GenAI Workbench

AI-Assisted Analysis and Synthesis of Engineering Systems from Multimodal Engineering Data

H. Sinan Bank Daniel R. Herber

sinan.bank@colostate.edu
Colorado State University – Department of Systems Engineering
CSU Systems Engineering Mini Conference, 2026

Outline

1

Introduction

Motivation: The Fragmented Engineering Stack

- Modern engineering design platforms excel at discipline-specific (e.g., eCAD, mCAD) and task-specific (e.g., CAD, CAM, CAE) tools, but lack native systems engineering frameworks
- System-level requirements and architectures typically live in separate tools of their own — distinct from the detailed component design tools

- Consequences:
  - Information silos that *break the digital thread*
  - Verification & validation fragmented across the product lifecycle
  - Increased integration risk and late-stage rework

Motivation: this fragmentation motivates GenAI Workbench — an integrated, comprehensive, and intelligent paradigm for systems engineering.

Question ?

> How can we bridge the gap between unstructured requirements, physical geometry, and system-level architecture — all within the designer's primary workflow?

→ Step-by-Step Methodology and Research Contribution

Publication-driven validation: the viability of the workbench is argued step by step through a sequence of papers, each covering one facet of the framework:

- CatalogBank [1] — structured catalog dataset & semi-automatic annotation tool (*DocumentLabeler*) for engineering documents (mCAD-leaning instantiation)
- Retrieval-Bench [2] — extends the CatalogBank notion into eCAD: multimodal dataset & retrieval benchmark across *BOMs, schematics, and netlists* on OSHWA-certified hardware *(under double-blind review)*
- GenAI Workbench [3] — integrated conceptual framework *(this talk)*
- Semantic–Physical–Relational linking [4] — multimodal linking mechanisms and alignment metrics
- Architecture Synthesis [5] — text $\rightarrow$ DSM generation from requirements
- Design-OS [6] — specification-driven framework for engineering system design

Central hypothesis: embedding AI-driven systems engineering capabilities directly into the designer's workflow establishes a unified digital thread that enhances engineering effectiveness.

Proposed instrument: the GenAI Workbench — a methodological proof-of-concept.

2

Background

State of the Art in AI for MBSE

- **AI-driven requirements engineering**
  - LLMs (GPT-4 [7], LLaMA-2 [8]) enable zero/few-shot extraction, but accuracy is limited (∼45–52% for formal spec extraction [9])
  - Two-step “annotate-then-convert” pipelines [9] mitigate hallucinations by narrowing the task
  - Motivates a multi-stage refinement approach
- **Vision-language models on engineering documents**
  - Hybrid YOLO + transformer parsers (e.g., Donut) extract dimensions and tolerances from drawings (93.5% F1 [10])
  - Fine-tuned VLMs currently outperform general-purpose multimodal LLMs on technical content [11]
- **Emerging MBSE Co-Pilot visions** [12]
  - GPT-4 integrated into SysML environments to auto-generate model elements [13]
  - Feeding MBSE models into LLM assistants reduces hallucinations in requirement generation [14, 15]

→ Graph-Based Integration and the Multimodal Gap

Graph-Based Integration

- **Graph databases for MBSE** (e.g., Neo4j-backed SysML stores [16]) enable traversal queries that traditional SysML tools struggle with
- **Intercax Syndeia** [17] and **OpenMBEE** [18] provide federated graph links across CAD, requirements, and simulation
- **Lifecycle-artifact graphs** [19] unify requirements, CAD parts, and process plans as typed edges across the digital thread

Multimodal Gap

- These tools link artifacts; they do *not* link semantic content to physical features
- Standards such as OSLC [20] define URI-based linking across tools, but leave semantic interpretation to the engineer
- **Our target:** link requirements directly to the elements of physical viewpoint (e.g., geometric features), not to CAD files as opaque artifacts

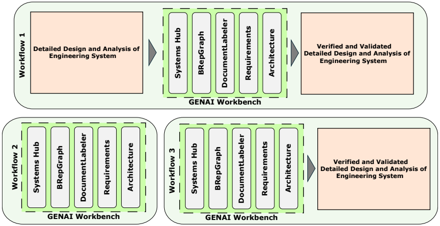
Important !

Open gap: Bridging *semantic* (text) ↔ *geometric* (B-rep) ↔ *relational* (graph) data through a unified MBSE workflow.

3

Methodology

→ GenAI Workbench — Proposed AI-Assisted Workflow



The figure shows three complementary workflows:

- (1) CAD-to-CAD mediator — Workbench acts as an *interface for CAD*: existing design in, processed (verified & validated) design out — a geometry-level interop bridge *with semantics*
- (2) Standalone Workbench — used on its own (no external CAD I/O); engineers work *inside* the Workbench via its components (SystemsHub, DocumentLabeler, BRepGraph, Requirements, Architecture)
- (3) Workbench $\rightarrow$ CAD — Workbench as the *source*: synthesizes from internal documents/requirements and outputs a verified & validated detailed design to CAD

Interoperability hinges on open formats (`STEP`/AP242, PDF, JSON) and the PLM/CAD backbone's programmable API.

System Architecture & Technical Foundations

PLM/CAD Backbone

- Programmable API (Python) for geometry and automation
- `STEP`/AP242 with B-rep access via an open geometry kernel
- Extensible for custom workbenches

Multimodal Linking

- A combination of complementary mechanisms ties *Document*, *Geometry*, and *Graph* together — no single identifier carries the whole load

Remark

Multimodal linking lets a requirement change ripple through geometry and architecture — details in [4].

→ Document Ingestion & Multimodal Linking

Document Ingestion

- xLM (and OCR) extraction of raw text (and images) from source PDFs/specs
- Prompt-guided identification of requirement statements
- Each requirement stored as a first-class PLM object, trace-linked to its source text
- Automatic compilation of a glossary of key terms

Multimodal Linking

- Binds each component's *semantic*, *geometric*, and *relational* representation
- Combination of complementary mechanisms rather than a single identifier
- Current implementation and evaluation: [4] (in preparation)

Remark

Architecture synthesis (text $\rightarrow$ DSM) is under review at the *Wiley Journal of Systems Engineering* [5], where we evaluated a range of proprietary and open-source LLMs across *hundreds of experiments* — not part of the current workflow slice.

→ Human Refinement & CAD Linking

Human-in-the-Loop Refinement

- Engineer reviews requirements: edits phrasing, corrects misinterpretations, adds missing items — PLM objects updated in place
- Reviews and edits the multimodal links (component ↔ geometry ↔ graph) through the UI [4]
- Behavior and script refinement follows the specification-driven loop of [6]
- **AI jumpstarts the tedious work; the human validates domain knowledge**

Linking with CAD

- Each confirmed system component is tied to its CAD part/assembly
- One-to-one correspondence between architecture nodes and CAD B-rep entities
- Enables bidirectional traceability: *requirement* ↔ *component* ↔ *geometric feature*

Remark

> The multimodal data model ties *Document*, *Geometry*, and *Graph* together through a combination of linking mechanisms — the technical backbone of the digital thread.

→ Core Components: Implementation Architecture

Project Database — SQLAlchemy persistence root, accessed through the hub.

Systems Hub — UID-linked registry and cross-modal index.

DocumentLabeler & **BRepGraph** — parallel extractors: semantic content from PDFs/images [1] and face-adjacency graphs from STEP B-Rep geometry; both feed the hub.

Requirements — first-class verification objects linked to architecture elements.

Architecture — hierarchical decomposition supporting states and analysis scripts.

Canvas — shared node-graph editing interface between Requirements and Architecture.

→ Verification & Validation within the Framework

Cross-Modal Compatibility

- Geometric compatibility: spatial alignment and dimensional consistency between connected parts
- Functional/Behavioral compatibility: parameters, capacities, operating ranges (from document annotations)
- Relational compatibility: graph-level consistency of connection types

Formal Verification

- Natural-language engineering constraints translated into formal specifications — e.g., grammars, temporal logics, symbolic AI techniques
- System models evaluated against derived specifications for behavioral and temporal properties
- Extends verification *beyond* pairwise checks

Important ! Continuous V&V is a product of the integrated data model, not an afterthought.

4

Applications

➞ Anticipated Impact — Realizing the Digital Thread

- UID-based integration makes end-to-end traceability practical rather than theoretical:
  - Click a CAD feature → see the requirements it satisfies
  - Query "*show all components affected by this requirement change*" and receive an accurate, automatically-maintained answer
- **Accelerating early-stage architecture exploration**
  - Weeks of manual decomposition collapse into hours of AI-assisted iteration
  - Architecture generation becomes a parameterizable, repeatable, comparable computational task
  - Early-phase decisions have the highest leverage — a broader design space earlier exposes superior architectures that would otherwise be missed

➞ Walk-through: Roller Runout Inspection

- One assembly, five panels — the same Misumi catalog entry flows through the whole workbench.
- **1. DocumentLabeler** — vendor PDFs ingested and tagged with the target schema (e.g., FBS, SysML, or custom).
- **2. Systems Hub** — documents, images, geometries, and other assets centralized for easy access.
- **3. BRepGraph** — parts as nodes; edges coloured by *intersecting / touching / adjacent / contains*.
- **4. Architecture** — whole/part block diagrams, graph, DSM-like, and list representations.
- **5. Requirements** — 18 shall-statements (L1–L3, Functional / Performance / Interface) seeded from the design.
- Same pipeline can be extended to different domains, identical flow.

5

Conclusion

→ Contributions

- A conceptual framework for integrated, multimodal systems engineering
- A unified multimodal data model based on UIDs that links semantic, geometric, and relational representations of the same system
- Three proposed AI-assisted workflows for analysis, synthesis, and generation of system models
- A hierarchical V&V framework combining cross-modal rule checks with formal verification
- An implementation strategy using an open-source PLM/CAD stack and an accessible geometry kernel

Remark

> The GenAI Workbench is a *scientific instrument*: its purpose is to demonstrate that the proposed integration is computationally feasible and methodologically sound — *for now* not to replace the commercial packages.

→ Limitations and Future Work

Current Limitations

- **Metrics & experimentation:** the proposed workflows still need broader empirical evaluation
- **Human studies:** formal user studies with engineering designers have not yet been conducted
- **Scope of validation:** so far focused on catalog-style documents; industrial requirement documents remain to be exercised

Future Work

1. **Implementation and validation:** complete the proof-of-concept and run user/case studies
2. **Models:** adopt spatially graph-aware approaches (e.g., graph-aware transformers) on the geometry side of the pipeline
3. **Agentic synthesis and verification:** autonomous Python analysis scripts inside the workbench, moving beyond human-in-the-loop toward more autonomous verification

→ References

[1] H. S. Bank and D. R. Herber. "CatalogBank: A Structured and Interoperable Catalog Dataset with a Semi-Automatic Annotation Tool (DocumentLabeler) for Engineering System Design". *Proceedings of the ACM Symposium on Document Engineering 2024*. 2024

[2] H. S. Bank. "Retrieval-Bench: Evaluating Agentic LLM Retrieval of Related Design Data Across BOMs, Schematics, and Netlists on OSHWA-Certified Hardware". Under double-blind review at ICLAD 2026; no preprint publicly released. 2026

[3] H. S. Bank and D. R. Herber. "GenAI Workbench: AI-Assisted Analysis and Synthesis of Engineering Systems from Multimodal Engineering Data". *Proceedings of the 2026 IISE Annual Conference*. Preprint: https://arxiv.org/abs/2603.00251. 2026. arXiv: 2603.00251. URL: https://arxiv.org/abs/2603.00251

[4] H. S. Bank and D. R. Herber. "Linking Semantic, Physical, and Relational Elements of CatalogBank for Engineering Design". In preparation for submission to ASME Journal of Computing and Information Science in Engineering (JCISE). 2026

[5] H. S. Bank and D. R. Herber. "Retrieval Augmented (Knowledge Graph), and Large Language Model-Driven Design Structure Matrix (DSM) Generation of Cyber-Physical Systems". *arXiv preprint arXiv:2602.16715* (2026)

[6] H. S. Bank and D. R. Herber. "Design-OS: A Specification-Driven Framework for Engineering System Design with a Control-Systems Design Case". Preprint: https://arxiv.org/abs/2603.20151. 2026. arXiv: 2603.20151 [cs.SE]. URL: https://arxiv.org/abs/2603.20151

→ References (Continued)

[7] OpenAI. *GPT-4 Technical Report*. 2023. arXiv: 2303.08774. URL: https://arxiv.org/abs/2303.08774

[8] H. Touvron, L. Martin, K. Stone, et al. "Llama 2: Open Foundation and Fine-Tuned Chat Models". *arXiv preprint arXiv:2307.09288* (2023). URL: https://arxiv.org/abs/2307.09288

[9] H. Li et al. "Extracting Formal Specifications from Documents Using LLMs for Automated Testing". *arXiv preprint arXiv:2504.01294* (2025). URL: https://arxiv.org/abs/2504.01294

[10] M. T. Khan et al. "From Drawings to Decisions: A Hybrid Vision-Language Framework for Parsing 2D Engineering Drawings into Structured Manufacturing Knowledge". *arXiv preprint arXiv:2506.17374* (2025). URL: https://arxiv.org/abs/2506.17374

[11] A. García, J. Paredes, and M. Yang. "TechMB: Exploring the Potential of Vision Language Models for Interpreting Technical Drawings". *Proc. of ICED 2023 (Design Society)*. 2023. DOI: 10.35199/dfx2025.19

[12] W. Zhang et al. "MBSE Co-Pilot: A Research Roadmap". *Systems Engineering* (2025). URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/sys.70011

[13] R. Longshore, R. Bell, and R. Madachy. *Leveraging Generative AI to Modify and Query MBSE Models*. Tech. rep. Acquisition Research Program, 2024. URL: https://dair.nps.edu/handle/123456789/5113

[14] A. Patel, Y. Maheshwaran, and P. Santhya. "Easing Adoption of Model Based System Engineering with Application of Generative AI". *2024 IEEE Space, Aerospace and Defence Conference (SPACE)*. IEEE. 2024. DOI: 10.1109/SPACE63117.2024.10667868

➞ References (Continued)

[15] R. M. García Alarcia et al. "Bringing Systems Engineering Models to Large Language Models: An Integration of OPM with an LLM for Design Assistants". *Proc. of the 12th Intl. Conf. on Model-Driven Engineering and Software Development (MODELSWARD)*. 2024. DOI: 10.5220/0012621900003645

[16] F. Schummer, M. Langer, et al. "An Approach for System Analysis with MBSE and Graph Databases". *Proc. of MBSE Cybernetics Workshop*. arXiv:2201.06363. 2022. URL: https://arxiv.org/abs/2201.06363

[17] Intercax LLC. *Syndeia: The Digital Thread Platform for Integrated MBSE*. Software documentation. Accessed 2025. 2021. URL: https://intercax.com/products/syndeia/

[18] NASA JPL. *OpenMBEE: An Open Model-Based Engineering Environment*. Project Documentation. 2018. URL: https://openmbee.org

[19] T. D. Hedberg Jr, M. Bajaj, and J. A. Camelio. "Using Graphs to Link Data Across the Product Lifecycle for Enabling Smart Manufacturing Digital Threads". *Journal of Computing and Information Science in Engineering* 20.1 (2020). DOI: 10.1115/1.4044921

[20] OASIS. *OSLC Core Version 3.0*. OASIS Project Specification 02. 2021. URL: https://docs.oasis-open-projects.org/oslc-op/core/v3.0/ps02/oslc-core.html

Thank you!

Questions &
discussion welcome

GenAI Workbench

AI-Assisted Analysis and Synthesis of Engineering Systems from Multimodal Engineering Data

H. Sinan Bank & Daniel R. Herber

Department of Systems Engineering, Colorado State University

sinan.bank@colostate.edu

- arxiv.org/abs/2603.00251 *(GenAI Workbench preprint)*
- arxiv.org/abs/2602.16715 *(Architecture Synthesis preprint)*
- arxiv.org/abs/2603.20151 *(Design-OS preprint)*
- doi.org/10.1145/3685650.3685665 *(CatalogBank, DocEng '24)*