

## PLM Road Map™ & PDT Europe 2023

*The Digital Thread in a Heterogeneous, Extended Enterprise Reality*

*A call for PLM Professionals to share their knowledge & experience*

**CIMdata**

November 15 & 16

**•eurostep**



**AEROSPACE & DEFENSE PLM ACTION GROUP**

# Model-based OEM/Supplier Collaboration Needs in the Aerospace Industry Driving Toolchain Requirements and Tool Provider Selection

Hartmut Hintze, HO Quality Cabin Program Operations (QCO), Airbus Operations GmbH

Administered by:

**CIMdata** | Global Leaders in PLM Consulting  
[www.CIMdata.com](http://www.CIMdata.com)

# MBSE Data Interoperability

## *Introduction*

---

- **Problem Statement**

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

- **Project Objectives**

To evaluate, identify, and promote methods of exchanging digital engineering design content, including system architecture models.



# Solution Provider Categories

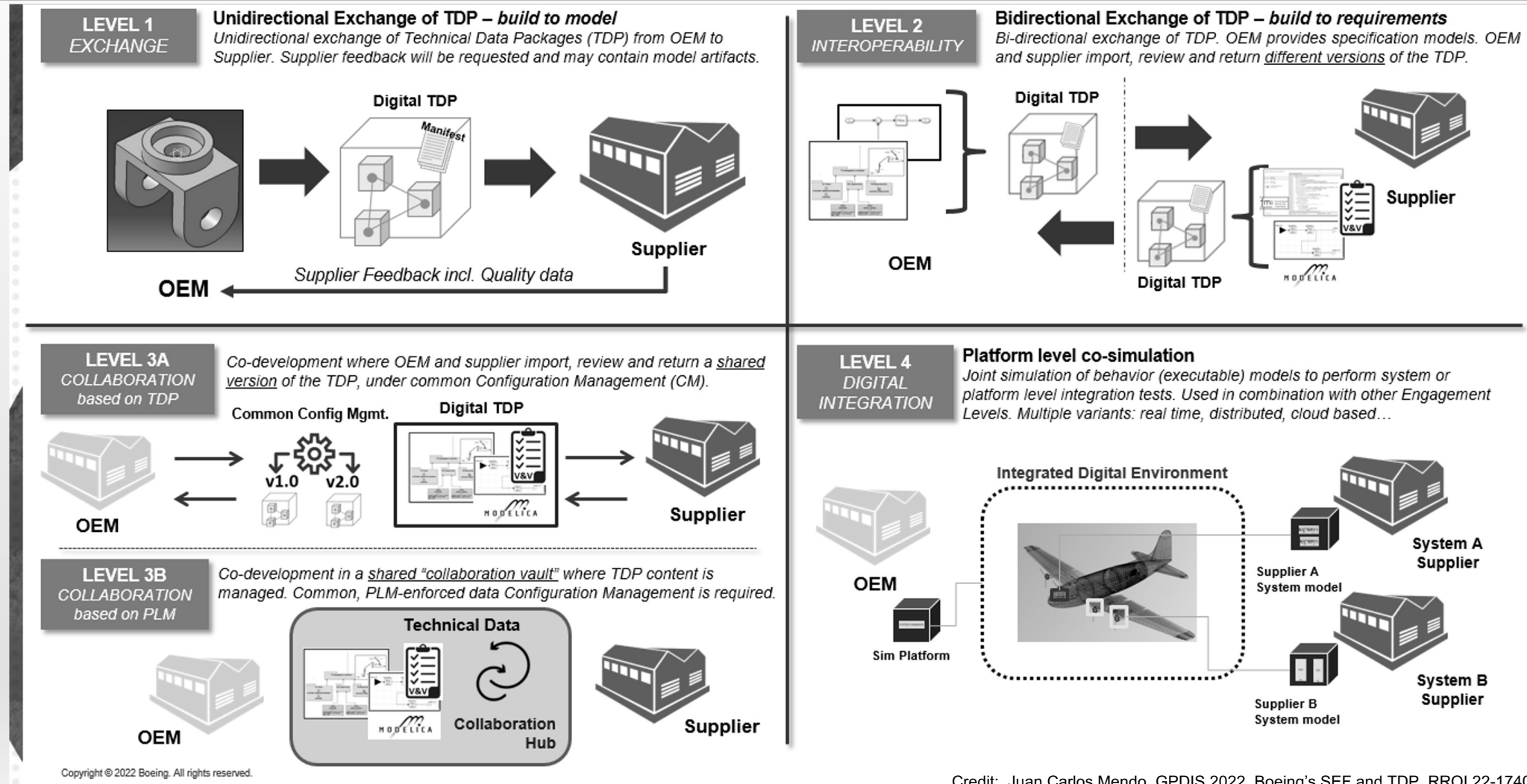
*Definitions used in this presentation*

- PLM Vendor – support for, or 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 Vendor – seller of a standalone architecture authoring tool that is ADL compliant. ADL examples include: AADL, Acme, ARCADIA, ArchiMate, OPM, Rapide, SysML, UML.  
(ADL = Architecture Description Language)
- 3<sup>rd</sup> Party Vendor – seller of an integration service or software tool(s) that supports the translation, exchange, or alternative representation of models generated from two or more brands of ADL compliant authoring tools



# Supplier Engagement Framework

## Different Level of Collaboration



Credit: Juan Carlos Mendo, GPDIS 2022, Boeing's SEF and TDP, RROI 22-174038



# Artifact Definitions

## *MBSE Working Team Perspectives – Model Types*

- System Architecture Models:

- *“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; the organizational structure of a system and its implementation guidelines.” (ISO/IEC/IEEE 24765)*
- System models created using an ADL (Architecture Description Language) compliant tool as defined by [ISO/IEC/IEEE 42010](#)

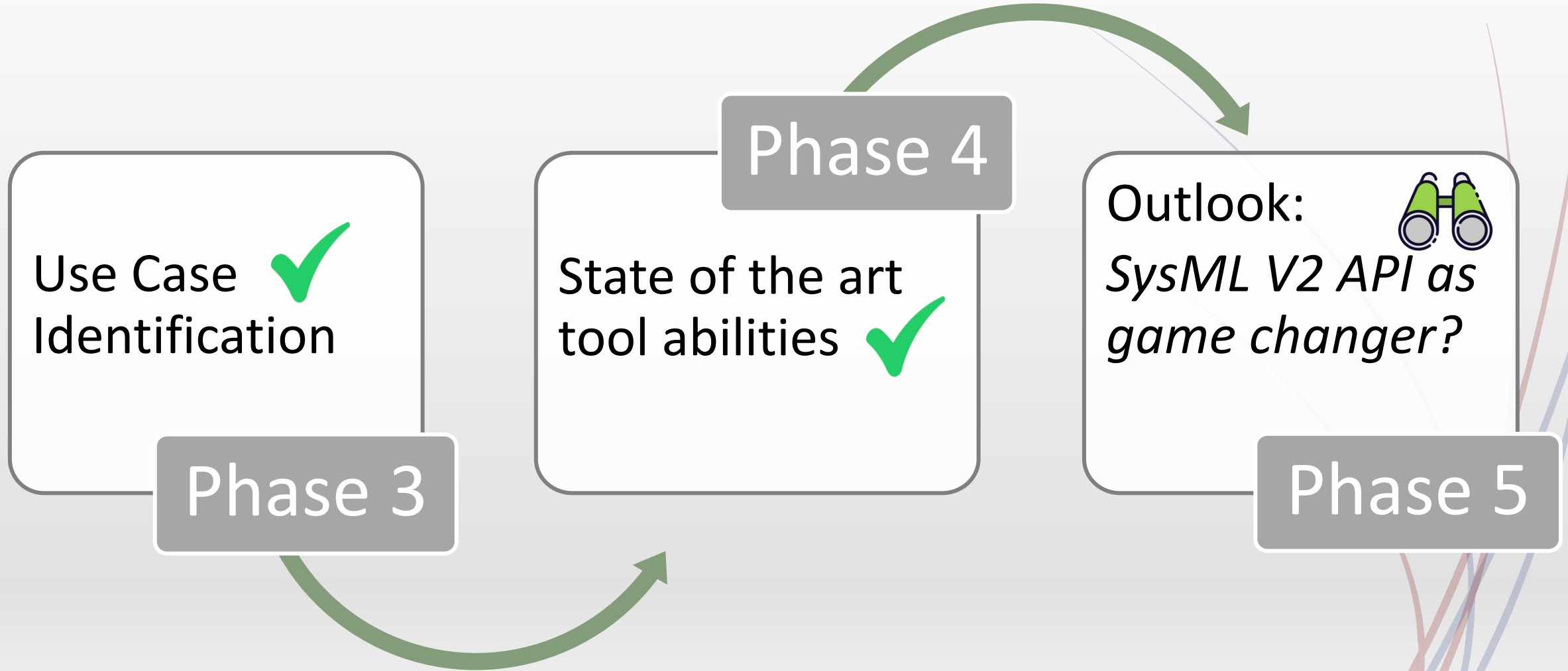
- Behavior Models:

- Quantitative assessments of System/Structural Plant Models. Lumped parameter models for behaviours and controls described by mathematical specifications or executable code, containing differential, algebraic and discrete equations. The application of a physics-based modelling environment.
- Models created using [MBD](#) – (Model Based Design/Development) tools, to evaluate complex equations that are not suited or easily executed in an architecture model.



# Supplier Engagement Framework

*Way of Working*



---

Use Case  
Identification

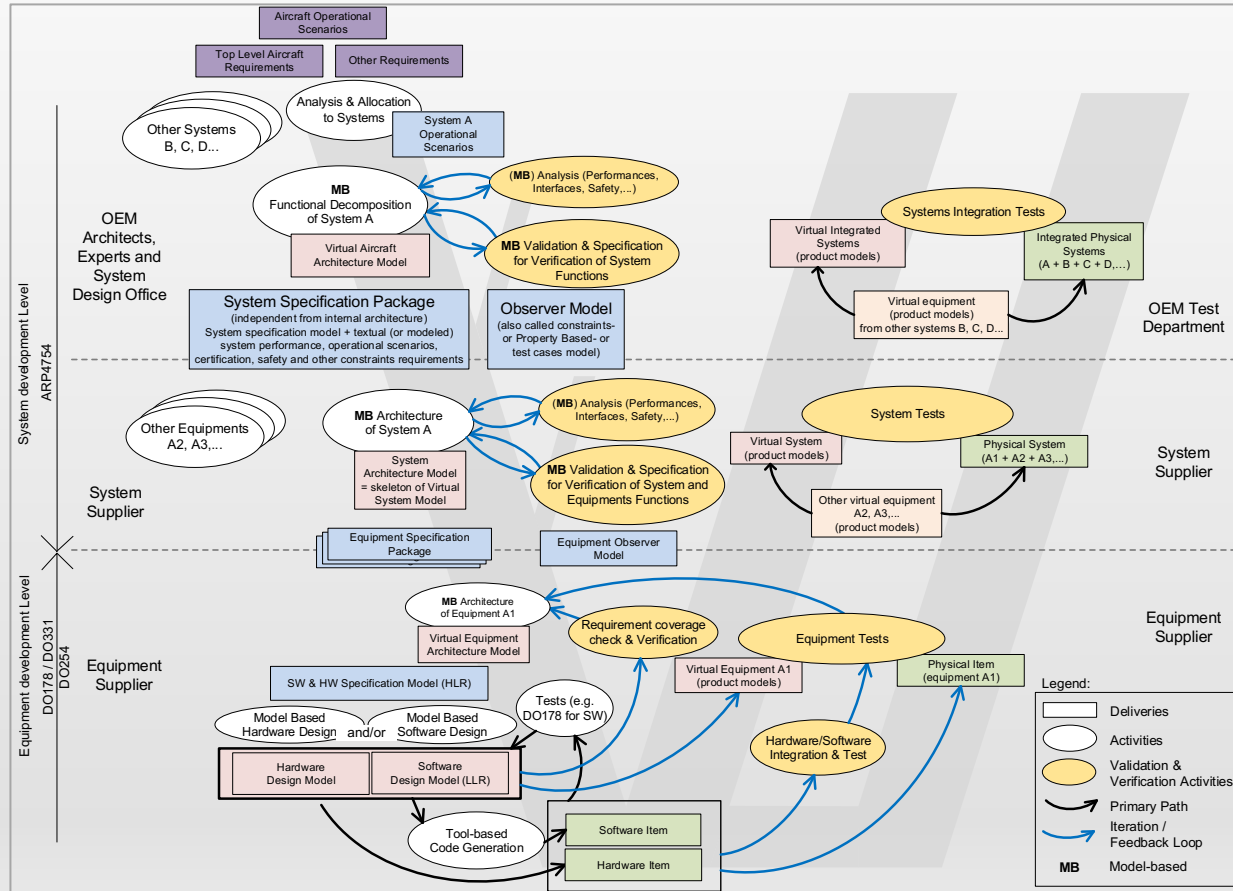
Phase 3





# Phase 3: MBSE Data Interoperability Specification

## *Correlation of Traditional Process Lifecycle with MBSE Use Cases Project*



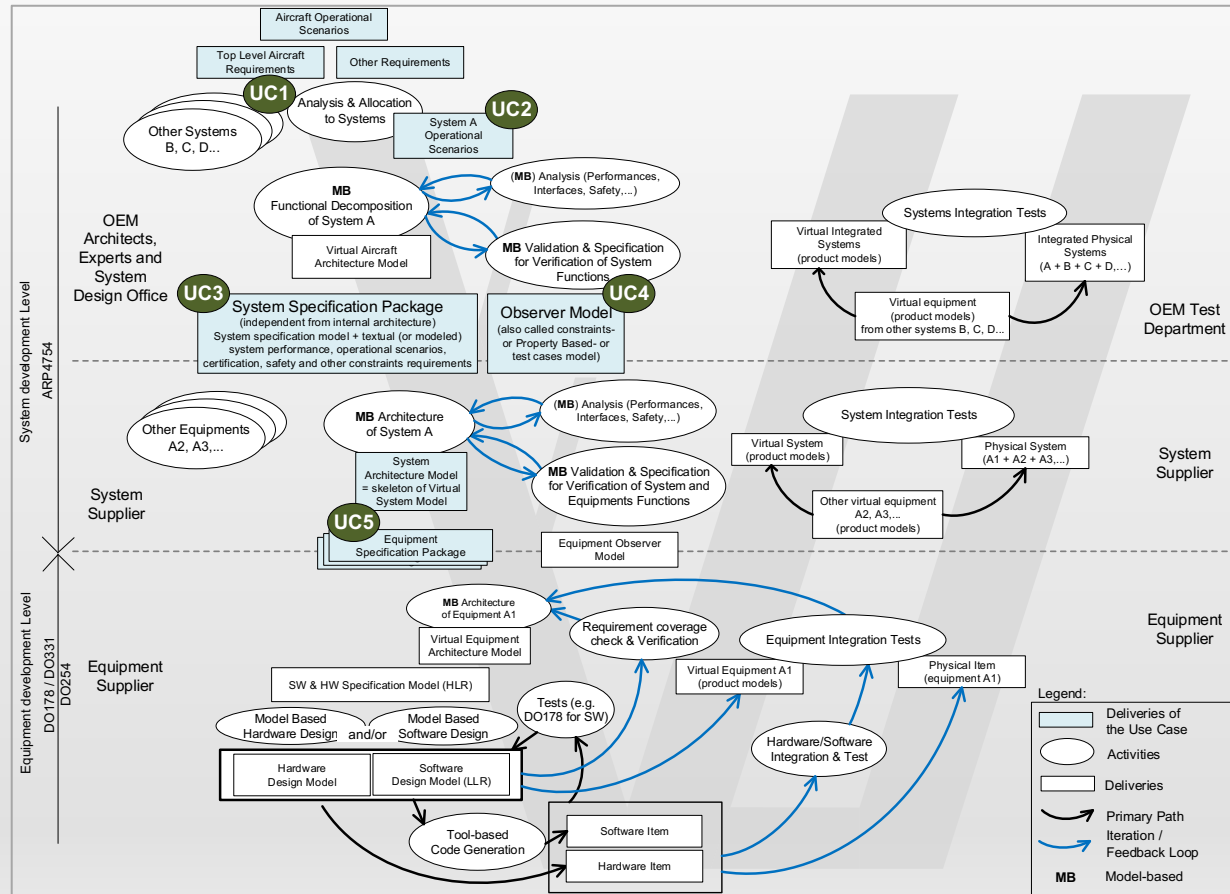
- The specification of dedicated use cases for an overall MBSE process requires input and agreement from the primary stakeholders. This figure describes the overall MBSE process mapped to the classical systems engineering “V” as the baseline for the use case definitions.
- The system development lifecycle process consists of three basic activities
  - Specifying and designing the system itself
  - Verifying and validating that system design
  - Managing the overall development project





# Phase 3: MBSE Data Interoperability Specification

## Identify Use Cases in need of Data Exchange

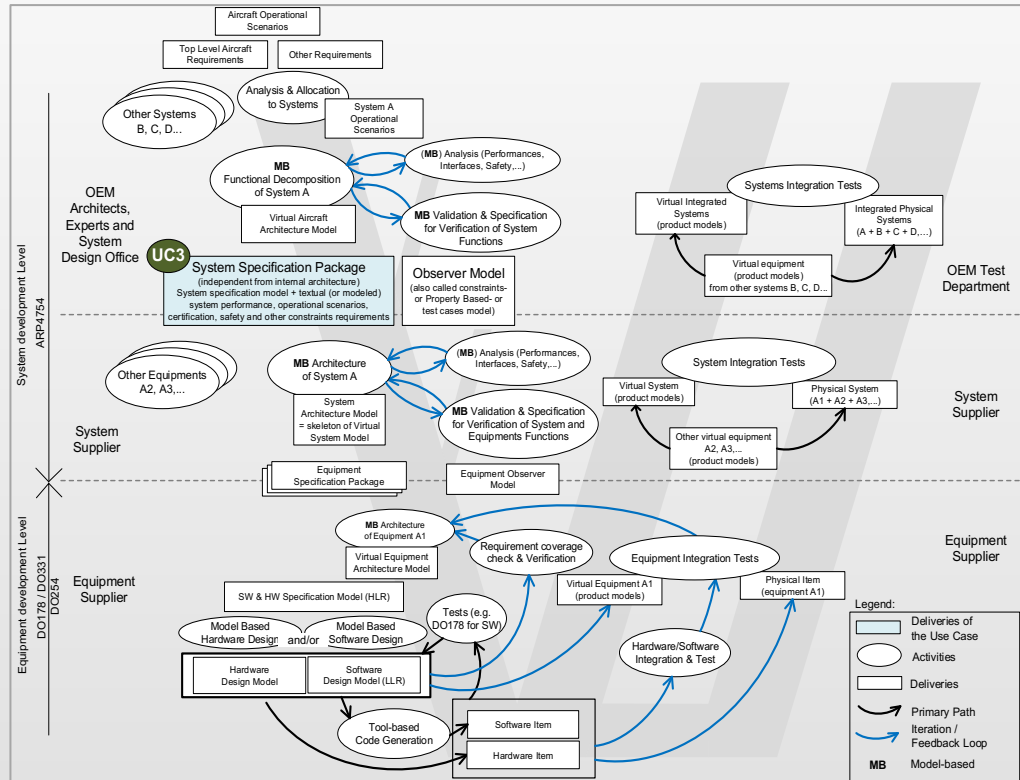


- The initial process can be divided into three phases: the conceptual phase, the preliminary design phase, and the detailed design phase.
- To identify the deliverables of the different steps within the overall process, the following use cases that describe the activities and deliverables in a top-down process are identified:
  - Use Case 1: System of Systems and Transitioning the Functional Interfaces to Logical Systems
  - Use Case 2: Define System Operational Scenarios
  - Use Case 3: Export System Functional Specifications
  - Use Case 4: How the Functional Specification and supplier product will be Validated (define the system context)
  - Use Case 5: Export Hardware/Software Functional Specifications



# Phase 3: MBSE Data Interoperability Specification

## *Data Exchange Criteria for Priority Use Case 3 & 4*



## Use Case 3 - Export System Functional Specifications

The first important Use Cases because it represents the Buy-Package interface between the OEM and the supplier. The drivers behind the exchange of architecture models include:

- collaboration on the contents of a Buy-Package
- common understanding of the model syntax
- model reuse at supplier side



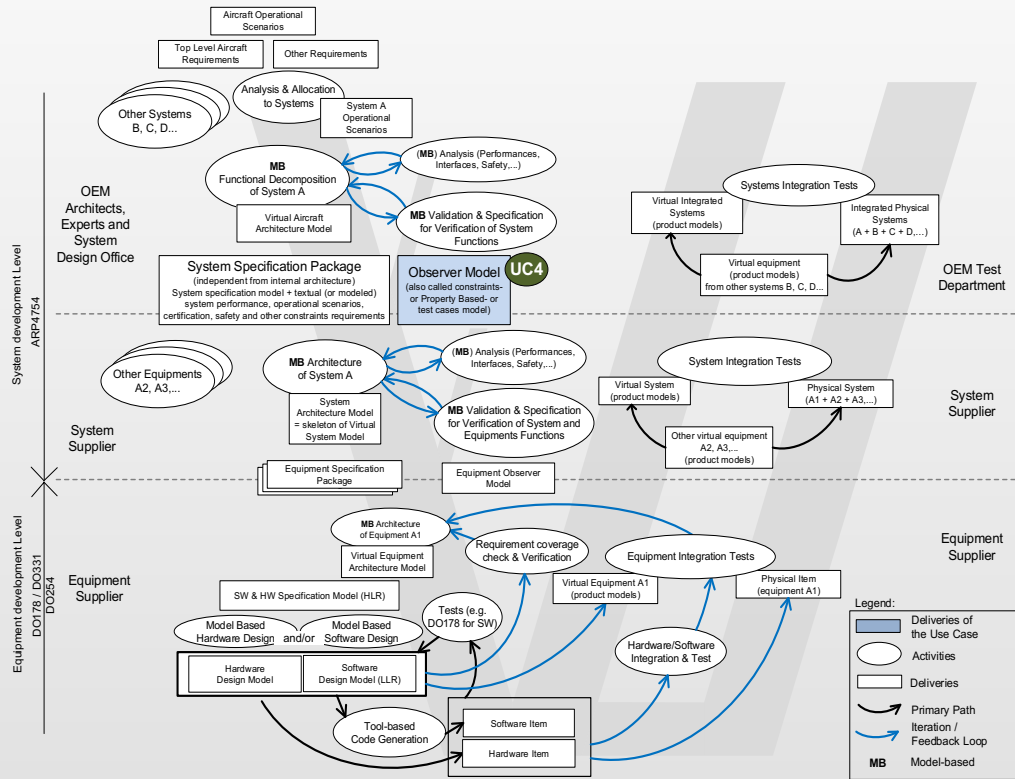
# Phase 3: MBSE Data Interoperability Specification

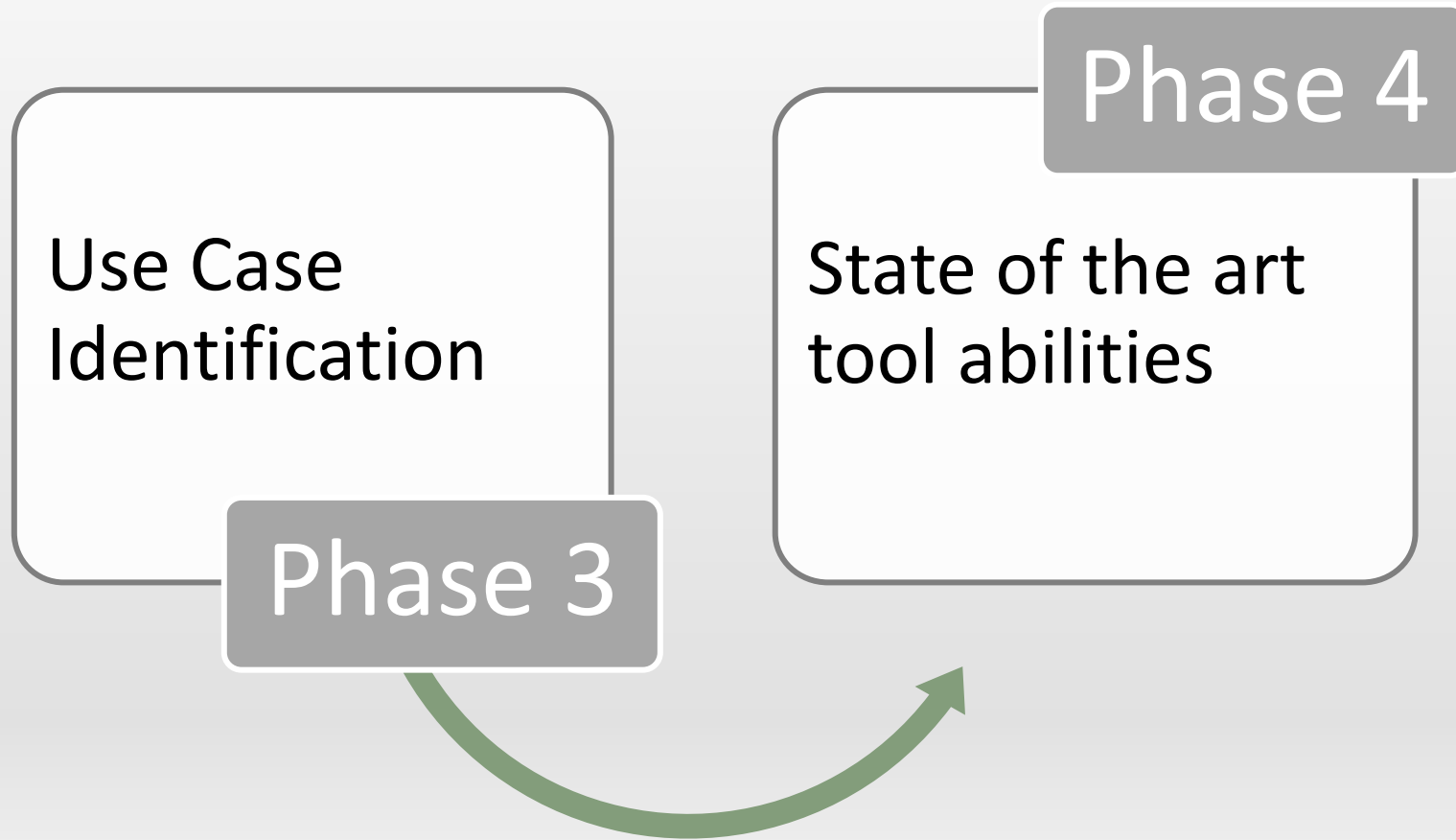
## Data Exchange Criteria for Priority Use Cases 3 & 4

### Use Case 4 - Validating the Supplier Models

The second important Use Cases because it supports the system functional validation at OEM and the supplier side. The drivers behind the exchange of V&V models include:

- a common understanding of the system context
- validation of functional specification completeness
- model reuse at supplier side for product validation before delivery





# Phase 4: OEM-Supplier Collaboration

*What can we do now?*

## OEM View

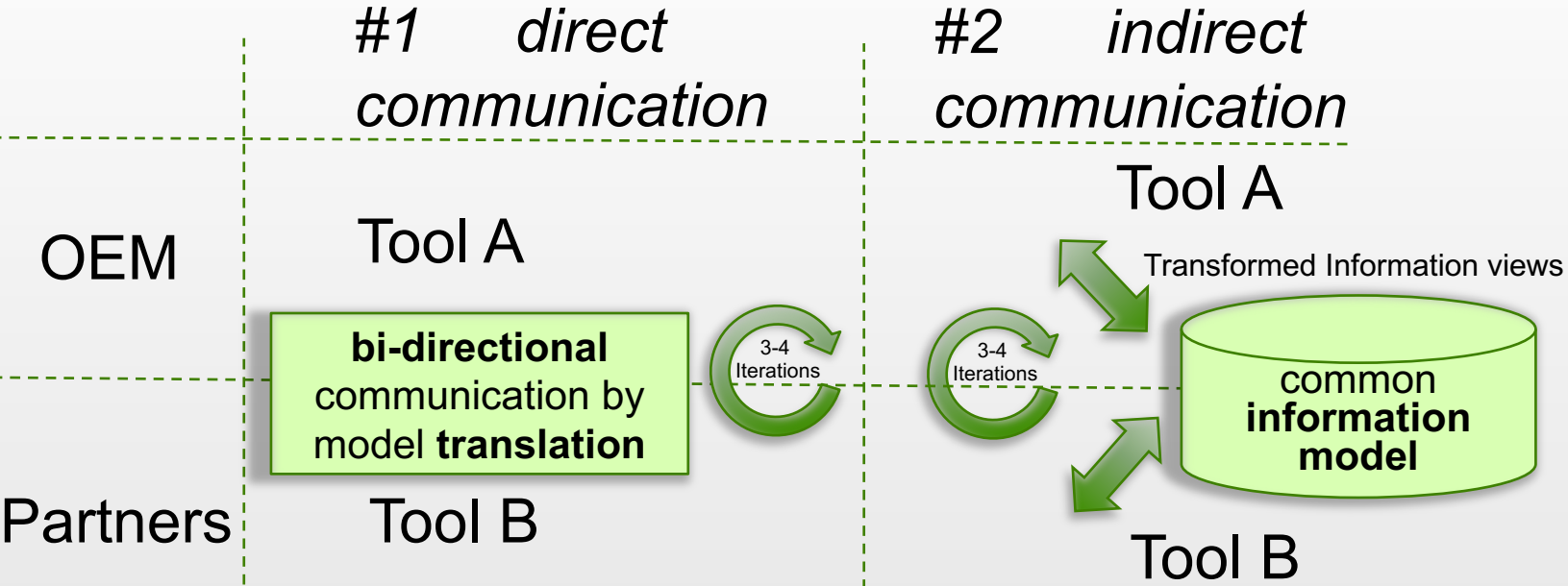
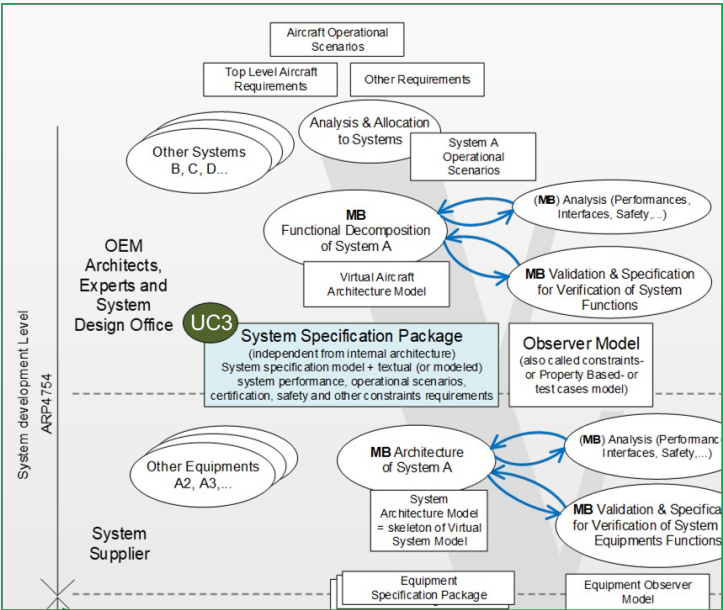
Plan for a new product development activity. It will involve multiple Tier 1 Suppliers. They each will design and deliver a major subsystem.

Integrate the design requirements, provide the architecture structure, interfaces, and functional behaviors **in digital models**.

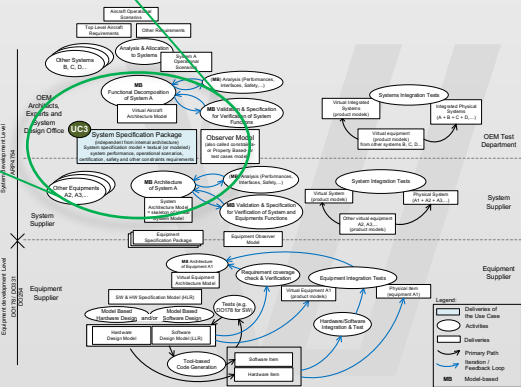
The design phase starts next month, what are my options?



# Phase 4: Interoperability Solution Scenarios 1 & 2



## Use Cases

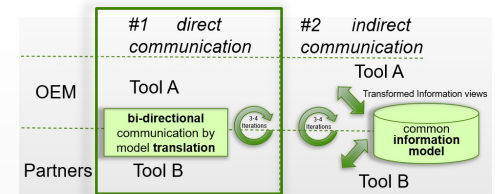


- Focus on Architecture Description Languages (ADLs)
- Assumes Partner uses different authoring tool
- Outcome - integrated models and designs



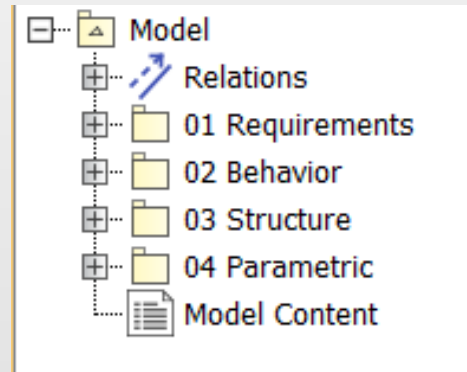
# Phase 4: Direct Communication

## Language Compatibility



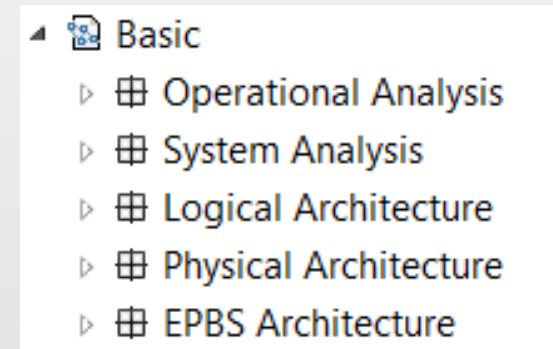
- For Use Case 3, an agreed modelling standard will be required to enforce consistency and enable model data exchange (assumes SysML  $\leftrightarrow$  ARCADIA, or SysML  $\leftrightarrow$  SysML)
- For SysML  $\leftrightarrow$  ARCADIA model data exchange requires a mapping of views and elements:

### SysML



The Package structure in SysML is defined by the user and could be completely different between models.

### ARCADIA



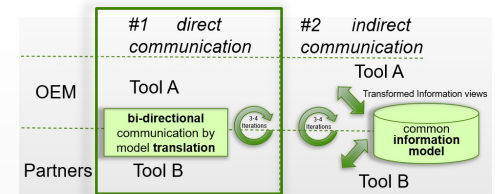
The Capella tool enforces the ARCADIA methodology as a framework. This means consistency across all Capella models.





# Phase 4: Direct Communication

## Model Mapping



## SysML <=> ARCADIA model data exchange

- A comprehensive mapping between SysML and ARCADIA (diagrams, model elements and relationships) defined for all the artifacts identified in Use Case 3 (partial table extract)
- OEM and Supplier model data exchange for Use Case 3 can be based on this mapping

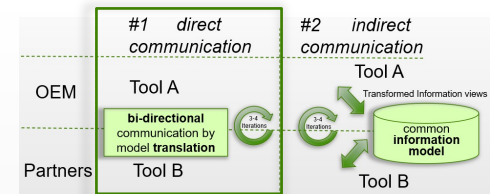
ARCADIA			SysML		
Diagrams	Model Elements	Relationships	Diagram	Model Elements	Relationships
(SA) Mission/Capability			Use Case		
	Capability			Use Case	
	Actor			Actor	
	Component			Block	
		Involved			Association
		Extends			Extends
		Includes			Includes
(SA) Architecture			Block Definition		
	Component			Block	
	Properties			Properties	
		Contained In			Generalization
		Contained In			Composition
		Contained In			Aggregation
(SA) Architecture			Internal Block		
	Component			Block Part	
	Port			Port	
		Exchange(Func,Comp, Phys)			Connectors
		Exchange(Func,Comp, Phys)			Item Flows
(SA) Functional Breakdown			Activity		
	Function			Action	
	Ports			Port	
	Control Node			Control Node	
		Functional Exchange			Flow
		Functional Exchange			Control
		Functional Exchange			Object Flow

Summary Example



# Phase 4: Direct Communication

## Common Issues

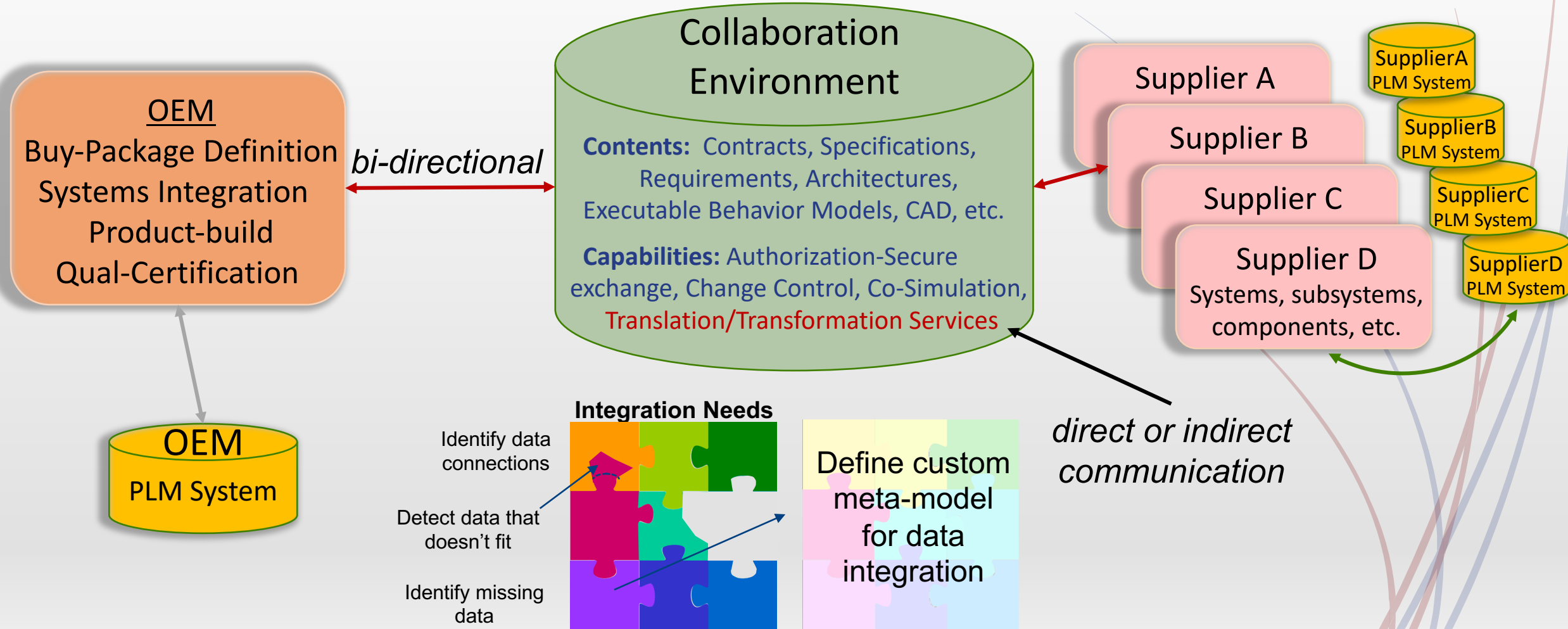
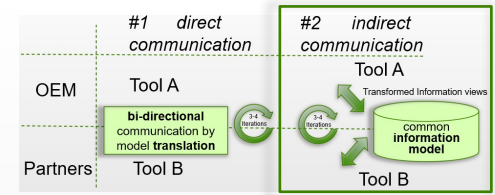


- The implementation of MBSE data standards is not consistent. This impacts the stability, compatibility, compliance and long term choice of any specific vendor's authoring tool.
- What priority each company assigns to MBSE modeling and data standards development
- How to assess the accuracy and completeness of a translation
- How to manage IP protection during model exchange and translation
- How to trade the labor + translation tool costs against the value of the exchange capability
- A tool vendor's on-going support for functionality used by a 3rd party translation service
- How to protect Enterprise tool investments that are impacted by changes to the exchange standards, advances in digital technologies, and redundant spending



# Phase 4: Indirect Communication

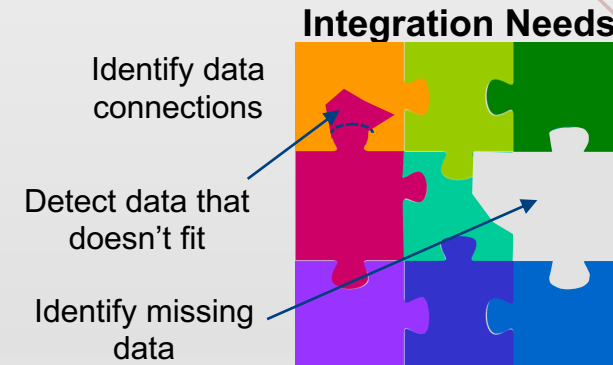
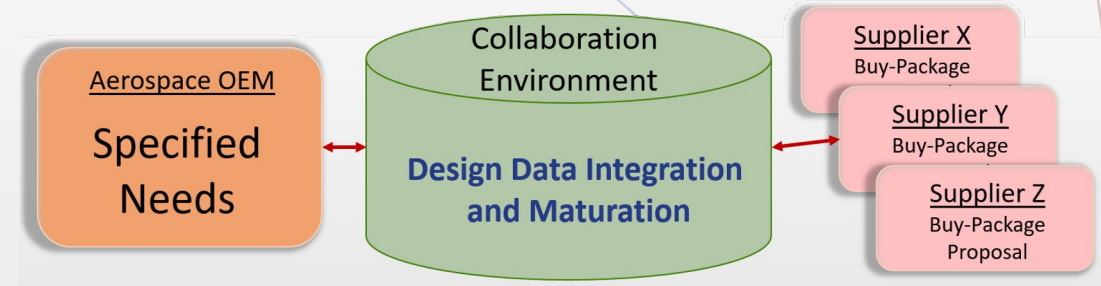
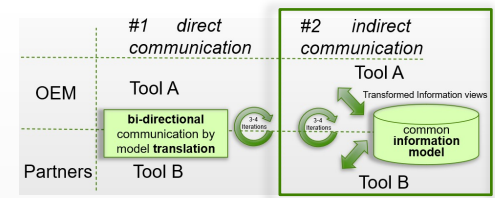
## Sample Infrastructure



# Phase 4: Indirect Communication

## *Challenges to Supplier Collaboration Environment*

- No metamodel, limited integration tools
- Diverse modeling perspectives
- No software support
- No Change/Config management
- Limited IP protections
- Not bi-directional
- Lack of standards



# OEM-Supplier Collaboration: Available Options

*STATUS of the Technology: What can we do now?*

## DIRECT Communication:

1. Limit SysML authoring products to one specific brand
2. Limit SysML authoring products to popular brands and use a direct translation tool

Exec Presentation: May 13, 2022

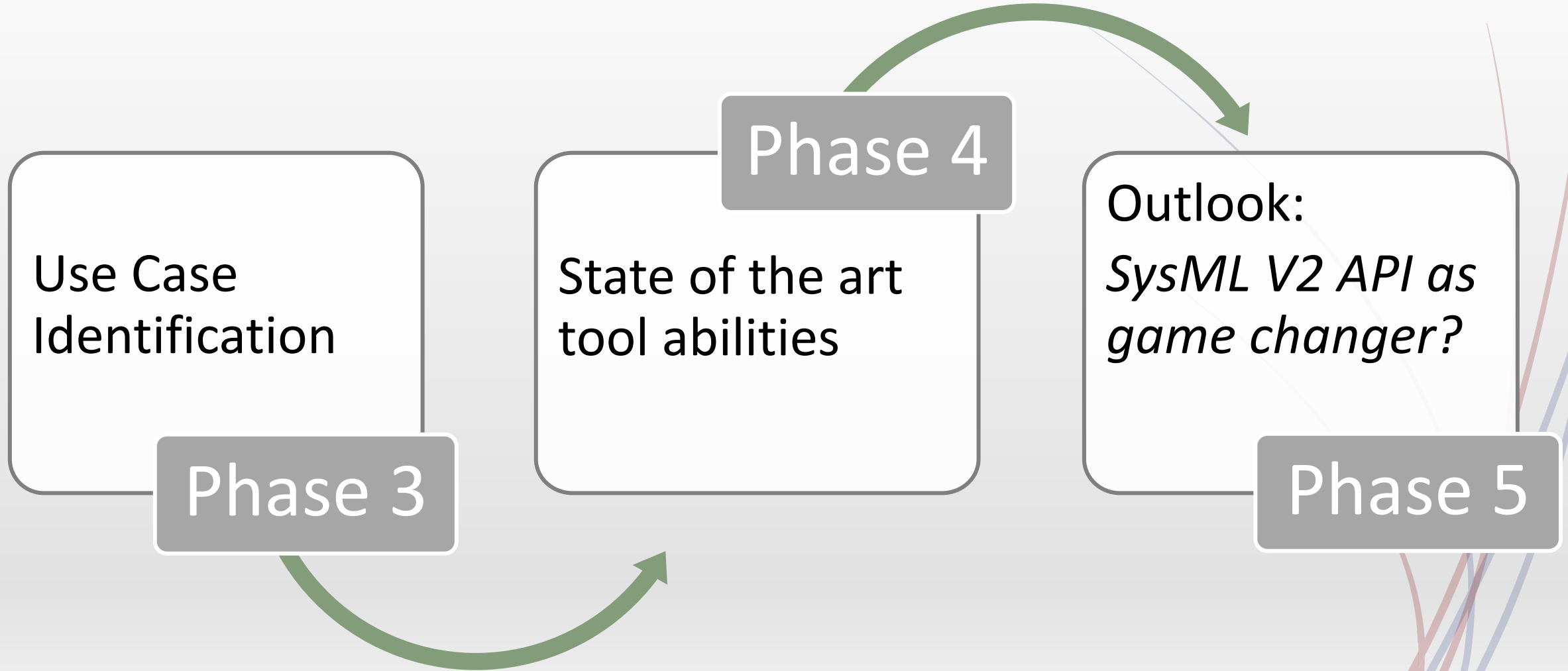
Native Model Source	Translation Tool
Rhapsody Rational Architect	<a href="#">SODIUS Publisher</a>
Cameo/MagicDraw	<a href="#">SODIUS Publisher</a>
System Architect (Unicom)	<a href="#">SODIUS Publisher</a>
Enterprise Architect (SPARX)	<a href="#">SODIUS Publisher</a>
SCADE Architect (Ansys)	<a href="#">SODIUS Publisher</a>

## INDIRECT Communication:

3. Use a in-direct transformation product to integrate multiple model types (requires integration of an additional data management environment)

Transformation Products
<a href="#">DaaS Platform</a> (eQube)
<a href="#">Syndeia</a> (Intercax)
<a href="#">SECollab</a> (SodiusWillert)
<a href="#">SES Engineering Studio</a> (REUSE Co)
<a href="#">SRX</a> (Taliesin Tech)





# Phase 5: Outlook

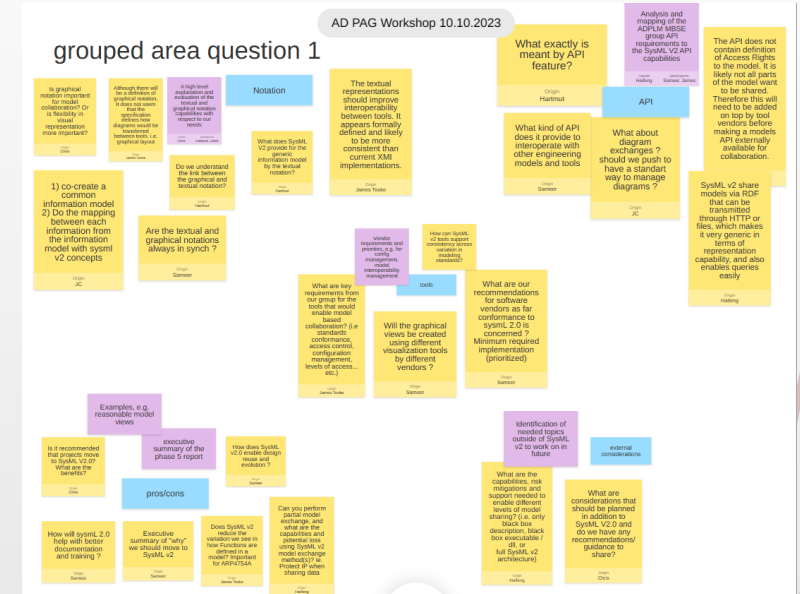
## *SysML V2 as game changer by the new API support?*

### What does SysML v2 provide and not provide, pros and cons?

- What are the other things needed to work beyond SysML v2 (e.g. reference model for aerospace) ?
- Which part of SysML v2 should we leverage? Conformance. What is a MUST and what is NOT?

### Topic clusters to be worked on:

- Notation:** A high level explanation and evaluation of the textual and graphical notation capabilities with respect to our needs
- API:** Analysis and mapping of the ADPLM MBSE group API requirements to the SysML V2 API capabilities
- Tools:** Vendor requirements and priorities, e.g. for config management, model interoperability management





# Phase 5: Outlook

*SysML V2 as game changer by the new API support?*

## Language = Syntax + Semantics + Vocabulary

- Syntax
  - *Concrete*: What you see (rectangles, lines, text).
  - *Abstract*: What you say (“block”, “item flow”)
  - *Interchange/API*: What computers read/write.
- Semantics
  - What’s possible to conclude about the things being modeled when using the syntax.
- Vocabulary (libraries)
  - Predefined syntactic (modeling) elements.

## Requirements (General)

- Uniform **syntactic** interpretation
  - Everyone looking at SysML diagrams should
    - Describe them the same way (using SysML terminology).
    - Agree on whether they are “legal” SysML (well-formedness).
- Uniform **semantic** interpretation
  - Everyone looking at SysML diagrams should
    - Reach the same conclusions about the things being modeled.
    - Including whether it is possible to draw any conclusions at all (consistency).

Every conformant SysML modeling tool shall demonstrate at least Abstract Syntax Conformance and Model Interchange Conformance.

Reference: [omgwiki.org](http://omgwiki.org)

SysML 2.0 Formalism: Requirement Benefits, Use Cases, and Potential Language Architectures Formalism WG, December 6, 2016



# Questions & Answers

---



For more information, visit the AD& D PLM Action Group website at [www.ad-pag.com](http://www.ad-pag.com)



# Backup

