59th ILMENAU SCIENTIFIC COLLOQUIUM
Technische Universität Ilmenau, 11 – 15 September 2017
URN: urn:nbn:de:gbv:ilm1-2017iwk-022:8

A METHOD FOR EFFICIENT AND TASK ORIENTED CONFIGURATION OF VIRTUAL REALITY (VR) MODELS FOR THE ANALYSIS OF TECHNICAL SYSTEMS

Atif Mahboob[1], Andreas Liebal[2], Stephan Husung[3], Christian Weber[1], Heidi Krömker[2]

[1]Institute for Design and Precision Engineering, Technische Universität Ilmenau
[2]Institute for Media Technology, Technische Universität Ilmenau
[3]:em engineering methods AG

ABSTRACT

The development of products for today's competitive and rapidly changing market demands the incorporation of digital methods and new technologies. An early evaluation of a product can help the product designer to gain better understanding of the product behaviour in its later life phases. In this paper, we will focus on building the product's life phase specific models in Virtual Reality (VR) for product evaluation. These will also include the life phase specific actor(s) and environment for the product. A new method based on an MBSE approach will be presented for the description of VR models that can help to reduce the effort needed to create them. The behaviour modelling process based on SysML will be explained using an example model. Moreover, a complete overview of the VR model configuration, its execution and interaction possibilities will be brought to light.

Index Terms **–** Product evaluation, Virtual Reality, Model Based Systems Engineering, SysML

1. INTRODUCTION

It is of great importance to consider the use-cases and related requirements of all stakeholders of a product in its different life phases during the design process. Designers today try to take into consideration the use-cases and requirements of different stakeholders and develop a solution in accordance with these requirements. According to the guideline VDI 2221[1] different life phases of a product are product planning, product development, production planning, production, distribution, use, service and disposal. Against these life phases there are different stakeholders/actors, e.g. product planner, product designer, production planner, production personal, user(s), service technician and disposal personal, respectively. In a similar way, there are different environments that a product is confronted with in its different life phases. Early evaluation of a product's later life phases can help the designer to get a better understanding of the necessary product's properties/behaviour and can help to identify new (i.e. not yet considered) requirements. The products today are complex and multidisciplinary as compared to their predecessors. The increasing competition between manufacturing companies and the decreasing time to the market are pushing companies towards the incorporation of new technologies and use of digital methods. The early evaluation of a product's later life phases can indeed contribute to the reliability of the product and can help to identify shortcomings in the design. However, the early evaluation can be a difficult task, as it requires the knowledge about the later life phases of the product along with the knowledge about possible actor(s) and environments. This requires creative imagination and depends greatly on the expertise of the product designer and can be prone to errors. Especially for the development of complex mechatronic products the use-cases and the requirements are difficult to imagine [2]. Mostly, the designer has to take up the role of an actor and has to imagine the later life phase situation. Therefore, there is a need for new methods and tools that can support the product designer in this highly complex early evaluation process.

Virtual Reality (VR) technology can be of great help to support this early evaluation. It can help the designer to analyse the product in its context (life phase specific actor(s) and environment) and verify the respective product behaviour. However, the preparation of appropriate VR models is currently based on complete and time-consuming programming of the whole scene in advance. This limits the application of VR in the industry today and is mainly due to the following reason:

- the great amount of time and the effort needed for the preparation of virtual models,
- poor/limited possibilities to reuse these models and/or parts of them and
- limited modification possibilities i.e. a small change can force a complete new preparation.

In order to tackle these issues, we are focusing on a new approach that divides the complete VR scene into three sub models, i.e. product, actor and environment [3]. These models have structural as well as behavioural aspects. The systematic recombination of these sub-models can make the preparation of new VR scenarios easier.

In order to have uniformity, in this paper the same terminologies are used as in [3]:

- "Context" – a specific combination of environment and actor (s) in a specific product life phase
- "Use-case" – a specific demand and interaction of environment and actor(s) with the product
- "VR-user" – the product designer
- "Environment" – the surroundings of a product in a particular life phase

The product along with its life phase specific environment and the actor(s) inside VR builds context based product evaluation scenarios (see figure 1). An observation of these scenarios can help the product designer to better understand the requirements of the product and its behaviour in later life phases.

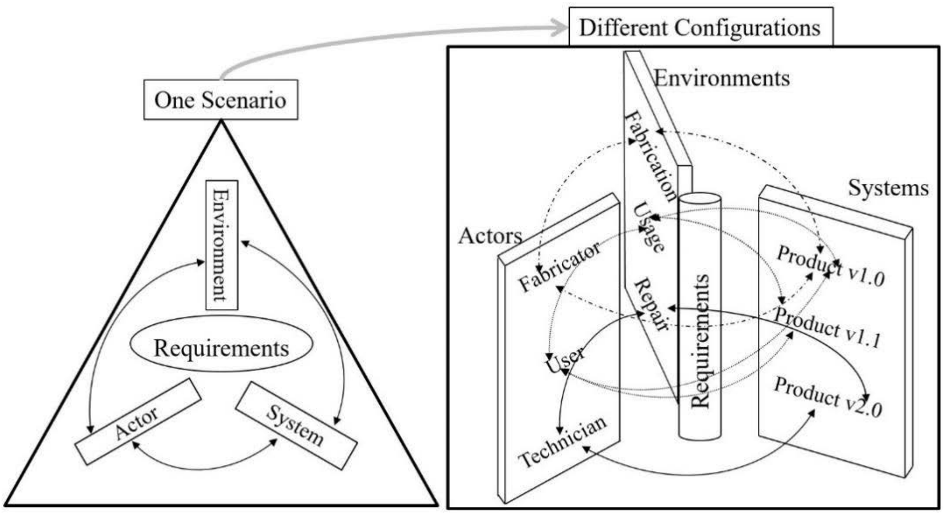

Figure 1: Scenario building in a model based approach for VR [3]

A product can have different actors and environments in its different life phases. The actors require different use-cases with the product in the specific environments. Figure 2 shows two of these use-cases. In each use-case, different aspects of the product are evaluated. In the use phase, the user is the actor, the fulfilment of the desired product functionality along with ease-of-use criteria will be most important. In the service phase, the needs of a service technician have to be considered, e.g. the ease of finding faults and the effort to replace broken or worn-out components.

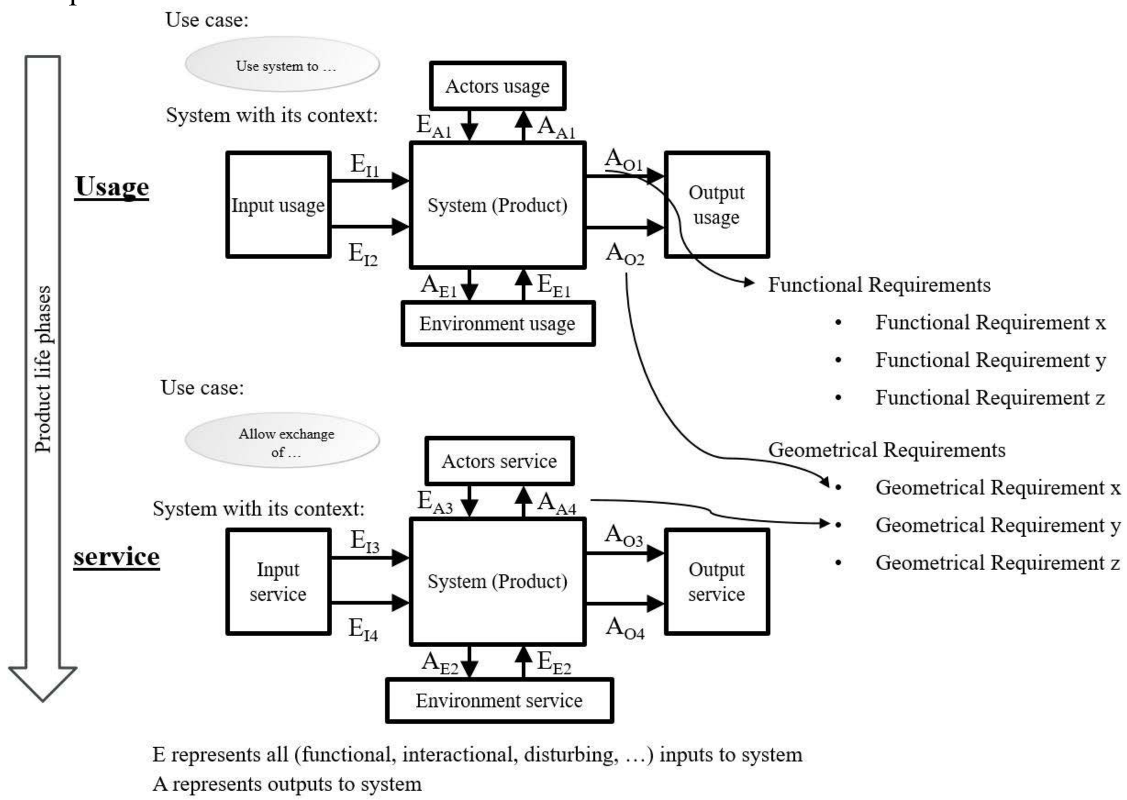

Figure 2: Product context in different product life-phases [3]

Figure 2 is in principle based on the formal representation of the relational properties introduced by [4] and further developed by [5] with the addition that the environment and the actor(s) are to be modelled independently from each other to achieve reusability. For the implementation of this model, there is a need of a new approach that can model the sub-models independently from each other – which in turn requires thorough definitions of the interactions between them. Model Based Systems Engineering (MBSE) with Systems Modelling Language (SysML) claim to be capable of doing just that, therefore we have decided to use (and test) this approach.
This paper aims to answer the following questions:

1) How can the sub-models be described independently from each other?
2) How can the interactions between the sub-models be modelled using SysML?
3) How can SysML models be used as core of the simulation process and cooperate with VR-software to achieve an interactive simulation in VR?
4) How can the physical behaviour of product, actor(s) and environment be incorporated inside VR – which needs real-time conditions?

2. RELATED WORK

Various methods and digital models are developed to support the product development process. These methods mainly focus on the product and its behaviour. The life phase specific actor(s) and environment build the context of a product and also play an important role. The use of VR can greatly support the product development process by realising a later life phase situation of a product with its context. [6] uses VR and focuses on a user centred approach to reduce the uncertainties in the development process. The work also brings the usefulness of VR into light as it can help to analyse designs by means of product use-case scenarios inside VR. [7] proposes a new methodology for virtual product development to analyse the dynamic and acoustic behaviour of the product by means of virtual prototypes. The use of VR for production lines is investigated by [8] by means of a simulation in VR. This work is very relevant as it uses SysML for behavioural descriptions in VR and also discusses the challenges for integration of a simulation tool in VR. [9] emphasises the use of the systems engineering approach for the development of today's multidisciplinary systems. Recently, two industrial projects (Fas4M and mecPro2) have focused on the use of SysML as unified language for the implementation of MBSE approaches. The Fas4M[1] [10] [11] project implements an interface between SysML and a CAD tool to support the modelling and development of mechatronic systems. The project mecPro2[2] works to develop the cooperation between MBSE and a Product Lifecycle Management (PLM) system and concludes that a combination of both can handle the complexity of the system under development [12] [13] [14]. The work of [15] recommends the use of SysML for the development of multidisciplinary mechatronic systems and explains the modelling process using a washing machine as an example. Another work [16] discusses behavioural modelling in SysML and explains the behaviour diagrams in detail by using an audio player as an example. Furthermore, there are already studies available that use model based approaches to realise simulations in VR. For example, [17] develops a framework based on modelling in UML and then uses transformations to integrate VR simulations; [18] develops process simulation in VR through the integration of VR and SysML. The related work shows the use of VR in product development and MBSE methods developed to support the product development process. However, there is still a need for a method to facilitate the use of VR for the early evaluation of a product; therefore, the method developed in this work will be presented in the next sections 3 and 4.

3. BEHAVIOUR MODELLING IN SYSML

The motivation for developing this method is to support the product designer by enabling him/her to have a look into later life phases of a product as explained in the introduction and elaborated in detail in [3]. The focus in this paper lies in the description and the preparation of a VR model using SysML as the description language. The sub-models of the VR-scene, i.e. product, actor and environment, have individual structural as well as behaviour descriptions. The structural description is usually available in the form of geometrical models that can be directly imported into the VR-software. However, the normal way of describing the behaviour of these models is based on simple programming that is time consuming and offers very limited reusability. Therefore, we are focusing on a model based approach with SysML as description language for the sub-models. In this section, the modelling process will be explained considering a vacuum cleaner as a product, a human user as actor and a living room as the environment. These three models are modelled using SysML, thus including the standard SysML diagrams, i.e. structural, behavioural and requirement diagrams. Figure 3 shows the Block Definition Diagram BDD that is used to model the structure inside SysML and describes different components/parts of a system. It is to notice that the structural description in SysML describes the hierarchical structure of a system and may be different from the geometrical (CAD etc.) description. The figure depicts the "User_Produ_Env Interaction" that has three sub-models defined in SysML as a structural element, i.e. "Block". Each block can have sub-parts as well as needed parameters to describe it. For example, in order to describe a vacuum cleaner the position of the cleaner, its current state and its weight can be important. However, for the sake

[1] Functional Architectures of Systems for Mechanical Engineers: http://fasform.de/content/ [last accessed on 26.07.2017]

[2] Modellbasierter Entwicklungsprozess cybertronischer Produkte und Produktionssysteme: https://www.mecpro.de/ [last accessed on 26.07.2017]

of clarity for the reader and to keep the discussion focussed, these sub-parts are not shown in figure 3. Furthermore, the sub-models have further sub-components that are not depicted in this diagram, e.g. product model has sensors, wheels, a handle, cables, etc. Similarly, the environment has a sofa, table, lamp, walls, etc. The actor model in same way has different structural components/parameters e.g. height, weight, age, etc. They are not discussed in this paper.

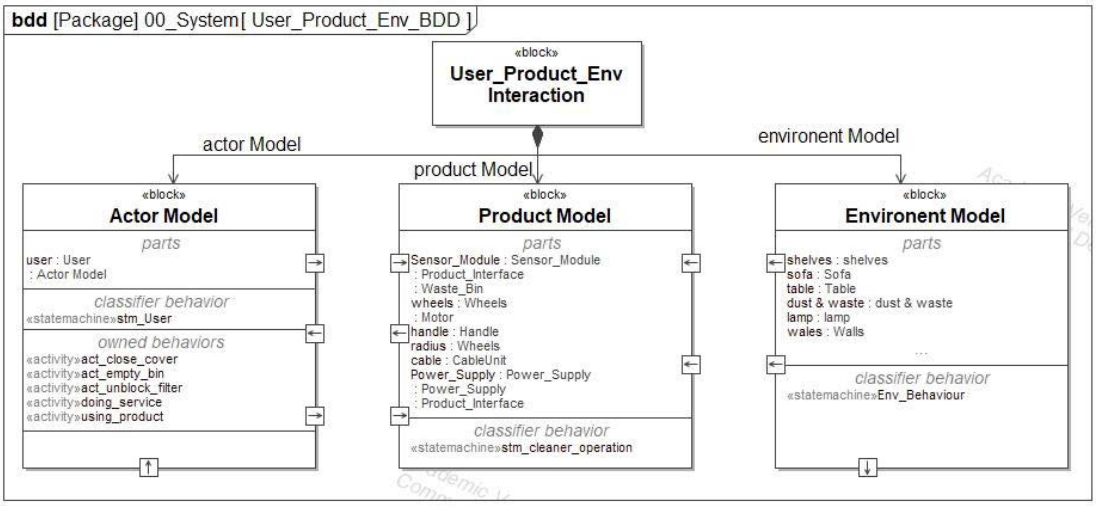

Figure 3: Block Definition Diagram (BDD) for an example model

Other important aspects that are indicated in this diagram are the interaction points, i.e. “Ports” in SysML. Ports define the interaction between two blocks in SysML and facilitate the flow of signals, values, behaviours, etc. between two blocks. However, figure 3 does not define which block is interacting with which block and how. This is depicted by another structural diagram in SysML – the “Internal Block Diagram” (IBD) – which shows the internal structure of a block and links between the blocks. For the vacuum cleaner example this is shown in the next figure.

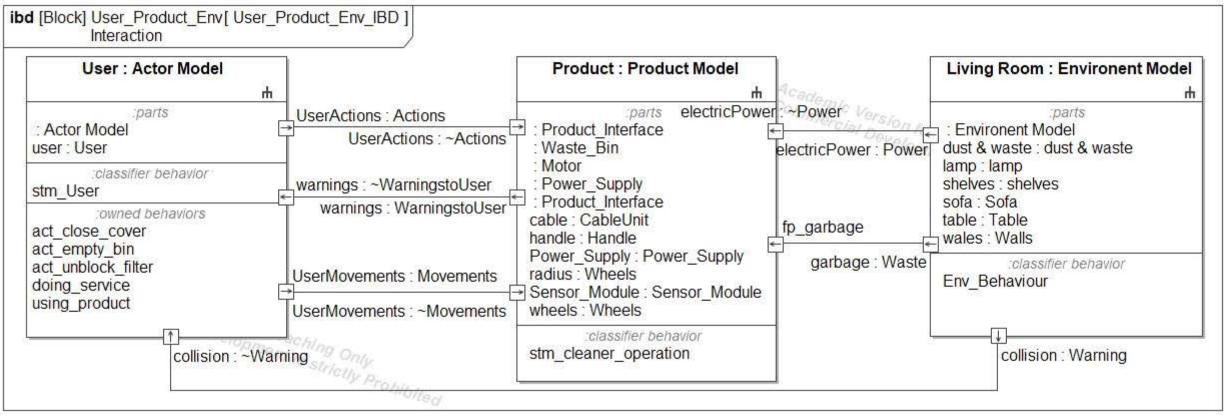

Figure 4: Internal Block Diagram (IBD) for an example model [2]

Figure 4 shows the links between ports of different blocks. The port “UserActions” communicates the actions performed by the user to the product, e.g. connecting the vacuum cleaner to the power supply, switching it on/off, etc. The port “UserMovements” communicates the movements performed on the product by the user, e.g. moving the vacuum cleaner forward, backward, turning, etc. The port “electricPower” indicates that the power needed by the product for completing its operation is coming from the living room, e.g. it can be an electric plug in a wall. We are interested in the decomposition of the use-cases into individual activities and possible states. In the following, the target behaviour is described (without entering the internal elements in detail) and the use of the ports in the behaviour description will be discussed. Figure 5 shows a State Machine Diagram (STM) that describes the different states that a system can find itself in and is a behaviour diagram expressed via SysML. For example, a vacuum cleaner can be on idle and can be waiting for the plug connected to the power supply. Once the power is supplied, it is waiting for a start signal (e.g. press of a button) and so on. A very important aspect to notice in this diagram is the indication of the port that will receive the actions (signals) coming from the user; in our case PowerPlugged, SwitchOn and SwitchOff are the signals from the model of actor in SysML and are communicated to the product model over "UserAction" port (figure 4).

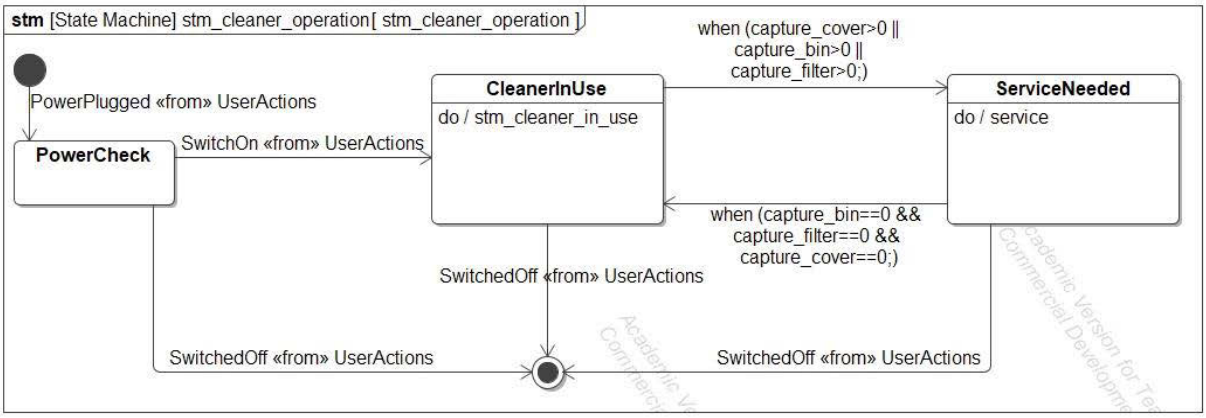

Figure 5: State Machine Diagram (STM) of a Vacuum Cleaner

In order to demonstrate the behaviour modelling, the authors would like to explain one state that has a sub-behaviour (here as a STM diagram). As indicated in figure 5, once the power button is pressed and the product is switched on it goes into the state "CleanerInUse" that has further sub-states (sub-behaviours) as indicated in figure 6. The vacuum cleaner can be moving or turning or be waiting for someone to move it (in the case of manual vacuum cleaner – for a robot cleaner this will be different).

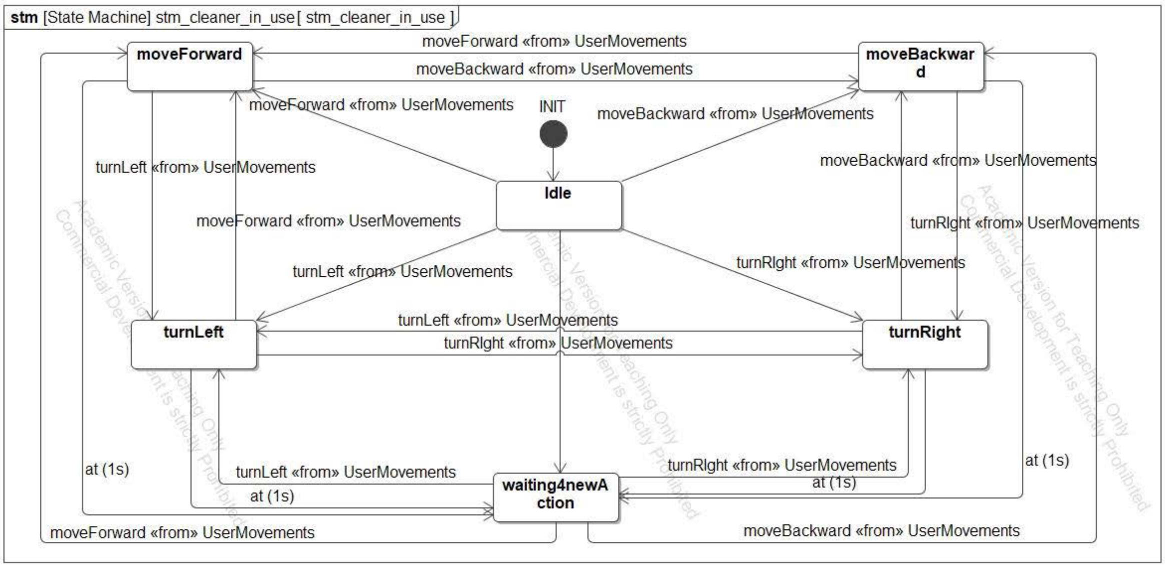

Figure 6: Sub-states of a vacuum cleaner during "CleanerInUse"

Similar to figure 5, figure 6 also shows that the signals (moveForward, moveBackward, turnLeft and turnRight) are received over the port "UserMovements". This kind of interaction through the port is essential for SysML models to be reusable, for example in our application: If the actor model (shown in figures 3 & 4) is replaced with a new model, the interaction will still

function properly – if the ports are the same in the new actor model as in the old one and if the signals expected by the product model are available.
SysML models – usually drawn up for complex systems – show ***that*** systems and their components have behaviour, they also show what causes and what is caused by the behaviour. However, SysML models do not contain information about ***how*** systems and components behave: In order to represent the details of the behaviour, additional simulation is needed, either via enhanced SysML models or with external simulation coupled to the SysML model. In a complex system, a SysML model will show the logic/sequence of simulations.

4. IMPLEMENTATION METHOD AND SIMULATION IN VR

For VR applications behaviour descriptions have to be executable (even in real-time at that). However, SysML was initially not thought to be an executable language. So how to come from models described via SysML to executable VR models?
There are considerable efforts already made to make SysML diagrams and models executable. There are already a couple of methods available that either transform the SysML models to another executable language/tool or try to achieve simulation by integrating SysML models into other executable tools [19] [20] [21] [22] [23] [24] [25]. Furthermore, there are already a couple of commercial tools that can execute the behaviour models of SysML. Cameo Systems Modeller (CSM) is one of those tools and it uses functional UML (fUML) based on token flow. Furthermore, CSM also utilises external solvers like Matlab, Maple, Modelica, Javascript, Jython and many more for the calculation of complex simulations. The same tool was used to carry research presented in this paper. CSM makes the SysML models executable and offers a Java based open Application Programming Interface (API) to extend this tool.
In order to integrate the SysML description and to use it for simulation in VR, we wrote a plugin for CSM that can communicate with VR and make the simulation in VR possible. This approach has an advantage over the approaches based on the transformation as it offers an integrated solution for modelling as well as execution of VR models inside CSM. The functionality of this plugin as well as a complete overview of the simulation process is described in figure 7.

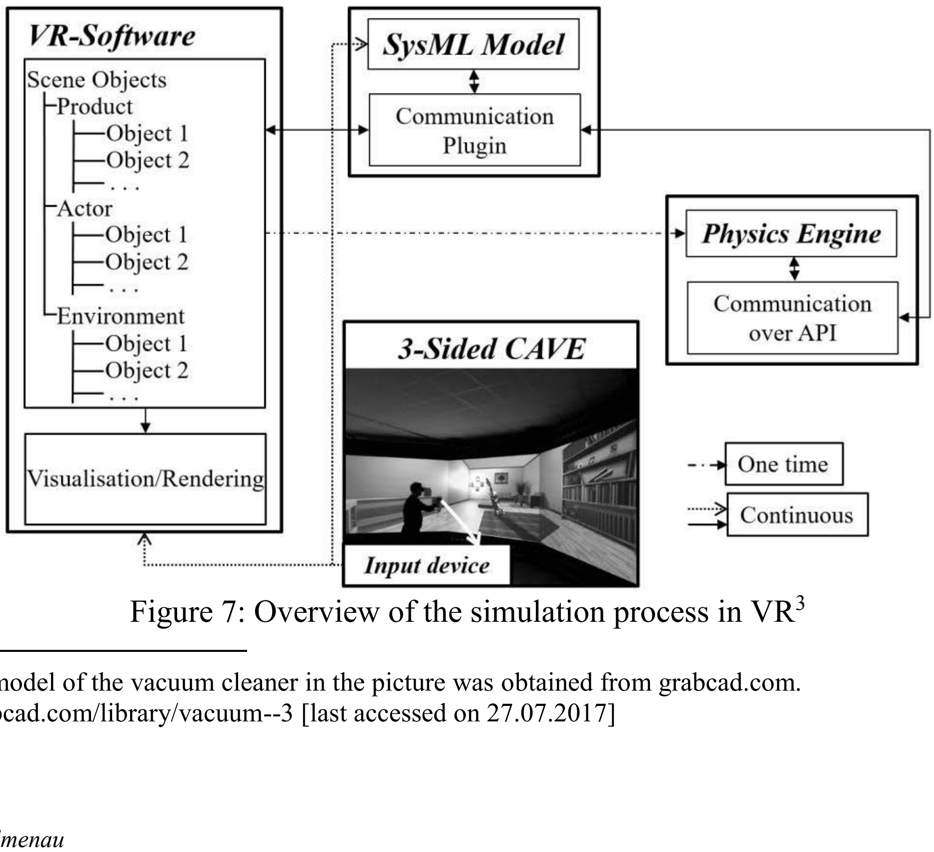

Figure 7: Overview of the simulation process in VR[3]

[3] The CAD model of the vacuum cleaner in the picture was obtained from grabcad.com. https://grabcad.com/library/vacuum--3 [last accessed on 27.07.2017]

The VR-software includes the geometrical models of product, actor and environment in the form of a VR-scene that can be visualised using a 3-sided CAVE type VR-system available at Technische Universität Ilmenau. The behaviour model developed with the help of SysML is used to simulate the objects inside the VR-scene. The Communication Plugin helps to communicate the real-time execution information from the SysML behaviour model to the VR-system that can be visualised in the CAVE. Also, the changes in the objects' properties as a result of scene manipulation from the VR-user, are received by this plugin. The currently available VR-software tools are only suitable for the visualisation of geometric objects and do not include any physical calculation. Therefore, to incorporate physical behaviour into the simulation, we are using Virtual Robot Experimentation Platform[4] (v-rep). V-rep offers a multi-physics framework for physical calculations and all the objects defined inside it are accessible over a multilingual API. Furthermore, it is easy to create the scene inside the v-rep using the objects from the VR-software with the same positions and orientations, etc. The Flystick (our main input device) is used to interact with the VR-system. In this way, the interaction of product, actor and environment can be viewed from different angles and points of view. The device can also allow the manipulation of the objects in the scene, e.g. hiding/un-hiding certain objects in the scene or changing their positions and orientations.
The simulation in VR will depend on the interplay of the tools mentioned in figure 7. The visualisation in the CAVE will depend on the objects in the VR-software and these objects' properties (position, orientation, …) will depend on the behaviour model in SysML. The SysML model works as a central information model which communicates on one side with the VR-software and on the other side with the physics engine (v-rep). The information about the objects will be sent to v-rep during an active simulation both in SysML and v-rep. The changes in v-rep simulation model against the information received from SysML will also be sent back to the SysML model. Hence, the VR-system will be updated by the SysML description based on the new information received from v-rep. The physical properties of the objects will be continuously read from v-rep, will be updated and sent to the VR-software which, in turn, will also update the visualisation in the CAVE. In this way, the presented approach can lead to a fast and easy configuration/preparation of complex VR models that are reusable as well. In the next section, the advantages and the limitations of the implementation method proposed in this paper will be discussed and conclusion will be presented.

5. DISCUSSION AND CONCLUSION

To summarise: Section 1 introduces the model based approach for the context based product evaluation using VR technology, and section 2 gives an overview of the most relevant related work done in this field. Section 3 explains the principles of behaviour modelling inside SysML. The interaction of sub-models over generic interfaces, i.e. ports, is shown. Section 4 discusses in detail how these SysML models can be enhanced by simulating behaviour as is required to control the VR model deduced from SysML descriptions.
The idea of separating the complete VR model into product – actor – environment is the main approach to enhance reusability of VR models. MBSE and SysML are tools that make the realisation of the idea possible by describing the sub-models independently, the interactions between these and (qualitatively) their behaviour. However, there are still severe problems, as was shown: To make these models (or any other type of model) "alive", SysML is not sufficient alone, in addition we need simulation (in our case a physics engine is used). It can be done (i.e. using MBSE and SysML for our purpose) – but only if and when simulation can be added to

[4] http://www.coppeliarobotics.com/ [last accessed on 26.07.2017]

and controlled by the MBSE/SysML models. In order to establish the presented approach one possible realisation has been shown in this paper.
In this way, the presented method can help the designer by making VR model configuration and modification efficient, thus supporting early evaluations of a product possible by means of use-case scenarios in VR.
However, in its current form the implementation of this method does not include the actor model because it is currently still under development. Also, so far only a very basic implementation of the communication between SysML, the physics engine and VR is achieved. These issues are in the focus of current and future work in the project.

ACKNOWLEDGEMENT

The authors would like to thank the Deutsche Forschungsgemeinschaft (DFG) for supporting this work under the funding label KR 3297/3-1 and WE 1989/6-1 within the research project "user and task oriented virtual model for product development"

REFERENCES

[1] VDI 2221, "VDI-Guideline 2221: Systematic Approach to the Design of Technical Systems and Products", VDI, Düsseldorf, 1987
[2] M. Ricks, M. Engelmann, M. Schwab, S. Kleiner and S. Husung, “Model-Based Systems Engineering and Simulation for Automotive Systems Development at GKN Driveline”, NAFEMS World Congress 2017, Stockholm, Sweden
[3] A. Mahboob, C. Weber, S. Husung, A. Liebal, and H. Krömker, “Model based systems engineering (MBSE) approach for configurable product use-case scenarios in virtual environments”, ICED17-21st International Conference on Engineering Design, University of British Columbia, Vancouver, Canada, 21-25 August 2017.
[4] M.M. Andreasen, and N.H. Mortensen, "Basic thinking patterns and working methods for multiple DFX", Proceedings of the 7th Symposium "Design for X", Universität Erlangen-Nürnberg, 1996/97, pp. 7-12
[5] C. Weber, “Looking at 'DFX' and 'Product Maturity' from the Perspective of a New Approach to Modelling Product and Product Development Processes”, The Future of Product Development (ed. by Krause, F.-L.), pp. 85-104. Springer, Berlin-Heidelberg 2007
[6] S. Metag, S. Husung, H. Krömker, and C. Weber, "User-centered Design of Virtual Models in Product Development", Proc. of 53rd IWK, Ilmenau, 2008.
[7] C. Weber, and S. Husung, "Virtualisation of product development/design – seen from design theory and methodology", DS 68-2: Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through Engineering Design, Vol. 2: Design Theory and Research Methodology, Lyngby/Copenhagen, Denmark, 15.-19.08. 2011.
[8] M.A. Abidi, B. Lyonnet, P. Chevaillier, and R. Toscano, "Contribution of Virtual Reality for Lines Production's Simulation in a Lean Manufacturing Environment", International Journal of Computer Theory and Engineering, 8(3), p.182, 2016.
[9] J. Gausemeier, R. Dumitrescu, D. Steffen, A. Czaja, O. Wiederkehr, and C. Tschirner, “Systems Engineering in der industriellen Praxis“, Heinz Nixdorf Institut (2013)
[10] G. Moeser, C. Kramer, M. Grundel, M. Neubert, S. Kümpel, A. Scheithauer, S. Kleiner, and A. Albers, “Fortschrittsbericht zur modellbasierten Unterstützung der Konstrukteurstätigkeit durch FAS4M”, Tag des Systems Engineering. München : Carl Hanser Verlag GmbH & Co. KG, 2015, S. 69–78, 2015.

[11] G. Moeser, M. Grundel, T. Weilkiens, S. Kümpel, C. Kramer, and A. Albers, “Modell-basierter mechanischer Konzeptentwurf: Ergebnisse des FAS4M-Projektes”, Tag des Systems Engineering. München : Carl Hanser Verlag GmbH & Co. KG, S. 417–428, 2016.
[12] M. Eigner, T. Dickopf, and H. Apostolov, “System Lifecycle Management - An Approach for Developing Cybertronic Systems in Consideration of Sustainability Aspects”, Procedia CIRP 61, S. 128–133, (2017).
[13] M. Eigner, C. Muggeo, T. Dickopf, and K. G. Faißt, “An Approach for a Model Based Development Process of Cybertronic Systems”, Proceedings of the 58th Ilmenau Scientific Colloquium. Ilmenau, 2014.
[14] H. Meissner, M. Cadet, N. Stephan, and C. Bohr, “Model-Based Development Process of Cybertronic Products and Production Systems”, Advanced Materials Research (Vol. 1018, pp. 539-546). Trans Tech Publications, 2014.
[15] M. Follmer, P. Hehenberger, S. Punz, and K. Zeman,” Using SysML in the product development process of mechatronic systems”, DS 60: Proceedings of DESIGN 2010, the 11th International Design Conference, Dubrovnik, Croatia, 2010
[16] R. Silhavy, P. Silhavy, Z. Prokopova, “Behavioral modeling in system engineering”, Proceedings of the 13th WSEAS international conference on Automatic control, modelling & simulation, S. 100–105, 2011.
[17] P. Chevaillier, T.-H. Trinh, M. Barange, P. de Loor, F. Devillers, J. Soler, and R. Querrec, “Semantic modeling of virtual environments using mascaret”, 5th Workshop on Software Engineering and Architectures for Realtime Interactive Systems (SEARIS), 2012, S. 1–8
[18] M.-A. Abidi, P. Chevaillier, B. Lyonnet, M. Kechiche, P. Baert, and P. Toscano, “How to Create a New Generation of Industrial Processes Simulation by Coupling Simulation Tools with VR Platforms”, 28th International Conference on Computer Applications in Industry and Engineering (2015)
[19] O. Batarseh, and L. F. McGinnis, “System modeling in sysml and system analysis in arena”, Proceedings of the Winter Simulation Conference, 2012, S. 258
[20] D. Bank, F. Blumrich, P. Kress, and C. Stoferle, “A systems engineering approach for a dynamic co-simulation of a SysML tool and Matlab”, 2016 Annual IEEE Systems Conference (SysCon), S. 1–6
[21] R. Bültermann, and M. Brandstätter, “Austausch von Modellen zwischen SysML und Modelica zur Simulation”, Tag des Systems Engineering. München: Carl Hanser Verlag GmbH & Co. KG, S. 289–298, 2014.
[22] D. Foures, V. Albert, J.-C. Pascal, and A. Nketsa, “Automation of SysML activity diagram simulation with model-driven engineering approach”, Proceedings of the 2012 Symposium on Theory of Modeling and Simulation-DEVS Integrative M&S Symposium, 2012, S. 11
[23] J. Gross, and R. Mukherjee, “Integrating Multibody Simulations with SysML”, Proceedings of the ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference: American Society of Mechanical Engineers, S. V006T10A038, 2015.
[24] W. Schamai, P. Fritzson, C. Paredis, and A. Pop, “Towards Unified System Modeling and Simulation with ModelicaML: Modeling of Executable Behavior Using Graphical Notations”, The 7 International Modelica Conference, Como, Italy: Linköping University Electronic Press, (Linköping Electronic Conference Proceedings), S. 612–621, 2009.
[25] A. Sindico, M. Di Natale, and G. Panci, “Integrating SysML with Simulink using Open-source Model Transformations”, SIMULTECH, S. 45–56, 2011.

CONTACTS

M.Sc. Atif Mahboob	atif.mahboob@tu-ilmenau.de
Dipl.-Ing. Andreas Liebal	andreas.liebal@tu-ilmenau.de
Dr.-Ing. Stephan Husung	stephan.husung@em.ag
Univ.-Prof. Dr.-Ing Chrisitan Weber	christian.weber@tu-ilmenau.de
Univ.-Prof. Dr. phil. Heidi Krömker	heidi.kroemker@tu-ilmenau.de