

Abstract for MBSE2021 Workshop

**Title: “SysML Version 2 Approaching Industrial Use”**

Author(s): Hans Peter de Koning ([hanspeter.dekoning@dekonsult.com](mailto:hanspeter.dekoning@dekonsult.com))

Affiliation: DEKonsult, Amsterdam, The Netherlands

LinkedIn profile: <https://www.linkedin.com/in/hans-peter-de-koning>

---

This presentation is proposed as a follow-up to “*What to Expect from SysML Version 2?*” [5], presented at the MBSE2020 workshop. As reported then, the *initial submission* of the major new version 2 of the Systems Modeling Language (SysML) standard to the Object Management Group (OMG) had just been done in August 2020. One year has passed since, and the international SysML v2 Submission Team (SST) has worked hard to complete the standard taking into account a number of reviews and evaluations that were held inside and outside OMG.

This will lead to – and by the time of the MBSE2021 workshop will have led to – the submission of the so-called *revised submission* of the complete SysML v2 standard in August 2021. As in 2020, the standard consists of three specifications:

1. Kernel Modeling Language (KerML)
2. OMG Systems Modeling Language (SysML)
3. Systems Modeling API and Services

In addition, the SST has developed full pilot implementations to validate the specifications while they were developed, with many example models and working software. Most of this software is provided in open source under the LGPL license, see [3] and [4].

In the ongoing adoption of Model-Based Systems Engineering (MBSE) the Systems Modeling Language (SysML) standard from the Object Management Group (OMG) plays a major role, as it is de facto the only implemented international standard in direct support of MBSE. Since SysML version 1.0 was released in 2008 and has increasingly been deployed in industry and government agencies across many industry sectors to support the development of complex systems. In addition, in 2017 SysML was adopted as the ISO/IEC 19514 standard. Since approximately 2012, SysML is used on an increasing number of space projects in the European and US space sectors, by industry as well as agencies.

As reported in 2020, the development of SysML v2 started in 2015 with extensive user needs collection and the formulation of a comprehensive Request for Proposal (RFP), the OMG equivalent of a requirements specification for a new standard or major upgrade. There are actually two RFPs: one for the language and one for the API and Services. Subsequently from 2018 to date, the SST has been developing SysML v2, both the Language and the API and Services parts as listed above. The SST consists of more than 180 experts from more than 80 organisations, that cover a wide range of industrial, academic, commercial, and institutional interests in systems engineering of complex systems. Both end-users and developers from many industry sectors are well represented. The RFPs contain a large set of requirements that drive the development.

Major points to be addressed are:

- lowering of the SysML learning curve;
- unification on how structure and behaviour are decomposed;
- full and simple access to (deeply) nested structures and behaviours;
- addition of a full textual notation, in addition to clean-up and improvements of the abstract language and graphical notation;

- integration of a powerful expression language;
- development of semantically rich normative model libraries;
- much better support for multiple viewpoints and views for the same model information.

The current presentation will provide an overview of the new and enhanced capabilities of SysML version 2, including but not limited to:

1. New simplified SysML meta-model, which is founded on a minimal set of key concepts.
2. The normative libraries, that provide all SysML basic concepts directly at the user modelling level (M1 for the experts) and thereby constitute a powerful and straightforward extension mechanism that replaces the profile and stereotypes approach of SysML v1 and UML.
3. The new textual notation, including a very powerful expression and constraint language, as well as the upgraded graphical notation, and the integral, flexible viewpoint / view capabilities. These overcome many limitations of the SysML v1 Block Definition and Internal Block Diagrams. They also provide much better and more precise ways to define interfaces and connections, including in deeply nested structures. Then there is the integrated approach to model behaviour (activities, functional architecture, time-based sequences, finite state machines, 4D lifecycle objects), both in precise textual notation and in diagrams that can be mixed and matched, which was requested by many users.
4. Much improved support for requirements engineering and use case definition, leveraging the parameterised constraint definition capabilities in SysML v2 and unification of concepts.
5. Support for variant modelling and product line engineering built into the language and/or connectable to external variant modelling tools.
6. The built-in meta-data and user-defined keywords functionality.
7. The extensive library in support of quantities, units, scales and dimensions, that covers all of ISO/IEC 80000 (International System of Quantities and SI) and US Customary Units. This library now also fully supports vector and tensor quantities as well as multi-dimensional coordinate systems and coordinate transformations.
8. The upcoming library to support basic geometry modelling, e.g., to allow definition of basic enveloping shapes and coordinate systems/transformations for the physical parts of an architecture.
9. The much-improved support for integrating SysML v2 models with external analysis and simulation paradigms and tools, founded on much more precise execution semantics.
10. The prototype implementations of the textual and graphical language, on the Eclipse Modeling Framework as well as in JupyterLab Notebooks, see [4].
11. The new API and Services (spec and implementation) that provide a much better and richer capability to interact with SysML models than SysML v1 XMI files. The technology neutral API specification allows for both static whole model transfers and simultaneous dynamic interaction of many client tools with one or more SysML repositories. In the current pilot implementations, REST and OSLC APIs are supported, as well as Java and Python client-side class libraries.
12. The way compatibility with SysML v1 is ensured via a standardised transformation specification from SysML v1 to SysML v2.

If there is sufficient time, it is possible to demonstrate selected SysML v2 modelling capabilities using the pilot implementation tools.

Finally, an outlook on the expected further standardisation and industrial deployment of SysML v2 will be provided.

The author has been an active member of the SysML v1 task forces since 2009 as well as the SysML v2 RFP working group and is currently a core member of the SysML v2 Submission Team at OMG.

## References

[1] Systems Modeling Language, OMG, <https://www.omg.org/spec/SysML/>, last version 1.6, November 2019

[2] Systems Modeling Language (SysML®) v2 Request For Proposal (RFP), OMG, December 2017, <https://www.omg.org/cgi-bin/doc.cgi?ad/2017-12-2>

[3] Systems Modeling Language (SysML®) v2 monthly public releases at:  
<https://github.com/Systems-Modeling/SysML-v2-Release>

Note: this comprises the specifications as well as training material and pilot implementation software releases.

[4] Systems Modeling Language (SysML®) v2 open source pilot implementations:

<https://github.com/Systems-Modeling/SysML-v2-Pilot-Implementation>

<https://github.com/Systems-Modeling/SysML-v2-API-Services>

<https://github.com/Systems-Modeling/SysML-v2-API-Java-Client>

<https://github.com/Systems-Modeling/SysML-v2-API-Python-Client>

[5] Hans Peter de Koning, “What to Expect from SysML v2”, 2020, presented at MBSE2020, see [https://indico.esa.int/event/329/contributions/5528/attachments/3886/5614/1740\\_-\\_Presentation\\_-\\_What\\_to\\_Expect\\_from\\_SysML\\_Version\\_2.pdf](https://indico.esa.int/event/329/contributions/5528/attachments/3886/5614/1740_-_Presentation_-_What_to_Expect_from_SysML_Version_2.pdf)

[6] General information on the OMG Systems Modeling Language (SysML), see <http://www.omgsysml.org>

[7] General information on MBSE across all industry sectors, INCOSE/OMG MBSE Wiki at <http://www.omgwiki.org/MBSE/doku.php>