**Aerospace Europe Conference 2023**
**Joint 10th EUCASS – 9th CEAS Conference**

Abstract #XXX *(to be filled by the organizers)*
Preferred Topics: CFDMPS (DEFAINE session)
Corresponding author: La Rocca, Gianfranco
e-mail of corresponding author: G.LaRocca@tudelft.nl
Type: Oral
Status of corresponding author: Regular

**Title**

# An MBSE approach to support Knowledge Based Engineering application development

**Authors**

*Gianfranco La Rocca [1]*, Akshay Raju Kulkarni [2], Fabio Mendes Fernandes [3], Robin Augustinus [4], Bram Timmer [5]*

** Corresponding author*
*[1] TU Delft, DELFT, 2629 HS, the Netherlands, G.LaRocca@tudelft.nl*
*[2] TU Delft, DELFT, 2629 HS, the Netherlands, A.RajuKulkarni@tudelft.nl*
*[3] TU Delft, DELFT, 2629 HS, the Netherlands, F.MendesFernandes@student.tudelft.nl*
*[4] GKN Fokker, HOOGERHEIDE, 4631 RP, the Netherlands, Robin.Augustinus@fokker.com*
*[5] GKN Fokker, HOOGERHEIDE, 4631 RP, the Netherlands, Bram.Timmer@fokker.com*

**Abstract**

Knowledge Based Engineering (KBE) applications are key enablers towards the automation of complex systems design. By means of the provided object-oriented language, caching and dependency tracking capabilities, and tight integration with a CAD kernel [1], KBE systems allow capturing engineering knowledge into powerful automation solutions for product configuration, design space explorations and multidisciplinary design optimization [2]. The development of a KBE app encompasses the acquisition from domain experts of the necessary product and process knowledge, and its translation into code, using the programming language provided by the KBE system at hand. This non-trivial process requires expertise both in knowledge acquisition and modeling, as well as in code generation and verification. At date, a convenient methodology to support KBE is missing, with consequences on the time required to develop applications, their quality (requirement compliance and traceability, maintainability, scalability, etc.) and, eventually, the capability to preserve and efficiently re-use engineering knowledge.

This paper proposes a novel methodological approach, based on Model Based Systems Engineering (MBSE), where a central knowledge model is generated first, making use of the industry standard SysML (System Modeling Language) [3]. The rich semantic and expressiveness of the SysML diagrams, together with their flexibility but conceptual rigor, are exploited to capture the required knowledge (requirements, data, product architecture, processes) by means of multiple interconnected and synchronized views. Dedicated translators have been developed to parse the knowledge model and automatically generate source code for the targeted KBE system, in this case the Python-based KBE platform ParaPy [4]. When the knowledge model is edited, a new KBE skeleton code can be automatically generated, guaranteeing synchronization of model and code The scope is currently limited to the generation of KBE app skeletons, whose development is acknowledged to be a critical part in the overall KBE app development process. Thanks to the visual modeling approach offered by SysML, app skeletons can be generated also by product architects with limited programming capabilities and then completed by KBE developers, directly in the ParaPy development environment.

In this paper we discuss the application and assessment of the proposed methodology by GKN Aerospace (Fokker Elmo) experts, on the development of a KBE app to automate the design of Electrical Wiring Interconnection Systems (EWIS) architectures. Ease of modeling, time required and quality of the generated code skeleton were evaluated. The application showed that modeling has a learning curve but is easy enough to learn. Although the generation of the SysML models does require more time than current processes, the higher level of achieved detail and formalism facilitates further tool development, extension and maintenance. The generated code skeleton was of good quality. It provided a well-structured correct starting point for further app development, and did realize time savings in app skeleton development. Besides, the proposed methodology also provides the traceability of requirements onto the various elements of the KBE app architecture, including direct links to the modules and code elements where such requirements are encoded, thereby mitigating the typical “black-box” effect of KBE apps. Furthermore, parts from previously generated knowledge models can be imported, allowing effective project-to-project knowledge transfer. The development of “round tripping” capabilities are planned to enable changes in the KBE app code to be fed back into the knowledge model, and even to build knowledge models from scratch, starting from existing KBE apps. The ability to operate in both directions will enable unprecedented capabilities in the domain of KBE development.

## Acknowledgement

The research presented in this paper has partially been performed in the framework of the DEFAINE (Design Exploration Framework based on AI for froNt-loaded Engineering) project and has received funding from ITEA 3 programme.

## References

[1] La Rocca, G., Knowledge Based Engineering: between AI and CAD. Review of a language based technology to support engineering design, Advanced Engineering Informatics, Volume 26, Issue 2, 2012
[2] Multidisciplinary Design Optimization Supported by Knowledge Based Engineering, by J. Sobieszczanski-Sobieski, A. Morris, M. van Tooren, Wiley, 2015
[3] L. Delligatti, SysML distilled, Addison-Wesley, 2013
[4] The ParaPy platform, www.parapy.nl, visited January 2023.