Alternative Visualization of MBSE

Bin Lin[1], Maximilian Peter Dammann[1], Hui Kong[1], Bernhard Saske[1], Kristin Paetzold-Byhain[1]

[1] TUD Dresden University of Technology, Dresden, Germany

** Korrespondierender Autor:*
Dip.-Ing. Bin Lin
TUD Dresden University of Technology
Faculty of Mechanical Science and Engineering
Institute of Machine Elements and Machine Design
Chair of Virtual Product Development
01062 Dresden
☎ *+49 351 - 46336721*
🖂 *bin.lin@tu-dresden.de*

Abstract

Model-based systems engineering (MBSE) is an essential tool for the development of complex systems in interdisciplinary teams. However, the complexity of MBSE tools often makes the system-model difficult for non-systems engineers to understand, which negatively impacts communication efficiency and costs. Traditional MBSE tools also tend not to support intuitive and dynamic visual presentations, which limits the potential for understanding and collaboration. Virtual Reality (VR) technology, with its immersive and intuitive nature, offers an innovative possibility. By using Metaphern in VR, an intuitive and interactive system modle can be created that reduces communication and training costs and improves the early validation of product designs.

Keywords

Model-based systems engineering, Virtual reality, Metaphern

 | DOI: 10.35199/dfx2024.15

1. Motivation for research

In 1993, Dr. Wayne Wymore first introduced the term Model-based systems engineering (MBSE) [1]. The INCOSE System Engineering Vision 2020(2007) describes MBSE as "the formalized application of modeling to support system requirements, design, analysis, verification and validation activities in the early concept, development and later lifecycle phases." [2] MBSE is often compared to a traditional document-based system engineering approach. Document-based systems engineering often requires the exchange of a large amount of system information between different engineers. This exchange of information among various engineers makes the overall system model more complex and increases the possibility of information loss, leading to a system that is more difficult to maintain. One of the advantages of MBSE is to improve the communication between engineers by centering on the system model during the development process.

Traditional MBSE tools typically do not support intuitive, dynamic, interactive visual presentations, which limits their potential for system model understanding and collaboration. In contrast, virtual reality (VR) technology has been widely applied in the field of mechanical engineering. For example, Ford Motor Company uses VR technology for the internal layout and dashboard design of automobiles. Construction engineers use VR technology for construction simulation to optimiz construction processes and safety measures [3, 4].

Virtual Reality technology not only provides an immersive and intuitive view of a system model, but it can also be used to create an interactive system model that is easier to understand and manipulate. This potentially reduces communication and training costs and helps teams better analyze product requirements in the early stages of the product lifecycle.

2. State of Research and Research problem

There are already many MBSE tools available on the market, such as Cameo Systems Modeler, Rational Rhapsody, Integrity Modeler, Genesys, and Enterprise Architect. Although these tools help systems engineers communicate effectively with other engineers, they are often designed from the perspective of systems engineers. The complexity and specialization of their software often make it difficult for non-systems engineers to understand and accurately construct system models. The MBSE system model serves as the "Single Source of Truth," where each engineer exchanges information based on the system model [5]. Therefore, when constructing MBSE system models, systems engineers need to constantly communicate with other engineers, or non-systems engineers need to spend a considerable amount of time learning traditional MBSE tools, which undoubtedly affects communication efficiency and cost.

Akshay Kande proposed integrating MBSE with virtual reality as early as 2011 [6]. In 2015, AZAD M. MADNI suggested that story-based engineering could be used to lay out systems engineering scenarios in the virtual world through storytelling methods using Unity 3D [7]. In 2020, Mostafa Lutfi proposed the framework of VR-MBSE [8]. In recent years, the Systems Modeling Language (SysML) has become one of the most common approaches for implementing MBSE models. SysML is a graphical modeling language for systems engineering [9]. It is an extension of the Unified Modeling Language (UML) and is specifically designed to support the modeling of complex systems [10]. However, since the graphical language built by SysML is two-dimensional, the information of the modeled system is distributed across different diagrams such as Block Definition Diagram and Requirement Diagram. The information contained in a single diagram is limited. Therefore, when transitioning system models from 2D to 3D space, corresponding adjustments need to be made [11, 12].

With the development of virtual reality technology, it is not only possible to display clearer and more realistic models in VR environments, but it also provides rich possibilities for human interaction with VR models. Both the display requirements of Structure Diagrams and Requirement Diagrams, as well as the interaction requirements of Behavior Diagrams in

SysML, can be realized in a VR environment using virtual reality technology. Furthermore, more system information can be displayed using relatively few VR models, such as various SysML diagrams and the relationships between components in the SysML method. Therefore, the use of virtual reality technology to virtualize the MBSE method is a potentially effective approach.

The central question of this research paper is how to effectively combine virtual reality technology with the MBSE methodology based on SysML to develop VR system models that are easy for non-systems engineering to understand and communicate.

The objectives of this research paper include:

1. Rationally and efficiently convert components in the system modeling language to the metaphern models, so that the models used in VR are intuitive and easy to understand.
2. Use the VR model to rationally construct VR system models in a VR environment.
3. Evaluate the constructed VR system models and identify further research directions.

3. Concept

In order to intuitively and accurately convert Components from SysML to a VR environment, the concept of metaphern is introduced as shown in Figure [1]. The system modeling language will be translated into the metaphern models. For example, a Block in SysML will be transformed into a proper Block metapher.

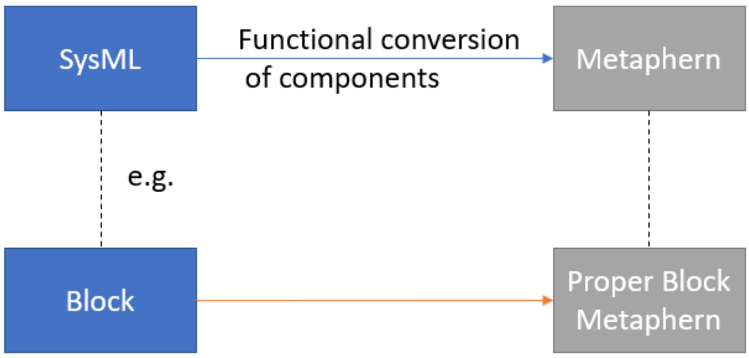

Figure 1: Conversion from SysML to Metaphern

A metapher is a model that is simplified as much as possible, yet easy for the user to understand. In everyday life, examples of metaphern include the figurines on safety exit signs and the folders on computer desktop. As shown in Figure [2], these are examples of using metaphern instead of SysML's components.

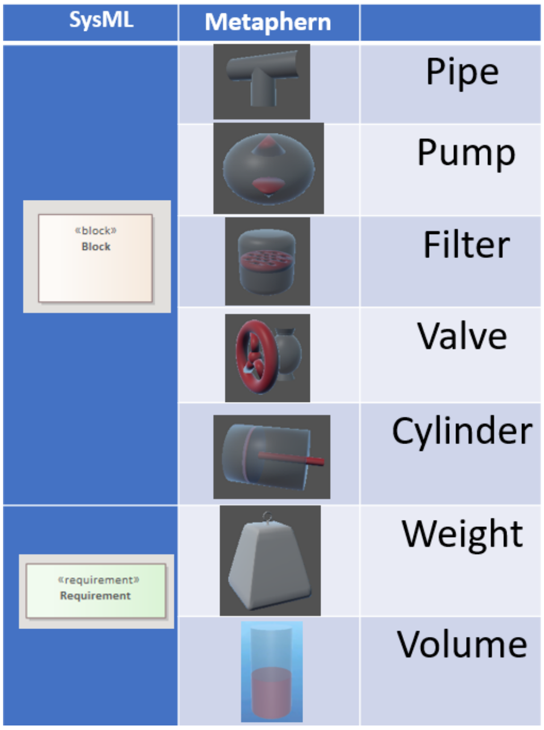

Figure 2: Example of using Metaphern instead of SysML's Components

Blocks and Requirements in SysML can be represented by different metaphern in a VR environment. These metaphern are more intuitive and easier to understand. To ensure that these metaphern can perform the same functions as Blocks, Requirements, and other components, they can also be reused in virtualized MBSE tools.

This leads to the discussion on the virtual engines used to create virtualization tools. There are two commonly used virtual engines on the market: Unity 3D and Unreal Engine. Unity 3D uses the C# programming language, which provides extensive cross-platform support and can be easily exported to a wide range of VR devices, including Oculus, HTC Vive, PlayStation VR, and others. Unity also offers a wealth of pre-built resources and plug-ins that can accelerate the development process. Unreal Engine primarily uses the C++ programming language and focuses more on graphics rendering and visual effects. For this paper, the chosen virtual engine is Unity 3D.

The most common way to reuse models in Unity is to save them as Prefabs, which can be preconfigured with parameters such as mesh, material, and size for direct use. Models created using Prefabs can also be further modified based on the original Prefab. This capability makes it possible to create unique VR system models using virtualized MBSE tools. For example, attributes like the length and color of a pipe can be adjusted.

4. Discussion

As shown in Figure [3], a CAD-Modell of the fluid storage system with four tanks contains pipes with different flow directions marked in different colors, along with filters, motors, and valves. A part of it is taken to build a traditional MBSE model in Enterprise Architect and a VR MBSE model in a virtual environment respectively.

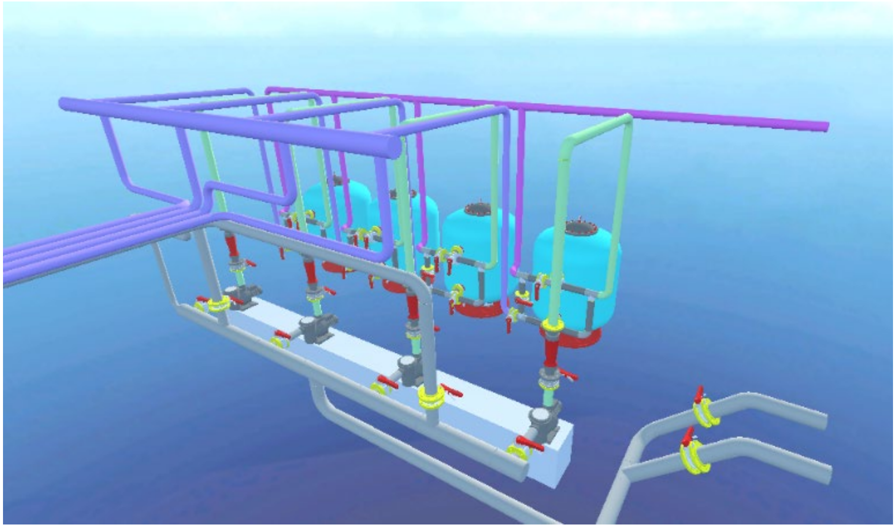

Figure 3: Fluid Storage System with four tanks

As an example, the liquid flow associated with one of the tanks can be constructed as the System-model in Enterprise Architect. This Block Definition Diagram (BDD) will illustrate that the basic fluid storage system consists of five parts: the tanks, the pipes, the motor, the valves and the filter. Additionally, the tanks or pipes are composed of smaller, distinct components. Although the structure of the liquid storage system has been simplified in BBD as shown in Fig. [4], it is still very difficult to show its complete structure in a display or a picture, and it is hard to be seen clearly. If all the components of the complete fluid storage system are included in the Block Definition Diagram (BDD), the diagram will become more complex and large, making it more difficult to clearly and completely display the entire structure of the fluid tank system on the screen.

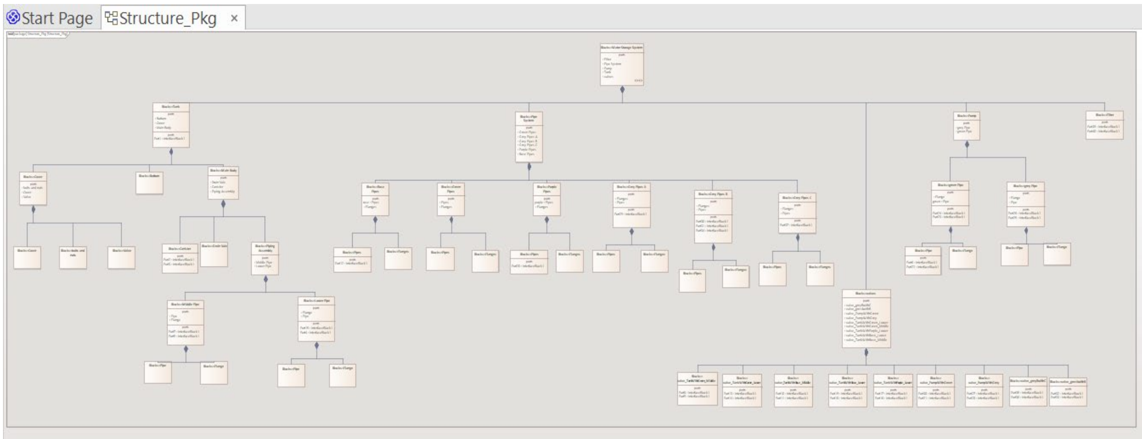

Figure 4: Structure Diagram of Liquid Storage System in Enterprise Architect

In the virtualized MBSE tools, the metaphern will be used shown in Figure [5] to construct such a fluid storage system.

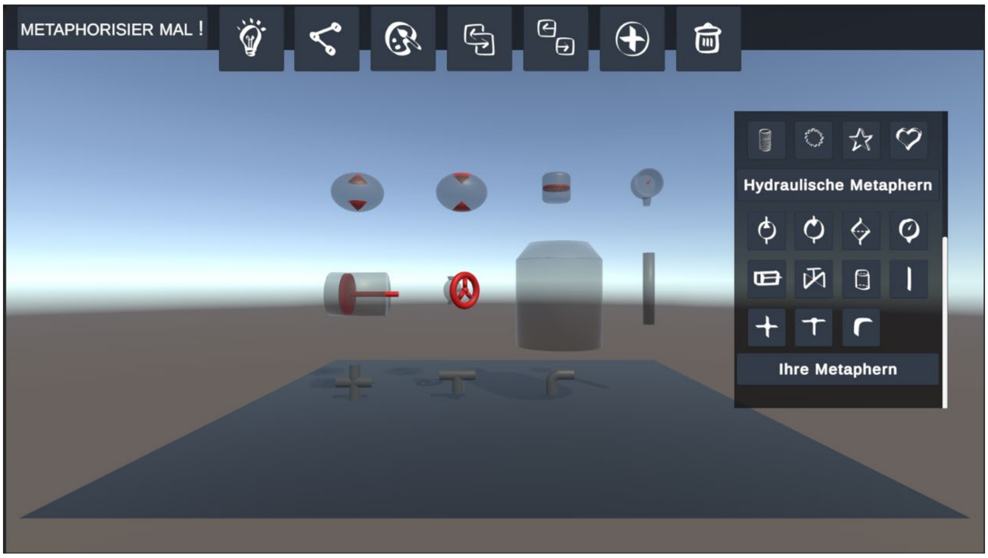

Figure 5: Metaphern's presentation in a VR environment

When using Metaphern to construct a VR system model, it is not a simple matter of copying the contents of the BDD into 3D space. the structure of the VR system model can be shown in the following figure [6].

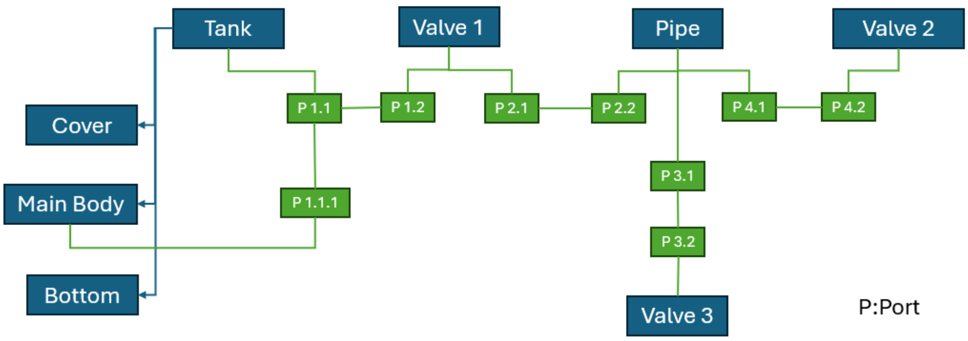

Figure 6: Structural design of the VR system model

The tank consists of the cover, main body, and bottom. The Port in Fig. [6] can represent both Relationships and Flow Properties in SysML, i.e., the Port in the VR system model can be used not only as a physical connection between components but also as an interface for components to transfer data and information. Ports can be used as metaphern in Unity with objects like attach points, but this is not the focus of this paper.

In traditional MBSE tools, this structure is built in the BDD, but in virtualized MBSE tools, it is edited in the project hierarchy. If the user wants, the structure edited in Hierarchy can also be displayed in the VR environment as in BDD by clicking a function key. This is one of the development directions of the research. As shown in Figure [7] below, it is a VR system model constructed using the Unity 3D virtual engine combined with Metaphern in Figure [5] and the VR system model structure in Figure [6]. At the area marked by number 1 in the figure is the structure relationships of the fluid storage system and the hierarchy of the Unity project.

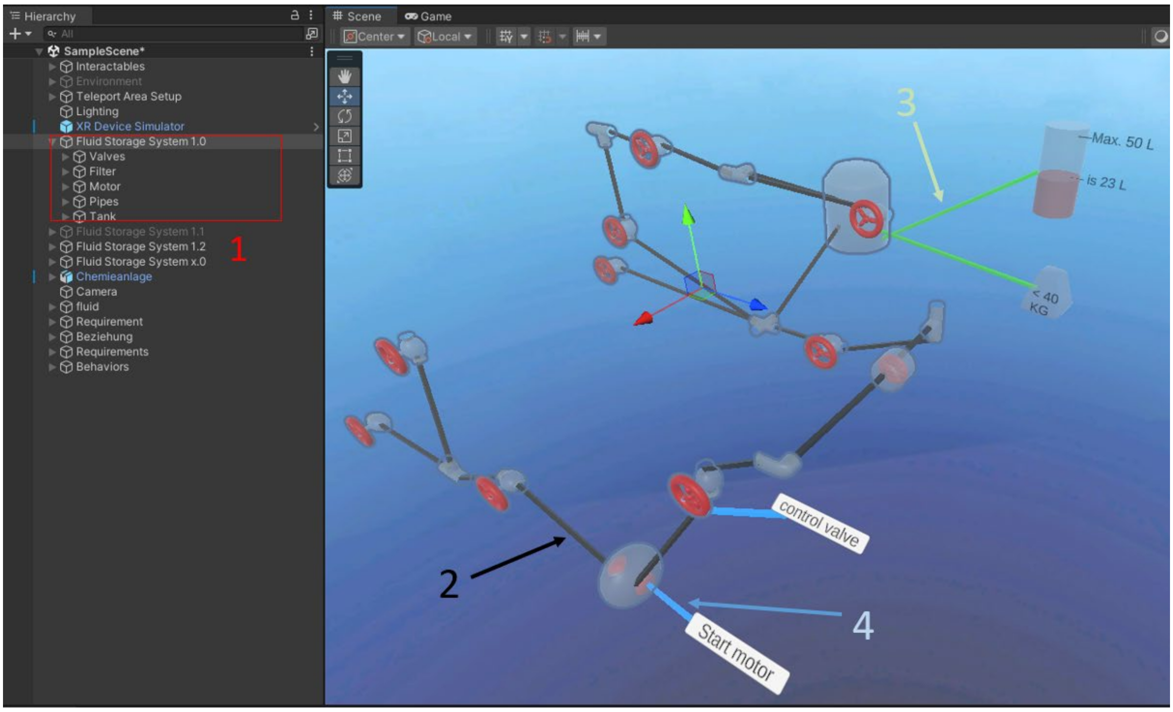

Figure 7: VR system model built with Metaphern

The Objects of the Fluid Storage System and Valves at position 1 in Figure [7] are Empty Objects created in Unity. As when building a structure diagram in BDD, some blocks point to a specific component, while others may represent abstract systems. These Empty Objects don't have properties such as materials by themselves, they function like those blocks in BBD that represent a particular system. The Metaphern used is an Object in Unity, and the empty Object created is also an object. The Empty Object of valves are created when different valve components are combined, just as pipes, filters, motors, tanks, and valves are combined to create a Fluid Storage System.

The structure in Figure [7] is suitable for engineers to think and discuss how to build the liquid storage system at the beginning stage of the project. The Metaphern representing the Block are connected in the middle by a black line marked with the number 2, which indicates the physical relationship between the Metaphern and also serves as a similar way to indicate the direction of the liquid flow in the Internal Diagram. The green line marked with the number 3 connects the two Metaphern from the Tank to the two Metaphern, indicating the weight and capacity of the Tank respectively. The blue line labeled with the number 4 connects from Motor and Valve to the two push-button Metaphern, indicating the interactivity of the Metaphern.That means that the functionality and requirements of the Behavior Diagram in SysML can be realized.

In traditional MBSE tools, the Internal Block Diagram uses flow properties and ports to define the flow direction of the fluid storage system, as shown in Figure [8]. Although the flow direction of the liquid and the relationships between different blocks can be successfully defined, the diagram still appears intricate and complex.

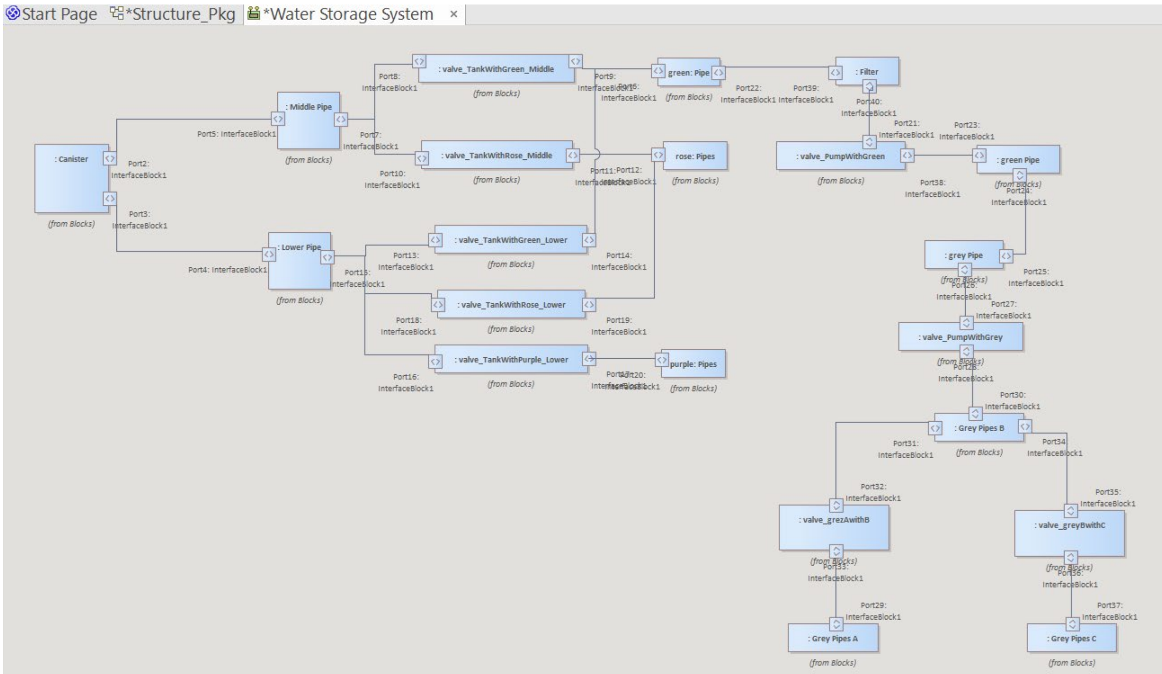

Figure 8: Presentation of fluid flow in Internal Diagram

In the Hierarchy on the left side of Figure [7] the liquid storage system can be seen with different iterative versions. Using the Port in Figure [6] to spatially locate and connect the different Metaphern, a more intuitive model of the VR system can be constructed in the VR environment as in Figure [9].

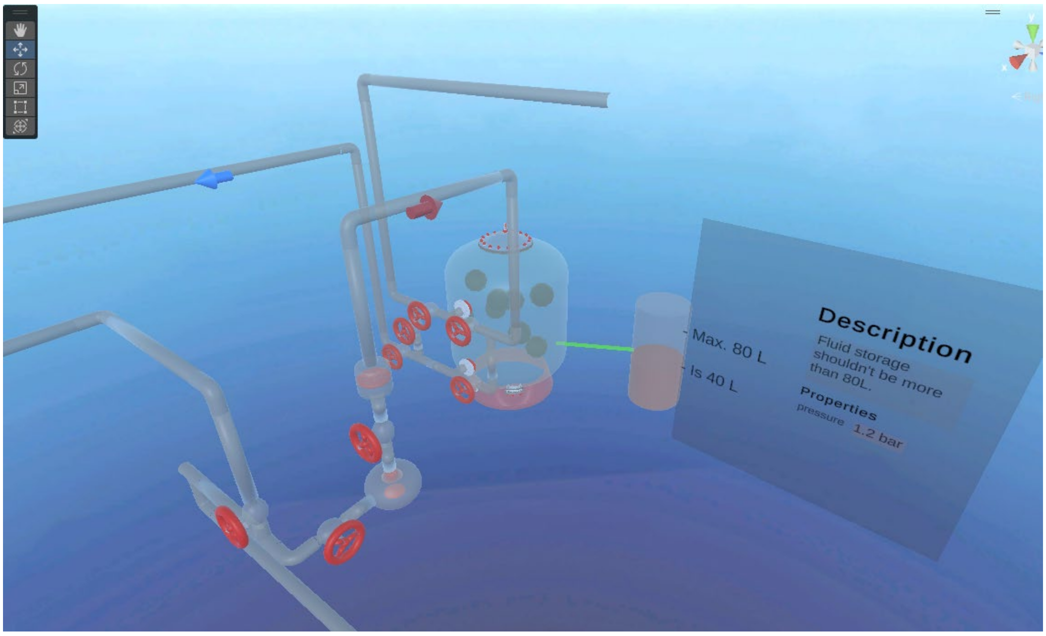

Figure 9 : Iterated VR system model

As the project progresses, users can use custom metaphern to build more accurate system models, as shown with the tank in Figure [9]. For example, engineers can simplify CAD models to a certain extent to replace the original simple tank metapher, use arrows to show the direction of liquid flow, and use material metaphern to describe the information about the liquid contained in the tank. When annotations or notes are needed, they can also be displayed in the VR environment using text. These are all directions for further research.

Comparing the system model in traditional MBSE tools with the system model constructed in VR, it can be found that constructing the system model in traditional MBSE tools requires creating structures in the BDD. Sometimes, it is necessary to reference the Block in the BDD as a link to be used in other diagrams, making the structure of the entire system model more complicated. The system model built in VR has an additional dimension, allowing the Structure, Requirement, and Behavior of the VR system model to be clearly seen in one Figure [7], making the whole VR system model easier to understand.

Therefore, it is feasible and effective to use metaphern to describe the system model in a VR environment based on the system modeling language. If the complete virtualized MBSE methodology tool can be built, other engineers can complete the VR system model with minimal assistance from system engineers, and the system engineers will be able to change their role from "Single Source of Truth" information exchangers to supervisors and controllers of the whole system model.

5. Conclusion and Outlook

The virtualized MBSE approach is a complex but rewarding research endeavor. Although the research is still in its preliminary stages, the goal is to improve the efficiency of communication and collaboration in systems engineering through the development of VR MBSE tools that help users understand system models. The initial VR-MBSE model is clearly more intuitive and easier to communicate for mechanical engineers. Since the current model is still relatively simple, there is no accurate validation against the improvement of collaboration efficiency. Future research will include further development and optimization of the tool, such as the development of Metaphern for other professional orientations such as construction engineers, electrical engineers, lawyers, etc., and extensive user testing, as well as exploring applications in other industries.

Further research is also needed to determine which parts of a system modeling language can and need to be effectively translated into metaphern models. It is also necessary to study modeling languages used metaphern in VR to standardize VR systems engineering. The standardized VR system model can be exported in an XML-like format. The exported file will identify the metaphern by GUID, type, etc., and will also contain the structural, spatial, and constraint information of different metaphern. Corresponding interface tools need to be developed for this purpose.

After completing the virtualized MBSE tool, it is necessary to construct a VR system model through real systems engineering cases to validate and continuously optimize the tool. These represent both directions for further research and challenges that the research will face.

Literaturverzeichnis

[1] Wymore, A. W.: Model-based systems engineering. An introduction to the mathematical theory of discrete systems and to the tricotyledon theory of system design. Systems engineering series. CRC Press, Boca Raton(1993). ISBN: 978-0-8493-8012-9

[2] Walden, D. D., Roedler, G. J., Forsberg, K., Hamelin, R. D., Shortell, T. M., and Kaffenberger, R., Eds. :INCOSE Systems Engineering Handbuch. Ein Leitfaden für Systemlebenszyklus-Prozesse und -Aktivitäten. GfSE, Gesellschaft für Systems Engineering e.V, München(2017). ISBN: 978-3-9818805-0-2

[3] Fant, J. S. and Pettit, R. G. : MBSE Mission Assurance. In Handbook of Model-Based Systems Engineering, A. M. Madni, N. Augustine and M. Sievers, Eds. Springer International Publishing, Cham(2023), 861–893. DOI=10.1007/978-3-030-93582-5_72.

[4] Sampaio, A. Z. and Martins, O. P. : The application of virtual reality technology in the construction of bridge: The cantilever and incremental launching methods. Automation in Construction 37(2014), 58–67. DOI= 10.1016/j.autcon.2013.10.015

[5] Vaneman, W. K. and Carlson, R. : Model-Based Systems Engineering Implementation Considerations. In IEEE International Systems Conference (SysCon). IEEE(2019), 1–6. DOI=10.1109/SYSCON.2019.8836888.

[6] Kande, A. : Integration of model-based systems engineering and virtual engineering tools for detailed design.(2011)

[7] Madni*, A. M. : Expanding Stakeholder Participation in Upfront System Engineering through Storytelling in Virtual Worlds. Systems Engineering 18, 1(2015), 16–27. DOI= 10.1002/sys.21284

[8] Lutfi, M. and Valerdi, R.: Virtual Reality in Model Based Systems Engineering: A Review Paper. In HCI International 2020 – Late Breaking Posters, C. Stephanidis, M. Antona and S. Ntoa, Eds. Communications in Computer and Information Science. Springer International Publishing, Cham(2020), 197–205. DOI=10.1007/978-3-030-60703-6_25.

[9] Peak, R. S., Burkhart, R. M., Friedenthal, S. A., Wilson, M. W., Bajaj, M., and Kim, I.: Simulation ‐ Based Design Using SysML Part 1: A Parametrics Primer. *INCOSE International Symp* 17, 1(2007), 1516–1535. DOI=10.1002/j.2334-5837.2007.tb02964.x

[10] Hampson, K. : Technical Evaluation of the Systems Modeling Language (SysML). Procedia Computer Science 44(2015), 403–412. DOI=10.1016/j.procs.2015.03.054

[11] Atif Mahboob, Christian Weber, Stephan Husung, Andreas Liebal, and Heidi Krömker.: Model based systems engineering (MBSE) approach for configurable product use-case scenarios in virtual environments. (2017).

[12] Lutfi, M. and Valerdi, R. : Integration of SysML and Virtual Reality Environment: A Ground Based Telescope System Example. Systems 11, 4(2023), 189. DOI=10.3390/systems11040189