DIGITAL ENGINEERING
/ SYSTEMS ENGINEERING

Synchronizing MBSE Models and Software Development in Robotic Autonomous Systems

Calvin Cheung, Andrew Pfeil, Shannon Griffith, Mark Petrotta, Mitchell Brooks, Michael Moore, Doyle Dennis, Reagan Grunwald

DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited. OPSEC#8846.

Elevator Pitch

- Robotic and Autonomous System (RAS) development needs Model-Based Systems Engineering (MBSE) and Agile Development
- Combining MBSE and Agile Software Development can be challenging
- U.S. Army Ground Vehicle System Center created XML/SysML Toolchains for model generation to help combine MBSE/Agile SW

Introduction

"RAS are viewed as the application of software, AI, and advanced robotics to create systems capable of achieving tasks and goals as directed by humans.[1]"

"[RAS is defined as]…the intelligent connection between perception and action[2]"

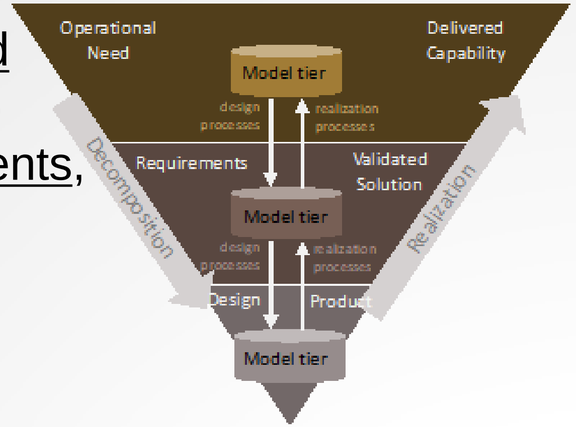
- RAS Benefits to the Warfighter
  1. Reducing the number of warfighters in harm's way
  2. Increasing decision speed in time-critical operations
  3. Performing missions impossible for humans
- Long Term Development Strategy
  - Near-Term
  - Mid-Term
  - Far-Term
- Army RAS Investment
  - Army R&D funding for RAS ground systems in FY23: ~$1.4B

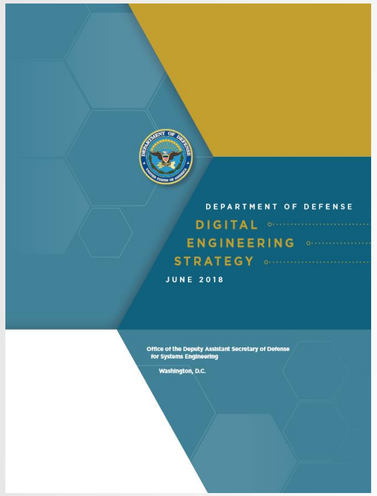
1. Robotic and Autonomous Systems Implementation Coordination Office (RICO), Future Land Warfare Branch, Australian Army Headquarters, Robotic & Autonomous Systems Strategy v2.0
2. Bruno Siciliano and Oussama Khatib. 2007. Springer Handbook of Robotics. Springer-Verlag, Berlin, Heidelberg.

Introduction

DIGITAL ENGINEERING / SYSTEMS ENGINEERING

MBSE: "…the formalized application of modeling to support system requirements, design, analysis, and verification/validation activities[1]" throughout a system's lifecycle





DoD INSTRUCTION 5000.97

DIGITAL ENGINEERING

Originating Component: Office of the Under Secretary of Defense for Research and Engineering

Effective: December 21, 2023

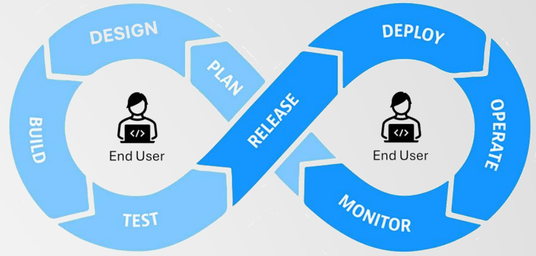
Releasability: Cleared for public release. Available on the Directives Division Website at https://www.esd.whs.mil/DD/.

Incorporates and Cancels: Department of Defense Directive 5000.59, "DoD Modeling and Simulation (M&S) Management," August 8, 2007, as amended

Approved by: Heidi Shyu, Under Secretary of Defense for Research and Engineering

Purpose: In accordance with the authority in DoD Directive 5137.02, this issuance establishes policy, assigns responsibilities, and provides procedures for implementing and using digital engineering in the development and sustainment of defense systems.

Agile: a modern development methodology that "relies on flexible requirements, regular user engagement, and an understanding of the value of what has been delivered[2]"



GAO

United States Government Accountability Office

Report to Congressional Committees

July 2023

DEFENSE SOFTWARE ACQUISITIONS

Changes to Requirements, Oversight, and Tools Needed for Weapon Programs

GAO-23-105867

DoD INSTRUCTION 5000.87

OPERATION OF THE SOFTWARE ACQUISITION PATHWAY

Originating Component: Office of the Under Secretary of Defense for Acquisition and Sustainment

Effective: October 2, 2020

Releasability: Cleared for public release. Available on the Directives Division Website at https://www.esd.whs.mil/DD/.

Incorporates and Cancels: Under Secretary of Defense for Acquisition and Sustainment Memorandum, "Software Acquisition Pathway Interim Policy and Procedures," January 3, 2020

Approved by: Ellen M. Lord, Under Secretary of Defense for Acquisition and Sustainment

Purpose: In accordance with the authority in DoD Directive 5135.02, this issuance establishes policy, assigns responsibilities, and prescribes procedures for the establishment of software acquisition pathways to provide for the efficient and effective acquisition, development, integration, and timely delivery of secure software in accordance with the requirements of Section 800 of Public Law 116-92.

1. E. R. Carroll and R. J. Malins, "Systematic Literature Review: How is Model-Based Systems Engineering Justified?," Sandia National Laboratories, Albuquerque, NM, 2016.
2. United States Government Accountability Office, "DEFENSE SOFTWARE ACQUISITIONS Changes to Requirements, Oversight, and Tools Needed for Weapon Programs," United States Government Accountability Office, Washington, DC, 2023.

MBSE/Agile Challenges

Challenge	Citation
Culture change needed to implement MBSE	[1] [2] [3] [4]
Leadership buy-in of MBSE	[1] [3]
Building and retaining modeling talent	[1] [3]
Lack of MBSE training	[1] [2] [3]
Using models to communicate with stakeholders that do not have MBSE experience	[1] [3]
Integration of modeling tools with agile project management and software development tools	[1] [2] [4]
MBSE processes considered extra work	[2] [3]
Not using Agile development as an excuse to bypass MBSE process	[2]

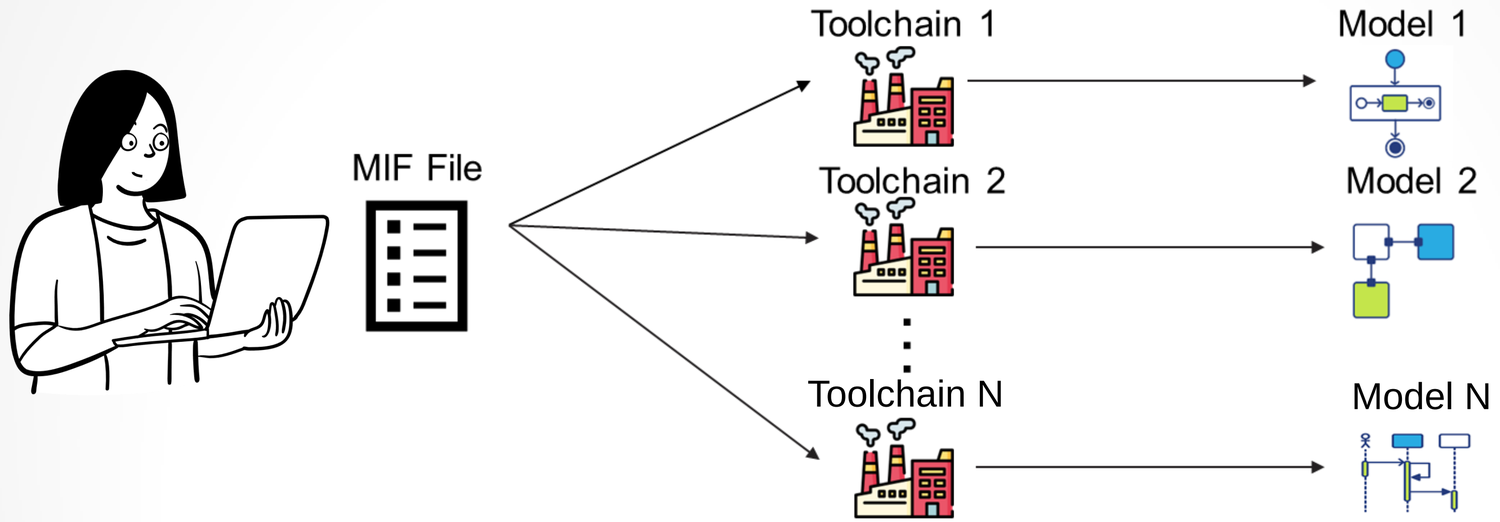
- Common themes:
  - Difficulty in getting MBSE support, tooling, or processes for developers
  - Resistance working with models / limited MBSE experience / perception: added effort, little value
- Bridging the gap between MBSE and Agile software development needs developer-friendly processes and toolchains to handle the modeling

1. M. Pantano and F. Galiber III, "Applying an Agile Approach with MBSE," in Agile in Government Summit, Washington, DC, 2019.
2. M. Hause, "How to Fail at MBSE," Atego, 202.
3. M. Chami and J.-M. Bruel, "A Survey on MBSE Adoption Challenges," in The Systems Engineering Conference of the Europe, Middle-East and Africa (EMEA) Sector of INCOSE, Berlin, Germany, 2018.
4. J. Kößler and K. Paetzold, "Integration of MBSE into existing development processes - Expectations and challenges," in DS 87-3 Proceedings of the 21st International Conference on Engineering Design (ICED 17) Vol 3: Product, Services and Systems Design, Vancouver, Canada, 2017.

Model Import Files

- Model Import Files (MIF)
  - Extensible Markup Language (XML) files tailored for the development of autonomous software.
  - Maintained by the software developers
  - Machine readable
  - Utilized in SysML toolchains to automate creation of diagrams

- Software developers are **NOT** burdened by SysML modeling



Model Import Files - ROS

- Robot Operating System (ROS)
  - Framework for the development of RAS software
  - Collection of tools, libraries, and conventions to enable RAS development and reuse
  - Widely used (approximately 550 million ROS “packages”)

ROS

Model Import Files - ROS

- MIFs leverage fundamental ROS concepts as a starting point

Concept	Definition
Nodes	A process that performs computation, synonymous with “software module”.
Messages	Strictly typed data structures that nodes pass to each other to communicate.
Topics	Named buses that nodes send messages on with a publish/subscribe message pattern.
Packages	Unit of organization in ROS software that provides all the files needed for some functionality (ROS nodes, a ROS-independent library, configuration files, etc.) for a logically standalone purpose.
Services	A reply/request process for synchronous transactions, analogous to a web service.
Parameters	A shared variable for ROS nodes stored on a parameter server.

Model Import Files – XSD

- MIFs are defined by an XML schema definition (XSD)
  - Defines elements, attributes, and datatypes in an XML
- MIF XSD captures important ROS interface concepts
- Every ROS package has an associated MIF that adheres to the XSD
  - Kept up-to-date along with code

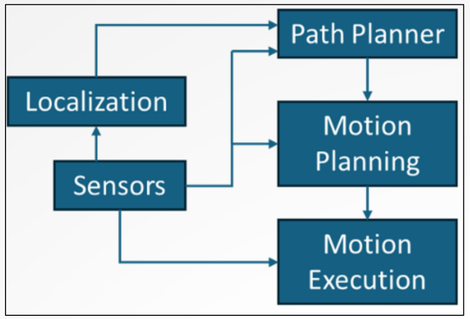
MIF/SysML Import Toolchains

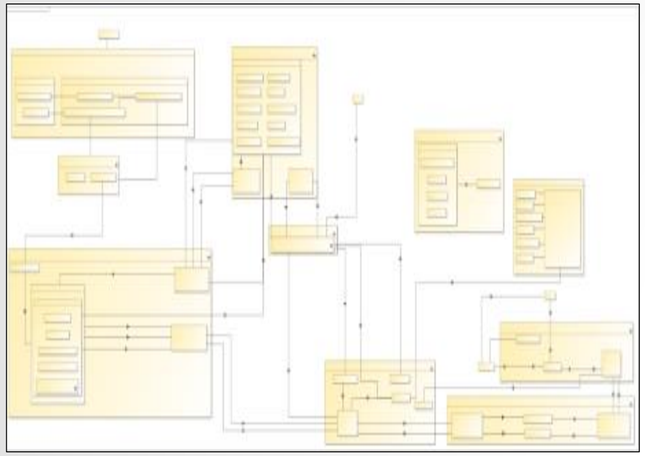
- MIFs provide a baseline of information for different SysML toolchains
- One set of MIFs can support different modeling needs for different programs
- Two specific current use cases:
    - Army Robotic Common Software (ARCS) Library Documentation Toolchain
    - Autonomous Ground Vehicle Reference Architecture (AGVRA) Interface Model Library (IML) - ARCS Toolchain.

MIF/SysML Import Toolchains

ARCS Library Documentation Toolchain (ALDT)

- Traditionally, software architecture diagrams have been “hand cranked”
- Gets out-of-sync with code

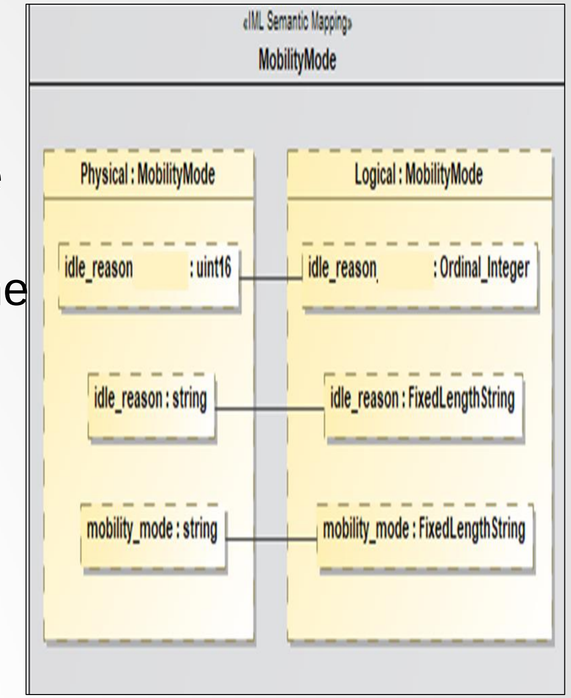




- ALDT uses MIFs to generate internal block diagrams of ARCS components
- Reduces “hand cranking” and ensure synchronization

IML-ARCS Library Toolchain (IALT)

- AGVRA IML profile defines stereotypes for describing logical & physical interfaces
  - Logical interfaces define data meaning
  - Physical interfaces define implementation details
- Mapping models show how the logical and physical interfaces are connected.
- IALT generates AGVRA IML diagrams for greater coherence/interoperability between RAS software and other systems



Conclusion

- MBSE and Agile software development are important priorities of the DoD system development
- Synchronizing MBSE and with an Agile software development approach can be challenging
- Utilization of MIFs and SysML Import Toolchains help bridge the gap between MBSE processes and Agile developer flexibility for RAS
- Using toolchains to synchronize between models and code allows us to expand the integration between Agile software development and MBSE for better RAS for Soldiers