Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 109 (2022) 550–555

www.elsevier.com/locate/procedia

32nd CIRP Design Conference

Approach for model-based requirements engineering for the planning of engineering generations in the agile development of mechatronic systems

Emily Windisch[a], Constantin Mandel[a]*, Simon Rapp[a], Nikola Bursac[b], Albert Albers[a]

[a]*IPEK – Institute of Product Engineering, at Karlsruhe Institute of Technology (KIT), Kaiserstr. 10, 76131 Karlsruhe, Germany*
[b]*TRUMPF Werkzeugmaschinen GmbH + Co. KG, Johann-Maus-Str. 2, 71254 Ditzingen, Germany*

* Corresponding author. Tel.: +49721-60847198; fax: +49721-60845487. *E-mail address:* constantin.mandel@kit.edu

Abstract

The crucial factor for a successful usage of modeling approaches of systems engineering is the interaction of language, method, and tool. For this, specific challenges arise for the application of MBSE in agile requirements engineering. From observations in agile development practice at a machine tool manufacturer, the challenges for model-based requirements engineering are described and each is assigned to its critical aspect of modeling: The language must formally represent the requirements data model, especially for planning engineering generations. The tool must support collaborative, interdisciplinary cooperation, and consider the dynamics of the requirements model during the development process. The method must individually support the requirements engineering activities, which are carried out several times in a sprint during the development process and must enable a target-oriented process for bundling the requirements into engineering generations. Taking these demands into account, an approach is then presented providing activity-based views in conjunction with activity steps based on a consistent ontology for the description of product requirements and verification activities. The activity steps are composed in activity patterns and support the user in making use of the views for modeling requirements for the engineering generations. The approach is implemented in the software JIRA at a machine tool manufacturer. The subsequent evaluation shows that the approach is used in development practice and offers the potential to plan engineering generation systematically and comprehensibly and to ensure a regular review of the implemented requirements.

Keywords: mbse; agility; validation; test planning

1. Introduction and motivation

In a study by the Automotive Agile PEP - Survey in 2018, representatives from automotive industry evaluate the status quo of agile working in mechatronic product development. This shows that unclear requirements are seen as one of the largest difficulties in agile development projects [1]. The reasons for this may lie in capturing the right requirements, consistently recognizing their interactions, and deriving from this the right conclusions for the development of the product. This is reinforced when late changes in requirements must be welcomed with establishment of an agile way of working.

To support requirements engineering, the benefits of model-based approaches such as MBSE - Model-Based Systems Engineering can be exploited, by consolidating information in a consistent model. Model-based approaches offer the possibility of making the complexity of systems manageable while providing, a holistic, cross-domain and cross-lifecycle view of the overall system, thus making it possible to represent and analyze the impact of changes in comprehensible manner [2]. As a mutual basis for communication, this can promote collaborative exchange among the participating business units based on a common understanding.

Validation, the central activity of product development [3], is aiming at ensuring that the planned product fulfills its intended purpose from the stakeholder's point of view. Therefore, it is essential to link the previously defined requirements with the validation activities. In addition, the results of validation must be linked to relevant requirements and an evaluation from the stakeholder's point of view to be able to assess their impact on product development. Moreover,

10.1016/j.procir.2022.05.293

it is necessary to investigate which views of the requirements model are required to derive the meaning for the requirement and the intended stakeholder benefit from the validation results. Based on this, the requirements can be further specified, and new activities and tasks can be identified for the next sprint.

2. Background and state of research

2.1. Agile product generation engineering of mechatronic systems

For successful product development, companies must be able to align their development processes in a customer-oriented and still economical way. The model of PGE - Product Generation Engineering offers companies the potential of targeted planning and control of their development activities. According to the model of PGE, development of a new product generation is always based on references [4]. The realization of a new product generation based on a reference system can be described by different shares of carry-over, attribute and principle variation. Through the systematic use of reference system elements, the assessment of development risk and demand for resources can be supported and strategically controlled for one new product generation as well as over several generations. Following the agile way of working, development is progressing in increments. The model of PGE is also applicable to increments within the development of a product generation [5]. Accordingly, a product generation comprises development increments that differ in their degree of maturity [5]. Since the complexity of today's mechatronic systems makes it unfeasible to generate increments from a blank sheet of paper, references play an essential role. In the planning of engineering generations, knowledge of the reference system elements and of previous engineering generations forms an integral basis [6]. Based on the existing reference system, a new increment scope can be planned which will be executed in certain timeframe in an iterative procedure. One way of defining an increment and its scope is the Minimum Viable Product (MVP) [7]. The MVP has the functional scope needed to obtain stakeholder feedback for further development within a defined timeframe [8] and thus to validate the realization based on stakeholder needs. Through this iterative, incremental approach, a complex task that is difficult to plan is broken down and processed sequentially in a prioritized manner. Agile development approaches such as Scrum [9] provide structured support for collaboration in interdisciplinary teams and ensure a transparency in processes.

2.2. System of objectives and requirements engineering

Based on systems theory, the engineering of a technical system can be described as the continuous transfer of an initially vague system of objectives into a concrete system of objects by an operating system. A system of objectives contains all relevant objectives, requirements and constraints, as well as their dependencies that are essential for the development of the product [10]. Objectives define what the product in development is supposed to achieve. Requirements, on the other hand, describe what the product should be able to do and are derived from the objectives [11]. The continuous extension and concretization of the system of objectives is only enabled by the gain of knowledge from continuous validation in the development process. Only by continuous analysis of the system of objects it is possible to gain case-specific knowledge and thus to make the system of objectives more and more concrete [10]. Accordingly, validation must be systematically addressed and consistently linked to the system of objectives.

There are interactions among the elements of the system of objectives as well as with elements that are not primarily assigned to the system of objectives. To be able to realize consistency in the system of objectives, these interactions must be explicitly mapped in form of modeled relations. Here, the usage of rigid requirement lists reaches its limits. It takes models to represent the existing interactions and to capture the dynamic development of the system of objectives.

A sub-activity of systems of objectives development can be seen in the widespread discipline of requirements engineering, which takes a systematic approach for specifying and managing requirements. Since systems of objectives are developed through the interdisciplinary cooperation of distributed business units, this is essential in managing the requirements and promoting a transparent development process. With consistent modeling of the requirements, as part of knowledge management in the company, successful product development can be supported by the transparency of the system of objectives' contents and their justifications [12].

2.3. Model-based requirements engineering

MBSE can overcome the limitations of the document-based approaches that are still frequently used in requirements engineering by supporting a consistent and common understanding of systems [2]. For a successful use of MBSE, Delligatti describes the integration of three needed aspects as the three pillars of MBSE: language, tool, method [13].

The language provides a notation for the elements and relations by which a model can be formulated. Only the transparent and traceable integration of objectives as elements of a modeling language allows to perform a consistent validation. A direct link therefore enables engineers to trace back to the objectives every time a change is made to the system at the levels of requirements, structure, or behavior [14]. The language is built on an ontology. Ontologies can be distinguished by their characteristic as lightweight or heavyweight. A lightweight ontology includes objects and relations between objects as well as properties describing objects. To represent the meaning of objects and relationships in a machine-interpretable way, the lightweight ontology is extended with basic mathematical rules and constraints. The resulting ontology can be understood as a heavyweight ontology and has a higher degree of formality. [15]

An established standard a MBSE modeling language is the Systems Modeling Language (SysML). Standardization through the modeling language is an important factor for formalization, in order to create a common understanding between people in a company. However, due to a high degree of formalization, the entry barrier for the use of SysML is described as quite high [16, 17]. In many applications, not all the elements of SysML are required and thus tend to complicate the application [18]. Therefore, a suitable adaptation is appropriate when utilizing a formal modeling language [19].

Regarding the existing need for flexible SE approaches, especially the utilized modeling tool becomes a crucial factor [20]. A modeling tool is needed to realize requirement models. The tool serves as a platform for implementing the language and providing a development environment for the model. In this way, the common exchange about the model contents can be realized. Many tools used in practice for requirements management are specialized tools that do not implement a standard like SysML. On the other hand currently available MBSE modeling tools are still perceived as expert tools and thus lack a broad acceptance in practice [18].

If requirements are modeled in a tool according to the rules of a modeling language, they are stored in the form of elements and relations to each other in a repository. In this form, information is not easily accessible, especially in the case of large data sets, and there is a lack of clarity due to the high level of interconnectedness. Views offer the possibility of displaying a section of the model as required, for example in dashboards, matrices, diagrams, or tables, which each serve as a filter. Views are usually constructed based on roles (see e.g. [21]). This approach does not help to promote mutual collaboration. Model frameworks, such as the SPES framework, are used to organize the views [22]. A procedure for developing MBSE approaches starting from an ontology and using the definition of views and a modeling framework is described by Holt and Perry [23]. How the views are to be used in the development process is often not described in detail [24].

A suitable modeling method must consider recurring activities which can be carried out in the development process according to the given situation. By avoiding rigid procedures, the barriers of modeling can be reduced and the iterative procedure in the development process can be implemented [24, 25]. Crucial for a successful use of MBSE approaches with all advantages, is the use in the agile workflow of engineering. How planning of engineering generations can be considered here needs to be investigated.

3. Aim of research and research methodology

In this paper, the research aim is to support the use of effective model-based requirements engineering in the agile development of mechatronic systems, especially for the planning of engineering generations. For this purpose, the following research questions will be investigated:

1. What are the requirements of an approach for model-based requirements engineering regarding the planning of engineering generations in the agile development?
2. How should an approach for model-based requirements engineering in the agile development be characterized that supports the planning of engineering generations?
3. What benefits can be seen with the approach for model-based requirements engineering, especially for planning of engineering generations in the agile development?

The approach follows the Design Research Methodology (DRM) from Blessing and Chakrabarti [26]. The results of this paper are based on the work [27] and are supported by activities of the BMBF funded research project MoSyS (see chapter 8).

4. Requirements of model-based requirements engineering in the agile environment

4.1. Procedures and study design

To address the first research question, three studies are performed. The first study takes place as participant observation at a machine tool manufacturer in a work group on the topic of requirements engineering. In this group, mainly team- and department managers from different disciplines participate. During the work group, a common understanding is created about how cross-functional, customer-oriented requirements engineering is developed in agile development practice, which different disciplines are involved, and which challenges are associated with this. In a document study, an analysis regarding how requirements of product development projects are modeled in a tool-based manner is carried out. The usage of the requirement models in the agile way of working is also investigated. The third study takes place as participant observation in an agile development project for a machine tool automation system. Therein, it is analyzed in which way the planning of engineering generations can be supported within tool-based requirements engineering. In the examined project phase, the focus lies on the execution of test planning for the qualification of requirements after the first engineering generation is delivered.

4.2. Requirements concerning the approach being developed

The studies clearly show that requirements engineering is shifting from a document-based approach to a model-based approach that is already being implemented step by step.

Demands for the approach for model-based requirements engineering	Critical factor in MBSE implementation
The data model can formally represent the system of objectives.	Language
Planning for engineering generations can be mapped in the data model.	
Support continuous interdisciplinary collaboration in system of objectives development.	Tool
Considering the dynamics of the system of objectives through the product development process.	
Consideration of heterogeneous projects and processes.	Method
Supporting activities in system of objectives development, which are executed cyclically several times in the product development process.	
Support planning into engineering generations in the activities of the product development process.	

Fig. 1. Elaborated demands for the approach for model-based requirements engineering

More detailed analysis clarifies that the actors are insufficiently supported in modeling the requirements, which also hinders the usage of the models. There is a lack of supportive, needs-oriented views to provide an overview and establish consistency in the requirements model. Such views build the central basis for the developed approach. From the findings of the studies, the demands described in Fig. 1 can be summarized for an approach for model-based requirements engineering in agile development that supports the planning of engineering generations.

The approach is essentially based on the three aspects of MBSE language, method, tool, which are closely interrelated. For each of the derived demand, the critical MBSE factor to be considered to fulfill the requirement has been identified. As an example, to support continuous interdisciplinary collaboration, this must be considered in the methodology and a common modeling language must be available, but the critical factor is the tool-side implementation. Without this, the common language as well as method cannot be used.

5. Approach for model-based requirements engineering for planning engineering generation in agile development

The lightweight ontology developed for the requirements model in this approach contains the required classes and their relations (see Fig. 2). Here, existing ontologies from MBSE approaches are built upon (see e.g. [23, 24]) and a focus is set on the extension and detailing of classes and their relations for requirements modeling according to the identified demands. By implementing the ontology in a tool, the contents of the ontology may be extended, e.g. by specifying attributes, or restricted. The resulting model structure can be described as a heavyweight ontology, which has a higher degree of formalization. In particular, the assignment of elements to the addressed engineering generation is modeled.

To support the actors in requirements engineering, the contents of the requirements model are provided by user-friendly views of the model. Based on the assigned attributes for the description of requirement elements, various analyses can be presented in visualized form in the view. In order to use the views continuously in the development process and to fill them with further information of the requirements model, an adapted and extended method based on the combination of modeling frameworks and modeling activities described by Mandel et al. [25] is presented (see Fig. 3). Requirements engineering activities are identified by analyzing user stories from the perspective of involved users. An additional acceptance criterion is applied to the so-called activity pattern and is intended to ensure that the activity has a sufficient level of granularity. Fig. 3 shows an example of this for the activity 'Bundle requirements for an engineering generation'.

The approach describes the activity steps for the modeling process in an activity pattern. Depending on the user story or the aim of the model usage, the activity patterns can be reused

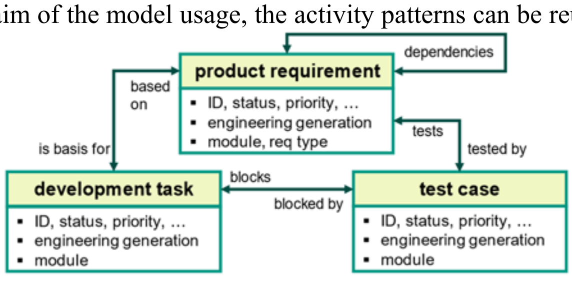

Fig. 2. Lightweight ontology for the developed approach (excerpt)

and combined. This maintains flexibility in the execution of the activity steps for different tasks in the product development process, which supports an agile way of working. According to the activity step of its activity pattern, the condensed view can be used for both analysis and synthesis of the requirements model. The deviation between the actual and target state can be systematically analyzed with the view. Information can be prepared in a way that it can function as a trigger for further activities, thus simplifying the analysis of the model content. For example, if a requirement has an insufficient description in terms of missing attributes or links, this can be explicitly presented as a trigger. In this way, the user is made aware of countermeasures. These and other synthesis activities can be supported by the developed view. It allows users to add or modify content of the requirement model.

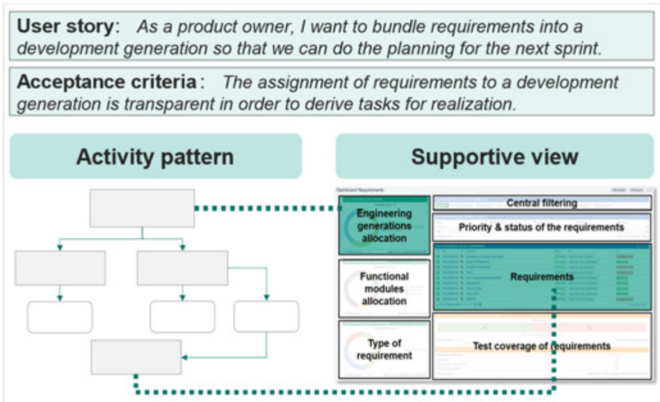

Fig. 3. Modeling method (the supportive view is shown enlarged in Fig. 4.)

The activity patterns in conjunction with the supportive views and the application of the specified modeling language support the user in continuously transferring the information of the requirements into a tool-based model. Such a procedure supports the aim of continuous validation in the product development process that was described earlier.

6. Implementation and evaluation of the approach

6.1. Implementation by test linkage

The presented approach is implemented in a development project for several user stories. For this, the tool JIRA from the vendor Altassian with the add-in R4J [28] is utilized. The three classes of product requirements, development tasks and test cases of the heavyweight ontology in JIRA are used to build the requirements model. Each class is described by attributes. For the description of the requirements, in addition to the standard attributes such as the status and the priority, appropriate attributes are implemented for the assigned engineering generations, the relevant modules and for the type of requirement. In the following, the implementation is exemplarily shown by the activity of linking requirements with test cases. The basis for executing the activity is provided by prepared activity patterns in combination with the view shown in Fig. 4 in form of a dashboard. The first step is to analyze the requirements that are not covered by a test case yet. Therefore, the filtering is set up the relevant engineering generation by the 'Central filtering' on the dashboard. In the area 'Test coverage of requirements', it is directly shown how many requirements have no coverage yet and the respective proportion assigned according to the type of requirement. Using this combination of attributes, the critical requirements for the development process can be identified, which supports the prioritization of most important tasks within the agile way of working.

To support the linkage of requirements and test cases, an additional view in terms of a two-dimensional traceability matrix is used. The requirements are displayed on the vertical axis and the test cases on the horizontal axis. By selecting a corresponding field in the matrix, the link between requirement and test case can be easily implemented. In order to adapt the scope of information as needed, filtering by attributes can help.

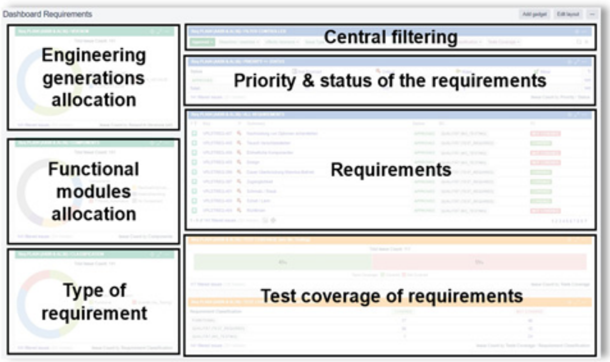

Fig. 4. Realization of the dashboard (left-hand side visualizes indicators for requirements engineering in the form of allocation to engineering generations, modules and type of requirements)

The presented procedure for the implementation of the views is exemplary. These can be extended for further processes by adding new attributes and adapting the views accordingly. Depending on the scope of the views, they can be specifically restricted by activity-based filter functions.

To address the descriptive study 2 of the applied DRM, two studies are performed. A data analysis on tool usage is carried out to evaluate the use of the approach. In a subsequent interview study, the utility of the approach is evaluated.

6.2. Evaluation of tool usage: data analysis

Within the analysis, data of the modeling activities, which were executed by the participants in the development project, was gathered. The modeling activities recorded in this case involve the linkage of the requirements with the related test cases. Multiple test cases can be assigned to one requirement or multiple requirements can be covered by one test case. A short briefing for using the supportive views was sufficient for the project participants to be able to carry out the modeling of requirements and, above all, their linkages by themselves, which indicates intuitive applicability. There are 83 requirements assigned to the given engineering generation, with 61 of these being testable requirements that must be verified by a test case during the further development process. Over a 2-month period, while test planning for the upcoming test phase was executed, data on modeling activities was collected. The results of our analysis indicate that the linkages evolved over time. Linkages include assigning additional test cases to the requirements or removing a link because the requirement cannot be verified by the planned test case. Only 10% of requirements have not been linked at all, while for 49% the linkage has been modified at least twice indicating that the implemented approach is been used actively.

This shows that linkages between requirements and test cases could be realized in a model-based manner that allows dynamically evolving the modeled links. This shows a productive usage of the approach in the development practice.

6.3. Evaluation the utility of the approach: interview study

An interview study with experts from the R&D and testing departments allowed for evaluating the utility of the approach. One partial result is the experts' assessment to the statements shown in Fig.5. The experts confirm that the supporting views provide a positive contribution to agile requirements engineering for planning in engineering generations.

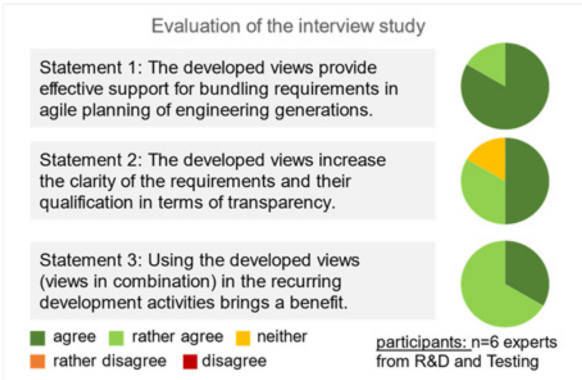

Fig. 5. Assessment of experts regarding the statements

Furthermore, additional insights could be gathered in the interviews. The participants agree that the approach offers useful support for the planning of engineering generations. According to the experts, the views serve as a central source of information and provide a communications basis that promotes interdisciplinary collaboration. Increased traceability of requirements to test cases is generally rated as positive. Comments from the experts reveal that this depends highly on the quality of the described requirements. In the experts' opinion, the use of the views can be integrated very well, particularly in recurring milestones, to be able to represent the current project status and product maturity.

7. Discussion and outlook

The presented MBSE approach offers the potential to handle complex engineering through engineering generations in a systematic and traceable way. A common, consistent language for requirements engineering is the basis for the use of the models. Activity-based views with an integrated filter function allow the handling of systematically bundled information and serve as a basis for communication. Recurring activities for both, analysis and synthesis of the requirements model are supported step-by-step by appropriate views. Based on the findings of this work, the presented activity patterns turn out to be a target-oriented way to support the user in modeling. Up to now, the activity patterns have been described independently of each other, but the activities in the product development process are not separate from each other. Therefore, future research should investigate how a connection between these individual activity patterns can be implemented. A first approach might be to make an output of an activity as trigger for the next activity to be initiated in the next sprint. In the presented approach, activity-based views are used. A framework for different views is investigated in the BMBF funded research project MoSyS.

So far, the presented work focuses on product requirements. To extend the approach for planning test cases and the test environment, the ontology must be extended accordingly. An ontology for classes and relations to describe validation activities is given by Mandel et al [24]. It is necessary to investigate how the elements of the ontology can also be integrated in the presented approach. A special focus should lie on the definition or extension of activity patterns for specific tasks in the validation context, e.g., for definition of test cases.

To support trainings on languages, methods, and tools of MBSE, the presented approach with the activity-based views and the assigned activity steps of the activity patterns offers great potential. In this way, individual activities of the modeling can be specifically instructed in connection with a view. An MBSE training concept based on views and activity patterns is already subject of further research.

8. Acknowledgment and references

This article is part of the research project MoSyS - Human-Oriented Design of Complex Systems of Systems. This project is funded by the German Federal Ministry of Education and Research (BMBF) within the "The Future of Value Creation – Research on Production, Services and Work" program (funding number 02J19B106 - 02J19B090) and managed by the Project Management Agency Karlsruhe (PTKA). The authors are responsible for the content of this publication.

[1] Automotive Agile PEP - Survey Report - we.conect Global Leaders GmbH, 2018. *Perspectives of Agile Product Development Processes in the Automotive Ecosystem*.

[2] Walden, D.D., Roedler, G.J., Forsberg, K.J., Hamelin, R.D., Shortell, T.M., 2015. *Systems Engineering Handbook - A Guide For System Life Cycle Processes And Activities*. John Wiley & Sons, Inc, Hoboken, NJ, USA.

[3] Albers, A., Sadowski, E., Marxen, L., 2011. A New Perspective on Product Engineering Overcoming Sequential Process Models, in *The Future of Design Methodology*, Springer-Verlag London Limited, London, p. 199.

[4] Albers, A., Bursac, N., Wintergerst, E., 2015. Produktgenerationsentwicklung – Bedeutung und Herausforderungen aus einer entwicklungsmethodischen Perspektive, in *Stuttgarter Symposium für Produktentwicklung (SSP)*, Fraunhofer Verlag, Stuttgart.

[5] Albers, A., Rapp, S., Birk, C., Bursac, N., 2017. Die Frühe Phase der PGE – Produktgenerationsentwicklung, in *Stuttgarter Symposium für Produktentwicklung 2017*.

[6] Wessels, H., Heimicke, J., Rapp, S., Grauberger, P. *et al.* Sprint planning in mechatronic system development based on reference system elements, in *17. Gemeinsames Kolloquium Konstruktionstechnik*.

[7] Ries, E., 2011. *The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. Currency.

[8] Schuh, G., Dölle, C., Schloesser, S., 2018. Agile Prototyping for technical systems–Towards an adaption of the Minimum Viable Product principle.

[9] Schwaber, K., Sutherland, J., 2020. *The Scrum Guide*.

[10] Albers, A., 2010. Five Hypotheses about Engineering Processes and their Consequences, in *Proceedings of the TMCE 2010*.

[11] Cross, N., 2008. *Engineering design methods: strategies for product design*.

[12] Bursac, N., Rapp, S., Waldeier, L., Wagenmann, S. *et al.*, 2021. Anforderungsmanagement in der Agilen Entwicklung Mechatronischer Systeme - ein Widerspruch in sich?, in *Entwerfen Entwickeln Erleben in Produktentwicklung und Design 2021*, Prof. Dr.-Ing. habil Ralph H. Stelzer, Prof. Dr.-Ing. Jens Krzywinski, p. 283.

[13] Delligatti, L., 2014. *SysML distilled: A brief guide to the systems modeling language*. Upper Saddle River, NJ; Munich [u.a.] Addison-Wesley.

[14] Horber, D., Wilking, F., Schleich, B., Wartzack, S., 2021. Durchgängige Systemmodellierung - Integration von Zielen in technische Systemmodelle im Kontext des Anforderungsmanagements, in *Stuttgarter Symposium für Produktentwicklung SSP 2021*, Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO, Stuttgart, p. 163.

[15] Gomez-Perez, A., Fernández-López, M., Corcho, O., 2004. *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer-Verlag, London.

[16] Bone, M., Cloutier, R., 2010. The Current State of Model Based Systems Engineering: Survey Results from the OMG SysML Request for Information 2009, in *8th Conference on Systems Engineering Research - CSER 2010*.

[17] Cloutier, R., 2019. 2018 Model Based Systems Engineering Survey: Conducted December 2018, in *Proceedings of the 2019 INCOSE MBSE Workshop*.

[18] Dumitrescu, R., Albers, A., Riedel, O., Stark, R., Gausemeier, J., 2021. *Advanced Systems Engineering - Value Creation in Transition: Engineering in Germany – Status quo in Business and Science*, Paderborn.

[19] Chami, M., Aleksandraviciene, A., Morkevicius, A., Bruel, J.-M., 2018. Towards Solving MBSE Adoption Challenges: The D3 MBSE Adoption Toolbox, in *28th annual INCOSE international symposium*, p. 1463.

[20] Albers, A., Zingel, C., 2013. Challenges of Model-Based Systems Engineering: A Study towards Unified Term Understanding and the State of Usage of SysML, in *Smart Product Engineering: Proceedings of the 23rd CIRP Design Conference*.

[21] Eigner, M., Dickopf, T., Schneider, M., Schulte, T., 2017. MECPRO² - A holistic concept for the model-based development of cybertronic systems, in *21st International Conference on Engineering Design (ICED17): Vol. 3: Product, Services and Systems Design*.

[22] Pohl, K., Broy, M., Daembkes, H., Hönninger, H., 2016. *Advanced Model-Based Engineering of Embedded Systems*. Springer International Publishing, Cham.

[23] Holt, J., Perry, S., 2013. *SysML for Systems Engineering – 2nd Edition: A model-based approach*. The Institution of Engineering and Technology.

[24] Mandel, C., Böning, J., Behrendt, M., Albers, A., 2021. A Model-Based Systems Engineering Approach to Support Continuous Validation in PGE - Product Generation Engineering, in *2021 IEEE International Systems Conference*.

[25] Mandel, C., Behrendt, M., Albers, A., 2021. Nutzer- und Aufgabengerechte Unterstützung von Modellierungsaktivitäten im Kontext des MBSE-Model-Based Systems Engineering, in *Entwerfen Entwickeln Erleben in Produktentwicklung und Design 2021*, Prof. Dr.-Ing. habil Ralph H. Stelzer, Prof. Dr.-Ing. Jens Krzywinski, p. 727.

[26] Blessing, L.T.M., Chakrabarti, A., 2009. *DRM, a Design Research Methodology,* 1st edn. Springer Publishing Company, Incorporated, London.

[27] Windisch, E., 2021. *Ansatz zur Modellbasierten Zielsystementwicklung zur Planung von Entwicklungsgenerationen in der agilen Entwicklung mechatronischer Systeme (unpublished master thesis)*, Karlsruhe.

[28] R4J - Requirements Management for JIRA. easesolutions.atlassian.net/wiki/spaces/REQ4J/overview.