2023 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
DIGITAL ENGINEERING/SYSTEMS ENGINEERING TECHNICAL SESSION
AUGUST 15-17, 2023 - NOVI, MICHIGAN

VERIFICATION OF AN ARCHITECTURE IN A SYSTEM MODEL USING DOMAIN-SPECIFIC OPERATIONAL SCENARIOS AND CONTEXTS

Macam Dattathreya, PhD[1]

[1]US Army DEVCOM/PEO Ground Combat Systems, Warren, MI

ABSTRACT

A proposed approach for verifying a Modular Open Systems Approach (MOSA)-enabled system architecture in Systems Modeling Language (SysML) based system models of military ground vehicle domains. Using this approach, Model Based Systems Engineering (MBSE) practitioners can verify that a given MOSA-enabled architecture in the system model is compliant to pre-defined MOSA-adopted reference architecture models. The approach utilizes military ground vehicle domain-specific operational scenarios and their associated domain context-specific architecture verification rules, architecture verification metrics and compliance-scope. MBSE professionals can implement the proposed approach using automated or semi-automated solutions.

Citation: Macam Dattathreya, "*Verification of an architecture in a* system *model using domain-specific operational scenarios and contexts*," In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 15-17, 2023.

1. INTRODUCTION

Model Based Systems Engineering (MBSE) practitioners develop models of Modular Open Systems Approach (MOSA)-enabled system architectures using modeling languages such as a Systems Modeling Language (SysML). System architecture models of military ground vehicles include design details in SysML-specific syntaxes and semantics. The MBSE practitioners represent a system model's compliance to a given military ground vehicle's performance requirements and MOSA-enabled reference architectures through satisfied relationships, simulation and validation results, or other verification methods. However, it's a daunting task for the verifiers to check model compliance with precision to a given MOSA-enabled reference architecture in military ground vehicle's system design contexts and operational scenarios.

The verification of the architecture in a model becomes very complex when the model represents the MOSA principles at conceptual, logical, and physical levels. Additionally, if the model represents its compliance to a given MOSA-adopted reference architecture, verification of that system architecture in that model becomes a very complicated undertaking. Therefore, an

DISTRIBUTION A. Approved for public release; distribution unlimited. OPSEC #: 7732

efficient verification approach should be developed prior to beginning the MOSA-enabled MBSE effort.

Several formal or informal model verification approaches [1-6] exist. However, they all lack a system domain-specificity, architecture-specific verification rule sets and architecture compliance checks according to given MOSA-adopted reference architectures.

This paper describes an approach for verifying a MOSA-enabled system architecture in SysML based system models of military ground vehicle domains. Using this approach, MBSE practitioners can verify that a given MOSA-enabled architecture in the system model is compliant to pre-defined MOSA-adopted reference architecture models. The approach utilizes military ground vehicle domain-specific operational scenarios and their associated domain context-specific architecture verification rules, architecture verification metrics and compliance-scope. MBSE professionals can implement the proposed approach using automated or semi-automated solutions.

Section 2 of the document describes an overview of the proposed approach; Section 3 discusses the approach in detail, Section 4 concludes the paper, and Section 5 lists the related literature references.

2. OVERVIEW OF THE PROPOSED VERIFICATON APPROACH

Prior to starting an MBSE effort, the practitioners should develop approaches for verifying architectures of the system being modeled. The verification considerations will enable the practitioners to efficiently verify the architectures continuously when the model is being built. The verification mindset will enable the modelers to adopt to appropriate modeling language-specific structural and behavioral constructs, and styles from the beginning of the effort. This mindset will influence practitioners to conduct efficient verification of the architectures in models repeatedly and successfully.

Figure 1 depicts the proposed high-level approach for architecture verification in system models. The key inputs for a successful verification approach include system performance requirements, system specific operational scenarios, domain contexts and appropriate MOSA-adopted reference architecture models. These inputs will enable practitioners to develop domain-specific verification scope, contexts, rules, architecture verification metrics, and

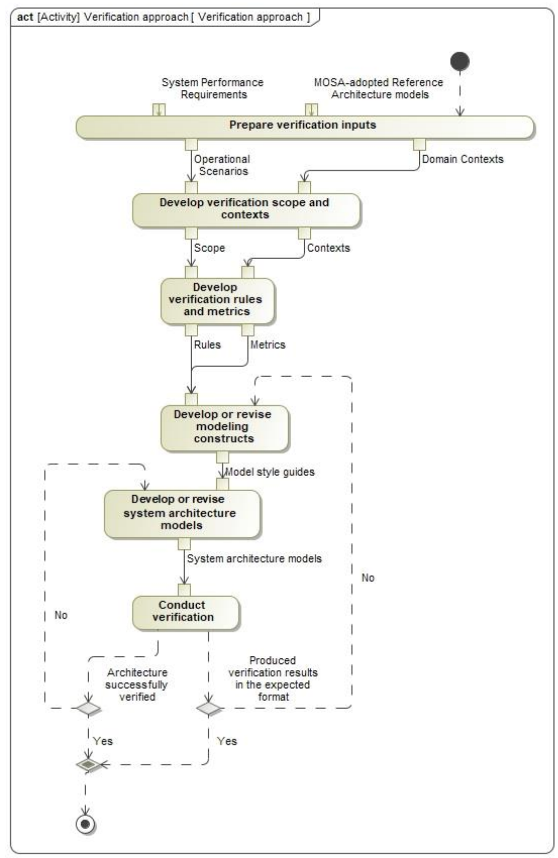

Figure 1: A high-level approach for an architecture verification in a svstem model.

modeling style guides. More importantly, the modelers can develop system architecture models using verification as one of the modeling contexts.

The outputs from the verifications can be used to improve modeling constructs and the model itself. This approach will influence continuous and rapid model improvements with verification in the loop always.

Section 3 describes the approach in detail.

3. THE VERIFICATION APPROACH DETAILS

Many existing verification approaches tackle the verification problem as a tactical solution and may not directly tie into the system domain contexts, operational scenarios. The proposed approach tries to fill that gap.

For brevity, the rest of the document uses the term *System* in place of a military ground vehicle. Sections 3.1 through 3.4 describe all the steps of the proposed verification approach.

3.1. Prepare verification inputs

A System's performance requirements and MOSA-adopted reference architecture models provide enough inputs to develop *System* domain-specific contexts and *System's* operational scenarios.

Practitioners can use the applicable SysML constructs for creating these contexts and scenarios. Each context should have clear definitions for testability during a verification process.

A maintainer replacing a sensor of a given weapon can be an example of an operational scenario and its associated domain-specific contexts can be modularity, modular boundary, and interfaces. Modularity is one of the principles of MOSA and is used for managing complexity of a system.

A system's parts communicate with each other only through the interfaces within a standardized architecture. Without modularity, the complex systematic interconnections cannot be eliminated [7].

A modular boundary of a component is at

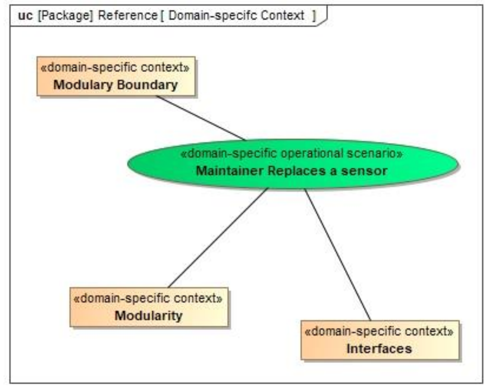

Figure 2: An example of a domain context.

its interfaces. An interface is an interaction portal between two components. Each interface will have its detailed definitions as a contract between communicating components of a system.

Figure 2 depicts an example domain-specific context and Figure 3 displays an example of a definition of a modular boundary.

3.2. Develop verification scope and contexts

A typical MOSA-enabled reference architecture model provides enough guidance for a *System's* design using the Army's MOSA implementation guide [8]. The model of such architectures articulates MOSA using SysML modeling constructs such as general blocks, interface blocks, constraints, structural and behavioral diagrams including state machines. Using those as guides, for each of the domain-specific contexts, the practitioners can develop appropriate architecture compliance verification scope and contexts.

As an example, for the interfaces as one of the domain-specific contexts, an appropriate verification scope can be to check all the interfaces defined in the structural context of the model with respect to a given standard interface's definitions, and provider and consumers of that interface.

Another scope can be to check the interactions surrounding a given interface in a model's behavior constructs such as activity or sequence diagrams.

Similarly, the practitioners can develop sufficient scope to ensure full coverage of the standard being used in the model.

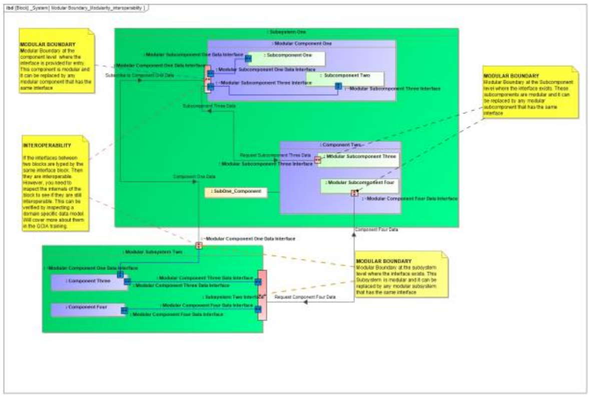

Figure 3: A modular boundary example.

3.3. Develop verification rules and metrics

Compliance verification scope and the domain-specific contexts provide inputs to develop appropriate domain context-specific architecture verification rules and architecture verification metrics. Figure 4 shows an example of a verification rule and Table 1 shows an example list of interface related verification metrics .

For example, an architecture-compliance verification rule can be to verify each interface's interaction definitions in the MOSA-enabled system architecture model with respect to the MOSA-adopted reference architecture model defined interfaces.

Table 1. Interface related verification metrics examples.

S/N	Name
1	# MOSA-Compliant interfaces
2	# Non-MOSA-compliant interfaces
3	#proprietary interfaces/#open interfaces
4	# Of components with no interfaces
5	% Of licensed interfaces

The practitioners can develop verification rules using the SysML constraint block constructs and utilize tool-specific validation engines or custom software scripts to execute the verification of the rules.

Use the results of the verification to revise modeling constructs and the system architecture being built. A SysML rule of construction can be used as a verification rule construct too. Figure 4 shows an example of a rule of construction for an interface's interaction type.

3.4. Develop or revise modeling constructs and architecture models, and conduct verification

Based on the verification rules, develop SysML style guides and model constructs. Based on the results from the verification execution, revise modeling constructs to ensure in-line with the verification approach.

As described earlier, use any tool-specific validation engines or custom software scripts to execute the verification of the rules. Based on the failed verifications, adjust the system architecture models to enable a continuous verification and improvement of the model being built for compliance with MOSA-adopted reference architecture model.

Figure 4: A SysML construct for a verification rule.

4. CONCLUSION

A streamlined compliance verification of system architecture models is an important method for successful MOSA-enabled MBSE efforts. The use of military ground vehicle domain-specific operational scenarios and their associated context-specific architecture verification rules with architecture verification metrics enhances the accuracy of MOSA verification approaches.

Continuous verification of the models using the proposed approach will enable the practitioners to adjust the system architecture models quickly and ensure compliance with the MOSA-adopted reference architecture model.

5. REFERENCES

[1]Burkhart, Rm et al. Simulation-Based Design Using SysML—Part 1: A Parametrics Primer. INCOSE International Symposium, 2007.

[2]Morkevicius, Aurelijus, and Nerijus Jankevicius. "An approach: SysML-based automated requirements verification." 2015 IEEE International Symposium on Systems Engineering (ISSE). IEEE, 2015.

[3]Myron Hecht, Jaron Chen, "Verification and Validation of SysML Models", INCOSE International Symposium, vol.31, no.1, pp.599, 2021.

[4]Debbabi, Mourad, et al. Verification and validation in systems engineering: assessing UML/SysML design models. Springer Science & Business Media, 2010.

[5] Apvrille, Ludovic, et al. "SysML models verification relying on dependency graphs.",2022.

[6] Ouchani, Samir, Otmane Ait Mohamed, and Mourad Debbabi. "A formal verification framework for SysML activity diagrams." Expert Systems with Applications 41.6,2014.

[7] "Modularity in Technology and Organization," Journal of Economic Behavior & Organization, Langlois, Richard, 2002

[8] Office of the chief systems engineer, ASA (ALT)/OCSE, "Modular Open Systems Approach (MOSA) Implementation Guide," January 2022.