Philipp Kranabitl, Bsc

Interaction of methods in product development

Master´s Thesis

for the acquisition of the academic degree

Diplom-Ingenieur

Master´s degree program: Production Science and Management

submitted to

Graz University of Technology

Assessor: Univ.-Prof. Dipl.-Ing. Dr.techn. Hannes Hick

Institute of Machine Components and Methods of Development

Graz, March 2019

Statutory Declaration

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 marked all material which has been quoted either literally or by content from used sources.

…………………………….. …………………………………………………………

(Date) (Signature)

Preface and Acknowledgement

At this point I would like to say a sincere thanks to all the people who have been helpful, patient and anxious to support me and this thesis. First of all I want to express my gratitude to the *Institute of Machine Components and Methods of Development*. I would like to thank Univ.-Prof. Dipl.-Ing. Dr.techn. Hannes Hick for his guidance and supervision over the whole process of working on this thesis. Special thanks to Dipl. Ing. Matthias Bajzek who has always taken time to answer my questions and has bothered himself to facilitate my actions.

Further, I would like to thank the *Institute of Innovation and Industrial Management* for their input in this thesis. Univ.-Prof. Dipl. Ing. Dr.techn. Christian Ramsauer and Dipl. Ing. Dr.techn. Hans Peter Schnöll have always challenged our ideas and provided great insights from an innovative view on product development.

I have to take this opportunity to thank those people who enabled my studying and without them I would not have had even the chance to apply myself in that, my parents. They allowed me not only to focus absolutely on my education, but they also permitted my studying abroad and kept me grounded in every situation.

Philipp Kranabitl, Graz, March 2019

Abstract

Companies nowadays operate in a dynamic environment. To stay successful, companies constantly have to operate in a state of innovation in terms of products they sell or provide, frequently introducing new products or modifying and improving existing products as needed and desired by the customers. In order to efficient and effective approach that challenge, the targeted use of development methods is becoming increasingly important. The term method in this thesis denotes the description of a rule-based and planned procedure, according to which specific activities have to be carried out to meet certain requirements. The target of the thesis is to give an overview of possible methods applications in the product development process and support the selection of suitable methods for development situations.

In order to meet this target, methods were analyzed and classified (according to their e.g., area of application, level of difficulty, generating output). The focus of the chosen methods is on the development of technical products. Nevertheless, most methods are suitable for various employments. The area of investigated methods reaches from mission analysis over stakeholder needs and design definition to the verification and validation of the product in the product lifecycle. For proper display, the analyzed methods are aligned alongside the V-model, which originally derived from software development, but is generical applicable. One problem of the V-model is the solely sequential display of methods. To solve that issue, the so-called *Münchner Vorgehensmodell* (MVM) is added where all methods are aligned to functions. That alignment offers the opportunity to select methods not depending on their linear or timely arrangement as in the V-model, but on their functional purpose.

Kurzfassung

Unternehmen arbeiten heutzutage in einem dynamischen Umfeld. Um erfolgreich zu sein müssen sich Unternehmen, in Bezug auf die von ihnen angebotenen Produkte, in einem konstanten Innovationsprozess befinden. Neue Produkte werden in den Markt eingeführt und bestehende Produkte nach Bedarf und Wunsch der Kunden modifiziert und verbessert. Um diese Herausforderung effizient und effektiv zu bewältigen, wird der gezielte Einsatz von Entwicklungsmethoden immer wichtiger. Der Begriff Methode in dieser Arbeit bezeichnet die Beschreibung eines regelbasierten und geplanten Verfahrens, nach dem bestimmte Aktivitäten ausgeführt werden müssen, um definierte Anforderungen zu erfüllen. Ziel dieser Arbeit ist es, einen Überblick über mögliche Methodenanwendungen im Produktentwicklungsprozess zu geben und die Auswahl geeigneter Methoden für unterschiedliche Entwicklungssituationen zu unterstützen.

Um dieses Ziel zu erreichen, wurden Methoden analysiert und klassifiziert (z.B. nach ihrem Anwendungsbereich, Schwierigkeitsgrad, Output). Der Fokus der ausgewählten Methoden liegt auf der Entwicklung technischer Produkte. Trotzdem sind viele Methoden für verschiedenste Anwendungen geeignet. Der Bereich der untersuchten Methoden reicht von der Missionsanalyse über die Bedürfnisse der Stakeholder und der Designdefinition bis hin zur Verifikation und Validierung des Produkts im Produktlebenszyklus. Zur Darstellung werden die analysierten Methoden neben dem ursprünglich aus der Softwareentwicklung stammenden, aber generisch anwendbaren V-Modell angeordnet. Ein Problem des V-Modells ist die sequentielle Darstellung von Methoden. Um dieses Problem zu umgehen, wird das sogenannte Münchner Vorgehensmodell (MVM) vorgestellt, bei dem alle Methoden nach Funktionen eingeteilt sind. Diese Zuteilung bietet die Möglichkeit, Methoden nicht nach ihrer linearen oder zeitlichen Anordnung wie im V-Modell, sondern nach ihrem funktionalen Zweck auszuwählen.

Table of Content

Abbreviations

CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
FEM	Finite element Method
FMEA	Failure Mode and Effect Analysis
FTA	Fault Tree Analysis
FTE	Full Time Equivalents
HiL	Hardware in the Loop
IIM	Institute of Innovation and Industrial Management
IME	Institute of Machine Components and Methods of Development
MVM	Münchner Vorgehensmodell
NPD	New Product Development
OEM	Original Equipment Manufacturer
PMTE	Process/ Method/ Tool/ Environment
SE	Systems Engineering
SiL	Software in the Loop
V&V	Verification and Validation

1 Introduction

"Methods are an integral part in the everyday life of an engineer, since it is only possible to develop target-oriented products if methods for design, recalculation, simulation, planning and conducting experiments are used."[1]

This citation has more validity than ever before. In times of globalization and exceedingly dynamic processes in industries, the need for a structured, target-oriented product development process supported by methods is eminent. This thesis intends to improve the effectiveness and efficiency of product development processes by a structured application of methods. It is going to be shown what kind of methods, depending on the development situation, are available and applicable.

1.1 Background

Successful products are an important requirement for a prosperous economy. Sufficient demand on the customer side is just as important as the economic value performance of the provider. The diversity of products sold in different markets range from services over natural products to technical products. This thesis focuses on the engineering sector. These are usually mechatronic products in which elements of mechanical engineering, electrical engineering and computer science work together intelligently. There are many factors that influence product development, such as the market, which demands products with specific properties within a tight quality, time and cost framework. Other factors may be the resources available, the used technologies, the legal framework as well as the employees.[2]

1.2 Motivation

Research facilities, such as *The American Product Development and Management Association*, have stated in their innovation surveys that a structured use of methods is an important influencing factor in product development. Despite this fact, existing research shows, that only a few companies use structured methods in their

[1] Lindemann 2009b. p1
[2] Lindemann 2009b. p7

development process. One of the reasons for this is that neither in literature nor in corporate practice a standardized approach exists. This results in a mostly very arbitrary and strongly person-dependent use of the methods. The nature and extent of the use of methods are therefore very different within the companies.[3]

1.3 Approach

After literature research, methods which support the product development process are selected and analyzed. In the next step the chosen methods are decomposed into elementary criteria (e.g., output information, number of participants). The elementary criteria describe certain aspects of methods (e.g., input, output, number of participants) to support the comparison and selection for certain development situations. In addition to the elementary criteria a model to graphicly display the methods is chosen. Subsequent, after analyzing multiple models, the V-model is selected due to its combination of the top-down and bottom-up approach, and its empathy on verification and validation. One problem of the V-model, as well as for other procedure models, is that the methods are only displayed in a sequential order. To solve that issue, the *Münchner Vorgehensmodell* is added to provide a function-oriented method selection approach.

[3] Graner 2013 p55

2 Product Development

In a static world without changes there would be no need to change business operations and methods or to realize what has changed and what works well. However, companies operate in dynamic environments, not stable ones, and both external competition and internal environments evolve over time. In response, processes have to be continuously adapted to enable those enterprises to remain competitive and profitable through the changing conditions.[4]

Therefore, successful companies constantly operate in a state of innovation in terms of products they provide, frequently introducing new products or modifying and improving existing products as needed and desired by the customers. The process of conceptualizing a product and designing, verifying, producing, and selling it is known by a generalized and comprehensive process called product development.[5] In this thesis the focus is on conceptualizing, designing and verifying part of the product development process.

2.1 Development and state of the art in product development

The focus of the thesis is on technical systems[6] and how to successfully choose adequate methods for specific development situations. In this subsection the evolution of the product development process, in which the methods are used, is discussed.

Once, industrial product development could be seen as a domain of designers and engineers who worked in different development phases. An exemplary development from back then looked like that: The product development receives the input as requirements of market research and sales. In the next step the development result is designed. The output is in form of design documents together with the necessary protection of the product characteristics in tests. In the last step it is passed on to production. This simple division of tasks in a sequential process with clear boundaries is not valid anymore today. The most important differences are:[7]

[4] Cooper 2005. p5
[5] Mital et al. 2014. p21
[6] Czichos 2015. p11
[7] Lindemann 2016. p3

- Due to legal and environmental reasons the subject of product development today includes all the phases of the product lifecycle. The product is thus developed in a holistic view.
- The phases of product development run no longer predominantly sequentially for a particular product, but at least partially simultaneously, to accelerate the development process and shorten the time to market.
- Product development thus becomes an interdisciplinary collaboration in which professionals of various disciplines and product phases communicate and work together.
- Products are increasingly not just purely physical products or hardware anymore. Mechatronics and embedded systems characterize many products. An ever-larger part of the development effort and added value is software.

There are different approaches to execute the product development process and the associated methods. The proposed methods don’t demand a specific development philosophy. It is to be left to the user which philosophy to apply. Common development philosophies are sequential, iterative, recursive, incremental, lean, new product development, or agile, etc.

In the next subsections the two most relevant philosophies for this thesis (new and agile product development) are discussed. In addition, chapter 2.1.3 discusses systems engineering, which is an approach for structured and successful development of complex systems.

2.1.1 New product development

In a dynamic economy, developing new and improving existing products is essential for a company´s survival. A number of studies have indicated that companies rely on new products to generate profits and that trend is going to continue, to a greater extend, in the future.[8]

The main topic of *New Product Development* (NPD) is the design phase with a strong focus on customer empowerment. A lot of methods described later, support the design

[8] Cooper 2005. p6

phase and customer empowerment. Therefore, this chapter should give a brief overview why these topics are essential in a successful new product development process.

A concept which is closely associated with NPD is *Design Thinking*. The term design thinking simply means, that one is approaching problems, and their solutions, as a designer would. Designers, whether in the arts or industry, tend to explore and solve problems through iteration. They quickly generate possible solutions, develop simple prototypes, and then iterate on these initial solutions, informed by significant external feedback, toward a final solution.[9]

The traditional view of new product development, in which companies are exclusively responsible for coming up with new product ideas and for deciding which products should ultimately be marketed, is increasingly being challenged. In particular, many have advocated the idea of democratizing innovation by empowering customers to take a much more active role in corporate new product development. This has become possible because the internet nowadays allows companies to build strong online communities through which they can listen to and integrate thousands of customers from all over the world. Extended research from *Fuchs and Schreier* has provided strong arguments that indicate that customer empowerment in NPD enables companies to develop better products and at the same time to reduce costs and risks if customers in a given domain are willing and able to deliver valuable input. Customer empowerment not only affects the company's internal NPD processes as reflected in the products that are ultimately marketed. It might also affect the way companies are perceived in the marketplace (by customers who observe that companies foster customer empowerment in NPD).[10]

Fuchs and Schreier propose that it would be useful to think of customer empowerment in NPD in terms of two basic dimensions (Figure 2-1):[11]

- Customer empowerment to create ideas for new product designs
- Customer empowerment to select the product designs to be pursued

[9] Griffin et al. 2015. p2
[10] Fuchs und Schreier 2011. p3
[11] Fuchs und Schreier 2011. p16

Therefore, customers may be empowered to submit ideas for new products (empowerment to create) or to vote on which products should ultimately be marketed (empowerment to select).

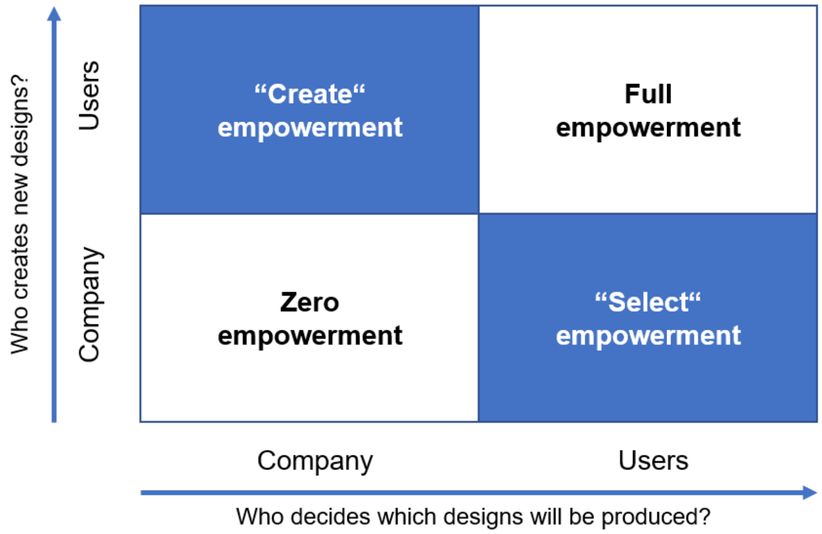


Figure 2-1: Customer empowerment strategies in NPD[12]

Despite all the positive aspects of customer participation on the new product development process there is also a negative side. *Morgan and Obal* released a study in which they explain a participation paradox. They say that, while customer participation in NPD may potentially help product performance, it could also lead to the development of products that are overly radical and are too difficult for potential customers to understand. To resolve this paradox, *Morgan and Obal* argue that companies with higher levels of expertise will be able to rein in the negative aspects of extreme product newness to create products that will be in high demand by the marketplace.[13]

In conclusion, there are strong arguments indicating that customer empowerment in NPD enables companies to develop better products and at the same time to reduce costs and risks if customers in a given domain are willing and able to deliver valuable input.[14]

[12] Fuchs und Schreier 2011. p18
[13] Morgan and Obal 2016. p8
[14] Fuchs und Schreier 2011. p19

2.1.2 Agile product development

In nature, the creature survives, that can best adapt to changes in its habitat. Transferred to various industries, it is expected that companies who cannot respond quickly enough to changing requirements, will have a hard time in the future. [15] Agile, originally derived in part from the manufacturing sector, has evolved into a set of principles and practices that have flourished within and found applications beyond the information technology sector. Its adaptive, value-driven, collaborative and empowering essence drives innovation in an iterative and incremental manner that is founded in organizational and experiential learning.[16]

Meyer defines five organizational and three technical principles which constitute the core of the agile canon.

The organizational principles are:[17]

- Put the customer at the center - Deliver the best return on investment to the customer
- Let the team self-organize - Deciding on their own tasks
- Work at a suitable pace - Refuse to have periods of intense pressure forcing a team to work exceptionally hard in preparation for an upcoming deadline.
- Develop minimal software and hardware - Building only the essential functions; building only what is requested, excluding extra work to prepare for future reuse and extension
- Accept change - Full requirements cannot be determined at the beginning. Needs emerge as the project develops

The technical principles are:[18]

- Develop iteratively - Agile development implies an iterative and incremental development approach
- Treat tests as a key resource - The primacy of tests embodies the approach´s focus on quality

[15] Heerwagen 2018. p9
[16] Moran 2015. p1
[17] Meyer 2014. p4
[18] Meyer 2014. p6

- Express requirements through scenarios - A scenario is a description of a particular interaction of a user with the system

Companies continually strive toward becoming efficient and competitive through various means. Most enterprises are severely constrained by their inability to change their processes in response to new market needs.[19] In practice agile teams tend to be small comprising of heterogeneous *generalizing specialists* capable of engaging in several distinct types of work (e.g. analysis, development, testing). Customer representatives are highly engaged, attend planning and demonstration events and be available on short notice should the solution team require their input. This is in contrast to non-agile approaches where business and development teams tend to be separated with relatively little contact beyond exchange of requirements and specifications.[20]

In 2001 the *Agile Manifesto* was developed and signed by seventeen experts. The mindset behind the manifesto was that in order to succeed in the new economy, to move aggressively into the era of e-business, e-commerce, and the web, companies have to rid themselves of their old manifestations of make-work and arcane policies. The manifesto consists of twelve principles. Out of those twelve, the six most relevant, in terms of influencing a sequence of methods and the method application itself are cited below: [21]

- *Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage*
- *Business people and developers must work together daily throughout the project.*
- *Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done.*
- *Continuous attention to technical excellence and good design enhances agility.*
- *Simplicity, the art of maximizing the amount of work not done, is essential.*
- *At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.*

[19] Srinivasan 2017. p20
[20] Moran 2015. p12
[21] Beedle et al.

Finally, what emerges is that agile product development copes adaptively with rapid change through feedback learning loops that iteratively create and incrementally deliver value.

2.1.3 Systems engineering

INCOSE, the international council in systems engineering, defines system engineering as follows:

"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"[22]

From the definition of *INCOSE*, systems engineering differs from mechanical, electrical, and other engineering disciplines in certain ways. Systems engineering is focused on the system as a whole and emphasizes its total operation. It looks at the system from the outside (at its interactions with other systems and the environment) as well as from the inside. It is concerned not only with the engineering design of the system but also with external factors (e.g. customer needs, system operational environment), which can significantly influence the design. Bridging the traditional engineering disciplines is also a core point of systems engineering. The diversity of the elements in a complex system requires different engineering disciplines to be involved in their development. For the system to perform accurate, each system part must operate properly in combination with other system parts. Thus, the various parts cannot be engineered independently of one another and then simply assembled to produce a well operating system.[23]

Basically, systems engineering has certain designated areas of emphasis: The most important ones are noted as follows:[24]

- A top-down approach is required to view the system as a whole. Although bottom-up engineering activities in the past have very adequately covered the

[22] INCOSE 2019.
[23] Kossiakoff 2011. p4
[24] Blanchard und Blyler 2016. p19

design of various system parts, the necessary overview and an understanding of how these parts effectively work together has not always been present.

- A lifecycle orientation is required, addressing all phases to include development, production, distribution, operation, sustaining maintenance and support, and retirement and recycling.
- A better effort is required relative to the initial identification of system requirements, connecting these requirements to specific design targets, the development of appropriate design criteria, the follow-on analysis, and early virtual verification effort to ensure the effectiveness of early decision making in the development process. In the past, the early analysis effort has been minimal. This has required greater individual design efforts downstream in the lifecycle, many of which were not well integrated with other design activities and have required modification later on.
- An interdisciplinary approach is required throughout the development process to ensure that all objectives are met in an effective manner.
- Managing the design of complex technical systems requires an understanding of many topics. Therefore, interface management is key for highlighting problems and for monitoring the system design and integration effort.

System engineers need a certain set of skills to accomplish the tasks required from them. From the definitions above, the three most fundamental tasks of systems engineers are derived:[25]

Task 1: Use an interdisciplinary system thinking approach and consider the complete problem in every system decision in every stage of the system lifecycle: The problems change over the system lifecycle. The initial problem statement from one decision maker or stakeholder is never the whole problem. Therefore, in each stage, an interdisciplinary approach to systems thinking and problem definition is required.

Task 2: Convert all customer needs and preferences to system use cases, requirements, functions, and properties: Working with stakeholders to determine the functions

[25] Parnell et al. 2011. p185

that the system must perform is a daunting task when dealing with complex, dynamic, interdependent systems involving many stakeholders.

Task 3: Lead the requirement analysis, design synthesis, and system validation to achieve an effective system realization: It is essential for system engineers to lead the resolution of requirements, configuration control, design integration, interface management, and test issues that will occur during the lifecycle stages.

To sum up, the essence of the systems engineering approach is in selecting the right pieces, bringing them together, orchestrating them to interact in the right way and so creating requisite emergent properties, capabilities and behaviors of the whole. Essential systems engineering is executed in a way that the parts and the whole are operating and interoperating dynamically in their environment, to which they are open and adaptive, while interacting with other systems in that environment.[26]

2.2 Dependences in the system lifecycle

The development of a new product is often a complex process in which several company departments are involved. Moreover, a successful product development requires a lot of specific knowledge (e.g., about customer wishes, technological know-how, optimal and cost-effective development, production and logistics as well as the early knowledge of possible sources of error and quality problems). The use of processes, methods and tools helps to make the product development more efficient and therefore to realize a more successful product.[27]

In order to better understand the next sections, it is critically important to establish the terminology associated with process, method, and tool. Figure 2-2 aims to give a holistic view to show how processes, methods and tool are embedded by starting with the system lifecycle.

[26] Hitchins 2007. p120
[27] Graner 2013. p3

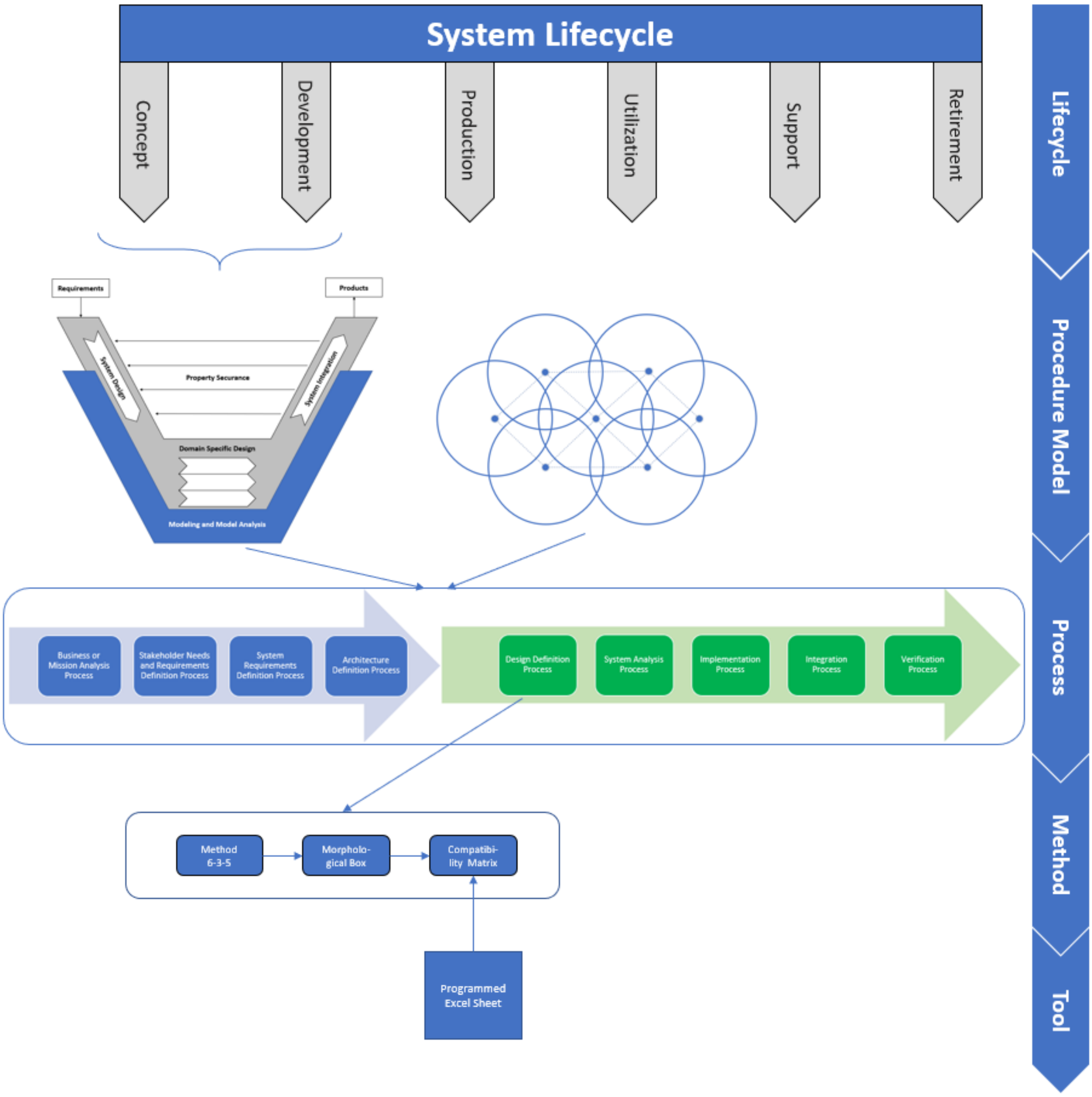


Figure 2-2: Correlation of a lifecycle, procedure models, process, methods and tool

The figure shows graphicly the correlation between a system lifecycle, procedure models, a process, methods and a tool. For better understanding

Table 1 displays the examples which are used in Figure 2-2:

System Lifecycle	ISO/IEC TR 24748-1[28]
Procedure Models	V-Model [29], Münchner Vorgehensmodell[30]

[28] ISO/IEC TR 24748-1 2010.
[29] VDI 2206 2004. p29
[30] Lindemann 2009a. p40

Process	ISO 15288[31]
Methods	Method 6-3-5, Morphological Box, Compatibility Matrix
Tool	Programmed Excel Sheet

Table 1: Examples for items displayed in Figure 2-2

In Figure 2-2 a system lifecycle and procedure models are displayed in addition to processes methods and tools . The reason is to illustrate the positioning of methods, which are the focus of this thesis, in a top-down approach starting with the holistic view of the system lifecycle. The procedure models describe what activities have to be performed in a logical manner. With the support of procedure models, processes are derived. Methods are chosen in order to execute the process as efficient and effective as possible. Tools support methods by executing their task. In the following subsections all mentioned terms are going to be discussed closer and some examples are given.

2.2.1 System lifecycle

A brief description of the system lifecycle supports to navigate and understand the correlation of processes, methods and tools used in the following chapters.

Products are dynamic in the sense that the passage of time affects their parts, functions, interactions, and value delivered to stakeholders. These observable effects are commonly known as system maturation effects. A system lifecycle is a conceptual model that is used by system engineers and managers to describe how a system matures over time. It includes the stages conceptualization, development, production, utilization, support, and retirement of the system.[32]

2.2.2 Procedure model

A procedure model is a logical model. Therefore, the challenge is to present the tasks and activities that generally occur in a development process in a logical and generic order. Procedure models should be tailor-made for specific projects and used to guide

[31] ISO/IEC/IEEE 15288:2015. p16
[32] Parnell et al. 2011. p7

concrete process planning. The targets for the use of procedural models are improved product quality and an improvement of processes.[33]

Basically, there are procedure models in different forms. Figure 2-3 presents a selection of procedure models used in product development. *Braun* exemplifies in that list which characteristics the procedure models underlie and how the models, through their different characteristics, pursue their use at different levels of abstraction and with different objectives.[34]

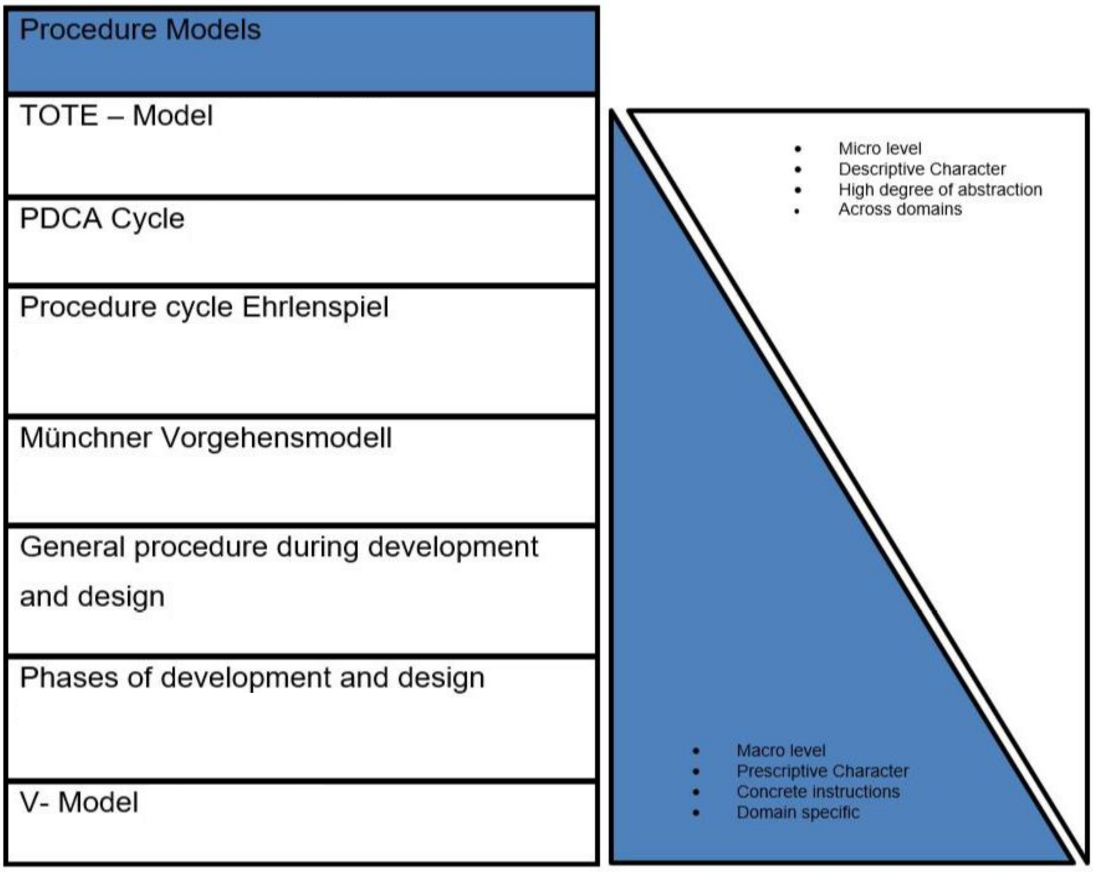


Figure 2-3: Procedure models of product development, inspired by Braun[35]

For the practical part of the thesis two procedure models are used to better display and choose suitable methods. Therefore, those two models are discussed more closely in the next subsection.

[33] Funke et al. 2000. p27
[34] Braun 2005. p28
[35] Braun 2005 p29

V-Model

In the VDI guideline 2206, the V-Model (Figure 2-4) is described as a model which originally derived from software development. It describes the generic procedure for the design of mechatronic systems and has a prescriptive character.[36]

The starting point is an actual development request. The next steps are *system design*, *domain-specific design* and *system integration*. The aim of the system design is to define a cross-domain solution concept that describes the structure and behavior of the system under development. Based on this jointly developed solution concept further concretization usually takes place separately in the participating domains (mechanical engineering, electrical engineering, software engineering). The results from the individual domains are then integrated into subsystems and an overall system in order to investigate the interaction and to be able to perform the verification and validation.[37] During system integration, starting with system parts through subsystems to the complete product, all properties of the product are tested against defined test cases in different stages.[38] The result of the V-model is the product. In this case, a product is understood as meaning not exclusively the finished, actually existing product but the increasing concretization of the future product. A complex product usually does not arise within one macrocycle. There are rather several cycles necessary to come to a desired result.[39]

[36] VDI 2206 2004. p29
[37] Ponn und Lindemann 2008. p16
[38] Lindemann 2016. p403
[39] VDI 2206 2004. p30

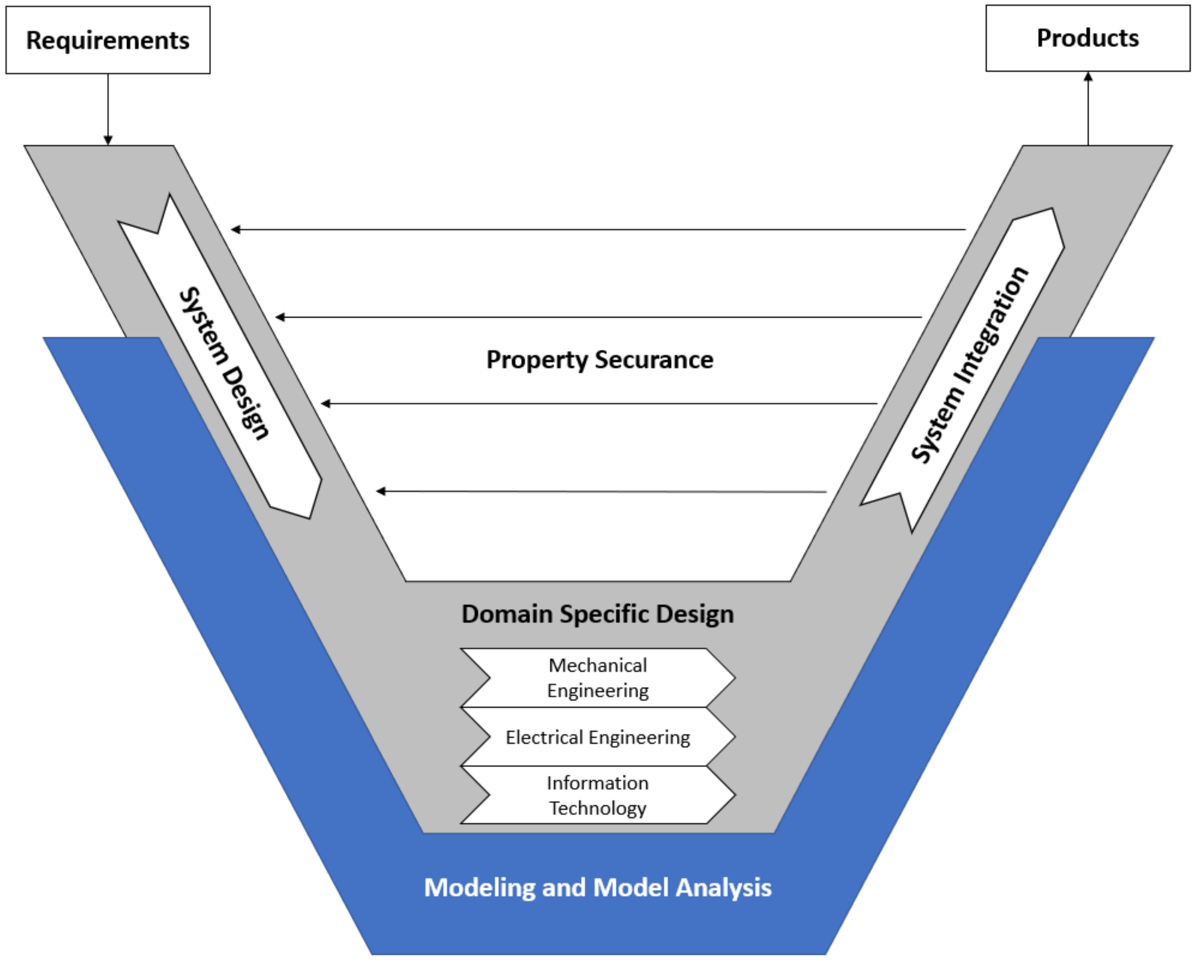


Figure 2-4: V-Model[40]

To further clarify which tasks have to be executed at a certain point in the V-model exemplary questions have to be answered in every phase. Those questions should be answered in detail at the right time to ensure the success of the project. Figure 2-5 helps teams focus their efforts to provide high value to stakeholders.[41]

The dotted line, after the first two questions represents the border between verification and validation. In theory, verification and validation (V&V) are different subjects and the differences between both should be understood.

- Verification: Did I build the thing right?
  Refers to a testing process that determines whether a product meets its specifications and derived requirements or compliant with applicable regulations.

[40] VDI 2206 2004. p29
[41] Medina 2015. p4

- Validation: Did I build the right thing?
  Refers to a testing process that determines whether a product satisfies the customer requirements of its intended customer or user.[42]

However, a dogmatic split-up in applying V&V may become counterproductive. Treating verification and validation totally separate has shown to lead to an explosion of plans and reports for proving compliance with each (derived) requirement.[43] Therefore V&V should be applied as shown in Figure 2-5.

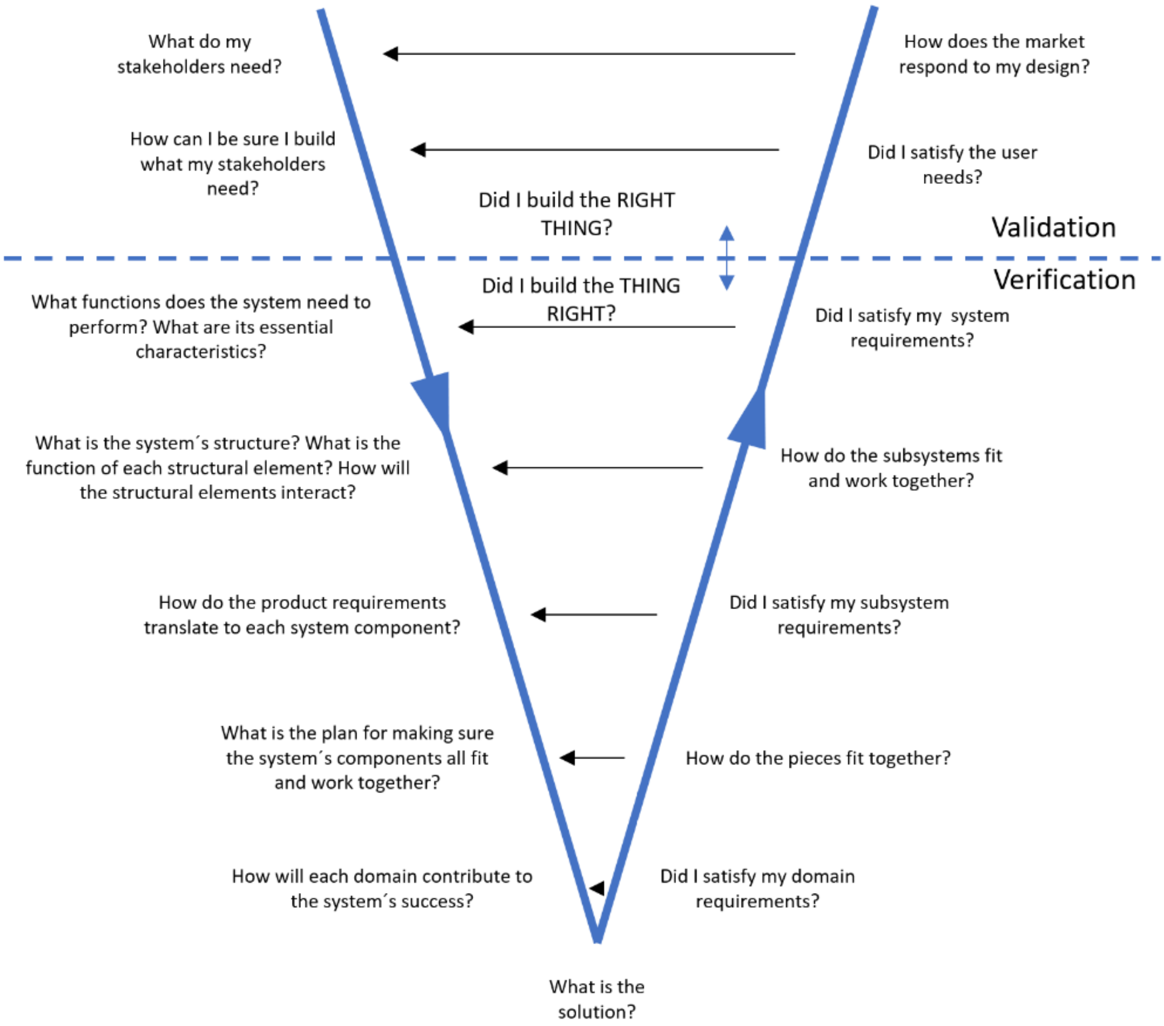


Figure 2-5: V-Model with allocated questions[44]

[42] Banks und Sokolowski 2009. p126
[43] Elich et al. 2012. p651
[44] Medina 2015. p7

Münchner Vorgehensmodell

Based on well-known procedure models as well as various research projects together with psychologists the *Münchner Vorgehensmodell* (MVM) was developed. The majority of prevailing procedure models in product development have a linear, often too fixed representation. Despite necessity users often fail to adapt the model and stay with the given basic pattern.

Clarify the problem, search for possible solutions, and decision support are the three main steps of problem solving. In the *Münchner Vorgehensmodell* (Figure 2-6) they are divided into smaller sub-steps to imitate a real problem-solving process. It can be gone through these steps sequentially as well as iteratively. The *Münchner Vorgehensmodell* contains the following seven steps as elements: plan target, analyze target, structure target, search for possible solutions, determine properties, decision support, secure target.[45]

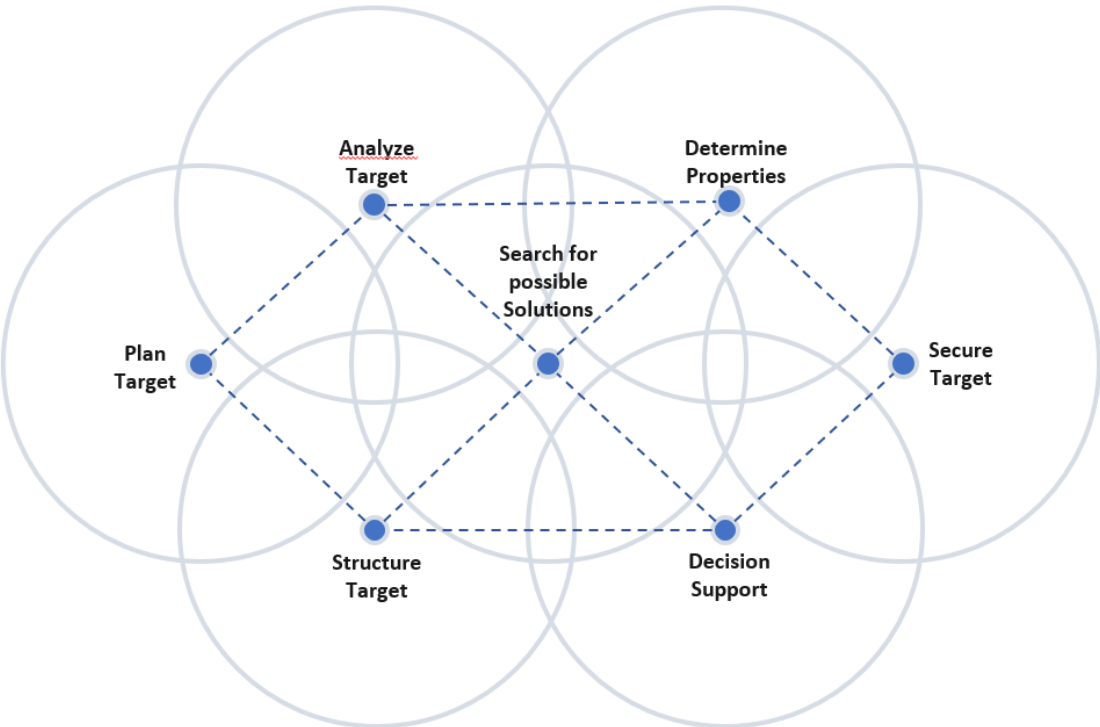


Figure 2-6: Münchner Vorgehensmodell[46]

The difference to existing process models is the special structure in form of a network. This comes closer to a real process than linear representations with possible returns.

[45] Lindemann 2016. p489
[46] Lindemann 2009b. p40

In practice, it is not possible to clearly separate the individual elements from each other. Therefore, the elements of the MVM are displayed using intersecting circles.[47]

2.2.3 Correlation of processes, methods, and tools

This section mainly focuses on processes, methods and tools but as they cannot be seen isolated in the product development process, some influencing factors are going to be discussed briefly as well. In Figure 2-7 *Estefan* provides a visual representation that does not only describe the relationship between the so-called PMTE elements (Process, Methods, Tools, and Environment) but also the effects of technology and people on the PMTE elements.

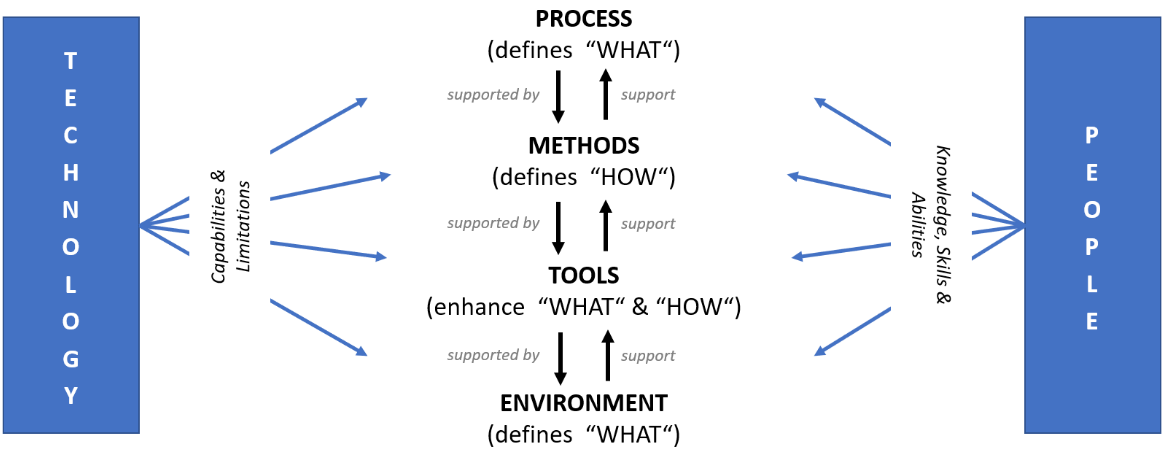


Figure 2-7:The PMTE Elements and Effects of Technology and People[48]

For the purpose of this thesis the following definitions and correlations for process, method and tool are used.

- A process is a logical sequence of tasks performed to achieve a particular objective. A process defines *WHAT* is to be done, without specifying *HOW* each task is performed.
- A method consists of techniques for performing a task. It defines *HOW* each task is performed. At any level, process tasks are performed using methods. However, each method is also a process itself, with a sequence of tasks to be performed for that particular method. Therefore, the *HOW* at one level of abstraction becomes the *WHAT* at the next lower level.

[47] Lindemann 2009b. p40
[48] Estefan Jeff A. 2008. p3

- A tool is an instrument that, when applied to a particular method, can enhance the efficiency of the task, provided it is applied properly and by somebody with proper skills and training. The purpose of a tool should be to facilitate the accomplishment of the *HOWs*. In a broader sense, a tool enhances the *WHAT* and the *HOW*. Most development tools are computer- or software-based, which are also known as *Computer Aided Engineering* (CAE) tools.[49]

Associated with the above definitions for process, methods, and tools is the environment. An environment consists of the surroundings, the external objects, conditions, or factors that influence the actions of an object, an individual person or group. These conditions can be social, cultural, personal, physical, organizational, or functional. The purpose of a project environment should be to integrate and support the use of tools and methods on that project. An environment thus enables, or in certain situations disables, the *WHAT* and the *HOW*.[50]

The capabilities and limitations of technology when creating a development environment are also to consider. Technology should not be used *just for the sake to use technology*. Technology can either push or hinder development efforts. Similarly, when choosing the right mix of PMTE elements, one must consider the of the people involved. When new PMTE elements are used, often the knowledge, skills and abilities of the people must be enhanced through special training and special assignments.[51]

2.2.4 Process

As stated in chapter 2.2.3 a process defines *WHAT* is to be done without specifying *HOW* each task is performed. For an organization to operate effectively, it must have many interrelated activities recognized, guided and directed. An activity that uses resources and that is executed to converting input into output can be considered a process. Often that results in one process generating direct input for the next.

In product development different processes can be observed: Existing products are changed, new products are developed, products already on the market are observed for quality and safety, patents are being examined etc. For this variety of total required processes, in this thesis a closer look is given at the process of product development,

[49] Estefan Jeff A. 2008. p2
[50] Estefan Jeff A. 2008. p2
[51] Estefan Jeff A. 2008 p3

in whose framework a new product is developed or an already existing one is developed further.[52]

Technical processes according to ISO 15288

The ISO 15288 defines four kinds of system lifecycle processes:

- Agreement processes
- Organizational project-enabling processes
- Technical management processes
- Technical processes

In this thesis the focus is solely on the technical processes. Figure 2-8 displays the processes that are dealt with in this thesis. Those processes are used throughout to orientate at what point certain methods are used. To better understand the positioning of the methods later on, every subprocess of the technical processes is described briefly (Table 2).

The validation process is the last subprocess relevant for the thesis. The last three subprocesses (operation, maintenance and disposal) are listed to complete the ISO 15288. [53]

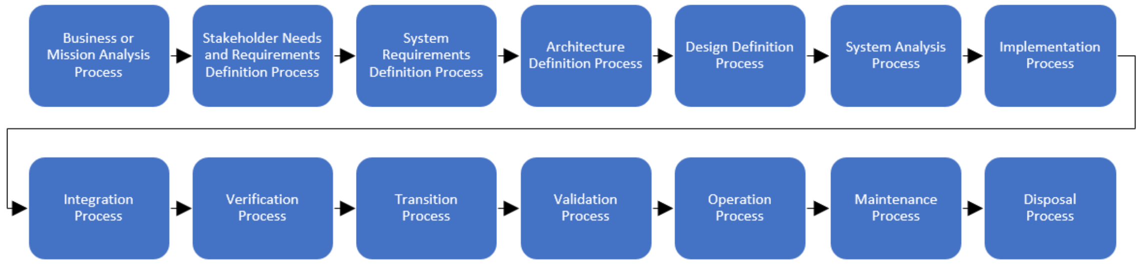


Figure 2-8 Technical system lifecycle processes, inspired by ISO 15288[54]

Business or mission analysis process	Define the business or mission problem or opportunity, characterize the solution space, and determine potential solutions that could address a problem or take advantage of an opportunity.

[52] Lindemann 2009b. p15
[53] ISO 15288 2015 p16
[54] ibidem

Stakeholder needs and requirements definition process	Define the stakeholder requirements for a system that can provide the capabilities needed by users and other stakeholders in a defined environment.
System requirements definition process	Transform the stakeholder, user-oriented view of desired capabilities into a technical view of a solution that meets the operational needs of the user.
Architecture definition process	Generate system architecture and alternatives, to select one or more alternative(s) that frame stakeholder concerns and meet system requirements, and to express this in a set of consistent views.
Design definition process	Provide sufficient detailed data and information about the system and its elements to enable the implementation consistent with architectural entities as defined in models and views of the system architecture.
System analysis process	Provide a rigorous basis of data and information for technical understanding to aid decision-making across the lifecycle.
Implementation process	Realize a specified system part.
Integration process	Synthesize a set of system elements into a realized system (product or service) that satisfies system requirements, architecture, and design.

Verification process	Provide objective evidence that a system or system element fulfils its specified requirements and characteristics.
Transition process	Establish a capability for a system to provide services specified by stakeholder requirements in the operational environment.
Validation process	Provide objective evidence that the system, when in use, fulfills its business or mission objectives and stakeholder requirements, achieving its intended use in its intended operational environment.
Operation process	Use the system to deliver its services.
Maintenance process	Sustain the capability of the system to provide a service.
Disposal process	End the existence of a system part or system for a specified intended use, appropriately handle replaced or retired parts, and to properly attend to identified critical disposal needs (e.g., per an agreement, per organizational policy, or for environmental, legal, safety, security aspects).

Table 2: Description of technical system lifecycle processes according to ISO 15288[55]

[55] ISO/IEC/IEEE 15288:2015. p16

2.2.5 Method

The term method denotes the description of a rule-based and planned procedure, according to which specific activities have to be carried out to meet certain requirements. A method is target-oriented and focuses on the generation of required output information based on existing input information (Figure 2-9). It is characterized by a strong operational character.[56]

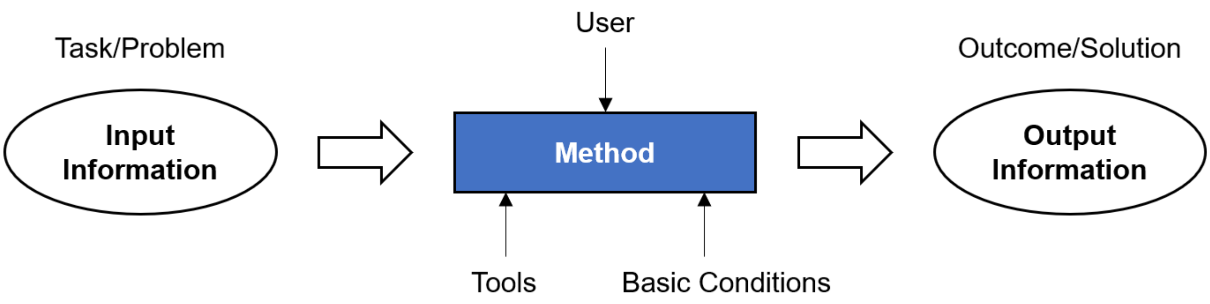


Figure 2-9: Information Conversion[57]

As mentioned in section 2.2.3, procedure models and processes help to navigate in the sense of *WHAT*, methods lead to the concrete work steps in the sense of *HOW*. Methods offer suggestions for the sequence of specific activities. A method must be varied if necessary and adapted to the respective situation. Often it is enough to adapt only individual modules of a method in order to meet the current and specific boundary conditions. The understanding of how individual steps in a method work is indispensable for their modularization and flexible adaptation.

The term method is broad and not always clearly definable. A method can consist of a few action sequences, such as in a *Pairwise Comparison*. However, the term is also used for the QFD (*Quality Function Deployment*) method, although in this case it is the combination of various individual methods (*Customer Survey*, *Benchmarking*, *Brainstorming* etc.). Even within less extensive methods, other methods are available. Within the method *Brainstorming* for example, methods such as *Mind Mapping*, *Gallery Methods* and others can be observed. Methods cannot simply be structured hierarchically. Better suited to this is the form of a network in which individual methods and their sub-steps can be used as modules in other methods.[58]

[56] Lindemann 2009b. p48
[57] Schwankl 2002. p38
[58] Lindemann 2009b. p49

In order to use methods successfully, several considerations are necessary. First, it has to be clarified whether there is a need for a method application in a specific development situation. If one concludes that the use of a method makes sense, an adequate method is to select. It is important to clarify whether the method supports the present task and the achievable effect agrees with the desired results. Some methods cannot be transferred unchanged to different situations. For this reason, the method must be adapted individually to the given application situation.[59] VDI 2221[60] describes the uncertainty of application suitability in practice by the following points:

- The qualification, education and experience of the employees
- The product program or development and design tasks to be solved
- The size and structure of the company
- The possibilities of the method itself

2.2.6 Tool

Methods are often supported by IT tools or other kind of tools. They should make the application more effective and efficient. The range covered by the term tool is large and reaches from simple tools (e.g., templates), to complex software (e.g., for simulation or statistical analysis). In general, tools have a major impact on the success of a method application. Therefore, the situation changes for the user if a tool is available and the user additionally is experienced in dealing with it.[61]

When selecting tools, analog to selecting a method, it is to take into account that the use of tools is associated with effort. For example, time and money has to be invested into training or licensing fees have to be paid. Therefore, the effort must always be weighed against the benefits that can be achieved.[62]

[59] Ponn und Lindemann 2008. p18
[60] VDI 2221 1993. p32
[61] Lindemann 2009b p52
[62] Ponn und Lindemann 2008 p20

3 Elementary View of Methods

The key to every successful use of methods is to work with the right method in the first place. This chapter focuses on how to build a database which supports finding the right method to a given development situation. Later on, the methods are going to be decomposed into various elementary criteria. The elementary criteria describe certain aspects of methods. The collection of those criteria is defined as the *elementary view of methods*.

In Figure 3-1 the actions which need to be considered in the whole process of using a method are displayed, in the so-called *Münchner Methodenmodell*. This chapter describes the essentials why the elementary view of methods is important and where it is used in the process of selecting and executing a method.

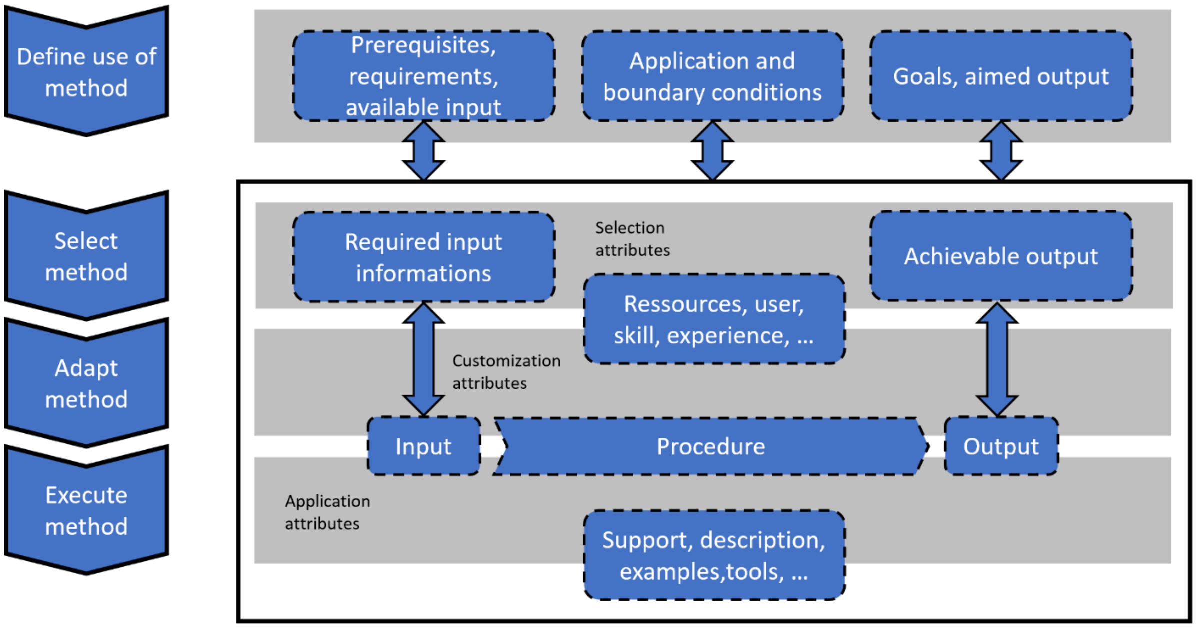


Figure 3-1: Münchner Methodenmodell[63]

The first step of the *Münchner Methodenmodell* is to examine which starting conditions (requirements, resources etc.) apply for the use of a method and to what extent the task or problem requires the use of a method at all. At the beginning of each use of method, based on the present task, the goal which should be achieved through the method, has to be defined.

[63] Braun 2005. p34

If the use of a method to complete the task makes sense, an adequate method has to be selected. The *Münchner Methodenmodell* indicates some aspects which are to be taken into account when selecting the method.[64]

As none of the methods fulfill all requirements resulting from the different influencing variables, it is necessary to choose from an extensive system of methods in order to select a method or a set of methods which suits the personnel, material, financial and organizational environment of the project.[65] Decisive for the right selection is to define targets and boundary conditions for the method used. It is essential to define the achievable output and also to observe the required input. The output of a method embodies the attainable effect and additional side effects of the method.[66]

The next step in the *Münchner Methodenmodell* focuses on the adaption of methods. In most cases, methods cannot be transferred unchanged to different situations. For this reason, methods need to be tailored to the individual application. Adjustments should, as far as possible, done before the actual method application. Nevertheless, continuous adaptation happens during the use of the method.

The application of the method itself includes the processing of the task at hand. Starting with the required input the methods generates a result, the output.[67]

3.1 Use of the elementary view

Computers and software nowadays offer a wide range of opportunities to improve the product development process because of the amount of data that can be processed in shorter time. To benefit from that possibility, the problem, that has to be solved, has to be broken down into numbers and categories to make it processable for computers.

The target is to find criteria for methods which are expressed numerically in order to make them processible for software applications. If it is possible to reduce all methods to representative numbers, they can be processed fast and for a lot of different practices. A software application could support the product development process by suggesting suitable methods depending on the development situation. In the next section

[64] Braun 2005. p34
[65] Schwankl 2002. p38
[66] Lindemann 2009b p49
[67] Braun 2005. p35

all criteria which were used to structure the methods are described. There are basically three different types of characterization used in this thesis:

- Numerical (processable) - characterization by a specific number
- Categorization (processable) - allocation to one of the predefined classes
- Description (not processable) - open wording

Even without a software application the elementary view of methods provides advantages for the user. The categorization also speeds up the search for methods without a software application. Users are capable to perform a structured research on methods and choose accordingly to their desired outcome.

3.2 Elementary criteria

As a basis for describing methods and defining elementary criteria the model for *Description of Methods* from *Lindemann* (Figure 3-2) is used. This model states that a method is described by input (input information), output (output information), required resources (number of employees, competence, tools, etc.), as well as control information (appointments, networking, etc.).

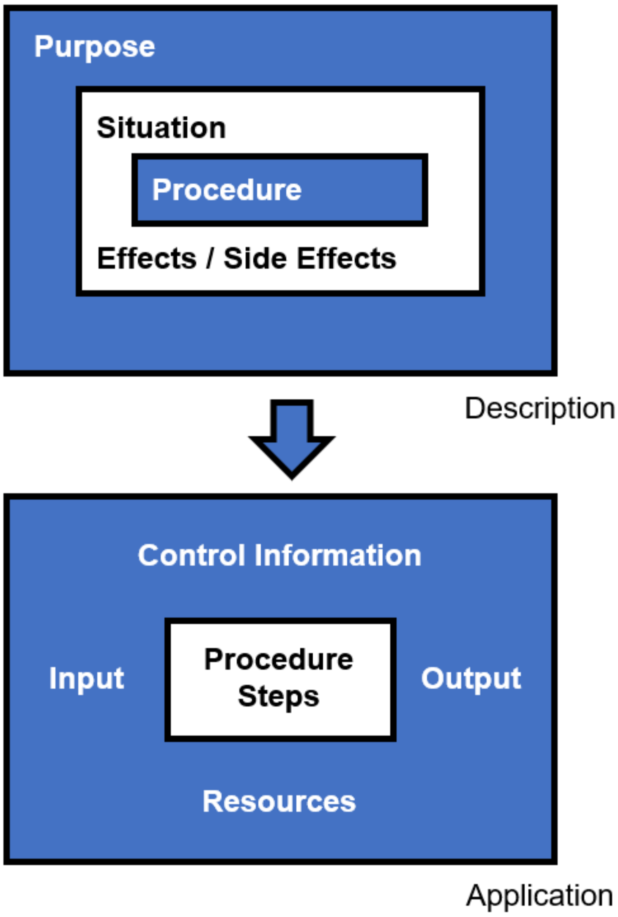


Figure 3-2: Model for description of methods[68]

In addition, some criteria to better choose and classify methods for specific applications are added (Table 3). All the criteria explained in Table 3 are discussed for the analyzed methods. The criteria are:

Purpose	In which situation should the method be used? What objective should the method support?
Input	What input information is needed to enable a successful use of the method?
Output	What is the output information of the method (does it align with my desired goals?)?
Qualitative / Quantitative	Is the output of the method qualitative or quantitative or can it be both depending on the use of the method?
Difficulty	Scale from 1-5 with the following definitions:

[68] Lindemann 2009b p52

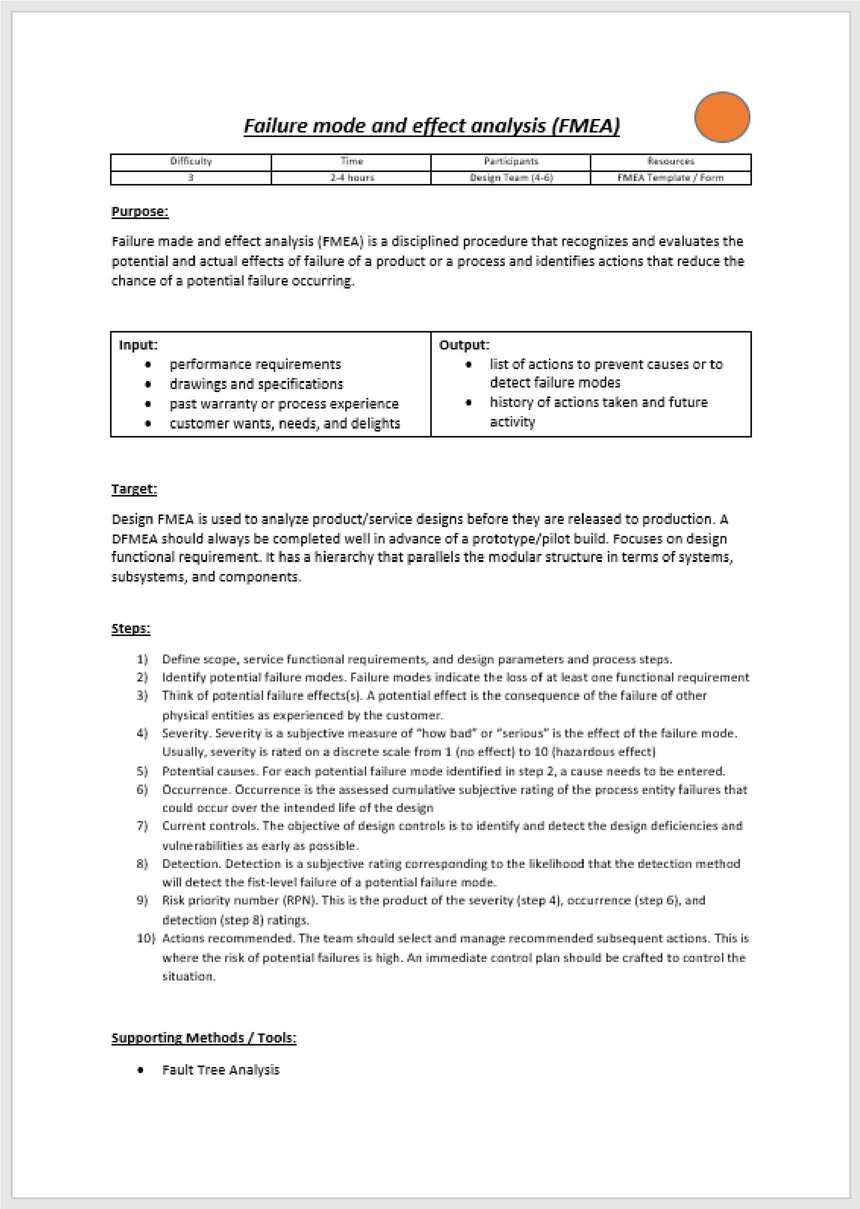
	• 1 – can be done in a short time (~10 minutes) of preparation without any previous knowledge (e.g., *Brain writing pool*) • 5 – only manageable with an expert who has profound experience with the method (e.g., *Load Matrix*)
Participants	What field of responsibility should the employees have to take on the method at hand?
# of Participants	How many full-time equivalents are needed for a standard use of the method?
One pager	A description of the method with all relevant information on one page.

Table 3: Elementary criteria

3.3 One pager

A one pager should give the user a brief overview of the method. After a rough preselection the one pager helps to narrow the selection down to a few. For methods with a difficulty level of 1, the one pager is enough to execute the method properly. For a higher degree of difficulty, it is only an orientation to get a better picture of what the method is about. In addition to the criteria from chapter 3.2, the one pager also provides information about the resources (e.g. computer programs, templates, etc.), possible supporting methods and tools, and an overview about the execution by explaining some of the key steps which need to be done. The circle next to the name of the method describes if the output of the discussed method is qualitative or quantitative (grey – qualitative, orange – quantitative).

Figure 3-3 shows an example of a one pager. It describes the method *Failure Mode and Effect Analysis (FMEA).* At this point, only a few methods are backed up with an exemplary one pager. The completion of that task and further plans are discussed in chapter 8.



Failure mode and effect analysis (FMEA)

Difficulty	Time	Participants	Resources
3	2-4 hours	Design Team (4-6)	FMEA Template / Form

Purpose:

Failure made and effect analysis (FMEA) is a disciplined procedure that recognizes and evaluates the potential and actual effects of failure of a product or a process and identifies actions that reduce the chance of a potential failure occurring.

Input:	Output:
• performance requirements • drawings and specifications • past warranty or process experience • customer wants, needs, and delights	• list of actions to prevent causes or to detect failure modes • history of actions taken and future activity

Target:

Design FMEA is used to analyze product/service designs before they are released to production. A DFMEA should always be completed well in advance of a prototype/pilot build. Focuses on design functional requirement. It has a hierarchy that parallels the modular structure in terms of systems, subsystems, and components.

Steps:

1) Define scope, service functional requirements, and design parameters and process steps.
2) Identify potential failure modes. Failure modes indicate the loss of at least one functional requirement
3) Think of potential failure effects(s). A potential effect is the consequence of the failure of other physical entities as experienced by the customer.
4) Severity. Severity is a subjective measure of "how bad" or "serious" is the effect of the failure mode. Usually, severity is rated on a discrete scale from 1 (no effect) to 10 (hazardous effect)
5) Potential causes. For each potential failure mode identified in step 2, a cause needs to be entered.
6) Occurrence. Occurrence is the assessed cumulative subjective rating of the process entity failures that could occur over the intended life of the design
7) Current controls. The objective of design controls is to identify and detect the design deficiencies and vulnerabilities as early as possible.
8) Detection. Detection is a subjective rating corresponding to the likelihood that the detection method will detect the fist-level failure of a potential failure mode.
9) Risk priority number (RPN). This is the product of the severity (step 4), occurrence (step 6), and detection (step 8) ratings.
10) Actions recommended. The team should select and manage recommended subsequent actions. This is where the risk of potential failures is high. An immediate control plan should be crafted to control the situation.

Supporting Methods / Tools:

- Fault Tree Analysis

Figure 3-3: One Pager FMEA

3.4 Attribute list

In total 58 methods were analyzed in the product development process for this thesis. All of these methods are arranged in a list and can be filtered depending on the situational selection criteria. Table 4 shows an extract of the method attribute list. The complete list is displayed in appendix A.

Method	Purpose	Input	Output	Qual. / Quan.	Difficulty (1-5)	Participants	# of Participants
4 Step Sketch	Creates ideas and concepts in a step-by-step process	Specification sheet, idea, problem, need	Ideas / concepts		1	Design team	2~6
6 Thinking Hats	Ideas are evaluated from different perspectives to find optimal solutions	Idea / concept	Critical evaluation of the idea		2	Design team	4~7
A-B Testing	Selection of the user prefered concept	Two or more testable concepts	User preference for one concept		3	Design team	2~5
Benchmarking	Compares one's product / business process / performance to the industry´s best	Product /process /area to improve	Inspiration and data for improvement		2	Design team	4~6
F.A.S.T	Develop a graphical representation showing the logical relationships between the functions of a project, process or product	Specification sheet	Logical relationships between functions		2	Design team	4~6
SWOT Analysis	Understand your strenghts and weaknesses, discover new opportunities and manage / eliminate threats	Internal and external research / benchmark	Functions, parts, areas to focus on		2	Design team	4~6
Brain Writing Pool	Generate a variety of ideas	Specification sheet, problem, need	Variety of ideas		1	Design team	5~8
Computer Aided Design	Use of computer systems to aid in creation, modification or optimization of a design	Concept of the design	Digital 3D model		3	Design team	2~4
CFD	Solve fluid mechanical problems approximately with numerical methods	CAD file	Fluid mechanical analysis of the System		4	Design team	1~2
Compatibility Matrix	Matrix for complete pairwise comparison of elements in terms of their compatibility	Set/variety of partial solutions	Possible combinations for final solution		2	Design team	3~6
Cost-Benefit Analysis	Decide whether to pursue an idea or not	Idea / concept	Worth pursuing or not		3	Design team	4~6
Durability Test	Test to see if the prototype works for the strived for time	Prototype	Failure modes		3	Verification team	4~6
FMEA	Recognizes and evaluates the potential and actual effects of failure of a product or a process	Concept / Layout / Detail design	Identification of critical components and weak spots		3	Design team	4~6
Function-Cost-Analysis	costs are being assigned to the individual functions	Detail Design, Development and production costs	Function and component attached costs		2	Design team, Production, Procurement	4~6
Load Matrix	To set up a test schedule which tests systems, subsystems or components in the sense of their stress collective	Stress collective	Failure modes		5	Verification team	4~6
Method 6-3-5	Generate a variety of ideas	Specification sheet, idea, problem, need	Wide variety of ideas		1	Design team	6

Table 4: Extract of the method attribute list

Methods from all different phases of the product development process were analyzed and categorized on the basis of the elementary criteria. There are different options to start the method selection process for any specific situation.

For example: The project team is under time pressure and has not a lot of knowledge about a certain activity. In that situation it would be advisable to start filtering the list by difficulty. After that the list shows all methods sorted either from difficulty 1 to 5 or the other way around. In the next step the list is filtered by the project team for the easiest method that generates the desired output. When a suitable method is found the team has to check if the method is not only easy to learn but also executable in the given timeframe.

3.5 Use of Münchner Vorgehensmodell

During the time of research and building up the database the perception arose that the list is not efficient enough for finding and selecting a method in a specific situation. The main problem is that methods for all different functions are in one database without any classification into their area of application. To overcome those problems and support the selection process even more, all methods additionally were classified into function groups. For that purpose, the classification criteria of the *Münchner Vorgehensmodell* are used.

In chapter 2.2.2 the principle of the *Münchner Vorgehensmodell* is explained. To better understand the assignment of methods every function is discussed briefly in Table 5.

Plan target	Analysis of the situation as well as the derivation of concrete actions. What factors play a role for the analysis of the situation depends on the desired output/goal.
Analyze target	Includes the clarification and description of the desired target. The general target in the product development process is to develop a requirement conform product. To accomplish that, it is necessary to formulate concrete and detailed requirements for the new product.
Structure target	Helps to determine focus areas and narrows down the search for possible solutions. Therefore, the system has to be viewed in a clearly arranged form which supports the problem-solving process.
Search for possible solutions	Describes the search for existing and the generation of new solutions. An important principle for the search of solutions is to think in alternatives. One should never be satisfied with the first idea as it may not lead to an optimal solution.
Determine properties	Determines the development of relevant characteristics by property analysis (primary the properties of the prepared solution ideas).

Decision support	Represents the evaluation of solution ideas and alternatives as well as making a selection. Not every solution idea is also an alternative. A solution alternative differs from a solution idea in the sense that the idea already went through a process of property analysis and evaluation.
Secure target	To reduce the risk at the realization of decisions. Even seemingly insignificant mistakes, both in the product and in the process, can have serious consequences. Therefore, a preventive verification of achievement of objectives should start early in the development process. Hence it is important to first identify and assess potential risks. If necessary, measures must be defined and implemented in order to minimize the identified risk.

Table 5: Functions of *Münchner Vorgehensmodell*[69]

In Figure 3-4 all analyzed methods are assigned to at least one of the functions of the *Münchner Vorgehensmodell*. In some cases, methods can occur in multiple areas. That happens when the output of a method can be used to tackle different problems.

For example: The method benchmarking is assigned to both functions- analyze target and search for possible solutions.

- Analyze target: On the one hand, a profound benchmark helps to clarify the desired target by analyzing what is already on the market. It may also reveal unsolved problems in the target market which have not been recognized before and therefore give the product a competitive advantage by targeting to solve that problems.
- Search for possible solutions: On the other hand, benchmarking can help to find new solutions. Most new products are just a new combination of already existing functions or systems. Thus, the key is to benchmark across industries and markets to look for partial solutions of the problem and arrange them in a new way.

[69] Lindemann 2009a. p48

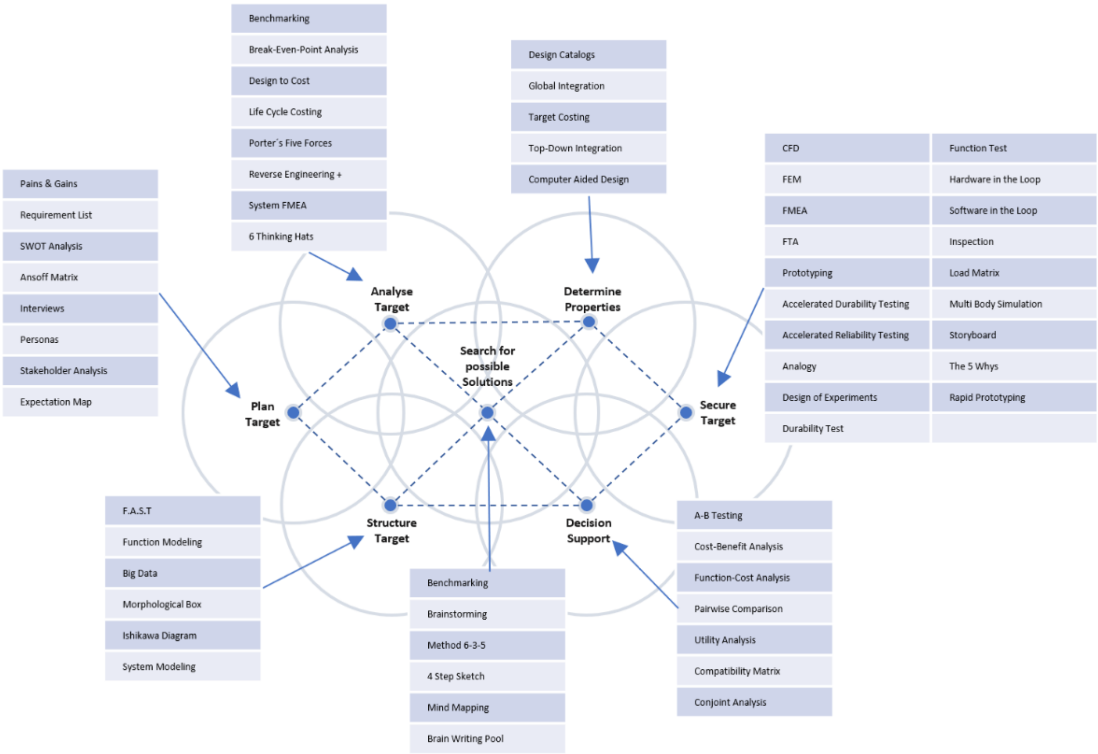


Figure 3-4: Münchner Vorgehensmodell with assigned methods

The allocation of the methods to functions cannot only be displayed in the *Münchner Vorgehensmodell* but also in the method attribute list (Table 6). An additional column with the heading *Münchner Vorghensmodell* attaches each method to a function. If a method, such as benchmarking, is allocated to multiple functions in Figure 3-4, the function which represents the most regular field of application is listed in the column.

Method	Purpose	Input	Output	Qual. / Quan.	Difficulty (1-5)	Participants	# of Participants	Münchner Vorgehensmodell
4 Step Sketch	Creates ideas and concepts in a step-by-step process	Specification sheet, idea, problem, need	Ideas / concepts		1	Design team	2~6	SEARCH FOR POSSIBLE SOLUTIONS
6 Thinking Hats	Ideas are evaluated from different perspectives to find optimal solutions	Idea / concept	Critical evaluation of the idea		2	Design team	4~7	ANALYSE TARGET
A-B Testing	Selection of the user prefered concept	Two or more testable concepts	User preference for one concept		3	Design team	2~5	DECISION SUPPORT
Benchmarking	Compares one's product / business process / performance to the industry´s best	Product /process /area to improve	Inspiration and data for improvement		2	Design team	4~6	ANALYSE TARGET
F.A.S.T	Develop a graphical representation showing the logical relationships between the functions of a project, process or product	Specification sheet	Logical relationships between functions		2	Design team	4~6	STRUCTURE TARGET
SWOT Analysis	Understand your strenghts and weaknesses, discover new opportunities and manage / eliminate threats	Internal and external research / benchmark	Functions, parts, areas to focus on		2	Design team	4~6	PLAN TARGET
Brain Writing Pool	Generate a variety of ideas	Specification sheet, problem, need	Variety of ideas		1	Design team	5~8	SEARCH FOR POSSIBLE SOLUTIONS
Computer Aided Design	Use of computer systems to aid in creation, modification or optimization of a design	Concept of the design	Digital 3D model		3	Design team	2~4	DETERMINE PROPERTIES
CFD	Solve fluid mechanical problems approximately with numerical methods	CAD file	Fluid mechanical analysis of the System		4	Design team	1~2	SECURE TARGET
Compatibility Matrix	Matrix for complete pairwise comparison of elements in terms of their compatibility	Set/variety of partial solutions	Possible combinations for final solution		2	Design team	3~6	DECISION SUPPORT
Cost-Benefit Analysis	Decide whether to pursue an idea or not	Idea / concept	Worth pursuing or not		3	Design team	4~6	DECISION SUPPORT
Durability Test	Test to see if the prototype works for the strived for time	Prototype	Failure modes		3	Verification team	4~6	SECURE TARGET
FMEA	Recognizes and evaluates the potential and actual effects of failure of a product or a process	Concept / Layout / Detail design	Identification of critical components and weak spots		3	Design team	4~6	SECURE TARGET
Function-Cost-Analysis	costs are being assigned to the individual functions	Detail Design, Development and production costs	Function and component attached costs		2	Design team, Production, Procurement	4~6	DECISION SUPPORT
Load Matrix	To set up a test schedule which tests systems, subsystems or components in the sense of their stress collective	Stress collective	Failure modes		5	Verification team	4~6	SECURE TARGET
Method 6-3-5	Generate a variety of ideas	Specification sheet, idea, problem, need	Wide variety of ideas		1	Design team	6	SEARCH FOR POSSIBLE SOLUTIONS

Table 6: Extract of the method attribute list with function allocation

3.6 In- and output information

Engineering tasks generate large volumes of data and information that must be available over the lifecycle of the system. An organization's ability to encode, communicate and organize information has continued to increase over the past several years. The analytical, computational and organizational tools that are used to manage information have also grown more and more powerful over this ensuing time period.[70]

During the course of a development process, the information about different system aspects matures. Methods have a key role in that process since they are primarily responsible for refining the information about the system. The target of this section is to show what kind of information enters and exits the analyzed methods.

Therefore, a possibility of how the information, which is gathered during the development process, may be grouped is explained. The following classification is the basis for allocating methods to certain *Information Classes*. An information about the system

[70] Simpson et al. 2005. p126

is part of a specific *Information Class* if it specifies the objective of the class. For this thesis the system information is divided into eight information classes. There is no right to completeness for this list. It has to be seen as an attempt and a start to show what kind of information is processed with certain methods. The two classes, *Decision* and *Structure* only occur on outputs of methods because of their nature of operating. *Geometry* in general is part of *Physical Data* but is listed separately because of the emphasis some of the analyzed methods place on geometry. The information class *Requirement* is subordinate to all other classes since they are (primarily in the beginning of the development process) part of the requirements. Nevertheless, it is listed separately because especially in the beginning of the development process the information is imprecise and only meaningful in combination. Table 7 shows all eight information classes with a brief description.

Requirement	Subordinate information class that represents a combination of rough information from other classes especially in the early phases of the product development process. (e.g. *4-Step Sketch*).
Description	General information about function, customer use, and environment of the product (e.g. B*rainstorming*).
Geometry	Information that deals with shape, size, relative position of objects, and the properties of space (e.g. C*omputer Aided Design*).
Physical Data	Includes information about kinematics (e.g., time, speed), mechanics (e.g., mass, force, impulse…), thermodynamics (e.g., temperature, energy…) as well as electrodynamics (e.g., amperage, charge…) (e.g., F*inite Element Method*).
Failure Modes	Information related with failure. Doesn't matter if the failure is only possible and the damage if it occurs is evaluated or it actually occurred in tests (e.g., *FMEA*).

Costs	Information which is related to costs in any way (e.g., B*reak-Even Analysis*).
Decision (only output)	For methods which do not develop any additional information for the product but help to make comprehensible and structured decisions between available options (e.g., P*airwise Comparison*).
Structure (only output)	Representation of the available information in a for a specific situation relevant way (e.g., f*unction modelling*).

Table 7: Description of information classes

An example would be the method *Failure Mode and Effect Analysis.* To even start the method (depending on the application point), information about geometry, physical data (e.g., loads) and a description of the function and usage of the system is required. It is important to mention that the quality of the output heavily depends on the accuracy of the input information. Therefore, the information about geometry, physical data, description of the function, and usage of the system should have reached a certain maturity in order to generate the desired output. If that is the case, the method application is executed efficiently. For the exemplary method *FMEA*, all that input information is required to generate a precise list about possible failure modes and their effects on the system if they occur. In conclusion, information from the class's geometry, physical data and description is needed to specify the information class of failure modes.

Every in- and output of the analyzed methods is allocated to one or more information classes. Table 8 shows an extract of the method attribute list with two (blue surrounded) additional columns. Those columns display the information classes where the input information is coming from and which information classes are going to be specified with the output information.

Method	Purpose	Input	Output	Qual. / Quan.	Difficulty (1-5)	Participants	# of Participants	Sys. Input	Sys. Output	ünchner Vorgehensmodell
4 Step Sketch	Creates ideas and concepts in a step-by-step process	Specification sheet, idea, problem, need	Ideas / concepts		1	Design team	2~6	Requirements	Description	EARCH FOR POSSIBLE SOLUTIONS
6 Thinking Hats	Ideas are evaluated from different perspectives to find optimal solutions	Idea / concept	Critical evaluation of the idea		2	Design team	4~7	Description	Failure Modes / Description	ANALYSE TARGET
A-B Testing	Selection of the user prefered concept	Two or more testable concepts	User preference for one concept		3	Design team	2~5	Description / Prototype	Decision	DECISION SUPPORT
Benchmarking	Compares one's product / business process / performance to the industry´s best	Product /process /area to improve	Inspiration and data for improvement		2	Design team	4~6	Requirements	Geometry/ Phys. Data / Failure Modes / Costs	ANALYSE TARGET
F.A.S.T	Develop a graphical representation showing the logical relationships between the functions of a project, process or product	Specification sheet	Logical relationships between functions		2	Design team	4~6	Requirements	Description	STRUCTURE TARGET
SWOT Analysis	Understand your strenghts and weaknesses, discover new opportunities and manage / eliminate threats	Internal and external research / benchmark	Functions, parts, areas to focus on		2	Design team	4~6	Description	Decision	PLAN TARGET
Brain Writing Pool	Generate a variety of ideas	Specification sheet, problem, need	Variety of ideas		1	Design team	5~8	Requirement / Failure Mode	Description	EARCH FOR POSSIBLE SOLUTIONS
Computer Aided Design	Use of computer systems to aid in creation, modification or optimization of a design	Concept of the design	Digital 3D model		3	Design team	2~4	Requirements / Description	Geometry	DETERMINE PROPERTIES
CFD	Solve fluid mechanical problems approximately with numerical methods	CAD file	Fluid mechanical analysis of the System		4	Design team	1~2	Geometry / Phys. Data	Physical Data	SECURE TARGET
Compatibility Matrix	Matrix for complete pairwise comparison of elements in terms of their compatibility	Set/variety of partial solutions	Possible combinations for final solution		2	Design team	3~6	Description / Geometry / Physical Data	Decision	DECISION SUPPORT
Cost-Benefit Analysis	Decide whether to pursue an idea or not	Idea / concept	Worth pursuing or not		3	Design team	4~6	Description	Costs	DECISION SUPPORT
Durability Test	Test to see if the prototype works for the strived for time	Prototype	Failure modes		3	Verification team	4~6	Geometry / Phys. Data	Failure Modes	SECURE TARGET
FMEA	Recognizes and evaluates the potential and actual effects of failure of a product or a process	Concept / Layout / Detail design	Identification of critical components and weak spots		3	Design team	4~6	Geometry / Phys. Data / Description	Failure Modes	SECURE TARGET
Function-Cost-Analysis	costs are being assigned to the individual functions	Detail Design, Development and production costs	Function and component attached costs		2	Design team, Production, Procurement	4~6	Description	Costs	DECISION SUPPORT
Load Matrix	To set up a test schedule which tests systems, subsystems or components in the sense of their stress collective	Stress collective	Failure modes		5	Verification team	4~6	Description / Geometry / Physical Data	Failure Modes	SECURE TARGET
Method 6-3-5	Generate a variety of ideas	Specification sheet, idea, problem, need	Wide variety of ideas		1	Design team	6	Requirements	Description	EARCH FOR POSSIBLE SOLUTIONS

Table 8: Extract of the method attribute list with system information classes

For completeness, it has to be mentioned that nowadays a trend is to intensively consider product and system functions. Therefore, an information class of *Functions* is to be considered in the future.

4 Arrangement of Methods

The target of this chapter is to arrange methods to the product development process in a logical and structured manner. As a basis, a modified V-model is used to display the methods. The following subsections discuss the joining of the V-model with the technical processes defined by ISO 15288 and the adaptions which were made to the V-model to develop a clearly arranged view of the methods.

4.1 Merging of the technical processes of ISO 15288 and the V-model

Figure 4-1 displays the allocation of the technical processes of the ISO 15288 to the traditional V-model. The initial point of the traditional V-model is formed by an actual development order. In the next step the task is specified more precisely and described in the form of requirements. These requirements at the same time form the measure against which the product is to be assessed later.[71]

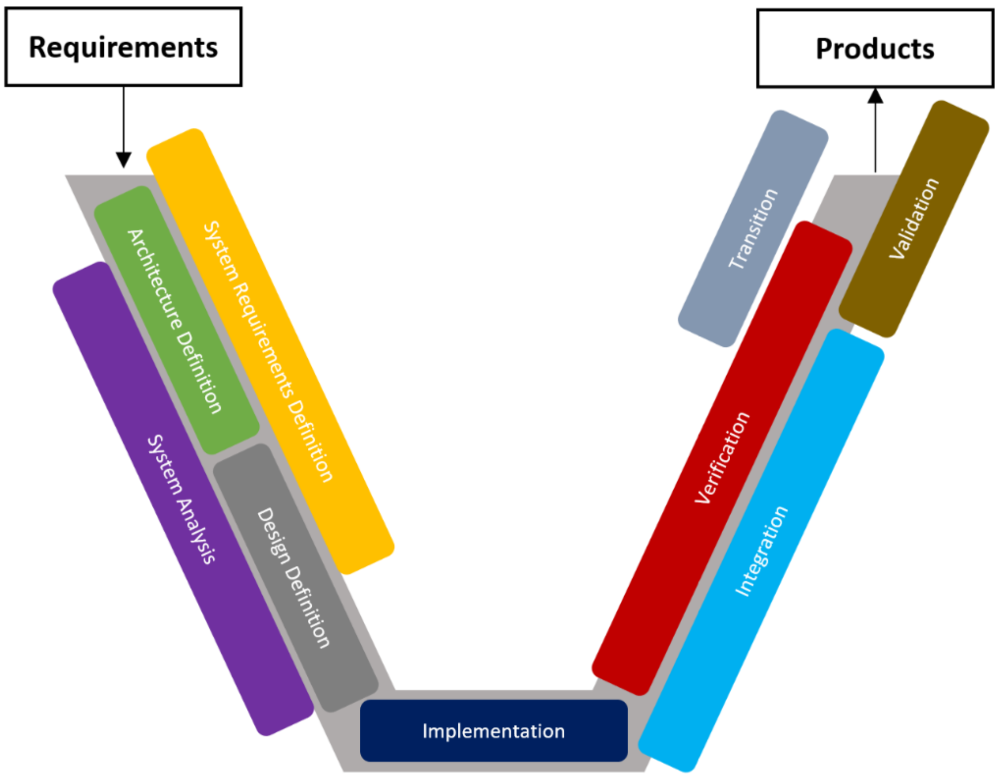


Figure 4-1: Traditional V-model with allocated technical processes of ISO 15288, inspired by Bajzek[72]

[71] VDI 2206 2004. p29
[72] Bajzek 2018. p48

The result of the V-model is the developed product. In this case, a product is understood as meaning not exclusively the finished, actually existing product but the increasing concretization of the future product which is called product maturity. Degrees of maturity are, for example, the laboratory prototype, the functional prototype, the pilot-run product, etc.[73]

4.2 Adaption of the V-model

In the system lifecycle the whole V-model is only executed in the development phase. That doesn't exclude, that segments of the V-model are also executed in the concept phase. Figure 4-2 shows an exemplary display of two segments of the V-model which are performed before the development phase.

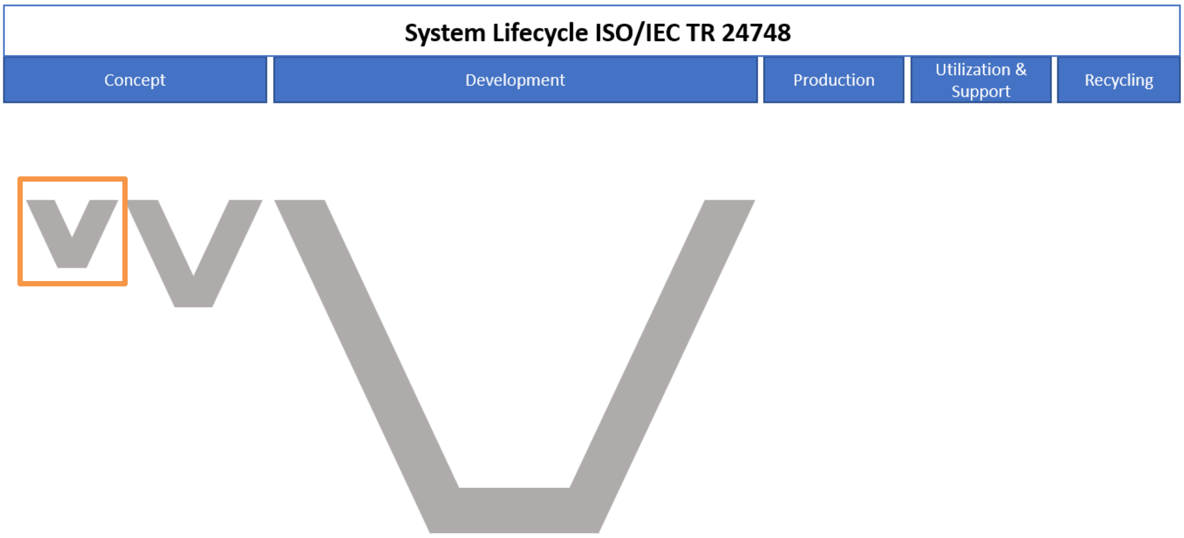


Figure 4-2: Example of multiple V-model segments in the system lifecycle

As an example, for a V-model segment in the concept phase, the orange marked v in Figure 4-2 is discussed. It represents parts of the business and mission analysis process. To deepen the understanding of the procedure of a V-model segment, exemplary statements and questions, which have to be carried out during this segment, are displayed in Figure 4-3.

[73] VDI 2206 2004 p30

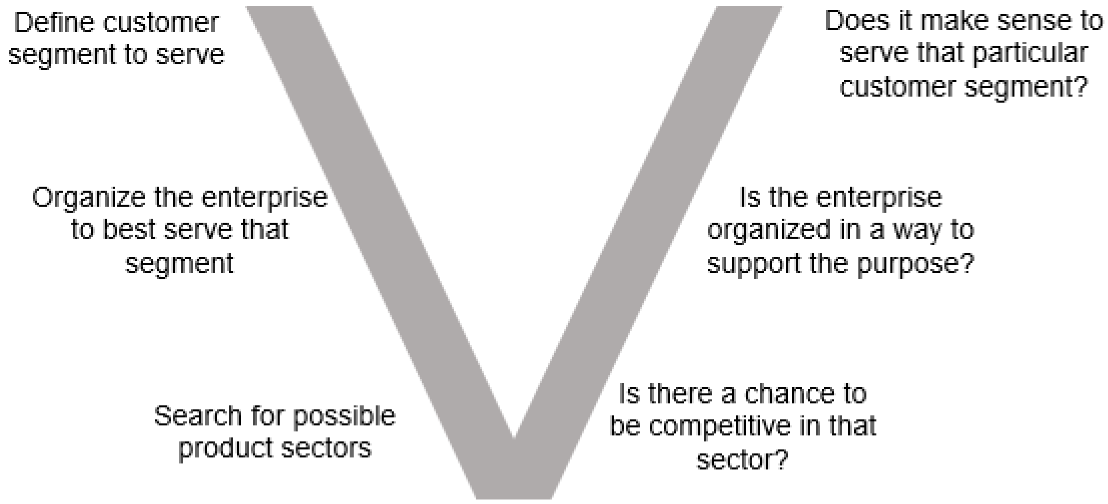


Figure 4-3: Segment of the V-Model for the business and mission analysis

At the INCOSE international symposium 2013, *Scheithauer and Forsberg* presented the paper *V-Model Views*[74]. This paper collects experiences and improvements from the past two decades. The authors extend the scope of the V-model from the development process to the lifecycle of the system, reaching from stakeholder needs to satisfaction. For the purpose of showing all analyzed methods in one model (chapter 4.3), the V-model segments from the concept phase are added to the traditional view. Therefore, the two processes (business or mission analysis, and stakeholder needs and requirement definition) are added to the traditional view shown in Figure 4-1. The extended V-model with the allocated technical processes of ISO 15288 is displayed in Figure 4-4.

[74] Scheithauer und Forsberg 2013.

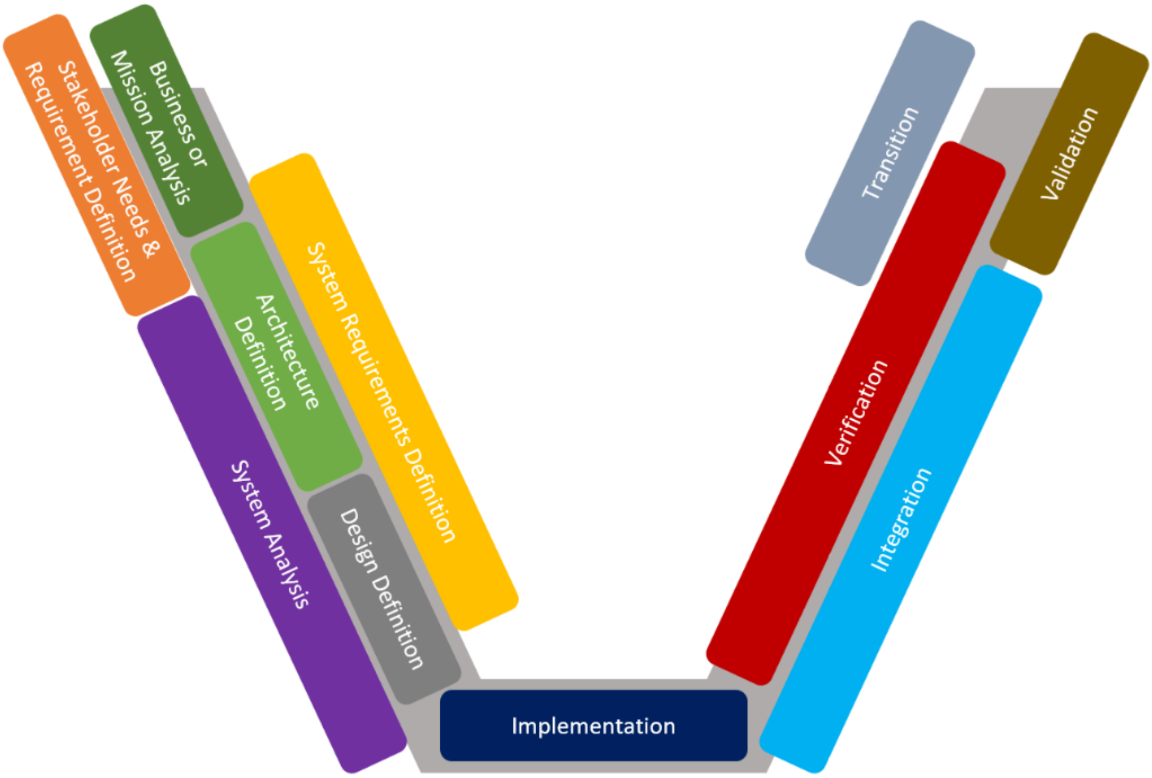


Figure 4-4: Extended V-model with allocated technical processes of ISO 15288, inspired by Bajzek[75]

Beside the extension of the scope of the V-model, another adaption is made in order to better allocate verification and validation methods. In the classic display of the V-model, verification and validation occur after the implementation phase. The detection of errors in this phase is notoriously expensive. As errors usually come from the first phases in the lifecycle, every phase, back to that error, has to be repeated. That fact increases the development costs dramatically. For this reason, one aim of this thesis is to provide the right methods to accomplish a system design verification and validation in order to detect errors during the early phases of development. [76]

Scheithauer and Forsberg state that verification and validation in general have either the focus on the substantiation that system requirements represent stakeholder needs adequately, or on the demonstration that the system requirements are implemented correctly and completely.[77] Compared to the traditional view, the paper splits the overall view of the V-model into different views. One of them is the so-called *Assurance-V*. The *Assurance-V* intends to put emphasis on the verification and validation topic by

[75] Bajzek 2018. p48
[76] Cambronero et al. 2010 p3
[77] Scheithauer und Forsberg 2013. p510

adding two additional V&V axes to the traditional V-model (Figure 4-5). Since those axes occur before the implementation, verification and validation are purely virtual.

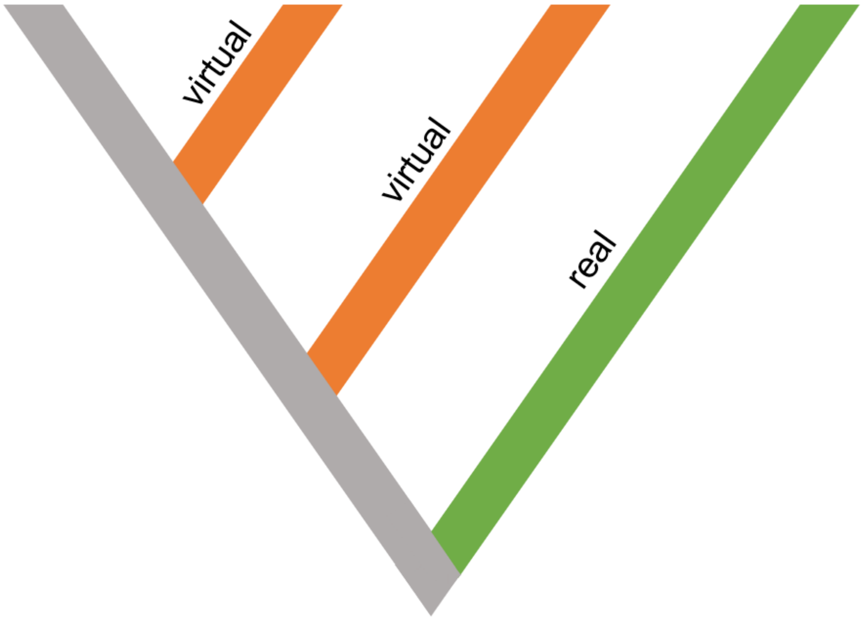


Figure 4-5: Assurance-V, inspired by Scheithauer and Forsberg[78]

4.3 Arrangement of methods

For companies, using an overview of methods in the product development process, several advantages arise. First of all, companies are able to define their as-is situation in the context of the use of methods in the product development process. Therefore, the previously used methods are compared to the methods displayed in Figure 4-6. If companies don’t use methods at several parts of the V-model, potential for possible improvement occurs. The displayed methods might support long existing processes to improve in certain areas by showing possibilities to tackle specific development phases. They are also the basis for new processes. According to prior defined requirements an appropriate chain of methods is chosen from the V-model. Ultimately, the V-model with aligned methods give organizations the chance to question their previous processes and might trigger an adaption for further applications.

[78] Scheithauer und Forsberg 2013 p510

Knowledge and experience enable engineers to perform most activities of the development process effectively and efficiently. If an engineer otherwise, is inexperienced or uncertain about a problem and the further course of action, a critical situation arises. In that case, routine actions have to be interrupted and replaced by a conscious and systematic approach (use of methods). Since those critical situations are not always recognized, it is important to develop a high sensitivity by reflecting on previous actions.[79] The display of methods in this chapter helps to start with a conscious and systematic approach in the first place or provides help if a critical situation arises.

With the defined changes in scope and the two additional *V&V* axes in the previous chapter, the final version of the V-model for the arrangement of methods is described. In Figure 4-6 all analyzed methods are aligned with the modified V-model. A problem with linear representation (as it is in the V-model) is that methods have to be put to one specific point in the model. Nevertheless, some methods can be executed in various parts of the V-model. For example, *Brainstorming:* on the one hand it is used to look for ideas for business opportunities in a very early phase. On the other hand, it is also applied in the design phase to generate a lot of different ideas for the actual product. It even is used in the verification phase to create ideas on how to best test a product or system in order to e.g., meet durability requirements.

Therefore, there are multiple options on where to put certain methods in the model. The final alignment happened with the intention to locate every method to the spot where it is most commonly used. The model in general doesn’t claim to be the only way to arrange those methods. It rather tries to give an overview of which methods could be possible at certain points in the development process. The red boxes in Figure 4-6 mark three details of the V-model which are going to be discussed in detail.

[79] Lindemann 2009a p48

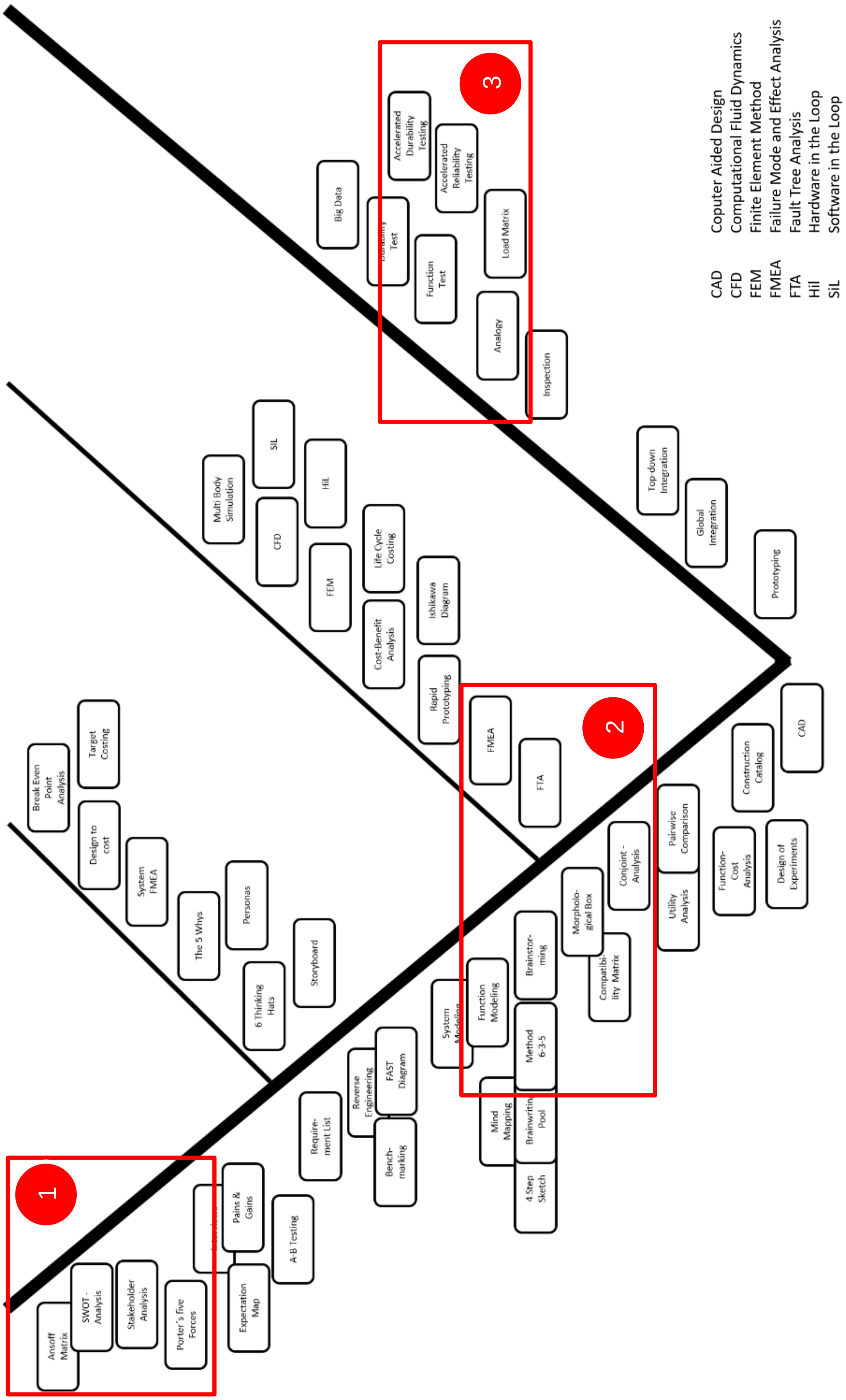


Figure 4-6: V-model with allocated methods

A lot of disciplines have to work together properly in order to develop a competitive and successful product (e.g., marketing, project management, technical development, controlling, logistics, purchasing). The focus of the model presented in this chapter is on technical development. Therefore, the majority of methods in Figure 4-6 intend to serve the technical development of a product. Nevertheless, some explementary methods from other disciplines are displayed as well (e.g., target costing, which is part of controlling). Possible future adaptions and specifications for other disciplines are discussed in chapter 8.

For better understanding of the model, three details (which are highlighted in Figure 4-6) are discussed in the following section. They all display a part of different phases of the model. The aim is to give an overview why certain methods are positioned as they are. Methods themselves are therefore not described in detail.

Detail 1 (Figure 4-7) displays a section of the very beginning of the model. According to Figure 4-4 the processes which are executed in this area are business and mission analysis, and stakeholder needs and requirements.

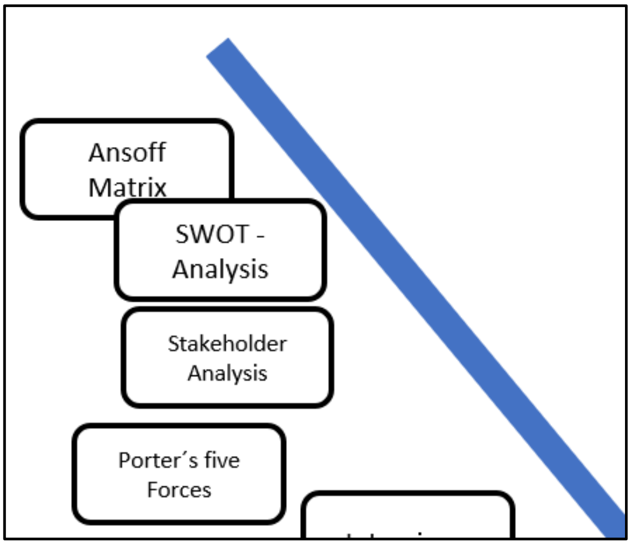


Figure 4-7: Detail 1

The A*nsoff Matrix* is a good example for a method in the process of business and mission analysis. It provides a framework to help executives devise strategies for future growth. If the management has decided on the future strategies the method *Stakeholder Analysis* could be advised. This method tries to identify groups or people who have an influential interest in the, to be developed, product. Relevant stakeholders are systematically collected, described briefly and their significance and influence on the outcome of the project is assessed (positive, negative, neutral).[80] Based on this, necessary steps to satisfy the different stakeholder are taken early in the process.

Detail 2 (Figure 4-8) is located at the start of the second virtual verification and validation axis.

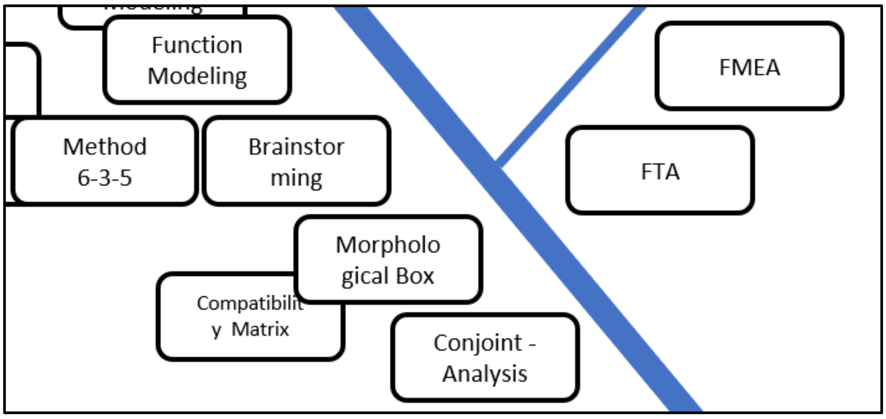


Figure 4-8: Detail 2

An example to execute the displayed part would be: with the method *Function Modeling* a solution-neutral representation of the functions of a system is created to ease its understanding and deal with its complexity. Those functions are the basis for the following used methods. *Method* 6-3-5 generates a lot of different ideas for every function defined earlier in a short amount of time. The method *Morphological Box* helps to handle system complexity by creating a matrix that contains functions and possible solutions. Subsequent to the *Morphological Box,* the method *Compatibility Matrix* is executed. It is a matrix for a complete pairwise comparison of elements in order to check their compatibility. At the end of this section, two to five different overall solution possibilities should be defined for the system.

In Figure 4-8 the verification and validation axis is positioned after the different methods for idea generation. That is just an example. The methods from the virtual V&V axes cannot only be applied at that exact point in the V-model. It is seen like an area of application which is shown exemplary in Figure 4-9.

80 Avgeropoulos 2014.

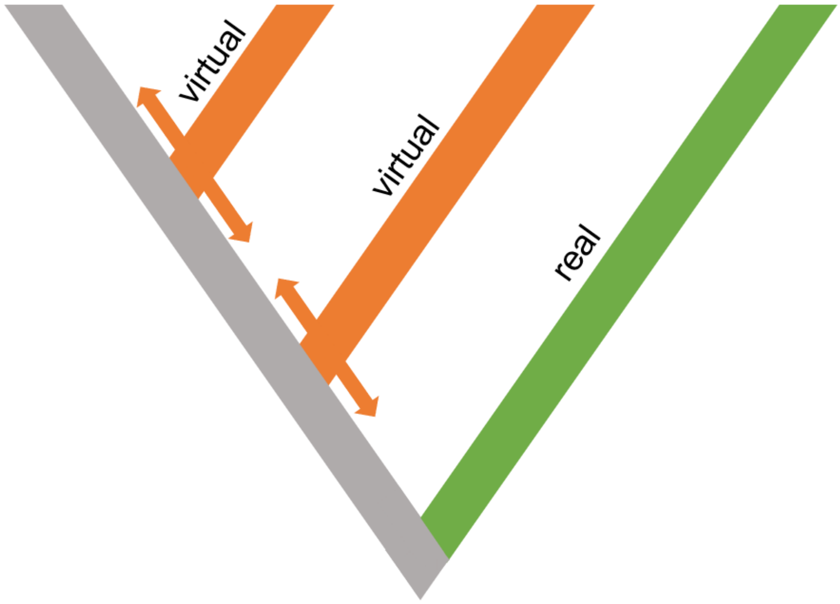


Figure 4-9: Application area for the virtual V&V axes

Figure 4-10 shows a detail which displays methods from the real (hardware available) V&V axis. Sufficient assurance of reliability and lifetime of products requires, with the background of increasing product complexity and shortened development times, special testing methods.[81] The method L*oad Matrix* is able to define an acceleration factor for durability testing to optimize the entire verification program. After the function tests, which confirm that all the required functions work properly, the accelerated durability tests start. The goal of this procedure is to verify the product as fast and as safe as possible.

[81] Denkmayr et al. 2003 p924

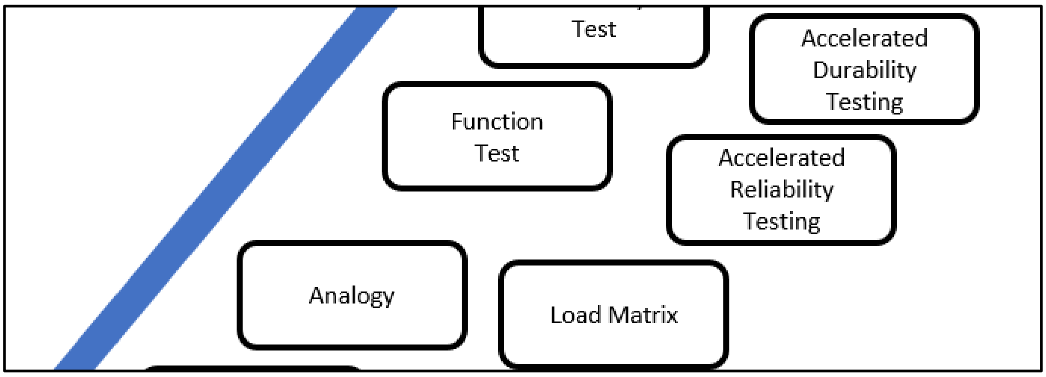


Figure 4-10: Detail 3

4.4 Differentiation between quantitative and qualitative methods

In product development it is distinguished between quantitative and qualitative methods. In Table 9 some characteristics of quantitative and qualitative methods are listed. If a method in this thesis is defined as quantitative or qualitative depends on the output information. The main indicator for differentiation is if the output is expressed numerically or not. That is important to define, since some methods have qualitative input but generate quantitative output and vice versa. An example for that case is the method *Pairwise comparison.* It is applied if different factors should be compared systematically. The input information consists of descriptions of the different functions which should be compared. After the method is carried out, every function is aligned to a number or percentage which indicates the importance of that specific function for the system. Further steps are planned according to the ranking the pairwise comparison generates.

Quantitative characteristics	Qualitative characteristics
Numerical data	Non-numerical data
Focus on measuring	Focus on understanding and interpreting
Standardized outcome	Open and flexible outcome
Test of hypotheses	Generation of hypotheses
Quantification of circumstances	Collection of suggestions for improvement
Verification of statistical correlations	Exploring of root causes

Table 9: Characteristics of quantitative and qualitative methods[82]

[82] Geller 2014 p4

The principle for the V-model view in Figure 4-11, where all methods are colored according to their affiliation is as followed. There are three options displayed:

- Orange: quantitative output
- Grey: qualitative output
- Orange/Grey: either qualitative, or quantitative output or combination of both depending on the operational situation. They are called mixed methods.

The method *interviews* is an example for a mixed method. An Interview may have specific questions which demand a precise numerical answer or open questions which give room for descriptions and explanations. Those two options can also be combined in one interview if it serves the purpose.

Figure 4-11 displays the same methods as Figure 4-6. The only difference is that the methods in Figure 4-11 are colored according to their affiliation to quantitative or qualitative methods. The figure shows that most methods at the beginning of the procedure model are qualitative. One reason for that is that the maturity of information of the product is not concrete enough in an early phase to break it down numerically.

On the second virtual, and the real V&V axis a lot of methods are quantitative or at least mixed methods. A numerical output is a main indicator if customer requirements are met or not because it is easily compared to the requirements defined at the beginning of the process. It is also important to check and display changes in the product numerically over time if there are iterations.

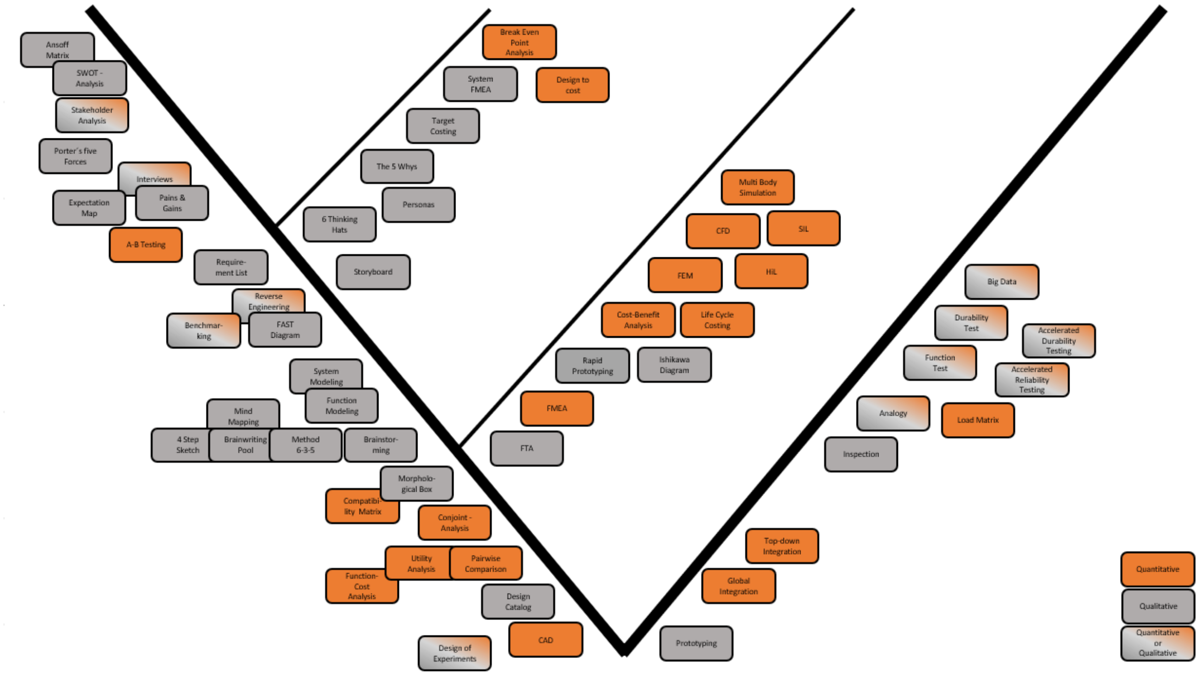


Figure 4-11: V-model with quantitative and qualitative methods

4.5 Input/output information in the V-model

As discussed in chapter 3.6, information classes are specified and developed throughout the product development process. Every method requires certain input information (from specific information classes) in order to generate the desired output. This section focuses on showing from which classes information is needed to start the method, and to which the output belongs.

Therefore, the in- and output of every method is linked to one or more information classes. In Table 8 the information classes are already linked to the methods and displayed in the attribute matrix. The graphical representation of methods and their information classes in combination with the V-model is shown in Figure 4-12. The different classes have already been described in chapter 3.6.

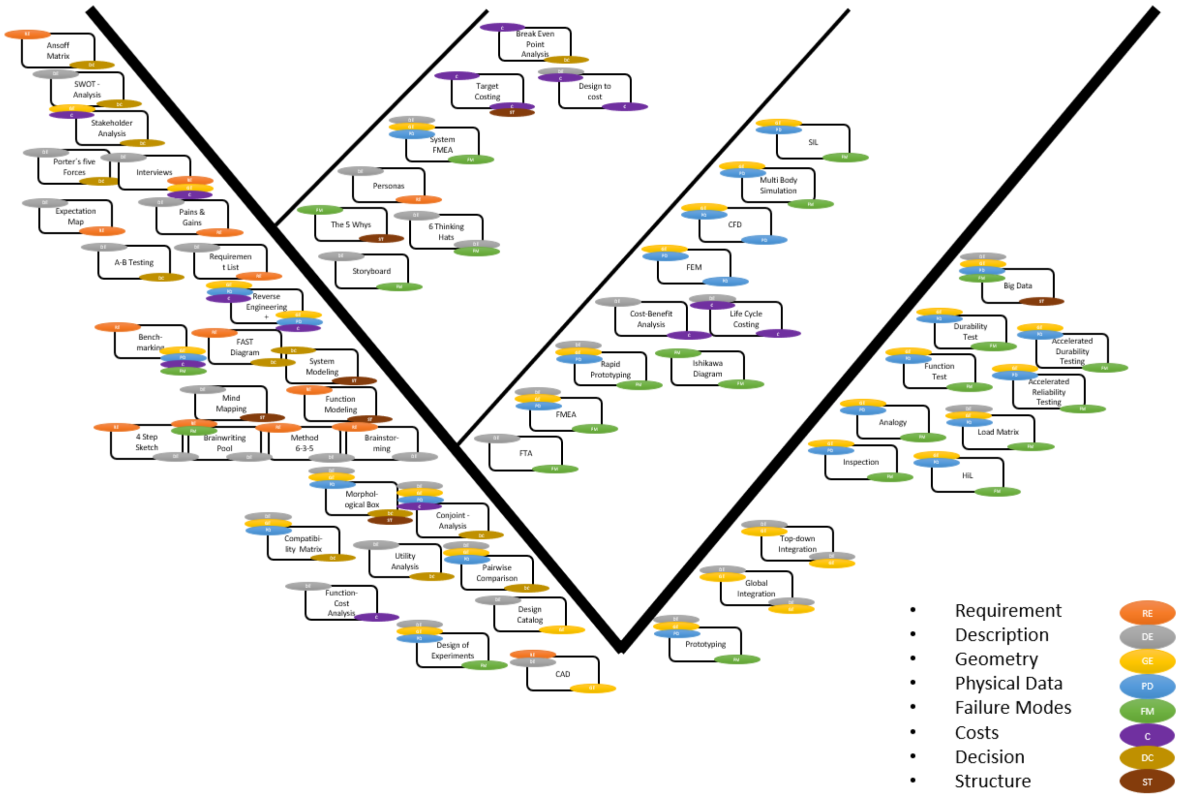


Figure 4-12: : V-model with in- and output information of methods

The classes, from which information is needed for the input, are positioned on the upper left corner of the method-button. On the bottom right corner, the classes which are associated with the output, are displayed.

Figure 4-13 shows the method *System FMEA* as an example. With the *System FMEA,* weaknesses of the system design should be identified early on. By implementing the measures derived from this method, the *System FMEA* contributes to increasing system security, reliability and availability.[83] In this case, the information classes for the input are *Description*, *Geometry*, and *Physical Data*. The *System FMEA*, to make accurate predictions about possible system weaknesses, needs a clear description of what the system is doing, what the rough geometry is, and what loads the system has to handle. The quality of the output information heavily depends on the maturity of the input information. If the method is processed accurately, possible failure modes of the system and their occurrence possibilities are discovered.

[83] Bertsche 2008.

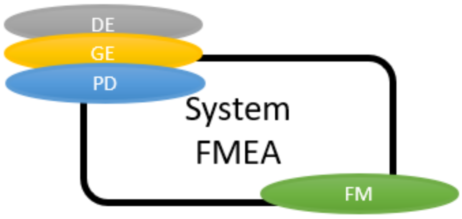


Figure 4-13: System FMEA in- and output information classes

In- output viewer

To better view all methods that contribute to the maturity of the same information class, a power point model with filter was developed. The target is to show all methods that either need input or generate output information for a specific information class at once. Thus, companies are able to look at methods of certain aspects of the product where they have problems. If for example, problems occur in detecting failure modes, the project team is able to (with the in-output viewer) have a look at all methods which either need failure modes as input information or generate output regarding failure modes. This overview, in comparison to the own set of methods, helps to define the room for improvement.

Figure 4-14 shows the principle of the in- output viewer without the names of the methods. At the start, the viewer displays the V-model as in Figure 4-12. The user now decides which information class has to be investigated in detail. If the button next to the desired information class in the legend is pressed, only the methods which either use input or generate output information for that class appear on the screen.

The example shown in Figure 4-15 displays a principle of all methods that are associated with the information class *Failure Modes.* All views of the in- output viewer are shown in appendix B.

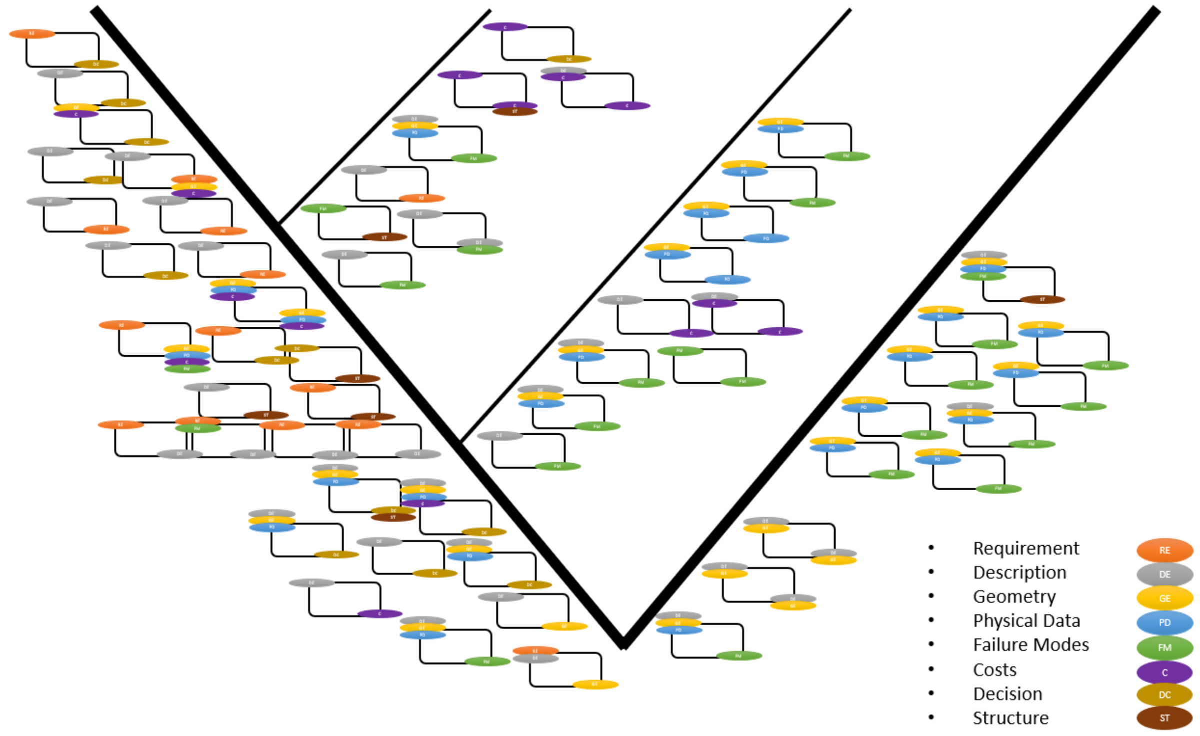


Figure 4-14: Principle of in- output viewer

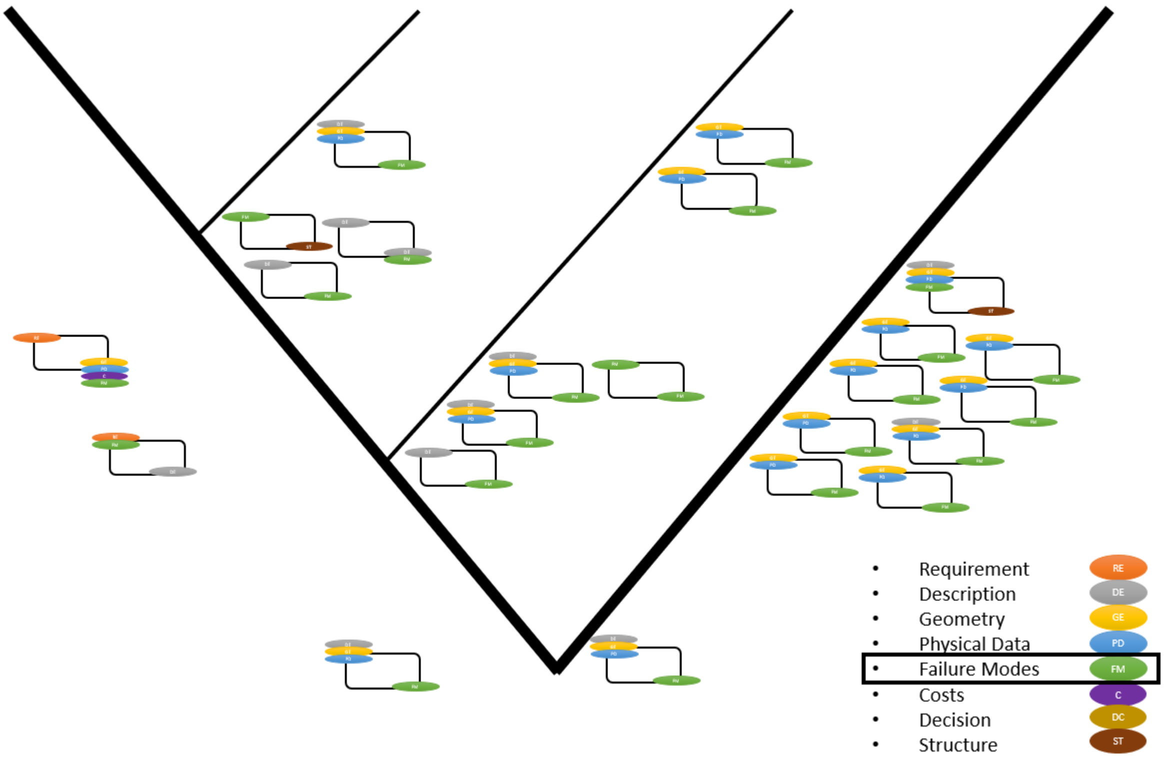


Figure 4-15: Principle of in- output viewer for the information class *Failure Modes*

5 Practical Approach

In this chapter the possibilities, of how the discussed models are applied in practice, are described. After a general explanation of the steps which are taken in a practical applications, three different positions in the product development process, of a not properly working exhaust valve in a car, are discussed. In the following examples the *Institute of Machine Components and Methods of Development* (IME) and the *Institute of Innovation and Industrial Management* (IIM) work together as an external provider for product development support. Considered companies outsource certain tasks to the institutes or use them for consulting.

5.1 Working principle

The main focus of this thesis are methods in the field of mechanical engineering. Nevertheless, the majority of methods are used universally and across divisions. The following sections explain the different steps which have to be proceed, in order to maximize the chance of generating the desired output of the development process.

5.1.1 Gather information

In general, if a business cooperation starts, a kickoff meeting is held. Project kickoff meetings must inform involved parties about the mission ahead of them. Project participants tend to be more effective if they understand the higher-level project targets and deliverables. Kickoff meetings are conducted in person or virtually.

Tres Roeder defines the three primary goals of the kickoff meeting as followed:[84]

1) Break the ice with project participants and begin building relationships.
2) Share the project scope, budget, and key deliverables with the team.
3) Communicate to each project member their role in achieving the project deliverables.

The third goal is key since there are three different options for a company on how tasks in the product development process are typically executed:

[84] Roeder 2013. p56

- *Done by the company itself*: possible if the knowledge and resources are available within the company and it is manageable economically.
- *Consulting*: to support the company complete certain tasks by leveraging their own expertise and experience together with collective expertise, experience and assets of the consulting organization, acting as a trusted adviser.[85]
- *Outsourcing*: describes the transfer of project tasks to external service providers. Outsourcing activities have grown as companies decided that they operate more effectively if they focus on their areas of core competence. When making outsourcing arrangements, it is essential that the company and the individuals responsible for managing the arrangement remember their responsibilities to their clients (activities can be delegated, but responsibility cannot).

As an external partner it is important for the two institutes, IME and IIM, to clearly define their responsibilities within the product development process with the customer. Therefore, it has to be communicated for which phases the customer requires consulting and which tasks are outsourced. Figure 5-1 displays a possible division of tasks in the product development process which supports the negotiation.

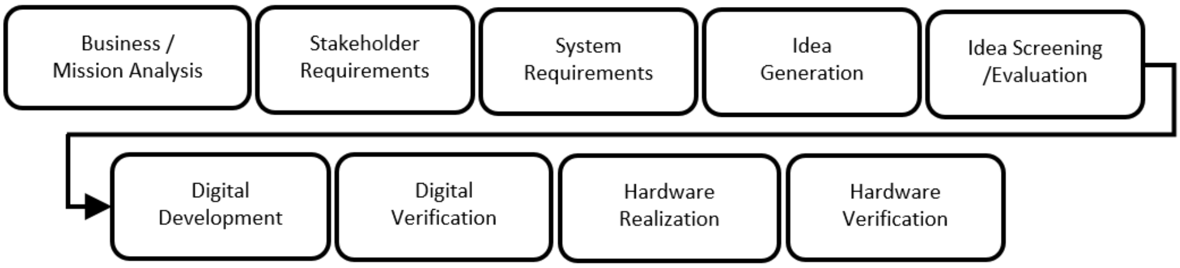


Figure 5-1: Example for division of tasks in the product development process

5.1.2 Propose methods

If the tasks of the institutes are defined, questions need to be asked in order to select the proper methods for the development process. For the best possible selection, general as well as open and closed (yes/no) specific questions about the product have to be answered by the customer. The differentiation between open and closed questions derives from the goal to develop a software which fully automated generates a chain of methods for any specific project (chapter 8). First of all, general - and specific open

[85] Parikh 2015 p6

questions have to be asked in order to get an overview of the situation of the project. Those questions are the basis for the specific closed questions. All closed questions are linked to methods. By answering the closed questions, the chain of methods emerges. In the following paragraphs some exemplary questions for each type are discussed.

General open questions are:

- How many *Full Time Equivalents* (FTE) are working on the project?
- How much time is scheduled for the project?
- How can the work of the consultants be integrated into the company?
- How are processes synchronized?
- Who are the contact persons in the organization?
- How does data exchange happen?
- What milestones have to be passed at what point of the project?
- Etc.

These and other general questions provide a structured approach for the consulter of which methods at which specific position in product development process are necessary. Depending on the time planned for the project and the number of FTEs involved, the number of possible methods increases or decreases. Extensive methods may not fit into the timeframe of certain projects and therefore have to be replaced with other methods. A different method might be faster and simpler to use but generates a less precise output. The consequences of possible information loss have to be considered when selecting the right methods and defining the timeframe for the project.

Specific open questions are:

- What are the main functions of the system?
- What is the added value for the customer?
- What is the desired durability?
- What is the use case of the system?
- Definition of the system environment?
- Etc.

As mentioned before, the target of the questionnaire is to answer as many closed questions as accurate as possible. Closed questions are assigned directly to the choice between methods. If it is possible to assess the situation exclusively with closed questions, a suggested chain of methods is generated automatically. Exemplary closed questions which has to be answered with yes or no are:

- Is the market known?
- Is it certain that the new product is needed/accepted by the customer?
- Is there a comparable system?
- Are the requirements defined?
- Are the conditions under which the system is used clear?
- Etc.

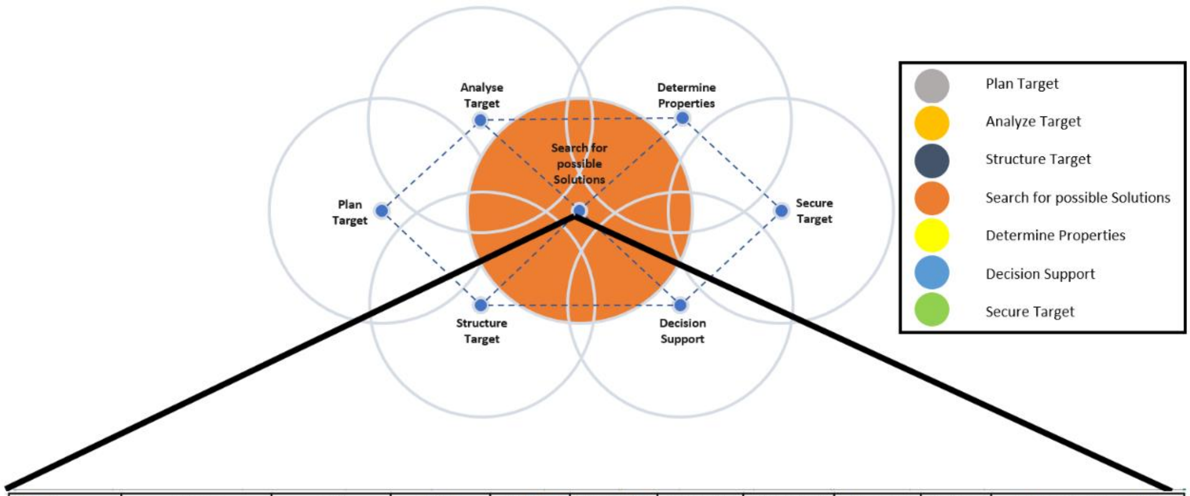
As an example, the following question is discussed: Are the requirements defined? If this question is answered with a no by the customer, the method *Requirement List* has to be added to the chain of methods. If the answer is yes, the method *Requirement List* doesn’t need to be added to the process.

After the questionnaire, the chain of methods is generated. In the future this chain is going to be generated automatically with a software program (chapter 8). Until then, the methods are selected manually by experts of the two institutes according to the answers of the questionnaire.

5.1.3 Adaption of the process with the MVM

During project execution a lot of change may happen due to changing circumstances. Therefore, the suggested methods might not generate the desired output. In that case, the initial chain of methods has to be adapted. A situation like that would be if an idea generation method like *Method 6-3-5* was executed and after further evaluation, none of the ideas or combination of ideas are suitable to solve the problem at hand. If that happens, another idea generating method (e.g. *Brainwriting Pool*) has to be added to the chain of methods.

When a situation occurs, where the output of the previous method is not sufficient to enable the execution of the next method, the *Münchner Vorgehensmodell* is to use. As discussed in chapter 3.5 the MVM groups the methods in functions. Depending on where in the process the *flow of information* is stopped, the function in the MVM is chosen. The example from the previous paragraph is shown again in Figure 5-2. That particular list of methods occurs when the MVM is scanned for the function *Search for Possible Solutions*.



Method	Purpose	Input	Output	Qual. / Quan.	Difficulty (1-5)	Participants	# of Participants	Sys. Input	Sys. Output	Münchner Vorghensmodell
Brainstorming	Generate a variety of ideas	Specification sheet, idea, problem, need	creative variety of ideas		1	Design team, partners, community members	4~6	Requirements	Description	SEARCH FOR POSSIBLE SOLUTIONS
Method 6-3-5	Generate a variety of ideas	Specification sheet, idea, problem, need	wide variety of ideas		1	Design team	6	Requirements	Description	SEARCH FOR POSSIBLE SOLUTIONS
4 Step Sketch	creates ideas and concepts in a step-by-step process	Specification sheet, idea, problem, need	ideas / concepts		1	Design team	2~6	Requirements	Description	SEARCH FOR POSSIBLE SOLUTIONS
Mind Mapping	A graphical way to represent ideas and concepts	Idea to develop	visual representation idea and related subtopics		1	Design team	2~5	Desire/ Description	Description / Structure	SEARCH FOR POSSIBLE SOLUTIONS
Brain Writing Pool	Generate a variety of ideas	Specification sheet, problem, need	Variety of ideas		1	Design team	5~8	Desire / Requirement / Failure Mode	Description	SEARCH FOR POSSIBLE SOLUTIONS

Figure 5-2: Methods located in *Search for possible Solutions*

Out of that list an additional method is selected to enable the subsequent method to continue. The next chapter discusses some different use cases to clarify the application of the method chain in combination with the MVM.

5.2 Car exhaust valve scenarios

This chapter discusses three different situations in reference to the exhaust valve of sports cars. The initial situation is the same for all three scenarios: The exhaust valve makes unwanted side noises expensive sports cars. In the next sections this problem is viewed from three different perspectives.

After the kickoff meeting in every scenario the two parties have to agree on the division of tasks. In addition to the three options already discussed in chapter 5.1.1 (done by the company itself, consulting and outsourcing), two additional options (not needed in this project, already done) are added clarify the status of each task in the development process.

5.2.1 Scenario 1: Sports cars customer

Initial situation:

A customer who has purchased multiple sports cars is not satisfied with the side noises of the exhaust valve in the cars. The customer complained at the OEM about the problem, but they never solved it. After several complaints the decision was made, to tackle the problem by himself. The customer has no experience in the automotive industry and therefore hires five engineers to work on an exhaust valve which keeps the side noises to a minimum. In addition, he asks for support from the two institutes IME and IIM.

The motivation for the customer is not to develop an exhaust valve which is ready to go into mass production, but to develop and produce a small number of products which are going to be installed in the owned vehicles only.

Involvement of Institutes:

The two parties agree on the following collaboration (Figure 5-3):

- Consulting the project team from project start until *Digital Verification*
- Outsourcing of *Hardware Realization* and *Verification* to the institutes

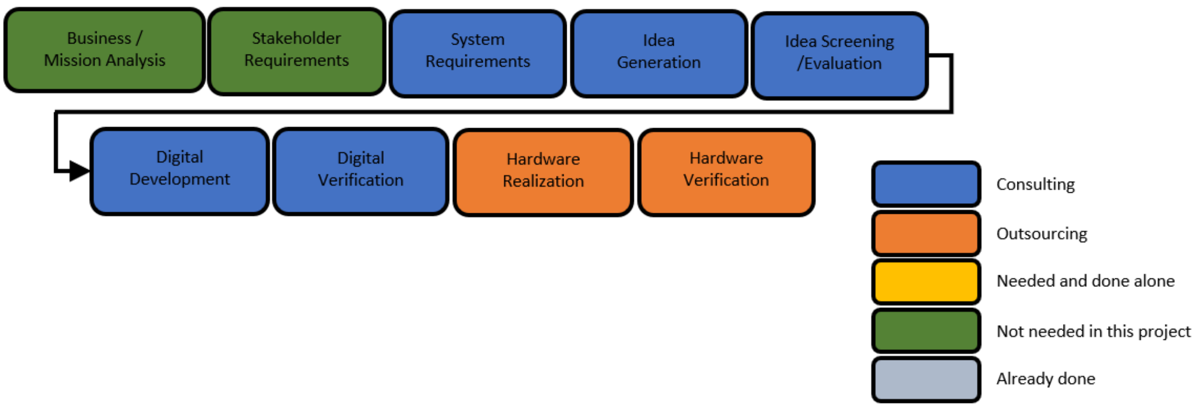


Figure 5-3: Tasks in the product development process colored according to scenario 1, sports cars customer

The first two tasks in the process are green, which means that they are not needed in this project. Since the customer is the one paying for the project and has no intention of bringing the improved exhaust valve to the market, there are no other wishes to consider than his.

Chain of methods:

If the involvement of the institutes is clear, the questions, which enable an accurate method selection, are discussed with the project leader. The ensuing suggested chain of methods is displayed in

Figure 5-4. Next to the methods, the purpose is described briefly. If two methods are displayed like *Benchmarking* and *Reverse Engineering* + in the Figure, it means that the first method is executed and if the output is not sufficient to enable the start of the following method, the other one is executed as well.

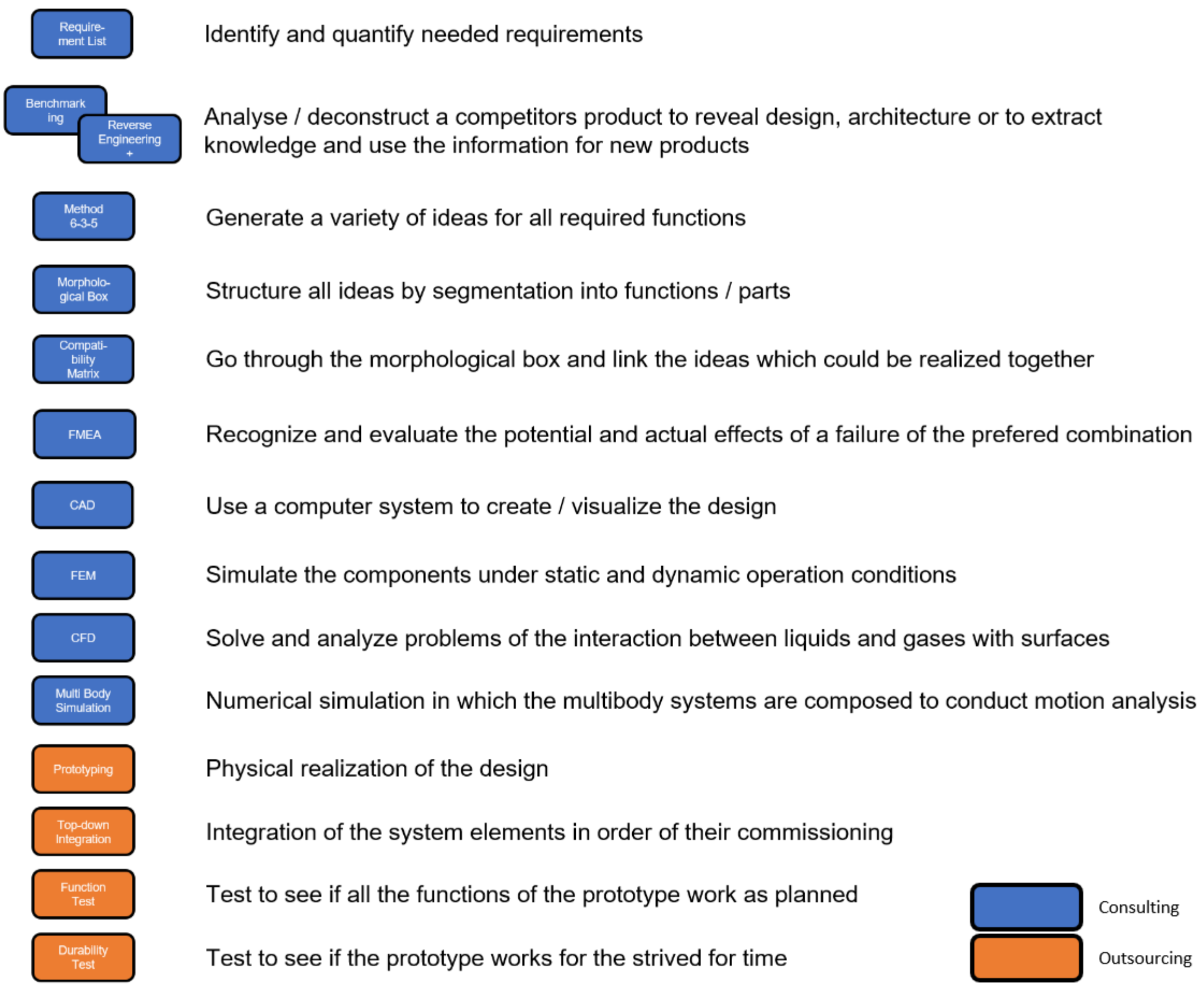
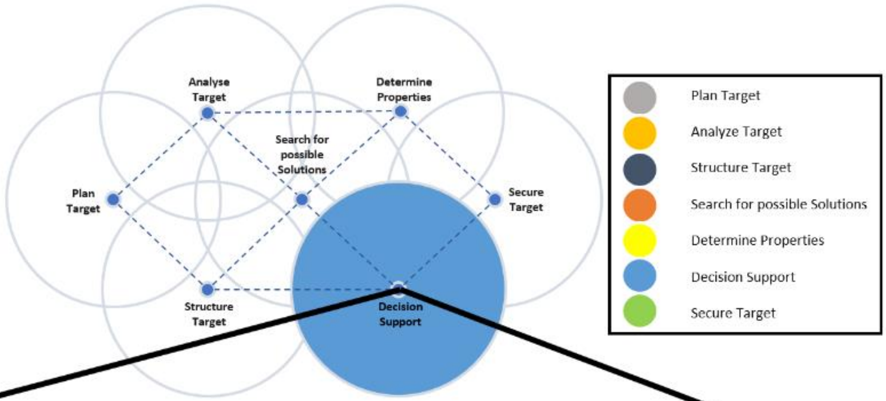


Figure 5-4: Chain of methods for scenario 1, sports cars owner

Adaptions:

While executing the methods a problem occurs. After the method *Compatibility Matrix* multiple options seem plausible. To continue with the process the team has to decide for one idea. To support that task the MVM with aligned methods is used. In the function D*ecision Support,* the methods which support decisions are listed (Figure 5-5).



Method	Purpose	Input	Output	Qual. / Quan.	Difficulty (1-5)	Participants	# of Participants	Sys. Input	Sys. Output	Münchner Vorgehensmodell
A-B Testing	selection of the user prefered concept	two or more testable concepts	user preference for one concept		3	Design team	2~5	Description / Prototype	Decision	DECISION SUPPORT
Cost-Benefit Analysis	decide whether to pursue an idea or not	idea / concept	worth pursuing or not		3	Design team	4~6	Description	Costs / Decision	DECISION SUPPORT
Function-Cost-Analysis	costs are being assigned to the individual functions	Detail Design, Development and production costs	Function and component attached costs		2	Design team, Production, Procurement	4~6	Description	Costs / Decision	DECISION SUPPORT
Pairwise comparison	To compare different factors systematically	Criteria, Functions, Operations	Priorization of criteria		2	Design team	4~6	Geometry / Phys. Data / Description	Decision	DECISION SUPPORT
Utility Analysis	used to make complex decisions. The total problem which is to be decided on, is broken down into sub-problems.	Different Concept Designs	Priotization of a concept		2	Design team	4~6	Description	Decision	DECISION SUPPORT
Compatibility Matrix	Matrix for complete pairwise comparison of elements in terms of their compatibility	Set/variety of partial solutions	possible combinations for final solution		2	Design team	3~6	Description / Geometry / Physical Data	Decision	DECISION SUPPORT
Conjoint - Analysis	Helps determine how people value different attributes that make up an individual product or service	Different combinations of possible partial solutions	prefered combination of attributes of the customer		3	Design team and customer	4~6	Description / Geometry / Physical Data / Costs	Decision	DECISION SUPPORT

Figure 5-5: Methods located in Decision Support

From that list the method *Utility Analysis* is selected to decide which idea is chosen going forward. After executing all proposed methods, the new designed exhaust valve works without significant side noises.

5.2.2 Scenario 2: Exhaust valve supplier

Initial situation:

The exhaust valve supplier receives complaints from the OEM about the product. The OEM explains, that customers have been complaining about unacceptable side noises of the exhaust valve during driving. The OEM threatens, that if the problem is not fixed, their contract is canceled. In order to keep the business, the exhaust valve has to be fixed fast. The supplier has years of experience in the field of exhaust valves and therefore some new design ideas. Nevertheless, the supplier reaches out to the two institutes IME and IIM for support on the backend of the development process. The motivation is not only to keep the contract with the OEM but also to not lose his reputation in the industry.

Involvement of Institutes:

The supplier is very experienced with the development of exhaust valves and is going to execute the tasks until digital design on his own. The task of *Digital Verification* is outsourced to the institutes. In addition, the supplier asks for consulting on *Hardware Realization* and *Hardware Verification* (Figure 5-6).

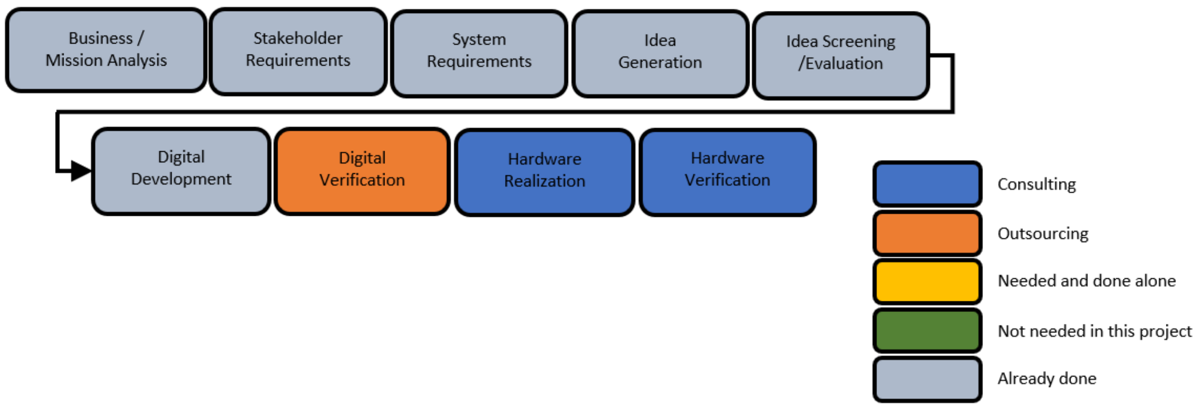


Figure 5-6: Tasks in the product development process colored according to scenario 2, exhaust valve supplier

Chain of methods:

The chain of methods that is generated for that specific project is shown in Figure 5-7. Since the cooperation didn't start at the beginning of the project, it is important to communicate certain standards the digital design requires in order to enable an efficient transition from D*igital Development* to D*igital Verification*.

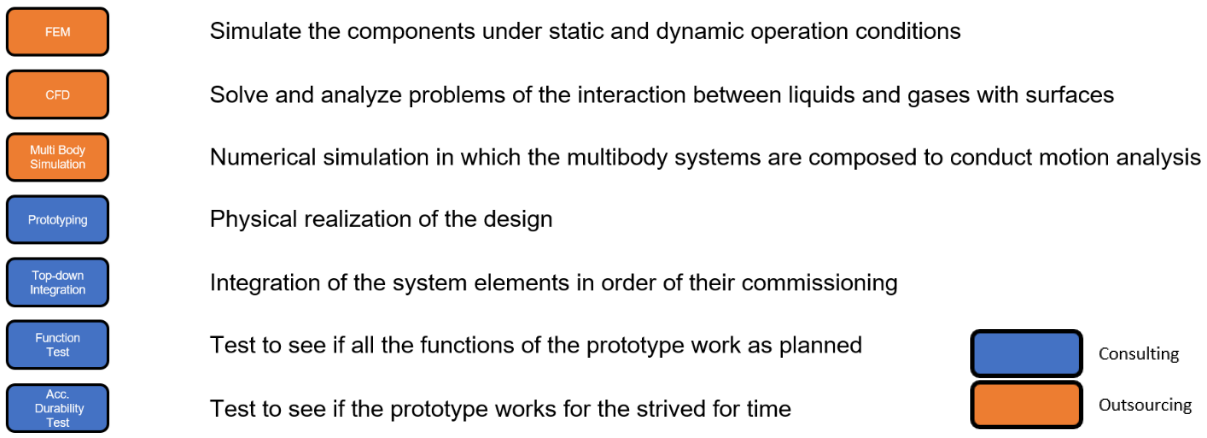


Figure 5-7: Chain of methods for scenario 2, exhaust valve supplier

5.2.3 Scenario 3: Original equipment manufacturer (OEM)

Initial situation:

The OEM received multiple complaints from customers due to loud side noises at the exhaust valve. Even if the problem with the exhaust valve is not primary the fault of the OEM, the OEM is responsible for the final product. The problem has to be fixed in order to keep the reputation and sales volume. The OEM held meetings with his supplier to address the problem but after some time for iteration, the side noises were still at a level that was not acceptable. Weighing in on options, the OEM thinks about insourcing the production of the exhaust valve. Since this is a decision with major implications, the OEM wants to gather some rough ideas to see if insourcing is economically feasible.

Involvement of Institutes:

Since the quality of the supplier is not adequate, the OEM thinks about manufacturing the exhaust valve itself. From years of experience, the system requirements are defined. The idea generation and screening process is outsourced to the institutes (Figure 5-8). The OEM expects some rough ideas to solve the problem. The output should help to assess if insourcing the exhaust valve is economically feasible or not.

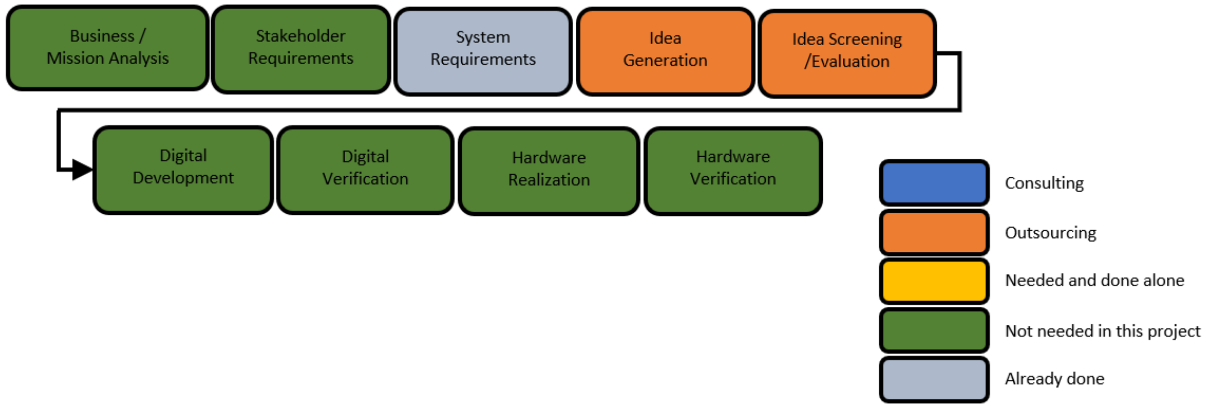


Figure 5-8: Tasks in the product development process colored according to scenario 3, original equipment manufacturer

Chain of methods:

A team of 3-4 FTE is able to execute the chain of methods for scenario 3 (

Figure 5-9) in two to three days. The target of that process is not a detail design of one option but an overview of possible mechanisms and designs.

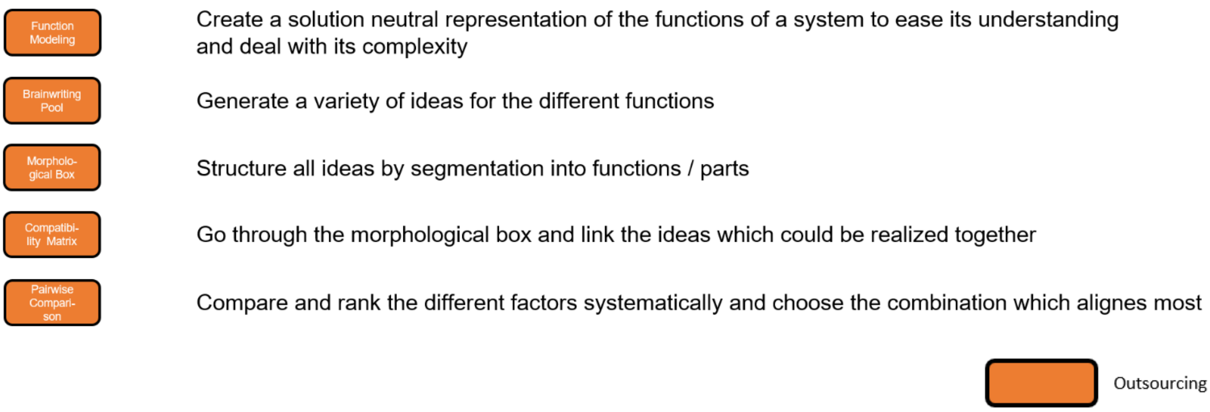


Figure 5-9: Chain of methods for scenario 3, original equipment manufacturer

6 Discussion

Today a paradigm shift to smarter products has happened and complicated product design and development. The need to coordinate mechanical, electronic and software components not only complicates product development, but it also can lead to launch delays and rising costs and risks when software or design changes are not effectively and accurately coordinated across disciplines.[86] Following that increase in complexity *Orphey* states, that the knowledge today doubles over a period of only 5 years. This increase in knowledge inevitably leads to a more complex problem-solving process. In addition, the number of external influences that need to be considered increase. Figure 6-1 displays internal and external factors that influence the ability to innovate and compete for organizations.[87]

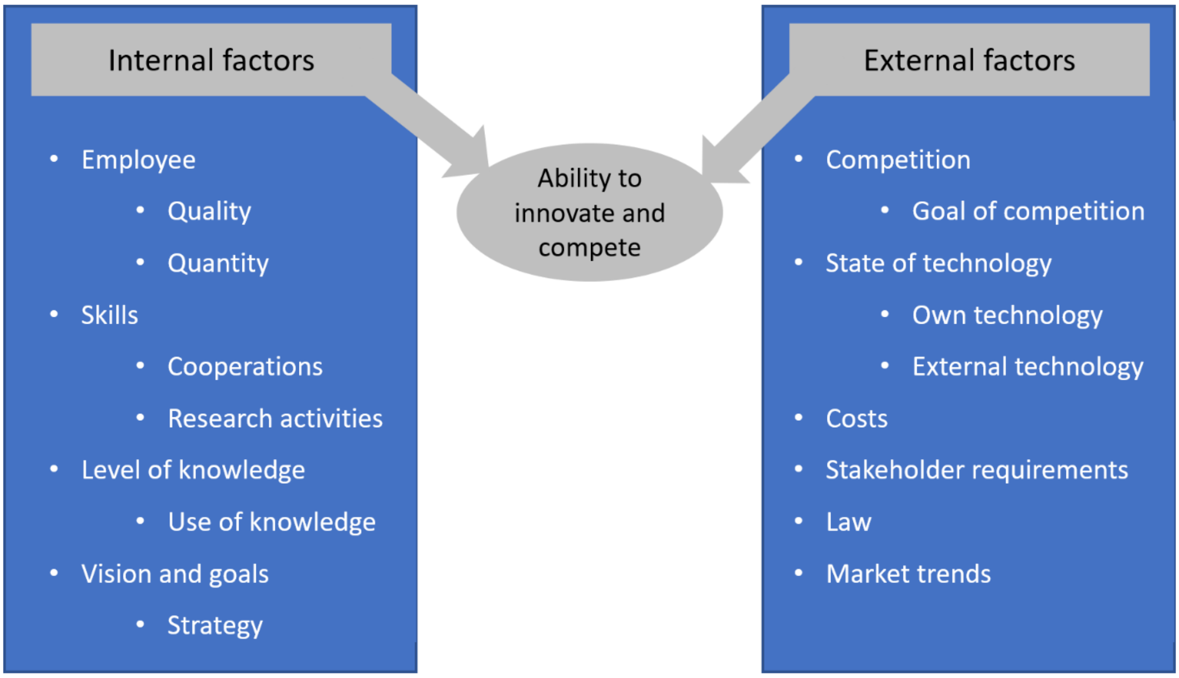


Figure 6-1: Factors of innovative and competitive ability inspired by Ophey[88]

All those influencing factors ultimately lead to the fact that product development processes are becoming more difficult to control. However, today there are a lot of different methods to support individual phases of complex processes. These methods support

[86] Greenstein 2013. p1556
[87] Ophey 2005. p3
[88] Ophey 2005. p4

the user to systematically question the current problem and work on a consequent and logical way to solve it.[89]

As mentioned in chapter 2.1, the models presented in this thesis do not demand any specific development philosophy. No matter what philosophy is chosen for the product development process, the key is to consider iterations. Considering iterations in product development is a valuable response in order to cope with the complexity of product and service development. Iterations may be pre-planned on the one, or event-driven on the other hand. Mastering both kinds of iterations in an integrated manner provides agility to the execution of the product development process.[90] Figure 6-2 displays three different kinds of possible iterations whereas the dashed lines represent different architecture levels.

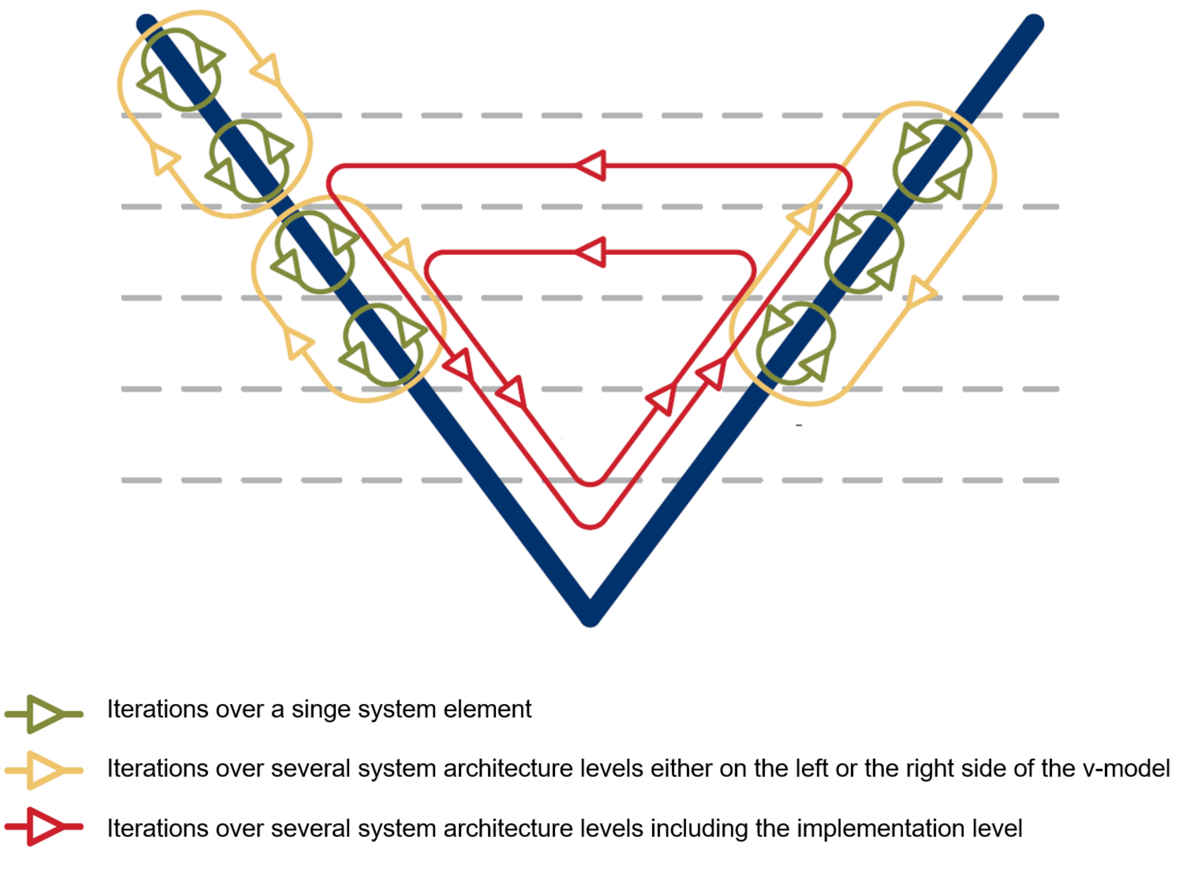


Figure 6-2: The dynamic V-model by *Scheithauer*[91]

[89] Ophey 2005. p4
[90] Scheithauer und Forsberg 2013. p512
[91] Scheithauer und Forsberg 2013. p512

Whenever feedback loops cross the implementation level the effort is going to be high. Consequently, the color red was chosen in the figure. The only exceptions are systems for which the implementation effort is low (e.g. some software applications). In all other cases, feedback loops of this kind should be avoided if possible or minimized in number. [92]

Chapter 2.2 displays how processes, methods, and tools are embedded in the system lifecycle. The main focus of this section is on defining the terminology for processes, methods, and tools. By analyzing the current state, it becomes obvious that there exist many different definitions of the terms process, method, and tool. There are several reasons for that. The terms are used in face-to-face communication, often resulting in a loss of exact meaning due to human language uncertainty and multiple possible interpretations. Transferring this basic thought to technical development, further different interpretations of the terms are in use. It is important for this thesis to develop a framework, which enables common understanding of processes, methods, and tools to support interdisciplinary communication. The fact that there is no common definition in literature shows, that those terms are not clearly to separate and define. Using the interrogative pronouns *HOW* and *WHAT* to describe the terms, help to provide an as clear distinction and understanding as possible. That definition doesn’t exclude an interleaving of processes and methods on different levels which often is the case.

The elementary view of methods described in chapter 0 primarily is an attempt to make the selection of methods processable. Computers offer a wide range of opportunities to improve the product development process because of the amount of data that is processed in a short amount of time. At this stage it is difficult to describe some elementary criteria as concrete as it would be necessary in order to generate the desired output. Nevertheless, those criteria provide (at least in comparison to each other) a structured overview of development methods. The one pager gives a more detailed view of the method and therefore concludes the overview of methods.

As mentioned before, the main problem in preparing this thesis was displaying the methods. Most models only allow a linear display which is not satisfactory if the aim is to generate a generic, logical and structured display of methods. The solution is to use the V-model and the MVM in combination. With that combination the advantages of

[92] Scheithauer und Forsberg 2013. p514

the V-model like the display of virtual/real development, emphasis on verification, and the change of level of detail are combined with the functional structure of the MVM which compensates the problem of linearity of the V-model.

In this thesis 58 methods have been analyzed. That number derived from the objective to display enough methods to make the use of the developed models comprehensible and at the same time not go beyond the scope of a master´s thesis. A lot of disciplines have to work together properly in order to develop a competitive and successful product (e.g. marketing, project management, technical development, controlling, logistics, purchasing, etc.). 58 methods of course cannot cover all those disciplines. The goal of the thesis is to develop a framework which is applicable and extendible to all disciplines which are involved in product development. As a start, the focus of the methods used in this thesis is on technical development. Exemplarily a few methods from other disciplines are displayed as well.

A critical point is the arrangement of methods. One may argue that some methods should be placed at another point of the V-model or even in another function of the *MVM*. The intention was to place the methods to the area where they are most commonly used (which is at least partly subjective). There is no *one true display* of the methods. Some methods may be moved for certain applications. Adaptions are allowed as long as the basic definitions of the V-model are still met.

The Information classes discussed in chapter 3.6 and displayed in chapter 4.5 intend to give an overview of what kind of information enters (input) and exits (output) the methods. The eight information classes requirement, geometry, physical data, failure modes, costs, decision, and structure were selected to show a continuous flow of information in the development process. Those eight classes are still very generic. They could be decomposed into subclasses to display the information in more detail and therefore improve their informative value. If the class physical data is decomposed, several new classes occur (e.g. kinematic, mechanic, thermodynamic, electrodynamic, etc.). If those classes are split up further, more than one hundred classes exist. Therefore, it is important to define what level of detail is relevant for the process in order to get specific enough information but not spend too much time on the development and documentation of the system itself.

To get the most out of the models developed in this thesis, it is critical to assess the initial situation as detailed as possible. A combination of open and closed questions support the selection of the right methods for a specific development situation. In the practical part the usage of the V-model in combination with the MVM is explained and demonstrated. In a standard application the MVM is only used if the proposed chain of methods from the V-model is not sufficient to keep the flow of information continuing.

7 Conclusion

The target of the thesis was the development of a generic overview of methods typically used in the product development process and to support the selection of suitable methods for product development situations. The methodical know-how of two institutes(*Institute of Machine Components and Methods of Development* (IME) and the *Institute of Innovation and Industrial Management* (IIM)) was used to discover development methods from innovation activities to the start of production in the product lifecycle. The two institutes methodically cover most parts of the discussed development process, but some phases are not covered very profound. For that phases methods from literature are added.

The V-model, considered as a combination of the top-down and bottom-up approach, presents the opportunity to display methods regarding their different levels of detail they operate in, from abstract overall system considerations and analysis to detailed investigations. An added value of using the V-model as base, is the opportunity to display integration, verification and validation methods on the opposite side of requirements and specification methods and therefore enable the visibility of these important dependencies.

The addition of the *Münchner Vorgehensmodell* offers the possibility to be more flexible in choosing the sequence of methods for the product development process. In case of necessary additional methods e.g., there are difficulties to make a decision, the MVM provides a recommendation of alternative methods. In order to provide that flexibility, the methods in the MVM are listed according to the functions they serve (e.g., plan target, search for possible solutions, secure target).

By implementing information classes, which group methods according to their generating output information, in addition to the two discussed procedure models, a new possibility of how to tackle difficulties in the product development process is offered. If difficulties occur within a certain information class, all methods across different phases which generate information concerning that class, can be displayed.

The use of the procedure models discussed in this thesis offer the chance to improve the product development process by discovering development methods in various development phases. Methods may be chosen for a specific development situation or suggested for the whole development process. Either way the efficiency of the development process increases due to a methods based, target oriented, structured approach.

8 Outlook

Over hundred years ago, in 1902 *Elbert Hubbard* stated:

„A machine can do the work of fifty ordinary people, but it cannot replace a single extraordinary one. "

-Elbert Hubbard

At the start of working on this thesis the focus was on analyzing methods which support the development of technical products. The target was to develop models and principles which are universally applicable for different technical development processes. In the first step, methods, with the focus on technical development, were aligned to the V-model and the MVM. In the next steps on the one hand methods for technical development could be enhanced and on the other hand more disciplines could be developed methodically. Disciplines which may be considered are marketing, project management, controlling, purchase, quality, logistics, and production. Figure 8-1 shows a part of a V-model and a possible form the different disciplines could be displayed in future.

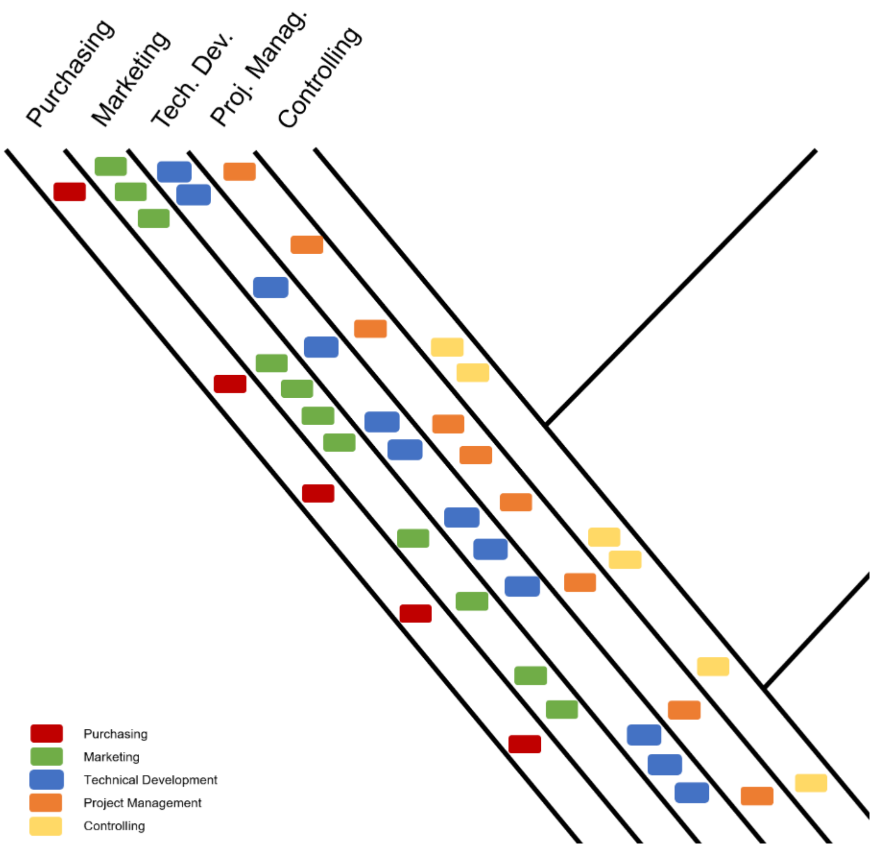


Figure 8-1: Possible future view with disciplines combined

In a display like Figure 8-1 the chronological order of the methods from different disciplines is shown precisely. Therefore, the different departments are able to improve the coordination of activities within the company .An overall goal in future is to develop a so-called *Method Chain Generator.* This generator automatically proposes a chain of methods for a certain development situation. Therefore, it is necessary to make all the information (about the development situation), which is inserted in the generator, processable. Closed questions are essential for that task and thus a precise and field-tested questionnaire has to be developed. As mentioned in the quote at the beginning of the chapter a machine can do the work of fifty ordinary people, but it cannot replace a single extraordinary one. This describes the fact that computers are able to process a huge amount of information in a short amount of time, but there will always be some information which cannot be processed by a computer (e.g. emotions). Thus, the proposed chain of methods, even with the best software possible, might need adaptions from an expert who assesses the situation as a whole.

The in- and output classes discussed in this thesis are held general until this point. In the next steps the classes could be extended to the different disciplines and decomposed into more detail. If the level of detail, between specific enough to offer the desired information and generic enough to not spend more time on working on the model than it creates additional value, is met in future, the *In-Output Viewer* will become a beneficial tool in product development. Companies may buy a future software application of the viewer or reach out for support to consultants.

List of Figures

List of Tables

List of Literature

Avgeropoulos, Stephanos (2014): Stakeholder Analysis. In: Cary L. Cooper (Hg.): Wiley encyclopedia of management, Bd. 20. 3. ed. Chichester, West Sussex: Wiley, S. 1–2.

Bajzek, Matthias (2018): Entwicklungsmethodik. Systems Engineering. Institut für Maschinenelemente und Entwicklungsmethodik, 2018.

Banks, Catherine M.; Sokolowski, John A. (Hg.) (2009): Principles of modeling and simulation. A multidisciplinary approach. Hoboken, N.J: John Wiley.

Beedle et al: Manifesto for Agile Software Development. Online https://agilemanifesto.org/iso/en/principles.html, 29.01.2019.

Bertsche, Bernd (2008): Reliability in Automotive and Mechanical Engineering. Determination of Component and System Reliability. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg (VDI-Buch). Online http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10281321.

Blanchard, Benjamin S.; Blyler, John E. (2016): System engineering management. Fifth edition. Hoboken: Wiley (Wiley series in systems engineering and management).

Braun, Thomas E. (2005): Methodische Unterstützung der strategischen Produktplanung in einem mittelständisch geprägten Umfeld. Zugl.: München, Techn. Univ., Diss., 2005. 1. Aufl. München: Techn. Univ; Verl. Dr. Hut.

Cambronero, M. Emilia; Valero, Valentín; Díaz, Gregorio (2010): Verification of real-time systems design. In: *Softw. Test. Verif. Reliab.* 20 (1), S. 3–37. DOI: 10.1002/stvr.405.

Cooper, Rachel (2005): Process management in design and construction. Oxford: Blackwell.

Czichos, Horst (2015): Mechatronik. Grundlagen und Anwendungen technischer Systeme. Wiesbaden: Springer Vieweg.

Denkmayr, Klaus; Hick, Hannes; Sauerwein, Ulrich (2003): Die Load-Matrix. In: *MTZ Motortech Z* 64 (11), S. 924–930. DOI: 10.1007/BF03227139.

Elich, Erik; Schreinemakers, Paul; Vullings, Maarten (2012): 5.1.1 Verification & Validation; An inconvenient truth. In: *INCOSE International Symposium* 22 (1), S. 650–662. DOI: 10.1002/j.2334-5837.2012.tb01362.x.

Estefan Jeff A. (2008): Survey of Model-Based Systems Engineering (MBSE) Methodologies.

Fuchs, Christoph; Schreier, Martin (2011): Customer Empowerment in New Product Development. In: *Journal of Product Innovation Management* 28 (1), S. 17–32. DOI: 10.1111/j.1540-5885.2010.00778.x.

Funke, Thomas; Noll, Reinhard; Niessen, Stefan; Weikl, Bruno (2000): Softwareentwicklung in mittelständischen Unternehmen mit ISO 9000. Berlin, Heidelberg: Springer (Xpert.press).

Geller, Bartholomaus (2014): Quantitative vs. Qualitative Methods. Online https://slideplayer.org/slide/672750/, zuletzt geprüft am 21.01.2019.

Graner, Marc (2013): Der Einsatz von Methoden in Produktentwicklungsprojekten. Eine empirische Untersuchung der Rahmenbedingungen und Auswirkungen. Zugl.: Cottbus, Techn. Univ., Diss., 2012. Wiesbaden: Springer Gabler (Springer Gabler Research).

Greenstein, Bret (2013): The Convergence of Engineering Disciplines in Modern Product Development. In: *INCOSE International Symposium* 23 (1), S. 1556–1561. DOI: 10.1002/j.2334-5837.2013.tb03108.x.

Griffin, Abbie; Luchs, Michael G.; Swan, Scott K. (Hg.) (2015): Design thinking. New product development essentials from the PDMA. Hoboken: Wiley.

Heerwagen, Mathias (2018): Entwicklung im Wandel Agile Methoden auf dem Vormarsch. In: *MTZ Motortech Z* 79 (4), S. 8–13. DOI: 10.1007/s35146-018-0016-8.

Hitchins, Derek K. (2007): Systems engineering. A 21st century systems methodology. Chichester, West Sussex, England, Hoboken, NJ: John Wiley (Wiley series in systems engineering and management).

INCOSE (2019): What is Systems Engineering? Online https://www.incose.org/systems-engineering, 28.01.2019.

ISO/IEC TR 24748-1 (2010): Life cycle management. Guide for life cycle management.

ISO/IEC/IEEE 15288:2015: ISO/IEC 15288.

Kossiakoff, Alexander (2011): Systems engineering. Principles and practice. 2nd ed. Hoboken, N.J: Wiley-Interscience (Wiley series in systems engineering and management, 67).

Lindemann, Udo (2009a): Methodische Entwicklung technischer Produkte. Methoden flexibel und situationsgerecht anwenden. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg (VDI-Buch).

Lindemann, Udo (2009b): Methodische Entwicklung technischer Produkte. Methoden flexibel und situationsgerecht anwenden. 3., korrigierte Aufl. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg (VDI-Buch).

Lindemann, Udo (2016): Handbuch Produktentwicklung. München: Hanser.

Medina, Casey (2015): The “V” Model Reloaded. In: *INCOSE International Symposium* 25 (1), S. 962–974. DOI: 10.1002/j.2334-5837.2015.00109.x.

Meyer, Bertrand (2014): Agile! The good, the hype and the ugly. Cham: Springer.

Mital, Anil; Desai, Anoop; Subramanian, Anand; Mital, Aashi (2014): Product Development. A Structured Approach to Consumer Product Development, Design, and Manufacture. 2nd ed. Burlington: Elsevier Science.

Moran, Alan (2015): Managing Agile. Strategy, implementation, organisation and people. Cham: Springer. Online http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&AN=969008.

Morgan and Obal (2016): Celebrating America’s Pastimes: Baseball, Hot Dogs, Apple Pie and Marketing? Proceedings of the 2015 Academy of Marketing Science (AMS) Annual Conference. 1st ed. 2016. s.l.: Springer-Verlag (Developments in Marketing Science).

Ophey, Lothar (2005): Entwicklungsmanagement. Methoden in der Produktentwicklung. Berlin: Springer (VDI). Online http://gbv.eblib.com/patron/FullRecord.aspx?p=323472.

Parikh, Samir (2015): The consultant's handbook. A practical guide to delivering high-value and differentiated services in a competitive marketplace. Chichester, West Sussex, United Kingdom: John Wiley & Sons Inc.

Parnell, Gregory S.; Driscoll, Patrick J.; Henderson, Dale L. (2011): Decision making in systems engineering and management. 2nd ed. Hoboken, NJ: Wiley (Wiley series in systems engineering and management).

Ponn, Josef; Lindemann, Udo (2008): Konzeptentwicklung und Gestaltung technischer Produkte. Optimierte Produkte - systematisch von Anforderungen zu Konzepten. Berlin, Heidelberg: Springer (VDI-Buch). Online http://d-nb.info/989986497/34.

Roeder, Tres (2013): Managing Project Stakeholders. Building a Foundation to Achieve Project Goals. 1. Aufl. s.l.: Wiley.

Scheithauer, Dieter; Forsberg, Kevin (2013): 4.5.3 V-Model Views. In: *INCOSE International Symposium* 23 (1), S. 502–516. DOI: 10.1002/j.2334-5837.2013.tb03035.x.

Schwankl, Ludwig (2002): Analyse und Dokumentation in den frühen Phasen der Produktentwicklung.

Simpson, Joseph J.; Miller, Ann; Grasman, Scott; Dagli, Cihan (2005): 1.3.2 A Generic, Adaptive Systems Engineering Information Model. In: *INCOSE International Symposium* 15 (1), S. 126–135. DOI: 10.1002/j.2334-5837.2005.tb00657.x.

Srinivasan, Venkat (2017): The intelligent enterprise in the era of big data. Hoboken, New Jersey: John Wiley & Sons Inc.

VDI 2206 (2004): Entwicklungsmethodik für mechatronische Systeme.

VDI 2221 (1993): Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte.

A List of Methods

Part 1

Method	Purpose	Input	Output	Qual / Quan	Difficulty (1-5)	Participants	# of Participants	Sys. Input	Sys. Output	Münchner Vorgehensmodell
4 Step Sketch	creates ideas and concepts in a step-by-step process	Specification sheet, idea, problem, need	ideas / concepts		1	Design team	2'6	Requirements	Description	SEARCH FOR POSSIBLE SOLUTIONS
6 Thinking Hats	Ideas are evaluated from different perspectives to find optimal solutions	idea / concept	Critical evaluation of the idea		2	Design team	4'7	Description	Failure Modes / Description	ANALYSE TARGET
A-B Testing	selection of the user prefered concept	two or more testable concepts	user preference for one concept		3	Design team	2'5	Description / Prototype	Decision	DECISION SUPPORT
Accelerarted Durability Testing	Accelerated test to see if the prototype works for the strived for time	Prototype	Failure modes		4	Mechanical development/ Verification team	depending on complexity of the system	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Accelerarted Reliability Testing	Accelerated t to see if all the functions of the prototype work	Prototype	Failure modes		4	Mechanical development/ Verification team	depending on complexity of the system	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Analogy	Comparing with already existing verification of comparable components	Prototype, verification of comparable components	Failure modes		3	Mechanical development/ Verification team	4'6	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Ansoff Matrix	Looks at the potenials and risks of four possible product-market combinations	internal and external research / benchmark	area to focus on		2	Management	4'6	Desire / Necessity	Requirement	PLAN TARGET
Benchmarking	Compares one's product / business process / performance to the industry's best	Product /process /area to improve	inspiration and data for improvement		2	Design team	4'6	Requirements	Geometry/ Phys. Data / Failure Modes / Costs	ANALYSE TARGET
Big Data	Improvement of data sets that are too large or complex for traditional data-processing application software to adequatly deal with	Data / Information	Improvement of data processing and information exchange		5	Overall	depending on complexity of the system	Description / Geometry / Physical Data / Failure Modes /	Structure	STRUCTURE TARGET
Brain Writing Pool	Generate a variety of ideas	Specification sheet, problem, need	Variety of ideas		1	Design team	5'8	Desire / Requirement / Failure Mode	Description	SEARCH FOR POSSIBLE SOLUTIONS
Brainstorming	Generate a variety of ideas	Specification sheet, idea, problem, need	creative variety of ideas		1	Design team, partners, community members	4'6	Requirements	Description	SEARCH FOR POSSIBLE SOLUTIONS
Break-Even-Point Analysis	Determines what you need to sell to cover your costs of doing business	Costs	Number of products I have to sell to cover my costs		3	Controlling team	2'4	Costs	Decision	ANALYSE TARGET
Computer Aided Design (CAD)	Use of computer systems to aid in creation, modification or optimization of a design	Concept of the design	Digital 3D model		3	Design team	2'4	Requirements / Description	Geometry	DETERMINE PROPERTIES
CFD	Solve fluid mechanical problems approximately with numerical methods	CAD file	fluid mechanical analyisi of the part		4	Design team	1'2	Geometry / Phys. Data	Physical Data	SECURE TARGET
Compatibility Matrix	Matrix for complete pairwise comparison of elements in terms of their compatibility	Set/variety of partial solutions	possible combinations for final solution		2	Design team	3'6	Description / Geometry / Physical Data	Decision	DECISION SUPPORT
Conjoint - Analysis	Helps determine how people value different attributes that make up an individual product or service	Different combinations of possible partial solutions	prefered combination of attributes of the customer		3	Design team	4'6	Description / Geometry / Physical Data / Costs	Decision	DECISION SUPPORT
Cost-Benefit Analysis	decide whether to pursue an idea or not	idea / concept	worth pursuing or not		3	Design team	4'6	Description	Costs	DECISION SUPPORT

Part 2

Design Catalogs	transfer known or stimulate new solutions by systematic assignment of known solution features to the conditions of the task.	Data base Specification sheet	Possible solution for design		2	Design team	4‘6	Description	Geometry	DETERMINE PROPERTIES
Design of Experiments	To determine the relationship between factors affecting a process and the output of that process	System	Cause and effect relationships		4	Design team	4‘6	Description / Geometry / Physical Data	Failure Modes	SECURE TARGET
Design to Cost	Systematic approach to controlling the costs of product development and manufacturing	Target costs	Costs are designed "into the product"		3	Design team / Controlling team	4‘6	Costs / Description	Costs	ANALYSE TARGET
Durability Test	Test to see if the prototype works for the strived for time	Prototype	Failure modes		3	Mechanical development/ Verification team	depending on complexity of the system	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Expectation Map	A visualization of customer jounrey emotions	Rough idea / Customer segment	Better understanding of customer desicions		1	Design team	4‘6	Description	Requirement	PLAN TARGET
F.A.S.T	develop a graphical representation showing the logical relationships between the functions of a project, process or product	specification sheet	logical relationships between functions		2	Design team	4‘6	Requirements	Description	STRUCTURE TARGET
FEM	Simulate components under static and dynamic operating conditions	CAD file	Load analyzed component		4	Design team	1‘2	Geometry / Phys. Data	Physical Data	SECURE TARGET
FMEA	recognizes and evaluates the potential and actual effects of failure of a product or a process	Concept / Layout / Detail design	identification of critical components and weak spots		3	Design team	4‘6	Geometry / Phys. Data / Description	Failure Modes	SECURE TARGET
FTA	systematic method of modeling different failures that can occur within a given system or component	Concept / Layout / Detail design	Malfunction and disturbance influences are being exposed		2	Design team	4‘6	Description	Failure Modes	SECURE TARGET
Function Modeling	Create solution neutral representations of the functions of a system to ease its understand and deal with its complexity	Determined requirements	solution neutral representation of the functions of the system		3	Design team	4‘6	Requirements	Structure	STRUCTURE TARGET
Function Test	Test to see if all the functions of the prototype work	Prototype	Failure modes		3	Design team / Mechanical development	depending on complexity of the system	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Function-Cost-Analysis	costs are being assigned to the individual functions	Detail Design, Development and production costs	Function and component attached costs		2	Design team, Production, Procurement	4‘6	Description	Costs	DECISION SUPPORT
Global Integration	All system elements are integrated in one big step	System elements	Combination of for the product relevant parts (prototype)		3	Mechanical development	4‘6	Description / Geometry	Description / Geometry	DETERMINE PROPERTIES
Hardware in the Loop	Test a complex real-time embedded system	Embeded system and HiL simulator	Verification of the embedded system		4	Design team / Mechanical development	4‘6	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Inspection	Visual examination of the element	Prototype	Failure modes			Design team / Mechanical development	4‘6	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Interviews	Understanding the hopes, desires and aspirations of your customers	Intention to serve a specific market segment	Better understanding of the customer		1	Design team, person you're designing for	2‘3	Desire	Requirement/ Costs/ Geometry	PLAN TARGET
Ishikawa Diagram	Visualization for catagorizing the potenital causes of a problem in order to identify ist root causes	Failure Modes	Root cause		2	Design team	4‘6	Failure Mode	Failure Mode	STRUCTURE TARGET

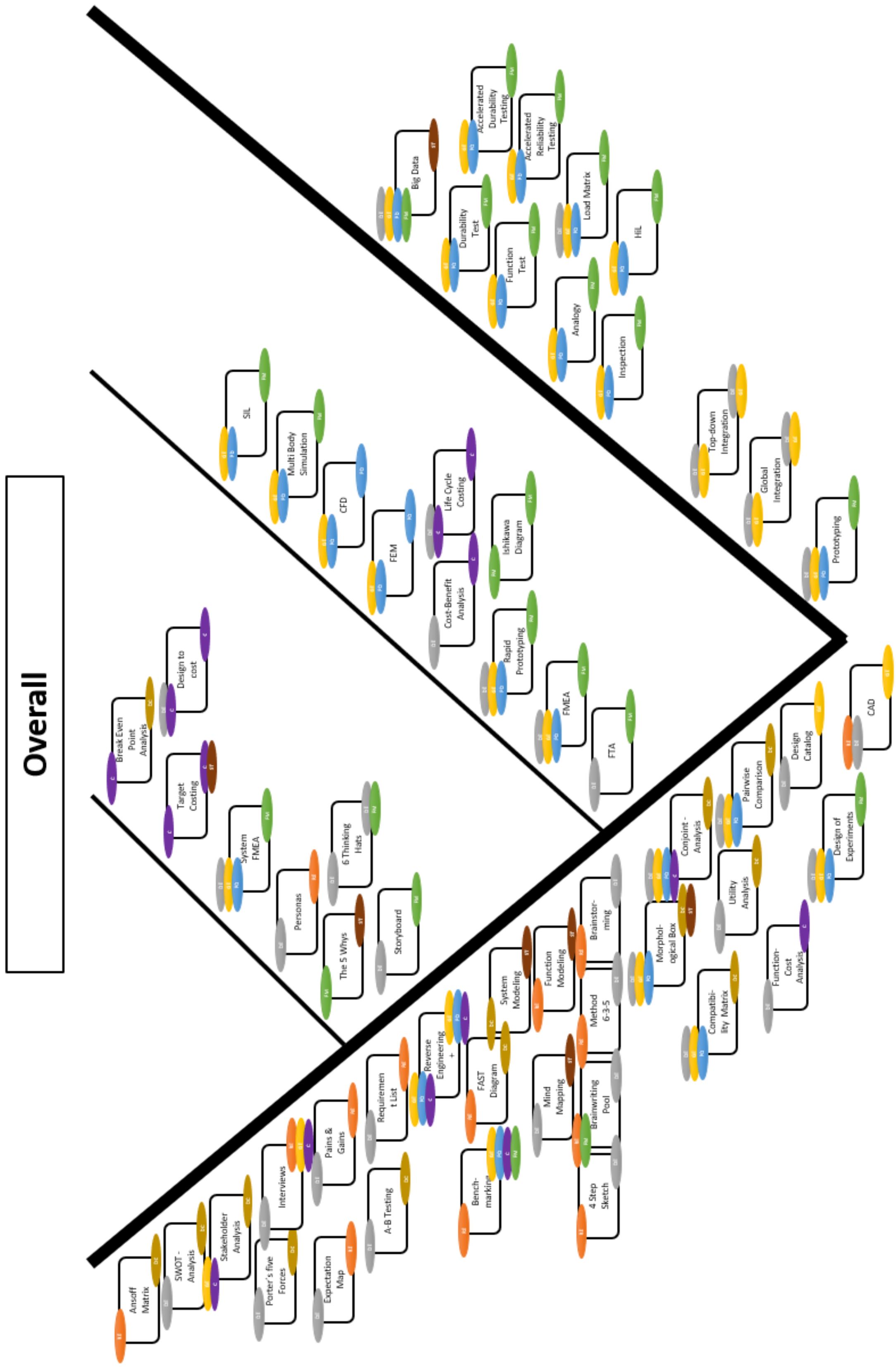
Part 3

Life Cycle Costing	Referes to the total cost of ownership over the life of a product	Financial, enviromental and social costs	Total cost of ownership		4	Design team / Controlling team	4´6	Costs / Description	Costs	ANALYSE TARGET
Load Matrix	To set up a test schedule which tests systems, subsystems or components in the sense of their stress collective	Stress collective	Failure modes		4	Verification team	4´6	Description / Geometry / Physical Data	Failure Modes	SECURE TARGET
Method 6-3-5	Generate a variety of ideas	Specification sheet, idea, problem, need	wide variety of ideas		1	Design team	6	Requirements	Description	SEARCH FOR POSSIBLE SOLUTIONS
Mind Mapping	A graphical way to represent ideas and concepts	Idea to develop	visual representation idea and related subtopics		1	Design team	2´5	Desire/ Description	Structure	SEARCH FOR POSSIBLE SOLUTIONS
Morphological Box	Simplify a problem by segmentation into its individual parts	Specification sheet Concept design problem	Design solution		1	Design team	4´6	Geometry / Phys. Data / Description	Structure	STRUCTURE TARGET
Multi Body Simulation	A numerical simulation in which multibody systems are composed to conduct motion analysis	CAD files, motion sequence	Failure modes		5	Design team	3´5	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Pains & Gains	Align product spezification with customer needs	Intention to serve a specific market segment	alignment of product spezifications with customer needs		2	Design team	4´6	Desire	Requirements	PLAN TARGET
Pairwise comparison	To compare different factors systematically	Criteria, Functions, Operations	Priorization of criteria		2	Design team	4´6	Geometry / Phys. Data / Description	Decision	DECISION SUPPORT
Personas	Provide a range of different perspectives on a product or service	Concept	Better understanding of the customer		2	Design team	5´7	Description	Requirements	PLAN TARGET
Porter´s Five Forces	Analyze the competition of a business	Research of the market	Attractiveness of an industry		2	Mix	5´7	Description	Decision	ANALYSE TARGET
Prototyping	learn through making and quickly get feedback	Concept	Strenghts and weaknesses of your design		4	Design team	4´6	Geometry / Phys. Data / Description	Failure Modes	SECURE TARGET
Rapid Prototyping	Used to quickly fabricate a scale model of a physical part or assembly using three- dimensional CAD data	CAD data	Hardware, Failure modes		4	Design team	3´5	Geometry / Phys. Data / Description	Failure Modes	SECURE TARGET
Requirement List	helps to identify and quantify needed requirements	Wishlist	structured, documented requirements		1	Design team	4´6	Desire	Requirements	PLAN TARGET
Reverse Engineering +	Deconstruction of a man-made object to reveal the designs, architecture or to extract knowledge from the object and use the information for new products	Product from the market	Specific information about existing products		4	Mix	4´7	Geometry / Phys. Data / Costs	Geometry / Phys. Data / Costs	ANALYSE TARGET
Software in the Loop	Test the software of a complex real-time embedded system	Software of the product and simulator	Verification of the software		4	Design team	4´6	Geometry / Phys. Data	Failure Modes	SECURE TARGET
Stakeholder Analysis	Assessment of a system and potential changes to it as they relate to relevant interested parties	Information about the product and potential changes, stakeholder	Stakeholders´ interests		2	Design team	4´6	Description / Geometry / Costs	Decision	PLAN TARGET
Storyboard	A quick low resolution prototype to visualize your concept	Idea	quality/problems of the idea		1	Design team	3´5	Description	Failure Modes	SECURE TARGET

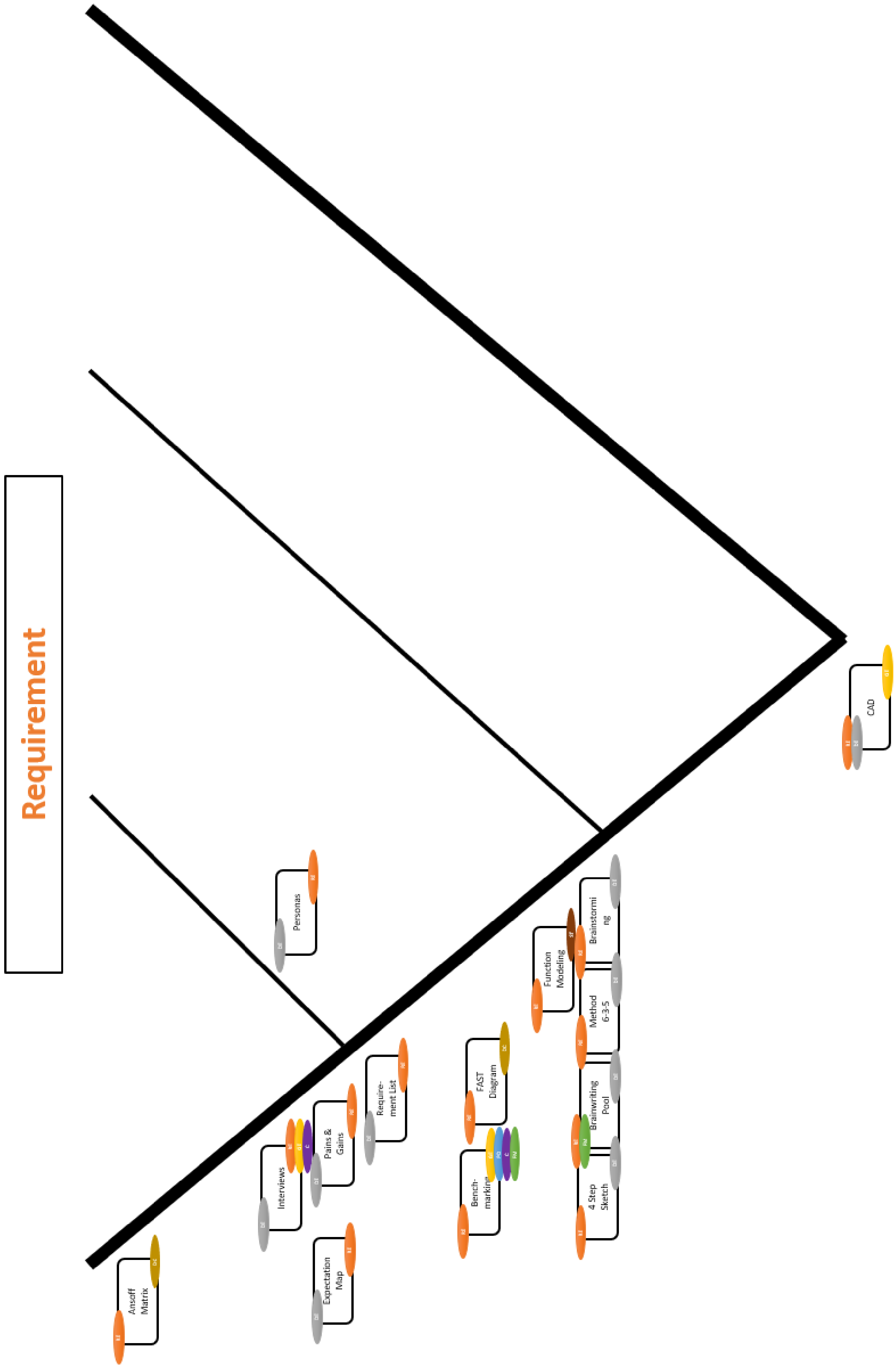
SWOT Analysis	Understand your strenghts and weaknesses, discover new opportunities and manage / eliminate threats	internal and external research / benchmark	Functions, parts, areas to focus on		2	Design team	4'6	Necessity	Desire / Decision	PLAN TARGET
System FMEA	To identify weak points in the system design early on	System description and definition of functions	Possible failures of the system design		3	Design team	4'6	Description / Geometry / Physical Data	Failure Modes	ANALYSE TARGET
System Modeling	Develop an abstract model of a system and ist environment to better understand the interactions	System description	Structure of the system and interactions		3	Design team	2'4	Description	Structure	STRUCTURE TARGET
Target Costing	To determine a product´s life-cycle cost	Target costs	Decomposition of costs from product to component level		3	Mix	4'6	Costs	Costs / Structure	DETERMINE PROPERTIES
The 5 Whys	To explore a specific problem in greater depth	Problem	Root causes		1	Design team	2'4	Failure Mode	Structure	SECURE TARGET
Top-Down Integration	System elements and modules are integrated in order of their comissioning	System elements	Combination of for the product relevant parts (prototype)		4	Mechanical development	4'6	Description / Geometry	Description / Geometry	DETERMINE PROPERTIES
Utility Analysis	used to make complex decisions. The total problem which is to be decided on, is broken down into sub-	Different Consept Designs	Priotization of a concept		2	Design team	4'6	Description	Decision	DECISION SUPPORT

B In-Output Viewer

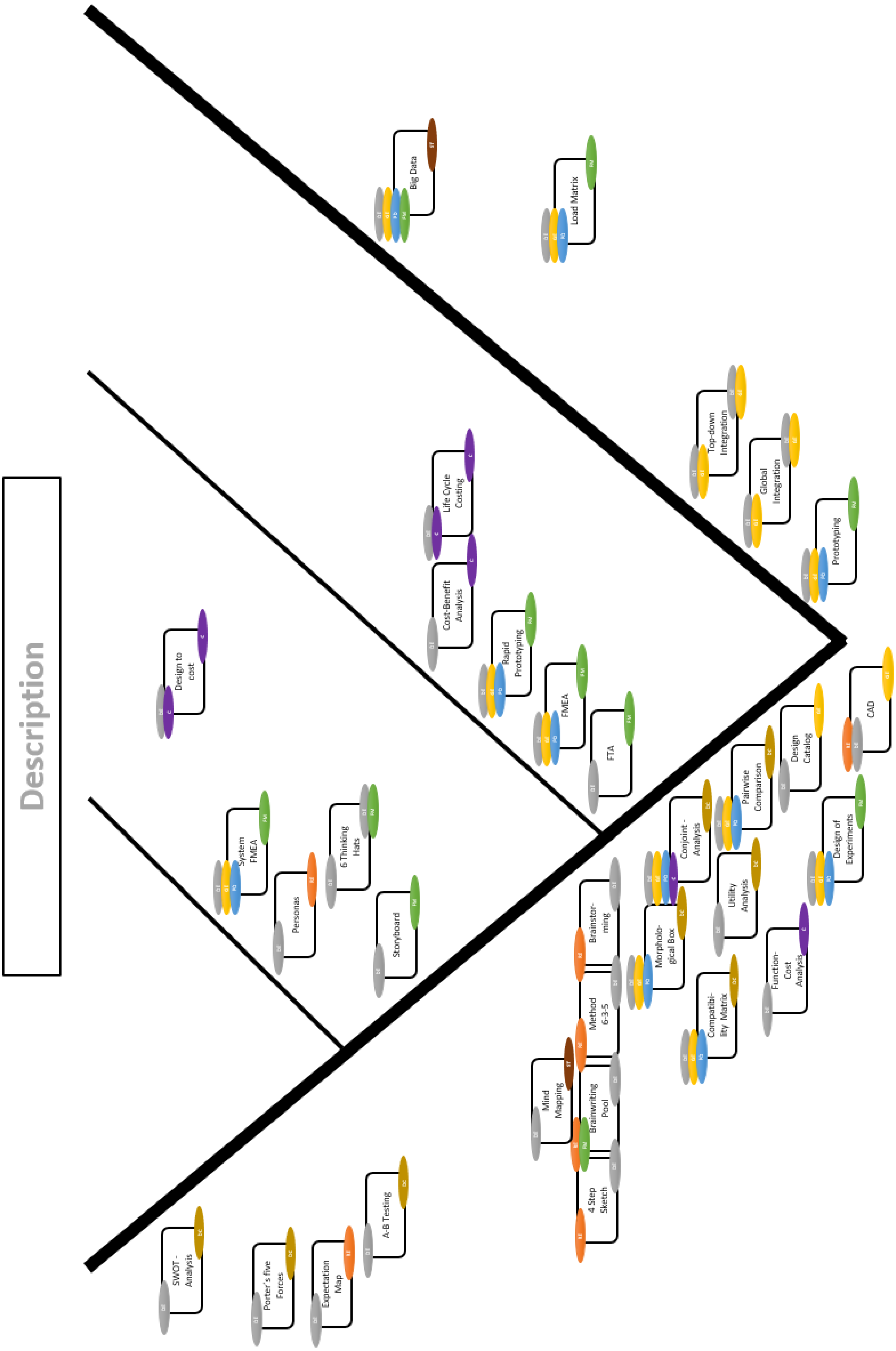
Overall (starting) view



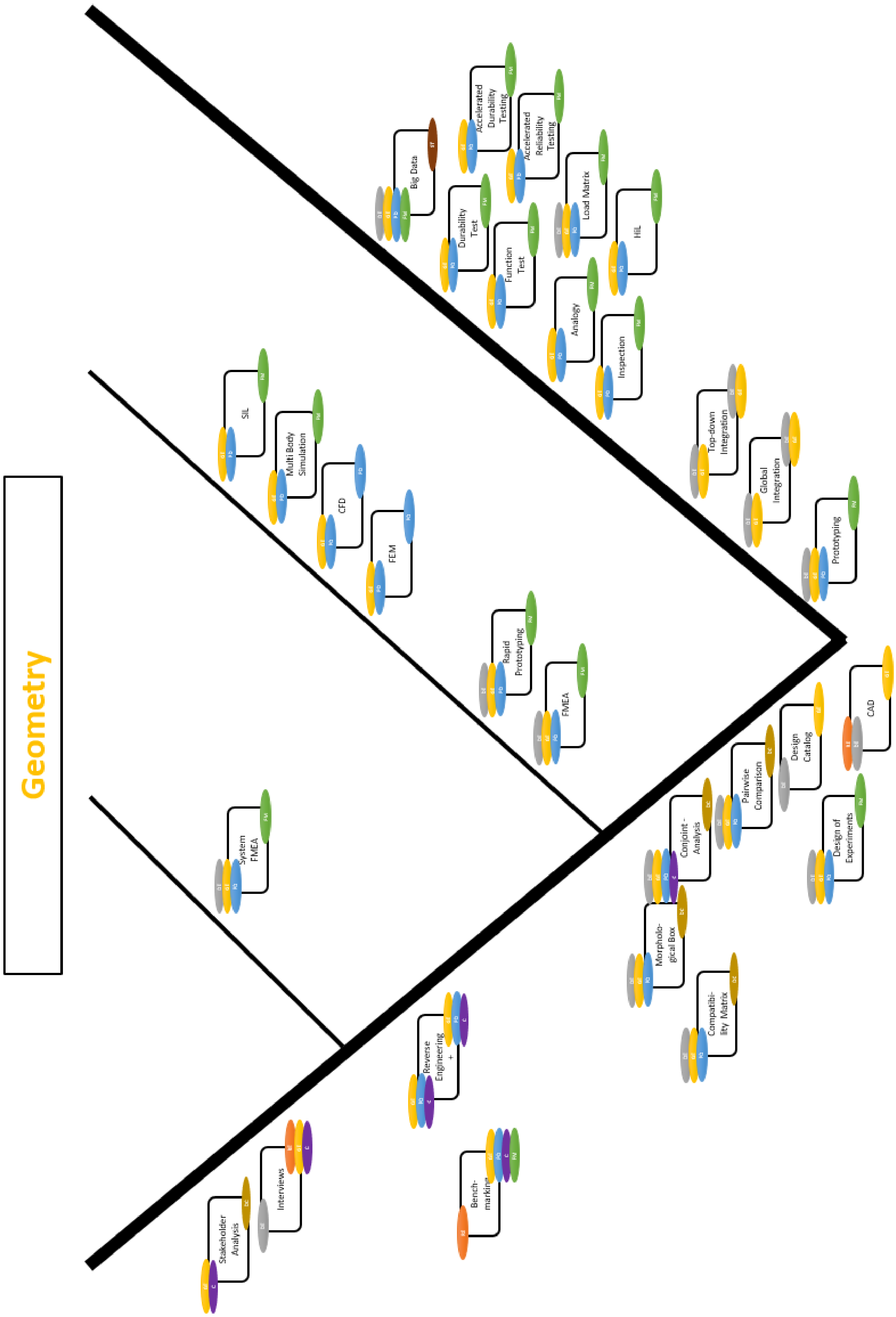
Requirement view



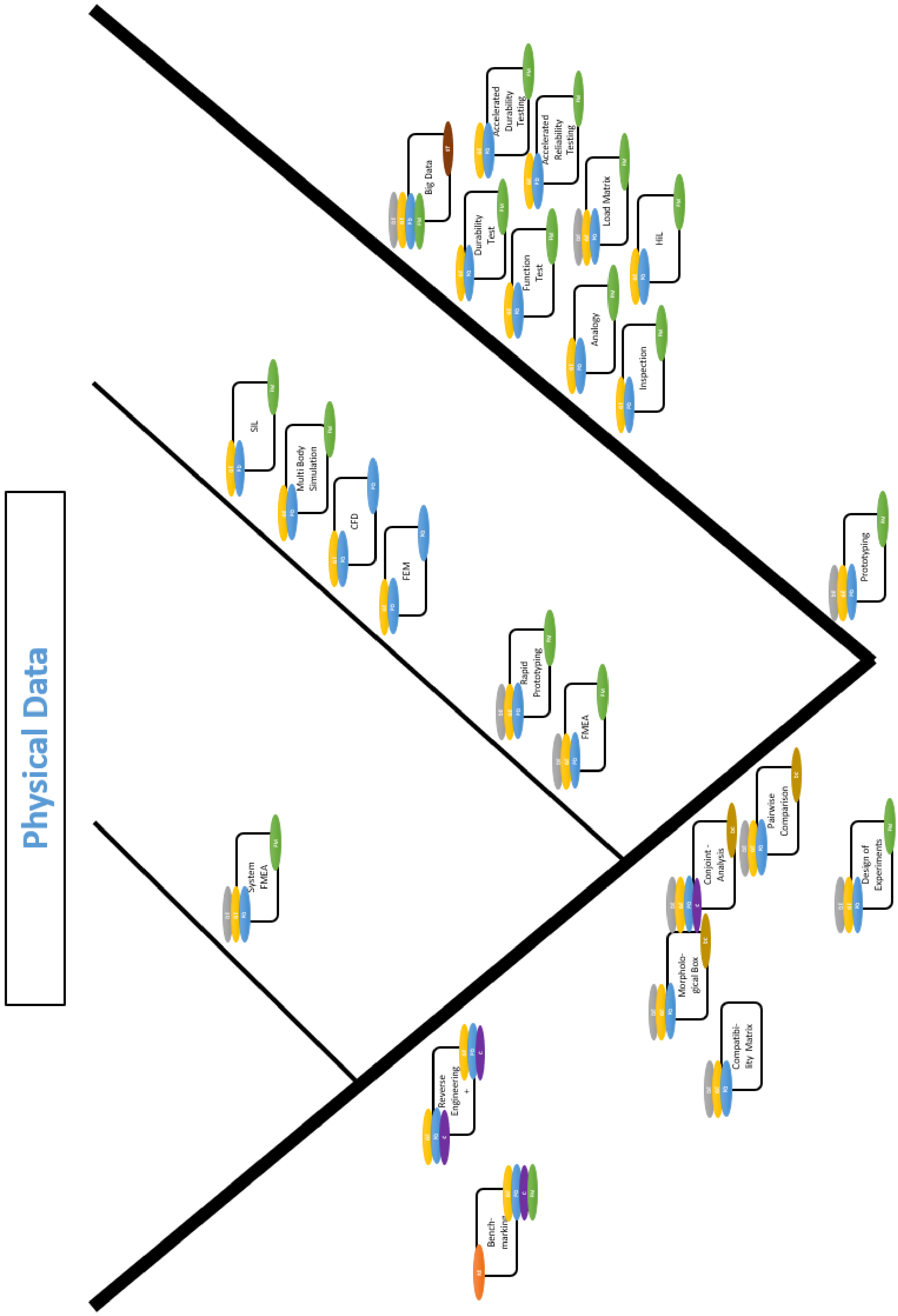
Description view



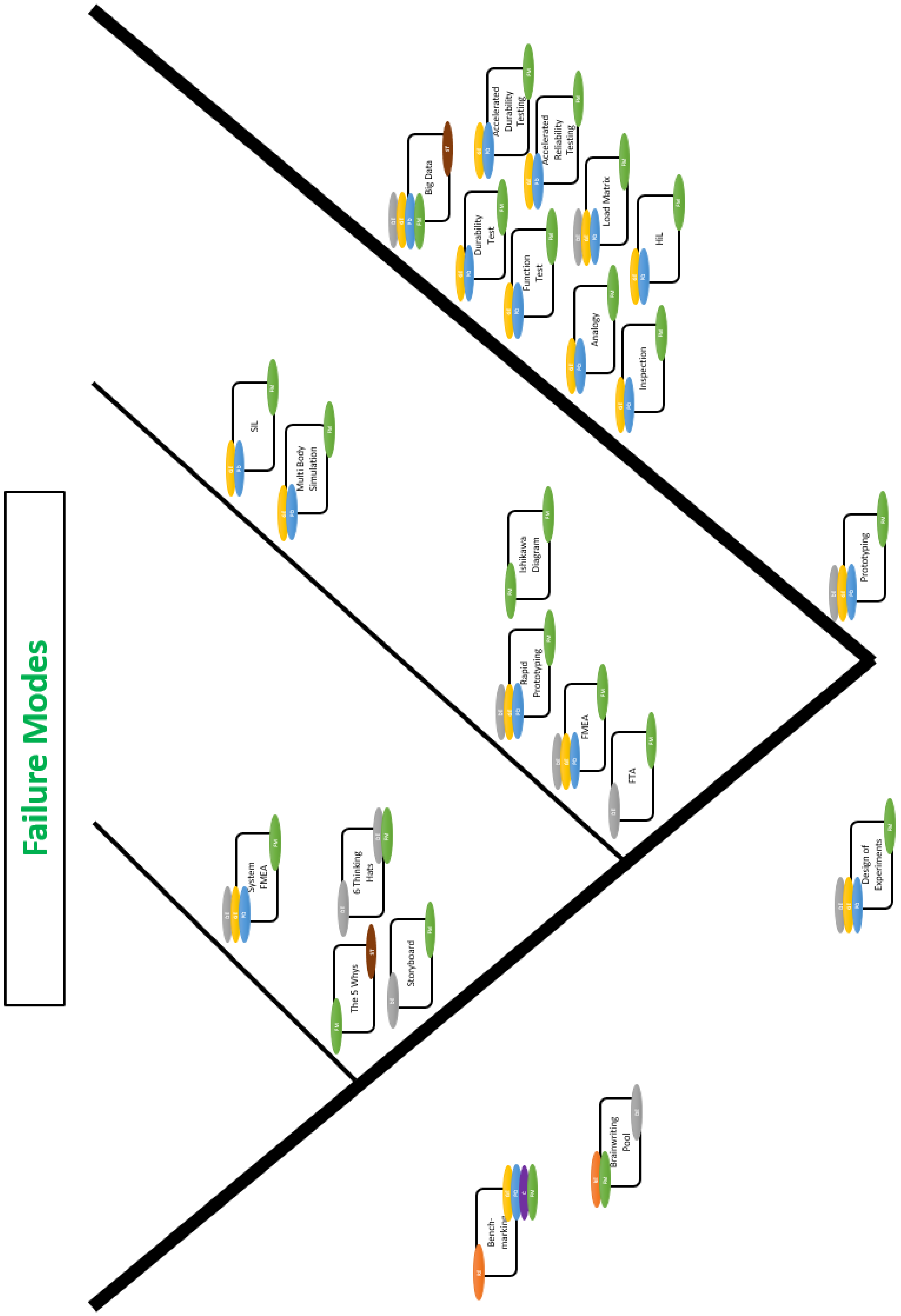
Geometry view



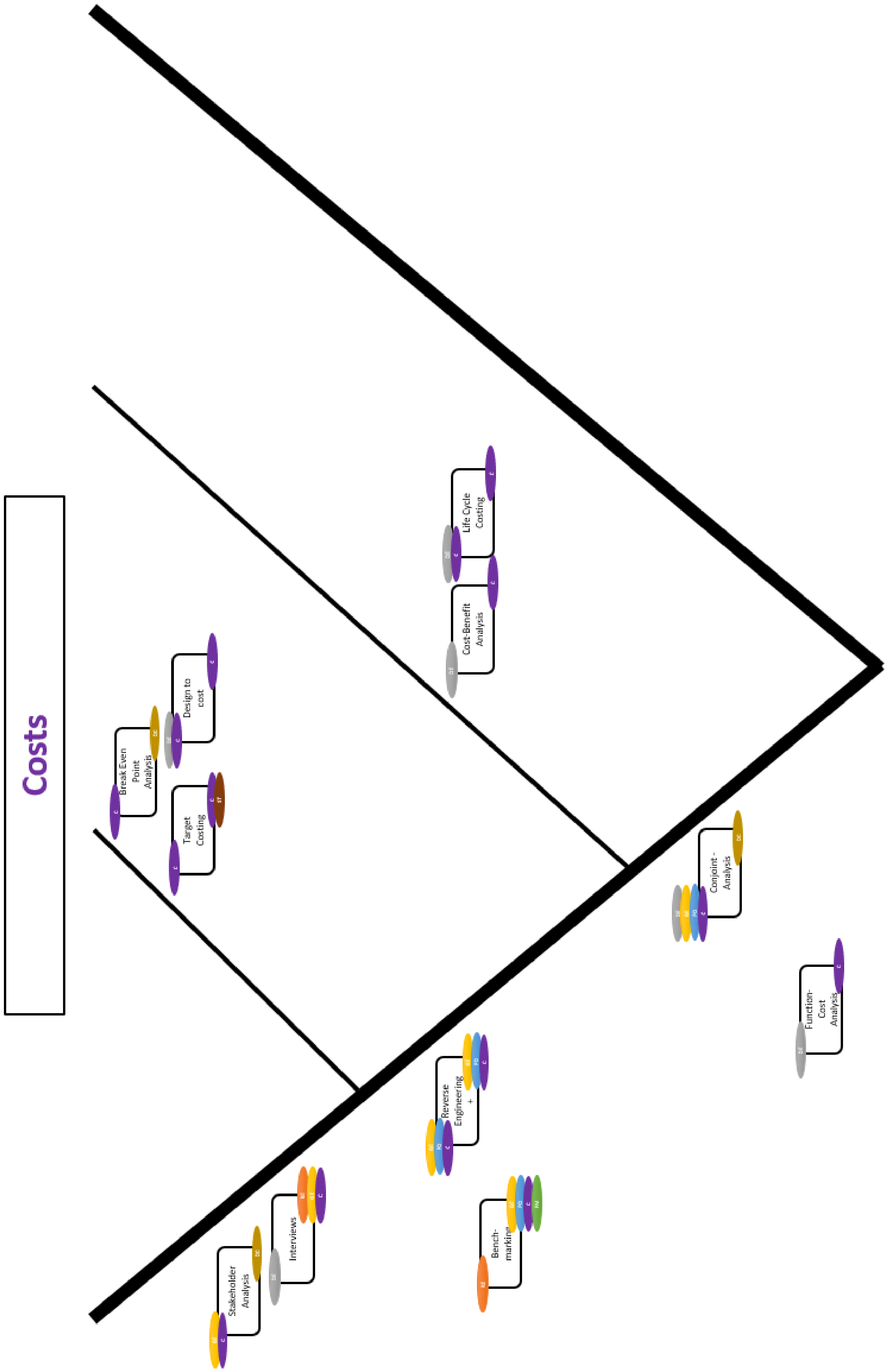
Physical Data view



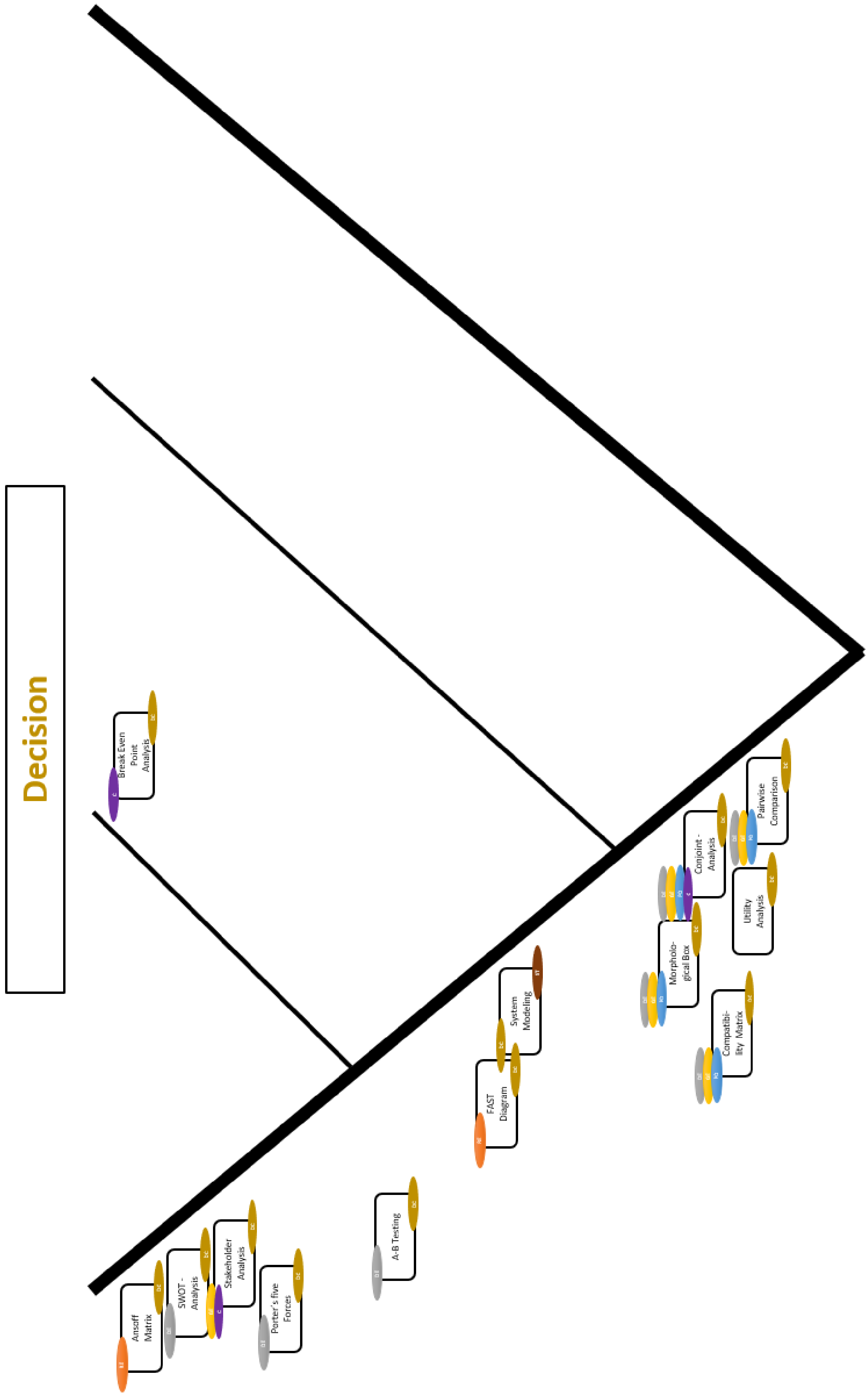
Failure Mode view



Costs view


Costs
Life Cycle Costing
Cost-Benefit Analysis
Design to cost
Break Even Point Analysis
Target Costing
Conjoint - Analysis
Function-Cost Analysis
Reverse Engineering +
Bench-marking
Interviews
Stakeholder Analysis

Decision view



Structure view