



# OMG: Transformation SysML4Modelica & simulations

L. RIOUX  
Thales Research & Technology

Date /Référence

Research & Technology

THALES

## SysML 4 MODELICA: (OMG)

- ◆ Modelica
- ◆ OMG standard: SysML4Modelica transformation

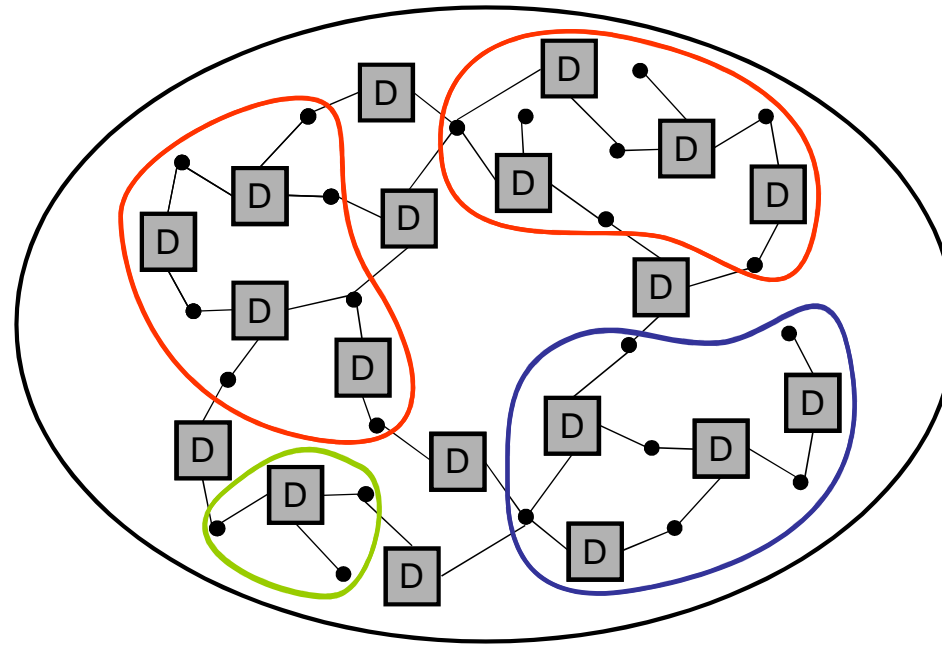
## Co-simulation: SysML & Simulink

- ◆ Simulink
- ◆ Co-simulation SysML/Simulink with Cosimate (IBM).

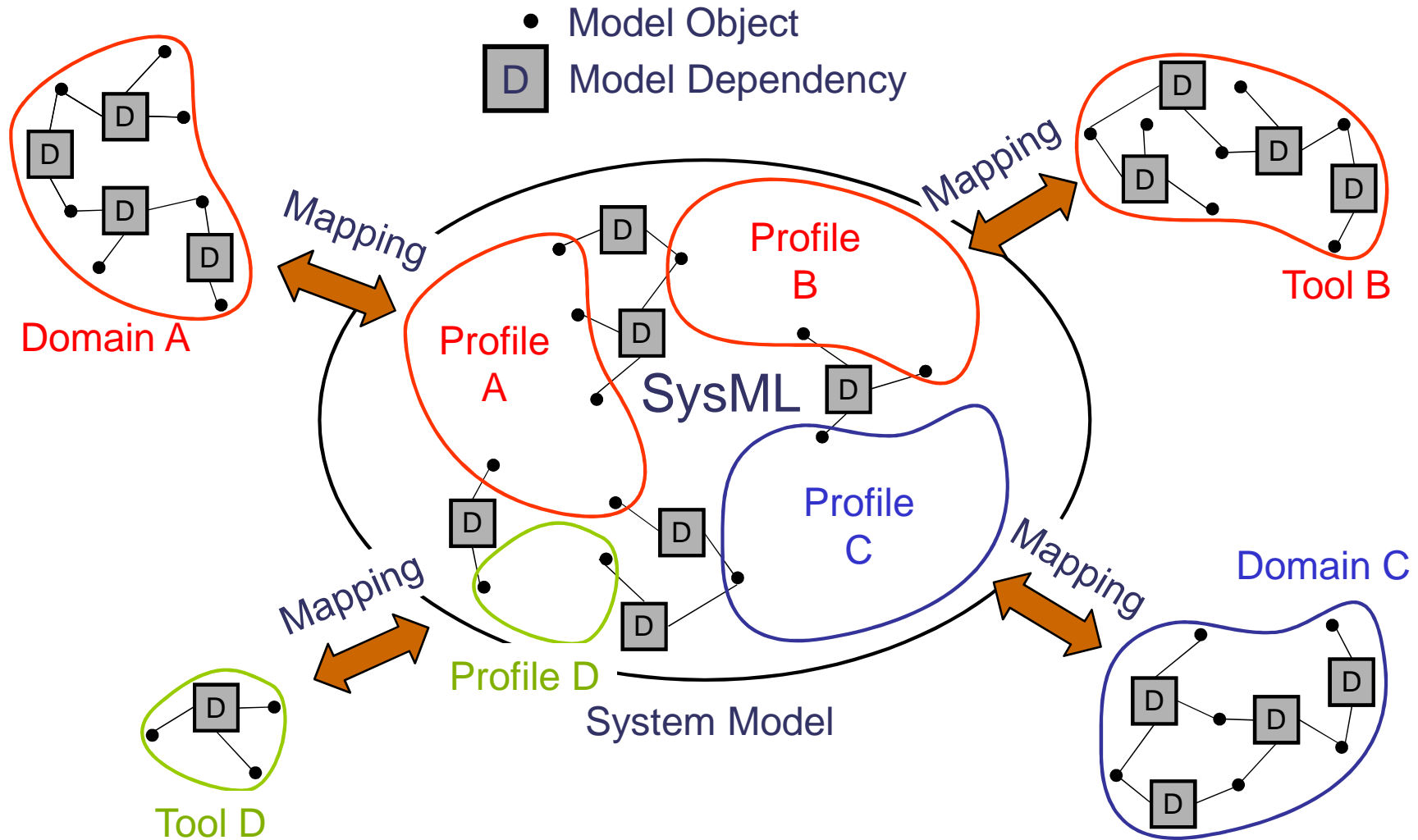
## Conclusions

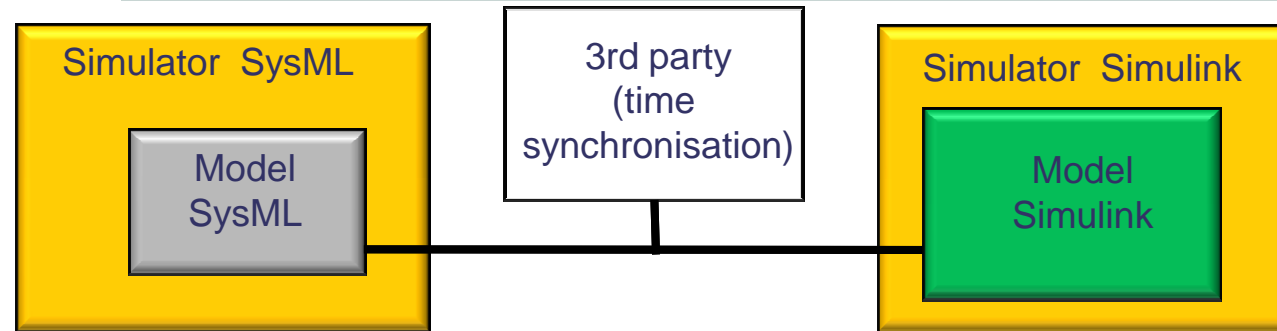
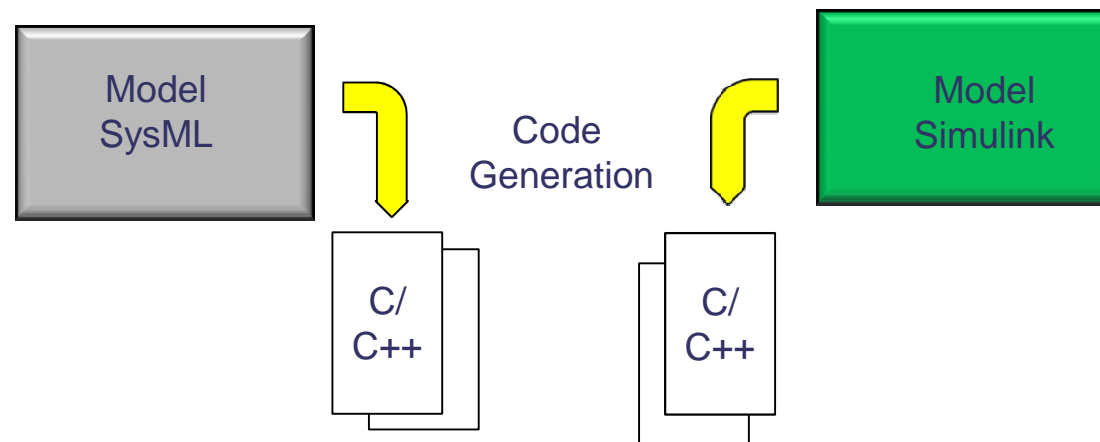
- Model Object

**D** Model Dependency



System Model



**1- Co-simulation****2- Code generation****3- Full Integration (SysML Centric)**

	Reichmann et al. (2004), Telelogic Rhapsody®	Hooman et al. (2004)A	ATEGO – ACE (2009) IBM-Cosimate (2010)	C. Nytsch-Geusen (2007), A. Pop et al., (2007), T. A. Johnson et al., (2008) <b>ModelicaML (2009)</b>  <b>SysML4Modelica (OMG 2010)</b>
Coupling method	Code Generation	Co-simulation of Simulink and Rose® RT	Co-simulation of Simulink / SysML execution	Full integration Conversion to Modelica
Simulation time management	By independent module (Rhapsody)	By Simulink	By independent module (Cosimate)	By Modelica

# Modelica language ([www.modelica.org](http://www.modelica.org))

## State-of-the-art Modeling Language for System Dynamics

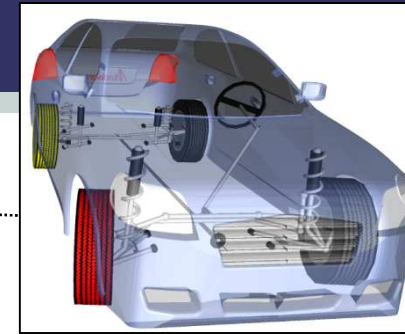
- ◆ Differential Algebraic Equations (DAE)
- ◆ Discrete Events

- ◆ Represents DAE models in an object-oriented, engineering-oriented language
- ◆ Multi-(physical)-domain modeling
- ◆ Permit **non-causal** modeling
- ◆ High level Specification

## Declarative Language

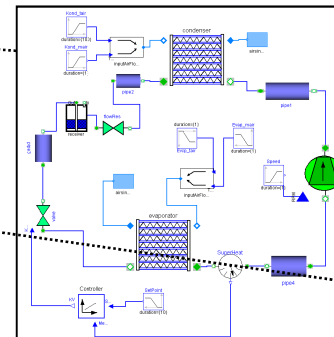
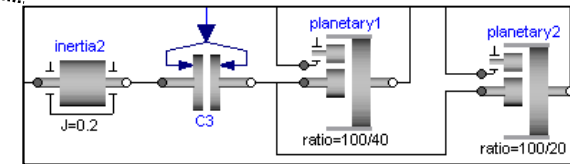
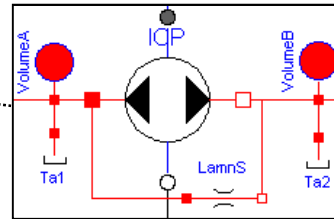
- ◆ With equations and mathematical functions
- ◆ Ports represent energy flow (undirected) or signal flow (directed)
- ◆ A causal, equation-based, declarative ( $f-m*a=0$ )



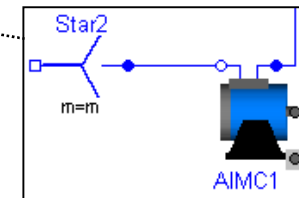
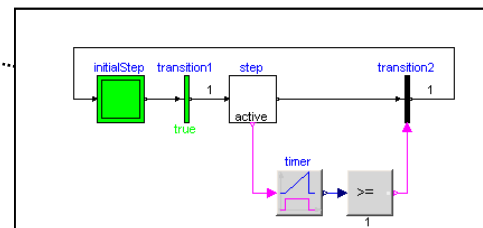
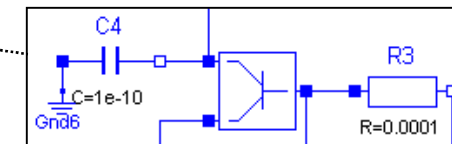
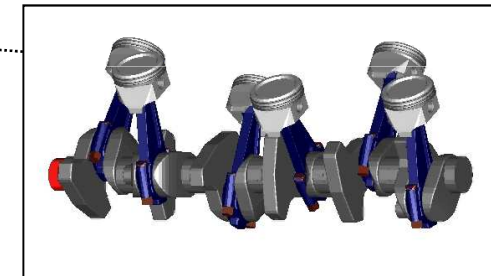


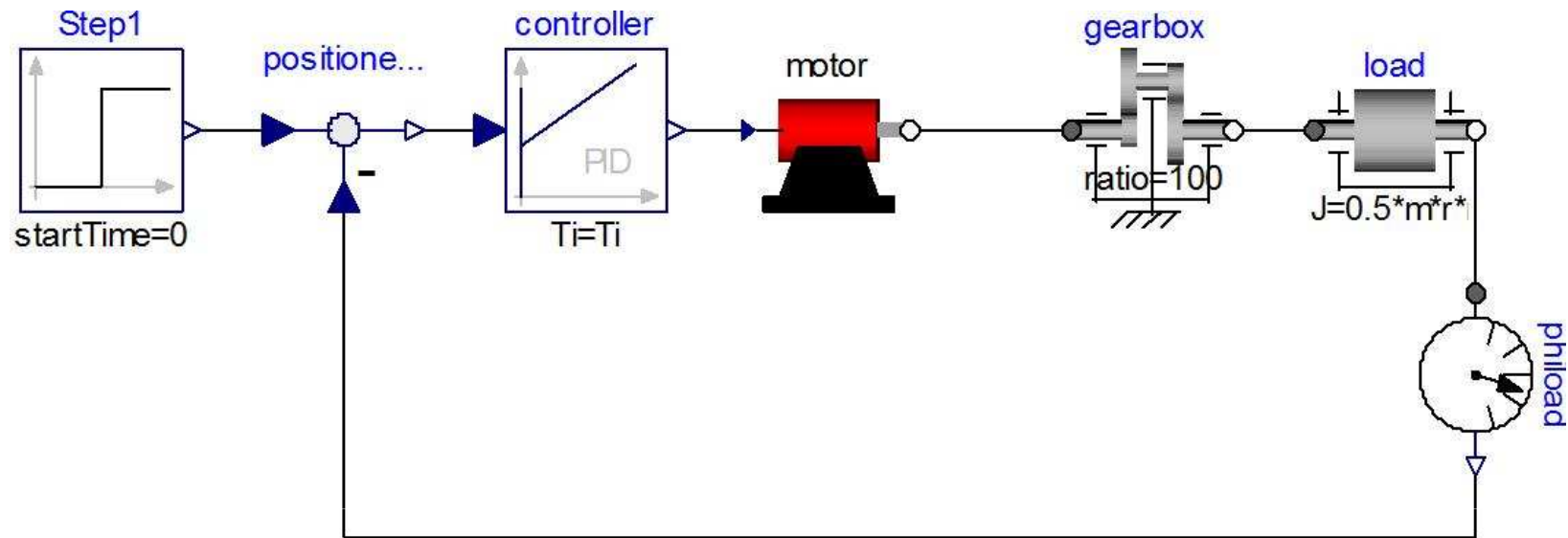
courtesy: Modelon AB

- **Vehicle dynamics** (3-dim. mechanics)
- **Drive trains** (1-dim. mechanics)
- **Hydraulics**
- **Combustion**
- **Air Conditioning** (Thermofluid systems)
- **Electrical/electronic systems**
- **Electrical machines**
- **Hierarchical state machines**
- **Control** (Input/output blocks, ...)



courtesy Modelon AB





```

class LowPassFilter
parameter Real T=1;
Real u, y(start=1);
equation
    T*der(y) + y = u;
end LowPassFilter;

```

```

algorithm
y := 0;
xpower := 1;

for i in 1:n+1 loop
    y := y + a[i]*xpower;
    xpower := xpower*x;
end for;

```

Full Integration

SysML & Modelica:

**SysML4Modelica**

**(standard OMG)**

**Version 1.0: march 2012**

THALES

## SYSML

- ◆ Language for architectural system design
- ◆ Contain Structure and Control Concepts
  - Control: Dataflow (IDB) and Events control (inherits from UML)
- ◆ No Equation language
- ◆ Compliant with MDA standard approach

## Mathematical Languages

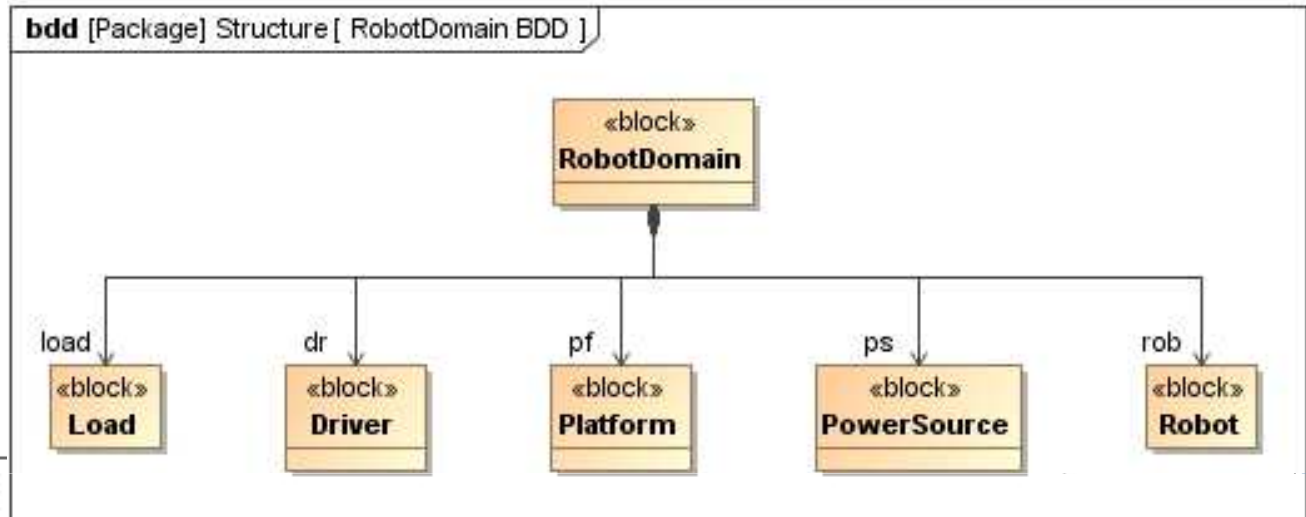
- ◆ Matlab / modelica
  - Clearly competitors: same usage domain with some differences
- ◆ Simulink / SysML4Modelica
  - Proprietary language - Standard Graphical language

## Two complementary languages for Systems Engineering:

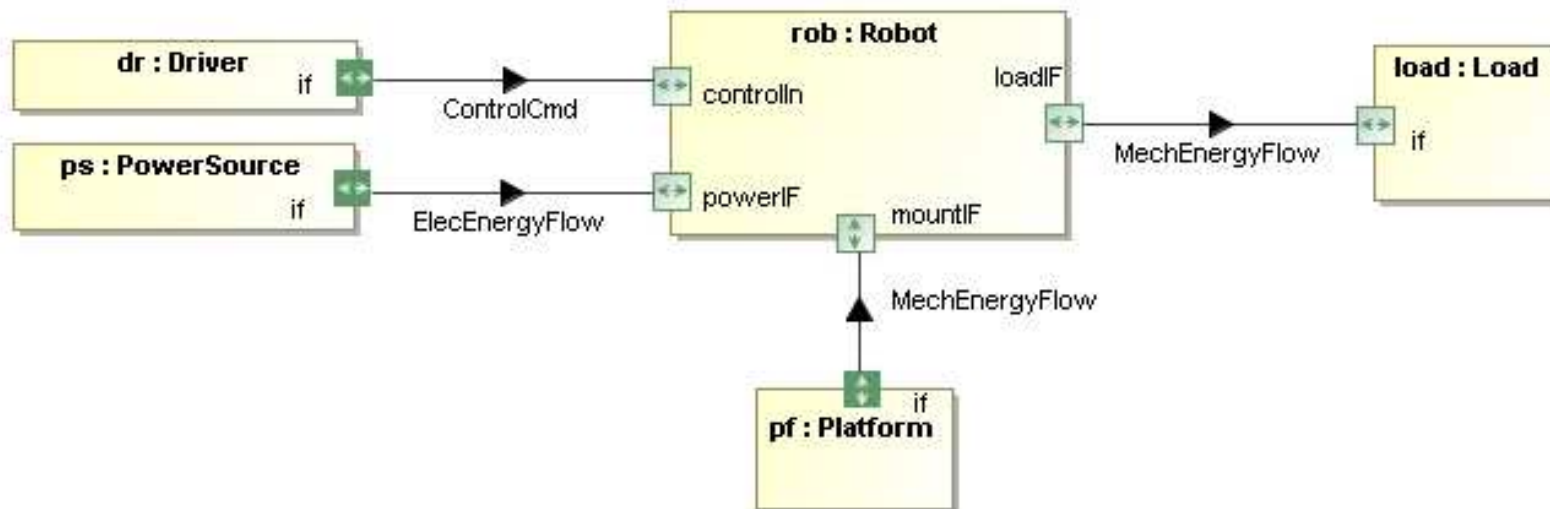
- ◆ Descriptive modeling in SysML
- ◆ Formal equation-based modeling for analyses and trade studies in Modelica

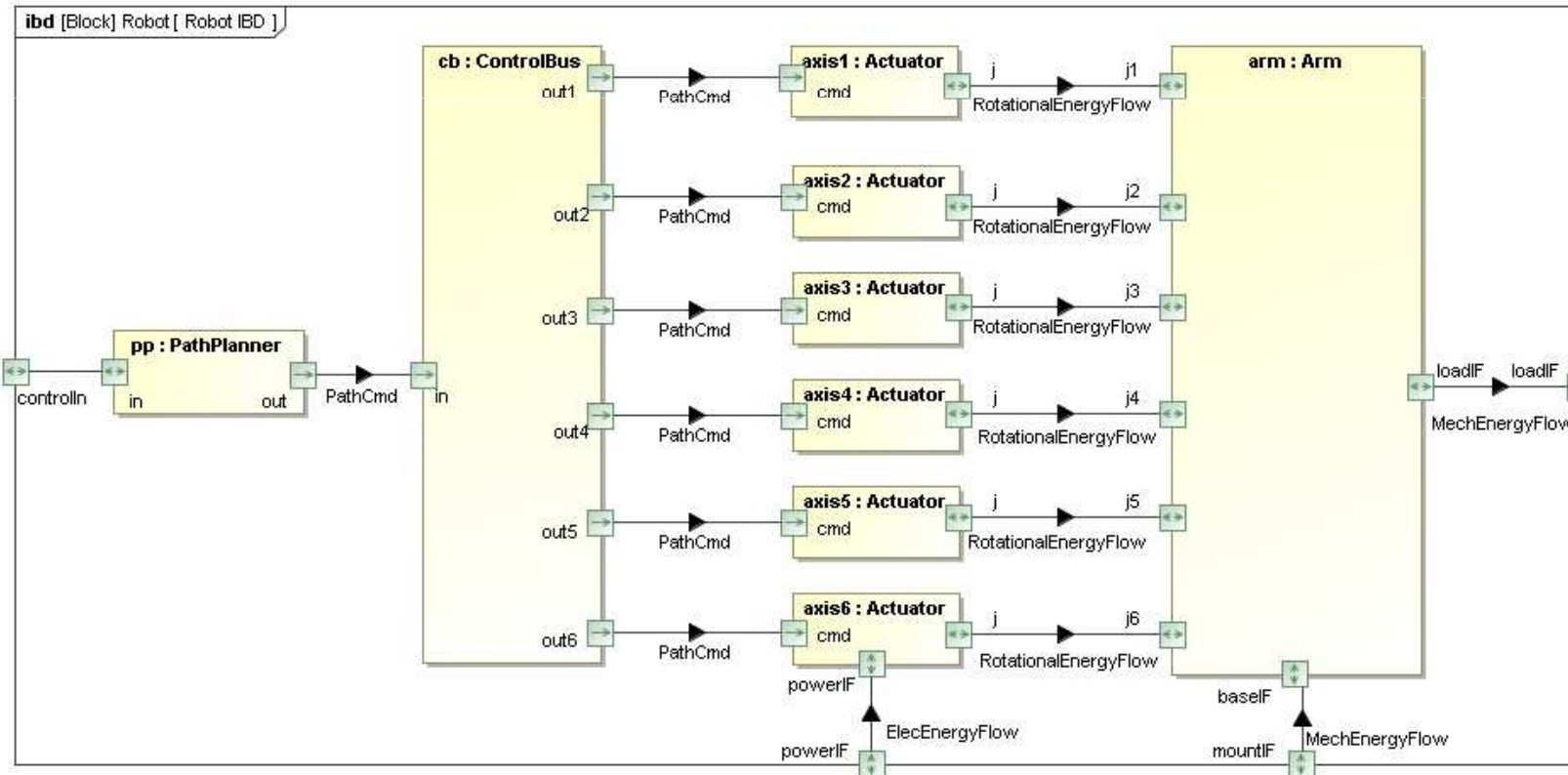
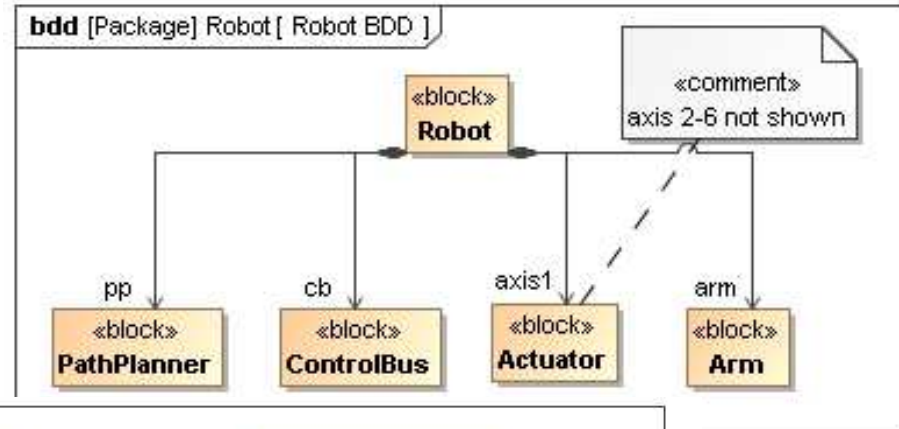
## Objective:

- ◆ Leverage the strengths of both SysML and Modelica by integrating them to create a more expressive and formal MBSE language.
  - Extend **only** the structural part of the SysML Languages
  - Add **Modelica semantics** to SysML (with stereotype)
- ◆ Define a formal Transformation Specification:
  - a SysML4Modelica profile
  - a Modelica abstract syntax metamodel
  - a mapping between Modelica and the profile



**ibd** [Block] RobotDomain [ RobotDomain IBD ]

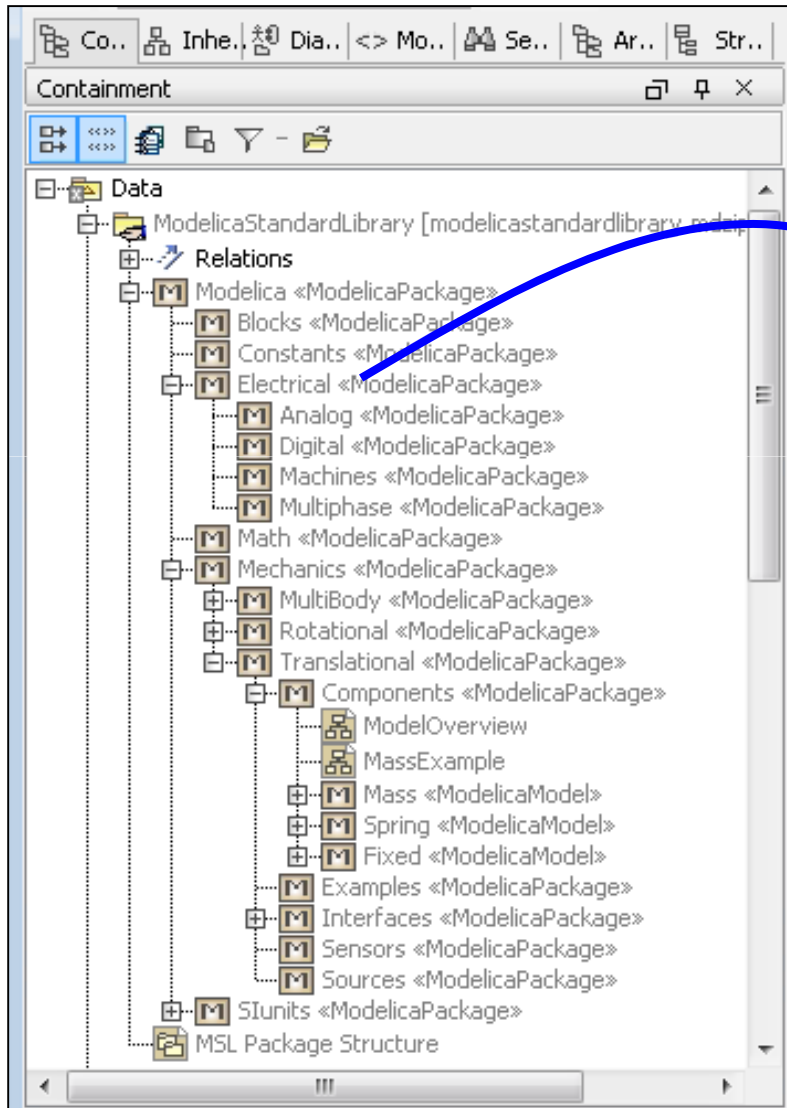




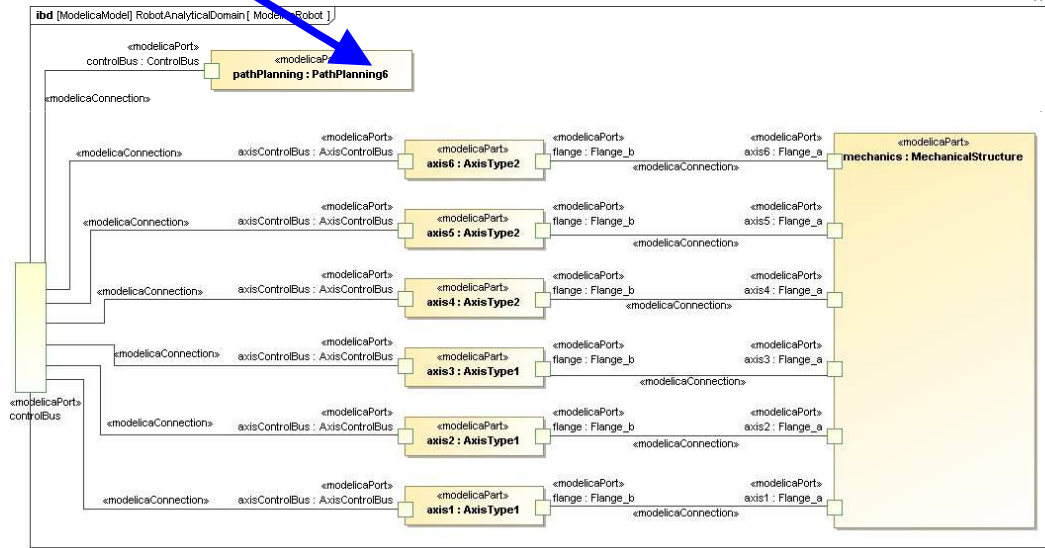
The information contained in this document and any attachments are the property of THALES. You are hereby notified that any review, dissemination, distribution, copying or otherwise use of this document is strictly prohibited without Thales prior written approval. ©THALES 2011. Template: rtp version 7.0.8



# SysML4Modelica Analytical Model: Compose Model from Standard Library



Drag and drop into IBD «ModelicaModel»

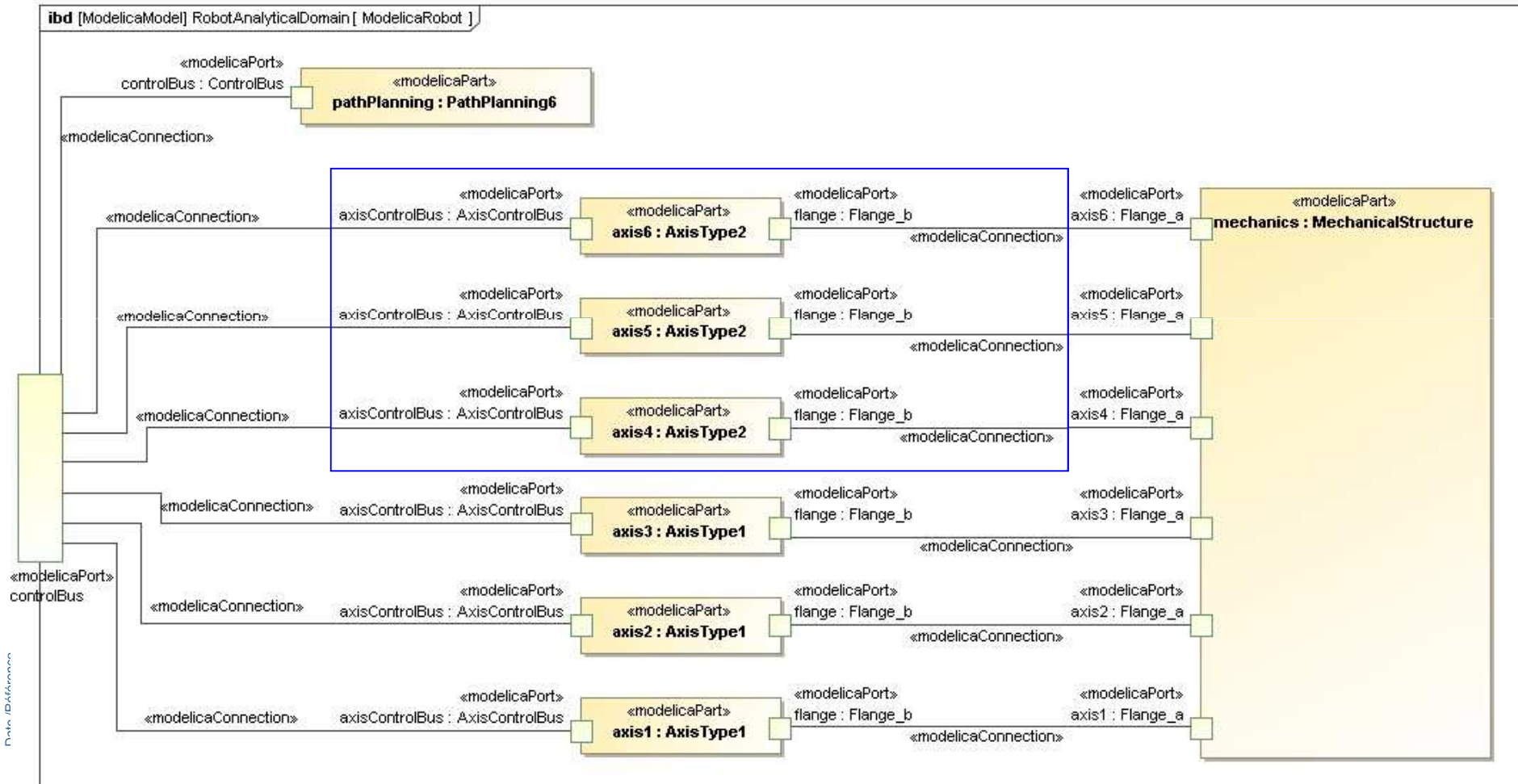


Date /Référence

verified that any review, dissemination, distribution, copying or late ttp version 7.0.8

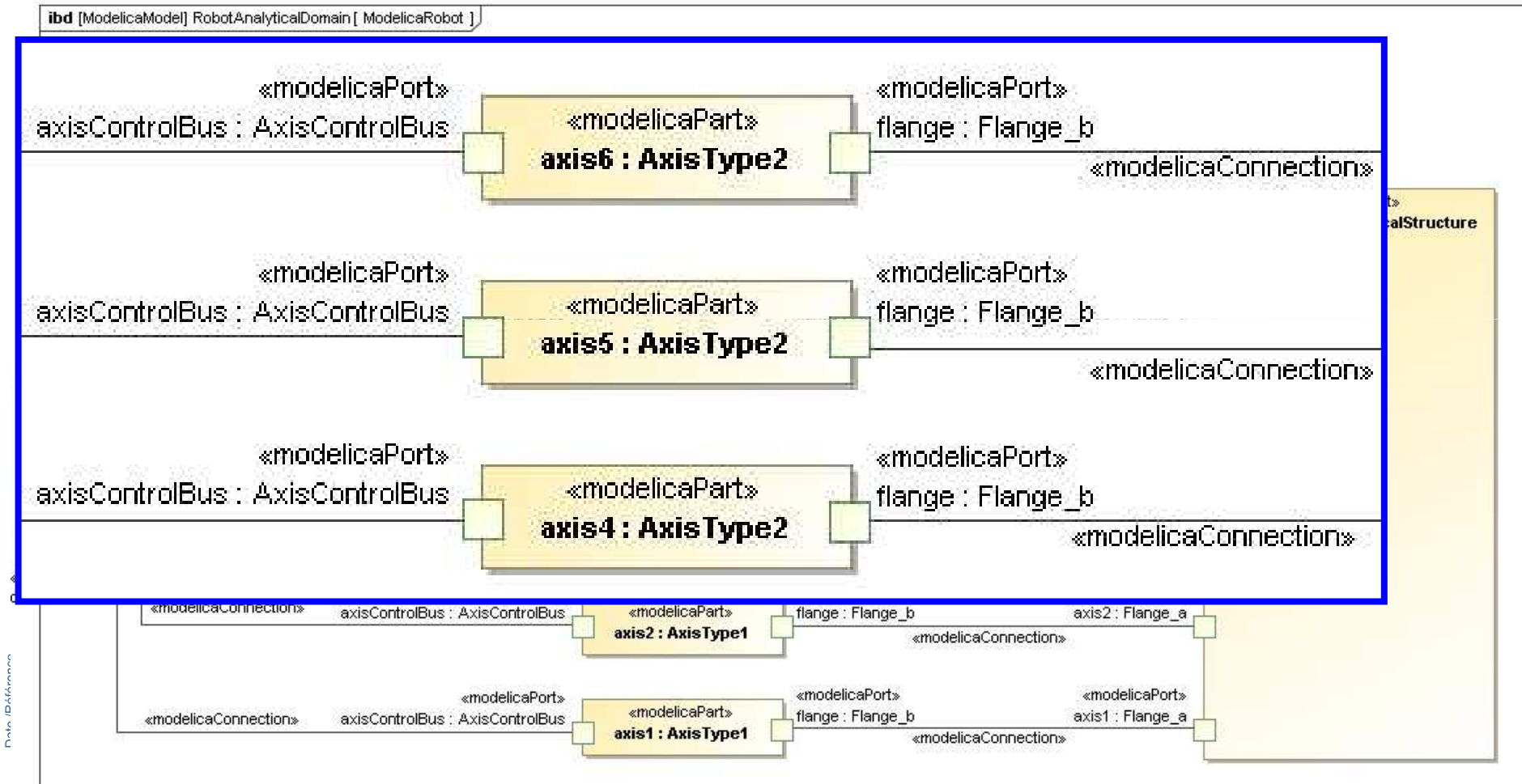
The information is otherwise use of





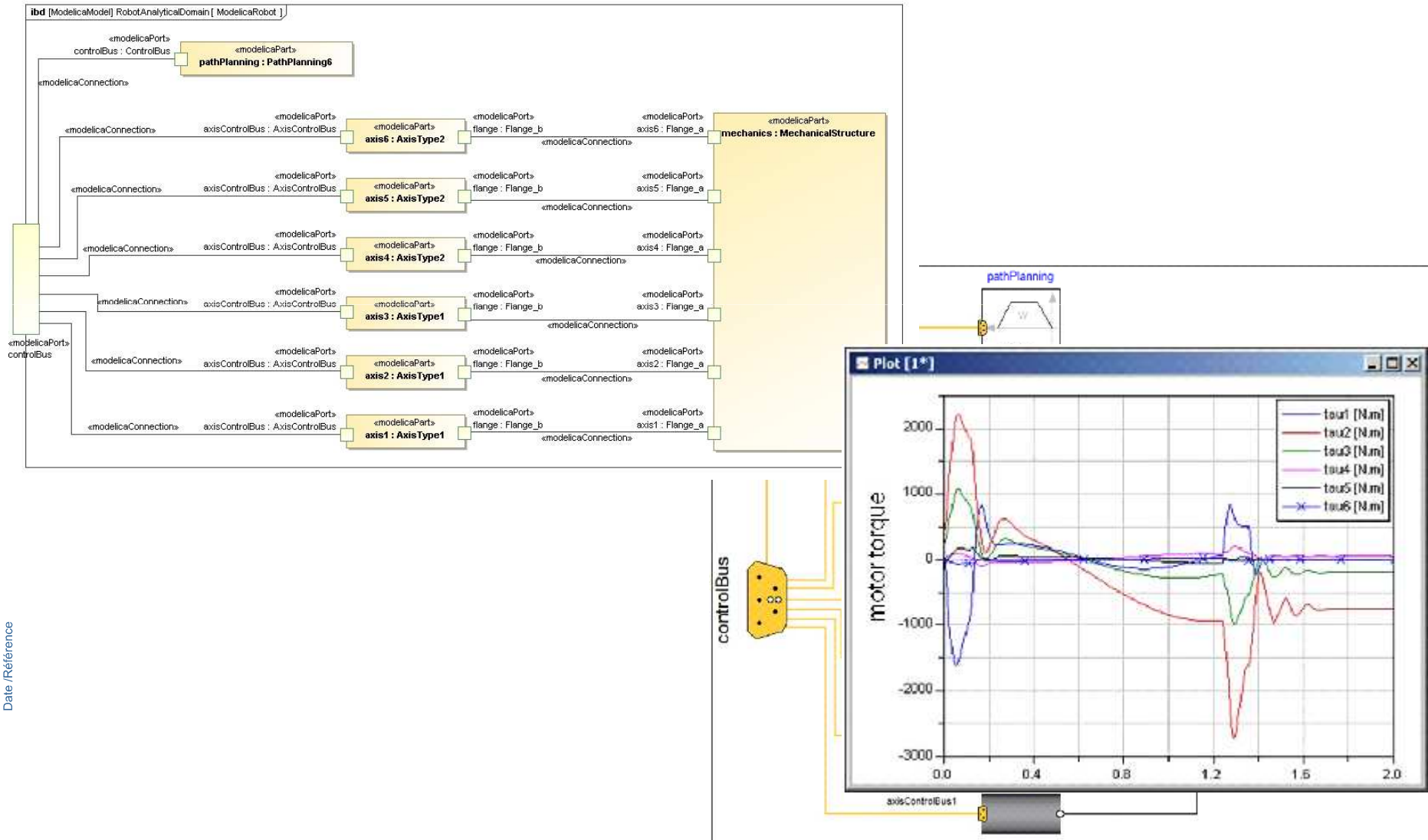
Data: IBD44444444

copying or



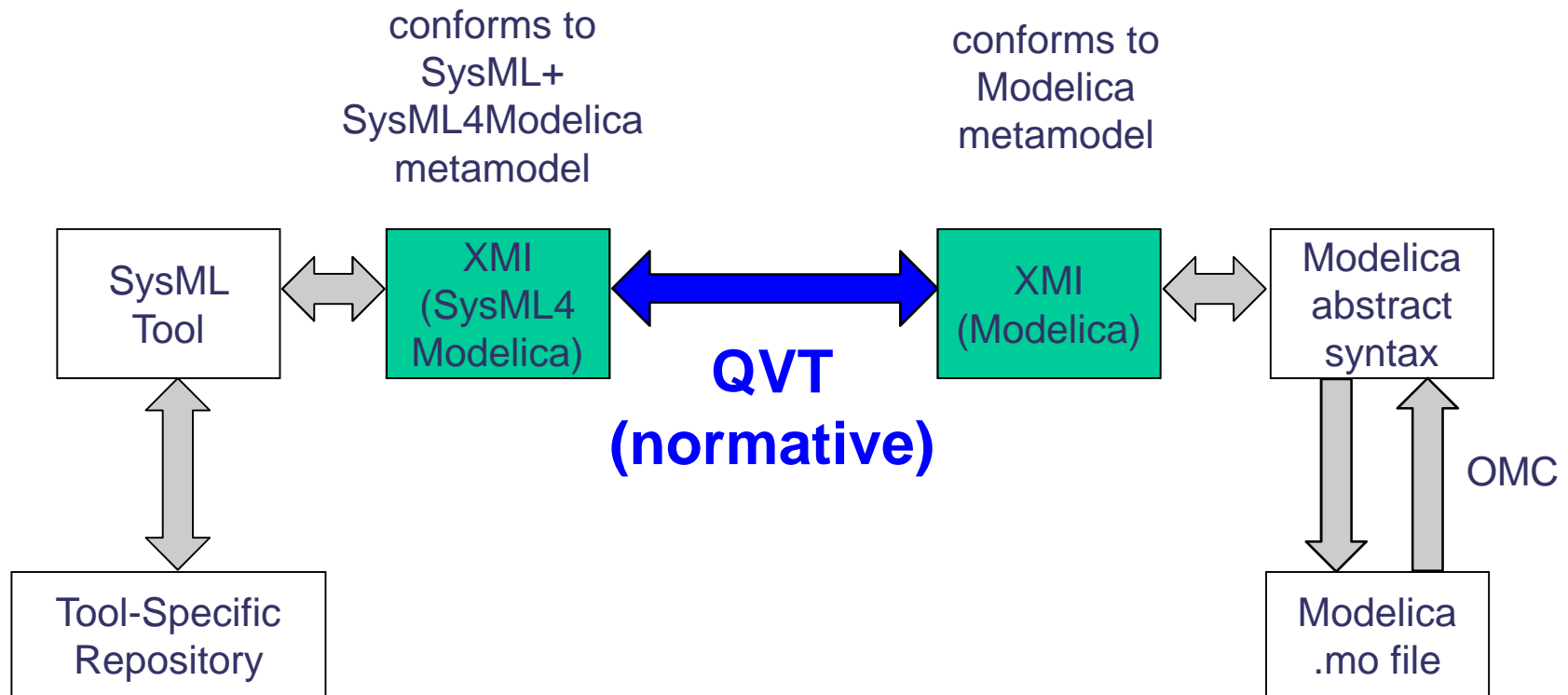
this IBD defines

# SysML4Modelica Analytical Model: Relation to Modelica Native Model



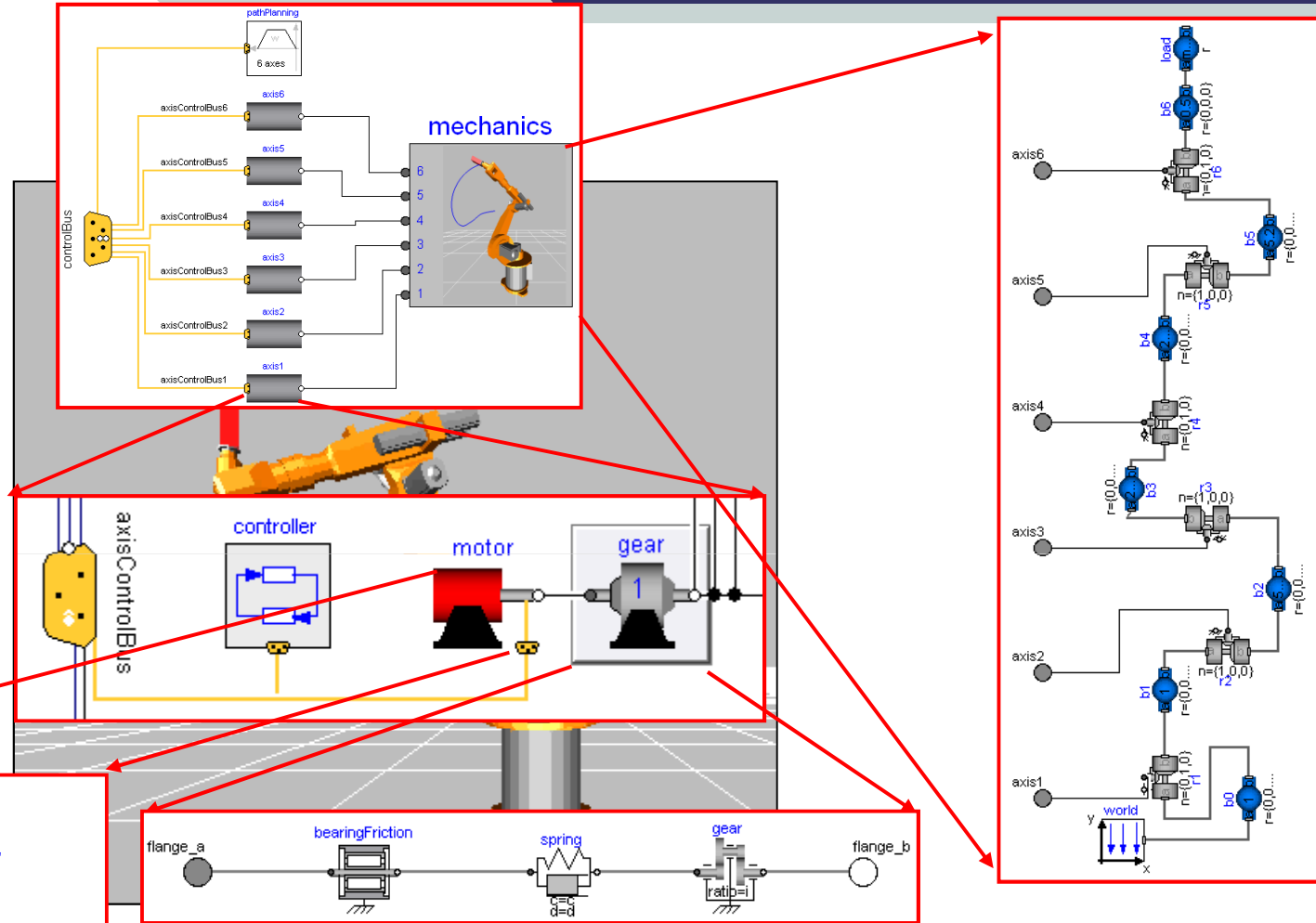
Date /Référence

## Standard Formal Mapping Between SysML4Modelica To Modelica



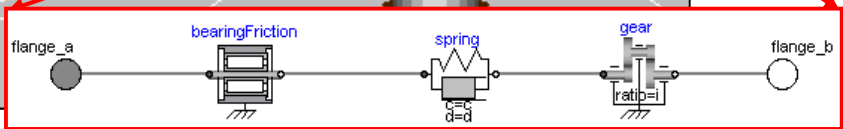
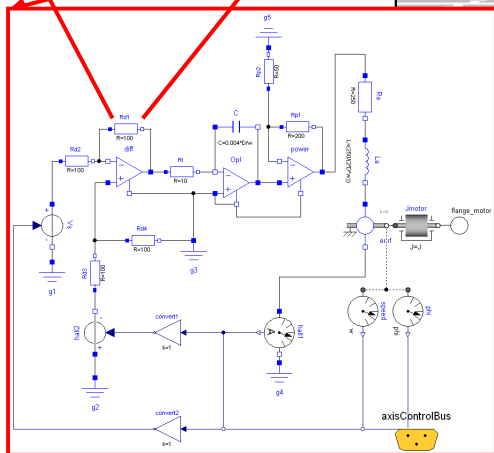
QVT = Query / View / Transformation

# Example: Industrial Robots



```

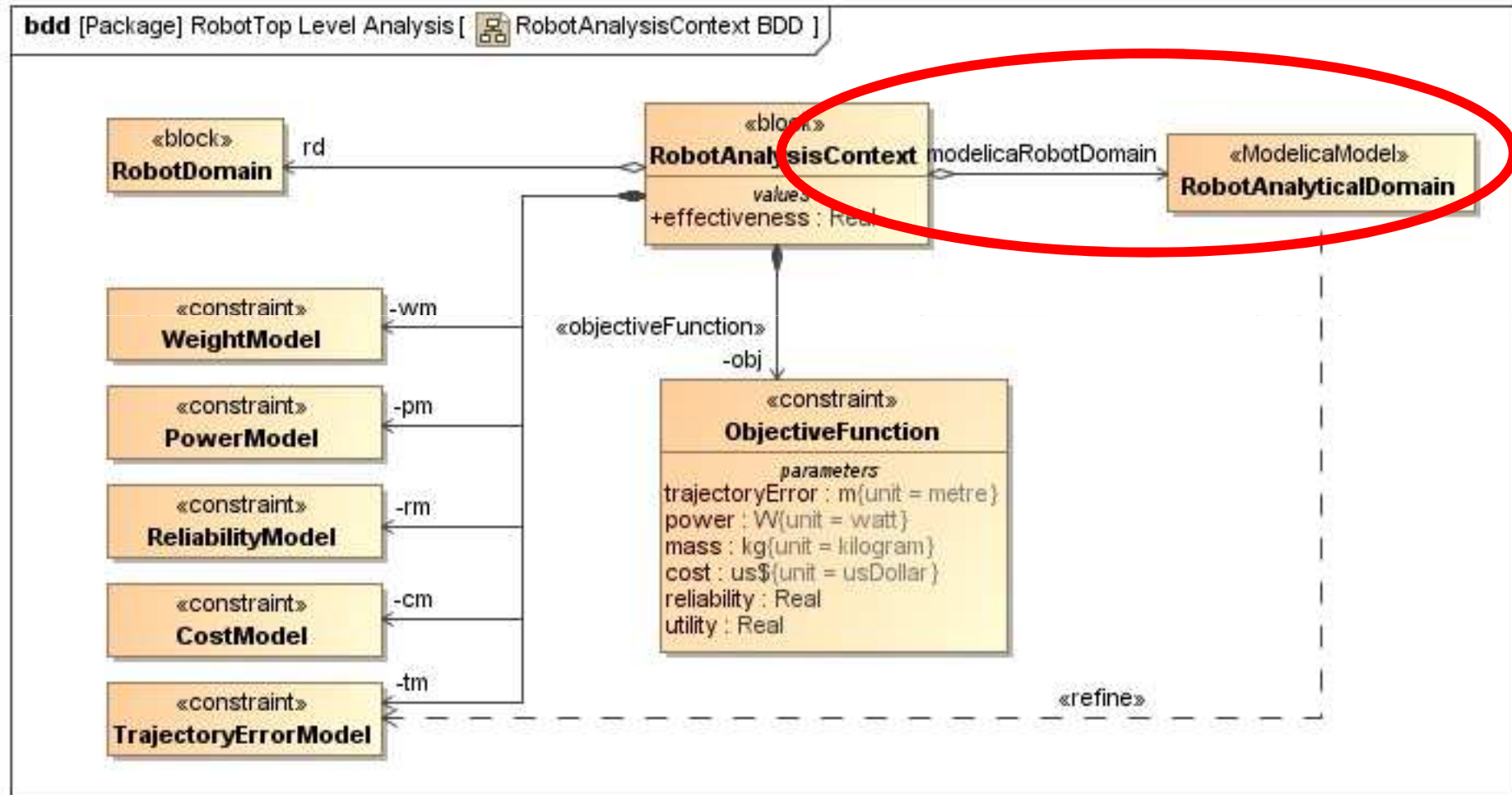
model Resistor
  extends OnePort;
  parameter Real R;
  equation
    v = R*i;
  end Resistor;
  
```

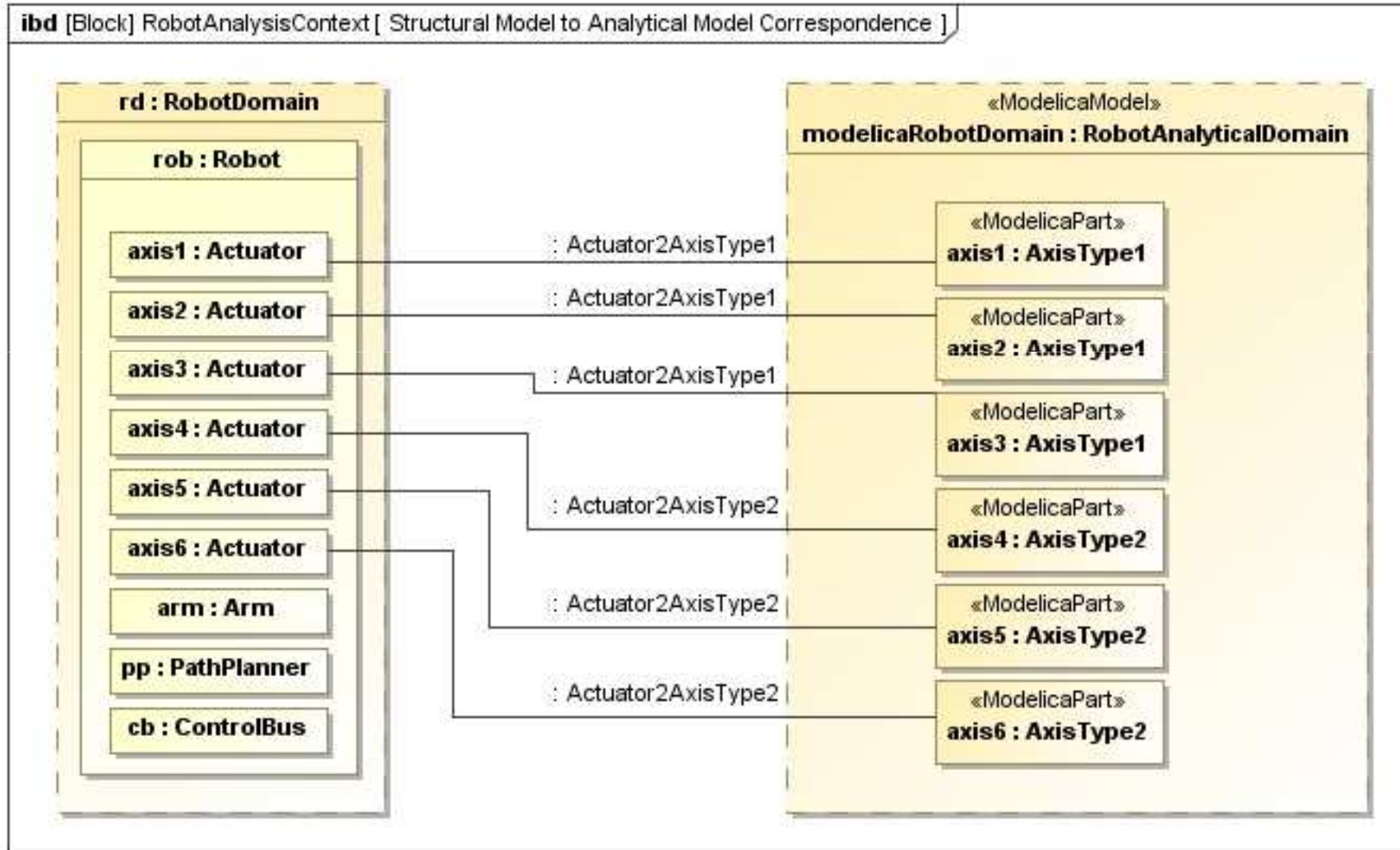


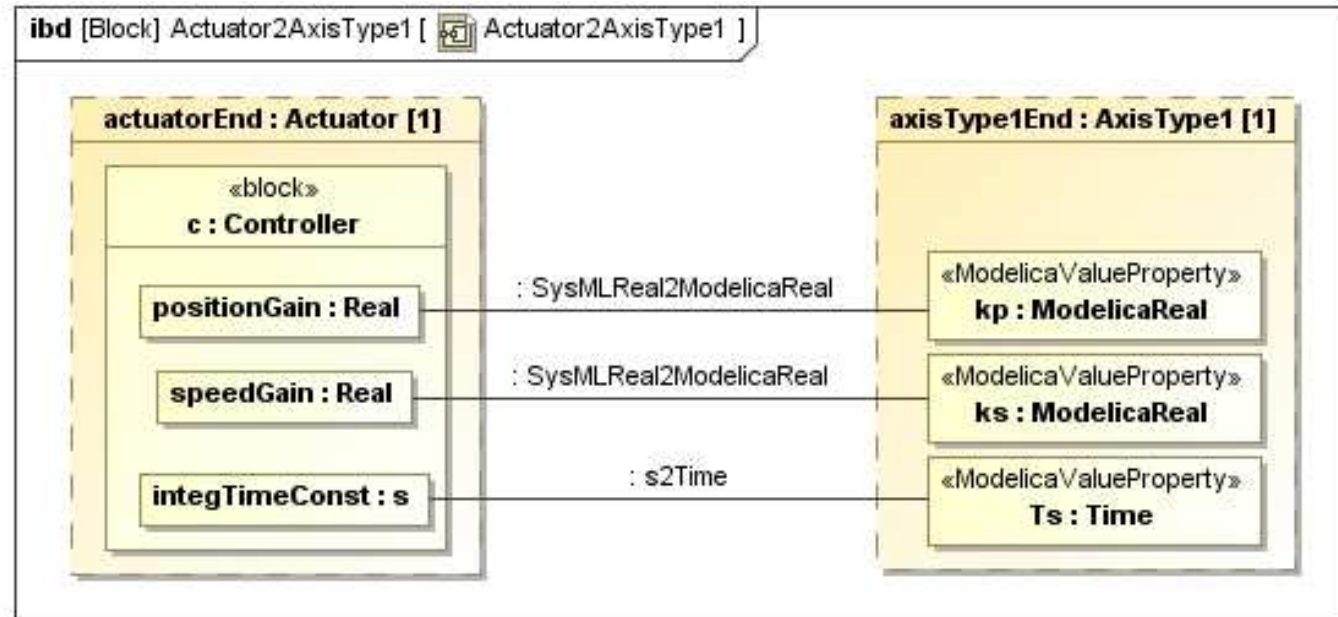
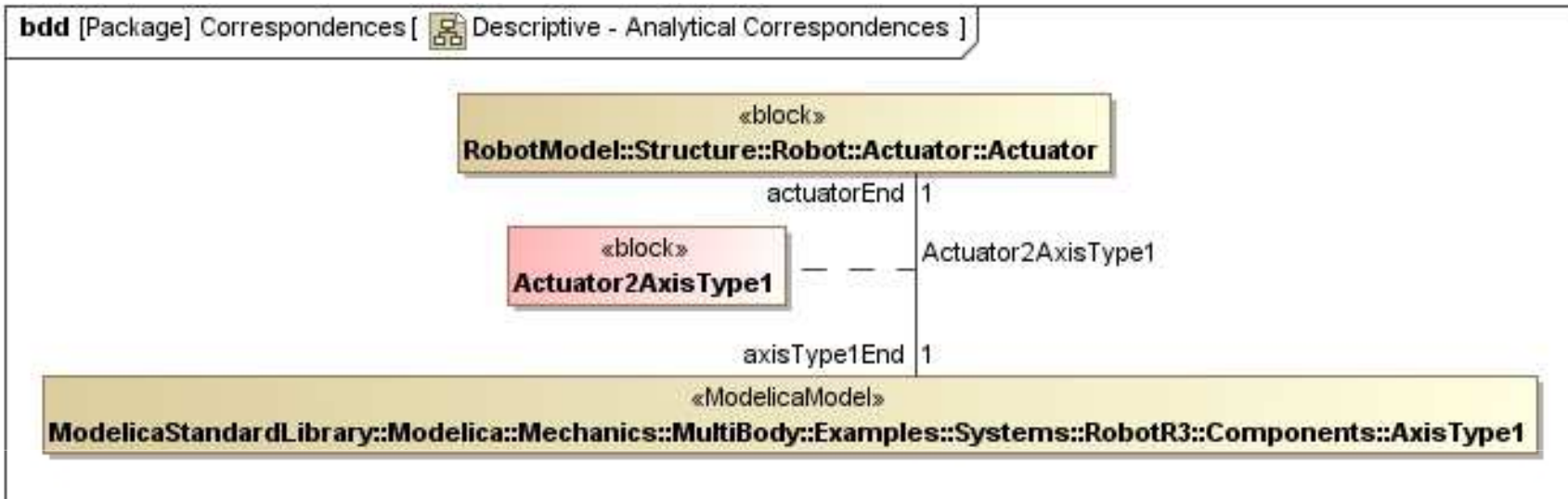
**1000 non-trivial algebraic equations, 80 states.**  
**Faster** as real-time on slow PC.

Date /Référence

The information contained in this document and any attachments are the property of THALES. You are hereby notified that any review, dissemination, distribution, copying or otherwise use of this document is strictly prohibited without Thales prior written approval. ©THALES 2011. Template ttp version 7.0.8

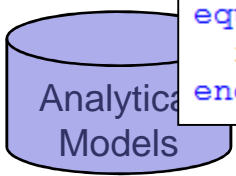
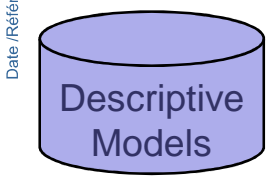
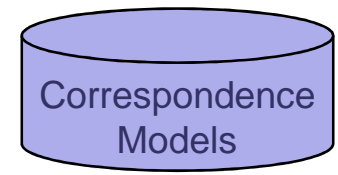
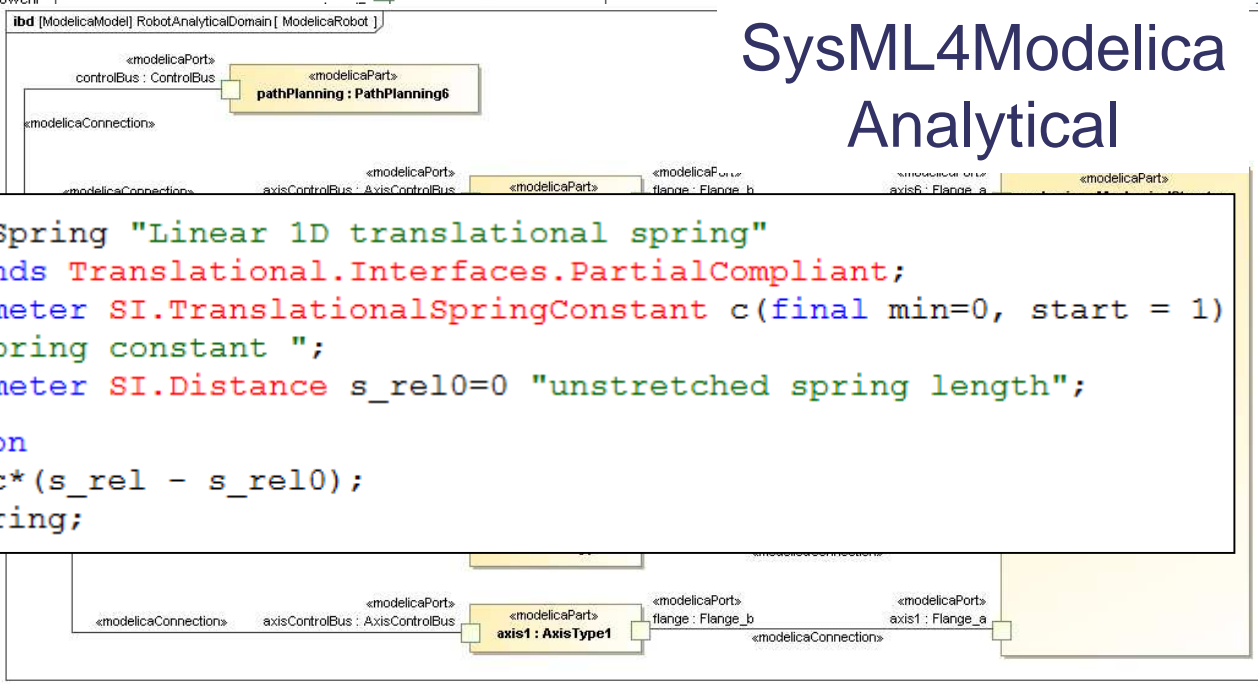
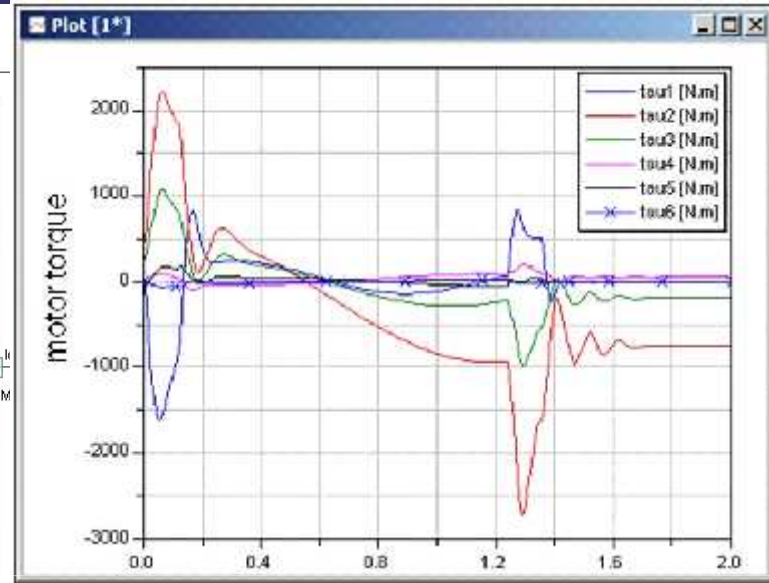
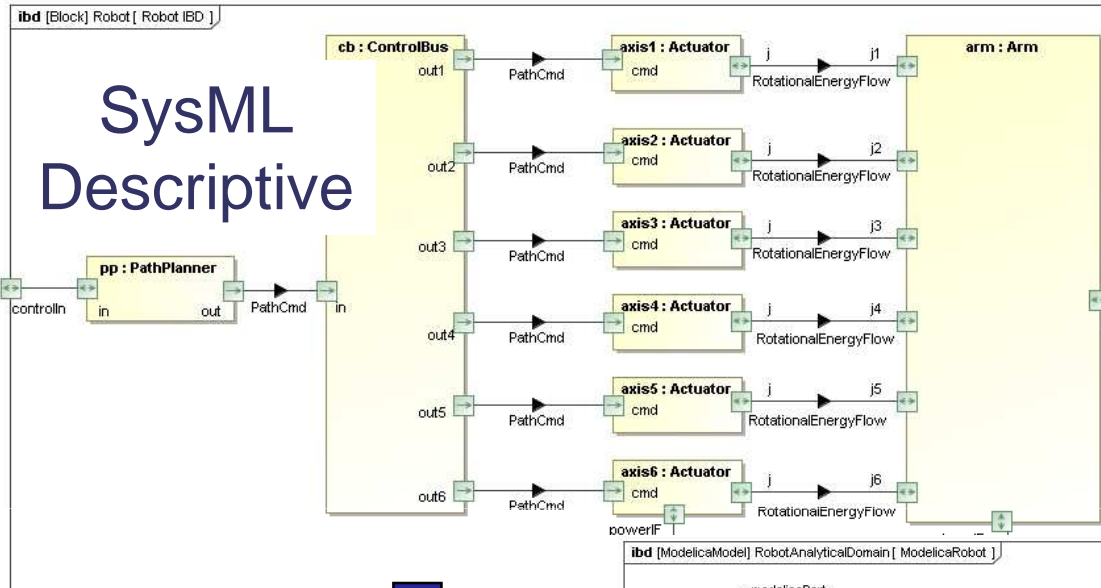




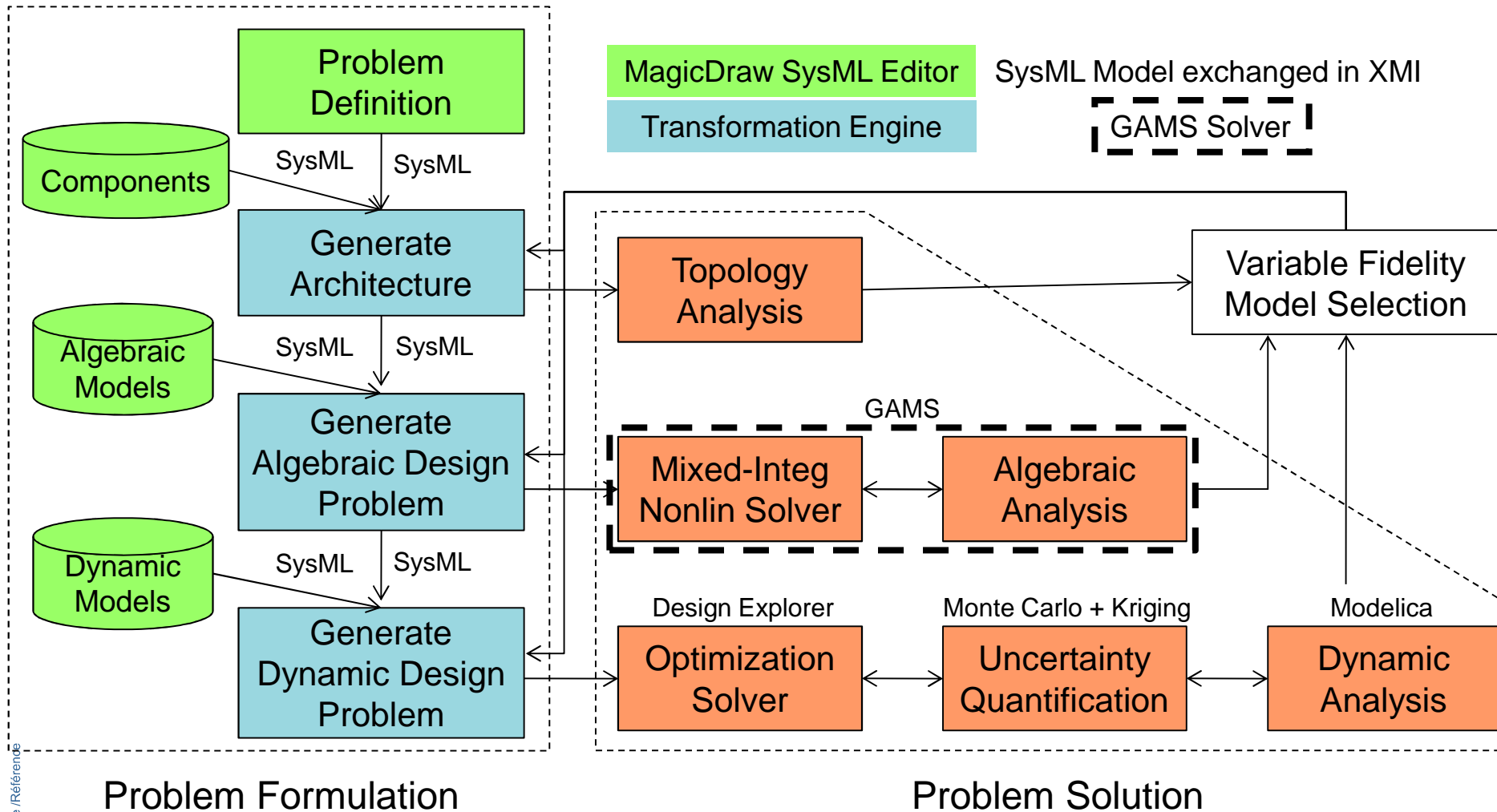




# (OMG) Model Transformations in MBSE: Descriptive to Analytical Transformation



Date /Référence



Date /Référence

The information contained in this document and any attachments are the property of THALES. You are hereby notified that any review, dissemination, distribution, copying or otherwise use of this document is strictly prohibited without Thales prior written approval. ©THALES 2011. Template rtp version 7.0.8

- ◆ 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://www.omg.org/spec/SyM/>

THALES

## Approach adopted by:

- ◆ **ATEGO (product Artisan)**
  - Parasolver (Sysml/Modelica)
  
- ◆ **No Magic (Magicdraw plug-ins: Cameo simulation)**
  - Prototype already available

## ModelicaML (UML profile for Modelica) – EADS IW (Uk)

- Extension UML Behavior for modelica (state-machine and activity diagrams)
- Will extend SysML4Modelica

# Another approach

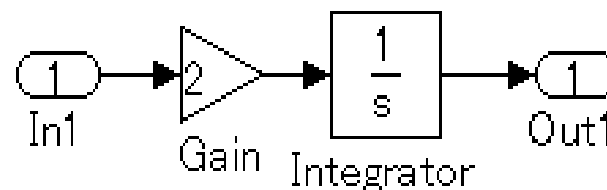
## SysML & Matlab/Simulink

### Co-simulation

## Simulink is the graphical notation of Matlab language

### Simulink uses Block Diagram

- ◆ Block diagram describes the flow of signal between blocks, which process the signal



- ◆ This example calculates:

$$f_{\text{Out1}}(t) = \int_0^t 2f_{\text{In1}}(\tau) d\tau$$

**Library: simulink**

File Edit Simulation Format Tools

Sources Sinks Discrete Linear Nonlinear Connections

Blocksets & Toolboxes Simulink Block Library 2.2 Copyright (c) 1990-1998 by The MathWorks, Inc. Demos

**Library: simulink/Nonlinear**

File Edit Simulation Format Tools

Abs Trigonometric Function Math Function Rounding Function MinMax

Product Combinatorial Logic Logical Operator Relational Operator Sign

Rate Limiter Saturation Quantizer Coulomb & Viscous Friction

Backlash Dead Zone Look-Up Table Look-Up Table (2-D)

Memory Transport Delay Variable Transport Delay Hit Crossing

f(u) Fcn MATLAB Function MATLAB Fcn S-Function

Switch Manual Switch Multipoint Switch

Relay Algebraic Constraint

**Library: simulink/Sources**

File Edit Simulation Format Tools

Constant Signal Generator Step

Ramp Sine Wave Repeating Sequence

Discrete Pulse Generator Pulse Generator Chirp Signal

Clock Digital Clock

From File From Workspace

Random Number Uniform Random Number Band-Limited White Noise

**simulink/Discrete**

File Edit Simulation Format Tools

Unit Delay Discrete-Time Integrator

Zero-Order Hold First-Order Hold

Discrete State-Space

Discrete Filter Discrete Transfer Fcn

Discrete Zero-Pole

**simulink/Linear**

File Edit Simulation Format Tools

Gain Sum

Integrator Transfer Fcn

State-Space Zero-Pole

Derivative Dot Product

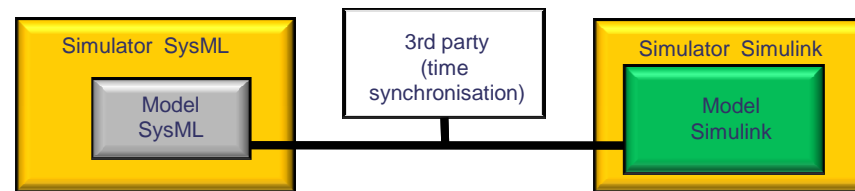
Matrix Gain Slider Gain

The information contained in this document is the property of THALES. The information contained in this document is strictly prohibited without THALES prior written approval. ©THALES 2011. Template: tip version 7.0.8

# Co-Simulation

## SysML / Simulink

### RSA – Cosimate - Simulink





**Use SysML for a system architecture description**

**Use specific languages for each domain**

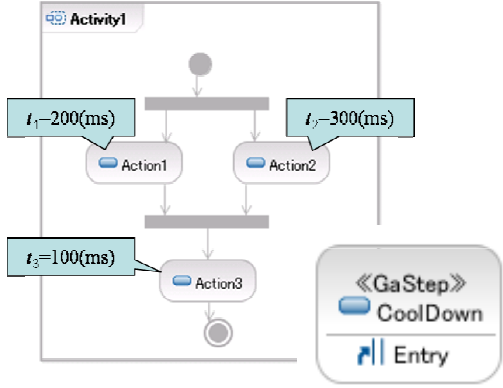
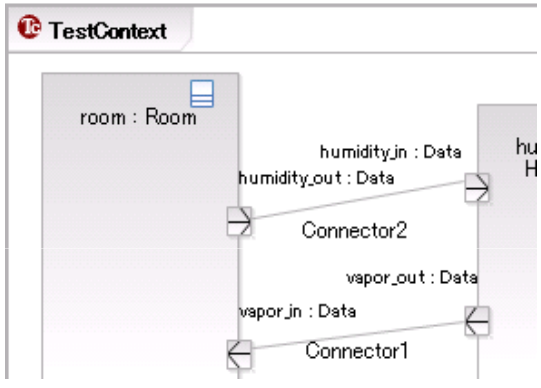
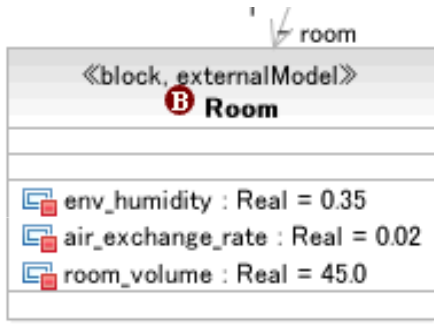
- ◆ Simulink for continuous-time behavior in a control system

**Time duration assignment to SysML action (MARTE)**

**Verify specification by collaborative simulation**

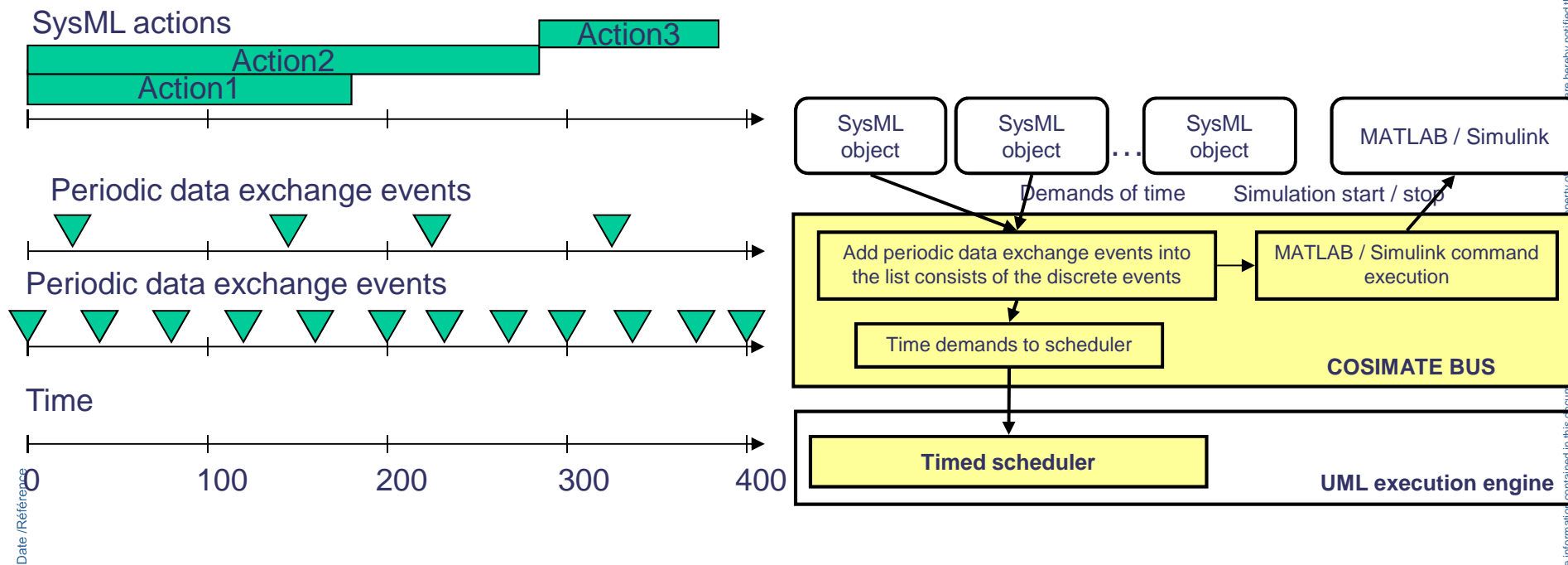
- ◆ Run multiple simulation in parallel with time synchronization
- ◆ Independent time management module

# Extended Modeling for Simulation in SysML (IBM)

<p>Time assignment to action</p>	<p>Continuous data exchange between blocks</p>	<p>Specifying Simulink model as a block behavior</p>
		
<p>MARTE &lt;&lt;GaStep&gt;&gt; stereotype, or sleep() method in action code</p>	<p>SysML flow ports with connectors in internal block diagram</p>	<p>&lt;&lt;externalModel&gt;&gt; stereotype to specify model file</p>

## Data is exchanged and the next Simulink step is executed at:

- ◆ End of timed SysML behavioral step (<<GaStep>> or sleep())
- ◆ Periodic data exchange event

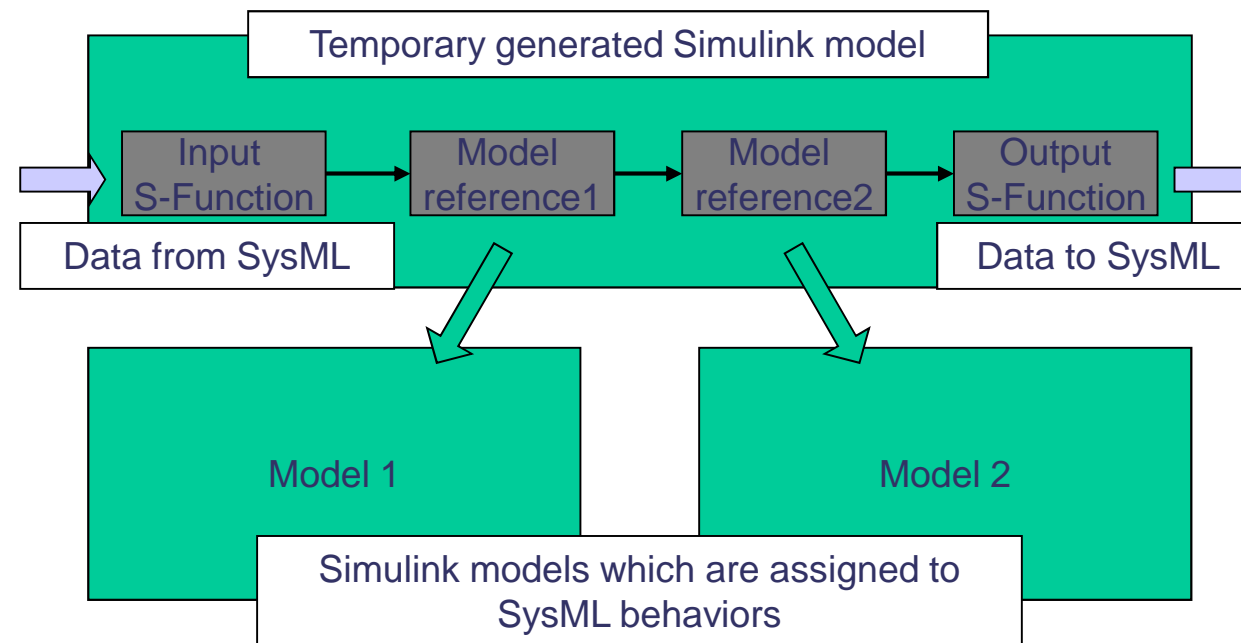


# Co-simulation with Simulink (IBM)

**All the Simulink models are referred from a single temporary Simulink model**

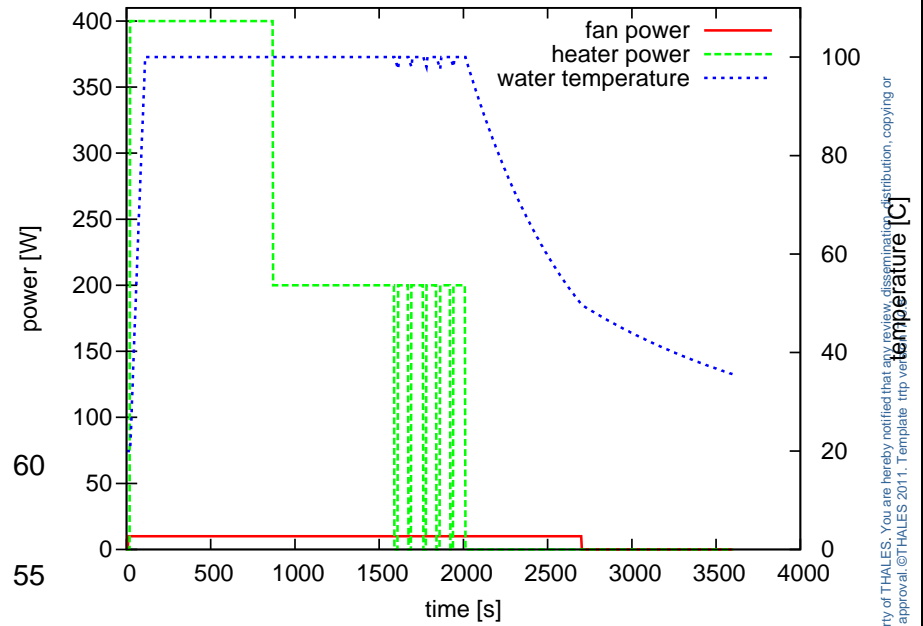
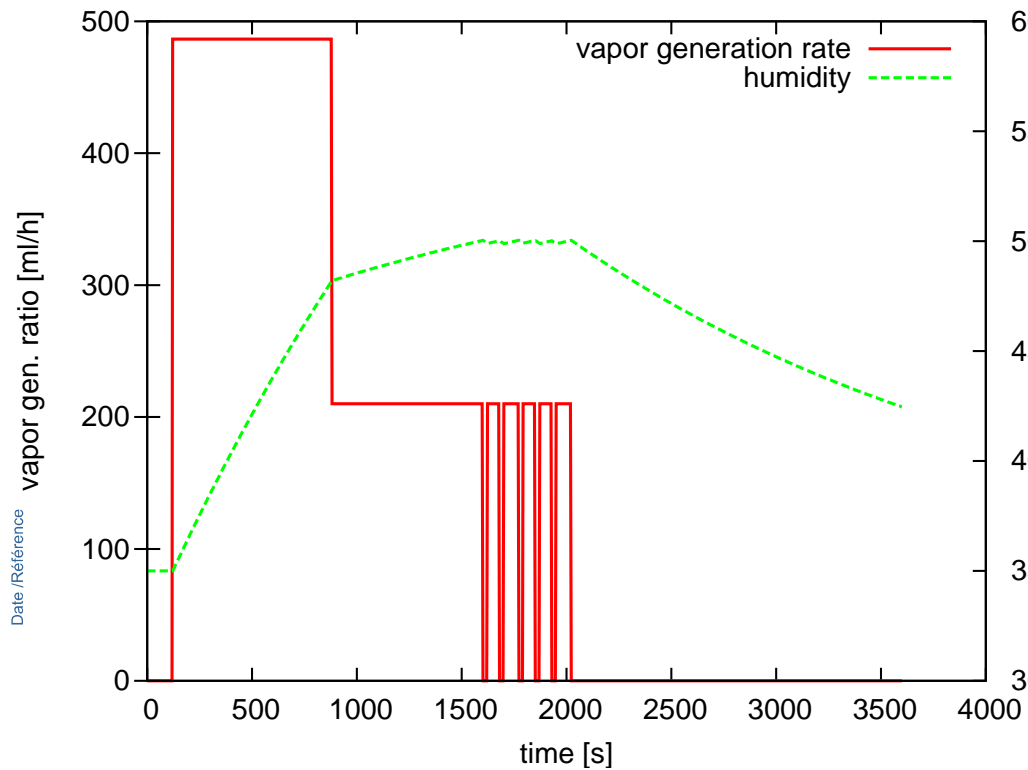
- ◆ Only one Simulink process is used

**Simulation data is sent or received through Input/Output S-Functions**



# Response Time analysis

Time(s)	Command
0	Test start
10	Power on
2000	Power off
3600	Test end



The information contained in this document and any attachments are the property of THALES. You are hereby notified that any review, dissemination, distribution, copying or otherwise use of this document is strictly prohibited without Thales prior written approval. ©THALES 2011. Template: ttp-vel-11-01-01

## 2 different solutions:

- ◆ **Fully based on SysML (OMG)**
  - Require to model everything inside the SysML Tool
  
- ◆ **Co-design: (promoted by tool vendors)**
  - SysML tool use to model the design of the system
  - Modelica tool use to model the analytical model and provide simulation

## Working Group Members

- ◆ Yves Bernard (EADS-AIRBUS)
- ◆ 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 (NASA JPL)
- ◆ Wladimir Schamai (EADS IW)

## Students / post-docs

- ◆ Kevin Davies
- ◆ Sebastian Herzig
- ◆ Alek Kerzhner
- ◆ Ben Lee
- ◆ Roxanne Moore

## Sponsors

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

With The contribution of the Modelica Association ([www.modelica.org](http://www.modelica.org))

### Commercial Modelica Simulation Environments (alphabetical list)

- ◆ **CATIA Systems** from Dassault Systèmes (based on Dymola kernel with PLM integration)
- ◆ **Dymola** from Dynasim AB, Sweden (Dynasim was acquired by Dassault Systèmes in 2006).
- ◆ **LMS Imagine.Lab AMESim** from LMS International
- ◆ **MapleSim** from MapleSoft, Canada.
- ◆ **MathModelica** from Wolfram Research, Sweden.
- ◆ **SimulationX** from ITI GmbH, Dresden, Germany.

### Free Modelica Simulation Environments (alphabetical list)

- ◆ **JModelica.org** from Lund University and Modelon AB, Sweden (under development; subset of Modelica is available).
- ◆ **OpenModelica** from Linköping University, Sweden (under development; subset of Modelica is available)

An up-to-date list of Modelica tools is available from: [www.modelica.org/tools](http://www.modelica.org/tools)



# Questions

