Heidelberg Laureate Forum 2016
Workshop on
Cyber Physical Systems

Laureates:
Joseph Sifakis
Vint Cerf

Organizers:
Marc-Oliver Pahl
Amel Bennaceur
Sai Manoj

September 19, 2016

Screen shot from: “Alex Gibney, Zero Days (2016)”

- https://youtu.be/2qaxJs8wYVw

Cut material from the “Alex Gibney, Zero Days (2016)” trailer

Heidelberg Laureate Forum 2016
Workshop on
Cyber Physical Systems

Welcome

Joseph
Sifakis

Vint
Cerf

Marc-Oliver
Pahl

Amel
Bennaceur

Sai
Manoj

Screen shot from: "Alex Gibney, Zero Days (2016)"

Agenda

- Introduction to the Workshop (5 min; Marc-Oliver Pahl)
- 3 Research Examples and their Links to CPS? (15 min; Sai Manoj, Amel Bennaceur, Marc-Oliver Pahl)
- **CPS and IoT** (15 min; **Vint Cerf**)
- **Linking Physicality and Computation** (15 minutes; **Joseph Sifakis**)
- A Practical CPS Example: an Infusion Pump (15 min; Amel, Sai, Marc-Oliver)
- Discussing Challenges and Opportunities of CPS (20 min; all)
- Wrap-Up (5 min; Joseph Sifakis, Amel, Sai, Marc-Oliver)

Introduction

3 Research Examples
CPS and IoT
Linking Physicality and Computation
A Practical CPS Example: an Infusion Pump
Discussing Challenges and Opportunities of CPS
Wrap-Up

Screen shot from: “Alex Gibney, Zero Days (2016)”

Your Definitions…

What is a Cyber Physical System?

- 5 **environment**
- 5 information
- 3 **networking**
- 3 controlled
- 3 algorithms
- 3 **software**
- 3 internet
- 3 devices
- 3 input
- 3 real

- 2 integrations
- 2 computation
- 2 integrated
- 2 components
- 2 interacts
- 2 processes
- 2 computer
- 2 interact
- 2 sensors
- 2 parts
- 2 world
- 2 many
- 2 used

What is a Cyber Physical System?

- Cyber-Physical Systems (CPS) integrate
  - **computation**
  - **physical processes**, and
  - **networking**

- This is **complex**, e.g.
  - physical processes are typically continuous
  - software is **discrete** in time (sampling) and value (quantization) space
  - software is **more difficult to verify** than hardware as it is more dynamic, etc.

- CPS bring new **opportunities** such as *flexibility* or *cost efficiency*.

- CPS bring new **risks** such as *attacks* (video) or *unwanted behavior*.

Introduction

3 Research Examples


Sai
Manoj


Amel
Bennaceur


Marc-Oliver
Pahl

Screen shot from: “Alex Gibney, Zero Days (2016)”

TECHNISCHE UNIVERSITÄT WIEN

Institut für Computertechnik
Institute of Computer Technology

Cyber-physical System-on-Chip Design for Health Monitoring

Sai Manoj P D
TU Wien

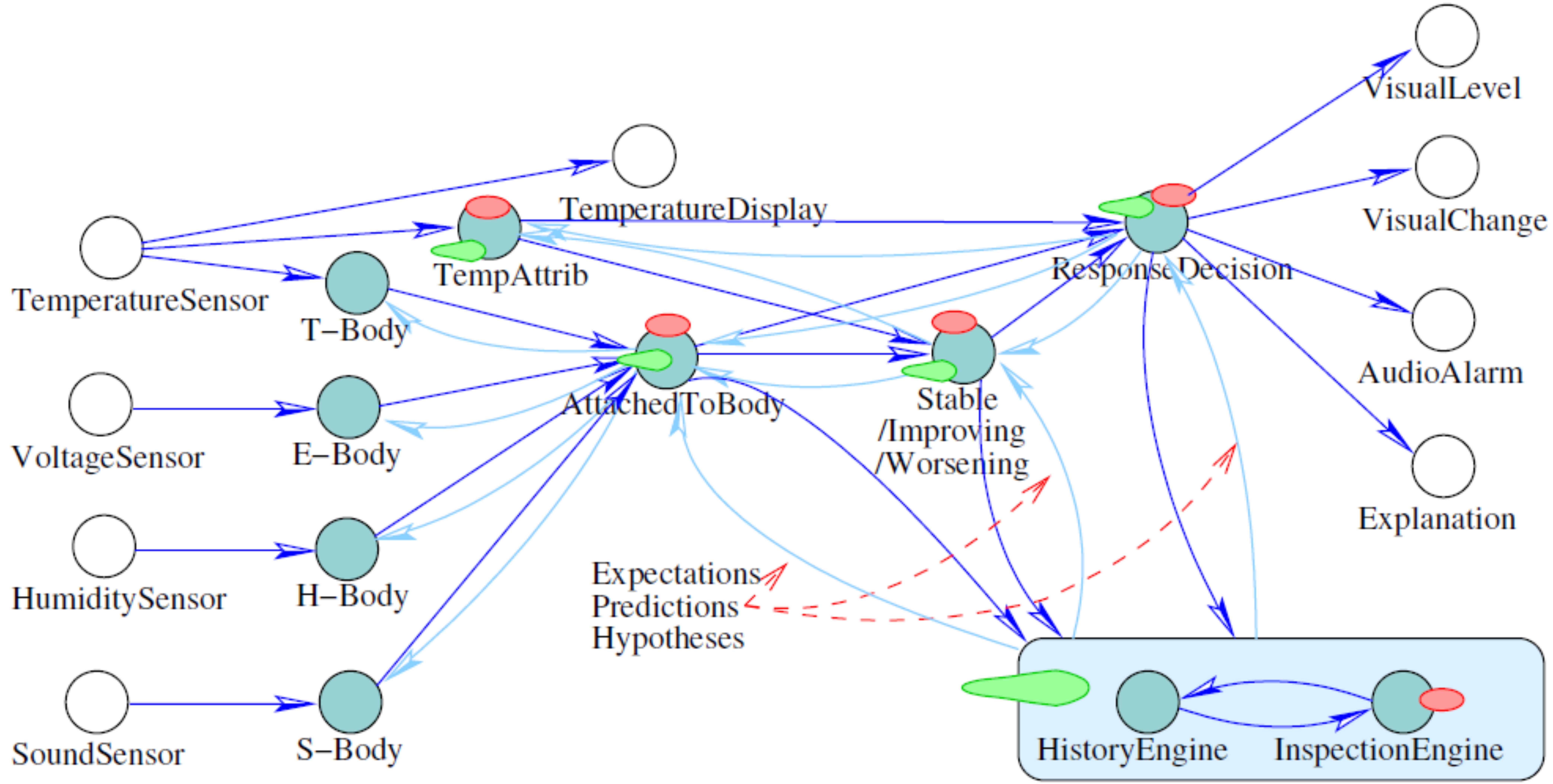
Introduction

- 2006-2010 B. Tech (ECE) from JNTU Anantapur, India
- 2010-2012 M. Tech (IT) from IIIT Bangalore, India
  - 2012 Thesis from TU Kaiserslautern, Germany
- 2012-2015 Ph. D. From NTU, Singapore
  - Thesis: 2.5D and 3D I/O Designs for Energy-efficient Memory-logic Integration towards Thousand core on-chip
- 2015-present Post-doc at TU Wien, Austria

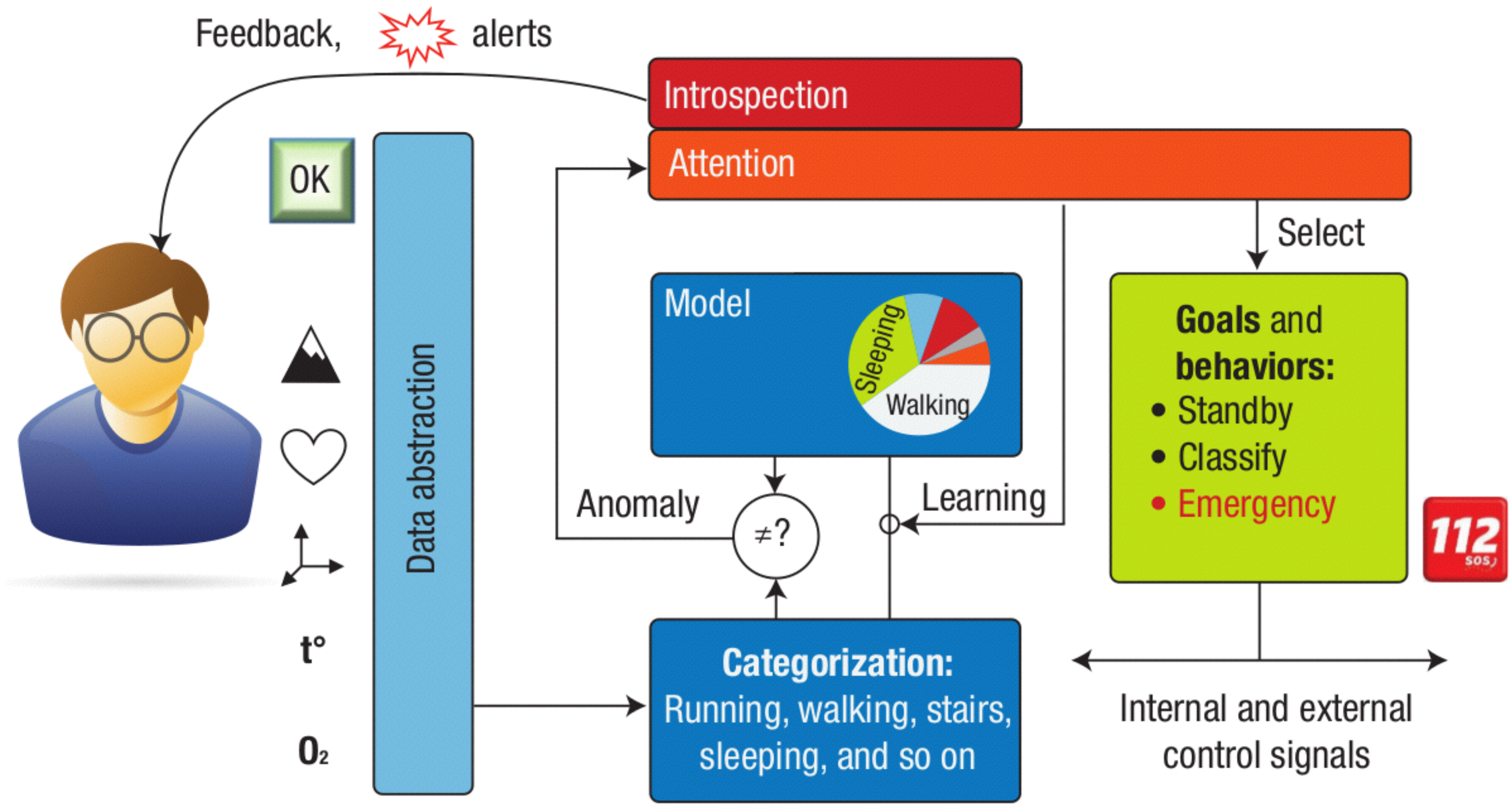
Overview of My Research

- Many-core systems are emerging with developments in IoT
- Many challenges such as integrating hetergeneous components, resource management, and so on needs to addressed
- Emerging integration technology systems such as 2.5D and 3D ICs are researched
- Power, thermal and bandwdith management are some of the issues focussed
- Health-care such as on-chip anomaly detection is one more additional area currently working on
- **Self-aware CPSoC** is currently under investigation

Bio-patch Modeling and Prediction

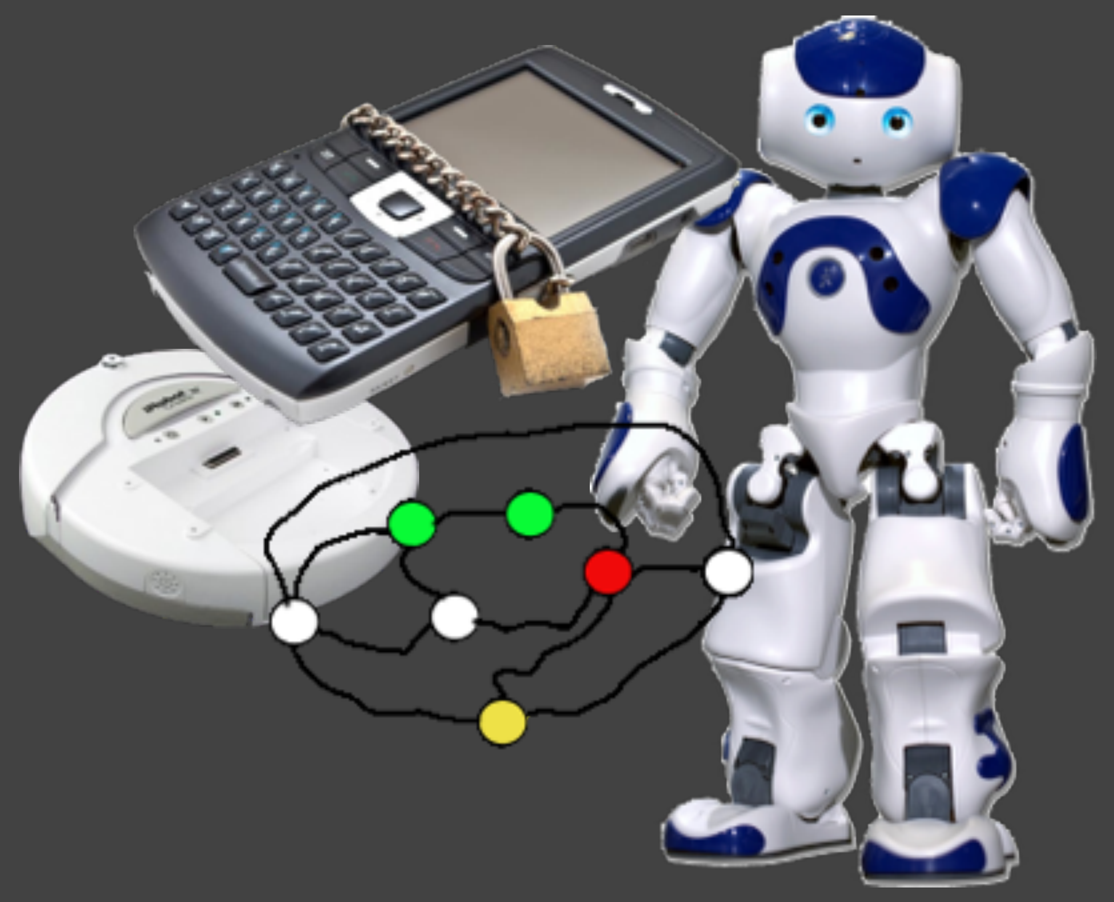


CPSoC for Health-care



Mediator Synthesis for Collaborative Adaptation and Security

Amel Bennaceur
The Open University, UK



My Research

Mediator Synthesis for Collaborative Adaptation and Security

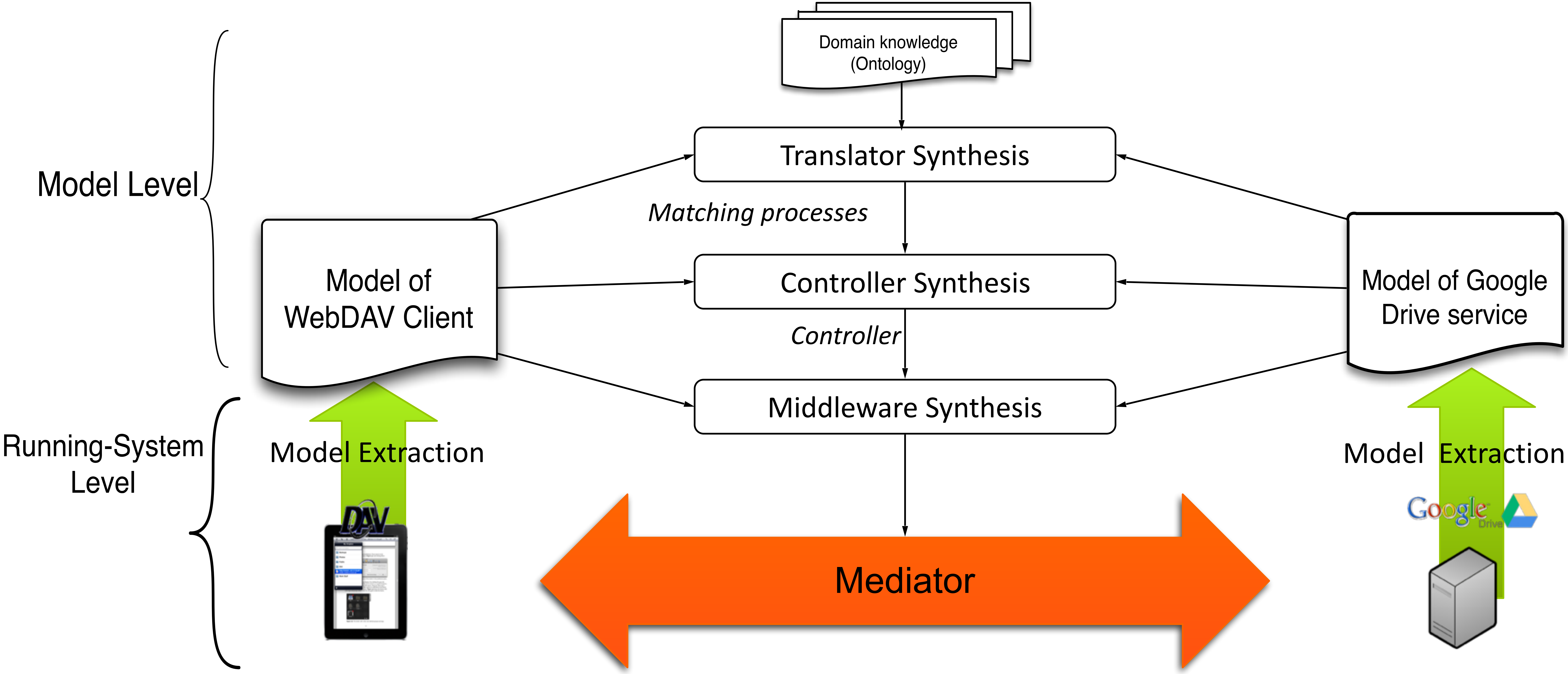
Creating software that can automatically make software systems interact with one another to support individual and people achieving their goals

Adaptive Software

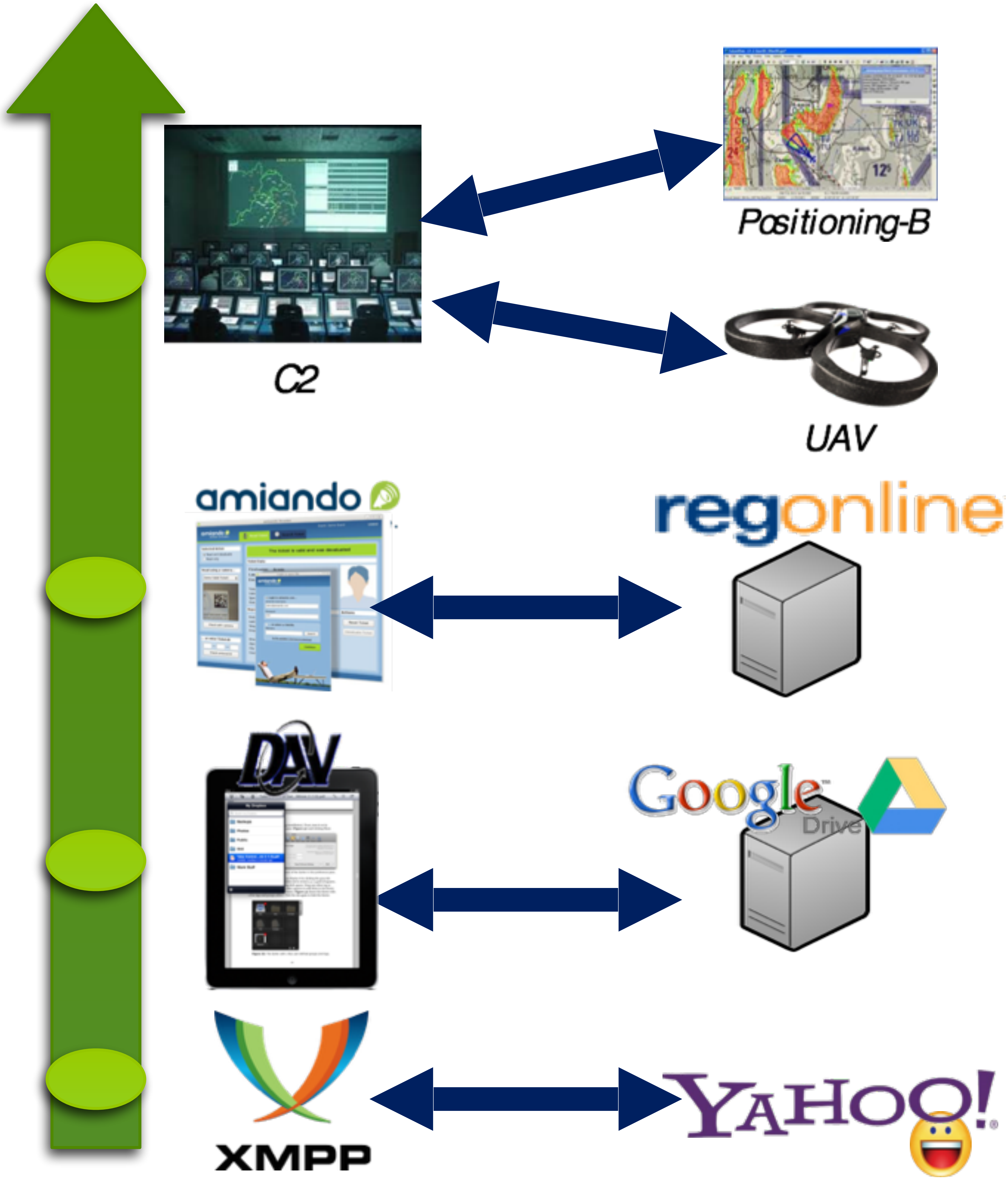
Distributed Systems

Formal Methods

Mediator Synthesis: The Process



Does it Work?



Emergency Management

- Heterogeneous interfaces & behaviours
- Heterogeneous middleware solutions
- mediation at runtime

Event Management

- Heterogeneous interfaces & behaviours
- Heterogeneous middleware solutions

File Management

- Heterogeneous interfaces & behaviours

Instant Messaging

- Heterogeneous interfaces

Mediator Synthesis in the Internet of Things

From composition of **services**

- ✓ Automated
- ✓ Correct by construction

- – Complete knowledge of the environment
- – Trustworthy individual services

To composition of ‘**Things**’

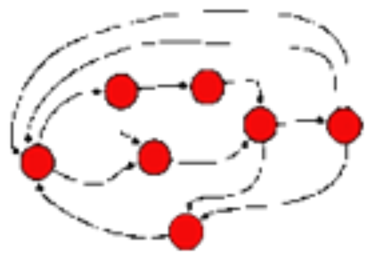
Cyber (Software)

Physical (Objects)

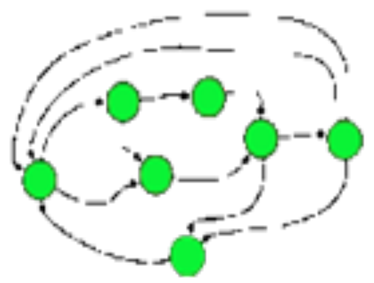
Social (People)

Mediator Synthesis for Cyber-Physical Systems

Cyber – Discrete-time controller



Environment, E



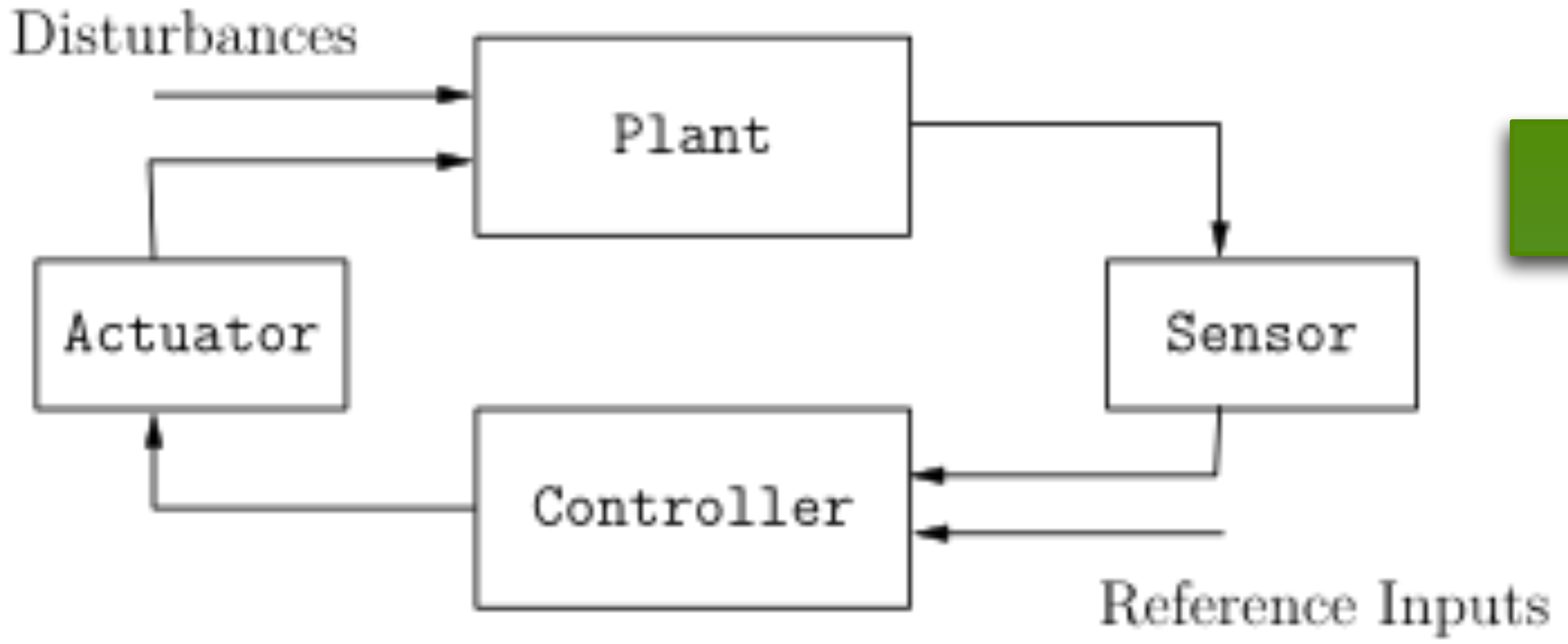
Desirable property, P

Synthesis

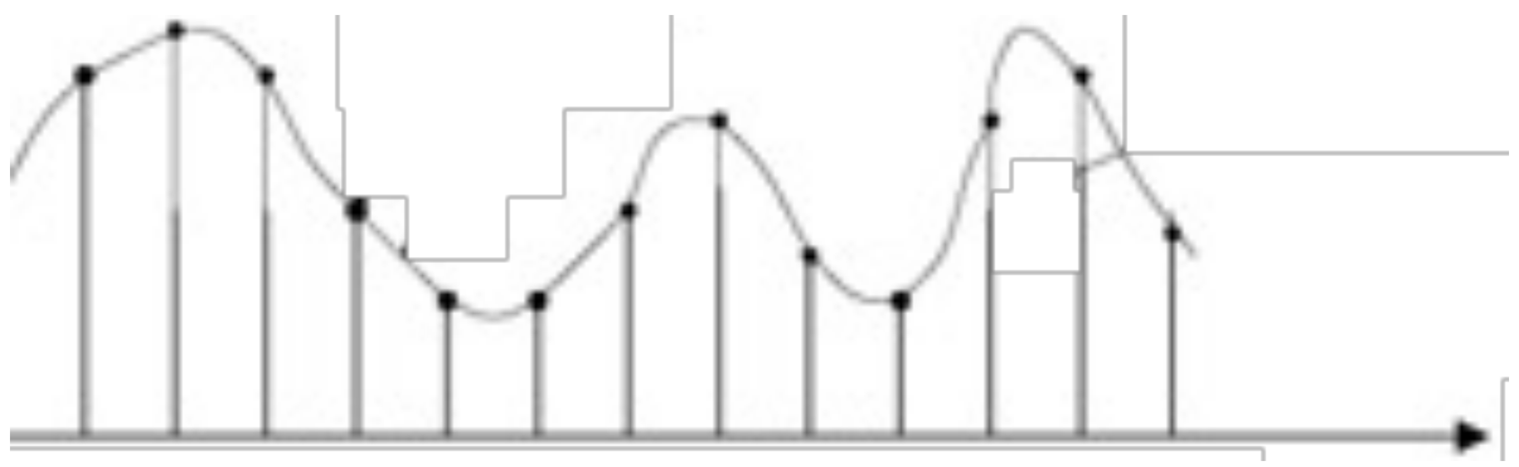
S such that E,S ⊨P

?

Physical – Dynamic controller



Synthesis

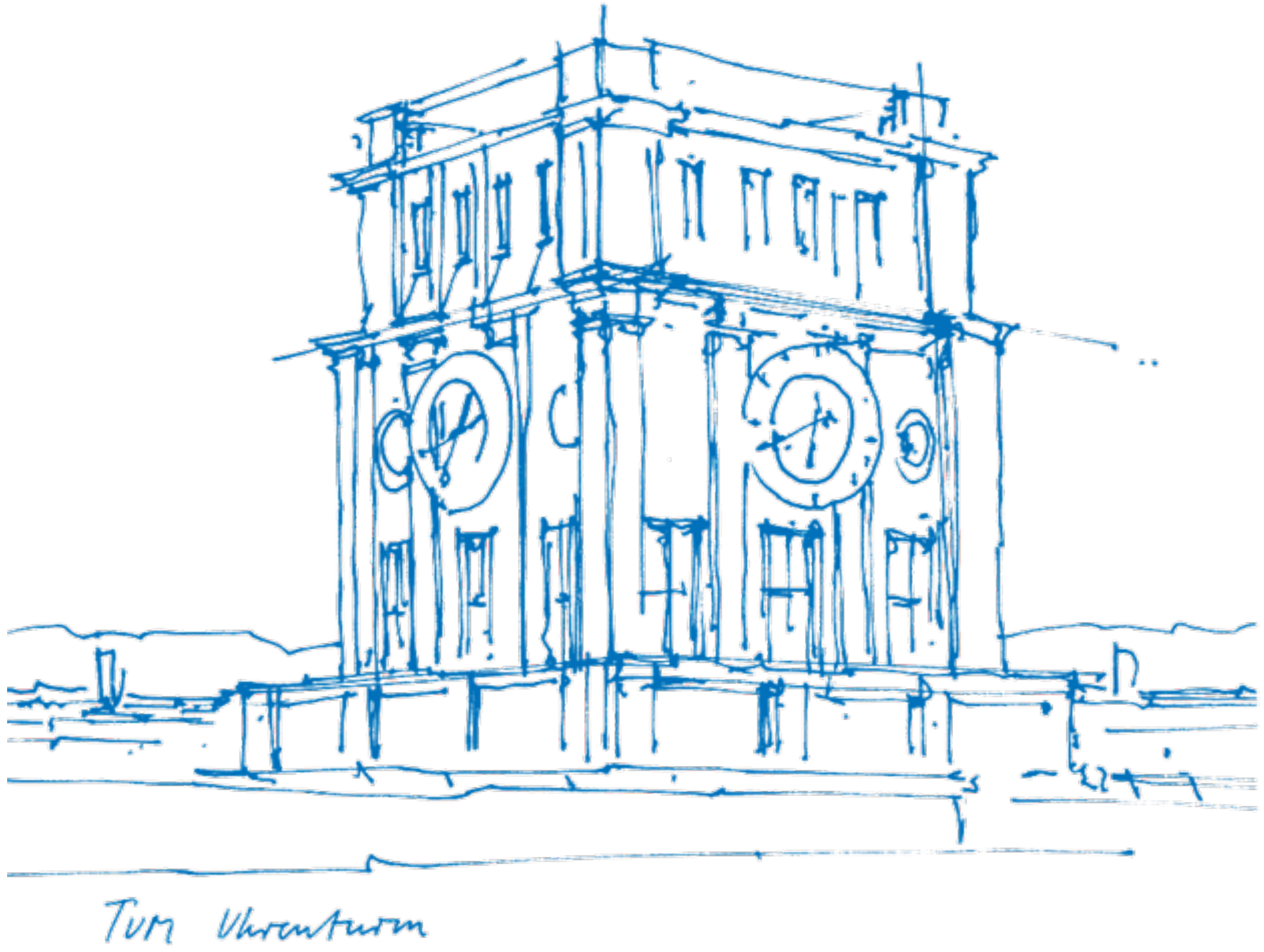


Continuous Signal

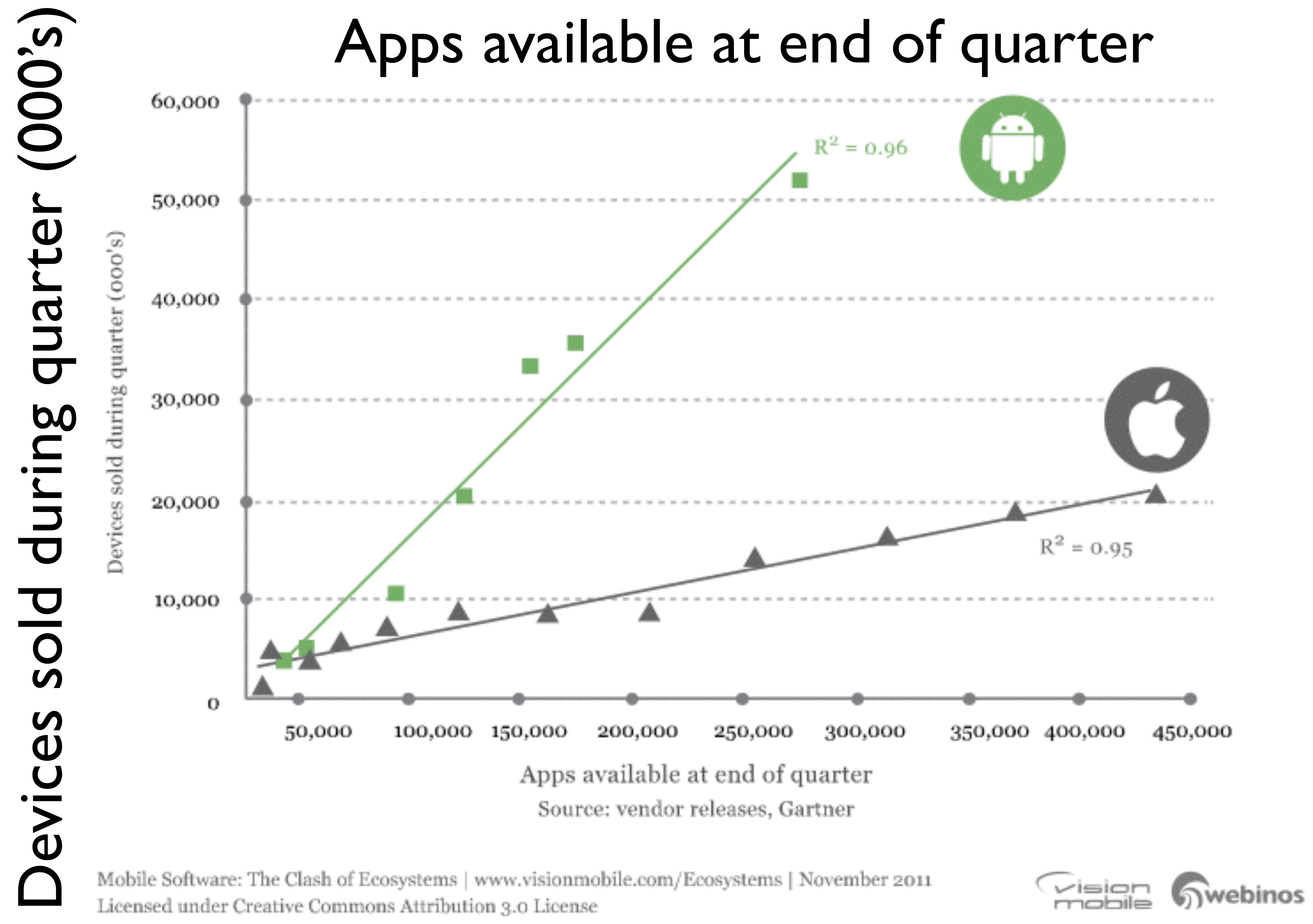
Case Study: Distributed Smart Space Orchestration

Marc-Oliver Pahl
Technical University of Munich
Chair for Network Architectures and Services

pahl@tum.de
http://s2o.net.in.tum.de/



Heidelberg
Karlsruhe
Stuttgart
Tübingen
München
4 Std. 11 Min.
410 km
Distributed Smart Space Orchestration
VSL middleware
Teaching
Blended
eLearning



open source™

How?

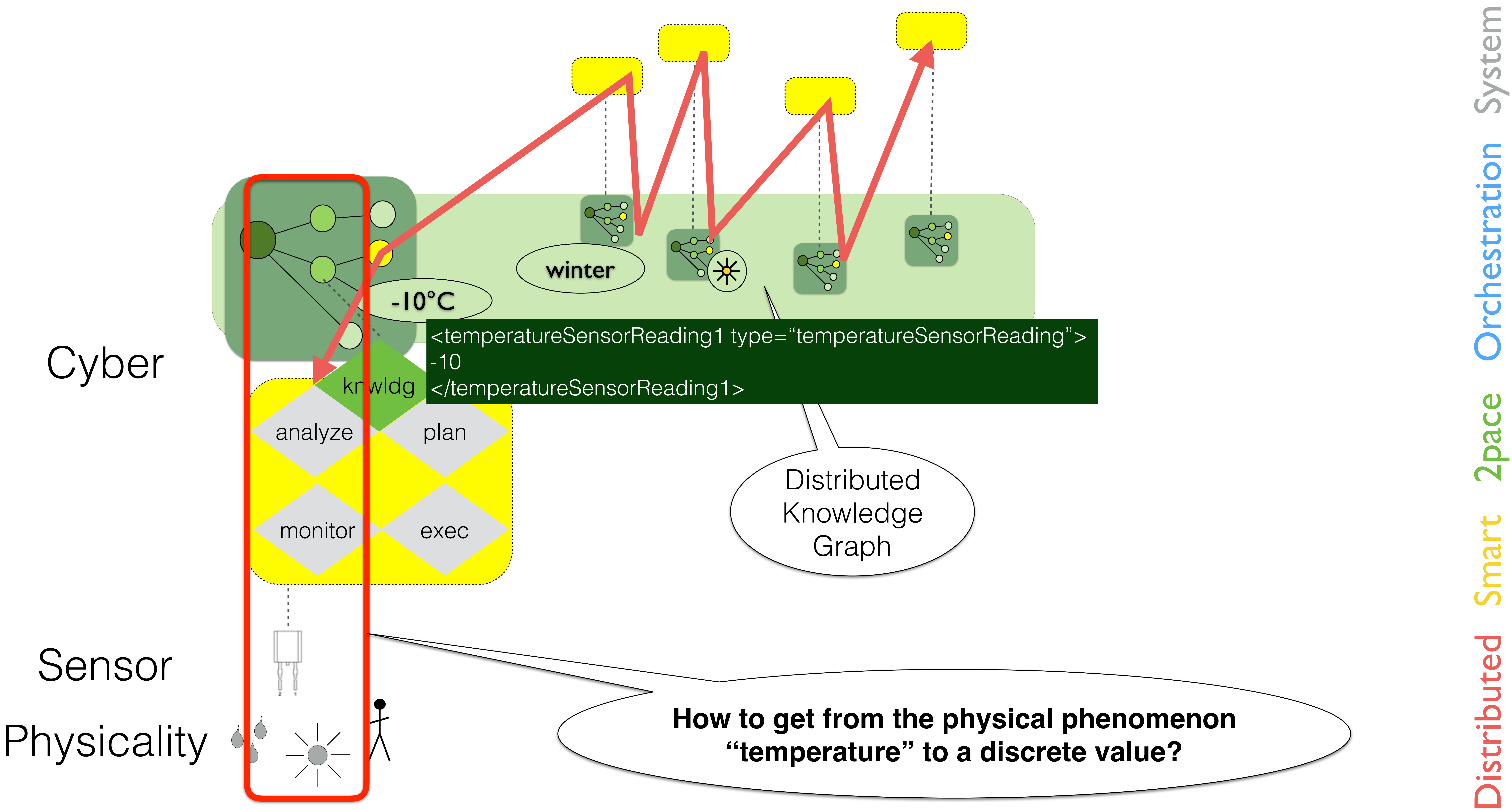
Smart Space

- A Physical Space that **contains Cyber-Physical Systems (CPS) and forms a CPS**.
- A Physical Space that contains Smart Devices* and is managed by software.

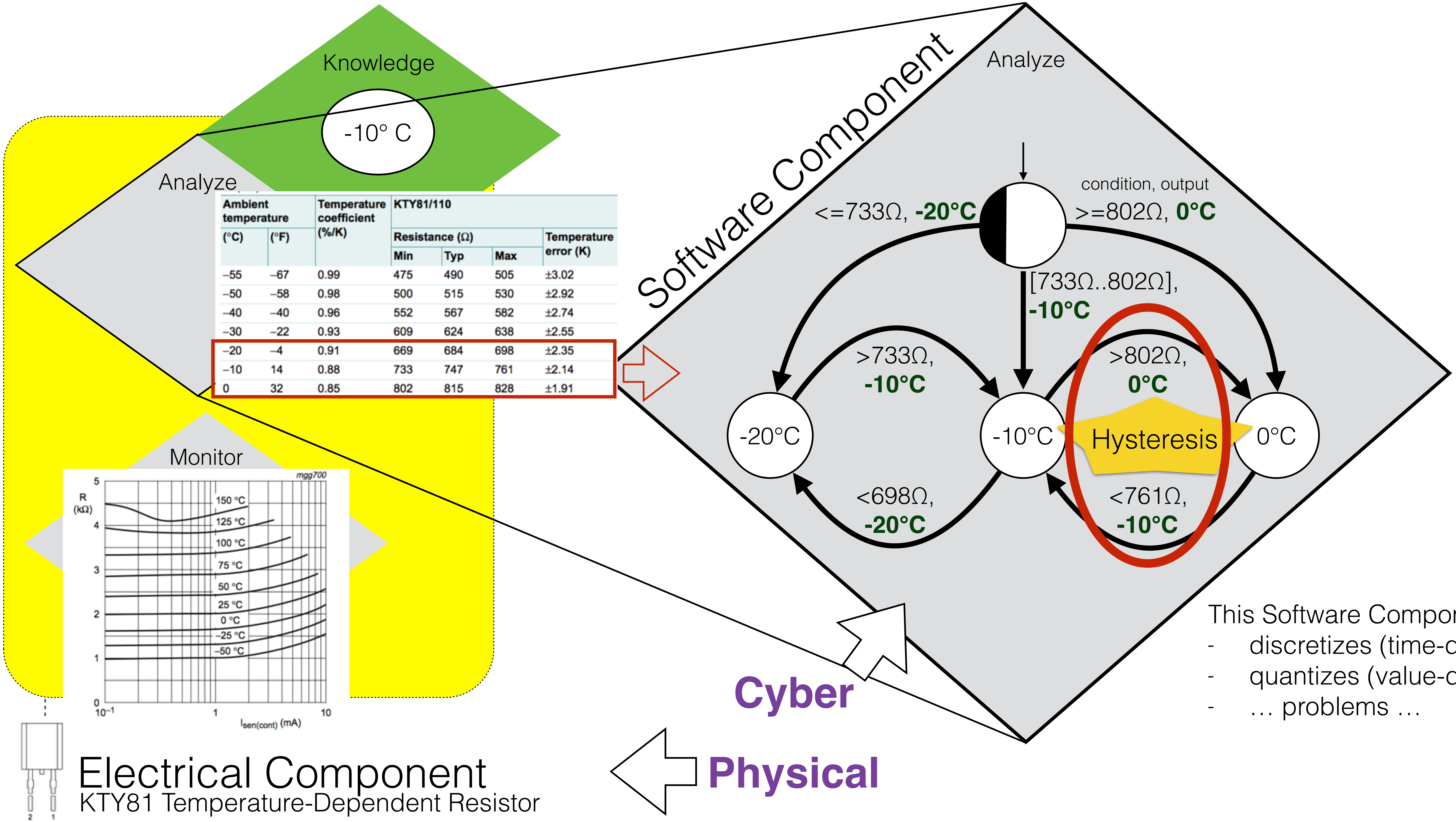
* embedded system that interact with their environments via sensors and actuators.

Some of my Challenges

- **Distributed** Entities
  - Sensors and Actuators
  - Computing Nodes
- **Interoperability** (How to bridge silos?) — modeling
- **Dynamic Reconfiguration** (How to integrate devices and Apps in a seamless way?) — modeling
- **Security**
  - **Safety** (How to ensure that a certain functionality is provided? e.g. factory automation) — modeling
  - **Privacy** (How to ensure that inherently collected data is only used as desired by the user?) — modeling
- **Usability** (How can “the crowd” develop Apps for complex Smart Spaces?)


Cyber
Sensor
Physicality
-10°C
winter
knwldg
analyze
plan
monitor
exec
<temperatureSensorReading1 type="temperatureSensorReading">
-10
</temperatureSensorReading1>
Distributed Knowledge Graph
How to get from the physical phenomenon "temperature" to a discrete value?
Distributed Smart 2pace Orchestration System

Adaptation Modeled as Labeled Transition System

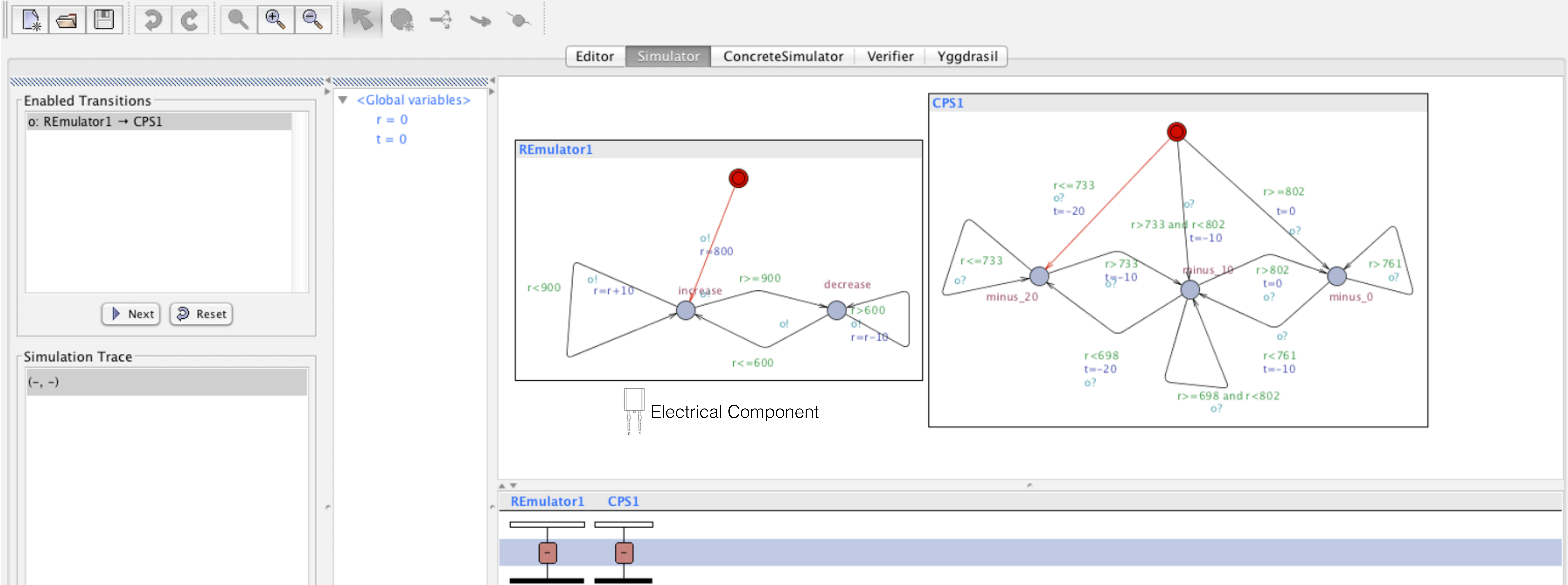


Ambient temperature (°C)	Ambient temperature (°F)	Temperature coefficient (%/K)	KTY81/110 Resistance (Ω) Min	Typ	Max	Temperature error (K)
−55	−67	0.99	475	490	505	±3.02
−50	−58	0.98	500	515	530	±2.92
−40	−40	0.96	552	567	582	±2.74
−30	−22	0.93	609	624	638	±2.55
−20	−4	0.91	669	684	698	±2.35
−10	14	0.88	733	747	761	±2.14
0	32	0.85	802	815	828	±1.91

This Software Component:
- discretizes (time-domain)
- quantizes (value-domain)
- … problems …

http://www.nxp.com/documents/data_sheet/KTY81_SER.pdf

Distributed Smart 2pace Orchestration System



Details:

https://youtu.be/WC4vpQt3mQs

Screencast from the Uppaal integrated tool environment for modeling, validation and verification of real-time systems modeled as networks of timed automata.

Distributed Smart 2pace Orchestration System

Introduction
3 Research Examples

CPS and IoT

Linking Physicality and Computation
A Practical CPS Example: an Infusion Pump
Discussing Challenges and Opportunities of CPS
Wrap-Up

Vint Cerf
Google
30 August 2016

Vincent
Cerf

Screen shot from: “Alex Gibney, Zero Days (2016)”

What are Cyber-Physical Systems?

- Robots, mechanical appliances, cars, ….
- What about automatic stock trading programs?
- Any programmable device? A massage chair? Microwave oven?
- What about Java-Script, Java, Python and other Web-languages?
- Where do sensor networks fit in? Their data may affect/control behavior.

It’s the Software, Stupid!

- Bugs
- Updates
  - Source? Integrity? Communication path(s). Confirmations?
- (Strong) authentication of parties
  - Who/What can control? Configure? Receive data?
- Scaling up configuration
  - The 200 device house

Machine Learning and Artificial Intelligence

- The artificial idiot
  - Learns the wrong things
    - Chatbot story and Thermostat that tries to be helpful
  - Configures the neighbors devices into your house system
- Ephemeral or Episodic Access
  - Fire Department, Police Department ,Emergency Medical Response

Access Control and Security

- Residents vs Guests

  Differentiating among residents and among guests

  What happens after guests leave?

- Parents vs Kids
- Shop floor operations vs Administrative staff and visitors

Long Term Considerations

- Digital picture frames
  - Storage corruption (data and executables) and Power supply failure
- Custom (entertainment, lighting) systems
  - Bugs/reprogramming and Wi-Fi or other protocol dependencies
- Programmable light switches
  - Battery format dependency and Programming user interface (blech)

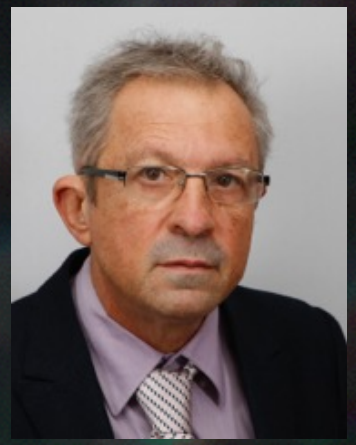
Major considerations

- Reliability and Ease of use
- Safety (https://arxiv.org/pdf/1606.06565v2.pdf)
- Security
- Privacy
- Autonomy (Your house should not stop working just because Internet access is down)
- Interoperability (of ensembles)

Introduction
3 Research Examples
CPS and IoT

Linking Physicality and Computation

A Practical CPS Example: an Infusion Pump
Discussing Challenges and Opportunities of CPS
Wrap-Up



Joseph
Sifakis

Screen shot from: “Alex Gibney, Zero Days (2016)”

Cyber-physical Systems

Cyber-physical systems

- refer to the next generation of engineered systems requiring tight integration of computing, communication, and control technologies
- tightly combine the continuous dynamics (systems of differential equations) with the discrete dynamics of cyber systems (SW+HW).
- are important in overcoming many challenges in energy, environment, transportation, and health care e.g., to achieve stability, performance, reliability, robustness, and efficiency

The emergence of Cyber-physical systems has been enabled by

- micro-scale and nano-scale design and fabrication technologies e.g., sensors, actuators, and processors that are small, cheap, fast, and energy efficient
- advances in system software, from high performance computing systems to real-time embedded systems
- wireless networks making feasible connectivity of mobile nodes.

Cyber-physical Systems – Extending the 3D-printing Paradigm

electrek

SEARCH

HOME BIKES CARS ENERGY SOURCES

MAY 24

TSLA: 217.91 1.69

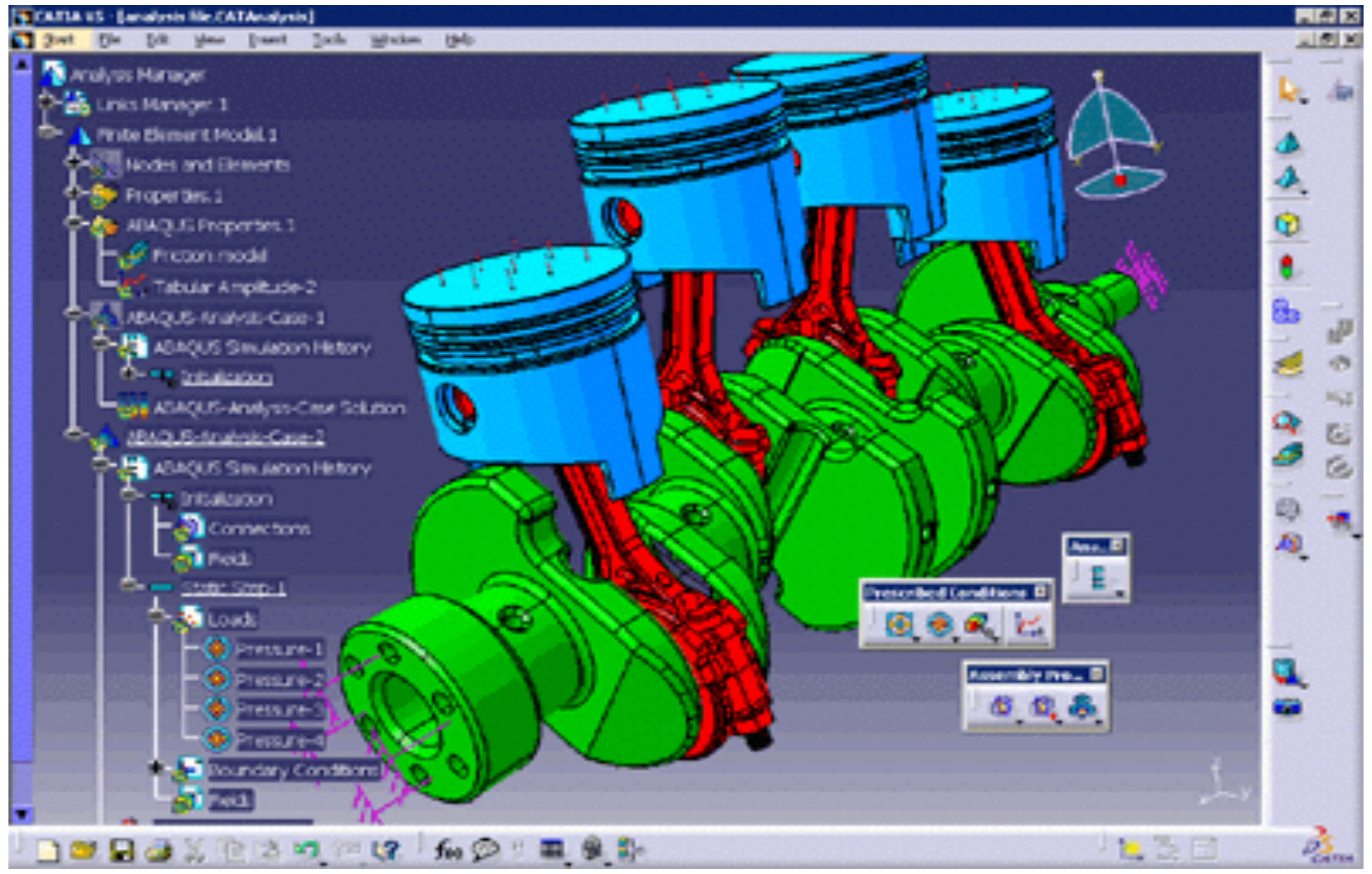
is not a 9 to 5 job'

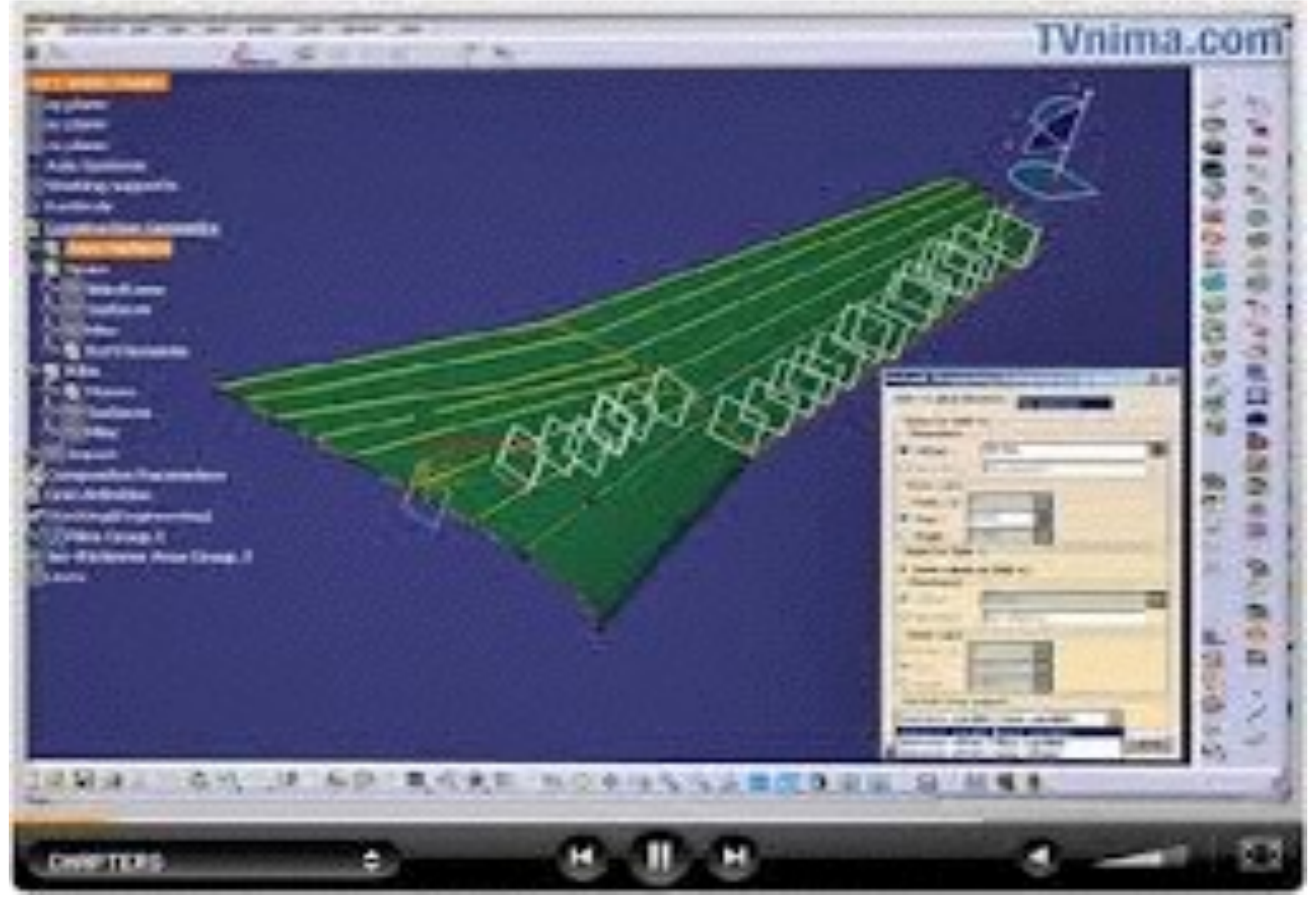
range options (Update: Base 6 sec 0-60 and

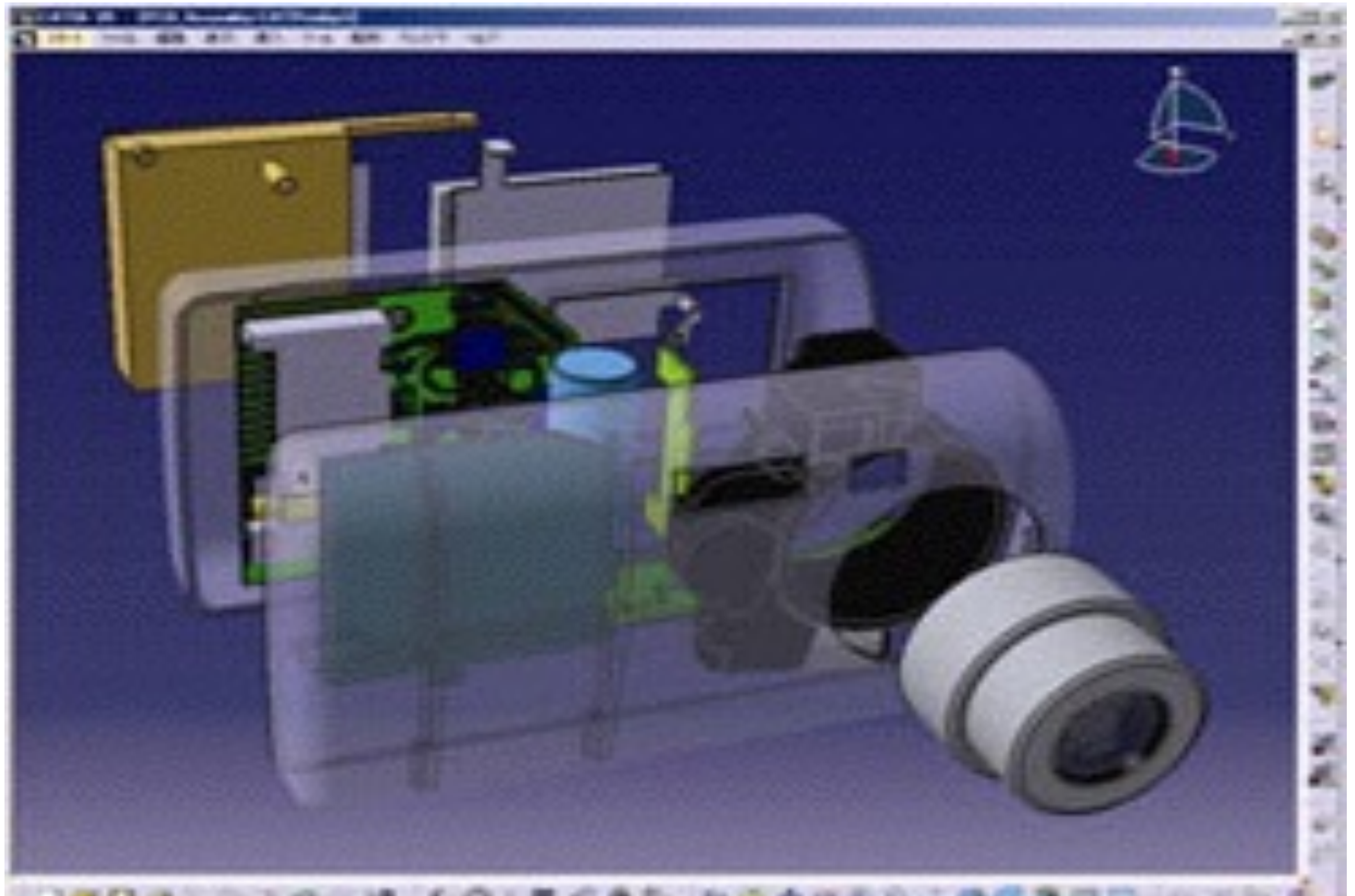
Airbus unveils the world's first 3D printed electric motorcycle: Light Rider

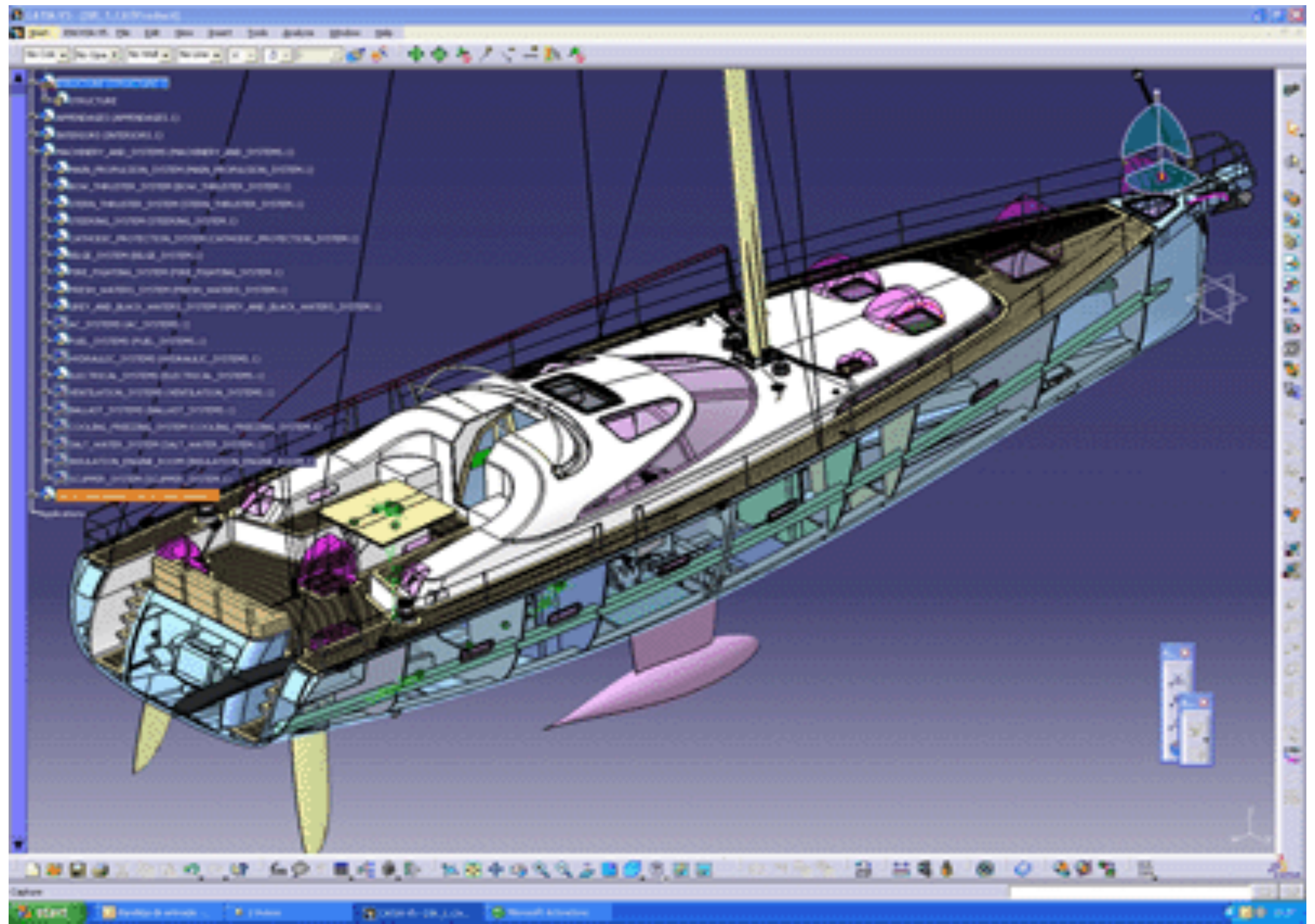
Cyber-physical Systems – Extending the 3D-printing Paradigm

Building smart systems as the composition of components whose cyber and physical parts are concurrently designed

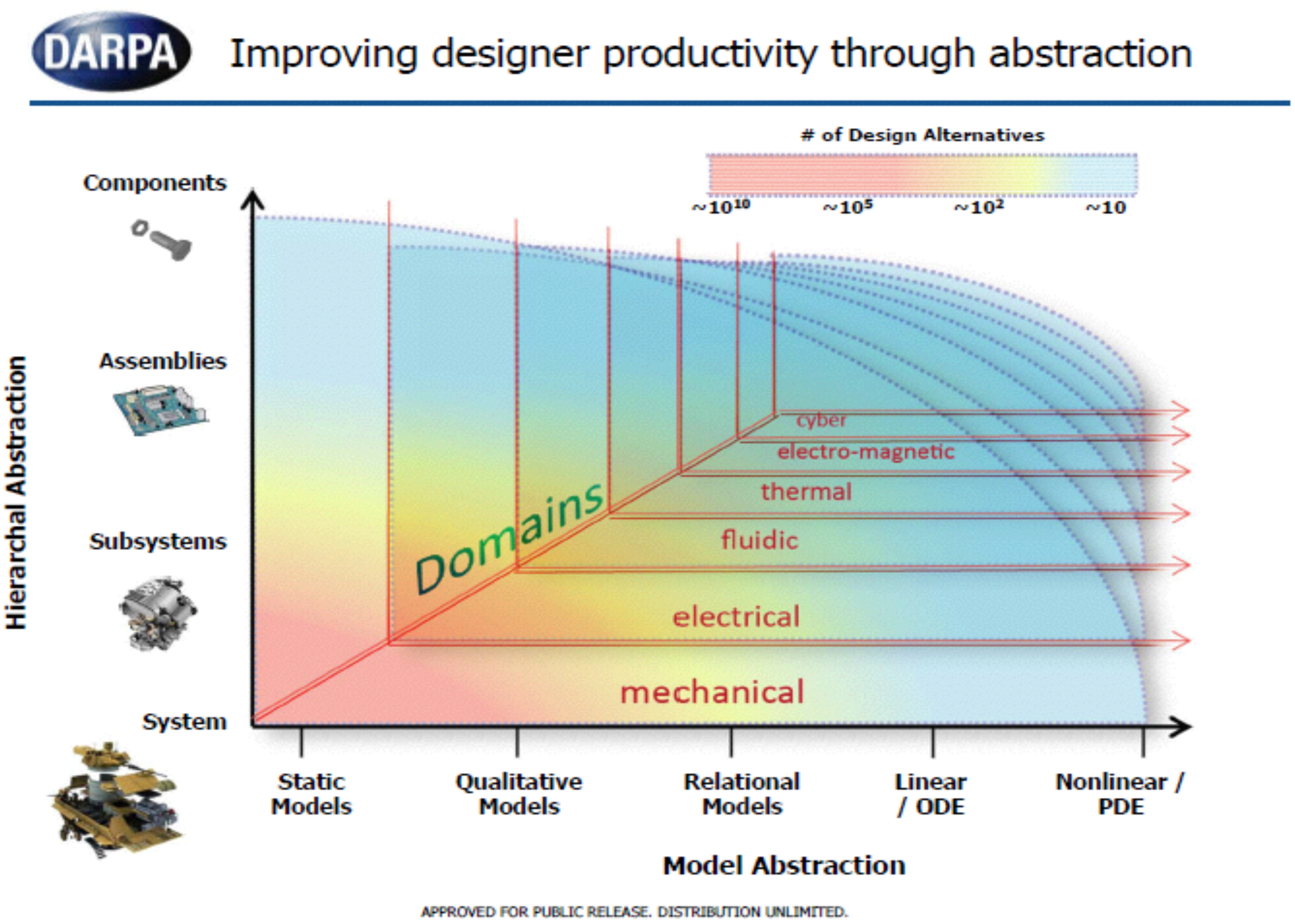




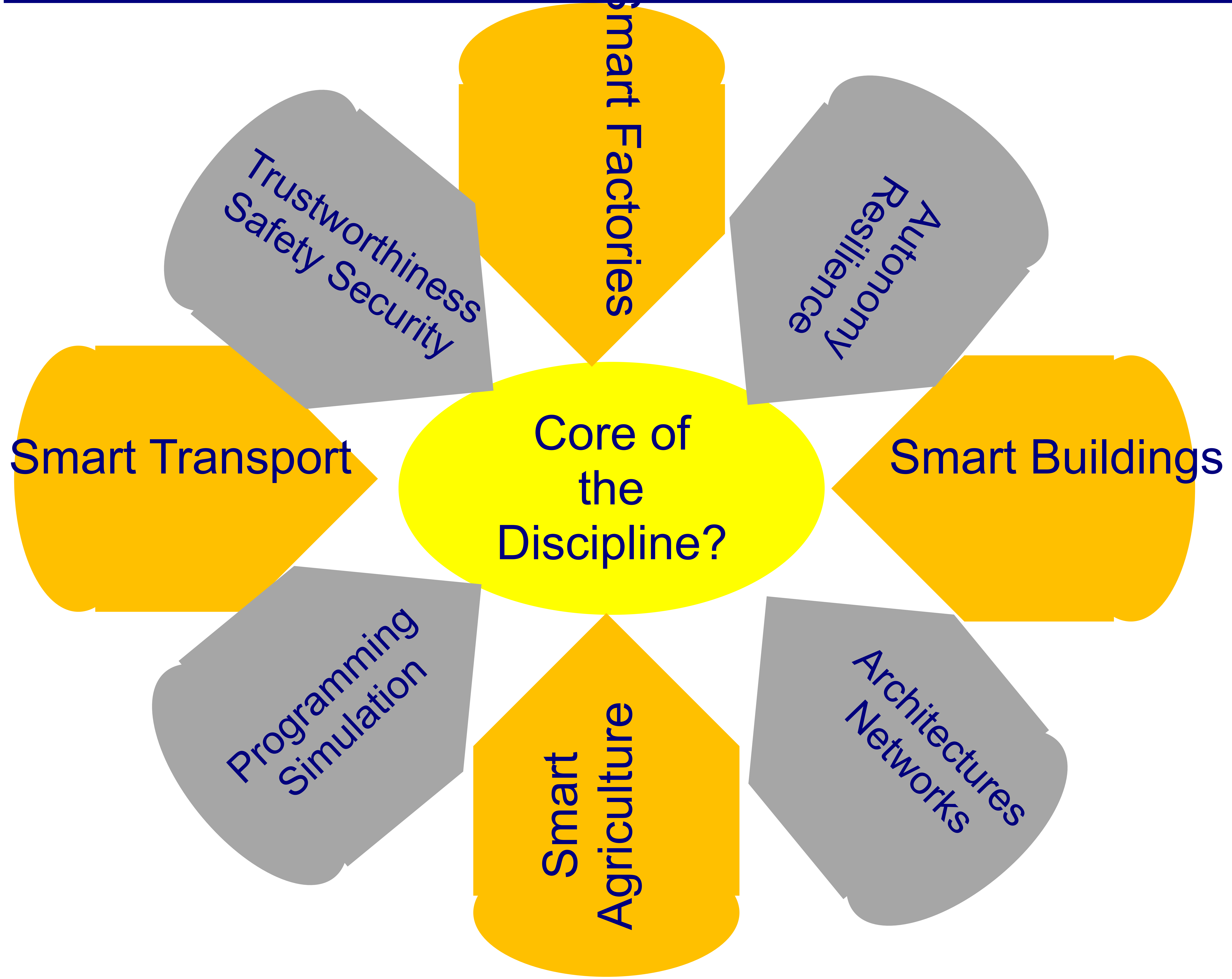




Multiscale and Multidomain Model Integration



Is CPS a Discipline in its Own Right?



The CPS Discipline

The Core of the Discipline

- Theory: Linking Physicality and Computation, in particular unification of modeling frameworks
- Engineering: Rigorous Design of CPS, in particular building CPS systems from CPS components

CPS Design – Three Main Problems

Modeling of CPS

- Structural Equational Modelling of Physical Systems
- Semantic issues
- Hybrid Models for Cyber Physical Systems

Discretization Techniques for Executability

- Discretization Algorithms
- Dataflow Models with Discrete Events

Execution and Implementation Techniques

- Modular Code Generation
- Co-simulation techniques
- Direct code generation from networks of physical components

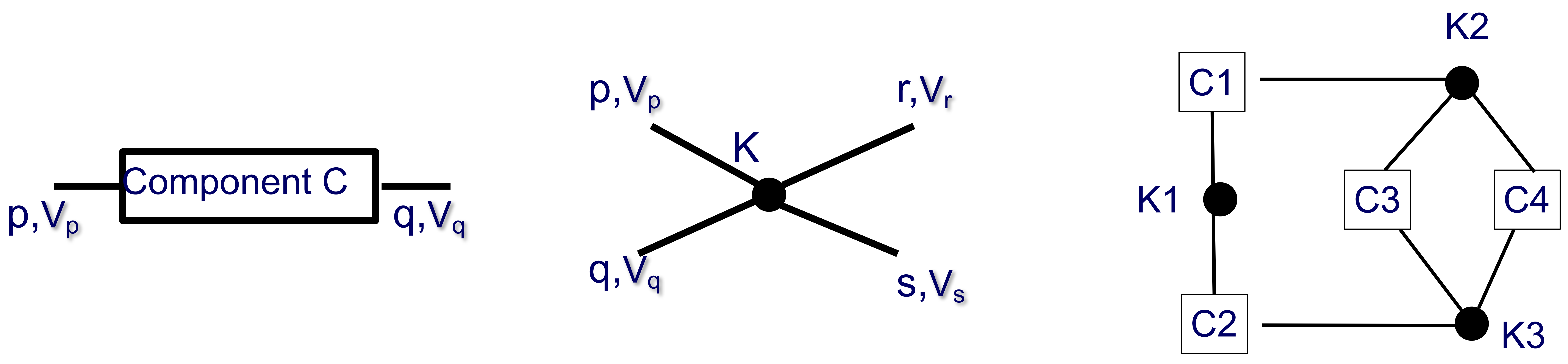
- Modeling CPS
- Discretization
- Execution and Implementation

OVERVIEW

Modeling CPS – Structural Equational Modeling

- A network of physical components is built from a limited number of types of components C and connectors K.
- Each component has ports p, q, r and associated general state variables V_p, V_q, V_r
- The network is a hypergraph: vertices are component ports and edges are connectors (sets of ports)
  - Component C has a characteristic equation of the form $E_C(V_p,V_q)$.
  - Connector K relating ports p,q, .. ,r has equations of the form $E_K(V_p,V_q ,..., V_r)$ – equations may be constrained in general.

Equational compositionality : The meaning of a network with sets of components $\{C_i\}_{i\in I}$ and set of connectors $\{K_j\}_{j\in J}$ is the union $\{E_{Ci}\}_{i\in I} \cup \{E_{Kj}\}_{j\in J}$.

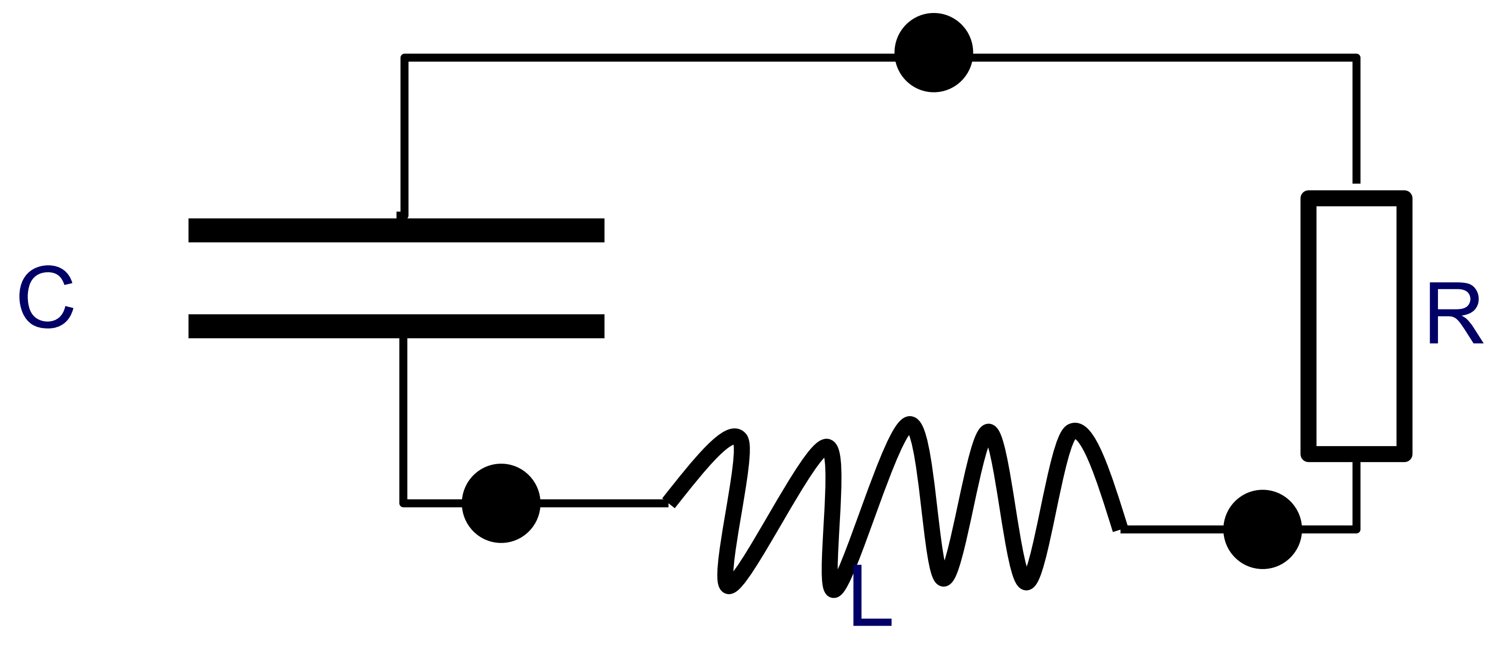


Component C: $E_C(V_p,V_q)$

Connector K: $E_K(V_p,V_q,V_r,V_s)$

Network of physical components

Modeling CPS – Structural Equational Modeling



$C\, v_C' = i_C$

$v_R = R\, i_R$

$L\, i_L' = v_L$

$v_C = v_{C1} - v_{C2}$

$v_R = v_{R1} - v_{R2}$

$v_L = v_{L1} - v_{L2}$

$i_C = i_R = i_L$

$v_{C1} = v_{R2}\ ,\ v_{C2} = v_{L1}\ ,\ v_{L2} = v_{R1}$

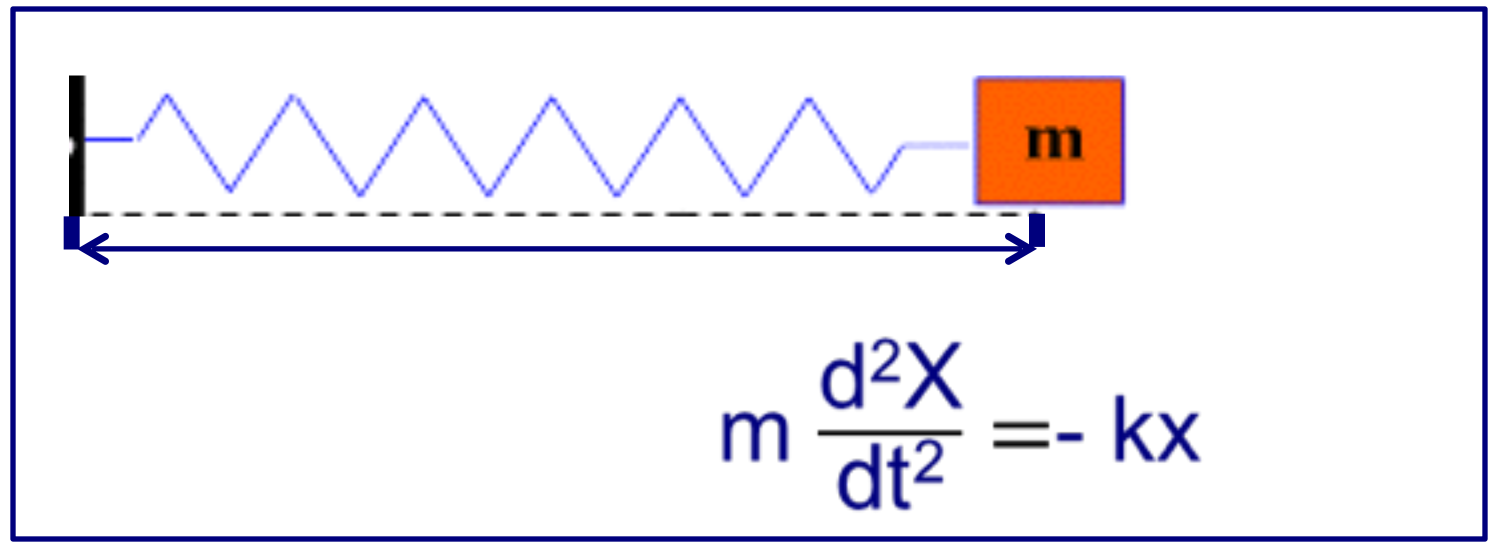
Open Problems:

- Given a physical system described as a network of interacting elements, is there a systematic approach to get faithful equational models?
- Is there a concept of faithfulness of modeling and how it can be formalized ?
- What is the value of techniques such as Bond Graphs, Linear Graphs?
- Modular specification – meaningful composition of equational specifications
- Abstraction and reduction techniques for equational models

Modeling CPS – Continuous Data Flow Models

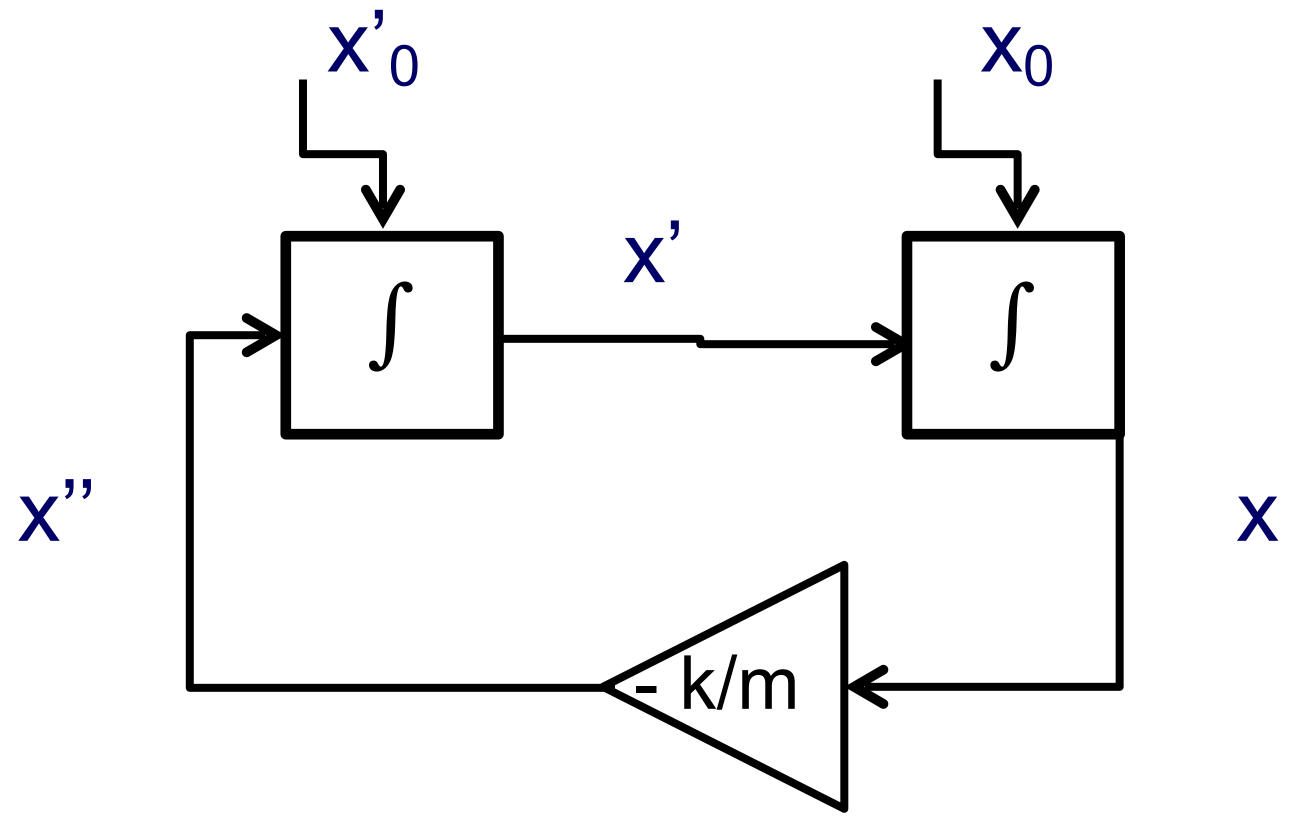
From an equational specification of a PS, a Data Flow Network (block diagram) may be derived

- The operators transform input flows into output flows – they are intrinsically parallel
- Integrators have an additional initialization input that determines the initial value of the integrated variable.
- Translation into data flow networks is the first step toward discretization

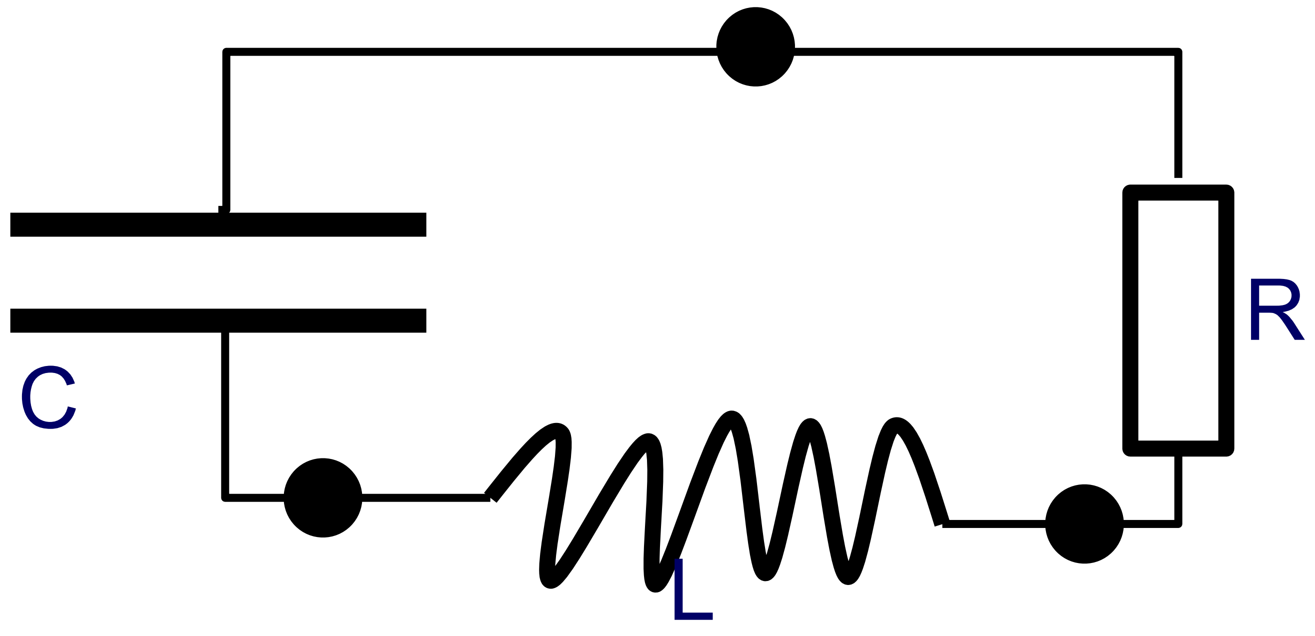


$$m \frac{d^2X}{dt^2} = - kx$$

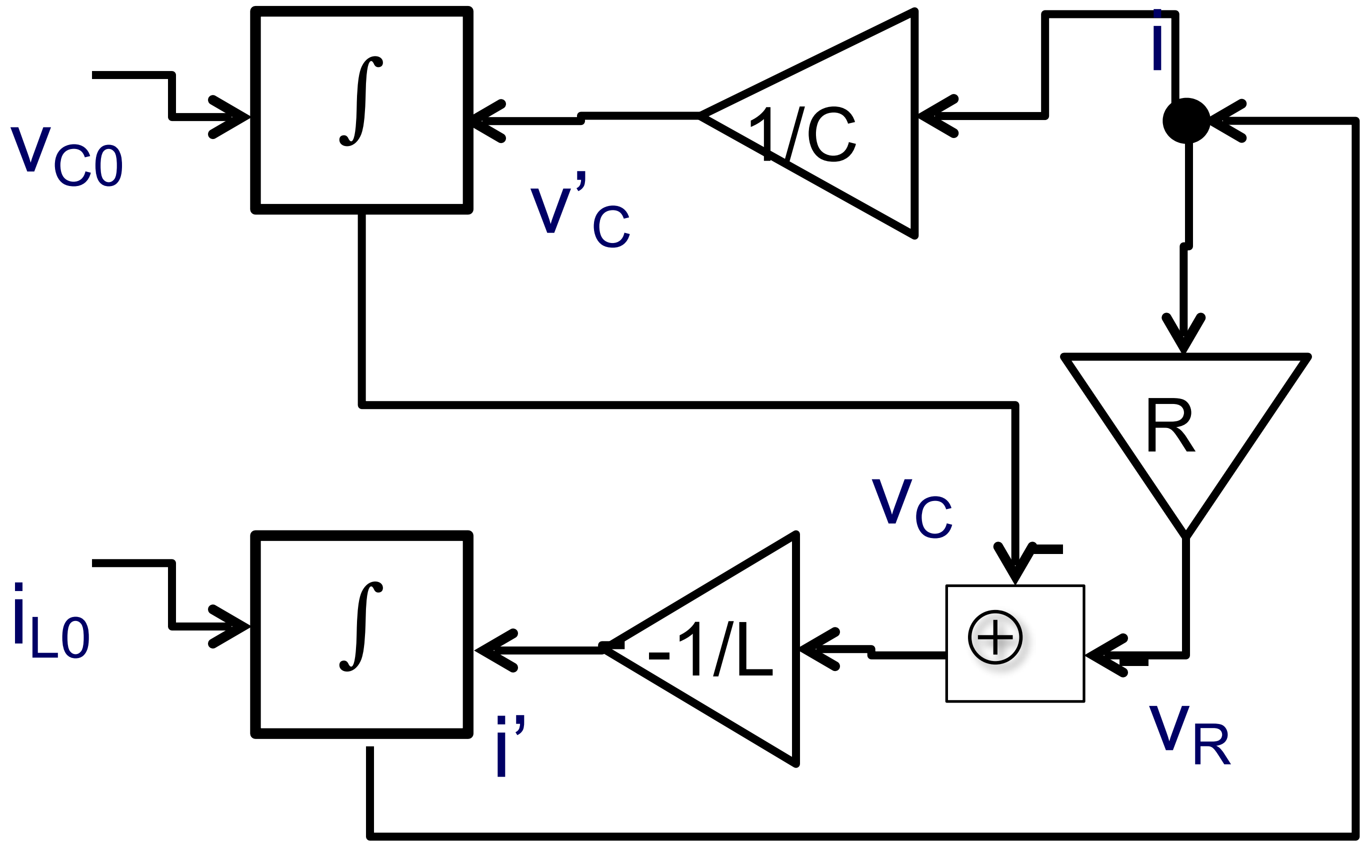
Mass-spring system



Continuous data flow model



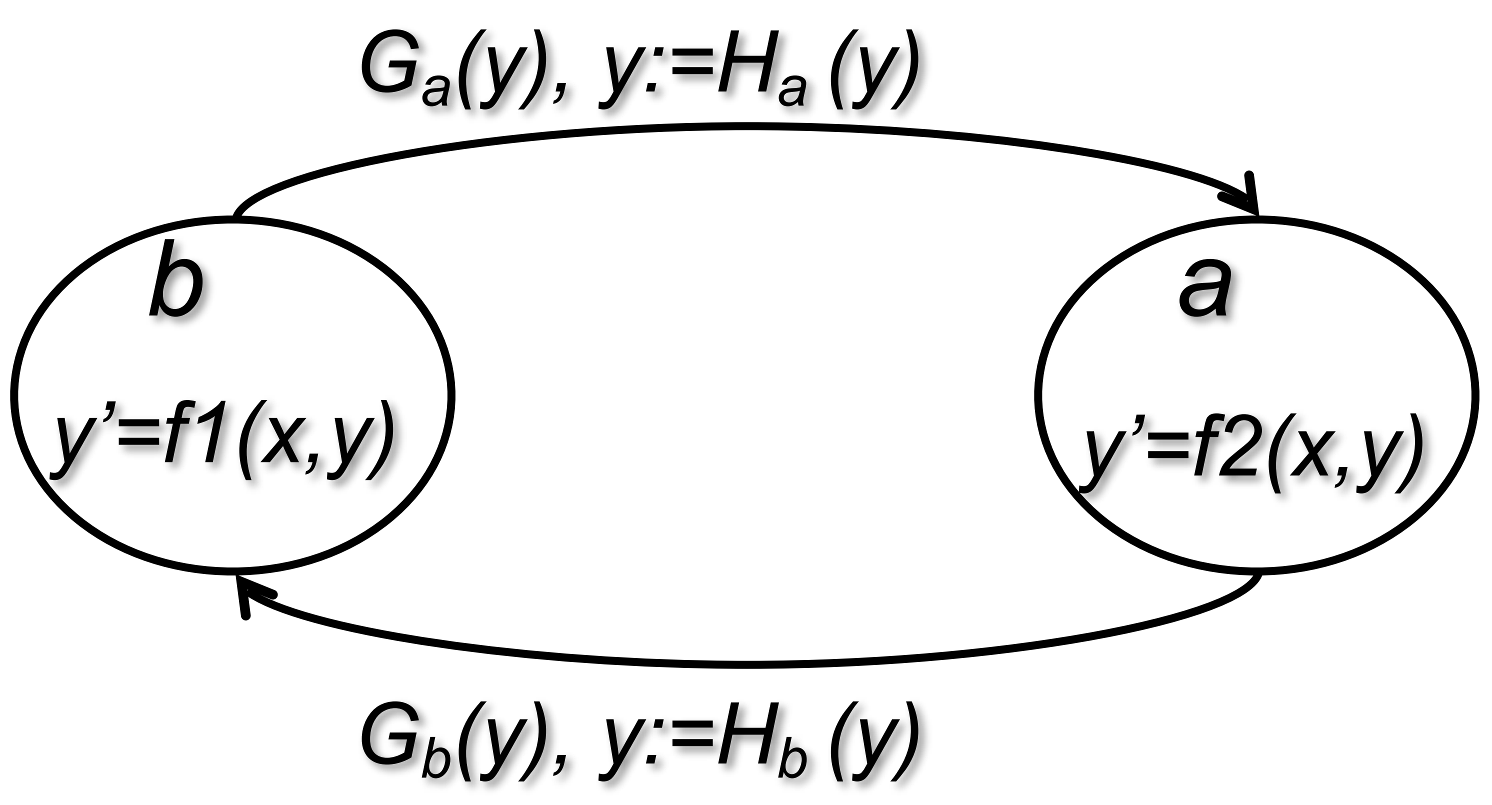
RLC Circuit



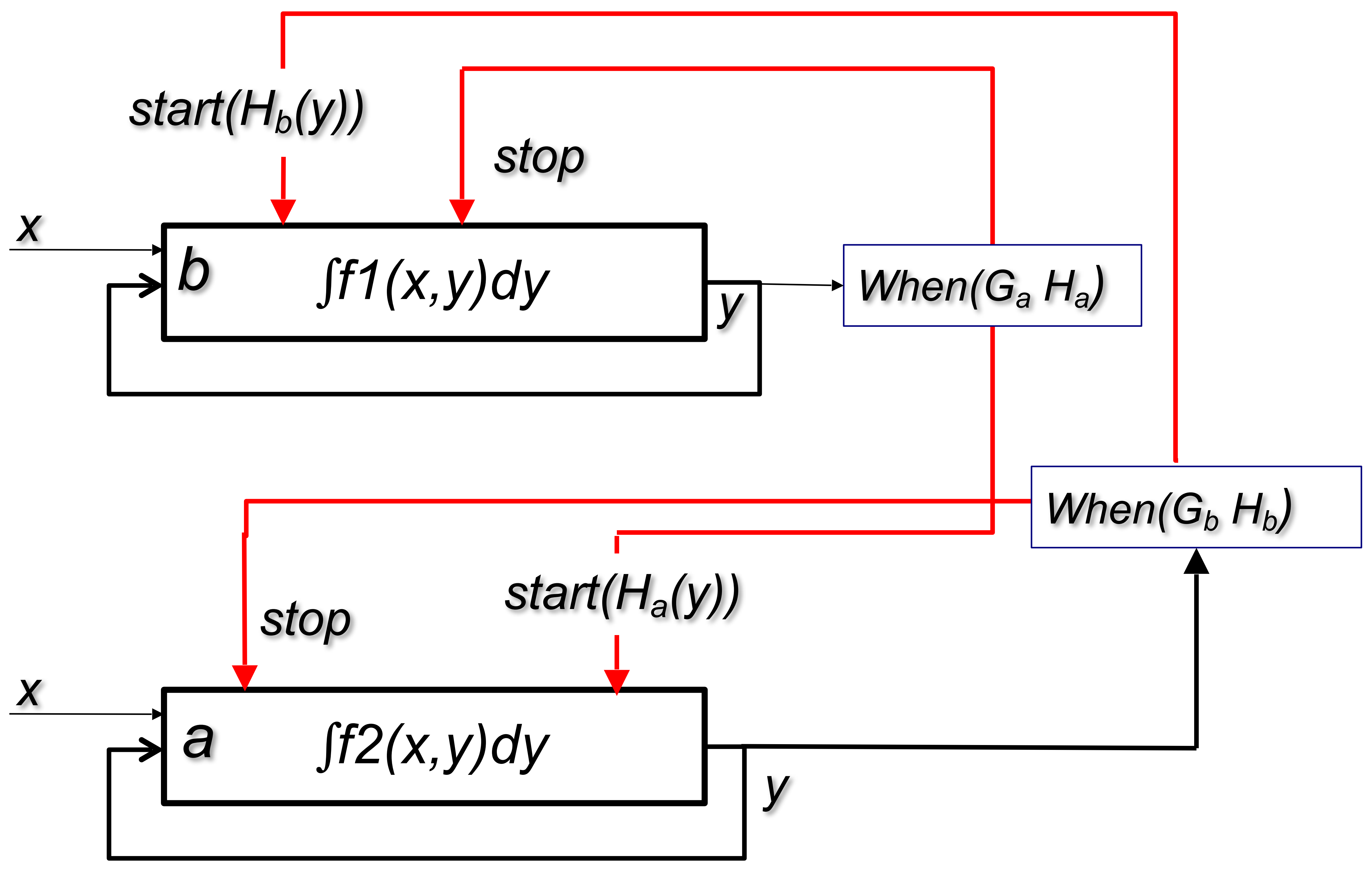
Continuous data flow model

Modeling CPS – Hybrid Models

Hybrid Automata vs. Hybrid Data Flow Models



HYBRID AUTOMATON



HYBRID DATA FLOW MODEL

OVERVIEW

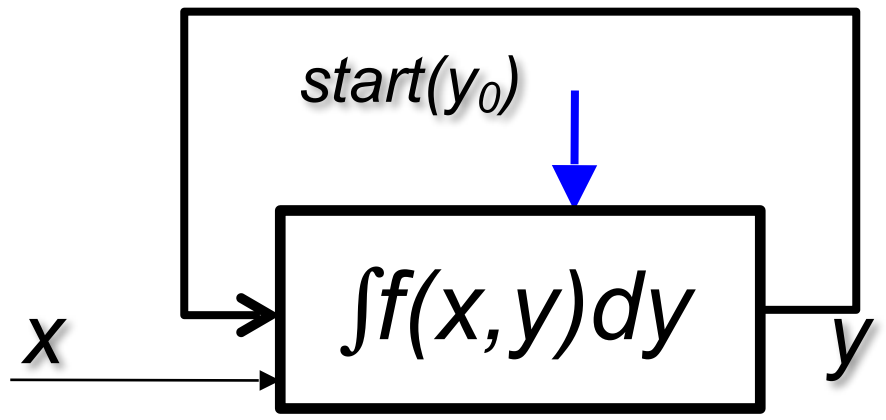
- Modeling CPS
- Discretization
- Execution and Implementation

Discretization – Discrete Data Flow Models

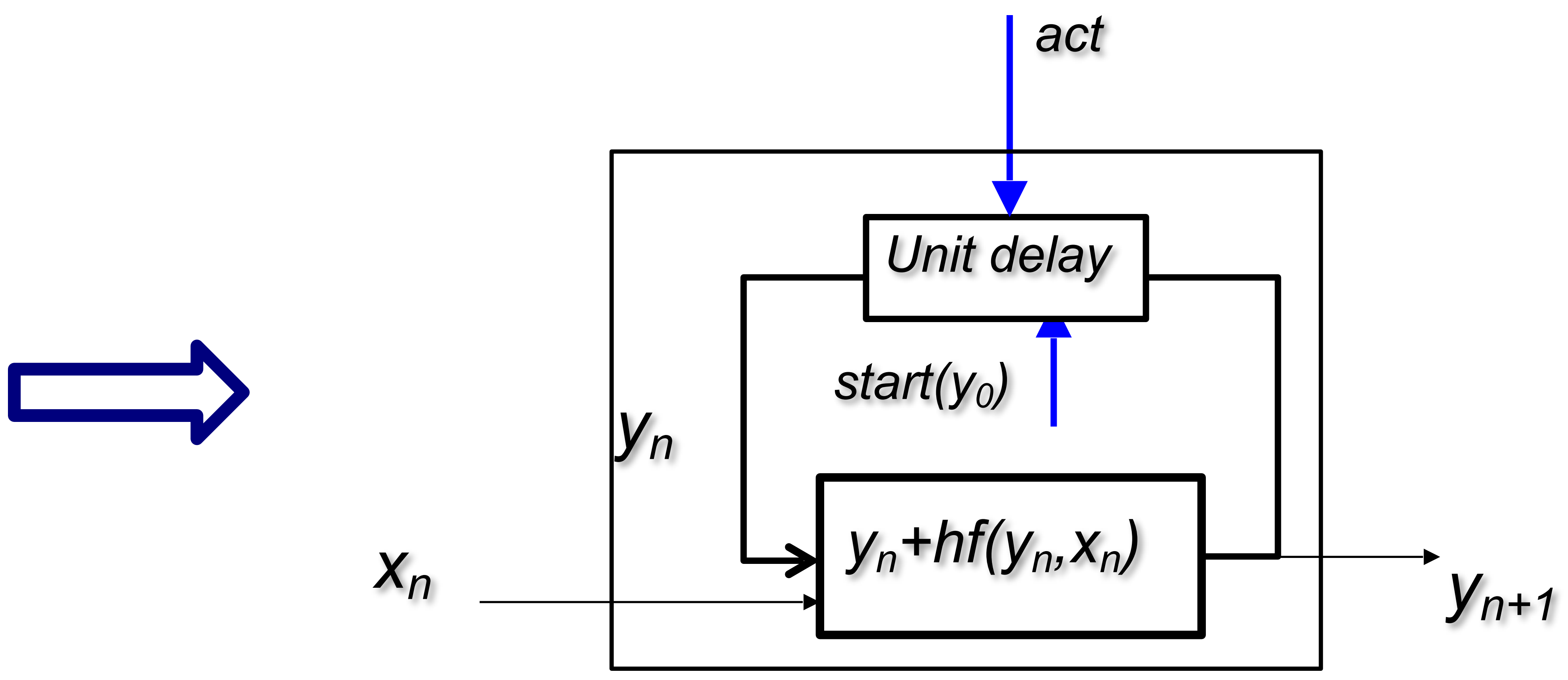
- By applying the Euler method, the solution of
  $y'(t)=f(t,y(t))$ with $y(t_0) = y_0$
  can be approximated by the iterative computation
  $y_{n+1} = y_n + h\, f(t_n, y_n)$ involving a time sequence $t_{n+1} = t_n + h\, t_i$

- A discretized system can be obtained by replacing each integrator by a discrete iterative program that applies a specific discretization technique.

- A discrete data-flow component has input and output data ports and an event port *act*. The event *act* triggers the cyclic computation of *f*. It plays the role of a *logical clock*.



Continuous dataflow model for
$y'=f(x,y)$ with $y=y_0$



Discrete dataflow model

Discretization – Discrete Data Flow Models

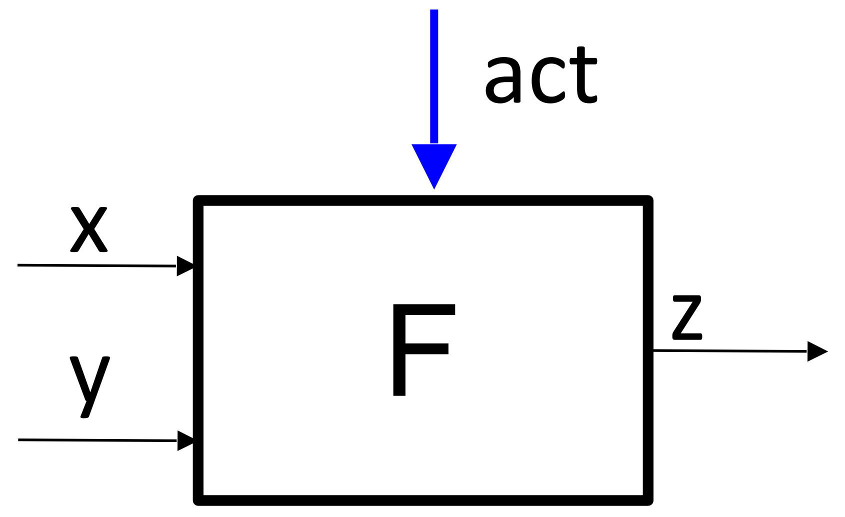
Are purely functional and do not directly support discrete events and non determinism e.g. Lustre, Simulink

Discrete data flow components:

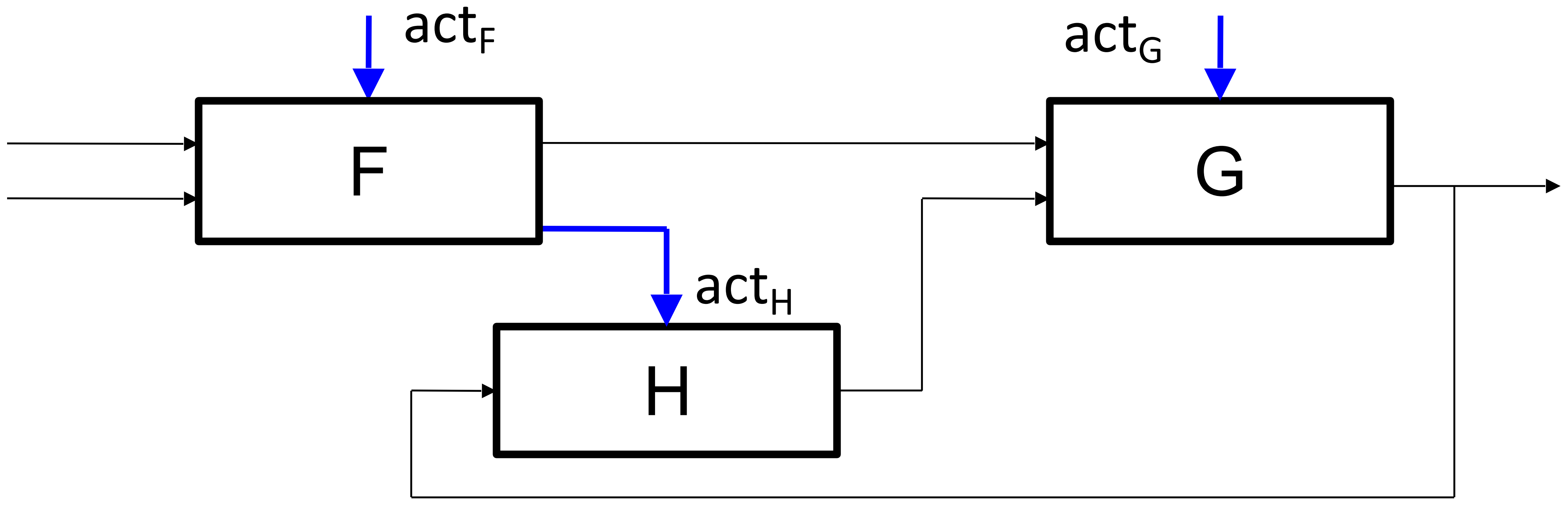
- have input and output data ports and a discrete event *act* .
- are characterized by a function F that transforms cyclically input data values into output values.
- the event *act* plays the role of a logical clock. At **each step** *t* the inputs *x* and *y* are updated and an output *z* is produced -- *z(t) =F(x(t),y(t))*

Discrete data-flow network:

- the interconnection of discrete data flow components.
- data output ports of a component are connected to data input ports of other components.
- *act* events can either external inputs or generated by using specific functions that generate events from data streams.



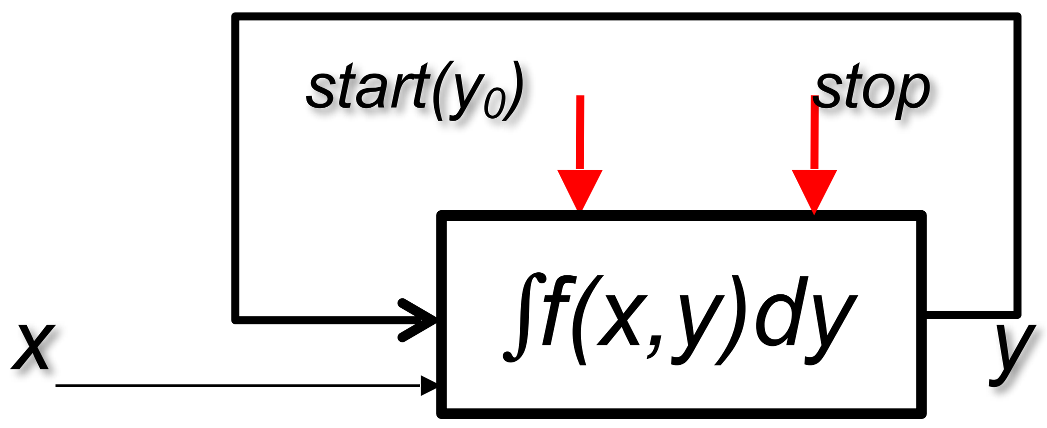
Discrete dataflow component



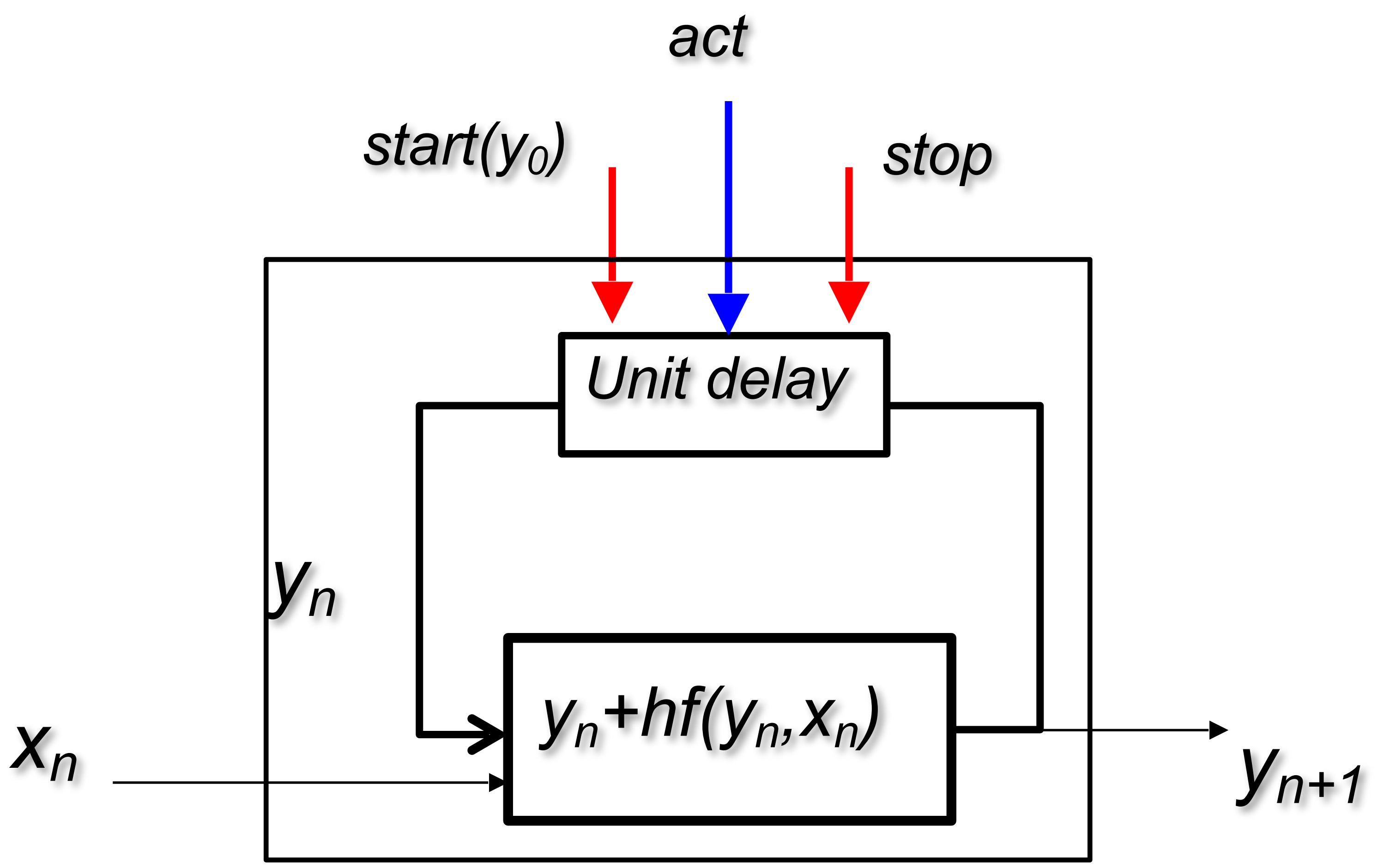
Discrete data flow network

Discretization – Discretized Hybrid Data Flow Models

- Are obtained by replacing each integrator in a hybrid data flow network by a discrete iterative program that applies a specific discretization technique.
- Discrete data flow components have in addition to *act* the external events *start* and *stop*.



Hybrid data flow model



Discretized hybrid data flow model

Discretization – Discretization Algorithms

Find easy-to-check criteria for executability:

- the model can be assigned a *causality* relation in a unique manner
- the model does not involve closed loops without integrators.

Learn from numerical analysis techniques. Numerical solvers combine different algorithms:

- some numerical integration schemas for stiff problems,
- of different consistency order, with the ability to increase the order according to the local error,
- some local error estimator, usually using formulae embedded in the above schema,
- a time step size adaptation heuristics that keeps the local error below the tolerance, and
- some event detection and “hot” restart mechanism to deal with reset equations.

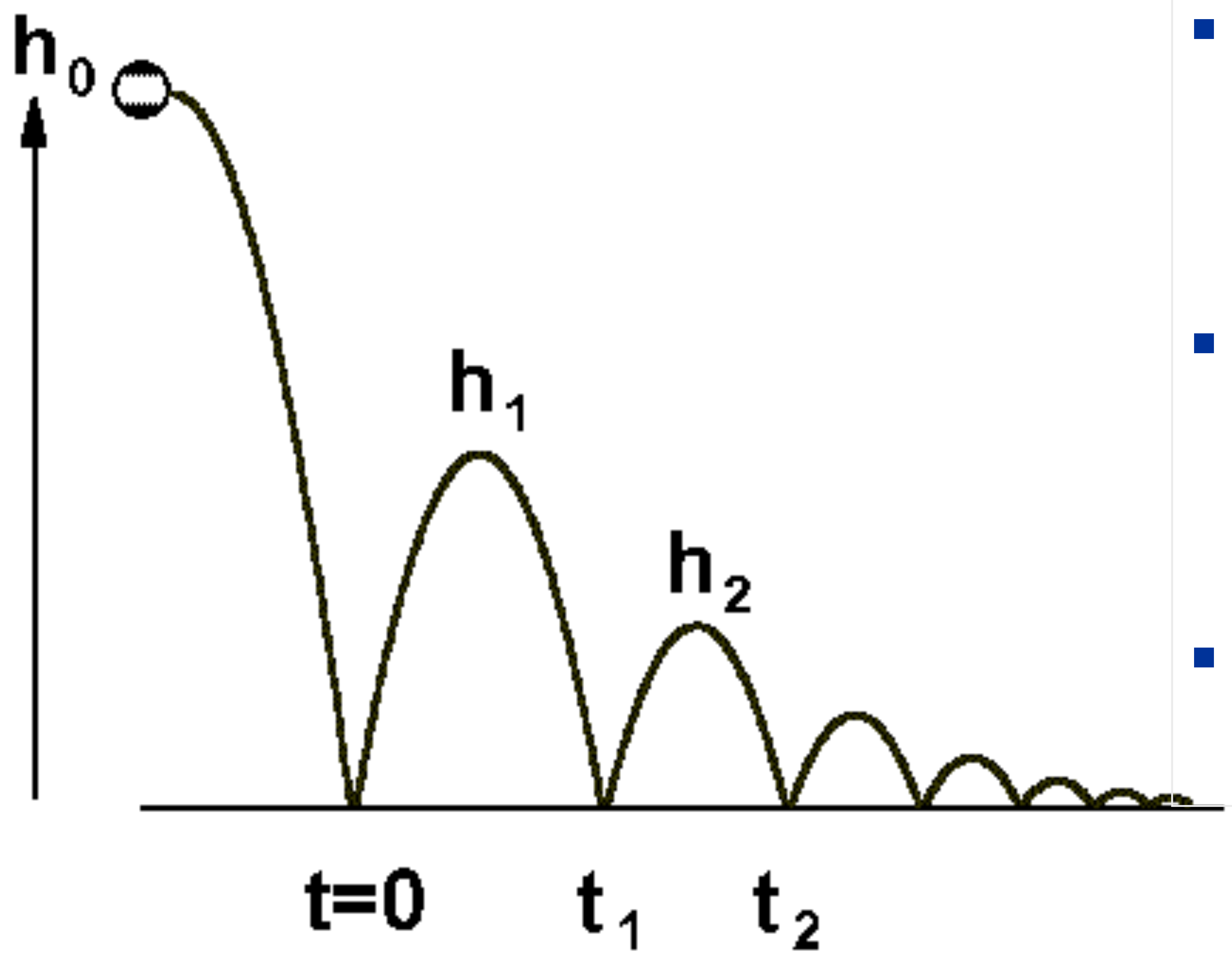
Define notions:

- what can be a measure of quality of approximation?
- how can be defined a notion of safety of approximation -- we do not discard when integrating critical discrete events

Discretization – Zeno behavior

Zeno behavior: existence of converging sequences of discrete events

- For a ball falling of an attitude h_0 and losing a percentage of its speed due to for non elastic shock with the ground
- The physical model is described by : $v' = -g$, $x'=v$, $v_{i+1}=0.8\ v_i$, $t_{i+1}-t_i = (2/g)v_i$
- $\lim_{t_n \to \infty} (t_n-t_0)$ is bounded, so time will not exceed the Zeno point

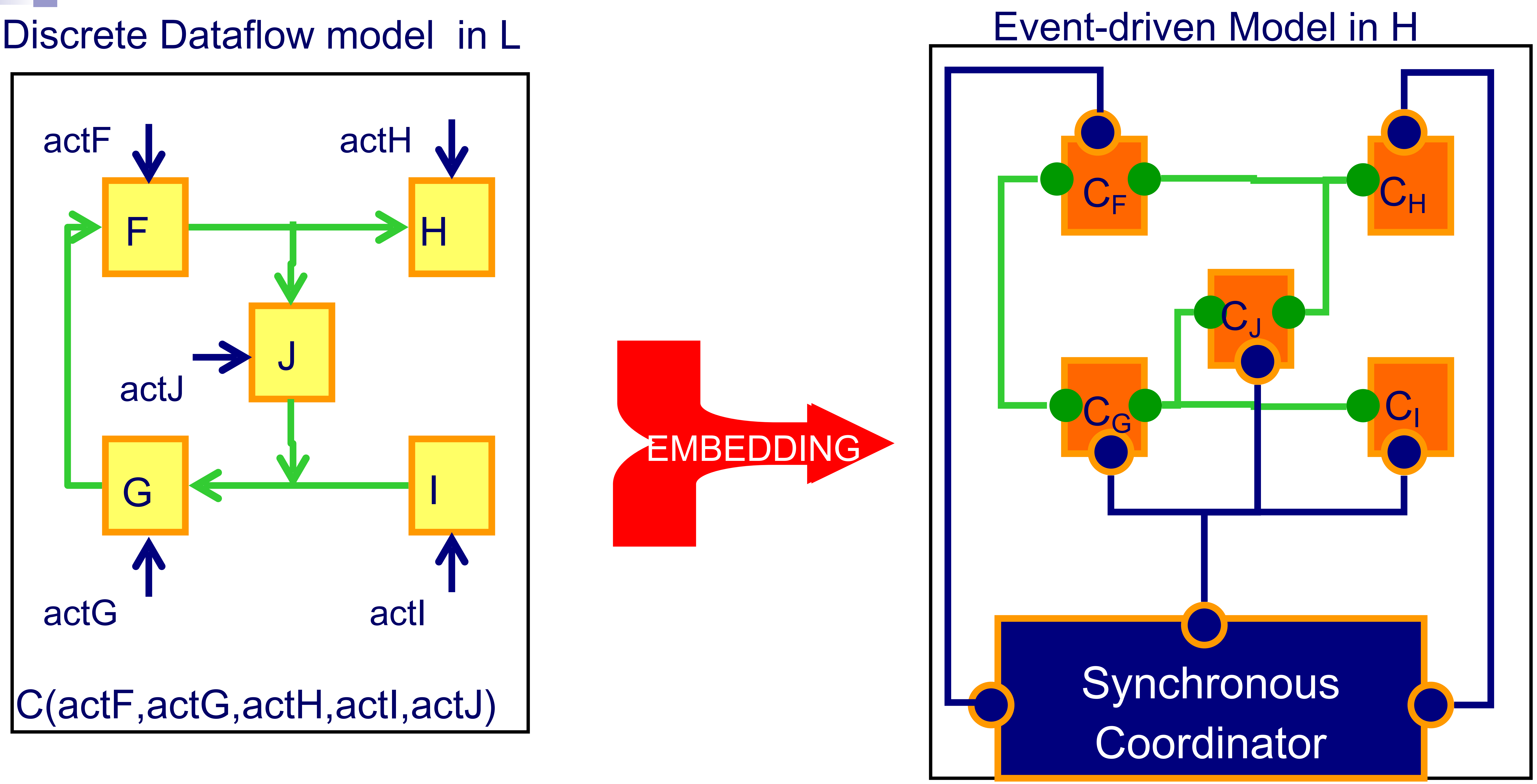


- Simulators with fixed integration step may either overshoot the limit or not reach convergence
- Proving nonZenoness is undecidable requires discovery and application of an induction hypothesis
- We need practicable theory to detect Zeno behavior

- ❑ Modeling CPS
- ❑ Discretization
- ❑ Execution and Implementation

OVERVIEW

Execution and Implementation



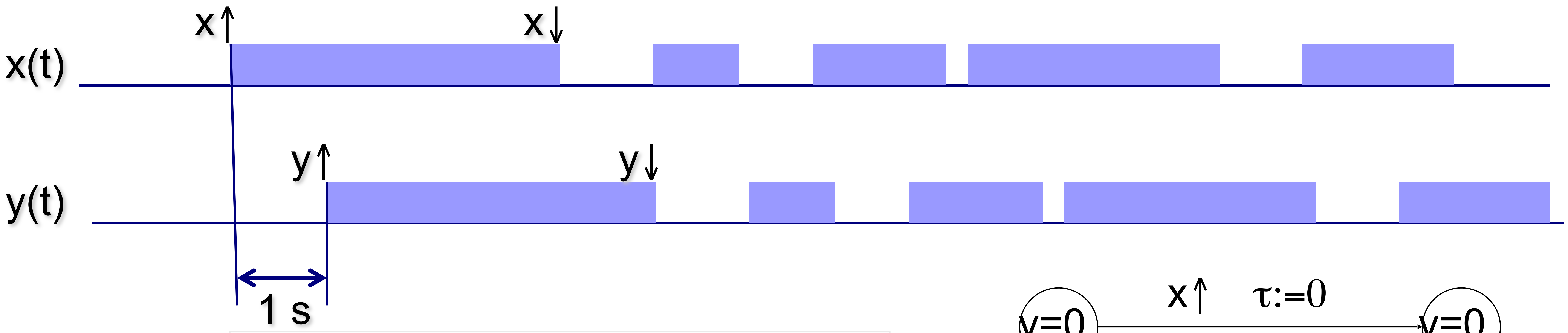
Apply embedding techniques to Discretized Hybrid Data Flow models (with start and stop events)

- modular code generation allowing separate compilation of components.

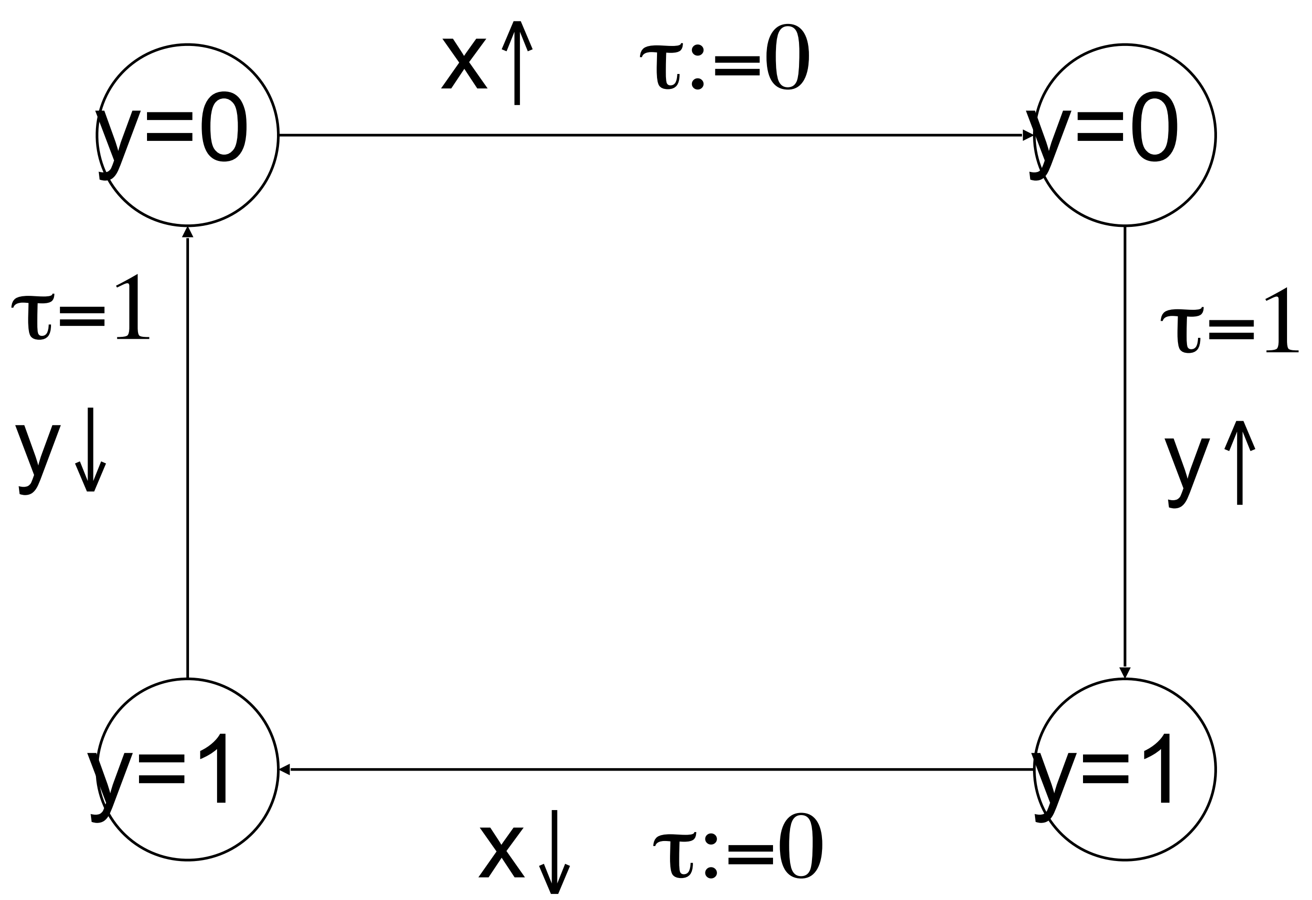
Execution and Implementation – Synchrony Assumption

Mathematically simple does not mean computationally simple!

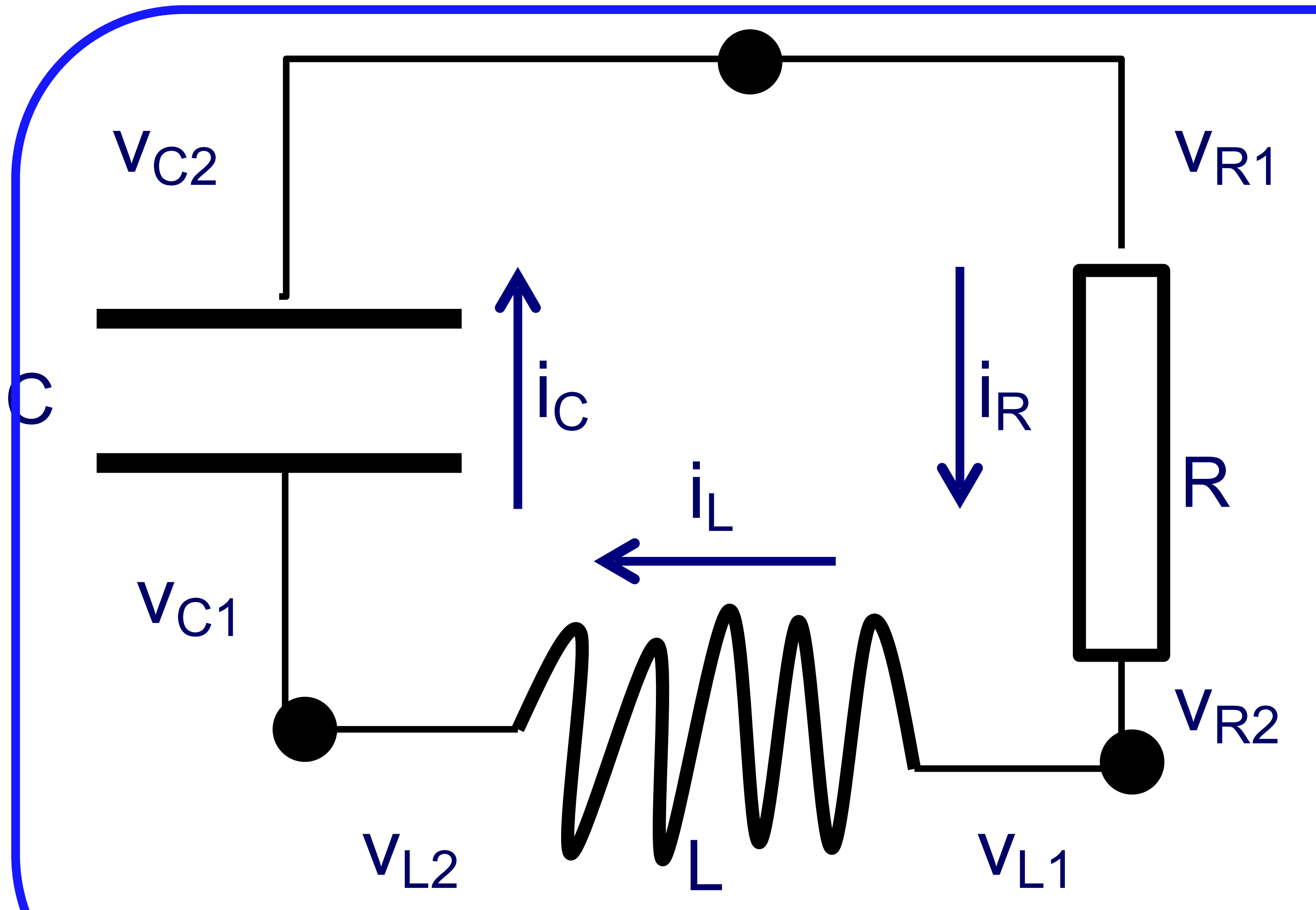
- The continuous data flow model may be sensitive to input changes even if they occur at very close times
- There can be an infinity of changes in a finite interval and this cannot be modeled computationally. There is no finite state computational model equivalent to a unit delay!



Equivalent timed automaton, provided that the distance between two consecutive input changes is more than 1sec – (synchrony assumption)



Execution and Implementation – Modular Simulation



Dynamics of components

$C(v_{C1}' - v_{C2}') = i_C$

$v_{R1} - v_{R2} = R\, i_R$

$L\, i_L' = v_{L1} - v_{L2}$

Constraints induced by connectors

$i_C = i_R = i_L$

$v_{C1} = v_{L2}$

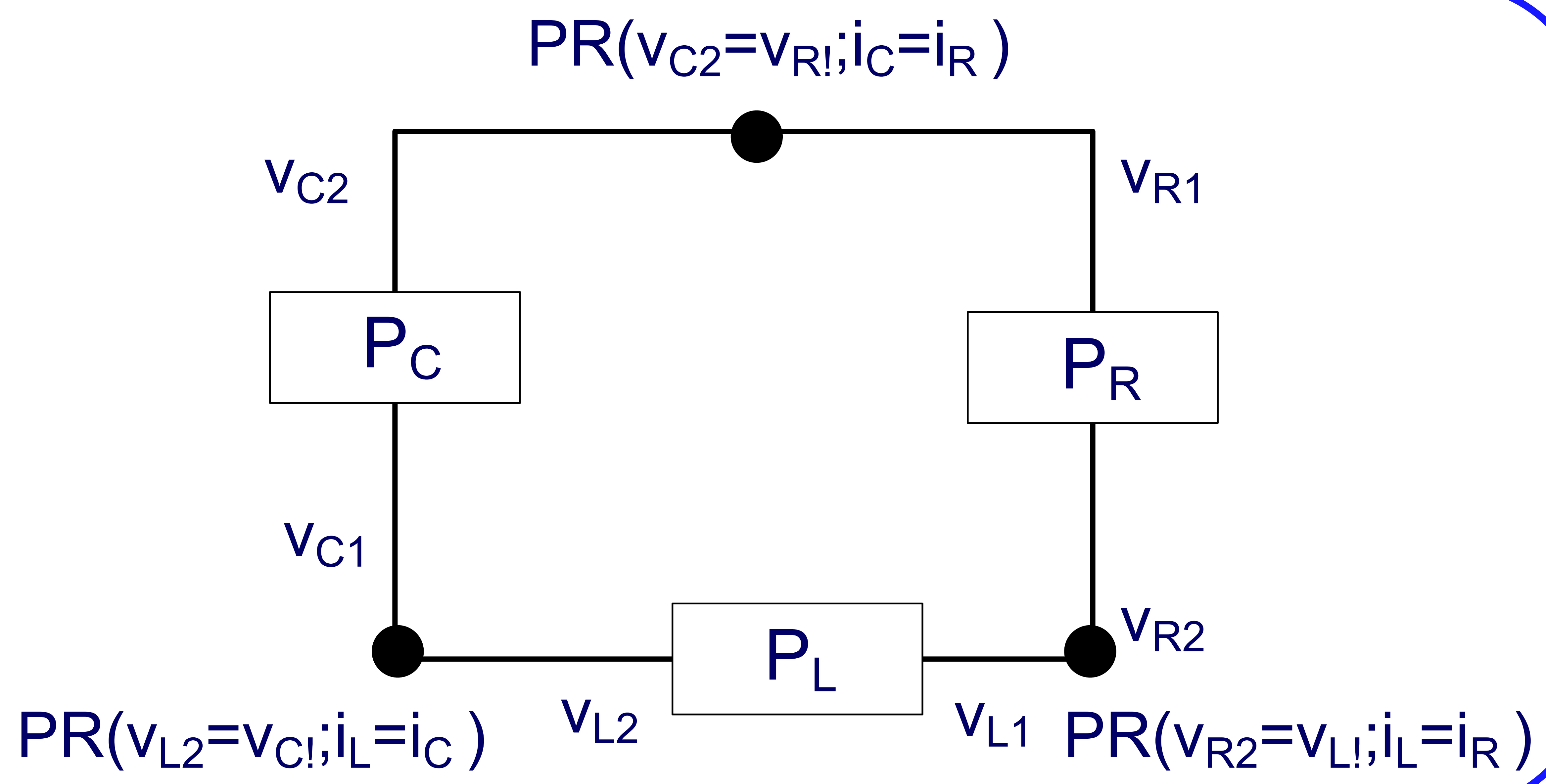
$v_{C2} = v_{R1}$

$v_{R2} = v_{L1}$

PHYSICAL MODEL

Modular Simulation

EXECUTABLE MODEL (CODE)



Thank you

Introduction
3 Research Examples
CPS and IoT
Linking Physicality and Computation

Practical Example: an Infusion Pump

Discussing Challenges and Opportunities of CPS
Wrap-Up

Screen shot from: “Alex Gibney, Zero Days (2016)”

Modeling Cyber Physical Systems — A practical example

An Infusion Pump Example

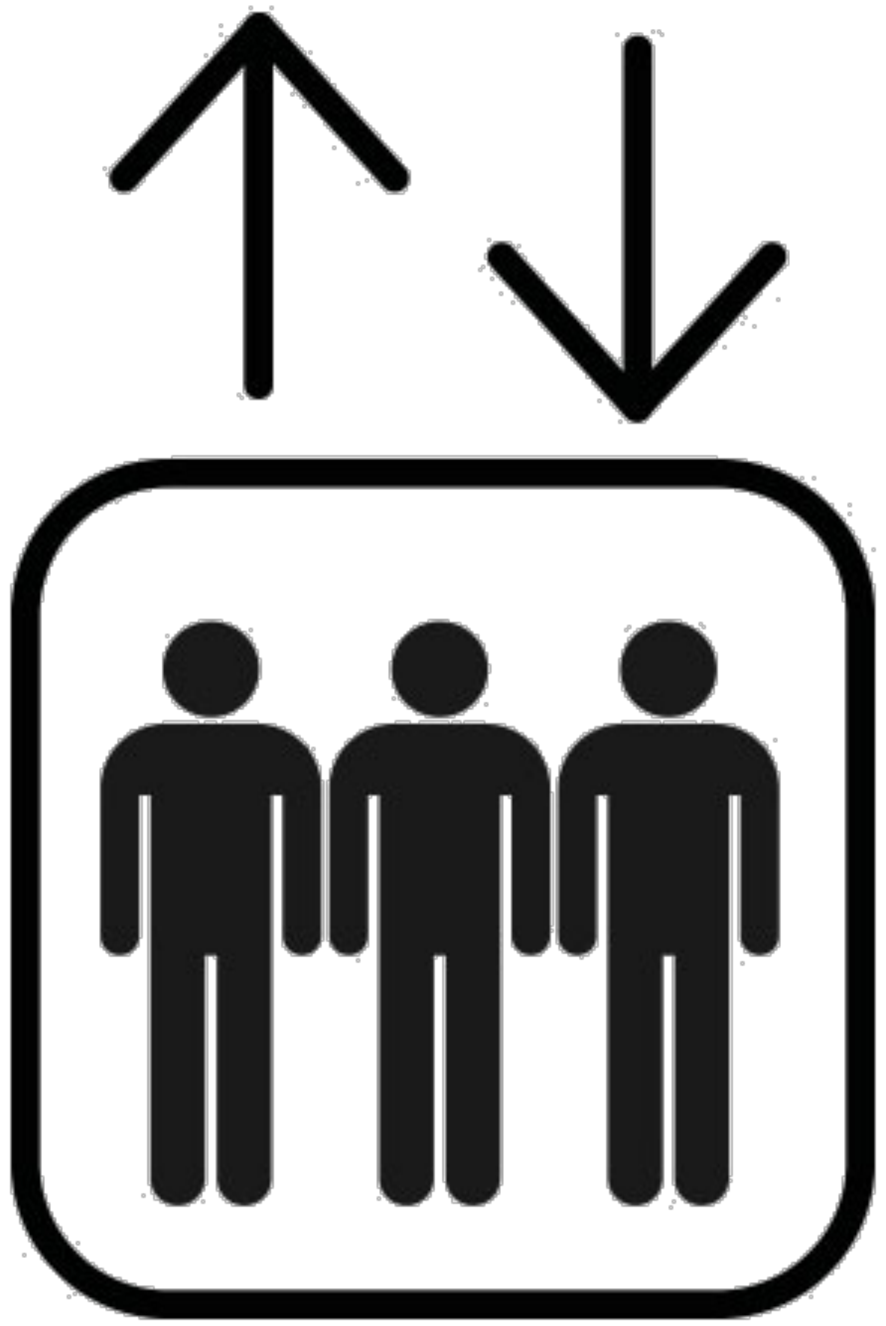
(15 minutes)

Overview

- Types of CPS
- Modelling of an Infusion Pump
- Challenges and open problems for CPS

Different Types of CPS

<u>Lift System</u>



Environment modelled using **discrete** events

Common Properties:

- Reactive Computation
- Concurrency
- Feedback Control of the Physical World
- Real-Time Computation
- Safety Critical

<u>Infusion pump</u>

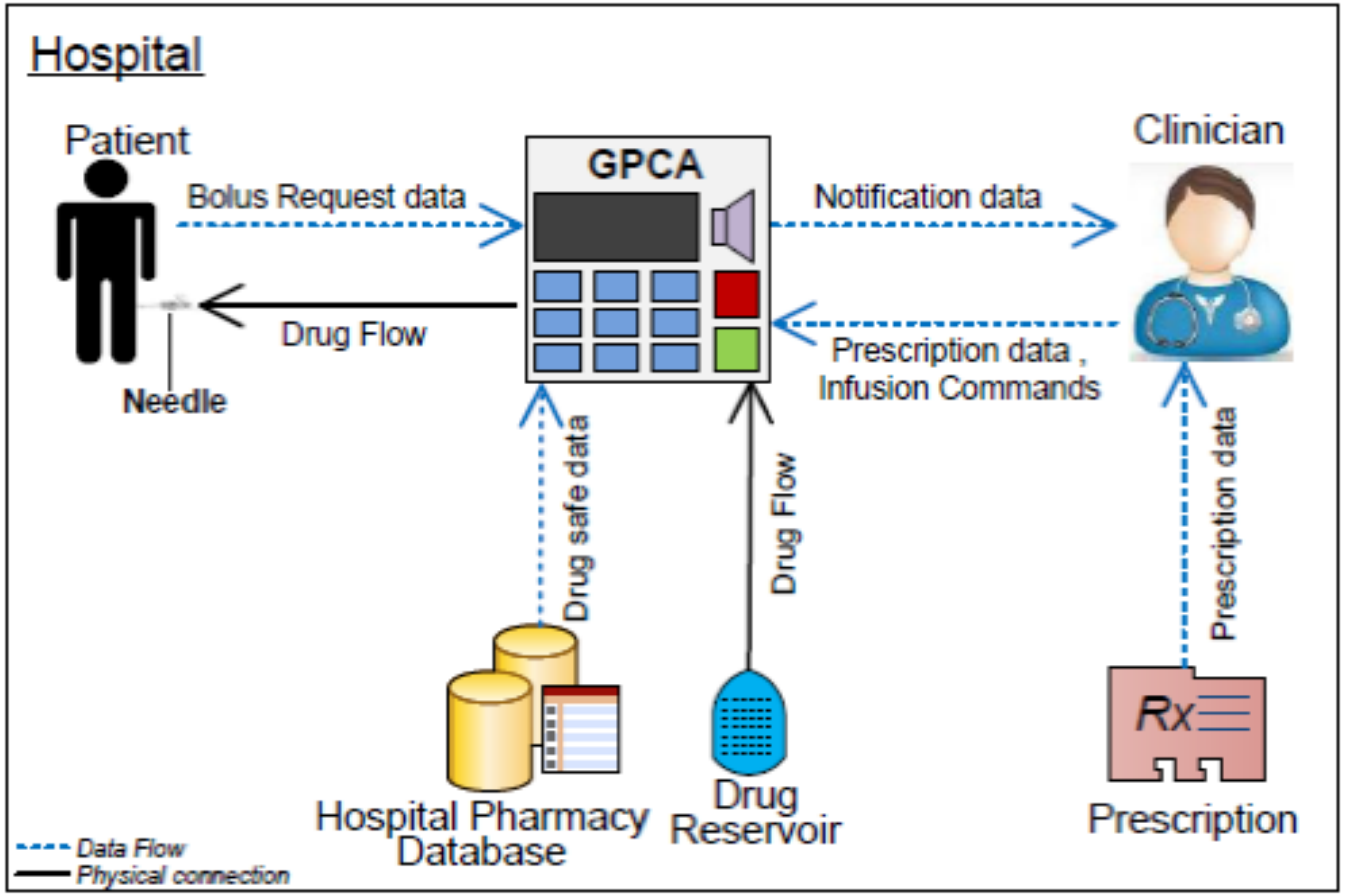
Environment represented using a **dynamic** model

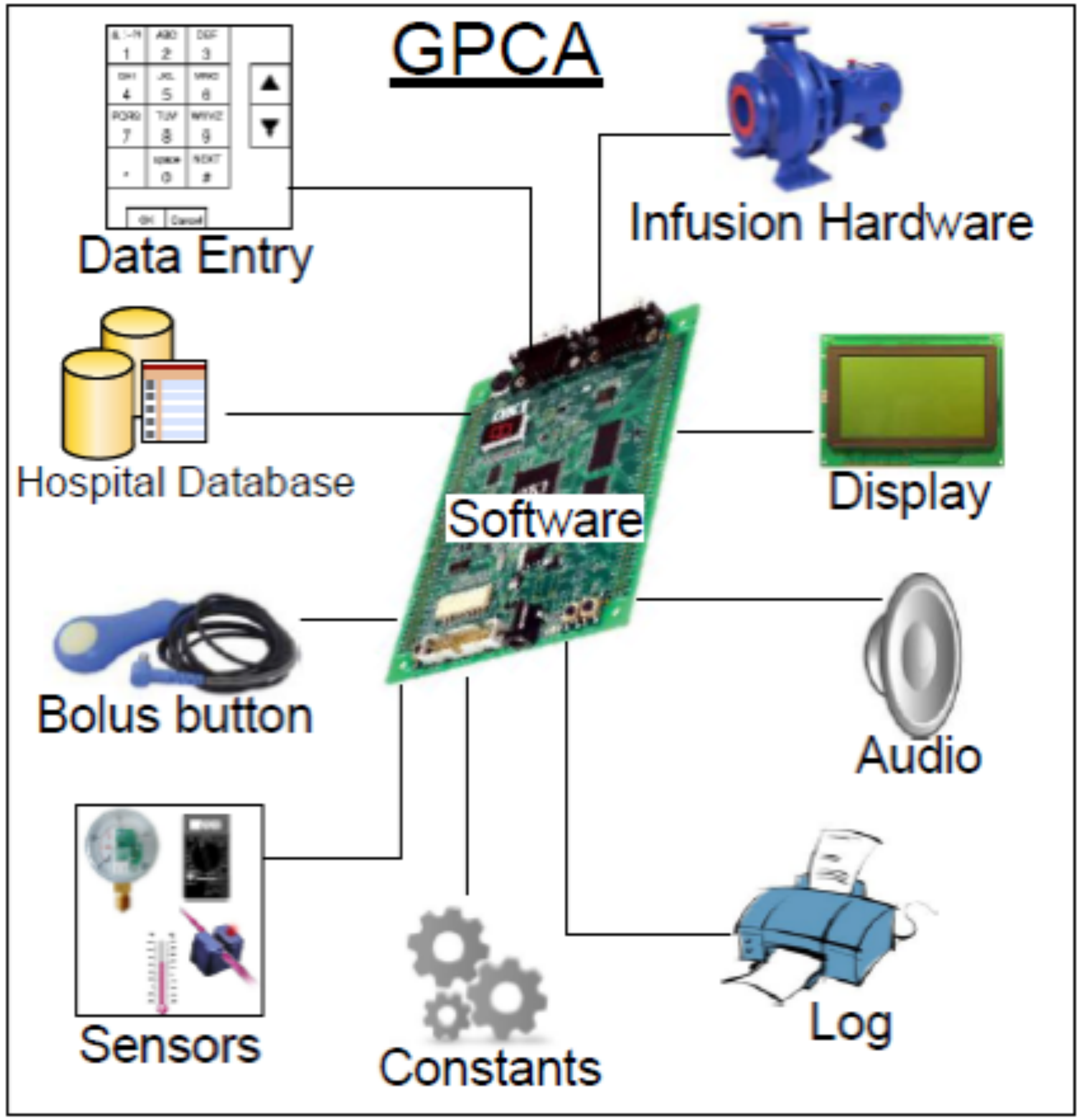
Tools for Modeling CPS

- System Model
  - Gives an overview of a system and its context.
- Architectural models
  - Visualize and understand the system boundaries
- Control models
  - Modeled in continuous domain
  - Helps to identify the missing requirements on the physical side of cyber-physical system
- Discrete Event models
  - understand detailed software requirements and perform rigorous validation

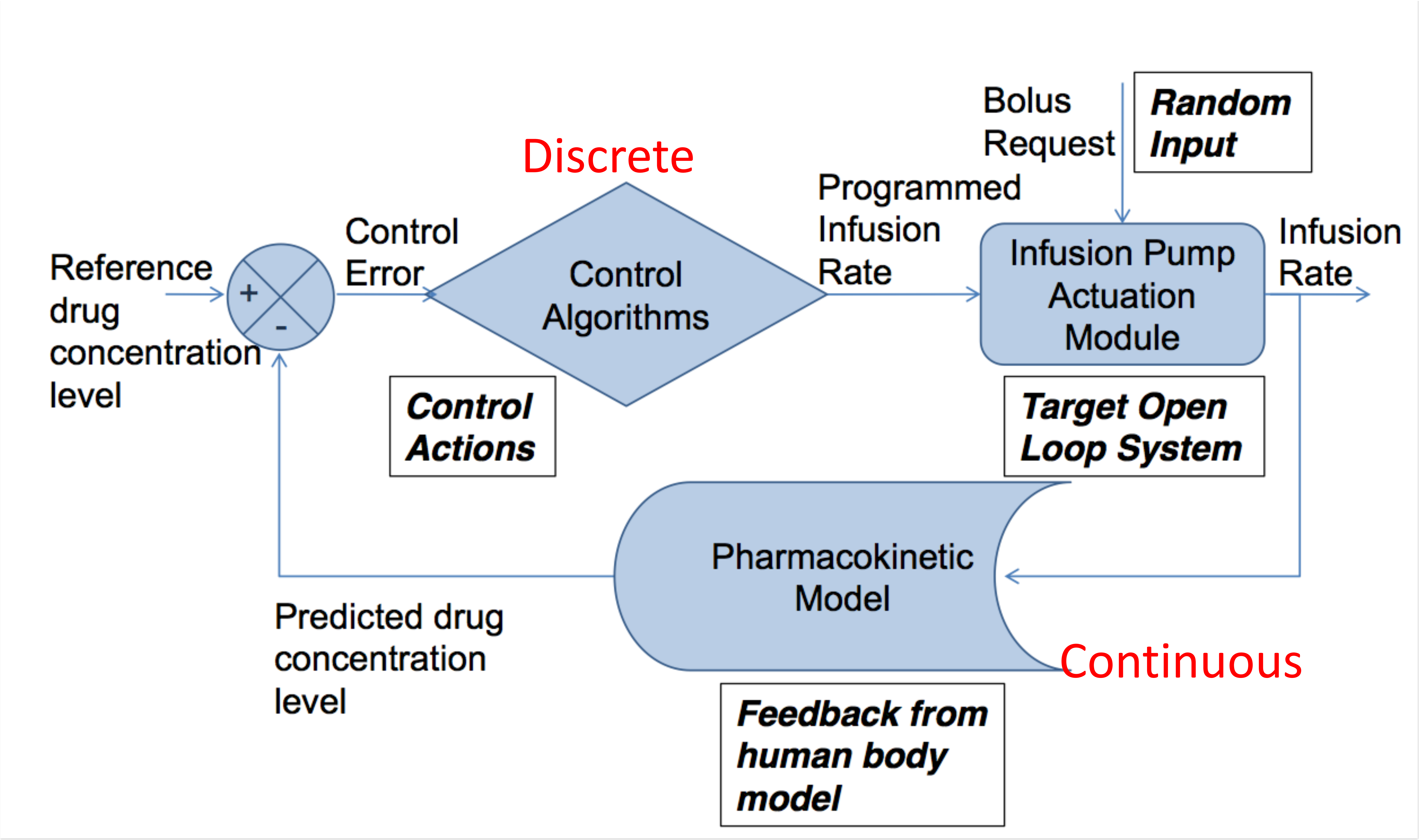
– …

Infusion Pump: System Model





Control Model



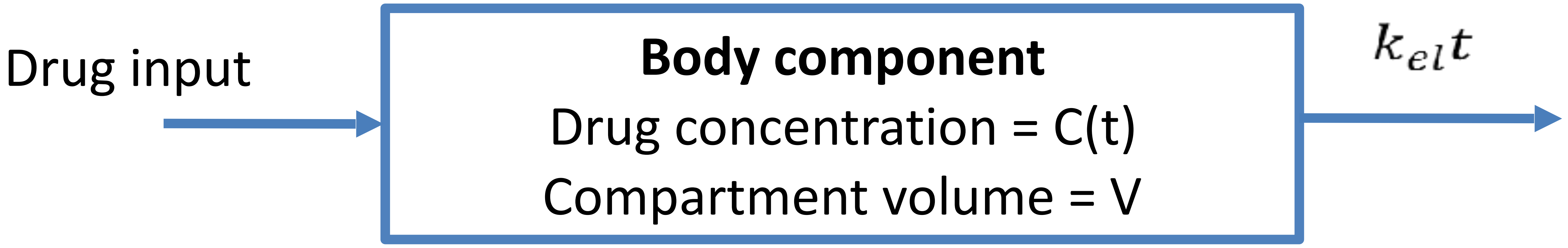
Pharmacokinetic Model: Drug Infusion

Infusion of the drug

- The diffusion process of the insulin/drug can be modeled using pharmacokinetic model in terms of infusion rate through a set of non-linear differential equations.
- A simple model (assuming body as one entity) could be as follows:

$$\frac{dC}{dt} = -k_{el}C(t)\ at\ t = 0, \qquad C_0 = \frac{Dose}{V}$$

$$C = C_0 e^{-k_{el}t} = \frac{D}{V_D}$$



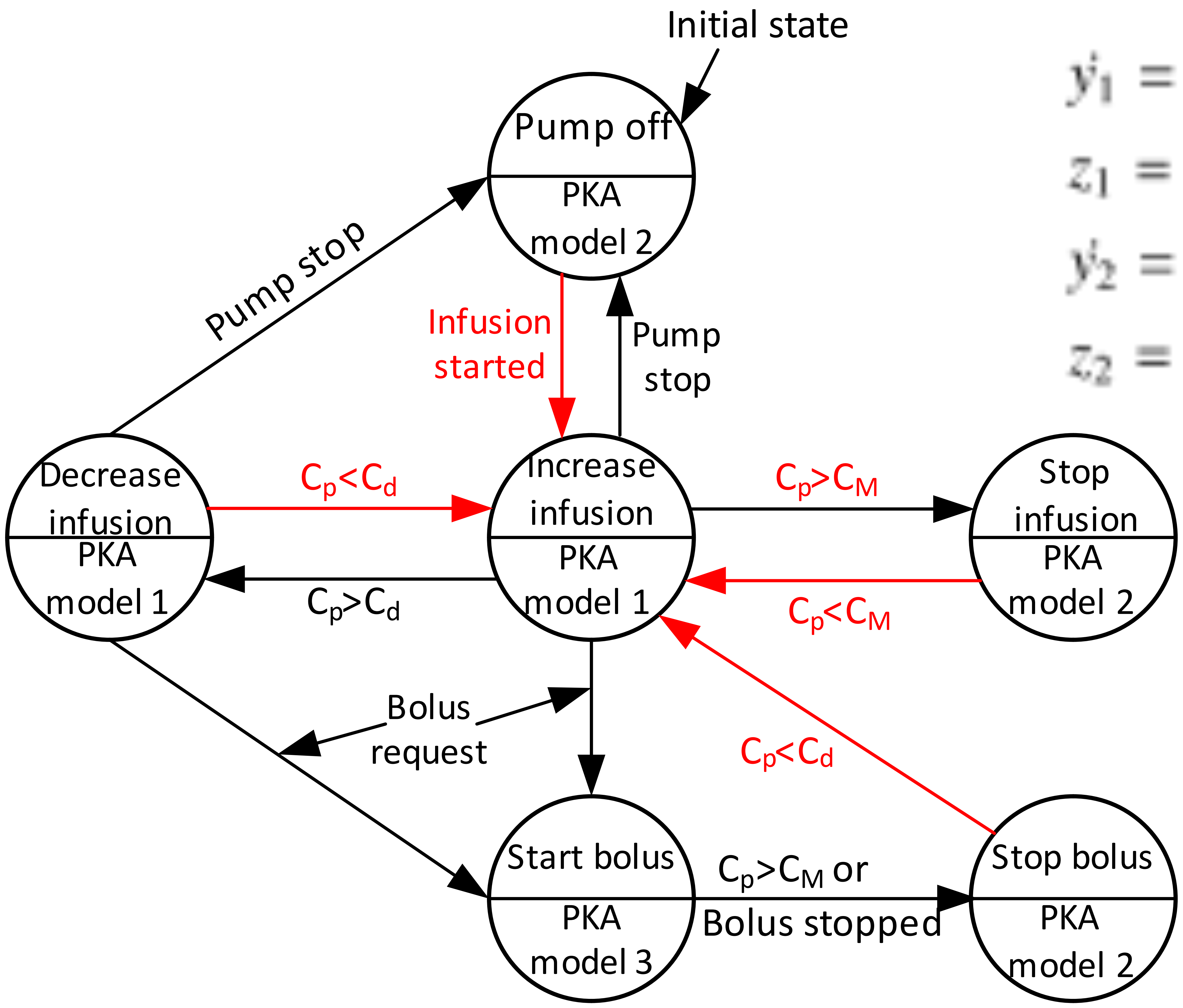
Infusion Pump Actuation Module

Control system of Infusion pump

- The control of the infusion pump can be modeled based on the insulin/drug concentration, feedback received and external input (by the user/patient)
- It can be modeled as spatio-temporal variations
- Drug diffusion model based on the infusion rate as input and the drug concentration in the blood as

$$\partial d/\partial t + \nabla(ud) = \nabla(D(r)\nabla d) + \Gamma(r)(d{\downarrow}B\,(t) - d) - \lambda d$$

Spatio-temporal Formal Model of Infusion Pump Control System



$$\dot{y}_1 = A_1 y_1 + B_1 z_2 + E_1 u(t - T_i)$$

$$z_1 = C_1 y_1(t - T_p)$$

$$\dot{y}_2 = A_2 y_2 + B_2 z_1$$

$$z_2 = C_2 y_2(t - T_r)$$

PKA model

Modeling Automata with Uppaal

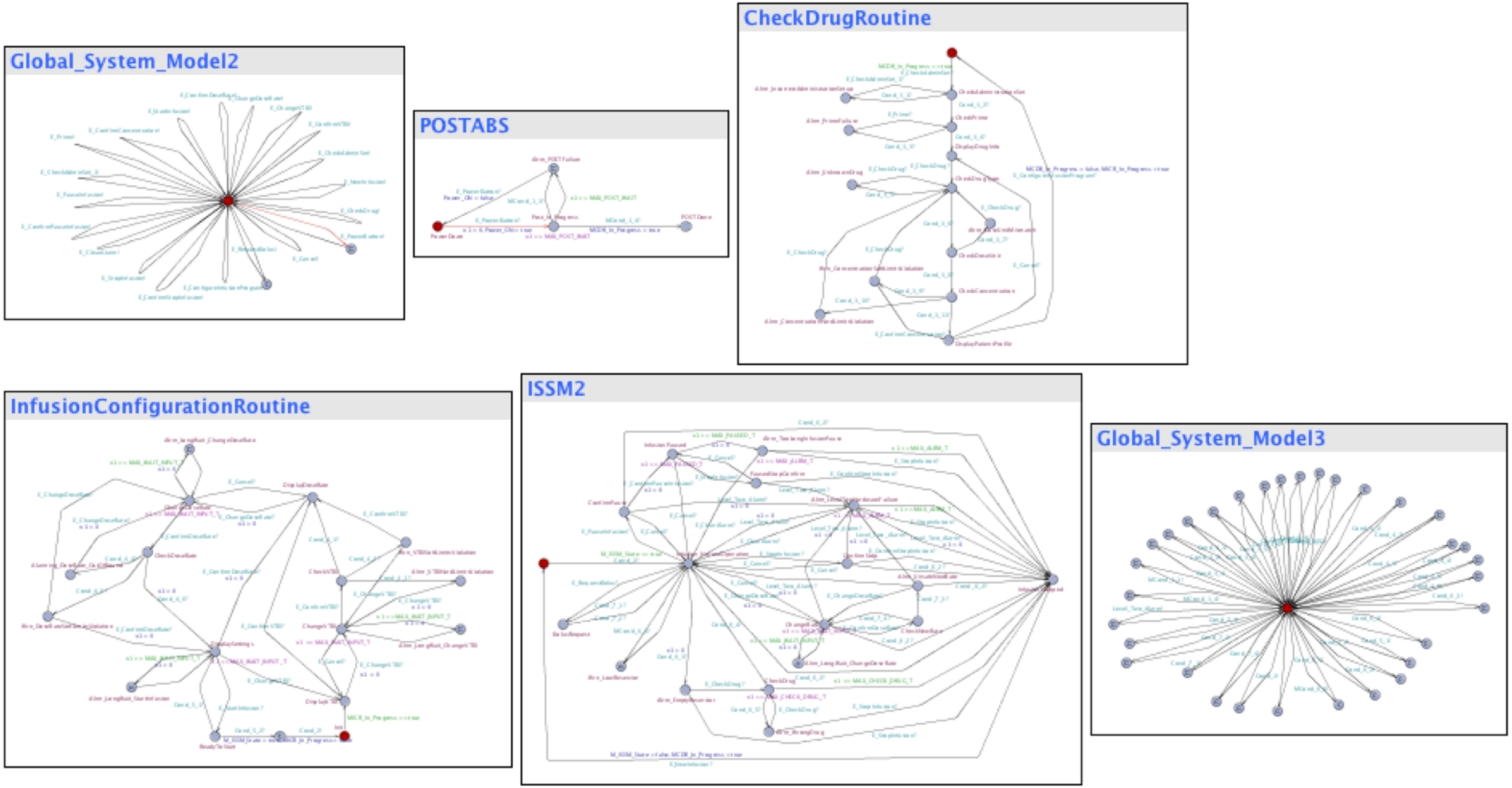
- Toolbox for **verification of real-time systems**

- Verify systems that can be modelled as **networks of timed automata** extended with integer variables, structured data types, user defined functions, and channel synchronisation.

Definition 1 (Timed Automaton (TA)). *A timed automaton is a tuple* (L, l_0, C, A, E, I)*, where* L *is a set of locations,* $l_0 \in L$ *is the initial location,* C *is the set of clocks,* A *is a set of actions, co-actions and the internal* τ*-action,* $E \subseteq L \times A \times B(C) \times 2^C \times L$ *is a set of edges between locations with an action, a guard and a set of clocks to be reset, and* $I : L \rightarrow B(C)$ *assigns invariants to locations.* □

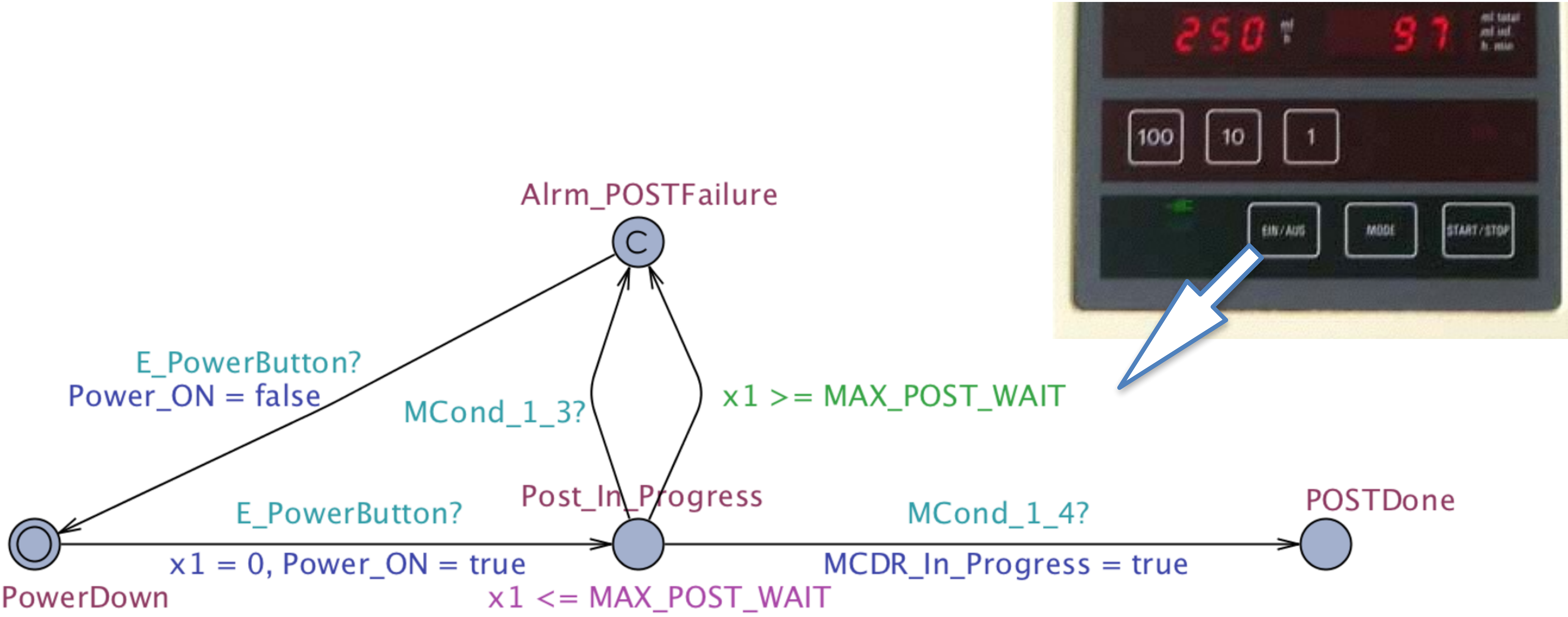
Behrmann, Gerd, Alexandre David, and Kim G. Larsen. "A tutorial on UPPAAL 4.0 (2006)."

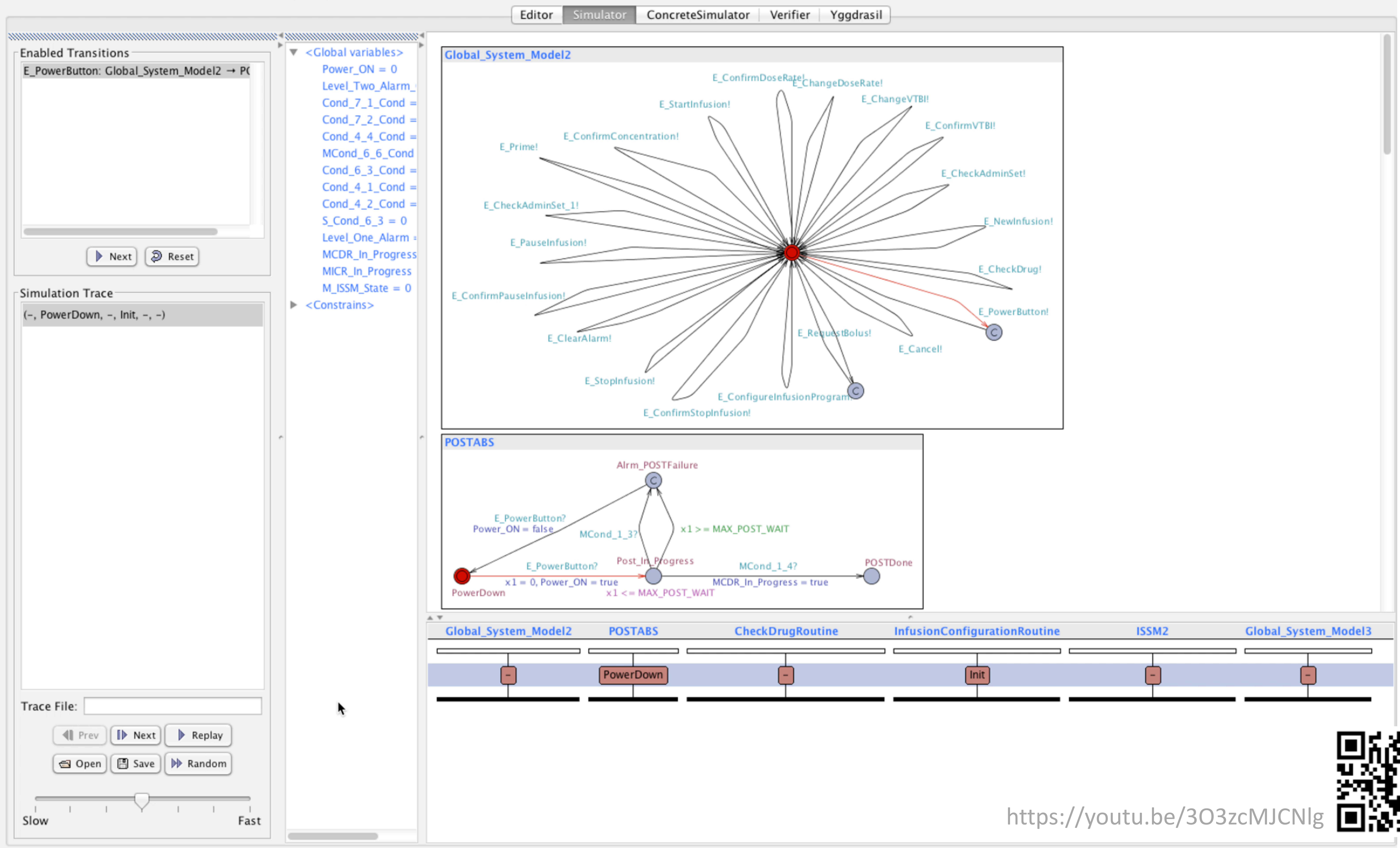
Overall Model



https://rtg.cis.upenn.edu/gip/#MBE

POST Finite State Automaton





https://youtu.be/3O3zcMJCNlg

Safety and Security of the Infusion Pump

- Dynamics of the physical models
  - Settling time is required for the drug concentration in blood to reach the reference level set by the operator.
  - A large settling time may lead to delay in the therapeutic effect of the drug.
  - Hysteresis relation with Zeno behaviour
- Assurance and uncertainty
  - Which parameters to consider when building the model – inherent incompleteness of physical models
  - How to react to unforseen risks
  - When to stop the analysis when building a safety case

Additional Challenges

- Concretisation
  - Simulation and analysis are performed at high level of abstraction but needs to be elaborated to actual implementations of the system
  - What kind of techniques can assist developers in specifying CPS?
- Runtime adaptation
  - Introspect and reason about changes
  - Find optimal configuration according to available resources
  - What kind of techniques can be used to learn/automatically extract the model of CPS post deployment?
  - How to monitor and react to untrustworthy systems?
- And many others: maintainability, dependability, security, resilience, ...

CPS – Current State

Formal Methods

- Modelling and reasoning about hybrid systems
- Rigorous methods for designing CPS

Software Engineering

- Eliciting and analysing requirements for CPS
- CPS an Human behaviour
- Techniques for assisting in implementing CPS

Distributed Systems

- Middleware solutions and standards to facilitate interactions between hybrid components
- Interoperability with legacy systems

Physical Sciences

- Control Theory
- Electrical Engineering
- Mechanical Engineering
- …

References

- A. Banerjee, S. Gupta, G. Fainekos, and G. Varsamopoulos, "Towards modeling and analysis of cyber-physical medical systems," Int. Symp. on Applied Sciences in Biomedical and Communication Technologies, 2011
- D. Arney, R. Jetley, P. Jones, I. Lee and O. Sokolsky, "Formal Methods Based Development of a PCA Infusion Pump Reference Model: Generic Infusion Pump (GIP) Project," Joint Workshop on High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability, 2007
- http://www.gatewaycoalition.org/files/hidden/deliv/ch3/3_5f.htm
- „Generic PCA Infusion Pump Reference Implementation," University of Pennsylvania, https://rtg.cis.upenn.edu/medical/gpca/gpca.html
- Behrmann, Gerd, Alexandre David, and Kim G. Larsen. "A tutorial on UPPAAL 4.0 (2006)."
- UPPAAL Team. "UPPAAL 4.0: Small tutorial, November 2009." URL http://www. it. uu. se/research/group/darts/uppaal/small_tutorial. pdf. Accessed: December 10 (2010).
- „Hinweise zur Arbeit mit UPPAAL,"http://st.inf.tu-dresden.de/files/teaching/ws10/SEW/Hinweise.pdf
- "Cyber-Physical Systems Challenges—A Needs Analysis for Collaborating Embedded Software Systems," in Software & Systems Modeling, Springer Berlin/Heidelberg, ISSN 1619-1366. vol. 15, nr. 1, pp. 5-16, 2016
- Uppaal model of the generic infusion pump from here: http://rtg.cis.upenn.edu/medical/gpca/model/ver0.86-release-v2.1.zip
- The Generic Infusion Pump (GIP), https://rtg.cis.upenn.edu/gip/#MBE

Challenges and Opportunities

Screen shot from: “Alex Gibney, Zero Days (2016)”

CPS Discussion

- Theory of Resilient CPS
  - Developing the foundation of **modelling**, **synthesis** and **development** of **resilient** cyber-physical systems
  - Unification of modelling frameworks encompassing **continuous and discrete systems modelling**
  - **Verification and simulation**: estimated quality, safety, and assurance
- Design and Engineering of **Resilient** CPS
  - Rigorous Design of CPS: **tools and execution platforms**
  - **Component-based engineering** for CPS (modularity, compositionality, composability)
  - Efficient **compilation of hybrid models** into executable code for implementation or simulation
  - Adaptive methods to cope with environment **uncertainty**
- Applications and Exemplars for CPS
  - **Killer apps for CPS**
  - **CPS and the Internet of Things (IoT)**
  - Metrics for **evaluating and comparing approaches for CPS**

Introduction

3 Research Examples

CPS and IoT

Linking Physicality and Computation

A Practical CPS Example: an Infusion Pump

Discussing Challenges and Opportunities of CPS

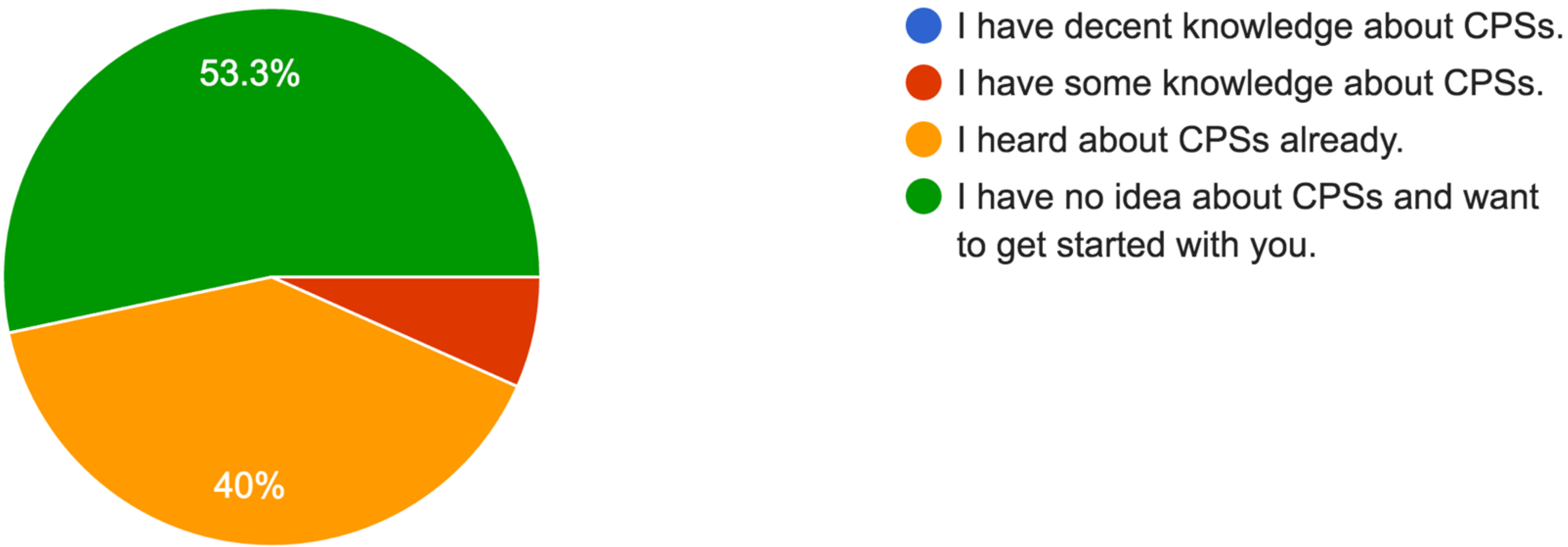
Wrap-Up

Screen shot from: “Alex Gibney, Zero Days (2016)”

Agenda

- Introduction to the Workshop (5 min; Marc-Oliver Pahl)
- 3 Research Examples and their Links to CPS? (15 min; Sai Manoj, Amel Bennaceur, Marc-Oliver Pahl)
- **CPS and IoT** (15 min; **Vint Cerf**)
- **Linking Physicality and Computation** (15 minutes; **Joseph Sifakis**)
- A Practical CPS Example: an Infusion Pump (15 min; Amel, Sai, Marc-Oliver)
- Discussing Challenges and Opportunities of CPS (20 min; all)
- Wrap-Up (5 min; Joseph Sifakis?, Amel, Sai, Marc-Oliver)

What is your level of expertise regarding Cyber Physical Systems? (15 responses)



- **https://youtu.be/2qaxJs8wYVw**

Cut material from the “Alex Gibney, Zero Days (2016)” trailer


Thank you!
Kein signal
04 : 24
Sai Manoj
Joseph
Sifakis
Vint
Cerf