



# SysML v2 Introduction

SysML v2 Summit  
Reston, VA  
March 20, 2025

Sanford Friedenthal  
OMG SMC Cochair  
INCOSE Liaison to OMG  
[safriedenthal@gmail.com](mailto:safriedenthal@gmail.com)



# Agenda

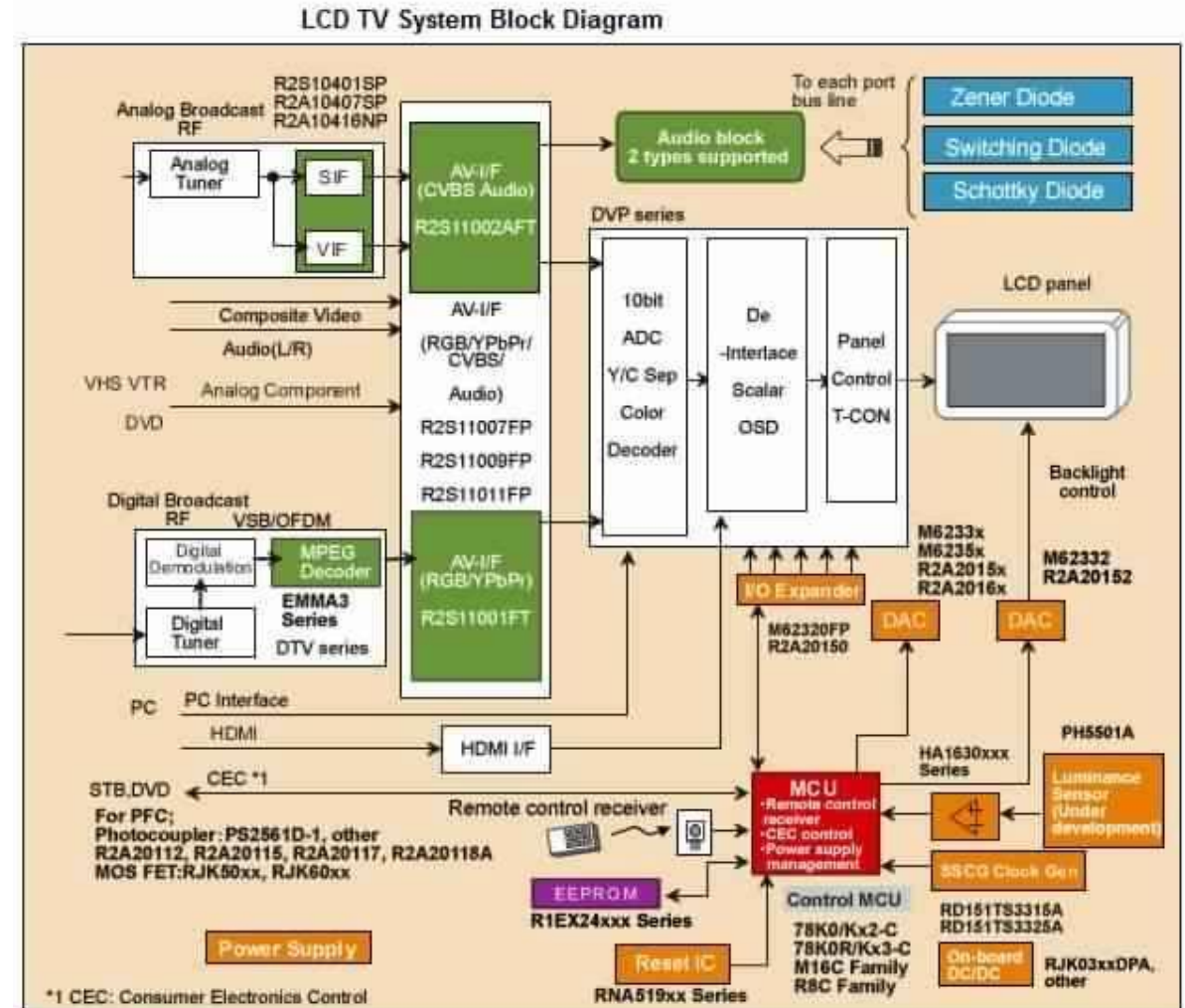
- MBSE Background
- SysML v2 Status
- SysML v2 Overview & Comparison with SysML v1
- Extended SysML v2 Community
- Representative SMC Working Group
- Wrap-up



# MBSE Background

# Traditional System Block Diagram

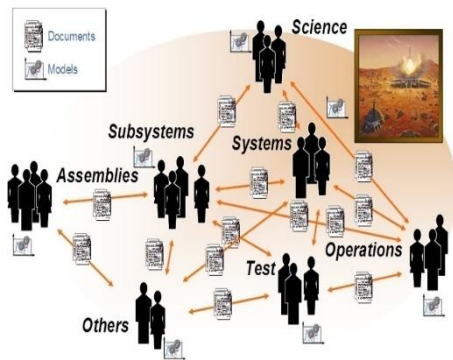
- System architecture captured using informal diagramming notation
- Good domain content but imprecise description of:
  - Component hierarchy
  - Interfaces
  - Functions vs components
  - Succession vs connection
- Disconnected from other system views
- Lack of traceability



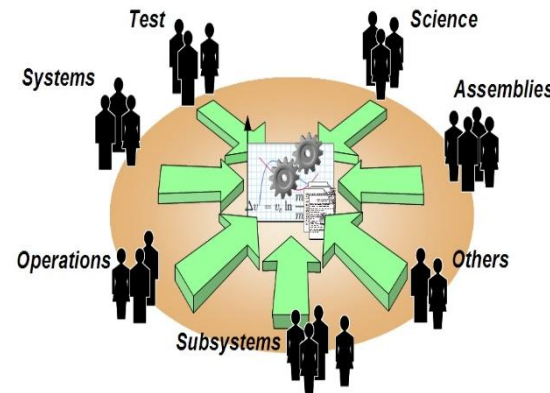


# Model-Based Systems Engineering (MBSE)

- A systems engineering approach where information about the system is captured in a system model
  - The model is the source of the data and managed throughout the lifecycle
- Contrasts with a document-based approach where the information is captured in a variety of documents, informal diagrams, and spreadsheets
- Provides a more complete, consistent, and traceable system design



**From:** System specification and design data related through documents



**To:** Shared system model with multiple views, and connected to discipline models

Source: Jet Propulsion Laboratory



# SysML v2 Status



# SysML v2 Status

- SysML v2 RFP issued by the OMG in December, 2017
- SysML v2 specifications developed by the SysML v2 Submission Team (SST)
- SysML v2 beta specifications (i.e., KerML, SysML v2, Systems Modeling API & Services) approved by the OMG in June 2023
- Submitted final specifications for adoption in March, 2025
  - Proceed through OMG process (review, voting, spec editing, ...)
    - Planned adoption in June 2025
    - Formal publication anticipated in September 2025
- Tool vendors working on their implementations
  - Anticipate commercial and open-source tools in 2025
- OMG Systems Modeling Community (SMC) in collaboration with INCOSE advancing the modeling practice with SysML v2



# SysML v2 Overview & Comparison with SysML v1





# SysML v2 Objectives

- **Increase adoption and effectiveness of MBSE with SysML v2 to support the evolving MBSE practice by enhancing...**
  - Precision and expressiveness of the language
  - Consistency and integration among language concepts
  - Interoperability with other engineering models and tools
  - Usability by model developers and consumers
  - Extensibility to support domain specific applications
  - Migration path for SysML v1 users and implementors

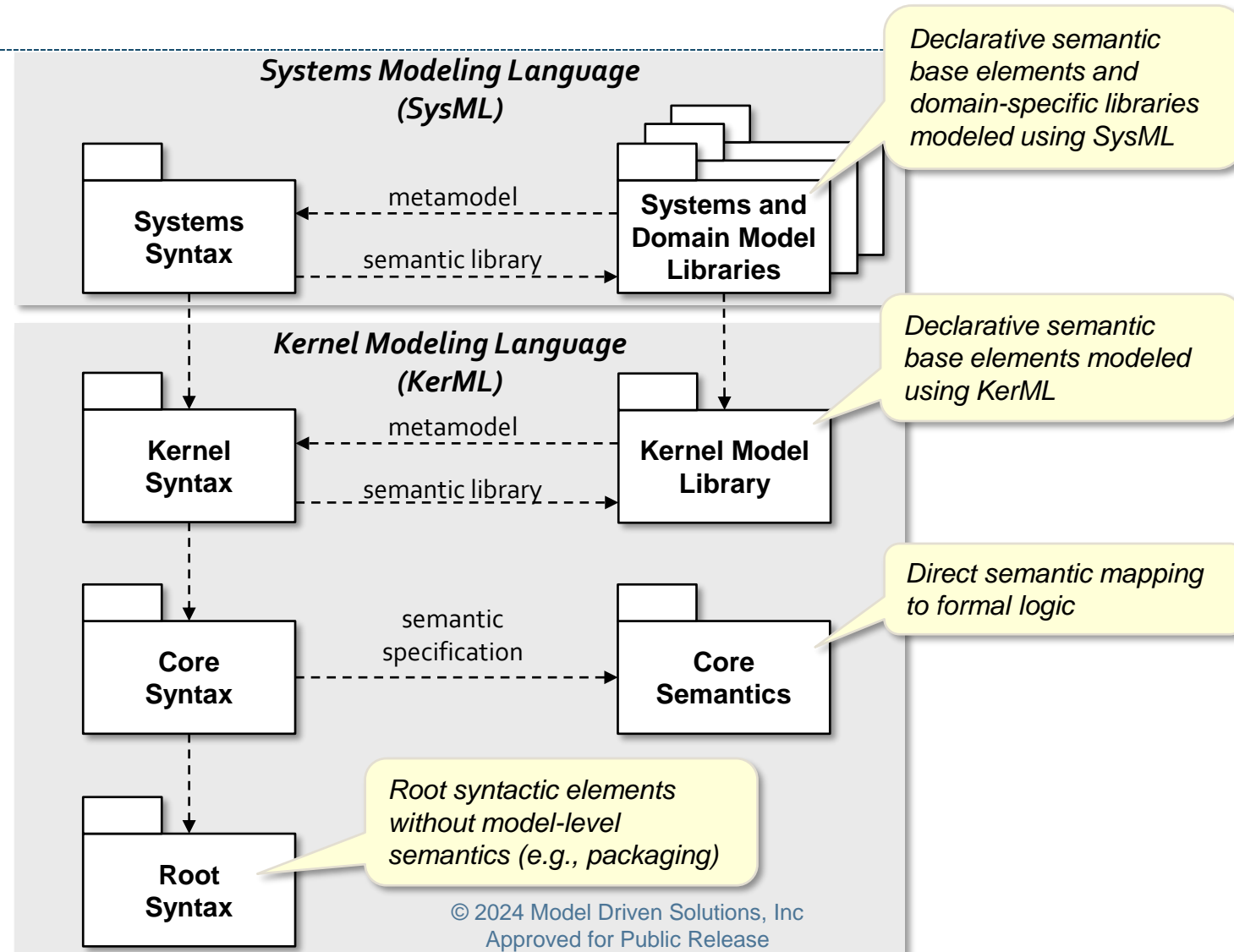


# Key Elements of SysML v2

- New Metamodel that is not constrained by UML
  - Preserves most of UML modeling capabilities with a focus on systems modeling
  - Grounded in formal semantics
- Robust visualizations based on flexible view & viewpoint specification
  - Graphical, Tabular, Textual
- Standardized API to access the model

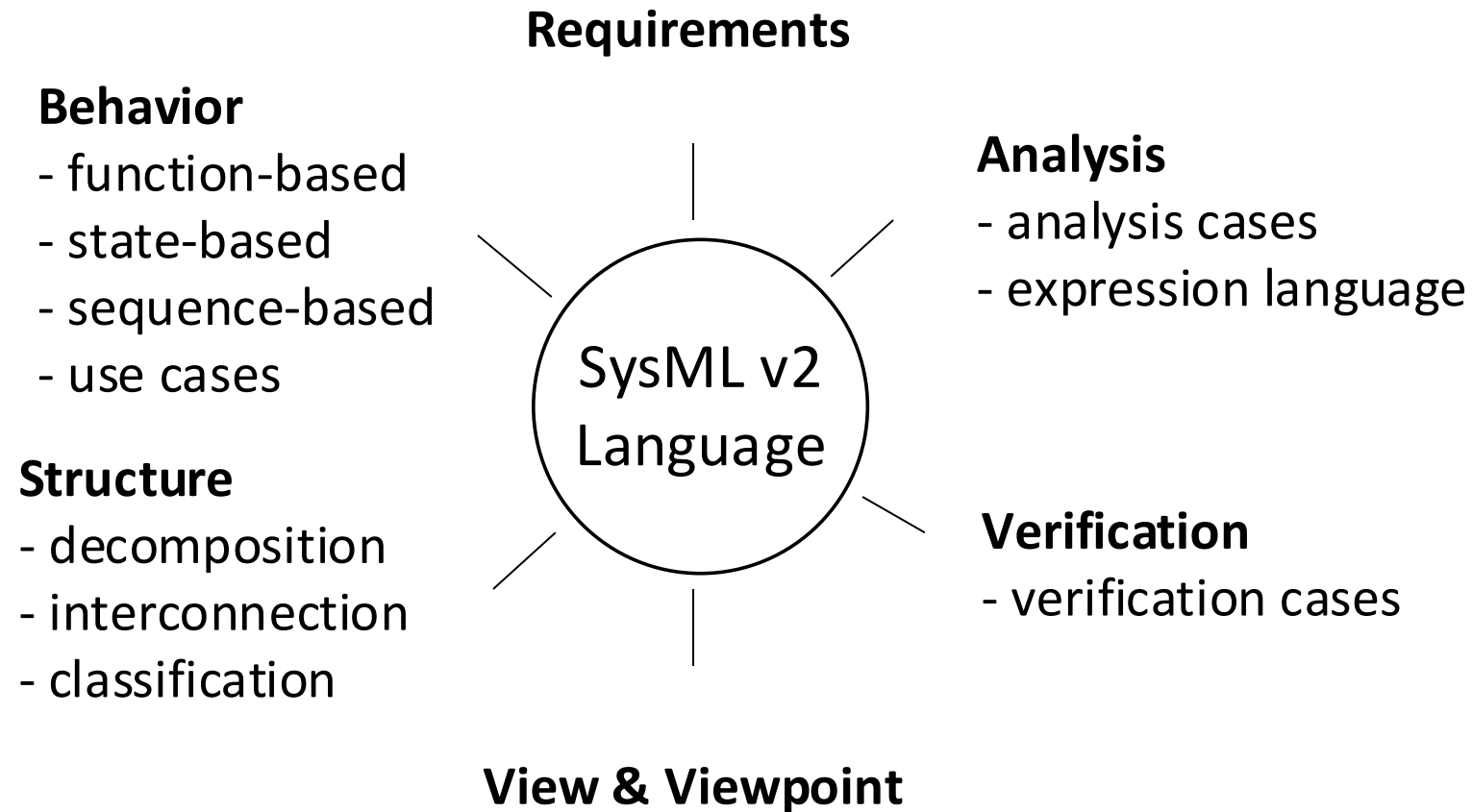


# SysML v2 Language Architecture





# SysML v2 Language Capabilities





# SysML v2 Reuse Pattern

- Definition and usage
  - A definition element defines an element such as a part, action, or requirement
  - A usage element is a usage of a definition element in a particular context
  - Pattern is applied consistently throughout the language



# SysML v2 to v1

## Terminology Mapping (partial)

SysML v2	SysML v1
part / part def	part property / block
attribute / attribute def	value property / value type
port / port def	proxy port / interface block
action / action def	action / activity
state / state def	state / state machine
constraint / constraint def	constraint property / constraint block
requirement / requirement def	requirement
connection / connection def	connector / association block
view / view def	view

*SysML v2 applies a consistent pattern of definition and usage*



# Acknowledgement

- The SysML v2 diagrams in this slide set were created using an early version of Dassault Systèmes Cameo Systems Modeler / CATIA Magic Cyber Systems Engineer.

Add one of these to every slide that has diagrams produced from CSM/MCSE:

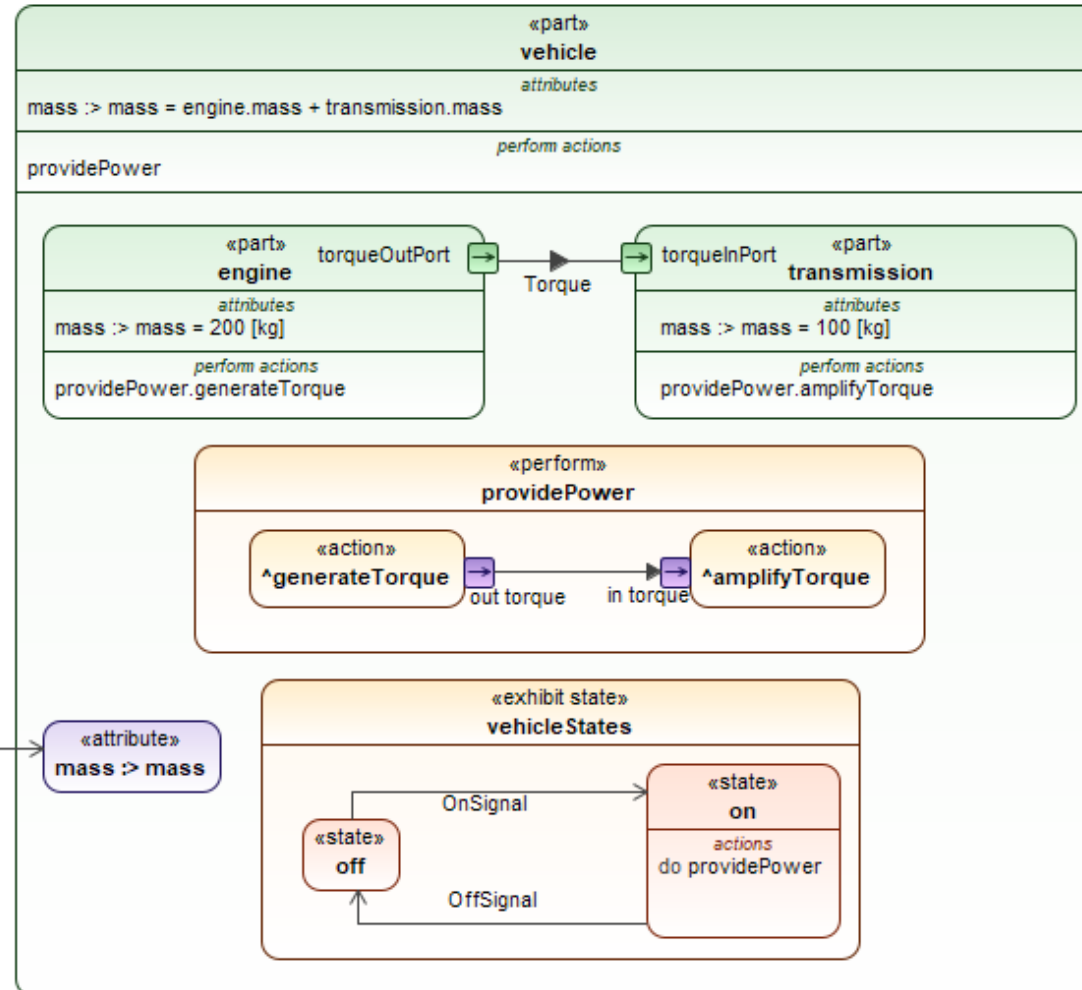
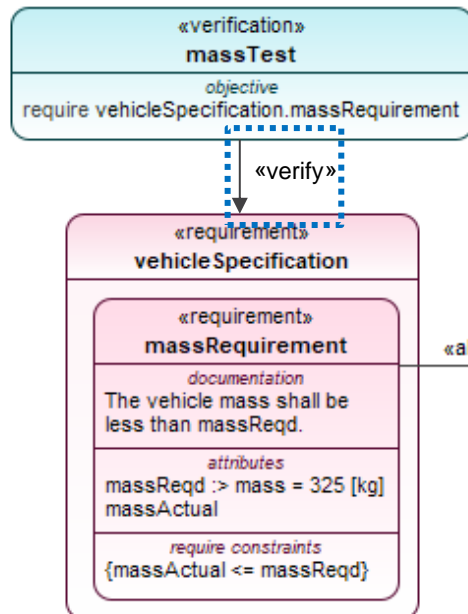
Produced with  
  
*Magic / Cameo Systems Modeler*

Produced with  *Magic  
Cameo Systems Modeler*

# Vehicle Example

Produced with  
**CATIA**  
 Magic | Cameo Systems Modeler

*Verify not  
from tool*



```

package vehicle_example {
  view vehicle_example;
  private import SI::*;
  item def OnSignal;
  item def OffSignal;
  attribute def Torque;

  part vehicle {
    attribute mass := ISQBase::mass = engine.mass + transmission.mass;
    perform providePower;
    part engine {
      attribute mass := ISQBase::mass = 200 [kilogram];
      port torqueOutPort {
        out attribute torque : Torque;
      }
      perform providePower.generateTorque;
    }
    part transmission {
      attribute mass := ISQBase::mass = 100 [kilogram];
      port torqueInPort {
        in attribute torque : Torque;
      }
      perform providePower.amplifyTorque;
    }
    connect engine.torqueOutPort to transmission.torqueInPort {
      flow of Torque from source.torque to target.torque;
    }
    exhibit state vehicleStates {
      state off;
      state on {
        do providePower;
      }
      transition off_to_on first off accept OnSignal then on;
      transition on_to_off first on accept OffSignal then off;
    }
  }

  action providePower {
    action generateTorque { out attribute torque; }
    action amplifyTorque { in attribute torque; }
    flow generateTorque.torque to amplifyTorque.torque;
  }

  requirement vehicleSpecification {
    requirement massRequirement {
      doc /* The vehicle mass shall be less than massReqd. */
      attribute massReqd := 325 [kilogram];
      attribute massActual;
      require constraint {massActual <= massReqd}
    }
  }

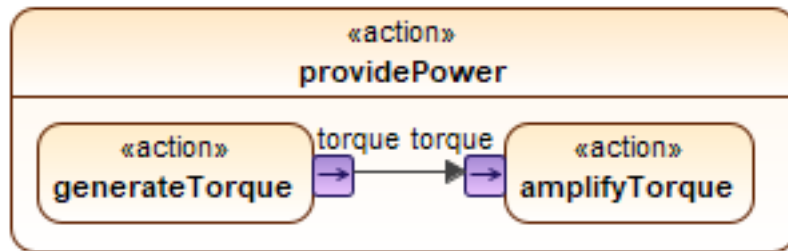
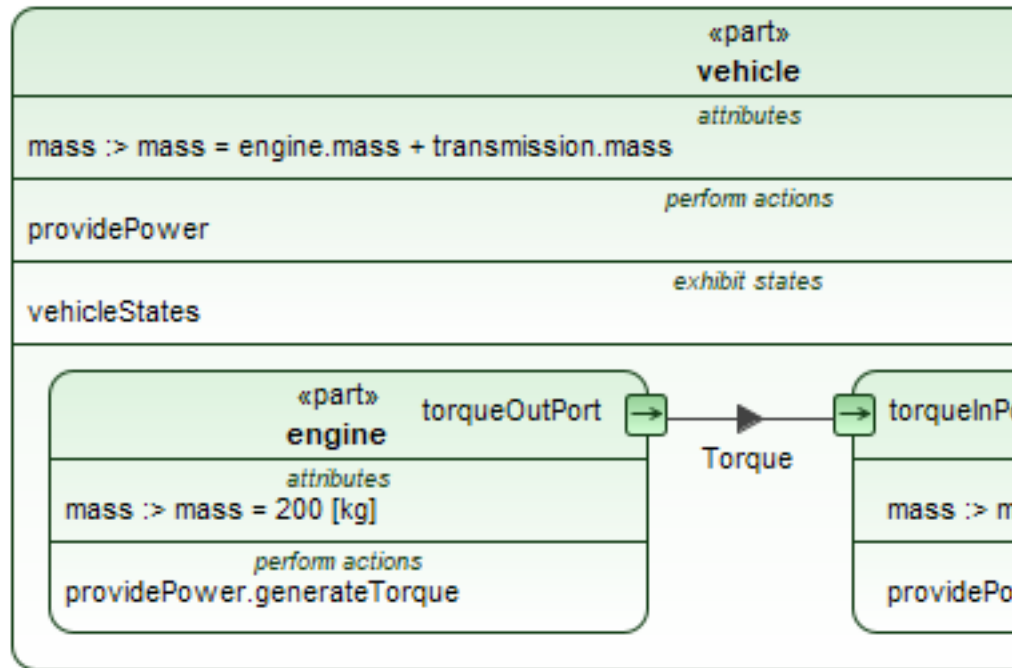
  allocate vehicleSpecification.massRequirement to vehicle.mass;
  verification massTest {
    objective {
      verify vehicleSpecification.massRequirement;
    }
  }
}

```



# Vehicle Example

## Graphical/Textual Correspondence



```

7  part vehicle {
8      attribute mass :> ISQBase::mass = engine.mass + transmission.mass;
9      perform providePower;
10  exhibit state vehicleStates {↔}
18  part engine {
19      attribute mass :> ISQBase::mass = 200 [kg];
20  port torqueOutPort {
21      out attribute torque : Torque;
22  }
23  perform providePower.generateTorque;
24  }
25  part transmission {
26      attribute mass :> ISQBase::mass = 100 [kg];
27  port torqueInPort {
28      in attribute torque : Torque;
29  }
30  perform providePower.amplifyTorque;
31  }
32  connect engine.torqueOutPort to transmission.torqueInPort {↔}
35  }
36  action providePower {
37      action generateTorque { out attribute torque; }
38      action amplifyTorque { in attribute torque; }
39      flow generateTorque.torque to amplifyTorque.torque;
40  }
  
```

Proc





# Nuclear Fission Model in SysML V2

## Source: SysML v2 Release Group

<vinay\_paliya@yahoo.com>

Feb 3, 2025, 12:36:15 PM

to SysML v2 Release

Friends,

I am trying to build a simple model of PWR Nuclear power plant on SysML V2. PWR produces energy by Nuclear Fission and I plan to develop the model bottoms-up starting with fundamental particles (Neutron, Proton, Electron) and going upto powerplant.

I created many elements (H, He, B, U235, Pu..) and compounds (H2O, H3BO3) etc to represent their Nuclear properties as used in PWR. The file attached herewith for your reference and comments.

I am wondering is there any better way (array or collection) to create full element table with all chemical and Nuclear properties? Is there any such library available?

Regards,  
Vinay

```
attribute u:MassUnit = 1.66054 * 10^-27 [kg];
attribute e:ElectricChargeUnit = 1.6*10^-19 [C];

part def Particle {
    attribute mass:MassValue;
    attribute charge:ElectricChargeValue;
}

part def Proton :> Particle {
    :>> mass = 1.0072766 [u];
    :>> charge = 1[e];
}

part def Neutron :> Particle {
    :>> mass = 1.008665 [u];
    :>> charge = 0 [e];
}

part def Electron :> Particle {
    :>> mass = 1/1837 [u];
    :>> charge = -1 [e];
}

part def Atom {
    attribute am:MassValue = pp.mass + nn.mass + ee.mass - bm;
    attribute bm:MassValue default = 0; /*mass equivalent to binding energy also called as mass defect */
    part pp:Proton [1..118];
    part nn:Neutron [0..176];
    part ee:Electron [1..118];
    assert constraint PnE { size(pp) == size(ee) }
}

part def <H>Hydrogen_Isotops :>Atom {
    :>> pp[1];
    :>> nn[0..2];
    :>> ee[1];
}
```

Example model  
from the community





# Comparing SysML v2 with SysML v1

- **Simpler to learn and use**

- Systems engineering concepts designed into metamodel versus added-on
- Consistent application of definition and usage pattern
- More consistent terminology
- Ability to decompose parts, actions,
- More flexible model organization with package filters

- **More precise**

- Textual syntax and expression language
- Formal semantic grounding
- Requirements as constraints

- **More expressive**

- Variant modeling
- Analysis case
- Trade-off analysis
- Individuals, snapshots, time slices
- More robust quantitative properties (e.g., vectors, ..)
- Simple geometry
- Query/filter expressions
- Metadata

- **More extensible**

- Simpler language extension capability
  - Based on model libraries

- **More interoperable**

- Standardized API



# Extended SysML v2 Community



# OMG Systems Modeling Community (SMC)

<https://www.omg.org/communities/systems-modeling-community.htm>

- Working groups focused on advancing the SysML v2 specifications and practices
  - API & Services WG – Manas Bajaj
  - Conformance WG - Vince Molnár
  - Digital Engineered Verification WG – Mark Malinoski
  - Execution WG – Richard Page
  - Formal Methods WG - Vince Molnár
  - Graphical Specification WG – Sandy Friedenthal
  - Real-Time Embedded Safety-Critical Systems WG - Jerome Hugues, Gene Shreve
  - Reference Implementation WG – Ed Seidewitz
  - Semantics WG – Conrad Bock, Karen Ryan
  - SysML v2 Certification WG – Rick Steiner
  - Transformation & Interoperability WG – Yves Bernard
  - User WG – Sandy Friedenthal



# OMG Systems Modeling Community (SMC)

## SysML v2 Public Repositories

- Open-source repositories
  - <https://github.com/Systems-Modeling>
- Current SysML v2 release
  - <https://github.com/Systems-Modeling/SysML-v2-Release>
- Release content
  - Specification documents for KerML, SysML and API
  - Training material for SysML textual notation
  - Training material for SysML graphical notation
  - Example models in textual notation (sysml/src)
  - Pilot implementation
    - Installer for Jupyter tooling
    - Installation site for Eclipse plug-in
  - Web access to prototype repository via SysML v2 API
- Google group for comments and questions
  - <https://groups.google.com/g/SysML-v2-Release>  
(to request membership, provide name, affiliation and interest)



# INCOSE

- Broad OMG/INCOSE Collaboration via Memorandum of Understanding (MOU)
- Participation with INCOSE (Chapters, Working Groups, IW, IS)
- OMG Systems Modeling Standards Supported by INCOSE
  - Systems Modeling Language (SysML)
  - Unified Architecture Framework (UAF)
  - Risk Analysis and Assessment Modeling Language (RAAML)
  - Cubesat System Reference Model (CSRM)





# SysML v2 Vendor Support

- The following vendors participated in the Vendor Roadmap Session at the INCOSE IW
  - **Ansys \***
  - **Celedon**
  - **Dassault Systèmes \***
  - **Ellidiss**
  - **IBM \***
  - **IncQuery \***
  - **Intercax \***
  - **LieberLieber**
  - **MathWorks \***
  - **Mgnite Inc. \***
  - **Obeo \***
  - **PTC**
  - **SBE Vision**
  - **Sensmetry \***
  - **Siemens**
  - **Sodius Wilert**
  - **Sparx**
  - **Tom Sawyer Software \***
  - **Vendor Booths at the SysML v2 Summit \***
    - **Qualtech Systems \***
    - **SysGit \***
    - **Model Driven Solutions \***

*SysML v2 Vendor Roadmap Videos from INCOSE IW at [mbse:incose\\_mbse\\_iw\\_2025](#) [MBSE Wiki]*



# A SMC Working Group Example

## *SMC Formal Methods*

### *Lead: Vince Molnár (BME)*



# Formal Methods WG Charter

*The mission of the Formal Methods WG is to **facilitate the application of formal methods** on KerML and SysML v2 models by sharing experience in the **integration of formal methods tools**, defining **precise mappings** to formal languages, and **providing feedback** to other working groups.*

## Possible community projects:

- Domain-specific libraries (e.g., temporal logic)
- Transformation rules to formal languages



# Formal Methods WG Charter – Where We Are

- *facilitate the application of formal methods*
  - *integration of formal methods tools*
  - *precise mappings*
  - *providing feedback*

# Formal Methods WG Charter

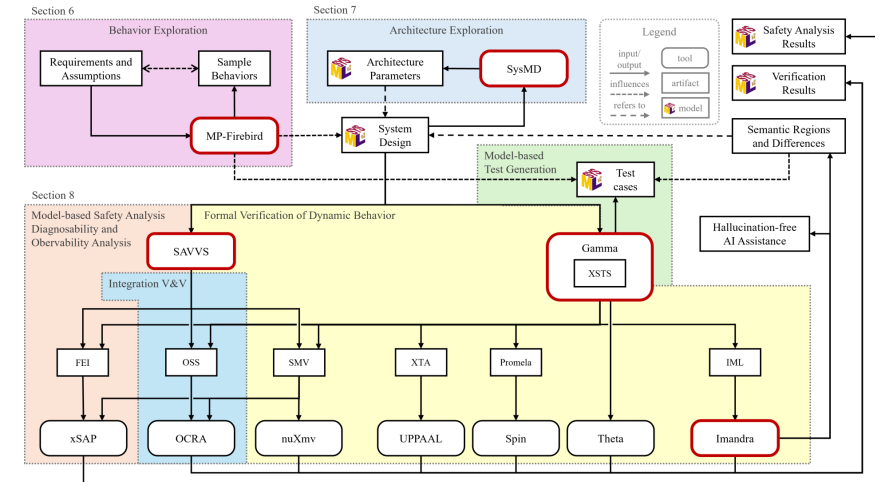
- *facilitate the application of formal methods*

- *integration of formal methods tools*

Vince Molnár<sup>1\*</sup>, Bence Graics<sup>1,6†</sup>, András Vörös<sup>1†</sup>, Stefano Tonetta<sup>2†</sup>, Luca Cristoforetti<sup>2†</sup>, Greg Kimberly<sup>3†</sup>, Pamela Dyer<sup>4†</sup>, Kristin Giammarco<sup>4†</sup>, Manfred Koethe<sup>5†</sup>, James G. Smith<sup>6†</sup>, Grant Passmore<sup>6†</sup>, Sebastian Post<sup>7†</sup>, Christoph Grimm<sup>7†</sup>. **Formal verification, validation and safety analysis of SysML v2 models: Towards a unified workflow.** Innovations in Systems and Software Engineering. Submitted, 2025.

- *precise mappings*

- *providing feedback*



# Formal Methods WG Charter

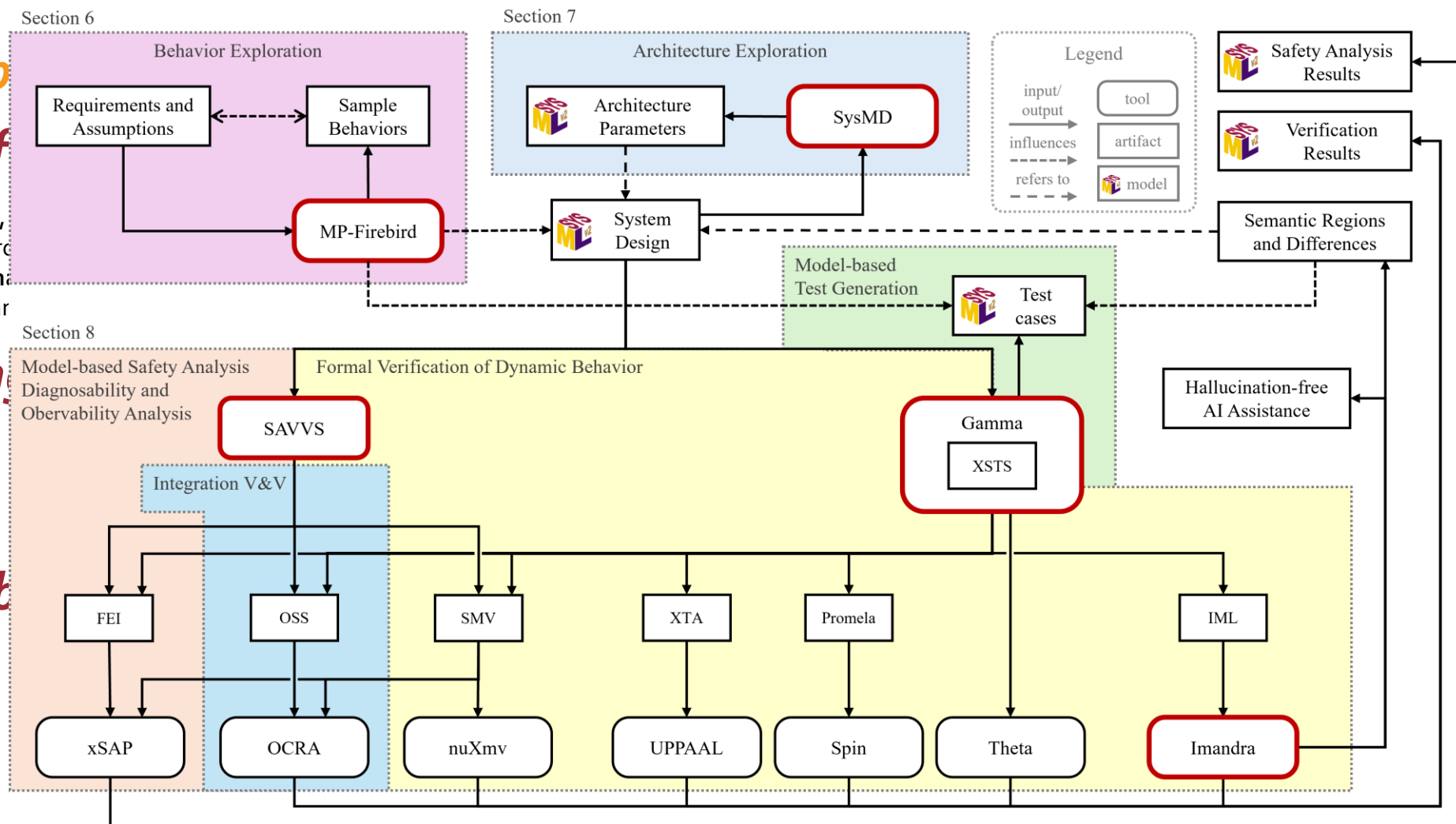
- *facilitate the app*

- *integration of f*

Vince Molnár<sup>1\*</sup>, Bence Graics<sup>1,6†</sup>,  
Pamela Dyer<sup>4†</sup>, Kristin Giammarco  
Post<sup>7†</sup>, Christoph Grimm<sup>7†</sup>. **Form**  
**Towards a unified workflow.** Inr

- *precise mappin*

- *providing feedb*



# Formal Methods WG Charter

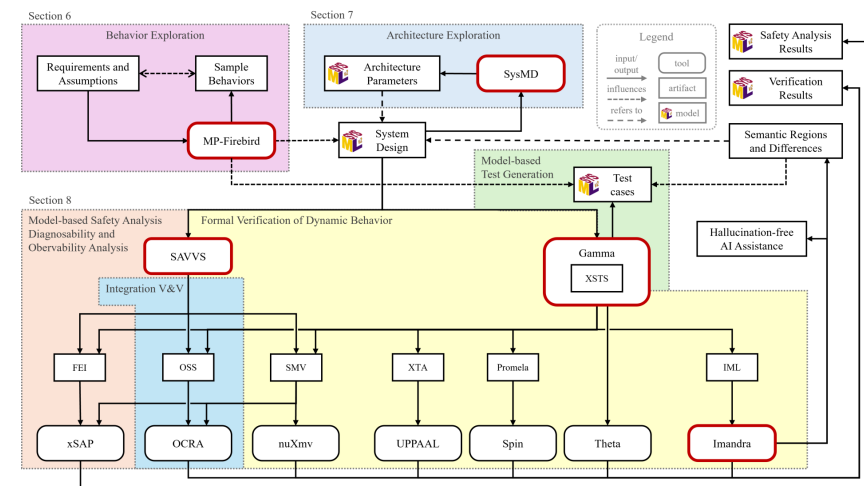
- *facilitate the application of formal methods*

- *integration of formal methods tools*

Vince Molnár<sup>1\*</sup>, Bence Graics<sup>1,6†</sup>, András Vörös<sup>1†</sup>, Stefano Tonetta<sup>2†</sup>, Luca Cristoforetti<sup>2†</sup>, Greg Kimberly<sup>3†</sup>, Pamela Dyer<sup>4†</sup>, Kristin Giammarco<sup>4†</sup>, Manfred Koethe<sup>5†</sup>, James G. Smith<sup>6†</sup>, Grant Passmore<sup>6†</sup>, Sebastian Post<sup>7†</sup>, Christoph Grimm<sup>7†</sup>. **Formal verification, validation and safety analysis of SysML v2 models: Towards a unified workflow.** Innovations in Systems and Software Engineering. Submitted, 2025.

- *precise mappings*

- *providing feedback*



# Formal Methods WG Charter

- *facilitate the application of formal methods*

- *integration of formal methods tools*

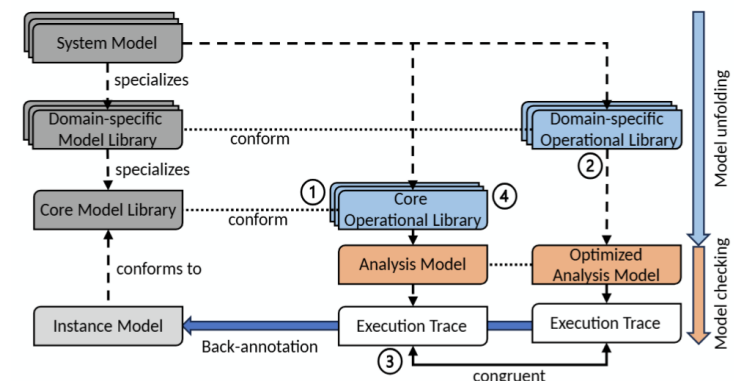
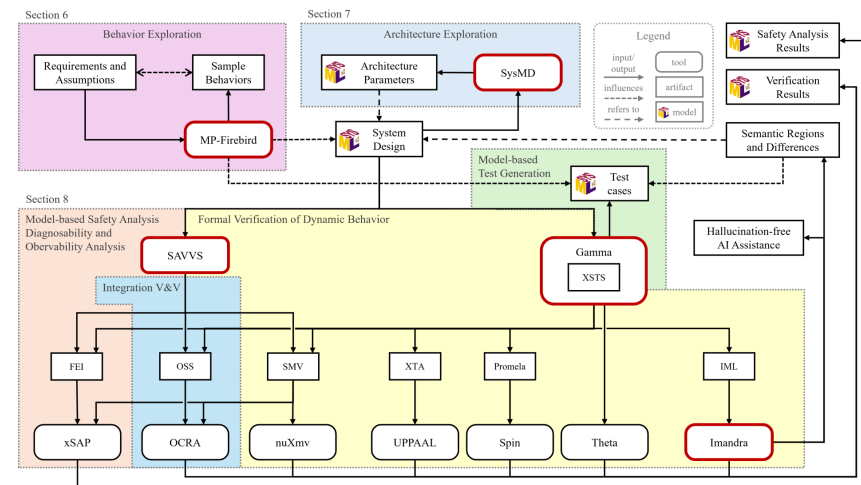
Vince Molnár<sup>1\*</sup>, Bence Graics<sup>1,6†</sup>, András Vörös<sup>1†</sup>, Stefano Tonetta<sup>2†</sup>, Luca Cristoforetti<sup>2†</sup>, Greg Kimberly<sup>3†</sup>, Pamela Dyer<sup>4†</sup>, Kristin Giammarco<sup>4†</sup>, Manfred Koethe<sup>5†</sup>, James G. Smith<sup>6†</sup>, Grant Passmore<sup>6†</sup>, Sebastian Post<sup>7†</sup>, Christoph Grimm<sup>7†</sup>. **Formal verification, validation and safety analysis of SysML v2 models: Towards a unified workflow.** Innovations in Systems and Software Engineering. Submitted, 2025.

- *precise mappings*

Ármin Zavada, Kristóf Marussy, and Vince Molnár. 2024. **From Transpilers to Semantic Libraries: Formal Verification With Pluggable Semantics.** In Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems (MODELS Companion '24). ACM, New York, NY, USA, 311–317

Brian Larson – Supplemental Formal Semantics for KerML

- *providing feedback*







# Wrap-up



# Summary

- SysML v2 is addressing SysML v1 limitations to improve MBSE adoption and effectiveness
  - New metamodel with both graphical and textual syntax and standardized API to access the model
  - More precise, expressive, usable, interoperable, and extensible than SysML v1
  - Consistent definition and usage pattern enables reuse, usability, and automation
- Progress/Plans
  - SysML v2 specifications submitted to OMG for final adoption in 2025
  - Continue to evolve SysML v2 specification, modeling practices, and domain specific extensions through the OMG Systems Modeling Community (SMC) in collaboration with INCOSE
- Organizations and practitioners should initiate their SysML v2 transition planning
  - Refer to SysML v2 Transition Wiki at [https://www.omgwiki.org/MBSE/doku.php?id=mbse:sysml\\_v2\\_transition](https://www.omgwiki.org/MBSE/doku.php?id=mbse:sysml_v2_transition)



**Thank You!!**