# Validating Design-Intent System Specification Models with State-of-the-art Large Language Models


1st Wiebke Outzen
*Airbus Operations GmbH*
Germany
wiebke.outzen@airbus.com

2nd Michael Oppermann
*Airbus Operations GmbH*
Germany
michael.oppermann@airbus.com

3th Tim Gonschorek
*Institute of Intelligent Cooperating Systems*
*Otto von Guericke University*
Magdeburg, Germany
tim.gonschorek@ovgu.de

4th Frank Ortmeier
*Institute of Intelligent Cooperating Systems*
*Otto von Guericke University*
Magdeburg, Germany
frank.ortmeier@ovgu.de



***Abstract*—This paper proposes an approach for ensuring the consistency and validity of MBSE models concerning their textual specifications. We outline a principal process for i) comparing architectural system information and ii) adapting the model for formal verification of the behavior with model checking.**

**The main novelty is the application of LLMs for the model comparison and integration of behavioral proof obligations.**



***Index Terms*—AI-based Design Model Validation, Model-based Systems Engineering and Safety Assessment, Crossfunctional Model Reusability**


## I. Introduction

In current aerospace (A/C) development processes, Model-Based Systems Engineering (MBSE) models typically serve as supplementary design-intent models, offering detailed yet incomplete representations of specific system aspects. These models are often created in conjunction with, rather than as a complete replacement for, textual specification documents. This dual-track approach raises a critical challenge: ensuring the equality and consistency between textual specifications and their corresponding design-intent models. For complex systems involving numerous domain experts and modeling formalisms, manual validation of this alignment is both cumbersome and error-prone.

To address this challenge, we propose a novel approach leveraging artificial intelligence (AI) and Large Language Models (LLMs) to automate and streamline the validation process. Our research hypothesis posits that most specification artifacts and requirements can be categorized into a limited set of types, including structural, behavioral, and safety requirements.

By incorporating AI and LLMs, we aim to develop a robust, scalable process for ensuring the validity and consistency of MBSE models in complex engineering projects. This approach has the potential to significantly enhance the effectiveness of MBSE, ultimately leading to safer, more reliable systems.

## II. Comparing Textual and Model Specifications

The proposed process involves the following steps:

- Information Extraction: Utilizing LLMs to extract relevant information from textual specifications, encompassing system components, interfaces, context properties, failure conditions, and intended behavior.
- Structural Validation: Employing LLMs to assess the alignment between textual representations (e.g., XML/XMI, FMU) and the structural architecture requirements captured in the model.
- Behavioral Validation: Translating behavioral requirements into model checking proof objectives, enabling rigorous verification of the system model's behavior.

Figure 1 illustrates the information flow in our proposed approach. It involves extracting and comparing specific information from both textual specification documents and the MBSE model. Based on the requirement categories introduced in [1], classes of system information are proposed. These include architectural components and communications, as well as system, equipment and environmental properties. The design specification document serves as the primary source for the system's architecture, parameters, interfaces between components, and interactions with external systems and the environment.

The process begins by extracting architectural components and their interfaces from both the design specification and the MBSE model. These extracted elements are then meticulously compared to ensure consistency and completeness. Concurrently, model checking proof objectives are extracted from the requirement specification document. Each proof objective is thoroughly analyzed to identify the involved architectural

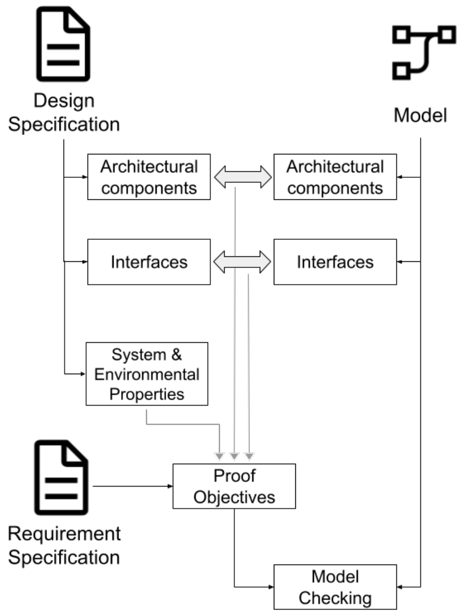


Fig. 1. Information extraction and comparison between specification documents and system model

components, interfaces, and relevant system/environmental properties. These identified elements are then cross-referenced with the information extracted from the design specification and the model. If all necessary information is available and consistent, the proof objective can be seamlessly integrated into the model, paving the way for a comprehensive safety analysis.

Figure 2 depicts the overall process. Similar to the approach in [2], in which an LLM is used for model slicing on an XML representation of a Simulink model, the MBSE model is exported into a textual representation (e.g., XML/XMI) and then reduced to include only the relevant information. This streamlined representation, along with excerpts from the design specification, is fed into a Large Language Model (LLM). The LLM is tasked with comparing the architectural components and interfaces, resulting in a consolidated list that captures the essential structural information of the system.

This consolidated list is subsequently returned to the LLM, accompanied by selected safety requirements from the requirements specification. The LLM then meticulously verifies whether all components, interfaces, and system/environmental parameters specified in the safety requirements are indeed present in the model.

If confirmed, we plan to translate the textual requirements into formal proof objectives suitable for model checking. In [3] GPT models improved by retrieval augmented generation are used to generate and modify SysML models. In our approach, the parameters are precisely located within the model, facilitating the incorporation of the safety requirements into the model's structure and behavior. Subsequently, the DIVE toolbox, a specialized safety analysis tool, is employed to conduct a rigorous safety analysis on the augmented model, assessing its compliance with the specified safety requirements.

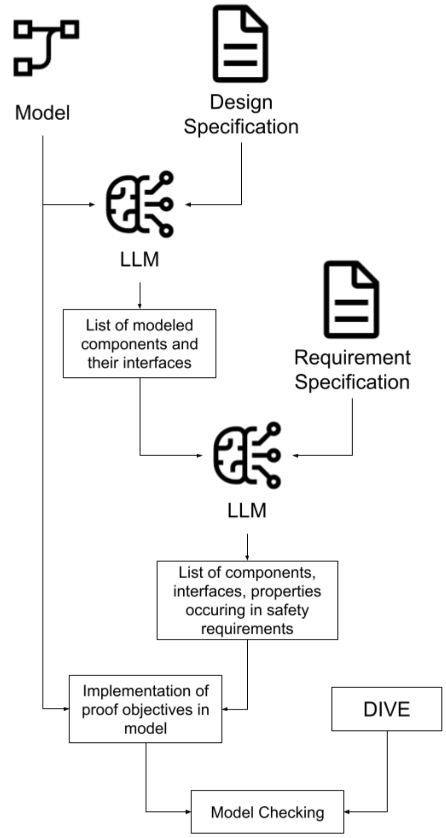


Fig. 2. Proposed process for document and model comparison

## III. Main Challenges

The proposed work contains, from our point of view, three main challenges:

1) The size and complexity of the specification documents must be handled by the LLM.
2) Finding a suitable textual representation of the system model an LLM can work with.
3) Evaluate which available LLM is most suitable either for the interpretation of the textual specification or the validation of the system model.

However, the question is, still, how the prompt must be written to lead an LLM to validate whether the required elements are within the model. And further, how much manual effort is still required or even desired.

## References

[1] A. Cimatti, M. Roveri, A. Susi, and S. Tonetta, "From informal requirements to property-driven formal validation," in *Formal Methods for Industrial Critical Systems*, D. Cofer and A. Fantechi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 166–181.

[2] D. Luitel, S. Nejati, and M. Sabetzadeh, "Requirements-driven slicing of simulink models using llms," 2024. [Online]. Available: https://arxiv.org/abs/2405.01695

[3] R. Longshore, R. Bell, and R. Madachy, "Leveraging generative ai to modify and query mbse models," Acquisition Research Program, Tech. Rep., 2024.