Proceedings of the Design Society, Volume 5: ICED25
https://doi.org/10.1017/pds.2025.10058

AI-augmented systems engineering: conceptual application of retrieval-augmented generation for model-based systems engineering graph

Fabian Hanke[1,✉], Isaac Mpidi Bita[1], Oliver von Heißen[1], Weller Julian[1], Hovemann Aschot[1] and Dumitrescu Roman[2]

[1] Fraunhofer-Institut für Entwurfstechnik Mechatronik IEM, Germany, [2] Paderborn University (HNI), Germany

✉ fabian.hanke@iem.fraunhofer.de

ABSTRACT: This paper presents the MBSE-Graph-RAG framework to address key challenges in Model-Based Systems Engineering (MBSE). Traditional MBSE tools suffer from usability barriers, limited accessibility, and integration challenges. By combining knowledge graphs with Retrieval-Augmented Generation (RAG), the proposed framework enables AI-Augmented engineering through natural language interactions and automated system architecture generation. A systematic literature review establishes a solid research foundation, identifying gaps in AI-assisted MBSE. Key contributions include a structured MBSE-Graph interface, improved usability via Large Language Models (LLMs), and automated graph construction aligned with SysML. A proof-of-concept demonstrates the potential of this approach to enhance MBSE by reducing complexity, improving data accessibility, and supporting engineering collaboration.

KEYWORDS: systems engineering (SE), artificial intelligence, ontologies, knowledge graph, retrieval augmented generation

1. Introduction

Engineering projects increasingly demand faster development cycles, innovation, and multidisciplinary collaboration. Although ***Model-Based Systems Engineering*** (MBSE) offers a structured way to manage complexity by modeling requirements, functions, and architectures, it faces two persistent barriers: (1) mainstream MBSE tools (e.g., Cameo Systems Modeler, Enterprise Architect) are complex to learn, inhibiting broader user adoption; and (2) leveraging historical data under ***Data Driven Systems Engineering*** (DDSE) is cumbersome due to data exchange issues and unwieldy plug-ins. ***Large language models*** (LLMs) such as GPT or Llama could alleviate these challenges by providing advanced analytics and natural-language interactions, yet their integration into MBSE remains underexploited. To address these gaps, this paper proposes ***retrieval augmented generation*** (RAG) with knowledge graphs as a means to represent MBSE system architectures more intuitively.
Problem Statement: Existing MBSE and DDSE approaches are limited by steep learning curves, proprietary tool constraints, and difficulties in harnessing unstructured historical data. Further, embedding AI technologies (like LLMs) into MBSE requires substantial, often prohibitive development efforts. As a result, automated model analysis and user-friendly, AI-driven interfaces remain elusive, preventing MBSE from reaching its full potential in complex engineering environments. A new approach is thus needed to streamline MBSE workflows, promote more natural user interactions, and simplify data integration.
Research Contribution: To overcome these challenges, we introduce the MBSE-Graph-RAG framework, which automatically generates system architectures from unstructured data, enabling an AI-Augmented SE process. By integrating LLMs with knowledge graphs, the framework provides natural-language access to

modeled information. Its key contributions include: ***(1) MBSE-Graph-RAG Framework*** – Combines knowledge-graph-based system architecture representation with LLMs to improve MBSE data accessibility and usability. ***(2) MBSE-Graph-RAG for Natural Language Interaction*** – Lowers the entry barrier for non-experts, providing intuitive access to MBSE data and reducing reliance on specialized modeling editors. ***(3) Automatic MBSE-Graph Construction*** – Demonstrates how unstructured data can be harnessed to enrich systems engineering, paving the way for broader DDSE adoption.

1.1. Design Science Research Methodology

The MBSE-Graph-RAG framework was developed following the ***Design Science Research Methodology*** (DSRM) (Peffers et al., 2007), ensuring a systematic approach to engineering research: Problem Identification identified the shortcomings of existing MBSE tools, the difficulty of integrating unstructured data, and the need for natural-language interaction. **Definition of Objectives** set goals to automate MBSE-graph construction, enhance data accessibility, and improve usability via Graph Retrieval-Augmented Generation (GraphRAG) and LLMs. **Design and Development** integrated AI techniques with knowledge graphs to automate system architecture creation and enable natural-language queries. **Demonstration** provided a use case that transforms unstructured engineering data into a structured knowledge graph, illustrating enhanced MBSE practices. **Evaluation** has not yet been formally conducted for the conceptual application. **Communication** involved disseminating the framework and findings through this paper, benefiting both academic and industrial stakeholders.
This iterative, feedback-driven methodology ensured that both the technical and user-centric needs of MBSE were addressed. The solution concept is currently under development, with the presented version being the result of the third development cycle.

1.2. Key concepts and definition of terms

Model-Based Systems Engineering (MBSE): Formalizes the entire system lifecycle via modeling languages like SysML (Lu et al., 2022). **Data-Driven Systems Engineering** (DDSE): Incorporates data analytics and machine learning to enhance decision-making and system performance (Borth & van Gerwen, 2018; R. S. Kenett et al., 2020; Tissen et al., 2023). **AI-Augmented Systems Engineering** (AI-ASE): Extends DDSE by applying advanced AI—especially LLMs—to handle unstructured data and reduce complexity in system models. **Ontologies**: Provide standardized data patterns and rules, crucial for automated reasoning in knowledge graphs (Fu et al., 2021). **Knowledge Graphs**: Capture semantically rich, interconnected nodes (e.g., requirements) and edges (e.g., relationships), vital for holistic MBSE data exploration (Gutierrez & Sequeda, 2020; Lu et al., 2022). **Graph Retrieval-Augmented Generation** (GraphRAG): Integrates LLMs with knowledge graphs for contextually accurate and explainable outputs (Enzo, 2024). **Semantic Web Technologies**: RDF, OWL, and SPARQL facilitate semantic data structuring and querying in knowledge graphs (Zindel et al., 2022). **Graph Analytics**: Focuses on analyzing node-link networks to uncover patterns and complex relationships beyond what traditional tabular data can reveal.

1.3. Problem analysis

Modern products and development processes are increasingly complex, pushing traditional, document-based SE to its limits. As systems grow in scale and interdisciplinary scope, it becomes challenging to maintain coherent, consistent models across various domains. Inefficient reuse of system models exacerbates the issue, as the growing number of model files makes it difficult to locate and repurpose existing knowledge for new products (Fu et al., 2021). Moreover, MBSE data is scattered across different disciplines and stored in heterogeneous modeling languages, formats, and syntaxes, complicating data integration and interoperability (Zindel et al., 2022). A key driver behind the adoption of graph analytics in MBSE is the need to analyze complex relationships and dependencies among system elements. Traditional approaches struggle with sophisticated analyses—for example, cyber resilience or life cycle assessments—because modern systems exhibit multi-layered interdependencies that are not readily captured or queried in siloed models (Dwivedi, 2018; G. M. Schweitzer et al., 2022). In the early design phase, limited system knowledge hampers informed decision-making, yet it is precisely during this stage that design choices have the greatest downstream impact (G. Schweitzer et al., 2023). Challenges also arise in design space exploration, where engineers seek to understand the implications of numerous design variations. Existing MBSE methods rarely provide robust techniques for managing the combinatorial complexity of design options (Timperley et al., 2024). Furthermore, conducting lifecycle-oriented analyses (e.g., LCE/LCSA) demands a holistic approach that captures extensive cause-and-effect chains—an area in which fragmented data landscapes fall short (G. M. Schweitzer et al., 2022; G. Schweitzer, 2023). Beyond these technical obstacles, the literature highlights additional adoption barriers. Many MBSE tools require extensive manual effort to model and maintain architectures, especially when incorporating historical and unstructured data (Chami and Bruel, 2018). The complexity and steep learning curves of existing solutions often deter non-expert users. In parallel, the lack of standardization and interoperability between tools results in data silos, limiting collaboration and reuse. The proposed MBSE-Graph-RAG Framework directly tackles these issues by automating system architecture generation through LLMs, introducing natural-language interfaces, and providing a tool-independent semantic layer with knowledge graphs. By addressing both the technical and usability hurdles, it seeks to promote more scalable, cost-effective MBSE practices, ultimately leading to faster innovation and higher-quality engineering outcomes.

2. Related works

This literature review follows the structured methodology proposed by Webster and Watson (2002), providing a comprehensive analysis of AI-ASE with a focus on the integration of AI, Knowledge Graphs, and MBSE. The review aims to assess how advanced AI techniques, particularly Graph Analytics, can enhance MBSE applications by enabling better insights, decision-making, and efficiency in the system development process. (Webster & Watson, 2002) A systematic search strategy was conducted using the IEEE database with a targeted search string, yielding 194 results, of which 6 relevant papers were identified. The relevant papers listed below in Table 1. Additional insights from peer reviewers and a snowballing approach contributed 5 more studies, ensuring a thorough coverage of the topic. The review employs a concept-centric organization, structuring insights around key themes such as DDSE, AI-ASE, GraphRAG, and Semantic Web Technologies.

Search string: *("MBSE" OR "Systems Engineering" OR "product lifecycle management" OR "PLM") AND ("Engineering Graph" OR "Knowledge Graph" OR "Graph-based Database" OR "Graph Database")*

In line with Webster and Watson's approach, the review synthesizes and analyzes the literature to highlight significant insights, contradictions, and trends. The analysis identified a growing trend towards using graph-based methods and integrating AI techniques within MBSE to improve system model generation and management.

Table 1. Overview related works

Reference	Titel
(Zindel et al., 2022)	Building a Semantic Layer for Early Design Trade Studies in the Development of Commercial Aircraft
(Fu et al., 2021)	Building SysML Model Graph to Support the System Model Reuse
(Huang et al., 2023 - 2023)	Design and Implementation of Ground PHM Sample Machine
(Lu et al., 2022)	Design Ontology Supporting Model-Based Systems Engineering Formalisms
(Dwivedi, 2018)	Implementing Cyber Resilient Designs through Graph Analytics Assisted MBSE
(Timperley et al., 2024)	Mapping the MBSE Environment and Complementary Design Space Exploration
(Chami et al., 2019)	A First Step towards AI for MBSE: Generating a Part of SysML Models from Text Using AI
(Schweitzer et al., 2022)	Detection of Cause-Effect Relationships in Life Cycle Sustainability Assessment
(Schweitzer et al., 2023)	Engineering Graph as an Approach to Support Design Decisions in Product Development
(Schweitzer, 2023)	Einsatz von KI in der Produktentwicklung auf Basis eines Engineering-Graph
(Kasper et al., 2024)	The Digital Thread for System Lifecycle Management with a Native Graph Database

2.1. Approaches to integrating graph analytics in MBSE: key findings and trends

Transformation of SysML into Knowledge Graphs: A prominent method for integrating graph analytics into MBSE involves transforming SysML models into Knowledge Graphs. This process converts SysML elements and their relationships into graph data structures, utilizing a SysML metamodel ontology as the schema (Chami et al., 2019; Fu et al., 2021; Lu et al., 2022; Zindel et al., 2022). By adopting this approach, existing modeling artifacts can be effectively reused, and advanced analyses of the complex interrelationships within system architectures become feasible. Research such as "Building SysML Model Graph to Support the System Model Reuse" demonstrates how these transformations enable semantic searches and the fusion of multiple subgraphs, thus enhancing the reuse and maintainability of system models. **Semantic Layers and Unified Ontologies**: Another strategy frequently highlighted in the literature is the development of a tool-independent semantic layer using ontologies created with OML, OWL, or RDF. Such semantic layers establish a single "source of truth" that standardizes terminologies, streamlines data exchange, and enhances interoperability across diverse modeling environments (Fu et al., 2021; Lu et al., 2022; Schweitzer, 2023; Zindel et al., 2022). The GOPPRRE framework, for example, presents a Unified MBSE Ontology that integrates core MBSE concepts and relationships, supporting data interoperability throughout the system lifecycle. This approach is exemplified in "Building a Semantic Layer for Early Design Trade Studies in the Development of Commercial Aircraft", where a semantic layer using Semantic Web Technologies enables automatic instantiation of MBSE data into a knowledge graph. **Engineering Graph and Digital Thread Concepts**: The concept of an Engineering Graph, as discussed by Schweitzer et al. (2023), extends the knowledge graph paradigm by linking and analyzing data from a wide array of engineering systems—such as PLM, ERP, and IoT systems—within a unified graph. This enables a holistic view of engineering data, supporting advanced analytics and decision-making (Schweitzer, 2023; Schweitzer et al., 2023). Complementarily, the Digital Thread approach leverages graph databases to connect lifecycle data across different phases of product or system development, enhancing data consistency and transparency (Kasper et al., 2024; Schweitzer, 2023; Schweitzer et al., 2022) These connected frameworks improve lifecycle management and allow early detection of design risks, offering a robust foundation for lifecycle sustainability assessments and design decision support. **Knowledge Graphs for Design Space Exploration:** Knowledge graphs are also used for Design Space Exploration (DSE), where they capture the complex relationships among MBSE formalisms, tools, methods, and design variables. By structuring these elements in a graph, engineers can efficiently navigate design alternatives and identify optimal solutions based on specific requirements and constraints (Dwivedi, 2018; Timperley et al., 2024; Zindel et al., 2022). For instance, the paper "Mapping the MBSE Environment and Complementary Design Space Exploration Techniques" illustrates how knowledge graphs support systematic evaluation of design trade-offs, enhancing the ability to select the most suitable MBSE environments and methods. **Ontologies, AI, and Lifecycle Management**: Ontologies, combined with AI methods, play a crucial role in achieving advanced lifecycle management. Tools such as OWL, RDF, and SPARQL facilitate robust data handling and querying, while AI techniques—including entity matching, pattern recognition, and Large Language Models (LLMs)—automate portions of the MBSE workflow (Chami et al., 2019; Fu et al., 2021; Huang et al., 2023 - 2023; Kasper et al., 2024; Lu et al., 2022; Schweitzer, 2023). These technologies assist in converting unstructured data into standardized models, identifying anomalies within large-scale architectures, and supporting constraint-based optimization of system configurations. The paper "A First Step towards AI for MBSE" highlights how LLMs can facilitate the generation of SysML models from textual descriptions, showcasing the early potential of AI-augmented MBSE solutions.

The literature consistently underscores the promise of knowledge graphs and semantic technologies in enhancing reusability, interoperability, and analytics within MBSE. Research trends indicate a growing focus on Digital Thread implementations, cross-domain ontologies, and AI-enhanced approaches for improved data integration and lifecycle management. However, challenges remain in scaling these methods to industrial contexts, automating ontology evolution, and incorporating advanced optimization techniques into a unified graph-based framework.

2.2. Prior work and felimitation from the state of the art

This paper builds on a rich body of research that highlights the synergy of semantic technologies, AI-driven techniques, and MBSE. Like Heissen et al. (2024) and Kulkarni et al. (2024), our work leverages LLMs to interpret and transform unstructured data into structured MBSE artifacts, thereby reducing manual modeling effort (Heissen et al., 2024; Kulkarni et al., 2024). We also draw on established best practices in SysML-to-knowledge-graph transformations (Fu et al., 2021) and ontology-based semantic layers (Zindel et al., 2022), reusing these core concepts to ensure interoperability and standardized data representation. In this respect, our approach shares with the state of the art a reliance on robust ontologies, graph-based data structures, and a recognition of LLMs' capacity to streamline MBSE workflows. **What differentiates** our solution is the systematic integration of RAG with knowledge graphs to form an adaptive architecture induction pipeline. Unlike prior studies that focus primarily on static rule-based transformations e.g., (Fu et al., 2021) or domain-specific semantic layers (Zindel et al., 2022), we emphasize an adaptive, LLM-augmented synthesis of system architectures. This paper goes further than typical Digital Thread architectures (Kasper et al., 2024) by embedding LLMs directly into the querying and modeling process, facilitating real-time expansions and modifications to the underlying graph. Moreover, while prior work underscores the importance of early trade studies and design space exploration (Timperley et al., 2024), our framework aims to make these activities more accessible through intuitive, natural-language interfaces rather than specialized modeling editors. Thus, while we inherit key ideas like graph-based data representation and SysML alignment from existing literature, we extend them with a focus on adaptive AI and user-centric design, establishing a new methodological foundation for MBSE workflows.

2.3. Challenges and research gaps

Complexity and Scalability of Large Graphs: As systems grow, their corresponding knowledge graphs can become exceedingly large and interdependent, leading to computational bottlenecks in graph analytics (Dwivedi, 2018). **Data Quality, Integration, and Tool Interoperability**: Heterogeneous data formats, modeling languages, and vendor-specific tools hinder seamless integration. Ensuring consistent data quality across these diverse sources remains an ongoing challenge (Fu et al., 2021; Schweitzer, 2023; Schweitzer et al., 2022). **Ontology Development and Maintenance**: Building and updating ontologies require significant domain expertise. The lack of automated processes for ontology evolution can stall the adoption of knowledge-graph-based MBSE in rapidly changing engineering environments (Fu et al., 2021; Schweitzer, 2023). **Proprietary Data Formats and Complex MBSE Tool Interfaces**: Many MBSE solutions rely on non-interoperable formats that complicate data sharing. Additionally, steep learning curves for specialized tools can deter users from fully leveraging MBSE's benefits. **Security Concerns and Dynamic Process Integration**: Aggregating design data in a centralized repository raises confidentiality risks, especially when multiple stakeholders are involved. Furthermore, integrating dynamic processes such as quality and change management into a unified knowledge graph is still a research frontier (Dwivedi, 2018). **Generative AI for adaptive Architecture Induction**: While LLMs show promise, many existing frameworks do not fully exploit generative AI to automatically evolve system architectures based on emerging data or requirements (Timperley et al., 2024). **Constraint Programming and Evolutionary Algorithms within Knowledge Graphs**: Few studies investigate how advanced optimization techniques can be embedded within a knowledge graph framework to systematically configure and instantiate complex system architectures (Schweitzer, 2023; Timperley et al., 2024; Zindel et al., 2022).

2.4. Research objectives of this paper

A primary motivation for these research objectives is to address the real-world obstacles highlighted in the preceding challenges and research gaps. The **first objective, Automatic MBSE-Graph Construction**, tackles issues of data quality, integration, and interoperability by unifying historically siloed engineering data into a single knowledge graph. This consolidation also helps mitigate the skill barrier linked to ontology development and maintenance, as it partially automates model creation from unstructured data. Hence, ***Research Question 1*** arises: *Can LLMs assist in structuring unstructured data and support the automated creation of MBSE-graphs?* The **second objective, Enhanced Usability and**

Information Retrieval via MBSE-Graph-RAG, targets the usability barriers caused by proprietary data formats and specialized MBSE tool interfaces. By leveraging LLMs for natural-language queries, it aims to reduce the expertise required to navigate and analyze complex system architectures. This leads to ***Research Question 2***: *Does the use of MBSE-Graph-RAG improve user interaction with system architectures?* Taken together, these objectives serve as a foundational response to multiple identified gaps, including the need for generative AI techniques, robust data consolidation, and greater scalability. They also establish the groundwork for potential future extensions, such as incorporating constraint-based optimization.

3. Development and demonstration proof of concept MBSE-Graph-RAG

The MBSE-Graph-RAG framework is guided by two main theories, both directly addressing the research objectives ***(1) Automated Knowledge-Graph Construction Enhances Data Accessibility***: We posit that structured knowledge graphs improve data accessibility and interoperability in MBSE by consolidating historically siloed information. Under this assumption, if LLMs can parse unstructured or semi-structured text to generate consistent nodes and relationships, the resulting graph will reduce manual modeling effort and mitigate skill barriers tied to ontology development. ***(2) Natural-Language Interaction Lowers Usability Barriers***: We further hypothesize that integrating LLMs for natural-language queries will enhance system-model usability. By translating everyday language into precise graph queries (e.g., Cypher), the framework aims to address common MBSE tool challenges like proprietary formats and steep learning curves. Ultimately, if LLM-driven interaction proves intuitive, even non-experts should be able to retrieve and analyze system data with minimal training. The following Figure 1 shows the underlying solution architecture.

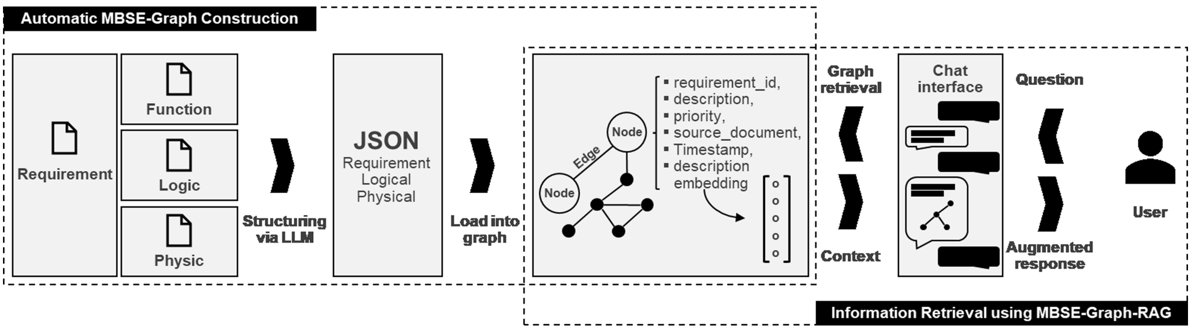

Figure 1. Solution architecture integrating knowledge graphs and LLMs in the MBSE-Graph-RAG framework

3.1. Automatic MBSE-graph construction

The first design component focuses on capturing and organizing MBSE data into a Neo4j-based knowledge graph. LLMs are used to interpret raw text, identify key MBSE entities (e.g., requirements, functions, logical elements), and place them in a coherent graph structure. The process: **Data Generation from Unstructured Engineering Documents**: We begin by extracting relevant information (requirements, functions, logical elements) from synthetic engineering documentation. LLMs receive domain-specific prompts to ensure that each entity is captured with sufficient detail and traceability. **Standardizing Terminologies and JSON Data**: To maintain consistency, extracted information is normalized into structured JSON, aligning with SysML-like attributes. This step mitigates common MBSE challenges around mismatched nomenclatures and tool-dependent formats. **Node & Edge Representation**: Each entity—requirement, function, or logical element—is mapped to a node with attributes (e.g., *requirement_id*, *function_id*), while permissible relationships (e.g., *"Refine"*, *"Satisfy"*) are enforced via constraints. Permitted constraints are displayed in the Table 2 below. This ensures semantic rigor and guards against invalid connections, preserving the integrity of the graph. By automating these tasks, the framework addresses Research Objective 1—namely, demonstrating that LLM-driven processes can reduce manual overhead and unify MBSE data into a more accessible format.

Table 2. Matrix permitted relations between system elements

Row to column	Requirement	Function	Logical
Requirement	Derive, dependency	Refine	Refine
Function	Satisfy	Connector, composition	
Logical	Satisfy		Connector, composition

3.2. MBSE-Graph-RAG for natural language interaction

The second design pillar leverages Graph Retrieval-Augmented Generation (GraphRAG) to enable natural-language queries on top of the MBSE-Graph. Key elements include:
LLM-Driven Query Translation: User queries such as *"Which functions satisfy the requirement 'The vehicle shall be energy-efficient'?"* are parsed by an LLM, which generates corresponding Cypher queries. **Natural-Language Interface**: A chat-based user interface reduces the learning curve by allowing engineers to ask questions in plain English. The LLM dynamically constructs or refines search criteria, freeing users from specialized query languages. **Real-Time Retrieval and Visualization**: Query results are displayed both textually and through subgraph visualizations, enabling more intuitive exploration of relationships and dependencies within the system. This addresses Research Objective 2, showing how LLM integration in MBSE can simplify model navigation and foster a more inclusive user experience. The following Figure 2 displays the user interface:

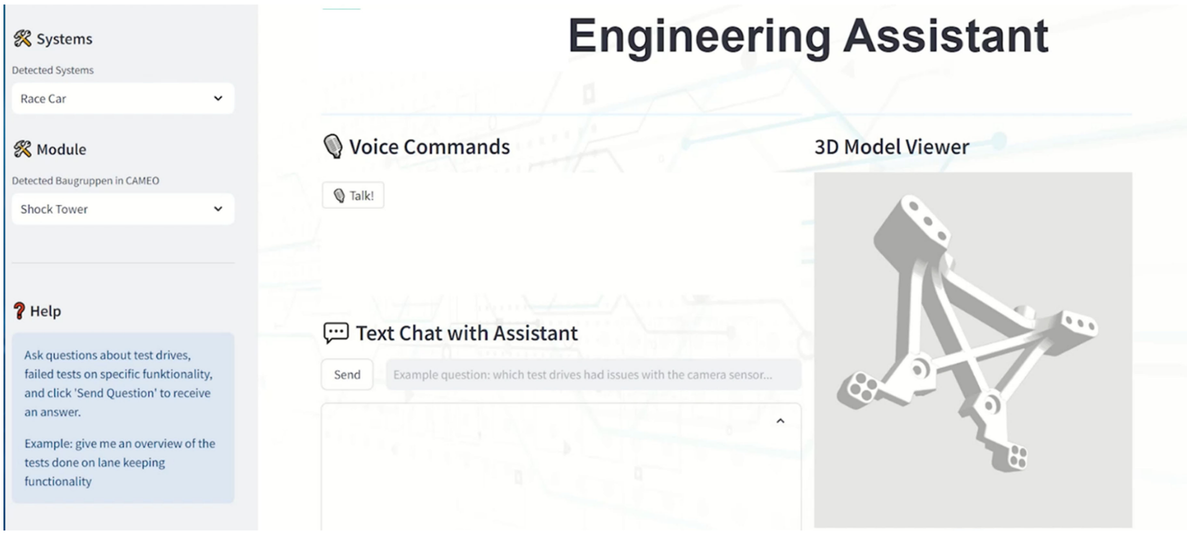

Figure 2. Screen capture user interface with chat area, voice command and preview area

3.3. Demonstration

To validate the proposed design, we implemented a proof-of-concept prototype, applying the above processes in a scenario involving synthetic automotive data. Users can issue queries to the MBSE-Graph —stored in Neo4j—via a chat-based interface. For instance, when asked, *"Trace all requirements linked to the function 'Optimize fuel efficiency,'"* the system automatically: **Interprets the Query**: The LLM transforms the natural-language prompt into a Cypher query, following the semantic rules established in the MBSE-Graph. **Retrieves and Presents Information**: Neo4j returns the relevant nodes and edges, which are then displayed as a text summary plus a subgraph visualization. This workflow confirms the system's ability to handle real-time data retrieval and to showcase how different MBSE entities interrelate.

4. Results and discussion

The MBSE-Graph-RAG framework was developed with two primary objectives:***(1)Automatic MBSE-Graph Construction*** and ***(2)Enhanced Usability and Information Retrieval via MBSE-Graph-RAG***. The **first objective** focused on automating the transformation of unstructured engineering data into a coherent knowledge graph, significantly enhancing data accessibility. The framework demonstrated its effectiveness by converting synthetic engineering artifacts into SysML-like nodes and relationships, automating parts of the ontology creation process. This result suggests a positive answer to ***Research Question 1****—Can LLMs assist in structuring unstructured data and support the automated creation of MBSE-graphs?*—as the LLMs effectively parsed complex textual inputs, reduced manual modeling efforts, and ensured consistent data representation. However, it is important to note that the testing was conducted using synthetic data, which may not fully capture the complexity and variability of real-world MBSE scenarios. This reliance limits the framework's generalizability until further validation with real-world datasets is performed. The **second objective** aimed to improve usability by integrating a natural-language querying mechanism, allowing users to interact with MBSE data without needing specialized syntax or deep expertise in MBSE tools. The system enabled natural language queries to be translated into Cypher queries, providing accurate and contextual information retrieval. This approach directly addresses ***Research Question 2****—Does the use of MBSE-Graph-RAG improve user interaction with system architectures?*—by demonstrating that participants could navigate and visualize large sets of MBSE entities more intuitively than with traditional MBSE tools.

4.1. Challenges and limitations

Despite these promising results, the study revealed several challenges and limitations. One major issue is the scalability of the framework when dealing with large, highly interconnected graphs. Real-time updates were sometimes slow, and visual feedback lagged, which could hinder usability in industrial-scale scenarios. The dependency on synthetic data also limits the system's applicability, as real-world datasets might introduce complexities and challenges not encountered during testing. The natural language processing (NLP) capabilities of the system need enhancement to handle ambiguous inputs, as such cases occasionally led to misaligned or incomplete queries.

4.2. Contributions and future outlook

The MBSE-Graph-RAG framework represents a significant advancement in bridging the gap between generative AI and data consolidation within MBSE. By integrating knowledge graphs with RAG techniques, the framework not only simplifies data management but also enhances usability, making MBSE tools more accessible to non-expert users. This increased accessibility has the potential to drive broader adoption within the industry. The proof-of-concept demonstrated tangible improvements in efficiency and data accessibility, showcasing how natural language interactions and automated graph construction can transform traditional MBSE practices. **Future research** will focus on validating the framework with real-world MBSE datasets to ensure robustness and scalability. Key development areas include enhancing NLP capabilities, optimizing visual performance for large and complex datasets, and incorporating domain-specific datasets to improve generalizability. Additionally, extending the framework to support advanced features such as automated ontology evolution, constraint-based optimization, and dynamic digital thread management will amplify its impact. These enhancements are expected to refine the framework's effectiveness, contributing to faster innovation cycles and higher-quality engineering outcomes. **Three primary research directions** have been identified to maximize the MBSE-Graph-RAG framework's potential: **(1) MBSE-Graph Metamodel with Advanced Embedding Capabilities**: A significant research direction involves developing an MBSE-Graph metamodel that can represent complex embedding rules, such as "include", "exclude" and other feature-model-like constraints. This capability is crucial for accurately modeling dependencies and variability within system architectures, ensuring a flexible yet consistent representation of product configurations. **(2) Automated AI-Augmented Induction of MBSE Graphs**: The framework aims to advance the automated creation of MBSE-graphs using AI technologies. However, achieving this at scale requires robust ontology development and maintenance mechanisms, as highlighted in the challenges identified in Section 2.3. Automating the evolution of these ontologies to adapt to changing requirements remains a key focus area. **(3) Automated Instantiation of System Architectures for Product Configurations**: One of the most critical and complex tasks is synthesizing concrete system architectures for specific product configurations. This process involves ensuring compliance with compatibility constraints, which has not yet been fully addressed in existing solutions. Effective design space exploration relies heavily on evolutionary algorithms, which can efficiently navigate complex option spaces to identify optimal configurations. For contradiction-free instantiation, the integration of constraint programming is particularly promising, as it can enforce compatibility rules and maintain system consistency.

5. Conclusion

This paper establishes the methodological foundation for the conceptual application of RAG in MBSE-Graphs. Through a structured literature review, the need and motivation were systematically compiled, and the underlying problem was analyzed. Based on these insights, the developed concept provides an initial outlook for future research and demonstrates its feasibility by presenting a proof of concept. The MBSE-Graph-RAG framework addresses critical challenges in MBSE by integrating generative AI with knowledge graphs, enhancing data accessibility, and enabling natural language interaction with system architectures. By transforming proprietary MBSE data into a graph-based format using Neo4j, the framework improves usability and lowers the entry barrier for non-expert users. The proof-of-concept demonstrated how the system automates the conversion of unstructured engineering data into structured knowledge graphs, reducing manual modeling effort and enabling efficient querying through intuitive natural language interfaces. While the results are promising, several limitations remain. The framework currently relies on synthetic data, which may not fully capture the complexity of real-world MBSE scenarios. Challenges also exist with scalability and the robustness of NLP. To fully realize its potential, future work will focus on validating the framework with real-world datasets, enhancing NLP capabilities, and optimizing visualization tools for large and complex graphs. Additionally, extending the system with advanced features like automated ontology evolution, constraint-based optimization, and support for complex embedding rules in the MBSE-Graph metamodel will be crucial. These advancements aim to establish a scalable and adaptive MBSE framework, promoting faster innovation and higher-quality engineering outcomes through enhanced automation, improved usability, and more comprehensive engineering workflows. Conducting usability studies with diverse user groups will provide valuable insights to refine the system further and ensure it meets the needs of both experts and non-experts effectively.

References

Borth, M., & van Gerwen, E. (2018). Data-driven Aspects of Engineering The Use of Operational Data in SoS Engineering: Chances and Challenges. *In 2018 13th System of Systems Engineering Conference (SoSE): June 19-22, 2018, Sorbonne université, campus Pierre et Marie Curie*, Paris, France (pp. 219–224). IEEE. https://doi.org/10.1109/SYSOSE.2018.8428711

Chami, M., Zoghb, C., & Bruel, J.-M. (2019). *A first step towards AI for MBSE: Generating a part of SysML models from text using AI.* https://www.researchgate.net/profile/mohammad-chami/publication/337338933_a_first_step_towards_ai_for_mbse_generating_a_part_of_sysml_models_from_text_using_ai/links/5dd4fc4da6fdcc37897a73e9/a-first-step-towards-ai-for-mbse-generating-a-part-of-sysml-models-from-text-using-ai.pdf

Dwivedi, A. (2018). Implementing Cyber Resilient Designs through Graph Analytics Assisted Model Based Systems Engineering. *In 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)* (pp. 607–616). IEEE. https://doi.org/10.1109/QRS-C.2018.00106

Enzo (2024, July 11). The GraphRAG Manifesto: Adding Knowledge to GenAI. *Neo4j*. https://neo4j.com/blog/graphrag-manifesto/

Fu, C., Liu, J., & Wang, S. (2021). Building SysML Model Graph to Support the System Model Reuse. *IEEE Access*, 9, 132374–132389. https://doi.org/10.1109/ACCESS.2021.3115165

Gutierrez, C., & Sequeda, J. F. (2020). Knowledge Graphs. In M. d'Aquin & S. Dietze (Eds.), *ACM Digital Library, Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (pp. 3509–3510). Association for Computing Machinery. https://doi.org/10.1145/3340531.3412176

Heissen, O. von, Hanke, F., Mpidi Bita, I., Hovemann, A., & Dumitrescu, R. (2024). Toward Intelligent Generation of System Architectures, 504–513. https://doi.org/10.35199/NORDDESIGN2024.54

Huang, S., Jiao, X., Zhang, Q., Chen, J., Hu, F., & Yang, K. (2023, October 12 - 2023, October 15). Design and Implementation of Ground PHM Principled Sample Machine for Airborne Fuel Pump Based on Three-Dimensional Multiphase Criterion. *In 2023 Global Reliability and Prognostics and Health Management Conference (PHM-Hangzhou)* (pp. 1–7). IEEE. https://doi.org/10.1109/PHM-Hangzhou58797.2023.10482759

Kasper, N., Pfenning, M., & Eigner, M. (2024). The digital thread for system lifecycle management with a native graph database in a polyglot architecture. *Proceedings of the Design Society*, 4, 2079–2088. https://doi.org/10.1017/pds.2024.210

Kenett, R. S., Swarz, R. S., & Zonnenshain, A. (2020). Systems Engineering, Data Analytics, and Systems Thinking. In R. Kenett, R. S. Swarz, & A. Zonnenshain (Eds.), *Systems engineering in the fourth industrial revolution: Big data, novel technologies, and modern systems engineering* (First edition, pp. 1–20). Wiley. https://doi.org/10.1002/9781119513957.ch1

Kulkarni, P. J., Tissen, D., Bernijazov, R., & Dumitrescu, R. (2024). Towards Automated Design: Automatically Generating Modeling Elements with Prompt Engineering and Generative Artificial Intelligence. In *Proceedings of NordDesign 2024* (pp. 617–625). The Design Society. https://doi.org/10.35199/NORDDESIGN2024.66

Lu, J., Ma, J., Zheng, X., Wang, G., Li, H., & Kiritsis, D. (2022). Design Ontology Supporting Model-Based Systems Engineering Formalisms. *IEEE Systems Journal*, 16(4), 5465–5476. https://doi.org/10.1109/JSYST.2021.3106195

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Schweitzer, G. (2023). *Einsatz von künstlicher Intelligenz in der Produktentwicklung auf Basis eines Engineering-Graph im Kontext von Model-based-Systems-Engineering*.

Schweitzer, G., Bitzer, M [Michael], & Vielhaber, M [Michael] (2023). ENGINEERING GRAPH AS AN APPROACH TO SUPPORT DESIGN DECISIONS IN PRODUCT DEVELOPMENT. *Proceedings of the Design Society*, 3, 1625–1634. https://doi.org/10.1017/pds.2023.163

Schweitzer, G., Mörsdorf, S., Bitzer, M [M.], & Vielhaber, M [M.] (2022). Detection of Cause-Effect Relationships in Life Cycle Sustainability Assessment Based on an Engineering Graph. *Proceedings of the Design Society*, 2, 1129–1138. https://doi.org/10.1017/pds.2022.115

Timperley, L., Berthoud, L., Snider, C., & Tryfonas, T. (2024). Mapping the MBSE Environment and Complementary Design Space Exploration Techniques. *In 2024 IEEE Aerospace Conference* (pp. 1–20). IEEE. https://doi.org/10.1109/AERO58975.2024.10521188

Tissen, D., Wiederkehr, I., Koldewey, C., & Dumitrescu, R. (2023). Exploring data-driven model-based systems engineering: A systematic literature review. *In 2023 IEEE International Conference on Technology Management, Operations and Decisions (ICTMOD): 22-24 Nov. 2023* (pp. 1–6). IEEE. https://doi.org/10.1109/ICTMOD59086.2023.10438129

Webster, J., & Watson, R. T. (2002). Guest Editorial: Analyzing the Past to Prepare for the Future: Writing a literature Review.

Zindel, A., Feo-Arenis, S., Helle, P., Schramm, G., & Elaasar, M. (2022). Building a Semantic Layer for Early Design Trade Studies in the Development of Commercial Aircraft. In *2022 IEEE International Symposium on Systems Engineering (ISSE)* (pp. 1–8). IEEE. https://doi.org/10.1109/ISSE54508.2022.10005324