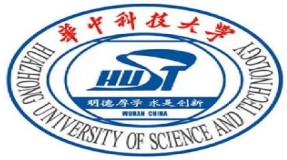


Rethinking the Role of Department of Defense Architecture Framework in System-of-Systems Architecture Design

Authors: Zhemei Fang, Yuxuan Liu, Jianbo Wang

Zhemei Fang, Associate Professor

zmfang2018@hust.edu.cn

School of Artificial Intelligence and Automation
Huazhong University of Science and Technology

Author's Bio Info

Zhemei Fang is an **Associate Professor** in the School of Artificial Intelligence and Automation in **Huazhong University of Science and Technology** in Wuhan, China. She received her Ph.D. degree from the School of Aeronautics and Astronautics Engineering at **Purdue University** in 2017.

Her primary research interests center on developing theories and methods for **System-of-Systems (SoS) architecture design and evolution**. She is currently exploring artificial intelligence approaches, like large language models, knowledge graphs, and reinforcement learning, for supporting **AI-driven, iterative SoS architecture design** (encompassing architecture representation, decision-making, and evaluation).

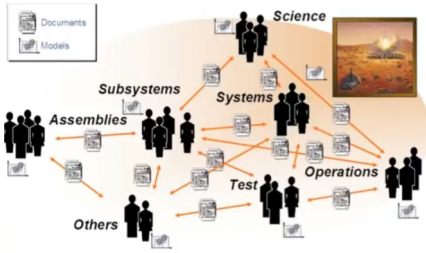
Outline

Motivation & Research Objectives

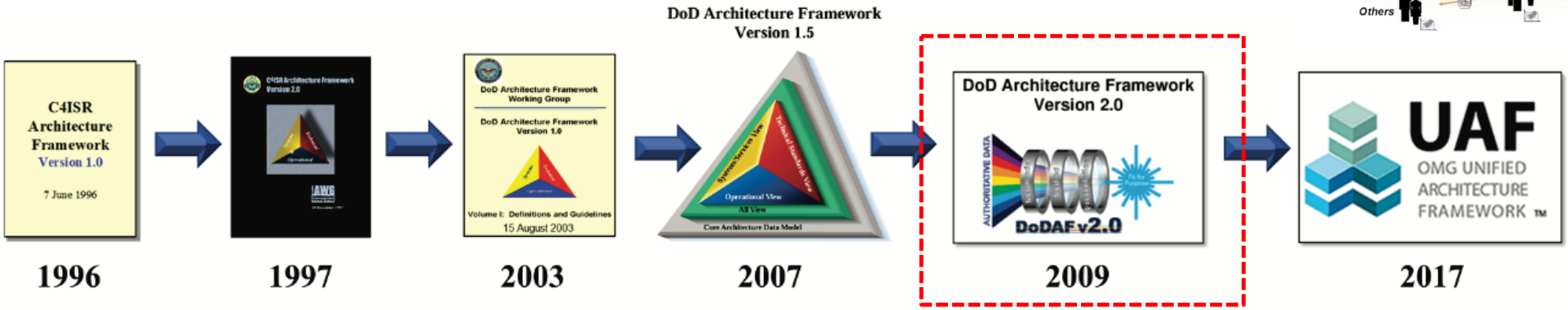
❑ Architecture & Architecture Description Framework (ADF)

Architecture: Fundamental concepts or properties of an entity in its environment and governing principles for the realization and evolution of this entity and its related life cycle processes

ISO/IEC/IEEE 42010:2022 standard (Software, systems and enterprise - architecture description)

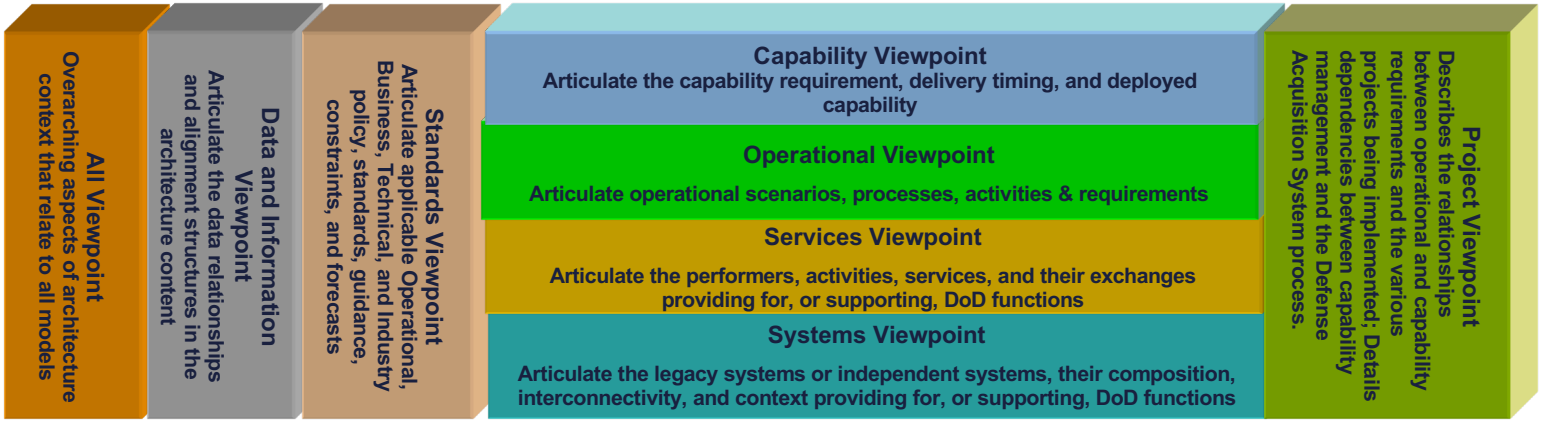


ADF: Formalize the conventions and common practices of architecture description - a tangible work product that communicates the otherwise intangible and abstract concept of architecture

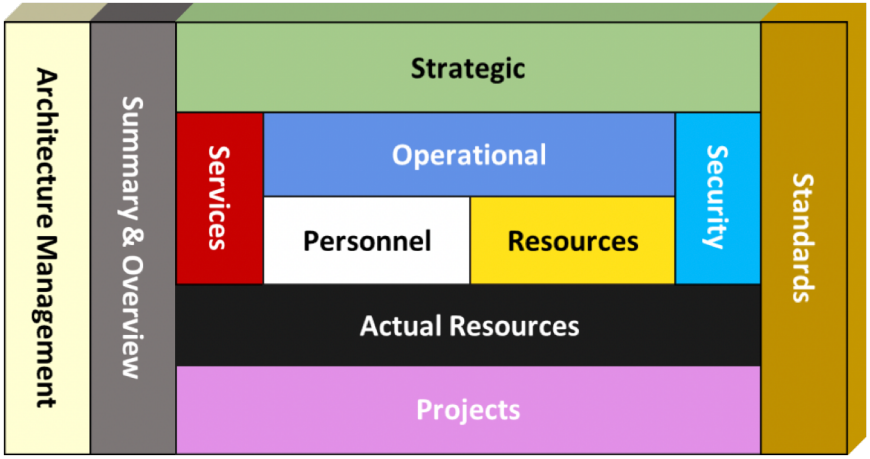


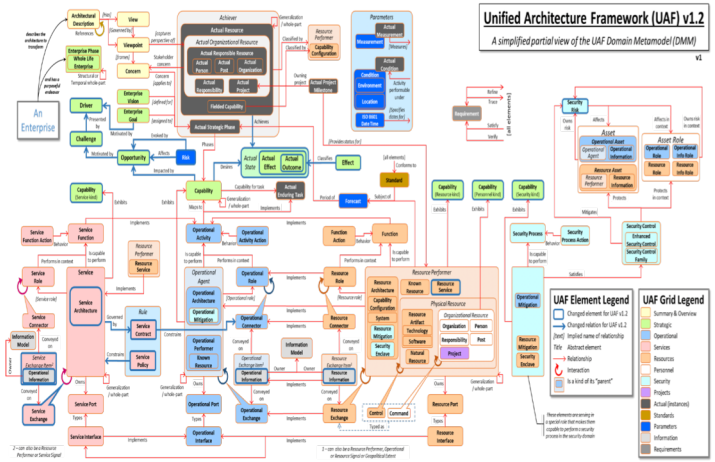
Motivation & Research Objectives

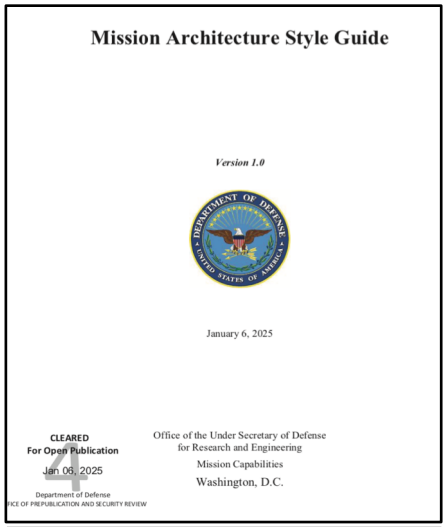
❑ Department of Defense Architecture Framework (DoDAF)



UAF





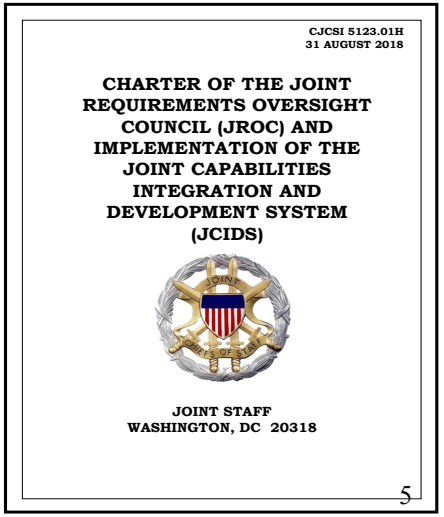

Mission Architecture Style Guide

Version 1.0

January 6, 2025

CLEARED
For Open Publication
Jan 06, 2025
Department of Defense
OFFICE OF PREPUBLICATION AND SECURITY REVIEW

Office of the Under Secretary of Defense
for Research and Engineering
Mission Capabilities
Washington, D.C.


CJCSI 5123.01H
31 AUGUST 2018

CHARTER OF THE JOINT REQUIREMENTS OVERSIGHT COUNCIL (JROC) AND IMPLEMENTATION OF THE JOINT CAPABILITIES INTEGRATION AND DEVELOPMENT SYSTEM (JCIDS)

JOINT STAFF
WASHINGTON, DC 20318

Motivation & Research Objectives

❑ Complaints from Practitioners

https://federalnewsnetwork.com/reporters-notebook-jason-miller/2022/09/you-dont-speak-dodaf-the-navy-feels-your-pain-with-its-new-plain-language-design-concept/

We work in this obscure language that nobody speaks, called the Defense Department Architecture Framework (DoDAF). It's great language, there's nothing wrong with it. We have **extra people** come on board to a project or program and those people go and **bother** the people who are **the engineers, the testers, the developers**, the people turning wrenches, and they say, 'Hey, I need information. So I can **fill out this DoDAF** view because we got to go back to the Joint Requirements Oversight Council (JROC) and **get it approved**, or we got to go back to the program officer to the **milestone decision authority**, the chief engineer has to approve these things in order for us to field this thing we're working on.

I have to **steal some of your time to go create this product**, so that it can be **checked by** guess who? **Another architect** whose whole job is to check the homework of the first architect. They will argue, believe me; they will go back and forth with one another because if the second guy doesn't reject the first guy's work, once or twice, he's not doing his job. In the meantime, what's actually happening? Everybody else is out there, like building stuff, actually testing things and delivering things, hopefully, to the end user who actually needs them. And the architecture really served as a gating function. It really **served as a thing that would slow you down**, and that would **prevent you from eventually delivering that capability**. So I hate architects, because that's what we do.

Don Yeske (architect for the Department of Navy's chief information officer's office)

Motivation & Research Objectives

❑ Research Objectives

DoDAF/UAF

MBSE

A persistent tension between the theoretical promise of ADFs and their real-world implementation barriers

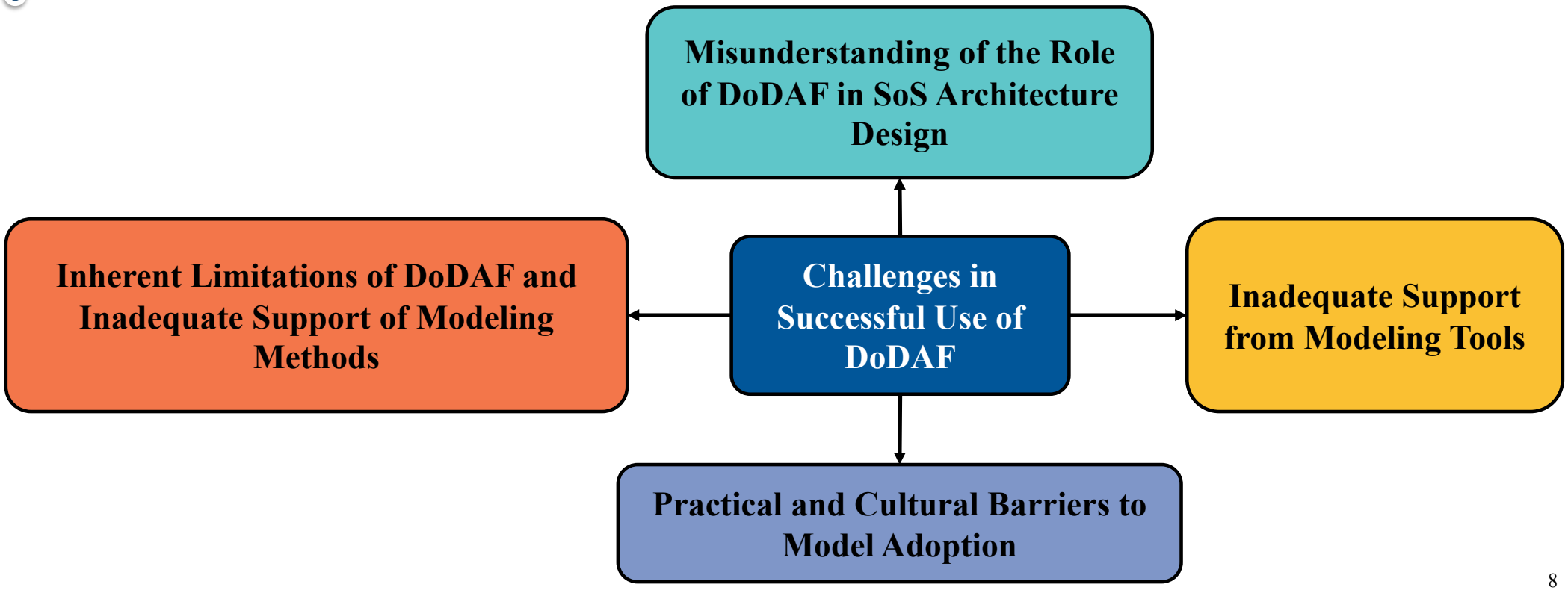
(issues are commonly acknowledged in informal exchanges)

☀ Research objective:

Systematically investigate the practical barriers that hinder the effective application of ADFs in complex SoS contexts and propose potential strategies for enhancing its practical utility

Challenges of DoDAF in SoS Architecture Design

Architecture Design = *f*(Architecture Representation, Architecture Decision-making, Architecture Evaluation, Human-in-the-Loop)



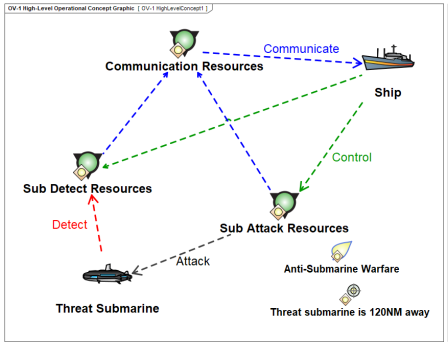
Challenges of DoDAF in SoS Architecture Design

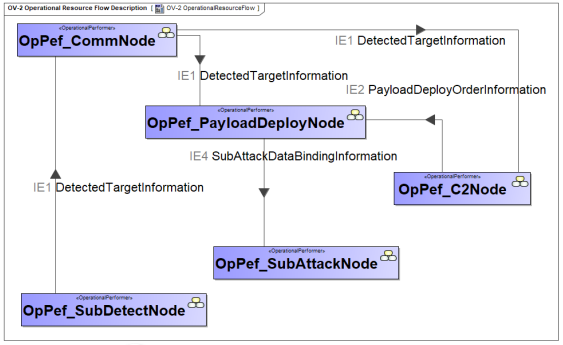
❑ Misunderstanding of the Role of DoDAF in SoS Architecture Design

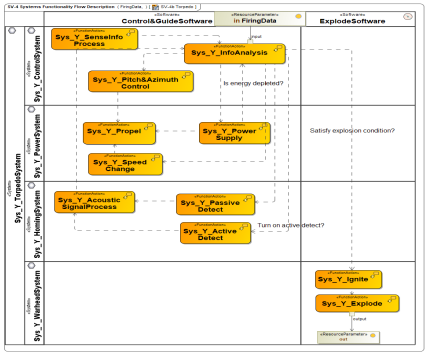
New to DoDAF → **Some experience with DoDAF**

DoDAF Modeling = SoS Architecture Design

DoDAF Modeling = Microsoft Visio Modeling







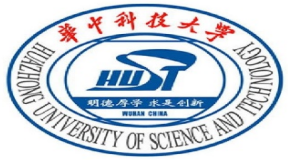
.....

DoDAF is Only for Documentation, Not for Analysis

DoDAF Models Are Static and Do Not Evolve

Products for passing milestone review only

Challenges of DoDAF in SoS Architecture Design



❑ Inadequate Support from Modeling Tools




.....

- **Steep Learning Curves for New Users**
  - Language (e.g., SysML); tool itself
- **Insufficient Support for Model Reuse**
  - Model modularity; model repository
- **Insufficient Support for Iterative and Agile Modeling**
  - Model modularity; version control
- **Limited Support for Modeling Dynamic Behavior**
  - Behaviors in architecture level
- **Difficulty in Managing Large-Scale SoS Complexity**
  - UI response; dependency traceability
- **Poor Interoperability with Other Tools**
  - Tool relationships; techniques
- **Lack of Intelligent Support**
  - AI methods (e.g., LLM) may help mitigate some of the issues above

Challenges of DoDAF in SoS Architecture Design

❑ Inherent Limitations of DoDAF and Inadequate Support of Modeling Methods

- **Over-Simplification of SoS Complexity**
  - ADFs may oversimplify the complexity of SoS
- **Inadequate Modeling and Verification Methods**
  - Inherent subjectivity of the modeling process
- **Lack of Methodological Guidance for SoS Analysis**
  - Users have different interpretations
- **Underestimation of Evolutionary Nature of SoS**
  - Requires iterative validation and continuous adaptation
- **Unclear Boundary Between Representation and Decision-making**
  - Ambiguity creates confusion

Challenges of DoDAF in SoS Architecture Design

❑ Practical and Cultural Barriers to Model Adoption

Models Focus on Compliance, Not Practical Use

Deliverables are created to pass reviews but rarely maintained or reused afterward

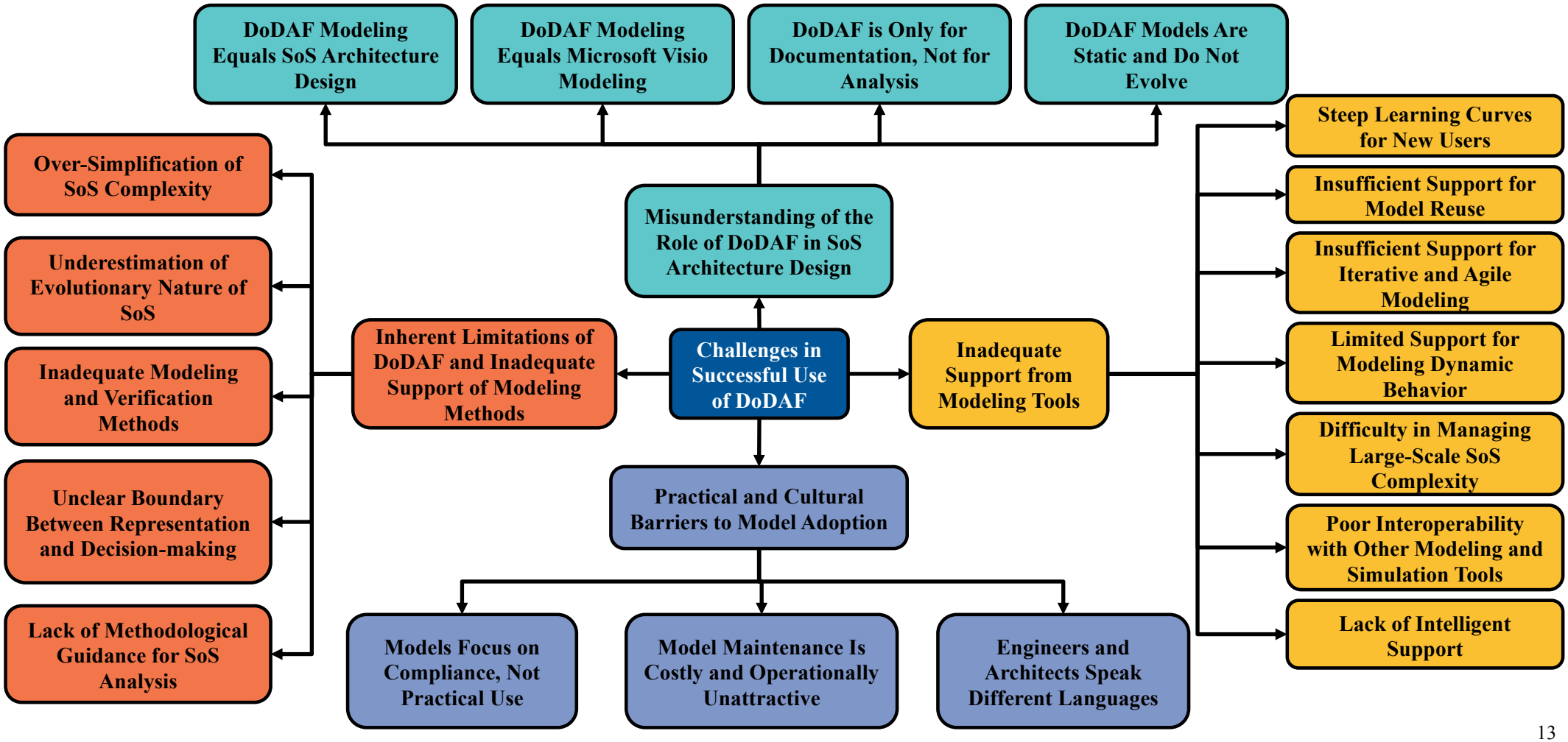
Model Maintenance is Costly and Operationally Unattractive

Teams may prefer to directly update prototypes or source code, bypassing architecture layer entirely

Engineers and Architects Speak Different Languages

This disconnect hampers collaboration and leaves the architecture models isolated from actual system implementation

Challenges of DoDAF in SoS Architecture Design



Opportunities for Improvement

❑ UAF's Capability to Address the Issues

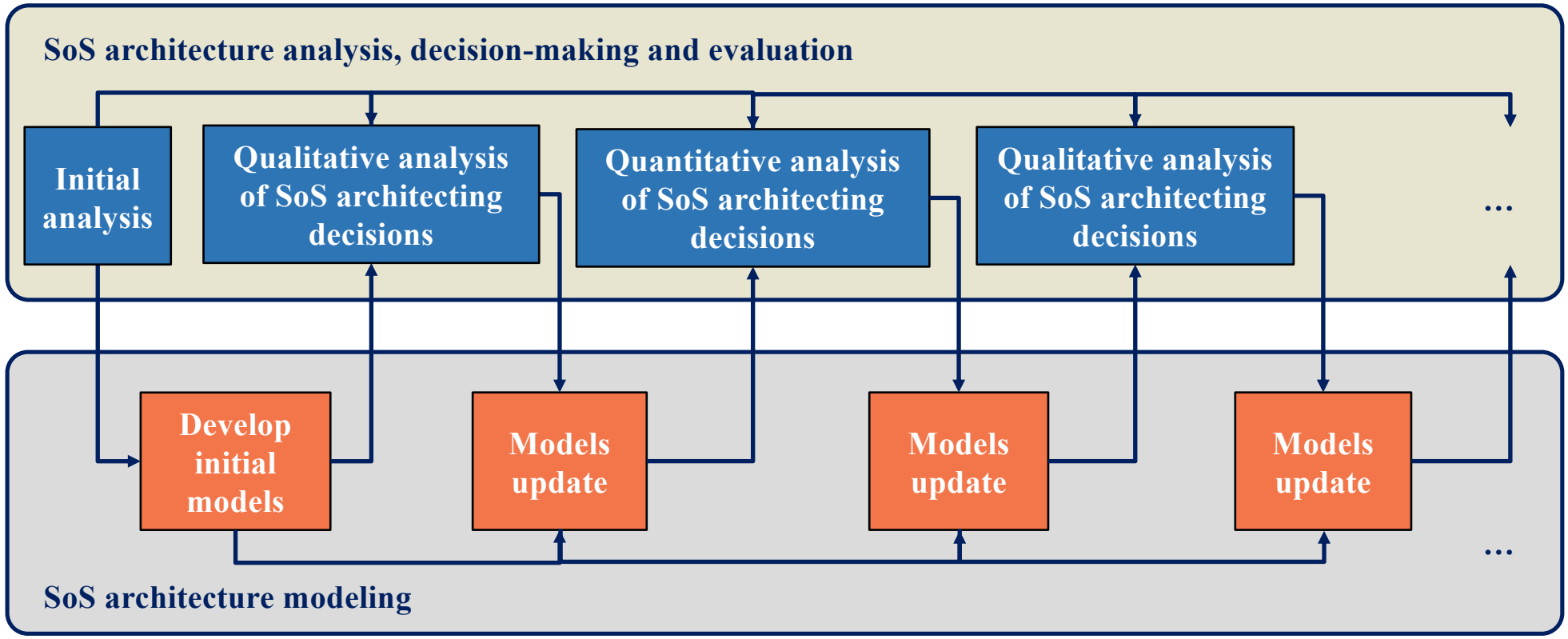
- ☀ UAF establishes an integrated meta-model that enhances the semantic consistency and structural rigor of architecture representations
- ☀ UAF aligns more closely with MBSE principles and SysML
- ☀ UAF still faces practical adoption challenges—particularly in terms of modeling methodology, tool maturity, and organizational constraints

❑ SysML 2.0's Capability to Address the Issues

- ☀ Integrates graphical and textual modeling approaches, bridging the language gap between system architects and domain engineers
- ☀ Defines APIs that enable seamless integration with simulation engines and verification tools
- ☀ Supporting a formal textual syntax makes it naturally compatible with LLMs
- ☀ Most of these anticipated benefits have yet to be validated in practice, and realizing them would require significant retooling of existing tools and workflows

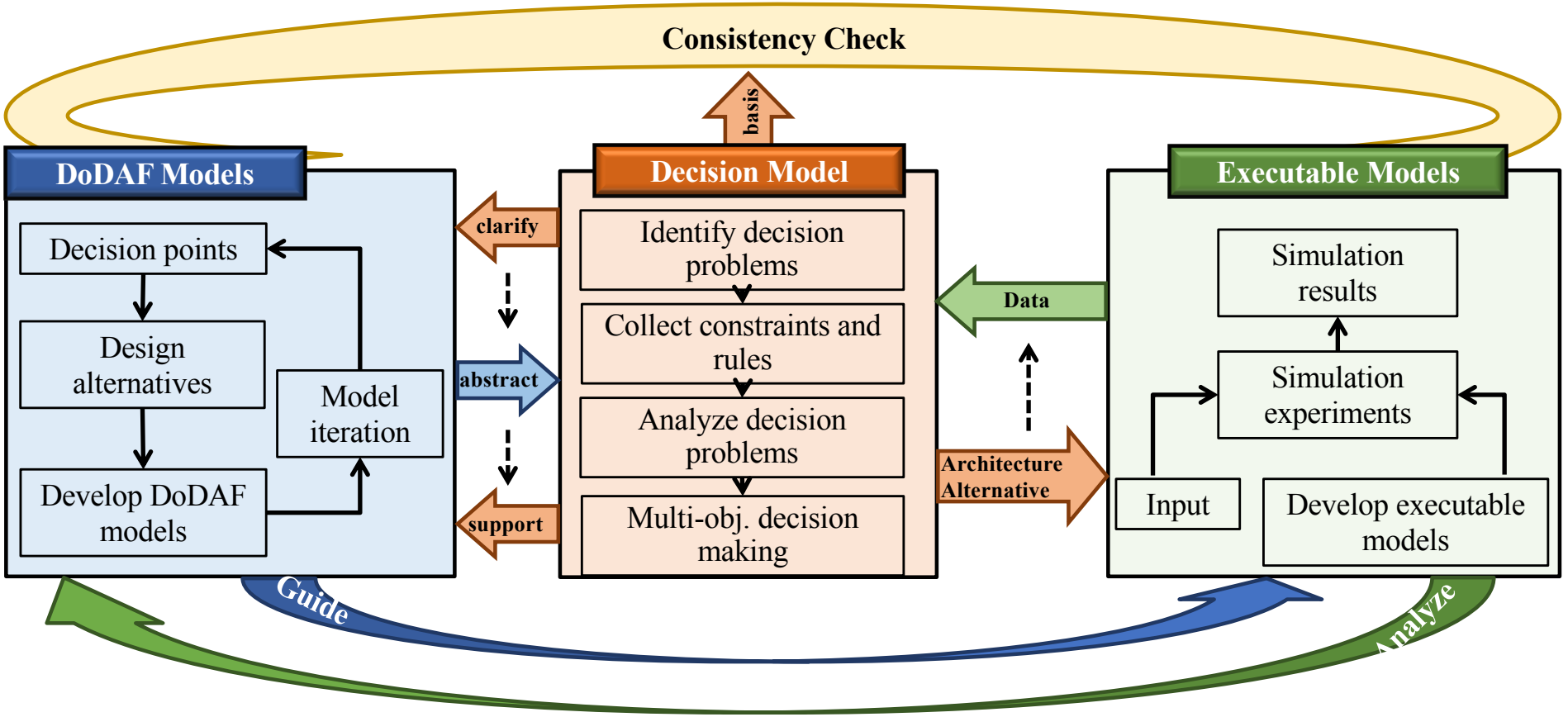
Opportunities for Improvement

- **Architecture Description Models Reflect Architecting Process more than Architecture Outcomes**



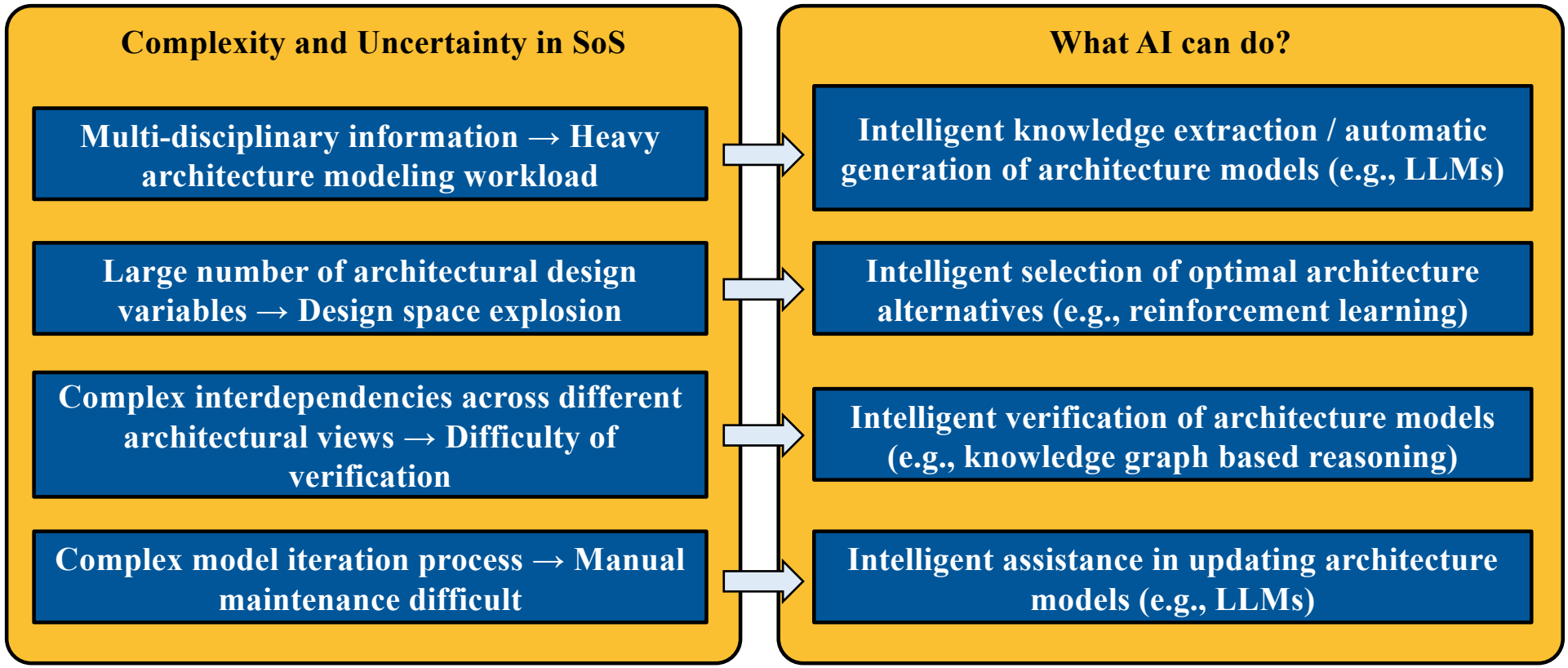
Opportunities for Improvement

- **Architecture Description Models Reflect Architecting Process more than Architecture Outcomes**



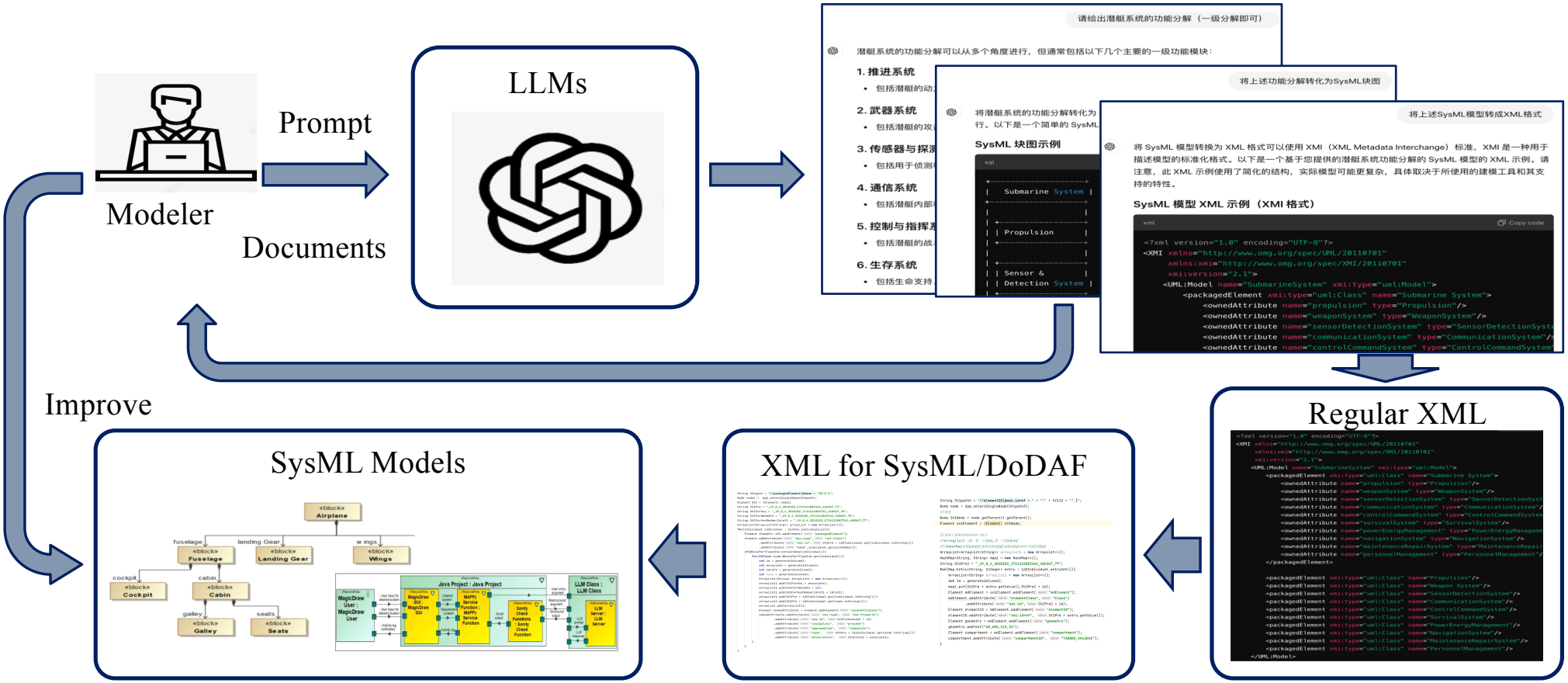
Opportunities for Improvement

❑ AI-Assisted Architecture Modeling and Design



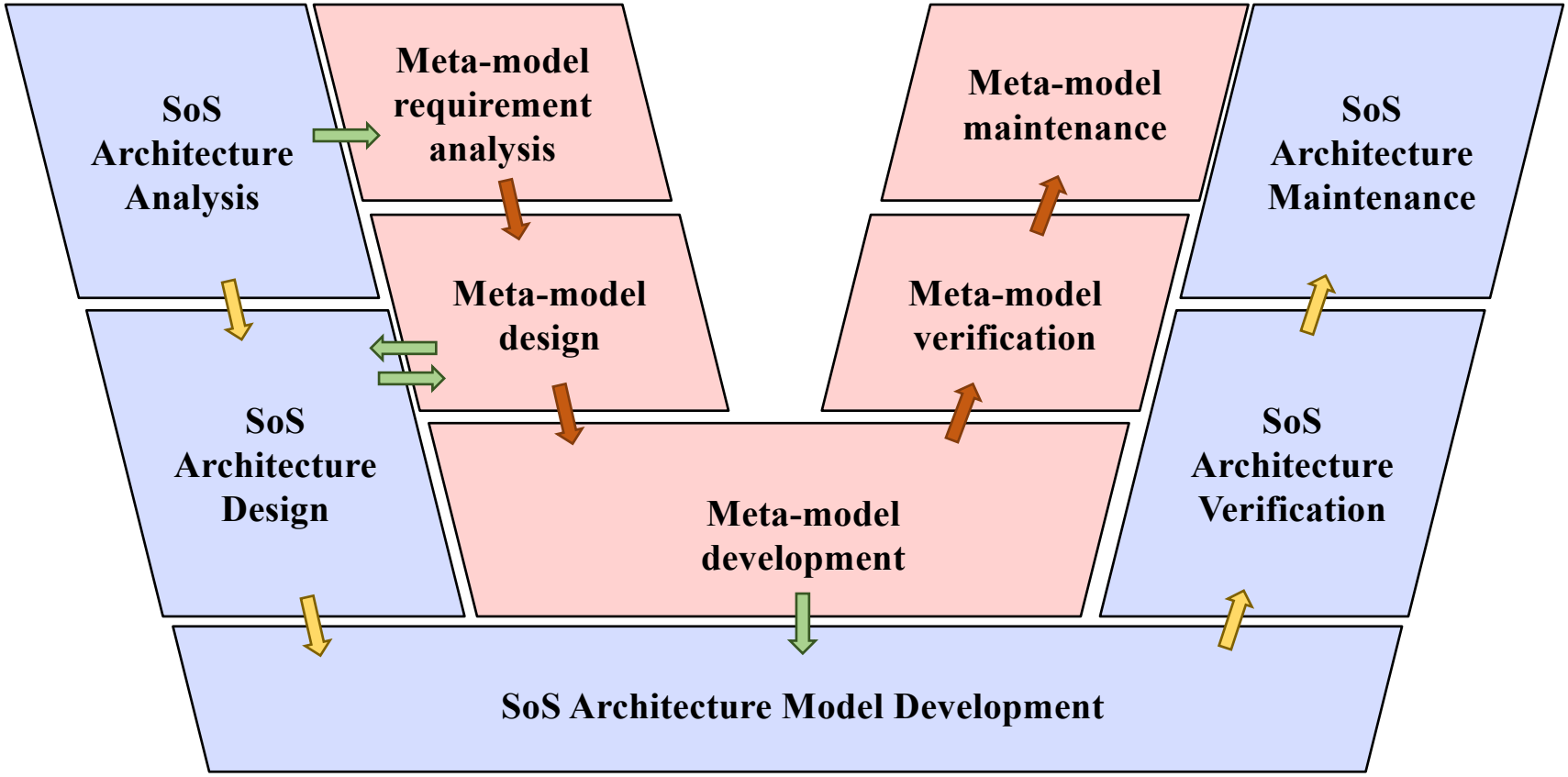
Opportunities for Improvement

❑ AI-Assisted Architecture Modeling and Design



Opportunities for Improvement

❑ Customized Metamodel Development and Underlying Consistency Assurance



Conclusion & Future Work

❑ Conclusion

- ☀ **Analyzed the key challenges facing DoDAF in the SoS architecture design**
  - **Misconceptions**
  - **Inadequate tool support**
  - **Method limitations**
  - **Organizational barriers**
- ☀ **Proposed some improvement suggestions**

❑ Future Work

- ☀ **Developing a lightweight iterative modeling plugin to integrate architectural models with decision-support tools**
- ☀ **Creating a specialized prompt engineering framework for LLMs tailored to SoS architecture tasks**
- ☀ **Transforming DoDAF from a documentation exercise into an evolving intelligent decision-support process**

References

1. ISO/IEC/IEEE, “ISO/IEC/IEEE 42010:2022(E) Systems, Software and Enterprise - Architecture Description,” 2022.
2. Joint Staff, “Charter of the Joint Requirements Oversight Council and Implementation of the Joint Capabilities Integration and Development System,” 2021.
3. G. Bangjun, C. Yunfeng, W. Xinyao, and Y. Wangwang, “Modeling the Anti-UAV Swarm System Architecture Based on DoDAF,” in 2023 9th Int. Conf. Big Data Inf. Anal., Haikou, China, Dec. 2023, pp. 409–412.
4. A. Aghamohammadpour, E. Mahdipour, and I. Attarzadeh, “Architecting Threat Hunting System Based on the DoDAF Framework,” J. Supercomput., vol. 79, pp. 4215–4242, 2023.
5. M. Vinarcik, “The Problem with DoDAF Models,” in 22nd Annu. Syst. Mission Eng. Conf., Tampa, FL, 2019.
6. S. W. Mitchell, “Transitioning the SWFTS Program Combat System Product Family from Traditional Document-Centric to Model-based Systems Engineering,” Syst. Eng., vol.17, no.3, pp.313-329, 2014.
7. M. Zaman, “Dynamic Resilient Enterprise Architecture Model (DREAM) Adoption in Defense Digital Architecture Management to Mitigate Disconnected Systems and Processes,” Ph.D. dissertation, George Mason Univ., 2025.
8. [Reddit], “Change My View: Model-Based Systems Engineering in 2024 is Still More Hype Than Value,” r/systems_engineering, 2024. [Online]. Available: https://www.reddit.com/r/systems_engineering/comments/1bpavpi/change_my_view_model_based_systems_engineering_in/?rdt=57765 [Accessed: Sept, 21st, 2025].
9. L. W. Wagenhals, I. Shin, D. Kim, and A. H. Levis, “C4ISR Architectures II. A Structured Analysis Approach for Architecture Design,” Syst. Eng., vol. 3, no. 4, pp. 248–287, 2000.
10. C. Piaszczyk, “Model Based Systems Engineering With Department of Defense Architectural Framework,” Syst. Eng., vol. 14, no. 3, pp. 305-326, 2011.
11. R. Wang and C. H. Dagli, “Executable System Architecting Using Systems Modeling Language in Conjunction With Colored Petri Nets in a Model-Driven Systems Development Process,” Syst. Eng., vol. 14, no. 4, pp. 383-409, 2011.
12. M. Amissah and H. A. H. Handley, “A Process for DoDAF-Based Systems Architecting,” in 2016 Annu. IEEE Syst. Conf., Orlando, FL, USA, 2016, pp.1-7.
13. J. N. Martin and D. P. O'Neil, “Enterprise Architecture Guide for the Unified Architecture Framework (UAF) ,” in Proc. INCOSE Int. Symp., vol. 31, pp. 242-263, 2021.
14. H. M. Hause, “Rebuilding the Tower of Babel: The Case for a Unified Architecture Framework,” in Proc. INCOSE Int. Symp., vol. 23, no. 1, pp. 1460–1474, 2013.

References

15. J. N. Martin, “Aspect-Oriented Architecting Using Architecture Frameworks,” in Proc. INCOSE Int. Symp., vol. 31, no. 1, pp. 210–226, 2021.
16. J. N. Martin, “Extending UAF for Model-Based Capability Planning and Enterprise Portfolio Management,” in Proc. INCOSE Int. Symp., vol. 32, no. 1, pp. 15-35, 2022.
17. K. Carroll, A. Lyle, R. Lewark, C. Medina, and A. Morkevičius, “SoS - Global Solutions to Global Problems Using UAF,” in Proc. INCOSE Int. Symp., vol. 34, pp. 2400-2412, 2024.
18. M. Hause and L. O. Kihlström, “Modeling Enterprise Software with UAF,” in Proc. INCOSE Int. Symp., vol. 34, pp. 2452-2475, 2024.
19. J. N. Martin and K. E. Alvarez, “Using the Unified Architecture Framework in Support of Mission Engineering Activities,” in Proc. INCOSE Int. Symp., vol. 33, pp. 1156–1172, 2023.
20. M. Gagliardi, M. C. Hause, J. N. Martin, and M. A. Phillips, “Darth Vader's Secret Weapon: Implementing Mission Engineering With UAF,” in Proc. INCOSE Int. Symp., vol. 34, pp. 1719-1747, 2024.
21. Office of the Under Secretary of Defense for Research and Engineering, Department of Defense Mission Architecture Style Guide. Washington, DC, USA: U.S. Department of Defense, 2025.
22. Z. Fang, X. Zhao, and F. Li, “Architecture Design Space Generation via Decision Pattern-Guided Department of Defense Architecture Framework Modeling,” Systems, vol. 12, no. 2, p. 336, 2024.
23. A. E. Trujillo and A. M. Madni, “MBSE Methods for Inheritance and Design Reuse,” in Handbook of Model-Based Systems Engineering, A. M. Madni, N. Augustine, and M. Sievers, Eds. Cham, Switzerland: Springer, 2023.
24. A. Morkevicius and G. Krisciuniene, “Towards UAF Implementation in SysML V2,” in Proc. INCOSE Int. Symp., vol. 34, pp. 2452-2475, 2024.
25. M. W. Maier, “Adapting the Hatley-Pirbhai Method for the Era of SysML and Digital Engineering,” 2022 IEEE Aerosp. Conf., Big Sky, MT, USA, 2022, pp. 1-12.
26. J. N. Martin and D. Brookshier, “Linking UAF and SysML Models: Achieving Alignment Between Enterprise and System Architectures,” in Proc. INCOSE Int. Symp., vol. 33, pp. 1132–1155, 2023.
27. Object Management Group, “OMG Systems Modeling Language (SysML) Version 2.0 Beta 2 (Revision 2024-02), Part 2: SysML v1 to SysML v2 Transformation,” OMG Doc. No. ptc/2024-02-01, Feb. 2024.
28. J. K. DeHart, “Leveraging Large Language Models for Direct Interaction With SysML v2,” in Proc. INCOSE Int. Symp., vol. 34, pp. 2168–2185, 2024.

THNAK YOU!