

# SysML for Modeling Co-Simulation Orchestration over FMI, INTO-CPS Approach

Alessandra Bagnato  
**SOFTEAM R&D**

Workshop Challenges and New Approaches for Dependable and  
Cyber-Physical System Engineering (De-CPS 2016)  
**Pisa, 17<sup>th</sup> June 2016**

**INTO-CPS** 

[www.into-cps.au.dk](http://www.into-cps.au.dk)



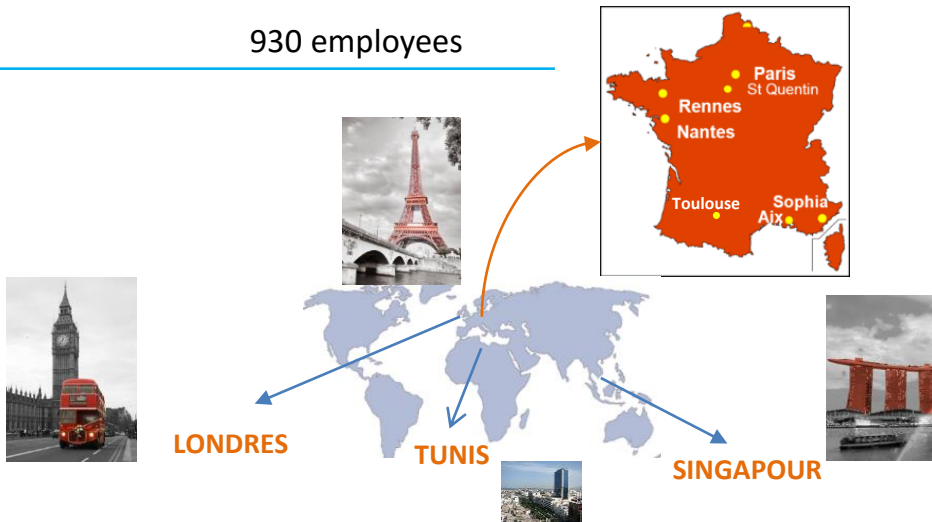
**21st International Conference on Reliable  
Software Technologies  
Ada-Europe 2016**



**SOFTEAM Cadextan**

# SOFTEAM Cadextan / IDENTITY

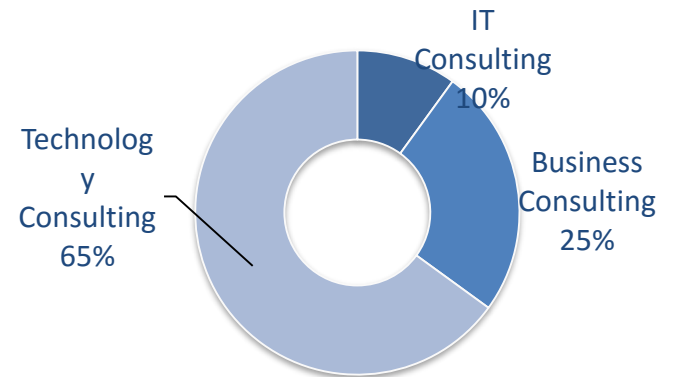
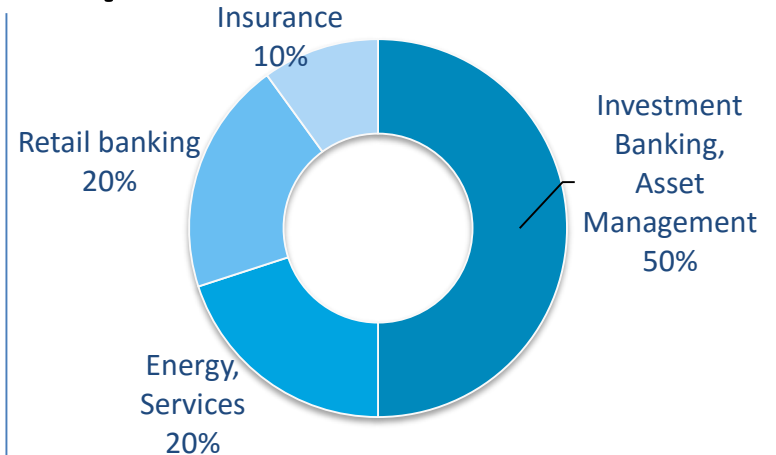
Capital	20 M€
Sales	90 M€
Staff	930 employees



Senior engineers  
650

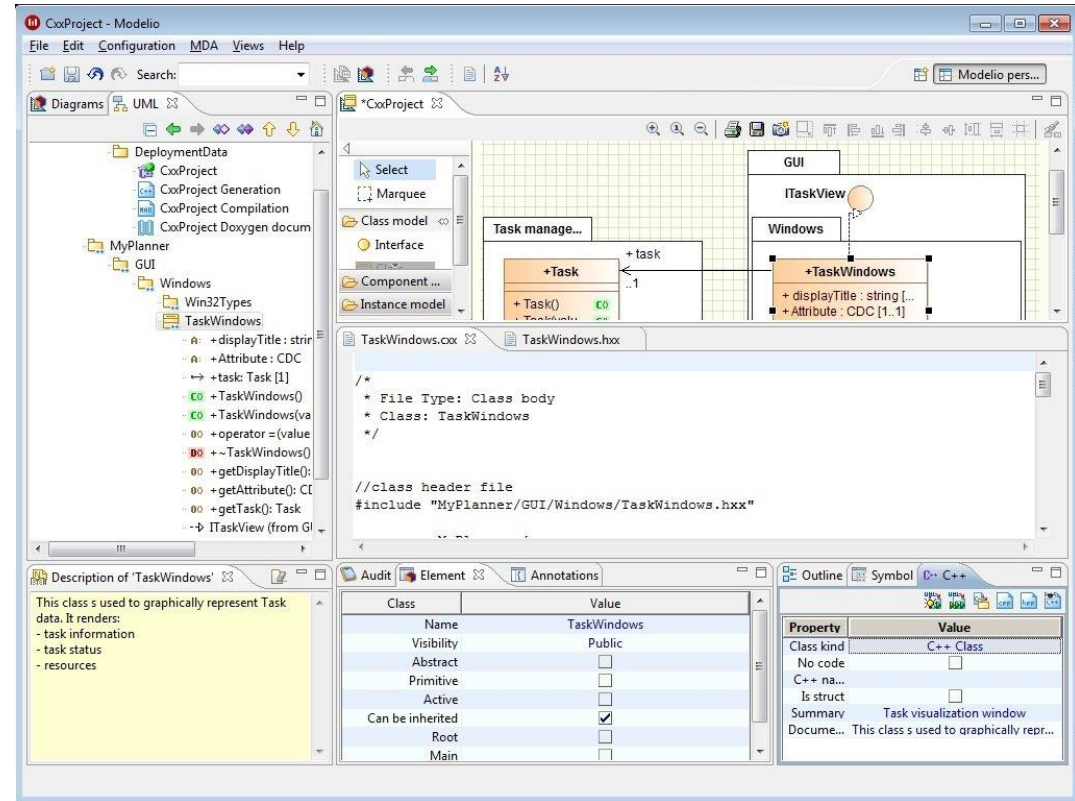
Architects  
250

.Net, JEE, C++, web, BI  
Architects, urbanists, experts, project managers, project directors

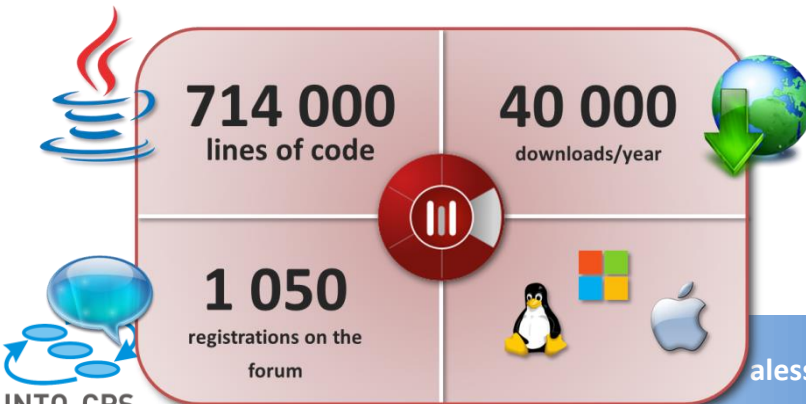


# Modelio for System Engineering

- UML editor with 25 years' history
  - SysML, MARTE, BPMN
  - Code generation
  - Documentation
  - Available under open source at [Modelio.org](http://Modelio.org)



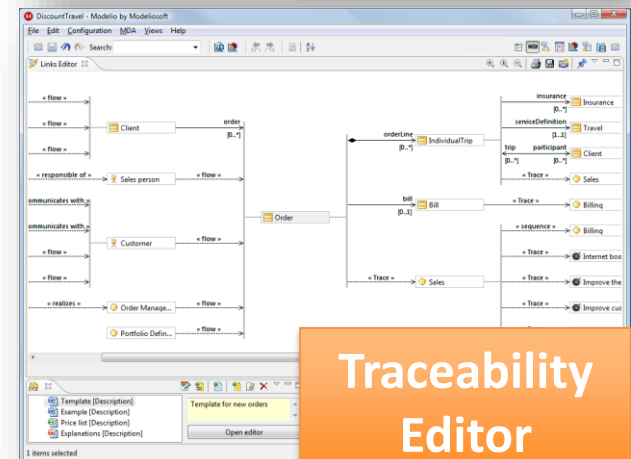
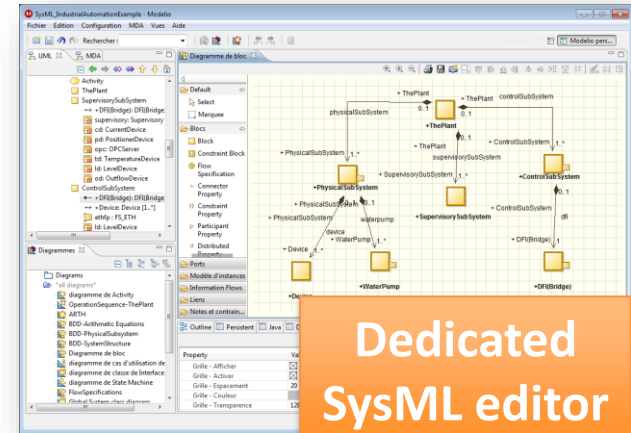
**Modelio 3**



# Modelio System Architect Solution

Dedicated to System architects modelling with SysML, UML or BPMN and carrying out Requirements based analysis

- Modelling with UML, SysML and BPMN
- Requirements Modeling
- Tabular editors
- Import/export MS Excel & Word
- Embedded Systems modelling via MARTE
- Traceability Editor
- Impact Analysis
- Document Generation
- Support for Collaborative activities (Constellation, SVN)
- Automatic diagrams creation
- Customisable, interfaces to external tools



# Cyber-Physical Challenges

## Agricultural Logistics

- Need to model control and planning/re-planning
- Models of locality and mobility
- Real-time behaviour modelling
- Domain-specific reference models



## Smarter Building Design

- Models of large-scale, open, diverse data integration
- ... coupled with models of physics
- Need to model learning behaviour
- Possible integration of models of human behaviour

# INTO-CPS: a 8 M€ H2020 Project

An integrated “tool chain” for  
comprehensive Model-Based Design  
of Cyber-Physical Systems

## INTO-CPS



Linköping University

THE UNIVERSITY *of* York



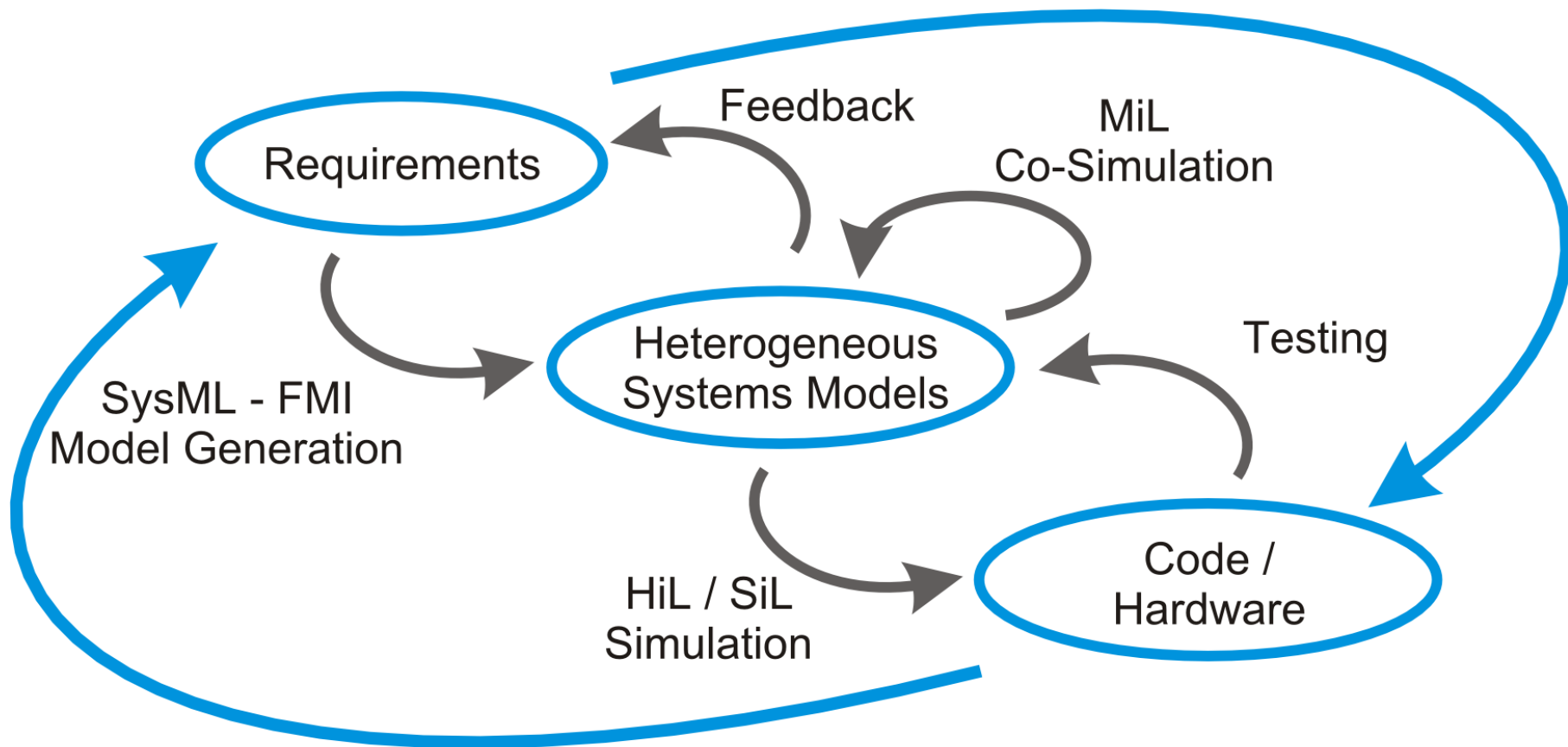


# INTO CPS Objectives

1. Build an open, well-founded tool chain for multidisciplinary model-based design of CPS that covers the full development life cycle of CPS
2. Provide a sound semantic basis for the tool chain
3. Provide practical methods in the form of guidelines and patterns that support the tool chain
4. Demonstrate in an industrial setting the effectiveness of the methods and tools in a variety of application domains.
5. Form an INTO-CPS Association to ensure that project results extend beyond the life of the project

# INTO-CPS

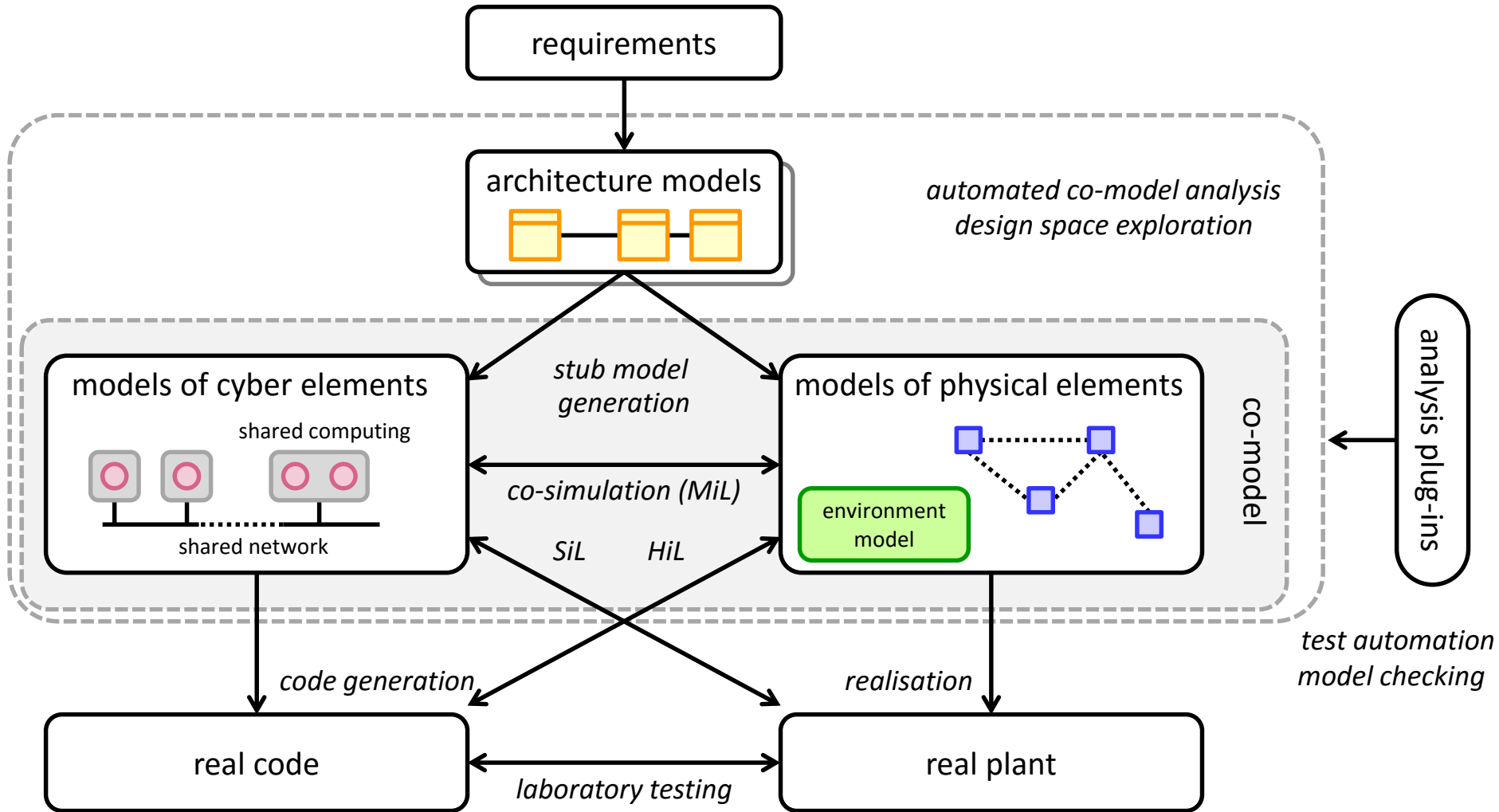
Design Space Exploration  
Test Automation



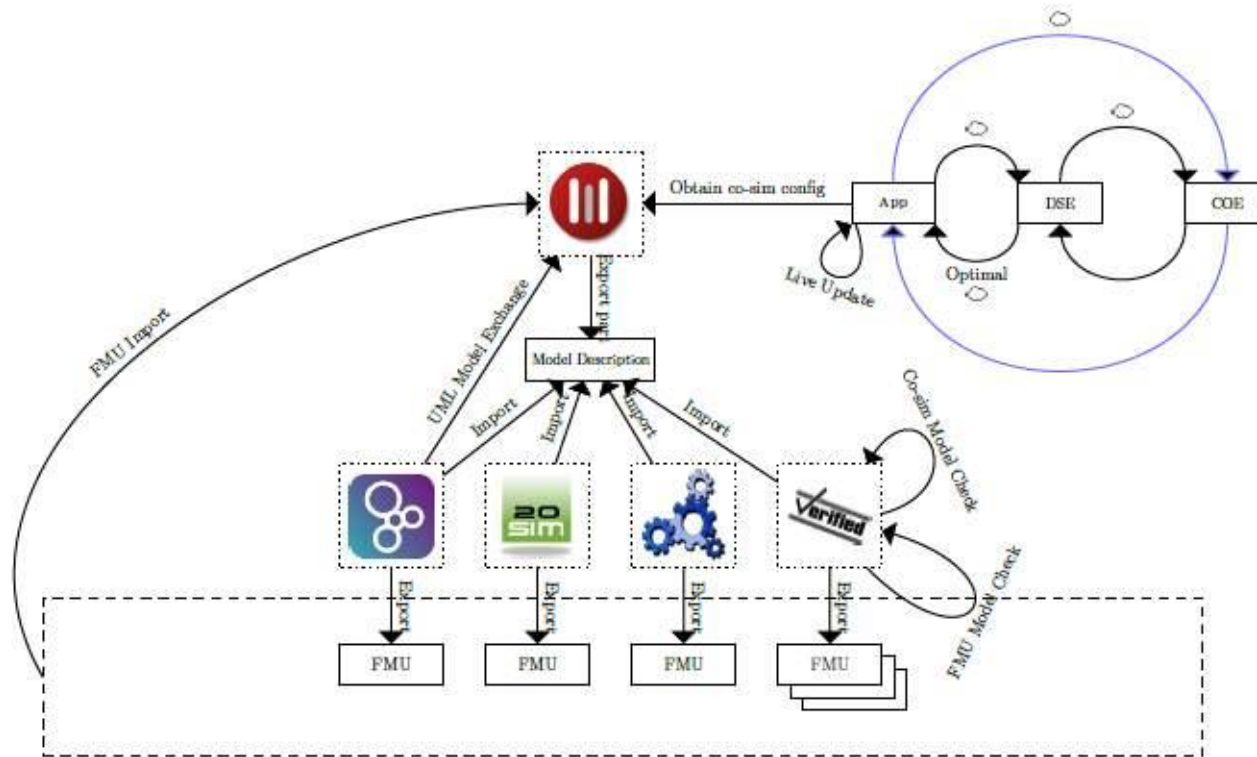
Strong Traceability  
Configuration Management



# CPS co-modelling



# Tools



**Modelio**

SysML  
modelling



**Overture**

Discrete-event  
modelling



**20-sim**

Continuous-time and physical-  
systems modelling



**OpenModelica**



**Crescendo**

Co-simulation solutions



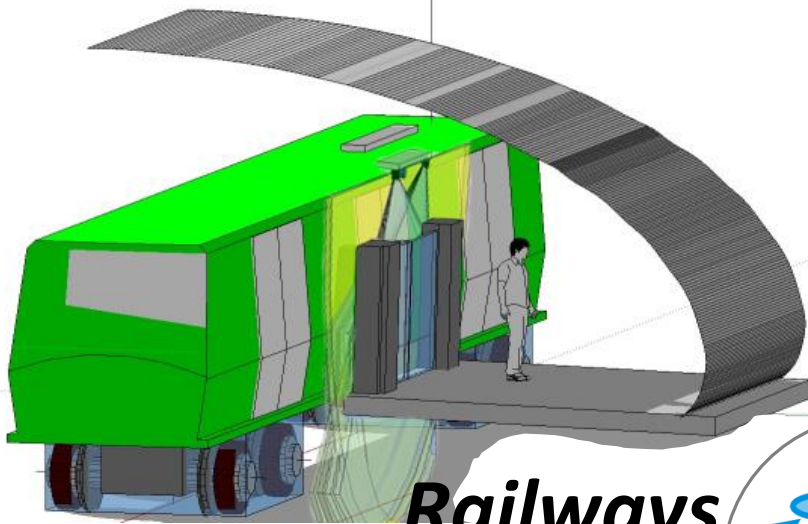
**TWT**

**TWT Engine**

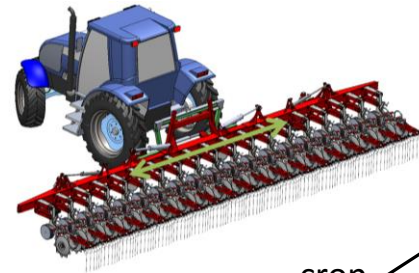


**RT-Tester**

Test automation /  
model checking

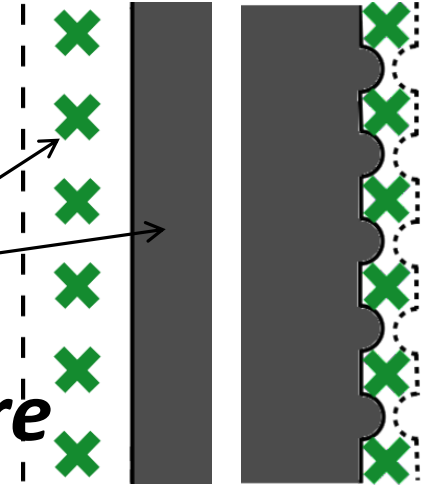


**Railways**



crop  
cleaned soil

Conventional Inter-crop



**Agriculture**

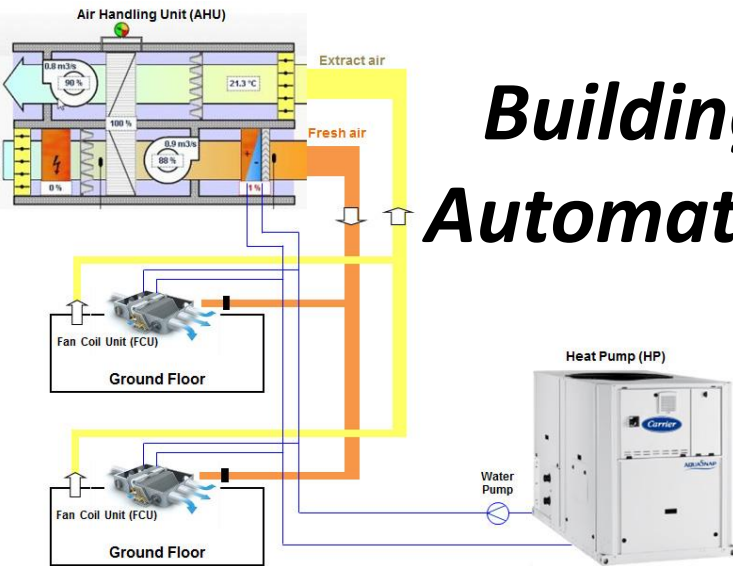


INTO-CPS

**Automotive**



**Building  
Automation**

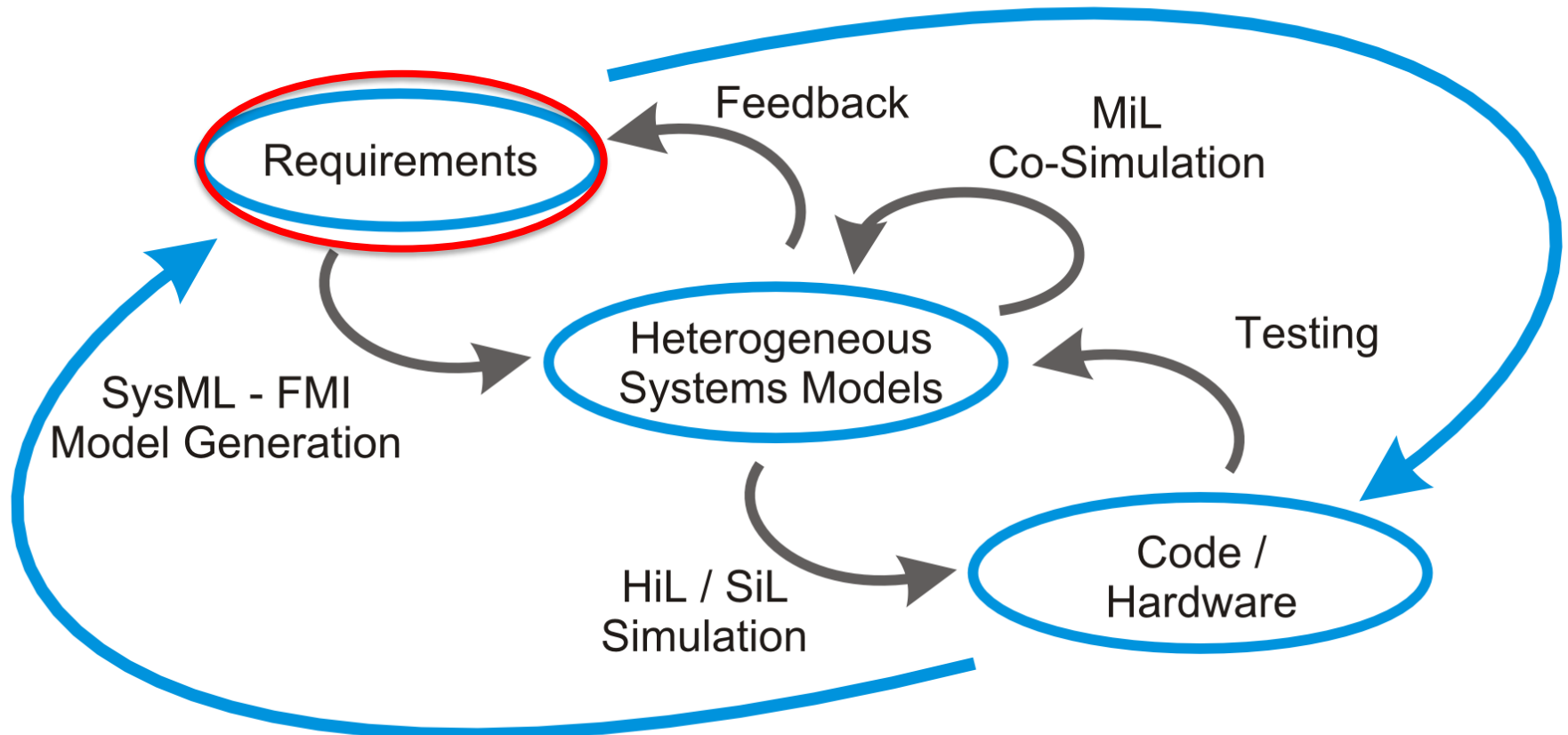


# Model-driven Design

- Modern systems are complex
- To cope with this, we can build models beforehand
  - To perform analysis (e.g. static analysis, proof, model checking, **simulation**)
  - Clarify our assumptions
  - Evaluate potential designs
  - Avoid expensive prototypes
- Different modelling paradigms for different aspects

# The Initial INTO-CPS Vision

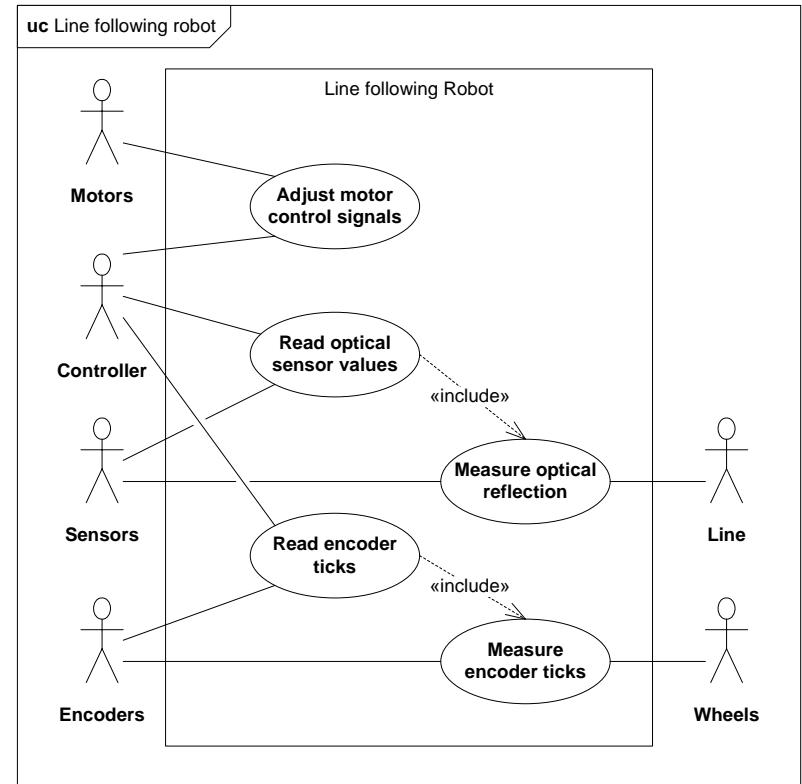
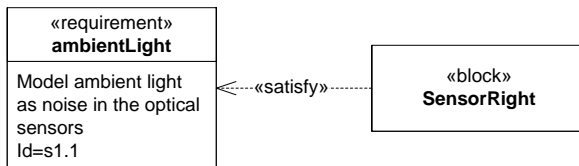
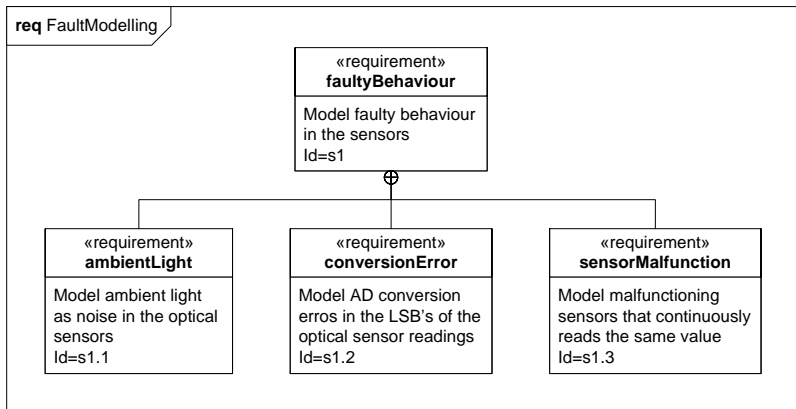
Design Space Exploration  
Test Automation



Strong Traceability  
Configuration Management

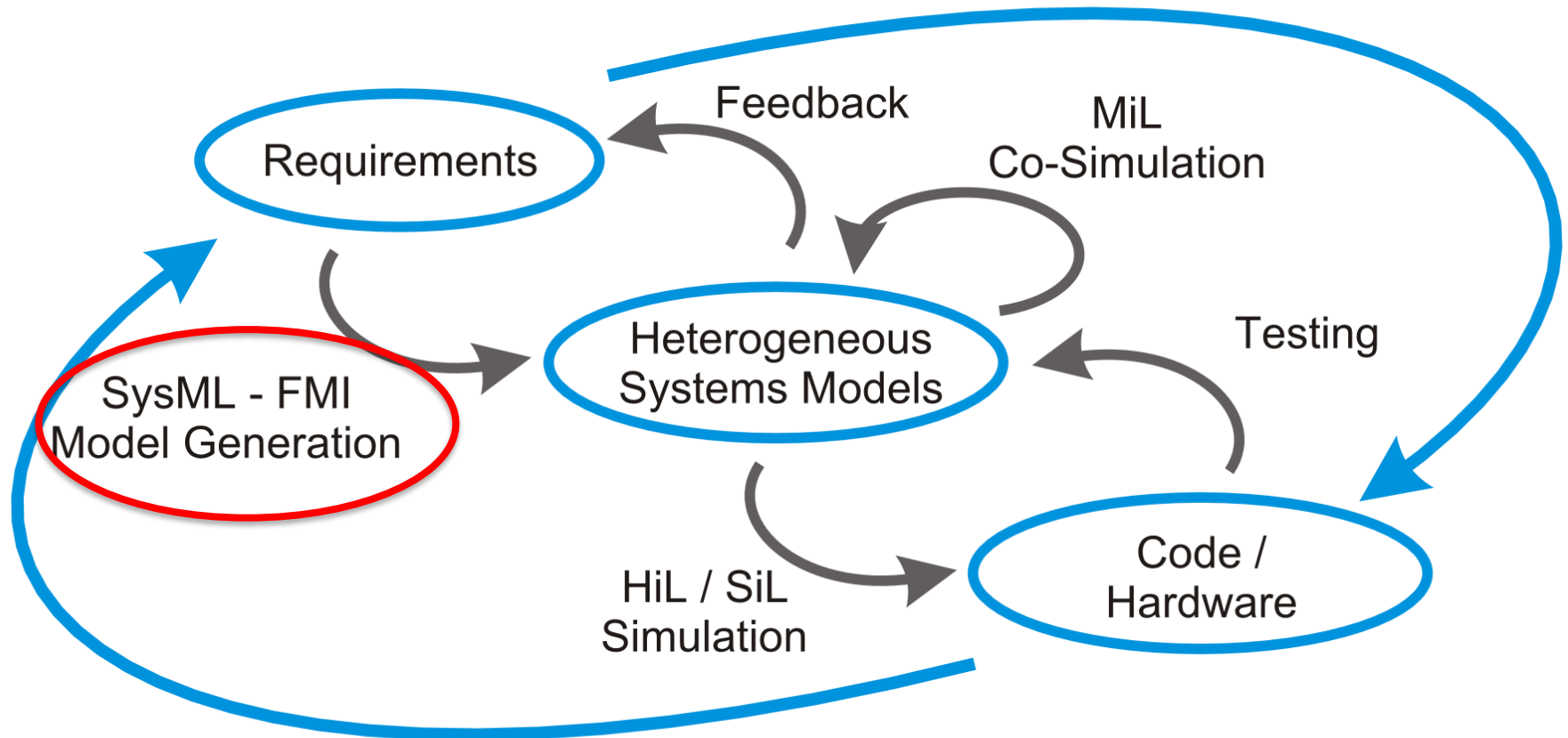
# Requirements Modelling

- SysML
  - Use Case diagrams
  - Requirements diagrams
    - Informal (link and traceable)
    - Formal (LTL, Test automation)



# The Initial INTO-CPS Vision

Design Space Exploration  
Test Automation



Strong Traceability  
Configuration Management

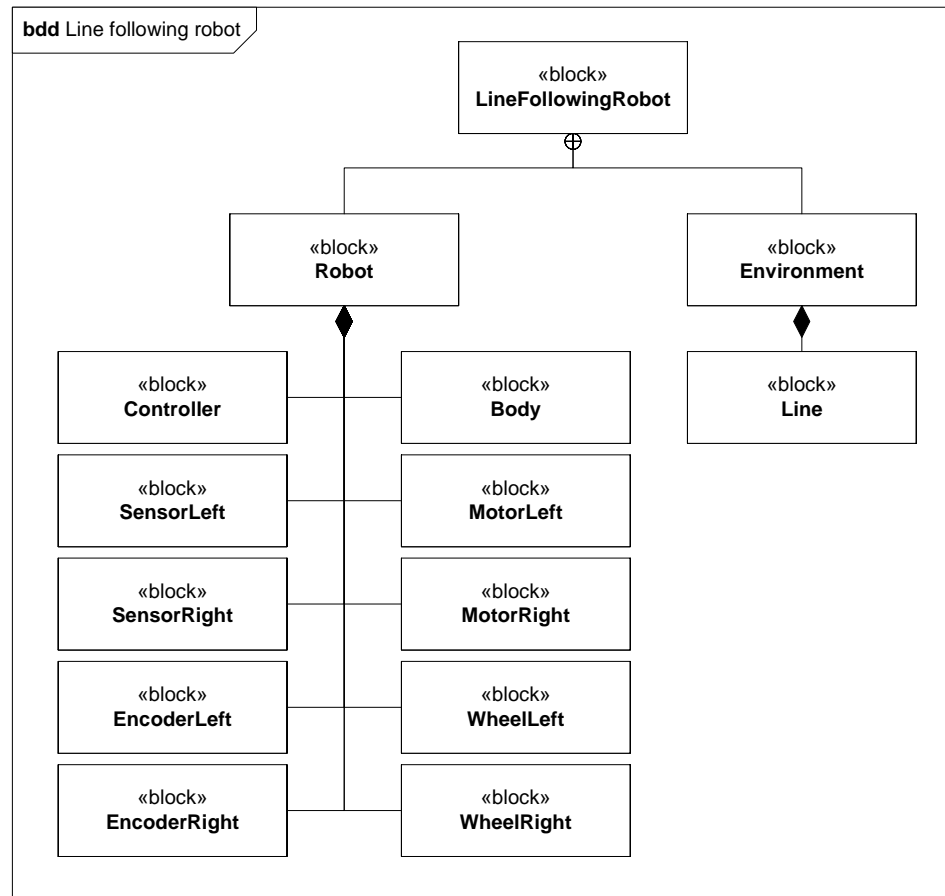


# Functional Mockup Interface (FMI)

- The Functional Mockup Interface (FMI) is a tool-independent standard to support both model exchange and co-simulation of dynamic models using a combination of XML-files and compiled C-code
- The FMI standards currently specify two types of protocols:
  - FMI for Model Exchange (import and export), and
  - FMI for Co-Simulation (master and slave).
- For FMI Model Exchange Import, the subsystem model is exported from a simulation tool in the form of an FMU archive containing the necessary FMU information (model description file, optional C source code, etc.); while in the FMI Model Exchange Export, the subsystem model is imported into the simulation system for system simulation.

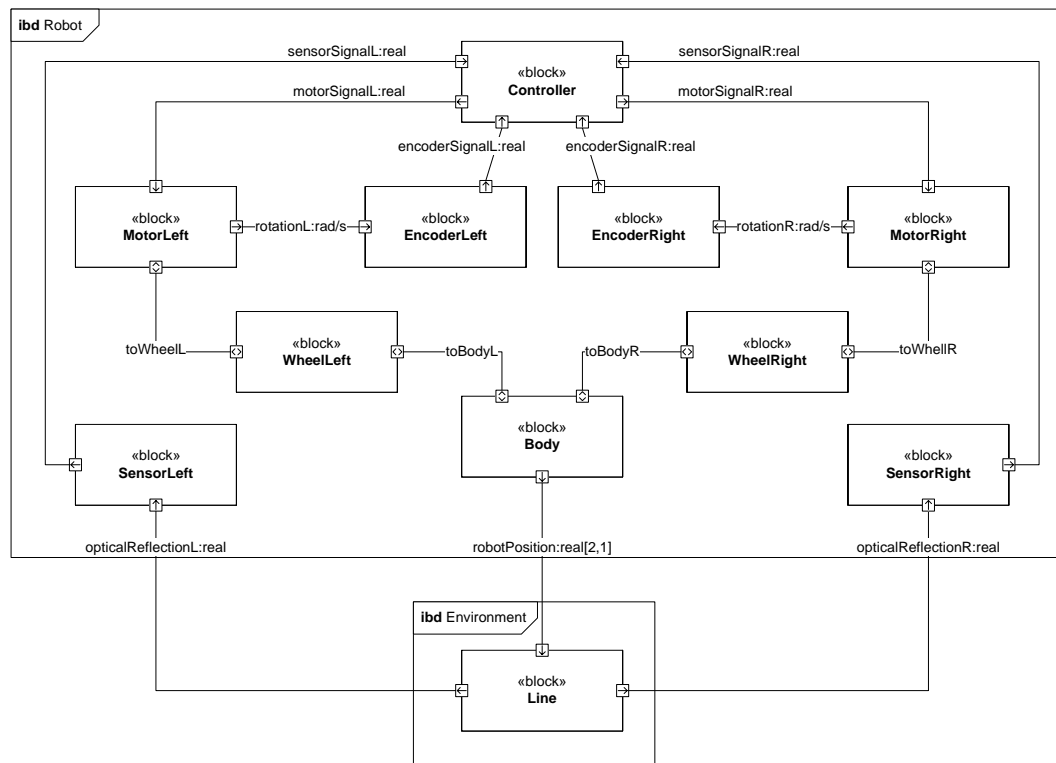
# System Decomposition

- Block Definition Diagram (top level)



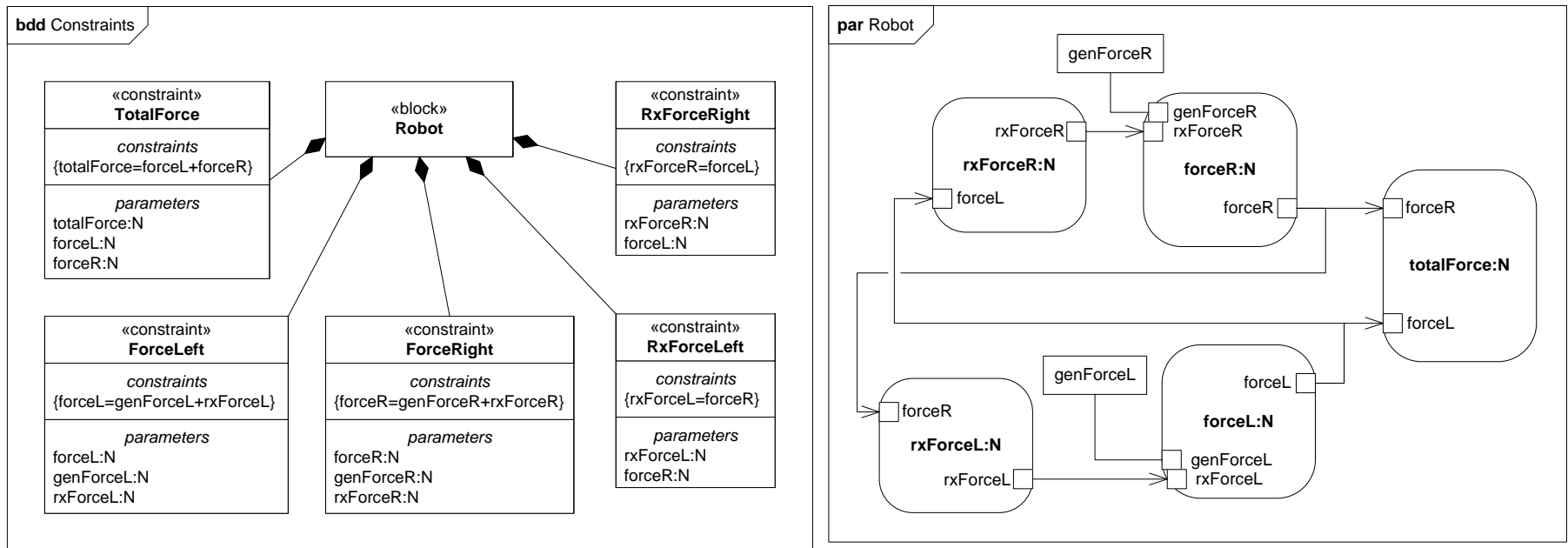
# System Interface Modelling

- Internal Block Diagram
  - Divide into CT/DE constituent models/systems/components
  - Define interfaces between different components

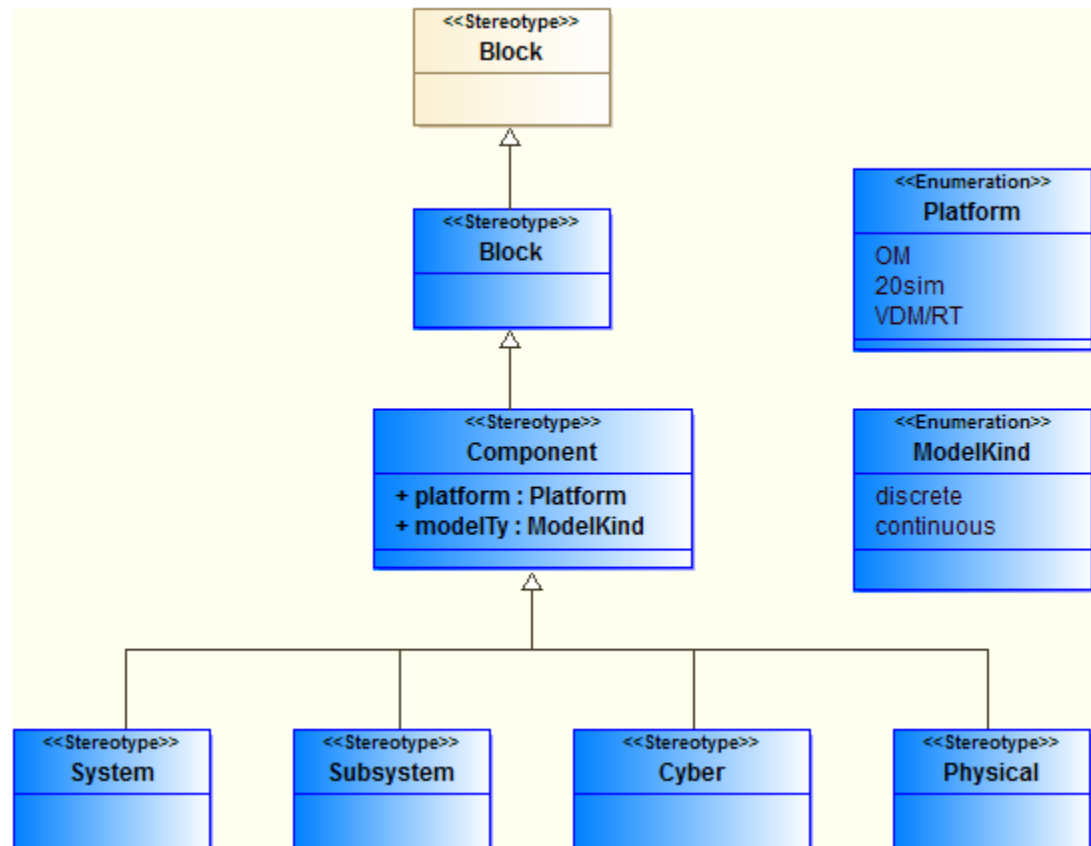
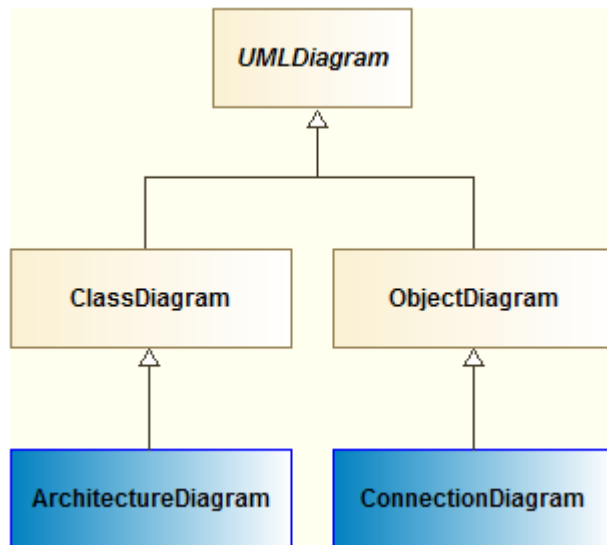


# System Behaviour

- Parametric Diagram
  - Define continuous behaviour of CT components
- State Machines (DE models generated for tests)
  - Define discrete behaviour of DE components



# INTO-CPS Diagrams and INTO-CPS Blocks



# Modelio for INTO-CPS (1/3)

The screenshot displays the Modelio 3.4 software interface. The main window is titled "Project1 - Modelio 3.4" and contains several panes:

- Model Browser:** Shows a tree view of the project structure. The "tankcontroller" component is expanded, showing its properties: "maxlevel: Real = calculated", "minlevel: Real = calculated", "input: Input", and "output: Output".
- Diagram editor:** Displays an architecture diagram for the "tankcontroller" component. The component is represented as a block with two ports: "output:Output" and "input:Input". To the right, two interfaces are defined: "Input" (with property "+ level: Real") and "Output" (with property "+ valve: Bool").
- Properties Panel:** Located at the bottom, it shows the properties of the selected component. The "INTO-CPS" tab is active, displaying the following table:

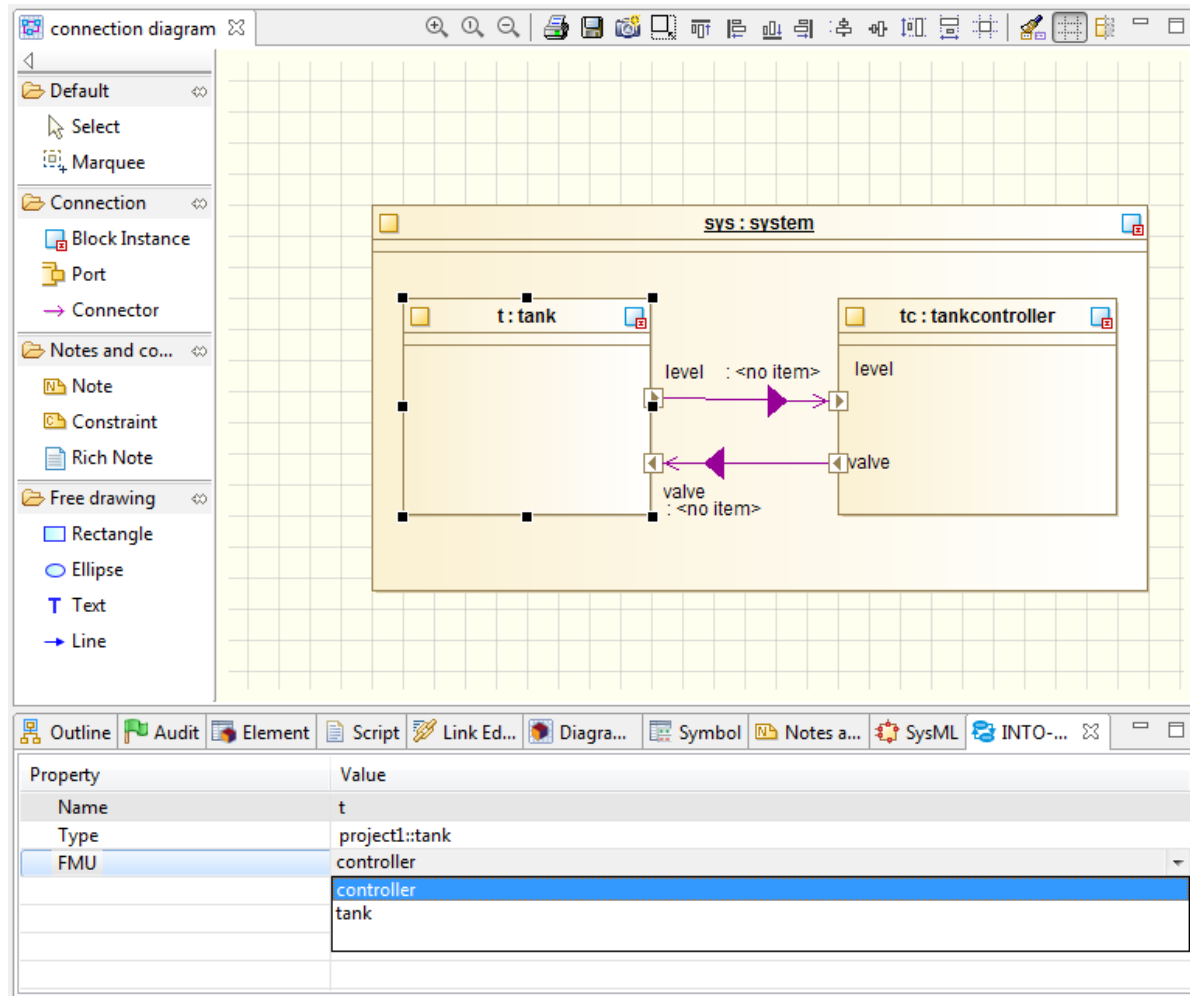
Property	Value
Name	tankcontroller
ComponentKind	Physical
ModelTy	discrete
Platform	VDMRT

A blue callout box on the left side of the image contains the text "Architecture Diagram". A red circle with three vertical bars is located in the bottom right corner of the screenshot.

Architecture  
Diagram

# Modelio for INTO-CPS (2/3)

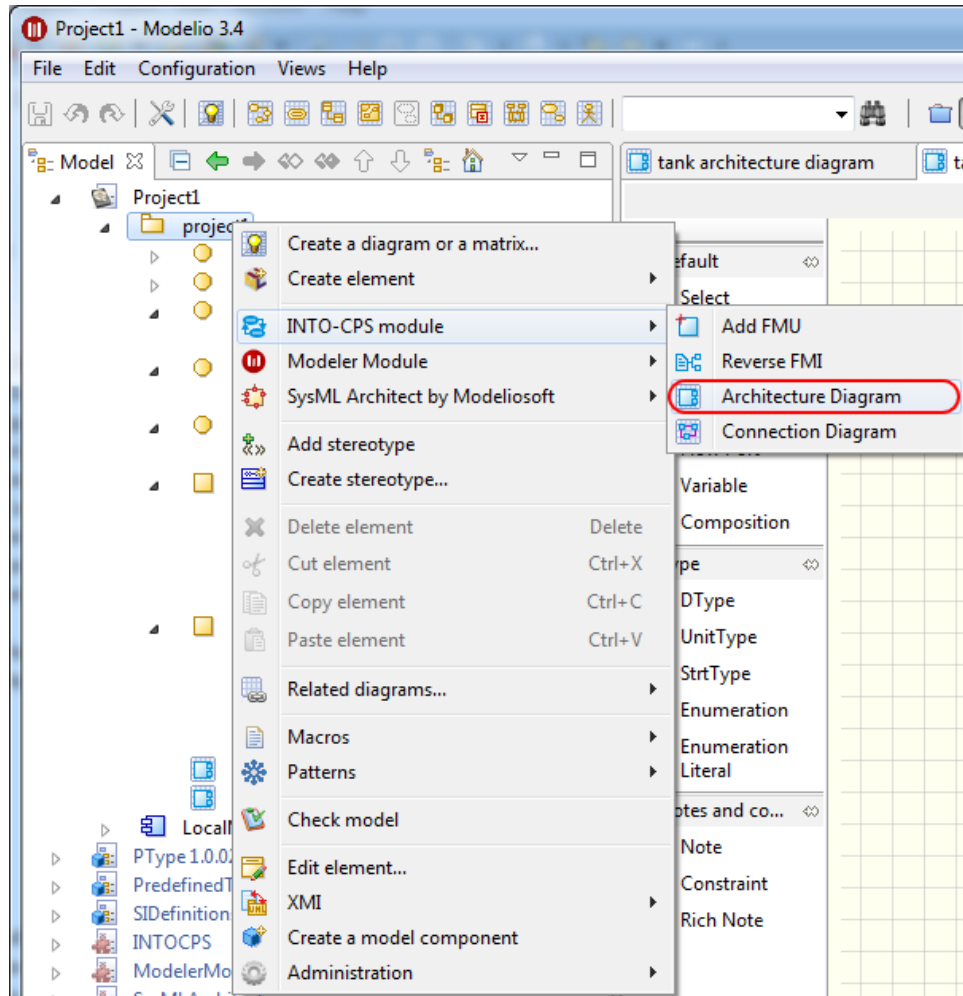
Connection Diagram





# Modelio for INTO-CPS (3/3)

Modelio  
INTO-CPS  
module



# Initial Industrial Follower Group

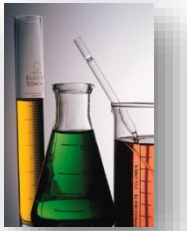


- AGCO, Denmark
- Alcatel-Lucent, Ireland
- Almende, Netherlands
- Altran, UK
- Bachmann electronic, Netherlands
- Bakker Sliedrecht Electro Industrie, Netherlands
- Carrier, France
- CeTIM, Netherlands
- Chemring TS, UK
- Compleks Innovation, Denmark
- Dredging International, Belgium
- DSTL, UK
- Goodrich, UK
- Grundfos, Denmark
- GN Resound, Denmark
- HMF, Denmark
- Huisman Equipment, Netherlands
- Irmato Industrial Solutions, Netherlands
- Jaguar Land Rover, UK
- National Institute of Informatics, Japan
- ONERA, France
- Rockwell-Collins, France
- Rolls-Royce, UK
- Seluxit, Denmark
- Siemens, Sweden
- Terma, Denmark:
- Thales, France
- UTC Aerospace Systems, UK
- West Consulting, Netherlands

# Contacts



IT Services /  
UML Tools



Research



Management



Standardization

Alessandra Bagnato, SOFTEAM  
[Alessandra.bagnato@softeam.fr](mailto:Alessandra.bagnato@softeam.fr)

**Modelio Web Site :**

<http://www.modelio.org>

<http://forge.modelio.org/projects/intocps>

**INTO-CPS Web Site** <http://into-cps.au.dk/>



@<https://twitter.com/IntoCps>



<https://github.com/into-cps>

# Thanks!

INTO-CPS 

[www.into-cps.au.dk](http://www.into-cps.au.dk)

