21st INTERNATIONAL DEPENDENCY AND STRUCTURE MODELING CONFERENCE, DSM 2019
MONTEREY, CA, USA, 23 – 25 September, 2019

From Visualizations to Matrices – Methodical support for New Development of Modular Product Families

Jan Küchenhof, Lea-Nadine Schwede, Michael Hanna, Dieter Krause
Hamburg University of Technology

Abstract: Modular product families have shown to be an effective way to handle product variety and enable economies of scale e.g. by commonality as well as economies of scope regarding lead time reduction or strategic flexibility. To benefit from these advantages right from the start, support for the new development of modular product families is tried to achieve by applying adjusted design for variety and technical-functional modularization methods. A new visual product model helps handling the emerging components and interfaces. By implementing the early stage product model in SysML, data consistency and traceability can be reached and a basis for optimization methods and further network analysis is created.

Keywords: Design for Variety, New Product Development, Component Interface Interaction, Design Structure Matrix, Model-Based Systems Engineering

1 Introduction

Diverse market requirements induced by different customer needs can lead to high product variety if not tackled appropriately. Development of modular product families with support of Design for Variety (DfV) is a possible solution to offering a high external product variety with relatively low internal product and process variety. The application of modularization approaches is mostly carried out on existing product families where the database is rich and can be used for further structuring and optimization after system analysis. (Krause and Gebhardt, 2018) We focus on new development of modular product families starting with concept creation which means that not only the knowledge about the product and its structure is low but also organizational and process structures are not well defined. Research often shows single case studies that mostly report successful attempts but miss some challenges manufacturing firms may face during product family development such as the balance of product integrity and component standardization and the management of the numerous emerging interdependencies (Sundgren, 1999).

In the following we present a possible way to generate the necessary product development models supporting the modular product architecture design as a first step within product family creation. As those product family models can be understood as dependency visualizations in network form, we then show the implementation in SysML as a database to reach consistency and traceability of the system elements as well as creating a foundation for further network analysis and optimization of the product architecture.
Methods for variant management and modularization as well as dependency visualization in networks and matrices will be presented. A methodical approach to develop early phase

interface support tools is shown which contains components and their linkage. Those are then implemented in SysML using the Cameo Systems Modeler (CSM).

2 Methods for Product Development and System Dependency Handling

In this section methods for product development, DfV and modularization are presented. The second part consists of the description of complex networks in visual and matrix representation. Finally, we give a short introduction to Model-Based Systems Engineering (MBSE) which will later be used to store the accruing data.

2.1 Design for Variety and Modularization in New Product Development

A classical generic product development process consisting of the six phases planning, concept development, system-level design, detail design, testing and refinement and production ramp-up is given by Ulrich and Eppinger (Ulrich and Eppinger 2012). Being in the early two phases strategic decisions have to be made. “The traditional paradigm for product competition and manufacturing relied on minimizing variety and change to achieve economies of scale, low cost, and quality” (Worren et al., 2002). To shorten New Product Development (NPD) lead times and to achieve not only economies of scale through commonality but also economies of scope (e.g. through customization, incremental innovation, flexibility and fast customer responsiveness just to name a few), modularity as an NPD strategy has been more and more focused. (Mikkola and Gassmann, 2003, Mikkola, 2006).

In order to extend the Integrated PKT-approach for Developing Modular Product Families in terms of NPD, DfV by Kipp as a core part of the approach finds consideration. It has proven helpful in many projects with the goal to reduce internal product variety and process complexity maintaining the external offer variety. The design method provides a variety-optimized product structure and prepares the following product-strategic modularization approach of Life-Phase-Modularization (LPM). LPM aims to harmonize the development activities between the different product life cycles such as product development, assembly and sales (Krause and Gebhardt, 2018). The correlation between external and internal variety can be analyzed and optimized with help of the Variety Allocation Model (VAM). The VAM consists of four levels, mostly supported by partial models. The external variety is analyzed and recorded in the Tree of External Variety (TEV) reflecting the customer relevant, differentiating product features on the first level. After structuring the external variety and clarifying the task, a functional structure is derived. The Product Family Function Structure (PFFS) is a flow-based model that supplements aspects of functional variety. The third level represents the active principles and active geometries. Solution finding is usually carried out with morphology, a model to represent the active structure regarding product variety has not been developed yet. To identify component variety on the fourth level, the Module Interface Graph (MIG) is developed by sketching the outlines of the chosen system and visualizing the components, modules and interfaces for each. Different colors and line types represent different interfaces and component variety. Connecting the elements on each level top down the relation between customer relevant product features can be mapped to components via functions and active principles trying to accomplish a 1:1 mapping of distinguishing characteristics to variant components (among other criteria) (Kipp, 2012)

A possibility to form modules based on flows within a functional structure such as the PFFS is given by Stone and will be used later on. The application of his heuristics helps standardizing sets of functions and flows and supports the functional decomposition which he describes as the most important step in the design process. Functional module building results from the application of the three heuristics "dominant flow", "branching flows" and "conversion-transmission". (Stone, 1997) On component level, the Integration Analysis Methodology based on Design Structure Matrices (DSM) can be used to build modules with the premise that components are coupled via information, energy and material transfer as well as spatial dependencies (Steward, 1981).

2.2 Complex systems – Dependency visualization in matrices and networks

Complexity in general describes the number, diversity and links of system elements as well as their states and dynamics. A system will be called complex if it is not-deterministic, uncertain and timely variable. Product families fit this description as the number and diversity of variant components and accordingly products and processes increase over time with product variety induced by new or changing customer requirements. To describe a complex system the number of different elements, the number of different links and the direction of links can be included as they make the problem quantifiable. (Krause and Gebhardt, 2018).

Visual support has been evaluated useful for complex problem solving. It is suited better for interdisciplinary team communication than lists, tables or matrices as combined information can be displayed (Gebhardt et al., 2014). Graphical support for dependency handling and analyzing coupled blocks in DSM by creating shunt diagrams is shown in by Steward and Smith (Steward, 1981, Smith, 1992).

Possible ways to systematically analyze complex systems are both, network and matrix views that can portray single systems such as the product structure in a DSM or mapping multiple domains as in a House of Quality (HoQ) with implied DSM and Domain Mapping Matrices (DMM). Networks can be viewed with the help of graph theory, resulting in visualizations providing a basis for system constellations and analysis of certain characteristics. The network consists of the different elements and its connections. (Lindemann et al., 2009) The elements can also be represented in visualization equivalents such as activity networks and DSM equivalents in matrix form. Basic activities such as serial, parallel and coupled processes can be shown in both representations as described by Smith (Smith, 1992) and Browning (Browning, 1998). The DSM represents the product system architecture in terms of its components and their relationships. The adjacency matrix shows system elements such as components as entries in the rows and columns and the relationship such as interfaces among them by matrix entries. There are different tools to handle matrices like DSMs. They can be presented in form of excel tables or other software (e.g. Loomeo, Boxarr, Lattix). (Hölttä-Otto et al., 2018)

2.3 MBSE for consistent data modeling

MBSE can also be used for supporting the development process of complex systems. A system in this context consists of a system architecture, system requirements and system behavior. System elements and their dependencies can be represented and linked in the modeling language SysML (Weilkiens, 2008). The CSM can be used as the software environment for this purpose (Holt, 2012).
MBSE has already been used in a number of ways in the product development process. Hanna introduced a consistent data model for the Integrated PKT-approach which also includes a representation of a DSM on component level. The data model considers three types of consistency needed to develop modular product families. Consistency of time allows an extension of the product family. The vertical consistency ensures that partial models are consistent across different hierarchical levels. Similarly, consistency between multiple models is enabled by having each model element only once. (Hanna et al., 2018) Bahns et al. used the model-based approach on modular product development concentrated on variant management (Bahns et al., 2015). Bursac introduces the consistent data management during iterative development cycles based on MBSE as a support for generational development. (Bursac, 2016)

3 Methodical Product Family Development and System Dependency Handling

The methodical approach chosen to initially develop a modular product family consists of the basic steps of Design for Variety adapted for NPD with early modularization on a functional level. The focus is on the identification of component interaction in visual models and matrices. The handling of future adaptions and system analyses is tried to accomplish by a consistent tracking of element variety by matching colors and flow definition during the implementing with SysML.

3.1 Challenges – Support in the early phase and consistent data tracking

The planning of whole product families, which will later consist of different variants instead of single products, poses a special challenge. Future uncertainties like changing legislation, new applications or arising technologies need to be considered to reduce internal complexity sustainably. New development of modular product families is characterized by a high risk for firms as it costs more resources than single product development and problems during development can affect the whole product family. Nonetheless companies may face innovation pressure that sometimes requires green-field development to seize new market segments. The beginning of the development activities is crucial as following work steps and costs will be derived from the selected concepts in this early phase. Processes such as development activities, supply-chain and assembly sequences can be derived and benefit from the modular product architecture in terms of standardization and flexibility (Sanchez, 1996).

Two questions emerge that will be dealt with. Firstly, how can new product family planning regarding product variety be methodically supported? And further how can the created data be efficiently stored and made available for further development activities. Those questions are approached by examining the interfaces in the product structure and between visual and software models. A development procedure was derived and carried out using a use case where visual models as well as software tools are combined. The connecting interface between both worlds is the newly introduced tool for component-interface definitions.

3.2 Initial Product Family Planning - Use-Case

Kipp proposes that higher levels in the VAM require more rework as new product functionality is introduced to extend the sales offer by implementing new product features (Kipp, 2012). It is only natural that new development of product families with initial feature realization can be achieved by following the design method top-down as development starts with identification of customer requirements. Often just a component redesign takes place as it is cheaper and easier than introducing different working principles or new features.

The aim of the use case is to simulate NPD activities and build a product architecture from which a modular product structure and a prototype can be derived and designed. The task is to build a modular product family of robots for further research. Starting with the task description, requirements are derived. Creativity techniques such as brainstorming can be applied to open the solution space. Four different applications are selected and requirement lists are created for each. Variation in requirements (energy storage, temperature resistance, regional differences, environmental influences) is analyzed and transferred in the TEV which represents the first model of the VAM showing the functionality required in formulation of the customer neglecting the technical realization at this point. For prototype development one application is chosen and will be defined by combining the different product features to one consistent solution for which a concept creation will be performed on. Next, the flow-oriented PFFS is drafted. The solution-neutral functional structure is suitable for application of Stone's heuristics to develop modular structures on functional level. Variant functions are marked grey and optional ones dashed to implement the view of variety. On basis of the functional modules, solution finding can be carried out as proposed in the VDI 2221 (VDI 2221, 1993). Discursive and intuitive methods help finding technical alternatives which can be stored in mind-maps or morphological boxes. After gathering enough possibilities to technically realize functional modules or single functions, alternative concepts should be created following different design rules (e.g. cheap & simple, innovative, best-fit) the concepts are sketched and evaluated with help of a technical-economic analysis. Often a combination of single solutions of each concept can lead to an optimal design regarding the task requirements.

The next tool in Design for Variety is the MIG. Without knowledge about component configuration, interfaces and layout, creating this model faces a challenge. Therefore, we introduce a new tool to support this development phase: The "Product Family Interface Graph" (PFIG) for initial structuring of the product components and interfaces. Due to the early modularization on functional level, components, modules and its interfaces can already be portrayed. Different components are represented by the boxes - the module formation is highlighted in color corresponding to the functional modules determined during technical-functional modularization if possible. The different flows are energy, signal/data, force and torque each represented in a different color. The coloring of the components stems from strategic decision-making regarding product variety. Standard components (white) are used in every product family member. Variant components (grey) can be parametrized or easily scaled (e.g. battery capacity, wheel diameter) while custom parts (blue) need to be redesigned for every variant. The four product family models can be seen in figure 1. The next step is the further definition and design as well as combination of the different system elements. The electrical components need to be matched to the technical requirements and harmonized with design components such as housing that need to be created in CAD. For control and sensor units, software for signal acquisition and control needs to be developed as well as this product contains mechatronic shares.

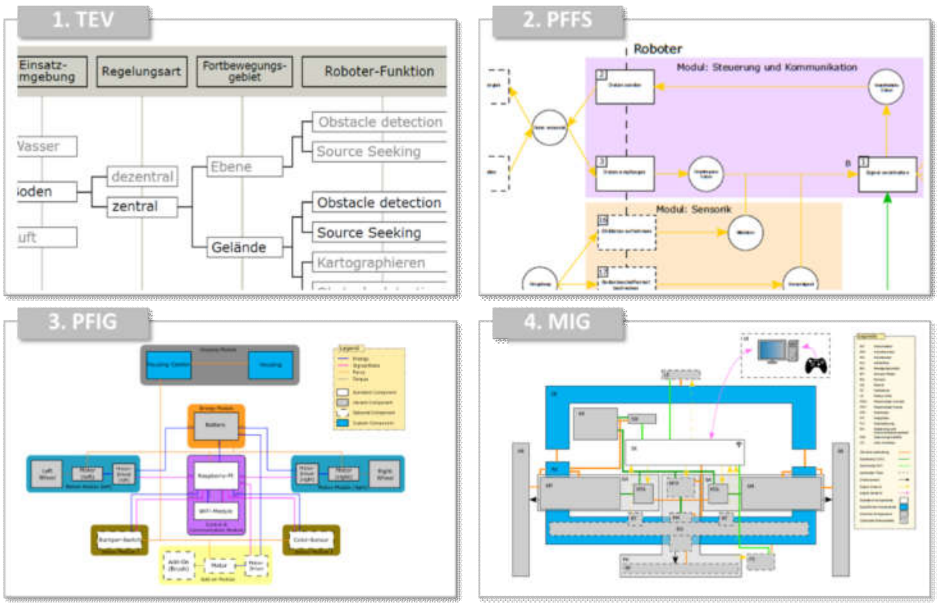

Figure 1. Partial models of the VAM in their order of emergence (except 3)

We want focus on the PFIG (figure 2) as this tool enables new possibilities. One of the most important features of the MIG is the representation of real shapes as it helps fast recognition of the components and aids interdisciplinary understanding (Gebhardt et al., 2014). The layout of the product is not yet defined at this stage but relevant components and their linkage can be seen which are prerequisites of handling complex systems. The information base is sufficient to work with as an early stage product structure with special regards to product variety represented by different colors. The developed product family structure can therefore be implemented in system modeling language to analyze and optimize the emerging interfaces. We now show the implementation in SysML via the CSM in form of a DSM on component and module level.

3.3 Support of the prototype development and analysis with SysML

In the PFID, different model-elements are shown which are connected to each other in different ways. The components are connected to each other via flows and are also connected to modules by local arrangement.

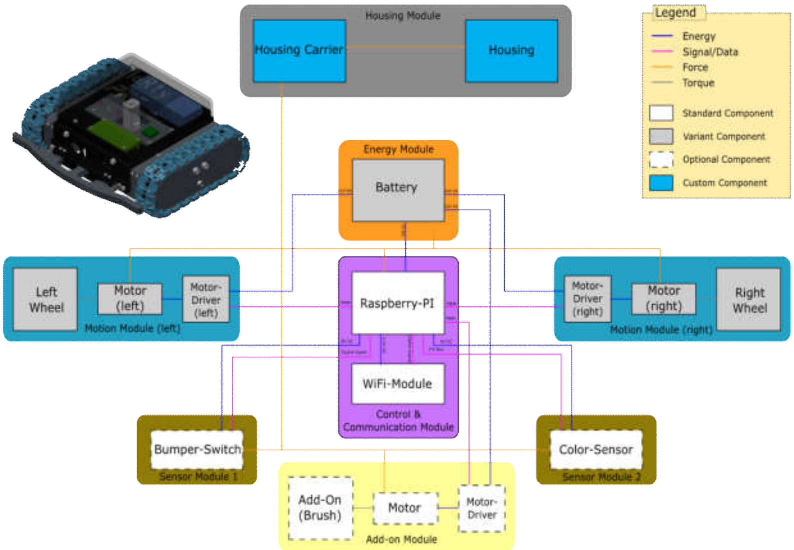

Figure 2. PFIG of the Product Family and CAD-model of the prototype under development

These data and data correlations can now be stored in a SysML-Model in CSM. This builds the foundation for consistent data overview and further analyses of the connections. Once the data has been initially inserted into Cameo, further product variants can be developed on this basis. The advantage is that information from the data and data contexts can be utilized consistently. For implementation the three different datatypes of the PFID and their attributes are identified and the SysML-notation was derived as can be seen in table 1.

Table 1. SysML-notation for Model Elements

Model Element	Attributes	Matching System Element (in SysML)
Module	▪ Color ▪ Contains other elements	(Smart)-Package
Component	▪ Color (incl. outlines) ▪ Linkable via flows	Block
Flow	▪ Color ▪ Name ▪ Connectable to components	Stereotyped dependencies

	▪ Directed	

At first, the different Element types were defined and the elements shown in figure 2 are created in the CSM. Then the components (blocks) are connected to the modules (packages) via the link type “containment” and the components are connected to each other via the flows (stereotype dependencies). In this example, the different stereotypes for the dependencies are: Energy (blue), Signal/Data (magenta), Force (orange) and Torque (grey).

Now, the data and their connections can be retrieved in a dependency matrix. The dependency criteria are the four stereotyped dependencies. The dependency matrix shows the coupling between components and between components and modules (figure3, left). Through implementation with SysML the data is stored consistently and traceable. The dependency matrix provides a DSM with foldable content which supports the analysis of interrelationships. The modules can be folded (figure 3, right). The sum of the stored links (both directions) then appears in the boxes. The interrelations are now clearly depicted and provide the basis for further product development e.g. in the context of new product generations. The matrix output can now be used for further analysis with DSM-optimization algorithms or network analysis software.

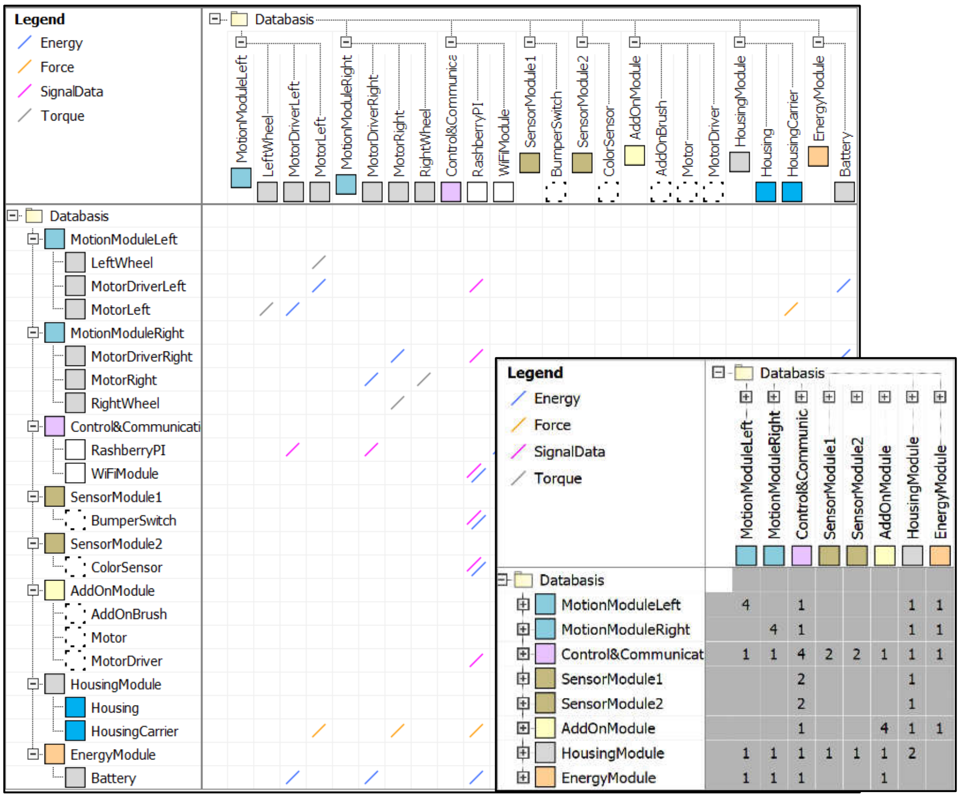

Figure 3. Left: DSM implemented in CSM, Right: Module View with summed interfaces in CSM

4 Evaluation of the Application

The initial development of a product structure and the representation of components and their relationships needs special consideration during the early phase of product development as they are hardly defined. The modular product structure is achieved by applying the heuristics after Stone on the PFFS within DfV after Kipp. As the strength of visualizations for concept creation in the early phase is used to easily develop product structures, matrix-based approaches are suited for optimization problems regarding module building and offering viable product variety. To combine the strength of both and enable consistency and traceability of information, SysML-models are a feasible way to create the matrices with its dependencies between elements that are needed for further development. Especially in the early phase dependencies between components and definition of interfaces are of high importance and therefore scrutinized in (Sundgren, 1999). The representation of the connections in matrices can help at this point. The couplings are systematically recorded and can be analyzed. The attachment of matrices can be used, for example, to analyze the degree of coupling of modules, which is a gradual feature of modularity. The aim is that the couplings of the components within a product are stronger than the couplings between the modules. (Salvador, 2007) The degree of coupling can be seen in figure 3: the numbers on the diagonal show the couplings (or the number of interfaces) within a module. Visual models support interdisciplinary teams as they strengthen a uniform understanding. The maintenance of visual models and implementation of changes during development activities are challenging though as the dynamics in modern markets is high and product development cycles shorten. Model-based approaches are a good way to track the data and support the interdisciplinary communication. Thus, the combination of the two can supplement a holistic product development process. The product structure can be used to derive organizational structures and production processes. A possible way to use DSMs and DMMs for such architectural constructs is shown in (Danilovic and Browning, 2006).

5 Conclusion

This paper shows the initial conceptualization of an early phase product structure derived within an evolving product architecture with support of DfV and modularization methods. The knowledge of the actual product structure is low in the concept phase while interface design is crucial to modular design as it enables component and process commonality. With the PFIG, a new tool is provided to support initial structuring of components and interfaces, which can then be implemented in the CSM for data storage and enabling analysis and optimization of the product structure.

References

Bahns, T., Melzer, S., God, R., Krause, D.. 2015. Ein modellbasiertes Vorgehen zur variantengerechten Entwicklung modularer Produktfmailien. Tag des Systems Engineering, Ulm.

Browning, T. 1998. Use of Dependency Structure Matrices for Product Development Cycle Time Reduction. Proceedings of the Fifth International Conference on Concurrent Engineering, ISPE, Tokyo.

Bursac, N.. 2016. Model Based Systems Engineering zur Unterstützung der Baukastenentwicklung im Kontext der frühen Phase der Produktgenerationsentwicklung. Karslruhe Institute of Technology, Karlsruhe.

Danilovic, M., Browning, T. R.. 2006. Managing complex product development projects with design structure matrices and domain mapping matrices. International Journal of Project Management 25, pp. 300-314.

Gebhardt, N., Bahns, T., Krause, D.. 2014. An Example of Visually Supported Design of Modular Product Families. 24th CIRP Design Conference.

Hanna, M., Schwede, L., Krause, D.. 2018. Model-Based Consistency for Design for Variety and Modularization. 20th International Dependency and Structure Modeling Conference, DSM, Trieste.

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

Hölttä-Otto, K., Valtteri, N., Eppinger, S., Browning, T., Stowe, H. M., Lampinen, R., Rahardjo, A.. 2018. Design Sprint for Complex System Architecture Analysis. Proceedings of the ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE, ASME, Quebec City.

Kipp, T.. 2012. Methodische Unterstützung der variantengerechten Produktgestaltung. Technical University of Hamburg, Hamburg.

Krause, D., Gebhardt, N.. 2018. Methodische Entwicklung modularer Produktfamilien: Hohe Produktvielfalt beherrschbar entwickeln. Springer, Berlin.

Lindeman U., Maurer, M., Braun, T.. 2009. Structural Complexity Management: An approach for the Field of product Design. Springer, Berlin, pp. 51-56.

Mikkola, J. H., Gassmann, O.. 2003. Managing Modularity of Product Architectures: Toward and Integrated Theory. IEEE Transactions on Engineering Management, Vol. 50.2, pp. 204-218.

Mikkola, J. H.. 2006. Journal of Product Innovation Management 23, pp. 128-146.

Salvador, F. 2007. Toward a Product System Modularity Construct: Literature Review and Reconceptualization. IEEE Transactions on Engineering Management Vol. 54, No. 2.

Sanchez, R.. 1996. Strategic Product Creation: Managing New Interactions of Technology, Markets and Organizations. European Management Journal, Vol. 14, No. 2.

Smith, R. P.. 1992. Development and Verification of Engineering Design Iterations. Massachusetts Institute of Technology, Cambridge.

Steward, D.. 1981. The Design Structure System: A Method for Managing the Design of Complex Systems, in IEE Transactions on Engineering Management, Vol. 28, No. 3, New York.

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

Sundgren, N.. 1999. Introducing Interface Management in New Product Family Development. Journal of Product Innovation Management 16, pp. 40-51.

Ulrich, K. T., Eppinger, S. D.. 2012. Product Design and Development. Fifth Edition. McGraw-Hill Irwin, New York, pp. 13-16.

VDI2221 Richtlinie. 1993. Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte. Düsseldorf.

Weilkiens, T.. 2008. Systems Engineering with SysML/UML: Modeling, Analysis, Design. Morgan Kaufmann Publishers Inc.. San Francisco.

Worren, N., Moore, K., Cardona, P.. 2002. Modularity, strategic flexibility, and firm performance: a study of the home appliance industry. Strategic Management Journal 23, pp. 1123-1140.

Contact: Jan Küchenhof, Institute of Product Development and Mechanical Engineering Design/Hamburg University of Technology, Denickestraße 17, 21073, Hamburg, Germany, +49-40-42878-2176, , jan.kuechenhof@tuhh.de, https://www.tuhh.de/pkt/