Mission Architecture Style Guide

Version 1.0

January 6, 2025

Office of the Under Secretary of Defense
for Research and Engineering

Mission Capabilities

Washington, D.C.

Department of Defense
Mission Architecture Style Guide

Office of the Under Secretary of Defense for Research and Engineering
3030 Defense Pentagon
Washington, DC 20301
osd.r-e.comm@mail.mil
https://ac.cto.mil/mission-engineering/

Distribution Statement A. Approved for public release. Distribution is unlimited.

DOPSR Case No. 25-T-0714.

UNCLASSIFIED

OFFICE OF THE ASSISTANT SECRETARY OF DEFENSE
3030 DEFENSE PENTAGON
WASHINGTON, DC 20301-3030

MISSION CAPABILITIES

MEMORANDUM FOR RECORD

SUBJECT: Mission Architecture Style Guide Release
Appendix to the Mission Engineering Guide, Version 2.0

This memorandum announces the release of the Department of Defense Mission Architecture Style Guide (MASG) as an appendix to the Mission Engineering Guide (MEG) 2.0. The development of this guide was led by the Deputy Assistant Secretary of Defense (DASD) for Mission Integration (MI) directorate within the Office of Defense for Research and Engineering, Mission Capabilities (OUSD R&E / MC) with support from the Department of Defense Mission Engineering Practitioners Forum and external ME stakeholders.

The MASG will aid model-based systems engineers and architects in the creation, presentation, and analysis of model-based mission architectures. The style guide was developed to synergize the creation and sharing of mission architectures across the Department of Defense (DoD). The MASG focuses on mission architecture development as a fundamental element of Mission Engineering as defined in the MEG 2.0.

This document is a direct result of the 2023 Digital Mission Architecture Workshop and subsequent guidance from the Deputy Assistant Secretary of Defense for Mission Integration to define a uniform process for model-based mission architecture development at the request of various DoD Components. The MASG is a foundational element for mission architects to develop models for Joint mission area analysis that leverages inputs from subject matter experts across the Department and documents a general consensus on mission architecture content. The MASG facilitates integration of Joint, model-based mission architectures across the Department.

The MEG 2.0 and its MASG appendix are available online from the Department of Defense Chief Technology Officer's website, at https://ac.cto.mil/mission-engineering. The ASD(MC) Mission Integration team will coordinate the release of future versions of this guide to clarify and expand on the mission engineering architecture applications and best practices.

ROMAN.ELMER.LUIS.1181775064

Digitally signed by ROMAN.ELMER.LUIS.1181775064
Date: 2024.12.27 17:23:47 -05'00'

Elmer L. Roman
Deputy Assistant Secretary of Defense
for Mission Integration

KANDIL.DINA.AHMED.1601691511 Digitally signed by KANDIL.DINA.AHMED.1601691511 Date: 2025.01.07 09:04:22 -05'00'

Dina A. Kandil
Mission Architect

LEESE.ANGELA.N.1408099591 Digitally signed by LEESE.ANGELA.N.1408099591 Date: 2025.01.07 09:23:09 -05'00'

Angela N. Leese
Mission Architect

SCHOEPF.MEGAN.M.1151998581 Digitally signed by SCHOEPF.MEGAN.M.1151998581 Date: 2025.01.07 09:15:10 -05'00'

Megan M. Schoepf
Mission Architect

Rock A. Mendenhall

Rock A. Mendenhall
Deputy Chief Engineer, Digital Mission Architecture

Digitally signed by BESTARD.JAIME.J.1292648150 Date: 2025.01.06 17:37:02 -05'00'

Jaime J. Bestard
Chief Engineer, Digital Mission Architecture

Table of Contents

Table of Figures

Table of Tables

1 Introduction

1.1 Background

Mission engineering (ME) is an interdisciplinary process encompassing the entire technical effort to analyze, design, and integrate current and emerging operational needs and capabilities to achieve desired mission outcomes.[1] This style guide is intended to help mission architects apply Model-Based Systems Engineering (MBSE) approaches and the principles outlined in the Office of the Under Secretary of Defense for Research and Engineering (OUSD(R&E)) *Mission Engineering Guide version 2*.0 (MEG 2.0) to create mission architectures that inform Department of Defense (DoD) decision making. Mission architectures align mission outcomes, requirements, and capabilities in a systematic way through graphical representations that enhance stakeholders' understanding of the mission space and associated problem areas. Artifacts are constructed to enable stakeholders to leverage completed studies, to run additional excursions, or extend the studies to address a greater scope.

Mission architectures ensure both problems and potential solutions are systematically decomposed so that leadership and stakeholders can use them to make informed decisions. Mission architecture modeling supports Mission Engineering Analysis through an iterative process of characterize, build, analyze and update. Digital modeling tools accelerate design cycles by providing a shared repository for information, allow reuse of model artifacts and enable the rapid reconfiguration of models to evaluate alternatives. Mission architectures help realize object-oriented systems engineering, enabling a digital life cycle representation. This guide provides examples for developing mission architectures to meet various stakeholders' intent.

1.2 Purpose of the Mission Architecture Style Guide

The Mission Engineering Architecture Style Guide (MASG) is targeted for use by model-based system engineers and architects supporting the Joint Staff, Combatant Commands, OUSD, Combat Support Agencies, and Military Services. The guide serves to standardize mission architecture development within the DoD, promote model federation and modularization, and support model development by reducing variances in the application and representation of elements and relationships used to define a model. The standardization of modeling within the defense enterprise enhances the shared model understanding and use across the mission engineering community. This joint effort affords collaboration and reduces duplications of effort. By utilizing expertise across the enterprise, model fidelity is increased. The objective of this guide is to facilitate the collective production of MBSE models that can be employed by the Department at large.

While the approaches described within should be broadly applicable, they are designed for the types of mission engineering studies typically conducted by OUSD(R&E)'s Mission Integration element. These studies traditionally involve the evaluation of technologies or systems as

[1] Department of Defense OUSD(R&E). (2023). Department of Defense Mission Engineering Guide 2.0. Washington, DC. Retrieved from https://ac.cto.mil/wp-content/uploads/2023/11/MEG_2_Oct2023.pdf

insertion options into a DoD mission. Despite this being a tailored approach, many aspects and practices can be further tailored for other modeling efforts.

As part of the modeling process, certain approaches, syntax, and presentation techniques were discovered to work well for the Mission Integration's use cases, and these are described in the guide. In alignment to a FY24 memorandum for the mission engineering executive steering council, MI and this guide evaluated how the Unified Architecture Framework (UAF) could conform to the style of mission threads described in the MEG 2.0[2]. The example approach defined is just one way to approach architecture when using the UAF and is intended to be informative. This Style Guide should be used in conjunction with the Enterprise Architecture (EA) Guide for UAF provided as part of the UAF standard.[3]

For purposes of providing modeling examples throughout the guide, architectures were created using the open-source example of Operation DESERT STORM, simplified to increase readability and comprehension of the models. These examples were produced utilizing the Unified Architecture Framework Modeling Language (UAFML) v1.2 in Cameo System Modeler 2021x, but the overall approach is intended to be software tool and language agnostic. UAFML is based on Systems Modeling Language (SysML) and further includes enterprise-level views that provide capability and enterprise modeling concepts.

The MASG is not a step-by-step handbook on how to create mission architectures; rather the guide outlines a scalable approach to promote mission architecture sharing and collaboration. The MASG:

1. Guides mission architects in the development of DoD model-based mission architectures that support various aspects of capability development through the lifecycle.
2. Facilitates the collective production and sharing of authoritative architectures that can be employed by the DoD community at large, while remaining in alignment with international Object Management Group (OMG) standards.
3. Advises mission architects in the development of views and presentation materials to inform Government and DoD Senior Leaders on key policy and investment decisions that drive technology development.

---

[2] Department of Defense OUSD(R&E). (2023, October 19). Memorandum for Mission Engineering Executive Steering Council. Enabling Digital Mission Architecture Integration Across the Department of Defense During Fiscal Year 2024. Washington, DC.

[3] Object Management Group. (2021). United Architecture Framework (UAF) Version 1.2 Enterprise Architecture Guide for UAF. Retrieved from https://www.omg.org/spec/UAF/1.2

2 Architectures in the Mission Engineering Process

2.1 Overview

The MEG 2.0 describes mission engineering as a five-step process depicted in Figure 1. There is an explicit step for "Mission Architectures", such that each of the other steps impacts and/or leverages the mission architecture[4]. This document contains one subsection for each of the MEG 2.0 process steps which describes the role of architecture and provides guidance on how to develop the supporting aspects of the architecture. The mission architecture development guidelines described in this document align with the mission engineering process. While the figure below highlights a serial process flow, mission architectures are created to support all steps of the ME process to include capturing intelligence information, analysis results, experimentation feedback, live testing, and recommendations. There is a digital thread feedback loop in which the results are used to update the model-based architectures and constructive simulations. For additional details related to the ME methodology, refer to Section 2.2 of the MEG 2.0.

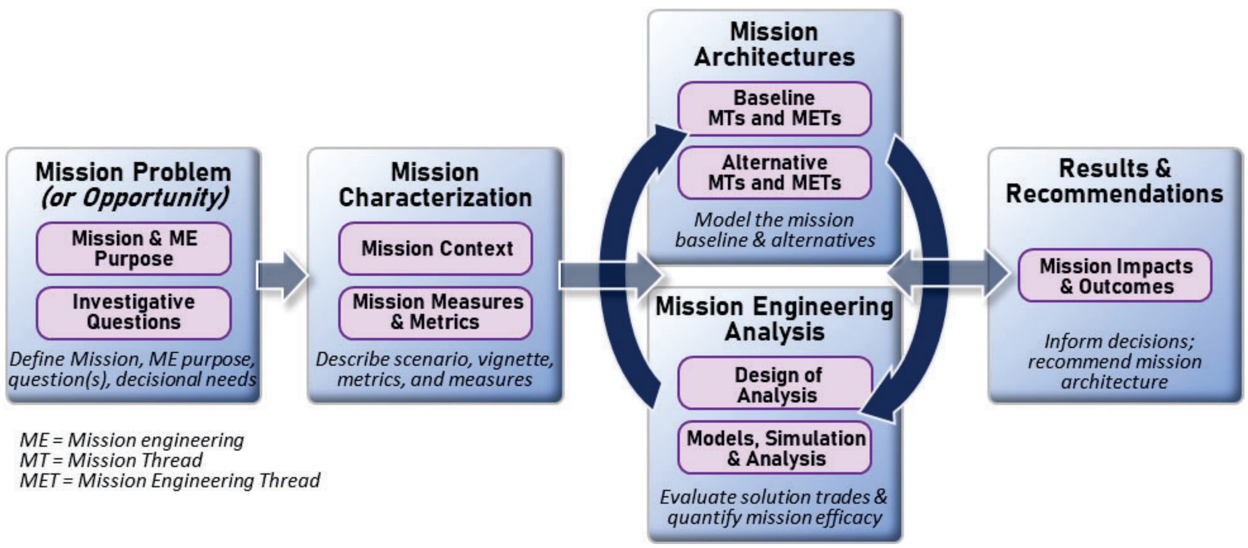

Figure 1. Mission Engineering Process (Source: MEG v2.0)

[4] Department of Defense OUSD(R&E). (2023). Department of Defense Mission Engineering Guide 2.0. Washington, DC. Retrieved from https://ac.cto.mil/wp-content/uploads/2023/11/MEG_2_Oct2023.pdf

2.2 Considerations

2.2.1 Types of Architecture and Key Terminology

Architecture Categories with associated definitions[5] are below:

Enterprise Architecture: Applies the basic definition of architecture (structure, behaviors, and global rules) to the top level of an ensemble of nodes, systems, elements, or other resources that collaborate to fulfill the functions of an overall organization or business process. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Mission Architecture: A view or representation that depicts the ways and means to execute a specific end-to-end mission, with relationships and dependencies amongst mission elements. This includes elements such as mission activities, approaches, systems, systems of systems, organizations, and capabilities. [Source: OUSD(R&E) MEG 2.0]

System Architecture: Applies the basic definition to an ensemble of elements (Ultimately hardware and software components) that collaborate to fulfill defined requirements allocated to a node or system (Implying that a clear system boundary and user interfaces are defined). [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Software Architecture: Applies the basic definition to software, focusing on frameworks, software requirements, application programs, infrastructure programs, workflow management, networking and messaging, interfaces, and other aspects of computer programming. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Hardware Architecture: Applies the basic definition to hardware, focusing on processors, storage, interconnects, operator stations, communications, sensors, effectors, and other hardware elements. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Reference Architecture: A logical/functional abstraction that defines the features and behaviors common to a domain or class of entities. An RA is instantiated by the addition of relevant detail to achieve a physical architecture that satisfies a specific set of requirements within the domain. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Executable Architecture: Represents the architecture in the form of computer models that can be run to simulate behaviors, perform automatic code generation, verify design correctness, etc. Executables exist at various levels of abstraction used to describe an architecture. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

[5] Borky, J. M., & Bradley, T. H. (2019). Effective Model-Based Systems Engineering. Springer.

2.2.2 Object Management Group Mission Engineering Guide for UAF

The OMG will publish an "ME Guide for UAF" that includes Mission Engineering model elements, extending the basic elements in the UAF metamodel. Once published, the OMG "ME Guide for UAF" will be included in the UAF v1.3 update and can be used in conjunction with this MASG and the MEG 2.0 to formulate mission architectures.

Both Enterprise Systems Engineering and Mission Engineering disciplines model a range of enterprise views and considerations, encompassing materiel, systems-focused solutions, and non-materiel (DOTMLPF-P) solutions. These approaches incorporate strategy, programs, people, processes, and capabilities that transcend traditional system boundaries (systems, components, and functions). The DoD, as an enterprise of enterprises, has complex logical and physical architectures that evolve over time. The UAFML builds upon the SysML to account for temporal and non-materiel aspects of architecture.

EA and mission architecture modeling break down complex problems into manageable components. Figure 2 illustrates a conceptual model focused on the key entities and relationships involved in enterprise modeling. To initiate the enterprise modeling process, architects can define and capture mission's needs, the required capabilities, desired effects, and outcomes in the Strategic viewpoint of UAF. Next, they can define mission-level behaviors from a solution agnostic perspective using the Operational viewpoint. The Resources viewpoint can then be used to codify how the capability need will be met. Finally, the Personnel and Projects viewpoints can be employed to model the Who, When, and Where aspects of the mission.

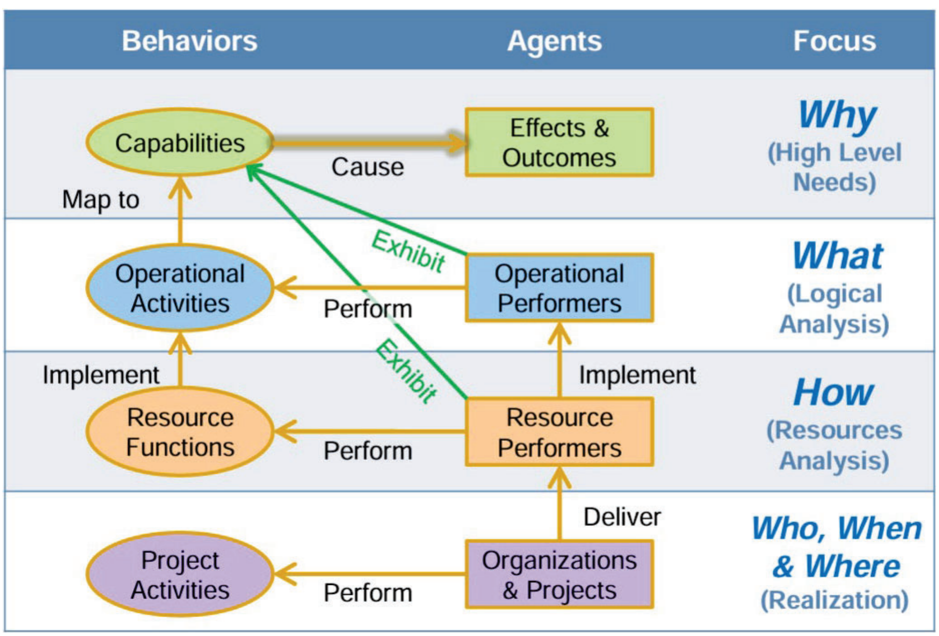

Figure 2. Key Relationships between Behaviors and Agents in UAF[6]

[6] Copyright 2024 James Martin. Used with Permission.

2.2.3 Relation to DoDAF

The Department of Defense Architecture Framework (DoDAF) and UAF share a common foundation on the high-level characterization of an enterprise's Strategic, Operational, and Personnel viewpoints. These viewpoints capture many of the key non-materiel aspects of the enterprise. The Strategic viewpoint in UAF outlines the key drivers and challenges to the enterprise, the critical opportunities to be pursued by the enterprise, and the mission and business capabilities needed (the WHY) for achieving enterprise objectives. The Operational viewpoint (in UAF and DoDAF) deals with the key operational performers and the activities (the WHAT) performed to support enterprise capabilities. In response to the capabilities and operations defined in this manner, the necessary services, personnel, and resources will be defined that equip the enterprise (the HOW) to achieve its enterprise objectives embodied in the mission and business capabilities.

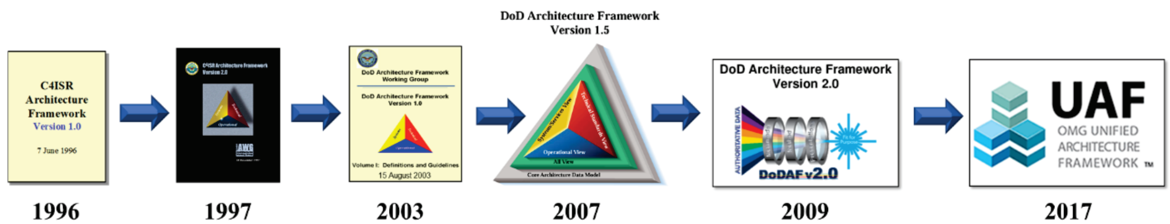

Figure 3. Evolution of DODAF to UAF[7]

Figure 3 illustrates the evolution of DoDAF to UAF. The UAF builds upon the foundation of DoDAF, providing a more refined analysis tailored to mission architecture and enterprise systems engineering. For a detailed mapping of UAF views to DoDAF 2.02, refer to Appendix A[8] Table 2.1 of the UAF version 1.2 specification, which contains a direct crosswalk of equivalent UAF views that will eventually supersede DoDAF 2.02. This resource will provide guidance for developing DoDAF views that complement those discussed in this guide.

2.2.4 Utilizing UAFML and SysML

UAFML is derived from the SysML and Unified Modeling Language (UML), so they are intended to work together. Using UAFML and SysML allows for modeling at an enterprise-mission and systems level, respectively, providing the desired level of detail to inform stakeholder concerns. SysML and the UAFML enable a MBSE approach by providing standard notations and semantics to model systems and system-of-systems with many configurations which is essential to enterprise modeling. UAF is the framework used to depict system-of-systems with many configurations which is essential to enterprise modeling. EA is used to align organizations with their business strategy and goals; it takes a holistic view of the interdependencies of an organization to ensure that the technology infrastructure supports the

---

[7] Martin, J. (2024, July). Enabling Enterprise Transformation Using Enterprise Architecture Principles and Concepts. 34th Annual INCOSE Symposium. Dublin.

[8] Object Management Group. (2022). Unified Architecture Framework (UAF) Traceability Appendix A Version 1.2. Retrieved from https://www.omg.org/cgi-bin/doc?formal/22-07-07.pdf

business processes effectively. Conversely, systems architecture focuses on the design and integrations of individual systems ensuring that each system is designed to meet the operational and programmatic requirements.[9] UAFML extends SysML to help model enterprises effectively analogous to the manner that SysML extended UML to support modeling systems, as UML was originally intended for visually modeling software architectures. The UAFML specifies UAF domain metamodel implementation in terms of the UML and SysML. It defines UML extensions (so-called stereotypes) that characterize the UAF DMM. It is also dependent on a SysML profile, which is another extension to UML. This is intended to provide more seamless integration with system modeling, using SysML, and to be able to fully leverage the capabilities of SysML in UAFML (Figure 4).

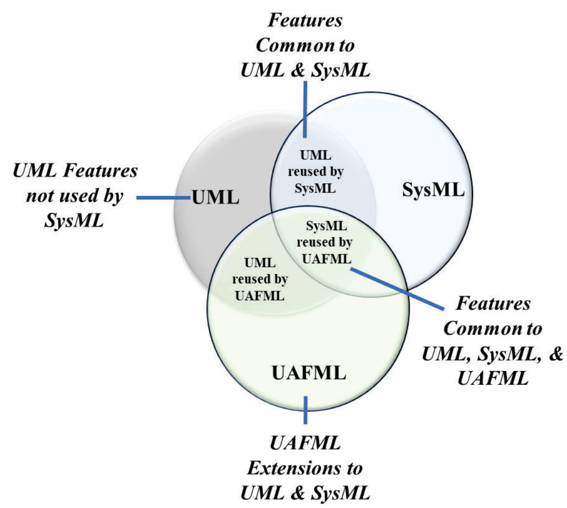

Figure 4. UML, SysML, and UAFML Overlay

UAFML is an international OMG standard that assists in development of architectural descriptions in federal government agencies, defense organizations, and commercial industry firms. UAFML supports a variety of use cases from Enterprise and Mission architecting to Cyber-physical Systems engineering. The UAF evolved from the Ministry of Defence Architecture Framework (MoDAF), the DODAF, the NATO Architecture Framework (NAF), and various frameworks to define a standard ontology between stakeholders when evaluating and supporting collaboration between organizations throughout the lifecycle and can support mission architecture collaboration. UAFML provides a set of extended stereotypes and rules to create consistent enterprise and/or mission architectures as digital models. These MBSE models become repositories from which various views can be extracted to highlight dependencies, traceability, and answer key questions for Senior Leaders and mission architects.

In UAF v1.2 Appendix A, there are “UAF 1.2 to DODAF 2.02” and “UAFML Stereotype to SysML and UML Metaclass” mapping tables.[10]

[9] Shreve, G. (2024, March 20). UAF or SysML - Yes? Retrieved from OMG UAF Summit: https://www.omg.org/events/2024Q1/special-events/UAF-presentations/1615-UAF_SysML_Yes_Updated-2024-UAF-Summit%20v2.pdf

[10] Object Management Group. (2022). Unified Architecture Framework (UAF) Traceability Appendix A Version 1.2. Retrieved from https://www.omg.org/cgi-bin/doc?formal/22-07-07.pdf

2.2.5 UAFML for Mission Architectures

There are multiple ways to document a mission architecture. This guide represents the OUSD(R&E) MI application of the UAF specification to create mission architectures. UAFML is designed to model an enterprise and is appropriate for modeling large mission architectures. Mission architectures can support capability planning and portfolio management by representing key enablers for fielded capabilities, such as systems, services, people, and processes. UAF models provide a means to develop an understanding of the complex relationships that exist between organizations, operations, systems, and services as well as enable analysis that ensures that expectations of the user community are meant.[11]

Figure 5 shows an overlay of the UAF grid that highlights the various views that can be leveraged to support the ME process. The UAF grid viewpoints (rows) and aspects (columns), together represent information in the model. The Strategic viewpoint is typically utilized to support the capability management process, and define the mission context, goals, capabilities, and threats. The Operational viewpoint maps to the mission thread level and provides a solution agnostic view of the mission; it describes the requirements, behaviors, structure, and necessary exchanges without solutioning. The Resources viewpoint represents a more solution-specific view into the mission through the mission engineering thread. At the mission engineering thread level, the organizations, technologies, systems, etc. are allocated to the different functions they will perform. For all viewpoints, the measures and measurements that are used to evaluate the architecture, such as the measures of success (MOS), measures of effectiveness (MOE), and measures of performance (MOP), can be captured in the Parameters aspect. Note that the views shown in this guide and highlighted in Figure 5 are a starting point for what can be created to support the mission engineering process. The Services, Personnel, Security, Projects, and Standards viewpoints provide additional perspectives into the model information that can be beneficial to represent a complete mission, especially for non-materiel elements. The UAF profile contains complementary views that can be leveraged in transition from other architecture modeling frameworks or languages, like DODAF and Business Process Modeling Notation (BPMN). For offices utilizing BPMN to depict activity sequences and the involved humans and organization, it is advisable to use the UAF Personnel-Processes Flow (Ps-Pr) diagram.

[11] Object Management Group. (2021). United Architecture Framework (UAF) Version 1.2 Enterprise Architecture Guide for UAF. Retrieved from https://www.omg.org/spec/UAF/1.2

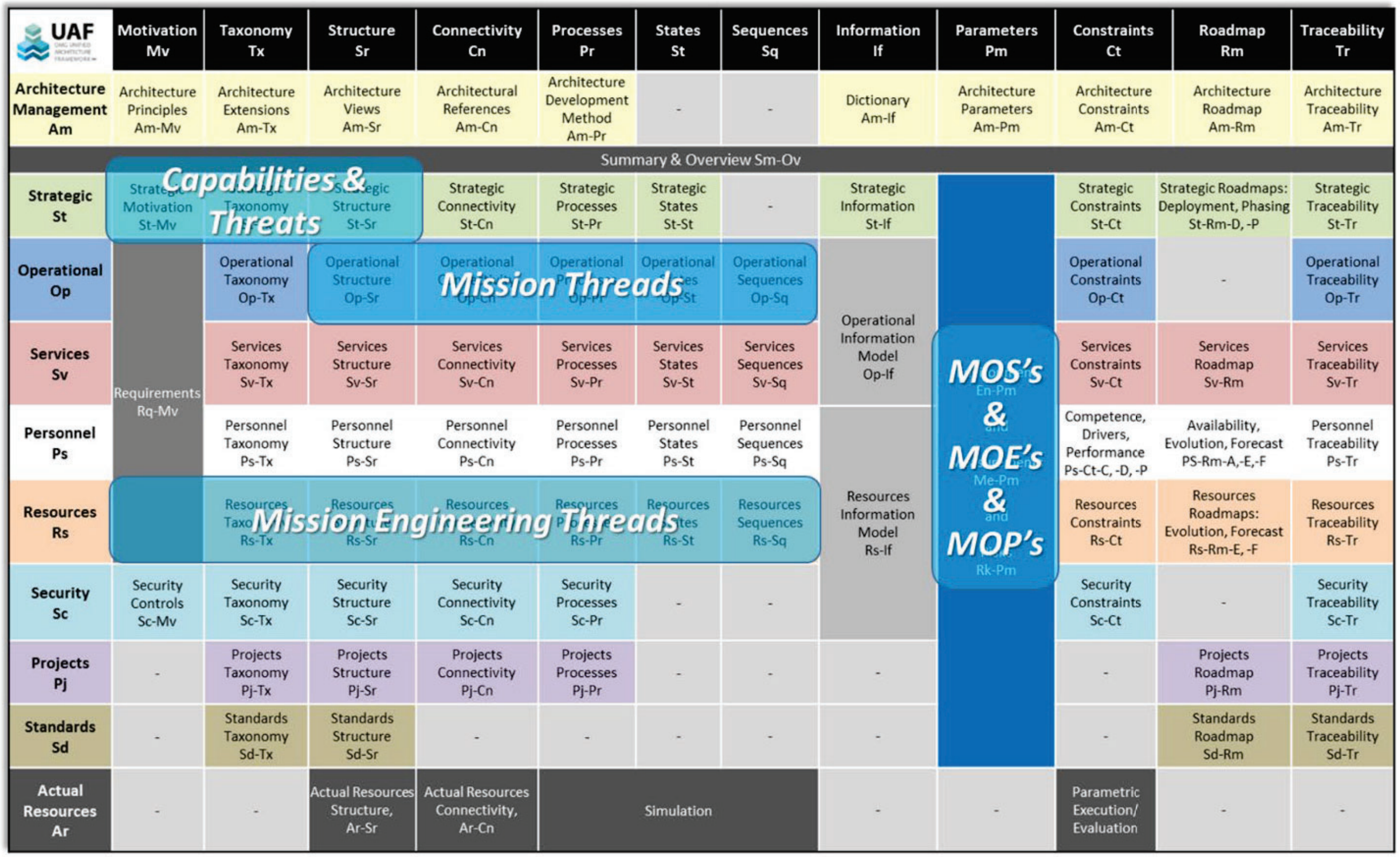

Figure 5. Mission Engineering Views in UAF v1.2[12]

[12] Copyright 2023 James Martin. Used with Permission.

2.3 Importance of a Federated (Modular) Architecture

Although there are infinite ways to separate ME models into reusable and case-specific information, UAF already segments model information such that one could simply create separate models based on the top-level packages: Strategy, Operational, Services, Personnel, Resources, Security, Projects, Standards, and Actual Resources. Federation needs should be considered as early as possible so models can be partitioned and built to facilitate architecture management and governance within the Department. This approach helps improve time to query the model, reduce model access conflicts among team members, promote reuse of authoritative model libraries, and allow for greater control over model changes and configuration control. Having a common architecture modeling framework and governance structure will serve as a balanced foundation that exhibits rigor and flexibility to enable interoperability, reuse, and maturity.

Federation is complex to execute but helpful in nature. Modularization promotes employing effective tenets of architecture. Modularity, interoperability, openness, loose coupling, composability, standards compliance, clarity, adaptability, and scalability are benefits of working together for creating architectures that can be used by many and for the greater DoD team, versus for a single organization. The tenet definitions mentioned here can be referenced in this guide's glossary for additional awareness.

A federated or modular model allows the reuse of models and their elements to completely define the architecture for a system of interest. This allows different stakeholders (Service Components, the Office of the Secretary of Defense, Federally Funded Research & Development Centers (FFRDCs), University Affiliated Research Centers (UARCs)), and other trusted organizations to possess a common source of truth when creating architectures, and to see only the relevant model elements and views. Common element libraries, profiles, and languages provide the standard architecture framework for basic compatibility between architecture models. Authoritative architecture information will be provided and validated by Services. Mission Integration will validate joint libraries using inputs from the Services. If no authoritative information can be identified or found, they must be documented in the model and/or included in any accompanying model documentation as a potential area for future investigation.

Using the Joint Publications, Universal Joint Task List (UJTL), Joint Common System Function List (JCSFL), Joint Capability Areas (JCAs), and inputs from the military components, common libraries will be populated and leveraged for the creation of mission architectures. To tailor to specific solutions, common element libraries (e.g., systems, organizations, roles, facilities, policy, doctrine) will be drafted by those with domain expertise and shared within the DoD community. This collaboration allows for a common source of truth (with versioning) that enhances the fidelity of the architectures created.

Figure 6 represents an overview of the proposed model organization of information from the Scenario layer down to the Platforms and Systems layer. At the most solution-agnostic level, doctrine and intel inform the mission. The first two layers, (Platforms and Systems, and Platform Configurations) will consist of basic starting templates for reuse when developing the necessary mix of contributors to accomplish a mission. These models will be maintained and provided as a library of possible options, matching capabilities to needs. Above the red dashed line, information becomes more and more mission specific, but maintains some re-usability when considering alternatives or excursions. Information flows from the bottom layer up only. Important considerations must be made when deciding where to define a model element so that it is visible to those other models that may need it. Specialization is always possible to add extra layers of information to existing model elements.

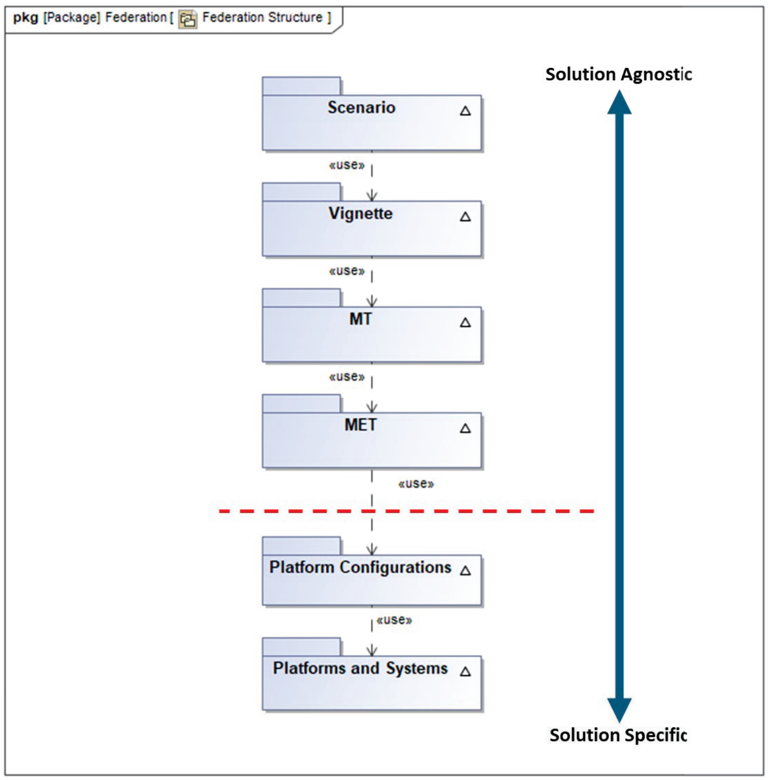

Figure 6. Federated Architecture Model Organization

The end vision is to enable unity of effort by bringing in Service stakeholder liaisons that can help serve on a holistic model integration team, like the U.S. Olympic team, while still primarily belonging to their home organizations (e.g., USAF, USMC). Figure 7 depicts a hub and spoke model with the Office of the Deputy Assistant Secretary of Defense for Mission Integration (ODASD(MI)) as the central hub for mission architectures and corresponding libraries. The spokes between ODASD(MI), the Service Components, Joint Staff, OUSD (Acquisition & Sustainment (A&S)) and other organizations is a bidirectional flow of architecture information going to and from all "hub and spoke" units. Analogous to an airline, the information is funneled to the central hub to help synchronize Department efforts. These mission architectures can also be used by various stakeholders as reference architectures, reference patterns, and a source overall.

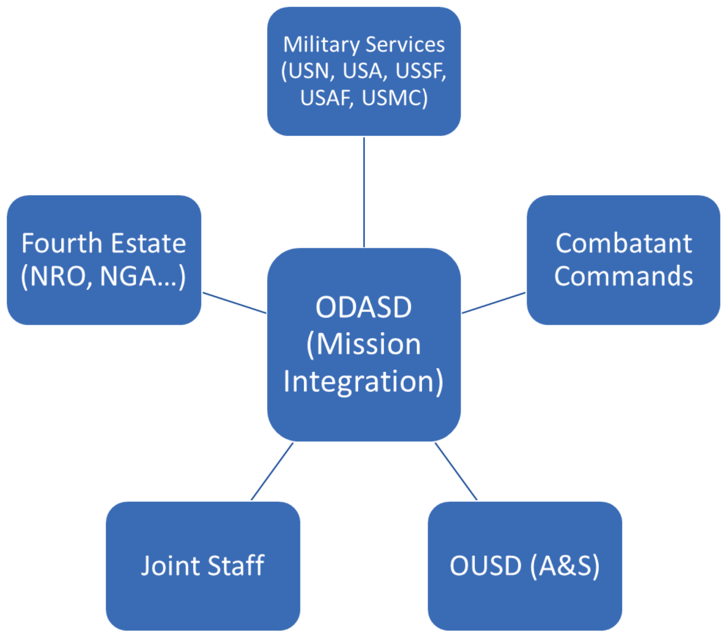

Figure 7. Architecture Hub and Spoke Model Exemplar

The hub and spoke mission architecture exemplar offers the following advantages:

- *Increased Insight and Connectivity*: Hub-and-spoke operations enable stakeholders to promote bi-directional connectivity between organizations and provide insight and transparency.
- *Centralized Architecture Integration and Decentralized Architecting*: ODASD(MI) can help serve as the central hub to perform governance, prevent duplication, focus resources, establish information sharing mechanisms, and share architecture constructs with stakeholders. This does not mean the central hub will necessarily control the model; MI will choose insight versus oversight and let the stakeholders perform decentralized architecting.

2.4 Mission Architecture Model and Views

2.4.1 Mission Engineering Model Organization

The important point to remember is that the model is the data/information and relationships among the elements. The views provide a slice of the model elements. Figure 8 shows a presentation flow view created in the digital engineering tool. This allows users to navigate through the model between the different views without the need to export the views, analogous to navigating through a PowerPoint briefing from start to finish.

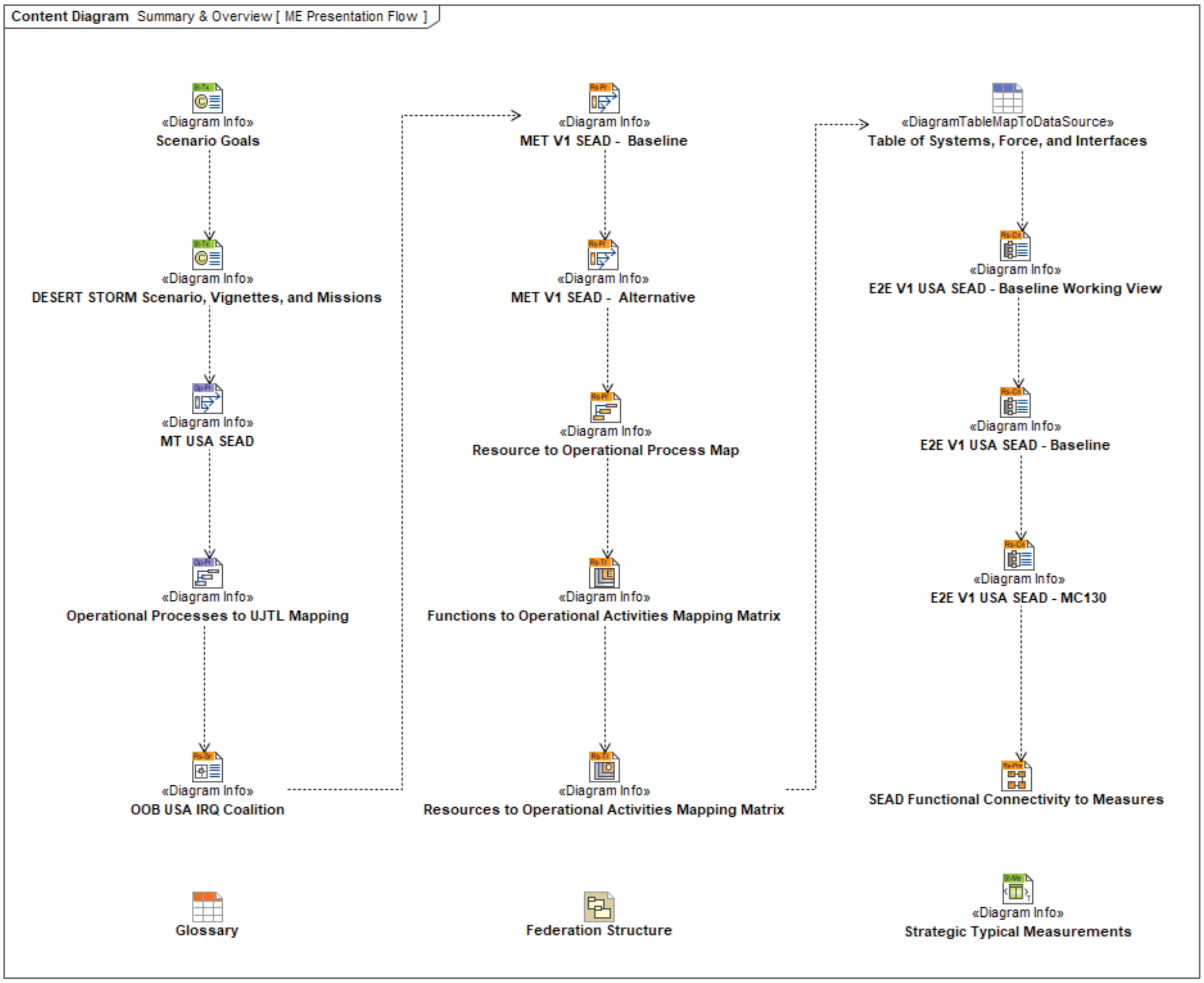

Figure 8. Summary & Overview Presentation Flow

Having a well-structured model is critical for organizing the information that is presented in mission engineering views. One of the major benefits of creating models rather than utilizing PowerPoint or other textual representations is that elements can be represented in different views and dynamically show connections. The following sections in the style guide show some views and how they align with the Mission Engineering process.

2.4.2 Mission Engineering Architecture Views

There are many methods to applying UAF to mission engineering as displayed in Figure 5. For the context of this style guide, a primary set of mission engineering architecture elements and views are identified to support a mission engineering initiative. This primary set is not exhaustive, but represent a starting point for capturing the mission elements (person, organization, platform, and/or system that performs a task), relationships, and processes as performed by OUSD(R&E) MI. These include:

Table 1. Primary Mission Engineering Architecture Artifacts

Primary Artifact	**Purpose**	**UAF View**
Scenario Goals	View of scenario and operation mapped to goals	Strategic Taxonomy (St-Tx)
Scenario Breakdown	Displays traceability/breakdown from scenario to vignette, mission, mission thread, and mission engineering thread.	Strategic Taxonomy (St-Tx)
Mission Thread (MT)	Doctrine-based and solution-agnostic; describe a set of tasks, activities, and events in an approach to conduct a mission	Operational Process Flow (Op-Pr)
Order of Battle (OB)/ Asset List	View of all mission elements for a scenario, operation, and vignette	Resources Structure (Rs-Sr)
Mission Engineering Thread (MET)	Assign the actors to functions that perform the tasks, activities, and events in the approach to conduct a mission	Resources Process Flow (Rs-Pr)
Mission Engineering Thread End-to-End (E2E) View	Connections between mission elements and the exchange between them. View recommended for presentation to Senior Leadership.	Resources Internal Connectivity (Rs-Cn)

Additional architecture artifacts may be captured based on the scope of the mission engineering problem to highlight additional relationships that exist between organizations, systems, and SoS. These diagrams include but are not limited to:

Table 2. Potential Additional Mission Engineering Architecture Artifacts

Additional Artifact	Purpose	UAF View
UJTL Task to MT Task Mapping	Traces back MT activities (tasks) to Doctrine	Operational Processes (Op-Pr)
MT Task to MET Function Mapping	Traces MET functions to MT tasks to confirm a complete battle plan	Resources Processes (Rs-Pr)
Table of Measurements	Represents the measurements (MOS, MOE, MOP) for a scenario/operation/vignette	Typical Measurements Matrix (Pm-Me)

3 Mission Problem or Opportunity

The initial step in the ME process is to identify and understand the mission problem. At the enterprise level, architects work with key stakeholders to understand the mission context for the mission engineering study. Architects collect and review existing relevant architectures and sourced artifacts to help understand the problem space. These mission architectures are living digital artifacts that are updated and reused over time. This includes information on how the problem or similar problems were described in the past, such as through validated MTs from Joint Staff. It is essential to understand the needs and scope of the mission problem to inform the study methodology. Refer to Section 3 in the MEG 2.0 for additional details related to identifying the mission problem and opportunity.

As discussed in the MEG 2.0, the mission engineering purpose can take one of four forms: identifying capability gaps, exploring cause and effect, evaluating trade space of potential solutions, and investing mission impact of new opportunities. The Operation DESERT STORM example below showcases the "identify capability gaps" mission purpose and associated questions to scope the problem.

Example Mission Problem: Operation DESERT STORM

The mission problem: In 1991, in response to Iraq's invasion of Kuwait and subsequent refusal to withdraw, Coalition forces, led by the United States, opted to use military force to liberate Kuwait and dismantle Saddam Hussein's war-making capabilities. Iraq fielded one of the most robust integrated air defense systems in the world. With this primary challenge, the Coalition force must identify a way to create an opportunity to seize air superiority to facilitate further military action.

Identify capability gaps: The goal of the study and the purpose for modeling the architecture is to identify any potential gaps in the plan that might impact mission success and evaluate potential solutions to address the gaps.

1. Does the planned mission support the Coalition force ability to open a route for further military movement and action in Iraq?
2. What are the limiting factors or gaps that can prevent the Coalition force from meeting their mission objectives?

Note: This example has been further simplified in this document for demonstration purposes relevant to Defense Department's oriented customers. The architectures produced can be expanded based on the problem set.

A Strategic Taxonomy (St-Tx) diagram in UAF can be created to demonstrate the goals and objectives of the mission.

Example Mission Problem or Opportunity: Identifying the DESERT STORM Goals

The UAF Strategic Taxonomy in Figure 9 represents the mission of Operation DESERT STORM and its four primary goals: (1) Iraq's unconditional withdraw from Kuwait, (2) to restore Kuwait's legitimate government, (3) to protect the lives of American citizens abroad, and (4) to promote security and stability of the Persian Gulf.

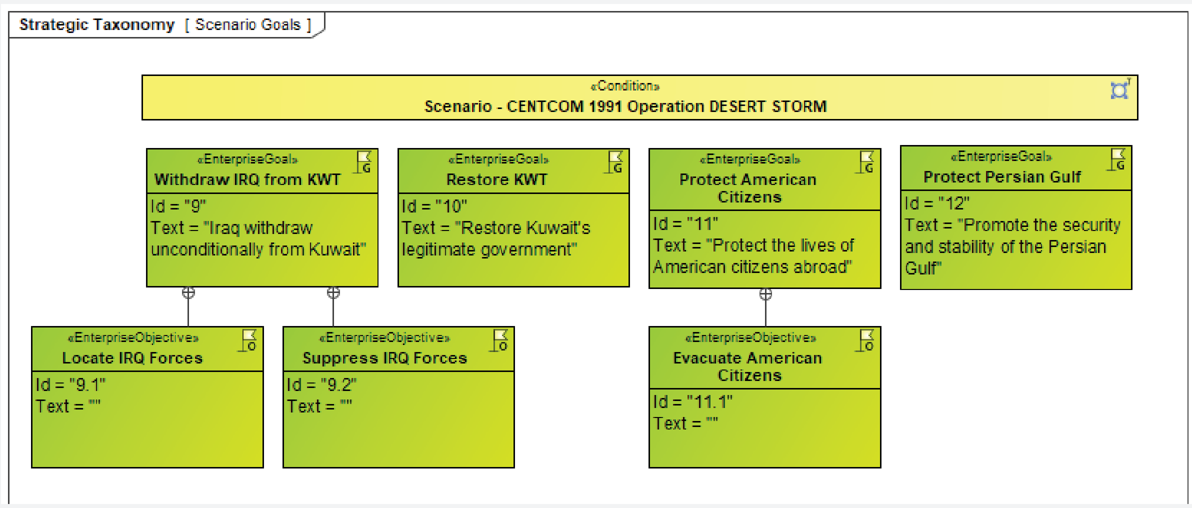

Figure 9. Operation DESERT STORM Goals (UAF Strategic Taxonomy)

Note: This notional example has been further simplified for demonstration purposes. The goals of the logistics and planning phases were not included.

4 Mission Characterization

Prior to planning the MT and MET, the goals of the mission are defined. Mission engineering starts by identifying two foundational elements:

1. What is the mission?
2. What is to be investigated about the mission?

These elements are crucial to scoping the mission engineering activities that follow. From the beginning, it's important to have a clear understanding of what goal or decision will be informed as this will drive subsequent choices throughout the process. Understanding the decisional needs focuses the effort to address the "so what" of the mission engineering investigation.

Refer to Section 4 in the MEG 2.0 for additional details related to mission characterization.

4.1 Develop Mission Context

> Mission characterization is dependent on the scenario, operation, and vignettes.
>
> Documenting the vignette is helpful in identifying MTs and the associated modeling that should be done with that goal in mind.
>
> Cross-collaboration is key to developing a common operating picture and unified understanding of the question that must be answered by all participants.

The mission characterization step focuses on understanding the mission context. The mission context is the background setting, conditions, timeframe, operational strategies, and the objectives of the mission that are specific to the focus of the ME effort and to answering the key questions. The joint conditions library is a starting point for defining conditions; the joint conditions provide common list that can be used to frame how mission architects define physical, military, and civil conditions that can be used to describe the operational context for the selected mission. Various mission characterization-oriented sources include defense planning scenarios, Operation Plans (O-Plans), Concept of Employment (CONEMP), Concept of Operations (CONOPS), and tactical employment guides. While the examples in this style guide are focused on describing mission characterization from the *blue* (USA) perspective, mission characterization is also completed for *green* (coalition) and *red* (adversary) forces. This includes defining the mission context for each of the parties involved in the scenario, operation, and vignette.

First, create an initial view to lay out the breakdown between scenario, operations, and vignettes. The scenario captures the specific description and intent of the mission along with its associated epoch. A scenario can be decomposed into operations and vignettes, smaller subsets of the scenario that are framed to concentrate on a set of events with behaviors, players, and systems. A clear understanding of the breakdown between the scenario, operations, and vignettes allows scoping of the problem set and informs what architectures will be created. Select architectures that support the study and comprehensively answer the study's questions. Note that most source documents will not lay out the scenario, operations, and vignette cleanly enough to model. Instead, architects need to make inferences from source documents and then review with subject matter experts in the relevant mission or domain and update.

[13]

Example Mission Characterization: Operation DESERT STORM Scenario and Vignettes

To liberate Kuwait in Operation DESERT STORM, the Coalition forces needed to achieve air superiority and garner freedom of movement into Baghdad. Coalition forces first needed to eliminate Iraq's robust air defense system, or a portion of it.

Scenario: United States Central Command (CENTCOM) 1991, DESERT STORM

Possible Vignettes:

- Establish Air Superiority
- Destroy Centralized Command and Control (C2)

Figure 10 shows the decomposition of the scenario to vignette, mission, MT, and MET. The "Establish Air Superiority" vignette is modeled in more detail in this guide. The examples are centered on the mission to negate Iraq's early warning capability using the suppression of enemy air defense (SEAD) MT. In this example, the *red* forces conduct an Integrated Air Defense mission with its own capabilities, conditions, measures, and performers. This view shows an example of how to represent the interaction between *red* and *blue* missions.

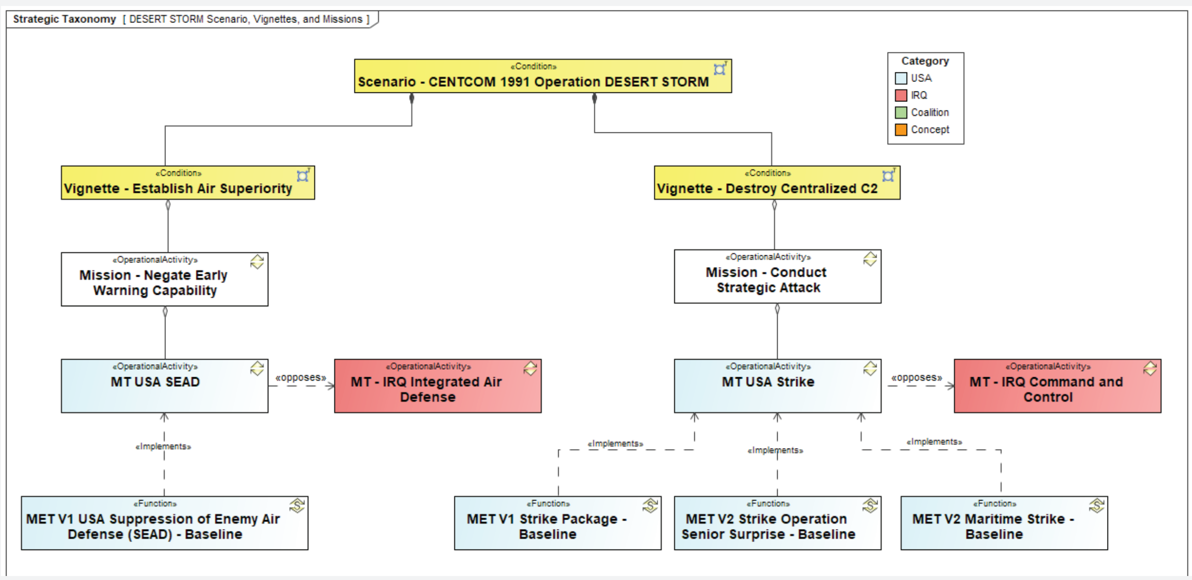

Figure 10. Scenario Breakdown for Operation DESERT STORM (UAF Strategic Taxonomy)

Note: This example has been further simplified in this document for demonstration purposes relevant to Defense Department's oriented customers. The architectures produced can be expanded based on the problem set. Additional METs can be created in support of this mission to represent alternatives and excursion mission approaches.

[13] U.S. Air Force. (2021). Air Force Doctrine Publication 3-70, Strategic Attack. Maxwell AFB. Retrieved from https://www.doctrine.af.mil/Portals/61/documents/AFDP_3-70/3-70-AFDP-STRATEGIC-ATTACK.pdf

U.S. Department of Defense. (2012). Joint Publication 3-01, Countering Air and Missile Threats. Washington, DC: Joint Chiefs of Staff.

4.2 Define Mission Measures and Metrics

A well-defined model will contain quantifiable measurements for success and will be defined prior to the selection of the systems and based on analysis of the situation. The scenario, operations, vignettes, and the mission drive the measures captured in the architecture. MOS quantify the objective to be evaluate and determine whether the mission is a success. The MOS is characterized by one or more MOEs; MOEs provide a means to evaluate the execution of a set of tasks. These mainly help to quantify the mission at the MT level and help answer are we doing the right things or does it require alternative actions. At the MET level, the MOPs quantify the target parameters or performance characteristics of systems or actors used to carry out the mission function or military effect. The MOPs help answer if the tasks were completed to standard.

Measures can be derived from source documents such as the UJTL. Each UJTL task is listed with its associated measures to quantify the effectiveness, and the success of the task. There is a logical decomposition of measures from the MOS at the vignette level down to the MOP at the system level. The MOS for a scenario can be associated to one-to-many MOEs. Along with using the UJTL and JCSFL, service-specific task list tasks have associated MOEs that the service maintains that can be used to provide context when creating joint mission threads and mission engineering threads. MOPs can be derived from tactical UJTL tasks and JCSFL functions as a starting point and the criteria can be added retroactively to architecture through results of constructive simulations, exercises, demonstrations, and experimentation. Operational performance metrics, such as operational effectiveness, suitability, survivability, and lethality (when appropriate), are evaluated for all programs by Operational Test & Evaluation (DOT&E). For a well-formed model, measures are defined for the desired effects, and outcomes are assessed against the allocation of system essential tasks within a MT. Note that the criteria, defined as the minimum acceptable level associated with a particular measure of task success, effectiveness, or performance, are not defined by the UJTLs. The scenario and its conditions will define the criteria.

<u>Operation DESERT STORM Example: MOSs, MOEs, MOPs</u>

For the Operation DESERT STORM SEAD mission, the objective of Task Force Normandy was to blind a portion of the Iraqi air defense system.

Figure 11 shows notional examples for the MOS, MOE, and MOS for Operation DESERT STORM and the establish air superiority vignette.

#	Property Set	Measurements
1	Measures of Success (MOS)	IRQ Presence in KWT : Integer Attrition : Percent
2	Measures of Effectiveness (MOE)	Air Defenses Destroyed : Percent Aircraft Lost : Integer
3	Measures of Performance (MOP)	Weapons on Target : Integer Aircraft Survivability : Integer

Figure 11. MOS, MOE, MOP for DESERT STORM (UAF Typical Measurements Matrix)

5 Mission Architectures

With a thorough understanding of the problem set and the questions that need answered, the next step is to develop the mission architectures. This process starts with the scenario and vignette as inputs and includes the following steps:

1. Perform information collection to understand the mission context and measures and the needed capabilities. A full understanding of the MOS and MOE enables accurate assessment and quantification for the vignettes against the desired end state.
2. Utilize existing MTs; if none exist, then develop new associated MTs by analyzing the desired effect.
3. Capture all relevant mission elements for the United States, allied combatants, and adversary forces.
4. Create METs by allocating the mission elements to the activities (tasks) in the MTs.
5. Represent the relevant communication, information, and information flow between mission elements as part of the architecture.

The first pass through these steps will yield an initial architecture. The baseline mission approach architecture represents the agreed upon starting point for how the mission will be executed to address the mission engineering effort. This will serve as the basis for comparison with any changes, additions, or deletions made to the architecture for each of the concepts. An alternative architecture represents a change to the baseline mission approach for how the mission will be executed.

Example Baselines, Alternatives, and Excursions:

DESERT STORM, Task Force Normandy

In the Task Force Normandy example, the **baseline mission approach** represents completing the mission assuming all mission elements are on station as fragged with clear visibility and weather factors.

The **alternative mission approach** represents introducing a new concept, the MC-130E, to perform communication relay assuming clear visibility and no weather factors.

In the **excursion mission approach**, the Task Force Normandy mission is executed during a sandstorm (varying conditions). The sandstorm will reduce visibility, which introduces a new risk to the mission success. The excursion architecture will represent a different execution approach than the baseline in which MH-53 helicopters drop light sticks to mark calibration points for the AH-64 helicopters.

An **alternative mission approach with an excursion** introduces a new concept, the MC-130E, to perform communications relay during a sandstorm.

Note: While the MC-130E is a real plane that employed for DESERT STORM and that retired in 2013, it is being represented as an existing airframe with a newly equipped technology Concept for demonstration purposes.

The baseline and alternative architectures serve as precise descriptions of what the study intends to analyze. Each alternative architecture is a variation on the baseline architecture. The starting set of alternatives are defined for each mission engineering study; however, the alternatives may evolve as the study progresses. Unlike alternatives, excursions are modelled to represent changes in the assumptions, mission elements, and behaviors in the baseline. It is important to note the difference between alternatives and excursions – alternatives are the introduction of new technology or concepts to a baseline architecture whereas excursions are a change to the mission context, mission elements, behaviors, and related assumptions made in the baseline architecture (Figure 12). Examples of excursions include switching out existing technology in the baseline architecture, changing the location of the assets, or changes to the countries/players involved in the scenario.

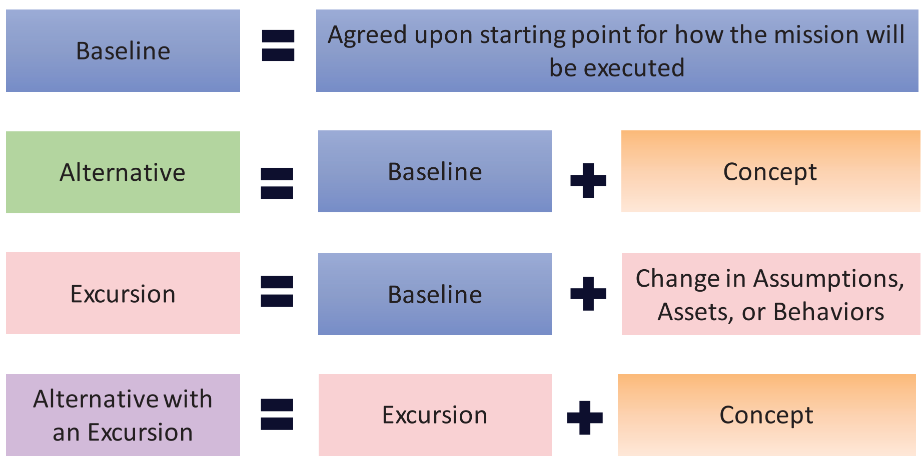

Figure 12. Explanation of Baselines, Alternatives, and Excursions

While there are usually differences in the METs architecture among the baseline, excursion, and alternative cases with any new or different way to execute the mission (e.g., a new doctrine or a new technology enables a new way to do things), the MTs should be abstracted to contain the same effects and measures. Please refer to the MEG 2.0, Section 5, for additional details related to mission architectures.

5.1 Developing Mission Threads

Develop MTs based on the desired effect of the scenario, operation, vignette, and mission. The DoD defines a mission as "the task, together with the purpose, that clearly indicates the action to be taken and the reason therefore"[14]; consequently, missions will be associated with key tasks. The goal is to develop one MT for each of the missions identified.

Keep in mind each MT is designed to be solution agnostic. When available and appropriate, doctrine-based threads and resources such as the UJTL, JCSFL, service-specific task lists, and the Joint Mission Essential Task List (JMETL) provide the preferred starting point for a MT.

Organizing tasks into a sequential description of the mission approach is useful. This exemplar uses *Find-Fix-Track-Target-Engage-Assess* (*F2T2EA*) kill chain to provide a consistent presentation format within OUSD(R&E) MI. Alternatively, other supporting missions may take a different construct so use an appropriate kinetic or non-kinetic effects chain, such as the *Detect to Engage* sequence from Air and Missile Defense, to frame the MT or MET. This guidance applies for kinetic, non-kinetic, cyber, and logistical applications.

MTs are modeled from an operational perspective. Operational performers can be used to abstract solutions to a functional-logical category. Introducing operational performers is consistent with the UAF perspective, while remaining agnostic to any specific mission element.

[14] U.S. Department of Defense. (2019). Joint Publication 3-0, Joint Operations. Washington, DC: Joint Chiefs of Staff.

Example Mission Thread: Operation DESERT STORM, Task Force Normandy

The objective or desired effect of the Task Force Normandy was to negate the early warning capability of Iraqi air defense system to create an undefended corridor for air attack. To meet that objective, the coalition forces conducted a suppression of enemy air defense (SEAD) mission against Iraq air defenses. A top-level SEAD MT can be created using a UAFML Operational Process flow (Figure 13).

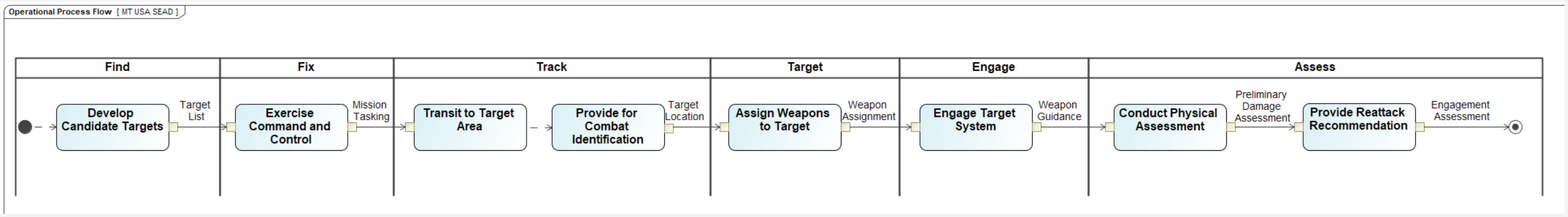

Figure 13. Suppression of Enemy Air Defense Mission Thread (UAF Operational Process Flow)

The tasks in Figure 13 were derived using various source documents, such as JP 3-01, the UJTL, and JCSFL. To ensure common lexicon, the MT tasks were compared and traced to the UJTL tasks (Figure 14). Note that the UJTL alone does not always contain all the tasks that can be associated to the activities performed in a MT. For additional details, the JCSFL and any domain or service specific task lists should be used as references for common language.

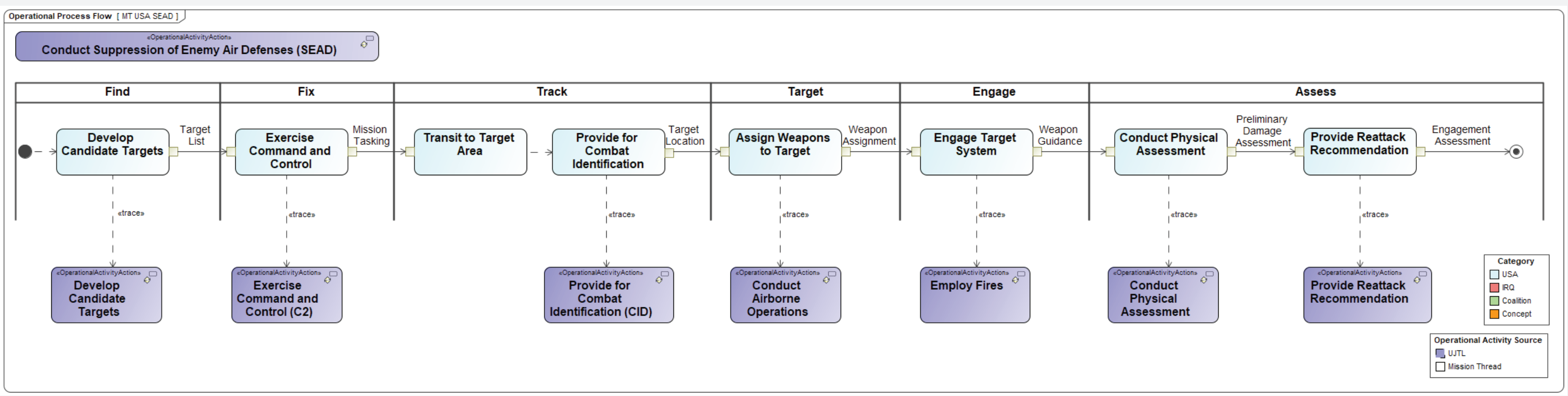

Figure 14. Suppression of Enemy Air Defense Mission Thread with UJTL Traceability

The UJTL is a list of joint military tasks at the strategic, operational, and tactical level maintained by the Joint Staff. Each UJTL task is listed with its associated measures to quantify the effectiveness, and/or success of the task. Note that the criteria or threshold values for these measures are not specified in the UJTL and must be determined the scenario and its specified conditions. The JCSFL is a common lexicon of system functions maintained by the Joint Staff. The JCSFL functions are used to describe and assess the capabilities and functions of systems and systems of systems (SoS) in the DoD across different domains and disciplines. The JCSFL is composed into functional categories, such as command and control, communications, and intelligence, surveillance, and reconnaissance (ISR). The functions in the JCSFL should be used as a starting point, in conjunction with other sources, to define the activities in the METs.

The UAF profile in Cameo contains an element library consisting of the unclassified operational activities (tasks) from the UJTL. However, please note that the UJTL library in the Cameo modeling tool is unclassified and may not reflect the latest updates. Architects should verify tasks against the authoritative UJTL library[15] and review higher classification UJTL libraries as necessary. MT steps may be a decomposition, generalization, or other appropriate relationship to one or more UJTL tasks. By mapping MT steps to UJTL tasks, architects can establish a direct link back to the doctrine and enhance the model's pedigree (Figure 15).

Example MT to UJTL Mapping: Operation DESERT STORM, Task Force Normandy

The individual steps shown in the SEAD mission thread (Figure 14) are mapped to the UJTLs in Figure 15 using a UAF Operational Processes (Op-Pr) diagram to show traceability back to doctrine within the model.

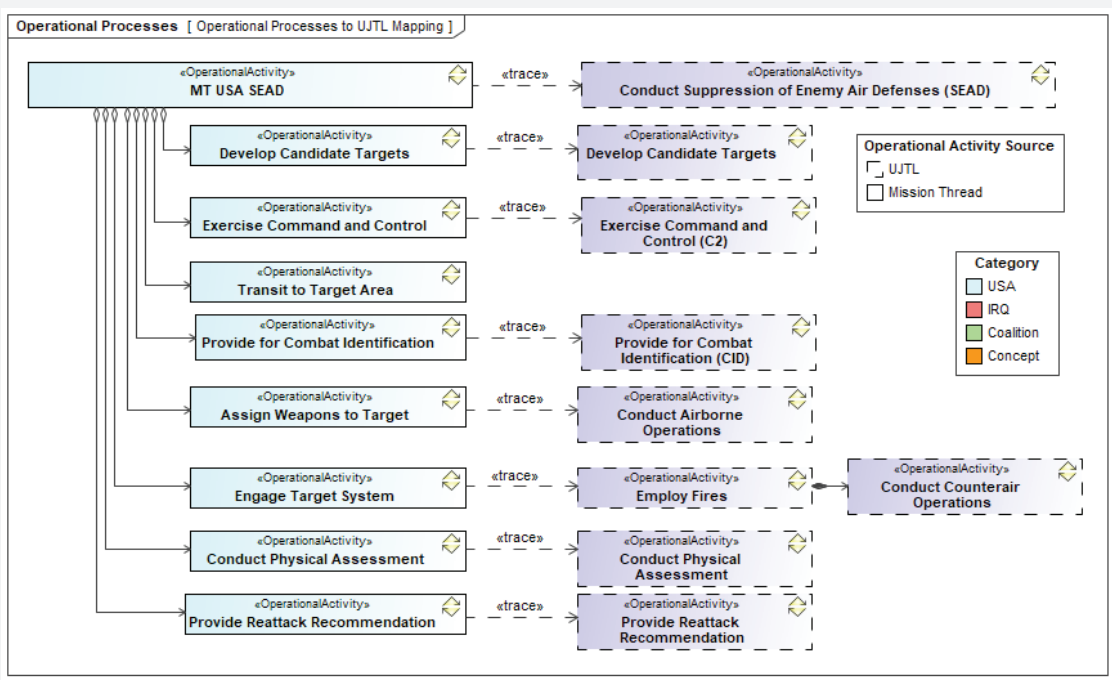

Figure 15. Mapping between MT Tasks and UJTL Tasks (UAF Operational Processes)

[15] Joint Chiefs of Staff. (2023, October 6). UJTL Task Development Tool (UTDT). Retrieved from https://utdt.js.mil/home

Key takeaways for MTs:

> ***How are Mission Threads used?***
>
> Mission Threads are based on doctrine and provide the foundation for all Mission Engineering Threads. Mission Threads help ensure that processes are consistent and correct.

Existent MTs may be used when appropriate. If needed, apply updates and modifications to tailor the fit to the current vignette.

MTs are solution-agnostic; they are distinct in that they describe the task execution sequence in a chain of events, not how or by whom each activity within the flow is to be accomplished.

MTs are focused on the needed information flows. Initial architecture work includes researching source documents and translating those authorized sources into modeling language elements. Where those sources are incomplete or ambiguous, consult and verify the information within architectures with subject matter experts (SMEs) and user communities. In practice, recognize that operational tactics may expand on areas described in doctrine.

Since top-level MTs are solution agnostic, it is wise to drill drown to sufficiently handle the fidelity required of the doctrine, Tactics, Techniques, and Procedures (TTPs), rules of engagement, UJTL tasks, and JCSFL functions. Further decomposition risks premature design constraints. Additional detail can be added to meet stakeholder requests, given that it is properly sourced.

As the MTs are developed, additional issues, capability gaps, and related information may be identified in the scenario, operations, and vignettes. Addressing these issues will require the clarification and update of those artifacts. An unresolved issue may serve as an indicator that a system may have a gap or vulnerability that needs to be identified.

For each vignette and mission, one to many MTs may be identified. MTs are created for *blue forces* based on Joint and Service-specific doctrine. *Green force* missions can be represented within a *blue* MT if they are working directly with the United States or can be represented separately based on doctrine, etc., if they are completing a separate mission in support of the same overall goal. *Red force* missions and MTs are generally based on information provided by the intelligence community.

For complex scenarios, refrain from using complicated diagrams that describe everything. This makes it difficult for subject matter experts to review. Decompose and nest diagrams as necessary by drilling into detail in additional views. When creating a MT, map out the nominal flows; this should consider known/realistic threats and conditions. Notable departures from the nominal flows and assumptions should be captured as notes or comments so that they can be considered for later analysis. Upon further analysis, this can result in the development of a unique MT.

The MTs represent specific mission execution within a vignette, but they remain system agnostic. The flow logic in the MT should be as complete as possible, but it still may not meet the requirements for model executability.

5.1.1 MTs using UAF Operational Process Flow in a Digital Engineering Tool

To develop a MT with UAFML, utilize an Operational Process Flow diagram. With SysML, utilize an Activity diagram.

- Create an **Operational Process Flow** diagram.
- **Vertical Swimlanes:** Vertical swimlanes are used to organize tasks into sequential phases. Use an appropriate effects chain relevant to the mission being conducted to frame the MT.
- **Horizontal Swimlanes:** Horizontal swimlanes can be used for allocating activities (tasks) to operational performers in the MTs, remaining agnostic to any specific mission element. If it is necessary to denote the type of asset which will conduct an activity, horizontal swimlanes at an abstracted level can be used. Swimlanes may be used for display purposes to organize the complex flows into sections.
- **Initial Node:** Represents the Start and is labelled with starting conditions of the MT.
- **Operational Activity Action:** Represents Activities (tasks) in the flow. Creating Operational Activity actions provides the ability to show traceability between the different mission engineering viewpoints.
- **Output Pin:** Each Operational Activity Action should have an output pin(s) that is assigned a "Type". These pins are used to show the operational information flow (e.g., key information, resource, etc.) being transferred between activities (tasks) in the MT. The "Type" for each Output pin should be shown.
    - Each pin is assigned **Operational Information** for the "Type" and is labelled with the information that is flowing between activities (tasks) (e.g., typically a noun or noun-state pairs). In UAF, Operational Information is used to show information that can be exchanged via operational activities (tasks) performed.
- **Input Pin:** Like the output pin(s), the input pin(s) also show the Operational Information flow between the activities in the MT. The Input pin(s) should have the same type as the preceding output pin(s). The "Type" and "Name" for each Input pin(s) should be hidden.
- **Operational Object Flow:** Connections between Operational Activity actions representing the information being produced or transmitted by the source activity and being consumed by the target activity. An object flow represents information being transmitted, whereas a control flow determines the order of operations for activities.
- **Activity Parameter Node or Activity Final Node:** Represents the End and shows the output/termination of the process.
- Use a **Legend** across all diagrams to visually indicate the country of ownership of the mission elements, activities (tasks), or functions. Based on the digital engineering tool, the legend can be applied manually or by setting parameters that allow for automatic legend item assignment.

Refer to Appendix 9.1 of this MASG 1.0 for more detailed guidance related to naming conventions, diagram formatting, standardized colors, and legends in a digital engineering tool like Cameo System Modeler.

5.2 Understanding the Order of Battle

The Order of Battle is defined as the "identification, strength, command structure, and disposition of the personnel, units, and equipment of any military force"[16].The Order of Battle (OB) architecture shows the entire line-up of mission elements involved in a scenario and is modeled utilizing the Resource Structure (Rs-Sr) diagram. This view calls out ownership of known mission elements and actors by country. This architecture will be used to ensure that all mission elements are accounted for in the METs and E2E views. In addition to allowing an easy view of all mission elements, it also serves as a checklist to ensure all mission elements are represented in the views and model overall. These mission elements will appear on the METs (UAFML Resources Process Flow) and the E2E Views (UAFML Resources Internal Connectivity).

[16] U.S. Department of Defense. (2022). Joint Publication 2-0, Intelligence. Washington, DC: Joint Chiefs of Staff.

Example Order of Battle: DESERT STORM, focus on Task Force Normandy

Figure 16 is a UAFML Resource Structure (Rs-Sr) for Operation DESERT STORM. This view shows the mission elements involved in the identified vignettes of DESERT STORM for each player category. For the United States (*blue*), this includes a variety of helicopters, weapons, command and control and the Air Operations Center. For this notional example, early warning radar sites are represented on the Iraq side in *red*. Coalition forces were not involved in Task Force Normandy but were a component of the greater operation and are represented in *green*. Utilize orange-gold to identify test concepts in the Order of Battle for ready identification. This example shows one way of articulating that the platform (AH-64) is equipped with weapons (30 mm Cannon Fire, AGM-114 Hellfire missiles, and Hydra-70), but there are other ways this relationship can be presented.

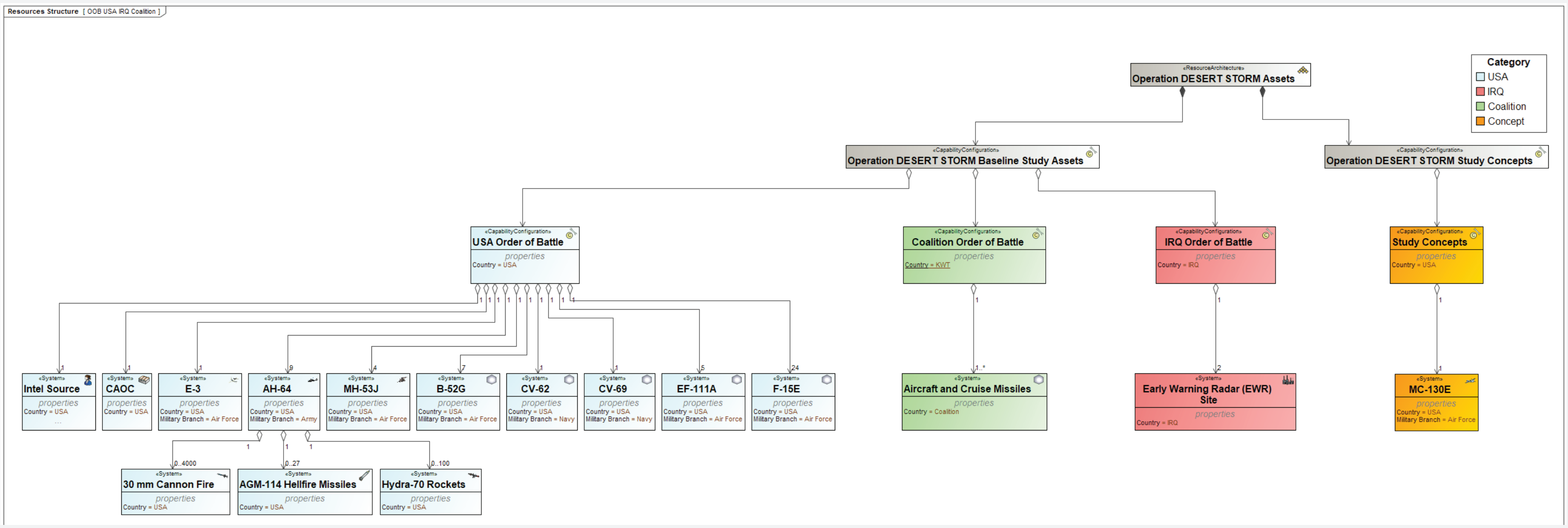

Figure 16. DESERT STORM Order of Battle Architecture (UAF Resources Structure)

5.2.1 Order of Battle using UAF Resources Structure in a Digital Engineering Tool

To develop an OB with UAFML, utilize a Resource Structure diagram as shown in Figure 15. With SysML, utilize a Block Definition Diagram (BDD) diagram and represent each mission element as a SysML block.

- Create a **Resources Structure** diagram.
- **Resource Architecture:** Represents the overall of the project or study. The Resource Architecture is used to represent a collection of resources against which the steps of the MET will be allocated.
- **Capability Configuration:** In the above example, capability configurations are used to separate the Order of Battle based on country of ownership or category. Capability configurations can also be used to represent the vignette order of battle, mission order of battle, and be used to show specific configurations or variants of a system. A separate capability configuration should also be created to define which mission elements are part of the baseline or alternative set of performers within the scenario.
- **System:** Defined as an integrated set of elements, subsystems, or assemblies that accomplish a defined objective. Systems are used to represent the mission elements in the Order of Battle. Each system or sub-system should be connected to one or more Capability Configurations.
    - Occasionally, more than one country uses the same type of mission element. In those cases, a separate Mission Element should be created for each country.
    - Systems should come from common system library that can be used for other architectures in the study.
    - Individual properties and parameters for each mission element needs to include the country of ownership. Architects must ensure that the information captured serves the purposes of the mission model and analysis, while striking a balance between the time allotted to complete the descriptive model and capturing all possible information required to define a system. Additional detail not shown in the architectures can be annotated through comments that highlight any critical gaps in the architecture so they can be reflected in derived simulations.
- **Directed Composition:** Connections between the Resource Architecture and each Capability Configuration are represented as Compositions because these orders of battles exist in the context of this project or study.
- **Directed Aggregation:** Connections between the Capability Configuration and the Systems are represented as aggregations because each of the mission elements can exist independently without being included as part of the order of battle.
- Use a **Legend** across all diagrams to visually indicate the country of ownership of the mission elements, activities, or functions. Based on the digital engineering tool, the legend can be applied manually or by setting parameters that allow for automatic legend item assignment.

Refer to Appendix 9.1 of this MASG 1.0 for more detailed guidance related to naming conventions, diagram formatting, standardized colors, and legends in a digital engineering tool like Cameo System Modeler.

5.3 Developing Mission Engineering Threads

METs are the implementation specific versions of MTs. To develop METs, take each MT from the previous process step and allocate each system, organization, or actor that performs it. The nominal process for doing so is:

1. Identify the mission elements (systems or actors) that perform each activity (task) in the MT.
2. Create swimlanes for each mission element that performs the activities.
3. Move the activity into the swimlane of the system that performs it. Review the JCSFL (maintained by Joint Staff/J6/AID), operational UJTLs, and tactical UJTLs as reference points for activities in the MET; they contain a standard lexicon to define the system functions. These resources are starting points that should be tailored to ensure intent of MET is clear to provide more detail for the activities being completed.
4. Identify required interfaces among mission elements by noting flows across the swimlane boundaries.

When developing the MET, the above process will not be sequential; it will require iteration back and forth between the MT, OB, and MET.

Example Baseline MET: Operation DESERT STORM, Task Force Normandy SEAD mission

The baseline MET below (Figure 17) depicts how the United States intends to counter the Iraqi early warning radar sites in a targeted area. Note the activities from the MT are allocated to the mission elements identified in the baseline. From the OB, there are three different weapons employed by the AH-64: AGM-114 Hellfire missiles, 30mm Cannon Fire, and Hydra-70 Rockets. In the MET, these are abstracted and represented in a single swimlane called "Weapons". If the initial engagement is not successful, the AH-64 would re-engage the target and go through the process until the target is destroyed.

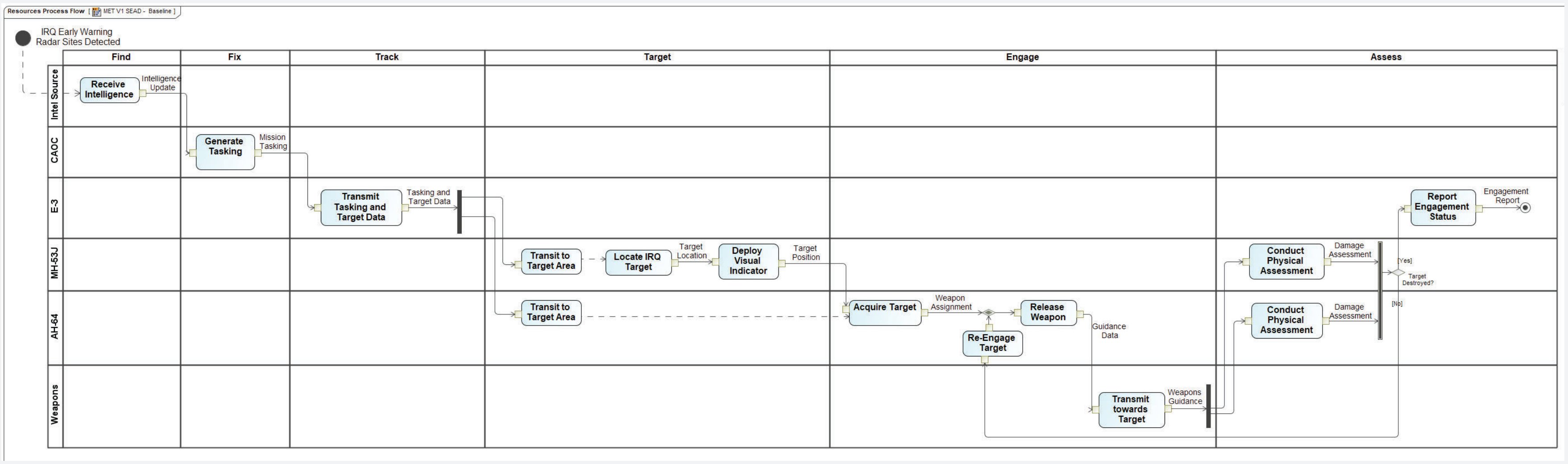

Figure 17. Baseline Operation "DESERT STORM, Task Force Normandy SEAD" Mission Engineering Thread (UAF Resources Process Flow)

Repeat this process for all MTs in the baseline; alternative METs will follow a similar process but are built after the entire baseline architecture is complete. Alternative METs are typically variants on the baseline METs as opposed to completely new threads.

Example Alternative MET: Operation DESERT STORM, Task Force Normandy SEAD mission

The alternative MET below (Figure 18) shows the insertion of the MC-130 concept into the baseline MET to perform communications relay.

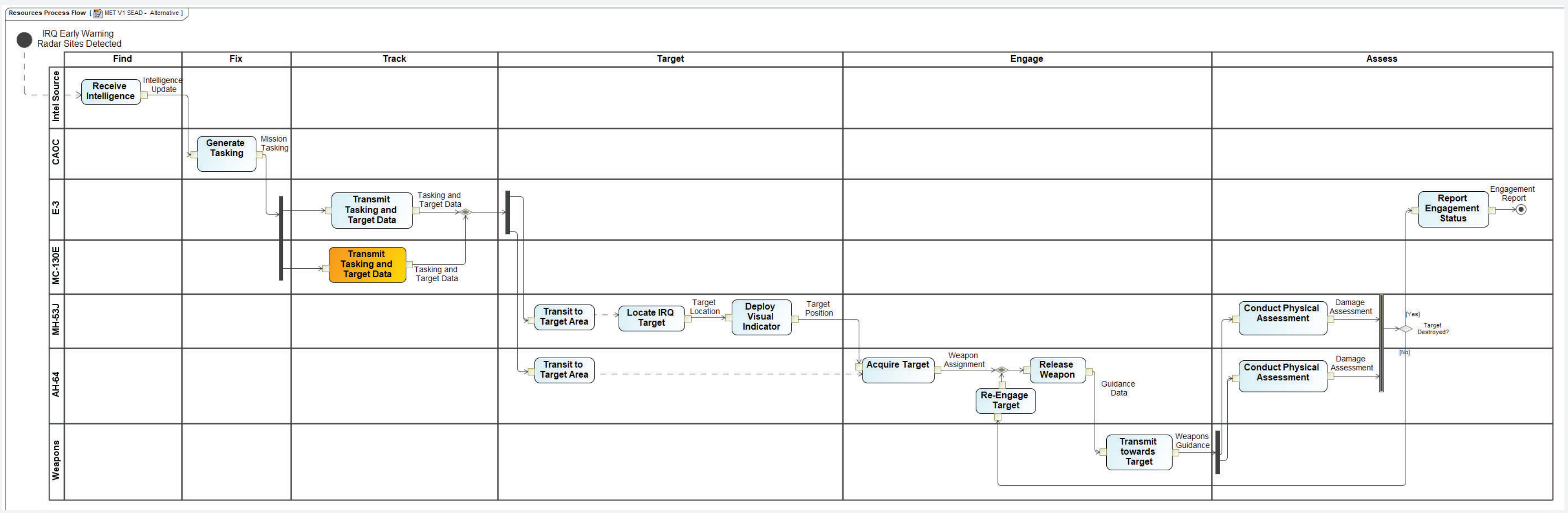

Figure 18. Alternative Operation "DESERT STORM, Task Force Normandy" Mission Engineering Thread (UAF Resources Process Flow)

A mapping between functions and tasks can be done to show which MET functions implement the MT operational activities. This mapping is essential to ensure a fully implemented battle plan. All steps of the MT should be mapped to activities in the MET, so it can be confirmed that the battle plan follows what is defined in doctrine and the UJTL tasks. The conditions of the scenario will vary the measure criteria for the MT and METs; different mission elements may accomplish a MT task in a different way. Conditions drive how many variations of the MET exist.

Example MET to MT Mapping: Task Force Normandy SEAD Mission

A resource function to operational activity map (Figure 19) shows the relationship between the SEAD MT activities and the MET functions. Each MET function implements a MT task.

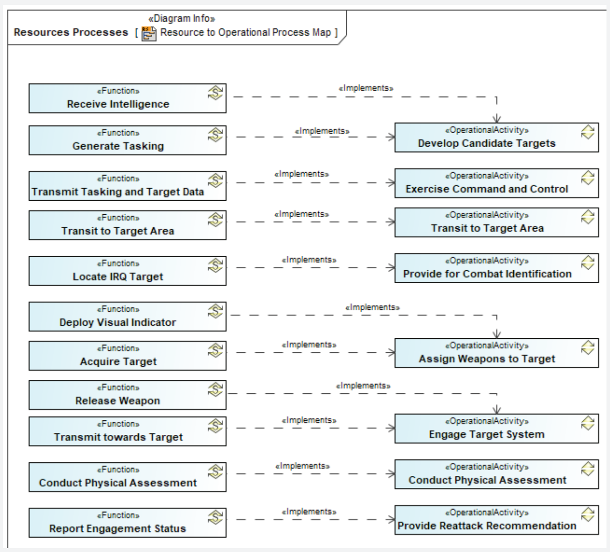

Figure 19. MET Function to MT Task Mapping (UAF Resources Processes)

This mapping can also be displayed in a matrix view created in the digital engineering tool. This provides a different perspective into the same information (Figure 20).

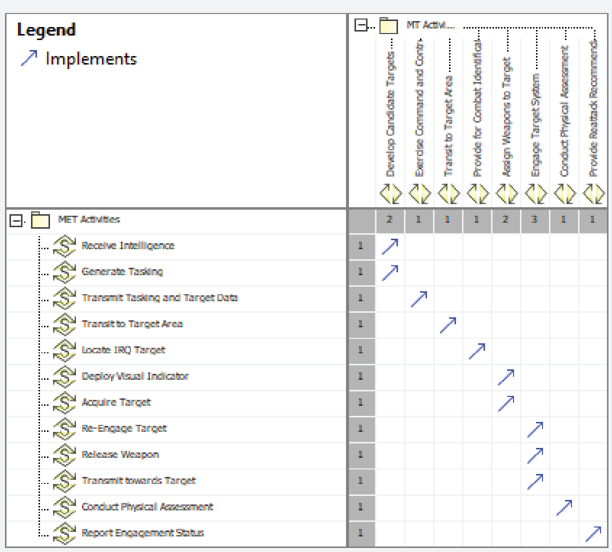

Figure 20. MET Functions to MT Task Matrix

Key takeaways for MET development:

By combining the mission elements shown in the OB and the activities (tasks) in the MT, the MET provides a solution specific view of the activities and mission elements involved. To construct the MET, start with the corresponding MT; the MT activity diagram and all associated activities are a key reference when creating METs. Existing architecture work, such as other implementations of the MT, is a good initial starting source but will need to be validated with leadership for the current application.

A key benefit of developing METs is that it drives the information collection process. Gaps, redundancies, and anomalous activities may be identified that were not considered in source plans. The activities and flows in the MET account for real-world implementation needs, while using the flows in the associated MT as the foundation. Certain portions of the MET will be more relevant than others. Drill down as needed to lower levels of detail in key areas. The level of detail that must be depicted in the architecture depends on the scenario, operation, vignette, and mission being represented and the supporting doctrine. The execution of a mission in a specific scenario may include the integration of multiple MTs and METs as *effects webs or kill webs*.

Based on the vignette or mission, one to many METs can be created for each MT. The mission elements allocated in the MET may differ based on the conditions. The conditions drive how many variations of the MET exist. METs are created for *blue forces* to represent the baseline, any excursions, or alternatives. *Green force* missions can be represented within a *blue* MET if they are working directly with the United States or can be represented separately if the information is available. *Red force* METs are generally based on information provided by the intelligence community.

The level of detail provided in a MET should be tailored to the purpose statement. Architects may make some assumptions regarding activities and events that are not central to the investigative questions—*these assumptions must be explicitly documented within the architecture* via *comments or in accompanying model documentation*. As with MTs, architects should validate METs with stakeholders and SMEs.[17]

For complex scenarios, refrain from using complicated diagrams that describe everything. This makes it difficult for SMEs to review. Decompose and nest diagrams as necessary. Include logic for important departures from the nominal flow that need to be considered in the subsequent analysis. The join, merge, and decision nodes can be used to represent activities occurring at the same time as well as alternatives in a process.

One goal of MET development is to identify specific interfaces between systems and what flows between the systems. For each identified system or actor, collect authoritative information sources to appropriately identify relevant sub-systems, capabilities, and key performance characteristics.

[17] Department of Defense OUSD(R&E). (2023). Department of Defense Mission Engineering Guide 2.0. Washington, DC. Retrieved from https://ac.cto.mil/wp-content/uploads/2023/11/MEG_2_Oct2023.pdf

5.3.1 METs using UAF Resources Process Flow in a Digital Engineering Tool

To develop a MET with UAFML, utilize a Resources Process diagram. Figure 16 shows a MET that was created using the Cameo System Modeler "Resources Process Flow" view. With SysML, utilize an Activity diagram. Define systems/mission elements using Blocks.

- Create a **Resources Process Flow** diagram.
- **Multidimensional Swimlanes:** Utilized to visually divide up large processes with phases or stages in vertical columns and mission elements in horizontal swimlanes.
  - **Vertical Swimlanes:** Represent the selected doctrine-based thread (e.g., *F2T2EA, Exploitation-Installation-Command and Control (C2)-Action on Objective*) as the labels in the vertical swimlanes.
  - **Horizontal Swimlanes:** Use horizontal swimlanes for allocating activities to mission elements (resources, systems, or actors). Swimlanes represent mission elements pulled from the list of mission elements (**Resource Artifact/System**) shown in the Order of Battle.
- **Initial Node:** Represents the initial item in the model and is labelled with starting conditions of the MET. The starting conditions for MTs and METs are scenario and mission dependent.
- **Function Action:** Represents activities in the flow and ensures that every call behavior in the diagram refers to a defined activity. This function action can be reused in different views to show connections and allocations in the model.
- **Function Object Flow:** Connections between Functions over which information flows. At the MET level, object flows are used more frequently than with the MT. One goal of MET development is to identify specific interfaces and what flows between them.
- **Output Pin:** Each Activity should have an output pin(s) that is assigned a "Type". These pins are used to show the object flow (e.g., key information, resource, etc.) being transferred between activities in the MET. The **Type** for each Output pin(s) should be shown.
  - Each Pin is assigned a Type. The types are created as **Resource Information** and labelled with the information that is flowing between activities (e.g., typically a noun or noun-state pair). With modularization, a library of common Resource Information will be available to be assigned as the Type.
- **Input Pin:** Like the output pin(s), the input pin(s) also shows the object flow between the activities in the MET. The input pin(s) **Type** and **Name** should be hidden to minimize clutter. Each output pin must terminate at a matching input pin; The input pin(s) should have the same type as the preceding output pin(s).
- **Activity Parameter Node or Activity Final Node:** Represents the end of the MET flow.
- Use a **Legend** across all diagrams to visually indicate the country of ownership of either mission elements or activities/functions.

Refer to Appendix 9.1 of this MASG 1.0 for more detailed guidance related to naming conventions, diagram formatting, standardized colors, and legends in a digital engineering tool like Cameo System Modeler.

5.4 Developing Mission Engineering Threads End-to-End View

Mission engineering thread end-to-end (E2E) views show the connections and the information exchange between mission elements. The E2E view (UAF Resources Internal Connectivity) can be used to identify potential gaps or vulnerabilities that exist within the selected employment depicted in the METs. For instance, communications architecture development often requires technical documentation and subject matter consultation. It may be discovered that two systems do not talk, identifying these types of gaps are relevant in both the modeling process and the actual mission employment process. This aspect of architecture development has the advantage of propelling information collection early in the mission engineering study through a low fidelity first path assessment, rather than discovering late in a study that it is not possible to close a MET.

Key takeaways for E2E development:

Align *blue* and *green* mission elements to the left with all *red* mission elements on the right to show the targets. Reverse this guidance when depicting *red* force models. This flow aligns with the readers natural left-to-right reading style. From left to right, arrange mission elements roughly by order in which they perform their functions. Typically, the order runs from sensors through command and control to weapon platforms and weapons/munitions.

Group mission elements by mission or function. Examples of groups: Operations Centers, Communications Relay, Battle Management Command and Control (BMC2), Interdiction, Offensive Counter-Air, Anti-Surface Warfare, Space-based ISR, etc.

Example Architect View of Mission Engineering Thread End-to-End (E2E) View: DESERT STORM, Task Force Normandy

Figure 21 depicts an architect's end-to-end (E2E) view for Operation DESERT STORM, Task Force Normandy. This is a working view that the architecture team uses to ensure that mission elements that are exchanging data have the appropriate port types/mechanisms (e.g., Link-16, Fiber, UHF Radio, etc.) to communicate. This "working" view is used to aid modelers; it is not intended for diagrams to be viewed outside of the modeling environment.

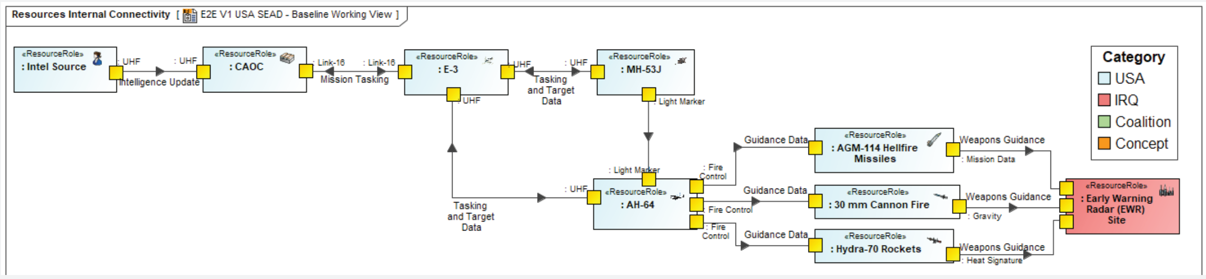

Figure 21. Working view of End-to-End (E2E) View for DESERT STORM, Task Force Normandy

The E2E view is the view primarily used when presenting to OUSD(R&E) leadership and stakeholders. To provide a consistent presentation format, show images of mission elements and group them by mission or function in the E2E view.

Example Mission Engineering Thread End-to-End (E2E) View: DESERT STORM, Task Force Normandy

Figure 22 shows an E2E view for Task Force Normandy. This logical view shows all the mission elements involved in the MET and the information flow between them at the highest level. The mission elements are grouped by mission or function. For example, the MH-53 and AH-64 are helicopters, but in the context of this vignette and MET their mission is strike.

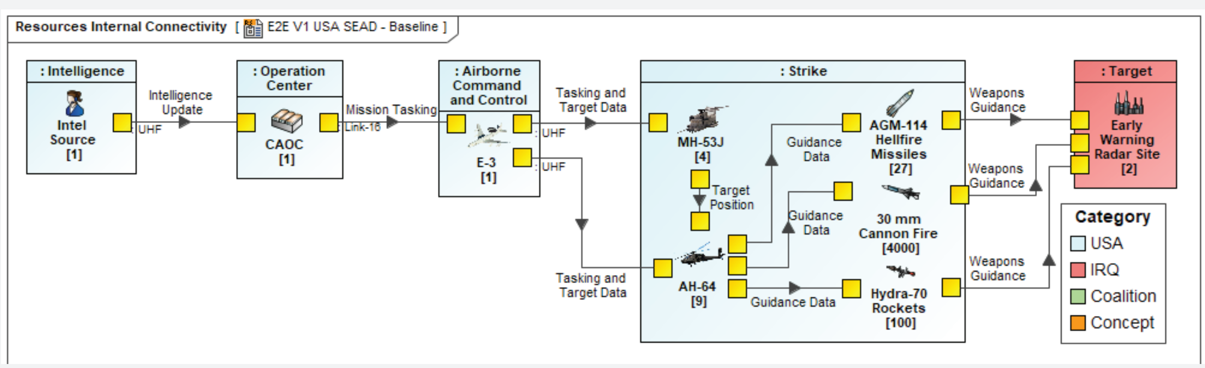

Figure 22. End-to-End (E2E) for DESERT STORM, Task Force Normandy

For this mission, it is assumed that the E-3 represents the essential C2 relay node such that its absence illustrates a potential gap, i.e., a single point of failure for the required communications relay. This example is a creative divergence from historical data to demonstrate a concept. These views identify the strengths and weaknesses of the mission.

5.4.1 E2E View using UAF Resources Internal Connectivity in a Digital Engineering Tool

To develop a E2E with UAFML, utilize a Resources Connectivity diagram. Cameo System Modeler has a viewpoint called "Resources Internal Connectivity" that was used in creating the example in Figure 21. With SysML, utilize a Block Definition Diagram. Define each SoS architecture using the SysML block defined in the OB. Define an Internal Block Diagram (IBD) for each SoS architecture. Define Interfaces using proxy ports typed by interface blocks.

- Create a **Resources Internal Connectivity** diagram.
- **System/Resource Role:** Represents mission elements in the MET. Mission Elements are represented using "Resource Role", which should match the mission elements used in the OB and MET. Elements should have their Stereotype label and icon hidden.
- **Resource Architecture:** For E2E views, a Resource Architecture can be used to group mission elements (systems) that collectively perform a capability or function. The Resource Architecture denote a model of the architecture and can be built of systems or capability configurations.
- **Resource Port:** To create a connection between mission elements, each **Resource Role** should have the **Resource Port** that is assigned a **Type**.
  - The port is typed by a **Resource Interface** that shows the transmission medium used (e.g., SATCOM, Fiber, Link-16, UHF Radio, VHF Radio, etc.). If a mission element has multiple possible transmission mechanisms, a new port should be created for each one.
  - Each **Resource Port** should also be assigned a **Name** that represents a generic category for the type of information transmission mechanism (e.g., Voice, Data Link). The port **Name** is not shown once the diagrams are done as they clog up the view, but they are useful when creating diagrams.
  - For the architectures shown in this guide, the ports between the weapon platforms and the weapons are assigned "Command" as the Name and "Fire Control" as the Type. The ports between a *blue*/*green* mission element (weapon) and a *red* mission element (Target) are assigned "Guidance" as the Name and "Gravity" or "Heat Signature" for the Type.
  - Create a Resource Connection in the E2E between systems/actors (resource ports) for which the flows between functions cross mission element horizontal swimlanes in the MET to indicate an interface between those two systems/actors.
  - When creating connections in the E2E diagram, connections should only be created at the mission element-to-mission element level. While the mission elements are categorized into abstracted mission elements groups, this is mainly for visualization purposes.
- To show relationship between nodes, draw a **Connection** between them**.** Create a **Resource Exchange** and add **Conveyed Items** as typed by **Resource Information.**
  - If there is a two-way connection between mission elements, do not draw a new line between the mission elements.

    - Display the name of the information being conveyed along with directional arrows (realized **Resource Exchanges**) both within and between systems in mission element groups.
    - A Port or Resource Exchange should exist on all mission elements including *blue*, *green*, and *red* mission elements.
- A **Legend** should be used across all diagrams to visually indicate the country of ownership of either mission elements or activities/functions.
- To assist audiences in reading the architectures, images of each mission element or a generic picture may be used to provide easy identification of what comprises each mission element. An **Image** can be added in the Specification window. To show these in the model, in Shape Properties select "Shape Image" as the option for Show Stereotype.

Refer to Appendix 9.1 of this MASG 1.0 for more detailed guidance related to naming conventions, diagram formatting, standardized colors, and legends in a digital engineering tool like Cameo System Modeler.

6 Mission Engineering Analysis

There is a two-way relationship between Mission Engineering Architecture and Mission Engineering Analysis. The core function of the mission engineering analysis is to evaluate mission architectures within the specific scenario-based mission context. In principle, one architecture can be defined for each major variant considered in the study experimental design. The purpose of the architecture is to precisely define what is to be analyzed in each scenario and ensure agreement amongst all stakeholders. Discoveries made during this mission engineering analysis step may result in changes to the study design and a need to update the mission engineering architecture. It can also be helpful to create specialized versions of architectures that show key assumptions and changes made to accommodate the analysis. Any deviations to the views should be shown along with the rationale.

Mission Engineering Architecture is the descriptive modeling foundation for constructive modeling and simulation. Mission Engineering Analysis may be performed through one or more simulation tools, experimentation, or other analysis techniques like Monte Carlo simulation or nodal analysis. Note that most analysis techniques, simulation or otherwise, require a model to conform with its own methodological assumptions.

Verification and validation (V&V) of mission architectures in the mission engineering process is two-fold: V&V of the mission information as well as V&V of the MBSE model elements is critical. Model verification is paramount throughout development of the model to ensure consistency and avoid model element broken links. Most modeling tools have this feature built in and it should be used to assist modelers during model development. Model profiles can also be developed to enhance the process of model verification. Uncertainty quantification can be incorporated to link descriptive architectures with mathematical analysis solvers that allow users to perform optimization runs. This promotes integrating probabilistic uncertainties and enables normative decision making. Further investigation is being conducted into specific analysis methods.

Please refer to Section 6 of the MEG 2.0 for additional information related to Mission Engineering Analysis and the various computations and simulation tools.

6.1 Test and Evaluation Support

Ensuring the advantage and comprehensive Joint readiness will stretch traditional test and evaluation (T&E) capabilities further than ever before. There is an increased emphasis on the operational and mission context in which a system under test is expected to perform through the system lifecycle. Shifting the way in which we think about system performance and how T&E contributes to the overall assessment of measures and outcomes, aligned with complex mission webs and Joint SoS is paramount. T&E can use mission architectures, reinforced by decision support tools, and supported by live test and a virtual constructive environment to assess material and non-material solutions' performance, interoperability, and impact to service and Joint mission execution.

This emphasis of using MTs and METs to support T&E aligns users, operators, acquirers, testers, and sustainers with the desired mission and capability outcomes. Mission architecting and engineering is interdisciplinary by nature, and it is value added to have MTs generate linkages to converge operational, acquisitions, and engineering personnel.

Mission architectures enable testing interfaces, visualizing test scenarios, and mapping a "system under test thread" upstream to a larger mission engineering thread to demonstrate how the system under test will employ in a larger Joint, all-domain context. This methodology promotes validation where user operators ensure the right system was built, while simultaneously allowing verification of performance measures by engineers ensuring the system was built correctly in a realistic, Joint mission setting. T&E metrics and models alongside live test events and real-world operations can serve as inputs to both system performance and mission engineering models.

MBSE enhances the understanding of system performance, interoperability, and impact on mission execution in a Joint and all-domain context. MBSE is a critical tool for creating traceability, linkages, and information storage to requirements, measures, test centers, and other essential elements needed to complete testing. By utilizing ME enabled by MBSE, the entire system lifecycle can be modeled in a digital environment, allowing for an understanding of how each component interacts and contributes to the overall mission success. One of the key benefits of MBSE is its ability to provide a visual representation of the system and its requirements. This not only aids in communication between stakeholders but also allows for a clearer understanding of how changes to one aspect of the system may impact others. Linking functional, performance, and system technical requirements directly to the model guarantees traceability is maintained throughout the development and testing process, ensuring that all necessary criteria are met.

Test & Evaluation considers both validation and verification. For verification, operational and test personnel can use MOPs to investigate how systems perform in simulated operational environments. Regarding validation, system users can validate the utility of a system by tracing to MOEs. Measures and Measurement Sets employed in models provide quantitative attributes that are beneficial to both experimentation and testing efforts.

Example Functional Connectivity to Measures: Task Force Normandy SEAD Mission

Figure 23 represents a notional example of the traceability between the AH-64 and the MH-53J and their functions in the SEAD MET. In this example, the performance of the AH-64 and the MH-53J are be evaluated against the MOPs for an Air Vehicle (lethality, survivability, recovery rate, and time to deploy).

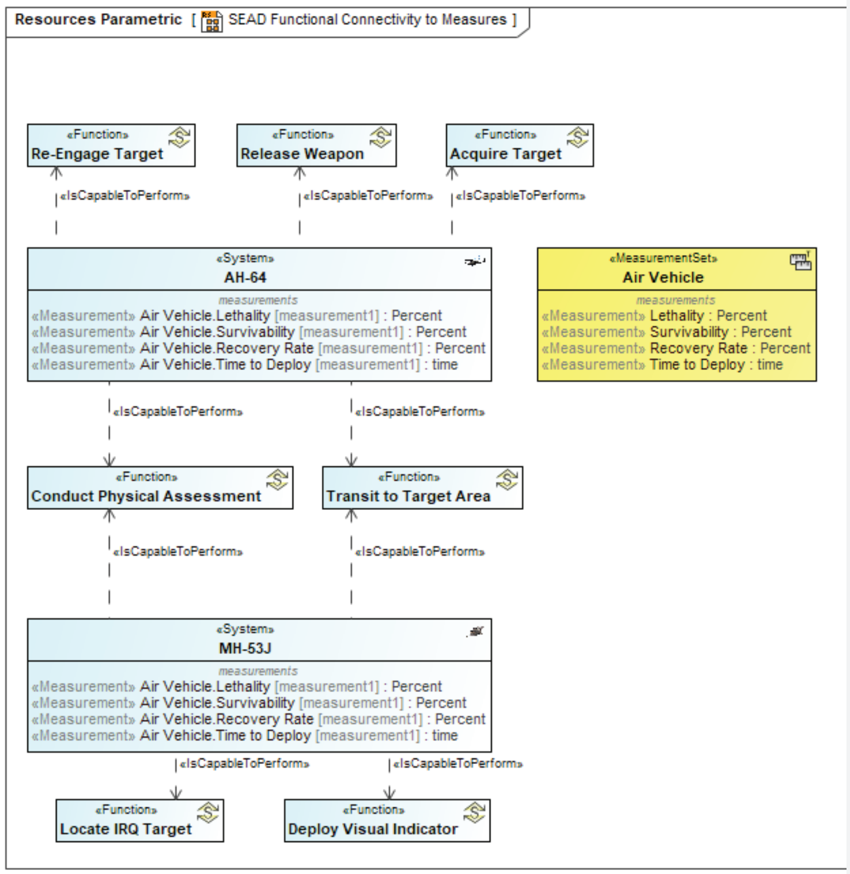

Figure 23. Example MET Function Connectivity to MOPs (UAF Resources Parametric)

7 Results and Recommendations

Upon completion, mission architectures should be made available for consumption via the various approved methods within the appropriate security construct within the DoD. This may include methods such as knowledge management webpages or addendums to joint doctrine publications. The diagrams used by architects to capture mission architectures are often too detailed for presenting to stakeholders. For presentation purposes, it is necessary to create a focused and annotated set of architecture views. It is advisable to hide low-level technical details of the architecture in presentation views as it makes them more difficult to interpret. Removing representations of elements and relationships on a diagram does not remove them from the model, meaning any context removed from the diagram to simplify the view is still captured in the model and can be accessed and represented in other diagrams in the model. This feature highlights one of the major advantages of using a dynamic model rather than a set of static architectures views. The views can be created on demand from the information stored in the model.

7.1 Presentation of ME Architectures to Leadership

When presenting to leadership, it is critical to document all starting assumptions and dependencies to ensure full comprehension in the architectures being presented. While the best practice is for the assumptions and starting conditions to be captured in the model itself, it is important to also present the information on a slide or in textual format for easy accessibility. Study documentation is submitted at the end of each study and contains detailed information related to the starting conditions, assumptions, dependencies, architectures, and other relevant items.

The appropriate way to present the architecture depends on the mission problem and the information the stakeholders need to see. The main view that OUSD(R&E)'s Mission Integration organization has found to work well when presenting to leadership is called the E2E view. The E2E view is a UAFML Resources Internal Connectivity architecture that displays the systems of interest in a mission architecture and how they logically share information to execute the mission. This E2E view is organized and annotated to highlight key pieces of the SoS architecture for stakeholder presentations. Figure 24 provides an example of how to present the E2E view.

Key considerations for presenting architectures to stakeholders:

Systems are represented and are organized from left to right to align with the order in which they are used during MET execution. The top-level steps (e.g., *kill/effects chain*) of the MET can be explicitly labeled along the top of the diagram. This is often much easier for non-architects to interpret than the associated swimlane diagram which may decompose these high-level steps into numerous low-level activities for allocation.

E2E views are often difficult to view and read if placed in another presentation format such as a PowerPoint slide. In that case, it is advisable to annotate the key features of the diagram with overlayed text and outlines like what is shown in Figure 24. In some cases, the same system can be used for more than one activity. In this case, it is important to not overload the content and context of the diagram. Aim to represent the E2E view with as little ambiguity as possible. Choose the first or most important activity that this system performs or create separate presentation views that emphasize important elements of the architecture.

Ensure that key interfaces are present. Stakeholders will want to have some understanding of how METs will be closed. However, if the number of interfaces shown clutter the view, it may be better to hide some and highlight the relevant interfaces. This can occur when there are many possible communications paths that can close the MT.

It is not advisable to include the entire communications architecture in the presented view, but it may be useful to depict targeted portions of the lower layers of the communications stack where it is important for understanding. Since the E2E view is derived from the complete model, this should be straightforward to achieve in a customized view while still being technically correct.

- Strategically display key parts within the systems. The key parts are dependent on the mission problem at hand. Do not display every part on each system. This clutters the view with irrelevant information. For example, if the sensor used by an ISR platform is important for the study, then show the sensors on the platform but hide the other parts on the E2E view.

Important mission elements can be boxed and labeled with drawing tools within the presentation software (e.g., PowerPoint) to both highlight the mission element and make them more readable. To maintain consistency when presenting architectures, use the following guidelines in PowerPoint:

- Add boxes around every mission element grouping; Create a rectangle with no fill and outline in dark blue (RGB (47 82 143)) with a 3-point line thickness.
- Based on the message being presented, overlay the E2E view with lines to help showcase the narrative in the baseline and alternative views.
- For Assess or other steps which are conducted by multiple mission element groups, add a green circle (RGB (146 208 80); 0.2" height and width; black outline in 1 point) to each.
- Add legends for the box colors and arrows as shown in Figure 24.

  Include a readable description of the architecture and the top-level MET execution shown in the view. While this information should also be stored at a lower level of detail in the architecture model, it can be helpful to use the presentation software to summarize and make it more readable and interpretable by non-architects.

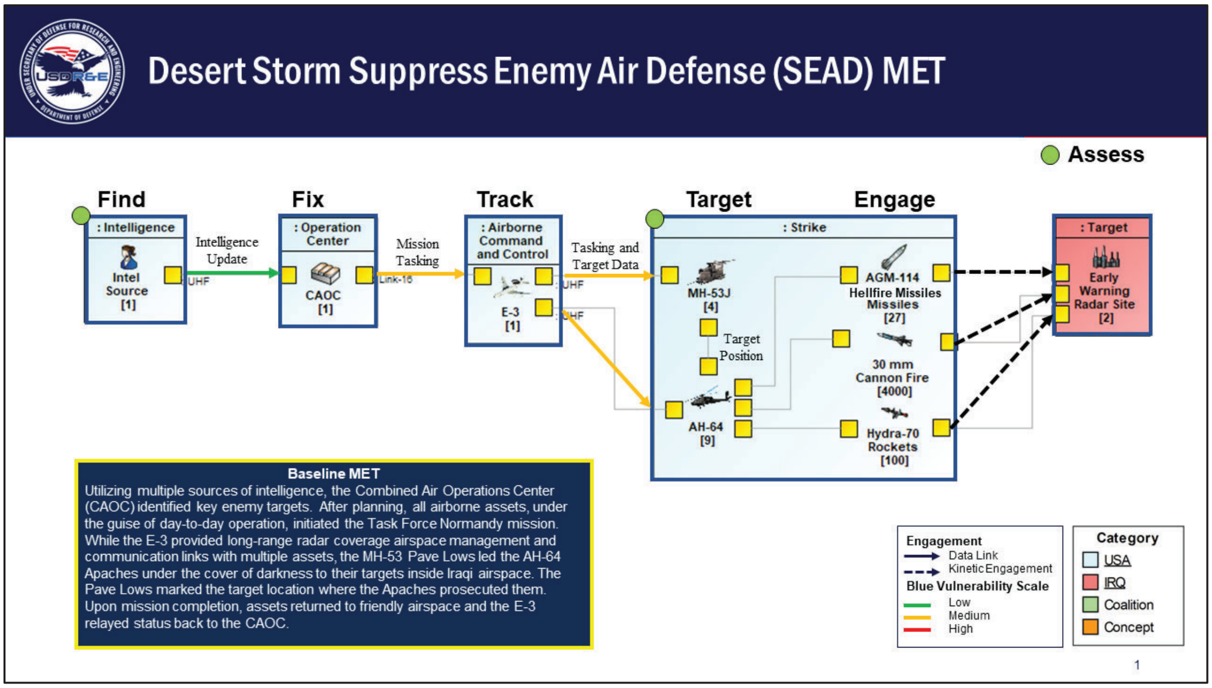

Figure 24. Presentation example for E2E View of baseline architecture

Typically, one E2E view is made for each major case in the study experimental design (e.g., baseline, alternative/concept 1, alternative/concept 2, etc.). It is helpful to highlight the differences in each alternative view relative to the baseline with color-coding as show in Figure 25.

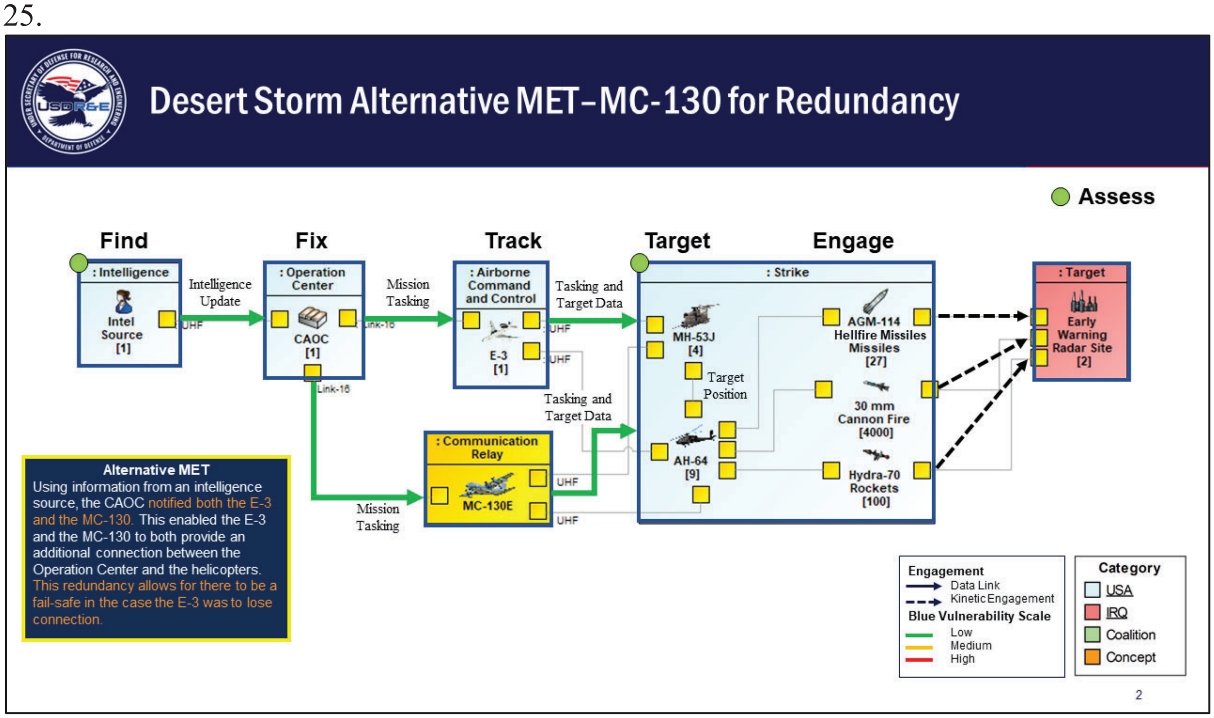

Figure 25. Presentation example for E2E View alternative architecture

When there are multiple adjustments to the E2E architecture between the baseline and alternative cases, it can be difficult for audiences to internalize the changes quickly. To avoid flipping back and forth between slides, it is also useful to create a side-by-side comparison view (see Figure 26) between the baseline and alternative architectures.

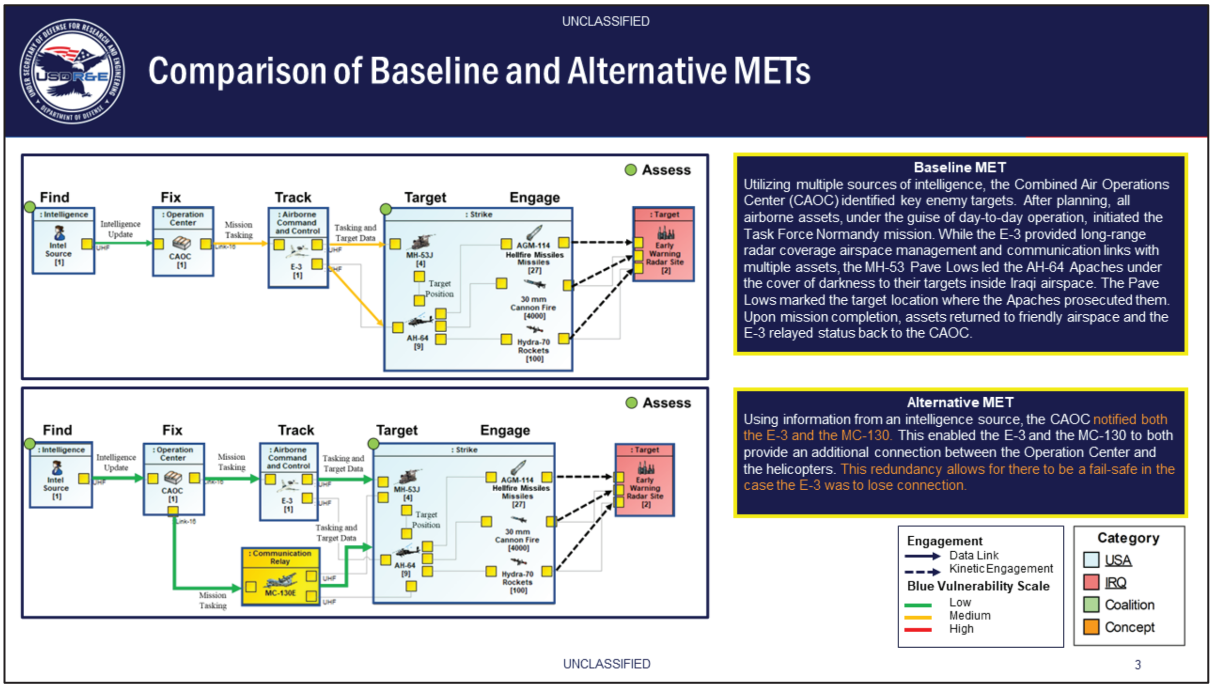

Figure 26. Presentation Example for E2E Comparison between baseline and alternative architectures

8 Summary

This guide has provided a brief description of how architecture modeling is used in the execution of mission engineering as defined by the mission engineering methodology. It describes the way that the mission architectures cut across all steps in the methodology and provide a precise description of the baseline and alternatives addressed in a mission engineering study or initiative. Utilizing a model, rather than PowerPoint, Visio, or other diagram tools, to develop these views is a more efficient and effective method to create and communicate MTs, METs, and larger mission architectures in a queryable and standardized way. A model allows for the collection and digital representation of a mission architecture in a centralized authoritative source. Models can be modularized to promote reuse and unity of effort. This guide is intended to be a living document, expected to be iterated like the MEG with subsequent version releases. And finally, it provides a laydown of the preferred E2E view representation which is the major product for results of the mission engineering process as presented to Senior leadership.

9 Appendix

9.1 Architecture View Conventions in Digital Engineering Tool

Below are a series of best practices to maintain DoD architecture consistency. While the architectures can be created in any digital engineering tool, instructions in this document were created using Cameo System Modeler. It is recommended that each organizational group consider it a best practice and define and document their own style guide while remaining compliant with existing style guides, including this MASG 1.0.

9.1.1 Diagram Conventions

- **Diagram Titles** should follow the following naming schema:
    - [Diagram type] [Short vignette ID] [ISO Country Code[18] of performer(s) or 3-letter representation of non-state actor] [Short name – less than 20 characters] – [Baseline or Concept name]
    - Diagram Types:
        - OB for Resources Structure/Asset List
        - MT for MT/Operational Process (Op-Pr) Flow
        - MET for MET/ Resources Process (Rs-Pr) Flow
        - E2E for E2E/ Resources Internal Connectivity (Rs-Cn)
    - Country codes should be listed in the following order: *blue* forces alphabetically, *green* forces alphabetically, *red* forces alphabetically.
- **Diagram Titles**, other than the special cases described above, should be succinct providing information about the purpose of the diagram. A rule of thumb is to keep titles shorter than 20 characters and not include the UAFML diagram type.
- **Comments** can be added to Diagrams to include any information that is important for the reader to understand. There can be any number of comments on any diagram, limited only by readability. To add a Comment, select **Comment** under **Note** in the diagram palette.
- **Diagram Frames** should be shown with the diagram type (full name, not abbreviated), type icon, and name shown. Parameters should be hidden.

[18] International Organization for Standardization (ISO). (n.d.). ISO 3166 Country Codes. Retrieved from https://www.iso.org/iso-3166-country-codes.html

- **Diagram Information** should be shown on all diagrams. This pane can be customized to include information such as drafter, reviewer, and approver, classification level, brief description of the architecture, and version (see Figure 27). Like engineering diagrams, the lower right corner of a diagram is the recommended location for the diagram information pane for OUSD(R&E) architectures. Each organizational group should set the standard for the position of the Diagram Information pane and should remain consistent on all views.

Approver	B. Jaime	**Project: Mission Architecture Style Guide**	**OUSD(R&E)**	5/30/24, 8:58 PM
Reviewer	M. Rock	**Diagram: OOB USA IRQ Coalition**	**UNCLASSIFIED**	**Version 0.1**
Drafter	K. Dina	**Mission elements in Operation DESERT STORM, Task Force Normandy**		

Figure 27. Custom Diagram Information

- **Classification:** Security Classification Markings
    - Security classification markings should be captured for each element and view in the model. Despite the lack of modeling tools having the inherent capability to appropriately mark, identify, and protect classified information, all models must be marked appropriately. Some versions of vendor tools support the ability to portion mark models, such as the Dassault Cameo Enterprise Architect 2022x release using its "Data Marking and Classification" plug-in (No Magic, Inc., 2022). Alternatively, users can create profiles or build other extensions to accommodate the capabilities like portion marking models and completing mass model data updates for a view or model.
- Cameo also allows additional information to be added for a specific element. In the **Specification Window** under the **Documentation/Comments** group additional information related to an object can be added, associated Military Service (e.g., Army, Navy, Air Force, Marine Corps, Space Force), and any other relevant background information.
    - **Comments** are elements within a model, whereas **Notes** are adornments on a diagram. Using Comments enables simpler methods for tracking since they exist within the model itself, as opposed to just existing in the single view of the diagram.
- The **Glossary** should be used spelled out acronyms.
- The following information should be included in each diagram through Documentation/ Comments or in Diagram Information pane:
    - *Diagram Classification*: The classification of the diagram as displayed in a read-only format.
    - *Mission:* The name of the mission being modeled in the diagram
    - *Purpose*: Description of the purpose of the diagram
    - *Narrative*: Textual description of what the diagram shows.
    - *Authors*: Name of individuals which contributed to the creation of the diagram including but not limited to modeler(s) and Subject Matter Experts which contributed to its creation.
    - *Date Created*: Day, month, and year that the diagram was originally created.

- *Date Reviewed and POCs:* Day, month, year, and name of individual(s) and/or organizations which reviewed the diagram. If a diagram was last reviewed in a previous version, this field should be set to a value of “Review Required”.
- *Version:* Using the standard major/minor update schema, the version associated with the current diagram. Major updates make a significant change to the content or organization of the diagram; minor updates include those of a minor or administrative nature (e.g., change the name of an Element to correct a spelling error).
- *Conditions/ Assumptions:* Textual description of all assumptions made and conditions for the diagram.

- **Sources** should be captured in the model. This can be done through the comments, a sourcing profile, or via a sourcing matrix companion for each diagram. The specific names, locations, dates, and overall classification of all sources used to generate the information captured on the diagram should be captured in the model. Architects should call out specifically which portions of the diagram originated from each source and reference any relevant security classification guides (SCG) and MIL-STD. When creating architectures, defer to MIL-STDs and superseding DoD guidance where applicable.
- Each mission element should have a **Value Property** added named “**Country**”. The **Default Value** should be set to the ISO 3-digit country code[19]. This can be done in the Specification Menu under the Attribute tab.
- Each United States mission element should have a **Value Property** added named “**Service Component”.** The **Default Value** should be set to the Military Service (e.g., Army, Navy, Air Force, Marine Corps, Space Force) to which they are resourced. This can be done in the Specification Menu under the Attribute tab.
  - A table can be created using the digital engineering tool using the information and properties in the model to represent the USA mission elements and the associated ownership (Figure 28).

#	Name	Country and Force	Physical Interface
1	Intel Source	USA	UHF
2	CAOC	USA	Link-16 UHF
3	E-3	USA Air Force	Link-16 UHF
4	MC-130E	USA Air Force	Link-16 UHF
5	MH-53J	USA Air Force	UHF Light Marker
6	AH-64	USA Army	UHF Fire Control Light Marker
7	AGM-114 Hellfire Missiles	USA	Fire Control Mission Data
8	30 mm Cannon Fire	USA	Fire Control Gravity
9	Hydra-70 Rockets	USA	Fire Control Heat Signature
10	Early Warning Radar (EWR) Site	IRQ	Mission Data Gravity Heat Signature

Figure 28. Example Table of Mission Elements, Service Components, and Interfaces

[19] International Organization for Standardization (ISO). (n.d.). ISO 3166 Country Codes. Retrieved from https://www.iso.org/iso-3166-country-codes.html

9.1.2 Diagram Formatting

- Minimize empty space as much as possible, wrapping text or resizing images as required.
- Font Size Suggestions:
  - Element names: Arial font, size 22
  - Flows: Arial Font, size 14
  - Ports and Pins: Arial font, size 16
- Properties such as font, word wrap, showing name/type, and others related to the appearance of the diagram can be found by right clicking an element and selecting **Symbol Properties** (Figure 29).

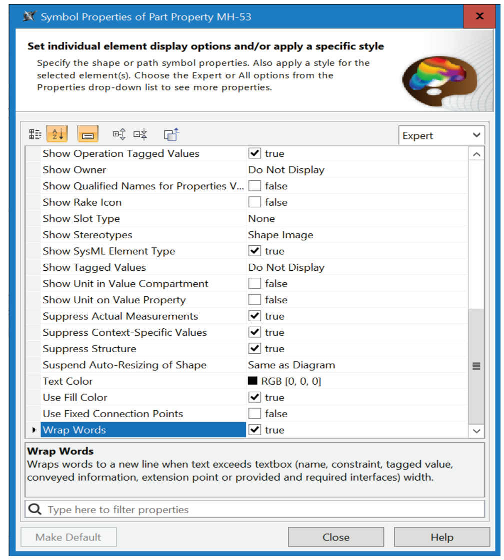

Figure 29. Example Symbol Properties

9.1.3 Legends

- A **Legend** can be used to visual code specific concepts using colors. Use a legend across all diagrams to visually indicate the country of ownership of either mission elements or activities/functions. Based on the digital engineering tool, the legend can be applied manually or by setting parameters that allow for automatic legend item assignment.
- In the architecture models, *blue* is used to represent United States forces, *green* represents allied combatants, and *red* represents adversary forces. For alternative architectures, a gold color is used to represent "concepts". A legend, like Figure 30, should be used in all architectures. To maintain consistency among the enterprise for force designation, use the following RGB codes:
  - USA *Blue Force* Mission Elements: RGB (214 255 255)
  - Adversary *Red Force* Mission Elements: RGB (237 124 123)
  - Allied combatants/Coalition *Green Force* Mission Elements: RGB (162 255 162)
  - Alternative/Concept Gold Mission Elements: RGB (255 153 0)
- All elements in the architecture must be assigned to a Legend Item
- In a diagram, right-click the shape of the element to which you want to apply the Legend to and select **Legend Item** from the drop-down menu.

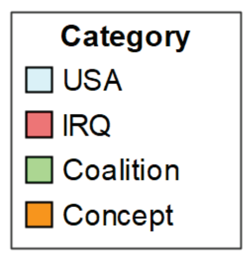

Figure 30. Example Legend

9.2 Glossary

Accreditability: Possesses the features required to support accreditation for classified information handling, safety, or other critical requirements. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Adaptability: An architecture tenet that extends the composability tenet with features that enable the ensemble and its elements to respond to new or unanticipated tasks and events in ways that make best use of their capability to achieve new goals. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Alternative Mission Approach: A change to the baseline mission approach for how the mission will be executed. [Source: OUSD(R&E) MEG 2.0]

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 [Source: ISO/IEC/IEEE 42020:2019]

Assumption: A specific supposition of the operational environment that is assumed true, in the absence of positive proof, essential for the continuation of planning. [Source: JP 5-0, Department of Defense Dictionary]

Baseline Mission Approach: The agreed upon starting point for how the mission will be executed to address the mission engineering effort; driven by the mission, scenario, and epoch. [Source: OUSD(R&E) MEG 2.0]

Blue Force: U.S. Combatants [Source: OUSD(R&E) MEG 2.0]

Capability Area: Joint Capability Area (JCA) – Collections of like DoD capabilities functionally grouped to support capability analysis, strategy development, investment decision making, capability portfolio management, and capabilities-based force development and operational planning. [Source: JCIDS Manual, DAU Glossary 13th Edition, DMO Model Developers Guide]

Capability Configuration: A composite structure representing the physical and human resources (and their interactions) in an enterprise, assembled to meet a capability. [Source: Enterprise Architecture Guide for UAF v1.2 Specification]

Capability Gap: The inability to meet or exceed a capability requirement, resulting in an associated operational risk until closed or mitigated. [Source: JCIDS Manual, DAU Glossary 13th Edition]

Capability Level Requirement: A singular documented need of what a particular product or service should be or do. [Source: DM2, DMO Model Developers Guide]

Capability Requirement: A capability which is needed to meet an organization's roles, functions, and missions in current or future operations. [Source: JCIDS Manual, DAU Glossary 13th Edition]

Capability Solution: A materiel solution or non-materiel solution to satisfy one or more capability requirements and reduce or eliminate one or more capability gaps. Also called, "solution". [Source: JCIDS Manual, DAU Glossary 13th Edition]

Capability Timeframe: MM/YYYY of the end/start of the capability in a phase or phases. [Source: DM2, DMO Model Developers Guide]

Capability: The ability to complete a task or execute a course of action under specified conditions and level of performance. [Source: MEG 2.0, CJCSI 5123.01H, DAU Glossary]; The ability to achieve a Desired Effect under specified [performance] standards and conditions through combinations of ways and means [activities and resources] to perform a set of activities. [Source: DM2, DMO Model Developers Guide]

Clarity: An architecture tenet that addresses the important characteristics of an architecture and its model that make them easy to understand and apply; this is central to the aesthetic dimension of architecture. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Composability: An architecture tent that provides the features at various levels to allow rapid assembly and integration of subordinate elements to create a capability package tailored to a specific task or situation, especially through interfaces that facilitate vertical and horizontal integration. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Concept of Operations (CONOPS): A verbal or graphic statement that clearly and concisely expresses what the commander intends to accomplish and how it will be done using available resources. [Source: JP 5-0, Department of Defense Dictionary]

Concern: A Concern is any interest in the system. Examples of concerns: (system) purpose, functionality, structure, behavior, cost, supportability, safety, interoperability. [Source: ISO/IEC/IEEE 42010]

Condition: 1. Those variables of an operational environment or situation in which a unit, system, or individual operates and that may affect performance. 2. A physical or behavioral state of a system that is necessary for the achievement of an objective. [Source: JP 3-0, Department of Defense Dictionary]; The state of an environment or situation in which a Performer performs or is disposed to perform. [Source: DM2, DMO Model Developers Guide]

Constraint: In the context of planning, a requirement placed on the command by a higher command that dictates an action, thus restricting freedom of action [Source: JP 5-0, Department of Defense Dictionary]. Constraints may also refer to the range of permissible states for an object [Source: Department of Defense CIO Architecture Framework]

Criteria: The minimum acceptable level of performance associated with a particular measure of task success, effectiveness, or performance. It is often expressed as hours, days, percent, occurrences, minutes, miles, or some other command-stated measure. [Source: CJCSI 3500.1 Series]

Data Curation: The ongoing processing and maintenance of data throughout its lifecycle to ensure long-term accessibility, sharing, and preservation [Source: OUSD(R&E) MEG 2.0, National Library of Medicine]

Driver: A factor which will have a significant impact on the activities and goals of an enterprise [Source: UAFML Version 1.2 Specification]

Effect: 1. The physical or behavioral state of a system that results from an action, a set of actions, or another effect. 2. The result, outcome, or consequence of an action. 3. A change to a condition, behavior, or degree of freedom. [Source: JP 3-0, Department of Defense Dictionary]

Effects Chain: A sequence of actions that must be successfully accomplished to deliver desired effects to meet a stakeholder need(s). Examples include Humanitarian Assistance/Disaster Response when providing defense support of civil authorities and stabilization operations.

Additional examples can include non-kinetic actions such as an information operations radio broadcast to encourage enemy surrender, and employment of electronic warfare capabilities. [Source: OUSD(R&E) MEG 1.0, 2023 DoD Electromagnetic Spectrum Superiority Strategy, Air Force Doctrine Publication 3-0]

Enterprise: A purposeful combination of interdependent resources that interact with each other to achieve business and operational goals. [Source: INCOSE SE Handbook, 5th edition, 2023]

Enterprise Architecture: Applies the basic definition of architecture (structure, behaviors, and global rules) to the top level of an ensemble of nodes, systems, elements, or other resources that collaborate to fulfill the functions of an overall organization or business process. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Excursion Mission Approach: A change to the assumptions, mission elements, and/or behaviors made in the baseline mission approach [Source: OUSD(R&E)]

Enterprise Systems Engineering: is the application of systems engineering principles, concepts, and methods to the planning, design, improvement, and operation of an enterprise. [Source: SEBoK Wiki]

Function: An activity which is specified in the context of a human or machine that is capable to perform it. [Source: UAFML Version 1.2 Specification]

Green Force: Allied combatants. [Source: OUSD(R&E) MEG 2.0]

Hardware Architecture: Applies the basic definition to hardware, focusing on processors, storage, interconnects, operator stations, communications, sensors, effectors, and other hardware elements. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Interoperability: An architecture tenet that provides the features to support information sharing, common understanding and collaborative action with other systems and nodes of an enterprise; implements enterprise services and data strategies. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Joint: Connotes activities, operations, organizations, etc., in which elements of two or more Military Departments participate. *Note that this definition of "joint" is applicable to capability requirements documents and capability solutions which apply to more than one DoD Component.* [Source: JCIDS Manual, DAU Glossary 13th Edition]

Joint Performance Requirement (JPR): A performance requirement that is critical or essential to ensure interoperability or fulfill a capability gap of more than one armed force, Defense Agency, or other entity of the Department of Defense, or impacts the joint force in other ways such as logistics. [Source: JCIDS Manual, DAU Glossary 13th Edition]

Kill Chain: A Mission Thread with a kinetic outcome. Dynamic targeting procedures often referred to as F2T2EA by air and maritime component forces; and Decide, Detect, Deliver, and Assess methodology by land component forces. [Source: OUSD(R&E) MEG 2.0, JP 3-09]

Kill Web: An inclusive set of multiple integrated Mission Threads and Mission Engineering Threads for the applicable scenario or vignette of interest. [Source: OUSD(R&E) MEG 2.0, OUSD(R&E)]

Loose Coupling: An architecture tenet that minimizes the detailed knowledge that one element requires about another to collaborate, exchange information, or provide services; maintains interface definitions independent of the implementation of functions or services accessed through the interfaces closely related to service orientation. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Measure of Effectiveness (MOE): Measurable military effects or target values for success that are derived from executing tasks and activities to achieve the MOS. [Source: OUSD(R&E) MEG 2.0]

Measure of Performance (MOP): Measurable performance characteristics or target parameters of systems or actors used to carry out the mission tasks or military effect. [Source: OUSD(R&E) MEG 2.0]

Measure of Success (MOS): Measurable attributes or target values for success within the overall mission in an operational environment. Measures of success are typically driven by the mission objectives of the *blue* force. [Source: OUSD(R&E) MEG 2.0]

Measure: The magnitude of some attribute of an individual. [Source: DM2, DMO Model Developers Guide]

Mission: The essential task or tasks, together with a purpose, that clearly indicates an action to be taken and the reason for the action. [Source: JP 3-0, Department of Defense Dictionary]

Mission Architecture: A view or representation that depicts the ways and means to execute a specific end-to-end mission, with relationships and dependencies amongst mission elements. This includes elements such as mission activities, approaches, systems, systems of systems, organizations, and capabilities. [Source: OUSD(R&E) MEG 2.0]

Mission Architecture Management: the planning, steering, controlling, and integration of Mission Architectures. Mission Architecture Management is typically covered by mission architects in large organizations.

Mission Area: The mission areas involve multiple Armed Forces and multiple programs and, at a minimum, examples include the following: (1) Close air support. (2) Air defense and offensive and defensive counter-air. (3) Interdiction. (4) Intelligence, surveillance, and reconnaissance. (5) Any other overlapping mission area of significance, as jointly designated by the Deputy Secretary of Defense and the Vice Chairman of the Joint Chiefs of Staff for purposes of this subsection. A mission area typically aligns the affected stakeholders (Users, Operators, Acquirers, Testers, Sustainers) with the desired mission and capability outcomes. [Source: 2017 NDAA, Section 855]

Mission Characterization: The aggregate of factors associated with military objectives and operations; this includes the mission to be accomplished in a specific time and place, the measures of success, the threats, and constraints. Changes in any factors of the mission characterization may cause the mission to be redefined. [Source: OUSD(R&E) MEG 2.0]

Mission Context: The elements that describe who, what, when, where, and why elements of the mission to be accomplished. Changes in any elements of the mission context may cause the mission to be redefined. [Source: OUSD(R&E) MEG 2.0]

Mission Element: A person, organization, platform, and/or system that performs a task. [Source: OUSD(R&E) MEG 2.0]

Mission Engineering Analysis: The approach to evaluate mission architectures within the specific scenario-based mission context to provide quantitative outputs that explore mission impacts. [Source: OUSD(R&E) MEG 2.0]

Mission Engineering Thread (MET): Mission threads that include the details of the capabilities, technologies, systems, and organizations required to execute the mission. [Source: OUSD(R&E) MEG 2.0]

Mission Engineering: An interdisciplinary process encompassing the entire technical effort to analyze, design, and integrate current and emerging operational needs and capabilities to achieve desired mission outcomes. [Source: OUSD(R&E) MEG 2.0]

Mission Integration Management: a core activity within the acquisition, engineering, and operational communities to focus on the integration of elements that are all centered around the mission. [Source: 2017 NDAA, Section 855]

Mission Objective: A clearly defined, decisive, and attainable end toward which every operation is directed. An objective is a specific, time-targeted, measurable, and attainable target that an enterprise seeks to meet to achieve its goals. [Source: DM2, DMO Model Developers Guide]

Mission Requirements: Requirements that relate to objectives of the stakeholders that are defined in the context of the supersystem, not the system itself. [Source: The Engineering Design of Systems, Models and Methods, by Dennis M. Buede, 2009]

Mission Tasks: A clearly defined action or activity specifically assigned to a system, individual or organization that must be complete. [Source: Adapted from JP-01]

Mission Thread: A sequence of end-to-end mission tasks, activities, and events presented as a series of steps to achieve a mission. [Source: OUSD(R&E) MEG 2.0]

Model Federation: A concept that connects system models from different organizations to create the system-of-systems model that guides enterprise-SoS engineering. Model federation enables distribution of model development efforts across the combined workforce and helps promote synergy. Additionally, model federation relies on libraries of high-level, predefined, common elements that can be used across organizations. [Source: OUSD(R&E)]

Model: A physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process. [Source: DoDI 5000.61, DoDI 5000.70]; Per the Systems Engineering Body of Knowledge, Models are often categorized as Descriptive, Analytic, etc. [Source: Systems Engineering Body of Knowledge] Per other sources, a model is any incomplete representation of reality, an abstraction. The essence of a model is the question or set of questions that the model can reliably answer for us. [Source: The Engineering Design of Systems, Models and Methods, by Dennis M. Buede, 2009]

Modularity: An architecture tenet that is hierarchically partitioned in a fashion that: 1) Groups related functions and information, minimizing required interactions across boundaries 2) Allows upgrading and technology refreshment 3) Facilitates the application of component-based design methods. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Objective: 1. The clearly defined, decisive, and attainable goal toward which an operation is directed. 2. The specific goal of the action taken which is essential to the commander's plan. See also target. [Source: JP 5-0, Department of Defense Dictionary]

Openness: An architecture tenet designed to help realize the characteristics of open system as described through 1) Modularity and standards compliance 2) complete, unambiguous definitions of internal and external interfaces 3) Absence of reliance on products and features that are proprietary or limited availability. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Operation: 1. A sequence of tactical actions with a common purpose or unifying theme [Source: JP 1 Vol 1, Department of Defense Dictionary]; 2. A military action or the carrying out of a military mission [Source: JP 3-0, Department of Defense Dictionary]

Operational Level Requirement: Operational requirements are those statements that "identify the essential capabilities, associated requirements, performance measures, and the process or series of actions to be taken in effecting the results that are desired in order to address mission area deficiencies, evolving applications or threats, emerging technologies, or system cost improvements." The operational requirements assessment starts with the Concept of Operations (CONOPS) and goes to a greater level of detail in identifying mission performance assumptions and constraints and current deficiencies of or enhancements needed for operations and mission success. Operational requirements are the basis for system requirements. [Source: MITRE, DMO Model Developers Guide]

Operational Logical Architecture: Defines the enterprise architecture in a solution independent form including behavior and structure, typically modeled as Mission Threads. [Source: DMO Model Developers Guide]

Operational Performer: Any logical entity - agnostic of human, automated, or any aggregation of human and/or automated - that performs a task. [Source: DM2, DMO Model Developers Guide]

Operational Viewpoint: Viewpoint in UAF focused on how capabilities are actualized by operational roles, activities, and performers [Source: Enterprise Architecture Guide for UAF v1.2 Specification]

Order of Battle (OB): The identification, strength, command structure, and disposition of the personnel, units, and equipment of any military force. Also called OB. [Source: JP 2-0, Department of Defense Dictionary]

Personnel Viewpoint: Viewpoint in UAF focused on how operational concepts are implemented through personnel [Source: Enterprise Architecture Guide for UAF v1.2 Specification]

Project Viewpoint: Viewpoint in UAF focused on how plans deliver the resources according to project activities and milestones [Source: Enterprise Architecture Guide for UAF v1.2 Specification]

Red Force: Adversary combatants. [Source: OUSD(R&E) MEG 2.0]

Reference Architecture: A logical/functional abstraction that defines the features and behaviors common to a domain or class of entities. An RA is instantiated by the addition of relevant detail to achieve a physical architecture that satisfies a specific set of requirements within the domain. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Resource Architecture: The Resource Architecture and contained Systems implement the Operational Architecture and both are mapped to the Actual Mission. [Source: Enterprise Architecture Guide for UAF v1.2 Specification]

Resources Viewpoint: Viewpoint in UAF focused on how operational concepts are implemented through resources/ mission elements [Source: Enterprise Architecture Guide for UAF v1.2 Specification]

Scalability: Allows performance or capacity to be increased by the addition of relatively small units of hardware or software with overall performance commensurate with installed resources. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Scenario: Description of the geographical location and timeframe of the overall conflict. A scenario includes information such as threat and friendly politico-military contexts and backgrounds, assumptions, constraints, limitations, strategic objectives, and other planning considerations. [Source: OUSD(R&E) MEG 2.0]

Software Architecture: Applies the basic definition to software, focusing on frameworks, software requirements, application programs, infrastructure programs, workflow management, networking and messaging, interfaces, and other aspects of computer programming. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Stakeholder: An individual organizational resource, or type of organizational resource (both internal and external to the enterprise) who has an interest in, or is affected by, outcomes or intermediate effects generated or influenced by the enterprise. [Source: UAFML Version 1.2 Specification]; Stakeholders are individuals, groups or organizations holding Concerns for the System of Interest. Examples of stakeholders: client, owner, user, consumer, supplier, designer, maintainer, auditor, CEO, certification authority, architect. [Source: ISO/IEC/IEEE 42010]

Standards Compliance: Applies mature, publicly available, consensus-based standards to architecture elements, services, and interfaces following applicable policy. [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

Strategic Viewpoint: Viewpoint in UAF focused on strategy, objectives, desired capabilities, phasing structure, MOEs, and roadmaps; The strategic plan deploys capabilities in phases to help address gaps and shortfalls [Source: Enterprise Architecture Guide for UAF v1.2 Specification]

System Architecture: Applies the basic definition to an ensemble of elements (Ultimately hardware and software components) that collaborate to fulfill defined requirements allocated to a node or system (Implying that a clear system boundary and user interfaces are defined). [Source: Borky and Bradley, Effective Model-Based Systems Engineering, 2019]

System Function: A function that is performed by a system. Although commonly used to refer to the automation of activities, data transformation or information exchanges within IT systems, it also refers to the delivery of military capabilities. [Source: DM2, DMO Model Developers Guide]

System Level Requirement: A singular documented need of what a particular product or service should be or do [Source: DM2, DMO Model Developers Guide]

System: A functionally, physically, and/or behaviorally related group of regularly interacting or interdependent physical elements. [Source: DM2, DMO Model Developers Guide]

Target: An entity or object that performs a function for the threat considered for possible engagement or other action. [Source: JP 3-60, Department of Defense Dictionary]
Task: A clearly defined action or activity specifically assigned by an appropriate authority

to an individual or organization, or derived during mission analysis, that must be accomplished. [Source: JP 1, Vol 1, Department of Defense Dictionary]

Test Plan and Procedure: Documents the overall structure and objectives of the Test and Evaluation (T&E) program. It provides a framework within which to generate detailed T&E plans and documents schedule and resource implications associated with the T&E program. The TEMP identifies the necessary Developmental Test and Evaluation (DT&E), Operational Test and Evaluation (OT&E), and Live Fire Test and Evaluation (LFT&E) activities. It relates program schedule, test management strategy and structure, and required resources to: Critical Operational Issues (COIs), Critical Technical Parameters (CTPs), objectives and thresholds documented in the Capability Development Document (CDD), evaluation criteria, and milestone decision points. For multi-Service or joint programs, a single integrated TEMP is required. [Source: DAU Glossary 13th Edition]

Threat: The sum of the potential strengths, capabilities, and strategic objectives of any adversary that can limit U.S. mission accomplishment or reduce force, system, or equipment effectiveness. The threat does not include (a) natural or environmental factors affecting the
34 ability or the system to function or support mission accomplishment, (b) mechanical or component failure affecting mission accomplishment unless caused by adversary action, or (c) program issues related to budgeting, restructuring, or cancellation of a program. [Source: DAU Glossary, CJCSI 5123.01H]

Validation: The process of determining the degree to which a model or simulation and its associated data are an accurate representation of the real world from the perspective of the intended uses of the model. Applicable to an expressed user need and consistent with program concept of operations. [Source: Space and Missile Systems Center Mission Engineering Primer and Handbook]

Verification: The process of determining that a model or simulation implementation and its associated data accurately represent the developer's conceptual description and specifications. [Source: JP 3-13.1, Department of Defense Dictionary]

View: An informative item, governed by an architecture viewpoint, comprising part of an architecture description that communicates some aspect of an architecture. [Source: UAFML Version 1.2 Specification]; An Architecture View in an AD expresses the Architecture of the System of Interest from the perspective of one or more Stakeholders to address specific Concerns, using the conventions established by its viewpoint. [Source: ISO/IEC/IEEE 42010]

Viewpoint: Conventions for the creation, interpretation, and use of an architecture view to frame one or more concerns that governs the creation of views. [Source: UAFML Version 1.2 Specification]; An Architecture viewpoint is a set of conventions for constructing, interpreting, using, and analyzing one type of Architecture View. [Source: ISO/IEC/IEEE 42010]

Vignette: A narrow and specific ordered set of events, and behaviors and interactions for a specific set of systems, to include blue force capabilities and red force (threats) within the operational environment. Vignettes can represent small, ideally self-contained parts of a scenario. [Source: OUSD(R&E) MEG 2.0]

White Force (White Units): Non-combatant or neutral units. [Source: OUSD(R&E) MEG 2.0]

9.3 Abbreviation List

AFTTP	Air Force Tactics, Techniques, and Procedures
AMD	Air and Missile Defense
BDD	Block Definition Diagram
BMC2	Battle Management Command & Control
BPMN	Business Process Modeling Notation
C2	Command and Control
CJCSI	Chairman of the Joint Chiefs of Staff Instruction
CONOPS	Concept of Operations
DAU	Defense Acquisition University
DMA	Digital Mission Architecture (OUSD(R&E) MI)
DoD	Department of Defense
DoDI	Department of Defense Instruction
DoDAF	Department of Defense Architecture Framework
DOT&E	Director, Operational Test & Evaluation
DOTMLPF-P	Doctrine, Organization, Training, Materiel, Leadership and Education, Personnel, and Facilities
DTE&A	Development Test, Evaluation, and Assessment
E2E	End-to-End
EA	Enterprise Architecture
ESE	Enterprise Systems Engineering
F2T2EA	Find, Fix, Track, Target, Engage, Assess
FFRDC	Federally Funded Research & Development Centers
IBD	Internal Block Diagram
ISO	International Organization for Standardization
ISR	Intelligence, surveillance, and reconnaissance
JCA	Joint Capability Areas
JCSFL	Joint Common System Function List
JICO	Joint Interface Control Officer
JMETL	Joint Mission Essential Task List
JP	Joint Publication
JTC	Joint Test Concept
M&S	Modeling and Simulation

MASG	Mission Architecture Style Guide
MBSE	Model-Based Systems Engineering
MC	Mission Capabilities
MCI	Mission Concept Integration (OUSD(R&E)-MI)
ME	Mission Engineering
MEA	Mission Engineering Analysis (OUSD(R&E)-MI)
MEG	Mission Engineering Guide
MET	Mission Engineering Thread
MI	Mission Integration (OUSD(R&E))
MIL-STD	Military standard
MoDAF	Ministry of Defence Architecture Framework
MOE	Measures of Effectiveness
MOP	Measures of Performance
MOS	Measures of Success
MT	Mission Thread
NAF	NATO Architecture Framework
OB	Order of Battle
ODASD(MI)	Office of the Deputy Assistant Secretary of Defense for Mission Integration
OMG	Object Management Group
O-Plan	Operation Plan
Op-Pr	Operational Processes
OSD	Office of the Secretary of Defense
OUSD(A&S)	Office of the Under Secretary of Defense for Acquisition and Sustainment
OUSD(R&E)	Office of the Under Secretary of Defense for Research and Engineering
Rs-Cn	Resources Connectivity
Rs-Pr	Resources Processes
Rs-Sr	Resources Structure
SA	System Architecture
SCG	Security Classification Guide
SE	Systems Engineering
SEAD	Suppression of Enemy Air Defense
SME	Subject Matter Expert
SoS	System of Systems

St-Sr	Strategic Structure
St-Tx	Strategic Taxonomy
SysML	Systems Modeling Language
T&E	Test and Evaluation
TLAMs	Tomahawk Land Attack Missiles
TTP	Tactics, Techniques, and Procedures
U.S.	United States
UAF	Unified Architecture Framework
UAFML	Unified Architecture Framework Modeling Language
UARC	University Affiliated Research Centers
UJTL	Universal Joint Task List
UML	Unified Modeling Language
USD(R&E)	Undersecretary of Defense for Research and Engineering
UTDT	UJTL Task Development Tool
V&V	Verification and Validation

10 Bibliography

Berg, C. P., Tilley, K. E., & Kem, J. D. (2018). *Deep Manuever Historical Case Studies of Manuever in Large-Scale Combat Operations.* Army University Press. Retrieved from https://www.armyupress.army.mil/Books/Browse-Books/iBooks-and-EPUBs/Task_Force_Normandy/

Borky, J. M., & Bradley, T. H. (2019). *Effective Model-Based Systems Engineering.* Springer.

Curtis E. Lemay Center for Doctrine Development and Education. (2016). *Air Force Doctrine Publication (AFDP) 3-0 Operations and Planning.* Retrieved from https://www.doctrine.af.mil/Portals/61/documents/AFDP_3-0/3-0-D19-OPS-Effects-Based-Plan.pdf

Data Markings and Classifications. (n.d.). Retrieved from Dassault Systemes: https://www.3ds.com/products/catia/no-magic/data-marking-classification-plugin

Defense Acquisition Library (DAU). (2020, July 21). Retrieved from https://www.dau.edu/tools/dau-glossary

Department of Defense OUSD(R&E). (2020). *Department of Defense Mission Engineering Guide.* Washington. DC.

Department of Defense OUSD(R&E). (2023). *Department of Defense Mission Engineering Guide 2.0.* Washington, DC. Retrieved from https://ac.cto.mil/wp-content/uploads/2023/11/MEG_2_Oct2023.pdf

Department of Defense OUSD(R&E). (2023, October 19). Memorandum for Mission Engineering Executive Steering Council. *Enabling Digital Mission Architecture Integration Across the Department of Defense During Fiscal Year 2024*. Washington, DC.

Deptula, D. (2021, March 1). *Desert Storm at 30: Aerospace Power and the U.S. Military*. Retrieved from War on the Rocks: https://warontherocks.com/2021/03/desert-storm-at-30-aerospace-power-and-the-u-s-military/

Desert Shield/ Desert Storm. (2023, September 21). Retrieved from Naval History and Heritage Command: https://www.history.navy.mil/browse-by-topic/wars-conflicts-and-operations/middle-east/shield-storm.html

Federal Aviation Administration. (n.d.). *14 CFR §§ 91.151 - 91.162-91.165 Visual Flight Rules.* Washington, DC: FAA.

Federal Aviation Administration. (n.d.). 14 CFR §§ 91.167 - 91.195-91.199 Instrument Flight Rules. Washington, DC: FAA.

Hause, M. (2024, March 20). *Darth Vader's Secret Weapon: Implementing Mission Engineering with UAF.* Retrieved from OMG UAF Summit: https://www.omg.org/events/2024Q1/special-events/UAF-presentations/1415-Hoth-Mission-Engineering-Slides-2024-03-20.pdf

History.com Editors. (2022, November 1). *Persian Gulf War*. Retrieved from History.com: https://www.history.com/topics/middle-east/persian-gulf-war

International Organization for Standardization (ISO). (n.d.). ISO 3166 Country Codes. Retrieved from https://www.iso.org/iso-3166-country-codes.html

International Organization for Standardization (ISO). (n.d.). ISO/IEC/IEEE 15288 - System Life Cycle Processes.

International Organization for Standardization (ISO). (n.d.). ISO/IEC/IEEE 42010 - Architecture Description.

International Organization for Standardization (ISO). (n.d.). ISO/IEC/IEEE 42020 - Architecture Processes.

International Organization for Standardization (ISO). (n.d.). ISO/IEC/IEEE 42030 - Architecture Evaluation Framework .

International Organization for Standardization (ISO). (n.d.). ISO/IEC/IEEE 42042 - Reference Architecture.

Joint Chiefs of Staff. (2015, March 19). *UJT to JCA Mapping*. Retrieved from https://jdeis.js.mil/jdeis/jel/training/jca_ujt_mapping.pdf

Joint Chiefs of Staff. (2023, October 6). *UJTL Task Development Tool (UTDT)*. Retrieved from https://utdt.js.mil/home

Lange, K. (2018, January 9). *6 Things to Know About Operation Desert Storm*. (DoD News, Defense Media Activity) Retrieved from Military.com: https://www.military.com/history/operation-desert-storm-6-things-know

Martin, J. (2024, July). Enabling Enterprise Transformation Using Enterprise Architecture Principles and Concepts. *34th Annual INCOSE Symposium.* Dublin.

Martin, J., & Alvarez, K. (2023). Leveraging UAF for Mission Engineering. *INCOSE Symposium.*

Martin, J., Fairley, D., & Lawson, B. (2024, May 06). *Enterprise Systems Engineering*. Retrieved from SEBoK Wiki: https://sebokwiki.org/wiki/Enterprise_Systems_Engineering

No Magic, Inc. (2022, July 1). *2022x Version News*. Retrieved from Cameo Enterprise Architecture 2022x: https://docs.nomagic.com/display/CEA2022x/2022x+Version+News

No Magic, Inc. (n.d.). *Using Libraries*. Retrieved from UAF 1.2 Plug-In 2022x: https://docs.nomagic.com/display/UAF12P2022x/Using+libraries

Object Management Group. (2021). *United Architecture Framework (UAF) Version 1.2 Enterprise Architecture Guide for UAF.* Retrieved from https://www.omg.org/spec/UAF/1.2

Object Management Group. (2022). *Unified Architecture Framework (UAF) Traceability Appendix A Version 1.2.* Retrieved from https://www.omg.org/cgi-bin/doc?formal/22-07-07.pdf

Operation DESERT STORM. (n.d.). Retrieved from U.S. Army Center of Military History: https://history.army.mil/html/bookshelves/resmat/desert-storm/index.html

Shreve, G. (2024, March 20). *UAF or SysML - Yes?* Retrieved from OMG UAF Summit: https://www.omg.org/events/2024Q1/special-events/UAF-presentations/1615-UAF_SysML_Yes_Updated-2024-UAF-Summit%20v2.pdf

U.S. Air Force. (n.d.). *1991 - Operation Desert Shield/Desert Storm*. Retrieved from Air Force Historical Support Division: https://www.afhistory.af.mil/FAQs/Fact-Sheets/Article/458965/1991-operation-desert-shielddesert-storm/

U.S. Air Force. (2021). *Air Force Doctine Publication 3-60, Targeting.* Maxwell AFB. Retrieved from https://www.doctrine.af.mil/Portals/61/documents/AFDP_3-60/3-60-AFDP-TARGETING.pdf

U.S. Air Force. (2021). *Air Force Doctrine Publication 3-70, Strategic Attack.* Maxwell AFB. Retrieved from https://www.doctrine.af.mil/Portals/61/documents/AFDP_3-70/3-70-AFDP-STRATEGIC-ATTACK.pdf

U.S. Department of Defense. (2012). *Joint Publication 3-01, Countering Air and Missile Threats.* Washington, DC: Joint Chiefs of Staff.

U.S. Department of Defense. (2017). *Joint Publication 5-0, Joint Planning.* Washington, DC: Joint Chiefs of Staff.

U.S. Department of Defense. (2019). *Joint Publication 3-0, Joint Operations.* Washington, DC: Joint Chiefs of Staff.

U.S. Department of Defense. (2020). *2020 Department of Defense Electromagnetic Spectrum Superiority Strategy.* Retrieved from https://media.defense.gov/2020/Oct/29/2002525927/-1/-1/0/electromagnetic_spectrum_superiority_strategy.pdf

U.S. Department of Defense. (2022). *Joint Publication 2-0, Intelligence.* Washington, DC: Joint Chiefs of Staff.

U.S. Department of the Navy. (2023, May 22). *DMO Model Developers Guide Wiki*.

U.S. Department of the Navy. (2023, May 22). *Naval Federated Architecture Reference Modelling Wiki*.