

What's new in the SysML V2 standard and demonstration of the SysON modeling software

Association Française d'Ingénierie Système | Afterwork Chapitre Rhône-Alpes (CRRRA)

June 18th 2024

Florian NOYRIT

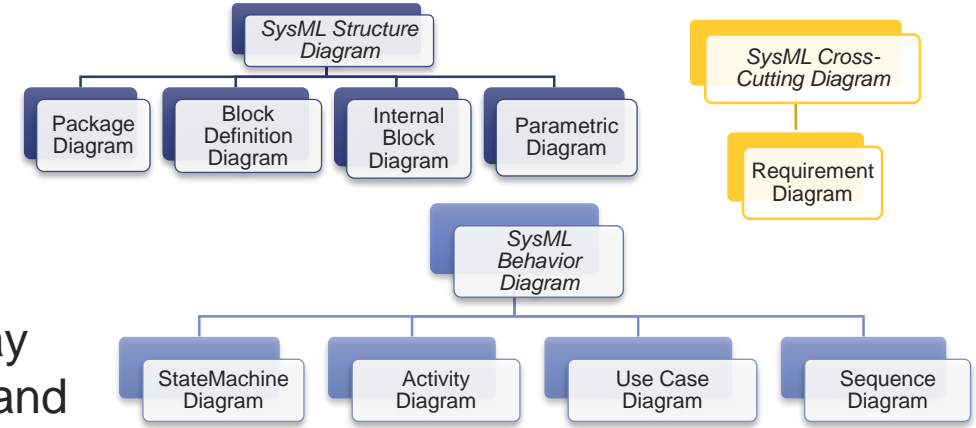
Jérémie TATIBOUET



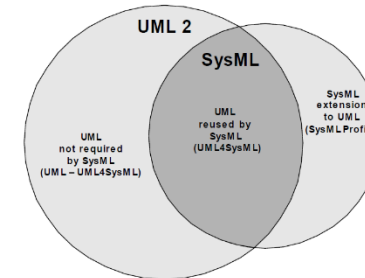
Context

SysML motivation and purpose:

”A general-purpose graphical modeling language for **specifying, analyzing, designing, and verifying complex systems** that may include hardware, software, information, personnel, procedures, and facilities.” (OMG)



Based on a subset of UML + UML profile :



SysML has evolved to address user and vendor needs:



SysML v2!

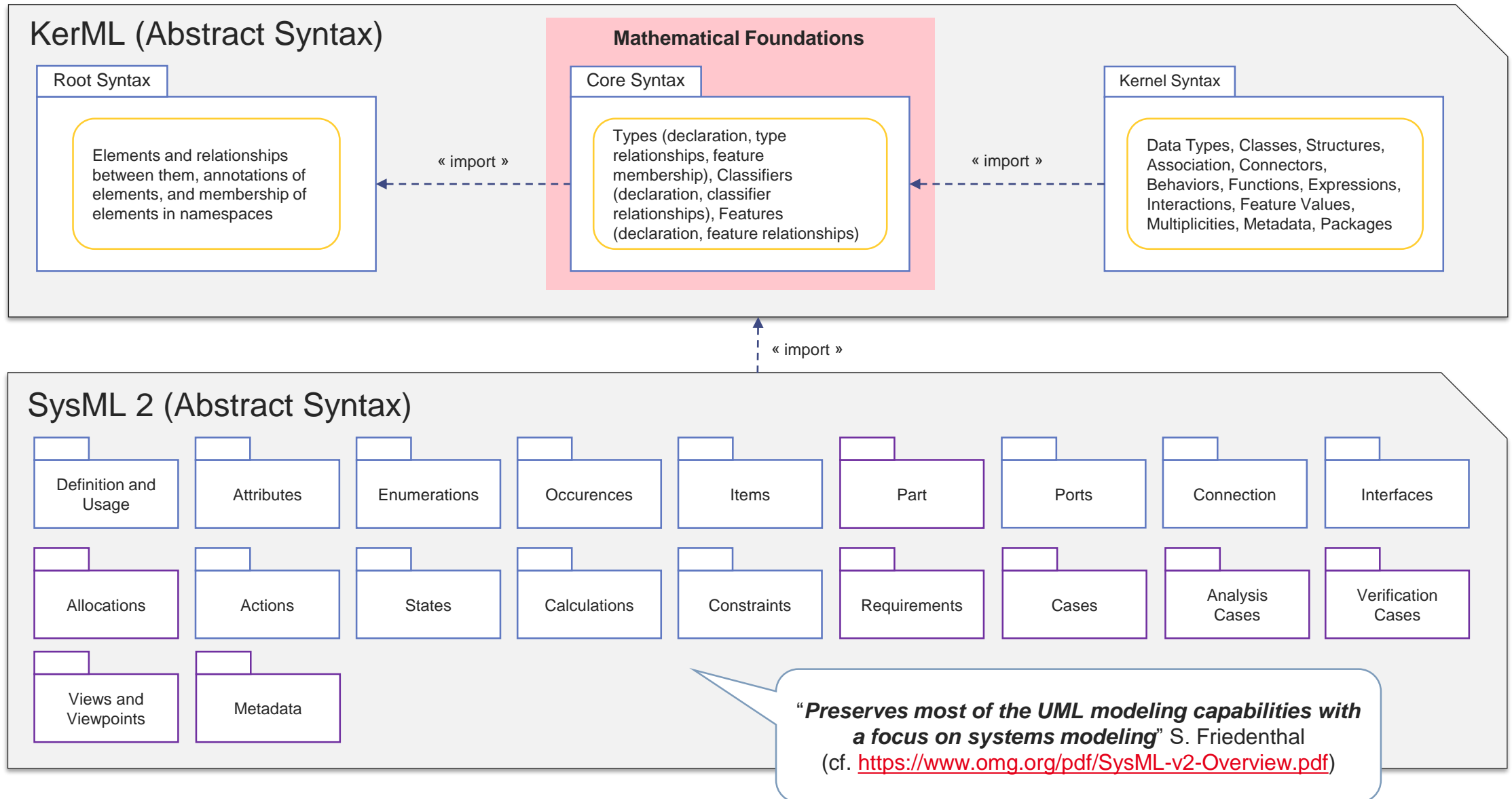
- New needs appeared:
 - INCOSE/OMG/AP233 requirements team.
 - must include **precise semantics**
 - must provide **flexible** and **rich visualization** and **reporting capabilities** to support a broad range of model users
 - must provide a **standard application programming interface**
 - ...
 - INCOSE Systems engineering VISION 2035
 - Full life cycle and from system of systems (SoS) to component level
 - “MBSE Descriptive models created using semantically rich modeling standards provide systems abstraction, data traceability, separation of views, and leverage AI/ML-based reference model reuse at both systems and product realization levels.”
 - ...
- Some issues with SysML v1
 - The UML profile mechanism has limits
 - UML is somewhat software oriented
- It is time for a new SysML version:



<https://www.incose.org/publications/se-vision-2035>



Language Architecture



Language Capabilities

▪ Capabilities

Requirements

- Define **required / assumed constraints** (on existing model properties)
- Define entity on which a requirement is specified: the subject
- Define actors (an external entity) and stakeholders (a person, an organization, a group)

Structure

- Define a system model structure with classifications, decompositions and interconnections

Behavior

- Define a system model behavior with actions, states and interactions

Analysis

- Define an analysis to **evaluate** the impacts of design choices on a system model according to various scenarios

Verification

- Define a verification to **check** the satisfaction of certain requirements on a system model

▪ Methodology

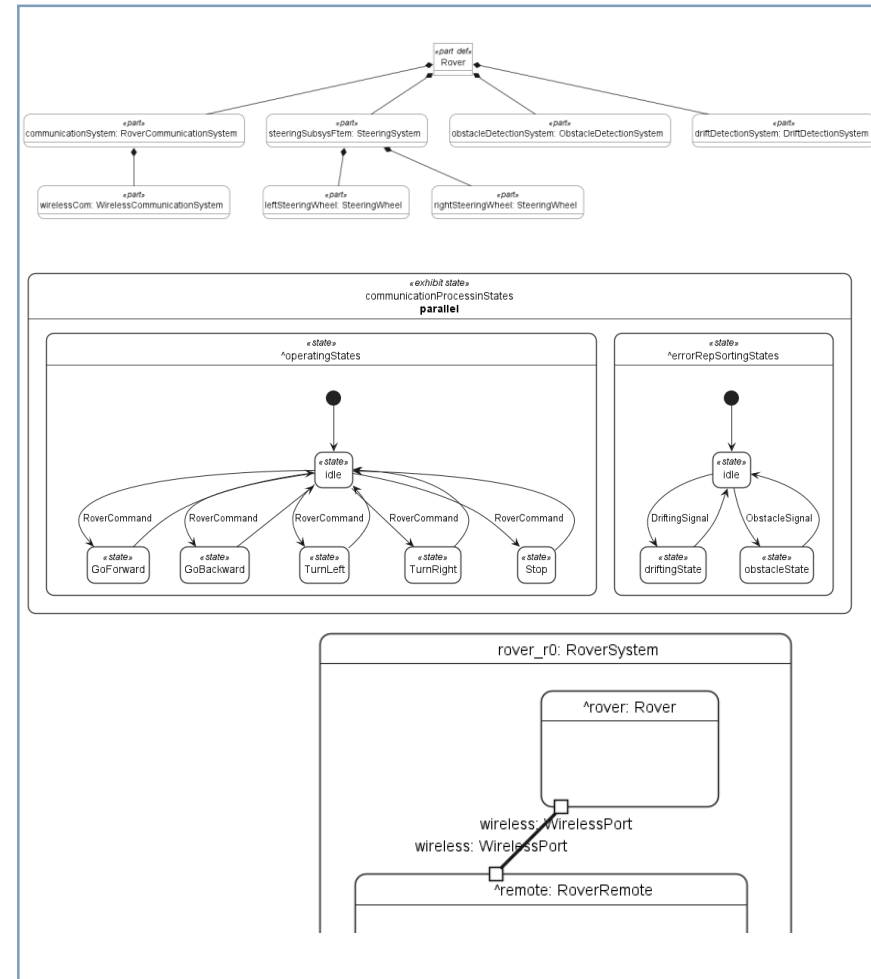
- “The language is intended to support **multiple system engineering methods and practices**. The specific methods and practices may impose additional constraints on how the language is used”. (cf., Systems Modeling Language Version 2.0 Beta 2 Part 1 – Clause 1)

Textual & Graphical language

Rich, expressive and human-readable textual syntax

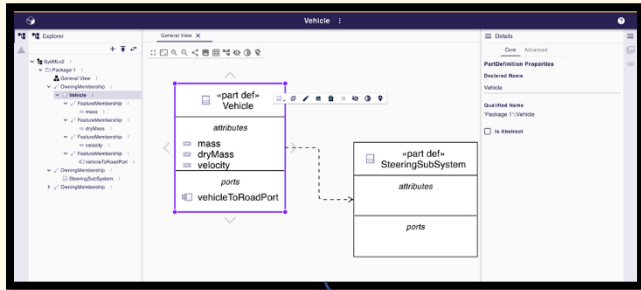
```
RoverExample > rover.sysml > Rover > System > CommunicationProcessing
1 package Rover {
2   import SI::M;
3   import SI::Q;
4   import SI::A;
5   import SI::V;
6   import SI::rad;
7   import SI::Hz;
8   import ISQ::PowerValue;
9   import ISQ::ResistanceValue;
10  import ISQ::ElectricCurrentValue;
11  import ISQ::ElectricPotentialValue;
12  import Quantities::scalarQuantities;
13  import ISQSpaceTime::AngularMeasureValue;
14  import MeasurementReferences::DerivedUnit;
15
16  package Requirements{
17    import System::RoverSystem;
18
19    requirement def <ReqDef6> PowerConsumptionRequirement {
20      doc
21      /*
22       * Assuming the max consumption greater than 0, the actual consumption shall be less than or equal to the max consumption.
23       * Assuming the min consumption greater than 0, the actual consumption shall be greater than or equal to the max consumption.
24       */
25      attribute actualPowerConsumption : PowerValue;
26      attribute minConsumption : PowerValue;
27      attribute maxConsumption : PowerValue;
28      assume constraint { maxConsumption > 0 [W] }
29      assume constraint { minConsumption > 0 [W] }
30      require constraint { actualPowerConsumption <= maxConsumption }
31      require constraint { actualPowerConsumption >= minConsumption }
32    }
33
34    requirement <Req6> roverSteeringPowerConsumptionReq : PowerConsumptionRequirement {
35      doc
36      /*
37       * Power consumption for the steering of the Rover must be 60W +/- 1%.
38       */
39    }
40  }
41 }
```

Multiple graphical views
(General, Interconnection, State Transition, etc.)

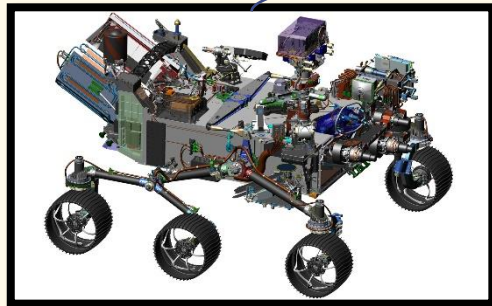
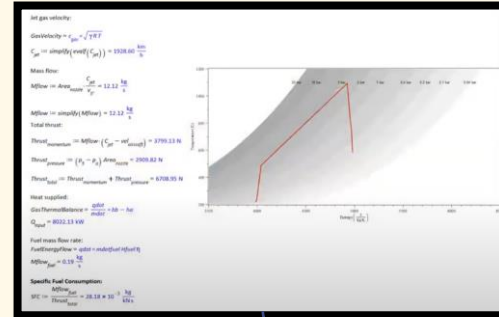


API and Services Interoperability

Authoring and visualization [2]



Analysis and solving [3]



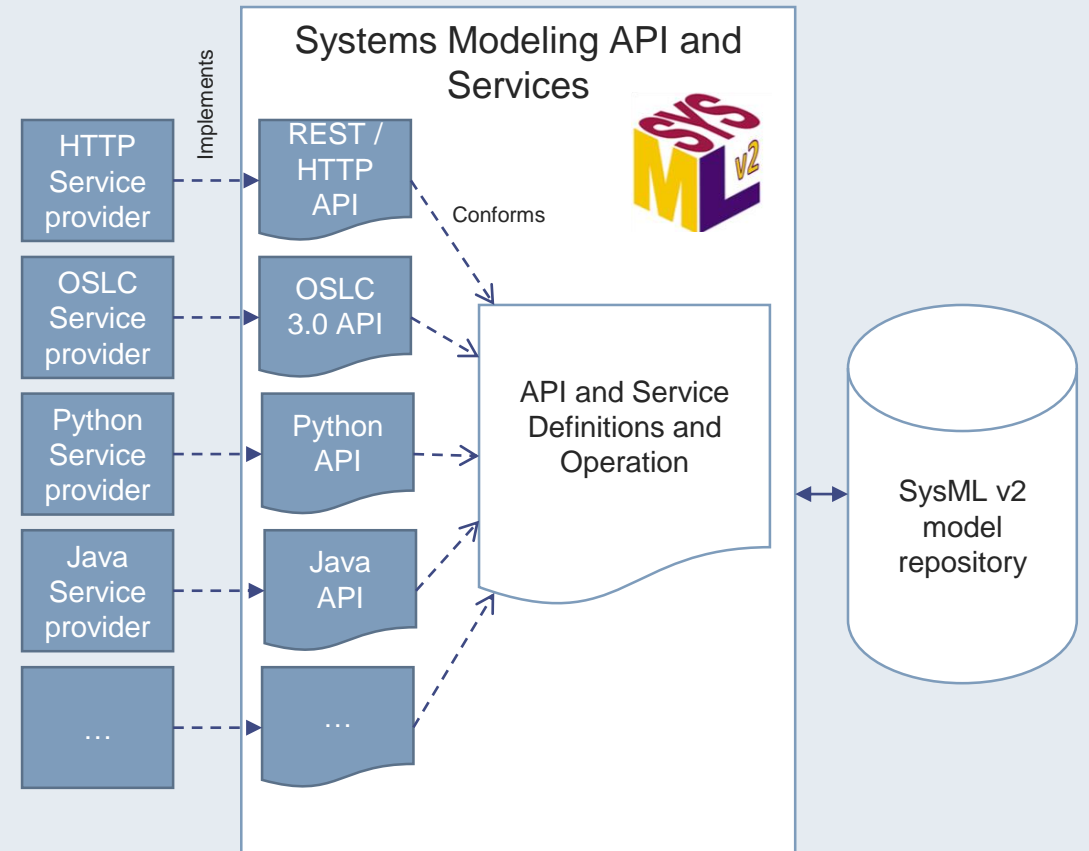
CAD [4]



Digital Thread management [5]

Service consumers

Systems Modeling API and Services



Services providing

[1] SysML v2 Basics, January 28, 2024, from Sanford Friedenthal

[2] <https://mbse-syson.org/>

[3] <https://www.maplesoft.com/products/Maple/demo/player/2023/Managing-Engineering-Calculations-as-Knowledge-Assets-with-the-Maple-Connector-for-SysML.aspx>

[4] <https://science.nasa.gov/resource/computer-design-drawing-for-nasas-2020-mars-rover/>

[5] [https://intercax.atlassian.net/wiki/spaces/SYN35/pages/2922152124/Syndeia+3.5+-+New+Features+and+Improvements#SysML-v2-\(OMG\)](https://intercax.atlassian.net/wiki/spaces/SYN35/pages/2922152124/Syndeia+3.5+-+New+Features+and+Improvements#SysML-v2-(OMG))

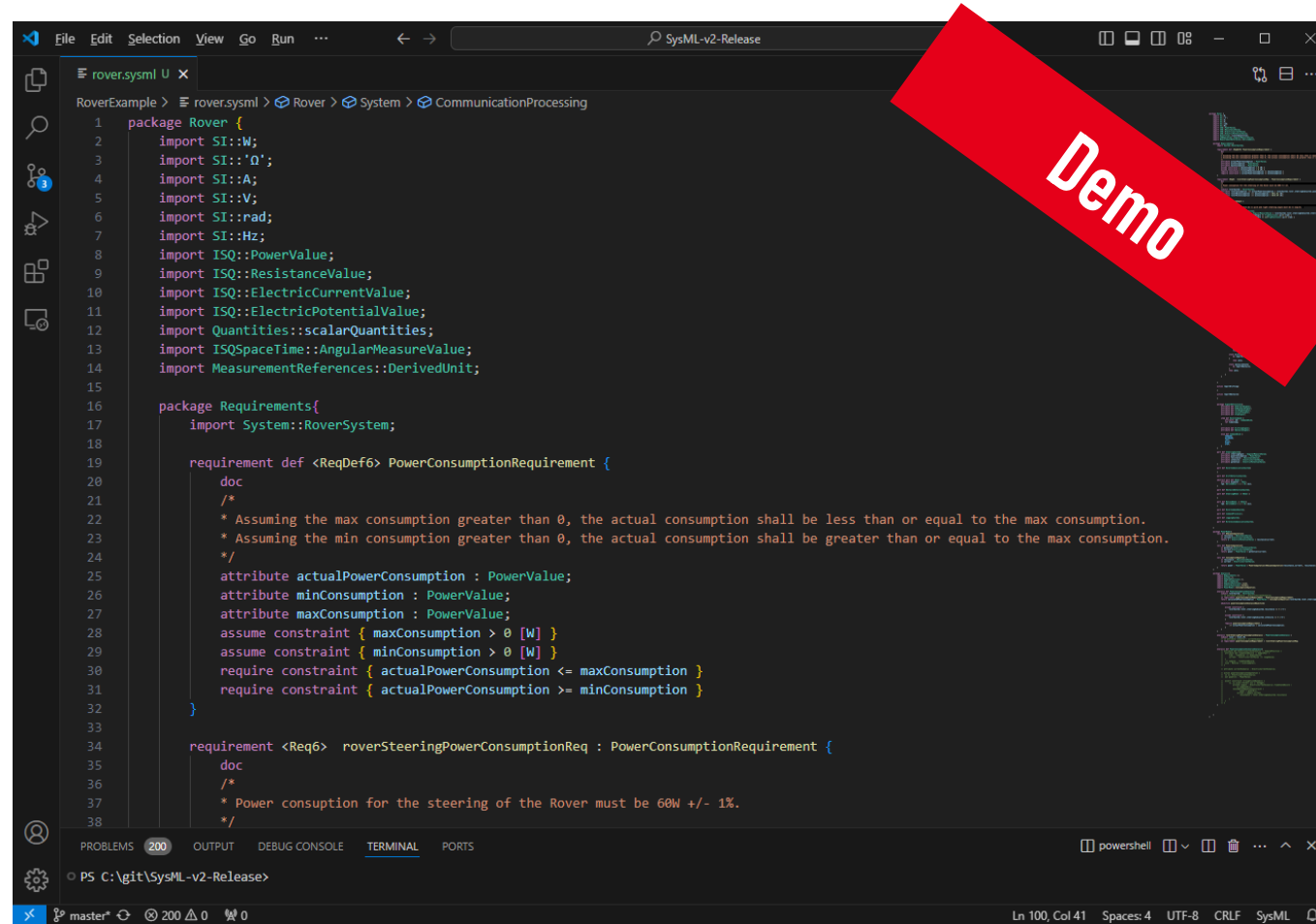
Some Implementations / Tooling

Name	Contributors	Standards Conformance	Ambition	Technologies	Expected Maturity	Availability	License
Pilot Implementation	OMG contributors and others	<ul style="list-style-type: none"> • KerML • SysML V2 • Notation : text. and graph. • API and Services 	Demonstrate the language capabilities and its implementability	<ul style="list-style-type: none"> • Eclipse • EMF • Xtext • PlantUML 	Demonstrator	https://github.com/Systems-Modeling/SysML-v2-Pilot-Implementation	LGPL 3.0
SysIDE	Sensmetry	<ul style="list-style-type: none"> • KerML • SysML V2 • Notation : text. 	“SysIDE (pronounced "seaside") provides SysML v2 language support in VS Code.”	<ul style="list-style-type: none"> • Typescript • Langium (LSP) • delivered as VSCode extension 	Industrial	https://github.com/sensmetry/sysml-2ls	EPL V2
SysON	Obeo, CEA List	<ul style="list-style-type: none"> • KerML • SysML V2 • API and Services • Notation : graph. 	Eclipse SysON project provides an open-source and interoperable tool for editing SysMLv2 models conforming to the OMG Standard for the MBSE community	<ul style="list-style-type: none"> • Sirius Web (Typescript, React-Flow, Spring boot, GraphQL, EMF) 	Industrial	https://github.com/eclipse-syson/syson	EPL V2

Others
(providers, consumers)

PTC (Windchill Modeler), Dassault (MagicDraw), Sparx (EA), IntercaX (Syndeia), Maple...

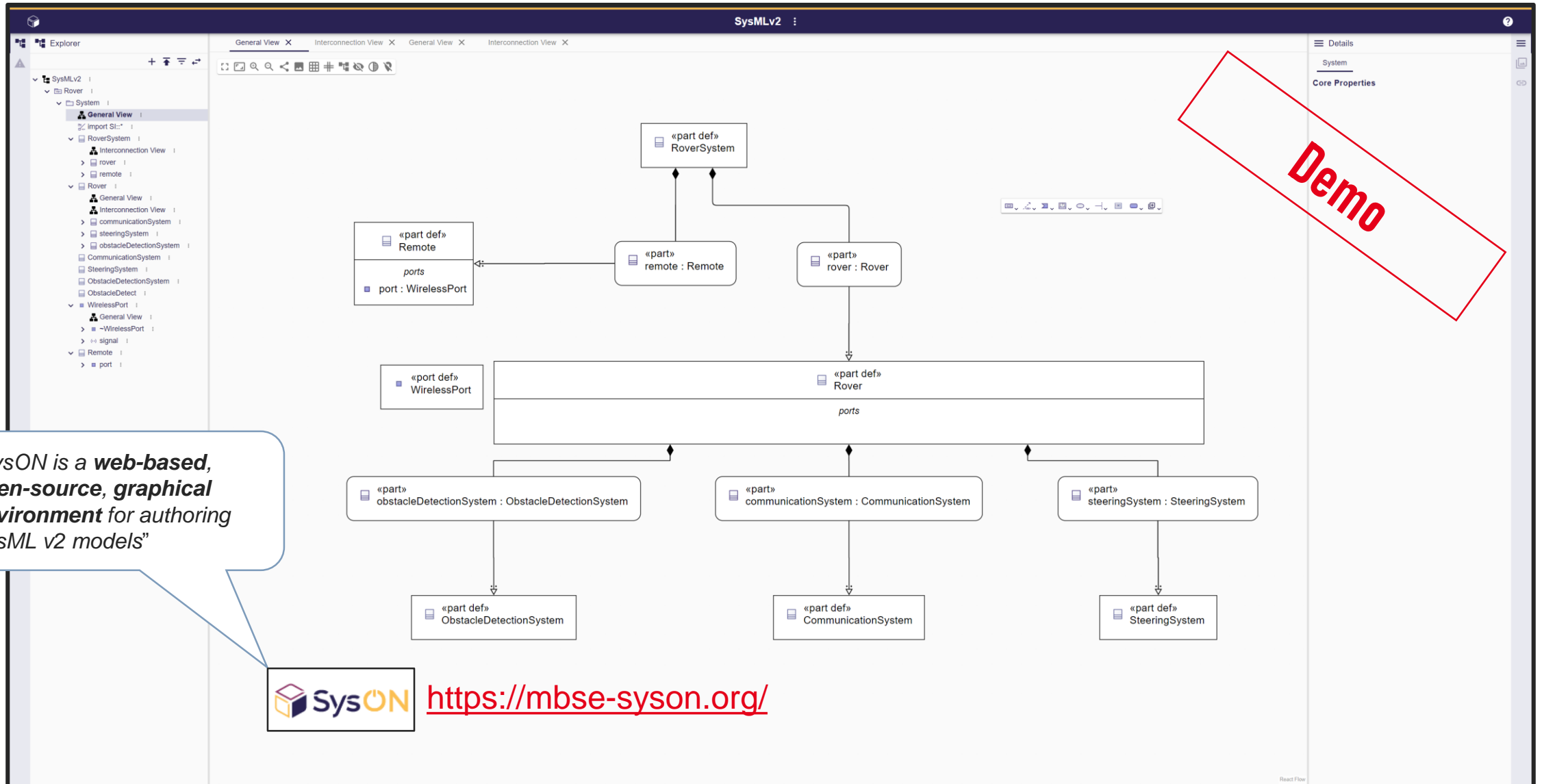
SysIDE Tooling Demonstration



The screenshot displays the SysIDE interface with a SysML model open. The model is titled 'rover.sysml' and is located within the 'RoverExample' project. The model structure is as follows:

- package Rover {**
 - import SI::W;
 - import SI::N;
 - import SI::A;
 - import SI::V;
 - import SI::rad;
 - import SI::Hz;
 - import ISQ::PowerValue;
 - import ISQ::ResistanceValue;
 - import ISQ::ElectricCurrentValue;
 - import ISQ::ElectricPotentialValue;
 - import Quantities::scalarQuantities;
 - import ISQSpaceTime::AngularMeasureValue;
 - import MeasurementReferences::DerivedUnit;
- package Requirements{**
 - import System::RoverSystem;
 - requirement def <ReqDef6> PowerConsumptionRequirement {**
 - doc
 - /*
 - * Assuming the max consumption greater than 0, the actual consumption shall be less than or equal to the max consumption.
 - * Assuming the min consumption greater than 0, the actual consumption shall be greater than or equal to the max consumption.
 - */
 - attribute actualPowerConsumption : PowerValue;
 - attribute minConsumption : PowerValue;
 - attribute maxConsumption : PowerValue;
 - assume constraint { maxConsumption > 0 [W] }
 - assume constraint { minConsumption > 0 [W] }
 - require constraint { actualPowerConsumption <= maxConsumption }
 - require constraint { actualPowerConsumption >= minConsumption }
 - requirement <Req6> roverSteeringPowerConsumptionReq : PowerConsumptionRequirement {**
 - doc
 - /*
 - * Power consumption for the steering of the Rover must be 60W +/- 1%.
 - */

SysON Tooling Demonstration



“SysON is a **web-based, open-source, graphical environment** for authoring SysML v2 models”



<https://mbse-syson.org/>

SysON (Roadmap)



The tool and the standard (i.e. SysML v2) are still ongoing developments.

- **2023**
 - Initial release in December
- **2024**
 - Releases in January, March and May (2024.5.2)
 - See <https://github.com/eclipse-syson/syson/packages/2020337/versions>
- **Current status**
 - Full implementation of SysML v2 and KerML metamodels
 - Graphical notation : general and interconnection views
 - Textual notation
 - Limited to a subset in direct edit capabilities proposed in diagrams
 - Import of models specified with the textual notation
 - Demo available at <https://www.youtube.com/live/GcRFW0YfFEI?si=Gch7wi5tPaYwzyy&t=2100>

- **Future works**
 - Graphical notation – support behavioral specifications
 - Action Flow View
 - State Transition View
 - Textual notation support
 - System Modeling API and Services
 - REST API implementation

SysML v1.x to v2.0 Legacy management

- **Normative formal (declarative) transformation** published along with KerML and SysMLv2 specifications

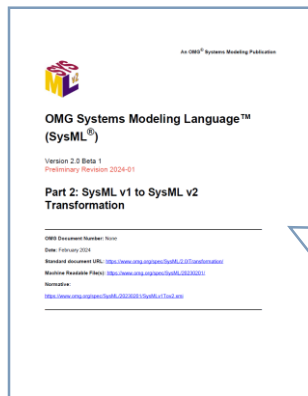
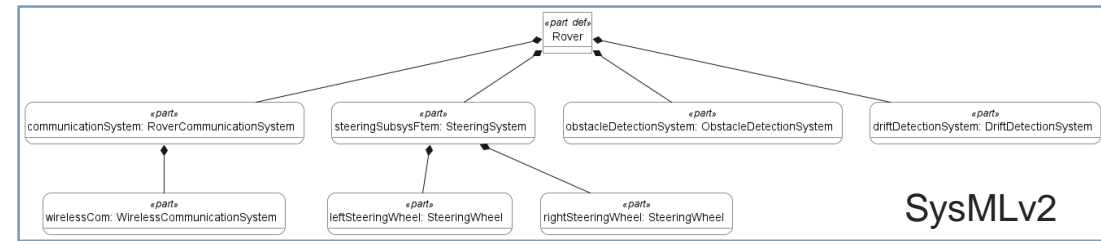
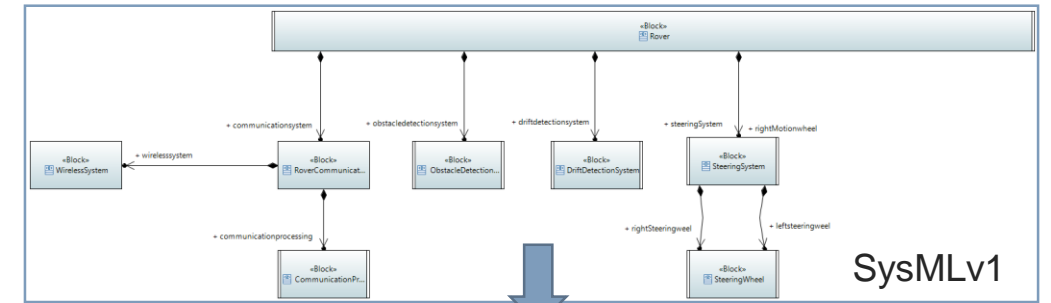
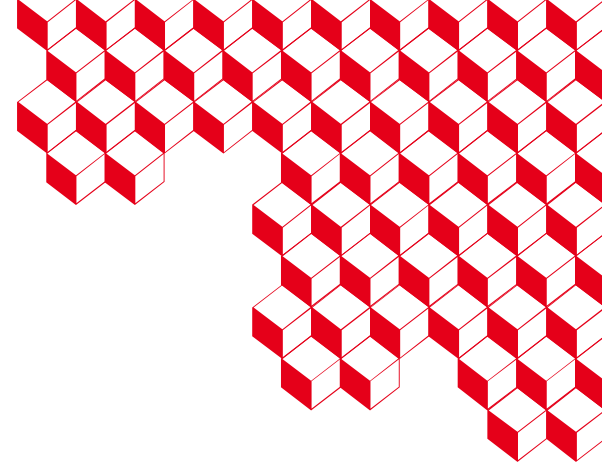


Table 26. List of all mappings

SysML v1 Abstract Syntax/Stereotype	SysML v2 Abstract Syntax
AdjunctProperty	
BindingConnector	BindingConnectorAsUsage
Block	PartDefinition PartDefinition
BoundReference	
ClassifierBehaviorProperty	
ConnectorProperty	
DistributedProperty	
EndPathMultiplicity	
NestedConnectorEnd	
ParticipantProperty	
PropertySpecificType	
ValueType	AttributeDefinition



- Not every construct from UML4SysML and SysML v1 stereotypes match a SysMLv2 construct.
- Generic transformation → **Manual and custom refactoring** may be required on the transformed model
- SysML v1 to SysML v2 Transition Guide Project [1]
- OMG dedicated Working Group



Thank you for your attention

CEA LIST
DRT/LIST/DILS/LIDEO

florian.noyrit@cea.fr

jeremie.tatibouet@cea.fr