Clemens Faustmann, BSc

Analysis of development methods for tribological systems supported by specified system models

MASTER'S THESIS

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

submitted to

Graz University of Technology

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

Graz, June 2019

Everything should be made as simple as possible, but not simpler.
Albert Einstein (1879-1955)

Danksagung

An dieser Stelle möchte ich all jenen danken, die durch ihre fachliche und persönliche Unterstützung zum Gelingen dieser Arbeit beigetragen haben. Herrn Univ.-Prof. Dipl.-Ing. Dr.techn. Hannes Hick danke ich für die Möglichkeit diese spannende Themenstellung am Institut für Maschinenelemente und Entwicklungsmethodik bearbeiten zu dürfen. Für die ausgezeichnete Betreuung danke ich Herrn Dipl.-Ing. BSc Matthias Bajzek, der mir immer mit fachlicher Expertise zur Seite stand.

Mein Dank gilt aber vor allem meiner Familie, insbesondere meinen Eltern, die mir mein Studium ermöglicht und mich in all meinen Entscheidungen unterstützt haben. Meiner Lebensgefährtin Andrea, die mir den nötigen Rückhalt in allen Lebenslagen gibt, danke ich ganz besonders.
Widmen möchte ich diese Arbeit meinem Vater, der den Abschluss dieser leider nicht mehr miterleben konnte. Ohne deine Unterstützung, dein Vertrauen und deine Weitsicht wäre all dies nicht möglich gewesen. Ich danke dir von ganzem Herzen dafür!

Eidesstattliche Erklärung
Affidavit

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

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

Ort, Datum

Unterschrift

Abstract

Increased system requirements regarding efficiency and ecological impact lead to new technical solutions. Especially in the automotive industry, new technologies emerge. This all leads to increasing system complexity of socio-technical systems. The number of system elements expands while interrelations between them become even more dynamic and non-linear. Development methodologies represent a set of processes, methods, models and tools, and support engineers to overcome these technical and organizational challenges.

The objectives of this thesis are to investigate the interactions of models, especially system models, with methods and processes and to clarify the term *system model*. Principles from systems engineering support the transition of globally defined concepts to its application for an use case.

Model-based approaches like *model-based systems engineering (MBSE)* cope with complexity by relying on principles from systems thinking and extensive use of models to describe a system. In general, models are created to enable better system understanding and to provide a base for communication among stakeholders. *System models* are a key concept of model-based systems engineering. Therefore, established understandings of the term were investigated and requirements for a new definition were derived. Based on that, a new definition of system models and a comparison to other development models were described. A classification scheme for models was developed and visualized with a three-dimensional cube. Established views on system models and approaches in model-based systems engineering were incorporated within the developed concepts. To provide an applicable methodology, the interactions between models and methods were deeply investigated. Furthermore, the application of tailored concepts to an use case was described. The development of a tribological system was chosen as example, and a specific development task for friction reduction in an internal combustion engine was investigated.

As a result, system models are incorporated within a procedure to select methods and models for development. An adaptable methodology was developed and the tailoring of it to a specific use case is exemplary shown for tribology.

Kurzfassung

Die Anforderungen an Systeme hinsichtlich Effizienz und ökologischer Aspekte führen zu neuen technischen Lösungen. Insbesondere neue Mobilitätslösungen und Antriebskonzepte stellen hohe Herausforderungen an die Entwicklung dieser immer komplexer werdenden Systeme. Die Komplexität von sozio-technischen Systemen wird weiter erhöht durch die steigende Anzahl an Systembestandteilen, größere Produktvielfalt und nichtlinieare, dynamische Wechselwirkungen zwischen Systemelementen. Entwicklungsmethodiken unterstützen das Engineering durch Prozesse, Methoden, Modelle und Tools um diese steigende Komplexität beherrschen zu können.

Die Konzepte, die in dieser Arbeit vorgestellt werden, basieren auf der Analyse von Wechselwirkungen zwischen Modellen, insbesondere Systemmodellen, und Methoden. Grundprinzipien von Systems Engineering wurden angewendet um diese definierten Konzepte auf ein Anwendungsbeispiel zu übertragen.

Modellbasierte Entwicklungsansätze, wie auch *Model-based Systems Engineering*, nutzen modellhafte Beschreibungen von Systemaspekten um komplexe Zusammenhänge zu beschreiben. Insbesondere *Systemmodellen* kommt eine herausragende Bedeutung zu. Aus diesem Grund wurden publizierte Konzepte zu Systemmodellen analysiert und auf Basis der identifizierten Anforderungen eine adaptierte Definition abgeleitet. Ein Ordnungsschema für Modelle in der Entwicklung technischer Systeme wurde in Form einer multidimensionalen Klassifizierung ausgearbeitet und mithilfe eines *Würfels* visualisiert. Insbesondere die Wechselwirkungen und Zusammenhänge von Modellen und Methoden wurden eingehend untersucht, um die *Befüllung* eines Systemsmodells mit Information nachvollziehen zu können.

Das Konzept eines Systemmodells wurde in eine Methodik integriert, mit welcher eine gezielte Methodenauswahl auf Basis von Kundenanforderungen erfolgt. Die ausgearbeiteten Konzepte wurden in weiterer Folge für eine tribologische Aufgabenstellung beispielhaft angewendet.

List of Abbreviations

CAD	Computer Aided Design
E/E	Electrics/Electronics
EHD	Elastohydrodynamics
FE	Finite Elements
FMI	Functional Mockup Interface
FMU	Functional Mockup Unit
HD	Hydrodynamics
INCOSE	International Council on Systems Engineering
ISO	International Organization for Standardization
IT	Information technologies
IODP	Integrated open development platform
MBD	Multi-body dynamics
MBE	Model-based engineering
MBSE	Model-based systems engineering
NASA	National Aeronautics and Space Administration
NVH	Noise Vibration Harshness
PLM	Product lifecycle management
QFD	Quality function deployment
SE	Systems engineering
SoS	Systems-of-systems
SysML	System modeling language
UML	Unified modeling language
V&V	Verification & Validation
VDI	Verein Deutscher Ingenieure

Contents

1 Introduction

The complexity of products is increasing, which brings huge challenges for product development with it. Most systems in mechanical engineering are socio-technical systems, which are hybrid combinations of hardware and software interrelated with people. In this kind of systems, principal actions are performed by hardware, but software and the *human factor* play essential roles as well (e.g., vehicles, computer-controller manufacturing machinery, etc.).[1] These systems do not only consist of many components, its interactions are also non-linear, which leads to dynamic system behavior. The high number of dynamically interrelated components make it a complex system. Therefore, the development of socio-technical systems proves to be especially challenging and demands sophisticated engineering approaches, which themselves require tailored organizations and expert networks.

These developments also lead to a change of organizational matters, development approaches, increase the number of involved stakeholders and much more. Especially in the automotive industry, additional triggers for increasing complexity of products are ever restricting legislative regulations regarding emission reduction. This empowers so-called *green technology*, where alternative technologies (battery electric vehicles, hydrogen fuel cell, etc.) aim to substitute conventional powertrains, powered by fossil fuels. This also covers all kinds of electrification of the powertrain in order to reduce emissions and increase overall efficiency, which leads to new technical solutions and therefore to greater product variety. Furthermore, legislative regulations set requirements regarding NVH (noise vibration and harshness), safety and more. Following the vision of autonomous driving and the demand for more embedded infotainment and entertainment systems in passenger cars, the system under development includes more and more aspects, while also time to market (including development and production time) needs to be shortened.[2]

In order to cope with these developments, systems thinking is considered as part of systems engineering principles. They become essential when considering a system not just as bunch of components, but as a whole consisting of subsystems and components which are strongly interrelated to operate together for a common purpose.[3] Systems thinking enables to look at a system as a whole to understand its interrelations with its environment as well as the structure within the system and its behavior.[4]

---

[1] Kossiakoff et al. 2011, p.361.

[2] Grebe and Fischer 2018, p.87 f.

[3] Forrester 1968, p.1.

[4] Senge 1990, p.13 ff.

A system, which can be a virtual or hardware product, has to be considered over its whole lifecycle where associated engineering tasks are supported by methods, models and (mostly IT-based) tools. Challenges in interdisciplinary development projects are different for each project and therefore it is hardly possible to provide an approach, which can fulfill all the different project specific requirements.[5] Therefore, connected organizations and processes are required as well as communication overarching not only departments but also cultures and time zones.
Major objectives of development approaches have always been to consider the whole lifecycle of a system and to implement enhanced customer involvement in early phases.[6] Tailored methodologies are required to deal with complexity and to enable lifecycle considerations of systems. Methodologies represent collections of methods, processes and tools[7],[8] to support product development. Many of them take up perspectives and concepts from systems thinking.[9]

Many approaches try to cope with increasing system complexity. Especially *systems engineering* proved to be beneficial when used to solve complex problems. Starting with an identification and analysis of customer needs, required system functions are derived. Based on that, a system concept is defined with ongoing planning activities regarding verification and validation, which are already considered in early project phases.[10] Systems engineering (SE) is a way of thinking, which is not only applicable for engineering of complex systems, but which is also frequently used to overcome challenges related to complexity. According to *Haberfellner et al.*, systems engineering is based on a handful of principles: Top-down approach, thinking in variants, structuring the procedure into phases and problem solving cycle.[11] These principles are further described in the course of this thesis. Product development often builds on model-based approaches, which have the objective to describe a system under consideration through models rather than with documents. Model-based development approaches (also called *virtual product development*) include IT-supported and model-based processes for the development of a system with the objective to save time and costs as well as to improve the quality of a product.[12]

In model-based development of complex systems, engineers have access to an immense range of available development methods and tools. In today's development projects, the task to decide which method should be used is often based on experience and best practices. Furthermore, the importance of effective knowledge management increases, because of the ever shortening time period it takes until technical knowl-

---

[5] Eigner 2014, p.2 f.

[6] Lindemann 2016, p.3.

[7] Dvorak 2017, p.72.

[8] Estefan 2008, p.1.

[9] Lindemann 2016, p.6.

[10] Long and Scott 2011, p.11.

[11] Haberfellner et al. 2015, p.55.

[12] Hirz et al. 2013, p.29.

edge becomes obsolete and changing possibilities and forms of data management.[13] It is therefore important to investigate methodologies, which support decision making with consideration of this changes in product development. Therefore, methodologies provide a framework for model-based product development independent of the application. Expert knowledge should not be replaced but should be implemented carefully and it should be possible for other engineers within the organization to use it for their own tasks. In the context of model-based development, this thesis has the objective to describe how models and methods are connected via system information. Based on that, an approach for model-based selection of development methods is outlined.
Figure 1.1 illustrates the observable trend of increasing system complexity for automotive powertrains, by exemplary showing the expanding diversity of subsystems and functionalities within a powertrain system. In the past, powertrain systems of passenger vehicles consisted mostly of few essential mechanical parts (engine, transmission, etc.), while today's systems (for this example the structure of a hybrid powertrain is shown in figure 1.1) are more complex. Not only does the system consist of two main subsystem (conventional powertrain and electrification system), also the control mechanisms in form of software and embedded systems increase system complexity by enhancing variability of interactions. This points to even higher product diversity and complexity for future systems where also new technical solutions may establish as new system types or as addition to conventional ones. Furthermore, *digitalization* has huge impact on technical systems like passenger cars. Not only does it change the product by adding new possibilities for entertainment, infotainment and communication, it also changes production, summarized by the catchphrase *industry 4.0*.[14] As conclusion, approaches are required which are neutral regarding the technical solution and which enable management of complexity.

Product development includes various aspects and system considerations can be done in different phases of the system's lifecycle. Therefore, it is important to clearly define the context of this thesis. To establish common understanding of frequently used terms, relevant fundamentals and its theoretical background are investigated and the most important ones are summarized in chapter 2 of this thesis. Fundamentals regarding product development are described as well as clarification of terminology, which is frequently used and which is important to comprehend the developed concepts. Established approaches to develop complex systems, like systems engineering, are investigated to understand the basic principles of them. Furthermore, the role of models in product development is discussed and advantages of model-based approaches are investigated. In chapter 3, a methodology is described which has the objective to select development methods supported by models. Therefore, a classification scheme for models is developed as well as extensive investigations of *system models* and an

---

[13] North and Kumta 2018, p.VIII ff.

[14] Lindemann 2016, p.5.

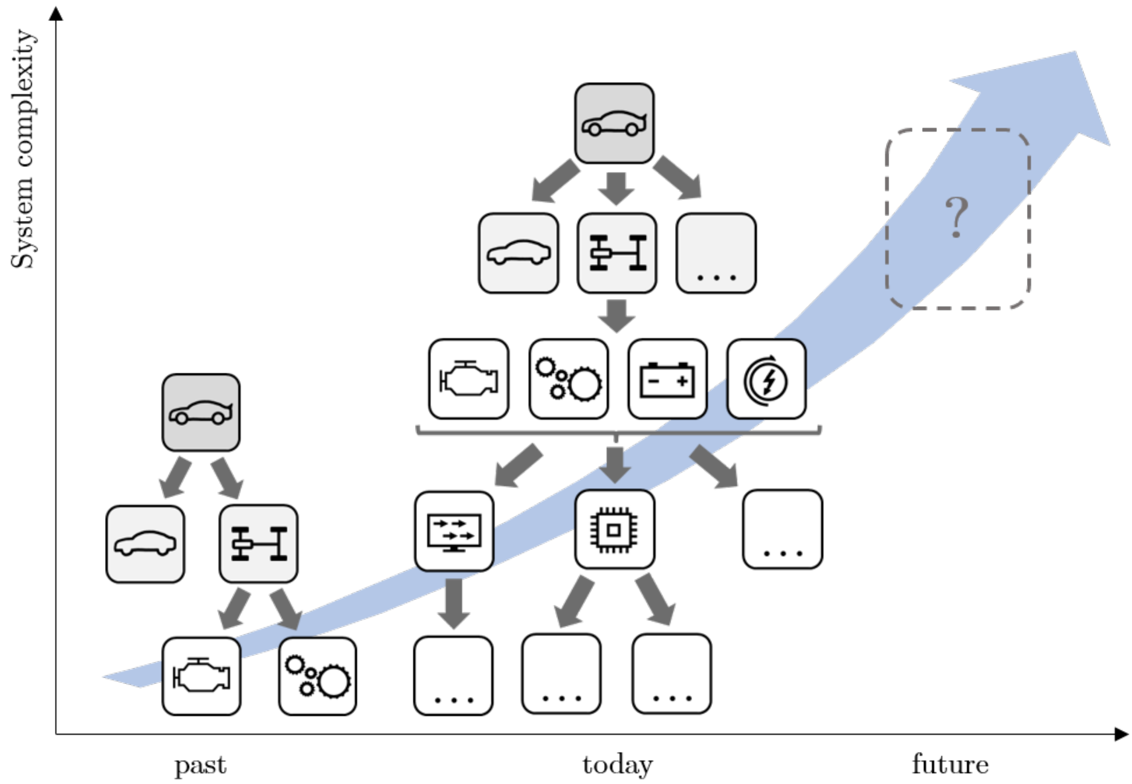


Figure 1.1: Increasing complexity of passenger vehicle powertrains[17]

adapted definition of it. The developed concepts and approaches are described in detail in chapter 3, where the emphasize is laid on interactions between models and methods to enable a more efficient use of them in development projects. To examine if the developed concepts are applicable in real development projects, they are applied to a use case. The mentioned evolution of powertrain systems in the automotive industry requires highly sophisticated approaches with emphasize on system considerations. Therefore, engineering approaches in that sector represent an interesting context for the developed concepts of this thesis to be analyzed and verified by applying them to an use case. To narrow the focus on a certain scientific discipline, involved in the development of powertrains for passenger cars, tribology[15] was chosen as an use case. The investigation of tribological systems is challenging because of the high number of involved scientific disciplines, different scales, on which tribological effects occur, and interdependencies of functionalities.[16]

---

[15] Jost 1966.

[16] Czichos and Habig 2010, p.12.

[17] Hick, Bajzek, et al. 2019.

These challenges make it an appropriate use case to demonstrate the benefits of the concepts presented in this thesis, which should support handling complexity in demanding development processes. The purpose of the developed concepts are demonstrated by applying them to a tribological development of an automotive powertrain subsystem. This is covered by chapter 4. Finally, the concepts are discussed and evaluated in chapter 5 and possible areas of its application together with required further investigations are outlined in chapter 6.

2 Fundamentals

To establish common understanding, terms and approaches, which are frequently used, are defined. Furthermore, it is necessary to describe the context for the application of the developed concepts, as many terms are differently used depending on the field of science.

2.1 Systems, Models and Methods

The complexity of state-of-the-art products is increasing as described in section 1. This leads to a need for approaches to consider systems not just as an assembly of parts, but as complex structures consisting of parts and subsystems, properties and interdependencies. In mechanical engineering, a variety of development methodologies are used for that reason. Many of them build on conceptual terms like system, method, model and tools. Therefore, these terms are defined in the following.

2.1.1 Systems thinking

As the term *system* is used in different contexts with varying scopes, this term has to be defined for the application of this thesis. In general understanding, a system consists of several components, which interact to fulfill the purpose of the whole system.[1] The *International Organization for Standardization (ISO)* defines the system in a technical context as follows:

> *“[...] a system is sometimes considered as a product or as the services it provides. [...] in practice, the interpretation of its meaning is frequently clarified by the use of an associative noun, e.g., aircraft system. Alternatively, the word “system” is substituted simply by a context-dependent synonym, e.g., aircraft, though this potentially obscures a system principles perspective. [...] a complete system includes all of the associated equipment, facilities, material, computer programs, firmware, technical documentation, services and personnel required for operations and support to the degree necessary for self-sufficient use in its intended environment.”*[2]

---

[1] Forrester 1971, p.13 ff.

[2] ISO15288 2018.

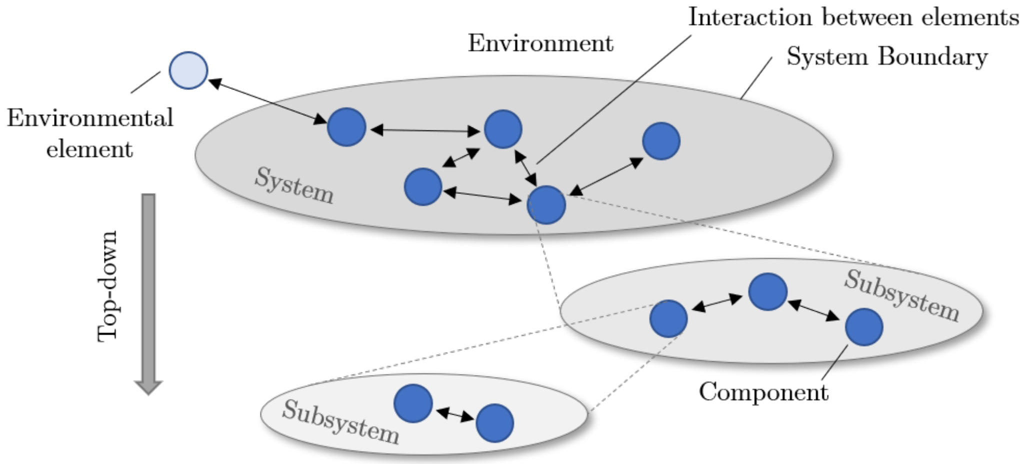


Figure 2.1: General illustration of a top-down approach to analyze systems, its subsystems and components *inspired by Dillerup and Stoi*[3]

Figure 2.1 illustrates that systems consist of several elements separated from the environment with a system boundary. A system consists of components and subsystems, which can be further decomposed.

In order to understand a system as a whole, interactions between system elements are important. The field of *systems thinking* was established by *Peter M.Senge* to consider systems as a whole and to enhance the understanding of a connected world. He recognized that this way of thinking is essential to understand an entity he called a *learning organization*, which represents a continuously evolving system.[4] Systems theory is an approach to focus not only on its components and subsystems, but to also lay emphasize on the interdependencies between them. *Aristotle* stated, that *the whole is greater than just the sum of its parts.* Especially with system complexity increasing more and more, the importance of system considerations rises. This approach is also called *holistic view* in contrast to an *atomic view* on a system.[5] Top-down approaches adopt this principles by firstly concentrating on the whole system, its system boundary and interaction with its environment, followed by considerations, where the focus is narrowed step by step (e.g., system - subsystem - components). Furthermore, systems science has the purpose to classify systems hierarchical, describe their structure, behavior, functions and relevant views on the system.[6] It depends on the perspective how a system is further described.

For instance, if the system is considered from a functional perspective, it contains

---

[3] Dillerup and Stoi 2016, p. 30.

[4] Senge 1990, p.13 ff.

[5] Dillerup and Stoi 2016, p.25.

[6] Ropohl 2009, p.75.

a set of interrelated functions, inputs, outputs and other related elements like the environment.[7]

Furthermore, so-called *system-technical methodologies* have the purpose to consider the wholeness of a system and the related technical discipline to describe the system under development not just as a structure consisting of components, but also as connected and dynamic combination of parts. It applies techniques from information technologies, biology and cybernetics (the science of control mechanisms and communication within systems[8]). Another frequently used term is *systems-of-systems (SoS)*. It is used for several systems, which are individually developed to be able to stand for their own but which provide greater benefit (e.g., value for the customer) when they are orchestrated to work together in a connected way.[9] For example, a navigation system and a passenger car can both act as single systems independently from each other, but if they are connected in a smart way, they can provide additional benefits like reduced fuel consumption through intelligent powertrain operating strategies adapted to the selected route.

2.1.2 Complexity of systems

As the *complexity* of a system is frequently mentioned to emphasize the challenges in state-of-the-art development projects, it is important to understand the meaning of the term. An established system classification regarding difficulty, rated by number of parts and system dynamics, is shown in figure 2.2. A system is defined as simple, when there are only a few elements and static interaction between those elements. In contrast, a complex system consists of numerous elements with dynamic interaction (e.g., hybrid vehicle powertrain).[10] This leads to system behavior, which is difficult to predict and which makes development especially challenging. It is widely acknowledged that increasing system complexity is one of the main reasons that make definition of system architecture and development in general challenging.[11]

---

[7] Weilkiens et al. 2016, p.91.

[8] Wiener 1948, p.32.

[9] Maier 1998, p.268 f.

[10] Ulrich and Probst 1988, p.61.

[11] Maier and Rechtin 2009, p.6.

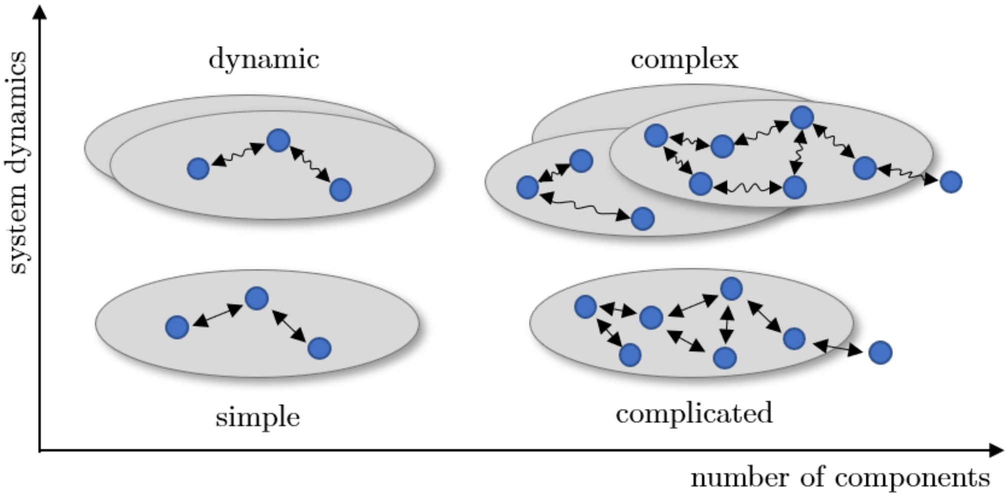


Figure 2.2: Principal classification of systems and meaning of complexity *inspired by Ulrich and Probst*[12]

2.1.3 Models

Models are used to describe systems. Based on *Stachowiak's* definitions, a model has three elementary properties:[13]

- *Mapping property*: Models always represent systems of natural or artificial origin, which can also be models itself.
- *Reduction property*: Models do not contain all attributes of its origin, they only cover those which are necessary for the creator/observer.
- *Pragmatic property*: Models have a replacement function and are not directly linked to their origin.

Based on these properties, every representation of a system can be seen as a model. To further refine the understanding of what models are, a general classification is required. Different types of models can be distinguished depending on the differentiation criteria. Some common classifications of models are incorporated in figure 2.3. Firstly, physically existing models (e.g., test specimen) and virtual models, which are processable by computers, are differentiated.[14] Model-based development approaches focus on

---

[12] Ulrich and Probst 1988, p. 61.
[13] Stachowiak 1973, p.130 ff.
[14] Law 2015, p.5.

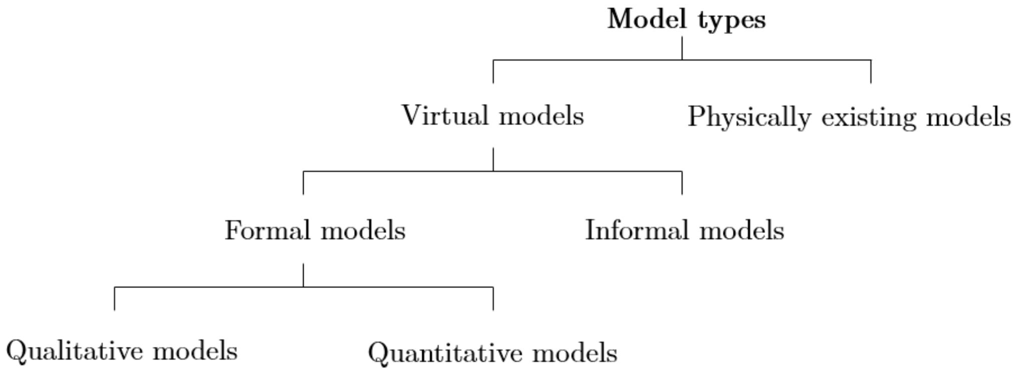


Figure 2.3: Different model types *inspired by Friedenthal et al.*[18]

formal and semi-formal models, which can be interpreted by computers and therefore provide re-usability. Informal models like documents are also often used, but do not provide distinct interpretability.[15,16] Generally, models can be quantitative and/or qualitative, depending if the information a models provides is a countable value or a qualitative conclusion.[17] Of course, other differentiation can be done, but the presented hierarchy should support the understanding, which types of models are investigated in this thesis.

A model is created by one or more views on a system, which may be combined. Views can be understood as narrow subsets of information about a system.[19] This fundamental building block of models is influenced by the scientific discipline the modeler is assigned to, the scope of the model and more. To visualize views, different notations are used, from text and tables to system modeling languages like SysML.[20]

2.1.4 Methods

The term *method* describes a rule-based and systematic task definition to achieve a certain objective. Methods are prescriptive and give proposals for a procedure. Therefore, methods in development have a strong operational character.[21] Methods are strongly linked to models and in many occasions, only an analysis of both, models and methods, make sense. To differentiate methods and models, the purpose of their use has to be investigated. With the already mentioned understanding of models, it becomes

---

[15] Madni and Sievers 2018, p.175.

[16] Koenig 2011, p.35 ff.

[17] Koenig 2011, p.31.

[18] Friedenthal and Moore 2012, p.526 ff.

[19] Weilkiens et al. 2016, p.90.

[20] Holt et al. 2016, p.12.

[21] Lindemann 2009, p.57.

clear that models represent a system and therefore contain system information within a more or less clearly defined system boundary. In summary, models *describe* systems and therefore contain system information, while methods *generate, manipulate or use* system information. For instance, if the system under consideration is a piston represented by a prototype (physically existing model), a related method is a durability test program to gain information about the reliability of the system.

In the context of product development, methods, which interact with models, are called *development methods*. Additionally, methods to support decision making are required in development projects.

2.1.5 Differentiation of processes, methods and tools

After differentiating models and methods, further terms have to be discussed. Methods define the *HOW* and are supported by instruments called *tools*. These tools are mostly IT-based. *Processes* are logical sequences of tasks which are performed to achieve a pre-defined objective. Processes define the *WHAT*, describing the required tasks in a certain time-dependent sequence and additional information about *WHEN* and *WHO* assigned to these tasks.[22,23] In most occasions, processes, methods, tools and also models are not used separately. It is therefore reasonable to analyze them in a connected way.

2.2 Product development

Lindemann describes product development as a combination of engineering, customer interaction and market situation. It considers not only the development of a product but also market mechanisms and customer demand to achieve a successful product market introduction and often relies on concepts from systems thinking.[24] Product development also integrates strategies for organizations and markets. Many of these concept like *integrated product development* provide approaches to cope with challenges in development.[25]

To transfer a problem into a solution (transformation of as-is situation into the should-be situation), development approaches and their fundamental principles have to be realized supported by a set of processes, methods, organization and tools. These so-called *four interlocking pillars* build the foundation for every system development. Successful development can be achieved by combination and coordination of processes, methods, organizations and tools.[26]

---

[22] Estefan 2008, p.2 f.

[23] Kranabitl 2019, p.27 f.

[24] Lindemann 2009, p.7 f.

[25] Ehrlenspiel and Meerkamm 2017, p.233 f.

[26] Haberfellner et al. 2015.

As other authors point out, especially in automotive industry, the design cycles of new vehicles and powertrains shorten. New approaches arise in order to fulfill development tasks in a very competitive environment within shorter time frames. Model-based approaches are therefore frequently used to support product development in many aspects.[27]

2.2.1 Model-Based Development

Model-based engineering (MBE) or model-based development, is about elevating models to a governing role in the engineering process including specification, design, integration, validation, and operation of a system.[28]

To support engineering of a system, a model-based approach focuses on the integration of models with different scope in contrast to traditional document-based approaches. It is not the purpose of models to fully replace documents, which also provide advantages in some applications, but the use of models throughout the system lifecycle proved to be beneficial. The most important reasons to use models in development are listed below:[29]

- *Characterizing an existing system*: Models are also used for knowledge capturing and documentation. Therefore, they are used to describe existing models, which may be poorly documented in an informal way.
- *Mission and system concept formulation and evaluation*: In early phases of the system life cycle, models support to synthesize and evaluate alternative missions and system concepts and also to check requirements fulfillment.
- *Data consistency*: Model-based definition of system architecture, design as well as verification and validation of related requirements ensures *data consistency* throughout a development project.

Model-based approaches support validation and verification activities in early phases, because inter alia no physical test specimen is needed to investigate system behavior. *Validation* is considered as checking if customer requirements are fulfilled - to check *if we did the right things* and if the executed activities were *effective* while it is the scope of *verification* to check if process results are in line with predefined objectives - to check *if we did the things right* and if the processes were *efficient*.[30,31]

Figure 2.4 a) schematically illustrates a traditional document-based approach, where standalone models are used in development, which are only loosely coupled and described in documents. Formal communication among different disciplines takes place

---

[27] Schlosser et al. 2007, p.25.

[28] Estefan 2008, p.10.

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

[30] Gilz 2014, p.68.

[31] Lindemann 2016, p.13.

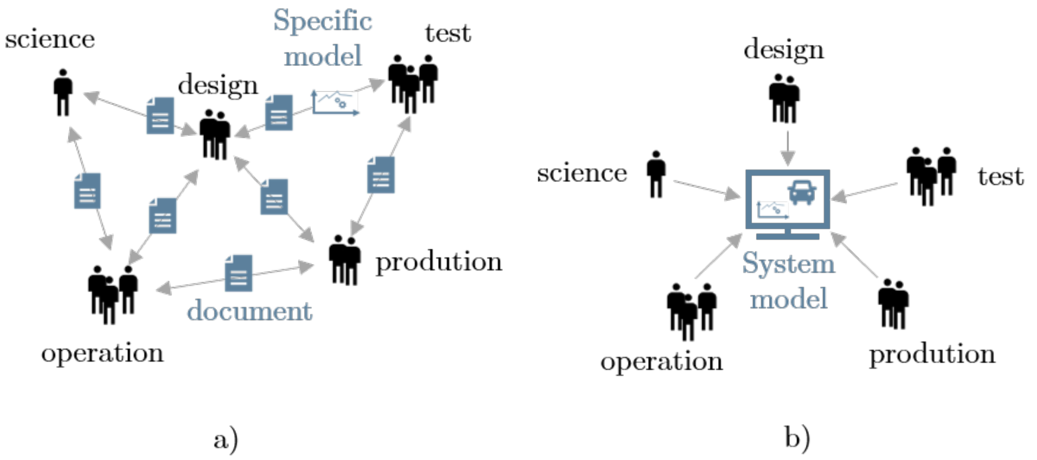


Figure 2.4: a) Standalone models related through documents (document-based) *inspired by Dvorak*
b) Multiple views on a shared system model (model-based)[33]

through a variety of documents that include human-readable text, diagrams, and spreadsheets. Figure 2.4 b) shows the principal approach to overcome the shortcomings of a traditional approach to assure consistency and completeness by using a shared system model, which is both human- and computer-readable. This provides a basis of communication about system understanding, because it integrates specific models of different disciplines.[32]

2.2.2 Systems Engineering

As the terminology hints, systems engineering is an approach established particularly in engineering of technical systems. The term *engineering* originates from the Latin term *genere* and means to generate something.[34] Over time, the term engineering evolved and nowadays it stands for a methodical approach to develop systems in a technical context. An engineer has insights in many technical disciplines from materials science over mechanical design to production science. The International Council on Systems Engineering (INCOSE) defines systems 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*

---

[32] Dvorak 2013, p.12.
[33] Dvorak 2013, p.12.
[34] Rambo and Weber 2017, p.18.

> *while considering the complete problem. […] Systems Engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation. Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs."*[35]

The *National Aeronautics and Space Administration (NASA)* sees systems engineering as a methodical and multi-disciplinary approach used in different phases of the system's lifecycle.[36]

Summarizing, systems engineering (SE) is an approach for especially challenging system developments to meet requirements within often opposed constraints. As the development of such systems is often multidisciplinary, systems engineering provides a holistic approach, which tries to balance the interests and requirements of different disciplines (e.g., different development teams and stakeholders) by avoiding a dominant perspective of a single discipline.[37] The perspective of systems engineering is based on systems thinking and the concepts of systems science. It has the objective to understand systems as dynamic entities, where interdependencies between elements define the behavior of the system. This way of thinking sharpens the awareness of wholes and how parts within a system interrelate.[38] To develop socio-technical systems it is not possible to strictly follow a predefined procedure, leading to the conclusion, that an iterative approach may be needed in some occasions. Figure 2.5 illustrates a principal approach in systems engineering. Starting with a problem or as-is situation, the problem solving process has the objective to find an appropriate solution. This process defines the tasks, whose sequence is influenced by the followed procedure model (e.g., v-model), best practices and principles of systems science. To fulfill each task, methods are needed to support of the process, which are themselves supported by tools and the organization. The procedure model affects not only the process but also the organization itself. For example, roles have to be assigned according to the subprocesses of the v-model (e.g., system designer for domain specific engineering and system integrator for coordination).

According to *Haberfellner et al.*, there are four fundamental ideas of systems engineering, which should be considered collectively rather than separately:[39]

- *Top-down approach*: To avoid conceptual issues, the approach to analyze or develop a system should start with a rough consideration followed by more and more detailed ones (e.g., from blackbox to whitebox thinking). This approach

---

[35] Walden et al. 2015, p.11.

[36] NASA 2016, p.2.

[37] NASA 2016, p.3.

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

[39] Haberfellner et al. 2015, p.55 ff.

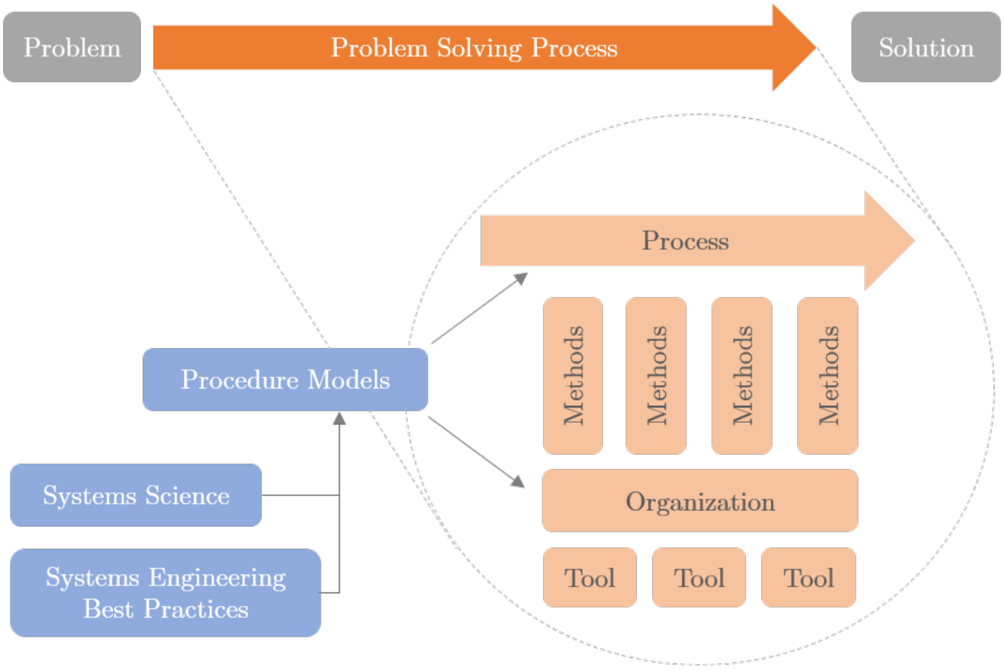


Figure 2.5: Processes, methods and tools in a systems engineering approach *inspired by Bajzek*[40]

has already been presented in the illustration of the general system structure in figure 2.1.

- *Thinking in variants*: Alternatives should be continuously developed in order to asses them in an objective way and to decide on the most promising variant.
- *Structuring the procedure in phases (macro logic)*: The process from a problem to a solution should be divided into several phases, which further support the implementation of a top-down approach.
- *Problem solving cycle (micro logic)*: To solve emerging problems in a development approach, the approach shown in 2.6 provides a structured methodology to make a profound decision.

Systems Engineering is widely used in English-speaking regions and is especially incorporated in development of military or aeronautic systems.[41] With the publication of *ISO/IEC 15288:2008 Systems and software engineering – System life cycle processes*, systems engineering got formal acknowledgment, which made it available for further applications.

---

[40] Bajzek 2018, p.61.

[41] Weilkiens 2014, p.19.

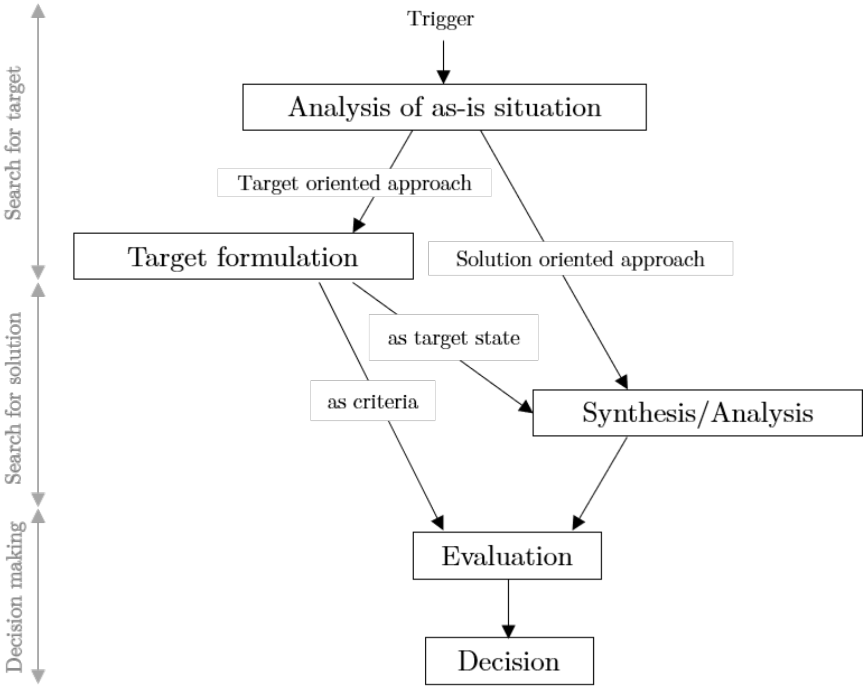


Figure 2.6: Problem solving cycle as micro logic to solve problem within the development process[42]

It seems to be a logical conclusion to build on the mentioned benefits of a model-based approach and to combine them with core principles of Systems Engineering. In conclusion, model-based systems engineering can be defined as follows, according to the *International Council on Systems Engineering (INCOSE)*:

> *"Model-based systems engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases. MBSE is part of a long-term trend toward model-centric approaches adopted by other engineering disciplines, including mechanical, electrical and software."*[43]

Weilkiens concluded in a simplified way, that model-based systems engineering is systems engineering with a *system model.*[44] There is no clear consensus among literature, what a system model actually is and how it differs from specific models. This question will be discussed extensively in the course of this thesis.

---

[42] Haberfellner et al. 2015, p.261.
[43] INCOSE 2007, p.15.
[44] Weilkiens 2014, p.22.

2.2.3 Procedure models

In order to support the product development process, researchers have conceptualized many procedure models. Most are declared as *procedural guidelines*, which can be followed iteratively as well as only once. Many established procedure models for development of mechanical or mechatronic systems have certain fundamental steps in common:[45]

- Requirements specification and planning
- Concept/draft generation
- Detailing
- Realization
- Integration
- Verification & Validation

Over time, numerous procedure models were developed depending on organizational and market specific demands.[46] From general descriptions of the product development process's phases according to *VDI2221*[47] and design guidelines according to *Pahl/Beitz*,[48] over to agile approaches like *spiral models*.[49] Of course, many combinations of different approaches and tailored models for the specific use case were developed in the past decades. These mentioned models are only an extract of many.

The v-model is a popular and frequently implemented procedure model. It is used to logically organize tasks and phases within product development. The concept of the v-model was derived from the *waterfall model*[50] whose core idea is that a phase can only be started if the information from the previous phase is available and complete.[51] *VDI* established the v-model, which is widely used in product development and which is well-known in the industry as well. Figure 2.7 shows the different procedural steps from requirements to a developed product. The v-model according to *VDI2206* is a fundamental approach to develop mechatronic systems through enhanced interdisciplinary cooperation. The v-model is well established in the industry and therefore, its procedural step are further described.

---

[45] Eigner 2014, p.15 f.
[46] Eigner 2014, p.15 ff.
[47] VDI2221 1993.
[48] Feldhusen and Karl-Heinrich Grote 2013.
[49] Boehm 1979.
[50] Royce 1970.
[51] Eigner 2014, p.4 f.

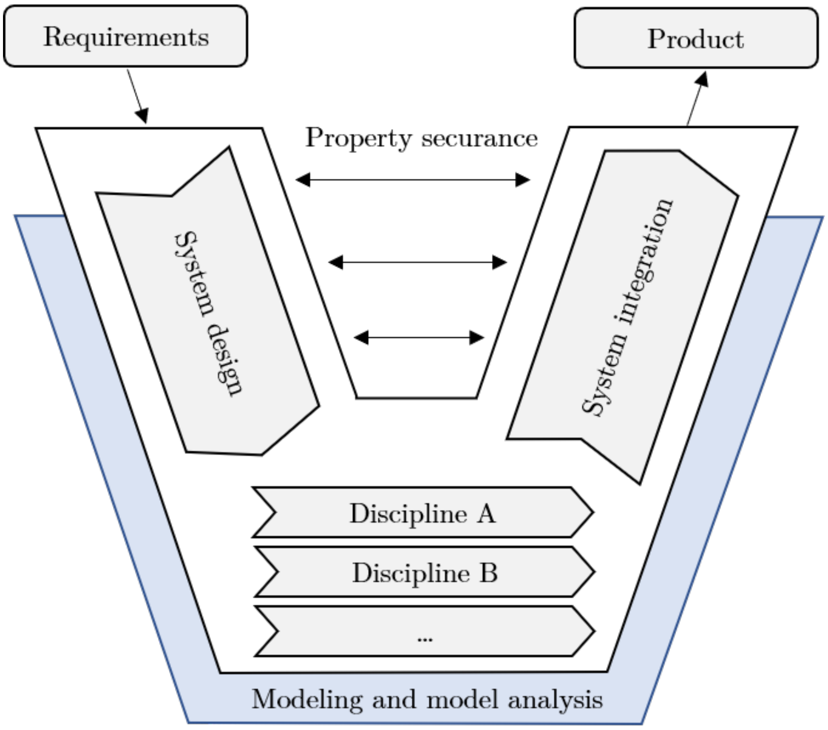


Figure 2.7: V-model *inspired by VDI2206*[53]

Requirements for development based on customer needs build the starting point of the v-model. As first step, in system design an initial system concept is developed, which is then detailed to deeper levels of the system structure (e.g., subsystems and components). After that, it is further processed by different involved disciplines (e.g., mechanic design, software design etc.). This procedure represents a top-down approach. When design and implementation are finished, system integration is carried out following a bottom-up logic to finally get to a well developed product. The v-model also emphasizes the importance of horizontal information transfer, which incorporates the idea to already consider system validation and verification tasks in system design in order to enhance the idea of early assurance of requirements fulfillment. The whole development should be supported by models, because of the reasons stated in section 2.2.1. For model-based approaches, the v-model is further evolved by associating tasks like model specification, simulation, virtual and hybrid tests to the classical v-model's procedure.[52]

Additionally, many variants of the v-model exist in literature as result of adaption of the principal procedure model to the specific needs of a certain discipline. It is also

---

[52] Zafirov 2014, p.86 ff.

[53] VDI2206 2004.

important to note, that the steps in the v-model define a procedure but not necessarily a process. Therefore, sub-sequences of the v-model can be followed iteratively (e.g., refining design after first verification tests).

In the further course of this thesis, the developed concepts and methodology are applied to an use case to investigate if they are adaptable without losing their core principles. An exemplary development task is defined in the field of tribology. Therefore, relevant fundamentals of tribology are described in the next section.

2.3 Tribology

The acknowledgment of tribology as an own field of science is relatively new. Of course, tribological phenomena like wear or friction are known since scientists like *Da Vinci, Euler* and others investigated them, but in these days, the focus was to understand friction from a mechanical point of view (e.g., to understand how friction can be reduced). In the twentieth century, the field of tribology was defined as an own field of science:

> *“Tribology is the science and technology of interacting surfaces in relative motion and of related subjects and practices”*[54]

Tribological development focuses on wear, friction, lubrication and its interactions, which are generally illustrated in figure 2.8. The interrelations between these aspects are complex and many factors like environmental conditions or material behavior affect them. After the mentioned *Jost report*, tribology became recognized more and more and its potential of improved system understanding, regarding tribological behaviour, is enormous.[55] For example, improved system durability, higher available performance, practically maintenance-free operation are typical objectives of tribological optimizations. Simply put, this is achieved by reducing wear and friction.[56] Nevertheless, friction has not to be avoided in all cases. Friction is often needed to transfer forces between different system elements or dissipation processes are used to transform energy (e.g., in brakes of a passenger car kinetic energy is transferred into thermal energy). In these applications, the objective is to increase the friction between the involved parts.[57] By reducing wear and friction, the world energy consumption can be reduced, as 23% of it is used to overcome friction and replacement of worn out parts. This has also huge impact on world wide emissions.[58]

---

[54] Jost 1966.

[55] Bunk et al. 1981, p.11.

[56] Czichos and Habig 2010, p.4 ff.

[57] Sommer et al. 2014, p.7.

[58] Holmberg and Erdemir 2017, p.1.

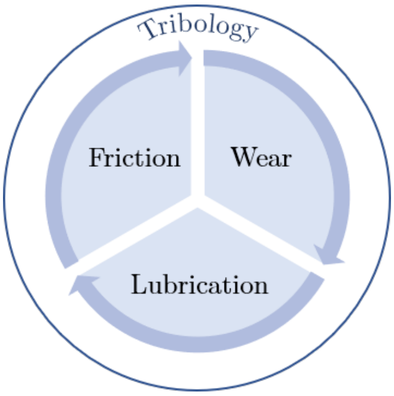


Figure 2.8: The three main aspects of tribology: friction, wear and lubrication

2.3.1 Tribological systems

Technical systems contain many tribological subsystems. As the definition of tribology stated, all relatively moved surfaced in contact or surfaces only separated by lubricants are tribological systems. Firstly, the structure of such a system is investigated.

In figure 2.9 a) the principal structure of a tribological system is shown. There are two solid elements (e.g., gear of a transmission), which are relatively in motion. These elements may be separated by a lubrication medium (e.g., oil). As a result of chemical reactions or pre-treatment, the solid elements may include a tribolayer, which is a solid surface layer with different properties in comparison to the base material. Tribolayers strongly influence the tribological behavior and may change over time (e.g., run-in phase). The environmental medium is also seen as a structural element, as it potentially affects all other elements (e.g., water vapour, dust etc.).[59]

To visualize the system structure in a more abstract way, figure 2.9 b) shows the elements as blocks representing elements with assigned properties, which are connected by lines to represent the interdependencies between them. In summary, the system structure consists of structural elements, its properties and interdependencies.

The functional description of a tribologial system can be illustrated as shown in figure 2.9 c). The system function is the transformation of an input X into an output Y. This process is influenced by disturbances, dissipation effects and occurring losses. For example, if the function of the system is to increase the temperature in the contact area to weld two parts together, the input is a normal force and a movement, while the output is heat dissipation in the contact area. This process is influenced by dust (disturbance), wear (loss) and dissipation effects which do not support the system function. Summarizing, a tribological system is characterized by its structure, its technical functions, the load collective and the dissipation effects like wear and friction.[60]

---

[59] Czichos and Habig 2010, p.8.

[60] GfT 2002, p.15.

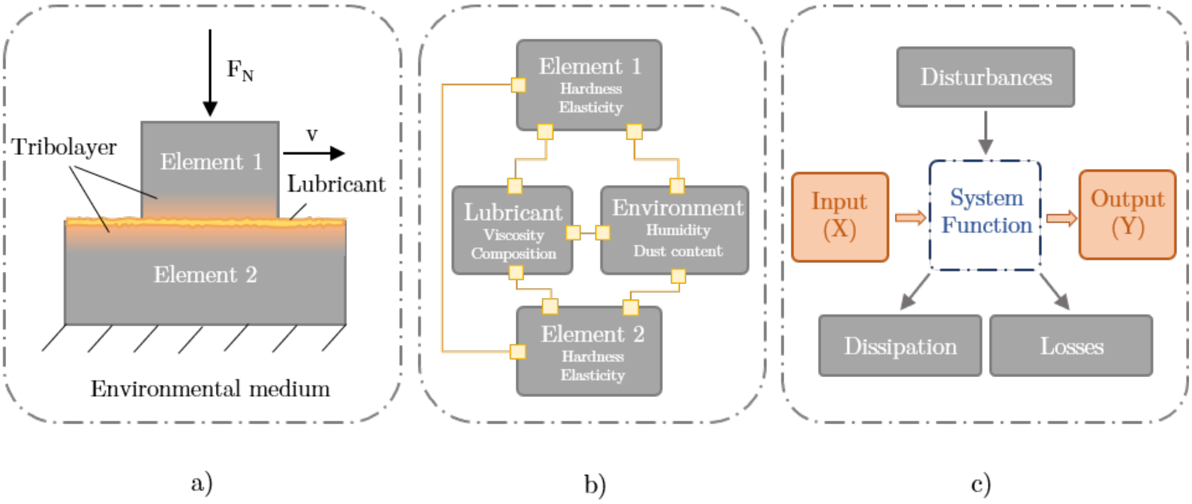


Figure 2.9: a) Generalized tribological system
b) System structure with interdependencies and properties
c) System function and influencing factors *inspired by Czichos and Habig*[61]

2.3.2 Complexity of tribological systems

Tribological effects are complex because of their interdisciplinar and multiscale nature. In development of mechatronic systems, the target is often to improve performance and reduce losses such as friction. Therefore, the investigated effects are mainly macroscopic. The interdependencies of tribological behavior on multiple scale levels - from nano to macro - make the development of such systems challenging. For example, the friction between a piston ring and a cylinder liner is influenced by the atomic forces on nano scale between the surfaces, which depend on chemical depositions on the surface and lubrication, but also on the microstructure of the surface material. In order to understand the different effects and to improve the behavior of the whole system, many discplines are involved, from chemistry (e.g., tribolayer formation), over materials science (e.g., disclocations, material properties, microstructure) and contact mechanics (e.g., contact of asperities) to applied sciences (e.g., surface coatings).

Figure 2.10 illustrates this interdisciplinarity over the relevant scales of tribological effects.

2.3.3 Friction

Basically, friction is a resistance against movement. It occurs as friction force between two surfaces, and acts as a counter reaction to the introduction of motion or as resistance against an continuing motion. There is also an inner friction within materials,

---

[61] Czichos and Habig 2010, p. 18.

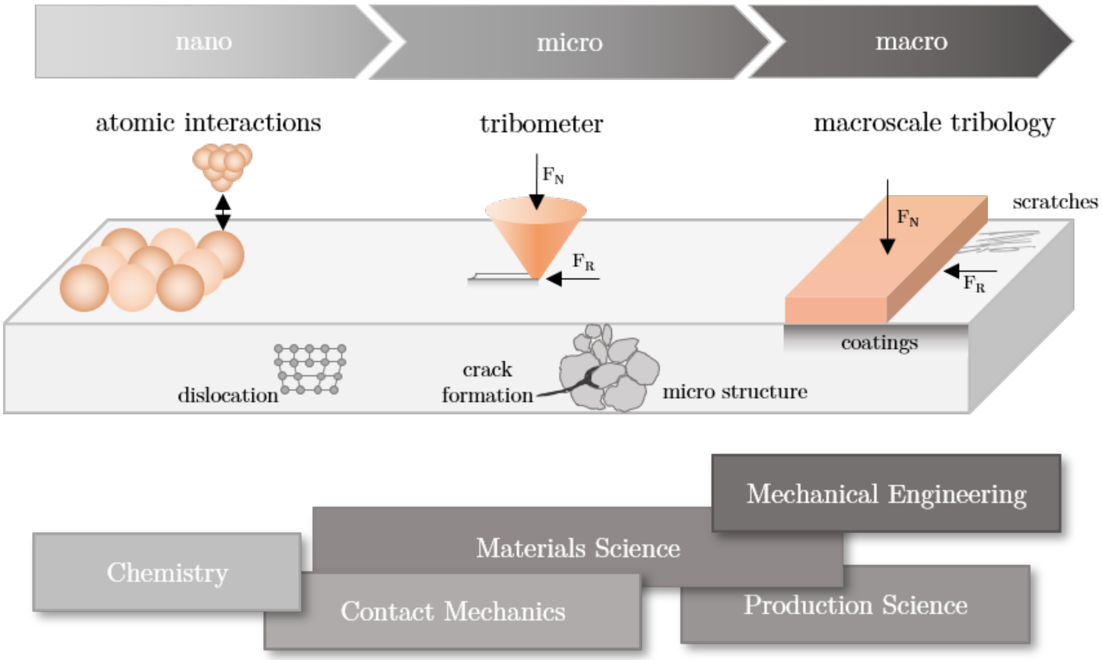


Figure 2.10: Tribological effects and applications on different scales and allocated disciplines *inspired by Lehigh Univeristy* [65]

which is characterized for fluids with the property *viscosity*. The viscosity of a fluid is influenced by many factors like temperature, shear stress, contamination and fluid pressure.[62] Early concepts to describe friction forces date back to *Coulomb*, who continued the research of previous contributors like *Da Vinci* and *Euler*. The friction model of *Coulomb* describes the occurring friction force as linearly dependent on the normal force with a proportionality factor called *coefficient of friction*.[63,64]

---

[62] Czichos and Habig 2010, p.81 f.
[63] Coulomb 1821.
[64] Sommer et al. 2014, p.7.
[65] Lehigh-University 2019.

$$F_R = COF * F_N \tag{2.1}$$

F_R ... friction force
COF ... coefficient of friction
F_N ... normal force

This fundamental formula can be seen as a *macroscopic* description of friction. For coarse approximations or quick estimations, it might be reasonable to use literature values for the coefficient of friction (mostly based on material pairing and lubrication). Nevertheless, it is important to understand, that this coefficient is strongly related to the system and is affected by relative movement, contact situation, materials, lubrication and many more influencing factors and can only be described as a *system property* and not just as independent material property.

Considering friction from a microscopic point of view, it is important to note that energy conversion processes mostly take place in surface boundary layers where physical and chemical interactions between the surfaces occur together with related surface and material transitions.[66] Based on the mechanisms of lubrication film formation, different lubrication states can be distinguished.[67,68] Table 2.1 provides short descriptions for each lubrication state.

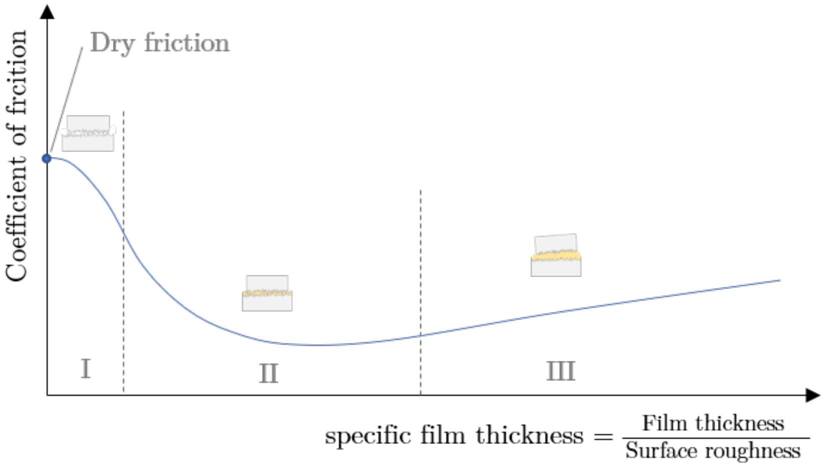
The described states depend on a comparative value, which includes the film thickness, surface roughness and other properties depending on the application (e.g., journal bearing carrying load). In figure 2.11, the coefficient of friction, which is the ratio of the friction force and the normal force, is principally shown as function of the lubrication states.[69] As described earlier, this is a system property and therefore depends on the used materials, lubrication medium, environmental circumstances etc. The hydrodynamic lubrication state, where the contact surfaces are completely separated by a lubrication fluid, is especially important in the context of this thesis. The focus is laid on elastohydrodynamic (EHD) approaches in the further course of this thesis.

---

[66] Sommer et al. 2014, p.7 f.
[67] Wen and Huang 2018, p.3 ff.
[68] Czichos and Habig 2010, p.81 f.
[69] Czichos and Habig 2010, p.81 f.

Table 2.1: Description of different lubrication states[70]

Lubrication state	**Description**
Dry friction	Contact of two dry surfaces without lubrication
Boundary lubrication (I)	Friction occurs between the surfaces of the contact partners. So called tribolayers on the surface differ from the base material regarding chemical composition.
Mixed lubrication (II)	Inner friction within the lubricant as well as friction between solid surfaces occur.
Hydrodynamic lubrication (III)	This state is described with elastohydrodynamic (EHD) approaches, which combine, inter alia, hydrodynamic influences and rheology.



I...Boundary lubrication
II...Mixed lubrication
III...Hydrodynamic lubrication

Figure 2.11: Lubrication states of a system. Dry frcition, Boundary lubrication (I), Mixed lubrication (II), Hydrodynamic lubrication (III))[71]

---

[70] Wen and Huang 2018, p. 3.
[71] Czichos and Habig 2010, p. 81.

2.3.4 Wear

The term *wear* includes different types of surface damage and removal of material from one or more surfaces of solid elements, which are in relative motion to each other (sliding, rolling or impact motion). Typically, wear occurs through interactions (mechanical or chemical) of asperities of the solid surfaces, which are influenced by frictional heating.[72] These energy dissipation effects in the surface layer of a solid part influence the material loss. These interdependencies should be considered in the system design phase (material selection, surface treatment, etc.) and understanding of them supports the evaluation of failures as well.[73]

As shown in figure 2.12, two main types of surface interaction can be distinguished. While mechanical interactions mainly depend on the direct contact situation either between two surfaces with or without abrasive particles, molecular interactions are caused by forces between the surfaces. In tribology, the relationships between different factors (e.g., mechanical, structural, physical, chemical) are complex and depending on the system and boundary conditions, they affect the processes more or less. These tribological processes result in different forms of surface damage - from abrasion, pitting, spalling, scuffing to micro wear.

These wear modes are interrelated with friction and lubrication, as dissipating friction heat and lubrication as convective cooling fluid influence wear - especially scuffing - by changing surface material interactions. Also, wear particle act abrasively and can therefore cause further wear.[74,75]

2.3.5 Lubrication

In order to reduce friction and wear, a separation of the contact surfaces is meaningful, which is done by a lubrication medium. This medium can be either gaseous (e.g., air), liquid (e.g., oil, water) or solid (e.g., graphite), but in most technical applications oil is used as lubricant.[76] The enormous influence of lubrication on occurring friction between two solid parts is described with the *Stribeck curve* in figure 2.11. Lubrication is affected by the medium and its properties and the resulting fluid flow in the gap.

The calculation of the fluid flow is based on *Reynolds' hydrodynamic lubrication theory*[77], which is a differential equation derived from *Navier-Stokes equations*[78]. Assumptions regarding the lubrication gap (gap distance much smaller than dimensions in flow direction), fluid pressure (constant in normal direction to flow), the fluid behavior

---

[72] Bhushan 2013, p.447.

[73] Sommer et al. 2014, p.14.

[74] Wen and Huang 2018, p.283 f.

[75] Kragelsky et al. 1977.

[76] Czichos and Habig 2010, p.163 f.

[77] Reynolds 1886.

[78] Stokes 1845.

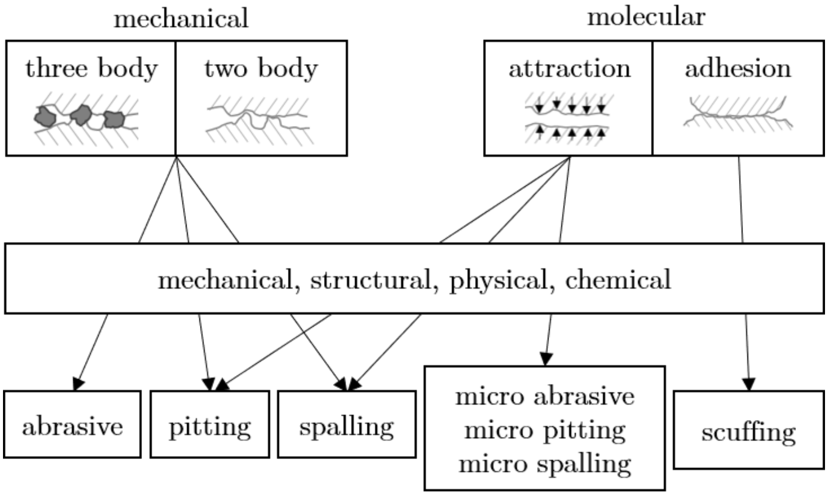


Figure 2.12: Classification of different types of wear *inspired by Weng and Huang*[82]

(no fluid inertia, stick condition) and the surface topography (ideally flat) are made to simplify these basic equations.[79] For tribological applications, these equations were adapted to include effects like micro flow.

Lubricants are characterized by the properties density and viscosity. These characteristics have huge influence on the fluid behavior in the lubrication gap.[80] The investigation of fluid properties, especially its dynamic viscosity, is called *Rheology*.[81] The influence of temperature, shear stress, pressure and other factors on the fluid behavior are of interest.

---

[79] Czichos and Habig 2010, p.182 f.

[80] Wen and Huang 2018, p. 6 ff.

[81] Czichos and Habig 2010, p.81 f.

[82] Wen and Huang 2018, p. 283.

3 Conceptual Framework and Methodology

To outline the general procedure of this thesis, as shown in figure 3.1, several principles are considered. First of all, the objectives and motivation behind this thesis are clarified. To develop an approach for development method selection based on models different questions have to be answered:

- How can a set of development methods and models be derived from customer needs?
- Which types of models can be distinguished and what is the scope of these different types?
- How do models and methods interact and what are the differences between them?
- Which types of development methods are used in model-based development and how can they be differentiated?

Figure 3.1 illustrates the principal procedure to answer these questions in the course of this thesis. The mentioned questions define the procedural steps on the left side of figure 3.1. Current understandings and approaches are analyzed and based on that a new methodology is developed. Furthermore, definitions for important terms are described to establish common understanding. In section 4, the developed concepts are applied on an use case to be able to evaluate its advantages and disadvantages. This procedure has similarities with the v-model and takes up several principles from systems engineering as well. Firstly, a concept is defined in which the understanding about models and methods is further detailed for the scope of this thesis. These steps follow a top-down approach as they start with an overall concept and then further detail definitions and classifications in the further course of this thesis. The implementation of the defined concepts is done by applying them to an use case. This is done iteratively following a bottom-up approach to improve the concepts with conclusions drawn from its application to tribology. Furthermore, the procedure is structured into phases (macro logic) to define feasible tasks.[1,2]

---

[1] VDI2206 2004.

[2] Haberfellner et al. 2015, p. 55.

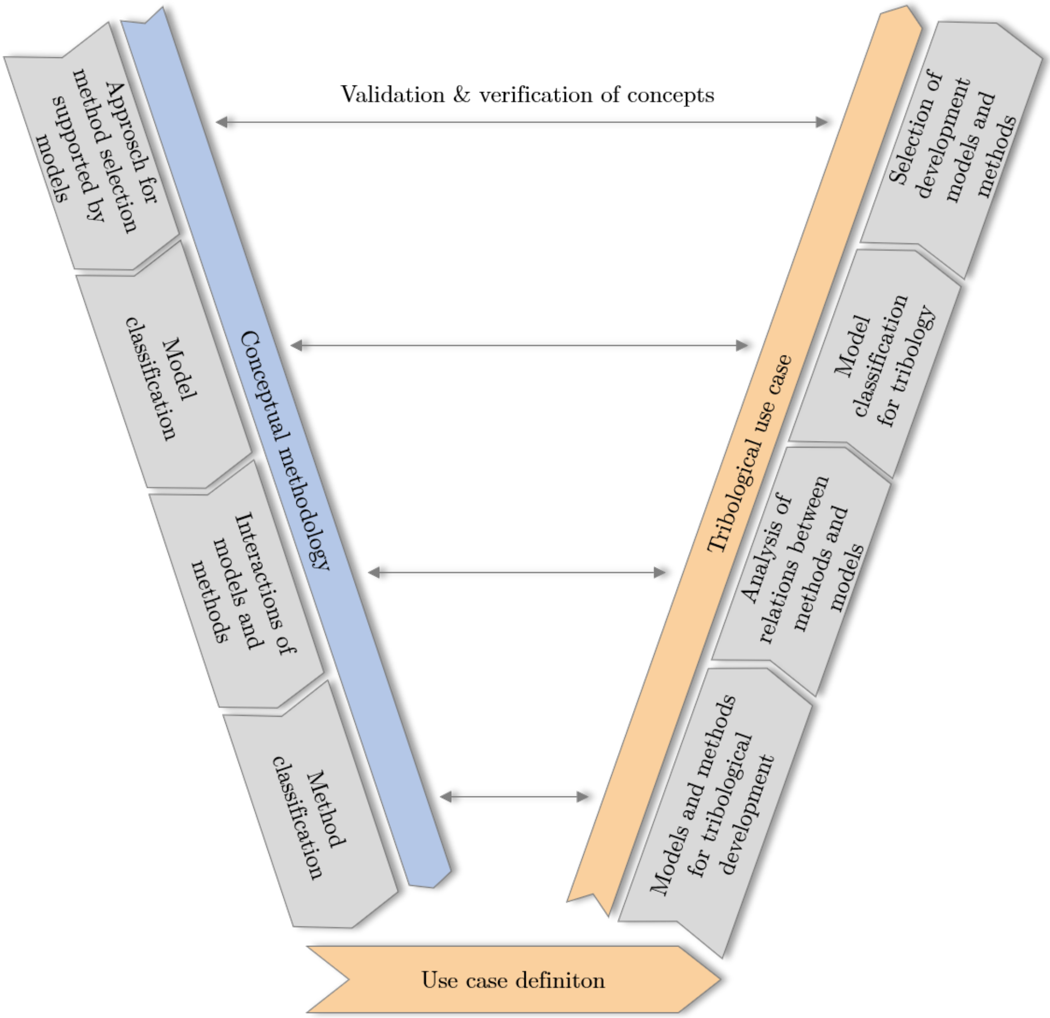


Figure 3.1: Procedure to develop and validate a method selection approach supported by models

3.1 Approach for method selection supported by models

As a first step, a general approach is described to connect models and methods in development in order to choose the appropriate ones. This should support decision making about what methods and models are used in the development of a system. In figure 3.2, a logical approach is visualized to derive the required models and methods from customer needs.

This approach uses an analogy to *Quality Function Deployment (QFD)*, which was

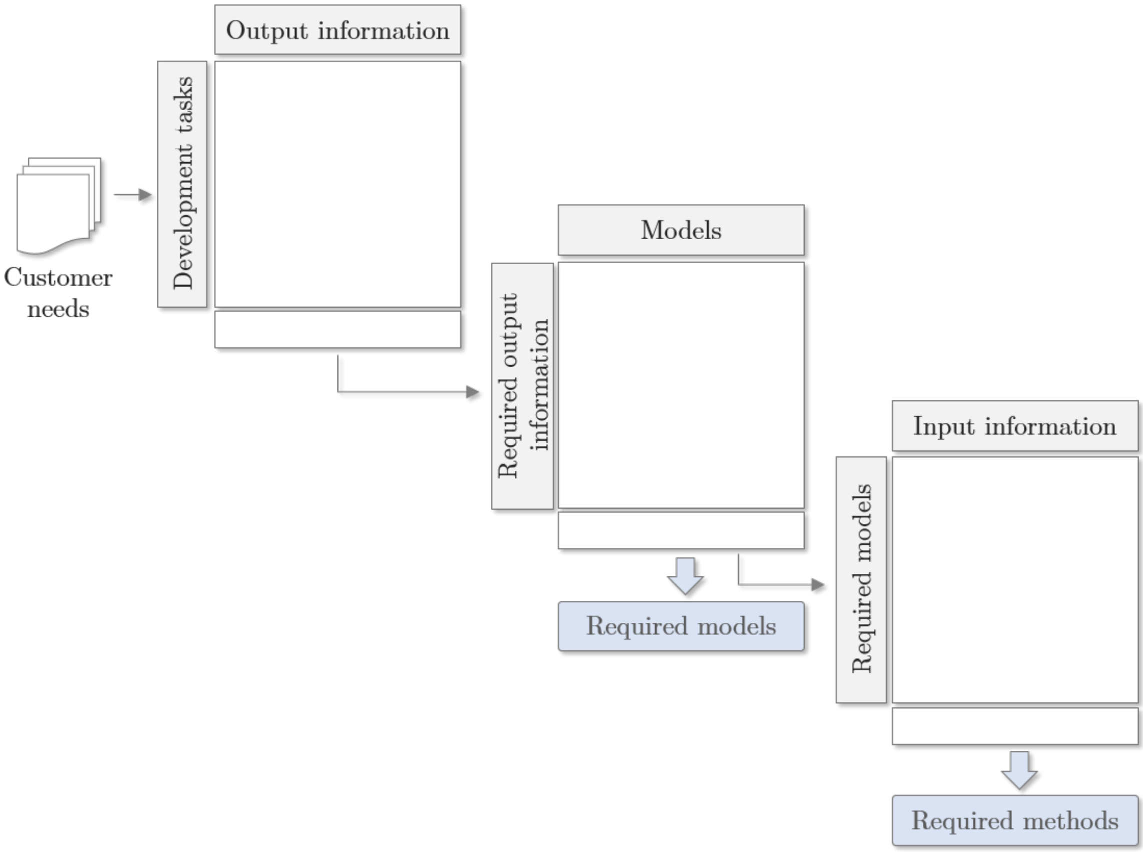


Figure 3.2: Approach for assessment of required models and methods in development based on customer needs

introduced by researchers in Japan in the twentieth century.[3] It is a customer-oriented method to ensure quality throughout product development by deriving product properties from customer needs. Several adaptions of this concept are well-established, but they all have the core idea of step-wise derivation and evaluation of product characteristics based on customer requirements in common.[4]

Figure 3.2 shows an adaption of this approach for model-based engineering. As a first step, customer needs are compared with possible output information (information about a system, which is an output of development methods) in order to asses which is needed to evaluate if customer needs were satisfied. For example, if the system under development is a vehicle powertrain and a customer demands *sporty acceleration*, the development task is to *assure acceleration from 0-100kph in 8sec.* In the first comparison, every system information which is associated with this development task is marked (e.g, torque, engine speed, mass). After this required output information is

---

[3] Akao 1992.

[4] Maritan 2015, p.11 f.

determined, the next step is to asses which models cover this information. As there exist numerous models in product development, an approach for classification of models and visualization of its contents is needed. This is discussed in section 3.2. Next, models are evaluated, if they contain the necessary information about the system or not. A set of required models is identified in this step. These identified models (e.g., torque-speed-model and geometry model) need to be *filled* with information in order to provide the required output information, which is needed to asses if customer needs were satisfied. The information described with these models is generated or transferred from other models, which is the task of methods. For example, a map which shows the engine torque as function of the engine speed (which is a formal model), has to be generated by *filling* it with data (measured torque values for different operating points). In the next step, the required model inputs have to be identified. For the example of a torque-speed-model, there are several possibilities to generate this model. In a classical development approach, engine test beds are used to acquire the required data. In virtual development, a proper method would be a combustion simulation which generates the required model. In order to identify applicable methods, a classification scheme and an assignment of methods and models in it is necessary. Section 3.3 and section 3.4 cover these tasks.

It is important to note, that it is not meaningful to select methods solely analytically, for that might not be feasible in complex development environments. The main reason for this is, that in large engineering departments the number of available methods is huge (this of course depends on e.g., the type of system under development or the experience of the engineers with a certain product). Also, more and more software solutions and IT-tools are purchasable which implicates that a decision for a set of development methods has huge impact. This approach serves as *navigation method* to find a path in a complex and interconnected map of models and methods. It is also important to look at this concept from different points of view. On the one hand, this approach may be used before starting a new product development in an organization without previous experience to support the selection of right methods. On the other hand, it can be used in existing engineering departments to make development more effective by focusing the development efforts on the most important methods. In both cases, this approach provides a pre-selection of methods and models, but the final choice depends on the skills of the engineers, financial resources (to invest in new development methods and IT-tools), already existing methods and other aspects. Therefore, another important and not avoidable influence is the so-called *human factor*. The way how a human uses a machine/an IT-tool/a simulation software depends on many factors (e.g., professional experience, collaboration in a department, corporate culture) and influences the outcome of a method essentially.[5] This affirms that socio-technical systems include humans as an integral part of the system. As already mentioned, it is not always possible to select methods analytically, as there may be more than one applicable method available

---

[5] Guastello 2014, p.2.

to describe a required system information. Mostly unconsciously, decisions are based on the belief that a chosen development path will be successful. It is important to understand, that many decisions in technical development are affected by credition (a term established as analogy to emotion and cognition for believing processes).[6] Recent research on processes of believing indicate, that there is no clear separation between cognition and emotion. Model-based approaches support decision making by providing information in form of models, which describe important aspects of the system under development (e.g., critical interdependencies).

3.2 Classification of models in development

As already illustrated and described in section 2.1.3, different model types can be distinguished. Exemplary, figure 3.3 a) shows a race car engine developed by *Porsche*[7] as system under consideration. Figure 3.3 b) illustrates different types of models to describe this system. A physically existing model would be a prototype of the engine, which can be used as test specimen (but which has not all of the properties of the produced engine in the final specification). A document-based approach describes dimensions and physical properties in form of textual documentation, which represents an informal model. Model-based development is based on the use of semi-formal and formal models, in this case a geometry model of the crank train which also includes information about dimensions, mass properties, kinematics, etc. As this thesis is based on the core concepts of model-based development, only formal and semi-formal models are further investigated. In this context, semi-formal means that these models obey a certain syntax and include notations but most of its contents is described using natural language. In contrast, formal modeling languages include syntax, notations and semantics, which are all formally defined.[8,9,10] The purpose of both, formal and semi-formal models, is to manage complexity by providing the following advantages:[11]

- *Maintain data consistency*: Connected data repositories and interconnected models enable a consistent system description.
- *Assure Traceability*: Through extensive investigation of interfaces between models, it is possible to trace information flow better in development.
- *Provide re-usability of information*: In comparison to informal and physical models, formal and semi-formal models can be re-used by adapting it for similar systems.

---

[6] Angel 2015.

[7] Porsche. 2016, p.1.

[8] Liu 2004, p.239.

[9] Koenig 2011, p.34.

[10] Friedenthal and Moore 2012, p.371.

[11] Madni and Sievers 2018, p.172.

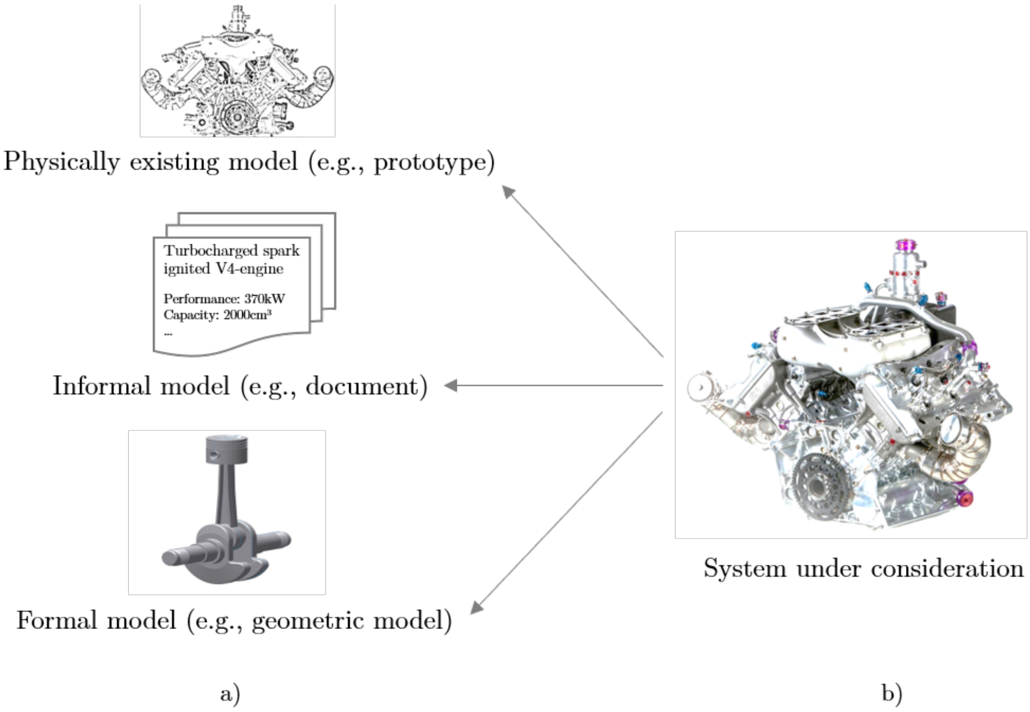


Figure 3.3: a) Principle illustration of different model types
b) Race engine developed by Porsche as exemplary system under consideration[14]

One of the main advantages of semi-formal and formal models is the fact, that they minimize the space for different interpretations of the meaning. In contrast, human language always leaves room for interpretation. In theory, virtual models - formal and semi-formal - provide explicit information described with standardized formalisms, which make it readable and understandable for humans as well as computers.[12,13]

In model-based development approaches, virtual models (sometimes also referred to as digital models), which are generated on computers, are used. As these models are processable by computers, they act as enablers for a more efficient development. (e.g., supporting early verification and validation and identification of critical interfaces). For example, verification and validation (V&V) activities and optimization projects are supported by the use of models and a (semi-) automatic documentation of products or systems can be implemented.[15,16] There exist several terms which are used similarly. Human language is often not precise, which causes different understandings

---

[12] Madni and Sievers 2018, p.175.

[13] Koenig 2011, p.35 ff.

[14] Porsche. 2016, p.1.

[15] Hick, Bajzek, et al. 2019.

[16] Zafirov 2014, p.81 f.

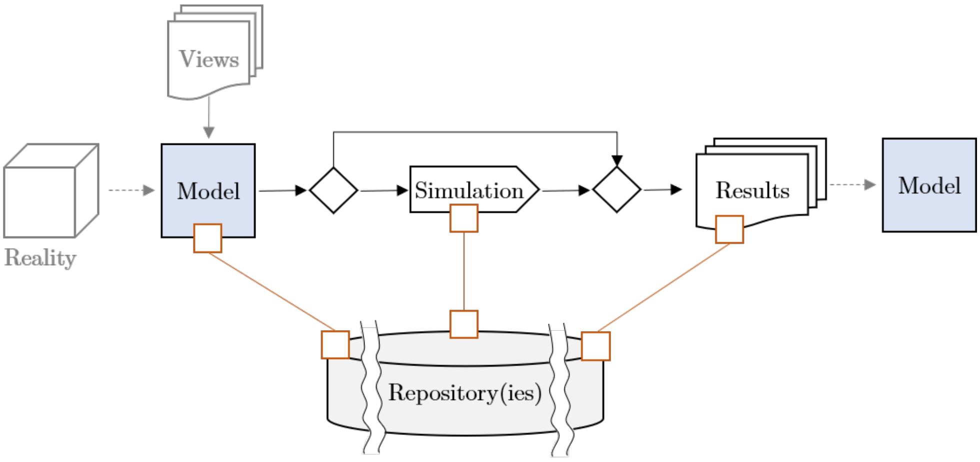


Figure 3.4: Differentiation of models, simulations and repositories *inspired by Hick et al.*[19]

of the same term. Models, methods, processes and tools were already distinguished in chapter 2. The interrelations of development models with simulation and repositories are visualized in figure 3.4. A model is made up of several views of a real system, which collectively describe its origin. Virtual models are stored in repositories and may be interconnected via simulations. There are numerous meanings of simulation in literature. According to *Friedenthal et al.*, a simulation is composed of a model and a set of initial conditions. The simulation engine has the purpose to create an instance of the model in the simulation environment, applies the initial conditions to that instance in order to determine the change of state as function of time.[17] The results of a simulation may represent models itself (e.g. fuel consumption map). Repositories act as data bases which collect data and information of models, simulations and results. The figure also shows the principle that not every model is created to be used for an ongoing simulation, but it hides further methods which are necessary to generate a model (by transferring the reality into a model), methods to *fill* a model with information and methods to provide exchange between models.[18]

3.2.1 Analysis of as-is situation

Referring to section 2.2.1, system models are essential for model-based systems engineering approaches. Several authors agree that system models are the most important attribute which distinguishes model-based systems engineering (MBSE) from systems

---

[17] Friedenthal and Moore 2012, p.526.
[18] Hick, Bajzek, et al. 2019.
[19] Hick, Bajzek, et al. 2019.

engineering (SE)[20,21] , there is no common understanding of what a system model actually is. Therefore, no consensus about system models, their boundaries and objectives exist. Some describe it as a structured representation of a system containing several different aspects as requirements, structure, properties and interconnections.[22] Other authors state, that system models are combinations of several models and repositories, which are able to act as one single model or central repository.[23] In order to act as repository connected to several combined models, data management tasks and clearly defined responsibilities between the different entities are required.

The reasons for this disparity of what a system model represents are different. The uncertainty associated with human face-to-face communication is one reason for that. Most of the time, human language leaves room for interpretation, which implies that no clear meaning and application area of a certain term can be defined. Especially, the terms *systems* and *models* are often used. As described in chapter 2, it depends on the point of view what a system is (e.g., a vehicle as well as an engine). A model is a representation of reality with properties as defined by *Stachowiak* (mapping , reduction and pragmatic property[24]). But not every representation of a system is automatically a system model. Therefore, it is important to establish a common understanding and clear definitions about system models.[25]

Another reason is that the term system model is used in various technical disciplines, from software engineering over mechanical engineering to business economics. As a result of that, the term represents different meanings according to its application. To understand the similarities and differences of system models in different disciplines, a few examples from literature are given in the followings.

In software engineering, system models are used as graphical representations of the software system, its elements and interactions with the environment and are mostly developed in the course of requirements engineering[26] and system design. The process of developing a system model is consequently called *system modeling*, where abstract models of a software system are generated. Each model represents a different view or perspective of the considered system. System modeling often uses graphical notations or mathematical methods to develop formal and semi-formal models.[27]

Systems thinking inspired approaches in management and organization, which have strong connections to system dynamics. They often implement semi-formal or formal system loop diagrams to visualize impact of decisions and market developments.[28]

---

[20] Friedenthal and Moore 2012, p.523.
[21] Zafirov 2014, p.81.
[22] Hart 2015, p.18.
[23] Weilkiens 2014, p.22.
[24] Stachowiak 1973, p.130 ff.
[25] Hick, Bajzek, et al. 2019.
[26] Lamsweerde 2009, p.3.
[27] Sommerville 2016, p.137 f.
[28] Buerlow n.d., p.119.

In the automotive industry, which often sets standards for technical engineering of mechatronic systems in general, system are structured on several levels, from the vehicle system over powertrain systems to subsystems and components. System models are used on different levels of technical detail and for different states of system maturity in development to document requirements, structure, behavior and verification & validation.[29,30]

Established approaches in product development are analyzed in the followings to build a base for common understanding of the term *system model.* As it is generally agreed that a standardized modeling language is an essential enabler of model-based system engineering (MBSE), many authors see the *System Modeling Language (OMG SysML^{TM})* as the appropriate solution.

> *"SysML is a graphical modeling language with a semantic foundation for representing requirements, behavior, structure, and properties of the system and its components"*[31]

It consists of four content-related pillars: requirements, structure, behavior and parametrics and is based on the *Unified Modelling Language (UML)*[32] The fundamental structure of SysML is shown in figure 3.5, were the four content blocks and interrelations between them are idealized. Many publications claim that system models follow the same principles and contain the same information categories as SysML [*] and further imply, that SysML is the only possibility to create system models.[33,34] With this approach, potentially useful views on a system (e.g., verification and validation test cases) might be ignored. It is also not possible to implement such a system model as executable model. Nevertheless, SysML provides semi-formalized modeling language and therefore enables better system understanding and a base for stakeholder communication by uniform notations, semantics and syntax.[35] It has to be stated, that SysML is just one way of system model generation, but system models should not be limited to SysML. In the following chapters, SysML based system models are integrated in a broader definition of system models and will be considered in a wider context.

In some approaches to implement model-based development, a system model is claimed to be a central model, which acts as a *single source of truth* (also referred to as sole source - or single point of truth). It is a logical conclusion, that an entity in a central position could provide 100% traceability and data consistency. In some publications, approaches are described where the task to manage all occurring engineering

---

[29] Hick, Bajzek, et al. 2019.

[30] Albers et al. 2007, p.16.

[31] Friedenthal and Moore 2012, p.XVII.

[32] Friedenthal and Moore 2012, p.17.

[*] SysML in its current version

[33] Albers et al. 2007, p.16.

[34] Hart 2015, p.18.

[35] Hick, Bajzek, et al. 2019.

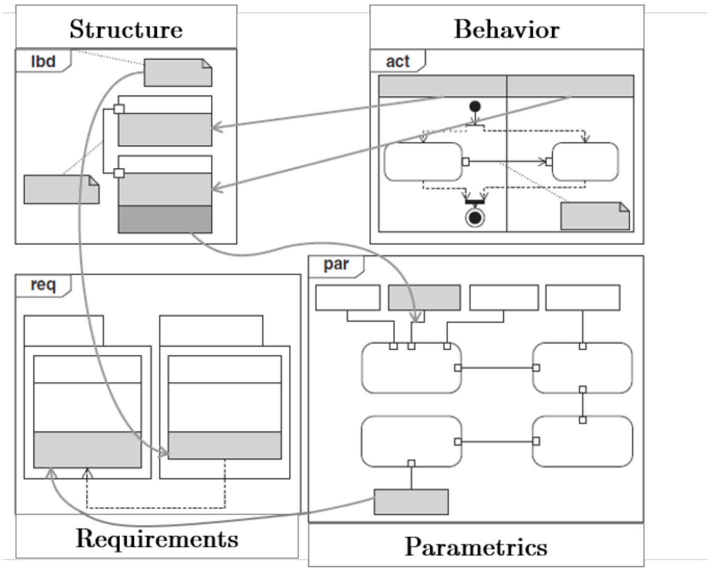


Figure 3.5: Illustration of SysML structure and content-related pillars[42]

data about a system are handled by a central system model. As SysML models are not able to fulfill these tasks, new concepts like the *total system model* were developed. In this exemplary approach, a central system model is complemented with tools for data management.[36] Many approaches already tried to provide some kind of single source of truth or single repository. An early concept for this purpose was the *engineering backbone*.[37] The developed ideas were further evolved and lead to *product lifecycle management (PLM)* approaches. The objective of PLM is to manage all product relevant data and information over the whole lifecycle of a system or product.[38] This is done to provide consistent information flow over the whole lifecycle of a product. Important to note is that PLM is not a finished IT-solution, rather it is a way of thinking to connect all department specific repositories, which emerged over time.[39] The lifecycle of technical systems and software systems is defined by the *International Organization for Standardization (ISO)* and includes all activities from concept, development, production, utilization, support until retirement.[40] The rising flood of available data and information about products, markets and customers triggers new applications in the area of *Big Data*. The requirements for big data approaches show the challenges

---

[36] Bajaj et al. 2016, p.5 f.

[37] Eigner and Stelzer 2009, p.44.

[38] Peschke 2017, p.30.

[39] Rudolf and Schrey 2015, p.3 f.

[40] ISO/IEC/TS24748 2018.

[41] Fels et al. 2015, p.263 ff.

[42] Friedenthal and Moore 2012, p.17.

of state-of-the-art data management.[41] System models cannot replace such approaches and especially SysML based approaches cannot cope with the challenges of increasing data variety. Therefore, intelligent integration of different concepts like product lifecycle management, development models and data management concepts is essential to create an added value for engineers, customers and other stakeholders. It depends on the objective of its application what purpose a system model has to fulfill and how it should be implemented in an engineering process. In conclusion, it is important to connect different approaches, because it is not feasible at the moment to implement a single source of truth in development.

3.2.2 Requirements for a system model definition

Based on the as-is analysis discussed in section 3.2.1, general requirements for a definition of the term *system model* are derived. With this as starting point and with the generic definitions of the term *system* and *model*, a definition of system models for model-based development can be outlined. It is not the aim of this thesis to state that other concepts are wrong. The definitions should enable fundamental understanding, established point of views should be incorporated or derivable to allow the use of them in further research activities. It should enable better understanding of the term *system model*, its possible applications and further connection to other types of models and methods. Moreover, it is not sufficient to only provide a global definition, as the objective is to develop a concept about system models that is adaptable to its application. Additionally to general valid definitions and a clear terminology, the positioning of a system model in a wider context of a development process is presented. In chapter 4, the concepts are applied to a use case in order to understand shortcomings and advantages of it. Apart from these general requirements derived from the as-is situation, there are basic requirements all kind of models have to fulfill. According to *Madni and Sievers*, there are two concepts to ensure the quality of an implemented model. *Model validation* checks if the model corresponds to a real system and if external requirements (e.g., incorporated standards and applicability) are fulfilled. *Model verification* ensures that a model has been implemented correctly. Correctness implies completeness, consistency and traceability.[43]

- *Completeness*: A model is complete if all elements, relationships, parameters, inputs, outputs, processes and constraints, which are relevant for its application, are sufficiently specified. It must be able to reflect the needs of involved stakeholders.

- *Consistency*: In order to provide a model, which is consistent with established concepts and definitions, general standards have to be implemented.

---

[43] Madni and Sievers 2018, p.175.

- *Traceability*: Concepts, relationships and results have to stay in line or should be derived from common standards, specified requirements or guidelines.

These requirements are valid for all kind of models and also for the developed concept of a system model. Of course, the validation can only be done if there is a real system the model corresponds to. This will be done by applying the developed concepts to a use case in order to validate the definition of a system model. Finally, the purpose of system models also defines requirements for its definition. The general advantages of a model-based approach were already discussed extensively in the previous sections. But what makes system models interesting and motivates a deeper investigation? Increasing product complexity, high number of involved disciplines (mechanics, software, electrics/electronics etc.), increasing amount of available data (of market, customers, development, suppliers etc.), the need for flexible production lines and many more trends make it impossible to provide detailed information about every part as well as overall structure and functions of a system with just one model. It has to be investigated, how a system model can provide general views on a system while interacting with other models which provide depth and detailed information.[44] According to *Zafirov*, requirements, structure, behaviour, parameters, context, validation and verification represent a set of modeling concepts, that incorporate various views of different disciplines.[45] These views have to be incorporated in a system model definition as well.

3.2.3 Definition of the term system model

The term *system model* is not just a simple combination of the terms *system* and *model*. Adapting the principles of *Aristotle*, a system model is more than just the sum of these two terms.

In section 2.2.1, the importance of system models for development approaches like model-based systems engineering, which build on system models as primary artifacts, was outlined. Therefore, it has to be investigated which purpose and scope system models have. According to *Friedenthal et al.*, a system model is generally used to describe multiple views on the system at a fairly abstract level. The scope of a system model is not to describe a single part of a system in detail but it has a broader scope. Other types of models focus on particular aspects of a system to represent them in detail. It is especially important to keep overlapping areas of models consistent.[46] In order to be able to distinguish different model types, it is important to clearly define applicable differentiation criteria. Firstly, it has to be defined what a *broad scope* means in the context of development of mechatronic systems. There exist a lot of models in science (physical, mathematical etc.) which describe physical or chemical phenomena. Consequently, these models provide *depth*, which means that processes or parts of a

---

[44] Hick, Bajzek, et al. 2019.

[45] Zafirov 2014, p.83 ff.

[46] Friedenthal and Moore 2012, p.526 f.

system are described in detail, and not *breadth*. In this context, breadth stands for holistic views on a system.

Following the systems engineering principles, a top-down approach leads from rough to detail (e.g., from the overall system over subsystems to components). System models are often only used on system level, but this depends on the point of view. Looking at the v-model described in 2.2.3, it has to be discussed which types of models are used on which levels of the development procedure. System models naturally have the purpose to *describe* the system under consideration to understand the interplay of its sub-elements in order to fulfill a greater objective. Generally, system models are used to support design. For complex products, it is meaningful to use system models on different levels (e.g., a system model describing the functional structure of an engine and a system model of the whole powertrain structure). Figure 3.6 shows that if this principal procedure is cascaded from system over subsystems to components there does not have to exist only one system model in development. The v-model, as defined in *VDI 2206*, can be drawn for each level. Therefore, a v-model of the whole product development can be seen as a *global v-model*, while further *local v-models* can be set up for each level.[47] As mentioned in the introduction, the context of this thesis is the development of socio-technical systems. In this type of systems, functions are typically performed by hardware which is controlled by software and humans.[48] Mechatronic systems are socio-technical systems and *VDI 2206* mentions that the development of such mechatronic systems include mechanical engineering, electrics/electronics and information technologies (IT).[49] System models need to be able to act as communication platform for different involved disciplines by combining views of several disciplines on the same system. System modeling generally has the objective to include aspects, which are externally visible like behaviour and properties of a system as well as internal aspects like architecture. Thus, the contents of a model may include information about requirements, structure, behavior, paramaterics, validation & verification (V&V).[50,51] These different aspects of a system or types of model contents can be summarized as *technical domains.*

In conclusion, a system under development is considered on different levels of detail, by multiple disciplines (mechanics, electrics/electronics, software, etc.) and with several technical domains (requirement, structure, behavior, V&V, etc.). Each combination represents a certain view on a system and the wholeness of views embodies every possible system information (e.g., system level - mechanics - structure). Models cover different views and therefore system information regarding their scope. A system model is a combination of multiple views. Extending the comprehension about system models of *Friedenthal et al.*, system models have the scope to provide breadth and

---

[47] Hick, Bajzek, et al. 2019.

[48] Kossiakoff et al. 2011, p.361.

[49] VDI2206 2004.

[50] Zafirov 2014, p.83 f.

[51] Hybertson 2009, p.78.

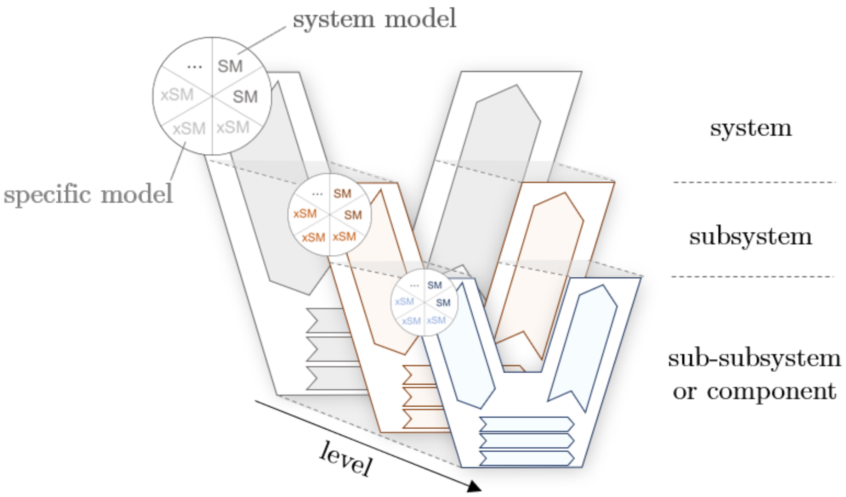


Figure 3.6: Positioning of system models (SM) and specific models (xSM) in the v-model *inspired by Hick et al.*[53]

width and not fidelity in all aspects.[52] Models, which aim to provide high level of detail are called *specific models*.

System models incorporate multiple views in breadth and width by including the views of at least two technical domains or disciplines. The objective of *specific models* is to provide more depth than a system model by representing the view of a single discipline on a single technical domain in detail.[54]

For example, a model generated using SysML represents a system model as it usually includes several technical domains (requirements, structure, behavior and parametrics). The view of a mechanical engineer on the structure of a system and the design of its subsystems and components, which is usually modeled using *Computer aided design (CAD)* tools, represents a specific model. Both types of models are needed in development and build the foundation of model-based approaches. As a system model combines several views to provide a better system understanding, the idea of one single model, which combines all views on a system may be feasible in the future, but this is far from practical realization. This also supports the idea of several existing system models in parallel in a connected development environment. Although, centralized thinking is important and is supported by the use of system models.[55]

---

[52] Friedenthal and Moore 2012, p.526 f.
[53] Hick, Bajzek, et al. 2019.
[54] Hick, Bajzek, et al. 2019.
[55] Hick, Bajzek, et al. 2019.

3.2.4 Multidimensional system cube

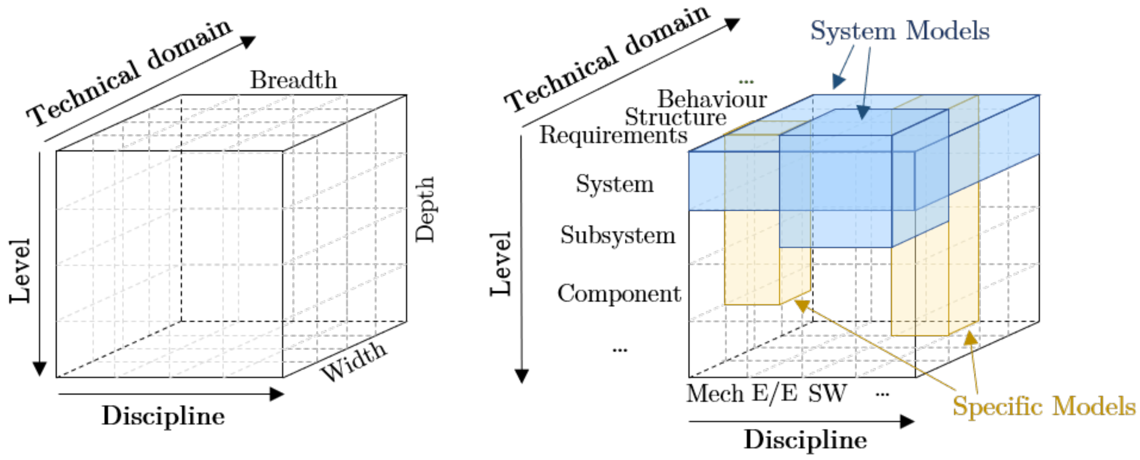
In order to be able to classify models into system models and specific models, it is necessary to define a generalized structure which is adaptable for different applications. As stated in section 3.2.3, different model types can be distinguished regarding the system information (or views) they contain. According to the conclusion of previous sections, three general dimensions are defined for a model-based development of mechatronic systems:[56]

- *Breadth*: There are many disciplines (e.g. mechanics, electrics/electronics, software, etc.) involved in the development of a system, which all have their own views on the system under consideration. These disciplines are represented by this dimension.
- *Width*: This dimension describes involved technical domains (e.g. requirement, structure, behavior, verification & validation, etc.) to consider different aspects of a system.
- *Depth*: Following a top-down approach for the consideration of a system (e.g., system - subsystem - component), the dimensions described before are available in different degrees of detail and on several levels of the system structure. Therefore, a third dimension, describing the depth of the model, is required.

The resulting three-dimensional grid leads to the visualization in figure 3.7 a). The defined dimensions can be visualized as a cube, where each dimension defines one side of the cube (x - breadth, y - width, z - depth). This concept is not limited to three dimensions and an database implementation can be set up following the same principles. In this case, the dimensions have to be tailored to the specific use case. Figure 3.7 b) visualizes the difference between system models and specific models. While a system model is typically placed at the upper half of the cube, specific models can be visualized as pillars. Several meaningful system models can be derived from this cube. For example, a system model covering the technical domains requirements, structure, behaviour as views of multiple disciplines on system level can be modeled using SysML. Another system model contains the views of the disciplines electrics/electronics and software on the aspects structure and behavior on system and subsystem level. This could be a system model showing the architecture and functionalities of an electronics system controlled and driven by software applications. Therefore, several system models with diverse purposes and scopes can exist in parallel. Combined views allow meaningful system statements and further provide a base for common understanding and discussions about systems. Specific models provide technical detail of one view on different levels in contrast to system models. For example, a mechanical engineer

---

[56] Hick, Bajzek, et al. 2019.



a)

b)

Figure 3.7: a) General illustration of classification scheme in form of multidimensional system cube
b) Exemplary models (system models and specific models) positioned within the system cube[58]

defines the design of the components (e.g., piston) as well as the assembly of subsystems (e.g., engine) and the complete system (e.g., vehicle) using CAD software tools. To understand the functionalities of the system, interdisciplinary considerations are required (e.g., mechanical parts, electrics/electronics and software act together to provide a certain function). Both, system models and specific models, are needed to successfully implement model-based development and to support decision making in product development.[57]

It is reasonable to look at similar approaches in data management. Multidimensional data modeling is based on arrangement of information within a grid, which is built up by dimensions. In these concepts, dimensions are further detailed by a hierarchical structure (a so-called classification scheme). These data models are also often visualized as cubes.[59] Figure 3.8 exemplary shows further detailed dimensions in a hierarchical order for the previously defined dimensions. The structure of a dimension is not limited to the presented solution. For example, the discipline mechanics can be detailed in different ways, depending on the criteria. Considering the medium's state of aggregation, solid and fluid mechanics can be distinguished. Based on force influence and movement kinematics, statics and dynamics can be differentiated. It is very important to tailor these definition of the dimensions to the use case or type of

---

[57] Hick, Bajzek, et al. 2019.
[58] Hick, Bajzek, et al. 2019.
[59] Farkisch 2011, p.13 f.

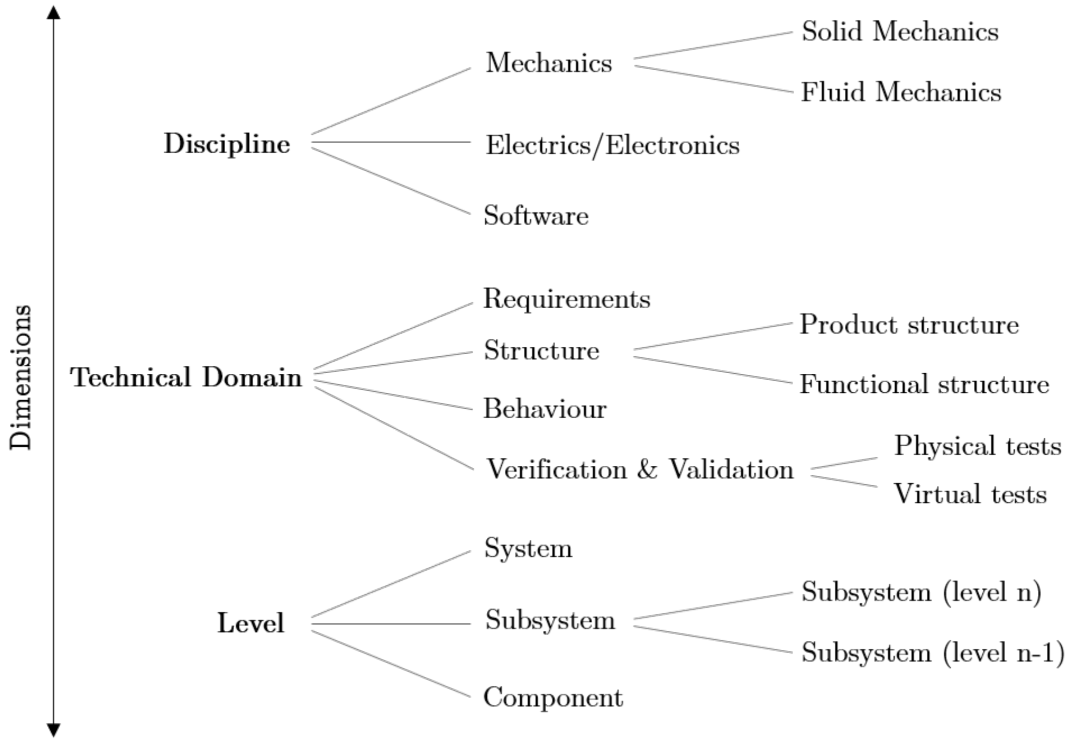


Figure 3.8: Hierarchical structure of dimensions

product. Moreover, the relevant level of detail is depending on the practical application. For example, in vehicle development the levels system-subsystem-component are reasonable while for materials science the levels macro-micro-nano might be suitable. This is further discussed in section 4, where the principal dimensions are tailored to the field of tribology for technical system developments.

3.2.5 Analysis of model structure

The main purpose of the system cube presented in figure 3.7 is to visualize how models can be classified. Nevertheless, this cube can also be used to identify the scope of different models visually. For instance, if a completely novel product development starts and it is unclear which models are needed to describe the system properly, required system information and appropriate models can be detected.

Decisions about purchases of new simulation software packages or test machinery bring high costs with it. Therefore, it is important to find an effective selection of methods and models. Comprehensible concepts and data structure support these kinds of decisions in development. Furthermore, the system cube is used to analyze a current state of available models. Empty spaces can be detected easily, followed by considerations if this missing system information should be described using a model or not. In

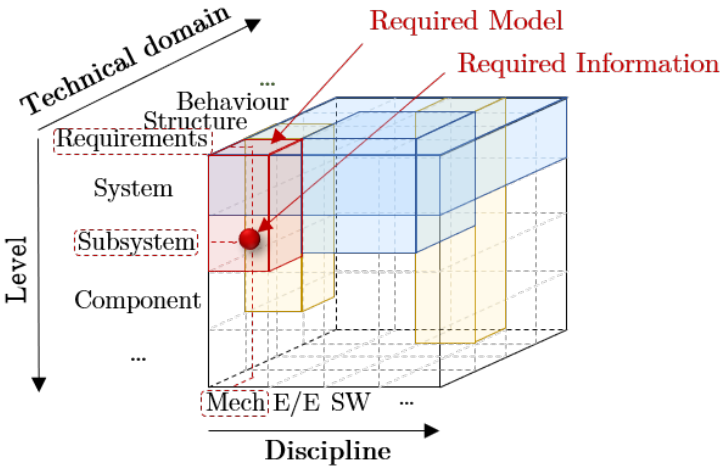


Figure 3.9: Identification of relevant models

figure 3.9, this procedure is principally shown. For example, if the requirements for an automotive powertrain (subsystem) need be to described as base for engineering tasks definition, this view (mechanics - requirements - subsystem) can be identified within the multi-dimensional grid of the system cube. Based on that, a decision has to be made if this information should be described using models or in another way. In this case it would make sense to expand the current structure of available models with a new model that describes requirements on subsystem level as derivation of system requirements. For example, methods from *requirements engineering*[60] could be used to generate this model. This model could be connected and integrated with other models (e.g., to connect a certain requirement with other system aspects like geometry).

To further use the analogy from data management, some established operations for multidimensional data structures are discussed. Figure 3.10 a) shows a *slice* operation, which filters the data in order to look at the system from the view of an mechanical engineer. This operation reduces the dimension of the data grid by focusing on the remaining two dimensions. Figure 3.10 b) visualizes the result of a *dice* operation. In this example, all hardware elements (mechanics and electrics/electronics) are considered on system and subsystem level regarding requirements and structure. With these operations, relevant data can be filtered in order to lay focus on it. Applicable models, which include the relevant views or system information, can be determined easily. *Farkisch* describes more standard operations for the analysis of multi-dimensional data structures like *pivot* (rotate the cube to change the perspective of the analysis) or *drill-down* (refining the view on a certain system aspect step by step).[61]

---

[60] Lamsweerde 2009, p.3.

[61] Farkisch 2011, p.39 ff.

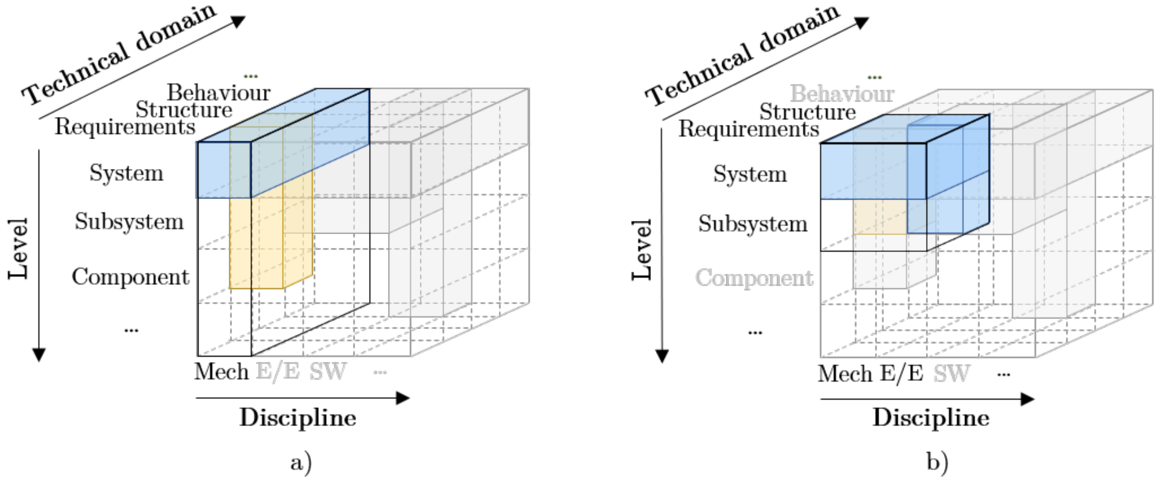


Figure 3.10: a) Slice operation b) Dice operation

3.3 Interactions of models and methods

Models describe systems, or aspects of a system, by combining views and system information. Methods process information or data and either generate, connect or alter models and the system information described by them. In well implemented model-based approaches, most of the relevant system information is described using models. Therefore, methods also act as connection between different models. As stated in section 2.1.5, tools support methods and are often IT-based. Three basic procedural phases or sub-methods can be identified for each development method.

- *Information acquisition*: Input information has to be identified and prepared for a certain method (e.g., it has to be transferred into an appropriate format)
- *Information processing*: This is the core step of each method as it provides the required output information. For example, a calculation uses a mathematical description together with given input information to provide a solution.
- *Information transfer*: Similar to information acquisition, relevant information has to be identified and prepared for further usage.

Figure 3.11 illustrates this sub-procedure within each method and exemplary shows its connections to models. In model-based approaches, most of the available system information is described with models. Therefore, a method is a connection between several models by defining how different system information is interrelated (e.g., fluid flow velocity is an input to calculate hydrodynamic friction). Tools are used to support a method.

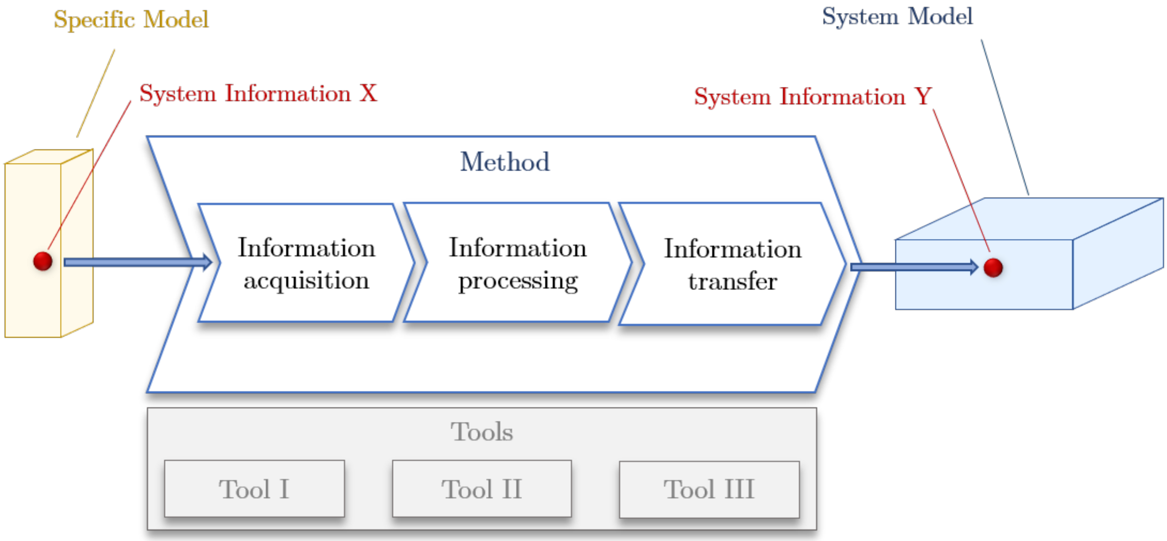


Figure 3.11: Principal structure of a method and input-output relations with system information described by models

As described in the sections before, the development of a growing number of products and rising complexity leads to often incomprehensible structures of models and methods within organizations. Referring to the definition of complexity in section 2.1.2, this implies that there is a high number of entities (models, methods) with not always clearly defined and sometimes non-linear interfaces. Providing a classification scheme for models like the one presented in section 3.2.4, is a first step to face these challenges. In figure 3.4, it was shown how models, simulations and repositories can be connected principally. It is important to understand how they interact in a network consisting of several entities. Figure 3.12 illustrates this network and gives a preview on how such a network might look like in a product development department using numerous models and simulations to develop its products. This should affirm the conclusion, that it is of highest importance to lay emphasize on interfaces between these entities. According to *Hick et al.* three main types of interfaces can be distinguished:[62]

- *Model transfer*: It is reasonable to use the contents of one model in another model (often in a slightly adapted form). For example, data exchange formats provide possibilities for transfer of information. In many cases, a model only needs slight changes to fit for another simulation (e.g., if a detailed geometry model in CAD should be used in a simulation to calculate stresses in the material, the model has to be simplified and adapted, but does not have to be set up from zero). Interfaces between models are required to implement changes not twice but only in one models and the changes are transferred to the other connected models

---

[62] Hick, Bajzek, et al. 2019.

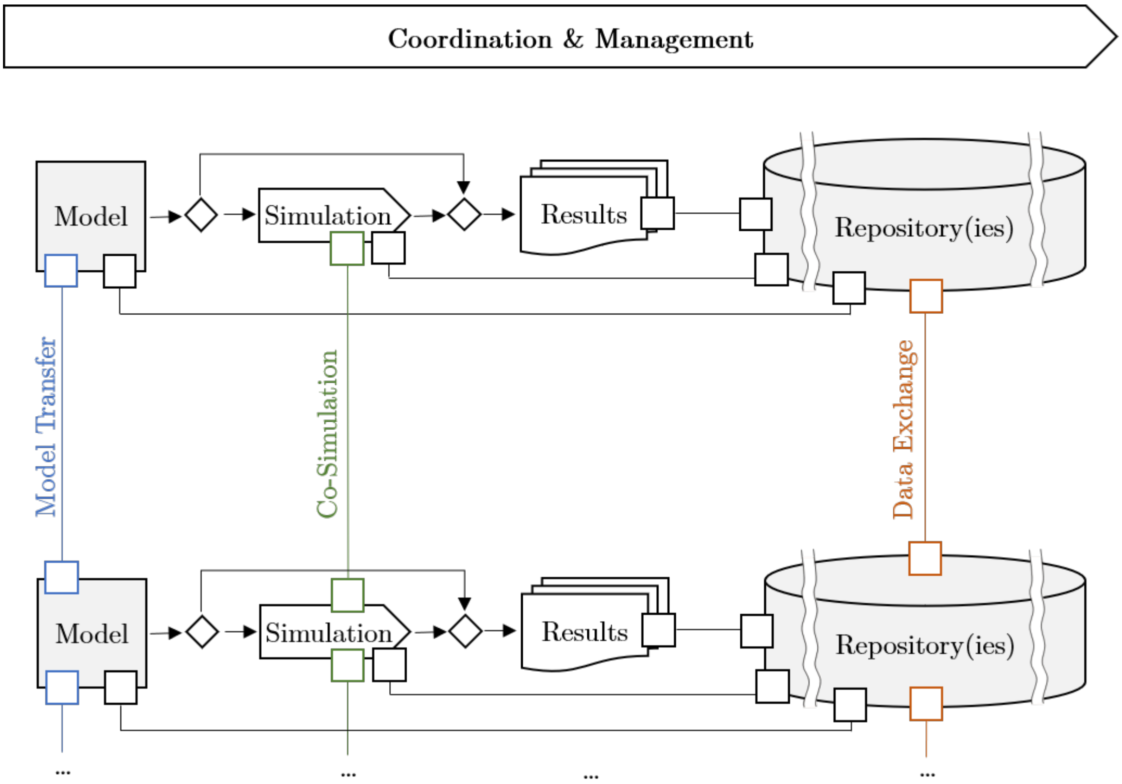


Figure 3.12: Network of models, simulations, repositories and interfaces between them *inspired by Hick et al.*[63]

automatically.

- *Co-simulation*: In order to connect simulations, standardized interfaces are required. To enable cooperation of different simulation and to connect simulations with input-output relations, concepts like *functional mockup interfaces (FMI)* enable the creation of so-called *functional mockup units (FMU)*.[64]
- *Data exchange*: The connection of different repositories is certainly the most important interface type to provide data consistency throughout a development project. It is a task of data management to enable these connections.

Figure 3.12 concentrates on simulation methods and leaves out that many more methods are used (e.g., generate models or process data in models). Another conclusion is that these networks of models, methods and repositories require coordination and management tasks. It was already postulated in this thesis, that a model combining all

---

[63] Hick, Bajzek, et al. 2019.

[64] Höll et al. 2018, p.805.

other models in development to cover all system information is far from realization. A connected repository, which provides the required consistency is not available, however several approaches like *product lifecycle management (PLM)* aim to provide a connected data backbone for the whole product lifecycle.[65]

3.4 Classification of development methods

After describing the principal sub-procedure within methods in figure 3.11, it is now the aim to classify methods. Methods are distinguished regarding their interactions with models, regardless if these models are system models or specific models. As shown in figure 3.13, three principal method types can be detected, regarding their interaction with models:

- Method type A: Methods to generate information about a system, to *fill* a model with it. Therefore, these methods generate models or parts of it.
- Method type B: Methods to use information contained in a model to generate new information (e.g., for another model).
- Method type C: Methods to improve or modify already existing information in a model (e.g., updating existing information or to combine different information to create a new conclusion).

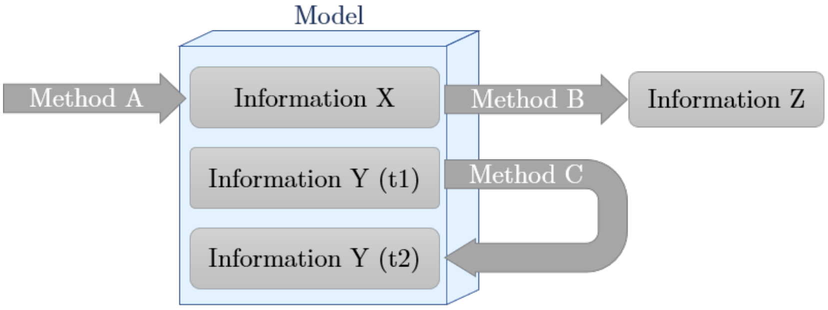


Figure 3.13: General types of interaction between methods and models and derived model types

Figure 3.14 illustrates the different types of interactions between methods and models. These types of interaction act as classification criteria. Figure 3.14 a) shows the most important method type which is used to generate models based on input information. In the further course of this thesis, the emphasize lays on this kind of methods. Methods as shown in 3.14 b) and c) are also used in many occasions, but as a kind of background process, which is to transfer information between models and to update or expand existing models.

---

[65] Zafirov 2014, p.86 f.

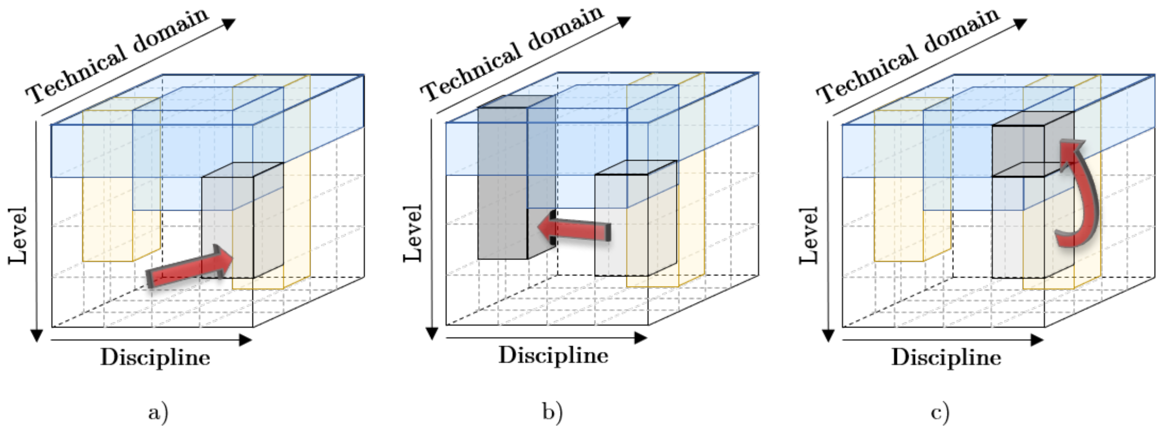


Figure 3.14: Principal illustration of different method types
a) Method to generate models (Method A)
b) Method to transfer information from one model to another one (Method B)
c) Method to process information within one model (Method C)

3.5 Summary of core concepts

Previous sections of this chapter described the described step-wise derivation of classification schemes (for models and methods) and tailored definitions, based on the determined requirements for new concepts.
Considering the scope of a model, *system models* and *specific models* are distinguished. *System models* incorporate multiple views in breadth and width and include the views of at least two technical domains or disciplines. On the contrary, *specific models* provide more depth than a system model by detailing the view of a single discipline on a single technical domain.

To classify several models in development, a scheme was introduced which is based on multidimensional data structures of system information. Therefore, dimensions have to be defined for the specific application (discipline - technical domain - level). This classification scheme can be visualized as multi-dimensional *system cube*.
The interactions of methods and models were investigated in terms of their relations to system information. While methods generate or modify information about a system, models describe systems containing a certain set of information, depending on the scope of the model. Based on these connections, methods for a certain development task can be selected depending on the required system information to generate specific models and system models.
For the context of this thesis, methods are classified in *methods to generate models*, *methods to transfer information from one model to another one* and *methods to process information within one model.*

Considering different model types and their interaction with methods, an approach was presented to support decision making about the right choice of development methods used to generate models. In this procedure, the relevant information to evaluate the development task is identified as a first step. Based on that, a derivation of required models and methods can be done. Basic interactions of models and methods are important to implement a combined use of them. Therefore, classification and clarified terminology are essential.

4 Practical Approach

After describing and developing relevant concepts in section 3, the next step in the procedure illustrated in figure 3.1 is the application of the developed concepts to a defined use case. As described in the introduction, the context of this thesis is the development of socio-technical systems. The presented methodology in chapter 3 has the objective to support the development of complex systems. To prove the advantages of this methodology, a sophisticated use case for system development has to be chosen. Tribological system developments proved to be challenging, as numerous different disciplines are involved and tribological effects occur on several scales with often complicated cause-effect relationships.[1] Additionally, system behavior in tribology, in contrast to other engineering disciplines, depends on system properties rather then only on constant material properties.[2] The following sections start with a description of the chosen technical system, its application and integration within a system. Next, available development methods for tribology are explained with a limited focus on those which support a model-based development approach. Especially important and established development methods in tribology are described more closely in the followings. Based on these selected methods, the interactions of methods and models are analyzed. Models describe different aspects of the system, by containing information about the system under consideration. Emphasize is laid on methods to generate those models. The identified models are in the next step further detailed and classified with the described concept of the *system cube*. This classification and the analysis of appropriate models and methods are the starting point for derivation of a selection of development methods based on customer needs and required models. This is done by identifying required output information and subsequent decisions on how this information can be provided. The procedural steps of this thesis were already illustrated in figure 3.1. In this section, the step *use case definition* and application of the methodology on this defined *tribological use case* are described in detail.

4.1 Use case: Piston-bore interface

Typical objectives of tribological developments are for instance improved system durability and higher available performance, which can both be improved through reduced

---

[1] Czichos and Habig 2010, p.12.

[2] Scherge et al. 2002, p.202.

wear and friction.[3] In the automotive context, the development objectives are often related to durability, reliability, performance, driveability, NVH, etc. Friction and wear reduction play an essential role to achieve that. Tribology is an own field of science, which investigates effects occurring when two surfaces are in direct contact or only separated by a lubricant and in relative motion.[4] It includes interrelated processes of friction, wear and lubrication. While a scientific investigation aims to understand the fundamental processes which define observable tribological behaviour, technical product developments have objectives regarding durability/reliability, function deployment, costs, safety, etc. Therefore, it has to be decided how deep a meaningful investigation of effects has to go in order to improve overall performance of the system, and often a compromise has to be accepted. Extensive research and high development efforts are invested in the improvement of the friction of automotive powertrains. Friction reduction in powertrain subsystems such as internal combustion engines is necessary, as friction losses lead to worse efficiency and increased fuel consumption.[5] Friction analyses of combustion engines show, that up to a third of all internal friction is caused by the piston group. The highest proportion of that is caused by the tribological contact of the piston and rings to the cylinder wall or liner.[6] Of course, these values are influenced by the engine's state (e.g., start, heat up, normal operation, start-stop) and many other factors (e.g., combustion process, operating point of the engine, lubrication, oil temperature) but can be seen as reference values. Furthermore friction of the piston group does not only depend on the operating point of the engine but also on constructive parameters like piston geometry, surface topography, tolerances between the contact partners and ring pre-load.[7] Furthermore, worse efficiency and therefore higher fuel consumption for the same output power leads to higher emissions of CO_2 and can also cause higher pollutants emissions.[8]

Figure 4.1 illustrates the considered subsystem in this use case. On the left side of the picture, an internal combustion engine is drawn schematically, with a detail showing the tribological subsystem piston ring - cylinder wall. This subsystem is also called *piston-bore interface* and is the use case to demonstrate the concepts developed in this thesis. Much research and engineering is invested in the geometrical design of the pistons rings, the surface topography and coatings of the cylinder wall. For this example, the exact geometrical implementation is not taken into account. Instead, overall relations between included parts and functionalities are investigated.

In the development of a vehicle it is always a very important objective to improve powertrain efficiency. Therefore, friction improvement is a derived objective as part of overall efficiency improvement. In order to reduce occurring friction within parts of a

---

[3] Czichos and Habig 2010, p.4 ff.

[4] Jost 1966.

[5] Maassen 2015, p.439.

[6] Speckens et al. 1998.

[7] Maassen 2015, p.447.

[8] Pucher 2015, p.825 ff.

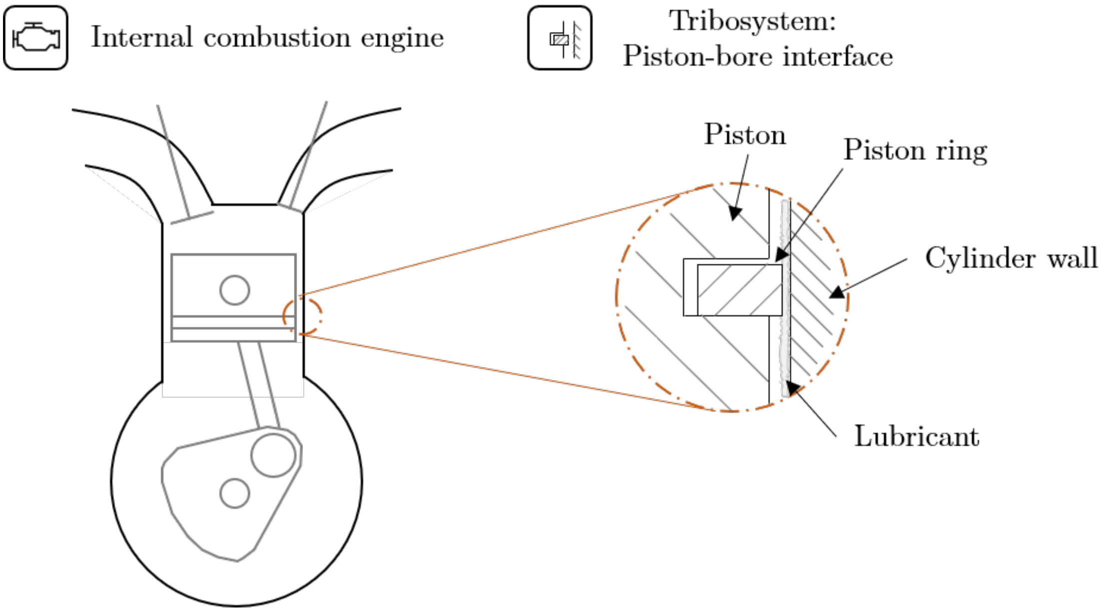


Figure 4.1: Schematic structure of internal combustion engine and the investigated subsystem (contact between piston ring and cylinder wall/liner)

vehicle, all subsystems and its contacts have to be analyzed regarding friction. The internal combustion engine might be the most important element regarding friction investigation. Other parts which contribute to overall losses in a vehicle are, among others, transmission, bearings. As stated before, the piston group and especially the contact between a piston ring and cylinder wall (often a liner is placed within the cylinder bore) is the system-of-interest, when the friction of the internal combustion engine needs to be analyzed. Other friction causing components inside the engine are the valve train, bearings, etc. Figure 4.2 illustrates that the friction map of the engine (as description of the combined friction behavior of the internal combustion engine depending on the operating point) has to be used in combination with information about friction of other subsystems and not just as isolated single information. For example, the operating point of the engine is affected by the chosen gear, flow and roll resistance of the vehicle and therefore the friction within the engine is interrelated with other subsystems and components of the vehicle.

4.2 Models for system description

Models have the objective to describe systems by providing different views on system aspects. The piston-bore interface, as described in section 4.1, has to be investigated regarding its tribological behavior. Therefore, relevant system aspects have to be described by several models to exploit the advantages they provide (according to section

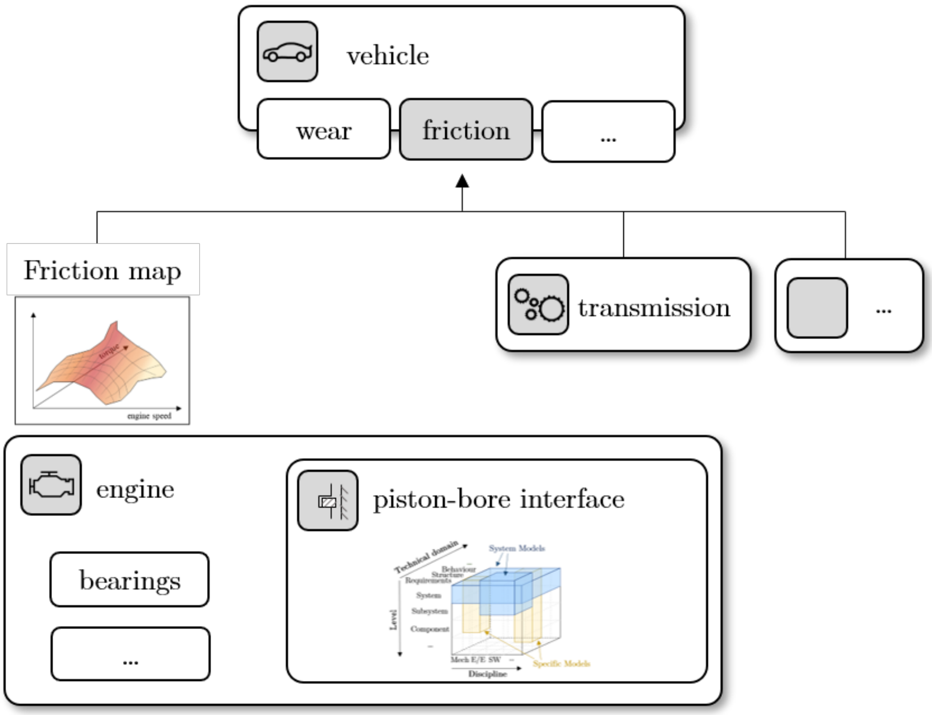


Figure 4.2: Structure of placement of the investigated subsystem within the development of the whole vehicle regarding friction[9]

2.2.1). Figure 4.3 illustrates the considered tribological system. It consists of two solid parts - the piston ring on the one and the cylinder wall (or liner) on the other side. The solid parts of this system are described by a geometry model and materials model and the change of geometry of the lubrication gap is described by a deformation model. The two surfaces are separated by an oil as lubricant - described by an oil model and a hydrodynamic model. However, this separation is affected by normal loads on the contact and motions of the two surfaces relatively to each other - described by multi-body dynamics model and a load model. The contact and the occurring wear and friction are defined by applicable models. The models used in the development of the contact between the piston ring and the cylinder wall (or liner) are principally described in table 4.1.

---

[9] Hick, Faustmann, et al. 2019.

Table 4.1: Description of models used for tribological development

Model	**Described system aspects**
Mixed Lubrication Friction Model	Description of friction between two surfaces in relative motion with and without lubrication (dry friction and hydrodynamic friction)
Wear Model	Wear volume of contact for different wear modes described as function of operating point
Contact Model	Contact pressure formation in both surfaces and determination of contact zones
Hydrodynamics Model	Description of flow behavior of the lubrication medium (usually oil), pressure distribution in the lubrication gap (incl. modeling of surface topography influence on micro flow) and temperature distribution within the fluid film
Oil Model	Description of the lubrication fluid properties (e.g., viscosity of the lubricant as function of temperature, shear stress and pressure)
Deformation Model	Material behavior in form of mechanical and thermal deformations and related change of gap geometry surfaces (often expressed as condensed stiffness matrices)
Multi-body Dynamics Model	Description of kinematic multi-body systems regarding motion and occurring reaction forces
Geometry Model	Geometrical information about the system structure, subsystems, components and surface topography
Load Model	Description of load parameters on the system under development according to defined test cycles including mechanical and thermal load
Materials Model	Properties of used materials for solid parts of the tribo-system regarding elasticity, stiffness and surface hardness
Elastohydrodynamics Model	Tribological system consideration by combined aspects described in other models

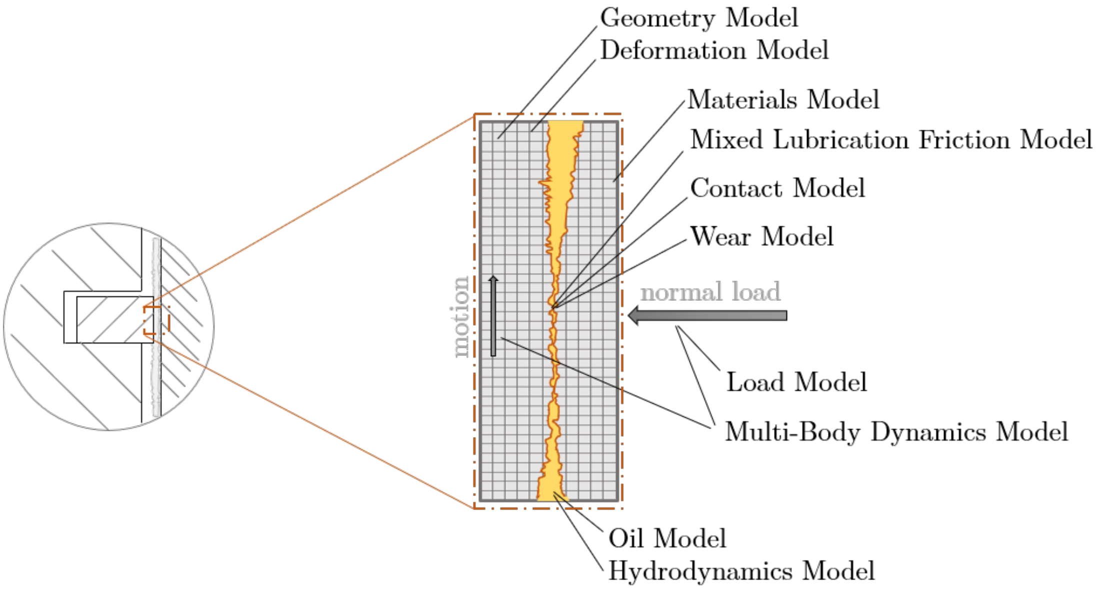


Figure 4.3: Principal illustration of models used to describe tribological contacts

4.3 Methods for tribological development

Generally, *test methods* (test procedures with an existing physical model or the produced product) and *simulation methods* (approximate imitation of the operation of the system under consideration[10]) are distinguished. In literature, development methods in tribology are divided into six categories, listed in table 4.2. While a higher category implies a higher degree of abstraction of the load situation and/or system description, a test of a lower category requires a test specimen, whose properties are near to the final produced product, or the final specification of the system itself.[11,12] For vehicle development, a test of category I would be a vehicle test under real driving conditions (e.g., including oil aging effects or intake air containing abrasive particles) to estimate wear. Analyses on an engine testbed to estimate the friction by the corresponding value of *friction mean effective pressure (FMEP)*[13] belong to category II or III (depending on the point of view if the engine is the complete system or a subsystem). Numerous methods evolved over time to estimate the friction losses of the engine indirectly on a testbed. For example, by measuring the cylinder pressure (to calculate mean indicated pressure) and the effective torque (to calculate mean effective pressure) of the engine, the difference corresponds to the FMEP. This method is well established to estimate friction losses within an engine but an engine as test specimen and elaborate

---

[10] Banks et al. 2001, p.3.
[11] GfT 2002, p.41 ff.
[12] Czichos and Habig 2010, p.193 f.
[13] Maassen 2015, p.439.

Table 4.2: Categories of tests for tribological systems[18]

Category	Description
I - Field test	Tests and investigations of complete tribological system under real world conditions
II - Bench test	Tests and investigations of complete tribological system on a test bench under conditions near to reality
III - Aggregate test	Tests and investigations of single original subsystems under conditions near to reality
IV - Component test	Tests and investigations of components (original or simplified) under conditions near to reality
V - Specimen test	Tests and investigations of component-type specimens with simplified load conditions
VI - Model test	Fundamental investigations of tribological processes with specific test specimens under variable but defined conditions

measurement tools are required.[14] To evaluate the friction of specific subsystems or components, extensive investigation like the strip-down method are required.[15] For direct friction measurement, the *floating liner method* was developed, where the friction force acting on the cylinder liner is measured.[16] Simulation methods can be assigned to the categories IV to VI, as they are always based on models which are simplified to a certain degree. Most physical or mathematical simulation models are based on laws of similarity and continuity, which cannot be transferred to tribological systems in any cases. Therefore, simulation in tribology is especially challenging.[17] Nevertheless, simulation methods provide possibilities to develop friction causing subsystems in early development phases, as no physical test specimen is required.

---

[14] Pischinger et al. 2009, p.359 f.

[15] Maassen 2015, p.440.

[16] Merkle et al. 2018, p.8.

[17] Czichos and Habig 2010, p.236 f.

[18] Czichos and Habig 2010, p.193.

4.4 Interaction analysis of models and methods

In a model-based development approach, methods are classified in *methods to generate models*, *methods to transfer information from one model to another one* and *methods to process information within one model* according to section 3.4. To enable virtual development, the generation of models, especially formal and semi-formal models, is essential. These models are used to describe the tribological system under development. Therefore, the process of model generation via methods and the related information flow is further analyzed. The focus of this thesis is on descriptive models, which means that these models are generated to describe a certain system. Of course, simulation methods are based on so-called *simulation models*, which are the base for the execution of a calculation (e.g., a geometry mesh as base for a *Finite Element Method*). These models strongly depend on the tool they are built with and are not the focus of this thesis.

4.4.1 Methods for model generation

In section 3.3, it is stated that models and methods are linked via system information. For example, the system information contained in one model is used as input for a calculation method, which generates another model. Therefore, there are input-output-relations between models and methods. For simulations in tribology, several specific calculation methods are well established and are further described in the followings.

As friction between two surfaces is influenced by many factors, the tribological behavior of a system is often illustrated as a *Stribeck curve*, where the influence of the specific film thickness on the *coefficient of friction (COF)* is shown. This COF was already mentioned in equation 2.1, and is described as proportionality factor between normal load and friction force. Therefore, the coefficient of friction can be used as a specific parameter representing friction. The absolute value of the friction force is calculated by multiplying the COF with the normal load an the contact. The principle relation of friction and specific film thickness dates back to *Stribeck*[19] , who investigated the influence of lubrication on bearings. To estimate the friction of a system depending on the operating point (load, relative speed, motion etc.) information about lubrication flow, the lubrication medium and the surface properties are required. Figure 4.4 illustrates the required input information to generate the mixed lubrication friction model, which contains information about the dry friction and the hydrodynamic friction of a system as output information provided by a method. It is also exemplary shown in this figure, that several models might be included. Models are classified into *system models* (marked blue) and *specific models* (marked orange) using the concept of the system cube according to section 3.2.4. In this case the models describe different system aspects and contain output and input information used by methods. However, it

---

[19] Stribeck 1902.

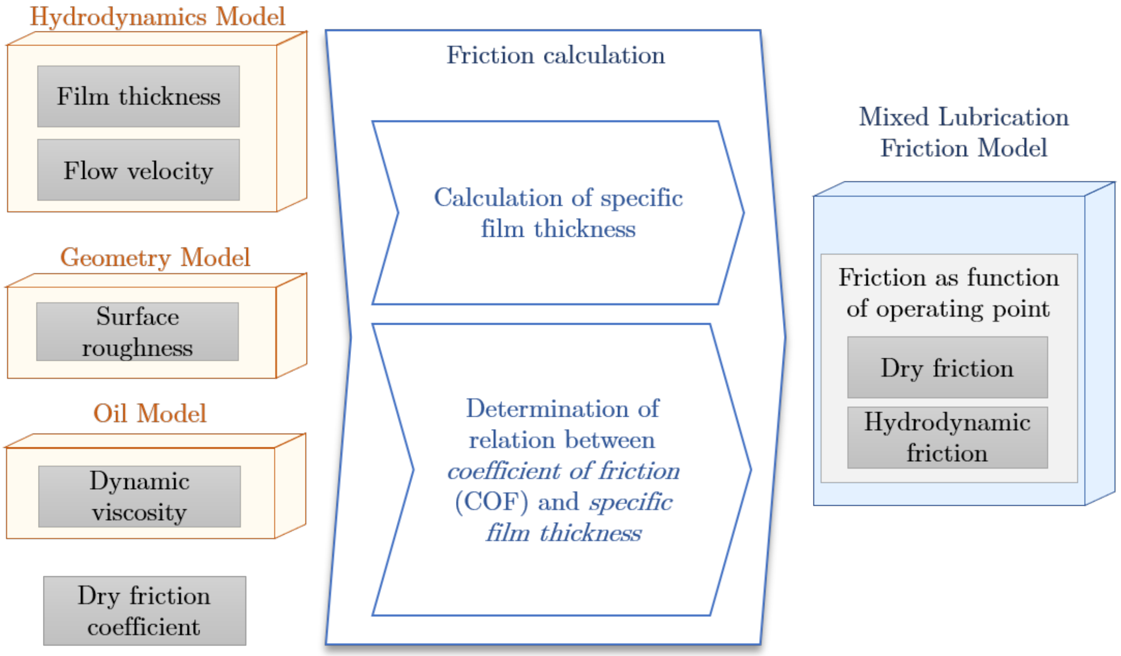


Figure 4.4: Illustration of a *friciton calculation method* to generate a *mixed lubrication friction model*

depends on the specific circumstances of a development project which models are available (e.g., not in all cases an detailed oil model is available or desired). To calculate absolute friction between the contact partners, information about friction under dry condition (without lubrication) is required. For example, this can be determined using a *tribometer test*[20] , which uses a simplified system structure as well as defined load and motion. Hydrodynamic friction for *newtonian fluids* depends on the fluid viscosity and the flow velocity gradient.[21] Therefore, information from a *hydrodynamic model* and an *oil model* (viscosity is affected by fluid pressure, temperature and shear stress and other influencing factors) is required. To calculate the specific film thickness, different approaches are established depending on the technical system or application. All of them have in common, that it relates to a film thickness (described by a hydrodynamics model) and the surface roughness.[22]
For instance, the specific film thickness is generally approximated as function of the film thickness and the surface roughness of the involved parts. For bearings, it is established to estimate it as function of the bearing load and angular velocity.[23]

Figure 4.4 shows the interrelations between models and methods, as methods require

---

[20] Czichos and Habig 2010, p.204.
[21] Wen and Huang 2018, p.201.
[22] Czichos and Habig 2010, p.238 f.
[23] Wen and Huang 2018, p.148.

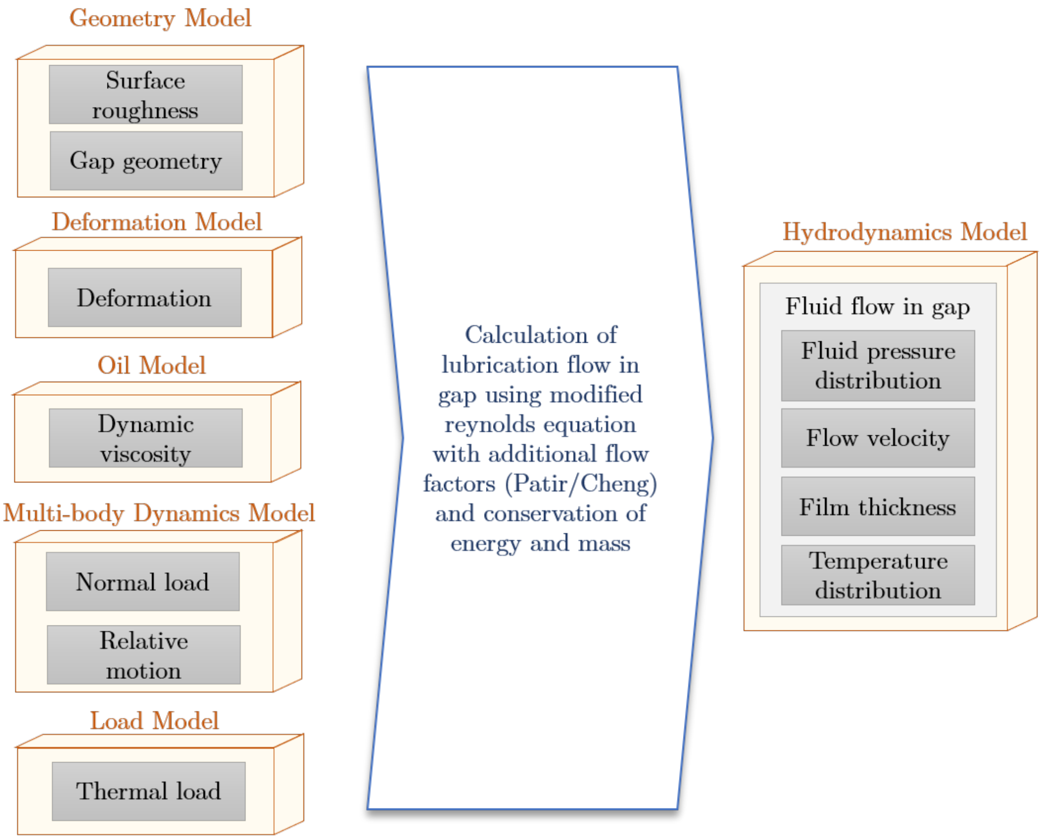


Figure 4.5: Fluid flow calculation of lubricant in gap with assigned input and output information and generated *hydrodynamics model* as result of the calculation

input information to generate certain output information. In most cases, this information is described in models. Figure 4.4 shows, that important input information for the generation of a *friction calculation method* is contained in a *hydrodynamics model*. Therefore, also the method to generate this model is further investigated and illustrated in figure 4.5.

The calculation of the fluid flow is based on *Reynolds' hydrodynamic lubrication theory*[24], which is a differential equation derived from *Navier-Stokes equations*[25] with simplifications for fluid flow in a lubrication gap.[26] This equation was adapted by *Patir and Cheng*, who introduced so-called *flow factors* to consider the influence of surface topography on the micro flow of the lubricant.[27] This is especially important for tribologial contacts, as it has great influence on friction if the contact zone is separated by lubricant in a micro scale or not.

---

[24] Reynolds 1886.
[25] Stokes 1845.
[26] Czichos and Habig 2010, p.182 f.
[27] Patir and Cheng 1978.

As input information, the fluid viscosity is required together with boundary conditions. These are represented by the gap geometry and deformations of the surfaces as well as loads (mechanical and thermal) and relative motion of the surfaces. In this case, the equation system with its adaptions represent a model of the system, which is the base for the method of fluid flow calculation. In tribology, friction and wear are the most important parameters of behavior. But also hydrodynamics have great influence on friction and wear as it describes the behavior of the lubricant within the gap. The methods and models illustrated in figure 4.4 and figure 4.5 were chosen to exemplary show the interrelations of different methods and models. For a meaningful system description regarding friction, the mixed lubrication friction model can only be generated, when a hydrodynamics models provides input information for the calculation. Nevertheless, the input information for a friction calculation can also be based on assumptions or literature values, but the accuracy of such an approach is not meaningful.

Further methods are required to fully describe the tribological system. A well established method to calculate wear is an approach according to *Archard*,[28] who describes an empirical relation of wear, surface hardness, normal load and sliding distance with a proportionality factor called specific wear rate. It was originally used to describe adhesive wear only and was updated to include effects of abrasion[29] and the influence of abrasive particles on the specific wear rate as well. Accordingly, to calculate the specific wear rate, information about the surface roughness (described in a geometry model), film thickness (described in a hydrodynamics model) and abrasive particles are required.[30]

Considering the contact situation, the geometry of the asperities has great impact on the resulting contact zone and pressure. *Greenwood and Tripp* established a way to consider surface roughness by idealizing the real asperities with normally distributed equally sized asperities.[31] Furthermore, material properties (described in a materials models), loads (described in a multi-body dynamics model) and information about the contact geometry (described in a geometry model and a deformation model) are required as input information for the simulation of the contact.

Additionally, models and required methods to generate them are implemented to describe deformations, multi-body dynamics, oil characteristics and more. These additional methods to generate relevant models are contained in the appendix.

4.4.2 Information flow between methods and models

As described in the previous section, there are various methods and models required to describe all relevant aspects of a tribological system. Figure 4.6 illustrates the infor-

---

[28] Archard 1980.

[29] Rabinowicz 1965.

[30] Sommer et al. 2014, p.16 ff.

[31] Greenwood and Tripp 1971.

mation flow to generate an elastohydrodynamics (EHD) model. *Elastohydrodynamic (EHD)* is a lubrication state according to table 2.1 and includes several effects (hydrodynamics, elastic deformation of gap geometry and contact of asperities). This lubrication state occurs in many technical systems (gear, bearings, etc.).[32] In elastohydrodynamic contact situations, many influencing factors affect each other. These factors are represented by several models and methods describing different aspects of tribological systems.

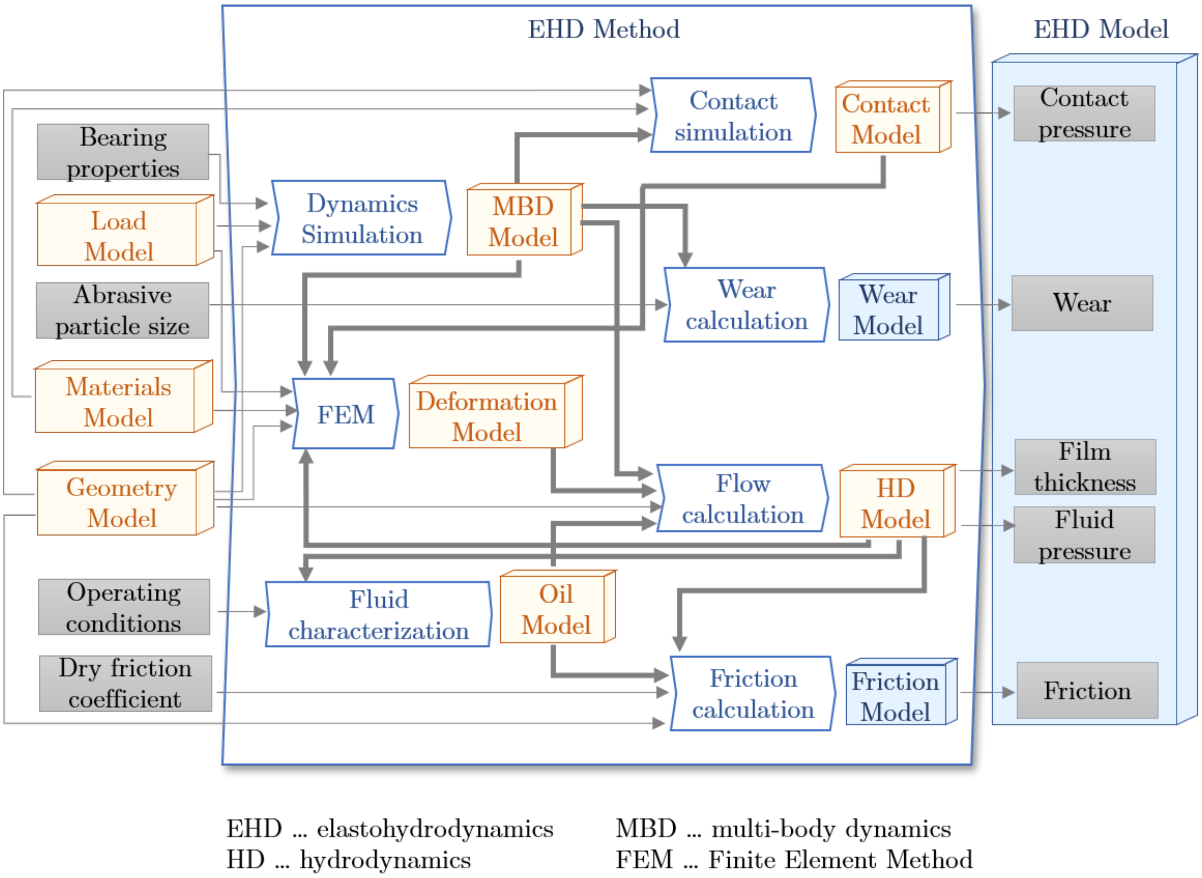


Figure 4.6: *Elastohydrodynamics (EHD)* model generated by a method which connects and orchestrates the interplay of several methods and models

Elastohydrodynamic approaches are used to described different tribological systems like journal and roller bearing, gears and others. A sophisticated model structure is required to gather combined system considerations. Various pieces of input information are required to generate the models, which are connected by the elastohydrodynamic

---

[32] Wen and Huang 2018, p.4.

method. Obviously, this figure illustrates how complex the interrelations between the different system aspects are. In the end, it is the objective to generate an elastohydrodynamics (EHD) model, which contains information about the tribological contact. Although the most relevant information related to friction and wear, other information about the contact (contact pressure, film thickness, fluid pressure) are of interest and described together in the generated elastohydrodynamics (EHD) model.[33]

Figure 4.6 shows models, methods and connection between them. This is illustrated in a simplified way, but it proofs how sophisticated modeling of tribological contact situations can be. Detailed descriptions of the interfaces (illustrated as arrows) and of the shared or transferred system information were exemplary shown for two methods in section 4.4.1. All other methods related to an EHD approach are illustrated together with its input-output relations to models in appendix figure A.1, A.2, A.3, A.4 and A.5. The source of input information my vary from project to project (e.g., information about certain system aspects can be re-used from previous projects). Figure 4.6 illustrates an ideal model-based approach, which means that it has the objective to describe as many aspects of a system with models as possible. The presented network of methods and models is a sophisticated approach, which has to be adapted according to requirements of the development. An approach to choose the right development path within this network was presented in section 3.1.

4.5 Classification of models in tribology

The interrelations between models and methods were already investigated in the section before. Furthermore, the identified models have to be classified, because as illustrated in figure 4.2 the model structure for this tribosystem is only one of many in vehicle development. Following this presented approach, it is possible to develop a *model landscape* for a whole development.

4.5.1 Clustering of tribological system information

In order to be able to classify the models used to describe this tribological system, the dimensions which build up the *system cube* have to be adapted to the use case. Therefore, the general dimensions are further detailed for tribology, by asking the following questions:

- *What disciplines have to be involved to describe the tribological system from their point of view?*
- *Which system aspects are of interest to successfully develop a system?*
- *On which levels of detail are the views on the system described?*

---

[33] Czichos and Habig 2010, p.186 f.

The general dimensions (breadth, width and depth) of the multi-dimensional data structure are detailed as follows:

- *Discipline (breadth)*: In tribology, many disciplines of science and technology are involved to generate broad system understanding.[34] For the development of the piston-bore interface, the disciplines *mechanics*, divided into *solid mechanics* (describing contact mechanics, kinematics etc.) and *fluid dynamics* (describing the lubrication fluid flow and pressure distribution in the lubrication gap, etc.), *materials science* (describing the properties of the involved solid parts regarding elasticity, plasticity, microstructure, crack formation, dislocations, etc.) and *rheology* (describing the lubrication fluid properties, especially viscosity) are relevant disciplines.
- *Technical domain (width)*: For this dimension, the aspects of interest of the considered system are included. Selected aspects are *structure* (describing geometry, surface topography, etc.) and *behaviour* of the tribological system. For this use case, system behavior is divided into *structural behaviour* (describing elastic deformations of solid parts, resulting load distributions, viscosity of fluids, etc.) and *tribological behaviour* (describing wear and friction effects).
- *Level (depth)*: As the focus of the models is to describe the piston-bore interface, this dimension is structured top-down from *macro*, over *micro* to *nano*. Tribological effects occur on all of these scales.

Regarding the dimension *level*, further explanations are required. Figure 4.7 applies the principal idea of a cascaded v-model[35] to a tribological system. Like in the development of a product on different levels (e.g., system - subsystem - component), a tribological system can be developed or improved in a top-down and implemented in a bottom-up approach. Starting with the principal tribological system structure on macro scale, zooming in narrows the view on microscopic aspects like the influence of surface roughness on the fluid flow and finally on contact of asperities on nano scale. To describe tribological systems, models are needed on different levels, but it depends on the development goals and how deep the understanding of the system should go, which models are meaningful.

4.5.2 Derivation of system cube

The identified models are allocated to the mentioned dimensions as shown in table 4.3. It is important to mention, that in some cases there is no distinct allocation for a model possible, as it depends on the point of view and the objective of the analysis.

---

[34] Czichos and Habig 2010, p.12.

[35] VDI2221 1993.

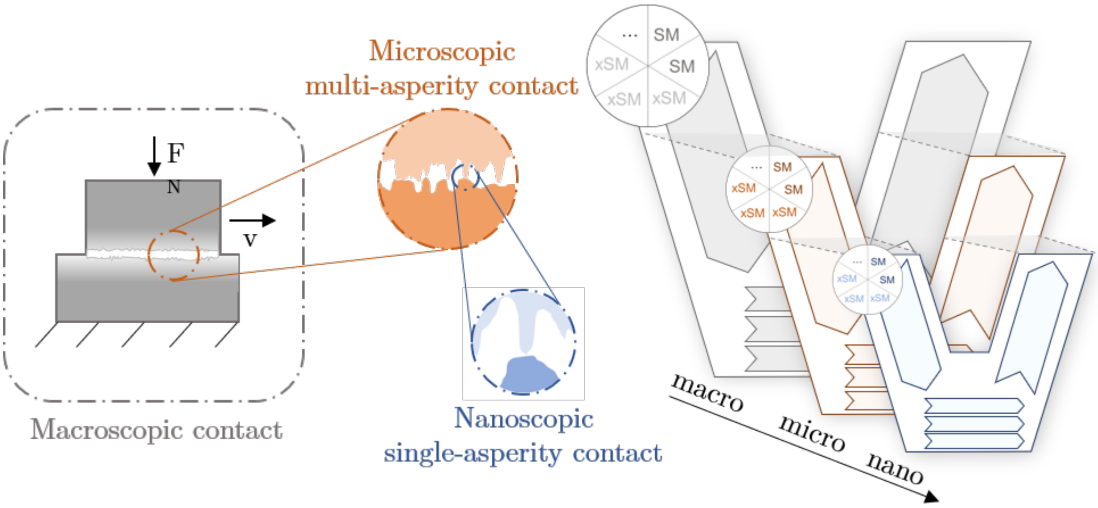


Figure 4.7: Principal illustration of models used to describe tribological contacts *inspired by Hick et al.*[37]

This concept should demonstrate a way how to structure numerous models to enable further analysis and does not claim to be an unique solution. Based on this allocation, the scope of each model can be identified. As stated in section 3.2.3, *system models* focus on breadth and width - including at least two technical domains or disciplines - while *specific models* provide depth by detailing one technical domain as view of one discipline.[36]

By analyzing table 4.3, three system models can be identified. The *elastohydrodynamic model (EHD)* combines and coordinates different models to gain knowledge about the system on macro scale (e.g., information from hydrodynamics model is used to determine the lubrication state and based on that the friction is estimated). An EHD model therefore provides a multi-disciplinary view on the system.

Another system model is the *mixed lubrication friction model*, which has the objective to provide information about the friction in form of dry and hydrodynamic friction. Therefore, the scope of this model is to give an interdisciplinary understanding of the interrelated friction modes and has not the objective to detail one mechanism. The *wear model* needs to include the views of several disciplines to describe different wear modes on different scales. For example, erosion is connected to fluid dynamics and also to materials science, as it depends on the impact energy of the fluid and the material structure, its properties and pre-damage. In general, system models represent sets of multiple views to support the distribution of information between models and to enable broad system understanding.

---

[36] Hick, Bajzek, et al. 2019.

[37] Hick, Faustmann, et al. 2019.

Table 4.3: Description of models used for tribological development

Model	**Discipline**	**Technical Domain**	**Level**
Mixed Lubrication Friction Model	Fluid Mechanics Solid Mechanics	Tribological Behavior	macro
Wear Model	Materials Science Fluid Mechanics Solid Mechanics	Tribological Behavior	macro micro
Contact Model	Solid Mechanics	Structural Behavior	micro nano
Hydrodynamics Model	Fluid Mechanics	Structural Behavior	macro micro
Oil Model	Rheology	Structural Behavior	macro micro
Deformation Model	Materials Science	Structural Behavior	macro
Multi-body Dynamics Model	Solid Mechanics	Structural Behavior	macro
Geometry Model	Solid Mechanics	Structure	macro micro
Elastohydrodynamics Model	Rheology Materials Science Fluid Mechanics Solid Mechanics	Structure Structural Behavior Tribological Behavior	macro micro nano

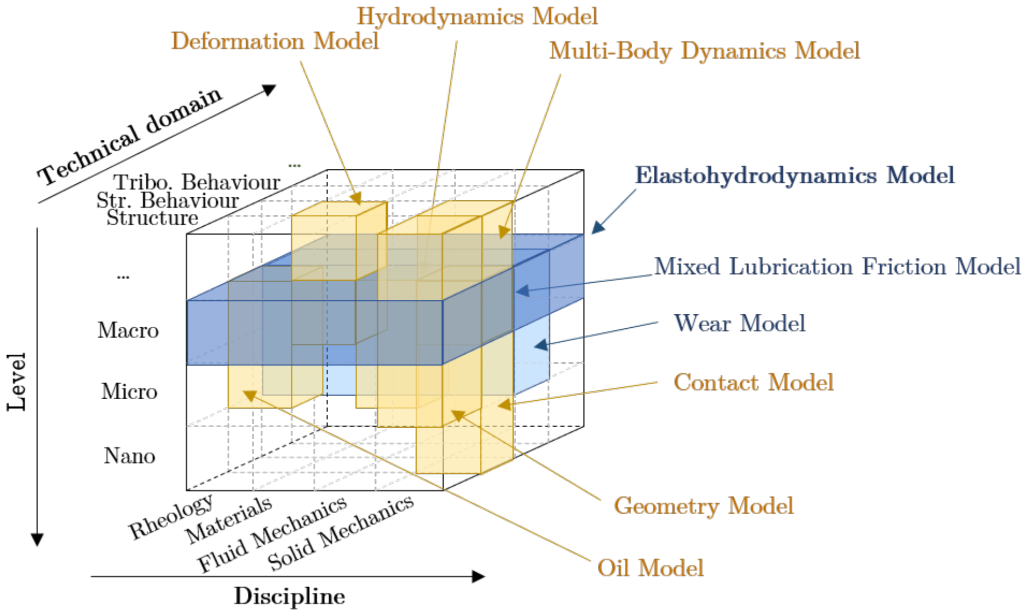


Figure 4.8: Tailored illustration of a system cube for tribological system development[38]

In order to visualize this model classification, a system cube is drawn based on the defined dimensions. Figure 4.8 illustrates the system cube with the models described in table 4.1 according to their allocation to the dimensions in table 4.3. The listed models are placed within the defined grid, where specific models are highlighted yellow while system models are highlighted blue. The boundaries of each models are not fixed, rather they depend on the scope how the model is used in a certain development. Furthermore, models can be extended to include more system information. For instance, a geometry model can be extended by including information about surface roughness and surface treatment to provide more depth.

Based on this classification, several analyses can be conducted. Firstly, it can be analyzed how much of the possible information about the system (represented by the whole cube) is covered by or integrated within models. Next, it has to be decided which areas of the cube are necessary to conclude relevant statements. In this example, the only model, which may contain information on nano scale is the contact model (e.g., including atomic contact forces measured by atomic force microscopy). Furthermore, most models are allocated to solid mechanics, which is a result of the tribological system's application within an internal combustion engine. To gain knowledge about the interactions of different effects within the system (e.g., deformations influence fluid flow, which affects friction and wear) a system model like a elastohydrodynamics model is needed to act as model integrator and connector.

---

[38] Hick, Faustmann, et al. 2019.

4.6 Approach for methods selection

After the analysis of models and methods, which can be used to develop tribological systems, the task to select a set of appropriate methods and models is investigated. Therefore, a case example is defined. For this use case, the starting point is a customer, who invented a new coating for cylinder liners. It has to be investigated if it is applicable for automotive powertrains and if it provides advantages regarding friction. Therefore, the development task is *friction evaluation*. An engineering department, which has only specific experience in tribological development, has to decide which models and methods are required. These decisions imply investments in software, hardware and employees and they influence durability and quality of the developed product as well. Therefore, these decisions are of high importance. This approach provides support to select a set of models and methods and does not provide a final decision. Figure 4.9 illustrates the available models within the system cube at the start of this exemplary development project.

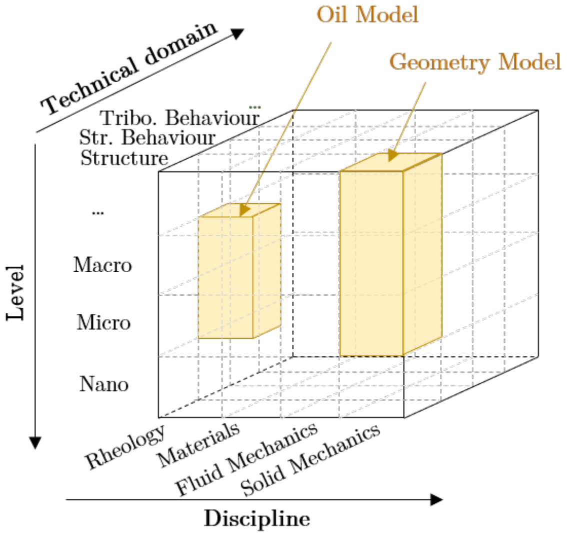


Figure 4.9: Available models at project start

In this case, the engineers have access to a geometry model (CAD model of the piston-bore interface and information about surface topography) and an oil model (oil viscosity as function of shear rate, temperature and pressure). The following procedure is based on the concept already illustrated in figure 3.2. Each step will be investigated and described in detail.

4.6.1 Identification of required output information

As a first step, the needs of the customer - *to evaluate if the new coating improves friction behavior of the piston-bore interface* - are used to derive a specific development task. The customer needs are in this case pre-defined. Nevertheless, in complex projects *requirements engineering*[39] is executed before this approach of method selection is triggered. The development task, based on customer needs, is the input for the matrix in figure 4.10.

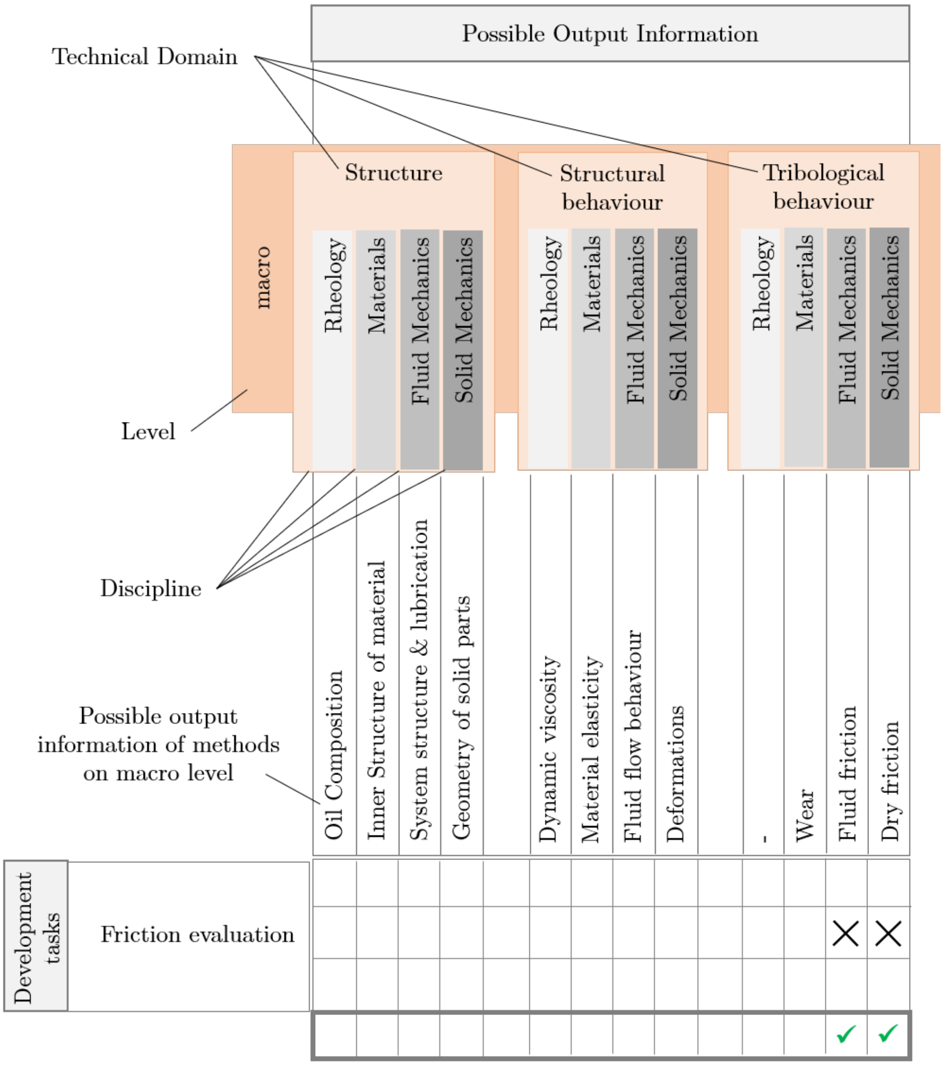


Figure 4.10: Derivation of required output information based on development task

---

[39] Lamsweerde 2009, p.3.

Furthermore, possible output information (in this case output information of virtual development of the tribological contact) is required. This information is derived from the system cube structure based on the dimensions defined in section 4.5.1. To answer how the piston-bore interface behaves regarding friction, information on *macro level* about the different *system aspects* (structure, structural behavior, tribological behavior) as views of different *disciplines* (Rheology, materials science, fluid mechanics, solid mechanics) on these aspects are listed. In this case, tribological behavior on macro level is of interest, which leads to the conclusion, that information about *fluid friction* and *dry friction* is required. This required information can also be identified within the system cube. Figure 4.11 shows the position of these views on the system (highlighted as red points) and the already available models (oil model and geometry model).

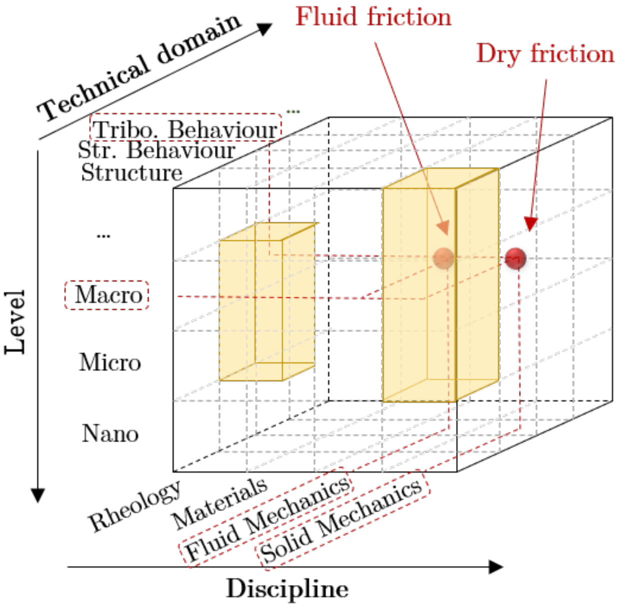


Figure 4.11: Illustration of identified and required system information

4.6.2 Identification of required models

In figure 4.12, the identified information from the first step (dry friction and fluid friction) is used and models which contain this information are identified. The allocation of system information to a certain model - regardless if it is a specific model or a system model - was already done with the concept of the multidimensional system cube. The system cube including possibly relevant models for elastohydrodynamic contact situations was already shown (figure 4.8). With this data model as background concept,

		Models							
		Mixed Lubrication Friction Model	Wear Model	Contact Model	Hydrodynamics Model	Oil Model	Deformation Model	Multi-Body Dynamics Model	Geometry Model
Required output information									
	Dry friction	×							
	Fluid friction	×							
		✓							

Figure 4.12: Derivation of required models from required system information

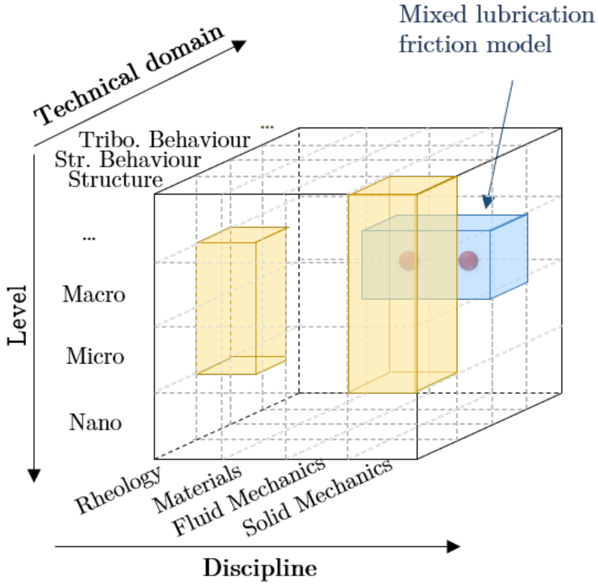


Figure 4.13: Illustration of identified model, which covers the required system information

it is easy to allocate a system information to a certain model. The required information describes tribological behavior on macro level as views of multiple disciplines (fluid and solid mechanics). The result of this step is the identification of a required model. The system cube in figure 4.13 shows the required model structure to cover information of interest for this example. In this case, a *mixed lubrication friction model* is required to describe the system information of interest, additionally to the already existing geometry model and oil model.

4.6.3 Identification of required input information

In the next step, the required input information has to be identified in order to use the information in the selected methods. Based on this, it is decided which input information is already existing and which methods should be implemented. In this case, a method for the generation of a *mixed lubrication friction model* needs *surface roughness* of each part of the tribological contact, *dry friction coefficient* providing information about the friction of the dry contact, *dynamic viscosity and flow velocity* of fluid and *film thickness* in the lubrication gap as input. These pieces of information are identified in figure 4.14 and highlighted in figure 4.15 a) (in this case five identified pieces of system information).

Methods are used to generate the required models. Therefore they use the identified input information to generate the required output information. These methods for model generation are visible as arrows in figure 4.14 and figure 4.15 b) and represent connections between input information and output information. In the previous sections the information flow for these methods to generate the *mixed lubrication friction model* were described in detail in figure 4.4.

As the required input information is identified, it has to be analyzed, if some of it is already existing. In this example, the oil model (description of lubrication fluid) is already available at project start as shown in figure 4.9. Therefore, the dynamic viscosity of the fluid is already described and no further investigations on the oil properties are required. Information about surface roughness is also already available and described as part of the geometry model on micro level. In other cases, classical test procedures (profile method to gain standardized parameters like average roughness value[40]) or optical methods (e.g., interference microscopy[41]) could be used to generate this information. To gain understanding of the dry friction between the two surfaces without lubrication, literature values can be used for coarse estimations. In more sophisticated approaches, tribometer tests can be done (for these tests, the system is simplified to replicate the motion and contact type).[42]

---

[40] ISO4288 1996.

[41] Czichos and Habig 2010, p.236 f.

[42] Czichos and Habig 2010, p.227 ff.

In this case, a *hydrodynamics model* is essential to calculate the input information film thickness and fluid flow velocity. To generate the required output information a method to generate the *mixed lubrication friction model* is required.

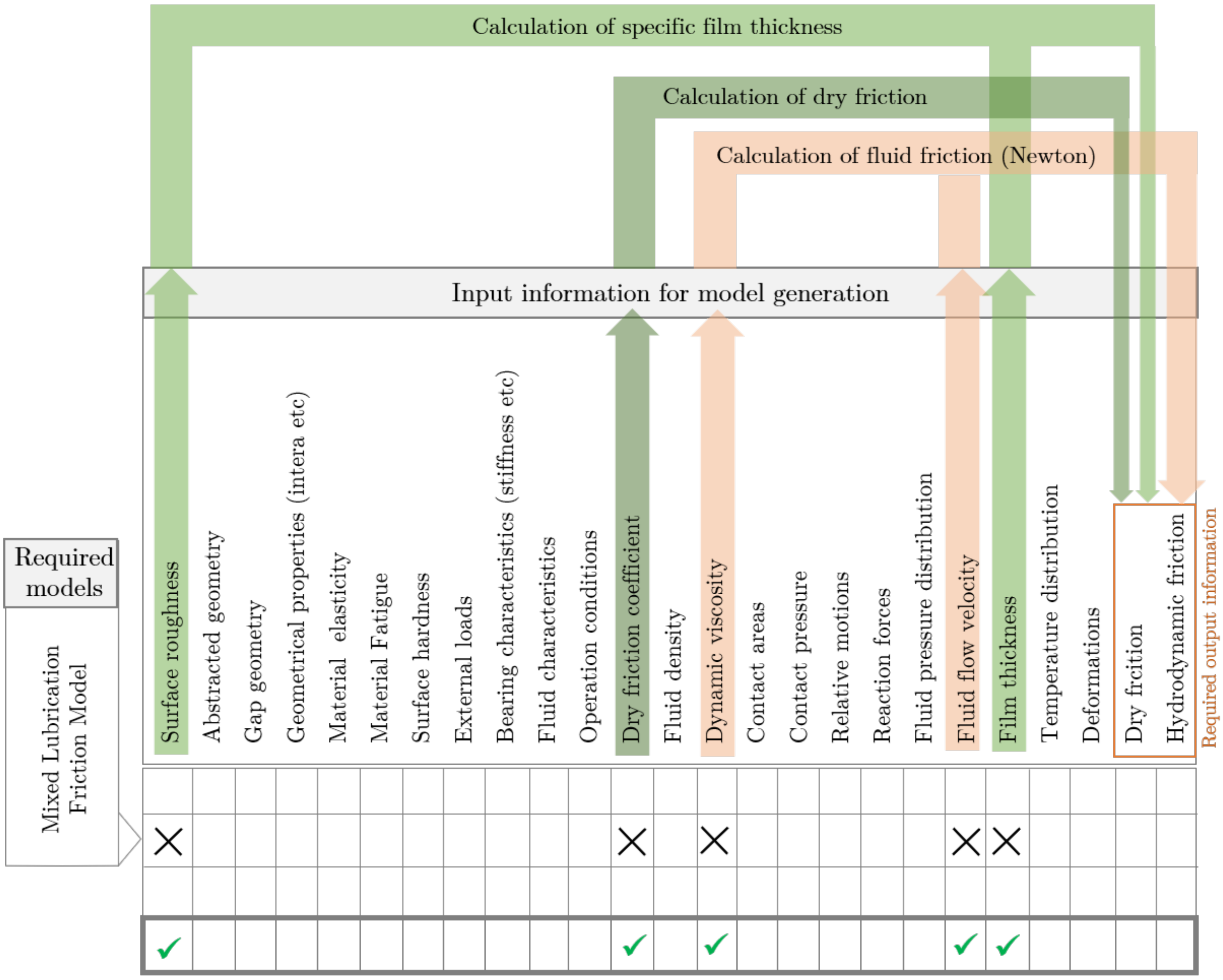


Figure 4.14: Derivation of required input information from required models

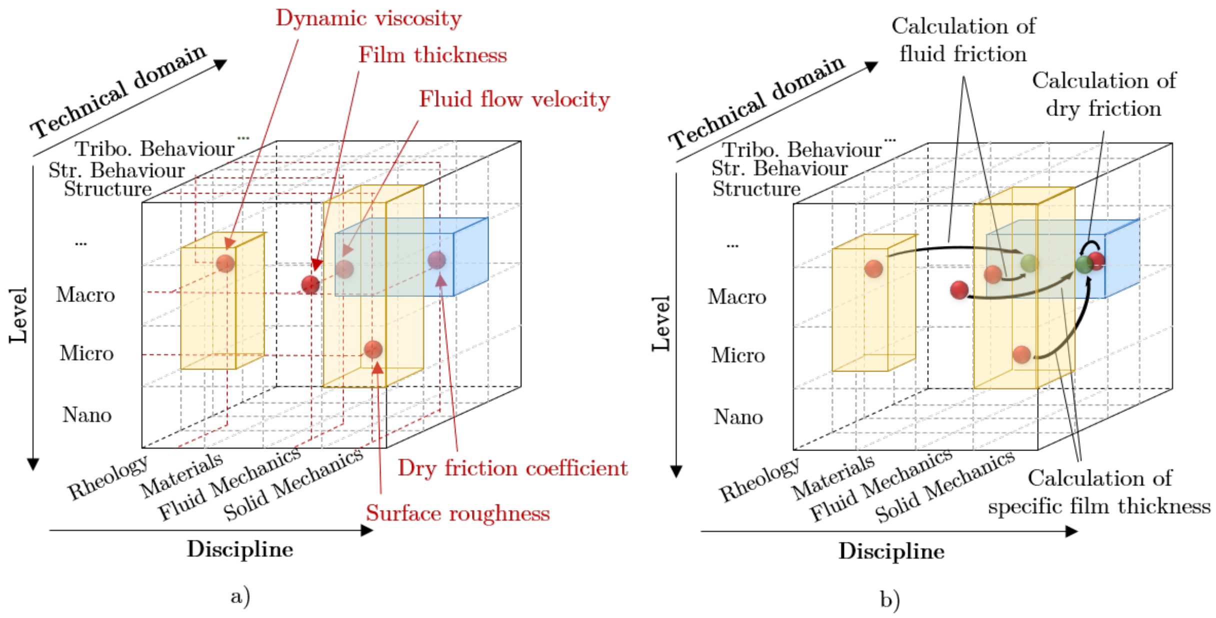


Figure 4.15: a) Illustration of required input information for *mixed lubrication friction model* generation
b) Methods to connect input and output information

This procedure - following the three described steps - provides a set of choices and limits the number of meaningful models and methods for a specific development effort. In the end, this supports decision making but can actually not replace a human to take the responsibility for an appropriate choice.

5 Discussion and Conclusion

Many approaches, especially in engineering, evolved over time to enable a development based on functions. Nevertheless, few sets of methods, processes and IT-based solutions are established for interdisciplinary development and intelligently connected production.[1]

Simulations and other virtual methods provide the enormous advantage to validate concepts in early project phases, because there is no need to manufacture a physical test specimen. In many cases this does not only reduce development costs, it also enables *frontloading*. The objectives of it are to shorten innovation process duration, while improving product quality and reducing development costs and risks (e.g., through early verification & validation tasks). Frontloading aims to provide these advantages by shifting result critical tasks to earlier development phases. The reasons for that are higher possibilities of cost reduction with smaller costs for concept changes in earlier phases.[2]

The result of the approach in chapter 4 was a selection of models and methods by identifying relevant system information. After determining the required model landscape, methods to generate these models and their required input information were outlined in section 4.6. Figure 5.1 illustrates some possibilities to provide these pieces of information. This should demonstrate, that further decisions are required, which cannot be done by an analytical method only. These decisions have to be made individually, as many factors affect them, but methodologies support decision making. For example, it depends on which test machinery is already available or how much experience a department has with a certain method. Furthermore, expert knowledge about valid assumptions and simplifications can be very valuable, especially in simulations.

5.1 Applicable system models

As *system models* are an essential concept of *model-based systems engineering*, the definitions presented in chapter 3 are a main output of this thesis. The clarification of frequently used terms and a differentiation of *system models* and *specific models* should enable common understanding. In general, models are created to deal with complexity by enabling better system understanding and providing a base for communication

---

[1] Eigner 2014, p.3.

[2] Eigner and Stelzer 2009, p.41.

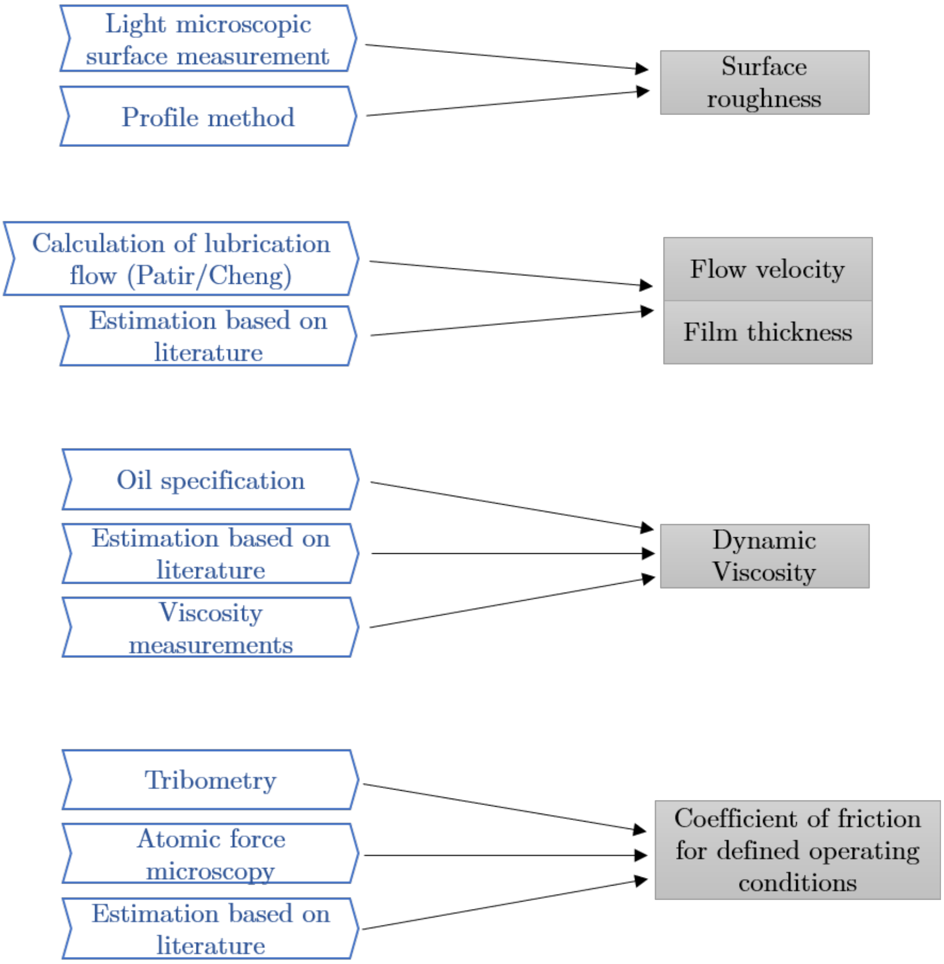


Figure 5.1: Required input information for model generation and possibilities to provide them

among stakeholders. Furthermore, better product quality and reduced risks can be achieved following a model-based approach.[3]

The classification scheme for models described in this thesis can be used as base for further steps towards a virtual development approach. Only if the purpose and scope of different model types are clear, decisions about which models should be implemented, can be made.

---

[3] INCOSE-UK 2015.

5.2 Improved product development

Processes of product development all have a general objective in common. They all support human beings to cooperatively work on a joint goal.[4] Purposefully implemented methods can support efficient and effective development of solution alternatives.[5]

Tribology was chosen as use case, as it proved to be especially challenging to fully understand tribological systems and associated processes. It has always been an objective of engineering to make technical systems even more efficient while improving its functionalities. Reducing losses contributes to overall efficiency gains, independent of the technical system. Not only friction in internal combustion engines have to be reduced, but also moving parts in power plants (e.g., gas turbines or rotors of wind power plants) in order to contribute to the global objective of reduced energy consumption through higher efficiency of technical systems.

---

[4] Lindemann 2016, p.19.

[5] Lindemann 2016, p.156.

6 Outlook

The presented definitions regarding *system models* should act as starting point for further research activities. In future investigations, especially the interdependecies with other approaches like *product lifecycle management* have to be evaluated. Embedding different approaches in a greater context could generate possible synergies, which could be utilized by implementing interfaces between them.

Furthermore, the interactions of models and methods have to be further investigated. The classification scheme for models visualized as *system cube* could be further evolved and implemented into a database. Methods represent connections between different pieces of system information and could therefore be integrated into this database. By observing the enormous number of available development methods and occurring data, interfaces between entities (models, methods, repositories) have a very important role. Companies like *AVL* developed platforms for integration. The so-called *integrated open development platform (IODP)* provides interfaces to implement co-simulation and connection of simulation models and the hardware.[1] Future approaches have to lay emphasize on interfaces to make a implementation in a bigger landscape of tools and models possible.

Many approaches have the objective to describe cybertronic systems by a digital representation, which reflects their information over the whole lifecycle. The concept of a so-called *digital twin* is strongly connected to product lifecycle managment approaches.[2]

It is often described as set of virtual information constructs fully describing an actually manufactured product or a potential future product from nano (atomic) level to macro level. Every information that could be obtained from the physical product could also be obtained from its digital twin.[3] To come closer to this vision of a digital twin, as many aspects of a system as possible have to be modeled in form of semi-formal or formal models. Therefore, the approach of this thesis supports the realization of a digital twin. Nevertheless, the challenges already mentioned in this thesis, from increasing system complexity to higher amount of emerging data, require new methodologies and connection of different approaches to realize a digital twin. This should lead to a more efficient development, decreasing engineering and production efforts in parallel to improved quality and availability of the product.

To establish a database about system information and to successfully associate models and methods in development to it, data warehouse management techniques have to

---

[1] Tao et al. 2017, p.2.

[2] Eigner, Koch, et al. 2017, p.52.

[3] Grieves and Vickers 2017, p.93 f.

be applied. Implementations of multidimensional data structures are well established for other purposes and can be adapted for model-based development.[4]

It seems not likely that the complexity of systems and also the variety of available methods, models and tools will reduce, rather they will further increase. Methodologies, classification schemes and procedure models have to be further developed to provide a competitive development environment, which enables successful product development.

---

[4] Farkisch 2011, p.11 ff.

List of Figures

List of Tables

List of References

Akao, Y. (1992). *QFD - Wie die Japaner Kundenwünsche in Qualität umsetzen.* Verlag Modeme Industrie.

Albers, A., K. Müller-Glaser, C. Schyr, and M. Kühl (2007). “Model-based powertrain development, Integrated tool chain from design to validation”. In: *ATZ* 2, pp. 15–17.

Angel, H. F. (2015). “No believing without emotion: The overlapping of emotion and cognition in the model of credition”. In: *Studies in Science and Theology* (15).

Archard, L. F. (1980). “Wear theory and mechanisms”. In: *Wear Control Handbook.* Ed. by Peterson. Ed. by Winer. American Society of Mechanical Engineers New York.

Bajaj, M., D. Zwemer, R. Yntema, A. Phung, A. Kumar, A. Dwivedi, and M. Waikar (2016). “MBSE++ Foundations for extended model-based systems engineering across system lifecycle”. In: *26th Annual INCOSE International Symposium.* Intercax LLD Atlanta.

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

Banks, J., J. Carson, B. Nelson, and D. Nicol (2001). “Prentice Hall International Series in Industrial and Systems Engineering”. In: *Discrete-Event Simulation.* Ed. by W. J. Fabrycky and J. H. Mize. Fourth edition. Pearson.

Bhushan, B. (2013). “Principles and applications of tribology”. In: *Tribology Series.* Second edition. John Wiley & Sons.

Boehm, B. (1979). “Guidelines for Verifying andValidating Software Requirements and Design Specifications”. In: *Euro IFIO 79.* Ed. by P. A. Samet. North-Holland Publishing Company Amsterdam.

Buerlow, I. von (n.d.). “Systemgrenzen im Management von Institutionen. Der Beitrag der weichen Systemmethodik zum Problembearbeiten”. In: *Wirtschaftswissenschaftliche Beiträge.* Physica-Verlag Heidelberg.

Bunk, W., J. Hansen, and M. Geyer (1981). “Tribologie. Reibung. Verschleiß. Schmierung.” In: *Metallurgie, Werkstoffentwicklung, Rückgewinnung.* Ed. by BMFT. Springer Berlin.

Coulomb, C. A. (1821). *Theorie des machines simple.* Bachelier Paris.

Czichos, H. and K.-H. Habig (2010). *Tribologie Handbuch. Tribometrie, Tribomaterialien, Tribotechnik.* Third edition. Vieweg+ Teubner Wiesbaden.

Dillerup, R. and R. Stoi (2016). *Unternehmensführung. Management & Leadership. Strategien –Werkzeuge – Praxis.* Fifth edition. Franz Vahlen München.

Dvorak, D. (2013). *Model-centric engineering. Part 1. An introduction to model-based systems engineering.* Jet Propulsion Laboratory. California Institute of Technology. URL: `http://nescacademy.nasa.gov/review/downloadfile.php?file=SEWG_MBSE_Module_1_v20.pptx&id=251&distr=Public` (visited on 02/21/2019).

Dvorak, D. (2017). *Model-centric engineering. Part 2: Introduction to System Modeling.* Jet Propulsion Laboratory. California Institute of Technology. URL: `http://nescacademy.nasa.gov/review/downloadfile.php?file=SEWG_MBSE_Module_2_v8.pptx&id=2` (visited on 02/21/2019).

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

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

Eigner, M., W. Koch, and C. Muggeo (2017). *Modellbasierter Entwicklungsprozess cybertronischer Systeme. Der PLM-unterstützte Referenzentwicklungsprozess für Produkte und Produktionssysteme.* Springer Vieweg.

Eigner, M. and R. Stelzer (2009). *Product Lifecycle Management. Ein Leitfaden für Product Development und Life Cycle Management.* Second edition. Springer Berlin.

Estefan, J. A. (2008). "Survey of model-based systems engineering (MBSE) methodologies". In: *INCOSE MBSE Initiative.*

Farkisch, K. (2011). *Data-Warehouse-Systeme kompakt. Aufbau, Architektur, Grundfunktionen.* Springer Berlin.

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

Fels, G., C. Lanquillon, H. Mallow, F. Schinkel, and C. Schulmeyer (2015). "Technik. Wirtschaft – Recht – Technik". In: *Praxishandbuch Big Data.* Ed. by J. Dorschel. Springer Fachmedien Wiesbaden.

Forrester, J. (1968). *Principles of Systems.* Second edition. Massachusetts Institute of Technology. Wright Allen Press.

Forrester, J. (1971). *World Dynamics.* Second edition. Wright Allen Press.

Friedenthal, S. and A. Moore (2012). *A Practical Guide to SysML. The Systems Modeling Language.* Elsevier.

GfT (2002). “Arbeitsblatt 7 - Verschleiß, Reibung. Definitionen, Begriffe, Prüfung”. In: *Tribologie.* Gesellschaft für Tribologie.

Gilz, T. (2014). “Requirements Engineering und Requirements Management”. In: Eigner, M., D. Roubanov, and R. Zafirov. *Modellbasierte virtuelle Produktentwicklung.* Ed. by Eigner. Ed. by Roubanov. Ed. by Zafirov. Springer-Verlag Berlin Heidelberg.

Grebe, U. D. and R. Fischer (2018). “Entwicklungen für die Mobilität in einer digitalisierten Welt”. In: *ATZ* Jubiläumsausgabe 2018, pp. 86–94.

Greenwood, J. A. and J. H. Tripp (1971). “The Contact or Two Nominally Flat Rough Surfaces”. In: *Proceedings of the Institution of Mechanical Engineers.*

Grieves, M. and J. Vickers (2017). “Digital Twin: Mitigating Unpredictable, Undesireable Emergent Behavior in Complex Systems”. In: *Transdisciplinary Perspectives on Complex Systems. New Findings and Approaches.* Ed. by F.-J. Kahles, S. Flumerfelt, and A. Alves. Springer International.

Guastello, S. (2014). *Human factors engineering and ergonomics. A Systems Approach.* Taylor & Francis Group CRC Press.

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

Hart, L. (2015). “Introduction To Model-Based System Engineering (MBSE) and SysML”. Presented at the Delaware Valley INCOSE Chapter Meeting July 30, 2015. In: URL: `https://www.incose.org/docs/default-source/delaware-valley/mbse-overview-incose-30-july-2015.pdf` (visited on 11/08/2018).

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

Hick, H., C. Faustmann, M. Bajzek, J. Edtmayer, and S. Walch (2019). *System models and model classification in tribological system development.* Manuscript submitted for publication.

Hirz, M., W. Dietrich, A. Gfrerrer, and J. Lang (2013). “Overview of Virtual Product Development”. In: *Integrated Computer-Aided Design in Automotive Development.* Springer Berlin-Heidelberg.

Höll, J., C. König, Y. Ritter, F. Auris, T. Bär, S. Süß, and M. Paul (2018). “Seamless simulation toolchain for virtual engineering and virtual commissioning of smart factories”. In: *18. Internationales Stuttgarter Symposium. Automobil- und Motorentechnik.* Ed. by M. Bargende, H.-C. Reuss, and J. Wiedemann. Springer Fachmedien Wiesbaden, pp. 797–812.

Holmberg, K. and A. Erdemir (2017). “Influence of tribology on global energy consumption, costs and emissions”. In: *Friction* 5 (3), pp. 263–284.

Holt, J., S. Perry, and M. Brownsword (2016). “Foundations for Model-based Systems Engineering. From Patterns to Models”. In: *IET Professional Applications of Computing Series*. The Institution of Engineering and Technology.

Hybertson, D. W. (2009). “Model-oriented systems engineering science. A Unifying Framework for Traditional and Complex Systems”. In: *Complex and Enterprise Systems Egineering Series*. Taylor and Francis Group.

INCOSE (2007). *Systems Engineering Vision 2020*. Document No.: INCOSE-TP-2004-004-02. Version Revision 2.03. Technical Operations. International Council on Systems Engineering (INCOSE). URL: `http://www.ccose.org/media/upload/SEVision2020_20071003_v2_03.pdf` (visited on 02/21/2019).

INCOSE-UK (2015). “Why do MBSE? Z9: What is Model Based Systems Engineering”. In: *Z-Guide* (1). Ed. by H. Woodcock. URL: `http://www.incoseonline.org.uk/Documents/zGuides/Z9_model_based_WEB.pdf` (visited on 05/05/2019).

ISO/IEC/TS24748 (2018). *Systems and software engineering - Life cycle management*. International Organization for Standardization.

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

ISO4288 (1996). *Geometrical Product Specifications (GPS) - Surface texture: Profile method Rules and procedures for the assessment of surface texture*. International Organization for Standardization.

Jost, H. P. (1966). *Lubrication (Tribology) Education and Research. A report on the present position and industry’s needs*. First edition. Her Majesty’s Stationery Office London.

Koenig, B. (2011). *Vorlesung Modellierung. Modellierungsmethoden der Informatik*. URL: `http://www.ti.inf.uni-due.de/fileadmin/public/teaching/mod/slides/ws201112/einfuehrung.pdf` (visited on 04/04/2019).

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

Kragelsky, I. V., M. H. Dobychin, and V. S. Kombalov (1977). *Foundations of calculations for friction and wear*. Trans. Russian by N. Standen. Mashinostroenie. Pergamon Press.

Kranabitl, P. (2019). “Interaction of methods in product development”. Master’s Thesis. Graz University of Technology.

Lamsweerde, A. van (2009). *Requirements Engineering. From System Goals to UML Models to Software Specifications*. John Wiley & Sons.

Law, A. (2015). *Simulation Modeling and Analysis*. Fifth edition. McGraw-Hill Education New York.

Lehigh-University (2019). *Tribology Lab*. URL: `https://lehigh.edu/~intribos/images/mattrib.jpg` (visited on 01/17/2019).

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

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

Liu, S. (2004). *Formal Engineering for Industrial Software Development. Using the SOFL Method*. Springer Berlin Heidelberg.

Long, D. and Z. Scott (2011). *A Primer for Model-Based Systems Engineering*. Second edition. Vitech.

Maassen, F. (2015). “Reibung. Grundlagen, Komponenten, Systeme, Perspektiven”. In: *Handbuch Verbrennungsmotor*. Ed. by R. van Basshuysen and F. Schäfer. Seventh edition. Springer Fachmedien Wiesbaden, pp. 439–455.

Madni, A. and M. Sievers (2018). “Model-based systems engineering. Motivation, current status and research opportunities.” In: *Systems Engineering*, pp. 172–190.

Maier, M. (1998). “Architecting Principles for Systems-of-Systems”. In: *Systems Engineering* 1, pp. 267–284.

Maier, M. and E. Rechtin (2009). *The art of systems architecting*. Third edition. CRC Press Taylor & Francis Group.

Maritan, D. (2015). *Practical Manual of Quality Function Deployment*. Springer International Publishing Switzerland.

Merkle, A., B. Huber, and T. Spitznagel (2018). “Symmetrische Formhonung in den neuen BMW 3- und 4-Zylinder-Ottomotoren. Systemverständnis eröffnet zukünftige Wege. 6. ATZ-Fachtagung Tribologie”. In: *Reibungsminimierung im Antriebsstrang*. Ed. by J. Liebl. Springer Vieweg.

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

North, K. and G. Kumta (2018). *Knowledge Management. Value Creation Through Organizational Learning*. Springer Texts in Business and Economics. Second edition. Springer.

Patir, N. and H. S. Cheng (1978). “An Average Flow Model for Determining Effects or Three Dimensional Roughness on Partial Hydrodynamic Lubrication”. In: *ASME Journal for Lubrication Technology* 100, pp. 12–17.

Peschke, F. (2017). “Product lifecycle management (PLM)”. In: *Pocket Power*. Ed. by Kamiske. Hanser.

Pischinger, R., M. Klell, and T. Sams (2009). “Thermodynamik der Verbrennungskraftmaschine. Der Fahrzeugantrieb”. In: ed. by H. List. Third edition. Springer Wien New York.

Porsche. (2016). *World champion turbo four-cylinder*. Porsche newsroom. URL: `https://newsroom.porsche.com/en/technology/porsche-motorsports-turbo-four-cylinder-engine-919-hybrid-12226.html` (visited on 04/04/2019).

Pucher, E. (2015). “Schadstoffe und ihre Entstehung. Grundlagen, Komponenten, Systeme, Perspektiven”. In: *Handbuch Verbrennungsmotor*. Ed. by R. van Basshuysen and F. Schäfer. Seventh edition. Springer Fachmedien Wiesbaden, pp. 825–830.

Rabinowicz, E. (1965). *Friction and wear of materials*. John Wiley and Sons New York.

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

Reynolds, O. (1886). “On the Theory of Lubrication and Its Application to Mr. Beauchamp Tower’s Experiments. Including an Experimental Determination of the Viscosity of Olive Oil”. In: *Philosophical Transactions of the Royal Society of London*.

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

Royce, W. (1970). “Managing the Development of Large Software Systems: Concepts and Techniques”. In: *Technical Papers of Western Electronic Show and Convention (WesCon) Los Angeles*.

Rudolf, S. and E. Schrey (2015). “Product lifecycle management – Herausforderungen einer ganzheitlichen IT-Unterstützung der Geschäftsprozesse”. In: *Complexity Management Journal*, pp. 8–12.

Scherge, M., B. Kehrwald, and A. Gerve (2002). “Tribologie und Motormechanik”. In: *MTZ - Motortechnische Zeitschrift* 63 (3), pp. 202–209.

Schlosser, A., B. Kinoo, W. Salber, S. Werner, and N. Ademes (2007). “Modern Powertrain Development with Model-Based Methods”. In: *MTZ* Volume 68 (02|2007).

Senge, P. (1990). *The fifth discipline. The Art and Practice of the learning organization*. Bantam Doubleday Dell Publishing Group.

Sommer, K., R. Heinz, and J. Schoefer (2014). *Verschleiß metallischer Werkstoffe. Erscheinungsformen sicher beurteilen.* second edition. Springer Wiesbaden.

Sommerville, I. (2016). *Software Engineering.* Tenth edition. Pearson Education Limited.

Speckens, F.-W., F.-G. Hermsen, and J. Buck (1998). “Konstruktive Wege zum reibungsarmen Ventiltrieb”. In: *MTZ - Motortechnische Zeitschrift* 59 (3), pp. 176–181.

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

Stokes, G. G. (1845). “On the Theories of Internal Friction of Fluids in Motion”. In: *Transactions of the Cambridge Philosophical Society*, pp. 287–305.

Stribeck, R. (1902). “Die wesentlichen Eigenschaften der Gleit- und Rollenlager”. In: *Zeitschrift des Vereins Deutscher Ingenieure*, pp. 1341–1348, 1432–1438, 1463–1470.

Tao, J., A. Ennemoser, H. Petutschnig, G. Lang, and A. Traussnig (2017). “Vehicle Thermal Management System Control Strategies - from MiL to HiL (CiL)”. In: *ATZ live. VPC – Simulation und Test 2016: Herausforderungen durch die RDE-Gesetzgebung.* Ed. by J. Liebl and C. Beidl. Springer Wiesbaden.

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

VDI2206 (2004). *Design Methodology for Mechatronic Systems.* Verein Deutscher Ingenieure.

VDI2221 (1993). *Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte.*

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

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

Weilkiens, T., J. Lamm, S. Roth, and M. Walker (2016). “Model-based System Architecture”. In: *Wiley Series in Systems Engineering and Management.* Ed. by A. Sage. John Wiley & Sons.

Wen, S. and P. Huang (2018). *Principles of Tribology.* Second edition. John Wiley & Sons Singapore.

Wiener, N. (1948). *Cybernetics or Control and Communication in the Animal and the Machine.* MIT Press.

Zafirov, R. (2014). “Modellbildung und Spezifikation”. In: Eigner, M., D. Roubanov, and R. Zafirov. *Modellbasierte virtuelle Produktentwicklung*. Hanser.

Appendix

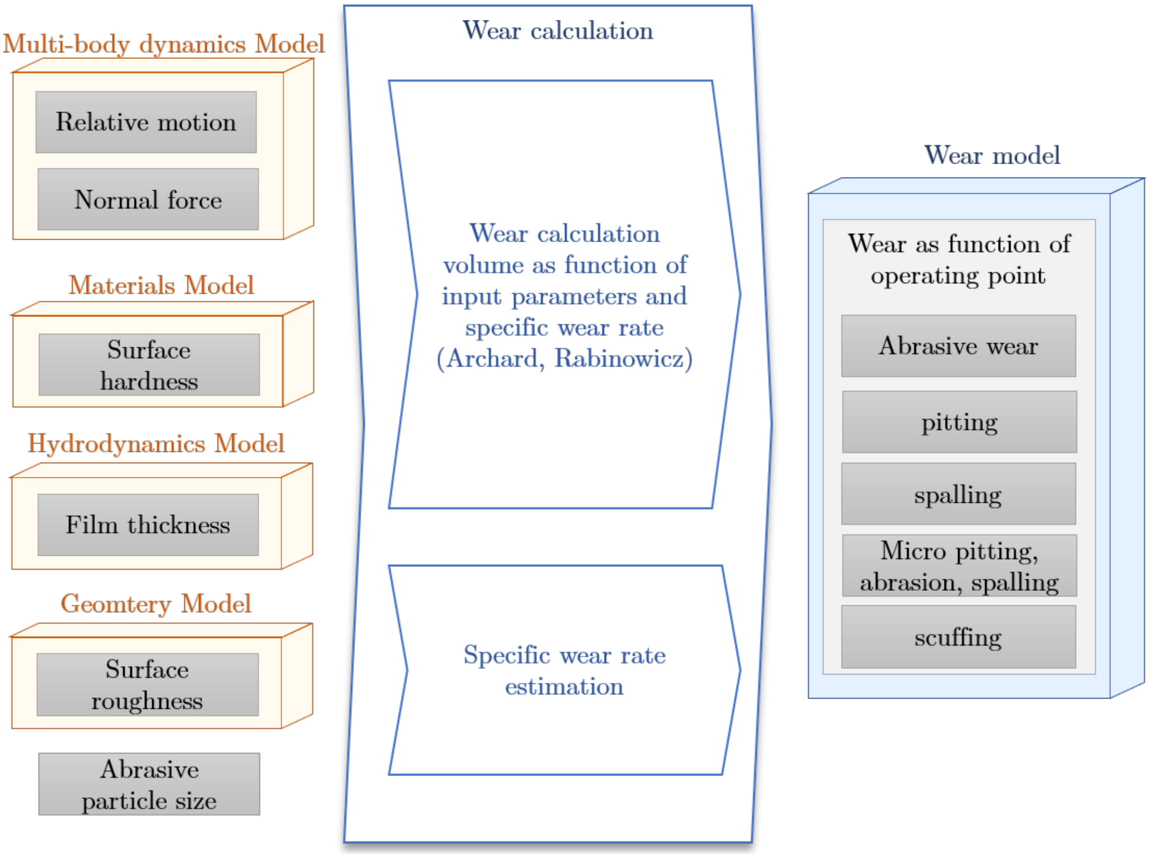


Figure A.1: Illustration of a *wear calculation*, consisting of specific wear rate estimation and a empirical expression for wear volume, to generate a *wear model*

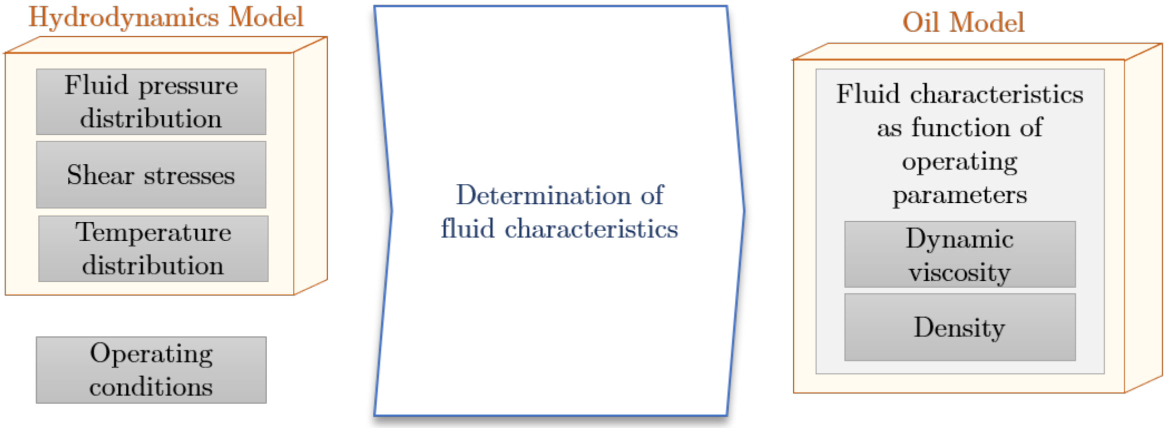


Figure A.2: Generation of an *oil model* by determination of fluid behavior under defined conditions

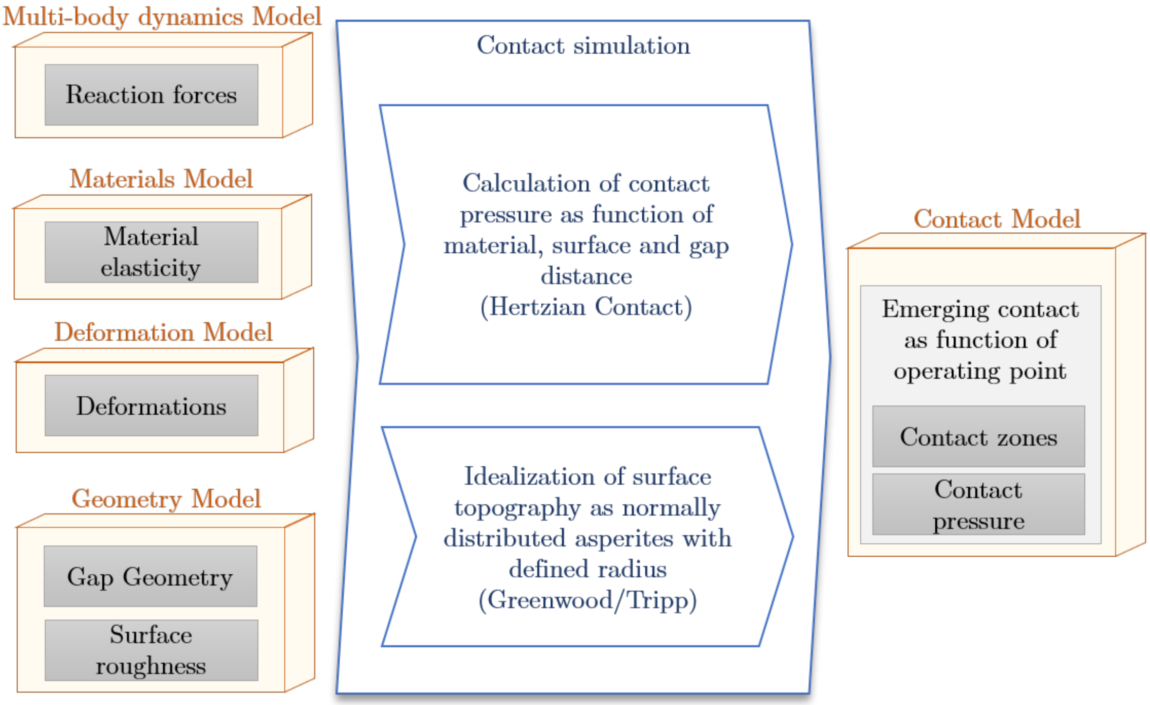


Figure A.3: Calculation, consisting of hertzian contact pressure calculation and idealization of surface topography, to generate a *contact model*

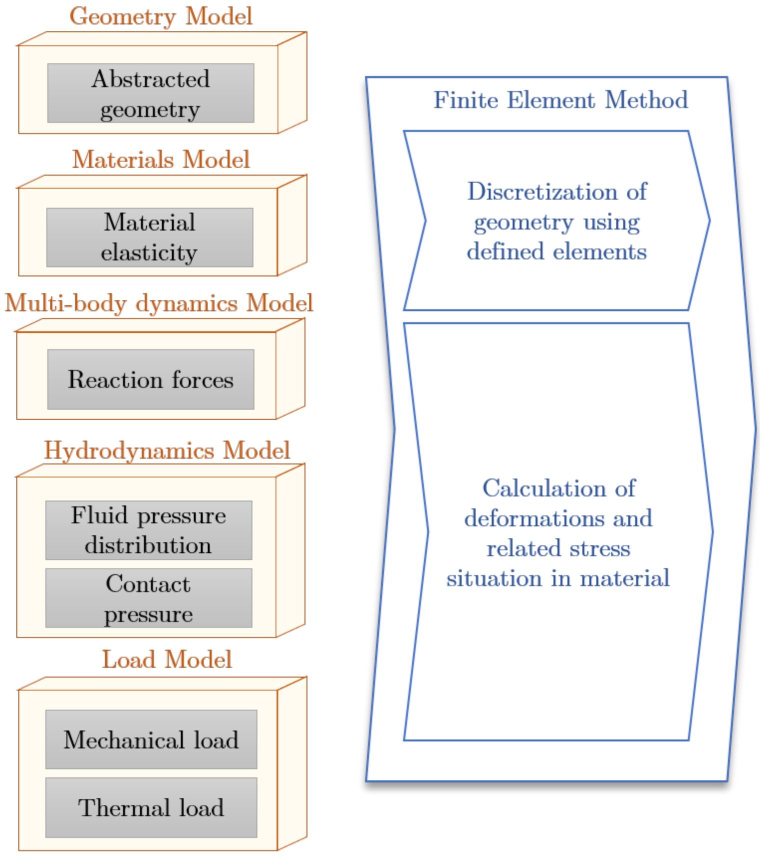


Figure A.4: Illustration of a *finite element method (FE)* to generate a *deformation model*

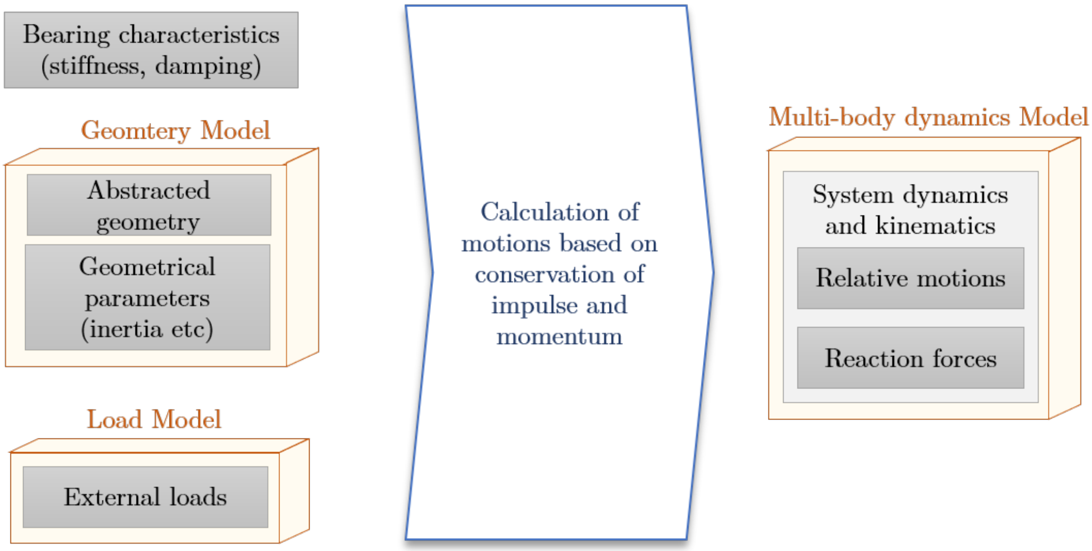


Figure A.5: Generation of a *multi-body dynamics model (MBD)* by calculation of kinematic relations and dynamics