the devices are designed
2. ... the desired effects of the behavior of the devices
3. ... intended behavior of those devices

The purpose of the first meaning concerns a state of affairs that has to be realized with the device, for example, as shown in table 4.4, for the light bulb is *illumination in the room.* The first meaning of functions is to be equated with the goal of the five-key-concepts. Therefore, this functional meaning is goal-oriented. The second meaning

---

[4] Vermaas 2010, p.185.

Table 4.4: Examples of different meanings of functions

	Light Bulb	**Powertrain**
1	Illumination in a Room	Drive the Vehicle
2	Convert Electricity into Light	Change Kinetic Energy of the Vehicle
3	Convert Electricity into Light and other, Thermal, Radiation	Change Kinetic Energy of the Vehicle by using Chemical Energy and producing Emissions (e.g. CO_2, NO_x, PM, NVH, etc.)

discribes the desired effect of a device, and the example may be to *converting electric energy into light.* Functions then still refer to the behavior, but conversation laws are ignored in their description. In the final meaning, a function refers to the behavior of the device, and its description meets physical laws for example *convert electricity into light and other thermal radiation.* The last meaning of functions is to be equated with the behavior of the five-key-concepts.[5]

It is possible with the three archetypical meanings of functions to describe devices/products very goal-oriented, function-oriented, or behavior-oriented. This range facilitates the specification of devices/products if the three different meanings of functions are used and applied in a targeted manner.

Considering this, and applying the three different meanings of functions to the powertrain, they could be as shown in table 4.4. The goal-oriented function for the powertrain could be *drive the vehicle.* The user function described in chapter 2.3.2 and the goal-oriented function, are equally comparable as they both implement the customer's functional expectations. The second meaning of functions describes the desired effect of the behavior in a technical form, which can be derived from the first meaning of functions. This meaning includes only the part of the behavior which can be derived from the customer's functional expectations and not all effects resulting from the continuity. For the powertrain example, this is *change the kinetic energy of the vehicle.* The last meaning of functions, the behavior-oriented description, expand the description of the desired effect of the behavior to the description of the whole behavior, which is for the powertrain example *change the kinetic energy by using chemical energy of the vehicle and producing emissions (e.g.* CO_2*,* NO_x*, PM, NVH, etc.).*

In summary, customer expectations can be translated into technical, behavioral descriptions using the functional description language. All three meanings are used in the following conceptual approach. The goal-oriented meaning is used to define customer

---

[5] Vermaas 2010, p.185.

expectations. The function-oriented meaning is used to identify the functions, and the behavior-oriented meaning is used to determine the resultant behavior. All three steps are described according to the rules of the functional description, which means a combination of verb and noun and a solution-neutral formulation (as described in chapter 2.3.1).

4.1.2 Conceptual approach and procedure

This section explains the conceptional approach and procedure to specify a system in the course of the master's thesis.

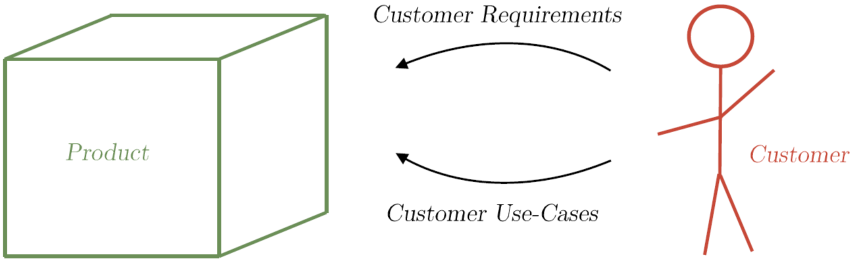


Figure 4.1: Customer-product-relationship

The approach begins with a holistic perspective of the product/system from the customer's point of view, as illustrated in figure 4.1. The first step describes the customer's requirements and use-cases of the product/system. The customer's requirements are the starting point for the technical requirements specification. The customer's use-cases describe how the customer will use the product/system. The customer's point of view is the starting point for the system specification of technical functionalities to achieve customers' expectations. The cube represents the product/system, which should lead to a solution neutral thinking and description of the costumer's requirements and use-cases.

Second, the system is delimited by the definition of system boundaries. It is essential for the following system specification process to define the system boundaries and the relevant system environment. For example can a system be part of a super-system or part of a system of a system as described in chapter 2.2.1. As a following step, the system can be divided into more detailed sub-systems. In the system context, the system model consists of four spaces: requirements, functions, behavior, and structure, as shown in figure 4.2. The four steps are inspired by the four steps of the *Eigner's et al.* model. Moreover, it is inspired by the spaces and levels of the *Münchner Produktkonkretisierungsmodell*. The red space represents the technical requirements for the defined system. The blue space describes the functions and behavior of the system according to the definition of *Vermaas*. The axes - vary, specify, and concretize - are taken from the *Münchner Produktkonkretisierungsmodell* and represent the same basic

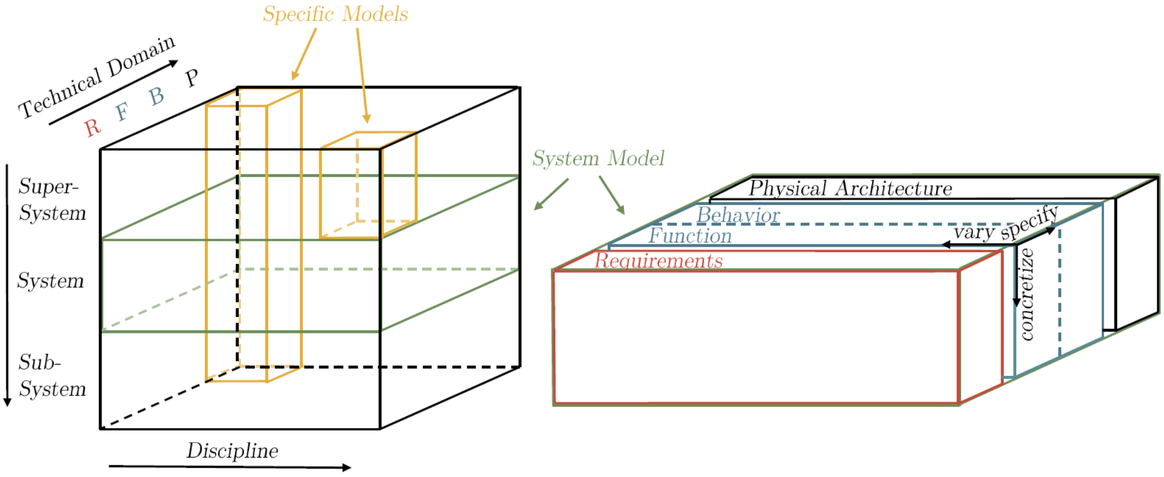


Figure 4.2: Clarification of the context by means of the system model

ideas. The final structure space is the area for the physical topology definition of the product based on the blue and red space.

The procedure follows the explained concept of the four spaces in the system model and is shown in Figure 4.3. The created customer's point of view is the starting point for the procedure. First, customer requirements are translated into technical requirements in consideration of the customer use-cases. The technical requirements specify the product with technical facts based on all stakeholder groups. Second, customer use-cases are translated into functions by considering the technical requirements. Therefore, the meaning changes from the purpose (customer/user function) to the desired effect of behavior. Third, the description of the desired effect of the system's behavior is extended to the description of the system behavior by using the black-box view. Fourth, the second stage concretizes and varies the functions of the previous first stage. This variation of functions is further evaluated and selected based on the technical requirements. The behavior of the system can be described piece by piece from a black-box view to a white-box view under consideration of the functional description. The cycle of concretizing, varying, evaluating, and selecting functions should be repeated as often as necessary. Besides, The behavior of the system is further and further specified. Fifth, the physical topology can be created based on technical requirements, functions, and behavior. The following subsection shows the conceptual approach and procedure applied to a light bulb.

4.1.3 Example: light bulb

This subsection applies the explained conceptual approach and procedure for system specification to a simple *light bulb* case study. The system is specified in the course of a new development of a light bulb. Figure 4.4 illustrates the explained case study.

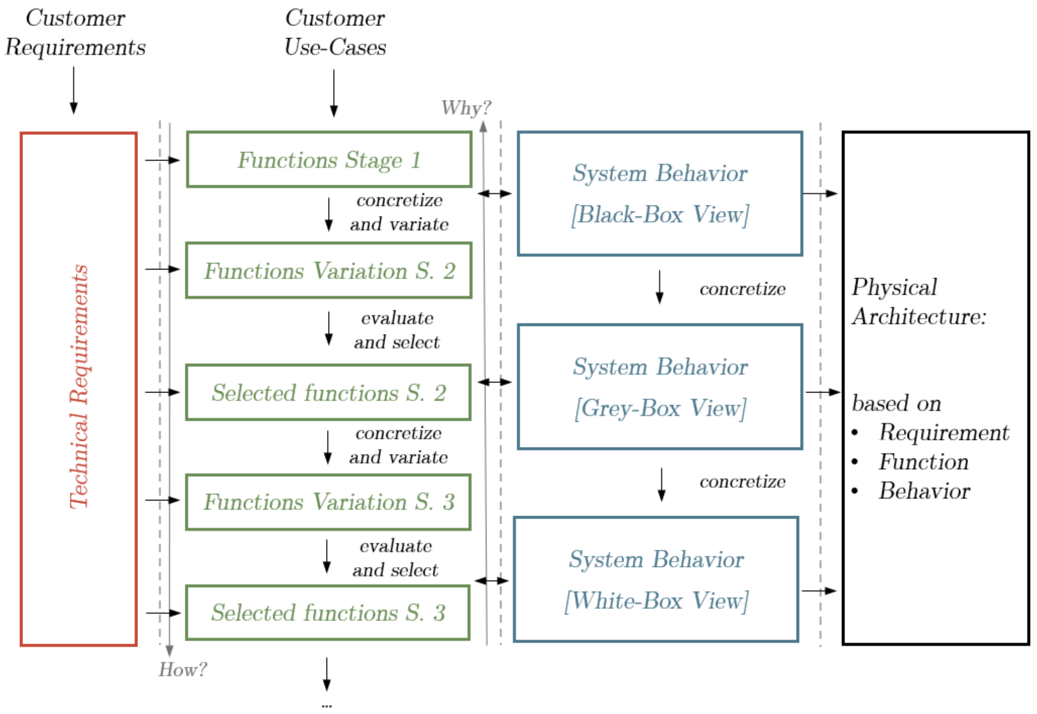


Figure 4.3: Top-Down Procedure

Customer requirements and use-cases

The case study begins with the consideration of customer requirements for a light bulb. The assumed customer requirements for this case study are a luminosity of 1000Lumen, a price lower than 5Euro, and the light bulb should be compatible with a specific lamp socket. The next step specifies the customer use-cases for the light bulb case study. The definition is based on the reflections of the purposes for which the devices are designed. For the light bulb example the use-cases *illuminate the room* and *mount light bulb* are defined.

System boundaries and technical Requirements

The definition of the system boundaries and the technical requirements of the system follows after the description of the customer-product relationship. An essential interface is the socket of the lamp because this interface is the primary connection to lamp. The system light bulb is part of the super system lamp.

To specify the technical requirements, all stakeholder requirements are essential in addition to the customer requirements. The customer requirements are the starting point for the definition of the requirements. The technical requirements consists of the luminosity of 1000Lumen, the production cost of 2Euro, the standardized E27 socket, the durability of five years based on average usage, and the maximum energy consumption should be 75kWh.

Stage 1

The definition of the product functions starts with the translation of the customer use-cases, which represent the purpose of the product, into the desired effects of the bahavior. The function of stage one are defined in consideration of the technical requirements. The use-case of *illumination in a room* can be translated into the function *emit light.* The second use of *mount light bulb* can be translated into *provide interface.* After the definition of the functions, the black-box view is used to illustrate the systems behavior. The black-box includes the described functions and the related input and output ports. For example, the function *emit light* has the input port energy and the output port light, and the function *provide interface* the port mount.

Stage 2

The following second stage specifies the product one step further. For this step, it is necessary to ask how the function of the first level should be fulfilled. A variation of functions in stage 2 are created by concretizing the previous stage 1 functions. For example, can the function *emit light* be achieved by the function *convert electric energy to light*, *convert chemical energy to light*, and *convert electromagnetic radiation to light.* The next step evaluates the detailed functions. One defined function of stage two has to be selected, to narrow the solution space. The evaluation and selection of the functions have to be based on the technical requirements.

All other functions of stage two are defined according to the same principle. For example, the function *provide interface* can be achieved by the function *make a detachable connection* or *make a undetachable connection.* Besides, the functions of stage two leads to a more detailed and specified systems behavior. This additional step is realized by rethinking the black-box view of phase one via using the functions of stage two.

Stage 3

The process from stage one to stage two can be repeated as often as necessary for a sufficient description. For example, the function *convert electric energy to light* can be achieved by the functions *emit light with electromagnetic gas discharge*, or *heat electric conductor with electric energy*, or *encourage semiconductor crystal with electric energy.* The last shown step of the figure is the translation of the selected function of stage three, *heat electric conductor with electric energy*, into the behavior. As described by *Vermaas*, a function is the desired effect of the behavior. By translating this meaning into the behavior, some additional functions are required. For example, produce the selected function of stage two *heat electric conductor with electric energy*, light and heat. Therefore the system needs a supporting function to *derive thermal energy.*

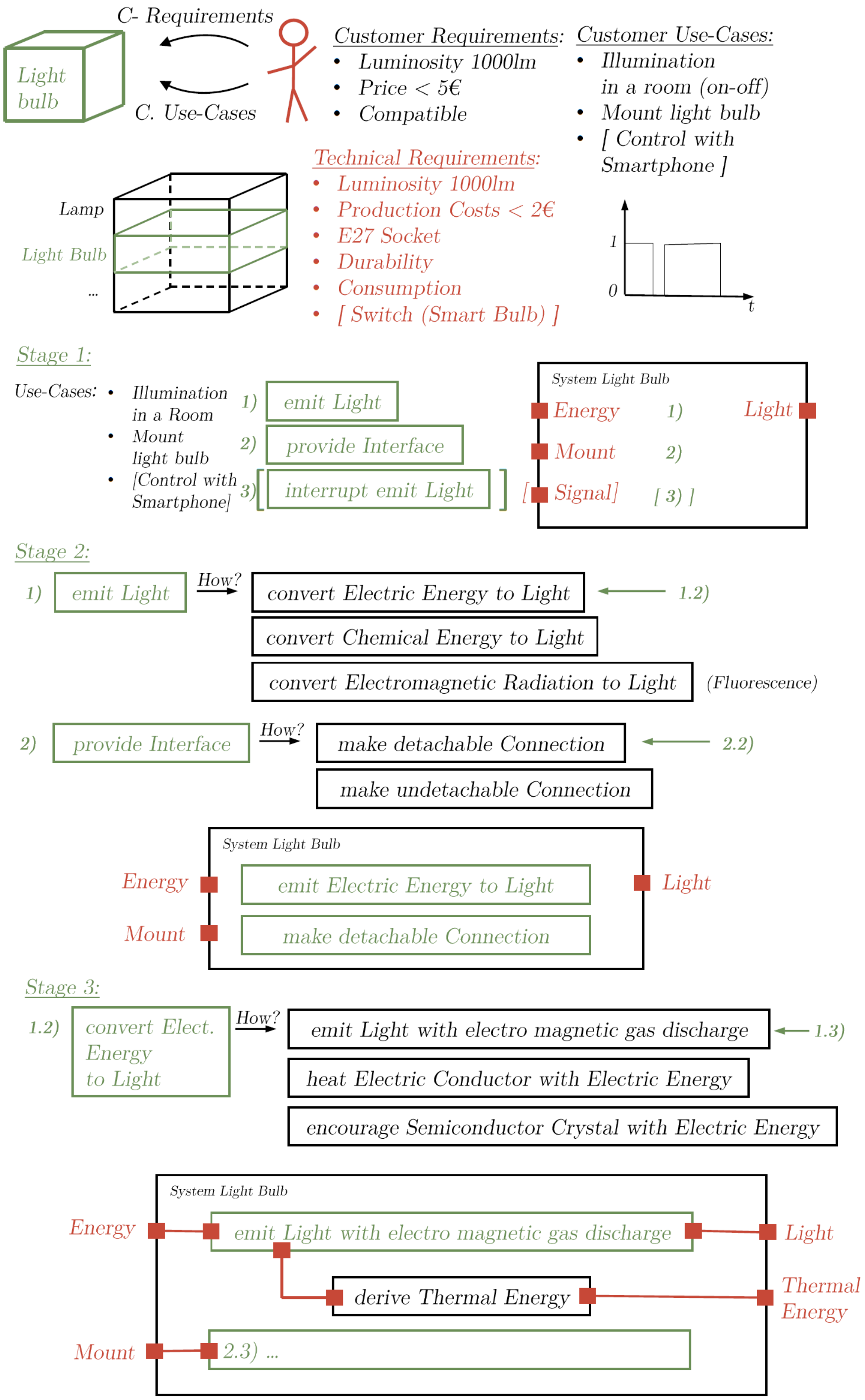


Figure 4.4: Case study example light bulb

The illustrated example also shows how an existing model can be extended. If the requirement for the product changes, for example, if the development team wants a smart light bulb instead of a standard one. Therefore additional goals can be added to the technical requirements. Furthermore, the functions can be efficiently adapted and supplemented. This addition also changes the behavior of the light bulb and requires an additional input signal.

4.2 Case study powertrain

This section deals with the application of the above described conceptual approach and procedure to the case study *powertrain*. This case study should show how a powertrain can be specified using the functions modeling approach. Figure 4.5 illustrates the case study.

Customer requirements and use-cases

In the beginning of the applied approach, customer requirements and customer use-cases are determined. The powertrain has to fulfill, the following assumed *customer requirements*:

- **Power:** 110kW
- **NVH:** smoothly and quiet
- **Torque:** 350Nm
- **Range:** 500km
- **Vehicle Class:** small car

In addition to the requirements, the use-cases describe the use of the powertrain concerning customer expectations. For this case study three use-cases are considered *drive the vehicle*, *stand still*, or *charging the vehicle*. The use-case *drive the vehicle* can be divided into *breaking* and *accelerating*.

System boundaries and technical requirements

The system boundaries are defined in chapter 3.1.1. In relation to the defined system boundaries the illustrated cube in figure 4.5 shows the system model (powertrain) and the according level structure. The vehicle is a super system from the observed powertrain system. Furthermore, the powertrain can be divided into sub-systems, for example, engine, transmission, etc.

The technical description of the system starts with the definition of the technical requirements of the system. The technical requirements are derived from the customer requirements and beyond that it included all stakeholder requirements. For the case study the technical requirements are described below:

- **Power:** 110kW
- **NVH** < 80dB
- **Torque:** 350Nm
- **range:** 500km
- **Emission:** Euro 6dTemp
- **production costs:** 4000Euro
- **packaging:** compact car

Stage 1

The definition of the functionalities starts with the translation of the customer use-cases into the desired effects of the behavior of stage one. The main use-cases *accelerating* and *Breaking* can be translated into the functions *increasing kinetic energy of the vehicle* and *reduce kinetic energy of the vehicle.* These two solution-neutral functions can be joined together to the function *change kinetic energy of the vehicle.* The functions for the description of the use-cases *stand still* and *energy charging* is *secure position of the vehicle* and *charging energy.* As described before, a black-box view can be created after the definition of the functions, which includes the description of the behavior, and the related input/output ports. The related ports for the function *change kinetic energy of the vehicle* are the input port kinetic energy and the output port kinetic energy. The function *charging energy* has the input port energy. Additionally, the whole system powertrain has the communication port signal.

Stage 2

The second stage of the system specification process describes the system functionalities in more detail. Therefore, the functions of stage one are described more precisely by answering the question, *how* the functions of stage two can fulfill the functions of stage one narrowly. Compared to the example light bulb, it is not possible to describe one function of stage one with just on detailed function in stage two. The first function *increasing kinetic energy of the vehicle* shows that the detailed functional description in stage two needs more than one function to describe the previous function narrowly. The chosen functions are *store energy*, *transform energy*, and *distribute energy.* The illustration of the behavior in the middle of the figure shows the connections between the functions.

The previous function *reduce the kinetic energy of the vehicle* is concretized to the variety of two detailed functions in stage two. For example, is this function fulfilled by the sum of the functions *transport - transform - and store energy* or only by the function *transform kinetic energy into thermal energy*. After the concretizing of the functional description and the variation of functions as described in the conceptual approach follows the evaluation and selection. This step is based on the technical requirements. For the case study, the function *transform kinetic energy into thermal energy* is chosen and further illustrated in the behavior model. The ports of the model are an input port kinetic energy and an output port kinetic energy and thermal energy.

Possible variations of the function *secure Position of the Vehicle* are *lock wheels* or *set torque against movement*. The function *changing energy*, is fulfilled by two functions *charge storage system* and *store energy*.

The case study powertrain describes the system's behavior initially for every function separately in comparison to the light bulb example. In a second step, these system behavior descriptions of each previous described function of stage one have to be merged into one system behavior model. The creation of sub-systems simplifies the merge of the individual system behavior models. A sub-system is a summarized block of functions with similarities. The creation of sub-systems makes it possible to merge different function-specific behavior descriptions into one system description. With this approach, solution spaces can be demarcated and further examined at the next level. Four various sub-systems are created: *store energy system*, *transformation system*, *transportation and distribution system*, and *brake system*. These sub-systems summarize the defined functions of level two. The sub-systems are connected by the ports which are previously elaborated in the individual representations.

The explained procedure from stage one to stage two may be repeated as often as necessary until the function and behavior model is sufficient enough to define the structure of the physical powertrain topology based on the developed model.

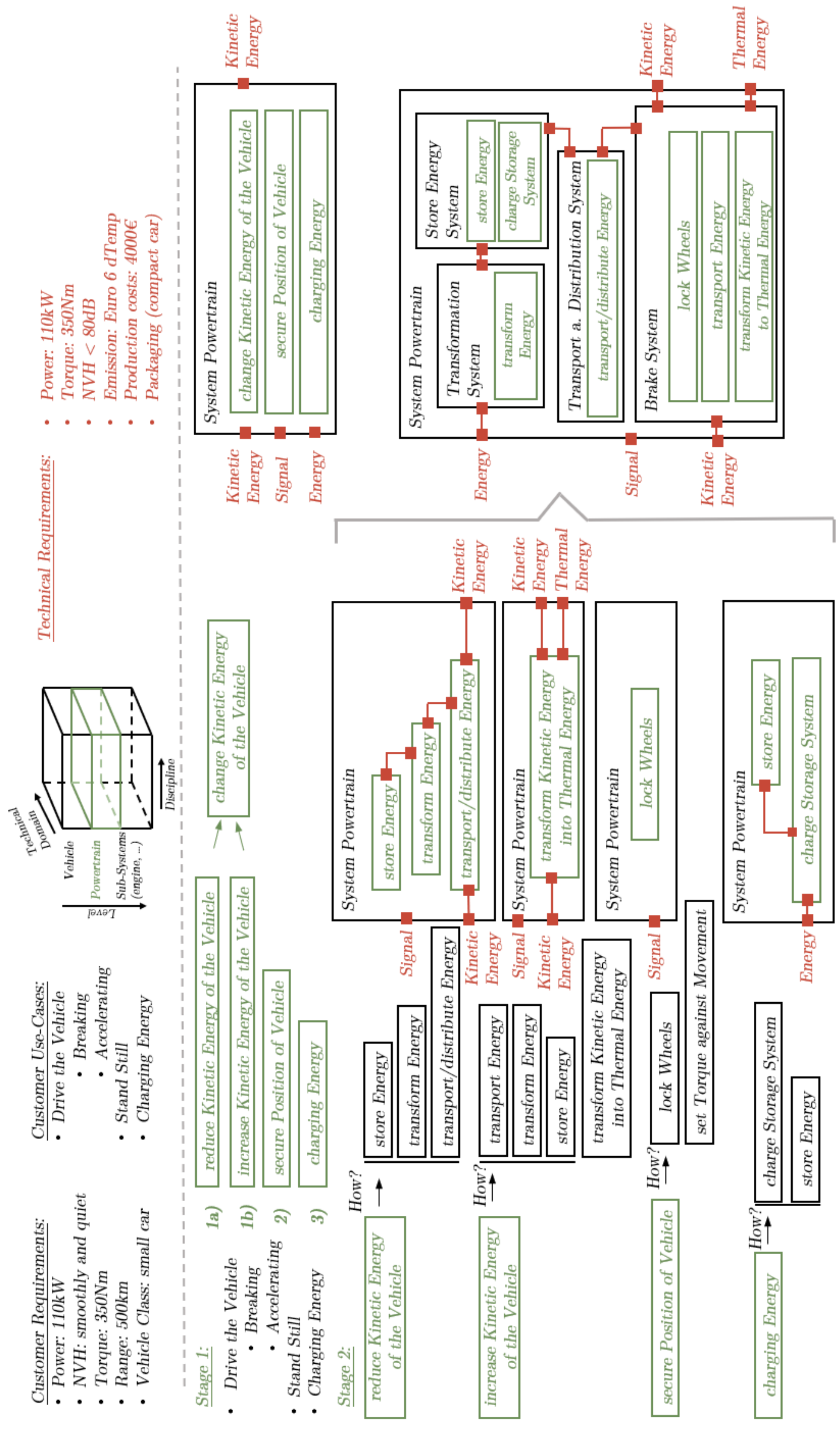


Figure 4.5: Case study powertrain

5 Discussion

This chapter discusses the functional modeling approach described in chapters 3 and chapter 4. First, the advantages and disadvantages of the functional modeling approach are discussed separately in conjunction with the individual chapter. The final section of this chapter summarizes the strengths and weaknesses of functional modeling based on the discussed advantages and disadvantages. Finally, the research questions of chapter 1 are discussed based on the following discussion.

5.1 Discussion of functional modeling according to the analysis of the powertrain System Architectures

This section discusses the advantages and disadvantages of the functional modeling approach in association with the bottom-up analysis. The advantages and disadvantages are formulated in general statements. Subsequently, the statements are discussed based on the previous chapters and supplemented by shown examples.

5.1.1 Advantages

The result of the bottom-up approach offers a disruptive way of thinking by translating specific physical architectures into a generic functional model.

This advantage is illustrated in Figure 2.14. The bottom-up approach starts with physical architecture topologies. These are analysed according to their logical topology and their functionalities in a generic way. By following this approach systems are described from a higher level of detail (specific physical architectures) to a lower level of detail (generic logical models and functional models). With the lower level of detail, it is possible to describe different physical architectures on a broader scale. The disruption of the specific solution leads to a generalized description of the system. This is useful to understand which functionalities a system possesses. Changing the way of thinking from a narrow level of focus to a broader perspective prevents to focus on details in a early development phase.

The result of the bottom-up approach offers a starting point for developing modified, optimized, or extended solutions for new development projects.

This advantage should be seen in combination with the first advantage. As described above, the bottom-up approach leads to a better understanding of the functionalities of systems. This widespread form of description can then be used as a starting point for modification, optimization, or extension of existing solutions. The considerations can take several directions. The functions of the system can be repurposed and compared with the functions of the competitor product. Or, as shown in chapter 4, the system can be specified further based on the defined functions and the new existing requirements.

The analyzed powertrain functionalities of chapter 3 are used in chapter 4, where the analyzed functionalities supported the definition of functionalities through the system specification process. This procedure helps to break up known thinking patterns and to find new concepts.

The bottom-up approach helps to define the system functionalities. It is more convenient to formulate functions based on the bottom-up approach rather than the top down approach.

The bottom-up approach first creates a generic logic model based on specific architectures as described in chapter 3. After the definition of the logical blocks follows the definition of the according functions. This systematic approach and especially the first created logical model simplifies the definition of the system functionalities. For example, the function of the *energy transformation system*, as shown in Figure 3.16, can be determined between the inputs and outputs. For example, the input *electric power* or *fuel* and output *mechanical power* lead to the function *transfer stored energy into kinetic energy*.

In comparison, it is much more challenging to derive the functions from higher-level functions, as shown in chapter 4. In the case of the shown example above, the function *change kinetic energy of the vehicle* can be subdivided into the functions *store energy, transform energy, distribute energy*. Then the inputs and outputs for the defined function are described. This procedure is more demanding than the above one. In summary, the definition of functions based on an existing logic models is more practicable than the specification of general functions, as described in chapter 4.

5.1.2 Disadvantages

Functional modeling according to the bottom-up approach needs additional demand by non-specifiable added value.

The achieved added value of analysing system functionalities is the support of the system definition process for new development projects. Therefore the analysis of functionalities based on specific solutions only makes sense in combination with a following specification process. This results in extended development processes and not directly determinable added value. Therefore, the bottom-up approach has to be an intermediate step of a following system specification and requires a broader context. To generate the added value efficiently and then use it profitably, a precisely defined procedure is required.

The success highly depends on the formulation of the individual functions.

Pohl and *Ropp* describe that a language is inherent and ambiguous. The statements made in this book are aimed at the formulation of requirements. The essential connections must also apply to the wording of the functions. By the described ambiguity, the functions are also interpreted differently. As a result, misunderstandings may arise due to the variety of people involved in the development process. Besides, in the two processes of perception and representation, so-called transformation effects occur, which differ from person to person.

Furthermore, formulated functions describe a solution space. For example, can a lamp be characterized by the function *illuminate space* or *convert electrical energy into the light*. This example already shows that depending on how the function is formulated, different solution spaces are defined. With the first formulation, a lamp could also emit light with fuel combustion, which is undesirable with the second function. In summary, there are two uncertainties: that the natural language is inherent, ambiguous, accordingly interpretable and that the formulation determines the solution space itself. The first point can be better understood by understanding the transfer effects. And the second point must be communicated very clearly so that it can be taken into account in the application.

Building a consistent functional structure model is more complicated by following the bottom-up approach rather than the top-down approach.

The third disadvantage can be explained by the comparison of the top-down and the bottom-up approach.

The top-down approach, as shown in figure 4.5, describe functions more in detail every stage. If a function needs more than one sub-function to describe the function on a detailed level, several sub-functions are related to each other. Afterwards, these relationships can be illustrated for every function of the precious stage. As shown in figure 4.5 the different functions and illustrations of each function are combined to a general representation of the system by categorise the sub-functions to sub-systems.

To describe the explained process of connection from functions between stages seems more complicated bottom-up rather than top-down. Above all, it is more challenging to remain consistent terminology by following the bottom-up instead of the top-down approach. This is the case because the functions of the bottom-up approach are defined separately instead of derived from higher-level functions.

To sum up, the bottom-up approach only makes sense up to a certain degree. The cross-sectional definition of functions limits the bottom-up approach.

5.2 Discussion of functional modeling according to the powertrain system specification

This subchapter discusses the advantages and disadvantages of the functional modeling approach in association with the top-down approach. The advantages and disadvantages are formulated in a general statement. Subsequently, the statements are discussed based on the previous chapters and supplemented by shown examples.

5.2.1 Advantages

To describe the functionalities of the product as part of the development process provides a structured approach from the requirements requisitions to the physical components.

Following the described RFLP approach from the requirements top-down to the physical components leads to a step by step description of how a product can be developed. This means that the development is divided into more steps instead of making one single creative step from the requirements of a product to the physical architecture. Doing a single step contains the risk of making many unknowingly decisions about the product. Defining the product requirements, functionalities, logical topology and finally the physical architecture as a step by step process will simplify the whole task of product development into manageable steps in a structured procedure.

The functional modeling approach used for the system specification leads to more transparent and traceability of the development process.

The figure 4.3 shows the applied conceptual top-down procedure. The conceptual procedure, especially the green functional procedure in figure 4.3, follows the sequence of figure 5.1. This sequence is repeated from piece to piece. It starts with the concretization and variation of the functionalities of the previous step. If a function is defined more specifically, this function leads to a reduction of the solution space. Therefore it is necessary to create different more specific functionalities by varying them in order to cover the solution space of the previous step. Eventually, the variety of the more specific functionalities have to be evaluated and selected in order to define the functionalities of the next step. This sequence requires automatically more decisions than a established system specification process where many decisions are made unconsciously. To sum, the system specification based on functions is an interplay between concretization - variation and evaluation - selection, which leads to a decision-based system specification

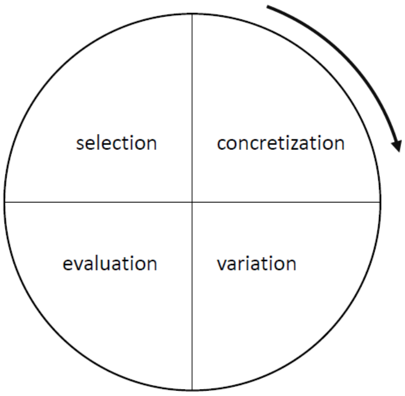


Figure 5.1: The sequence of system specification

with more transparency and traceability of the development process.

The system specification based on functional modeling is easily modifiable.

The requirements, as shown in the conceptual approach (figure 4.3), can be interpreted as a basis for the evaluation and selection decision in each step. If a subsequent project has different requirements than the previous development project, the basis of the decisions has changed. Because of the transparency and traceability of the specification process, the functional model can easily be modified in the case of changed requirements. That means that the subsequent steps of the specification process has to be adapted to the new decision. Because of this adaptability, the specification process is more reusable for subsequent projects.

The system specification based on functional modeling is easily extendable.

This statement can be explained based on figure 4.4, which shows the light bulb example. As an example, it is able to change the requirements from a standard light bulb to a smart device. Therefore, the product must additionally fulfill the requirement of a switchable light bulb (smart). The new requirement requires an additional function in the first stage. The corresponding function is called *interrupt emit light.* The additional functionality also changes the behavior and system boundaries. For example needs the function *interrupt emit light* the port signal. With this port is it possible to connect the light bulb, for example, with a cell phone to control the light bulb. This example shows that additional functions can easily extend to the functional model as an answer to changed requirements. Therefore, it is not necessary to rethink the whole model.

The varied interpretation of functionalities enables different points of views on the development problem.

With the functional modeling language, it is possible to describe different points of view deliberately. This characteristic is consciously used with the various description possibilities offered by *Vermass* to bring the different views together step by step. Additionally, figure 2.10 shows the difference between a user function and a technical function. This effect is used in the description of the use-cases and the following translation to the technical functions of stage one. The functional description of stage one also follows the second description of *Vermaas*, the desired effect of the behavior. The difference between the second and the third meaning of *Vermass* is used to translate the functional description into the behavior of the system. In sum, it is possible to describe with the functional modeling language different points of view, which can be beneficially used to specify the functionalities and further the behavior of the system.

5.2.2 Disadvantages

Functional modeling does not lead to a higher reproducibility of the system specification process.

The outcome depends on many factors of the specification process. The first fact is that the result depends heavily on the engineering team, which is also the case with classic development processes. The second fact is that language is inherent and ambiguous, which is also discussed in the second disadvantage of functional modeling based on chapter 3. The ambiguity leads to different interpretations, and misunderstandings may arise due to the variety of people involved. The third fact is that the outcome depends on the applied method, process, tools, etc. The result can also vary due to many different models and methods which use the functional modeling approach like *METUS Raute*(2.3.5), *Münchner Poduktkonkretisierungsmodell* (2.3.5), modified V-Model (2.2.2), etc.. These factors cause the functional modeling approach to have poor reproducibility. Therefore, it is essential to define general conditions before a specification process starts.

State of research of methodological development approaches supported by functional modeling is imprecise in the detailed description.

At the state of research are many different methodologies developed. This master's thesis shows some of the current approaches. Current approaches are, for example the RFLP approach from *Eigner et al.*,*METUS Raute*(2.3.5), or the *Münchner Poduktkonkretisierungsmodell* (2.3.5). The individual approaches are very similar concerning the functional modeling but very different in the details. Usually, there are no defined workflows according to the different approaches which are of great interest for a targeted application of the shown approaches. The consequence is that there is currently no uniform opinion on the application of functional modeling. Missing workflows lead to unclear processes and misunderstanding of functional modeling in general. The future task will be to compare the different approaches and to develop a standard method and extent workflows for the application of functional modeling.

Functional modeling leads to more transparency in the development process but requires additional effort. The balance between additional effort and more transparency is hard to define, because of unclear functional description scopes.

The system specification process, as shown in chapter 4, has no defined functional description scope and consequently, no clear end of modeling. The question is how far a functional description in detail makes sense for the subsequent definition of the physical system architecture. A comparison to the well known Pareto principle can be made. The Pareto principle describes economic relationships and is known as the 80/20 rule. It says that roughly 80 percent of the effects come from 20 percent of the cause. Much energy must be used to increase the effects even further. This link can also be seen in functional modeling. However, it is not very easy to estimate to what extent a functional description makes sense. Concerning functional modeling, clear description targets can be defined in advance in order to set a targeted description scope.

The functional modeling approach requires a high number of decisions during the specification process, which increases the development time.

The advantage that functional modeling leads to more transparency also has its price. Conscious decision making at every level costs time. It forces to evaluate variations that might not have resulted in consciously. Depending on the freedom of the individual decisions, it can be very time-consuming. As a result, the functional modeling approach leads in combination with the higher number of decisions during the modeling process to slower development processes.

5.3 Strengths and weaknesses of functional modeling

This section merges the separate discussed advantages and disadvantages of the previous sections of strengths and weaknesses from functional modeling. Every strength and weakness is briefly explained in detailed in the following list.

Strengths

- **Solution-neutral description changes thinking patterns**: This strength results from the discussed fact that narrow solution-oriented thinking patterns can be avoided by using the functional modeling approach. This approach changes the path to problems into solution-neutral thinking, which overall has a positive impact on finding innovative solutions.

- **Functional modeling enables function-based development for mechatronic systems**: Functional modeling provides a starting point for further development and contributes to a functional understanding of the product. By concentrating on functions, a component-independent view can be guaranteed.

- **Functional modeling contributes beneficially to a structured development of mechatronic systems based on controlled solution space**: Functional modeling restricts the solution space in a controlled manner step by step, which contributes to structuring. Therefore, functional modeling is the link between the requirements and the physical architecture topology definition of mechatronic systems, e.g. by using the RFLP approach.

- **Functional modeling increases the transparency and traceability of a development process**: The strength of a structured development leads to the strength of transparency and traceability, which in turn facilitates the determination of the architecture.

- **Functional models can be efficiently expanded and modified, resulting in high reusability and durability**: The functional model can be used for subsequent product development projects because it is modifiable and extendable. Therefore, functional models have high reusability and durability.

- **Functional modeling enables different points of view on the development problem**: As shown, the functional description method can be used to describe different points of view on a development problem. This characteristic leads to a holistic interdisciplinary and solution-neutral approach.

Weaknesses

- **The success of functional modeling depends highly on the formulation of the single function**: The strengths of functional modeling are solution neutrality, controlled solution space, and different perspectives on a problem rely very much on the formulation of the individual function. Therefore, success is directly dependent on the formulation of a single function.
- **High effort resulting in more extended development phases with unpredictable added value**: The intermediate step of functional modeling costs time and effort, which extends the development process. The added value is not predictable because functional modeling does not necessarily lead to a different solution.
- **Research of methodology is unprecise**: As previously described, the current state of the functional modeling approach is unprecise according to specific workflows. There is no universally accepted workflow view for functional modeling.
- **Subjective influence of personal factors (e.g. numbers vs. words affinity)**: Functional modeling describes the development problem linguistically rather than mathematically. This approach does not play to the strengths of a mathematical mindset, therefore causing difficulties in applying the functional modeling approach.
- **The functional modeling approach has no defined stop of the description scope and consequently unclear functional modeling process ends**: There is no defined description scope up to where functional modeling makes sense. Not only does the process have no stopping point, there is no guarantee for an added benefit in continuing the process.

5.4 Added value of functional modeling

To answer the research question, *if functional modeling is valuable and useful for overall product development*, value and effort are discussed separately in order to clarify the added value of functional modeling.

The value of functional modeling can be explained based on the characteristics of it. The functional modeling approach enables a solution-neutral description of product development problems. Furthermore, the approach disrupts existing solutions and brakes up existing ways of thinking. Functional modeling enables the focus on function-based development for mechatronic systems. The structured approach based on a controlled solution space leads to transparency and traceability of the development process and therefore offers the possibility to support the physical architecture definition significantly. The functional model can be efficiently modified and adapted, which makes it reusable for subsequent developments. The description type of functions can be used to realize different points of view on a development task, which enables development based on different interest groups.

In order to achieve the desired value, a general functional description of a product is necessary, which involves a great deal of time and effort. One of the main problems is to estimate the effort for functional modeling process. The main reason for this problem is that there is no clear end to functional modeling. Besides, there are currently different models and methods according to the functional modeling approach, but they are very different in detail. Furthermore, there are no workflows that describe how to proceed in detail. These facts lead to an inefficient use of functional modeling. In summary, it is infeasible to estimate the added value for product development by using the functional modeling approach.

However, the functional modeling approach should support the product development of innovative products. If the next level of innovation could be achieved with the support of functional modeling, the added value is given.

6 Conclusion and Outlook

First of all, the current concepts of the functional modeling approach is summarized as a theoretical background of this thesis. Superordinate topics like product development and systems engineering are explained, and finally, the theory of functional modeling is described. In addition to the basics, procedure models such as the modified V-model, the *Münchner Produktkonkretisierungsmodell*, or the *METUS Raute* are explained.

Next, the methodology of functional modeling is analyzed by applying the modified V-model according to *Eigner et al.* to the powertrain system by following the bottom-up approach. Three specific powertrain architectures are considered to identify the generic logical system architecture and the generic functions of the system. The analysis shows that the translation of existing solutions to a generic functional description breaks up existing ways of thinking and thus provides a good starting point for a new development based on the solution-neutral functional understanding of mechatronic systems. It is important to note that the result of analysis depends strongly on the description of the individual functions and that a comprehensive correlation of the functional structure is challenging to determine with the bottom-up approach.

In addition, the functional modeling approach is used to specify systems by following the modified V-model and the top-down approach. A comprehensive conceptual procedure is established in order to describe the system specification process in detail. A case study shows the procedure applied to the powertrain. The results of the system specification process demonstrates that functional modeling contributes to a structured approach and provides more transparency and traceability. Furthermore, functional models are easily modifiable and extensible and allow different perspectives on a problem. Besides, the current methodologies are imprecise in detail and the additional effort for the application of functional modeling is difficult to estimate because of unclear description scopes. Functional modeling extends the development process and added value cannot be estimated.

Finally, the advantages and disadvantages of the functional modeling approach according to the top-down and bottom-up approach are critically discussed. Furthermore, the strengths and weaknesses of the functional modeling approach are summarized. Subsequently, these findings are used to answer the research questions *if functional modeling is valuable and useful for overall product development.* The added value of functional modeling increases the likelihood to make a new step of innovation. Functional modeling has many advantages, as described, which can lead to innovative system development.

In order to be able to use the advantages of functional modeling in a more efficient way in product development processes in the future it is essential to develop a comprehensive concept that leads to the targeted use of functional modeling. A comprehensive concept includes besides the general procedure model also specific workflows in detail.

The following described points would contribute to a comprehensive concept based on the content of this thesis. The functional analysis process according to the bottom-up approach builds up a basic understanding of functions which have a positive impact on the following system specification. For the system specification process based on functional modeling, a precise procedure is necessary to build a function model effectively and efficiently. Besides, for the efficient usage of functional modeling it is also necessary to define how a function has to be defined for every part of the procedure. For this purpose the description of *Vermass* is a useful definition. Also, to work through the different points of view in a suitable sequence helps to develop the model. An efficient use of the approach also includes the definition of an system hierarchy, which has to be described precisely with functions. If a precise procedure is assumed, including workflows and a detailed understanding of how to define functions, functional modeling can provide a significant added value for product development processes.

List of Figures

List of Tables

List of References

Aristoteles (1960). *Nikomachische Ethik*. Buch I u. IV; vgl. Arendt.

Bajzek, M. (2018). “Lecture Notes: Entwicklungsmethodik. Systems Engineering”.

Bajzek, M., J. Fritz, and H. Hick (2019). “Systems Engineering Principles”. In: Hick, H., K. Küpper, and H. Sorger. *Systems Engineering in Automotive Powertrain Developments*. Springer Verlag (in press).

Berghaus, M. (2011). *Luhmann leicht gemacht: Eine Einführung in die Systemtheorie*. Böhlauer Verlags GmbH.

Braess H. H. und Seiffert, U. (2013). *Handbuch Kraftfahrzeugtechnik*. 7th ed. Springer Viewig.

Brown and Blessing (Sept. 28, 2005). “The Relationship between Function and Affordance”. In: *Processing of the ASME 2005 IDETC/CIE Conference*.

Ehrlenspiel, K. and H. Meerkamm (2017). *Integrierte Produktentwicklung. Denkabläufe, Methodeneinsatz, Zusammenarbeit*. Sixth edition. Carl Hanser Verlag München.

Eigner, M. (2014). “Modellbasierte Virtuelle Produktentwicklung”. In: Eigner, M., D. Roubanov, and R. Zafirov. *Modellbasierte virtuelle Produktentwicklung*. Springer-Verlag Berlin Heidelberg.

Einstein, Albert (Dec. 30, 2010). *Start the New Year Like Albert Einstein*. Ed. by Albert Einstein. URL: `https://www.forbes.com/sites/erikaandersen/2011/12/30/start-the-new-year-like-albert-einstein/#3be9318c3e12` (visited on 02/16/2020).

Ellinger, R. and W. Schöffmann (2019). “Challenges of Powertrain System Diversity”. In: Hick, H., K. Küpper, and H. Sorger. *Systems Engineering in Automotive Powertrain Developments*. Springer Verlag (in press).

Feldhusen, J. and K.-H. Karl-Heinrich Grote (2013). *Pahl/Beitz Konstruktionslehre. Methoden und Anwendung erfolgreicher Produktentwicklung*. Eight edition. Springer Vieweg.

Gabler (2013). *Kompakt-Lexikon Wirtschaft*. 11th ed. Springer Gabler.

Haberfellner, R., O. de Weck, E. Fricke, and S. Vössner (2015). *Systems Engineering. Grundlagen und Anwendung*. Thirdteenth edition. orell füssli Verlag.

Härtl, H. (2008). *Implizite Informationen: Sprachliche Ökonomie und interpretative Komplexität von Verben.* Akademie Verlag.

Hick, H., M. Bajzek, and C. Faustmann (2019). *Definiton of a system model for model-based development.*

Hirz, M., W. Dietrich, A. Gfrerrer, and J. Lang (2013). *Integrated Computer-Aided Design in Automotive Development.* Springer Berlin-Heidelberg.

Hirz, Prof. Dr. Mario (2018). "Lecture Notes: Innovative Powertrains Indroduction".

INCOSE (2007). *Systems Engineering Vision 2020.* Document No.: INCOSE-TP-2004-004-02. Version Revision 2.03. Technical Operations. International Council on Systems Engineering (INCOSE). URL: `http://www.ccose.org/media/upload/SEVision2020_20071003_v2_03.pdf` (visited on 02/10/2020).

Isaacson, W. (2011). *Steve Jobs: Die exclusive Biography.* Little Brown London.

ISO15288 (2018). *Systems and software engineering - System life cycle processes. Part 1: Guidelines for life cycle management.* International Organization for Standardization.

Koller, R. and N. Kastrup (1998). *Prinziploesungen zur Konstruktion technischer Produkte.* Springer-Verlag Berlin Heidelberg.

Kossiakoff, A., W. Sweet, S. Seymour, and S. Biemer (2011). "Systems Engineering Principles and Practice". In: *Wiley Series in Systems Engineering and Management.* Ed. by A. Sage. Second edition. John Wiley & Sons.

Lindemann, U. (2005). *Methodische Entwicklung technischer Produkte. Methoden flexibel und situationsgerecht anwenden.* Springer-Verlag Berlin Heidelberg.

Lindemann, U. (2009). *Methodische Entwicklung technischer Produkte. Methoden flexibel und situationsgerecht anwenden.* Third edition. Springer-Verlag Berlin Heidelberg.

Lindemann, U. (2016). *Handbuch Produktentwicklung.* Carl Hanser Verlag München.

Louie, C. (May 22, 2014). *Model Based Systems Engineering Readiness for Complex Product Development.* URL: `http://www.engineeringnewworld.com/?p=240` (visited on 02/14/2020).

Lührig, Tobias (2006). *Risikomanagement in ter Produktentwicklung der deutschen Automobilindustrie: von der Konzeptentwicklung bis zum Produktanlauf.* Shaker-Verlag.

NASA (2016). *Systems Engineering Handbook.* Second edition. National Aeronautics and Space Administration.

Ponn, J. and U. Lindemann (2008). *Konzeptentwicklung und Gestaltung technischer Produkte. Optiomierte Produkte - systematisch von Anforderungen zu Konzepten.* Springer-Verlag Berlin Heidelberg.

Rambo, J. and H. Weber, eds. (2017). *Systems Engineering. Die Klammer in der technischen Entwicklung.* Gesellschaft für Systems Engineering Verlag.

Ropohl, G. (2009). *Allgemeine Technologie. Eine Systemtheorie der Technik.* Third Edition. Universitätsverlag Karlsruhe.

Schömann, S. O. and M. J. Ringlstetter (2011). *Produktentiwcklung in der Automobilindustrie.* Gabler Verlag - Springer Fachmedien Wiesbaden.

Siemens (2020). *Powertain Performance Engineering.* URL: `https://www.plm.automation.siemens.com/global/it/industries/automotive-transportation/powertrain-performance-engineering.html` (visited on 02/17/2020).

Stachowiak, H. (1973). *Allgemeine Modelltheorie.* Springer Wien.

Ulrich, H. and G. Probst (1988). *Anleitung zum ganzheitlichen Denken und Handeln. Ein Brevier für Führungskräfte.* Haupt Verlag.

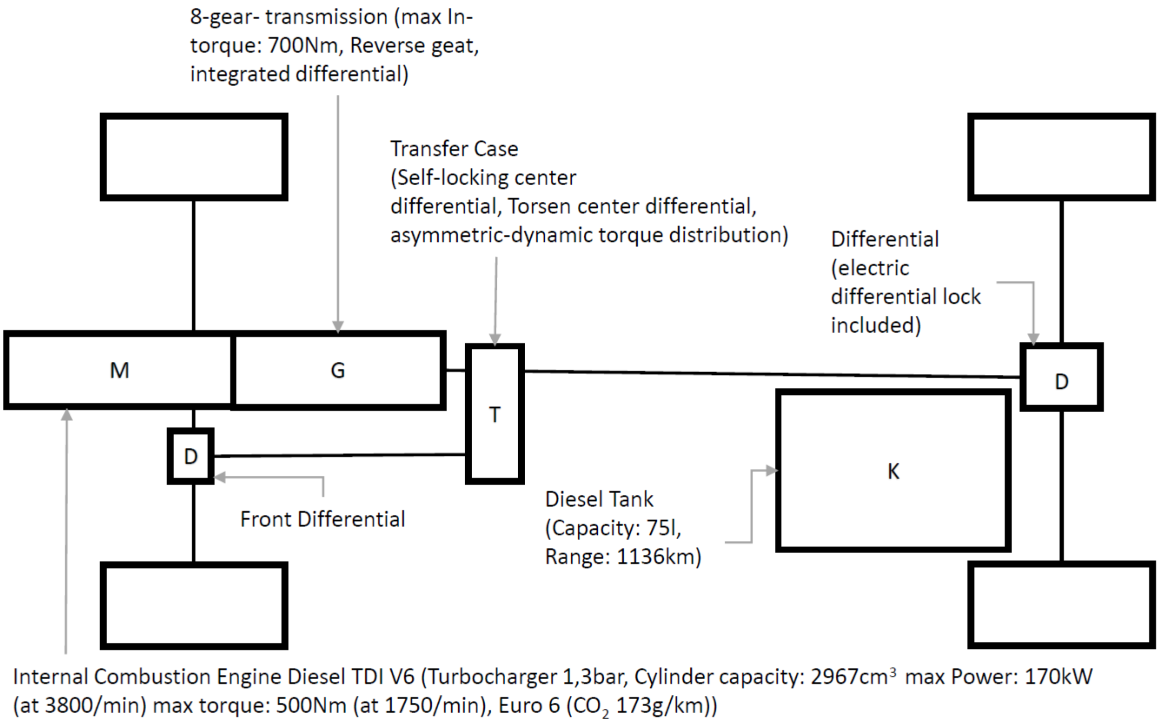
Vermaas, Pieter E. (Apr. 12, 2010). “Technical Functions: Towards accepting different engineering meanings with one overall account”. In: *Processing of TMCE 2010 Symposium.*

Volkswagen (Jan. 20, 2019). *Powertrain Architekturen.* URL: `https://www.vw.com/` (visited on 02/10/2020).

Walden, D., G. Roedler, K. Forsberg, K. Hamelin, and T. Shortell (2015). *INCOSE Systems Engineering Handbook. A Guide for System Life Cycle.* Fourth edition. Wiley New Jersey.

Weilkiens, T. (2014). *Systems Engineering mit SysML/UML. Anforderungen, Analyse, Architektur.* Third edition. dpunkt.verlag Heidelberg.

7 Appendix



8-gear- transmission (max In-torque: 700Nm, Reverse geat, integrated differential)

Transfer Case (Self-locking center differential, Torsen center differential, asymmetric-dynamic torque distribution)

Differential (electric differential lock included)

Front Differential

Diesel Tank (Capacity: 75l, Range: 1136km)

Internal Combustion Engine Diesel TDI V6 (Turbocharger 1,3bar, Cylinder capacity: 2967cm^3 max Power: 170kW (at 3800/min) max torque: 500Nm (at 1750/min), Euro 6 (CO_2 173g/km))

Figure 7.1: Schematic representation of the powertrain *architectures 1*, the VW Touareg

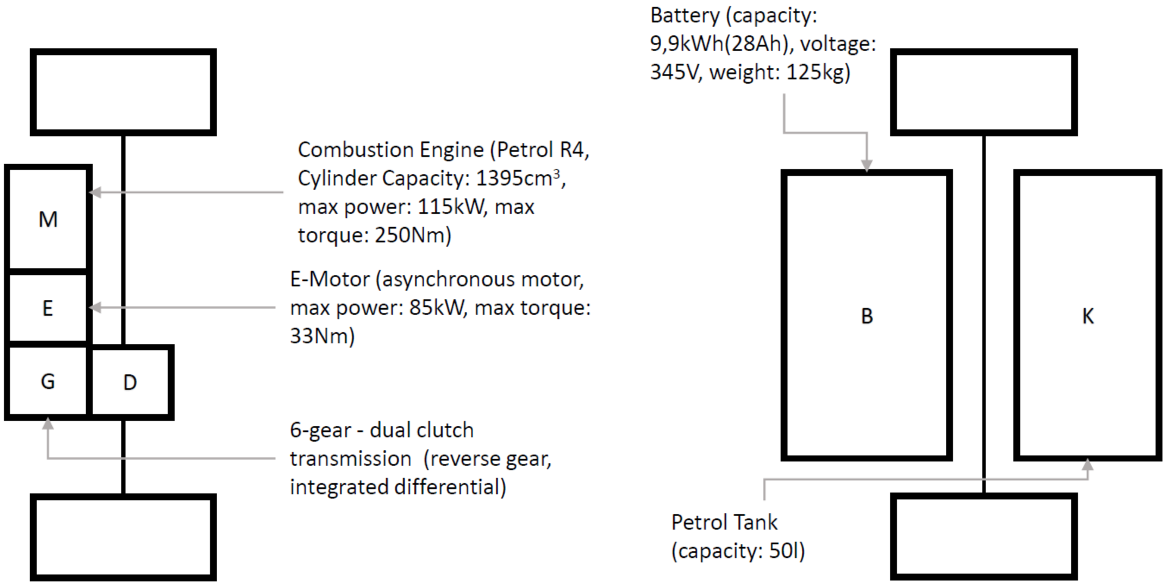


Figure 7.2: Schematic representation of the powertrain *architectures 2*, the VW Passat GTE

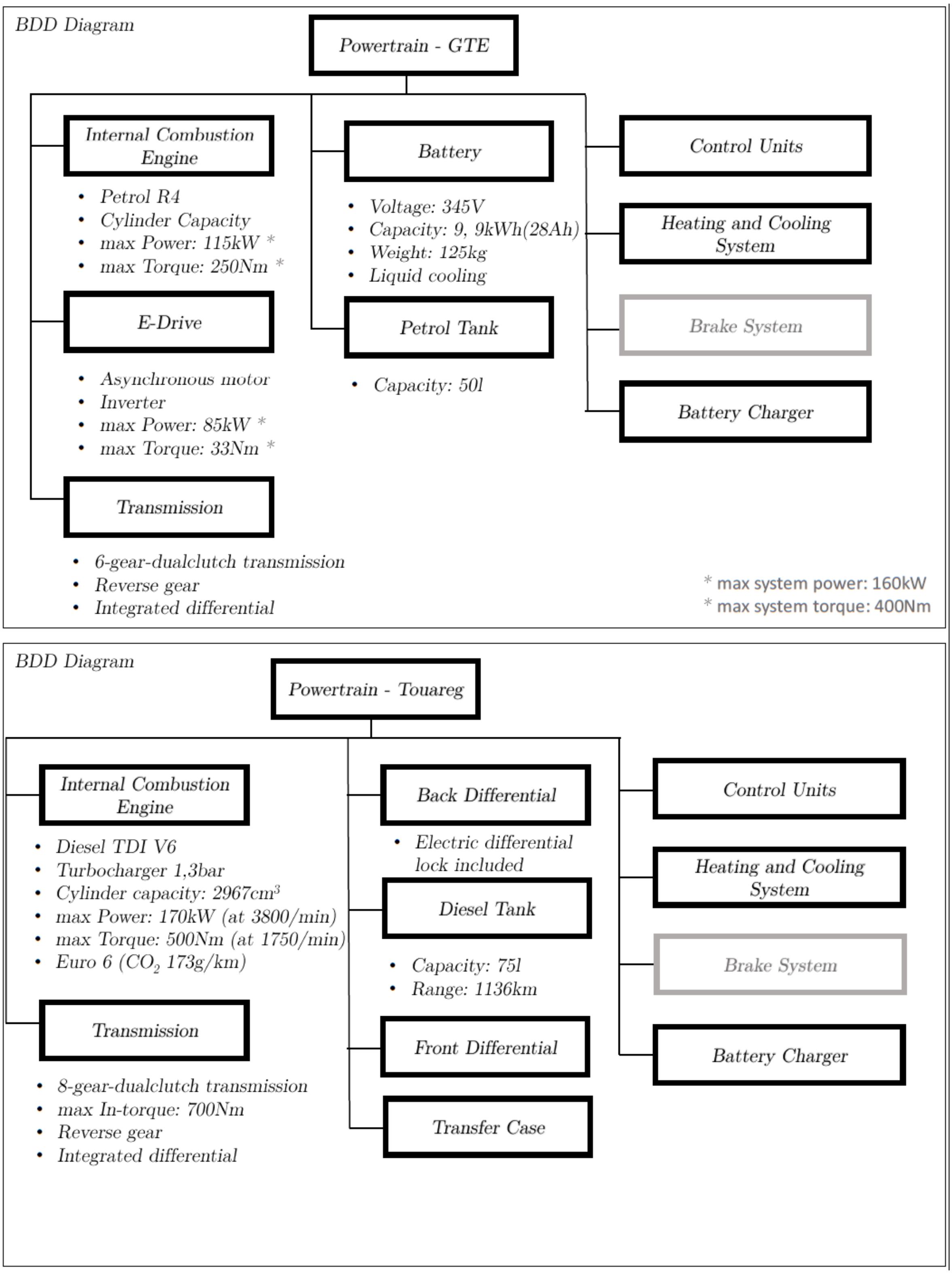


Figure 7.3: BDD of the *Architecture 1 and 2*, the VW Touareg and the VW Passat GTE

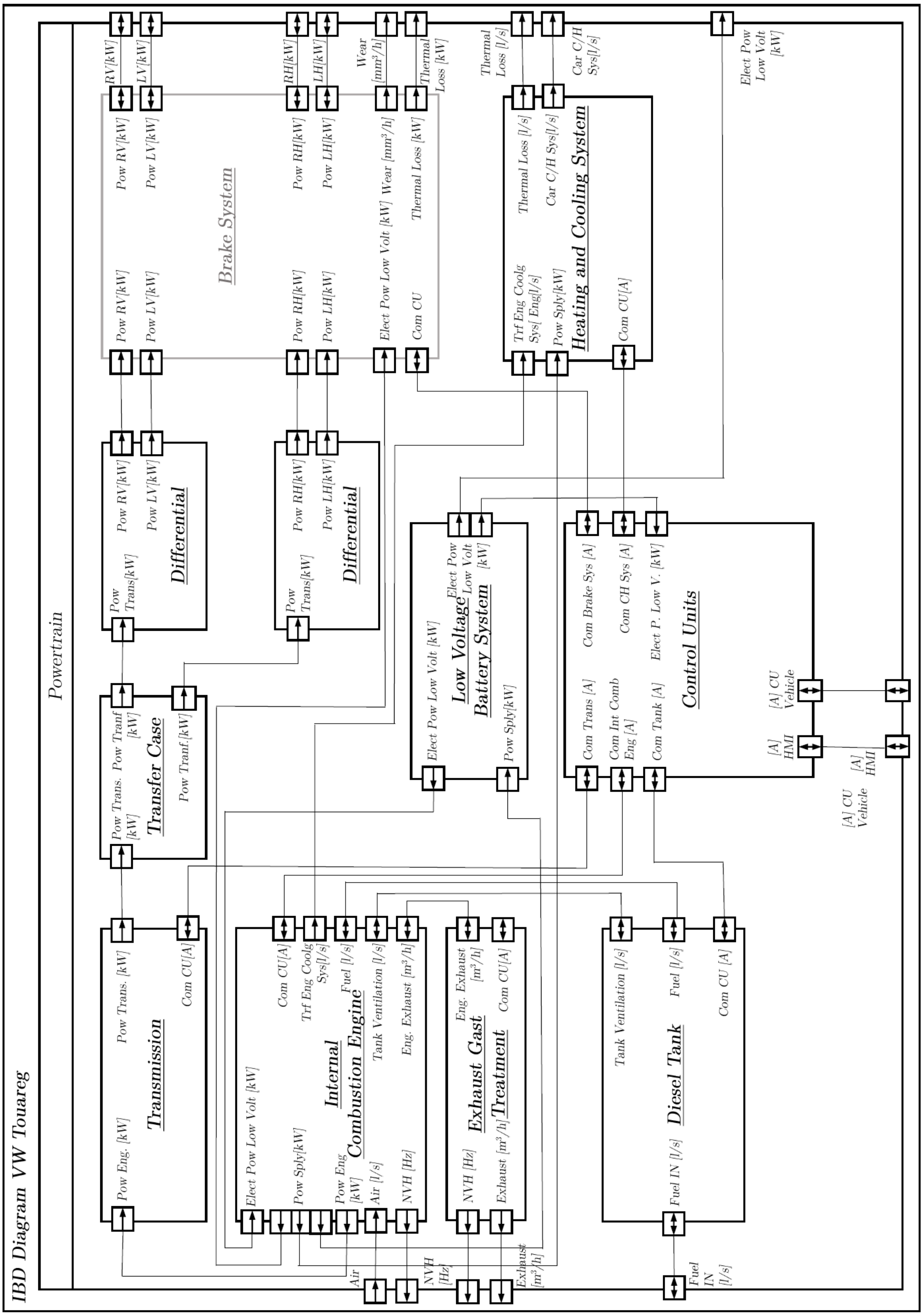


Figure 7.4: IBD of the *Architecture 1*, the VW Touareg

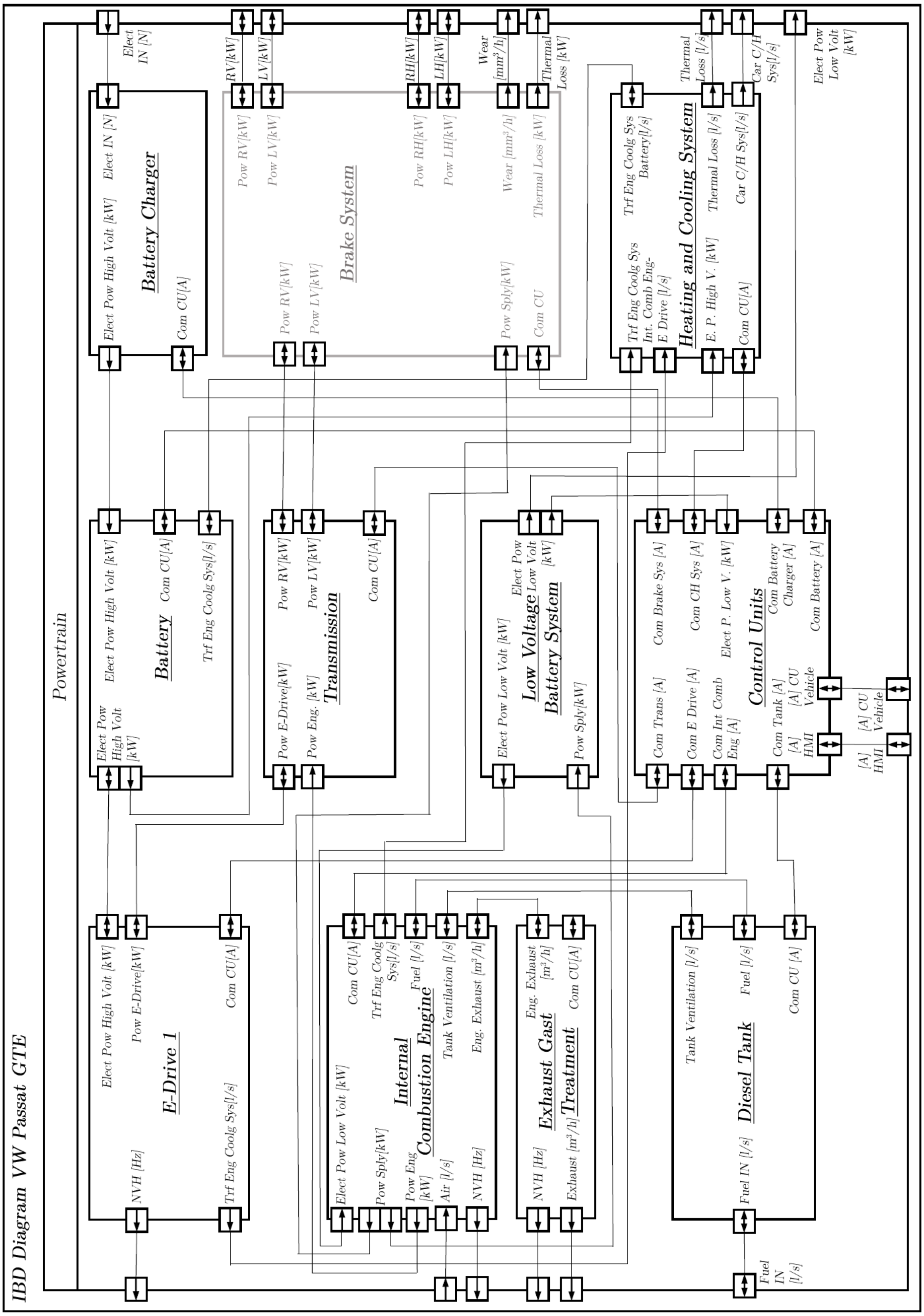


Figure 7.5: IBD of the *Architecture 2*, the VW Passat GTE

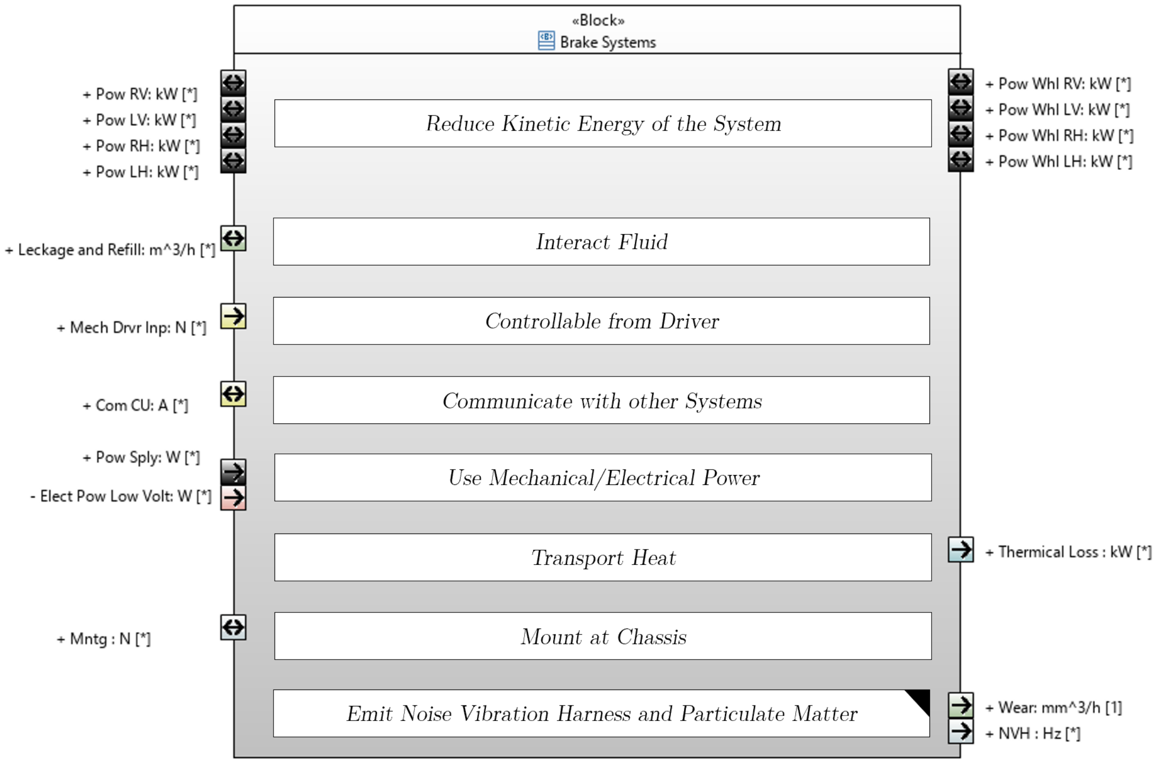


Figure 7.6: Boundaries and functions of the brake system

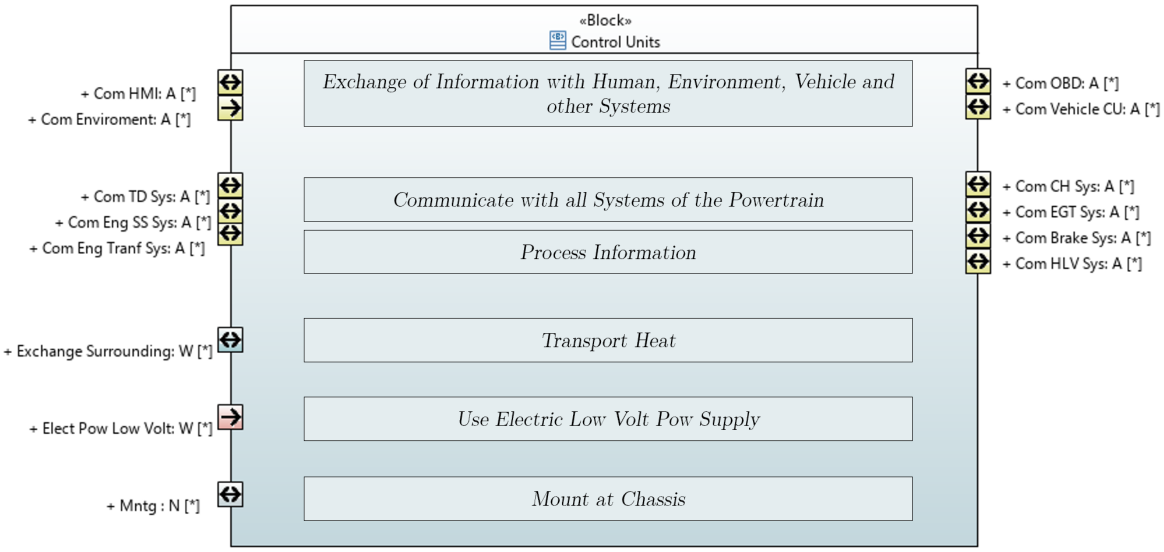


Figure 7.7: Boundaries and functions of the communication unit system

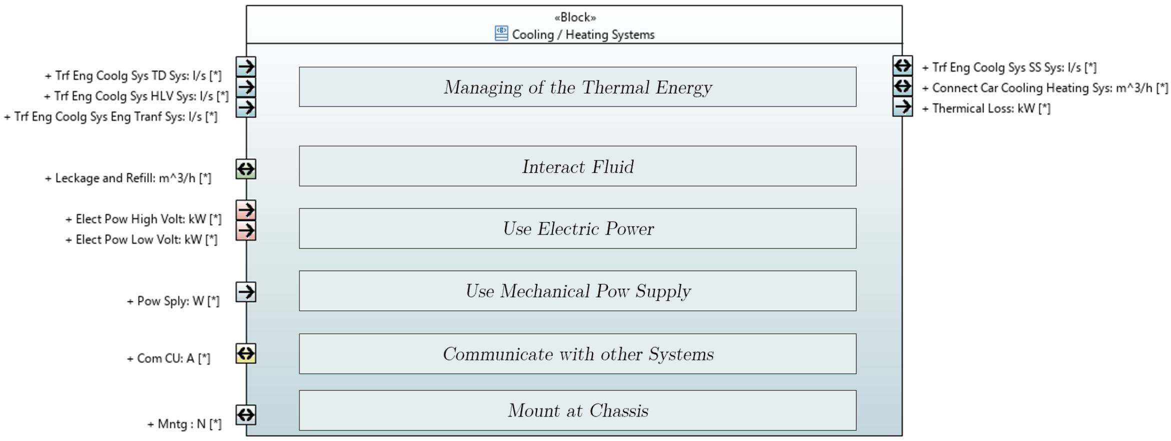


Figure 7.8: Boundaries and functions of the cooling heating system

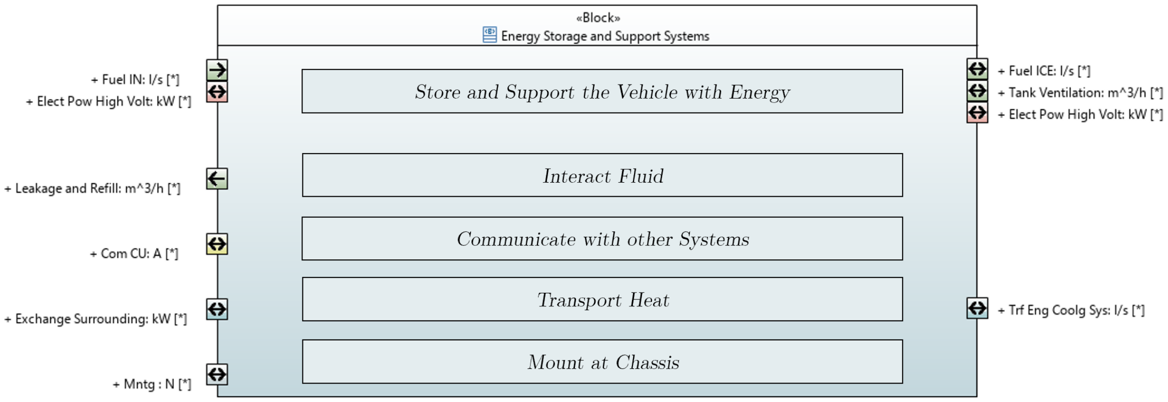


Figure 7.9: Boundaries and functions of the energy storage and support system

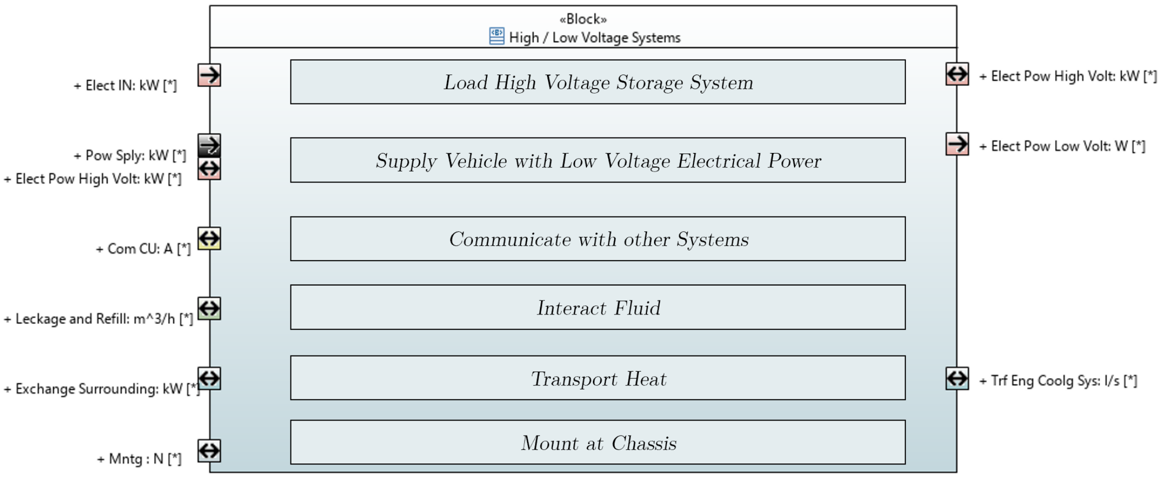


Figure 7.10: Boundaries and functions of the High and low voltage system

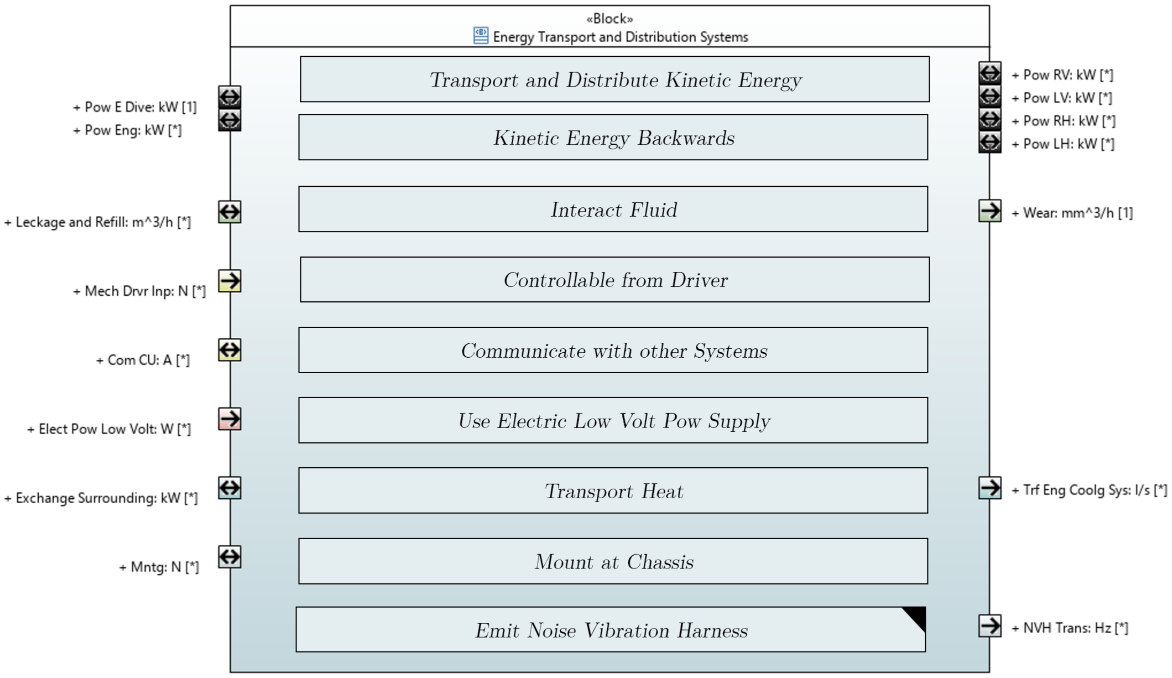


Figure 7.11: Boundaries and functions of the energy transportation and distribution system

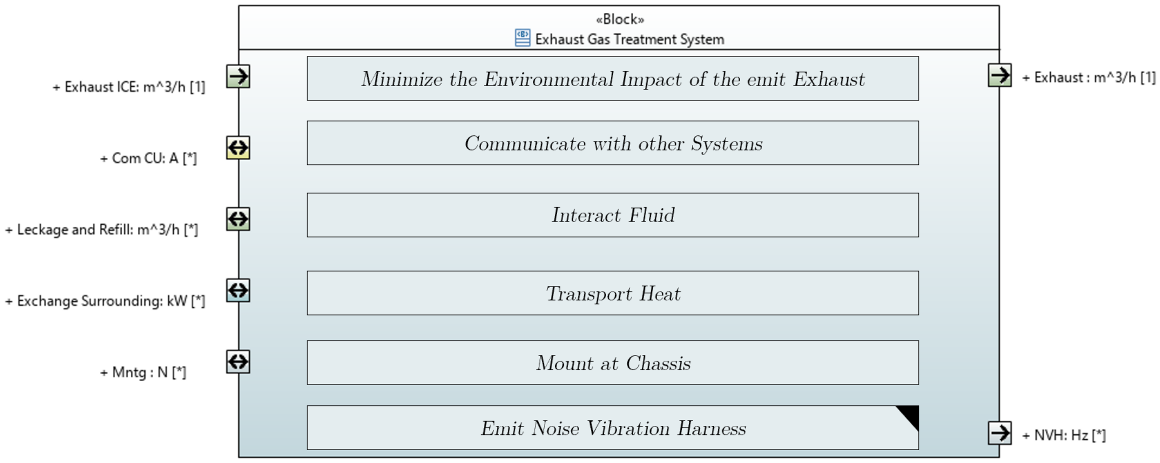


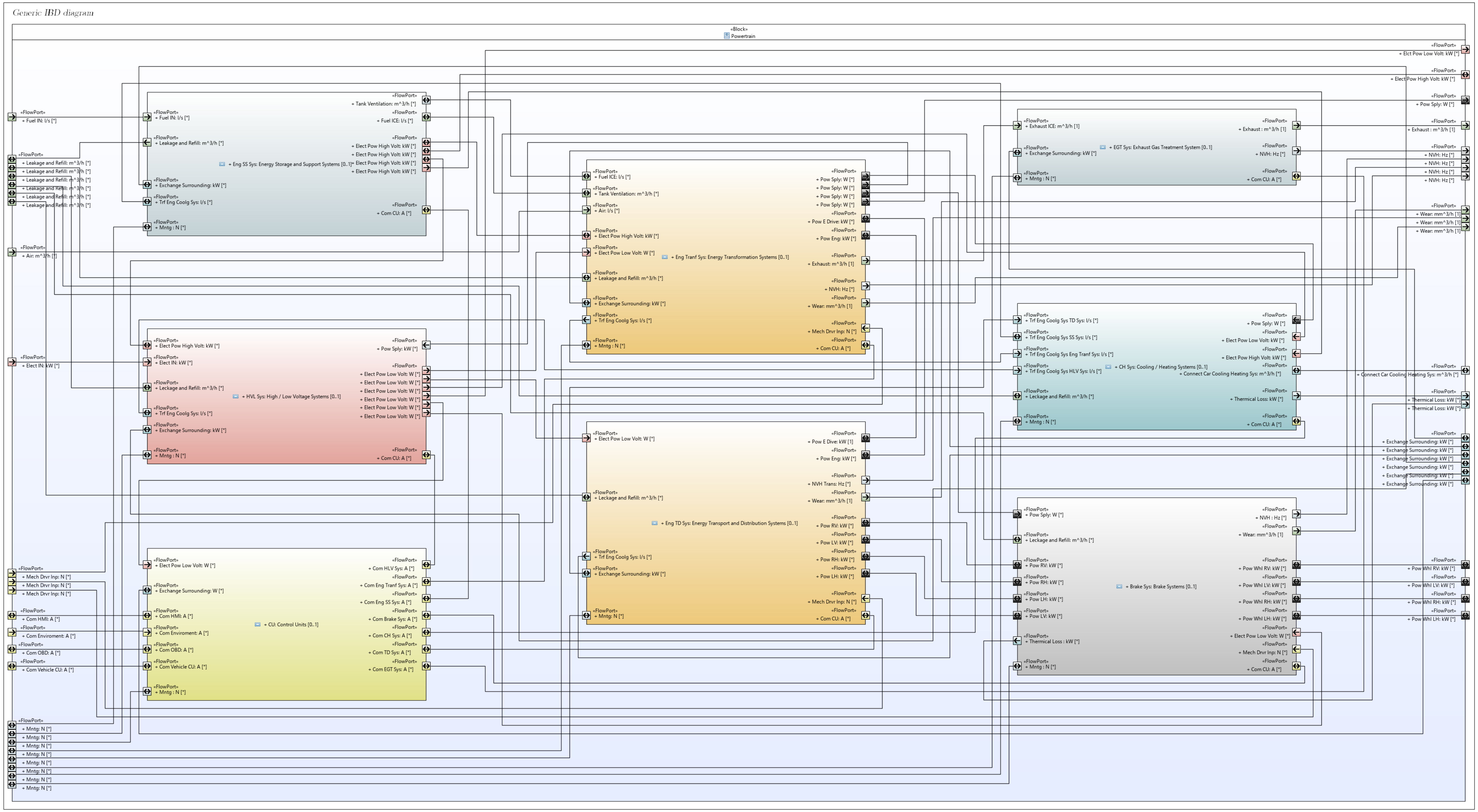
Figure 7.12: Boundaries and functions of the exhaust gas treatment system

Table 7.1: Description of used ports 1/2

Port Name	Unit	Stero Type	Description
Elect Pow Low Volt	W	Electric Energy	Low-voltage electrical energy
Elect Pow High Volt	kW	Electric Energy	High-voltage electrical energy
Elect IN	kW	Electric Energy	Charge input for electrical energy
Fuel ICE	l/s	Matter	Fuel supply for the internal combustion engine
Fuel IN	l/s	Matter	Fuel tank connection
Tank Ventilation	m/h	Matter	tank ventilation
Air	l/s	Matter	Air intake for the internal combustion engine
Leekage and Reefill	m/h	Matter	Losses and replenishment of liquids from systems
Wear	mm/h	Matter	General wear of components
Exhaust ICE	m/h	Matter	Exhaust of internal combustion engine
Exhaust	m/h	Matter	Exhaust after the system exhaust gas treatment
Mntg	N	Mechanic Energy	Mounting the systems

Table 7.2: Description of used ports 2/2

Port Name	Unit	Stero Type	Description
Pow E Drive	kW	Mechanic Energy	mechanical energy from the e drive
Pow Eng	kW	Mechanic Energy	mechanical energy from the internal combustion engine
Pow Sply	W	Mechanic Energy	Mechanical erngy suppling diverse systems in the vehicle environment
Pow RV/ LV/RH/LH	kW	Mechanic Energy	Distributed mechanical energy
Pow Whl RV LV/RH/LH	kW	Mechanic Energy	Connection to the wheels for transmission of speed and torque
Trf Eng Coolg Sys	l/s	Thermal Energy	Heat dissipation of excess heat
NVH	Hz	Thermal Energy	Noise Vibration Harshness
Thermical Loss	kW	Thermal Energy	Thermal losses of the powertrain
Exchange Surrounding	kW	Thermal Energy	Heat exchange with the environment across system boundaries
Connect Car Cooling Heating Sys	m/h	Thermal Energy	Excess thermal energy is released to the vehicle
Com OBD	A	Communcation Signal	Communication via onboard diagnostic
Com Vehicle	A	Communcation Signal	Communication with systems in the supersystem vehicle
Com CU	A	Communcation Signal	Communication with other control units in the system
Mech Drvr Inp	N	Communcation Signal	Communication via mechanical driver input



*Large version of **figure 3.14**: generic IBD diagram (Page 47)*