
SysML Models for Discrete Event Logistics Systems

Timothy Srock and Conrad Bock

National Institute of Standards and Technology,
Gaithersburg, MD 20899

timothy.srock@nist.gov
conrad.bock@nist.gov

Software DOI: <https://doi.org/10.18434/M32203>

Key words: Model-based Systems Engineering; Production Systems; Smart Manufacturing; SysML.

Accepted: July 22, 2020

Published: July 29, 2020

<https://doi.org/10.6028/jres.125.023>

1. Summary

System models and model-based engineering methods have the promise of transforming the way that industrial engineers interact with production and logistics systems. Model-based methods play a role in improving communication between stakeholders, interoperability between systems, automated access to consistent analysis models, and multi-disciplinary design methods for complex systems. However, there remains a need for a foundation for modeling these kinds of systems – a foundation that tailors methods and tools developed in other engineering domains to the unique concepts and semantics of production and logistics. This foundation is the topic of these models.

This repository contains model libraries for modeling discrete event logistics systems (DELS), an abstraction that covers manufacturing plants, material handling and transportation systems, warehouses, supply chains, etc. The DELS abstraction was created by identifying and modeling commonalities across the kinds of systems that industrial engineers typically encounter, and analysis models they use to analyze those systems. It extends well-known product, process, and resource (PPR) ontologies to incorporate a library of operational control model components, and is connected to Commodity Flow Network (CFN), modeling networks, flow networks, and process networks. The relationship between DELS and CFN formally links system models to abstractions used to create analysis models, such as discrete event simulation.

This repository is the first public release of models and documentation capturing many years of refinement and application by the authors. As a first release, the goal is to solicit additional use cases and feedback from the community to improve the models and make them the foundation for the model-based industrial and systems engineering community.

The models are implemented in the Systems Modeling Language (SysML) [1] developed using MagicDraw¹ 18.5 by No Magic. The models are documented in NISTIR 8262 [2].

¹Certain commercial software is identified in this paper to specify the models 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 software identified is necessarily the best available for the purpose.

2. Software Specifications

NIST Operating Unit	Engineering Laboratory, Systems Integration Division, Systems Engineering Group
Category	Systems engineering model
Targeted Users	Industrial and Systems Engineering Practitioners and Researchers.
Operating Systems	N/A
Programming Language	This software data item is created using SysML 1.4 and stored as Extensible Markup Language (XML) Model Interchange (XMI) models and Hypertext Markup Language (HTML) reports. MagicDraw 18.5 or later with SysML plugin is required to edit the model (also provided in proprietary mdzip format). The HTML documentation is provided for readers without the MagicDraw software.
Inputs/Outputs	N/A
Documentation	NISTIR 8262: https://doi.org/10.6028/NIST.IR.8262 Source models: https://github.com/usnistgov/DiscreteEventLogisticsSystems
Accessibility	N/A
Disclaimer	https://www.nist.gov/director/licensing

3. Methods

3.1 Implementation Tool

The SysML model was implemented using a proprietary graphical modeling tool called MagicDraw. MagicDraw supports the Unified Modeling Language (UML) [3] and SysML [1] among other graphic modeling languages. The SysML models can be exported in OMG XML Metadata Interchange (XMI) [4] file format and Hypertext Markup Language (HTML) reports.

3.2 SysML Diagram Types

The following SysML diagrams are used for communicating block relationships and architecture:

- Package Definition Diagram (PKG) – used to capture package containment and relationships;
- Block Definition Diagram (BDD) – used for capturing blocks, interfaces, and their relationships;
- Class Definition Diagram (CD) – used for capturing software classes, interfaces, and relationships;
- Activity Diagram (ACT) – used to capture behaviors, sequence of execution, and flows between executions;
- Internal Block Diagram (IBD) – used for capturing the internal composition of a single block.

3.3 Model Structure

- Commodity Flow Network
 - Networks

- Flow Networks
- Process Networks
- Discrete Event Logistics Systems
 - Context
 - Plant
 - * Product
 - * Process
 - * Resource
 - * Facility
 - Control

Acknowledgments

The authors acknowledge Leon F. McGinnis and George Thiers at the Georgia Institute of Technology for their contributions to earlier versions of the models.

4. References

- [1] OMG (2015) OMG Systems Modeling Language (OMG SysML) Version 1.4. Available at <http://www.omg.org/spec/SysML/1.4/>
- [2] Srock T, Thiers G, McGinnis LF, Bock C (2020) Theory of Discrete Event Logistics Systems (DELS) Specification (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency/Internal Report (NISTIR) 8262. <https://doi.org/10.6028/NIST.IR.8262>
- [3] OMG (2015) OMG Unified Modeling Language (OMG UML) Version 2.5. Available at <http://www.omg.org/spec/UML/2.5/>
- [4] OMG (2014) OMG XML Metadata Interchange (OMG XMI) Version 2.4.2. Available at <https://www.omg.org/spec/XMI/2.4.2/>

About the authors: *Timothy Srock is an Industrial Engineer in the Systems Engineering Group under the Systems Integration Division of the Engineering Lab at the National Institute of Standards and Technology. He is interested in developing and applying model-based engineering methods to production and logistics systems, in particular the design and operation of manufacturing operations management systems.*

Conrad Bock is a Computer Scientist in the Systems Engineering Group under the Systems Integration Division of the Engineering Lab at the National Institute of Standards and Technology specializing in formalization of engineering modeling languages. His previous experience includes business process modeling with SAP and Microsoft. He leads efforts on process modeling in UML and SysML at the Object Management Group.

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