26th annual INCOSE international symposium

Edinburgh, UK
July 18 - 21, 2016

MINES Alès

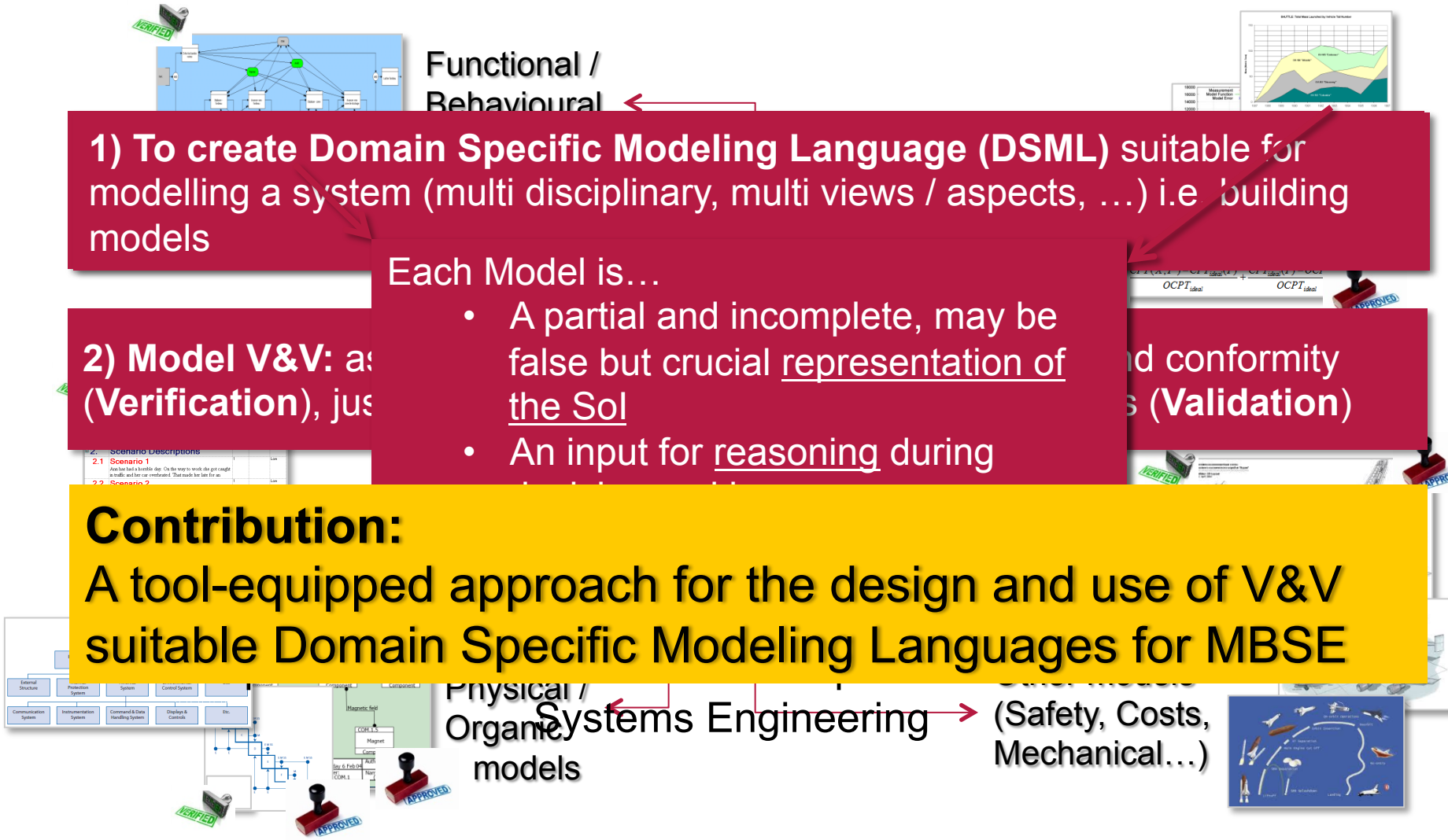
Towards V&V suitable Domain Specific Modeling Languages for MBSE: a tooled approach

Blazo Nastov[1], Vincent Chapurlat[1], Christophe Dony[2] and François Pfister[1]

[1]*LGI2P, Ecole des mines d'Alès, Nîmes, France*
{blazo.nastov, vincent.chapurlat, francois.pfister}@mines-ales.fr

[2]*LIRMM, University of Montpellier, Montpellier, France*
dony@lirmm.fr

MBSE context: research problematic



1) To create Domain Specific Modeling Language (DSML) suitable for modelling a system (multi disciplinary, multi views / aspects, …) i.e building models

Each Model is…

- A partial and incomplete, may be false but crucial representation of the SoI
- An input for reasoning during

2) Model V&V: as … conformity (**Verification**), jus… (**Validation**)

Contribution:
A tool-equipped approach for the design and use of V&V suitable Domain Specific Modeling Languages for MBSE

Agenda

- Illustrated background
- Contributions
- Case study
- Conclusion and Perspectives

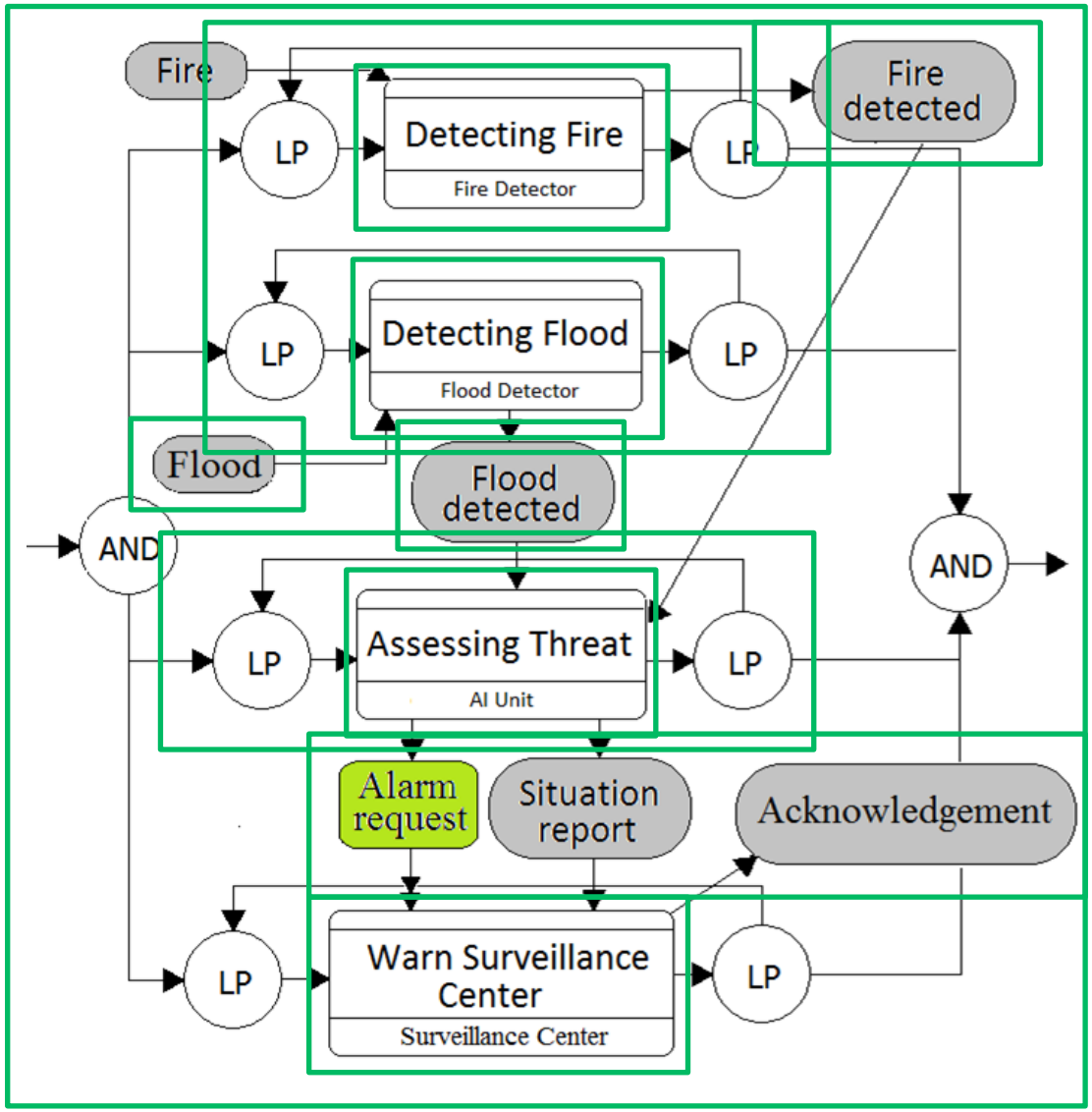
Illustrated background

Illustration: enhanced Functional Flows Block Diagram (eFFBD)

Function

Flow
(I/O, Trigger)

Construct
(parallelism, choice, loop, …)



Functional / Behavioural model (dynamics of a system)

1) To create DSML: things to consider

1) To create Domain Specific Modeling Language (DSML) suitable for modelling a system (multi disciplinary, multi views / aspects, …) i.e. building models

Semantics

Static

Structural pr

Behavioral p

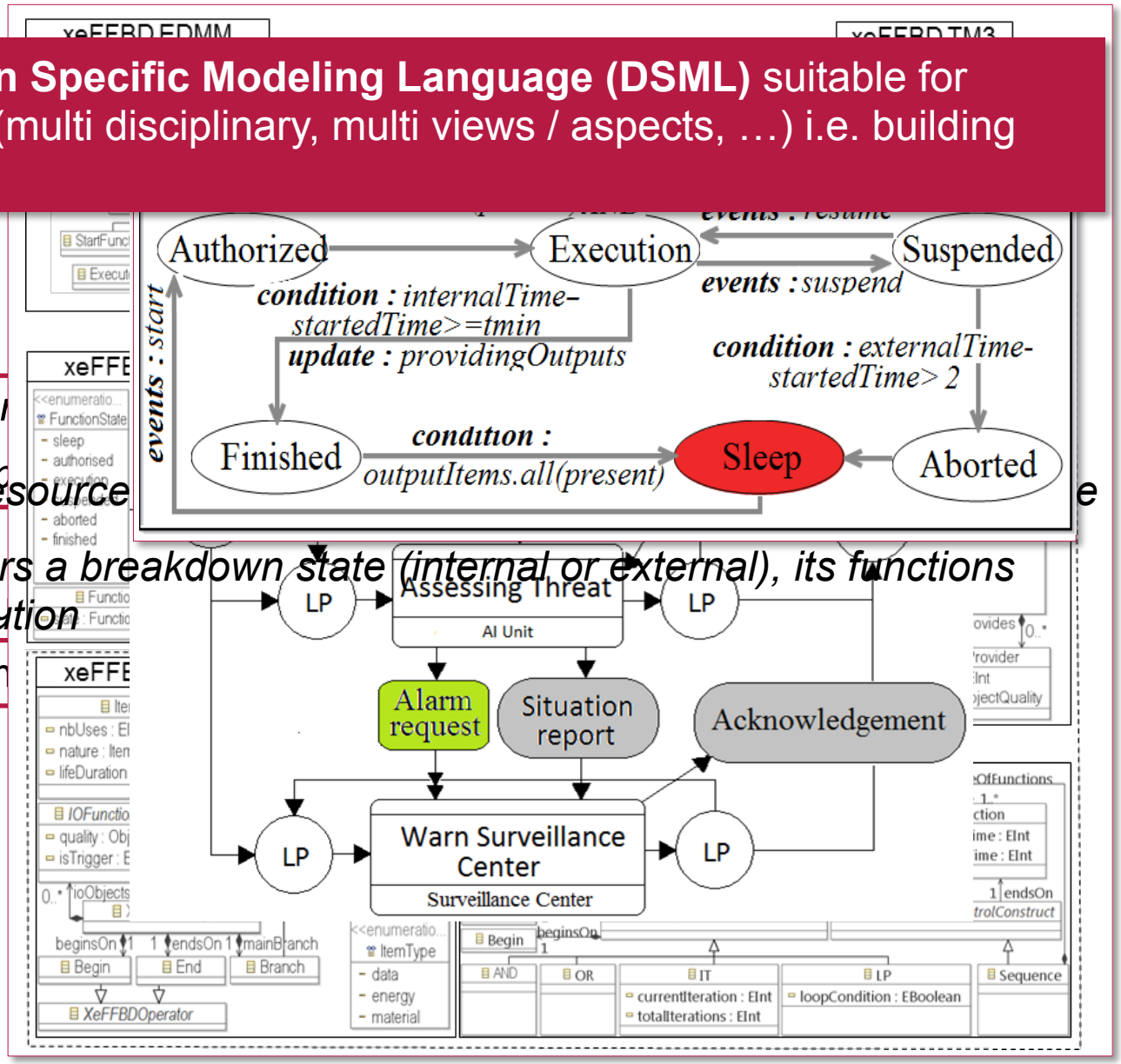
Dynamic

Action langu

Behavioral n

The quantity of a Resource … e or zero"

If a component enters a breakdown state (internal or external), its functions must suspend execution



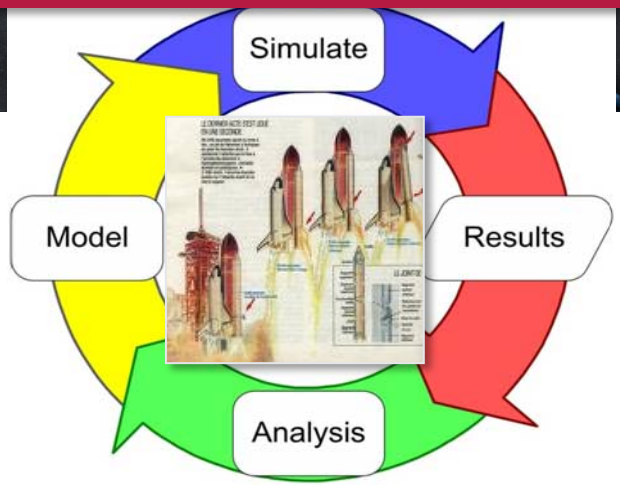
2) Model V&V

2) Model V&V: assure model's well-formedness, coherence and conformity (**Verification**), justifiability and relevance for expert's objectives (**Validation**)

Expert's model evaluation

Model simulation

Formal properties proof



Based on 3rd party formal approaches (M2M transformation)
(+) Reuse of existing approaches
(-) Information loss, relevance between the source and the target

Detailed contribution

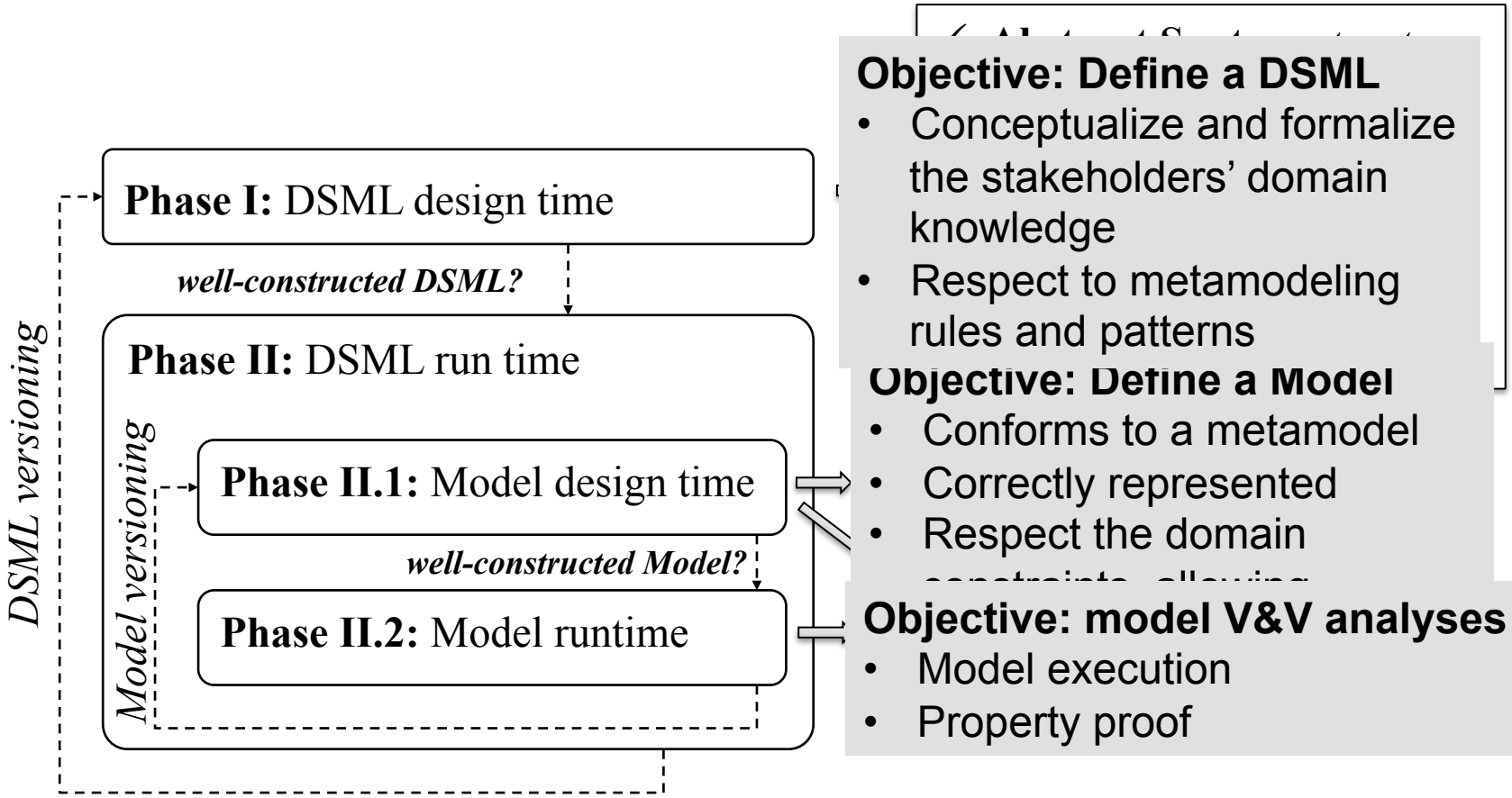
A *tool-equipped approach* for the *design of DSMLs* allowing Verification and Validation of models by using simultaneously *simulation* and *formal properties proof*

xviCore: eXecutable, Verifiable and Interoperable Core

Contributions

Conceptual contributions: life cycle

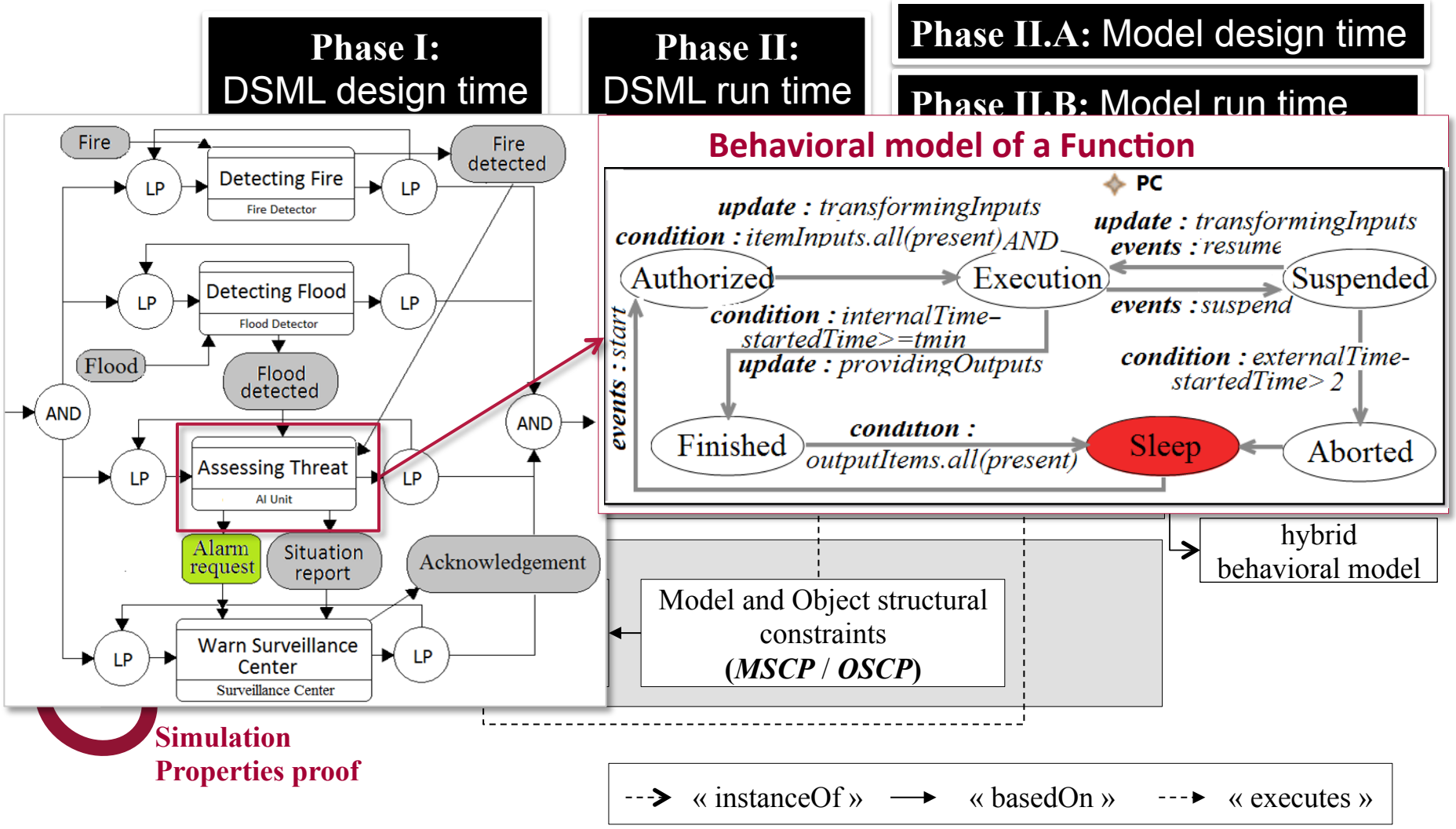
Contribution goal: to provide stakeholders with a life cycle including the design of V&V suitable DSMLs and models



Conceptual contributions: meta languages

xviCore follows previously proposed lifecycle

xviCore promotes 3 meta-languages



Conceptual contributions: simulation mechanisms

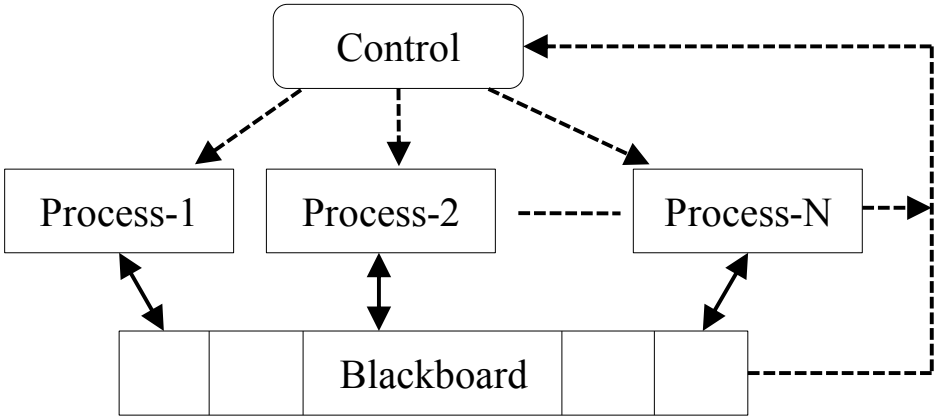
Observation

The operational semantics is defined as a set of **behavioral models** that are associated to chosen domain concepts of a DSML

Problem

How to coordinate all behavioral models from one or several DSMLs during simulation?

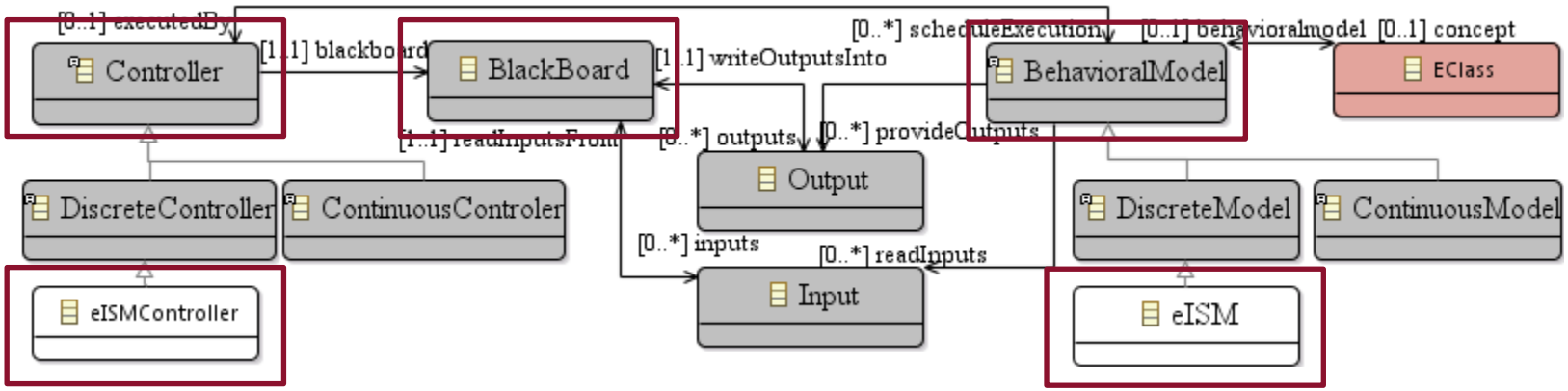
State of the art: Blackboard Design Pattern (Engelmore and Morgan, 1988)



Engelmore, R., and Morgan, T. 1988. Blackboard systems, *edited by Robert Engelmore, Tony Morgan. Addison Wesley Publishing Company*

Methodological contribution: simulation mechanisms

Objective: applying the Blackboard Design Pattern in the MBSE context



- Allows data exchange between behavioral models
- Based on a simple execution algorithm
  - Read inputs from blackboard
  - Calculate future state
  - Write outputs into blackboard
- and other original rules

Proof of SM)

(Vanderı

Oper

Sepa

Underlying structure based on the LTL (Linear Temporal Logic)

Vandermeulen, E., 1996. Machine Séquentielle Interprétée. *PhD Thesis University of Montpellier II*, (in French).

Methodological contribution: verification mechanisms

Formal properties proof mechanisms of xviCore approach

Formal specification

- syntaxes i.e. structure of the model
- dynamic semantics i.e. behavioral models)

Formal properties (structural and behavioral properties)

- using property modelling language: UPSL-SL(Chapurlat, 2013)

A **model-checker tool** (bridging specification and properties)

Reuse of existing tools :

OCL interpreter for structural properties

LTL model checker for behavioral properties

Chapurlat, V. (2013). UPSL-SE: A model verification framework for Systems Engineering. Computers in Industry, 64(5), 581-597.

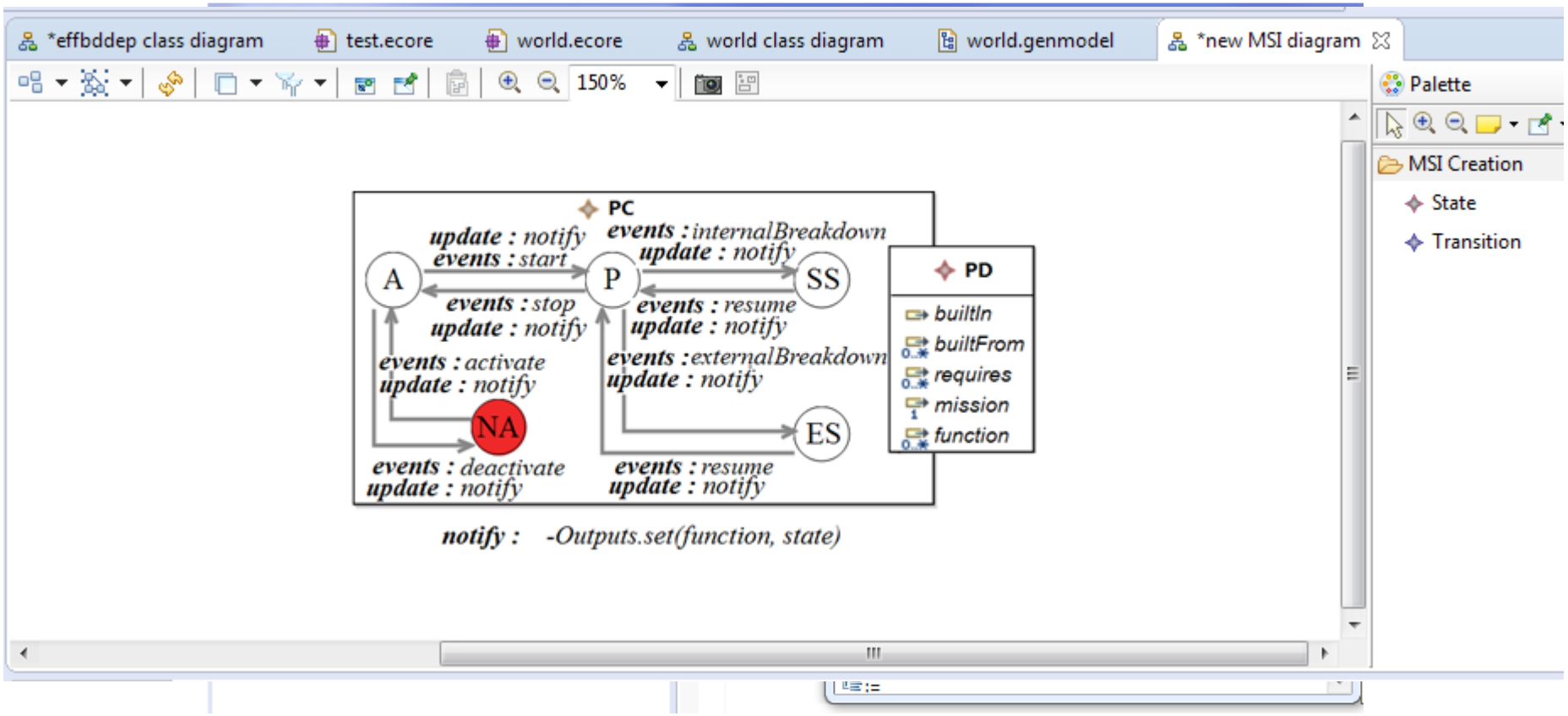
Technical contribution: current editors

DSML design

Editor for the DSML structure

Editor for the DSML behavior

Editor for the DSML constraint

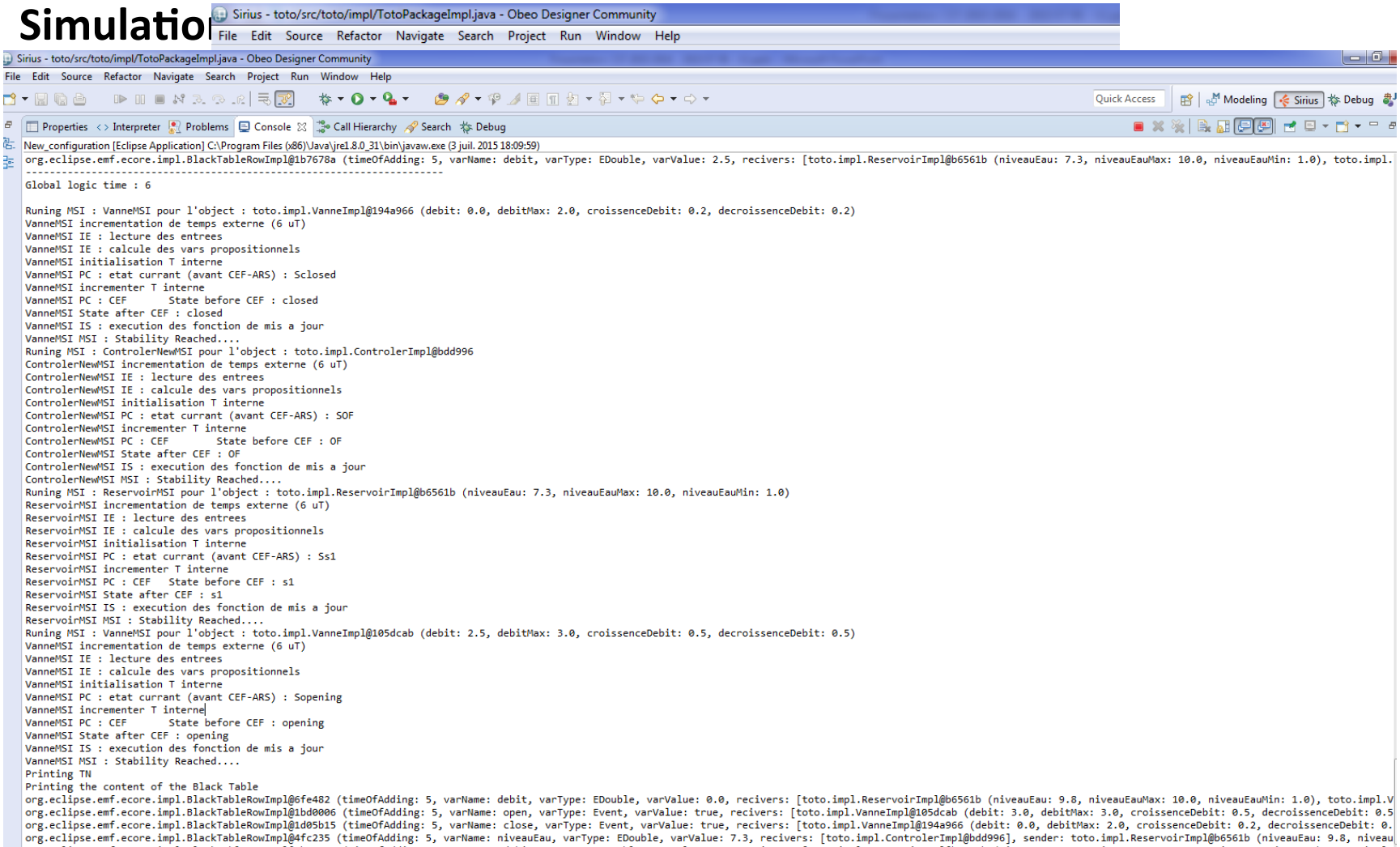


Technical contribution: extending EMF

We have extended the code generation library of EMF

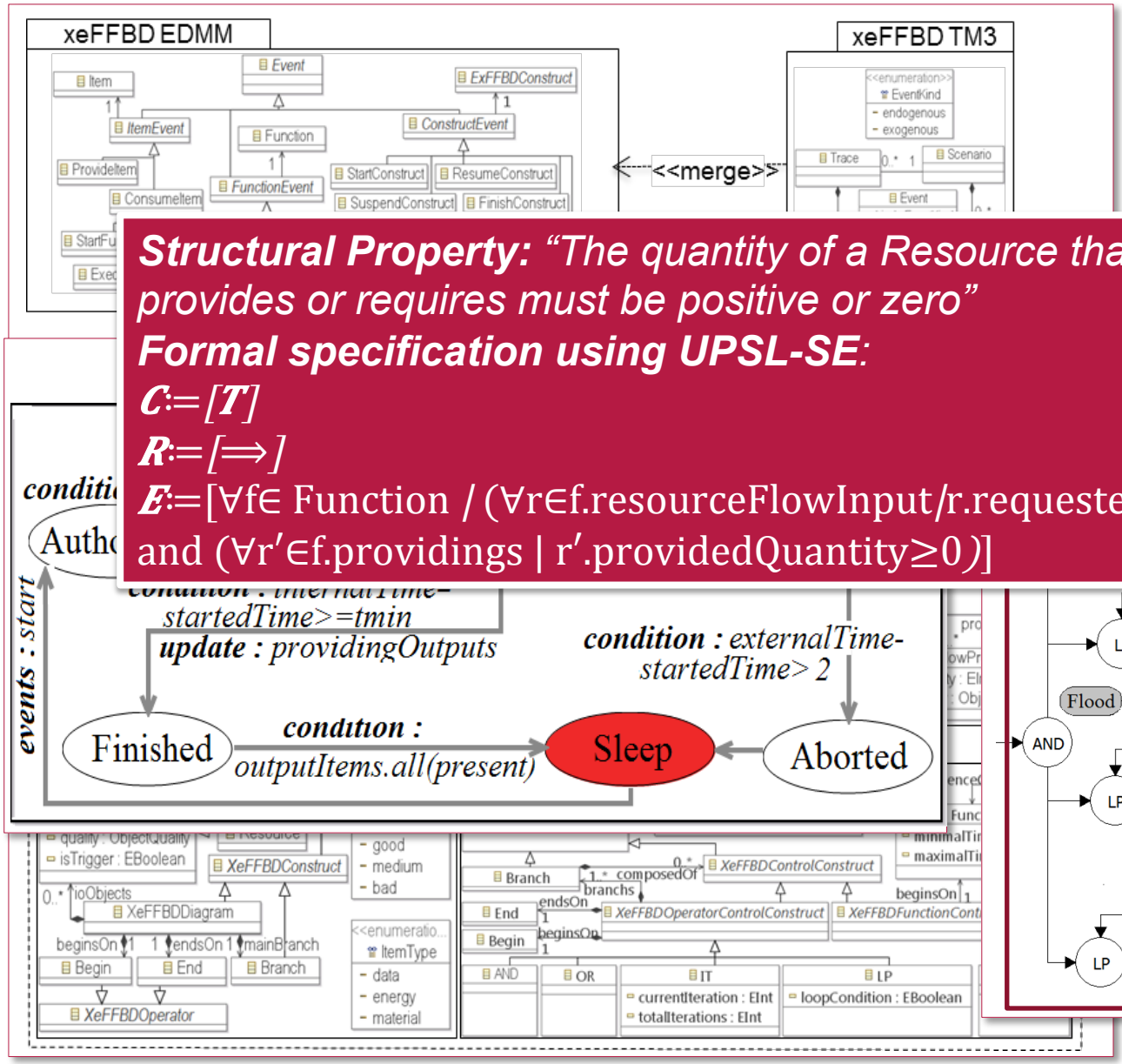
Generation of a simulation library

Simulatio



Application case: EFFBD and PBD

Reminder: enhanced Functional Flows Block Diagram (eFFBD)



Structural Property: *"The quantity of a Resource that a function provides or requires must be positive or zero"*

Formal specification using UPSL-SE:

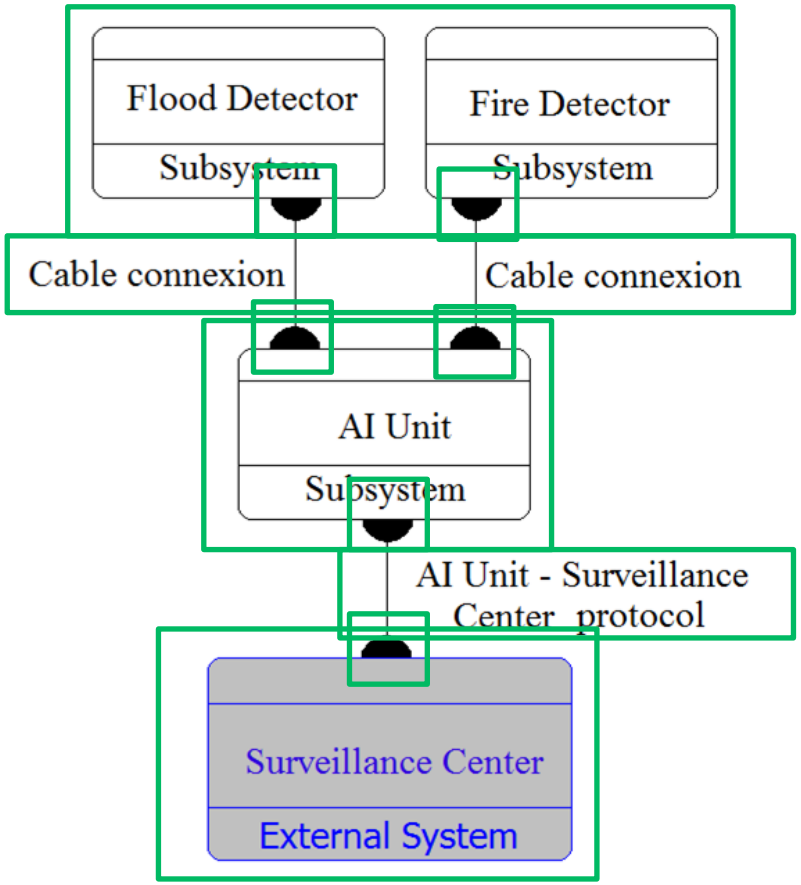
$C := [T]$

$R := [\Longrightarrow]$

$E := [\forall f \in \text{Function} \,/\, (\forall r \in f.\text{resourceFlowInput}/r.\text{requestedQuantity} \geq 0)$ and $(\forall r' \in f.\text{providings} \mid r'.\text{providedQuantity} \geq 0)]$

odel (MSP)

Sum up and application to Physical Block Diagram (PBD)



Components

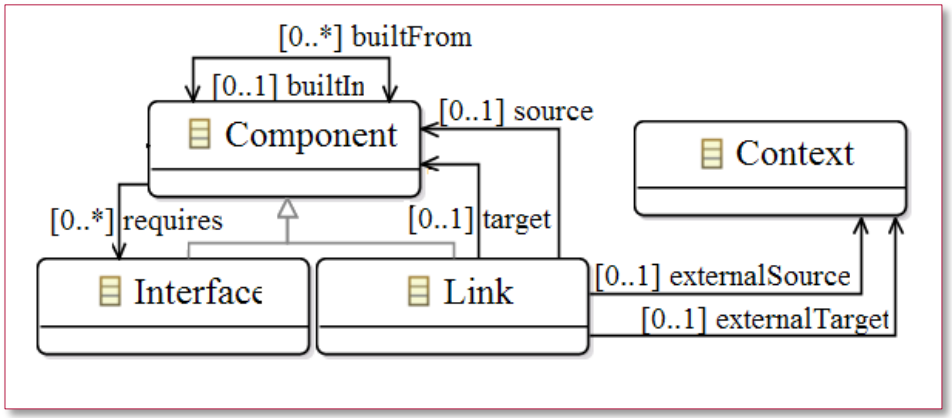
Interfaces

Links

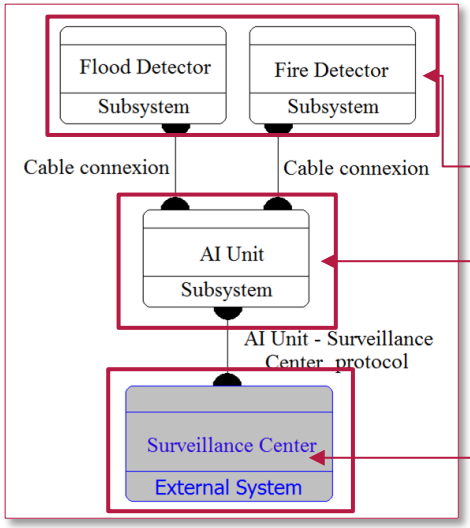
Model of architectural composition (the physical components of a system)

Example: Physical Block Diagram (PBD)

PBD abstract syntax (SP)

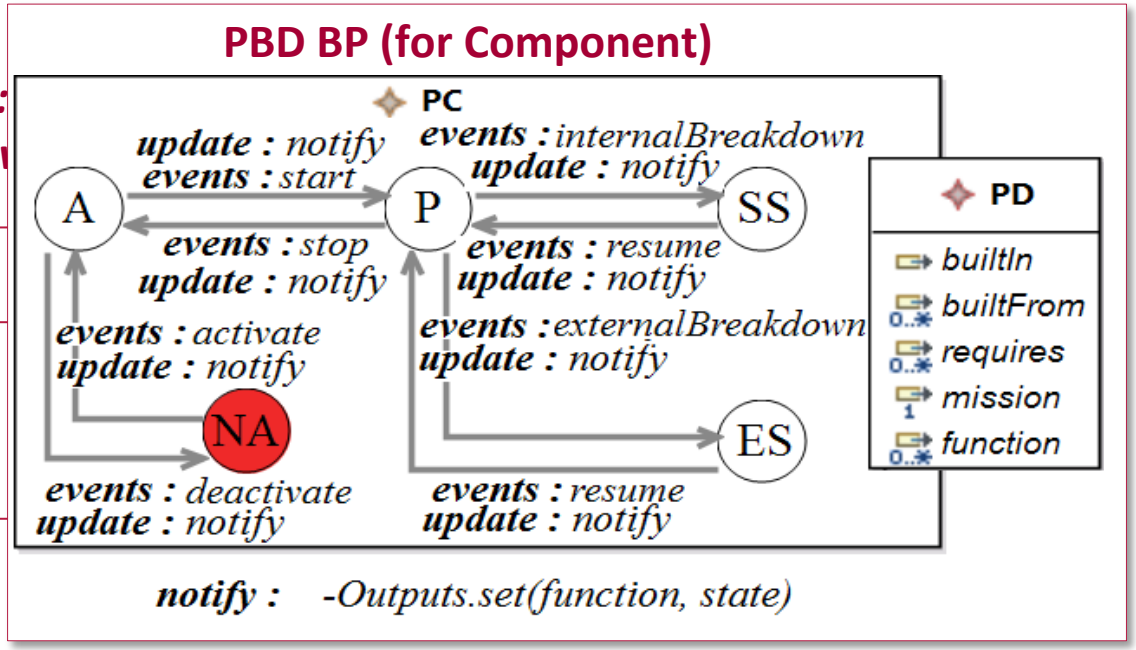


PBD model (MSP)



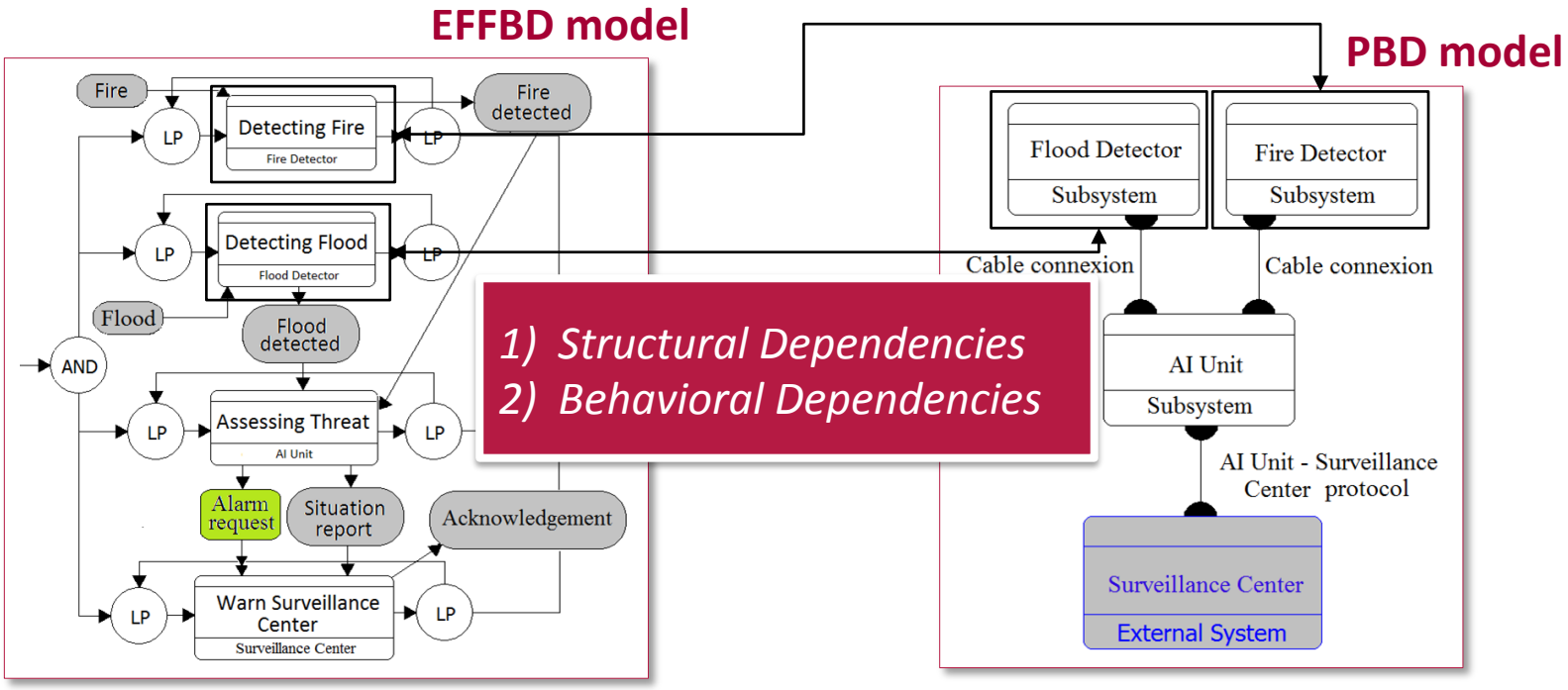
Request: behav

PBD BP (for Component)



Dependencies between eFFBD and PBD

*"Physical components of a system **perform** one or more functions and functions are **allocated** to a component"*



Dependencies between eFFBD and PBD

EFFBD DSM

DSML

Constraint:

"If a component enters a breakdown state (internal or external), its functions must suspend execution"

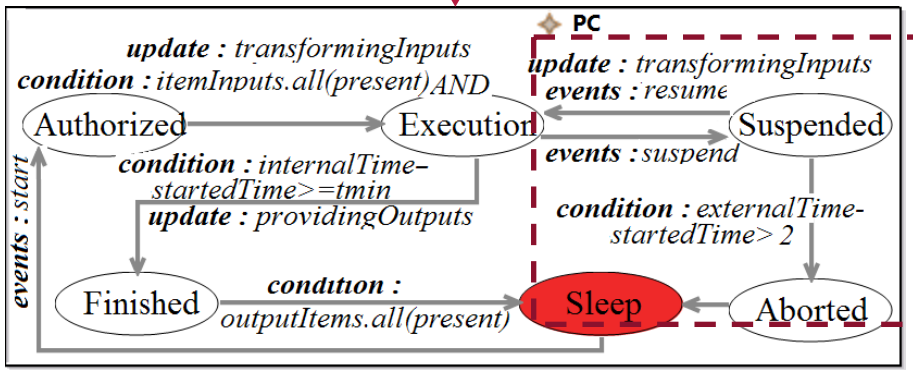
Formal specification using UPSL-SE:

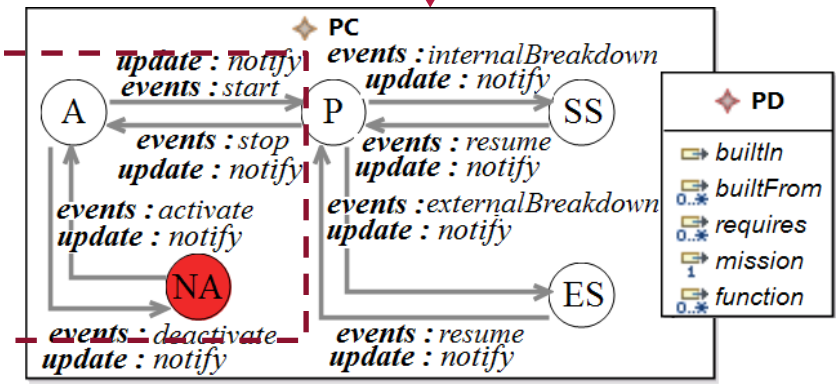
$C := [\forall c \in Component \mid c.State = SS \text{ OR } c.State = ES]$

$R := [\Longrightarrow]$

$E := [o(\forall f \in c.performs \mid f.State \neq Execution)]$

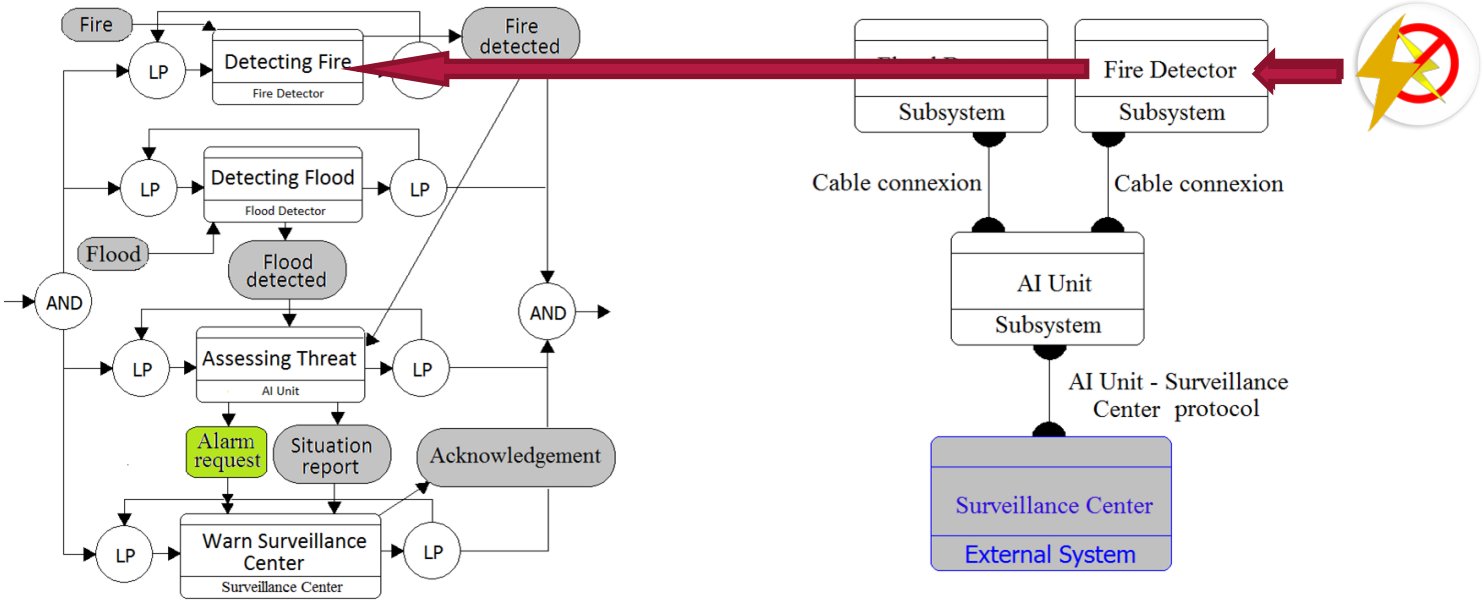
Behavioral dependencies

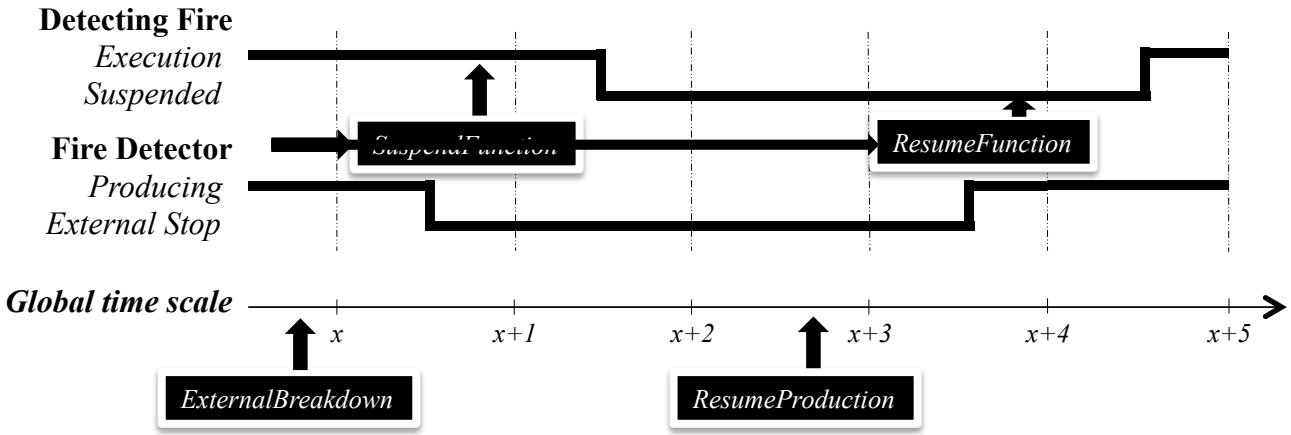




notify : -Outputs.set(function, state)

Simulation of eFFBD and PBD





Conclusion and Perspectives

Research focus

- Modelling Complex Systems

Contributions

- A lifecycle process for modeling, verification and validation
- An approach for the design of DSMLs and models
- Simulation and Formal verification of properties

Tool support

Application case: eFFBD and PBD

Future works

- Continuous and Hybrid behavioral models

Thank you
for your attention

Questions?