INCONSISTENCY MANAGEMENT FOR PRODUCT FAMILIES WITH MANY VARIANTS THROUGH A MODEL-BASED APPROACH IN MODULAR LIGHTWEIGHT DESIGN

M. Hanna ✉, J. Schwenke and D. Krause

Hamburg University of Technology, Germany

✉ michael.hanna@tuhh.de

Abstract

In methodical product development, numerous data are used and linked with each other, especially variant-related data. This paper presents a model-based solution for avoiding inconsistencies in the development of product families with many variants and extends it to modular lightweight design. In addition, the inconsistencies in methodical product development were classified and solution approaches were shown. Thus, inconsistencies can be avoided with the presented elaborated data model for an integrated product and process model based on the presented procedure.

Keywords: modularisation, lightweight design, systems engineering (SE), modelling, consistency management

1. Introduction

In methodical product development, numerous data are used and linked with each other. Particularly in methodical approaches for the development of modular product families, numerous visualization and analysis tools are available. In this special case, numerous models and elements are linked, especially variant-related data. This, however, can often lead to contradictions that are recognized late or never and complicate the development process and, in the worst case, impair the quality of the developed concepts. This aspect is particularly critical in interdisciplinary development procedures (Inkermann, 2019). Research work to date in this area has focused on the identification of inconsistencies or model-based procedures in general, as well as initial approaches to implementation in Model-Based Systems Engineering (MBSE). A continuous inconsistency management, which implements the variety context model-based, is missing so far.

An example of an interdisciplinary approach is modular lightweight design, in which both data of modularization and lightweight design interact with each other and have to be changed continuously. When applied to existing product structures in companies, historically grown product families with many variants complicate the data management additionally. Ongoing changes are often insufficiently documented, which is why only a few models of current development statuses are available, e.g. for FE modeling and the product structure. A uniform and explicitly documented development process are missing. Within the design process, FE models are to be standardized to limit their calculation and modeling effort. However, due to numerous product variants and different load cases, there are numerous FE models, which continue to increase with an increasing number of variants. This leads to considerable effort since for each newly developed variant a complete development process must be run through,

which is usually not clearly defined. Different persons involved with backgrounds from different disciplines make it even more difficult to maintain consistency. A uniform process optimized for lightweight design and modularisation should be supported by a consistent, holistic and digitized data management.

The following second section provides a brief overview of the main topics of modular lightweight design, inconsistency management and model-based development. In the third section, a classification scheme for inconsistencies will be presented, as well as initial solutions with a focus on model-based development. This is then applied to modular lightweight to show where inconsistencies are possible and to what extent a model-based approach can solve them. This paper concludes a discussion with an outlook.

2. State of the art

In this section, the methodological development of modular lightweight design, inconsistency management and model-based systems engineering will be discussed first.

2.1. Methodical development of modular lightweight design

Modular product structures are often used to reduce internal variety when external variety is high. According to Salvador, modularisation can be understood as a gradual property that can be described by the five gradual properties decoupling, interface standardization, commonality, combinability and functional binding (Salvador, 2007). Various methods for the development of modular product families are presented in the literature (Krause et al., 2014). These methods can be divided into two strategies: technical-functional modularization and product-strategic modularization. Design structure matrices (Pimmler and Eppinger, 1994) and structural complexity management MDM (Lindemann et al., 2009) are examples of modularization through technical-functional relationships. With the procedure of Stone, module formation is based on the flows within the functional structure. Three heuristics are determined: dominant flow, branching flow and conversion-transmission (Stone, 1997). Modular Function Deployment (Erixon, 1998) is an example of modularization according to strategic aspects. The Integrated PKT approach for developing modular product families combines these two strategies and contains numerous method units for reducing internal variety while leaving external variety unaffected.

Specialized visualization tools are offered, for example, the Module Interface Graph (MIG) is used for visualizing all components and their types of the product family, whereas the Tree of external Variety (TEV) for visualizing the variant product properties is relevant to the customers and the product variants offered. Resulting module wishes of the individual life phases are compared in the Module Process Chart (MPC). The data of the numerous visualization and analysis tools are continuously developed and supplemented in the sense of an iterative procedure. Data preparation is extensive, especially when considering an entire product family and not just individual variants. (Krause et al., 2014) A modular product structure can achieve advantages in all phases of life, but can sometimes lead to oversizing and weight gain due to the standardized interfaces (Krause et al., 2017). In the literature, a distinction is made between economical, functional and eco-lightweight design with regard to the objectives of lightweight design. In integral design, a combination of components results in a densification of functions with minimal interfaces. With the differential design method, in contrast, partial functions are realized by individual components and combined to form an overall function via interfaces. The integrating design method tries to combine the advantages of the integral and differential design method among other things by targeted positioning of individual interfaces. (Wiedemann, 2007)

In order to solve the contrast between lightweight design and modularization, modular lightweight design has so far been proposed as a first approach. With the sequential procedure in four phases, first a system model is created, a module division is revised, the module dimensioning is adapted and only then module-specific lightweight design measures are derived (Krause et al., 2017). In Gräßler and Yang's (2019) approach to the development of mechatronic systems and lightweight module design, also a sequential approach is proposed which, based on the V-model for the development of mechatronic systems, includes the steps of defining modular structure, revising module division and interfaces, dimensioning modules across variants and increasing the degree of lightweight module-specific design.

An approach of modular lightweight design, which considers both simultaneously and at an early stage, is the parallel approach. In the procedure for the development of modular lightweight design,

918

six steps are proposed within the framework of a process model, which is to be continuously linked with the product data model. Here the two core steps for the development of variant-oriented concepts for modular and lightweight design and their harmonization will be iteratively run through in order to skillfully combine the advantages of both design methods.

2.2. Inconsistency management

If a system has no contradictions, it is considered to be consistent. (Herzig, 2005). In their consistency management, Herzig et al. (2011) focus on the early identification of existing inconsistencies. Levels of inconsistency can be classified as internal or external inconsistencies. Herzig's types of inconsistencies are contradictions from modeling language, relationships between different models, beliefs and preferences of decision-makers, nature and its description. According to Finkelstein et al., (1992) the work of many people on a composite system (software, hardware, mechanics, ...) always leads to overlaps that must be coordinated and the overlaps are not easy to find due to the distributed way of working. A system created from physically separate teams uses a central database for communication. Consistency is made possible by strict access rights, version management and a common data model (Easterbrook et al., 1994). Continuousness can be divided into horizontal and vertical continuousness according to Albers and Lohmeyer (2012). With horizontal continuousness, the changes in a model are described by the procedure in the product life cycle, while vertical continuousness describes the change in the granularity of a model. In this context, consistency in models is described in such a way that they contain no contradictions when representing the same interests of systems. Baserati et al., (2018) show how inconsistencies in product-service systems can be systematically identified. They propose the following classification of inconsistencies: types, namely, existence, equivalency, refinement and satisfaction.

2.3. Model-based approaches for supporting product-development

For complicated products, a model-based approach promises better consistency management, which requires a model and consistency management. A model is only usable if it is consistent with the underlying formal system, the designer's views and preferences, and the laws of nature (Quamar and Paradeis, 2012).

A model is an abstraction of reality that has three main characteristics. Representation means that models represent something; Reduction indicates that not all attributes are represented in the model, only the relevant ones; Pragmatism means that models are not clearly assigned to their originals. A meta-model is itself a model that can be used to describe modeling. (Holt et al., 2012; Stachowiak, 1973)

In product development the models can be divided into those that represent the process and those that represent the products. Product models are used to represent the emerging product and its environment, focusing on different aspects such as requirements, behavior, structure and parameters. Process models are generated to design, communicate, plan and monitor the process at different levels of abstraction. These models are therefore used to express, analyze and communicate the design itself. (Eckert et al., 2017) Inkermann expands the idea of supporting tools in Model-based Process Engineering (MBPE). In the associated framework, the system of processes includes processes and activities at different levels of abstraction, the system of product models includes all models used to represent the emerging product at different stages of development with respect to different views and development areas. The system of tools represents the variety of tools used in the product development process and their required compatibility caused by the interaction between activities (Inkermann, 2019). For collaborative design an explanation model is proposed which includes the three views process, personal and technical-methodical. (Bavendiek et al., 2018)

For a model-based support of product development, the Model-Based Systems Engineering (MBSE) was developed using abstract system models (Holt et al., 2012). The language SysML was developed for MBSE. The modeling software Cameo Systems Modeler uses SysML notation as well as diagrams and tables. It is based on nine diagrams, five of which are used to create the structure and four to create the behavior (Weilkins, 2008; Holt et al., 2012).

Albers et al show how the MBSE approach can be used to model the complex real problem of vehicle kits. The goal is to investigate the potentials in this context, how modular kits and products can be modeled efficiently, and finally, how MBSE can support modular design (Albers et al., 2019). A data

model for the development of modular product families based on the integrated PKT-approach was also developed and implemented in MBSE. For this purpose, the methodological tools considered were linked and both hierarchical and version-related aspects were considered in order to develop a consistent data model (Hanna et al., 2018a).

3. Classification of inconsistencies and how to avoid them

In this section, the inconsistencies are described and a classification and initial solutions are proposed. The described methodical approach focuses on a model-based approach.

3.1. Motivation of inconsistency management

Inconsistencies can occur when different models are used. These inconsistencies are often a source of contradictions in models caused by a lack of inconsistency management and not explicitly documented knowledge. Especially in methodical product development, where many unconnected visualization tools are used, inconsistency problems often occur. This methodical approach encounters a company's structure that works a lot with individual models that are not networked, as well as individual documents and implicit knowledge of acting persons. Due to the development of modular product families to reduce the internal variance, not only existing support tools need the tools existing in the company, but also data and documents of different variants have to be considered. This is only insufficiently considered with past inconsistency analyses and also model-based solution suggestions and is to be focused particularly in this work. In terms of product generation development, the versioning of data and models is also relevant in this context (Albers et al., 2015). Often different versions of a product variant are available at the same time and it is not clearly determined which one is currently valid. Also, by methods, which recommend an iterative procedure, different states are defined at the same time, which are continuously changed. In interdisciplinary development projects, cross-domain data is used, for example in mechatronics, in which not only mechanical but also electrical and informatic information is required to be networked. (Inkermann, 2009)
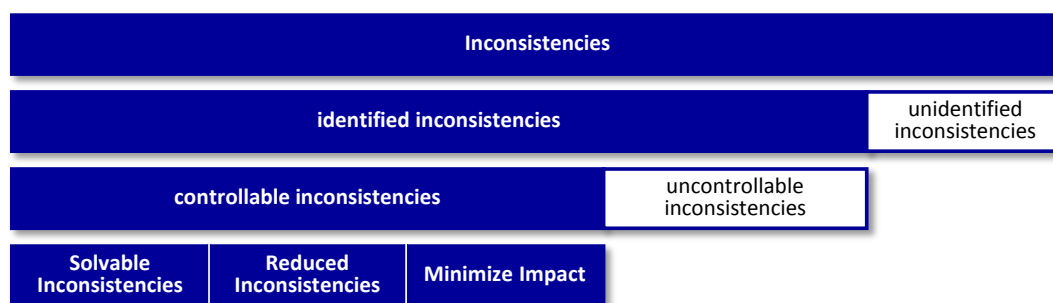

Figure 1. Classification of inconsistencies

In this context, all inconsistencies within the development of a system are divided into inconsistencies that can be identified and those that are not. The identified inconsistencies can be divided into controllable and uncontrollable inconsistencies. The controllable inconsistencies are divided into solvable, reduced and those with little impact.

3.2. Inconsistencies and possible solutions in methodical product development

In methodical product development, methodical visualization tools meet data of a company structure, which in turn are supposed to represent the real products. In addition, they will also be inconsistent due to the user of the data and the interaction of the individual users. Therefore, different inconsistencies will be described here and basic solutions will be described. Inconsistencies between different models can often occur if they contain the same model elements but contradict each other in the different models. As a special form of the inconsistency source between different models, the inconsistency between different variants becomes relevant during the development of product families, which is why it should also be considered in this case. Also, the inconsistency that can occur due to different versions of data. Contradictions can also occur within a model if individual data in a model are to describe the same

element, but are called different, for example. There is also a data transfer between the data within a company and the data of the research-side methodical product development. On the one hand, the data of the company is transferred to research when a company situation with its product strategic and technical situation is recorded, on the other hand, the data is also transferred to research when developed concepts are to be implemented in company models. This is therefore also a source of inconsistencies. These internal inconsistent company data can now also be external inconsistencies to the real system. By applying the methods, the user can also make an inconsistency possible if he or she makes an application in contrast to the formulated modeling. Inconsistencies can also occur between persons in the product development process if, for example, there is a different understanding of a situation.

There are several ways to control these assumed inconsistency types. For example, the inconsistency source within models can be solved by a consistent use of existing data that is stored centrally. The inconsistency source between models is solvable by using a product data model. The inconsistency between different variants can be resolved by using only one model for the entire product family, whereas the inconsistency between different versions can be resolved by using a process model. The source of inconsistency between methods and enterprise data can be reduced by using an interface between enterprises and methods. Inconsistencies between the real system and the enterprise data can be minimized by performing a consistency adjustment. The inconsistency between the involved persons can be reduced if a common understanding improves communication through a workshop-based approach. Inconsistencies caused by the application can be reduced if a user interface is used with a software solution.

3.3. Methodical approach for solving inconsistencies

Many of the sources of inconsistency can be avoided by a consistent data model as shown in section 3.2. A model-based approach is particularly helpful for inconsistencies between models and within models. In contrast to the procedure in the literature, however, the variance reference should be taken into account. The application can be simplified by a user interface and the resulting inconsistencies can be mitigated, which is why a software solution using MBSE is a good solution. At the product model level, all data, models and documents should be linked to each other to ensure consistency. The variance-induced sources of inconsistency are particularly in focus. The version-related inconsistencies can also reduce the inconsistency through a continuous process, which is why, as already suggested in the literature, a continuous process model linked to the product model is also used here. This enables cross-process data exchange. If possible, inconsistencies should be prevented and a strategy to uncover them can lead to better consistency management.

To this end, the sources of inconsistency that occur should in the first step be identified and then classified using the classification system proposed in Section 3.2. In the second step, these sources of inconsistency should then be analyzed to determine to what extent they can be resolved and where a model-based approach can avoid these inconsistencies. In the third step, all product data should be recorded and their modeling implemented within a meta-model for the inconsistency sources to be solved and should be linked by means of a continuous process model. The last step is the implementation of the concrete data within a modeling language such as SysML MBSE-based.

4. Using model-based consistency management for modular lightweight design

This section shows how the approach presented in section 3 can be applied to the aircraft galley using modular lightweight design as an example.

4.1. Presentation of the application case

Aircraft cabin galleys are installed in different positions in the aircraft. They are characterized by the fact that they can have a high variance since the airlines have different configurations available in the aircraft cabin in order to provide individualized catering according to their requirements. At this point, a modular product structure would be appropriate to be able to offer the high external variety with a reduced internal variety and thus a more favorable production process. In addition, all components in the aircraft should be as weight-efficient as possible to reduce fuel consumption. As mentioned above, the two goals of a

modular and weight-saving product structure are contradictory, making the aircraft cabin a good application for modular lightweight design. However, the information and data for aircraft cabin dismantling are often not always explicit, since the product structure has grown historically, aircraft cabin monuments have long development times due to their long service life, and generally high requirements apply in aviation. From the point of view of lightweight design, an integral design with as few interfaces and high functional integration as possible leads to a weight reduction. Furthermore, the use of as light a material as possible, for example, composite material and a suitable design like the used sandwich-structured composite is advantageous, as well as a load-path-optimized design. A modular design can be used to achieve product strategic advantages. For production, the modular design is ideal for taking the variety of variants into account in order to produce product variants that are as individualized and suitable for production as possible and to simplify interchangeability with MRO. These opposing designs cannot be implemented in a common design, but a suitable compromise must be found. Therefore, a hybrid design, which combines several of these designs, is the best solution.

This results in numerous inconsistent sources of different data. Therefore, it is an ideal example for demonstrating the potential of a model-based avoidance of inconsistencies for modular lightweight design.

4.2. Inconsistencies that occur and how they can be resolved

When considering the situation in which both methodical product development and methodical lightweight design encounter the existing company structure and have to interact with each other within the framework of modular lightweight design, the inconsistencies described in Section 3 can be identified at various points. Figure 2 shows examples of the data, documents and models used and their possible sources of inconsistency.

Identification and classification of inconsistencies	Analysis of inconsistencies, whether they can be avoided	Development of a metamodel for product and process structure	Implementation with a modelling language

Figure 2. Approach for reducing inconsistencies

Starting on the left research side, modularization methods are often not always free of sources of inconsistency. The PKT approach for developing modular product families (Krause et al., 2014) used here as an example for the development of modular product families uses numerous visualization tools (see section 2.1) that are not linked to each other, but partly contain the same elements. For example, MIG displays components that are also used for modularization using the MPC. An adjustment of the component types, however, does not take place, which can lead to errors, especially with changes. The version management is not given, therefore 2 versions of the MIGs and the MPCs are exemplarily represented in the picture, between which an inconsistency source is present, as they occur frequently with iterative development processes. On the other hand, variants are considered in the approach, which is shown by the fact that all variants of an entire product family are represented within the tools (such as the MIG). In this respect, inconsistencies between variants in the methodical approach do not occur within the models. Since the same data appears several times in the model MPC with the components, the source of inconsistency within a model prevails here. For methodical lightweight product development, numerical optimizations and working with functional structures are examples that may be inconsistent with each other but also do not have to be consistent with the respective versions and variants. The data consistency between the methodical models is also not made possible, which is why inconsistency sources between models can also occur here. Especially if different design concepts are to be developed during the development of modular lightweight design, these must be harmonized, which has not yet been done for different design methods. The modularization expert and the lightweight design expert must work together on the same components or modules in different models such as the MIG or numerical optimization within the framework of the parallel procedure of modular lightweight design, which enables inconsistencies due to their application as well as interpersonal inconsistencies due to different understandings.

Producing companies of aircraft cabin monuments such as galleys have inconsistencies mainly due to outdated versions of model data due to the historically grown product structure. For example, parallel

model versions may exist due to the subsequent 3D design of models that were formerly available in 2D. Inconsistencies can also occur between different departments that are relevant for modular lightweight design. An example is a CAD-model and an FE-model for an aircraft cabin. Existing identical functions can be solved differently. A clear assignment of requirements via functions to partial solution concepts is missing. The existing partial solution concepts can thus be inconsistent to each other. Inconsistency in the company is also due to the fact that the data basis is incomplete since the aircraft cabin development and its documentation originate from a time in which digitization has not yet advanced far.

4.3. Development of a data model for modular lightweight

As a solution to the essential inconsistency problems, a product data model is a good choice for modular lightweight design, which links both the models and data of modularization and lightweight design with one another and thus provides a redundancy-free and consistent basis for the development of a modular lightweight design strategy. On the other hand, a process model is needed due to the missing process situation and the parallel and iterative approach of modular lightweight design. Particularly missing defined development process steps as well as consistency problems caused by ambiguous versioning can thus be mastered. Some of the inconsistency problems described above are addressed with the product data model in Figure 3, which is based on the data model of the PKT approach (Hanna et al., 2018b).
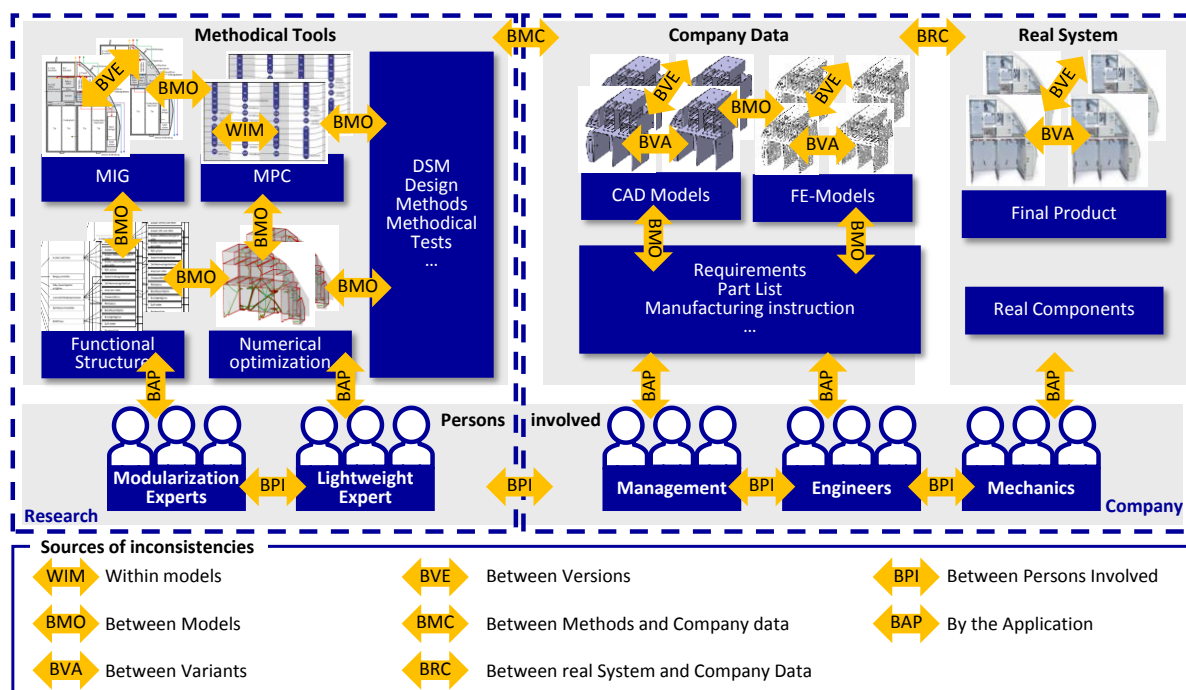

Figure 3. Sources of inconsistencies in the application of modular lightweight design

Data elements used in modularization are colored blue, whereas data of lightweight design are colored white. Modular development and lightweight development should use a product family functional structure (PFS) as the central functional structure. In particular, the design relevant for modular lightweight design must link the components with the functions as well as with the material properties of the components. Requirements are linked to the certification and variant costumer properties.

In the process model of modular lightweight, the first step is a detailed requirements analysis in order to consider the requirements of all relevant stakeholders from the perspective of lightweight design and modularization. For the second step, the analysis of the actual state of the product structure, design data such as CAD models are taken into account, as well as when using the integrated PKT approach for the development of modular product families of the Module Interface Graph are identified. With regard to lightweight design, existing design processes and existing data, for example from FE models, are analyzed. On this basis, the development of the product structure is started, whereby concepts for modular and lightweight design are developed iteratively. In the sense of modular product structuring, concepts are

developed for a modular design method on the one hand, and for a lightweight design method on the other, and existing data adapted. Subsequently, harmonization with the modular lightweight design method takes place, in which the module and lightweight design concepts are combined. In an iterative approach, the concepts are aligned in order to achieve optimum gradually between modularisation and lightweight design. In the modular lightweight design, for example, individual modules can be designed in different lightweight designs, which leads to a hybrid design. The resulting concept is consistently incorporated into modularization and lightweight design in the product data model.

4.4. Proposed implementation in cameo

The implementation by means of the Cameo-System Modeler can be done by a process model modeled with activity diagrams as shown in Figure 4. The product models contained therein can be modeled by internal block diagrams, like the MIG. The arrows show the directions of the process, while the product models are linked to the corresponding process steps in the lower-left corner. In this example, the focus is not on the linking and transfer of various company data such as CAD or FEM, but rather on the methodical model-based linking of lightweight design and modularisation methods.
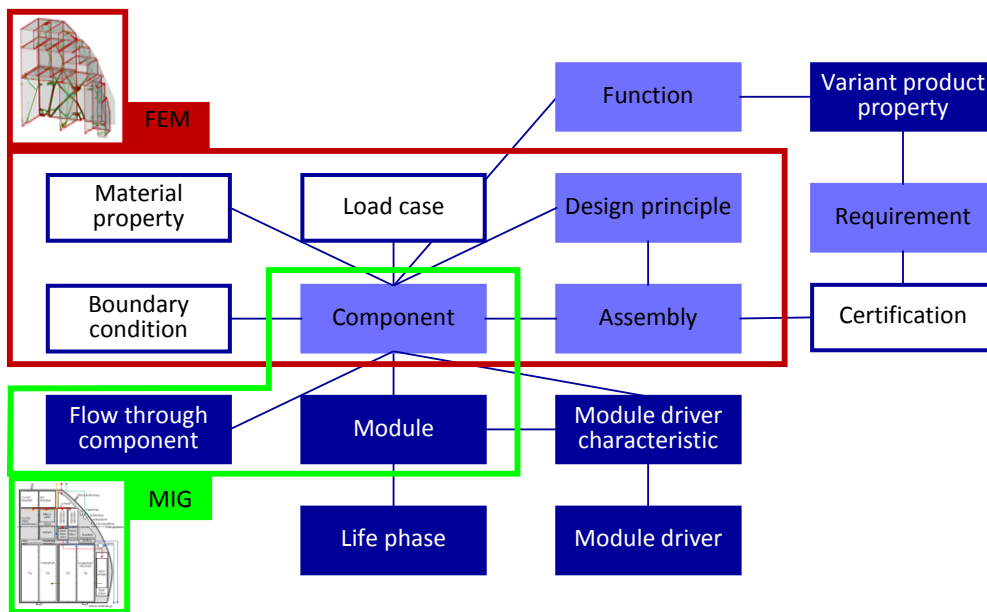

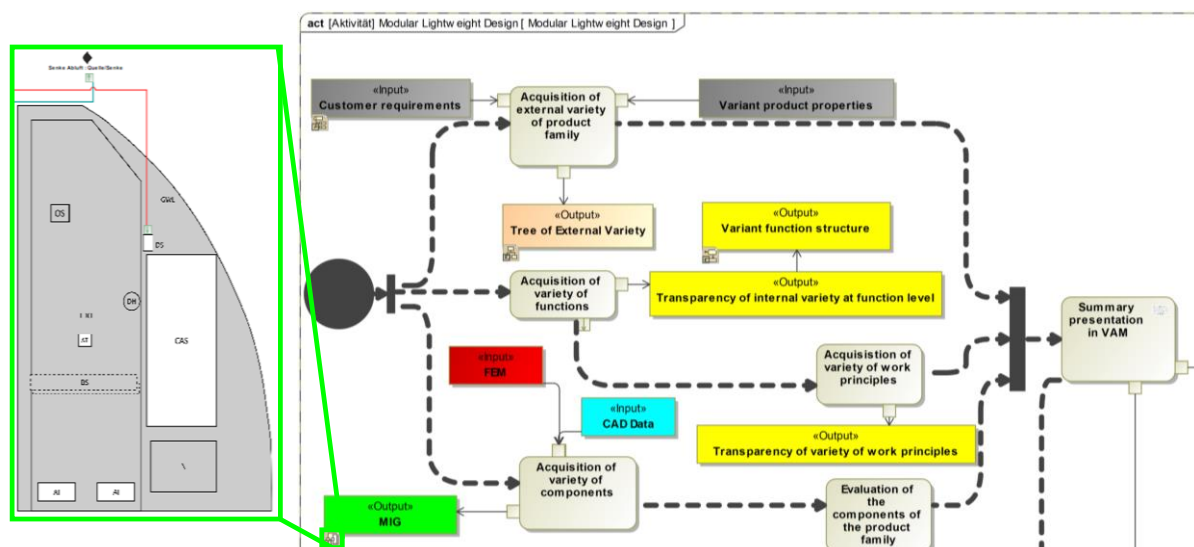**Figure 4. Detail of a data model of modular lightweight design**

Figure 5. Section of a possible implementation in SysML of modular lightweight design

On the right side, there is a section of the design process and on the left side, there is a product model of the MIG for an aircraft cabin monument. The gray-shaded model elements indicate the interface to data and models that can be integrated into the data model.

5. Discussion and outlook

This paper presents a model-based solution for avoiding inconsistencies in the development of product families with many variants and extends it to modular lightweight design. For the first time, the connection between the topics consistency management, Model-Based Systems Engineering, variant management, modularization and lightweight design is demonstrated. In addition, the inconsistencies in methodical product development were classified and solution approaches were shown. Thus, inconsistencies can be mastered with the presented elaborated data model for an integrated product and process model based on the presented approach. This makes it easier to minimize inconsistencies. Thus, it is now also possible to resolve the contrast between modularization and lightweight design within the framework of the parallel development of modular lightweight design for an entire product family and to provide the basis for software support. The exemplary application to an aircraft galley has already shown the advantages of an MBSE-based implementation of a part of the data model.

However, the interfaces to company data have not yet been implemented at this point and require further research in order to further increase the consistency between methodical product development and industry. In addition, the further development of modular lightweight design concepts should be further detailed. The data model should be extended by both. The application to the concrete application case should also be further detailed in the future and the effectiveness of the model-based approach should thus be further improved. In order to further focus on the sustainability of the lightweight modular system, the sustainability process should be taken into account (Bahns et al., 2015). These should then be linked with consistency management, model-based product development, variant management, lightweight design and software support in order to be able to apply it efficiently to the modular lightweight design.

References

Albers, A. et al. (2019), "Model-based systems engineering in modular design", *Design Science Journal*, Published by Cambridge University Press 2019.

Albers, A., Bursac, N. and Wintergerst, E., (2015), "Product generation development - importance and challenges from a design research perspective". In: *Proceedings of INASE Conferences 2015*, INASE, Wien, Austria.

Albers, A. and Lohmeyer, Q. (2012), "Advanced Systems Engineering – Towards A Model-Based and Human-Centered Methodology". In: *Proceedings of the Symposium on Tools and Methods of Competitive Engineering (TMCE)*, Karlsruhe, 2012.

Bahns, T. et al. (2015), "Sustainability of Modular Product Families", *20th International Conference on Engineering Design*, ICED15, Mailand.

Basirati, M.R. et al. (2018), "Towards systematic inconsistency identification for product service systems", *Proceedings of the DESIGN 2018 / 15th International Design Conference*, Dubrovnik, Croatia, May 21-24, 2018, The Design Society, Glasgow

Bavendiek, A., Inkermann, D. and Vietor, T., (2018), "Interrelations between processes, methdos, and tools in collaborative design - a framework", *21. International Conference of engineering Design*, ICED17, 2017 Vancouver.

Easterbrook, S. et al. (1994), "Coordinating Distributed ViewPoints: the anatomy of a consistency check", September 1994 *Concurrent Engineering Research and Applications,* Vol. 2 No. 3, pp. 209-222.

Eckert, C.M. et al. (2017), "On the integration of product and process models in engineering design", *Design Science*, Vol. 3 No. 3. http://doi.org/10.1017/dsj.2017.2

Erixon G. (1998), "Modular function deployment: a method for product modularization", Dissertation, The Royal Institute of Technology, Stockholm.

Finkelstein, A. et al. (1992), "Viewpoints: A Framework for Integrating Multiple Perspectives in System Development" *International Journal of Software Engineering and Knowledge Engineering,* Vol. 2 No. 1, pp. 31-58, March 1992, World Scientific Publishing Co.

Gräßler I., Yang X. (2019), "Entwicklung mechatronischer Systeme und Modulleichtbau: Anforderungen am Beispiel von Unmanned Aerial Vehicles (UAVs)", March 2019 http://doi.org/10.17619/UNIPB/1-774 Conference: Fachtagung Mechatronik 2019 At: Paderborn

Hanna, M., Schwede, L.-N. and Krause, D. (2018a), "Model-Based Consistency for Design for Variety and Modularization", *20th International DSM Conference*, DSM 2018, Trieste, Italy, 2018, S. 239-248.

Hanna, M., Schwenke J. and Krause, D. (2018b), "Modularer Leichtbau – Chancen und Herausforderungen im digitalisierten Entwicklungsprozess" *30. DfX-Symposium*, Hamburg 2019.

Herzig S. (2005), "A Bayesian Learning Approach to Inconsistency Identification in Model-Based Systems Engineering", Dissertation.

Herzig, S. et al. (2011), "A Conceptual Framework for Consistency Management in Model-Based Systems Engineering", In: *Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE*, 2011 August 29-31, Washington, DC, USA.

Holt, J., Perry, S. and Brownsword, M. (2012), "Model-based requirements engineering", The Institution of Engineering and Technology, London.

Inkermann, D. (2019), "Towards model-based process engineering", *International Conference of Engineering Design*, ICED 19, Delft.

Krause, D. et al. (2014), "Integrated Development of Modular Product Families: A Methods Toolkit", In: Simpson, T.W., Jiao, J.R., Siddique, Z. and Hölttä-Otto, K., *Advances in Product Family and Product Platform Design*, Springer, New York, pp. 245-270.

Krause, D. et al. (2017), "New Trends in the Design Methodology of Modularization", *11th International Workshop on Integrated Design Engineering*, IDE Workshop, 5-7. April 2017, Magdeburg.

Lindemann U., Maurer M. and Braun T. (2009), "Structural complexity management: an approach for the field of product design", Springer, Berlin.

Pimmler T. and Eppinger S. (1994), "Integration analysis of product decompositions", In: *Proceedings of the 6th design theory and methodology conference*, New York, pp. 343-351.

Quamar, A. and Paradeis C., (2012), "Dependency Modelling And Modell management in Mechatronic Design", *Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE 2012*, August 12-15, 2012, Chicago, IL, USA.

Salvador F. (2007), "Towards a Product System Modularity Construct: Literature Review and Reconceptualization", In: *IEEE Transactions on Engineering Management*, 2007, 54; S. 219-240.

Stachowiak, H. (1973), "Allgemeine Modelltheorie", Springer, New York.

Stone, R. B. (1997), "Towards a Theory of Modular Design", University of Texas, Austin.

Weilkins, T. (2008), "Systems Engineering with SysML/UML: Modeling, Analysis, Design", Morgan Kaufmann.

Wiedemann, J. (2007), "Leichtbau. Elemente und Konstruktion", Springer Verlag, Berlin, 3. Auflage, 2007.