

## Research

# A fundamental concept for linking methods, system models, and specific models

Philipp Kranabitzl<sup>1</sup> · Clemens Faustmann<sup>1</sup> · Matthias Bajzek<sup>1</sup> · Stefan Kollegger<sup>1</sup> · Hannes Hick<sup>1</sup>

Received: 9 November 2023 / Accepted: 2 January 2024

Published online: 20 January 2024

© The Author(s) 2024 [OPEN](#)

## Abstract

In the field of model-based development, and especially model-based systems engineering, the efficient and effective use of system models and specific models is of great interest. However, models strongly depend on the method they are linked to. Methods require models as input and associated models that are part of the method itself. Subsequent, methods are applied to adapt, enhance or generate output models. If a model is analyzed regarding its fidelity, level of detail, or maturity, only a combined consideration of methods and models in the development approach leads to a valid assessment. Therefore, the principle link of methods, system models, and specific models is defined and a concept for the application in product development is presented. It is stated, that a combined view on methods and models leads to improved accuracy of information interpretation and therefore optimizes efficiency and effectiveness during the product development.

## Highlights

- Basis for method-model landscape modeling that provides a more efficient and target-oriented documentation and management of models and methods
- Fundamental concept that defines the terms models and methods for system development
- Examples for methods and models in system development are described including their interactions

## 1 Introduction

In order for a company to survive, it has to stay competitive on the market. For this purpose, the company continuously develops/produces products that attract customers on the market who are willing to pay more for the product than the company spends on creating the product. [1] To reach this goal, companies have to work effective and efficient towards the company's goals.

One of the biggest challenges of projects is to harmonize project costs, time and quality. The movement of one of these criteria may put pressure on the other criteria. This interdependence is called the 'iron triangle' and is one of the most widely accepted success criteria in project management [2]. For example if the time to bring a product to market is shortened because of the pressure of a competitor, the quality might suffer due to less time for extensive verification and validation activities. Additionally, supplier have to meet shorter delivery times which might increase the costs. Van

---

✉ Clemens Faustmann, clemens.faustmann@tugraz.at; Philipp Kranabitzl, philipp.kranabitzl@tugraz.at; Matthias Bajzek, matthias.bajzek@tugraz.at; Stefan Kollegger, stefan.kollegger@tugraz.at; Hannes Hick, hannes.hick@tugraz.at | <sup>1</sup>Institute for Machine Components and Methods of Development, Graz University of Technology, Graz, Austria.



Wyngaard puts it like this: “good, fast or cheap—pick two” [3]. Due to globalization and the competitiveness of the markets, the concept stated by Wyngaard [3] is not satisfying for companies. To maximize all three of the aspects of the ‘iron triangle’, it is of critical importance to focus on which information is available in what level of detail to a certain point in the development process.

The available information at the beginning of a project heavily depends of what kind of development it is. There are three types of development namely variant, adaption and new development [4]. If the system under development is new, which means that there is no system knowledge available in the company as well as on the market from competitors, system understanding needs to be build up from scratch. This often results in longer research and development time. In comparison, variant and adaption development is characterized by the availability of a certain basis of information about the system, which is available from the beginning of development. In variant and adaption development, requirements from a previous development can be partly reused and is changed by one or more stakeholder to meet the variants requirements. The critical aspect in these development types is to understand what information from the previous projects is still valid in the current one, and what type of information is not useful anymore due to the change of requirements. To get the best out of technology for a business, the understanding of the importance of information and the consequent steps are key. Lawyers, journalists, engineers, logisticians and many more are heavily dependent on sound information about their subject for professional success [5].

The information over the development process about the system is primarily generated and refined by methods. Therefore, methods, besides some other influencing factors, define the quality of available information [6]. In a model-based systems engineering (MBSE) related development approach, the information generated by methods are contained within models. MBSE is an approach that relies on the principles and best practices of systems engineering and is characterized by the formalized application of modeling to support system development [7]. The core of every MBSE approach is the system model. The system model provides the basis for interdisciplinary collaboration and therefore the selection of appropriate methods, specific models, and tools. The use of methods, system and specific models in general is a measure to increase the effectiveness (doing the right things) and the efficiency (doing things right) of product development as well as to ensure the quality of the results throughout the entire product life cycle [8]. The model itself cannot conclude statements based on the information contained. Only in combination with the method which generated the information, an adequate interpretation of the information is possible. Consequently, this paper focuses on the interdependence of methods and models to gain development relevant statements.

The central statement of this contribution is, that only with a landscape considering both, methods as well as models, it is possible to provide traceability and to assess the quality of information regarding fidelity, level of detail and maturity. The following sections define the relevant terms and principle concepts regarding the linking of methods and models. Based on that, a generic concept is illustrated to visualize these links, which is explained based on an example of a shaft in an automotive powertrain development. Finally, this generic concept is used to connect multiple methods and models of a development approach, in order to show the relevance of this concept for practical application.

## 2 Methods and models in system development

According to Orloff [9] the human brain as a biological object has not changed for thousands of years. The structure, as well as the principles of the way the brain works, are still the same as they were 50.000 years ago [9]. Nevertheless, changes in the economy have developed at such a speed and diversity in recent years. This development affects what the human brain has to deliver and what human system such as societies or companies have to cope with. It can be observed, that it became difficult for companies to cope with this speed of changing and evolving technologies, for example. Companies will have to come to terms with the fact that the environment is becoming more turbulent, events are following each other ever more quickly and the boundary conditions are becoming less and less reliable. All these influences ultimately lead to the fact that product development processes, for example, are becoming increasingly difficult to control. Since our brain is evolutionary not able to cope with this fast-changing situation, methods exist that help in individual phases of these difficult and complicated processes. These methods support the user to systematically analyze the current problem, to work on and find a consistent and logical solution for it [10].

Based on the requirements defined by the stakeholders, engineers have to transfer the specific development problem into a solution which is at least acceptable, if not succeeds the expectations of the stakeholders. In addition to the technical and functional requirements, the engineers also have to take production and assembly into account [8]. The key to

develop the product or system in the required time, with a certain quality and not exceeding a cost target, is to apply appropriate methods. The term “method” according to VDI 2223 [11] describes a well-planned procedure for achieving a defined objective. Lindemann states that methods provide a sequence of certain steps and how these steps are to be carried out. Methods are additionally characterized by a strong operational character [12]. There is a wide variety of methods available for companies. On a high level, methods can be distinguished between development, organizational, and generally applicable methods. The focus in this paper is on development methods which aim to achieve a given objective, e.g., a document or object [13]. Consequently, they differ from organizational methods, whose goal is the design of a process, and from generally applicable methods which can be applied to various situations with different objectives.

Choosing the right methods for the development of a certain project is critical for the success of the project. Boundary conditions like what time is required to execute the method, the costs for applying the method in the project, the required knowledge and experience of employees who work with the method, the required tools which support the method execution and many more influence the selection and of course the successful application of methods [14]. One aspect which has been neglected in the past as an important aspect for method selection is the system model. Besides all the aspects which Weevers [14] argues, the system model provides a structured view on the mechatronic system and therefore accurately points out all the system aspects which need to be developed in detail.

The main reason why a method is chosen to be applied is because it delivers a certain output information which is needed to further proceed with the development. There are many different methods which deliver the same type of output information but on a different level of quality. For example, you can calculate the needed diameter of the gearbox shaft by a simple calculation by hand or by a very precise finite element simulation. Just because the one method is more accurate, does not mean that the other one cannot be beneficially applied. In an early stage of the development of an automotive powertrain, a rough diameter is needed in order to gain first estimations. In this phase of the development, the less accurate method would be suitable because the required level of detail of the output is low (in this case the geometrical information in form of a diameter). The key is to know which information is required and how accurate the output of the applied method has to be and to act accordingly.

In the context of MBSE the output information of methods is contained within models. Therefore, the understanding of the term ‘model’ has to be discussed on a general basis and elaborated for the context of this contribution. Generally speaking, models represent certain aspects of a system under consideration [15]. From a theoretical viewpoint, models can be defined by considering three elementary properties. First, the mapping property describe that models always represent a certain system—an original entity—or parts of it. Secondly, models reflect pragmatism, as models have the purpose to enable further investigations of the system of interest by replacing the artificial system or the reality. And thirdly, models do not represent all aspects and properties of their original counterpart, rather they are simplified representations that focus on defined aspects [15, 16].

These elementary properties define the general term model, but allow space for interpretation and ambiguity. Therefore, different types of models have to be discussed in order to limit the focus of this contribution regarding models. There are several ways to classify and differentiate models. For example, system models and specific models can be distinguished depending on their focus regarding the system hierarchy [17]. Moreover, regarding its nature, models are classified in physical and empirical models [18] or depending on the type of information representation in graphical, technical, semantic and semantic-scientific [19]. Depending on the purpose of the model for development activities, the following types are distinguished: models for specification, models for design exploration, models for documentation, and models for communication [16]. The following sections focuses on development models. This term is defined as all models, that are developed, utilized, and purposefully generated in the course of system development. Development models are not limited to a certain model type. For example, a system model (digital, descriptive, high level of simplification) in SysML [20], a specific simulation model (digital, quantitative, low to high level of simplification) and even a rig for engine testing (physically manifested/hardware-based, quantitative, low level of simplification) are referred to as development models.

In the context of system development, models are not an end in itself. Models are embedded in a development framework consisting of processes, methods, organizations and tools in order to provide an added value [21]. The focus of this contribution is the interaction of models and methods. Processes, organization and tools are of course important to implement a model-based development approach within a company as well but are not the focus of this paper. Nevertheless, it is important for the understanding of the links between methods and models to briefly discuss the elementary links between methods and processes on which this paper is based. A process in general is a logical sequence of tasks that need to be performed in order to meet a particular objective. Following, a process defines “WHAT” is to be done, without answering in detail “HOW” it is executed [22]. A method in comparison refers to a rule-based and planned

procedure for executing a task in order to achieve the defined goal. In other words, it defines “HOW” the task has to be performed [12]. From a system development standpoint it can be stated that the process is in the lead and logically aligns the methods which perform the tasks.

Development models represent a certain value in the project, as they describe and specify the system, and are used to prove that the system characteristics meet the requirements and required functions can be fulfilled. Friedenthal et al. [20] state, that the purpose of the generation and use of models should therefore be specified in this regard. This purpose is determined by model breadth, depth, and fidelity and should be balanced with the available schedule, budget, skill levels, and other resources [16, 20, 23].

### 3 Fundamental connection between methods, system models and specific models

As mentioned in the introduction, product development is a knowledge-intensive process. Simplified, it can be seen as a systematic process of knowledge and information generation, selection and transfer. However, this is often difficult because product development is not a very rigid process to a certain extent [24]. The necessary change of methods, models, tools and people in development phases inevitably leads to challenges in information continuity. In consequence, the project success heavily depends on the quality of available information at any given time during the product development process. At this point, the principal differentiation of data, information and knowledge has to be discussed. In literature, information is seen as data with meaning. This can be illustrated with an example of measurement data: a signal of a sensor is considered as data while the type of sensor and the measured medium define the meaning and lead to information (e.g., sensor voltage signal corresponds to fluid temperature). The information is contextualized within a model structure. For example, the model cube structure proposed by Hick et al. [17] is used to classify and structure models in order to define the context, e.g., the sensor data reflects the temperature of a cooling liquid within a vehicle at a specified position [17, 25, 26].

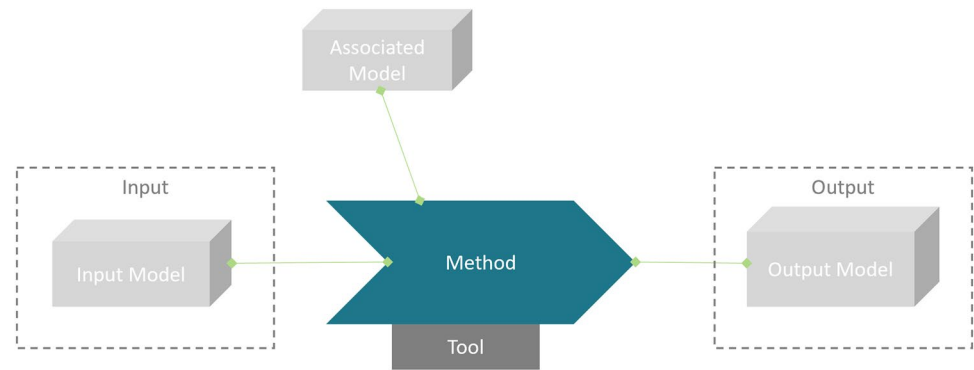
The quality of information within a model is essential for a successful development. By simply viewing a model, the user is not able to assess the information quality according to fidelity, level of detail and maturity. Model fidelity can be interpreted as the preciseness of the representation of a specific system aspect and how well impact parameters are considered, e.g., the derivation of a simulated temperature from the measured temperature in reality [27]. The level of detail of a model is related to the extent of the model and its content. However, it does not necessarily imply the fidelity, as also simple models can describe a system aspect in an appropriate quality. For instance, a linear function represents a linear behavior as good as a square model, even though the square model includes more parameters. The maturity of a model reflects how well the model is implemented and verified. For example, a model that predicted the temperature at a certain operating point with low derivation several times, has higher maturity compared to a model which has not been applied previously.

Without the declaration of fidelity, level of detail and maturity of the model, it is not possible to make beneficial decisions since the information basis is not clearly declared. However, the quality of a model is not the only factor to be considered. If it is not transparent how the information was generated and what the purpose of the model initially was, the information in the model cannot be interpreted in a proper manner. Therefore, only the combination of development methods and the corresponding models provides all information that is necessary to maximize development success.

In Fig. 1 the basic interdependence of a method and models is defined. The figure sets the spotlight on the methods and the models to emphasize the connection between them. Nevertheless, the link between the method and the development process is essential as well but is discussed in later sections. One important aspect that is shown is that a method is typically linked to several models. As described in previous sections, the input and the output of methods have to be considered. The input information could be a model or parts of a model, the same counts for the output of the method. Especially in model-based development approaches such as MBSE [20], models are preferred as information carrier or container because of the advantages described previously. Furthermore, there are also models that are essential for the processing of the method itself. For example, if the method relies on a numerical calculation, the model describing the physical effect as well as the model that describes the numerical discretization are an essential basis for the method, or in other words, they are integral parts of the method itself. These models that are integral parts of the method, are referred to as associated models.

Additionally, the application of methods may be supported by hardware or software tools [21]. The prerequisite for using a tool is a compatibility with the corresponding method (e.g. FMEA with the support of a spreadsheet program) or the method itself is partly integrated within the tool. In general, the tool follows the method. Therefore, the engineer has

**Fig. 1** Generic links between models and methods

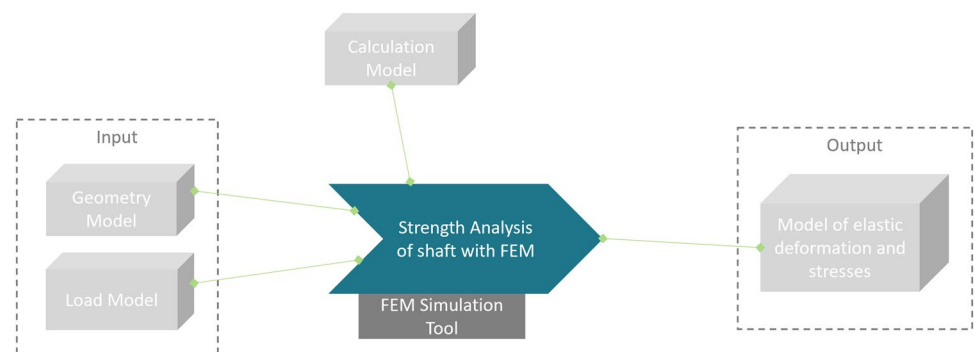


to define the method that has to be applied first, followed by the decision which tool could support this method application. However, the available tool landscape within an organization, the experience with a tool, or customer demands could be boundary conditions for decisions regarding method and tool selection.

As already stated, on the one hand an isolated review of a model does not provide all relevant information in order to assess its content regarding quality. On the other hand, the method application and the development strategy (which method should be applied, when and by whom in order to fulfill development objectives) is strongly influenced by related models. To emphasize this statement and to show the value of the generic concept in Fig. 1, an example is considered. As illustrated in Fig. 2, the method 'strength analysis of shaft with FEM (finite element method)' is considered as part of a system development. The objective of this method is to generate an output, namely a 'model of elastic deformation and stresses' of the shaft.

In order to assess the quality of the generated output, the aspects fidelity, level of detail and maturity are considered. The fidelity of the output model can be described in this case as how well the output model reflects the real behavior of the shaft under load with focus on elastic deformations and stresses. Firstly, this depends on the way how the output model was generated, in this case the method and the associated calculation model, and secondly on the input models of the method, e.g., how well is the geometry of the real shaft described with the input model 'geometry model'. To prove a certain fidelity, the model has to be verified by other methods such as a hardware test on a testbed with the real shaft loaded according to the load model. When moving forward to hardware tests, the production of the prototype is of utmost importance. There may be major differences between single prototype production and series line production shafts. The production process therefore needs to be considered in detail when assessing the model. Considering the level of detail of the output model, the method plays an essential role once again. The method and its applicant define if behavior regarding mechanical deformation and stresses is modeled for the whole shaft or if only certain critical (previously defined) areas are investigated in detail. This depends on the purpose of the output model and is heavily influenced by the development approach. Finally, the maturity of the output model depends on several influencing factors. Of course, the method application and the maturity of related input models and associated models influence the maturity of the output. For example, is the geometry model based on a first draft of the shaft geometry or is it the final shaft geometry model in CAD (computer aided design) that is used for production as well? Are loads on the shaft, described in the input 'load model', rough assumptions or are they measured in real life in a previously developed similar system? Furthermore, the method applicant or development team, with its experience and knowledge strongly influence

**Fig. 2** Example for links between models and methods



the outcome of the method as well as the tool that was used to apply the method. Concluding, the model fidelity, level of detail and maturity are relative ratings and it is not feasible to define an exact quantitative value for them.

The generic concept to illustrate a method and its relations to models can be applied for several types of methods in system development. It has to be stated, that this concept emerged out of the principles of MBSE but is not limited to digital models. Generally, also hardware models (e.g., an engine testbed that represents parts of the powertrain of the vehicle and reflects relevant aspects of the powertrain to enable tests of an engine) can be linked to methods based on this concept.

The question which model should be selected is a difficult one and is not the focus of this contribution. However, in accordance with the goal of modeling, the models are shortened to application-oriented views, i.e. only the information is represented which is required [15]. The individual perception of humans additionally affects the work with the model. If a common model is used by several individuals, ultimately not the model creator, but the model users define whether the model is beneficial for the development process or not [28]. In summary, the selection of the most appropriate models as input for a method or also the selection of the associated models (e.g., different equations for the same physical phenomenon) heavily depends on the development approach and related involved stakeholders.

#### 4 Methods and models in the context of a development approach

The concept of the elementary connection between methods and models will only provide substantial benefits if applied to the whole development process. The dependencies on a micro level are the basis for development relevant statements over multiple phases. To work efficiently on reaching the development goal and utilize on all the developed models, it is critical to plan right at the beginning what methods are going to be applied and what models should be developed. This is important since there are different strategies possible for every development. One company might focus on extensive virtual verification methods and tries to reduce hardware testing to a minimum. Another company may argue that with virtual testing methods a certain model fidelity is not reachable. Thus, virtual testing is done to a certain extent, but extensive hardware testing is additionally necessary. Generally, it cannot be said that one approach is better than the other, but it is critical to know and understand if (like in this example) the information provided in models is based on virtual or hardware tests.

Based on the requirements defined by the stakeholders, a rough “path” with aligned methods and models to fulfill those requirements needs to be defined, which has to go in line with the development approach. Processes and related activities and methods are distinguished as follows: In general, the process defines the ‘what’ while the methods define the ‘how’. Consequently, the process defines phases and activities with corresponding milestones and quality gates. Furthermore, inputs, outputs and assigned people (responsible, accountable, supportive and informative) are defined within processes. In contrast, methods are described in a more neutral way, which means that they can be applied for several process activities. Methods describe the structured step-wise approach to come from an input to an output, but they have to be assigned to a process, its phases and activities in order to define when the method has to be carried out and by whom [29].

In the considered example, a first draft of a set of methods and models can be derived based on the mentioned requirements, the involved stakeholder and the system model, which represents a generic system architecture (for example modeled with SysML or Capella). This first draft should serve as a kind of landscape for all people involved in the process. Following, engineers who are specialist and/or responsible for one method, can align themselves with the methods which are performed before and after their own to maximize the output. The engineer is also able to interpret the models accordingly, since it is known how the data from the models was developed. Of course, this first rough “path” is being adjusted along the development process and must not be seen as set in stone. Lower level requirements for example cannot be defined in detail at the beginning since they depend on higher level system solutions. Therefore, this path needs to be substantiated over the course of the development process.

For the information to be available over the course of the development process for the method responsible, a model management system needs to be in place. Every model has to be equipped with certain attributes like ID, creation date, creator, method/s used and many others to ensure traceability. Since many models are being improved and changed over time, version management is also a big topic. If these aspects are not considered, it is difficult to interpret the model in a right way or worse, the required model cannot be found at all. This becomes even more important if not all models are developed from scratch but are also used from previous projects. In this case, which is the most common in industry, the requirements for which the previous model was developed need to be compared precisely to the requirements of

the current one. If there are differences in the requirements (even small ones) it needs to be identified how the model information changes in comparison to the current project. In this context, product lifecycle management (PLM) has to be mentioned, which aims to manage product related data over the lifecycle of a product. State-of-the-art product lifecycle management systems include functionalities that support model management tasks. Meta data of the models as described above (ID; creation date, creator, etc.) can be included and managed within a PLM system. However, not only the data in form of models has to be managed but also the related methods [17]. Today, PLM systems do not cover method management and do also not include all aspects that would be required to cover the whole product lifecycle [30].

Figure 3 shows an exemplary path of a method-model-landscape over the schematic V-model. It has to be noted, that the V-model represents a procedure and no process. That means that iterations in a project are not visible in the V-model. Therefore Fig. 3 also illustrates a possible path through the methods and the V-model including one iteration. The planning of which methods will be applied is at least as important as the method execution itself. In the figure it can be seen that a model may be input to multiple methods and that a method can develop multiple output models. The links represent the data & information flows between the methods and the models. In this approach the system model (output of the first method) is considered to support the selection of methods in different phases and system levels. The system model improves the accuracy of the first method path by providing an overview of relevant development aspects. The red dotted arrows represent the dependencies of the methods on the system model. This figure represents the first draft of a method selection. During the development process the system model is updated frequently and corresponding methods might be changed due to new information. Nevertheless, the first rough method selection on the basis of the system model provides a foundation for planning and executing further development steps. Applied on the shaft example mentioned before, the method after the system model could develop a first rough sketch of the shaft (which is the output model of this method). The rough sketch and the loads, which were calculated for critical areas with respect to their durability provide the basis for a following method—CAD (computer aided design). The output of the CAD method is a detailed structural model which can be built for hardware testing methods. After the CAD method it is possible to apply the previously discussed method FEM. As shown in Fig. 2 the load model and the geometry model need to be developed previously by suitable methods. With the developed output model concerning the elastic deformations and stresses, the hardware test methods can be planned accordingly. Following, the shaft is tested for its fatigue strength

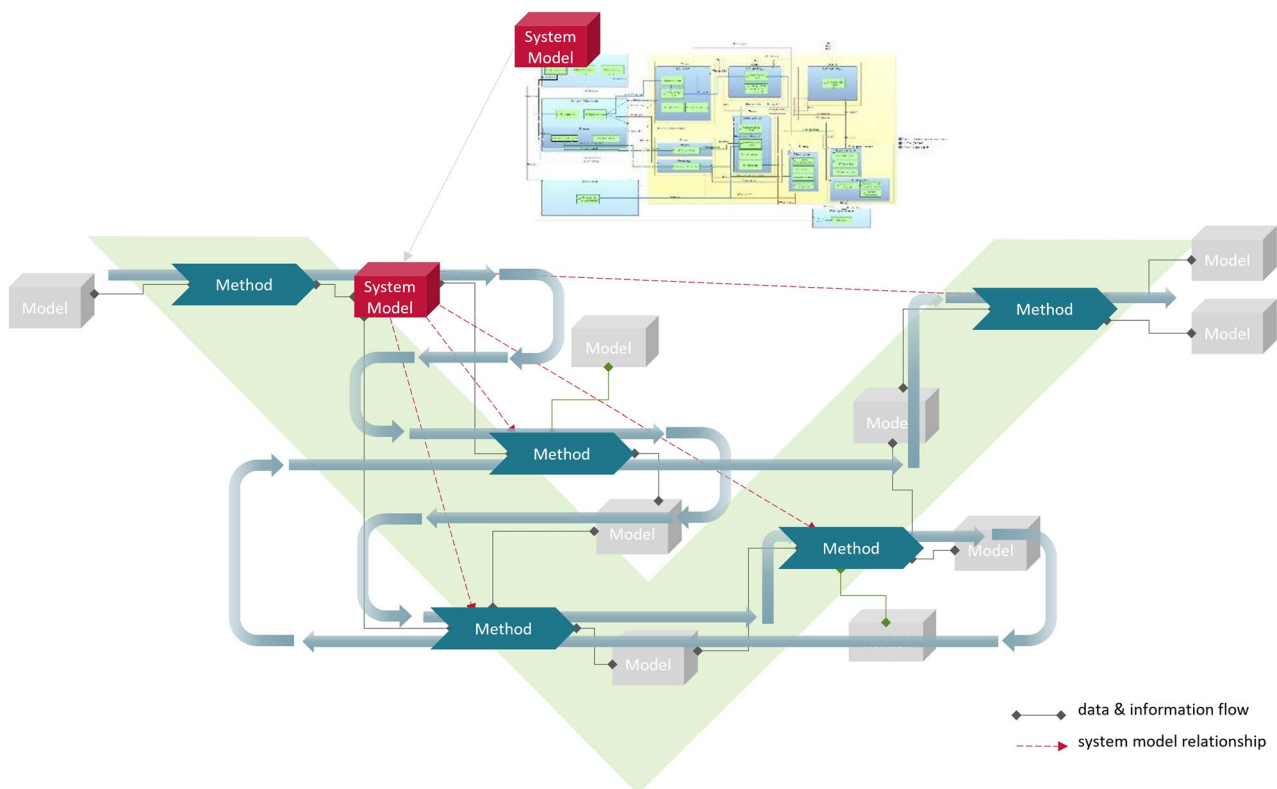


Fig. 3 Method—model—landscape over the v-model

on a test bench by an accelerated durability test. As can be seen in the figure, not only the methods on the left side of the v-model, but also the verification and validation methods are planned on the basis of the system model.

This in Fig. 3 exemplary shown method—model landscape is of critical importance in nowadays development of mechatronic systems. A major benefit compared to process modeling is the possibility of practical implementation on all company levels (on system level as well as on very specific activities of each discipline). On the one hand, an executive manager is able to steer the development by defining the actual methods and models developed starting from a very high level to deeper levels. On the other hand, even a very specialized engineer who works in depth with the FEM method for example, is supported by this approach since she or he is addressed on the actual level they are working on. One of the key deficits of modeling, namely the huge effort for the first development, is diminished by to the linkage of the models to the designated methods with respect to future projects. This stated linkage is the key factor of reusability in further projects and therefore justifies additional effort in the first development. Today, PLM systems do not offer possibilities to illustrate and model methods and its links to models. Many links between the system model and other models (especially discipline-specific models) are not supported by IT tools and also require defined methods include [31].

State-of-the-art PLM tools offer a wide variety of functionalities that turn PLM systems into a backbone of system development. Selected functionalities are described as well as the opportunities they provide in order to integrate methods and models more effectively in PLM systems of the future [30–33]:

- **Data Management:** This functionality includes meta data management of objects, file and artefacts. In order to efficiently document models and methods within a company, this functionality can be utilized to manage status, responsibilities, versioning, etc. of methods and models.
- **Workflow Management:** Similar to processes, defined methods require workflows when they are documented, revised or discarded. Therefore, workflow management within PLM systems (including tasks such as release management or definition of responsible employees) can be utilized to process such activities. Also, model generation and adaption workflows can be managed and tracked within the PLM system.
- **Variant Management:** As other development artefacts, models are generated in versions. As one version is an update to a already existing previous version, a variant represents a similar and derived model (e.g., a FEM model variant for a crankshaft of a inline 4 cylinder engine and a variant for a V6 crankshaft).
- **Configuration Management:** Different system configurations require different models to describe them but more importantly, they require a different set of methods to develop and especially verify and validate each configuration. Therefore, opportunities for optimization exist when connecting configurations of the system with configurations of method-model landscapes.

Innovative approaches include process modeling and modeling of scenarios PLM systems. This approach is heading in the right direction but does not cover the level of method modeling, which is described in this paper. Therefore, interfaces to PLM system have to be considered when further developing this fundamental concept into a solution for the industry.

Only the precise link of methods and models provides an understanding of the model origin and following the necessary adjustments of the model for future projects. To be competitive in future markets, this concept is one important aspect based on the benefits on all company levels and the efficiency improvement over several projects. The proposed methods-model landscape requires a new modeling language in order to provide usability for industry. Current modeling languages such as UML, SysML, BPMN, or other, do not represent a satisfying way of modeling methods and would need sophisticated extensions in order to support this task. A dedicated method-model modeling language needs to be developed, which requires further fundamental research as well as cooperation with academia and industry in order to establish an applicable solution.

## 5 Conclusion

This publication deals with the relevance of a combined view on methods and models in the development of systems on a generic basis. The approach enables a combined consideration of methods and models as a platform to support all phases of product development in terms of sustainable, effective and efficient development. Especially system models and their connection to methods in a system development approach require further research. On the one hand, fundamental concept to link method and models as the presented concept are required, on the other hand also system



modeling itself requires further investigation. System models provide essential information in order to define a set of methods and models for a development approach [22].

Without knowing which methods are applied, the considered model cannot be assessed regarding fidelity, maturity and level of detail. Only knowing which method-model combinations were used up to a certain point of development allows to evaluate model qualities at this point of time. In contrast, if these method-model links are neglected or not considered, the consequences can lead from inefficiencies, over worse system quality, to a developed system that does not fulfill its requirements.

From the system model, over the first sketch to the validation of the target system (system-of-interest), a set of methods is used, which is applied by people or teams with different roles, disciplines and system-specific knowledge. Depending on the development phase and system level, these interdisciplinary teams/experts must exchange information in order to identify common interfaces and to take them into account for further development activities. This exchange of information from one expert/team to another expert/team poses major challenges [21]. A non-structured approach in exchange with other experts/teams leads to risks in the integration phase and thus e.g. to fatal costs. The method-model connection presented in this publication is essential for a structured cooperation in order to maintain the overview of the models, to define the interfaces through the inputs and outputs of a method and to depict the model and method relationship.

A combined view on methods, models and related inter-dependencies serve as a basis to bring together the interdisciplinary field of developers at system level as well as component level and provide a framework for discussions between involved stakeholders. The mindset in line with the combined view on methods and models and its impact on the organization and the process can lead to significant added value. For example, gaps that arise with regard to the interdisciplinary interfaces in the organization (author of the information, availability of information etc.) and related risks can be identified more quickly in order to be able to intervene in the development process in short time. The visualization of the combined consideration of models and methods requires appropriate semantics, syntax and notation in order to achieve uniform added value. Therefore, it is necessary to enhance the presented generic concept with a modeling approach in form of diagrams. Implementing this concept into an IT-based tool, could empower development teams to coordinate and apply their tasks in an optimized way. Therefore, this is seen as a potentially powerful approach for multiple domains that deal with system development.

**Author contributions** All authors contributed to the study conception and design. Material preparation, literature review, figure preparation and concept development were performed by PK and CF. The first draft of the manuscript was written by all authors in cooperation and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

**Funding** Open access funding provided by Graz University of Technology. The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

**Data availability** There is no empirical data that can be accessed elsewhere. This research is purely conceptual.

## Declarations

**Competing interests** The authors have no relevant financial or non-financial interests to disclose.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Engeln W. Methoden der Produktentwicklung. In: Reiner D, editor. Strategisches Wissensmanagement in der Produktentwicklung. Wiesbaden: Deutscher Universitätsverlag; 2004. p. 20–30.
2. Pollack J, Helm J, Adler D. What is the iron triangle, and how has it changed? *Int J Manag Proj Bus.* 2018;11(2):527–47.
3. Van Wyngaard CJ, Pretorius JHC, Pretorius L. 2012. Theory of the triple constraint -A conceptual review. In: *ieee international conference on industrial engineering and engineering management.* 2019. pp. 1991–1997.

4. Gladysz B, Waldeier L, Jahn H, Albers A. Priorisierung von Funktionsumfängen zur Risikobeurteilung. *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb*. 2018;113(1–2):42–5.
5. Bytheway, A 2014, 'Investing in information technology - The Information Management Body of Knowledge', *J Property Finance*, Springer.
6. Hick H, Angel H-F, Kranabittl P, Wagner-Skacel J. Decision-making and the influence of the human factor. In: Hick H, Küpper HF, Klaus K, Sorger H, editors. *Systems engineering for automotive powertrain development*. Cham: Springer Nature; 2021. p. 355–80.
7. INCOSE Systems Engineering Vision 2020, INCOSE (2007) Document no: INCOSE-TP-2004-004-02; Version: 2.03
8. Kirchner E. 'Werkzeuge und Methoden der Produktentwicklung Werkzeuge und Methoden der Produktentwicklung. Kossiakoff. In: Sweet A, Seymour WN, Biemer SJ, editors. *SM 2011 systems engineering: principles and practice*. New Jersey: Wiley; 2020.
9. Orloff M A. Grundlagen der klassischen TRIZ : Ein praktisches Lehrbuch des erfinderischen Denkens für Ingenieure. 2006.
10. Opey L. *Entwicklungsmanagement - Methoden in der Produktentwicklung*. Berlin Heidelberg: Springer; 2005.
11. VDI 2223 2004, *Systematic embodiment design of technical products*.
12. Lindemann U. 'Methodische Entwicklung technischer Produkte', *Methodische Entwicklung technischer Produkte*. Springer, Berlin Heidelberg: Berlin Heidelberg; 2009.
13. Ehrlenspiel K, Meerkamm H. *Integrierte Produktentwicklung*. Konstruktion: Carl Hanser Verlag; 2017.
14. Weevers T. Method selection tool for user centered product development master thesis. 2011.
15. Stachowiak H. *Allgemeine Modelltheorie*. Allgemeine Modelltheorie: Springer; 1973.
16. Bajzek M, Fritz J, Hick H, Maletz M, Faustmann C, Stieglbauer G. Model based systems engineering concepts. In: Hick H, Küpper K, Sorger H, editors. *Systems engineering for automotive powertrain development*. Cham: Springer Nature; 2021.
17. Hick H, Bajzek M, Faustmann C. Definition of a system model for model-based development. *SN Appl Sci*. 2019. <https://doi.org/10.1007/s42452-019-1069-0>.
18. Paulweber M, Lebert K. *Mess- und Prüfstandtechnik*. Wiesbaden: Springer Vieweg; 2014.
19. Jokisch M, Rosendahl J. 'Klassifikation von Modellen. In: Bandow G, Holzmüller H, editors. *Das ist gar kein Modell!* Unterschiedliche Modelle und Modellierungen in Betriebswirtschaftslehre und Ingenieurwissenschaften. 1st ed. Wiesbaden: Gabler; 2010.
20. Friedenthal S, Moore A, Steiner R. *A Practical Guide to SysML: the systems modeling language*. 3rd ed. Morgan Kaufmann; 2014.
21. Bajzek M, Fritz J, Hick H. 'Systems engineering principles. In: Hick H, Küpper K, Sorger H, editors. *Systems engineering for automotive powertrain development*. Cham: Springer Nature; 2021.
22. Estefan J A. Survey of model-based systems engineering (MBSE) methodologies. 2008.
23. Bajzek M, Faustmann C, Krems D, Hick P, Kranabittl H. A concept for a digital thread based on the connection of system models and specific models. *INCOSE Int Symp*. 2021;31(1):1035–48. <https://doi.org/10.1002/j.2334-5837.2021.00885.x>.
24. Meboldt M. *Modellbildung in der Produktentstehung-als Beitrag zum integrierten Produktentstehungs-Modell (iPeM)*. Institut für Produktentwicklung, Universität Karlsruhe. 2008;29:259.
25. North K. *Wissensorientierte Unternehmensführung*. 7th ed. Wiesbaden: Springer Gabler; 2021.
26. Faustmann C, Kranabittl P, Bajzek M, Fritz J, Hick H, Sorger H. Future of systems engineering. In: Hick H, Küpper K, Sorger H, editors. *Systems engineering for automotive powertrain development*. Cham: Springer Nature; 2021. p. 855–82. [https://doi.org/10.1007/978-3-319-68847-3\\_32-1](https://doi.org/10.1007/978-3-319-68847-3_32-1).
27. Jacobs G, Konrad C, Berroth J, Zerwas T, Höpfner G, Spütz K. 'Function-oriented model-based product development. In: Krause D, Heyden E, editors. *Design methodology for future products data driven*. Cham: Agile and Flexible, Springer; 2022. p. 243–63.
28. Braun A. *Modellbasierte Unterstützung der Produktentwicklung*. Karlsruhe: Karlsruher Institut für; 2014.
29. Bajzek M, Fritz J, Hick H. Systems engineering processes. In: Hick H, Küpper K, Sorger H, editors. *Systems engineering for automotive powertrain development*. Berlin: Springer Nature; 2021. p. 1–35. [https://doi.org/10.1007/978-3-319-68847-3\\_9-1](https://doi.org/10.1007/978-3-319-68847-3_9-1).
30. Zamazal K, Denger A. Product lifecycle management in automotive industry. In: Hick H, Küpper K, Sorger H, editors. *Systems engineering for automotive powertrain development*. Berlin: Springer Nature; 2021. [https://doi.org/10.1007/978-3-319-68847-3\\_16-1](https://doi.org/10.1007/978-3-319-68847-3_16-1).
31. Hick H, Sanladerer S, Trautner J, Wilking F, Horber D, Ryan K, Piguet J, Faustmann C, Kranabittl P, Kollegger S, Bajzek M, Wartzack S. *Combining system models and CAD for change scenario management*. INCOSE Int Symp. 2023. <https://doi.org/10.1002/iis2.13012>.
32. Stark J. *Product lifecycle management Volume 1: 21st century paradigm for product realisation*. 3rd ed. New York: Springer International Publishing; 2015.
33. Eigner M, Stelzer R. *Product lifecycle management*. 2nd ed. New York: Springer-Verlag; 2009.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.