

Department of Defense (DoD) SysML v2 Transition Guide Project

26th Annual National Defense Industrial Association Systems and Mission Engineering Conference

Daniel Hetteema
Director, Digital Engineering, Modeling & Simulation
Office of Systems Engineering and Architecture
Office of the Under Secretary of Defense
for Research and Engineering

Norfolk, Virginia
October 2023





Agenda

- **DEM&S Overview**
- **Context for SysML v2 Transition Guidance**
- **SysML History**
- **How big is the change?**
- **SysML Transition Guidance**
- **Next Steps**



Digital Engineering, Modeling & Simulation's Place in the Federal Government



Joe Biden
President



Lloyd J. Austin III
Secretary of Defense



Heidi Shyu
Under Secretary of Defense
(OUSD) for Research and
Engineering (R&E)

SE&A
Systems Engineering
& Architectures



Tom Simms
Executive Director, SE&A

DEM&S
DIGITAL ENGINEERING
MODELING AND SIMULATION



Daniel Hettema
Director

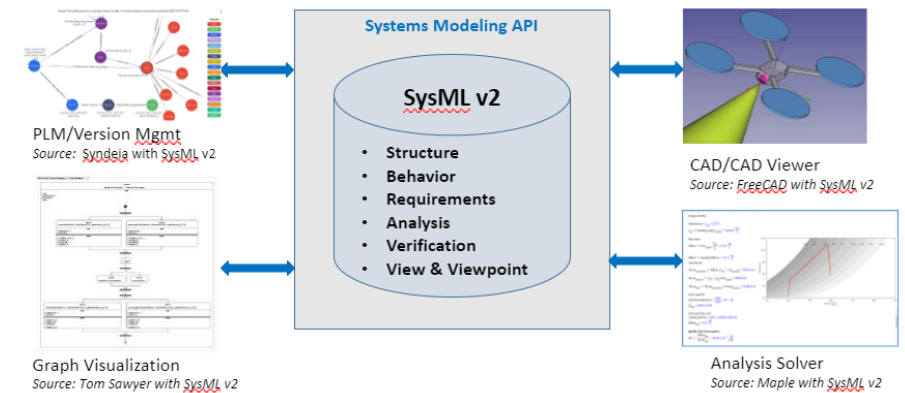


The Future of Systems Engineering Is Model-Based

- Part of the digital transformation
- Full life cycle from SoS to component level
- Agile system development including automated workflow and Configuration Management of the digital thread
- Model patterns and reuse
- Facilitates:
 - The management of complexity & risk
 - More rapid response to change
 - Reuse and design evolution
 - Reasoning about & analysis of the system
 - Shared stakeholder understanding
 - Automated documentation & reporting



Source: INCOSE SE Vision 2035

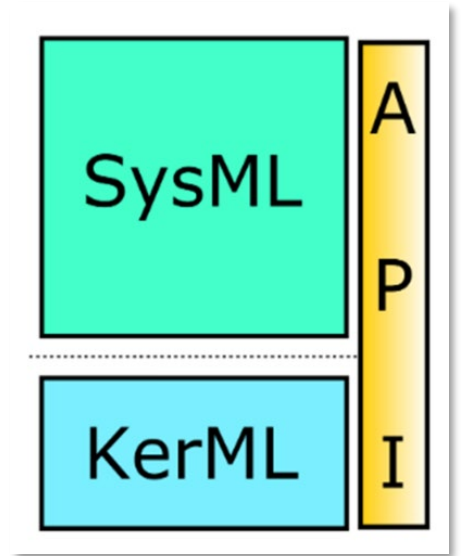


Connecting SysML v2 through the API



Difference between SysML v1 and SysML v2

- **SysMLv2 has a new architecture**
 - KerML: provides core concepts, constructs and semantics
 - SysMLv2: a NEW Language –and revised Methodology
 - Free of dependencies on UML
 - SysMLv2 has two syntaxes: Textual Syntax and Graphical Syntax
 - API: A standardized interface, supports managing model storage, model exchange, and model interaction with a wide variety of external tools (i.e., CAD, PDM, Simulation, Analytics, etc.)





Syntax Differences

SysML v1

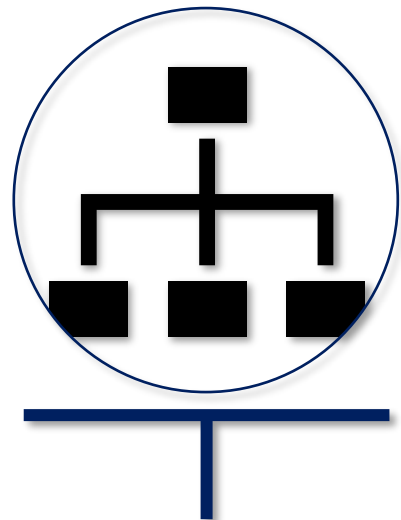
- block / part property
- value property / value type
- proxy port / interface block
- action / activity
- state / state machine
- constraint property / constraint block
- Requirement
- connector / association block
- view

SysML v2

- part / part def
- attribute / attribute def
- port / port def
- action / action def
- state / state def
- constraint / constraint def
- requirement / requirement def
- connection / connection def
- view / view def



What is the scope of the change?



15,000+

models in DoD
*Based on sampling
counts of government
owned ecosystems*



25%

DoD engineers use models
*Based on reported data and
user assumptions of those
ecosystems*

Size estimates do not include contractor ecosystem models and people using them



Collaborating across the community





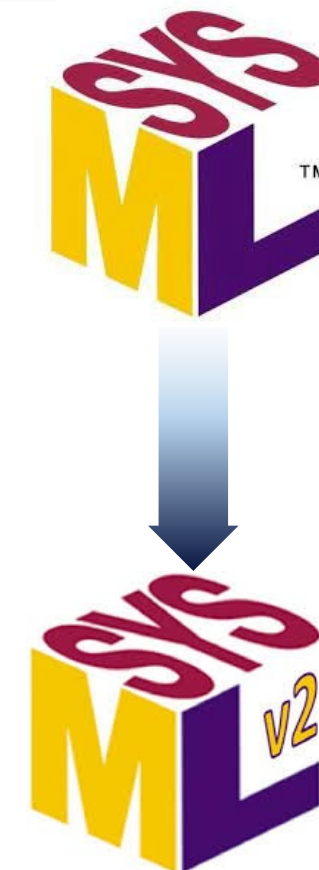
How to Transition from SysML v1 to v2

Derived from ChatGPT, 5/12/2023

Steps to transition from SysML v1 to SysML v2:

- **Familiarize yourself with the changes:** New concepts, syntax, and semantics.
- **Identify the impacts:** Analyze the impacts of the changes in SysML v2 on your existing models, modeling tools, and processes.
- **Update your modeling tools:** Consider switching to a tool that support SysML v2.
- **Convert your existing models:** This may involve manual conversion or using automated tools, depending on the complexity of your models.
- **Update your processes:** This may involve changes in your modeling guidelines, templates, and reviews.
- **Retrain your team:** This may involve training on the new concepts, syntax, and semantics of SysML v2 and any new features in your modeling tools.
- **Plan for a gradual transition:** Gradually transition your existing models to SysML v2 as needed.
- **Communicate and Collaborate:** Collaborate with other stakeholders in your project to ensure they understand the changes and updates, and help them to adapt their processes, tools and methods to align with SysML v2.
- **Establish pilot projects:** Gather and align efforts towards transitioning to v2 and demonstrate initial implementation of a transition plan

A clear plan and a structured approach is needed to successfully transition to SysML v2 and take advantage of the new features and benefits of the language.





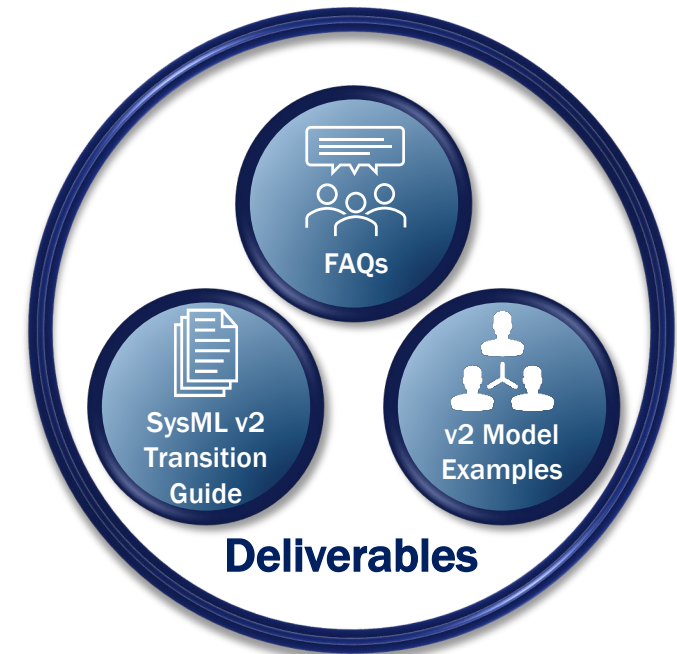
SysML v2 Transition Guidance

Guidance will:

- Address community questions and concerns on transition
- Define differences in syntax, semantics from v1 to v2
- Model examples that further explain guidance principles
- Provide stylistic suggestions to help streamline the modeling process
- Align transition and adoption effort
- Communicate the values of transitioning from v1 to v2

Outcomes:

- Greater transition success
- Faster transition
- Improve modeling practices and model quality
- Avoid duplications of effort
- Share lessons learned





FAQs

- **Sample Questions**

- What changes to the MBSE methodology are required?
- What are the risks in transitioning from SysML V1 to SysML V2?
- What vendors are implementing SysML V2?
- What are the benefits of switching to V2?
- When should DoD programs put SysML V2 on contract (required or requested)?



DEBoK

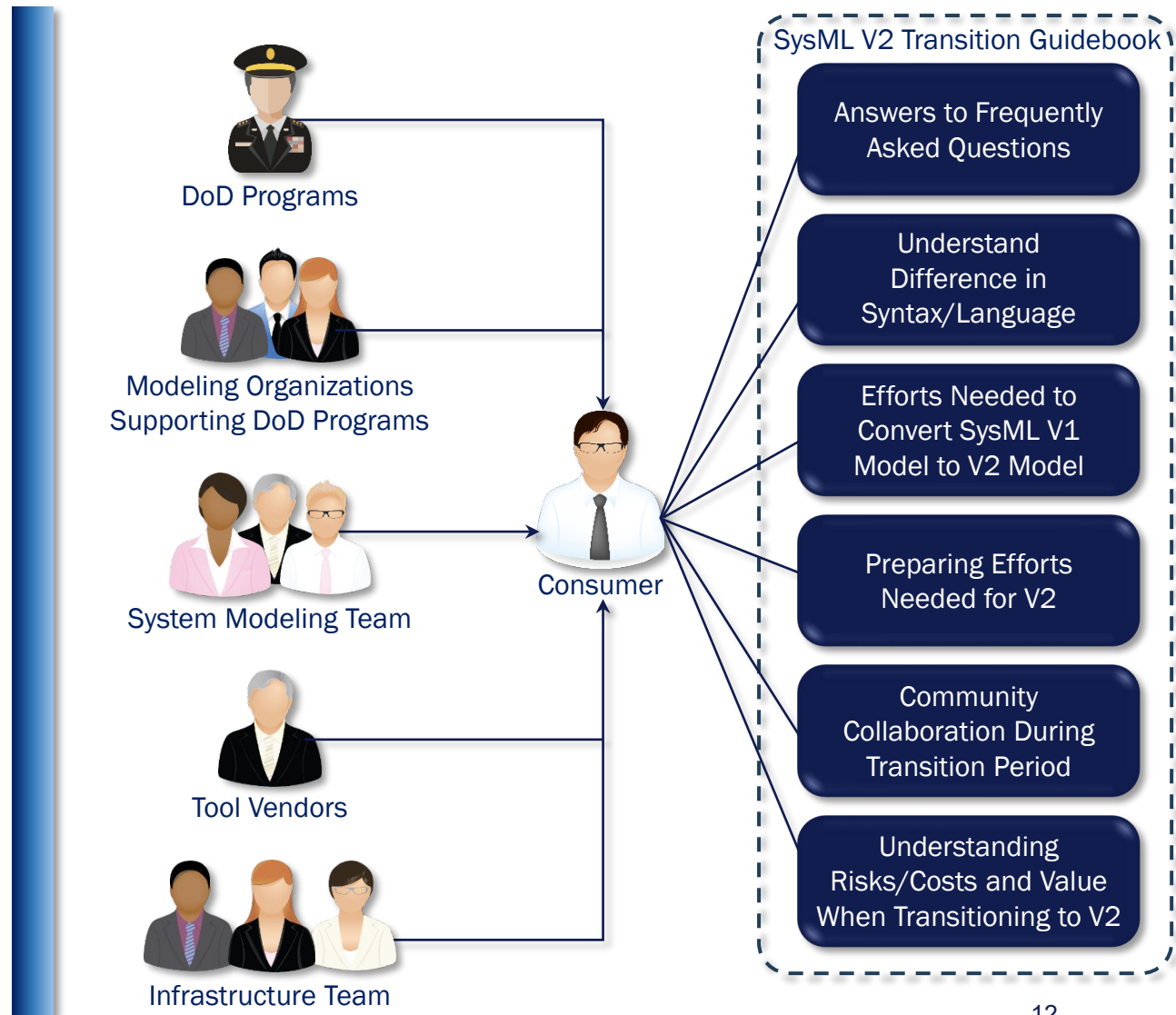
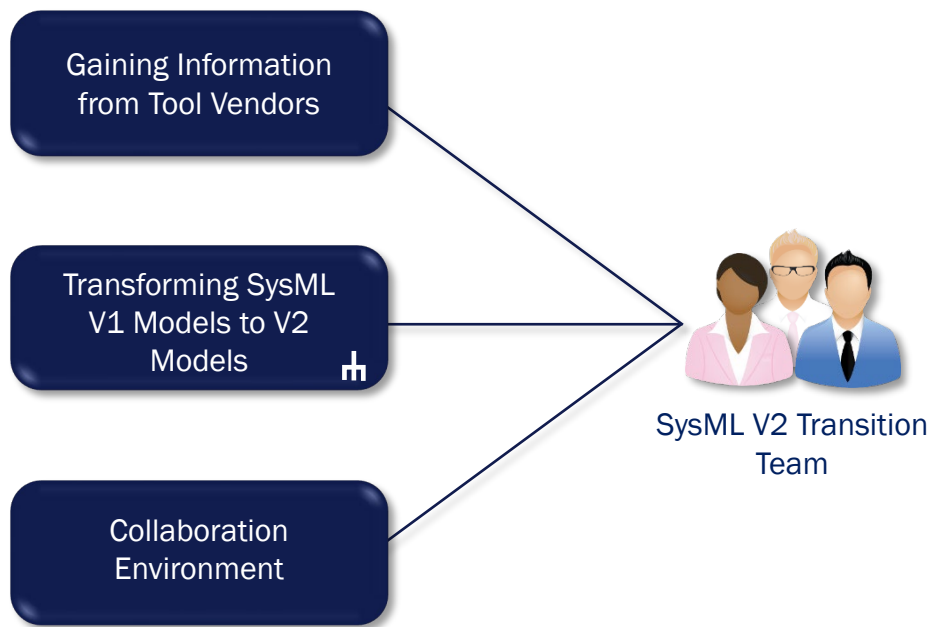


OMG WIKI

50+ FAQs across 12 categories



Transition Guidance Use Cases





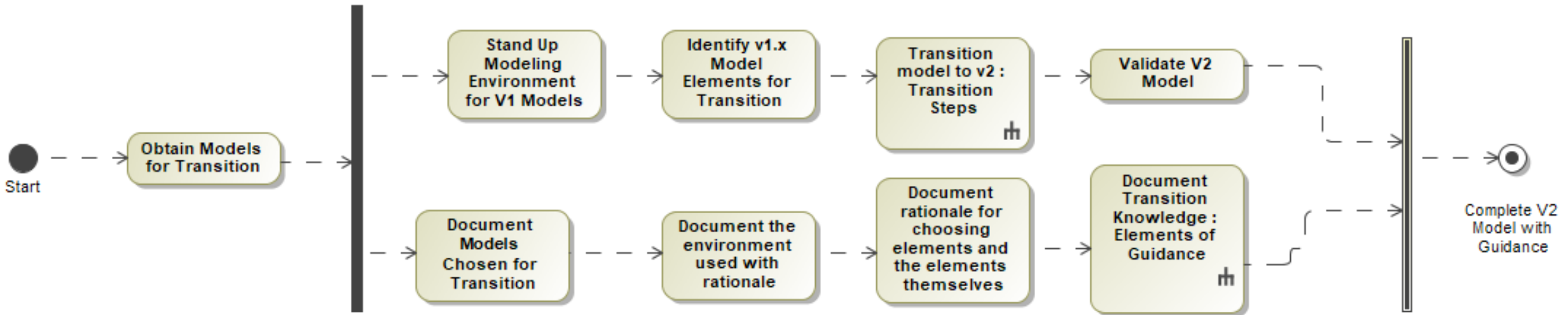
Modeling in v2 Today

- SysML V2 Kernel is publicly available now on SysML v2 Submission Team (SST) Git repository <https://github.com/Systems-Modeling/SysML-v2-Release>
- Install is available for Eclipse and Jupyter Labs
- Installation instructions available for Jupyter Labs on OMG Wiki
- DoD practitioners can utilize ARCUS to install v2 environment



SysML v1 to SysML v2 Model Conversion

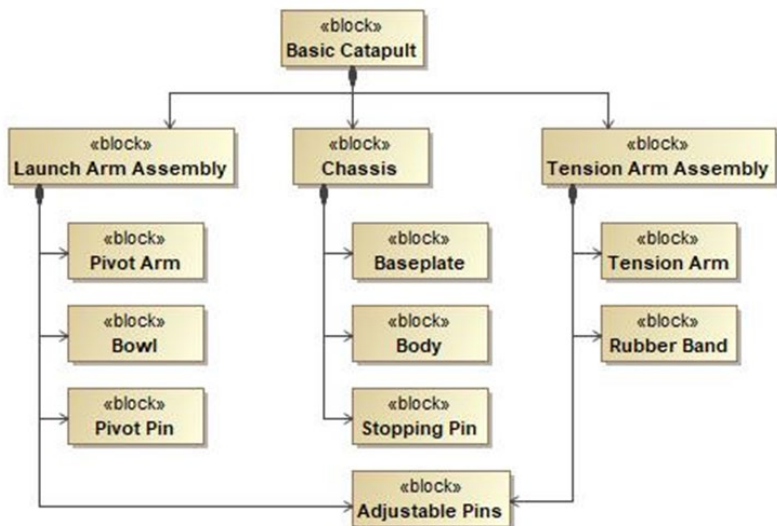
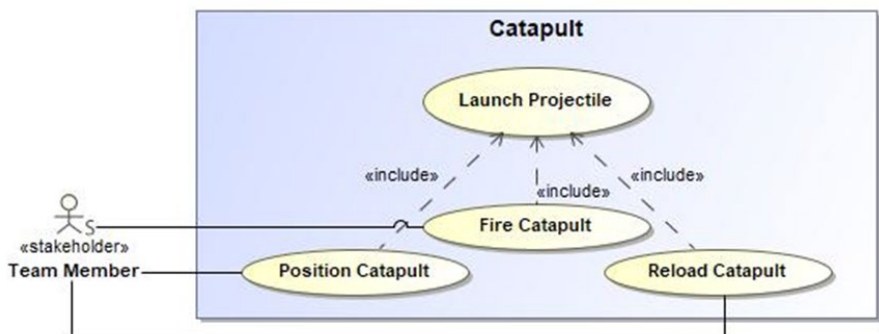
- Use models originally developed in SysML v1.x and convert a selection to SysML v2
 - Pre-process v1 model to remove customizations, plug-ins, match v1 spec
 - Transform v1 model into v2 environment
 - Post-process v2 model to use v2 constructs, validate model once complete
- As this is being worked, there will be ongoing dialogue with the community



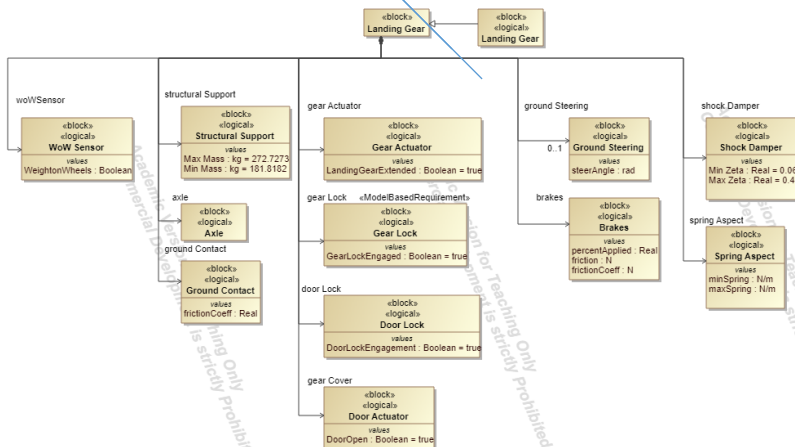
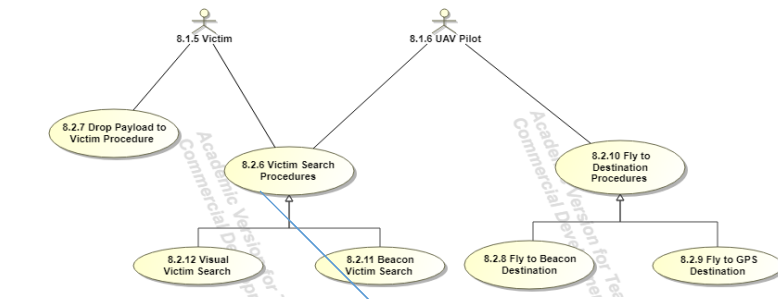


Sample Transition Models

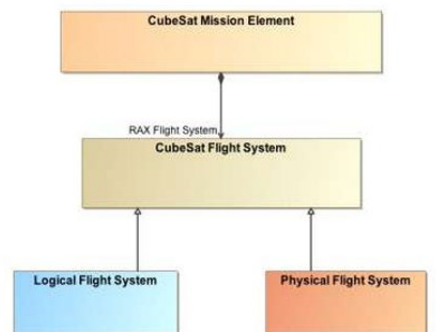
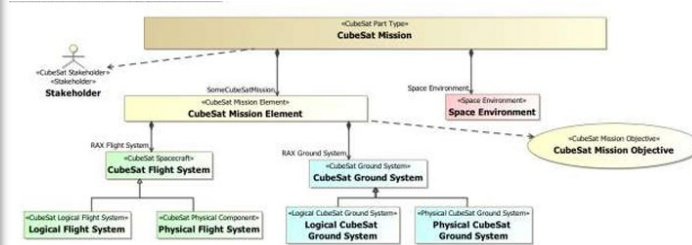
Armaments Catapult Model



Skyzer Surrogate Pilot Model

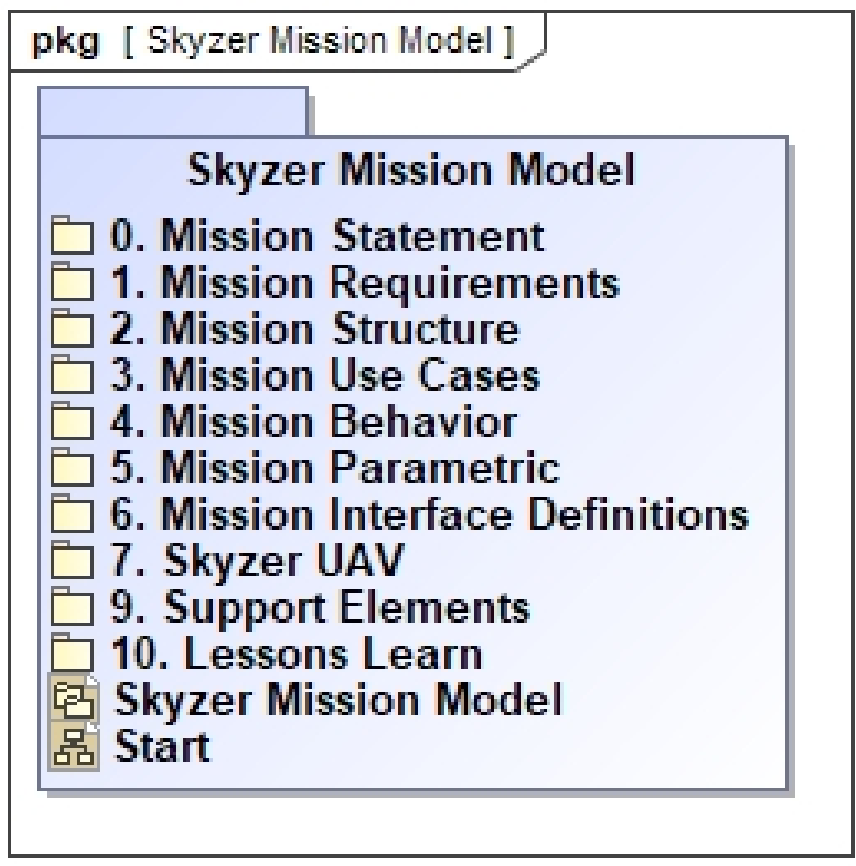


CubeSat Reference Model





Conversion Example – Package Structure



Skyzer SysML v1 Mission Model Package Structure

```
package SkyzerMissionModel_transformed{
    import '0.MissionStatement'::*;
    import '1.MissionRequirements'::*;
    import '2.MissionStructure'::*;
    import '3.MissionUseCases'::*;
    import '11.LanguageCustomization'::*;

    package '0.MissionStatement'{↔}
    package '1.MissionRequirements'{↔}
    package '2.MissionStructure'{↔}
    package '3.MissionUseCases'{↔}
    package '4.MissionBehavior'{↔}
    package '5.MissionParametric'{↔}
    package '6.MissionInterfaceDefinitions'{↔}
    package '7.SkyzerUAV'{↔}
    package '9.SupportElements'{↔}
    package '10.LessonsLearned'{↔}
    package '11.LanguageCustomization'{↔}
}
```

Skyzer SysML v2 Mission Model Package Structure



Conversion Example – Operational View



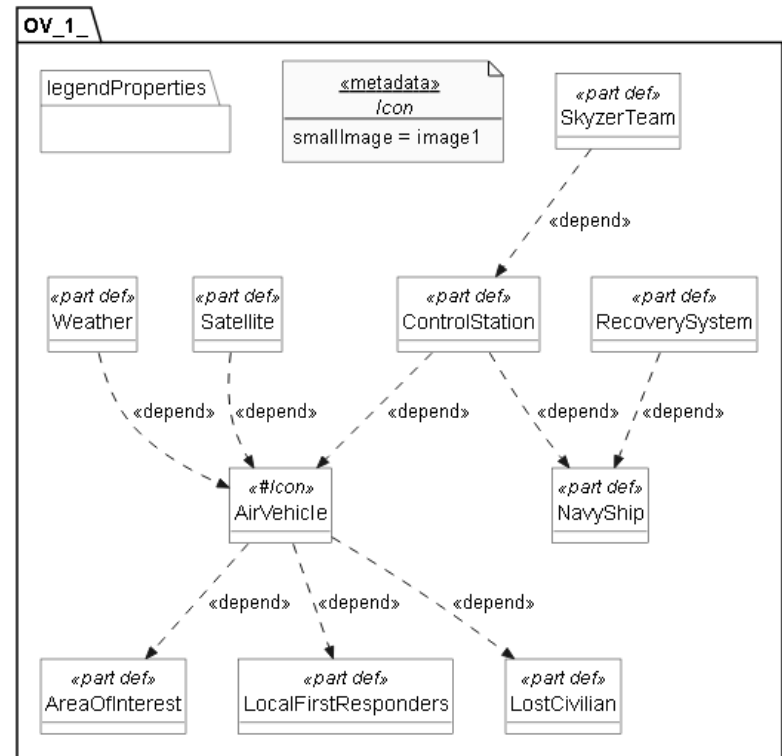
Skyzer SysML v1 Mission Model OV1 Diagram

```

package OV_1{
  package legendProperties{
    // map to a rendering method
  }
  dependency from AirVehicle to AreaOfInterest;
  dependency from AirVehicle to LocalFirstResponders;
  dependency from AirVehicle to LostCivilian;
  dependency from ControlStation to AirVehicle;
  dependency from ControlStation to NavyShip;
  dependency from RecoverySystem to NavyShip;
  dependency from Satellite to AirVehicle;
  dependency from SkyzerTeam to ControlStation;
  dependency from Weather to AirVehicle;
  part def AirVehicle{
    @Icon{
      smallImage = image1;
    }
  }
  part def AreaOfInterest;
  part def LocalFirstResponders;
  part def LostCivilian;
  part def ControlStation;
  part def NavyShip;
  part def RecoverySystem;
  part def Satellite;
  part def SkyzerTeam;
  part def Weather;
}
  
```

OV1 Skyzer SysML v2 Mission Model

%viz --view tree --style comptree SkyzerModel_v2::SkyzerMissionModel_transformed::OV_1_



OV1 Skyzer SysML v2 Mission Model OV1 View



SysML v1 to v2 Transition Project Plan

			2022	2023				2024				
	WBS	Phase	Task Area	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
	1		Phase 1: Transition Guidance									
	1.1		Outreach									
	1.1.6		NDIA 10/16/23					X				
	1.1.7		INCOSE IW 01/24						X			
	1.2		Establish SysML V2 Transition Team (STT)									
	1.3		SysML V1 to V2 Transition FAQ		D1	D2	R1	R2				
	1.4		SysML V1 to V2 Model Conversion Guidance				D1	R1	R2			
	1.5		Transition Plan Template					D1 D2	R1			
	1.6		SysML V2 Modeling Environment			D1	R1					
	1.7		SysML V2 Training					D1	D2 R1			
	1.8		Identify Other SysML V2 Transition Activities					D1	R1			
	1.9		SysML V1 to V2 Transition Guidance Website				D1		R1 R2			
	2		Phase 2: Organizational Transition (pilots)									
	3		Phase 3: Project Deployment									



Next Steps

- **Continue work on transition guidance products**
- **Monthly community meetings**
 - Dec 6th
- **Update products based on feedback**
- **Prepare for IW**

Please let us know of any outreach opportunities



Contact Info

**Office of the Under Secretary of Defense for
Research and Engineering**

Systems Engineering and Architecture

osd-sea@mail.mil | Attn: DEM&S

<https://www.cto.mil/sea>