MBSE Data Interoperability Specification Report

Process Use Cases and Data Exchange Criteria

Release 1.0

December 2020

AEROSPACE & DEFENSE PLM ACTION GROUP

Abstract

Phase 3 of the Model-Based Systems Engineering (MBSE) interoperability project evaluates the business context and tool alternatives for exchanging requirements and architecture representations depicting systems and system components. It attempts to identify the minimum content that must be exchanged based on popular authoring technologies. The project distinguishes the use cases that describe the exchange scenarios and the most appropriate technical data needed for interoperability. A mapping between two major modelling languages, SysML and ARCADIA, is defined for each use case. The outcome was a scoring of capabilities needed to implement a full-scale solution across a multi-tier supply chain. The project was based on the following problem statement and objective.

Table of Contents

Revision Record

Release	Date	Description
1.0	**December 2020**	Phase 3 - Report for Public Release, includes use cases, evaluation criteria, and a go forward plan

MBSE Data Interoperability Specification Report

Executive Summary

The Aerospace and Defense Product Lifecycle Management (PLM) Action Group (AD PAG) is an association of aerospace Original Equipment Manufacturers (OEMs) and aircraft engine manufacturers within CIMdata's globally recognized PLM Community Program, which functions as a PLM advocacy group.

The AD PAG sponsored a team of systems engineering domain experts to evaluate current data interoperability standards intended to enable a Model-Based Systems Engineering (MBSE) conceptual design process. The activity was to assess the feasibility of exchanging digital models, instead of documents, between aerospace & defense OEMs and their supply chain partners. To date, three project phases have been completed.

Phase 1 identified a gap in the capability of commercial SysML-based (Systems Modeling Language) authoring tools offered by the major PLM and MBSE software solution providers. The tools in their out-of-the-box configurations do not support bi-directional MBSE data and model exchange.

Phase 2 considered options to address those exchange issues in both the short and the long term. The recommended interim solution is for the aerospace and defense industry to jointly identify 1) independent, third-party software tools that can be used standalone or as adapters/plug-ins to the SysML-based authoring tools or 2) a services-based translation facility for MBSE data and model exchange. In the long term, the group recommends Application Programming Interface (API) interoperability improvements identified for future versions of the SysML standard and its implementation by the providers of MBSE architecture authoring tools.

Phase 3 involved the definition of targeted MBSE application use cases spanning the systems engineering V-model lifecycle (the WHY); the definition of what key elements of system architecture data that needs to be digitally created and collaboratively exchanged between aerospace and defense (A&D) OEMs and their suppliers/partners (the WHAT) and a high level evaluation of the current MBSE standards and commercial tools available to affect a digitally enabled conceptual design process (the HOW).

This technical report and a related summary PowerPoint presentation document the results of the Phase 3 project activities and include recommended actions for future industry efforts to achieve business value from the industry's digital transformation initiatives.

Given the current limitations in the data standards, a third-party translation service is still considered the most viable long-term solution for a positive aerospace industry-wide outcome.

Introduction

Digitization of the product development process from the early conceptual design stage all the way through the entire product lifecycle is a major business strategy across all industries today. This is especially true in aerospace and defense (A&D) programs, which are motivated to resolve the design complexities of cyber-physical systems with ever-increasing amounts of embedded software and electronics. Many economic business drivers are cited for achieving this end-to-end digital and model-based process, often referred to as the *digital thread*:

- Increased market share and profitability via faster time to market (i.e., shorter product design and development cycles)
- Innovation leading to unique product features (i.e., design features directly linked to meeting end-customer functional requirements)
- Increased enterprise engineering productivity and supply chain collaboration through the use of robust digital models versus documents throughout the product development lifecycle
- Improved collaboration in model-based design, fewer engineering change orders (ECOs), fewer physical prototype iterations
- Better product quality and reliability via continuous design validation and integrated requirements (i.e., optimized designs via digital modeling and simulation of performance before hardware)
- Reduction in total lifecycle costs, including manufacturing, warranty, and in-service operations (i.e., digital models evolve to meet other domain requirements - digital twins)
- Compliance with global industry regulations (e.g., safety certification, re-use/green, etc.)

Specific to the opportunities above, the A&D Original Equipment Manufacturers (OEMs) recognize that collaboration within a large, globally distributed supply chain of design and development partners is being seriously hindered by reliance on traditional, document-based development processes. To address this challenge, OEMs are expanding their use of digital and model-based software tools that define and manage the overall system requirements, associated system architectures, product safety, and regulatory/contractual obligations. To ensure an inclusive implementation, they have engaged their supply chains to closely collaborate through the exchange of conceptual digital models that communicate the design intent back and forth in a robust and iterative product development process that does not rely on traditional documents, singular artifacts, and/or drawings.

For this purpose, an AD PAG project team was formed to evaluate current data interoperability standards for enabling a Model-Based Systems Engineering (MBSE) conceptual design process based on digital models instead of documents. The AD PAG members represent a multi-national cross-section of engineering skills representing the OEM companies. Participation was on a voluntary basis, and the participation of at least three different companies was required to establish a forum that could make decisions. The meetings were conducted using the English language.

The AD PAG member companies listed below provided subject matter experts (SMEs), also known as *domain experts*, to participate in the evaluation. Those who were assigned have decades of aerospace PLM and configuration management experience.

- Airbus
- Boeing

- Embraer
- GE Aviation
- Pratt & Whitney
- Rolls-Royce
- Safran

CIMdata administers the AD PAG operations, coordinates research, and manages the progression of policy formulation. For this project, CIMdata was engaged to facilitate and assist with the planning and execution of the solution evaluation process. In total, the project team has more than twenty contributing MBSE domain experts from these companies, as well as a CIMdata representative.

The goal of the project team was to evaluate the current capabilities of an aerospace and defense OEM and its suppliers to develop and communicate a set of system design requirements, behavior models, and a corresponding system architecture using digital data modeling standards. The project team agreed that the data exchange standards for individual packages of requirements and behavior models are relatively mature[1], so the focus shifted to how to leverage the standards supporting the exchange of ADL (Architecture Description Language) compliant modeling languages.

In Phase 1, the data exchange exercises evaluated the feasibility of producing, exchanging, and consuming a model-based system architecture defined in a SysML-based (Systems Modeling Language) authoring tool. A corresponding set of requirements for a new system design was allocated to model features and elements. A very simple subsystem example was used, namely a light control system.

In the initial conceptual design scenario, the OEM sent a model-based request to suppliers to develop and define their solutions. The OEM would analyze the returning models and determine if the supplier's response communicated their ability to meet the stated OEM performance requirements. The conceptual design scenario mimicked how an OEM would solicit design proposals from suppliers with the intent of establishing a contractual relationship for design and development of a system or sub-system. The digital deliverables included the system specification—SysML or SysML-based architecture diagrams—and the associated design requirements. The OEM had the option to specify the initial digital exchange format, and the responding project team members could use any tool available to complete the digital data exchange. Our previously published position paper, *Model-Based Systems Engineering (MBSE) Data Interoperability*[2], provides an overview of the Phase 1 MBSE data exchange exercises and subsequent conclusions on the industry's state of readiness.

For the Phase 2 MBSE project activities, conducted during 2018, the goal of the project team was to agree on the most promising strategies and best practice for digital data exchange across the

---

[1] Support for the translation and co-simulation of behavior models created by different software tools is generally enabled by the FMI (Functional Mockup Interface) specification. RIF/ReqIF are the industry standards used to exchange product design requirements, and the associated metadata, between software tools.

[2] *Model-Based Systems Engineering Data Interoperability, Problem Statement, Assessment, and Go Forward Plan*, AD PAG Position Paper Release 1, January 2019.

aerospace and defense industry. Phase 2 was based on the current maturity level of the most suitable set of MBSE data standards (e.g., SysML, ReqIF, XMI, UMLDI[3]) and related MBSE data authoring tools. The previously published position paper reviews the various solutions considered by the group and makes initial recommendations of the most promising approaches to enable OEM/supply chain design collaboration based on MBSE standards, both short term and long term.

In summary, the primary short-term recommendation, based on the Phase 2 effort, was to focus the Phase 3 project activity on the identification and evaluation of one or more independent, third-party, software-based adapter tools and/or a tool-neutral MBSE data interoperability service. The tool/service would support model/data exchange between the most widely used ADL authoring tools, particularly those used regularly by the AD PAG member companies and those that are provided either as standalone, commercially available software products or by the major PLM/MBSE solution providers.

As demonstrated in the previous Phase 1 efforts, the common MBSE authoring tools based on SysML do not currently support a bi-directional and collaborative systems design process for OEMs and their supply chains. This may require the incorporation of other MBSE data interoperability standards, such as canonical XMI and UMLDI. But overall, the current MBSE data exchange standards were not considered to be adequate near-term options within the scope of the Phase 3 project activities.

Other MBSE architecture authoring tools, including open source software such as Capella, based on the ARCADIA framework, were also included within the scope of the Phase 3 activities.

Problem Statement

No standards-based tools support the exchange of digital system architecture models across the aerospace industry. The Aerospace OEMs and suppliers have not identified a common solution that enables their transition to a collaborative model-based business process.

Project Objective

The objective of this project is to evaluate, identify, and promote methods of exchanging digital content that includes the system architecture models.

[3] ADL – Architecture Description Language, or EADL – Executable Architecture Definition Languages
AADL – Architecture Analysis & Design Language, Carnegie Mellon University, https://wiki.sei.cmu.edu/aadl/index.php/Main_Page
Acme – Carnegie Mellon University, https://www.cs.cmu.edu/~acme/
Arcadia Capella – https://www.obeo.fr/en/capella-professional-offer
ArchiMate – Architecture-Animate, The Open Group, http://www.opengroup.org/aboutus
OPM – Object Process Methodology (ISO 19450), https://en.wikipedia.org/wiki/Object_Process_Methodology
Rapide – https://complexevents.com/stanford/rapide/
SysML – Systems Modeling Language specification, http://www.omgsysml.org/
UML – Unified Modeling Language specification, http://www.omg.org/spec/UML
UMLDI – Unified Modeling Language Diagram Interchange specification, http://www.omg.org/spec/UMLDI

About this Document

This section of the report describes the document's purpose and scope and identifies the model definitions and solution provider categories.

Purpose

This document is the final report on Phase 3 of the MBSE Data Interoperability Specification project, which encompasses the following structure, objectives, and deliverables:

- Formulate and document the most prevalent A&D industry use cases for MBSE system architecture data exchange between A&D OEMs and their global suppliers/design partners (WHY).
- Identify the minimum viable technical requirements and deliverables for a successful system architecture model exchange between A&D OEMs and their global design partners (WHAT).
- Based on the highest priority MBSE use cases (WHY) and the associated data deliverables (WHAT), develop a set of solution-scoring criteria that would enable the evaluation, ranking, and potential selection/implementation of commercially available MBSE data interoperability solutions (HOW). The combination of these viewpoints would meet some or all of the AD PAG's short-term business objectives.
- Based on completion of the Phase 3 research, develop an outline of recommendations with respect to a potential software/provider selection, and then solicit guidance from the AD PAG Leadership on how best to execute or utilize the recommendations.
- Define a Go Forward Plan that can be used to accomplish the collective goals and objectives of the AD PAG, based on the AD PAG Leadership's guidance.

To realize the above objectives and deliverables, the project team was split into three sub teams: the WHY, WHAT, and HOW. The sub teams worked in parallel, with the inputs and outputs of the three sub teams being synergistic to achieve the project objectives. The deliverables of the three sub teams are defined as follows:

WHY sub team – Airbus, Boeing, Bombardier, GE, Rolls-Royce

- Develop and prioritize detailed application use case scenarios and data exchange requirements for a successful OEM/supplier MBSE data exchange of system design intent (i.e., prioritize the exchanged architecture views and diagrams relative to the lifecycle).

WHAT sub team – Boeing, Airbus, Bombardier, GE, Rolls-Royce

- Review current industry MBSE standards specifications and modeling requirements.
- Define the minimum list of system architecture modeling elements necessary for a successful MBSE data exchange, based on the highest priority use cases.

HOW sub team – Boeing, Airbus, CIMdata, Embraer, GE, Rolls-Royce

- Develop a solution evaluation scoring model based on the implementation variables, the technical modeling criteria, and the relative importance of the MBSE data requirements.
- Identify and assess commercial solutions, based on web-based research. Characterize the commercially available software tools potentially capable of supporting an OEM/supplier MBSE data exchange, based on the use cases and requirements developed by the WHY and WHAT sub teams.

The following sections of this document reflect the combined work of the three sub teams. An MBSE-specific glossary is provided as Appendix C. An extensive AD PAG glossary is available for download at www.ad-pag.com.

Scope

This report describes MBSE use cases that represent a sampling of A&D MBSE processes. Inputs and outputs of these use cases are listed along with the corresponding modeling artifacts. A mapping to the SysML data standard modeling language is defined first, then ARCADIA is evaluated for comparison purposes and as a potential ADL alternative to SysML.

The system requirements are also part of the use case inputs and outputs but are presumed to be managed outside of the ADL models in documents or other software tools. Requirements management software tools, in-house developed custom tools, and/or office automation tools such as spreadsheets typically author and administer the authoritative view of system requirements today, even though SysML provides direct support through a Table or a Requirements diagram. This is in recognition of the "standard practice" used in industry of a separate authoring and configuration management system for requirements.

The document's last section identifies the scoring criteria used to evaluate an MBSE data exchange solution. The scoring criteria prioritizes the results from Use Case 3 and Use Case 4 as defined later in this document, using weighting factors applied to each of the decision criteria. The weighting factors were developed as a basis to engage with potential MBSE software tool providers and/or service providers. The evaluation criteria provide a framework to support potential future engagement with industry MBSE tool providers by AD PAG members, using a Request for Proposal (RFP) business evaluation and procurement process. To verify neutrality in the criteria categories, a scoring exercise was conducted by eight different project team members who were randomly assigned a potential product solution. Their scoring relied exclusively on information available in the public domain. A similar "hands on" exercise could be used to rank and down-select from a list of Commercial Off-the-Shelf (COTS) products that might represent a suitable near term MBSE data interoperability solution.

Model Definitions Used within this Document

The two model definitions used within this document are outlined below.

System Architecture Models

- The fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution (ISO/IEC/IEEE 42010)
- The organizational structure of a system or component and its implementation guidelines (ISO/IEC/IEEE 24765)
- System models created using an ADL-compliant tool as defined by ISO/IEC/IEEE 42010

Behavior Models

- The quantitative assessment of functional systems represented by environment models, System/Structural Plant Models, and/or regulated Control Loop models. These are lumped parameter models representing physics-based behaviors and controls. They are described by mathematical specifications or executable code, containing differential, algebraic, and discrete equations.
- The models are created in a dedicated digital authoring and simulation environment using MBD (Model-Based Design/Development) tools to evaluate complex equations that are not suited to or easily executed in a system architecture model.

Solution Provider Categories Used within this Document

The solution provider categories include PLM, ADL, and third-party interoperability suppliers and are described below.

PLM Software Supplier

A PLM software supplier is a provider of support for, or has the ability to combine, a comprehensive set of authoring tools and/or data management system(s) supporting the product development lifecycle (PLM = Product Lifecycle Management).

ADL Software Supplier

An ADL software supplier is the seller of standalone architecture authoring tool(s) that are ADL compliant. ADL language examples include the following: AADL, Acme, ARCADIA, ArchiMate, OPM, Rapide, SysML, UML.[4]

[4] ADL – Architecture Description Language, or EADL – Executable Architecture Definition Languages
AADL – Architecture Analysis & Design Language, Carnegie Mellon University, https://wiki.sei.cmu.edu/aadl/index.php/Main_Page
Acme – Carnegie Mellon University, https://www.cs.cmu.edu/~acme/
Arcadia Capella – https://www.obeo.fr/en/capella-professional-offer
ArchiMate – Architecture-Animate, The Open Group, http://www.opengroup.org/aboutus
OPM – Object Process Methodology (ISO 19450), https://en.wikipedia.org/wiki/Object_Process_Methodology
Rapide – https://complexevents.com/stanford/rapide/
SysML – Systems Modeling Language specification, http://www.omgsysml.org/
UML – Unified Modeling Language specification, http://www.omg.org/spec/UML
UMLDI – Unified Modeling Language Diagram Interchange specification, http://www.omg.org/spec/UMLDI

Third-Party Interoperability Software Supplier

A third-party interoperability software supplier is a seller of an integration service and/or software tool(s) that support the translation, exchange, or alternative representation of models generated from two or more brands of ADL-compliant authoring tools.

Industry and Technology Overview

The A&D industry is comprised of a wide range of multi-national corporations that specialize in the design, manufacture, and support of complex products and operational systems. A generalization of their product families includes both commercial and defense aircraft, space vehicles/satellites, weapon systems, propulsion/launch systems, and the supporting infrastructure. Although this project attempts to maintain an A&D focus, the same design authoring tools are used by multiple industries, including automotive, oil and gas, transportation, construction, medical, electrical-mechanical hardware, consumer goods, and more. Therefore, it is assumed that both the problem set and the potential solutions could be applicable to the design process of any complex cyber-physical product.

Business Realities for the A&D Industry

- DARPA, NIST[5], and AVSI estimate the interoperability opportunity cost to exceed one billion dollars per product across the lifecycle.
- Four major, enterprising PLM/MBSE software systems dominate the aerospace industry; however, the interoperability of their systems architecture authoring capabilities is limited
- Without model integration, the current industry solution is to exchange documents defining the logical architecture and associated requirements derived from proprietary behavior models.
- Without system architecture model exchange, leveraging the digital transformation of our supply chain has significant limits with no clear path to creating the digital thread/digital twin

Project Overview, Assumptions, and a Shared MBSE Vision

- Aerospace OEMs and Tier 1 and Tier 2 suppliers are invested in their own PLM systems and MBSE tool chains. (This assumes digital transformation is a common goal and each company's unique digital capability is a core competency.)
- OEMs use many of the same suppliers, and historically they have added to the suppliers' business costs by identifying specific tool brands, data schemas, and integration processes.
- Independent of the spatial design representations, there are three basic building blocks for the MBSE definition: the integration of Requirements, Behavior, and Architecture models
- Data exchange standards for Requirements and Behavior models are mature, readily available in the tools, and easily adopted. Exchanging architecture models has proven very difficult.

[5] Cost Analysis of Inadequate Interoperability in the U.S. Capital Facilities Industry, NIST GCR 04-867, National Institute of Standards and Technology, 2004

OEM – Supplier Collaboration, Multiple Capabilities, and Languages

The industry's Architecture Modeling Capabilities are estimated in Table 1. The tier designations are assigned according to a supplier's contribution to the final product. An OEM is described as a large system integrator and product owner with a wide range of design and manufacturing capabilities.

The Tier 1 suppliers are given responsibility to design and manufacture major product systems with expertise in many specialized technologies. The Tier 2 suppliers are selected based on their own detail design and manufacturing knowledge and competencies. Tier 2 suppliers generally specialize in specific hardware categories and fabrication capabilities. They support both Tier 1 suppliers and OEMs.

Table 1 - Variation in Modeling Capabilities

OEMs	Tier 1 Suppliers	Tier 2 Suppliers
80% SysML 10% ARCADIA 10% other	50% SysML 30% ARCADIA 20% other	10% SysML 20% ARCADIA 10% OPM 10% other 50% None

MBSE Interoperability Specification

In brief, the standard systems engineering design method begins with a stakeholder's requirements defining a product or service. These requirements are analyzed and decomposed into functional requirements and elements. The resulting product functions are synthesized into logical elements, which are then verified and validated by analysis and simulations. As depicted in Figure 1, this design cycle is iterated in the early design phases until a final product is fully conceptualized and integrated.

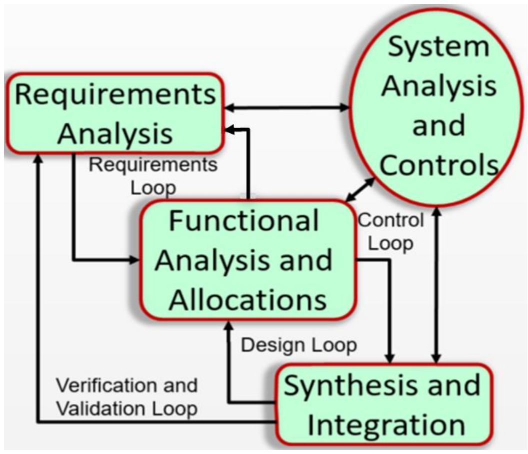

Figure 1 - Traditional System Design Process

In the conversion to a model-based paradigm, the same process is followed except that requirements are decomposed and applied to modelled elements. The models can be elaborated and translated into multiple views, which are then linked, synchronized, and integrated.

Although the models may be captured in a single tool and represented in separate diagrams, a minimum of three model types are needed to represent the traditional design process:

- The hierarchy, allocation, and integration view of the design requirements
- The system's functional and logical structure
- Validation of the product's behavior (with potentially separate models representing the functions, subsystems, components, connectivity, and product software behaviors)

Figure 2 is a simplistic view of the model types. In complex systems, the conversion to digital models extends the design capabilities needed to support thousands, even millions, of elements and relationships. In this case, evaluating the consistency of elements, attributes, and properties, as well as exploring design alternatives, is very difficult to execute and manage when relying exclusively on manually transcribed documents.

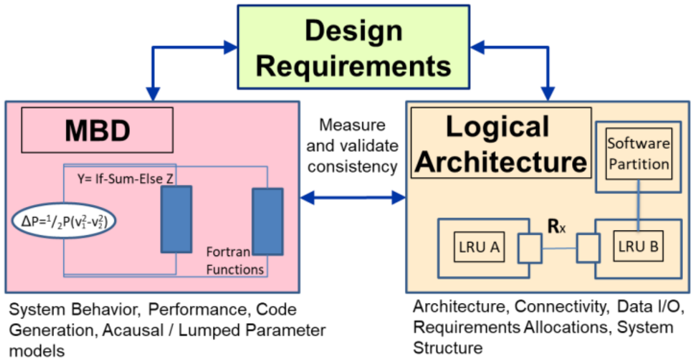

Figure 2 – Implementing basic MBSE Design Models

The transition to an MBSE process is significantly enhanced when common industry data standards are supported by the authoring tools, applied during execution (compilation), and/or used to link and synchronize the models. To validate the results, MBD tools simulate the behavior of the architecture model(s). The diversity of the data standards supporting MBSE is clearly indicated in the MBSE Roadmap[6] depicted in Figure 3.

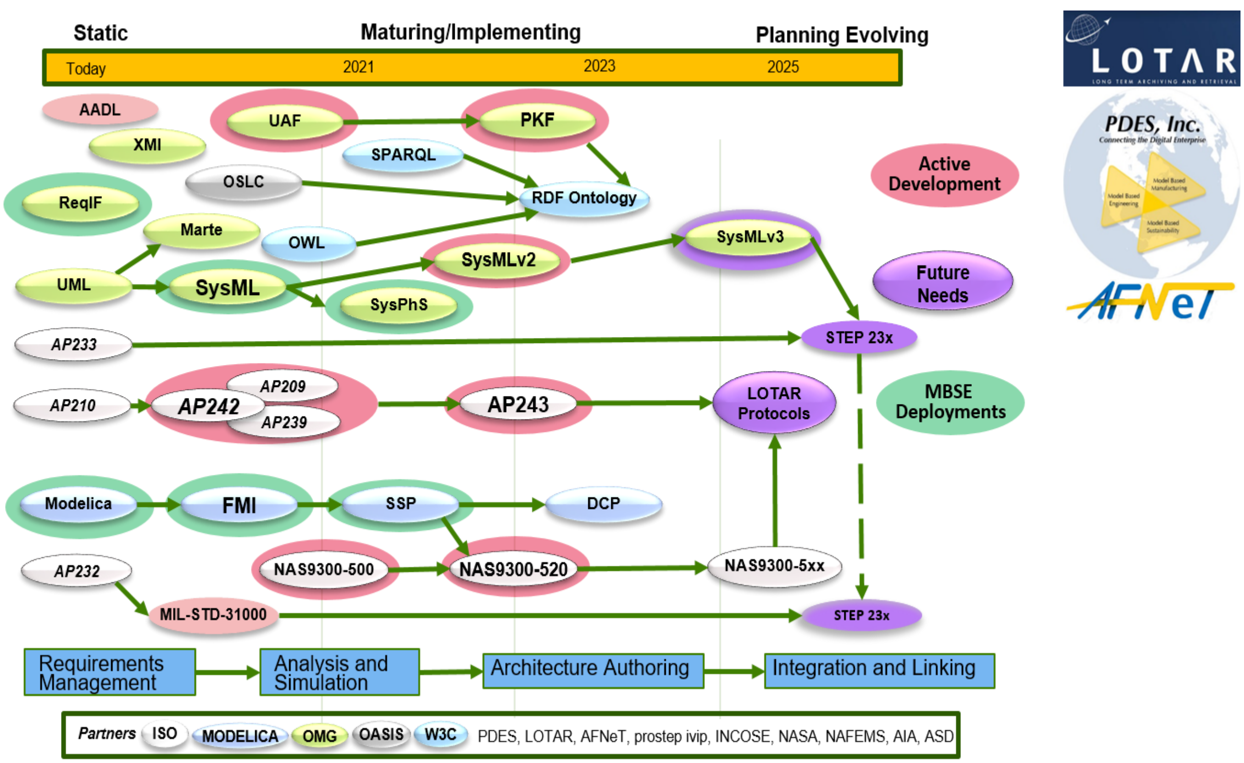

Figure 3 - MBSE Data Standards Roadmap

6 PDES-LOTAR MBSE Conference, May 8th, 2019. Revised Dec 11th, 2019. Reference ASD Radar Chart for detailed descriptions.

When applied to the basic MBSE model types depicted in Figure 2, the ability to exchange Design Requirements and Behavior models is well supported and readily available in tools that support these specific design domains. However, the subject of this report is the exchange and interoperability of system architecture models, and the original assumption was that the availability of standard languages, frameworks, and methodologies would make this a straightforward task. The previously mentioned Phase 1 results and Phase 2 report indicated otherwise.

Although SysML was the initial data standard target, the scope has grown to add a second popular modelling methodology implemented by the Capella tool. The ARCADIA framework is distinct from SysML in that it is both a methodology and a language, whereas SysML is just a language independent of predefined sematics. However, this does not prevent comparison or translations between SysML and ARCADIA model elements and diagram types.

As previously defined, the Phase 3 project intended to identify and evaluate an independent, third-party, software-based adapter tool and/or an MBSE data interoperability service. To accomplish this task, the project plan defined the following three deliverables:

1. Develop process-driven use case specifications for MBSE model exchange to enable OEM/supply chain design collaboration (WHY)
2. Extend the use cases to include all system architecture model interface needs, the design artifacts, and how to map the language alternatives (WHAT)
3. Evaluate MBSE model interoperability, including software supplier capabilities with respect to the use case requirements, and definition of maturity scores for the third-party tools (HOW). Understanding that the traditional method was to transmit and receive documents and images comprised of graphs, tables, and text descriptions, the team defined the following three modes for data exchange:
    - Basic Model Exchange – Usually defined as contractual requirement. A one-way transmission of specific content. (Prevalent for sharing 3D CAD content, and limited capability for other model types.)
    - Interoperability – Models are exchanged, edited, and re-shared between companies. Assumes that multiple versions may exist. (several examples in aerospace, but common in automotive industry by enforcement of common tools)
    - Collaboration – One model version is maintained as the master and accessible to both companies. (The marketing vision of PLM software suppliers, but licensing issues when mixing brands will require mature data standards are the basis of all shared models.)

Architecture Modelling Options and Comparisons

Due to the multiple ADL authoring tools available, it was necessary to compare differences in their implementation languages and modelling methodologies. Although the user interfaces may be different, the tools supporting the SysML language specification are generally understood, due to the formal specification as an extension to UML. However, ARCADIA is strikingly different, and its popularity required a closer examination of its major elements and capabilities. Despite its distinctions, the alternatives evaluation criteria and scoring, as defined in Appendix B: Criteria/Weighting Scores Extract, were not significantly impacted by the contrasts between SysML and ARCADIA.

Refer to Appendix A: SysML/ARCADIA Diagram Mapping for a table of mapping between SysML diagrams and their model elements to equivalent diagrams and model elements in the ARCADIA framework.

SysML Diagram Type Identification

Systems Modeling Language is a general-purpose modeling language for systems engineering applications and complex designs across multiple domains. The first open source SysML specification was released as SysML 1.0 in November 2005. Implemented as a UML profile, the language is very extensible and executes a major portion of the systems engineering data representation defined by ISO 10303-233. The SysML diagram taxonomy is composed of nine diagram types for mapping language elements across diagram types as shown in Figure 4.

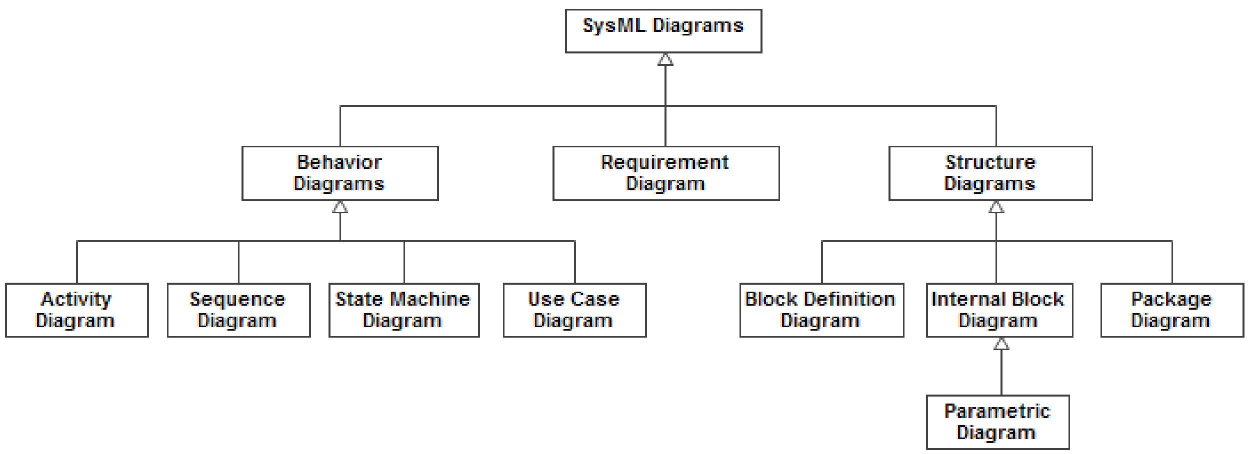

Figure 4 – Overview of SysML Diagram Types

ARCADIA Diagram Type Identification

Popularity of the ARCADIA methodology has grown rapidly; therefore, it has been included in this MBSE interoperability study. ARCADIA and the Capella authoring tool were both developed and open sourced by Thales in 2015. The language is open source, extensible, and distributed as a no cost extension of Papyrus UML. It has been adopted by leading PLM software suppliers.

ARCADIA is both a framework- and a domain-specific modeling language[7]. As illustrated in Figure 5, its diagram options are particularly suited for large complex mechanical systems where emergent behavior is prevalent. It is unconstrained by Object Oriented (OO) principles of encapsulation, aggregation, and composition and supports hierarchical architectural decomposition of complex systems.

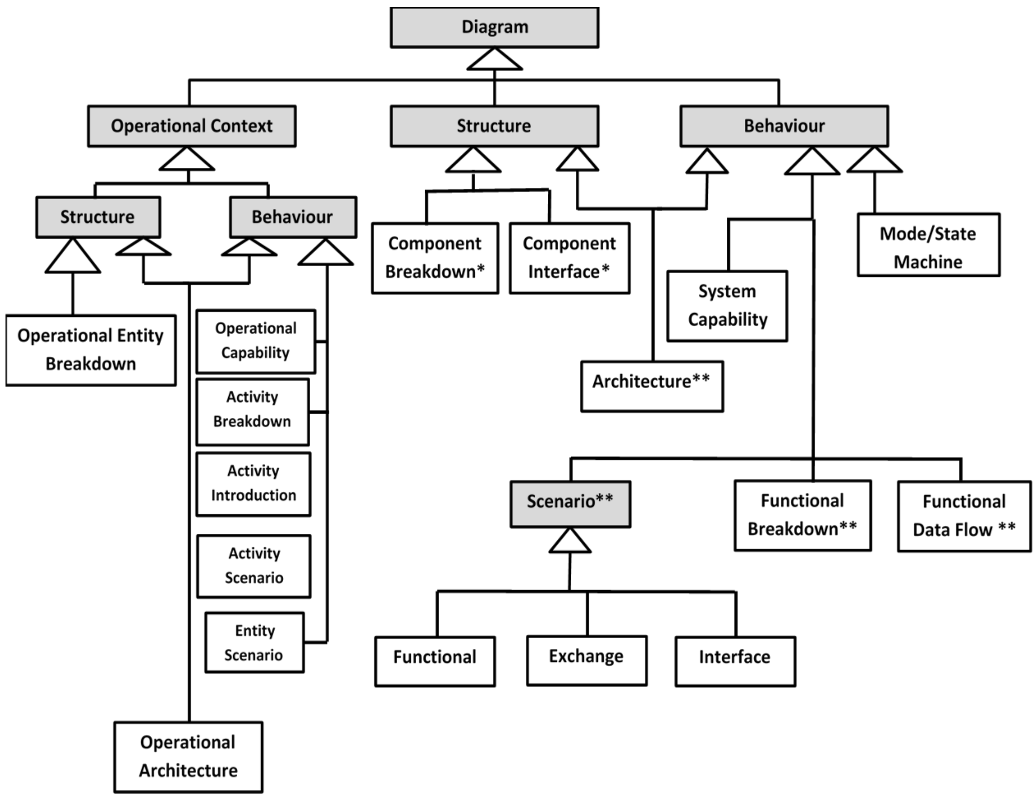

Figure 5 – ARCADIA Taxonomy Diagram

** Available in the following ARCADIA levels: Logical Architecture, Physical Architecture*

*** Available in the following ARCADIA levels: System Analysis, Logical Architecture, Physical Architecture*

[7] Representational state transfer (REST) is a W3C architecture that defines a set of constraints to be used for creating Web services and interoperability between computing systems. It is a stateless protocol that invokes a predefined list of standard operations.

MBSE Use Cases

The team identified five use cases along the classical systems engineering V-model development process where a formulized information exchange in the form of models between different development entities is needed.

- The first use case explored a System of Systems and Transitioning the Functional Interfaces to Logical Systems.
- The second use case defined the System Operational Scenarios.
- The third use case described how to Export System Functional Specifications. It also combines important information coming from Use Cases 1 and 2, such as operational scenarios, into one delivery package to be forwarded to the supplier.
- The fourth use case pre-staged the Validate and Verify Process using what is often referred to as an *observer model* for system functional specification and supplier product validation.
- The fifth use case defines an equipment specification package, which includes the Hardware and Software Functional Specifications, for the supplier.

In summary, Use Cases 3 and 4 represent the most challenging and relevant use cases with respect to OEM and supplier collaboration for model-based collaboration related to system architecture definition. Both address the model-based information flow between the OEM and the supplier, which is where a consistent modelling environment cannot be ensured without an interoperability strategy and plan. The goal and result are the identification of the minimum set and appropriate diagrams for exchange and use by the OEM and supplier.

Overall MBSE Process

The specification of dedicated use cases for an overall MBSE process requires input and agreement from the primary stakeholders.

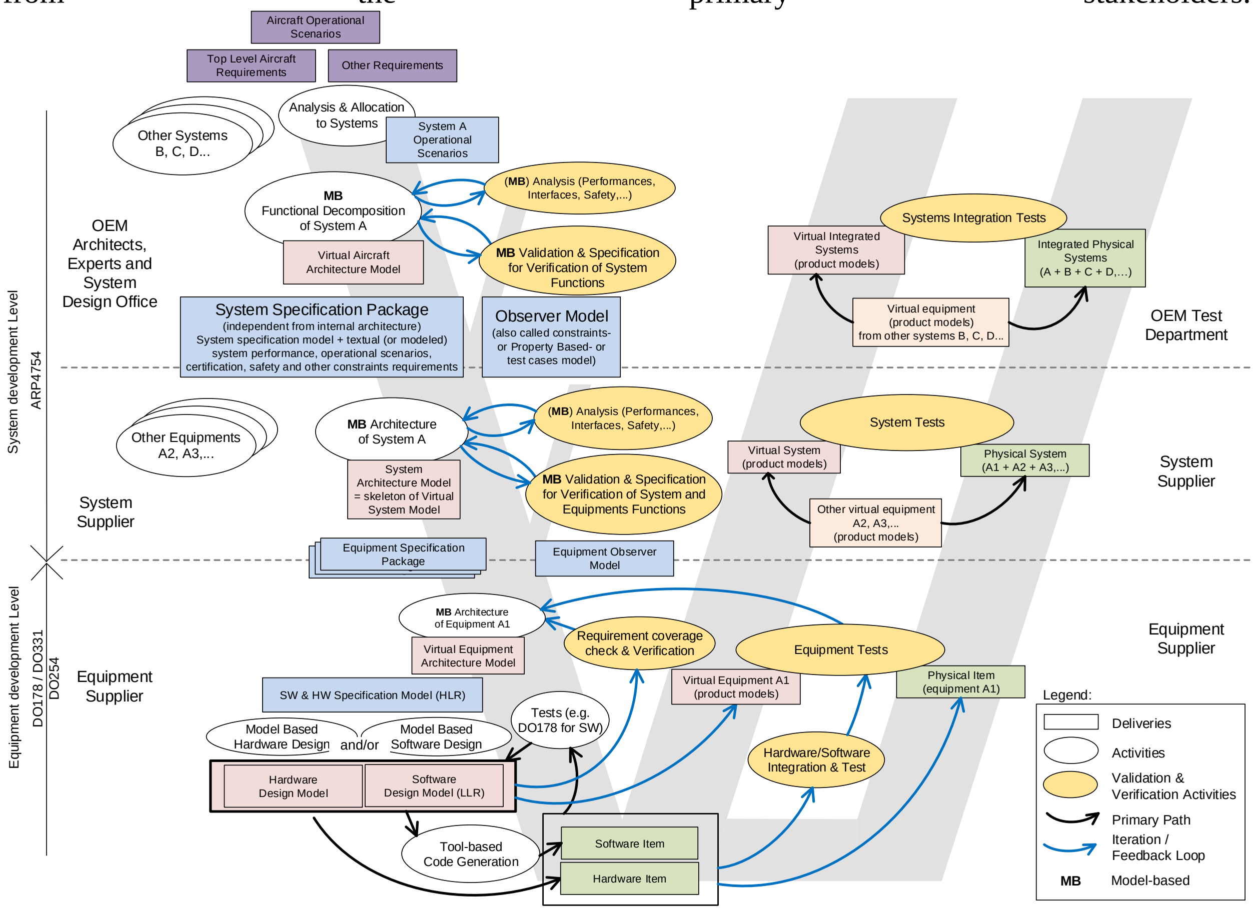

Figure 6 shows the overall MBSE process mapped to the classical systems engineering V-model as the baseline for the use case definition.

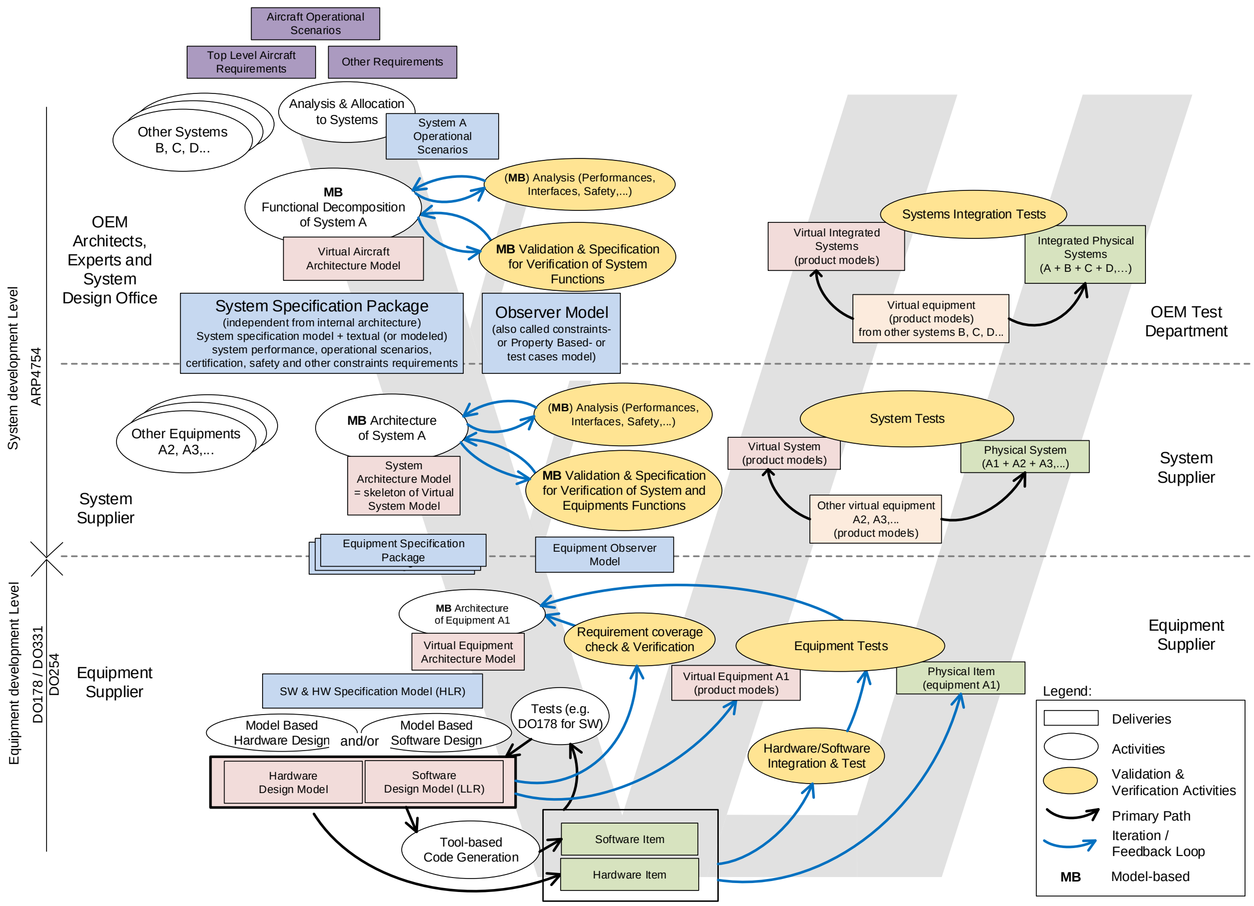

Figure 6 - Overall MBSE Process Definition

The system development lifecycle process consists of three basic activities:

- Specifying and designing the system itself
- Verifying and validating that system design
- Managing the overall development project

When developing a system of systems (SoS), the process is recursively applied at each level of the system hierarchy (i.e., systems, elements, components) until specifications are available for components and parts. The initial process can be divided into three phases:

- Conceptual phase
- Preliminary design phase
- Detailed design phase

To identify the deliverables of the different steps within the overall process, as shown in Figure 1, the following subsections contain different use cases that describe the activities and deliverables in a top-down process. For certain systems, the OEM will have the domain knowledge and will maintain the architectural accountability and responsibility. In this case, the OEM defines the system specifications and the only external deliverables are the Equipment Specification Package and the Equipment Observer Model. This is the condition assumed for the following simple use cases.

The alternative scenario is when the supplier is the best technology source and provides many of the details comprising the system specification, or even defines an alternative to the original specification. In this scenario, the supplier makes significant contributions to the system architecture, the verification and validation models, and the design requirements. The need for data exchange significantly increases, as does the complexity of the models and their integration.

Use Case 1: System of Systems and Transitioning the Functional Interfaces to Logical Systems (UC1)

As a system engineer, I want to decompose the system of systems into manageable system models. Systems and components are considered "black box" parts of the overall system of systems. The transition shall include functionality, interfaces, and non-functional requirements. Refer to Figure 7.

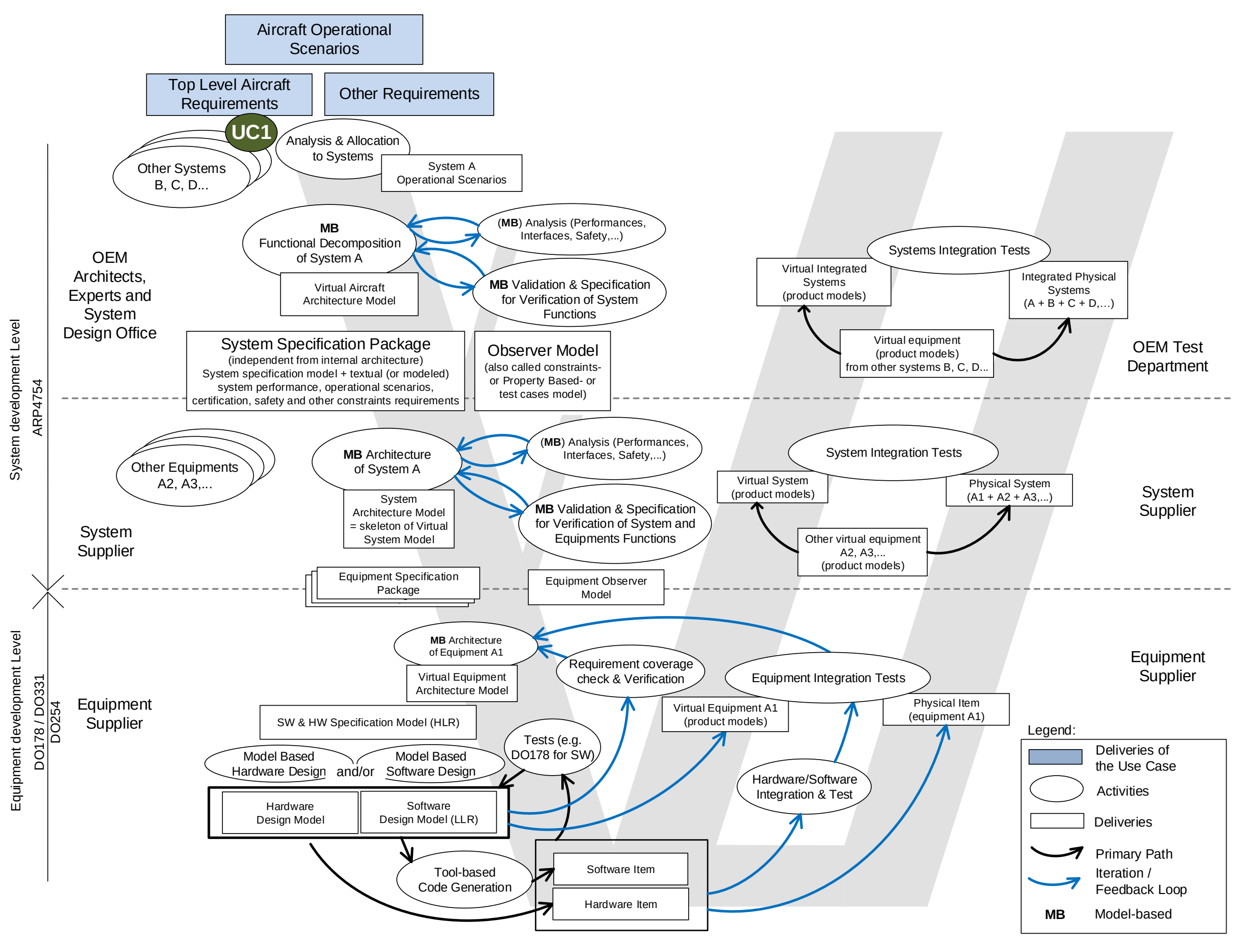

Figure 7 - Use Case 1: System of Systems (Functional & Interface) Transition to Individual Logical Systems

Preconditions/Inputs: Specification of system of systems, design requirements, system context, critical system properties and design constraints, ICDs (Interface Control Definitions), requirements for operational scenarios and use cases

Activities:

Table 2 - Use Case 1 Required Diagram Types

Origin	Artifact	ARCADIA Diagram	SysML Diagram
UC1	System function allocation	Logical Architecture diagram Physical Architecture diagram Mode State Machine diagram (+ Operational/System Analysis diagrams to capture precondition/inputs: Aircraft Operational Scenarios)	Activity diagram Block Definition diagram Internal Block diagram Sequence diagram State diagram

Post conditions/Outputs: A set of models for the overall system of systems logical architecture, with models for individual systems and components that describe the allocated functional requirements, interfaces, and state definitions for the overall system

Use Case 2: Define System Operational Scenarios (UC2)

As a system engineer, I want to be able to better understand the operational context of the system to be developed. For that reason, the system operational scenarios shall be specified to describe the necessary system behaviors for all modes and possible conditions of its intended operation. Such scenarios are operational scenarios in ARCADIA or SysML use cases. Refer to Figure 8.

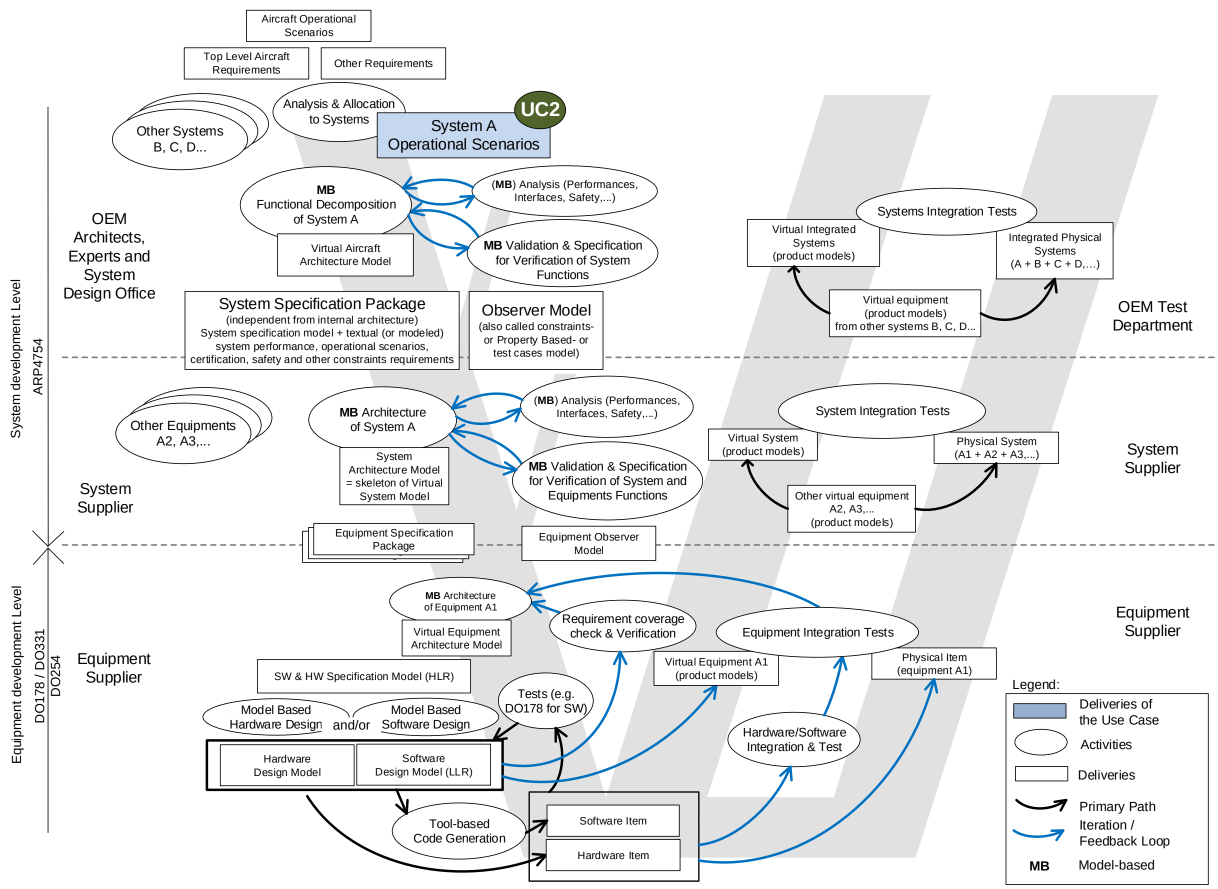

Figure 9 - Use Case 2: Specification of System Operational Scenarios

Preconditions/Inputs: Models for individual systems and components that describe the allocated functional requirements, interfaces, and state definitions for the overall system

Activities:

Table 3 - Use Case 2 Required Diagram Types

Origin	Artifact	ARCADIA Diagram	SysML Diagram
UC2	Operational scenarios	SA Level - Mission Capabilities diagram SA Level - Data Flow diagram SA Level - Functional Scenario diagram SA Level - Exchange Scenario diagram Mode State Machine diagram System Architecture diagram	Use Case diagram Activity diagram Sequence diagram Block Definition diagram Internal Block diagram State Machine diagram

Post conditions/Outputs: System operational scenarios that describe the behavior of the system under development in its contextual usage

Use Case 3: Export System Functional Specifications (UC3)

As a system engineer, I want to deliver a final system specification package so the supplier can proceed with the system design. Figure 10 highlights where the deliverable is located within the V-model. Use Case 3 describes the minimum deliverables that would normally be exchanged between the OEM and the system supplier. Refer to Figure 10.

The following information describes the "functional decomposition of System A", including the input, and output results representing Use Case 3.

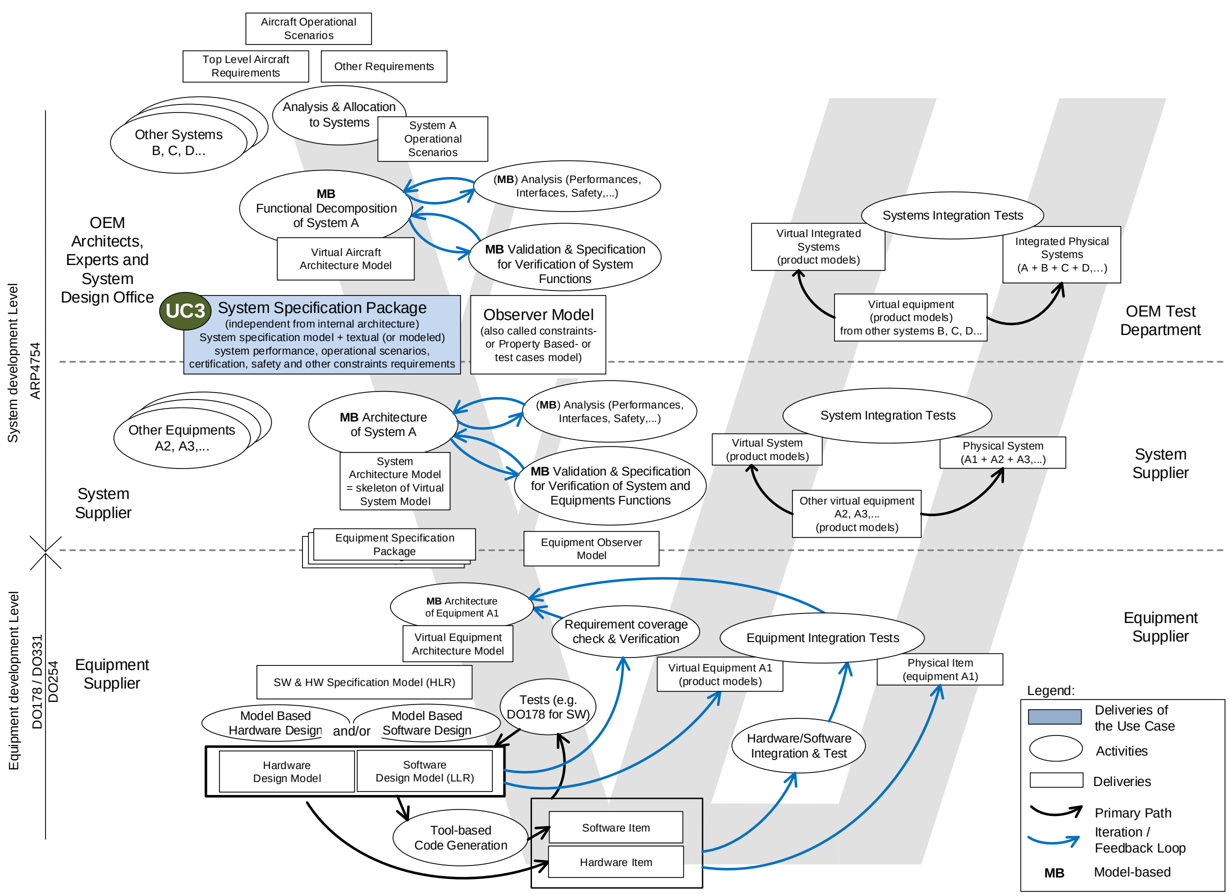

Figure 10 - Use Case 3: Delivery System Specification Package

Preconditions/Inputs (delivered by other uses cases):

- Functional top-level requirements
- System operational scenarios
- Clear identification of non-functional requirements, such as performance and certification requirements
- External interface definitions

Activities:

The activity shown as "functional decomposition of System A" contains, as its major part, the definition of the System Specification model. The System Specification model includes an executable functional model of the nominal system behavior in terms of defined inputs, outputs, states of the system, as well as performance, design, and safety constraints. The summary representation may be defined textually or modelled from the viewpoint of an observer, depending on the purpose and needs of the project. The System Specification model is used to unambiguously state the system requirements.

The activity of Use Case 3 requires the development of diagram types as described within Table 4.

Table 4 - Use Case 3 Required Diagram Types

Origin	Artifact	ARCADIA Diagram	SysML Diagram
UC3	Boundary diagram	System/Physical Architecture diagrams	BD + IBD diagrams
	Interface diagram	System/Physical Architecture diagrams	BD + IBD diagrams
	Context diagram	System Architecture diagram	Use Case diagram Activity diagram BD + IBD diagram
	Functional Flow diagram[1]	Functional Chain diagrams** Functional Data Flow diagrams**	Activity diagram Internal Block diagram
	Behavioral (white box temporal) diagrams	Model State Machine Diagram Functional Scenario diagrams** Entity Exchange Scenario diagrams** Interface Exchange Scenario diagrams** Architecture diagrams (Parametric viewpoint)**	State diagram Sequence diagram Parametric diagrams
	Logical architecture[1]	Logical Architecture diagram Logical Component Breakdown diagram	BD + IBD diagrams
	Physical (system) architecture[1]	Physical Architecture diagram Physical Component Breakdown diagram	BD + IBD diagrams Package diagrams

[1] The extent to which these artifacts are developed depends upon the OEM – system supplier relationship and the type of system under development. Where domain knowledge resides with the system supplier, these artifacts are likely to be less developed to protect proprietary knowledge. (Note: The ARCADIA Sub-system transition allows for model generation with different ARCADIA levels populated, i.e., commensurate with the domain knowledge of the system supplier).

******_Available in each of the ARCADIA levels (System Analysis, Logical Architecture, Physical Architecture)_

As is shown in the previous table, for several artifacts there is more than one ARCADIA or SysML diagram that can be used to represent the artifact.

In support of a minimal, optimized solution, the project team has prioritized the following diagrams. The list that follows is the minimum set of diagrams needed to represent all of the artifacts for Use Case 3.

Table 5 – Prioritized ARCADIA and SysML Diagrams

ARCADIA	SysML
Component Breakdown diagram*	Block Definition diagram
Component Interface diagram*	Internal Block definition diagram
Architecture diagrams**	Activity diagram
Functional Data Flow diagram**	Sequence diagram
Functional Scenario diagram**	

** Available in each of the following ARCADIA levels: Logical Architecture, Physical Architecture*

*** Available in each of the following ARCADIA levels: System Analysis, Logical Architecture, Physical Architecture*

Post conditions/Outputs:
System Specification Package (independent of internal architecture):

- System Specification model (may include parametric diagram)
- Operational scenarios
- Textual (or modelled) system performance
- Certification, safety, and other constraints requirements

Use Case 4: Pre-Stage the Validate & Verify Process and Co-Develop Behavior Models (UC4)

As a system engineer, I want to be able to validate and verify my model-based functional specification. The outcome shall be a statement of how the functional behavior was validated and a report or other output of the compliance and completeness of the system specification as defined by Use Case 3. The observer model, defined in Figure 12, specifies the system context. It provides the different inputs to the information scenarios and specifies the expected system output behavior. The use of observer models requires executable system specifications to interact with and validate. Because different validation scenarios are needed to describe the expected system behavior, the observer models are forwarded to the supplier for product validation before delivery. The Validate & Verify (V&V) activities shall be defined independently of the model-based system specification Refer to Figure 11.

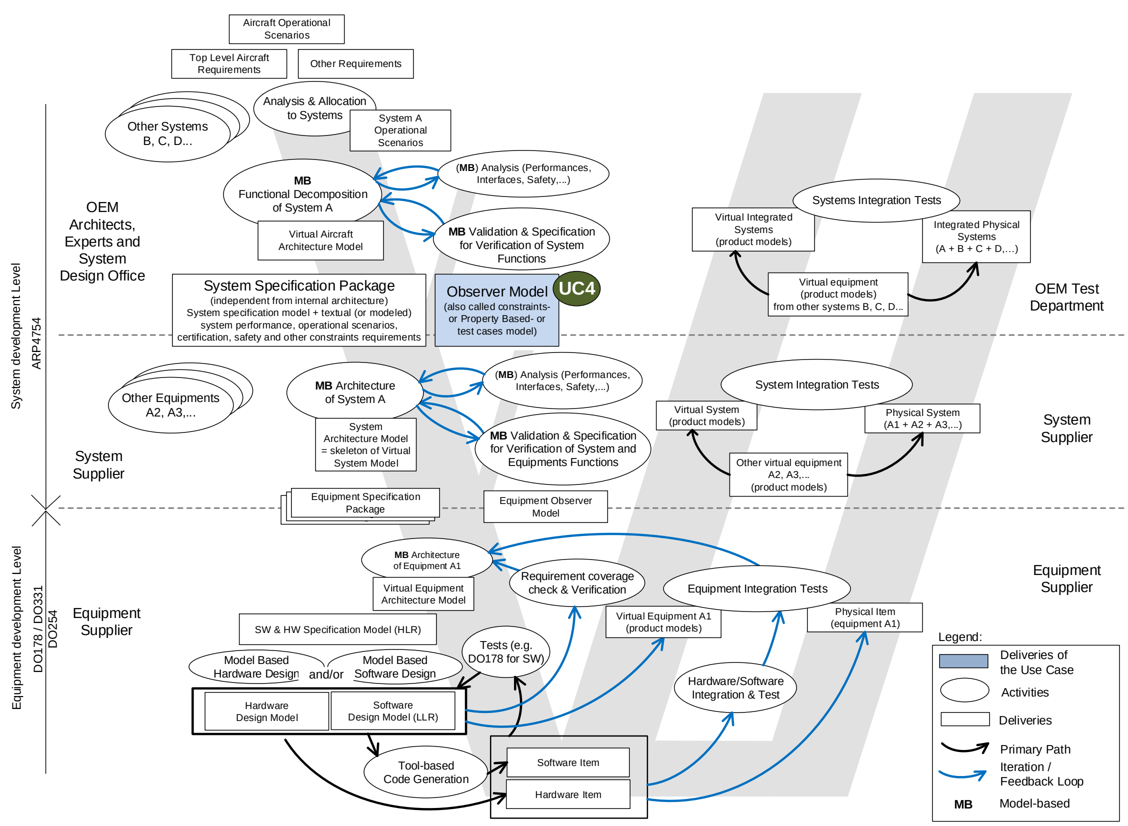

Figure 12 - Use Case 4: Validate & Verify System Functional Specification Model-Based

Preconditions/Inputs:

- Specification of system top-level requirements
- Modeling of operational scenarios and use cases
- Modeling of an executable system functional specification
- Modeling of system validation scenarios

Activities:

Table 6 - Use Case 4 Required Diagram Types

Origin	Artifact	ARCADIA Diagram	SysML Diagram
UC4	V&V Process	Entity/Functional Scenario diagrams Logical/Physical Architecture diagrams (Parametric viewpoint) Mode State Machine diagram	Sequence diagrams Parametric diagrams State diagrams

Post conditions/Outputs: The V&V report shall verify how the requirements and the architecture modeling relate to the behavior modeling and demonstrate, through simulation or other means, that the modeling behaviors perform according to the parameters provided from the requirements and architecture model content. The outcome includes collaboration with the supplier and co-development of the behavior models.

Use Case 5: Export Hardware/Software Functional Specifications (UC5)

As a system engineer, I want to deliver an equipment specification package so that the supplier can proceed with the detailed system design, which specifies the software and/or hardware solution for the equipment. Figure 13 shows where the deliverable is located within the V-model. The process involves co-development of an ICD that defines the network configuration, the signal interfaces, the protocols of each interface, and the software communications and messages.

Three important deliverables produced by the design process include the following:

- Functional Hazard Assessments (FHAs) that define the component/software/system failure modes
- Fault Isolation diagrams/Trees
- Identification of the associated operational conditions that must be considered when analyzing the system's response to the failure modes

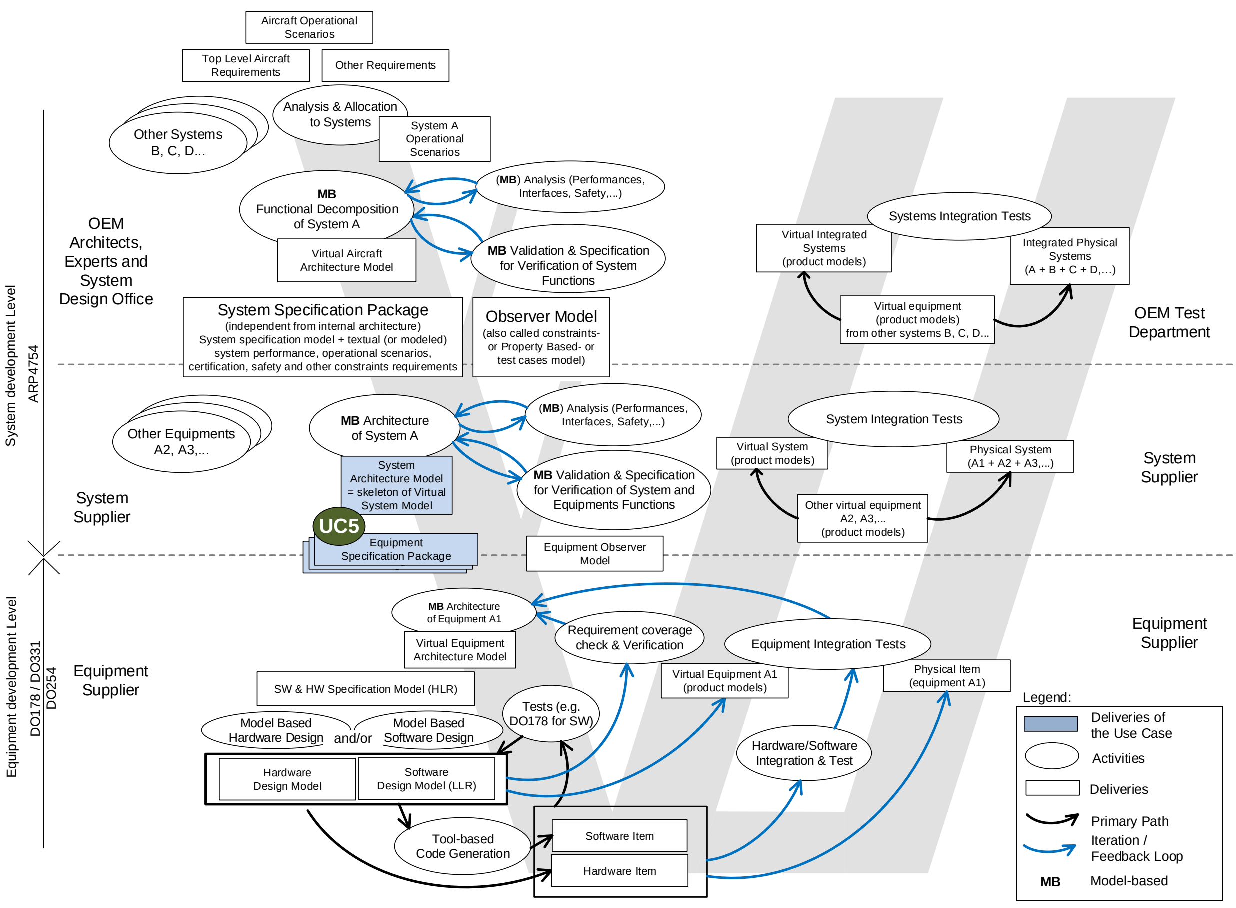

Figure 13 - Use Case 5: Equipment Functional Specification Package

Preconditions/Inputs:

- System Specification model
- Behavior models and derived textual descriptions of system performance
- Operational scenarios and simulations of the scenarios
- Certification, safety, and other constraining requirements

Activities:

Table 7 - Use Case 5 Required Diagram Types

Origin	Artifact	ARCADIA Diagram	SysML Diagram
UC5	Interface diagram	Physical Architecture diagram	BD + IBD diagrams
	Behavioral (white box temporal) diagrams	Mode State Machine diagram PA Level - Entity/Functional Scenario diagrams	State diagram Sequence diagram Parametric diagram
	Physical (system) architecture	Physical Architecture diagram	BD + IBD diagrams

Post conditions/Outputs:

- Equipment Specification model
- Textual (or modelled) equipment performance
- Certification, safety (Functional Hazard Analysis (FHAs) and System Functional Test (SFTs)), and other constraining requirements
- Physical architecture and system/software/equipment behaviors

Use Case Summary of Critical Diagram Types for Interoperability

Based on the objectives of the project and the broad scope of the lifecycle use cases defined, the MBSE project team identified Use Cases 3 and 4 as the highest priority for the more in-depth evaluation. The team focused on defining the minimal set of system architecture definition diagrams that are required, using both the SysML and the ARCADIA languages in order to enable a bi-directional model-based collaboration process between an OEM and its suppliers (i,e., the WHAT).

SysML Interoperability – High Priority Use Case Diagrams

- Block Definition diagrams
- Internal Block diagrams
- Activity diagrams
- Sequence diagrams
- Parametric diagrams
- State diagrams

ARCADIA Interoperability – High Priority Use Case Diagrams

- Component Breakdown diagrams
- Component Interface diagrams
- Architecture diagrams
- Functional Data Flow diagrams
- Functional Scenario diagrams
- Entity/Functional Scenario diagrams
- Logical/Physical Architecture diagrams
- Mode/State diagrams

MBSE Interoperability Solutions Evaluation

Based on the widespread use of the SysML and ARCADIA languages, the team conducted a “paper study" of the COTS software solutions available to A&D companies in support of MBSE model interoperability. This study did not include any hands-on use of the COTS tools or any in-depth benchmarking activity. The study relied on the review of commercially-published information about the third-party MBSE interoperability solutions, augmented by personal experience and knowledge shared by the MBSE project team subject matter (domain) experts. The published/known capabilities of each solution were evaluated and ranked against a set of requirements and scoring criteria developed by the project team.

Interoperability Options

Point-to-point model data translation (SysML <–> ARCADIA or SysML <–> SysML) is possible; however, the following must be considered:

- A translation capability based on the current SysML standard (v1.6) is not a long-term solution (although, identifying an interim capability could be cost effective).
- SysML v2 is a paradigm shift from the current SysML standard (v1.6), and translation systems would be fundamentally different.
- As represented in Figure 14, SysML v2 will offer multiple data interoperability options. We assume it will be at least two years (2022?) before the industry deploys the first interoperability alternatives based on SysML v2.
- The SysML v2 solutions do not guarantee data exchange. The specification options include exposing a tool's API, provisions for RESTful services1, or incorporating OSLC2 (Open Services for Lifecycle Collaboration) support for a "data linking" solution.

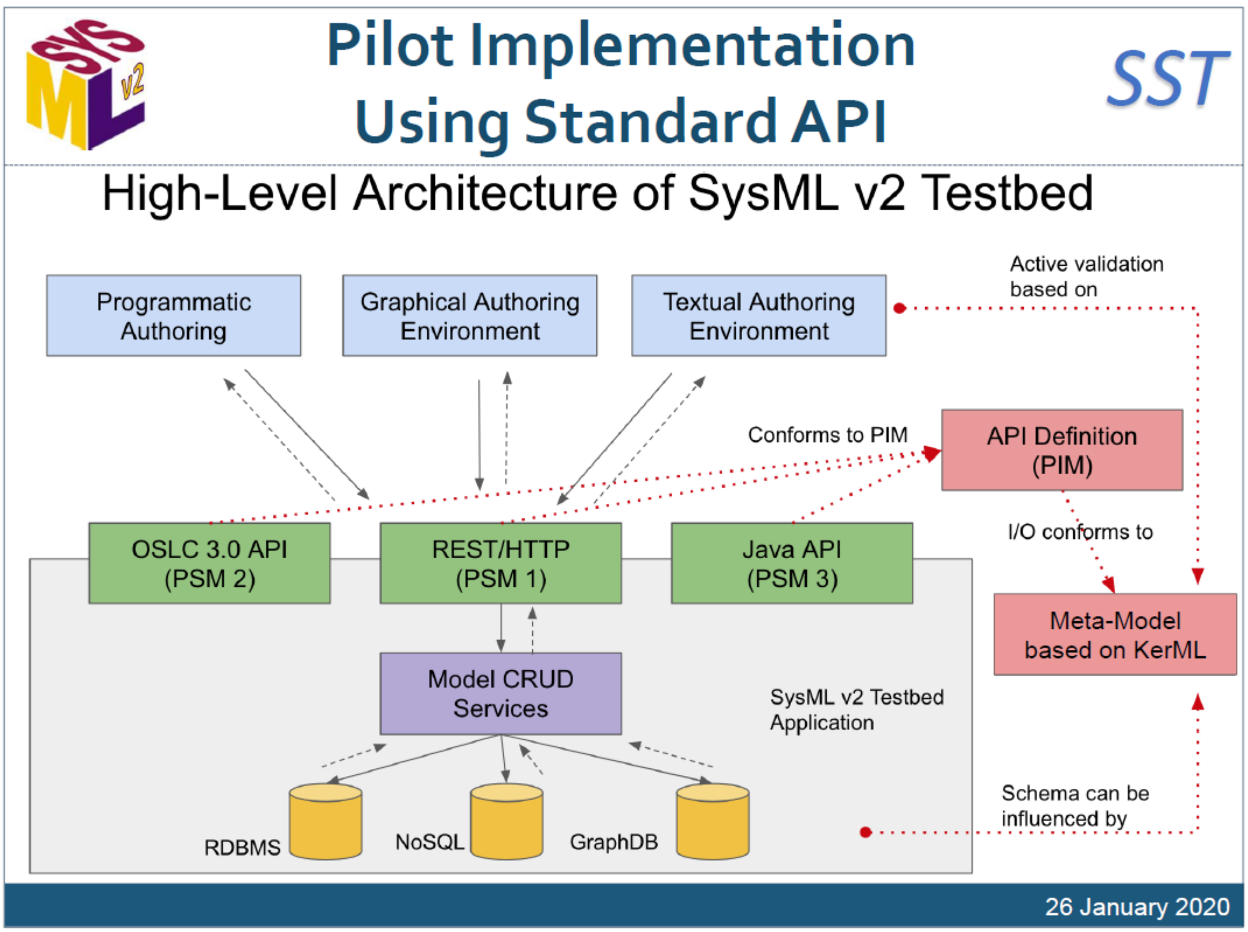

Figure 14 - SysML v2 Interoperability Options[8]

[8] Sandy Friedenthal, INCOSE IW2020 presentation

Exploration of Third-Party Capabilities

The HOW sub team conducted a paper analysis (i.e., “paper study”) of potential alternatives and solutions. A sample is depicted in Figure 13.

- Evaluated twelve products comprising two categories of capability (categories include a point-to-point translation or a federated architecture using an integration database)
- Did not identify a dominant COTS solution with expected functionality
- Did not identify an easy path to benchmark and validate use cases
- Could not agree on a common business case between project team members and instead decided to score three different scenarios

Summary

			Experienced ADL Exchange Companies/Tools			Engineering Services	
Companies			Company1	Company2	Company3	Company11	Company22
Products			Product1	Product2	Product3	Product1	Product2
CRITERIA	weight	describe criteria					
Language							
SysML	1000	target language	100%	100%	100%	50%	Custom Service
Max score:	43100	Total score:	23000	19500	800	14500	21000
		Percentage of maximum score:	52.00%	45.00%	18.00%	33.00%	48.00%
		How many criteria scored:	54	60	21	47	62

Figure 15 - Sample Alternatives Analysis

Based on the knowledge of the MBSE data exchange requirements, the capabilities of existing tools, and the diversity of the products available in the authoring tools industry, the sub team (HOW) developed a list of evaluation criteria and a scoring model by which each of the current potential MBSE interoperability solutions could be evaluated. The evaluation criteria and proposed scoring model were verified using a “virtual” assessment of existing providers. Each potential provider was scored against the solution criteria to simulate a packaged capability that would provide the optimum functionality to the AD PAG (see Appendix B: Criteria/Weighting Scores Extract).

In parallel with the activities of the other two sub teams, the HOW sub team investigated how to develop a list of commercially available MBSE interoperability software and service solutions. Multiple commercial and academic resources were used to derive a directory. Project team members were then assigned a company to research, via data available in the public domain, to gain a better understanding of the capabilities of the different MBSE interoperability software solutions. The investigator presented the results to the project team, which then discussed, modified, and extended the assessment categories, criteria, and related descriptions.

The HOW sub team also solicited input from other AD PAG project team members and industry colleagues who had experience with or knowledge of the existing list or potential MBSE providers and their software solutions. Based on these interviews, the project team concluded that on-going research and development by these companies deferred a conclusive assessment and identification of a clear front-runner. New providers and products continue to emerge across a wide range of industries due to the rising interest in MBSE authoring tools.

Although very few solution providers publicly identified an off-the-shelf interoperability solution between Capella and other SysML-based authoring tools, multiple results of commissioned projects were identified. This provided more evidence that the environment is rapidly changing, indicative of company mergers, acquisitions, partnerships, market share, and overall domain growth that is decreasing the number of customized ADL products and increasing interest in the use of data exchange standards or tailorable interoperability solutions.

The sub team did not thoroughly evaluate the use of a consulting or services company to host a secure OEM/Supplier MBSE data exchange environment. It was assumed this solution would forego an automation product and rely on a customized pay-as-you-go business model. Multiple companies are marketing the potential to navigate the most complex exchange scenarios. However, notional differences in the cost model and associated performance were difficult to analyze and correlate with a COTS-based user-controlled solution. The guidelines agreed to and used by the combined sub teams were to work toward a future optimized study, with the intent to use test lab conditions that verify the minimal and most important content that needs to be exchanged.

In the future, a practical solution may combine a COTS product with a consulting service and represent an optimized solution for data packages that contain a mix of modelling styles, including both linked and integrated data. Therefore, the services-based approach will require additional consideration if the software solution providers are not able to provide an adequate out-of-the box result that meets the short-term business requirements of the AD PAG.

While five potential MBSE use case scenarios across the V-model lifecycle were documented during this project phase, the final recommendations are focused on Use Case 3 (Export System Functional Specifications) and Use Case 4 (Pre-Stage the Validate & Verify Process and Co-Develop Behavior Models). These two use cases defined the highest priority data exchange requirements and addressed the focus of this MBSE project, which is **how to enable a collaborative conceptual systems design activity between an aerospace/aircraft OEM and its design partners in the supply chain**. The other use cases are relevant to an overall product development lifecycle activity but are out of scope for this current effort. The key system architecture diagram types supporting Use Cases 3 and 4 were assigned the highest criteria scores and are highlighted in *italics*—rows 2.3, 2.4, 2.8, and 2.11—in Appendix B: Criteria/Weighting Scores Extract.

When implementing a MBSE interoperability solution, the HOW sub team also considered the different business scenarios defining the interactions between the OEM and a supplier. In some cases, both parties may require access to the software tools, or only one party will require access to the MBSE interoperability tools. Or, both parties could rely on a service provider. Those scenarios are illustrated in Figure 16 and were incorporated into the solution evaluation criteria (see the rows labeled 4.9 in Appendix B: Criteria/Weighting Scores Extract).

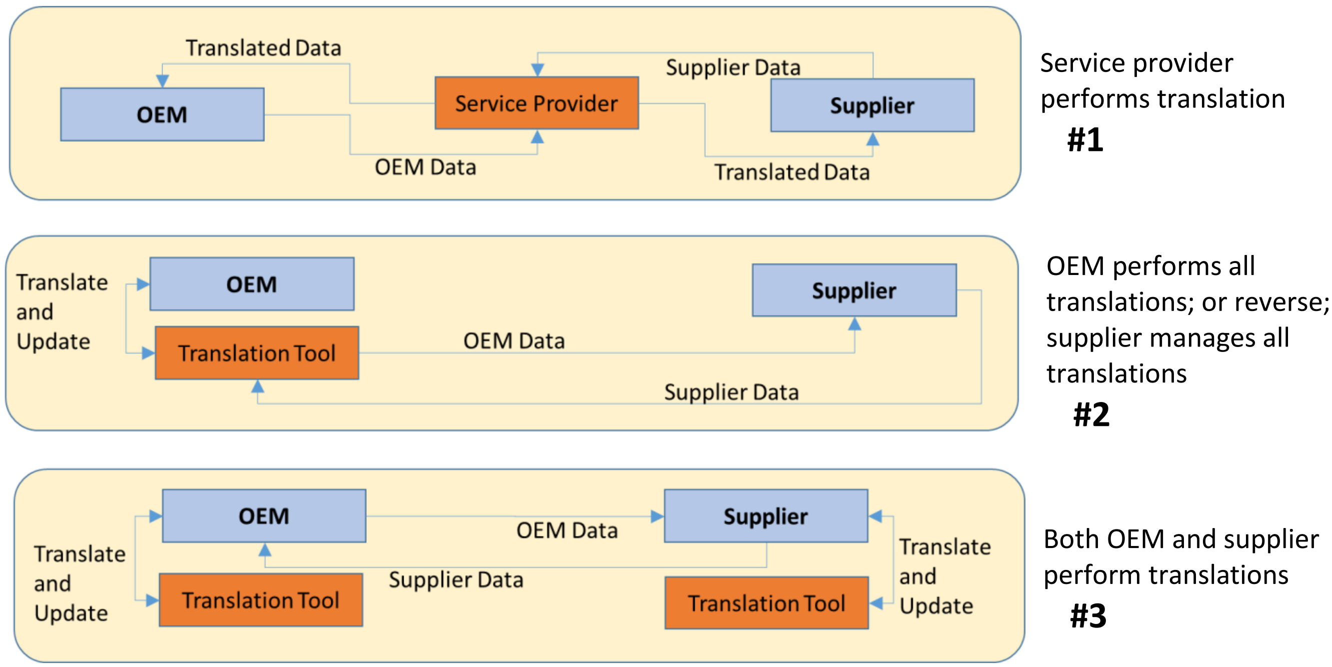

Figure 16 - OEM/Supplier Solution Deployment & Interaction Scenarios[9]

Phase 3 Summary and Go Forward Plan

This section presents a project summary that includes MBSE data interoperability alternatives and interim solutions, as well as observations and issues, and that provides a go forward plan.

Summary

The MBSE project team created multiple use cases defining the specifications for process-driven model interoperability across the lifecycle. With a focus on Use Cases 3 (how it will be specified) and 4 (how it will be used for validation purposes), the team generated a definition of the primary MBSE artifacts (diagrams) to be exchanged between the OEM and the supplier. It was assumed that by conducting a thorough analysis of the use cases and identifying the minimum list of models and relevant diagrams, we could reduce the complexity of the interoperability and translation requirements. The goal was to simplify what data must be exchanged while remaining effective in achieving design collaboration. After developing a comprehensive mapping between SysML and ARCADIA (i.e., comparing the diagrams, model elements, and relationships), the team assumed the minimum list of exchange artifacts would be affected by differences in the modeling languages. However, evaluation of the content produced by the two languages found very few inconsistencies with respect to the use case priorities. The distinguishing difference in functionality is the limited options when representing requirements in ARCADIA. Overall, a third-party interoperability software supplier product or a translation service is still considered to be the most viable near-term

[9] Scenario 3 could involve/require multiple MBSE interoperability tools.

solution. The solution will not necessarily support model-to-model exchange, but it will support collaboration, using a limited set of diagrams and depending on the availability of APIs from the most dominant software suppliers.

MBSE Data Interoperability – Alternatives and Interim Solutions

Plenty of examples exist that define the cost of developing complex products using historical text-based methods. It is the reduction in these costs that continues to inspire the industry to move forward. The following proposals support the future interoperability of architecture models.

- The aerospace community must be proactive and collectively aligned on the development of interoperability standards for bi-directional model exchange and design collaboration.
- The MBSE community must establish and participate in an Implementer's Forum to validate the data exchange use cases and assess the overall capabilities of the individual ADL product brands and related translation alternatives.
- The third-party translation products generally rely on each authoring tool's API. Collectively the industry should engage with, encourage, and incentivize the ADL software suppliers to publicly expose and standardize their APIs. Both positive and negative financial incentives should be considered as required (i.e., the carrot and the stick).
- Although best suited for mathematically verifying the behavior characteristics of the design architectures, many of the behavior modeling tools, such as Modelica and Simulink, can be used to represent the implementation architecture. With the added support of the Functional Mockup Interface (FMI) and System Structure Parameterization (SSP) specifications, and data exchange and co-simulation data standards, these mathematical tools can be used as a partial substitute for conceptual architecture model exchange. Although this type of solution is currently being used with varying degrees of success by several major OEMs in the automotive industry, this approach is not considered a viable long-term solution for the aerospace/defense industry needing to support the complexities of integrated distributed systems and Integrated Modular Avionics (IMA) architectures. The same may be said for autonomous vehicles.
- In the interim, without a common model exchange methodology, the aerospace industry must actively encourage the individual PLM and MBSE software suppliers to reach consensus on an exchange philosophy that supports interoperability between the most commonly used ADL tools.

MBSE Data Interoperability – Observations and Issues

Other factors impacting the exchange of system architecture models are as follows:

- Implementation of MBSE data standards (i.e., the variety of versions designated SysML v1.x) is not consistent between the different authoring tools. While this issue is expected to improve with the eventual release of SysML v2, no standards exist for the definition of the graphical diagrams.
- The SysML specification supports the creation of customized profiles that support user-defined stereotypes, tagged values, and constraints.
- Significant differences in the modeling methods used by each engineering team within both OEMs and their supply chain partners.
- No industry standards measure model quality and compliance or how to assess the accuracy and completeness of an MBSE model translation.

- No industry standards manage the protection of intellectual property.
- There are significant differences in the priority that each company assigns to MBSE modeling and the associated data standards development
- When do the costs of translation tools and accompanying labor exceed the value of exchange capability? Standardized business cases need to be developed and supported by the OEMs.
- Will the major MBSE software suppliers provide open APIs and on-going support for a third-party software translation service?
- Will changes to the data exchange standards and advancements in the digital technologies out-perform the value of a translation product or process?

In summary, a COTS translation product will not cover all aspects of interoperability. To develop the foundation for a successful process, the OEM and suppliers must predefine how the data will be used, how the data will be organized, the underlying data model, and the authoring methodology.

Go Forward Plan

As described within this report, model interoperability is one of the key factors for successful implementation of MBSE in the A&D industry. This evaluation took a minimalistic approach, focusing on the architecture models, but the methodologies must be consistently applied to all of the MBSE model representations, including the Requirements and Behavior models.

ARCADIA is generally implemented by one tool brand, so paths to interoperability with other ARCADIA models already exist. The main issue is SysML v1.x (version designation) and the differences inherent in the products produced by multiple software suppliers. For that reason, one of the major opportunities for the SysML v2 RFP (Request for Proposal), referenced in Figure 17, is to increase the exchange potential by stipulating a standardized API that is publicly available as part of the specification. Through a standardized API, SysML interoperability will be ensured either by the ADL software supplier or a third-party interoperability software supplier. With respect to the importance of this target, a separate RFP called *SysML v2 API and Services* was created by the Object Management Group's (OMG's) SysML Submission team.

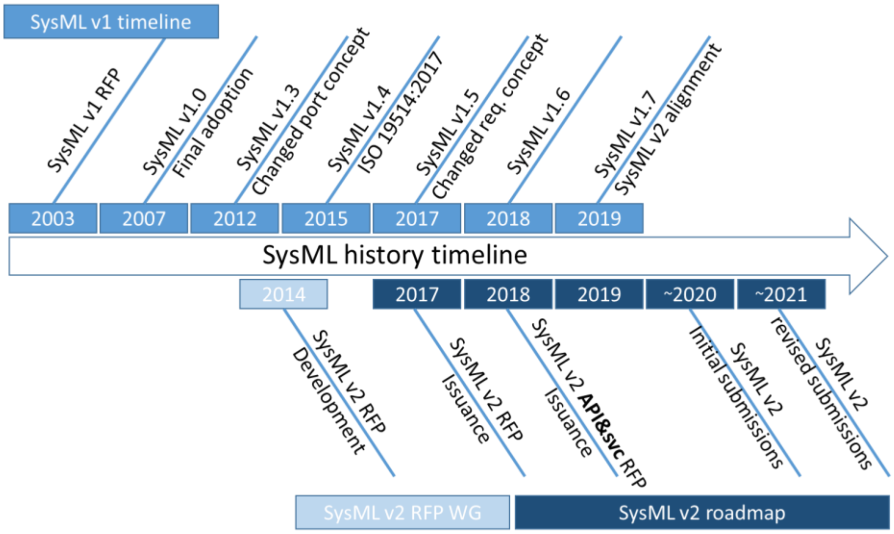

Figure 18 – SysML Timeline with API & Services (svc) Target

The MBSE team's Phase 3 activity was a natural extension of the Phase 2 report, but developing the next steps has not produced a simple resolution. As of this writing, SysML v2 has not been finalized, and the specification's provisions for interoperability are optional. For that reason, the interoperability situation hasn't changed and is potentially getting worse as the list of solution providers continues to grow.

From a data exchange standpoint, most of the SysML products do not facilitate the export and import of Canonical XMI, and proprietary plug-in products are needed to support a limited point-to-point exchange (between brands). In some cases, the MBSE tool providers have apparently devalued the exchange provisions in the new SysML specification and have not promoted alternative solutions. The OMG forum may not be a suitable environment to resolve the interoperability issues, and the major PLM MBSE providers have yet to agree on an interoperability solution.

The project team had initially recommended a hands-on software evaluation/benchmark activity leveraging the MBSE use cases. The addition of ARCADIA has added new complexities. The project team had also recommended that a third party be engaged to assist in the evaluation. Whether this is a viable next step is still under discussion and consideration by the AD PAG members.

The project team is also aware of other MBSE data interoperability initiatives being planned by other industry groups, including INCOSE, National Defense Industrial Association (NDIA), prostep ivip, French Network Users Association (AFNeT), academia, OMG, and the United States Department of Defense.[10] It would be highly beneficial to both the AD PAG and the entire A&D industry if a multi-industry consortium could collaborate with the major PLM providers to leverage these other initiatives and propose a consolidated effort and collaborative solution.

---

[10] SOSA - Sensor Open System Architecture Standard originated at the Air Force Life Cycle Management Center (AFLCMC) and is now managed by The Open Group. An over-arching open-systems standard identified in a Tri-Service memorandum, signed in 2019 by the secretaries of the U.S. Navy, Army, and Air Force.

The MBSE project team concluded that additional work is necessary to develop solutions that integrate the architecture models exchanged in these use cases. We did not explore the potential of using provider-neutral models based on alternative industry standards, such as Object Process Methodology (OPM). It is necessary to understand what modeling tools apply to variations in the business model, similar to the criteria that would drive the use of large-scale 3D CAD tools as well as mid-range CAD tools. The project team believes that these additional tasks are worth exploring before initiating an interoperability software product evaluation activity.

About AD PLM Action Group

The Aerospace & Defense PLM Action Group (www.ad-pag.com) is an association of aerospace and defense companies within CIMdata's globally recognized PLM Community Program, which functions as a ***PLM advocacy group*** to:

- Set the direction for the aerospace & defense industry on PLM-related topics that matter to members (*including promoting, not duplicating, the work of standards bodies*)
- Promote common industry PLM processes and practices
- Define requirements for common interest PLM-related capabilities
- Communicate with a unified voice to PLM solution providers
- Sponsor collaborative PLM research on prioritized industry and technology topics

CIMdata administers Group operations, coordinates research, and manages the progression of policy formulation.

About CIMdata

CIMdata, a leading independent worldwide firm, provides strategic management consulting to maximize an enterprise's ability to design and deliver innovative products and services through the application of Product Lifecycle Management (PLM) solutions. Since its founding over thirty years ago, CIMdata has delivered world-class knowledge, expertise, and best-practice methods on PLM solutions. These solutions incorporate both business processes and a wide-ranging set of PLM-enabling technologies.

CIMdata works with both industrial organizations and providers of technologies and services seeking competitive advantage in the global economy. CIMdata helps industrial organizations establish effective PLM strategies, assists in the identification of requirements and selection of PLM technologies, helps organizations optimize their operational structure and processes to implement solutions, and assists in the deployment of these solutions. For PLM solution providers, CIMdata helps define business and market strategies, delivers worldwide market information and analyses, provides education and support for internal sales and marketing teams, as well as overall support at all stages of business and product programs to make them optimally effective in their markets.

In addition to consulting, CIMdata conducts research, provides PLM-focused subscription services, and produces several commercial publications. The company also provides industry education through PLM certification programs, seminars, and conferences worldwide. CIMdata serves clients around the world from offices in North America, Europe, and Asia-Pacific.

To learn more about CIMdata's services, visit our website at www.CIMdata.com or contact CIMdata at: 3909 Research Park Drive, Ann Arbor, MI 48108, USA. Tel: +1 734.668.9922. Fax: +1 734.668.1957; or at Oogststraat 20, 6004 CV Weert, the Netherlands. Tel: +31 (0) 495.533.666.

Appendix A: SysML/ARCADIA Diagram Mapping

The table below provides mapping between SysML diagrams and their model elements to equivalent diagrams and model elements in the ARCADIA framework. These model diagrams, elements, and relationships are considered the minimum necessary to provide a model-based digital data exchange.

SysML Diagram	SysML Model Elements	SysML Relationships	ARCADIA Diagram	ARCADIA Model Elements	ARCADIA Relationships
Use Case diagram	Use Case Actor	Association Extends Includes	Capabilities diagram, ^	Capability, ^ Actor, ^ SA Mission, SA System	OA Involvement, Extends, ^ Includes, ^ SA Capability Exploitation, SA Involved Actor
Block Definition diagram	Block Properties	Generalization Association Composition Aggregation Connector	Component Breakdown diagram, * + Component Interface diagram, *	Component *	Contained In, Port Delegation
Activity diagram	Action Port Control Node	Flow Control Flow Object Flow	Architecture diagram, ** Functional Data Flow diagram, ** Functional Chain diagram, **	System/ Component, ** Function (Non Leaf and Leaf), ** Function Ports, ** Functional Chains, ** Control Nodes ** (AND/OR/IT)	Functional Exchange, Functional Allocation, Port Allocation, Sequence Links

SysML Diagram	SysML Model Elements	SysML Relationships	ARCADIA Diagram	ARCADIA Model Elements	ARCADIA Relationships
Internal Block diagram	Block Part Port Activity Parameters	Connectors Item Flows	Architecture diagram * Functional Breakdown Diagram (+ All views from Activity diagrams)	Component, * Actor * Component/ Physical Ports, * (+ All elements from Activity diagrams)	Functional Exchange, Component Exchange, Physical Link
Sequence diagram	Block Events	Signals Messages	Functional Scenario diagram,* Entity Scenario diagram*	Execution, * Instance Role, * State Fragment *	Sequence message
Parametric diagram	Parameter	Connector	Architecture diagrams (Parametric viewpoint) **	Property Value, Constraint	Applied Property Value, Constraint Element
State Machine diagram	State Operations	Events Messages Call Events	Mode/State Machine diagram	Mode, State, Choice, Join, Fork	Transition, Entry, Do Activity, Exit, Operational Activities/ Functions

^ *Available in the following ARCADIA levels: Operational Analysis, System Analysis*

* *Available in the following ARCADIA levels: Logical Architecture, Physical Architecture*

** *Available in the following ARCADIA levels: System Analysis, Logical Architecture, Physical Architecture*

Appendix B: Criteria/Weighting Scores Extract

The following table provides an extract of the criteria and weighting scores that will be applied to exchange tool capability. These evaluation criteria apply to a standalone software tool or translation service using such a tool. The criteria for an engineering services company to create or manage the exchange service capability would be different and is not shown here.

Experienced ADL Exchange Companies/Tools

	Companies			Company1
	Products		Explanation	Product1
item #	**CRITERIA**	*weight*	*describe criteria*	
1	**Language**			
1.1	SysML	1000	target language	100%
1.2	UML	500	lower usage	1%
1.3	AADL	100	rarely used	1%
1.4	Marte	100	rarely used	1%
1.5	Plug-In Interface	1000	plug-ins for existing ADL language products	50%
1.6	Proprietary	10	no source code or user modification allowed	
2	**Diagram Types**			
2.1	Requirements	10	decomposition and relationships	100%
2.2	Structural	500		
2.3	*Block Definition*	1000	target diagram type	100%
2.4	*Internal BD*	1000	target diagram type	1%
2.5	Package	100	composite view of populated diagrams	1%
2.6	Parametric	500	constraints and equations	100%
2.7	Behavior	10	activities, states, controls, interactions	1%
2.8	*Activity*	1000	target diagram type	100%
2.9	Use Case	10	SoS functional operational description	100%
2.1	State	500	simulations and controls	100%
2.11	*Sequence*	1000	target diagram type	100%
3	**OS**			
3.1	Windows	1000	majority of the users	
3.2	Linux	100	supports math-based application integration	

	Companies			Company1
	Products		Explanation	Product1
item #	**CRITERIA**	*weight*	*describe criteria*	
3.3	Both	500	support for mixed environments	100%
3.4	DB Only	10	proprietary database	
3.5	Web/Cloud	500	supporting service technology	100%
4	**Product-Packaging**			
4.1	Pricing	500	500 < $5k/yr. average seat cost	
4.2	Availability	1000	Std. product available to any purchaser	100%
4.3	Distributed Licensing	500	node locked	
4.4	Floating License	1000	flexibility in hosting	
4.5	Enterprise License	500	scalable for large companies	100%
4.6	Open Source	1000	API included	
4.7	Services	1000	installation support, infrastructure analysis	
4.8	Version Stability	10	public announcements	
4.9a	User Level Capability #3	1000	both OEM and suppliers perform translations	100%
4.9b	Special skills Capability #2	500	OEM performs all translations; or reverse	100%
4.9c	Service Level #1	100	Service company performs translation	
5	**Product Functionality**			
5.1	Scalability	500	number of users	1%
5.2	Interfaces	1000	support for multiple tool brands	100%
5.3a	Bi-Unidirectional Links	500	translate the characteristics of a link	100%
5.3b	Traceability Links	1000	connections between elements	100%
5.3c	RESTful Links (OSLC)	10	maintain Restful links after translation	100%
5.4	Multi-Disciplinary Deliverables	100	supports data from all domains	100%
5.5	Concurrent Engineering	100	multiple users editing at one time	100%
5.6	Data Federation	100	independent model/product management	100%
5.7	Graphics Rendering	1000	support for diagrams	
5.8	Model Translation-Export	1000	output is tool-brand compatible SysML	100%
5.9	Inclusive Processing	1000	or requires license of each I/O product	1%

	Companies			Company1
	Products		Explanation	Product1
item #	**CRITERIA**	*weight*	*describe criteria*	
5.1	Support Custom SysMLProfiles	1000	translation accommodates custom profiles	1%
6	**Data Management**			
6.1	Config. Mgmt.	1000	product, derivative, dependencies	100%
6.2	Change Mgmt.	1000	tracking, marking, history	100%
6.3	Maturity	10	workflow designation, new/in-work/approved	1%
6.4	Security	1000	encryption, authentication, handling	1%
6.5	System Lifecycle Management	500	manages design evolution	1%
6.6	Requirements Management	10	replaces RMS tools (config. mgmt.)	1%
6.7	Model Scale	500	size of the model(s), large > 1GB	
7	**User Interface**			
7.1	Results Visibility	100	success / fail flags	100%
7.2	Diagnostics	100	error detection, % success	1%
7.3	Reports/Metrics	100	before-after metrics reporting of results	100%
7.4	Ease of Use	100	UI with component controls	1%
8	**Product Support**			
8.1	User Support/Training	100	user guides, demos	100%
8.2	Bug Reporting System	10	translation product errors	
8.3	Product Update Regularity	1000	1000 = twice yearly, patches when needed, release plans available	
9	**Company Resources**			
9.1	Multiple Commercial Products	100	variety of software products	
9.2	Translation Experience	1000	demonstrated existing capability	100%
9.3	Applicable Partnerships	500	contracts with existing SysML providers	100%
9.4	Engineering Design Services	100	consulting on design capabilities	1%
9.5	Architecture Modeling Services	100	consulting on ADL/MBSE capabilities	100%
9.6	User Base	1000	1000 = notable market share using product	1%

	Companies			Company1
	Products		Explanation	Product1
item #	**CRITERIA**	*weight*	*describe criteria*	
10	**Product Capability/Integration**			
10.1	CAMEO / MagicDraw	1000		100%
10.2	IBM Rhapsody / Rational	1000		100%
10.3	ARCADIA-Capella	1000		
10.4	Enterprise Architect	1000		100%
10.5	PTC Integrity Modeler	1000		100%
10.6	SCADE Architect	1000		
10.7	Papyrus	100		
10.8	System Architect	100		
10.9	Modeler	100		
10.10	CORE	100		
10.11	Innoslate	100		
10.12	Modello	100		
10.13	Cradle	100		
10.14	Matlab/Simulink	500	API tool integration	100%
10.15	Mathematica	100	API tool integration	100%
10.16	Modelica	500	API tool integration	
10.17	DOORS/DOORS NG	1000	API tool integration	100%
10.18	Jama	100	API tool integration	100%
10.19	Microsoft Excel	1000	API tool integration	100%
	Max score:	43100	Total score:	23000
			Percentage of maximum score:	52.00%
			How many criteria scored:	54

Appendix C: Glossary

Term	Definition	Source
ADL	Architecture Description Language	ISO/IEC 42010
Allocation	The process of associating a function or requirement to an element of the architecture.	
AFNeT	French Network Users Assoc., digital practices in industry	AFNeT
ARCADIA	ARChitecture Analysis and Design Integrated Approach	Thales Group
AVSI	Aerospace Vehicle Systems Institute	AVSI
BDD	SysML block definition diagram	
Component	One of the hardware, electronic, software, or documentation items (e.g., parts, sub-assemblies) that make up a system. A component may be subdivided into other components.	CIMdata PLM Glossary
Constraints Model	See Observer Model.	
COTS	Commercial Off-the-Shelf (software, hardware)	NIST
Design Model	A model that describes the system solution which implements a functional model. The model is based on the component breakdown, with each component having their own interfaces and allocated functions. A software design model defines any software design such as a low-level requirement, a software architecture, algorithms, component internal data structures, and data flow and/or control flow. A model used to generate source code is a design model.	
ECO	Engineering Change Order	Synopsys, 2011
Equipment	A combination of parts, components, accessories, attachments, firmware, or software that operate together to perform a function(s) within a System/Sub-System Architecture, as, or for an end-item or a system. Equipment may be a subset of an end-item based on the characteristics of the equipment. Equipment that does not meet the definition of an end-item is a component, accessory, attachment, firmware, or software. Equipment are certified end items, per TC (Type Certificate), TSO, or equivalent standard, supplied by	CIMdata AD PLM AG – Multi-view BOM project team

Term	Definition	Source
	external companies and their support and spares are under the original suppliers' responsibility.	
Formalized Functional Requirement	A format of a modeled requirement in which the computed output or state is a function of the inputs and/or states. Example: y = f(x,t).	
Formalized Requirements	Requirement defined by unambiguous semantics and syntax.	
Function	The action or actions that a product is designed to perform	EIA-649-A 2004
Functional Decomposition	The process of breaking down a function into sub-functions.	SEBoK
IBD	SysML internal block diagram	
ICD	Interface Control Definition	
IMA	Interface Modular Avionics	IEEE/AIAA DASC Conference, 2015
MBD	Model Based Design, or Model Based Development defines real-time, high fidelity, causal/acausal system models that use mathematical and visual methods to define behaviors, controls, and performance. Often constructed as lumped parameter models representing the systems functions, signal processing, and communication systems. The MBD methodology supports code generation and the design of embedded software.	
MBSE	Model-Based Systems Engineering is the formalized application of various levels of modeling (from 0D to 3D) to evaluate system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later lifecycle phases.[11] In its most direct form, MBSE applies a continuous modeling paradigm (0D, 1D, 2D, 3D...) to define systems, progressing from the most simple (0D) form to a fully defined 3D representation, and on to higher order models to understand temporal issues. This is done in addition to and in the context of written requirements and 2D and 3D CAD designs. The models are	CIMdata PLM Glossary, See footnote.

[11] INCOSE Systems Engineering Vision 2020. INCOSE-TP-2004-02. San Diego, CA. September, 2007.

Term	Definition	Source
	used to validate from very early stages that the system will function as conceived and defined by its requirements.	
Model	An abstraction, or information item (separately identifiable body of information that is produced, stored, and delivered), aiming at understanding, communicating, explaining, or designing aspects of interest. This information item respects a certain model kind (i.e., conventions for a type of modeling) to support analyses (as opposed to a document written in informal natural language).	
NDIA	National *Defense* Industrial Association	NDIA
NIST	National Institute of Standards and Technology	NIST
Observer Model	A format of a modelled requirement that defines an unambiguous constraint relationship between the inputs, outputs, and states without specifying a complete design of exactly how to compute the output from the inputs and states. Example: $y < f(x,t \pm dt)$ This kind of model is also called a Constraints Model or Property-Based Model.	
OEM	Original Equipment Manufacturer	
Operational Scenario	An operational scenario is a temporally ordered set of activities needed to fulfill a mission, under normal and abnormal conditions.	
OSAM	Overall System Architecture Model, comprising a view for functional decomposition, a view for product breakdown into components, and a view for a 3D envelope.	
PLM	Product Lifecycle Management	CIMdata
Product Model	A Product Model is a model that represents a solution architecture including its constituent elements (components, hardware, and software), their interactions, and the functions allocated to those elements.	
Property-Based Model	See Observer Model.	
prostep ivip	International trade organization focused on standards and methods supporting the industry’s digital transformation	prostep ivip

Term	Definition	Source
RBE (Requirement-Based Engineering)	Specification and design process based on requirements tracing, allocation, and decomposition as an answer to ARP4754/DO178 policy.	
ReqIF	Requirements Interchange Format	OMG
Requirement	A statement that translates or expresses a stakeholder need and its associated constraints and conditions.	ISO/IEC/IEEE 29148:2011
Requirement Decomposition	The process of detailing / breaking down requirements to the next lower level of requirement.	
Requirement Validation	Determination that the requirements are correct and complete.	ARP4754
Software	Software is a set of instructions, data, or programs used to operate computers and execute specific tasks.	
Software Architecture	The structure of a software system comprising software elements, relations among them, behaviors, and properties of both elements and relations.	SEI
Specification Model	A model that represents requirements which specify functional, performance, interface, or safety characteristics without consideration of the design implementation. An executable specification is a model of a system's behavior that can be executed at least for purposes of its own validation. The execution can be either simulation (e.g., in Simulink) or animation (e.g., SysML in Rhapsody).	
SysML	Systems Modeling Language	OMG
System	A system is an ordered arrangement of components with links to comprise a whole where the components mutually interact together toward achieving a common objective. It can be regarded as a set of interrelated components that perform one or more functions.	ISO/ IEC 15288
System Architecture	Fundamental concepts or properties of a system, its environment embodied in its elements, relationships, and the principles of its design and evolution.	ISO/IEC/IEEE 42010:2011
System of Systems (SoS)	A system of systems (SoS) brings together a set of systems for a task that none of the systems can accomplish on its own. Each constituent system keeps its own management, goals, and resources while coordinating within the SoS and adapting to meet SoS goals.	ISO/IEC/IEEE 15288 Annex G (ISO, 2015)

Term	Definition	Source
UML	Unified Modeling Language	OMG
UMLDI	Unified Modeling Language Diagram Interchange	OMG
V-Model	Systems development lifecycle	INCOSE, *Systems Engineering Handbook*
Validation	The evaluation if the right thing was developed in its functions.	SEBoK
Verification	The evaluation if the development was done right in its requirement traceability.	SEBoK
XMI	XML Metadata Interchange (Canonical XMI)	OMG