Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 16 (2013) 353 – 362

Procedia
Computer Science

Conference on Systems Engineering Research (CSER'13)
Eds.: C.J.J. Paredis, C. Bishop, D. Bodner, Georgia Institute of Technology, Atlanta, GA, March 19-22, 2013.

Extending SysML for Engineering Designers by Integration of the Contact & Channel – Approach (C&C²-A) for Function-Based Modeling of Technical Systems

Albers, Albert[a]; Zingel, Christian[a]*

[a]IPEK – Institute of Product Engineering at Karlsruhe Institute of Technology (KIT), Kaiserstr. 10, D-76131 Karlsruhe, Germany

---

Abstract

Model-Based Systems Engineering (MBSE) is advancing rapidly in the domains of software and embedded systems. In contrast, mechanical engineers still have trouble in application of MBSE. SysML has established as the leading modeling language for multidisciplinary systems, but some limitations still hinder mechanical engineers from its application. The provided behavioral and structural diagrams seem to not being sufficiently capable to represent all relevant information regarding mechanical systems. This paper presents an extending profile which aims to overcome some of those limitations. The profile is based on the Contact & Channel – Approach (C&C²-A), which is well-proven in function-based modeling of technical systems comprising function-relevant structural properties. The goals of the C&C²-A are to retain a maximal solution space, to overcome component-afflicted thinking and to provide an adequate modeling approach for mechanical relevant information. A prototypic tool implementation of the extending SysML-profile, complemented by some automatisms in form of a plugin, is evaluated at the example of a small gearbox.

"Keywords: Function-based modeling; state-based structure modeling; SysML; Contact & Channel - Approach; C&C²-A; Engineering design"

---

1. Introduction

Modern technical products are highly networked systems of mechanics, electrics and software in close interaction with their environment. These systems are called mechatronic systems, embedded systems, self-optimizing systems or cyber-physical systems, standing for a complex environment that challenges product development through factors like physical reliability or unpredictability but coincidentally offers an enormous economical and societal potential [1]. Development trends like the transition from document-based development approaches to Model-Based Systems Engineering (MBSE) approaches try to cope with the inherent complexity of such systems. Several efforts have

---

* Corresponding author. Tel.: +49-721-608-45486; fax: +49-721-608-6051.
E-mail address: christian.zingel@kit.edu.

doi:10.1016/j.procs.2013.01.037

been done to establish adequate modeling languages to describe embedded systems, consisting of software and electrics/electronics. Popular examples are AUTORSAR, MARTE or EAST-ADL. A more generic language is the Systems Modeling Language (SysML), which intends to support the specification, analysis, design, verification, and validation of a broad range of complex systems. These systems may include hardware, software, information, processes, personnel, and facilities [2]. Since its initial release in September 2007, SysML has been frequently applied and evaluated in research and industrial pilot projects, where its potential has been recognized but coincidentally several improvement issues have been identified. Bone and Cloutier present results from an OMG Request for Information that identified several aspects which inhibit successful adoption of the MBSE approach [3]. They recommend conducting further surveys regarding the root cause(s) of the steep learning curve of MBSE, the affection of tools and the application of diagrams. A survey has been done by the authors of this paper among German Systems Engineers in 2012 in order to determine more information about the state of common term understanding and SysML application [4]. The results from this study as well as experiences from several cooperation projects with industry motivated the authors to face SysML's limitations in function-based modeling with consistent derivation and representation of dynamic physical system structures. On the one hand, promising functional modeling approaches are emerging, which will be presented in the next chapter. On the other hand, there is still a lack of adequate modeling approaches for clear specification of function-relevant physical properties of system structures preserving a maximum solution space before deriving the resulting physical structures. Furthermore, dynamic structural changes depending on states of systems are not yet considered. The IPEK – Institute of Product Engineering has developed the Contact & Channel – Approach (C&C²-A) for integrated modeling of functions and including function-relevant physical properties of technical systems, which is well-proven for mechanical systems in academia and industry [5], [6]. The work at hand builds on our previous research regarding C&C²-A in SysML [7], [8] by modeling state-dependent structures, adding a functional structure using CSS and WSP and inheriting physical structures. The long-term goals of these efforts are to improve the usability and comprehensibility of SysML for engineering designers and increasing integration of physical and geometrical aspects into multidisciplinary system models. A functional modeling approach aims to facilitate a transition from component-afflicted to functional thinking. The next chapter introduces related functional modeling approaches and contrasts them to the approach presented in this paper.

2. Related research work

Shah et al. introduce an approach for multi-view modeling to support embedded SE in SysML [9]. They focus on describing the interrelations between involved disciplines and automated generation of discipline-specific models from a common multidisciplinary system model. An extending profile facilitates modeling of necessary entities and relations for these model transformations. The presented findings are considerable in terms of subsequent tool integration using a function-based modeling approach. The approach presented in this paper provides a basis for model transformations by distinguishing functional from physical structures for synchronization of other models (i.e. simulation models can be synchronized with functional structures and CAD models with physical structures).

Gausemeier et al. present the modeling language CONSENS (Conceptual Design Specification Technique for the Engineering of Complex Systems) for computer-aided cross-domain modeling of mechatronic systems [10]. Those models describe the principle solution as the result of the conceptual design phase of mechatronic and self-optimizing systems. The modeling language comprises several coherent partial meta-models in order to specify different aspects of mechatronic systems. For instance, "application scenarios" concretize the system behavior in a certain state and events causing state transitions. The system behavior itself is described in states and activities, similar to SysML. One core concept is the "active structure", which describes the system elements, their attributes and the relation of system elements. Functions are only described hierarchically, a transition methodology from activities over function-relevant physical properties to a resulting physical structure as well as dynamical changes of physical structures have not been addressed.

Wölkl and Shea apply SysML for the development of an integrated product model for mechanical design in early phases of product development [11]. They describe functions as Use Cases in SysML for the user perspective and decompose them for the device perspective using Activity Diagrams. According to Pahl et al., functions are described solution-neutrally as verb-noun-pair transitioning inputs to outputs [12]. Wölkl and Shea recommend the application of the NIST function database provided by Hirtz et al. [13] for decomposition to basic functions.

Afterwards, working principles are allocated to the abstract functional structure - similar to the "behavior"-level in Umeda's and Tomiyama's Function-Behavior-State (FBS) modeling approach [14] - using Block Definition Diagrams and Internal Block Diagrams. Blocks therein represent classes of working principles and Constraints are applied to describe physical equations. Working structures are established by interconnected Working principles. The matching of functions to working principles is told to be conducted traditionally using construction catalogues. The application of the FBS-approach and the NIST function database has been advanced by Kruse et al. [15]. They present an FBS-based approach using SysML, extended by additional stereotypes for user-specific "functions" and reusable "elementary functions" from the NIST function database. In this approach, a direct mapping from functions to structures is realized by allocations using allocation matrices, which is in addition graphically represented in Activity Diagrams using Activity Partitions (swimlanes). The NIST function database is extended by additional flows for modeling mechatronic systems. The modeling of system behavior is left out for future work. Both lastly introduced approaches propose a direct mapping from solution-neutral functions (modeled as activities) to components (modeled as blocks and block properties). The transition using working principles and working structures is promising; unfortunately it has not been adopted in the advancement. Furthermore, explicit modeling of function-relevant physical properties in order to preserve a maximum solution space before deriving physical components as well as modeling of dynamic structures has not been addressed within both approaches.

Eigner et al. present a modeling approach based on the Requirements-Functions-Logical Structure-Physical Structure (RFLP)-Approach using SysML, enhanced by additional stereotypes [16]. Their aim is to integrate the functional system model as part of a PLM solution in interdisciplinary product development. They propose a distinction into three views of modeling: firstly "Modeling and specification" using descriptive SysML models, secondly "Modeling and first simulation" using discipline-crossing tools like Matlab or Modelica and thirdly "Discipline specific modeling" using CAx-tools like mechanical or electrical CAD tools. A PDM environment using XMI as data interface format shall facilitate the networking of requirements, functional and logical structures. Specialized SysML elements like a block extended by the stereotype "function" or the specialized allocation "realize" are added using the profile mechanism of UML/SysML. Logical system elements represent the realization of functions comprising physical effects and properties and are networked to the physical Bill Of Material (BOM) within the PDM environment, which are again linked to M- or E-CAD-elements and software. This approach considers function-relevant physical properties, but it requires two additional views for functional and logical structure. Again, the transition from logical structures to physical structures is not conducted in SysML (they are only loosely allocated). Moreover, this approach does not address modeling of dynamic structures.

Lamm and Weilkiens present the FAS-Method (Functional Architectures for Systems), an approach for function-based development of technical systems enabling re-use of concepts across multiple generations of technology [17], [18]. Again, the approach applies SysML and enhances it by a FAS-profile, comprising stereotyped elements in order to facilitate modeling according to the FAS-Method. The approach starts with identification of functional and non-functional Requirements. Those are refined by Use Cases, which are further decomposed through Activities. Those Activities are grouped into functional groups using heuristics, defined in [17]. For instance, grouping criteria like shared data can be applied. The resulting, automatically generated functional blocks are similar to functional blocks in Simulink and could be transformed and detailed there using modeling tool interfaces like i.e. presented in [8]. Methods for mapping of functional blocks to physical blocks are currently under development. The approach uses concepts which are also applied in the presented approach at hand, wherefore it is intended to combine the approach at hand with the FAS-Method in the future, using the cooperation platform of the GfSE[b] FAS working group. The main advancement will be to apply concepts of the Contact & Channel – Approach ($C\&C^2$-A) in combination with the FAS-Method in order to facilitate a methodology for modeling function-relevant physical properties including the transformation from functional to physical structures as well as modeling dynamic structures depending on system states in SysML. For this purpose, the $C\&C^2$-A and its core concepts will be introduced in the next chapter.

---

[b] GfSE stands for Gesellschaft für Systems Engineering, the German chapter of INCOSE

3. The Contact & Channel – Approach (C&C²-A)

Numerous approaches for modeling software and embedded systems using object-oriented, graphical modeling languages are emerging, but none on them is capable to represent all relevant information of physical systems. SysML in combination with upcoming approaches for function-based modeling are promising, but still offer a lack of representing function-relevant physical properties. Moreover, dynamically changing system structures with changing system states cannot be represented. For instance, a clutch can be either opened or closed and the contact between the clutch plates either exists or not. Depending on the state of the clutch ("opened" or "closed"), the function of transmitting torque can be performed or not. This state-dependent structure (a contact between the clutch plates) cannot be modeled in existing modeling languages using available approaches today. The Contact & Channel – Approach (C&C²-A)[c] was developed at IPEK – Institute of Product Engineering in order to facilitate a structured analysis and synthesis of form and function of technical systems. This is realized by the basic elements **Working Surface (WS)**, where two of them can form a **Working Surface Pair (WSP)** when being in contact as well as the **Channel and Support Structure (CSS)**. The basic elements describe all function-relevant information of technical system elements or subsystems, which can be physical or chemical effects (i.e. friction, heat emission) or properties (i.e. material, length). Furthermore, a **Connector (C)** carries properties of adjacent systems, which are relevant for describing system functions in its environment. Three basic hypotheses define the rules for a consistent application of the approach (according to Matthiesen and Ruckpaul [19]):

- **Hypothesis 1:** Every technical system fulfills its function by interacting with adjacent systems. Effects can only take place if a WS is in contact with a further WS and thus a WSP is built up.
- **Hypothesis 2:** Functions are represented by at least two WSP's, the connecting CSS and at least two Connectors which embed the system model into its environment. The properties of WPSs, CSSs, Connectors and the effects taking place in the WSPs and CSSs are determining for the fulfillment of the function.
- **Hypothesis 3:** Every system and subsystem can be described by the basic elements WSP, CSS and Connector on different levels of abstraction and detail.

Using the basic elements, technical sketches or drawings, sectional representations or even pictures of a real system can be analyzed. Examples for the application of the C&C²-A can be found in [5], [6] and [19]. Therefore, the geometrical arrangement in addition to the functional analysis supports the comprehension of the function principle of a technical system. In fact, the basic elements can be used in an abstracted manner for form-neutral specification of system functions using property parameters to describe function-relevant physical parameters. This is realized in the approach that is presented in the next chapter.

4. Function-Based Modeling Approach for Technical Systems using C&C²-A@SysML

The particular challenge addressed in this paper is to enhance SysML by a profile according to the needs of engineering designers. Similar to the FAS-method that enhances Weilkiens' SYSMOD approach [20], the approach presented in this chapter does not aim to replace elaborated MBSE methodologies (like those surveyed by Estefan [21] or the methodology presented by the authors of this paper in [8]) but rather intends to enhance them, wherefore they are not contrasted here. The approach aims to provide a continuous development approach with model support from solution-neutral activities over functional structures capturing function-relevant physical properties up to the resulting physical structure. Hitherto, functional structures were only mapped to physical structures using allocations and their properties (Values) and interfaces (Flow Ports) have hence not been transmitted yet. Therefore, they had to be modeled again. The method at hands uses specializations for inheriting the properties that have been defined in the function-based structure to physical structures using a matrix for mapping the functional elements to physical elements. Moreover, an extending property of the SysML Connector facilitates to assign system states, whereby two flow ports (Working Surfaces - WS) can form a WSP only for certain states. The previously described approach with adjacent modeling activities is depicted in Fig. 1.

---

[c] The square at the C² stands for the element "Connector", which is usually not quoted in the full name

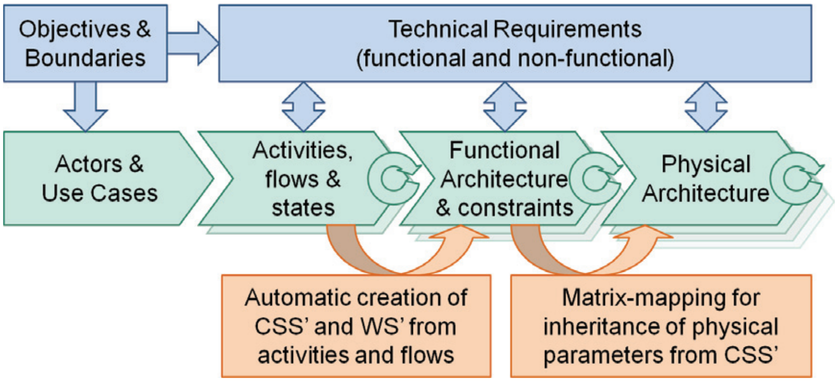


Fig. 1: Scheme of the modeling approach

Firstly, stakeholder objectives and boundary conditions are captured. Use Cases, standing for Top-Level system functions, are defined and linked to their actors in order to refine function-affecting objectives. In the next step, those Use Cases are decomposed into logical sequences of activities, representing the sub-functions. Coincidentally, system states are defined in order to subdivide, when the required WSP's for certain functions exist and they thus can be performed. For instance, a continuous function is performed within a certain state (i.e. torque can only be transmitted as long as a gear is engaged). Discrete functions are performed within state transitions (i.e. shifting between gears). Furthermore, the object flows (flows of matter, energy or information) are added into the activity diagrams in order to establish active structures. This decomposition of functions is, similar to the following steps, an iterative process which is represented in Fig. 1 using the circular arrows. Activity Partitions (or swim lanes) are used to structure the activities into functionally coherent groups. This can be conducted i.e. by applying the heuristics used in the FAS-Method by Lamm and Weilkiens [17] or similar, depending on the system to be modeled. From those activity partitions, CSS are generated automatically by a procedure of the prototypic tool plugin coming with this method, when it is executed by menu command. Furthermore, the plugin identifies object flows crossing the activity partition borders and automatically creates WS (see Fig. 2).

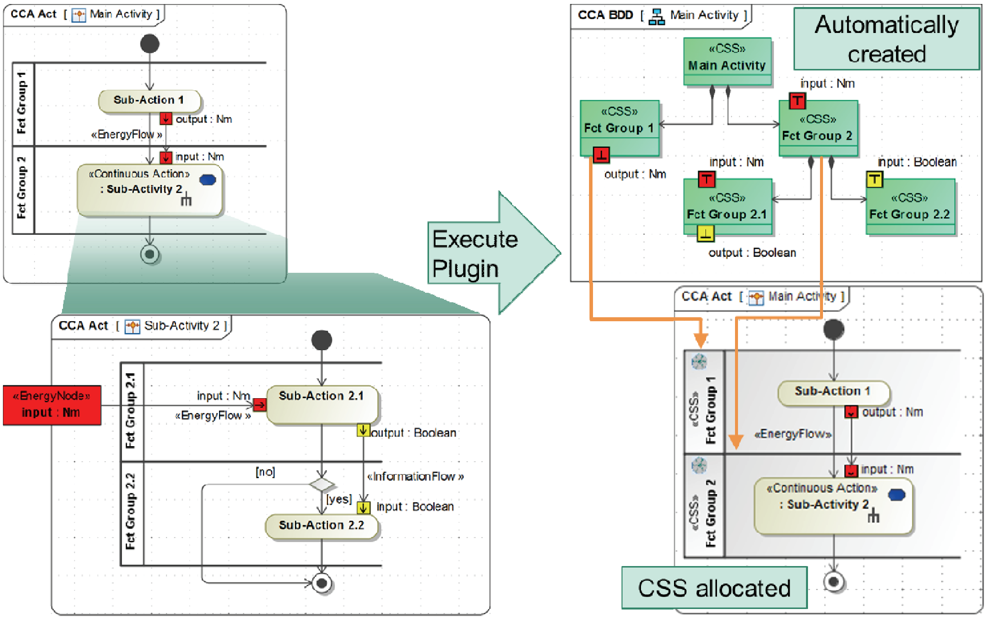


Fig. 2: Automatic creation of CSS and WS as well as CSS allocation to Activity Partitions

The WS act as interface for the object flows and are modeled in SysML using stereotyped Flow Ports as matter-, energy- and information-interfaces. The purpose of these partitions and the generated CSS is to enable modeling function-relevant physical properties (i.e. by values) and to establish a functional structure, which can explicitly be mapped to physical components. The abstract example in Fig. 2 shows a decomposed activity with activity partitions (left side), the automatically created CSS and WS for these activities (right upper side) as well as the allocated CSS to the activity partitions (right bottom side).

When the CSS are created, the functional structure can be modeled using specialized Internal Block Diagrams. Therein, stereotyped SysML-Connectors are used to network WS to WSP. The Connectors have an additional property for the assignment of system states for which they shall exist. The purpose of this assignment is to consider that WSP’s not necessarily exist for every system state. For instance, a WSP between two clutch plates only exists for the clutch’s state “closed” but not when it is “opened”. Using this Connector property, the model can be checked against correct assignment of activities to states, conducted within state diagrams using consistency rules.

Frequently, one physical component takes part of more than one function and can therefore inherit the function-relevant properties of all participated functions. When components are created, CSS can be mapped to components using a matrix (see Fig. 3 left side). The mapped components inherit all properties from the CSS including their WS (interfaces). The resulting relationships between CSS (green) and Physical Components (grey) are depicted in the Block Definition Diagram (see Fig. 3 right). When having mapped the components to CSS, the physical structure can be modeled using another specialized IBD for physical structures.

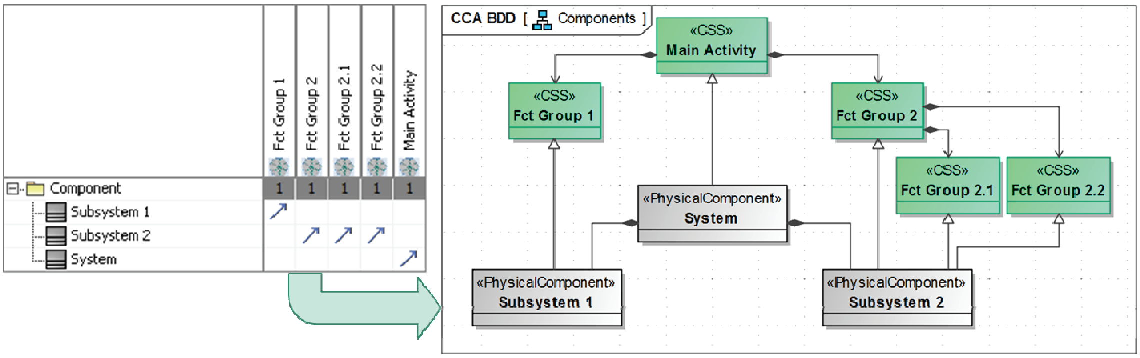


Fig. 3: Mapping of CSS to components

Some functions need constraints for their description when being decomposed, because relations of function-relevant physical properties must be defined in order to realize a function. An illustration of a dynamic structure including the according states and a function using a constraint is presented in chapter 5 using the application example of a small gearbox with 2 gears. In parallel to all those modeling activities, technical requirements are derived and iteratively and continuously complemented throughout the entire product development process (for details towards the applied continuous requirements modeling and tracing methodology ref. to [8]). All enhancements of SysML were done using the profiling mechanism; automatisms were programmed in a Java Code using the API of the modeling tool. The specialized diagram types (one specialized IBD for functional and one for physical structures) were added in order to reduce the tool bars onto the required minimum. In other respects, they are identical to regular Internal Block Diagrams.

4.1. Modeling rules

In order to facilitate correct modeling, several automatisms have already been realized within the prototypic tool plugin. For instance, the provided types of object flows (matter, energy, information) only allow the assignment of adequate datatypes. Furthermore, the correct relations between activity partitions and CSS as well as between CSS and components are automatically created. Coincidentally, the WS automatically receive the correct datatypes that are applied for the object flows crossing activity partition borders. For clearer modeling of mechatronic systems, additional rules can be applied in order to network only subsystems which are capable to interact in reality as well. Bertsche presents helpful modeling rules, which base on a 5-layer-model that is derived from the OSI 7-layer model for software systems (the 5 layers are the following system layers: mechanical, electro-mechanical, electronic, drivers and software) [22]. For instance, he proposes that electro-mechanical systems (sensors and actors) are not able to interact directly, but either via mechanical or via electronic systems.

4.2. Tool interoperability options

The presented profile including the additional diagram for modeling functional structures (using CSS and WSP) is not only capable to facilitate more continuous modeling but also to associate partial models with other development tools and models. Fig. 4 shows the three main modeling steps in the extended SysML using C&C²-A in the left (green) area and possibly linked tools for model synchronization in the right (blue) area.

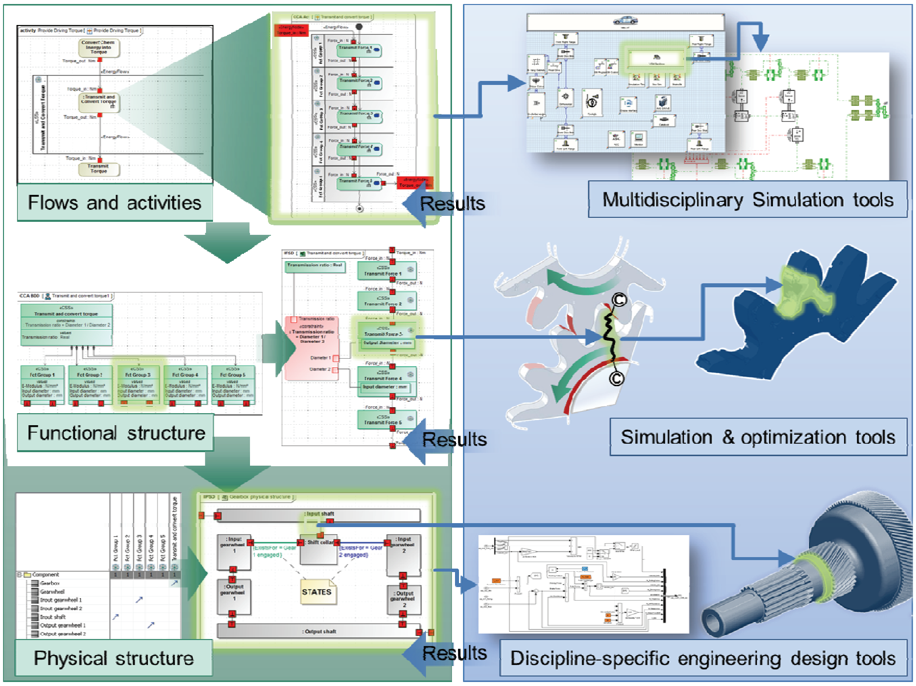
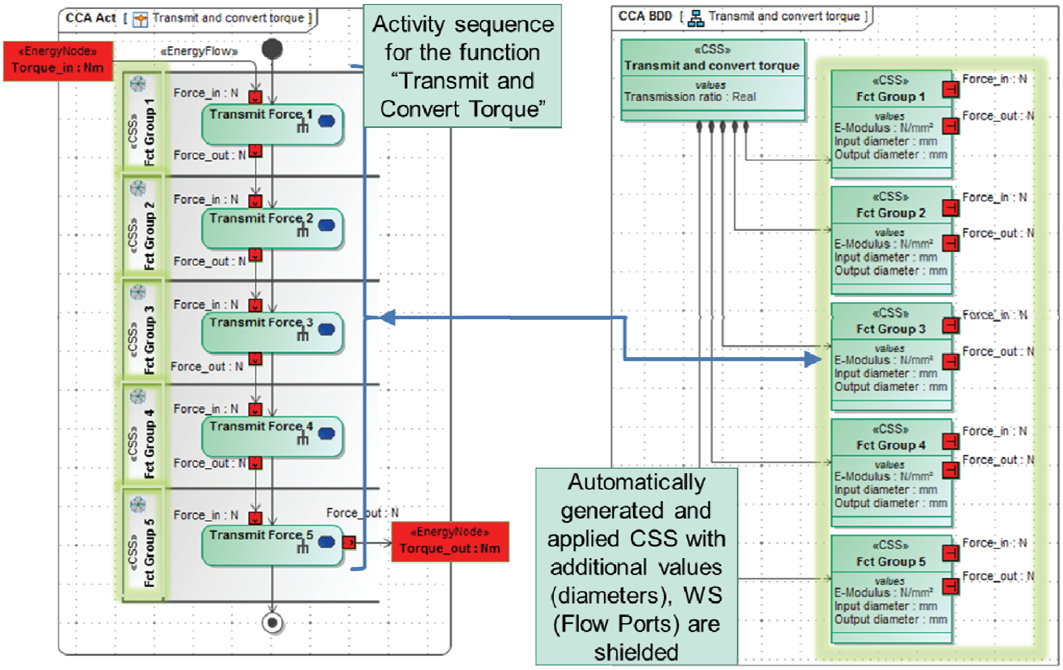


Fig. 4: Modeling steps and possible tool interaction

Information from activities, object flows and activity partitions can be used for multidisciplinary high-level simulation tools (i.e. to define model frameworks with interfaces and black boxes). Simulation results like geometrical forces could be transferred back to SysML and stored as parameters in the functional structure. CSS carry function-relevant physical properties. Using the example of a gearwheel, its function is to transfer force from the cog to the shaft. For performing this function, a certain structure – the Channel and Support Structure (CSS) - is required. Such a CSS is highlighted in the middle of the right side of Fig. 4. The resulting required material tensile modulus from a finite element simulation tool using predefined geometries will again be transmitted back to SysML. From such kind of information, physical components could be chosen and modeled in the physical structure. The next chapter illustrates the presented approach using the example of a small gearbox.

5. Application Example “Gearbox”

In this chapter, some modeling aspects of a small gearbox with 2 gears are introduced. Though this system is purely mechanical and the complexity is not too high, it would need much more room than the available 10 pages to present all modeled aspects for the entire gearbox in this paper. Therefore, diagram examples show partial aspects in order to elucidate how to use the presented profile. The modeling of stakeholder objectives, boundary conditions, Use Cases and technical requirements is not shown here, but more information and examples regarding those aspects can be found in Zingel et al. [8]. Modeling of the example starts with decomposing the activities of the main function of a gearbox (Transmit and convert torque) and generation of CSS and WS (see Fig. 5).


Activity sequence for the function "Transmit and Convert Torque"

Automatically generated and applied CSS with additional values (diameters), WS (Flow Ports) are shielded

Fig. 5: Activies of function "Transmit and Convert Torque" (left) and automatically generated CSS (right)

The Activity Diagram on the left shows five sub-activities which are performed within the main function "Transmit and convert torque". The five resulting CSS have additionally obtained three value properties each (input and output diameters in order to calculate torque from forces as well as the value E-modulus). The activities derive from a retrospective analysis of an existing gearbox. The transmission and conversion of torque is conducted by the interaction of an input shaft, a shift collar, two toothed gearwheels and an output shaft. This decomposition was done in order to demonstrate the application of the profile and the modeling approach, in a real scenario the sequence of modeling steps is not that stringent like it is shown here but rather much more iteratively. CSS could also be created manually first and allocated to activity partitions subsequently. Hence, the way of modeling using the profile is flexible, according to numerous possible application scenarios in a real product development process.

The next logical step in this example is to model the functional structure using the CSS and their WS in order to establish WSP. Due to that the CSS within the gearbox in fact transmit forces using their WSP's, the function-relevant diameters are needed. The conversion of torque is not realized by one single sub-function, but comprising the relation of the diameters shown in the diagram. Constraints can be applied to express such parametric relations (i.e. the quotient of the two affected diameters yields the transmission ratio respectively the conversion factor). The resulting transmission ratio is a value of the main function as shown in the diagram on the left in Fig. 6.

Afterwards, physical components can be created. They must apply the hitherto modeled function-relevant WS and values. This is done by specialization of CSS, assigned in a matrix for easier handling and better clarity. The right diagram in Fig. 6 exemplifies the inheritance of properties by physical components from CSS and (as shown here) of additional abstract element classes (here: "Gearwheel"). Therefore, every component inherits a modulus, which is a general parameter of gearwheels.

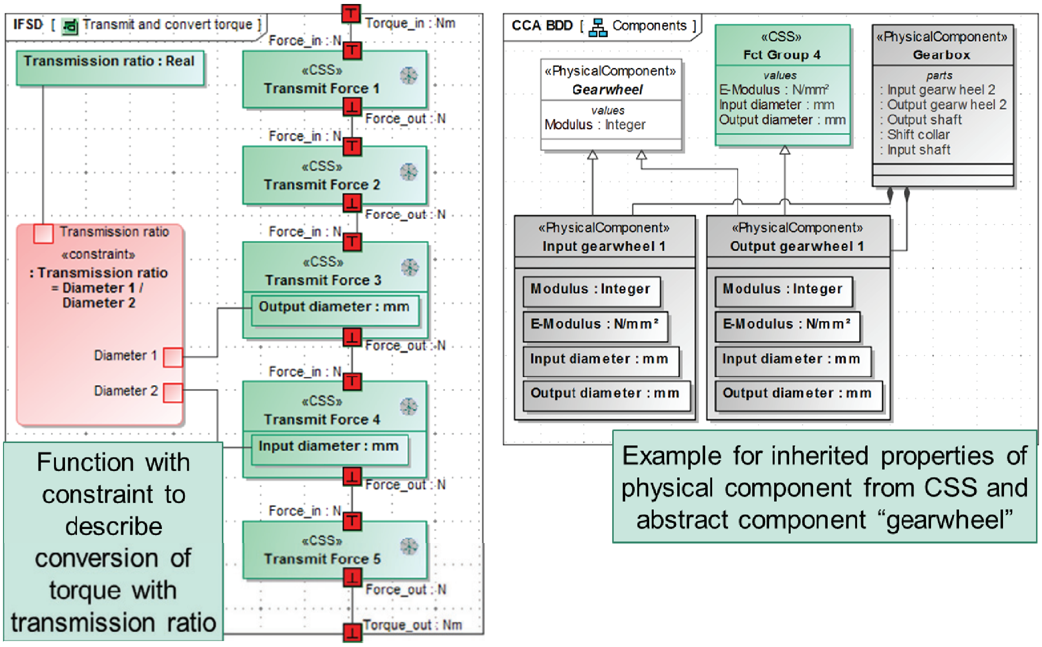


Fig. 6: Functional structure of "Transmit and Convert Torque" (left), inheritance of properties to physical components (right)

Finally, physical structures can be modeled. The left diagram in Fig. 7 depicts the dynamic physical structure of the gearbox, where states (right diagram) have been applied to the connectors between the shift collar and the two gearwheels. Using this method, the model contains the information, which WSP can carry object flows within which system state. Moreover, the tool plugin is able to hide or display connectors depending on the system states.

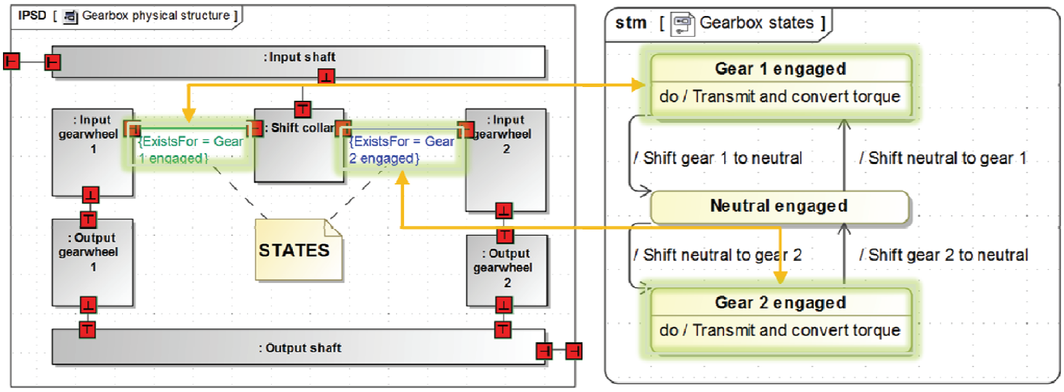


Fig. 7: Physical structure (left) and states (right) of the gearbox example

6. Discussion and Outlook

The presented profile extends SysML in order to facilitate function-based modeling of mechatronic systems with dynamic structures. The presented modeling approach supports mechanical engineers in function-based modeling and can be applied as part of established modeling methods like presented in [8], [17] or [21]. Moreover, possible future integrations of other models in product engineering to their according aspects in SysML were introduced.

Currently, the profile and the according approach are applied within the funded research project "Function-based controlling of mechatronic products" in cooperation with the Virtual Vehicle Competence Center (Graz), AVL List GmbH (Graz), BMW AG (Munich) and the Institute of Product Development (TU Munich). The aim is to establish function-based product portfolio management, based on a semantic knowledge domain. Function-based SysML-product models are the main information basis within this framework in order to provide decision rationales [23].

In the future, the profile will be further advanced, i.e. by providing artifacts for modeling physical properties (joint method, relative movement) and effects (friction, corrosion or heat emission) of WSP. The scientific challenges are to achieve a tradeoff for the level of detail as well as to identify relevant physical properties and effects for multiple disciplines or improve system understanding in order to substantiate whether this information has to be integrated in a multidisciplinary SysML model.

References

[1] Lee, E.A.: Cyber Physical Systems: Design Challenges. In: Proceedings of the 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC). Orlando, Florida, USA, 2008

[2] Object Management Group (Pub.): Systems Modeling Language (SysML). Specification of Version 1.3, 2012, available at http://www.omg.org

[3] Bone, M.; Cloutier, R.: The Current State of Model Based Systems Engineering: Results from the OMG™ SysML Request for Information 2009. In: Proceedings of the 8th Conference on Systems Engineering Research (CSER2010). Hoboken, NJ, USA, 2010

[4] Albers, A.; Zingel, C.: Challenges of Model-Based Systems Engineering: a study towards unified term understanding and the state of usage of SysML. In: Proceedings of the 23rd CIRP Design Conference. Bochum, Germany, 2013 (accepted)

[5] Albers A.; Braun A.: Contact and Channel Modelling to support early design of technical systems. In: Proceedings of the 17th International Conference on Engineering Design (ICED'09), Stanford, USA, 2009

[6] Matthiesen, S.: Seven Years of Product Development in Industry – Experiences and Requirements for Supporting Engineering Design with 'thinking tools'. In: Proceedings of the 18th Int. Conference on Engineering Design (ICED'11). Copenhagen, Denmark, 2011

[7] Albers, A.; Zingel, C.: Interdisciplinary Systems Modeling Using the Contact & Channel-model for SysML. In: Proceedings of the 18th International Conference on Engineering Design (ICED'11). Copenhagen, Denmark, 2011

[8] Zingel, C.; Albers, A.; Matthiesen, S.; Maletz, M.: Experiences and Advancements from One Year of Explorative Application of an Integrated Model-Based Development Technique Using C&C²-A in SysML. In: IAENG International Journal of Computer Science, Volume 39, Issue 2, 2012

[9] Shah, A.A.; Kerzhner, A.A.; Schaefer, D.; Paredis, C.J.J.: Multi-view Modeling to Support Embedded Systems Engineering in SysML. In: Graph Transformations and Model-Driven Engineering (Lecture Notes in Comp. Sc., Vol. 5765). Springer, Berlin, 2010

[10] Gausemeier, J.; Dorociak, R.; Pook, S.; Nyßen, A.; Terfloth, A.: Computer-aided cross-domain modeling of Mechatronic Systems. In: Proceedings of the 11th International Design Conference (DESIGN '10). Dubrovnik, Kroatien, 2010

[11] Wölkl, S.; Shea, K.: A Computational Product Model for Conceptual Design using SysML. In: Proceedings of the ASME 2009 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC/CIE 2009). San Diego, California, USA, 2009

[12] Pahl, G.; Beitz, W.; Feldhusen, J.; Grote, K.: Engineering Design – A Systematic Approach. 3rd edition, Springer, London, 2007

[13] Hirtz, J.; Stone, R.; Mcadams, D.; Szykman, S.; Wood, K.: A Functional Basis for Engineering Design: Reconciling and Evolving Previous Efforts. In: Research in Engineering Design, Vol. 13, No. 2. Springer, 2002

[14] Umeda, Y.; Tomiyama, T.: FBS Modeling: Modeling Scheme of Function for Conceptual Design. In: Proceedings of the 9th International Workshop on Qualitative Reasoning about Physical Systems. Amsterdam, Netherlands, 1995

[15] Kruse, B.; Münzer, C.; Wölkl, S.; Canedo, A.; Shea, K.: A Model-Based Functional Modeling and Library Approach for Mechatronic Systems in SysML. In: Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC/CIE 2012). Chicago, IL, USA, 2012

[16] Eigner, M; Gilz, T.; Zafirov, R.: Proposal for Functional Product Description as Part of a PLM Solution in Interdisciplinary Product Development. In: Proceedings of the 12th International Design Conference (DESIGN '12). Dubrovnik, Croatia, 2012

[17] Lamm, J.G.; Weilkiens, T.: Functional Architectures in SysML. In: Proceedings of the Tag des Systems Engineering (TdSE '10). Munich, Germany, 2010

[18] Korff, A.; Lamm, J.G.; Weilkiens, T.: Tools for Forging the Functional Architecture. In: Proceedings of the Tag des Systems Engineering (TdSE '11). Hamburg, Germany, 2011

[19] Matthiesen, S.; Ruckpaul, A.: New insights on the Contact & Channel Approach - Modelling of Systems with several logical states. In: Proceedings of the 12th International Design Conference (DESIGN '12). Dubrovnik, Croatia, 2012

[20] Weilkiens, T.: Systems Engineering with SysML/UML: Modeling, Analysis, Design. Morgan Kaufmann, Burlingston, USA, 2007

[21] Estefan, J.A.: Survey of Model-Based Systems Engineering (MBSE) Methodologies (Revision B). INCOSE, Seattle, USA, 2008

[22] Bertsche, B.; Göhner, P.; Jensen, U.; Schinköthe, W.; Wunderlich, H.-J.: Zuverlässigkeit mechatronischer Systeme: Grundlage und Bewertung in frühen Entwicklungsphasen. Springer, Berlin, Germany, 2009

[23] Denger, A.; Fritz, J.; Kissel, M.; Parvan, M.; Zingel, C.: Potentiale einer funktionsorientierten Lenkung mechatronischer Produkte in der Automobilindustrie. In: Proceedings of the Tag des Systems Engineering (TdSE 2012). Paderborn, Germany, 2012