

# MBSE AI Platform for Productivity (MAPPy): Combining AI and Digital Engineering

Melinda Ong  
Nicholas Godshall  
Jeremiah Crane

# Agenda

---

Background

Improvements Since Last Year

Methodology

Demo Walkthrough

Next Steps and Conclusion

# AI and MBSE Background

Demand for digital engineering is notably increasing across the DoD technological landscape, driven by DoD Instruction 5000.97.

“...mandates the incorporation of digital engineering for all new program and calls for the replacement of documents with the use of digital models as the primary means of communicating system information. Starting sentence with action word”  
- DoD Instruction 5000.97

### MAPPy

Chat History

- User says hello.
- "Share Another Joke Please"
- Here's a joke
- Funny joke, please!
- Funny joke, please!
- Funny Joke Request
- "User greeting: Hello"
- Funny joke requested.
- User greeted (Hello)
- Airlock: Airtight chamber for ...
- Tissue Box Specifications
- Testing system functionality.
- "Testing system functionality"
- Crew Accommodation Requir...
- Enhanced MBSE model gener...
- Benefits: Efficient Model-Base...
- Benefits: Enhanced Design. Ef...
- Generative AI benefits: Model...
- Crew Accommodation Requir...
- Crew Accommodation Requir...
- Crew Accommodation Requir...
- Crew Accommodation Actors ...
- Crew Accommodation Requir...
- \*Generative AI aids MBSE in ...
- Generative AI values in MBSE...
- Collection Manager

What is the requirement for crew accommodation in the lunar habitat?

The requirement for crew accommodation is that "The lunar habitat shall provide a human hospital space non-human crew." This can be derived from the given context, specifically from the Requirement with its habitat shall provide a human hospital space for 5 human and 3 non-human crew."

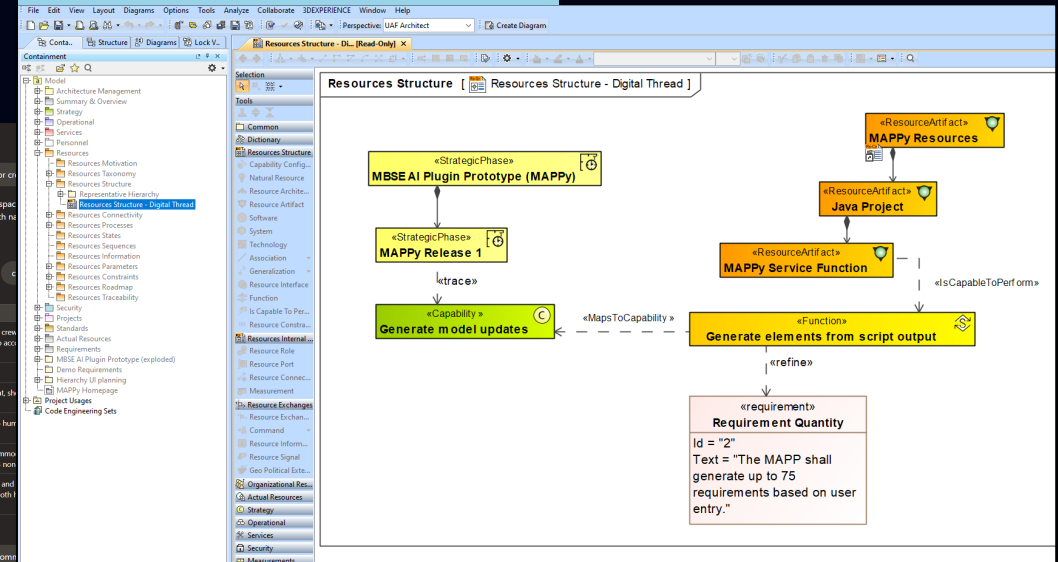
NAME	STEREOTYPE	DOCUMENTATION
Spacesuit Storage	Requirement	The system shall store spacesuits for at least 3 human crew and 3 non-human crew derived from the main requirement "3.6.6.5 The Airlock Hatch shall be sized to acco and non-human crew."
System Status	Requirement	The system shall notify users of the system operational status.
Lunar Habitat BD	Diagram	This diagram represents the Interface Breakdown Diagram for the lunar habitat, sh item flows between the Lunar Habitat, Crew Habitat, and Airlock blocks.
Lunar Habitat	Block	The lunar habitat is a modular unit that provides a human hospital space for 5 hur crew. It is composed of the Crew Habitat and Airlock subsystems.
Crew Habitat	Block	The crew habitat is a modular unit within the lunar habitat that provides accomo members. It has dimensions capable of safely accommodating 5 human and 3 non
Airlock	Block	The airlock is a modular unit within the lunar habitat that allows crew to enter and controlled environment. It is one-hand operable and sized to accommodate both crew.

NAME	TEXT	STEREOTYPE	VERIFYMETHOD
Crew Accommodations	The system shall have dimensions capable of safely accommodating 5 human and 3 non-human crew.	FunctionalRequirement	Analysis

Chat Message MAPPy

MAPPy can hallucinate. Review all AI-generated responses.

### MagicDraw MBSE Application



Integrating AI and MBSE transforms previously manual systems engineering processes, increasing efficiency & engineering rigor.

# Updates

---

## Then

### *MBSE AI Plugin Prototype*

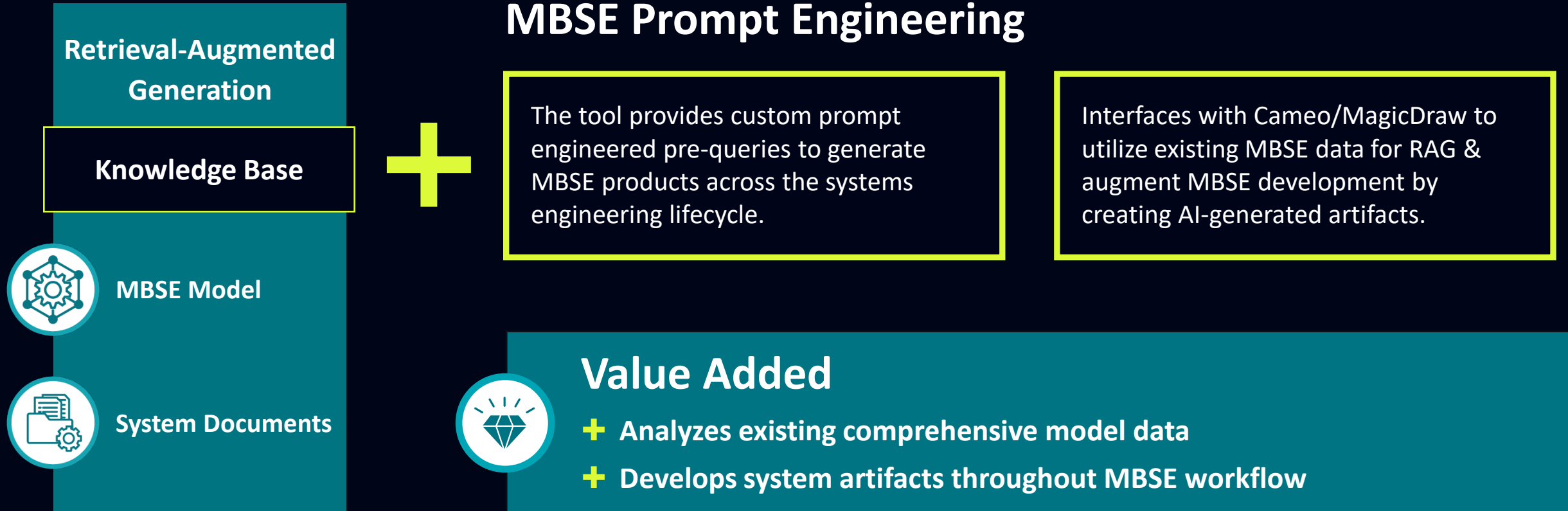
- Only a cameo plugin
- Limited to OpenAI API
- Limited to generation of requirements and blocks

## Now

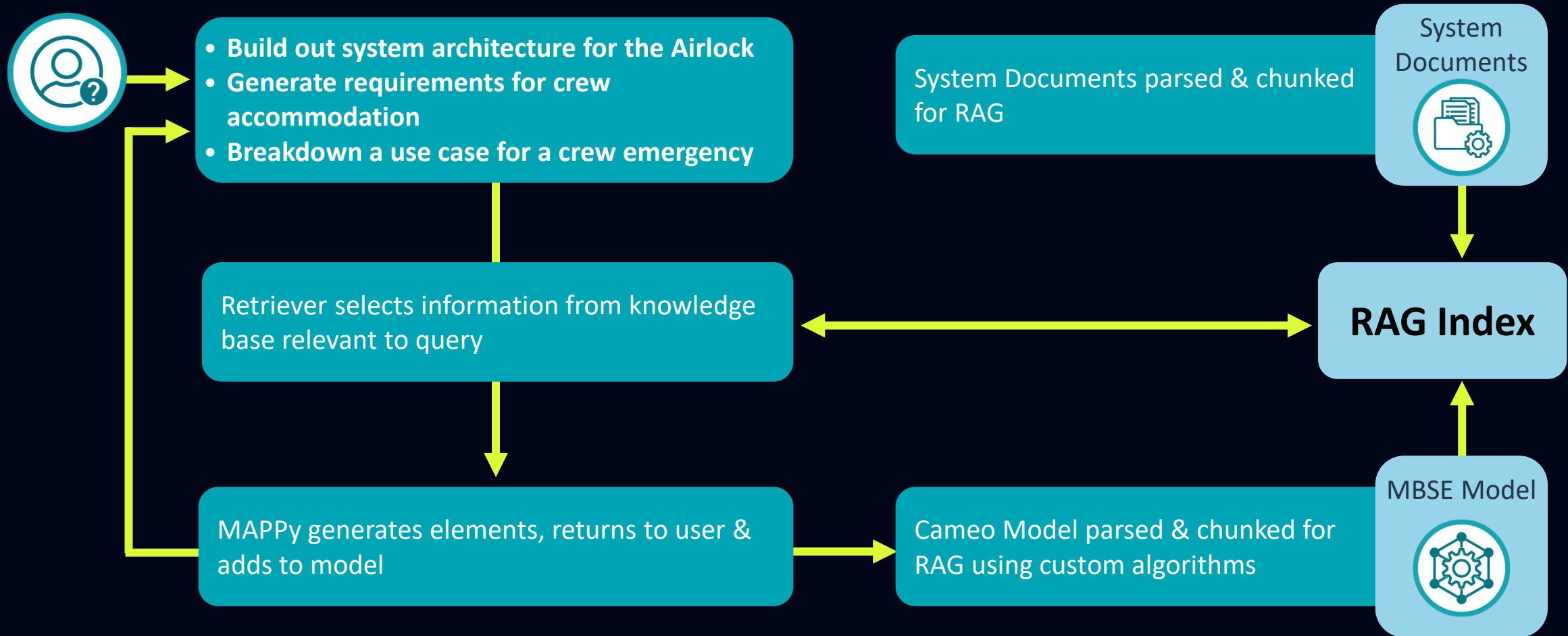
### *MBSE AI Platform for Productivity*

- Cameo plugin + webapp + custom RAG / Agent pipeline using open-source large language models.
- Uses on premises AI server for all processing
- Support for a wider array of stereotypes
- MBSE Model + Documents used for RAG
- Containerized code to facilitate ease of deployment

# Prompt Engineering



# RAG Enabled Search and Response



# Compliance with INCOSE Guidelines

## Re-prompting based on validation suite

Crew safety requirements for a lunar habitat

NAME	TEXT	STEREOTYPE	VERIFYMETHOD
LunarHabitat_SafetyEntryDoors	Lunar habitat shall feature lockable, gas-tight access doors, operable by the crew members under their own initiative.	functionalRequirement	Test
LunarHabitat_Emergency	INCOSE 4.5.1 R24: Do not use 'it', 'this', 'that', 'he', 'she', 'they', and 'them'. use.	Requirement Analysis	

The LLM is prompted to regenerate requirements not passing the built-in validation suite to deliver compliant & complete requirements.

INCOSE Guide to Writing Requirements v3.1 – Summary Sheet

Rules for Need and Requirement Statements and Sets of Needs and Requirements

<p><b>Accuracy</b></p> <p>R1 - Use a structured, complete sentence: subject, verb, object.</p> <p>R2 - Use the active voice in the main sentence structure of the need or requirement statement with the responsible entity clearly identified as the subject of the sentence.</p> <p>R3 - Ensure the subject and verb of the need or requirement statement are appropriate to the entity to which the need or requirement refers.</p> <p>R4 - Define terms in a glossary, data dictionary, etc.</p> <p>R5 - Use definite article "the" rather than the indefinite article "a."</p> <p>R6 - Use appropriate units when stating quantities. All numbers should have units of measure explicitly stated.</p> <p>R7 - Avoid the use of vague terms such as "some", "any", "allowable", "several", "many", "a lot of", "a few", "almost always", "very nearly", "nearly", "about", "close to", "almost", and "approximate".</p> <p>R8 - Avoid escape clauses such as such as "so far as is possible", "as little as possible", "where possible", "as much as possible", "if it should prove necessary", "if necessary", "to the extent necessary", "as appropriate", "as required", "to the extent practical", and "if practicable.."</p> <p>R9 - Avoid open-ended clauses such as "including but not limited to", "etc." and "and so on..".</p> <p><b>Concision</b></p> <p>R10 - Avoid superfluous infinitives such as "... be designed to be able to..." or "... be capable of..."</p>	<p><b>Completeness</b></p> <p>R24 - Avoid the use of pronouns and indefinite pronouns.</p> <p>R25 - Avoid relying on headings to support explanation or understanding of the requirement.</p> <p><b>Realism</b></p> <p>R26 - Avoid using unachievable absolutes such as 100% reliability, 100% availability, all, every, always, never, etc.</p> <p><b>Conditions</b></p> <p>R27 - State applicability conditions explicitly.</p> <p>R28 - Express the propositional nature of a condition explicitly for a single action instead of giving lists of actions for a specific condition.</p> <p><b>Uniqueness</b></p> <p>R29 - Classify the needs and requirements according to the aspects of the problem or system it addresses.</p> <p>R30 - Express each need and requirement once and only once.</p> <p><b>Abstraction</b></p> <p>R31 - When defining design inputs avoid stating a solution unless there is rationale for constraining the design. Focus on the problem "what" rather than the solution "how."</p> <p><b>Quantifiers</b></p> <p>R32 - Use "each" instead of "all", "any" or "both" when universal quantification is intended</p> <p><b>Tolerance</b></p>
--	--

The validation suite checks the LLM generated requirement text for compliance with the INCOSE Guide to Writing Requirements (when applicable).

MAPPy leverages INCOSE guidelines for writing requirements to automatically validate its own responses.

# Promptfoo

- + Models, prompts and other factors can be systematically tested using promptfoo
- + Queries are repeatedly sent to llm, with single variable changed, in this case the model being used
- + Each response is tested against the defined ruleset, and a final score is assigned

The screenshot shows the Promptfoo interface with three model test results:

- mistral:latest**: 2 FAIL 9 PASS (0.90). INCOSE 4.6.1 R26: Do not use '100%', 'all', 'every', 'always', and 'never'.
- llama3:70b** (highlighted): 4 FAIL 7 PASS (0.84). Includes text before table formatting.
- gemma:2b**: 1 FAIL 10 PASS (0.98). INCOSE 4.1.7 R7 Vague Terms: Do not use 'some', 'any', 'allowable', 'several', 'many', 'a lot of', 'a few', 'almost always', 'very nearly', 'nearly', 'about', 'close to', 'almost', and 'approximate'.

Pass	Score	Reason
✗	0.00	Includes text before table formatting
✓	1.00	n/a
✓	1.00	n/a
✓	1.00	n/a
✗	0.60	INCOSE 4.2.1 R10: Do not use 'able to' and 'capable of'.
✗	0.80	INCOSE 4.3.5 R16: Do not use the word 'not'.
✗	0.80	INCOSE 4.5.1 R24: Do not use 'it', 'this', 'that', 'he', 'she', 'they', and 'them'.
✓	1.00	n/a
✓	1.00	n/a
✓	1.00	n/a
✓	1.00	n/a

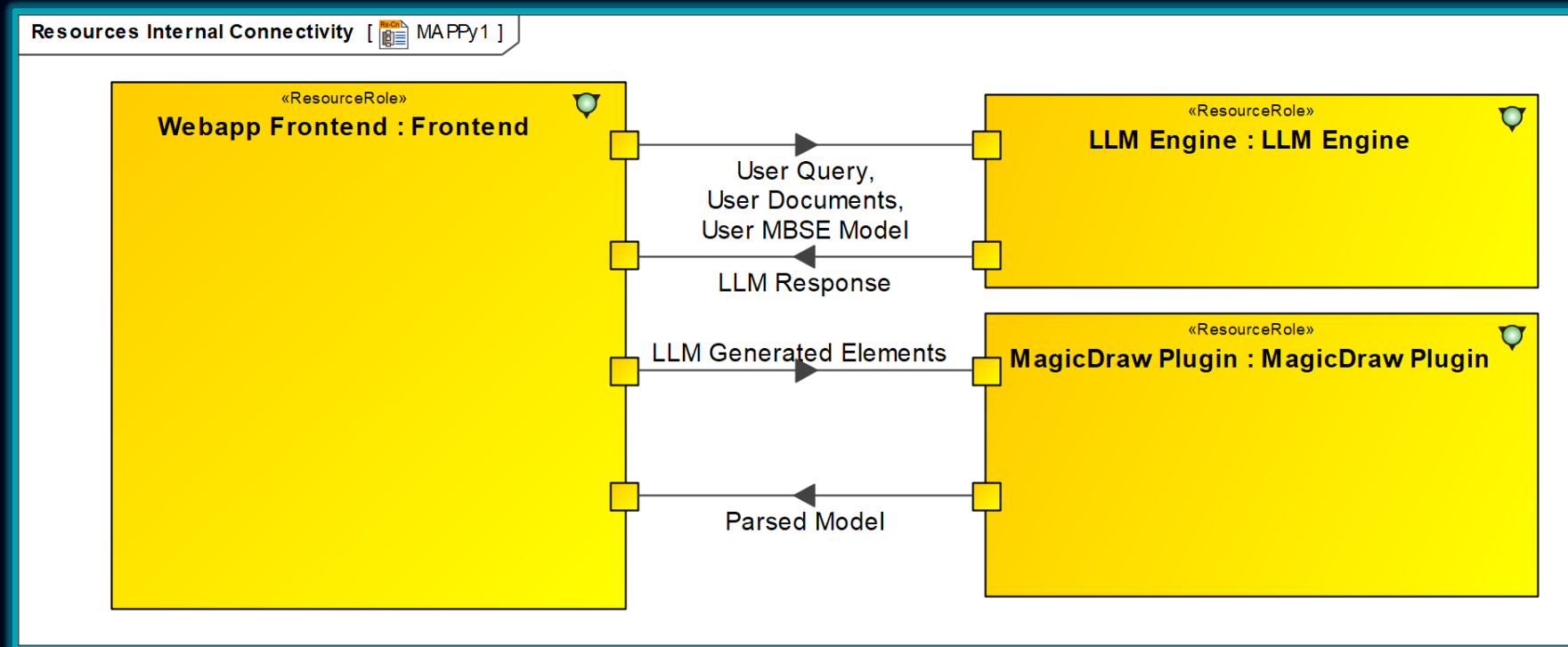
Using the scores from promptfoo, variables such as the large language model, pre-prompts, or RAG parameters can be made using objective data.



# System Engineering for Artificial Intelligence (SE4AI)

**Frontend:** Provides a user-friendly interface to generative AI capabilities. Users may upload documents and query a fine-tuned LLM.

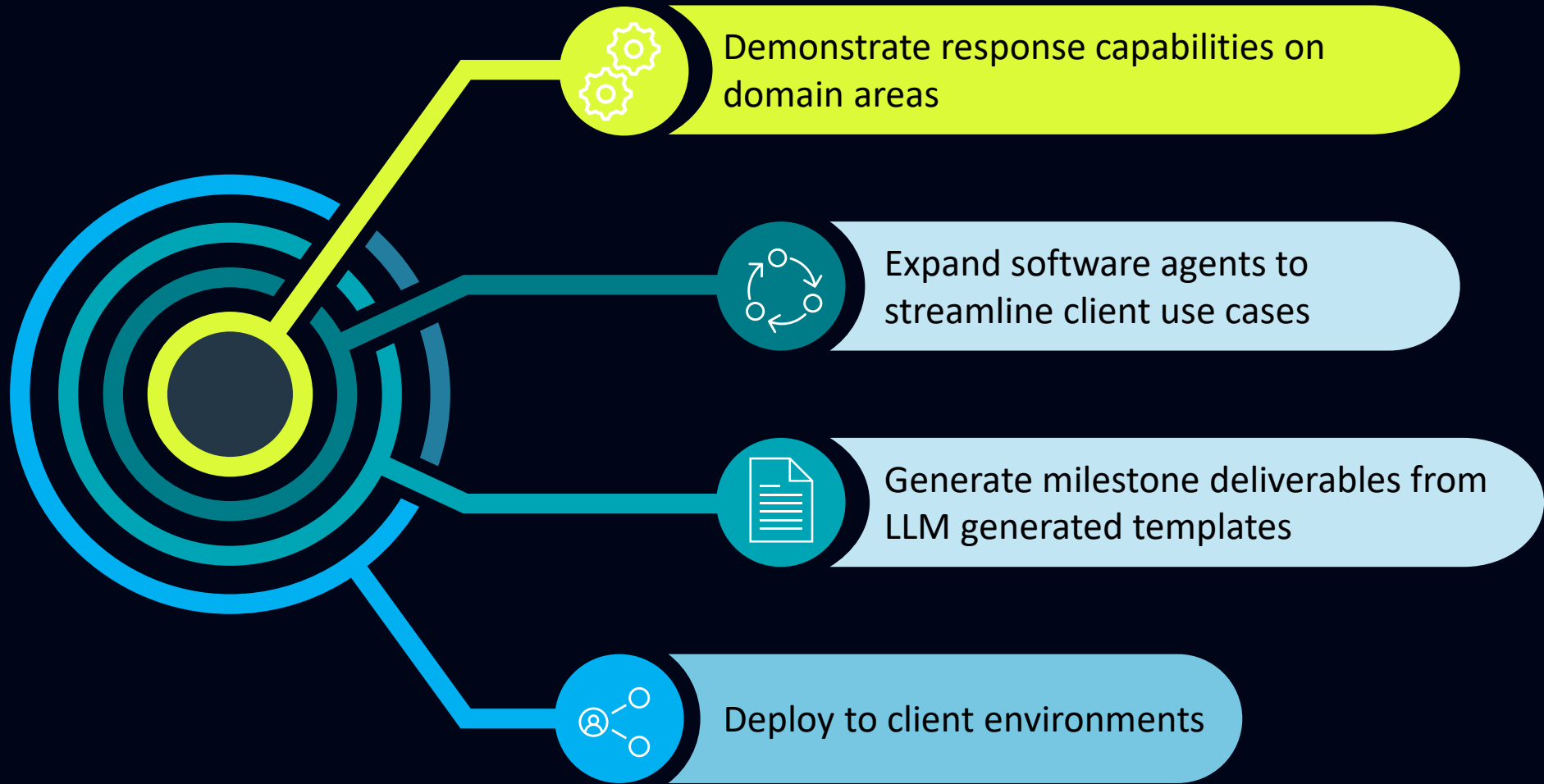
**Backend:** Optimizes document embeddings into chunks for RAG. Through NLP techniques, the LLM is to generate a comprehensive response which automates the creation of MBSE products.



The tool's internal connectivity showcases how resources interact, capturing all necessary interfaces considered for design

# Next Steps

---



# Conclusion

---

Demonstrates successful integration of generative AI and MBSE development

Facilitates search and response for program knowledge retrieval

Provides tailored prompts for SE lifecycle product development support

Enables developer extensibility with the modular architecture and API

# Thank You

---

For more information contact:



**NICK GODSHALL**

*Staff Engineer*  
*Godshall\_Nicholas@bah.com*



**MELINDA ONG**

*Lead Engineer*  
*Ong\_Melinda@bah.com*



**JEREMIAH CRANE**

*Chief Engineer*  
*Crane\_Jeremiah@bah.com*



**LEONARD BROWNLOW**

*Distinguished Engineer*  
*Brownlow\_Leonard@bah.com*