Translator from Extended SysML to Physical Interaction and Signal Flow Simulation Platforms, Version 1.1

Raphael Barbau[1,2], Conrad Bock[3], and Mehdi Dadfarnia[3]

[1]Engisis LLC,
Bethesda, MD 20817, USA

[2]Associate, National Institute of Standards and Technology,
Gaithersburg, MD 20899

[3]National Institute of Standards and Technology,
Gaithersburg, MD 20899

raphael.barbau@nist.gov
conrad.bock@nist.gov
mehdi.dadfarnia@nist.gov

1. Summary

The design of complex systems often requires engineers from multiple disciplines (mechanical, electrical, production, and so on) to communicate with each other and exchange system design information. Systems engineering models are a cross-disciplinary foundation for this process, but are not well-integrated with specialized engineering information, leading to redundant and inconsistent system specifications. The software provided here translates system models in the Systems Modeling Language (SysML) [1] to physical interaction and signal flow (also known as lumped-parameter, one-dimensional, or network) models on two simulation platforms used in many engineering domains.

The translator implements the SysML Extension for Physical Interaction and Signal Flow Simulation (SysPhS) version 1.1, published by the Object Management Group [2]. It can generate Modelica[1] or Simulink/Simscape files from SysML models extended according to the SysPhS specification. This translator is an updated version of the SysPhS 1.0 translator [3].

Several example models are provided to demonstrate the translator in various engineering domains.

---

[1]Certain commercial equipment, instruments, materials, or software are identified in this paper to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

2.   Software Specifications

NIST Operating Unit	Engineering Laboratory, Systems Integration Division
Category	System model translator
Targeted Users	Systems engineers, simulation engineers
Operating Systems	See systems requirements of Java 8 or later
Programming Language	Java 8
Inputs/Outputs	Input: SysML models created using the SysPhS profile Output: Modelica or Simulink file corresponding to the model
Documentation	Instructions on how to run the translator are provided in the software package, see README.txt file. The translator is an implementation of the SysPhS specification, available at https://www.omg.org/spec/SysPhS/1.1. The source code of the translator is available at https://doi.org/10.18434/mds2-2414
Accessibility	N/A
Disclaimer	https://www.nist.gov/director/licensing

3.   Software details

The software translates SysML models extended with SysPhS into simulation files for Simulink/Simscape and Modelica, per the SysPhS specification. The SysML models must be serialized in XML Metadata Interchange file format (XMI) [4]. The software deserializes these files into instances of the SysML metamodel using the Eclipse Modeling Framework (EMF) library and its extension for the Unified Modeling Language (UML) [5], on which SysML is based. EMF was used to create metamodels of the Simulink/Simscape and Modelica languages. Java libraries were automatically generated from these metamodels. These libraries are then used to instantiate the metamodels based on the input SysML models. The result is then serialized into simulation files that can be opened by their simulation platforms. Additional explanation is available in [6, 7].

The translator also supports the reverse translation, from Simulink/Simscape and Modelica files generated by the translator to extended SysML.

4.   Methods for Validation

A set of SysML example models are included to demonstrate the translator. These sample models cover multiple aspects of SysPhS:

- An electrical circuit model demonstrates how to use physical interaction modeling in the electrical domain.

- A hydraulics model demonstrates how to use physical interaction modeling in the hydraulic domain.

- A signal processor model (with low-pass and high-pass filter) demonstrates how to use signal-flow modeling.

- A humidifier system model demonstrates how to use signal-flow modeling to describe the operation of a humidifier in a room. The model also includes simple state machines.

- A cruise controller model demonstrates mixed usage of physical interaction and signal-flow. The signal flow portion covers speed sensors and engine actuators, while the physical interaction portion covers the flow of mechanical energy that powers the car and makes it move.

The sample models also include another version of the cruise controller model, which is used in a publication on debugging physical interaction and signal for models [8].

All these models were translated into Modelica and Simulink/Simscape files that simulate successfully, showing how the systems will behave over time.

Acknowledgments

5. References

[1] Object Management Group (2019) Systems Modeling Language, version 1.6. Available at
https://www.omg.org/spec/SysML/1.6/About-SysML/
[2] Object Management Group (2021) SysML Extension for Physical Interaction and Signal Flow Simulation, version 1.1. Available at
https://www.omg.org/spec/SysPhS/1.1/About-SysPhS/
[3] Barbau R, Bock C, Dadfarnia M (2019) Translator from Extended SysML to Physical Interaction and Signal Flow Simulation Platforms. *Journal of Research of the National Institute of Standards and Technology* 124:124017.
https://doi.org/10.6028/jres.124.017
[4] Object Management Group (2015) XML Metadata Interchange, version 2.5.1. Available at
https://www.omg.org/spec/XMI/2.5.1/About-XMI/
[5] Object Management Group (2017) Unified Modeling Language, version 2.5.1. Available at
https://www.omg.org/spec/UML/2.5.1/About-UML/
[6] Bock C, Barbau R, Matei I, Dadfarnia M (2018) An Extension of the Systems Modeling Language for Physical Interaction and Signal Flow Simulation. *Systems Engineering* 20(5):395–431. https://doi.org/10.1002/sys.21380
[7] Dadfarnia M, Bock C, Barbau R (2016) An Improved Method of Physical Interaction and Signal Flow Modeling for Systems Engineering, Conference on Systems Engineering Research. Available at
https://www.nist.gov/publications/improved-method-physical-interaction-and-signal-flow-modeling-systems-engineering
[8] Dadfarnia M, Barbau R (2019) Platform-Independent Debugging of Physical Interaction and Signal Flow Models. *2019 IEEE International Systems Conference (SysCon)*, pp 1–8. https://doi.org/10.1109/SYSCON.2019.8836890

About the authors: *Raphael Barbau is a senior consultant at Engisis LLC, and a Computer Scientist in the Systems Integration Division at NIST. He is involved in various projects for improving interoperability among systems engineering tools.*

Conrad Bock is a Computer Scientist in the same division at NIST. He leads systems analysis integration in the Model-Based Enterprise Program, and chairs the SysML Extension for Physical Interaction and Signal Flow Simulation Revision Task Force at the Object Management Group.

Mehdi Dadfarnia is an Electrical Engineer, also in the same division at NIST. He has a PhD in Mechanical Engineering at the University of Maryland, College Park, as well as an M.S. in Systems Engineering and B.S. in Electrical Engineering from the same institution. He worked on interoperability of systems models with simulation tools.

The National Institute of Standards and Technology is an agency of the U.S. Department of Commerce.