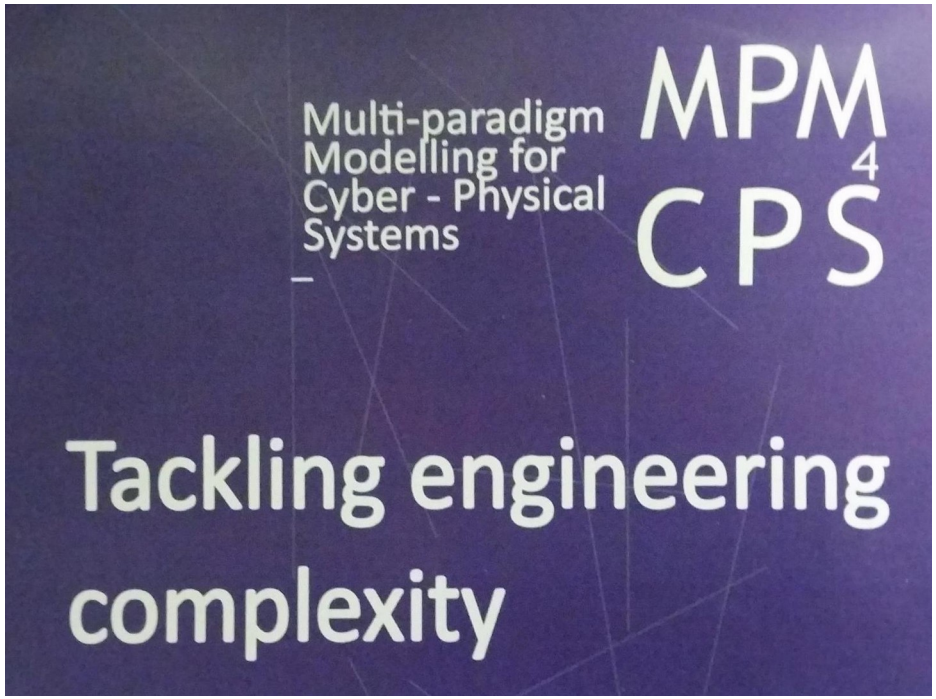
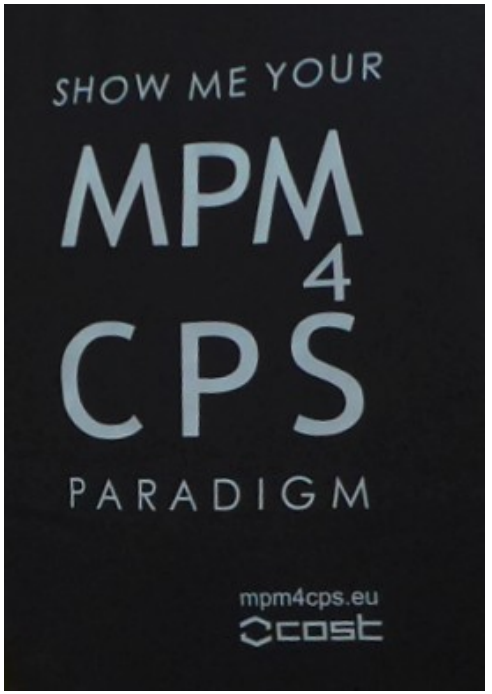
MSDL Research Day

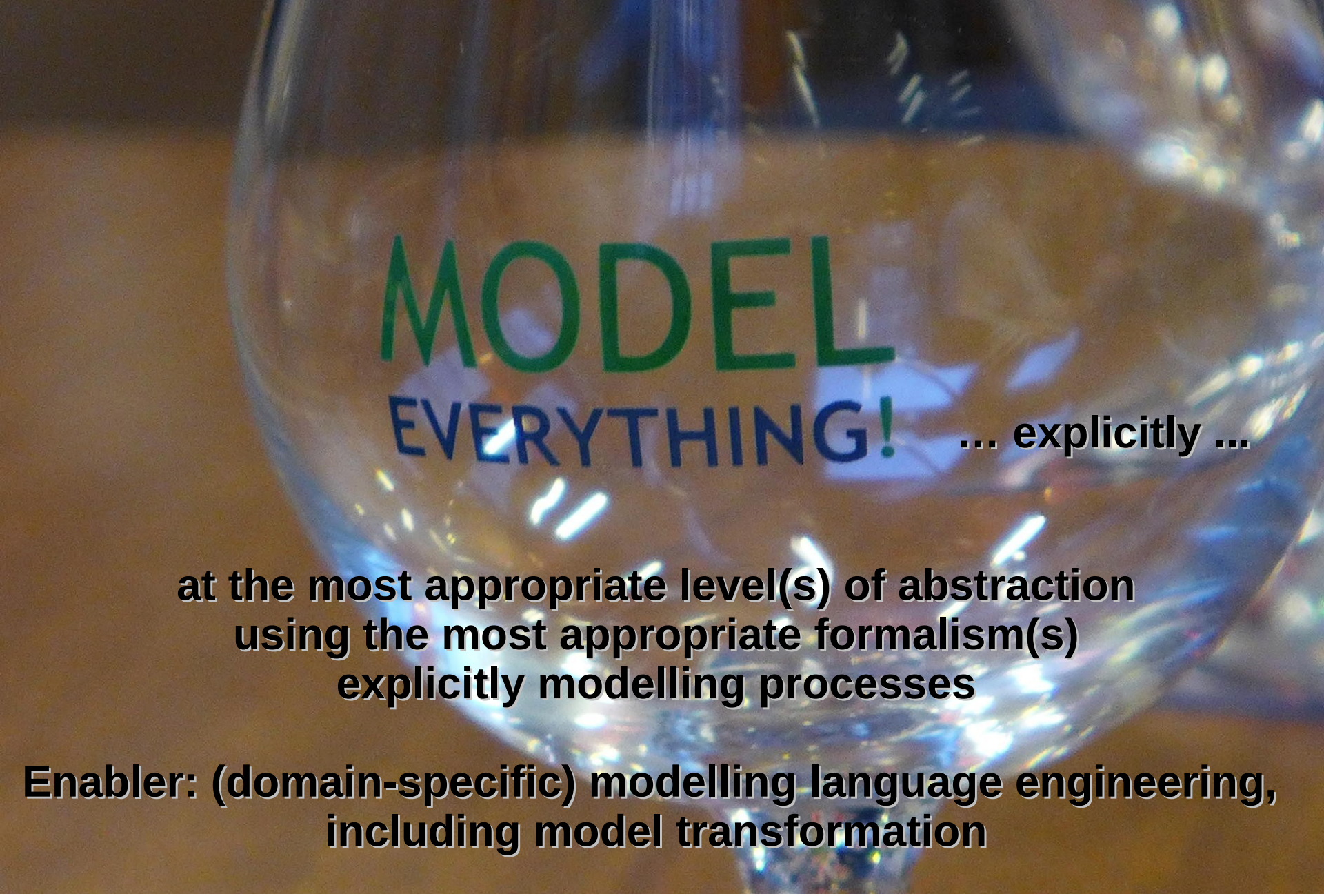




Monday 22 October 2018

Middelheim Campus M.A.301
Antwerp

Universiteit Antwerpen | Ansymo Antwerp Systems & Software Modelling University of Antwerp | Nexor Cyber-Physical Systems University of Antwerp | FLANDERS MAKE MANUFACTURING INNOVATION NETWORK | McGill | MPM4CPS | cost



… explicitly …

at the most appropriate level(s) of abstraction
using the most appropriate formalism(s)
explicitly modelling processes

Enabler: (domain-specific) modelling language engineering, including model transformation

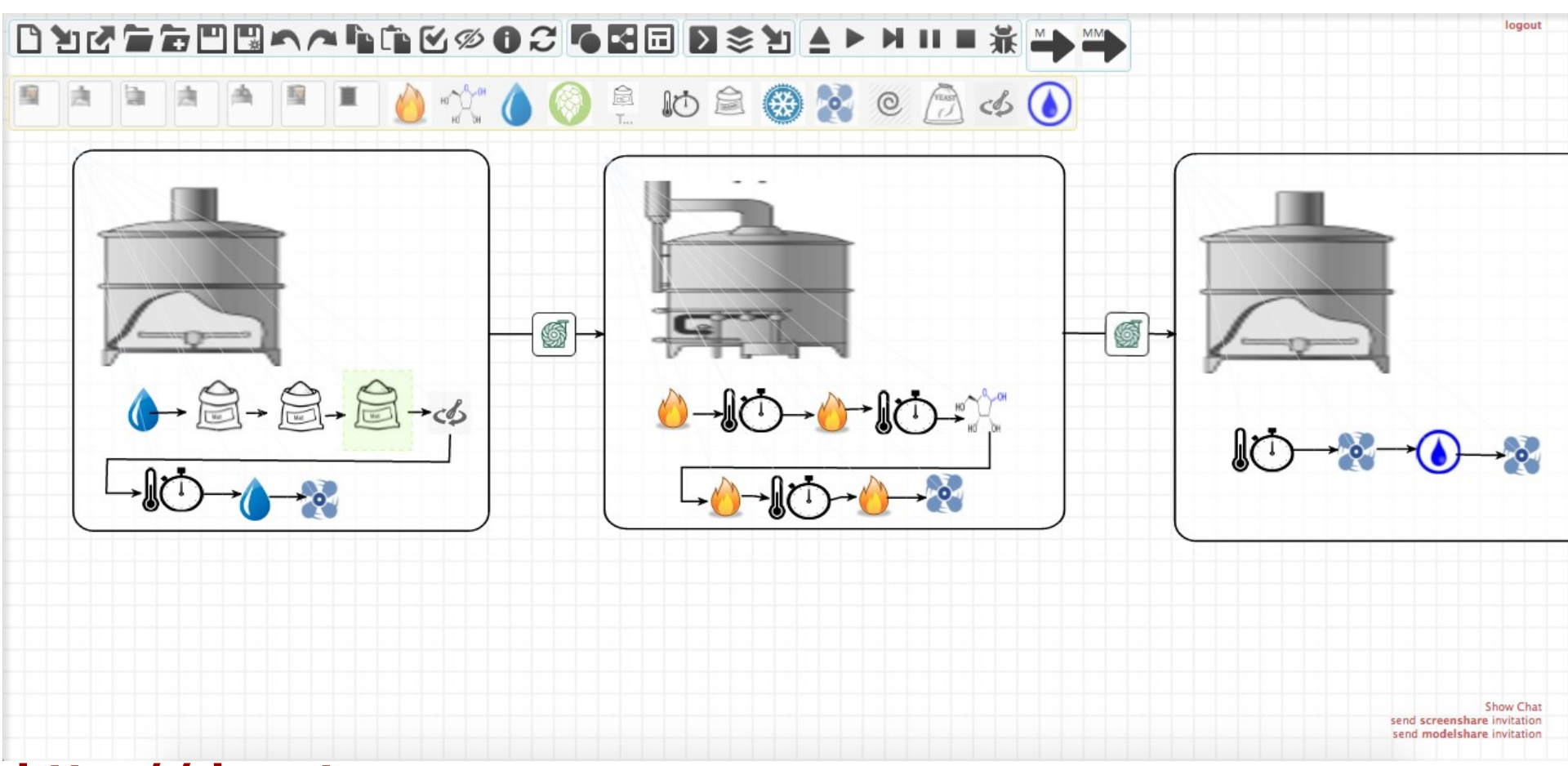
Pieter J. Mosterman and Hans Vangheluwe. Computer Automated Multi-Paradigm Modeling: An Introduction. Simulation: Transactions of the Society for Modeling and Simulation International , 80(9):433- 450, September 2004. Special Issue: Grand Challenges for Modeling and Simulation.

DSM TP 2014
Theory and Practice

5th International Summer School
on Domain Specific Modeling

Antwerp, Belgium
25 - 29 August

Thomas Kühne



http://dsm-tp.org

Eat Your Own Dogfood!

Goals of today:

1. Get to know each other's work

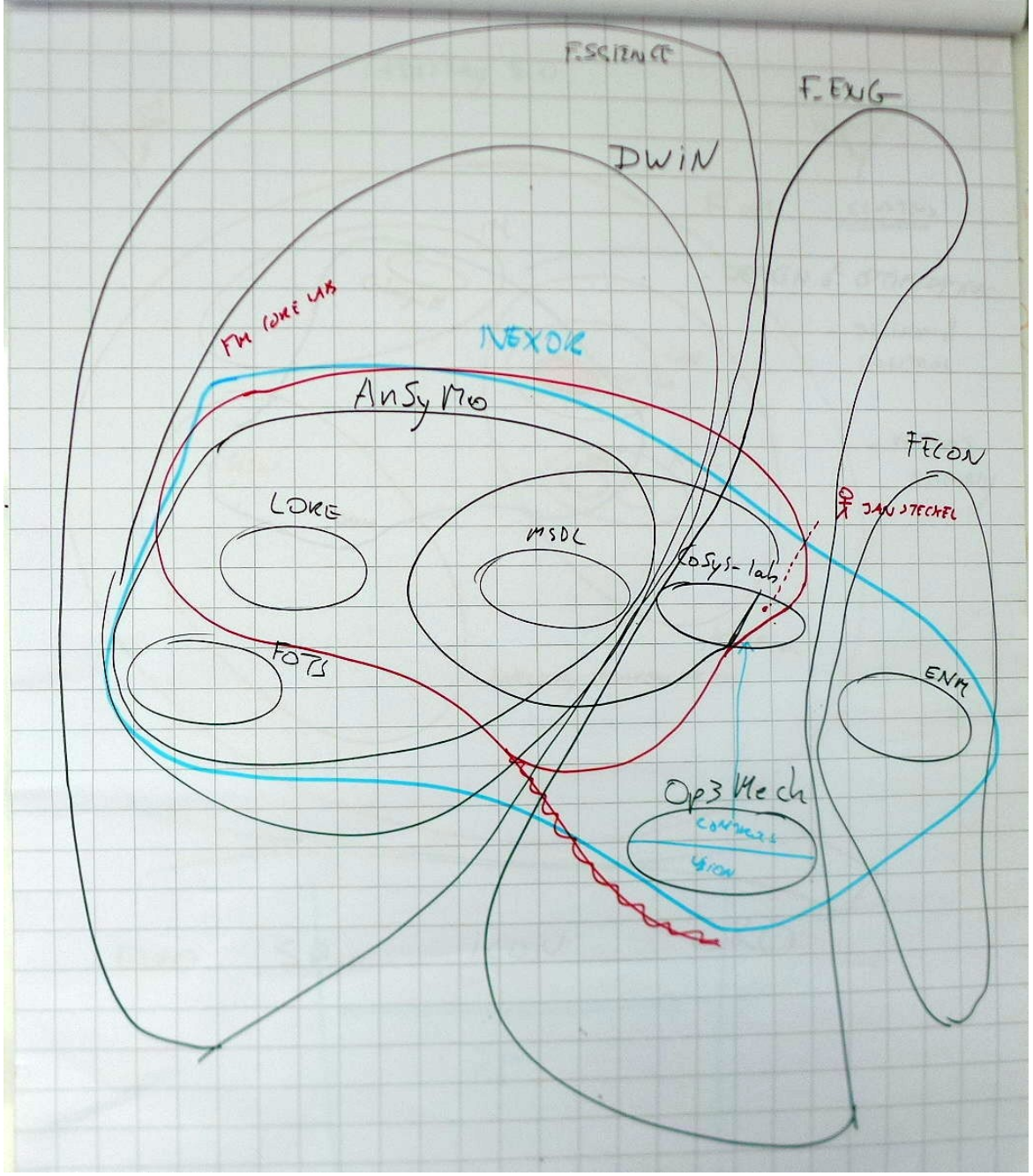
   Why? What? How?

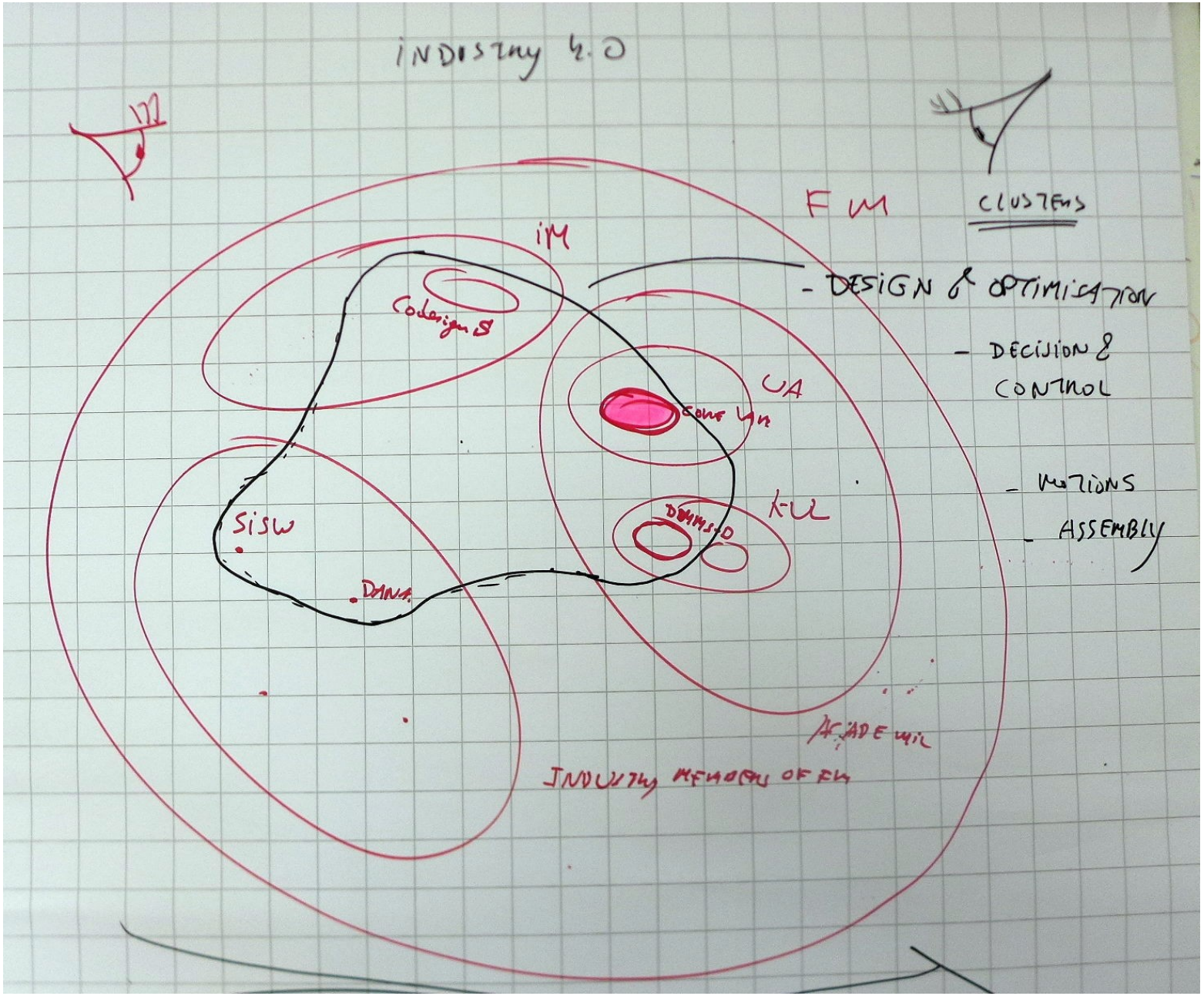
2. See the big picture (research/projects/...)

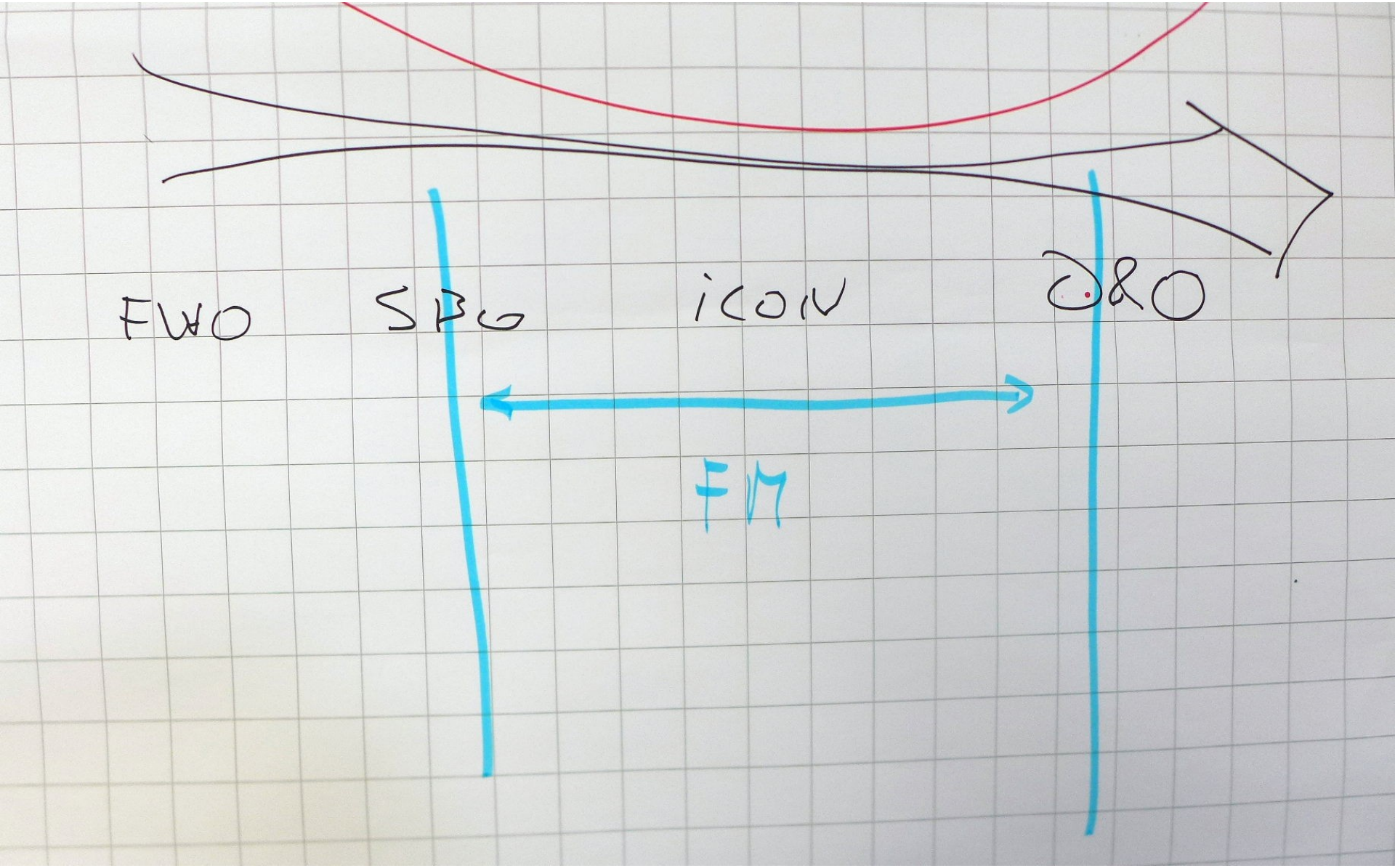
3. Roadmapping

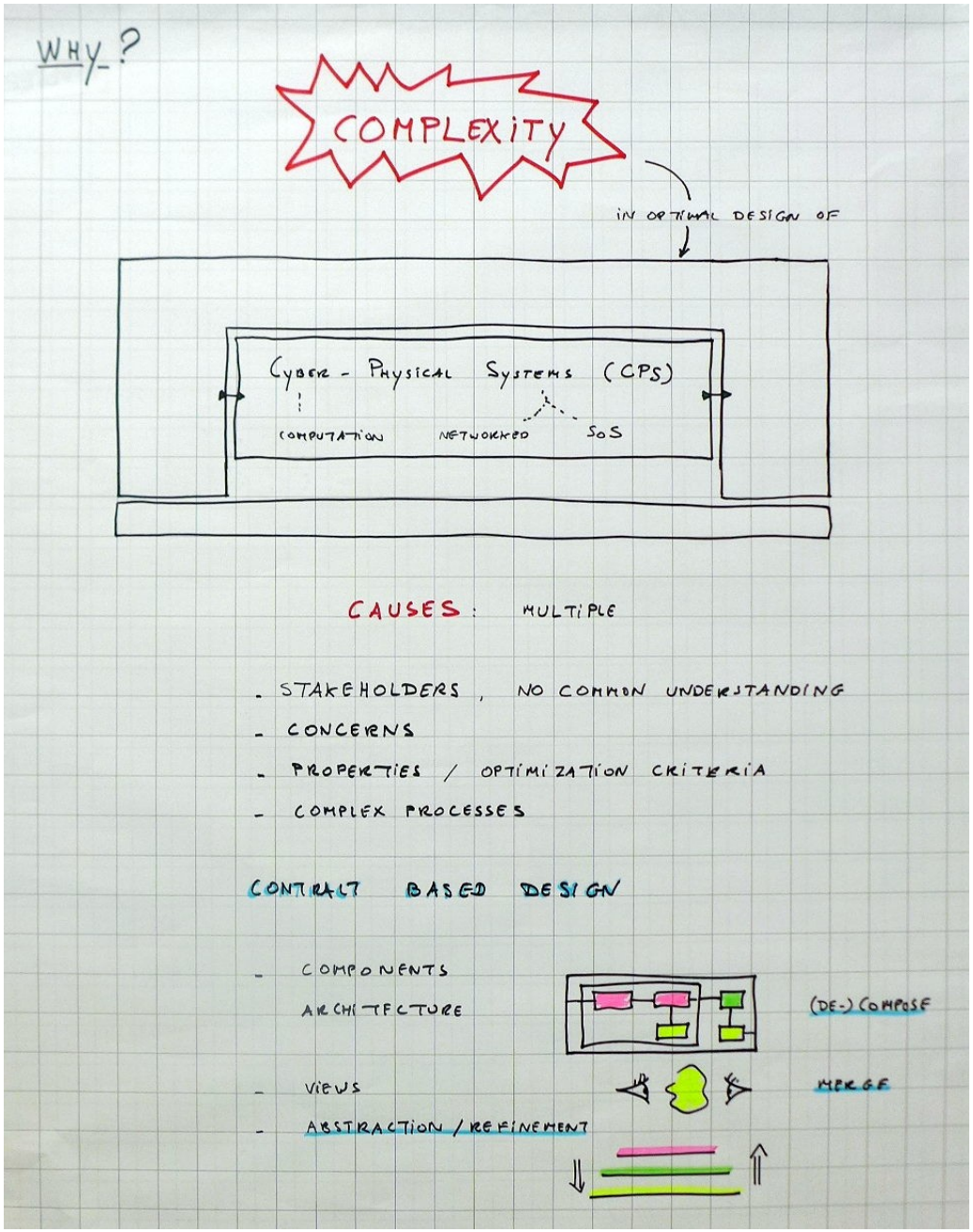
4. Build Portfolio(s)

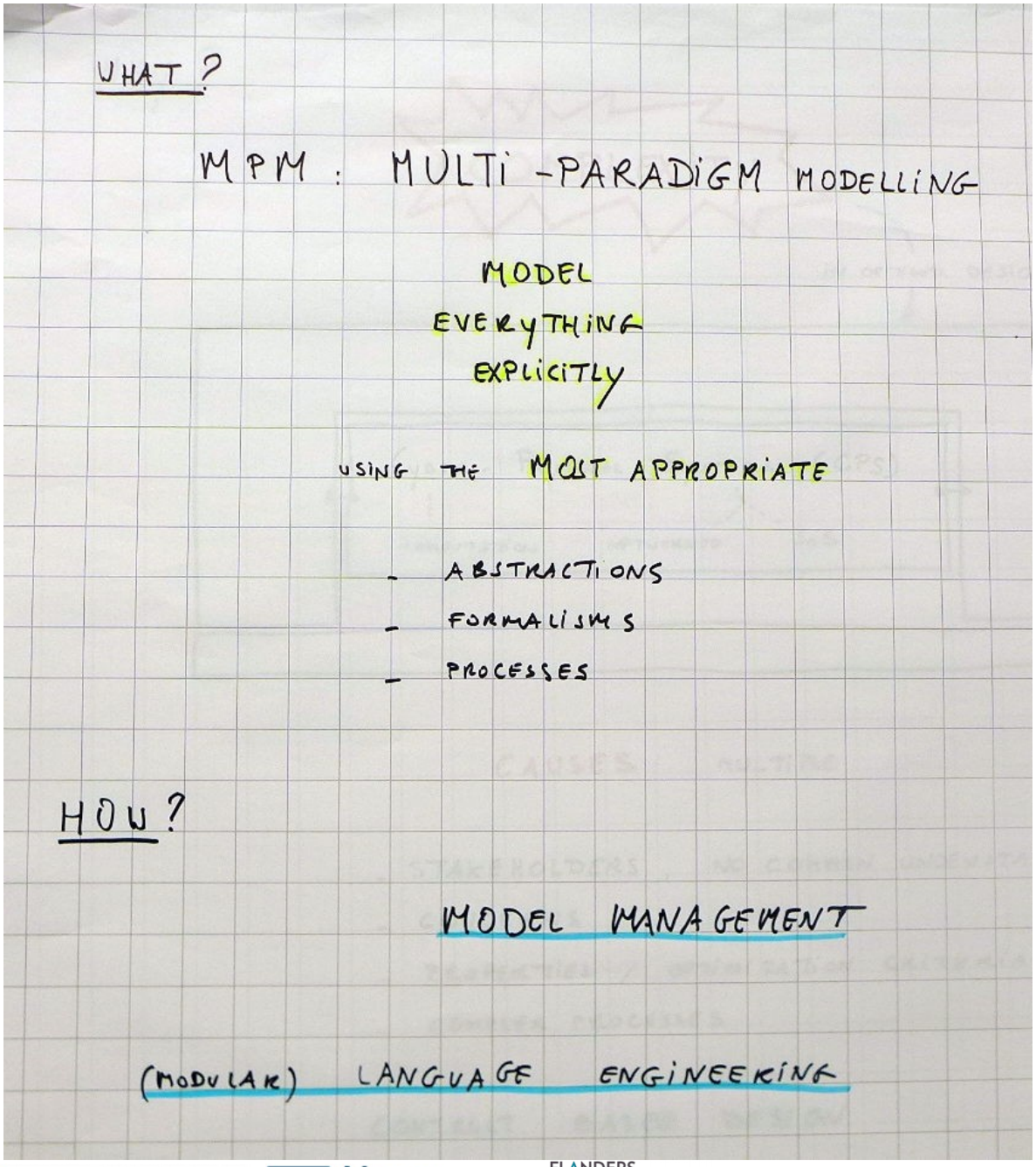
MSDL
AnSyMo
NEXOR
Flanders Make
…
n context

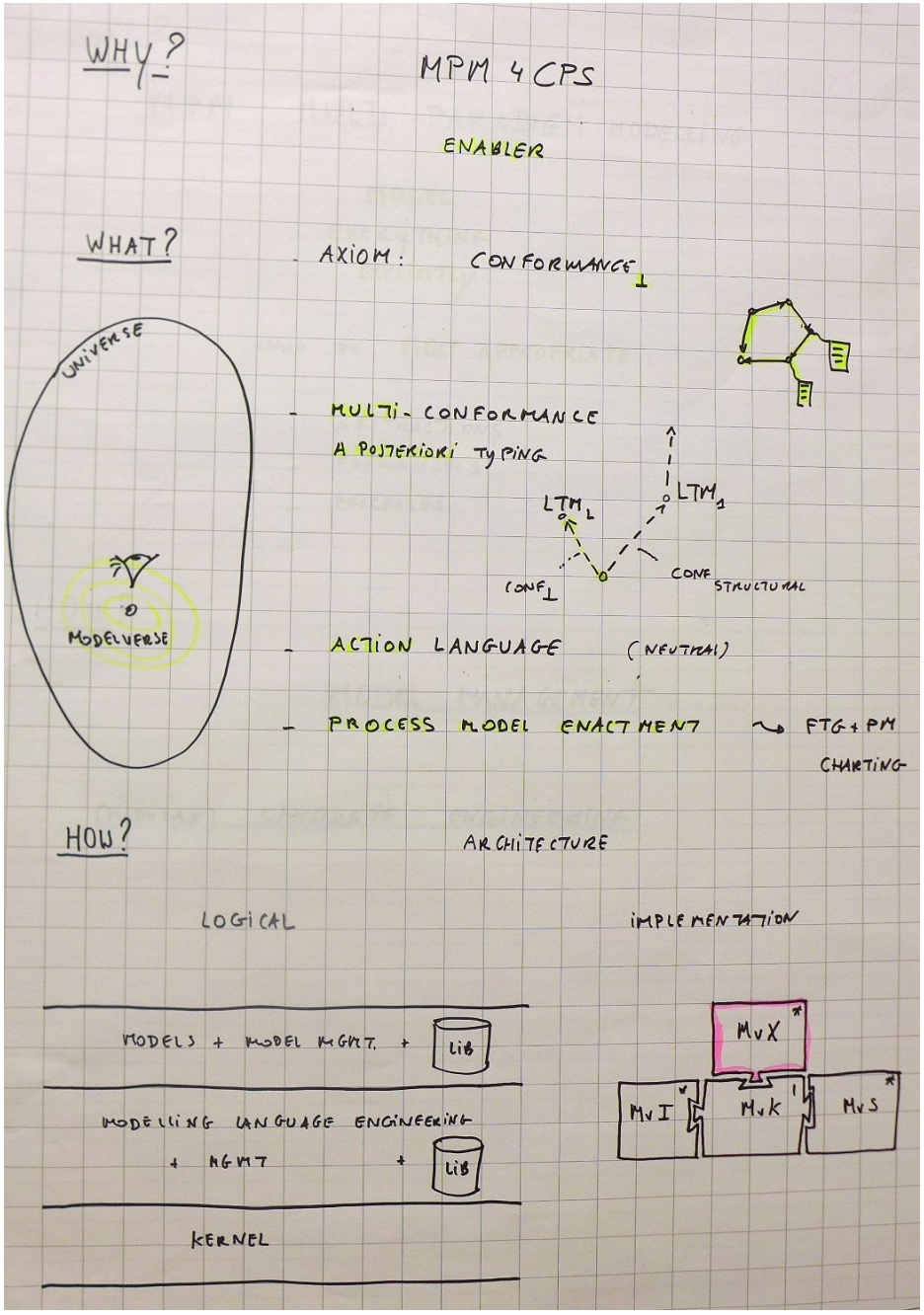



INDUSTRY 4.0
FM
CLUSTERS
- DESIGN & OPTIMISATION
- DECISION & CONTROL
- MOTIONS
- ASSEMBLY
iM
Codesigns
UA
SiSW
DOMAIN
ACADEMIC


FWO
SPO
ICON
O&O
FM


WHY ?
COMPLEXITY
IN OPTIMAL DESIGN OF
CYBER - PHYSICAL SYSTEMS (CPS)
COMPUTATION
NETWORKED
SoS
CAUSES: MULTIPLE
- STAKEHOLDERS, NO COMMON UNDERSTANDING
- CONCERNS
- PROPERTIES / OPTIMIZATION CRITERIA
- COMPLEX PROCESSES
CONTRACT BASED DESIGN
- COMPONENTS
ARCHITECTURE
(DE-) COMPOSE
- VIEWS
MERGE
- ABSTRACTION / REFINEMENT


WHAT?
MPM: MULTI-PARADIGM MODELLING
MODEL
EVERYTHING
EXPLICITLY
USING THE MOST APPROPRIATE
- ABSTRACTIONS
- FORMALISMS
- PROCESSES
HOW?
MODEL MANAGEMENT
(MODULAR) LANGUAGE ENGINEERING


WHY?
MPM 4 CPS
ENABLER
WHAT?
- AXIOM: CONFORMANCE ⊥
UNIVERSE
MODELVERSE
- MULTI-CONFORMANCE
A POSTERIORI TYPING
LTM⊥
LTM1
CONF⊥
CONF STRUCTURAL
- ACTION LANGUAGE (NEUTRAL)
- PROCESS MODEL ENACTMENT ⇝ PTG + PM CHARTING
HOW?
ARCHITECTURE
LOGICAL
IMPLEMENTATION
MODELS + MODEL MGMT. + LIB
MODELLING LANGUAGE ENGINEERING + MGMT + LIB
KERNEL
MvX
MvI
MvK
MvS

UNIVERSITÉ DE GENÈVE
Centre Universitaire d'Informatique

FACULTÉ DES SCIENCES
Professeur D. Buchs, directeur
Professeur G. Falquet, codirecteur

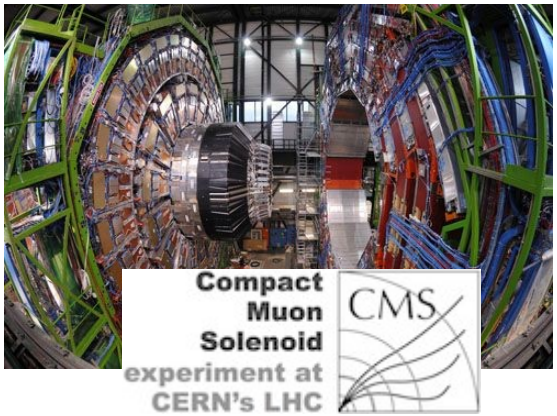
A Methodology For The Development Of Complex Domain Specific Languages

THÈSE

présentée à la Faculté des sciences de l'Université de Genève
pour obtenir le grade de Docteur ès sciences, mention informatique

2010

Matteo Risoldi



CMS Tracker Cosmic Rack

Design

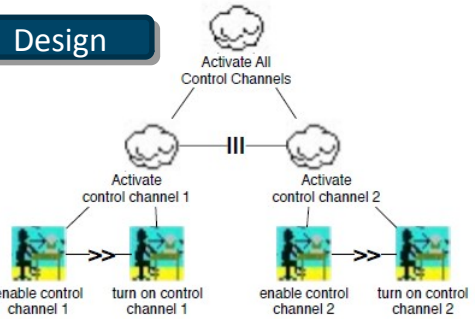


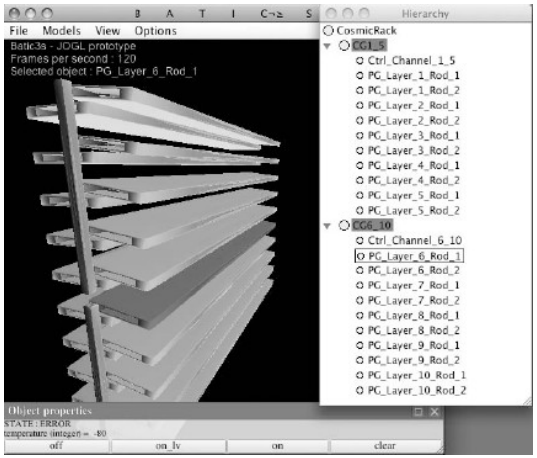
Figure 4.8. CTT for the turn on control channels *task*

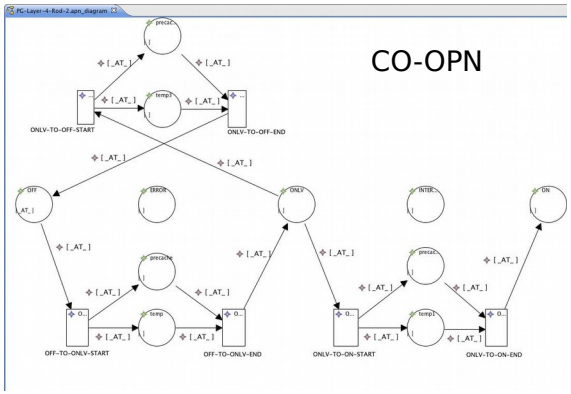
Property

```
*PG-Layer-4-Rod-2.apnmm_diagram   properties.prop
import 'PG-Layer-4-Rod-2.apnmm'
import 'blackToken.adt'

Expressions

  MUTUAL_EXCLUSION : (((card($on in ON) + card($onlv in ONLV)) + card($off in OFF
  NOSTATE : (((card($on in ON) + card($onlv in ONLV)) + card($off in OFF)) + card

   TEMP : card($tmp in temp)=1;
   TEMP1 : card($tmp in temp1)=1;
   TEMP2 : card($tmp in temp2)=1;
   TEMP3 : card($tmp in temp3)=1;
   TEMP4 : card($tmp in temp4)=1;
   TEMP5 : card($tmp in temp5)=1;
   TEMP6 : card($tmp in temp6)=1;
   TEMP7 : card($tmp in temp7)=1;
   TEMP8 : card($tmp in temp8)=1;
   TEMP9 : card($tmp in temp9)=1;
   TEMP10 : card($tmp in temp10)=1;
   INTERMEDIATE_STATE : (((((((((@TEMP | @TEMP1) | @TEMP2) | @TEMP3) | @TEMP4) |

Check
  (!(@INTERMEDIATE_STATE) => @MUTUAL_EXCLUSION);
```

Application

UI prototype



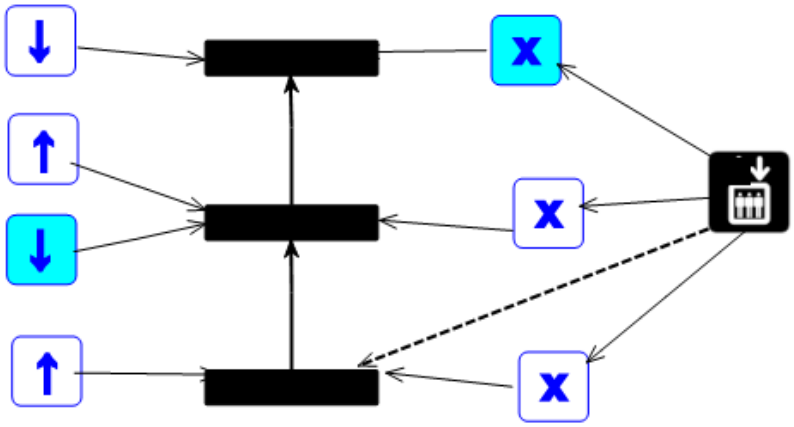



```
Properties  Specification Imports  (x)= Variables  Console  Problems
AlPiNA Model Checker Engine; [Java Application] /System/Library/Frameworks/Java
************************************************************
Compute State Space...
Reachability Time : 8 ms
State Space has been fully generated.
************************************************************
Check the properties...
Check property : [(!(((((((((((Card(tmp in temp:TRUE) EQUALS 1) or (Card(tmp in
 temp1:TRUE) EQUALS 1)) or (Card(tmp in temp2:TRUE) EQUALS 1)) or (Card(tmp in t
emp3:TRUE) EQUALS 1)) or (Card(tmp in temp4:TRUE) EQUALS 1)) or (Card(tmp in tem
p5:TRUE) EQUALS 1)) or (Card(tmp in temp6:TRUE) EQUALS 1)) or (Card(tmp in temp7
:TRUE) EQUALS 1)) or (Card(tmp in temp8:TRUE) EQUALS 1)) or (Card(tmp in temp9:T
RUE) EQUALS 1)) or (Card(tmp in temp10:TRUE) EQUALS 1))) implies ((((((Card(on in
 ON:TRUE) plus Card(onlv in ONLV:TRUE)) plus Card(off in OFF:TRUE)) plus Card(er
ror in ERROR:TRUE)) plus Card(int in INTERLOCKED:TRUE)) EQUALS 1))]
Property holds : OK
************************************************************
Property Check is finished.
```

Alpina

Designing Requirements/Property Languages

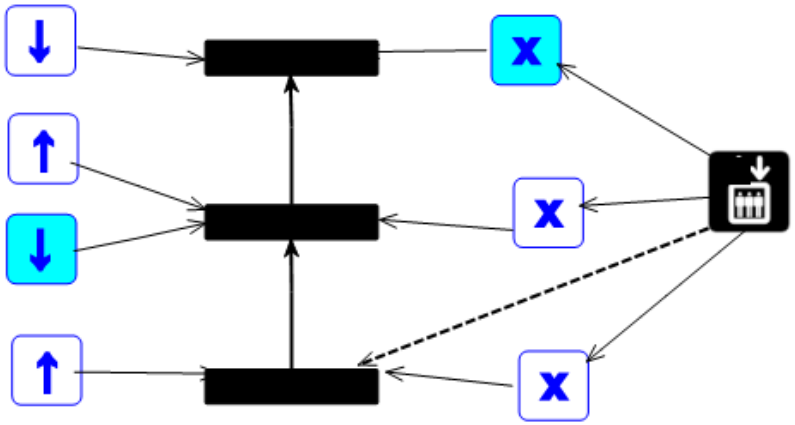
Design ⊨ Property

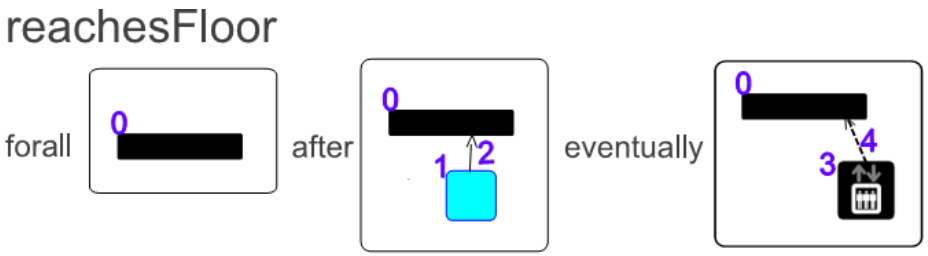


$$\Box(((go0 \wedge up0) \vee \Diamond(floor0 \vee idle)) \rightarrow ((\neg(floor0) \vee \neg(floor0 \vee idle)) \mathcal{U}((floor0 \vee idle) \wedge (((floor0) \vee \neg(floor0 \vee idle)) \mathcal{U}((floor0 \vee idle) \wedge ((\neg(floor0) \vee \neg(floor0 \vee idle)) \mathcal{U}((floor0 \vee idle) \wedge (((floor0) \vee \neg(floor0 \vee idle)) \mathcal{U}((floor0 \vee idle) \wedge (\neg(floor0) \mathcal{U}(floor0 \vee idle)))))))))) \vee \Box(((go1 \wedge up1 \wedge down1) \vee \Diamond(floor1 \vee idle)) \rightarrow ((\neg(floor1) \vee \neg(floor1 \vee idle)) \mathcal{U}((floor1 \vee idle) \wedge (((floor1) \vee \neg(floor1 \vee idle)) \mathcal{U}((floor1 \vee idle) \wedge ((\neg(floor1) \vee \neg(floor1 \vee idle)) \mathcal{U}((floor1 \vee idle) \wedge (((floor1) \vee \neg(floor1 \vee idle)) \mathcal{U}((floor1 \vee idle) \wedge (\neg(floor1) \mathcal{U}(floor1 \vee idle)))))))))) \vee \Box(((go2 \wedge down2) \vee \Diamond(floor2 \vee idle)) \rightarrow ((\neg(floor2) \vee \neg(floor2 \vee idle)) \mathcal{U}((floor2 \vee idle) \wedge (((floor2) \vee \neg(floor2 \vee idle)) \mathcal{U}((floor2 \vee idle) \wedge ((\neg(floor2) \vee \neg(floor2 \vee idle)) \mathcal{U}((floor2 \vee idle) \wedge (((floor2) \vee \neg(floor2 \vee idle)) \mathcal{U}((floor2 \vee idle) \wedge (\neg(floor2) \mathcal{U}(floor2 \vee idle)))))))))))$$

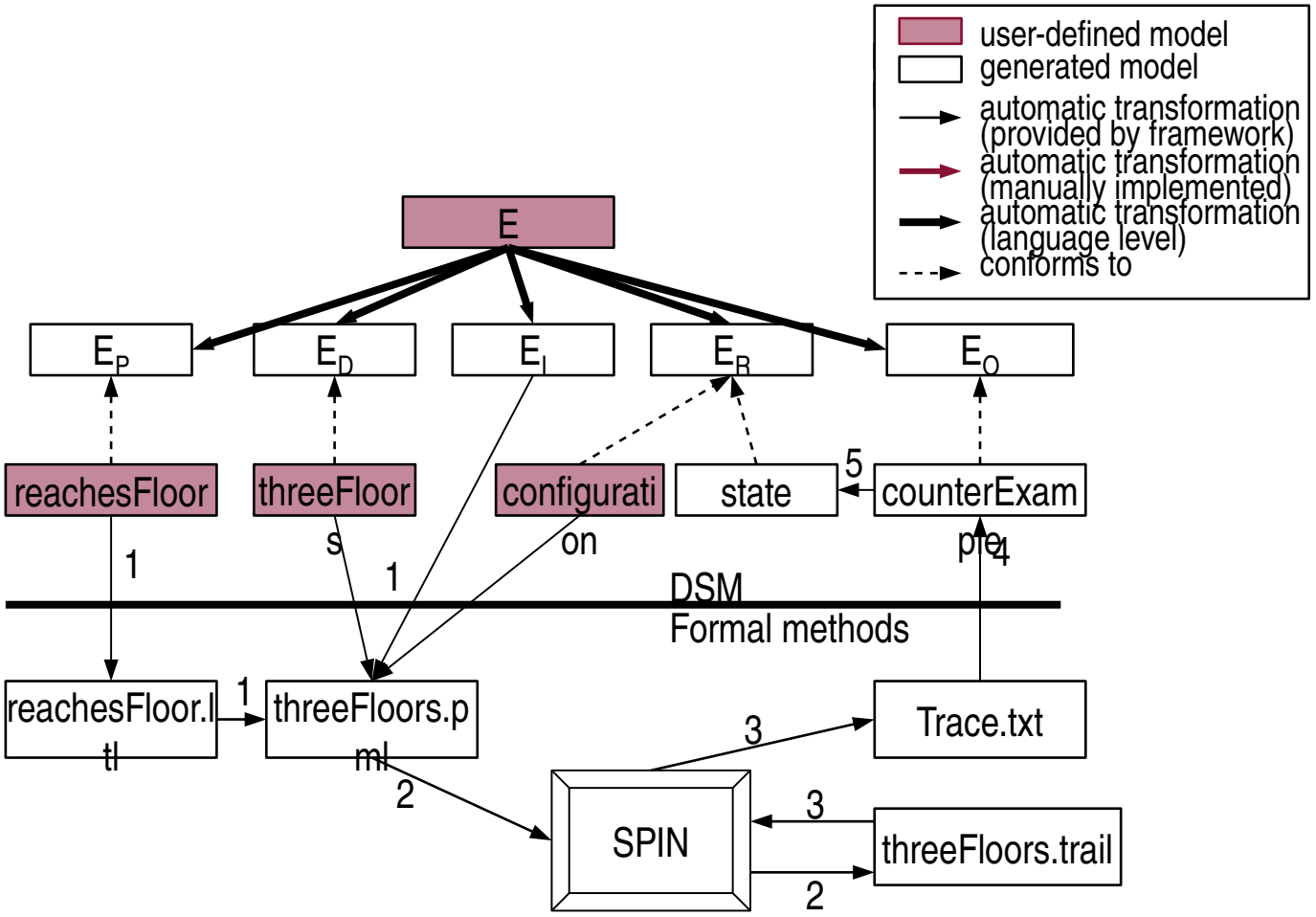
Designing Requirements/Property Languages

Design ⊨ Property



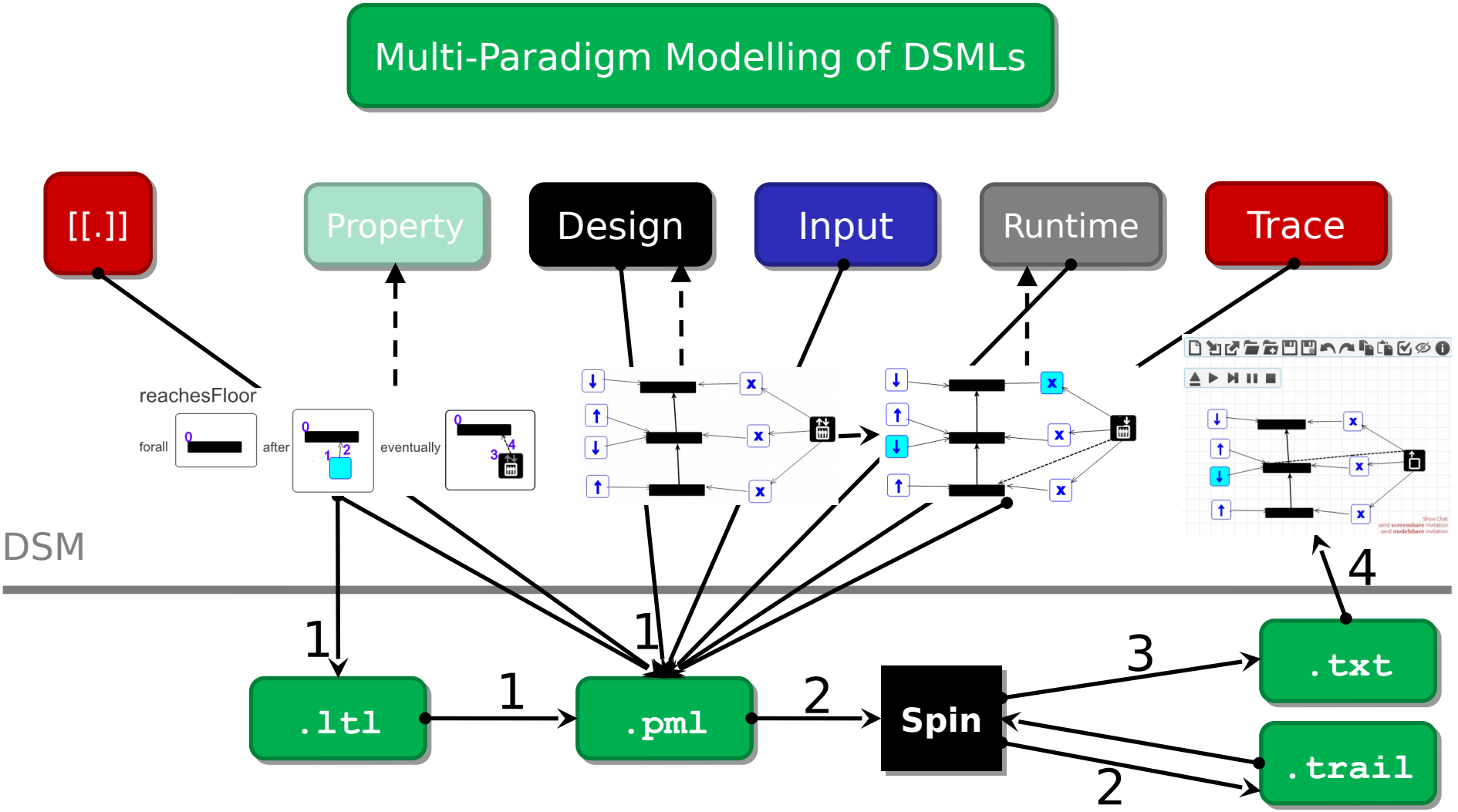


Designing Requirements/Property Languages



Bart Meyers, Romuald Deshayes, Levi Lucio, Eugene Syriani, Hans Vangheluwe, and Manuel Wimmer. ProMoBox: A Framework for Generating Domain-Specic Property Languages. In Software Language Engineering (SLE), Vasteras, Sweden, volume 8706 of Lecture Notes in Computer Science (LNCS) , pages 1- 20. Springer. September 2014.

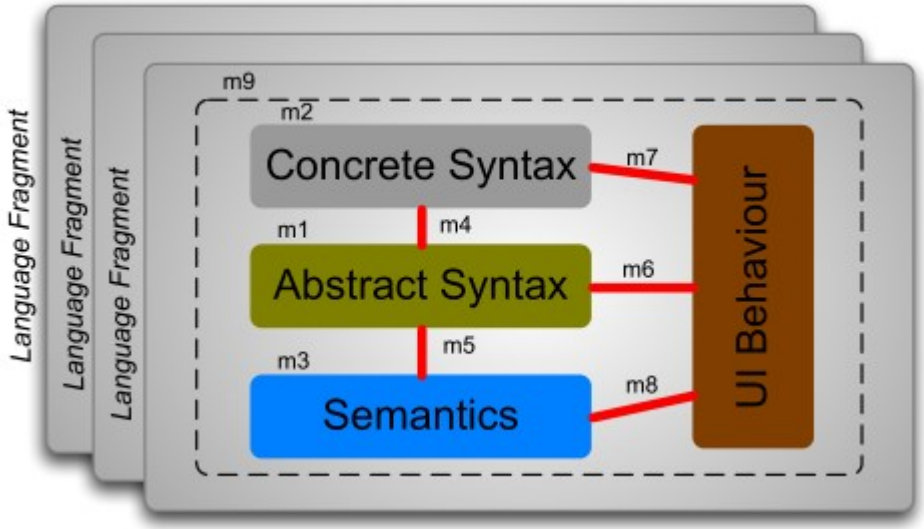
Designing Requirements/Property Languages



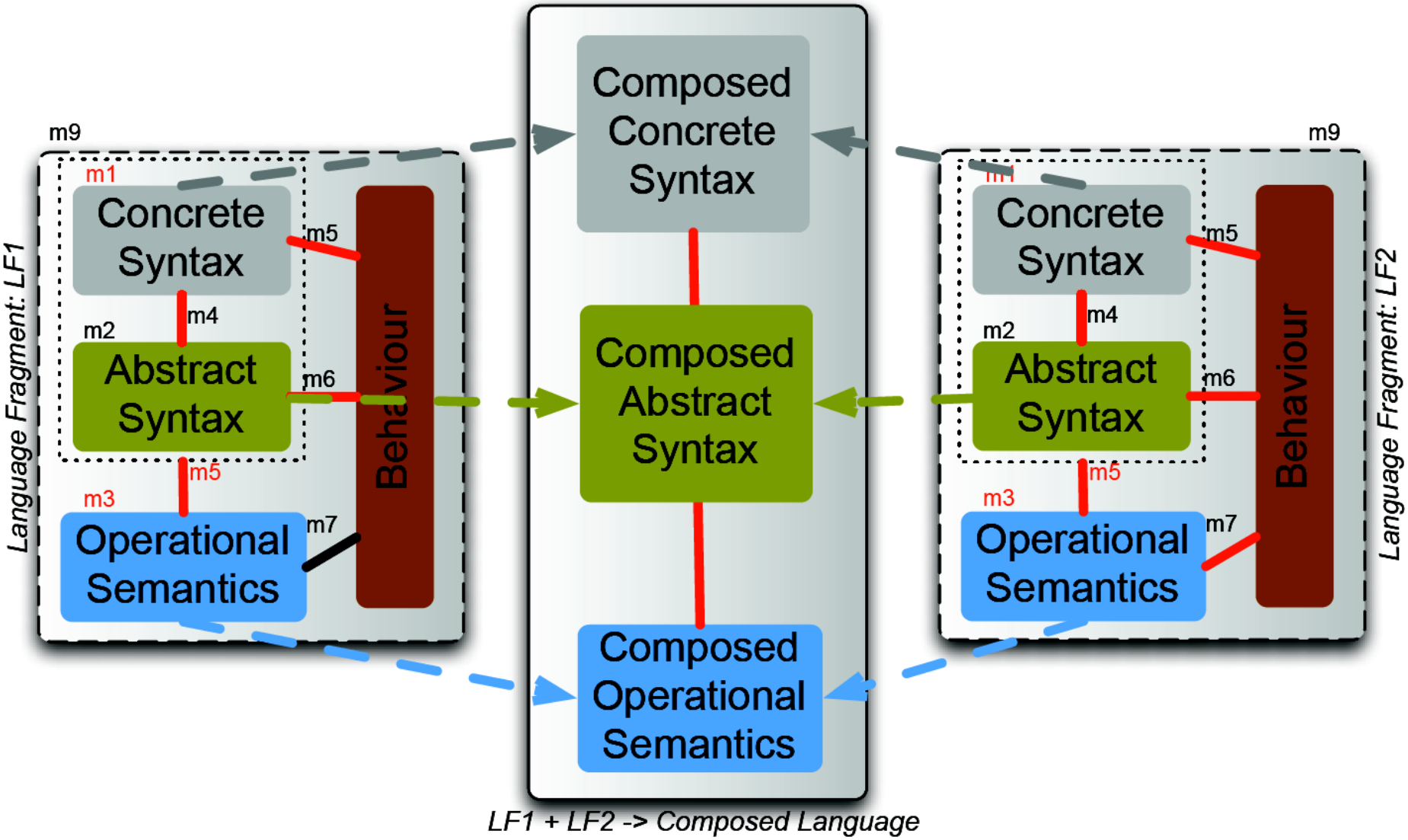
Bart Meyers, Romuald Deshayes, Levi Lucio, Eugene Syriani, Hans Vangheluwe, and Manuel Wimmer. ProMoBox: A Framework for Generating Domain-Specic Property Languages. In Software Language Engineering (SLE), Vasteras, Sweden, volume 8706 of Lecture Notes in Computer Science (LNCS) , pages 1- 20. Springer. September 2014.

Modular Language Environment Engineering (L+E)Spec

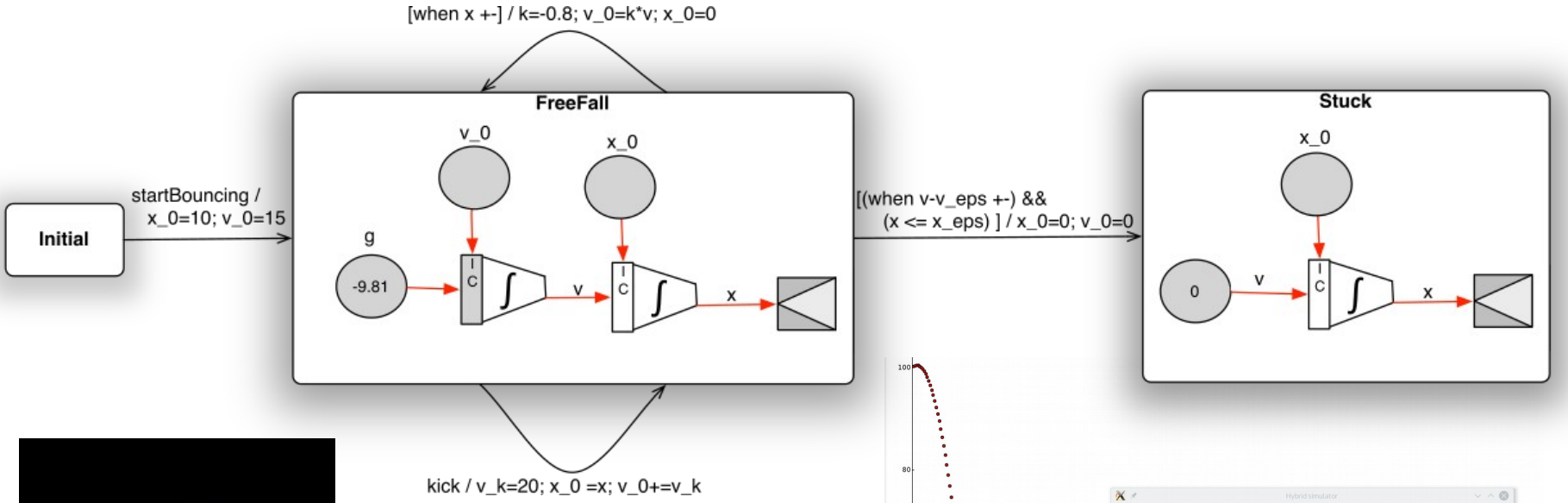
- Reusable components of a language environment specification
  - Syntax (Concrete/Abstract)
  - Operational semantics
  - Interaction (UI) Behaviour
- Combine fragments to create environments for hybrid languages

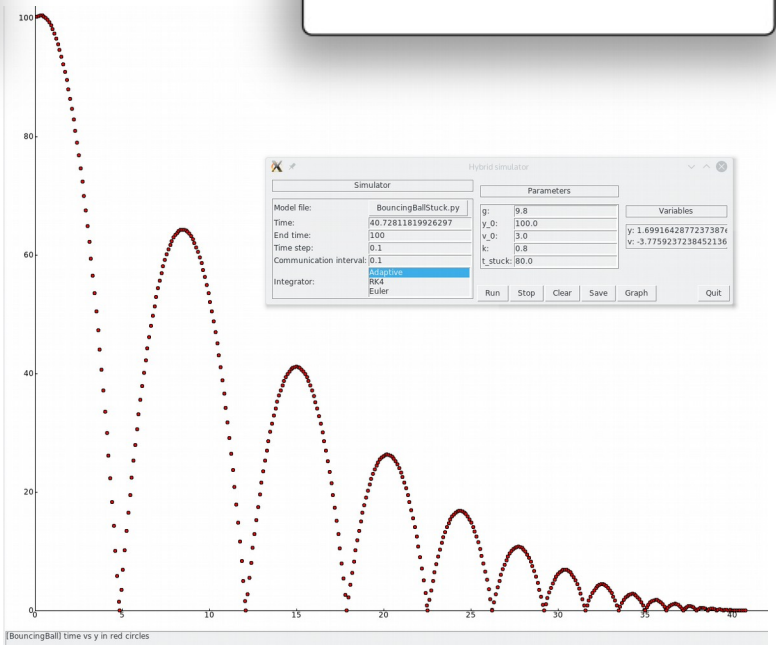


Composition of Language Fragments



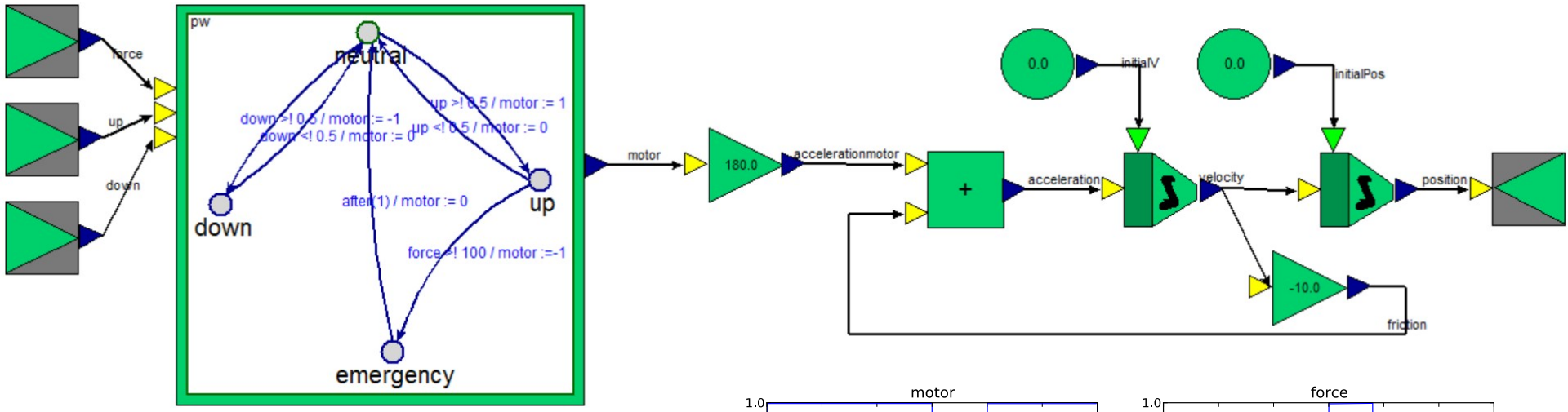
FSA+CBD composition 1

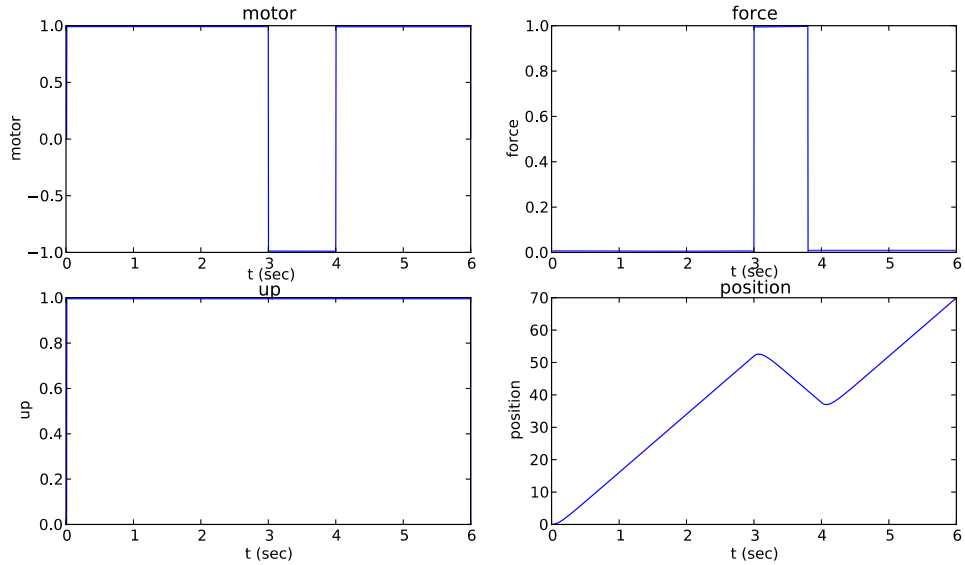




Simon Lacoste-Julien, Hans Vangheluwe, Juan de Lara, and Pieter J. Mosterman. Meta-modelling hybrid formalisms. In Pieter J. Mosterman and Jin-Shyan Lee, editors, IEEE International Symposium on Computer-Aided Control System Design , pages 65 - 70. IEEE Computer Society Press, September 2004. Taipei, Taiwan.

FSA+CBD composition 2

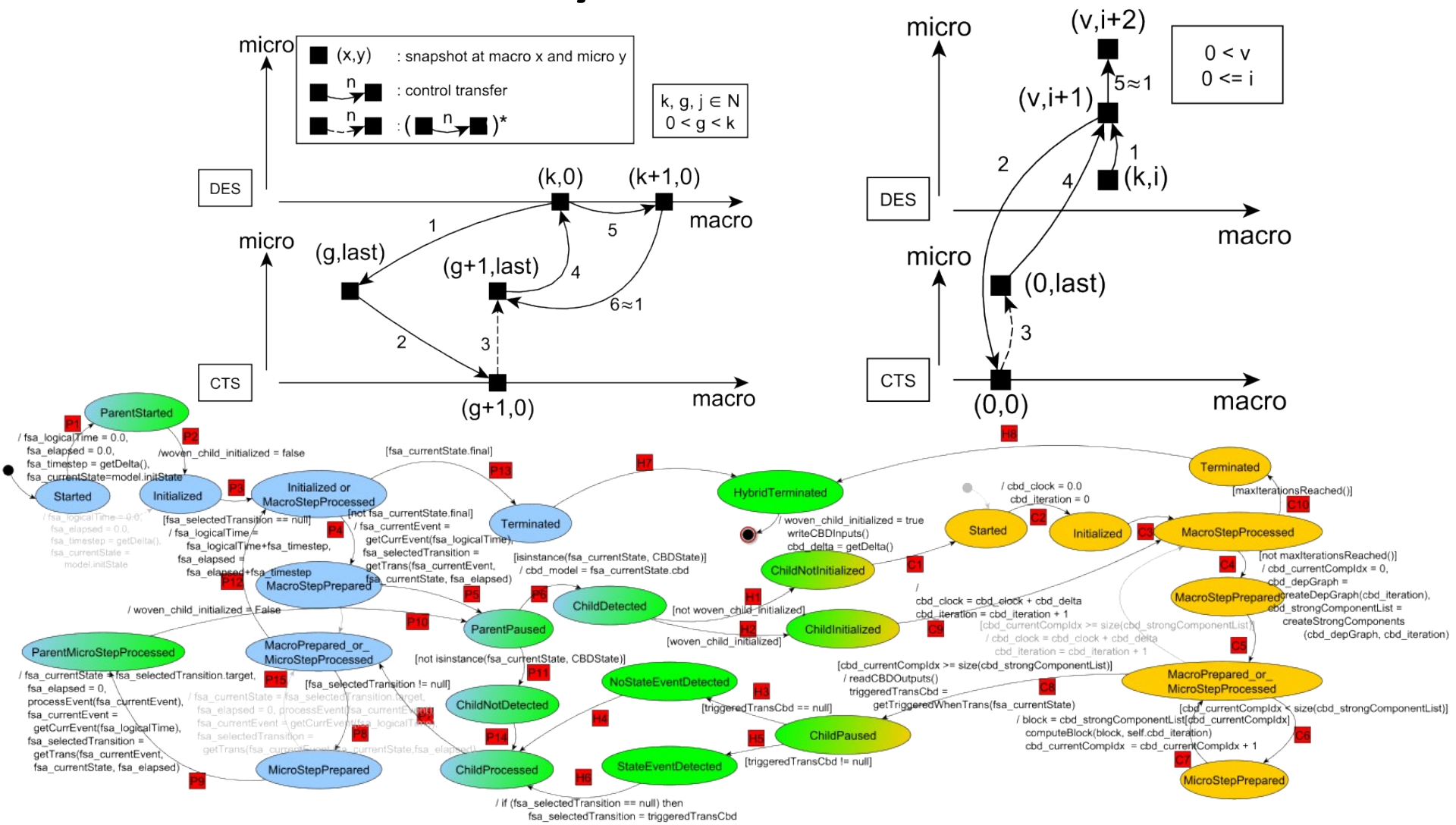




Bart Meyers, Joachim Denil, Frederic Boulanger, Cecile Hardebolle, Christophe Jacquet, Hans Vangheluwe. A DSL for Explicit Semantic Adaptation. MPM@MoDELS 2013:47-56.

Joachim Denil, Bart Meyers, Paul De Meulenaere, and Hans Vangheluwe. Explicit semantic adaptation of hybrid formalisms for FMI co-simulation. In Proceedings of the 2015 Spring Simulation Multi-Conference, pages 852 - 859. SCS, April 2015.

Hybrid TFSA


micro
(x,y) : snapshot at macro x and micro y
n : control transfer
n (n)*
k, g, j ∈ N
0 < g < k
DES
(k,0)
(k+1,0)
macro
1
5
micro
(g,last)
(g+1,last)
4
6≈1
2
3
CTS
(g+1,0)
macro
micro
(v,i+2)
0 < v
0 <= i
5≈1
(v,i+1)
2
1
4
(k,i)
DES
macro
micro
(0,last)
3
CTS
(0,0)
macro
ParentStarted
P1
/ fsa_logicalTime = 0.0,
fsa_elapsed = 0.0,
fsa_timestep = getDelta(),
fsa_currentState=model.initState
P2
/woven_child_initialized = false
Started
Initialized
P3
Initialized or
MacroStepProcessed
[fsa_selectedTransition == null]
P4
[not fsa_currentState.final]
/ fsa_currentEvent =
getCurrEvent(fsa_logicalTime),
fsa_selectedTransition =
getTrans(fsa_currentEvent,
fsa_currentState, fsa_elapsed)
/ fsa_logicalTime =
fsa_logicalTime+fsa_timestep,
fsa_elapsed =
fsa_elapsed+fsa_timestep
P12
MacroStepPrepared
[fsa_currentState.final]
P13
Terminated
H7
HybridTerminated
[isinstance(fsa_currentState, CBDState)]
/ cbd_model = fsa_currentState.cbd
/ woven_child_initialized = true
writeCBDInputs()
cbd_delta = getDelta()
ChildNotInitialized
C1
/ cbd_clock = 0.0
cbd_iteration = 0
Started
C2
Initialized
C3
MacroStepProcessed
Terminated
H8
[maxIterationsReached()]
C10
[not maxIterationsReached()]
/ cbd_currentCompIdx = 0,
cbd_depGraph =
createDepGraph(cbd_iteration),
cbd_strongComponentList =
createStrongComponents
(cbd_depGraph, cbd_iteration)
C4
MacroStepPrepared
/ woven_child_initialized = False
P5
P8
ChildDetected
[not woven_child_initialized]
H1
ChildInitialized
C9
/
cbd_clock = cbd_clock + cbd_delta
cbd_iteration = cbd_iteration + 1
[cbd_currentCompIdx >= size(cbd_strongComponentList)]
/ cbd_clock = cbd_clock + cbd_delta
cbd_iteration = cbd_iteration + 1
C5
P10
ParentPaused
[woven_child_initialized]
H2
ParentMicroStepProcessed
MacroPrepared_or_
MicroStepProcessed
[not isinstance(fsa_currentState, CBDState)]
P11
[fsa_selectedTransition != null]
P15
/ fsa_currentState = fsa_selectedTransition.target,
fsa_elapsed = 0,
processEvent(fsa_currentEvent),
fsa_currentEvent =
getCurrEvent(fsa_logicalTime),
fsa_selectedTransition =
getTrans(fsa_currentEvent,
fsa_currentState, fsa_elapsed)
P6
MicroStepPrepared
NoStateEventDetected
H3
[triggeredTransCbd == null]
ChildNotDetected
H4
P14
ChildProcessed
H6
ChildPaused
H5
[triggeredTransCbd != null]
StateEventDetected
/ if (fsa_selectedTransition == null) then
fsa_selectedTransition = triggeredTransCbd
[cbd_currentCompIdx >= size(cbd_strongComponentList)]
/ readCBDOutputs()
triggeredTransCbd =
getTriggeredWhenTrans(fsa_currentState)
C8
MacroPrepared_or_
MicroStepProcessed
[cbd_currentCompIdx < size(cbd_strongComponentList)]
/ block = cbd_strongComponentList[cbd_currentCompIdx]
computeBlock(block, self.cbd_iteration)
cbd_currentCompIdx = cbd_currentCompIdx + 1
C6
C7
MicroStepPrepared

F FACILITY

FOR

C COLLABORATIVE

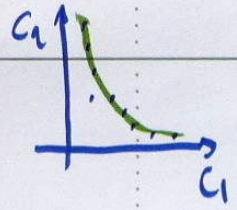
WHAT?

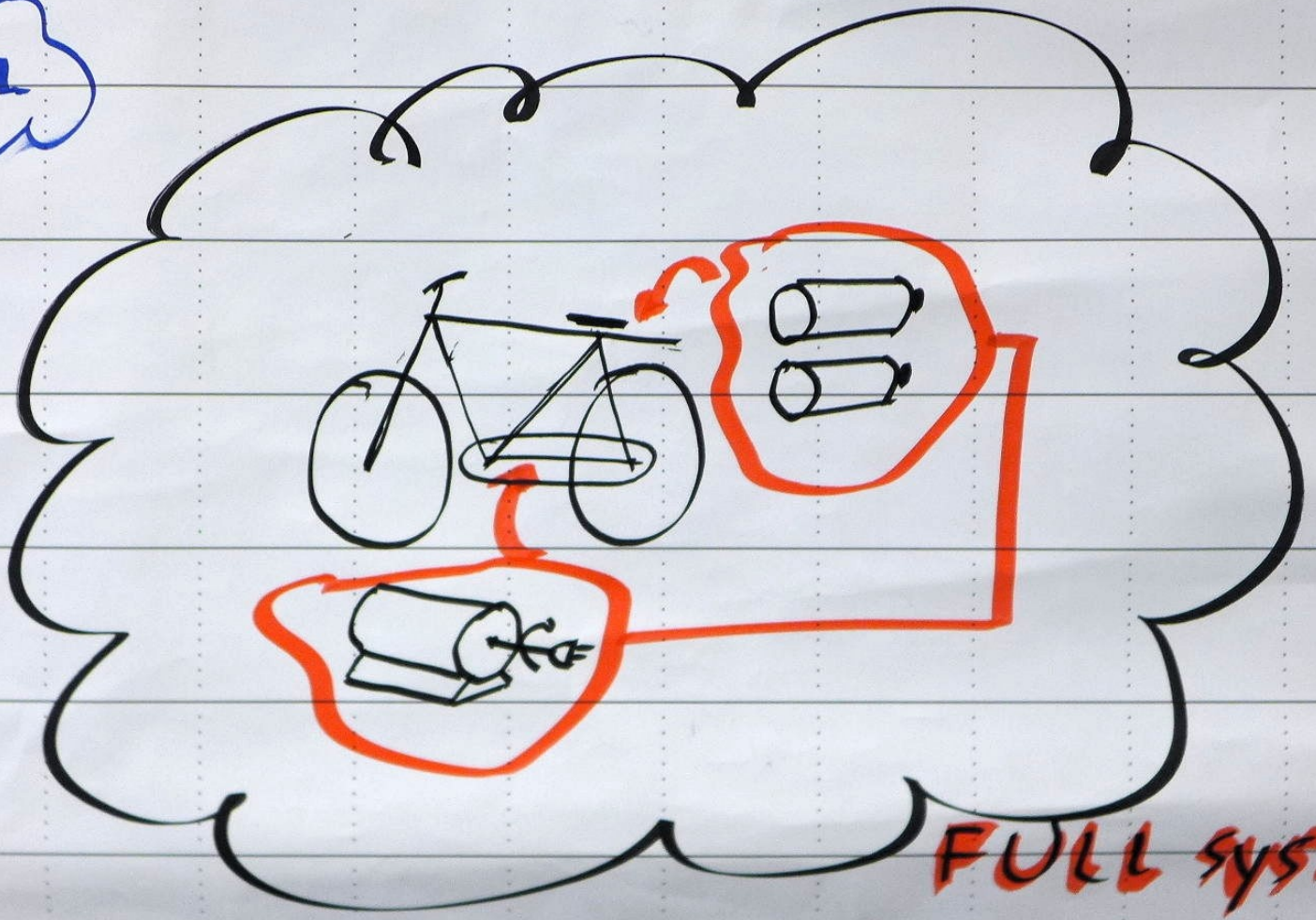
- SHOULD BE FUNDED

MBSE MODEL-BASED SYSTEMS ENGINEERING

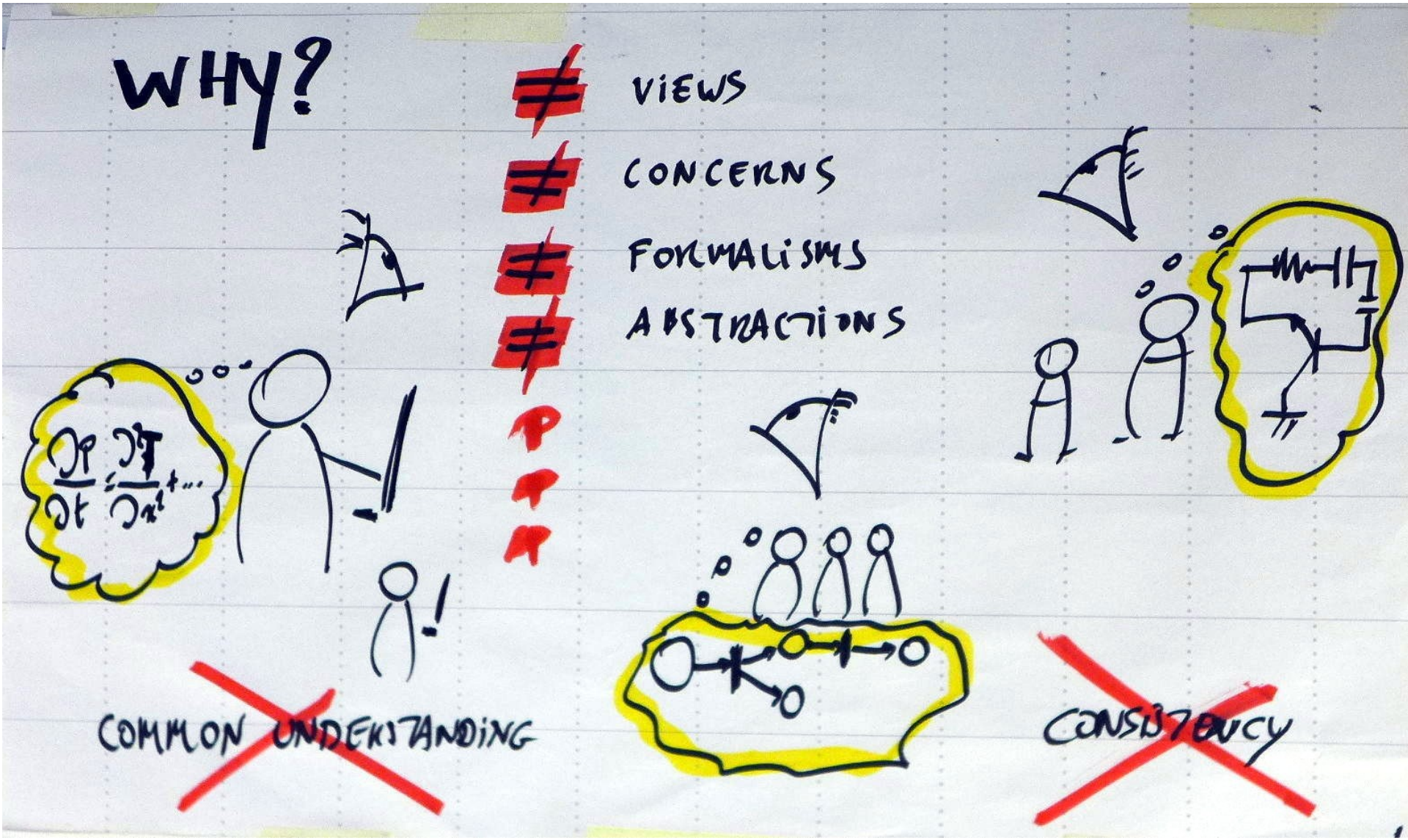
FC-MBSE

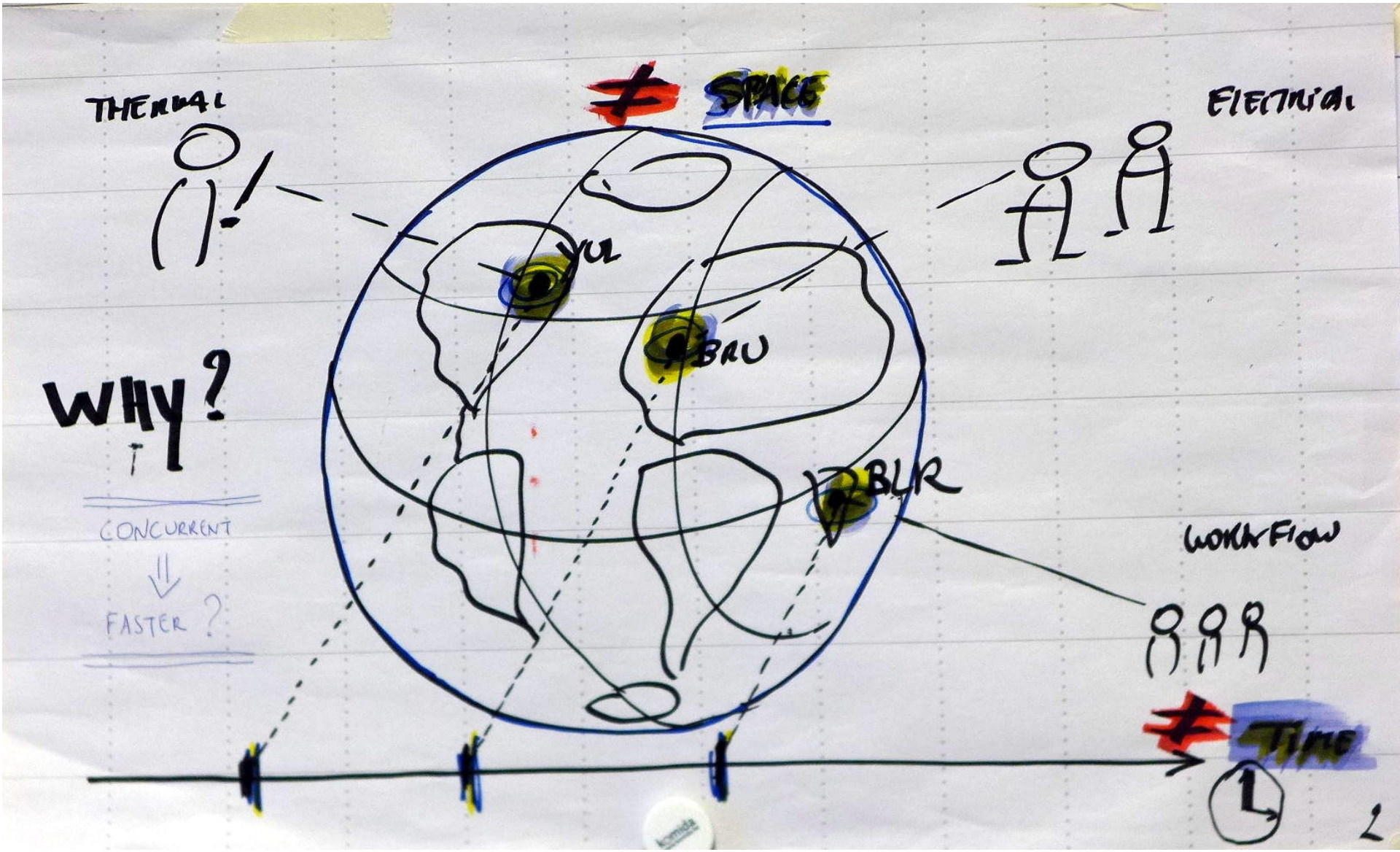
OPTIMAL

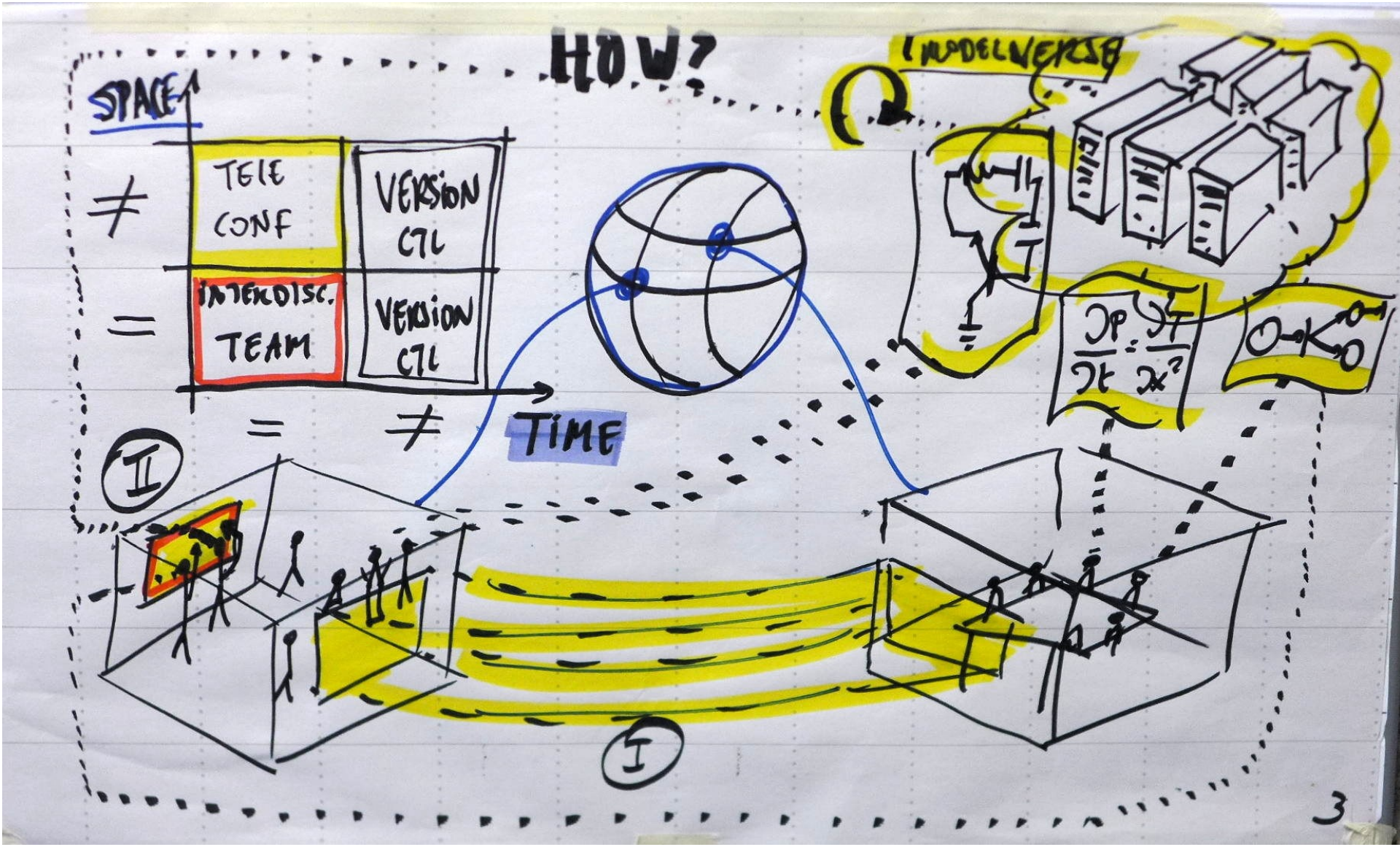




FULL SYSTEM


WHY?
≠ VIEWS
≠ CONCERNS
≠ FORMALISMS
≠ ABSTRACTIONS
≠
COMMON UNDERSTANDING
CONSISTENCY


THERMAL
≠ SPACE
ELECTRICAL
YUL
BRU
BLR
WHY?
CONCURRENT
FASTER ?
WORKFLOW
≠ TIME
1
2


HOW?
SPACE
≠
=
=
≠
TELE CONF
VERSION CTL
INTERDISC. TEAM
VERSION CTL
TIME
I
II
MODEL VERSIO
3

Research Topics

Analysis, Validation, Verification, Testing and Accreditation

Analysis and Verification of Model Transformations, Debugging, Instrumentation, Tracing, etc.

Language Engineering

Domain-Specific Languages, Model Transformation, Design-Space Exploration(web-based) Visual and Textual Modelling Environments, etc.

Simulation

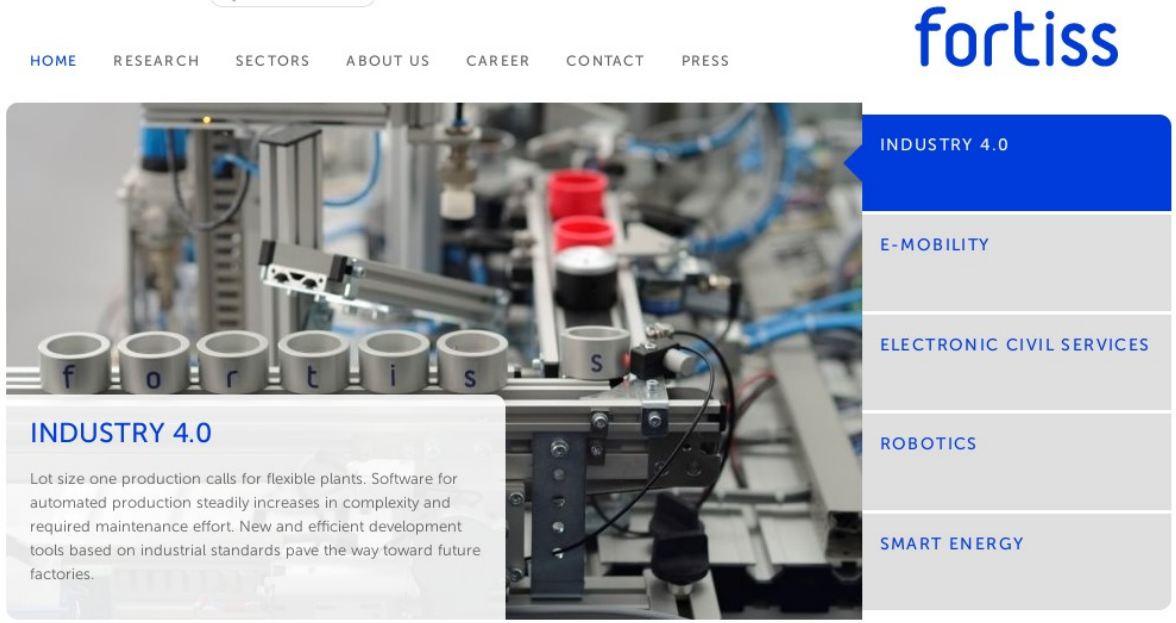
Co-Simulation, Discrete-event, DEVS, continuous time, a-causal (e.g., Modelica), physics-based (e.g., Bond Graph), etc.

Deployment & Resource-optimized Execution

Platforms (e.g. AUTOSAR, CAN, etc.), Deployment-Space Exploration, Virtualization, Models@run-time, Efficient execution of model transformations, etc.

Model Management and Process

FTG+PM, Safety (ISO 26262, Railway, etc,), Agile Modelling, Consistency management, contracts, Experimental frames, etc.



Future: towards ...

The Role of Models in Engineering of Cyber-Physical Systems – Challenges and Possibilities

Bernhard Schätz, fortiss GmbH
schaetz@fortiss.org

Roadmap:

1. Where are we (and where are others)?

2. Where do we want to be?

3. What's the path from 1. to 2. ?

MSDL-level questions,

also to be answered individually
(roadmap: now vs. future):

- who (people/labs) are our peers?

- what are our conferences?

- what are our publication venues?

- ...

SWOT:

- **S**trengths

- **W**eaknesses

- **O**pportunities

- **T**hreats

some Projects/Funding/Collaborators (academic collaborators not listed)

MPM4CPS http://www.mpm4cps.eu

http://www.cost.eu/COST_Actions/ict/Actions/IC1404

http://www.modelwriter.eu/

http://www.necsis.ca/

http://www.mbse4mechatronics.org/