An Overview of the SysML-Modelica Transformation Specification

Chris Paredis (Georgia Tech)

Y. Bernard (Airbus), R. Burkhart (Deere & Co), H. de Koning (ESA/ESTEC), S. Friedenthal (Lockheed Martin Corp.), P. Fritzson (Linköping University), N. Rouquette (JPL), W. Schamai (EADS)
Two complementary languages for Systems Engineering:
- Descriptive modeling in SysML
- Formal equation-based modeling for analyses and trade studies in Modelica

Objective:
- Leverage the strengths of both SysML and Modelica by integrating them to create a more expressive and formal MBSE language.
- Define a formal Transformation Specification:
  - a SysML4Modelica profile
  - a Modelica abstract syntax metamodel
  - a mapping between Modelica and the profile
Presentation Overview

- What is SysML?
- What is Modelica?
- Motivating Example: Design & Analysis of Robot
- SysML-Modelica Transformation Specification
- Transformations in Systems Modeling
- Model reuse and composition
- Summary
What is SysML?

- The Systems Modeling Language (OMG SysML™) is a visual, general purpose modeling language.

- **Is** a modeling language that provides
  - Semantics = meaning
  - Notation = representation of meaning

- **Is not** a methodology or a tool
  - SysML is methodology and tool independent

- Developed by the Object Management Group to support Model-Based Systems Engineering
What Can be Expressed in SysML?

- SysML is a language to express the information and knowledge generated and processed during the application of a systems development methodology.

- Specification
- Analysis
- Design
- Verification
- Validation

- Hardware
- Software
- Data
- Personnel
- Procedures
- Facilities
SysML Diagram Taxonomy

Think of SysML as an integrated collection of languages…

Some History...

  - v1.0: 2007-09
  - v1.1: 2008-11
  - v1.2: 2010-06
  - v2.x: RFI preparation workshop - 2008-12

- Strong vendor support
  - MagicDraw (No Magic), Artisan Studio (Atego), Enterprise Architect (Sparx Systems), Rhapsody (IBM),…

- Good learning infrastructure
  - Books, short courses, academic courses, INCOSE/OMG tutorial, public examples, etc.

- OMG Certified Systems Modeling Professional
  - http://www.omg.org/ocsmp/
What is SysML? (www.omgsysml.org)

1. Structure

```
<<block>>
Library: Electronic Processor
<<block>>
Library: Anti-Lock Controller
<<block>>
Traction Detector
<<block>>
Brake Modulator
```

2. Behavior

```
interaction state machine
activity/ function
```

```
stm TireTraction [State Diagram]
```

```
dd ABS_ActivationSequence [Sequence Diagram]
```

```
req [Package] Vehicle Specifications [ Braking Requirements ]
```

```
Vehicle System Specification
Stopping Distance
Id = "10.2"
Text = "The vehicle shall stop from 60 miles per hour within 150 ft on a clean dry surface"
```

```
Braking Subsystem Specification
Anti-Lock Performance
Id = "33.7"
Text = "The braking system shall prevent wheel lockup under all braking conditions."
```

3. Requirements

```
e1 : Braking Force Equation
\( f = (t + bf)^{(1.4t)} \)
```

```
e2 : Acceleration Equation
\( a : m/sec^2 \) \( f = m \cdot a \)
```

```
e3 : Velocity Equation
\( v = \frac{dx}{dt} \)
```

```
e4 : Distance Equation
\( x : m \)
```

4. Parametrics

```
What is SysML? (www.omgsysml.org)

1. Structure

- Vehicle Specifications
- Braking Subsystem Specification
- Vehicle System Specification

- Requirements Diagram - Stopping Distance
- Requirements Diagram - Anti-Lock Performance

2. Behavior

- Straight Line Vehicle Dynamics

3. Requirements

- Verify (via interaction)

4. Parametrics

- Distance Equation
- Velocity Equation
- Acceleration Equation
- Braking Force Equation

What is Modelica?
(www.modelica.org)

- State-of-the-art Modeling Language for System Dynamics
  - Differential Algebraic Equations (DAE)
  - Discrete Events
- Formal, object-oriented language
- Standardized by the Modelica Association
  - Open language specification – tool independent
- Multi-domain modeling
- Ports represent energy flow (undirected) or signal flow (directed)
- Acausal, equation-based, declarative (f-m*a=0)
Modelica: Active and Mature Community

- Modelica association — 20+ free libs (www.modelica.org)
- 6 commercial solvers, 3 open-source solvers (Dymola, MapleSim, SimulationX, OpenModelica, ...)
- EUROSYSLIB project — 20+ libs under development (http://www.itea2.org/public/project_leaflets/EUROSYSLIB_profile_oct-07.pdf)
A Robot Example in Modelica
Connections represent Kirchhoff semantics
- Across variables (voltage, pressure,…) are equal
- Through variables (current, flow rate,…) add to zero

Graphical symbols defined as annotations in textual models

```model Spring "Linear 1D translational spring"
extends Translational.Interfaces.PartialCompliant;
parameter SI.TranslationalSpringConstant c(final min=0, start = 1)
    "spring constant ";
  parameter SI.Distance s_rel0=0 "unstretched spring length";
equation
  f = c*(s_rel - s_rel0);
end Spring;```
Presentation Overview

- What is SysML?
- What is Modelica?

Motivating Example: Design & Analysis of Robot

- SysML-Modelica Transformation Specification
- Transformations in Systems Modeling
- Timeline towards Specification Adoption
- Summary
A Robot Example in Modelica
SysML-Modelica Robot Example: UseCases & Requirements

- Operator
- Maintenance
- Programmer

- Emergency Shutdown
- Execute Robot Task
- Program Trajectory

- Robot System
- Cost
- Performance
- Reliability
- Energy Consumption
- Weight
- Tracking Accuracy

- Workspace Safety Violation
- Workspace Safety Violation
SysML-Modelica Robot Example: Robot Domain BDD & IBD
SysML-Modelica Robot Example: Robot Arm BDD
Analysis models depend on descriptive models
SysML4Modelica Analytical Model: Compose Model from Standard Library

Drag and drop into IBD «ModelicaModel»
SysML4Modelica Analytical Model: Detailed IBD
SysML4Modelica Analytical Model: Detailed IBD
SysML4Modelica Analytical Model: Relation to Modelica Native Model
SysML-Modelica Robot Example: Modelica model with simulation results
SysML-Modelica Robot Example:
Analysis and Trade Study

Analysis results are incorporated in Trade Study
Presentation Overview

- What is SysML?
- What is Modelica?
- Motivating Example: Design & Analysis of Robot

SysML-Modelica Transformation Specification
- Transformations in Systems Modeling
- Timeline towards Specification Adoption
- Summary
SysML-Modelica Transformation Specification

SysML-Modelica Transformation follows the principles of Model-Driven Architecture (MDA)
SysML4Modelica Profile

pkg [Package] Classes [ Modelica Class Stereotypes ]

- isPartial is derived from isAbstract
- isFinal is derived from isLeaf

<<metaclass>>
Classifier

<<stereotype>>
ModelicaClassDefinition
+isFinal : Boolean [1] = false
+isPartial : Boolean [1] = false
+isModelicaEncapsulated : Boolean [1] = false
+isReplaceable : Boolean [1] = false
+fromLibrary : String [0..1]

<<stereotype>>
ModelicaModel
<<stereotype>>
ModelicaRecord
<<stereotype>>
ModelicaConnector
+isExpandable : Boolean [1] = false
<<stereotype>>
ModelicaType
+scope : ModelicaScopeKind [1] = none
+externalLibrary : String [0..*]
+externalInclude : String [0..1]

<<stereotype>>
ModelicaClass
<<stereotype>>
ModelicaBlock
<<stereotype>>
ModelicaOperator
<<stereotype>>
ModelicaPackage

<<stereotype>>
Block
<<stereotype>>
Enumeration
<<stereotype>>
DataType
<<stereotype>>
ValueType
model Spring  "Linear 1D translational spring"
    extends Translational.Interfaces.PartialCompliant;
    parameter SI.TranslationalSpringConstant c(final min=0, start = 1) "spring constant ";
    parameter SI.Distance s_rel0=0 "unstretched spring length";
    equation
        f = c*(s_rel - s_rel0);
end Spring;
Reference implementation: Based on OMG QVT

QVT = Query / View / Transformation
Transformations in Systems Modeling

- Model Object
  
  Model Dependency

System Model
Model Reuse in MBSE

Creating Models is Expensive and Time-Consuming

Reusable Models in MBSE

- Physical components are reused
- Portions of the systems model repeat
- Patterns for instantiating these portions

- Component models → Domain specific model libraries
- Application of pattern = model transformations

Electric Motors
Model Library of Hydraulic Components

- Needs to be carefully designed and managed
- Encodes domain knowledge

```
<<block>>
Hydraulic
<<block>>
Pump
Displacement
<<block>>
Valve
<<block>>
Tank
Volume
<<block>>
VariableDisplacementPump
PressureMargin
<<block>>
CheckValve
<<block>>
6Way3PosOpenCenter
<<block>>
6Way3PosOCParallel
```

```
<<block>>
HydraulicComponent
<<block>>
Pump
suction: FluidPort
discharge: FluidPort
input/Shaft: Shaft
housing: Mount

<<block>>
Fixed_Displacement_Pump
ratedPressure: Pressure [Nm²/2]
displacement: Displacement [cc/rev]
speedAtRatedPressure: AngularVelocity [rad/s]
cost: Currency [Dollar]
mass: Mass [kg]

<<block>>
VendorOTS_Pump
partNumber: string = AXD123
ratedPressure: Pressure [Nm²/2] = 1E5
displacement: Displacement [cc/rev] = 100
speedAtRatedPressure: AngularVelocity [rad/s] = 210
cost: Currency [Dollar] = 350
mass: Mass [kg] = 10
...```
Other Perspectives of Cylinder are Reusable

- When cylinder is used, other corresponding models are often used also

→ Capture the reuse pattern
Model-Based Systems Engineering Center

Correspondence Patterns

- Analytical Model refines the Descriptive Model
- Structural ports are allocated to corresponding analytical ports
- Descriptive properties bound to analytical properties
Model Composition using Model Transformations

Descriptive Systems Model

Analytical Systems Model

Correspondence Models

Descriptive Models

Analytical Models
Descriptive to Analytical Transformation

Correspondence Models

Descriptive Models

Analytical Models
Summary

- **Objective:**
  - Leverage the strengths of both SysML and Modelica by integrating them to create a more expressive and formal MBSE language.

**Descriptive Modeling in SysML**

+ **Formal Equation-Based Modeling for Analyses and Trade Studies in Modelica**

http://doc.omg.org/syseng/2010-6-8
Acknowledgements

Working Group Members
- Yves Bernard (EADS)
- Roger Burkhart (Deere & Co)
- Wuzhu Chen (Univ. Braunschweig)
- Hans-Peter De Koning (ESA)
- Sandy Friedenthal (Lockheed Martin)
- Peter Fritzson (Linköping University)
- Nerijus Jankevicius (No Magic)
- Alek Kerzhner (Georgia Tech)
- Andreas Korff (Atego)
- Chris Paredis (Georgia Tech)
- Axel Reichwein (Georgia Tech)
- Nicolas Rouquette (JPL)
- Wladimir Schamai (EADS)

Students / post-docs
- Kevin Davies
- Sebastian Herzig
- Alek Kerzhner
- Ben Lee
- Roxanne Moore
- Marc Pare
- Axel Reichwein
- Wladimir Schamai

Sponsors
- Deere & Co
- Lockheed Martin
- National Science Foundation

http://doc.omg.org/syseng/2010-6-8