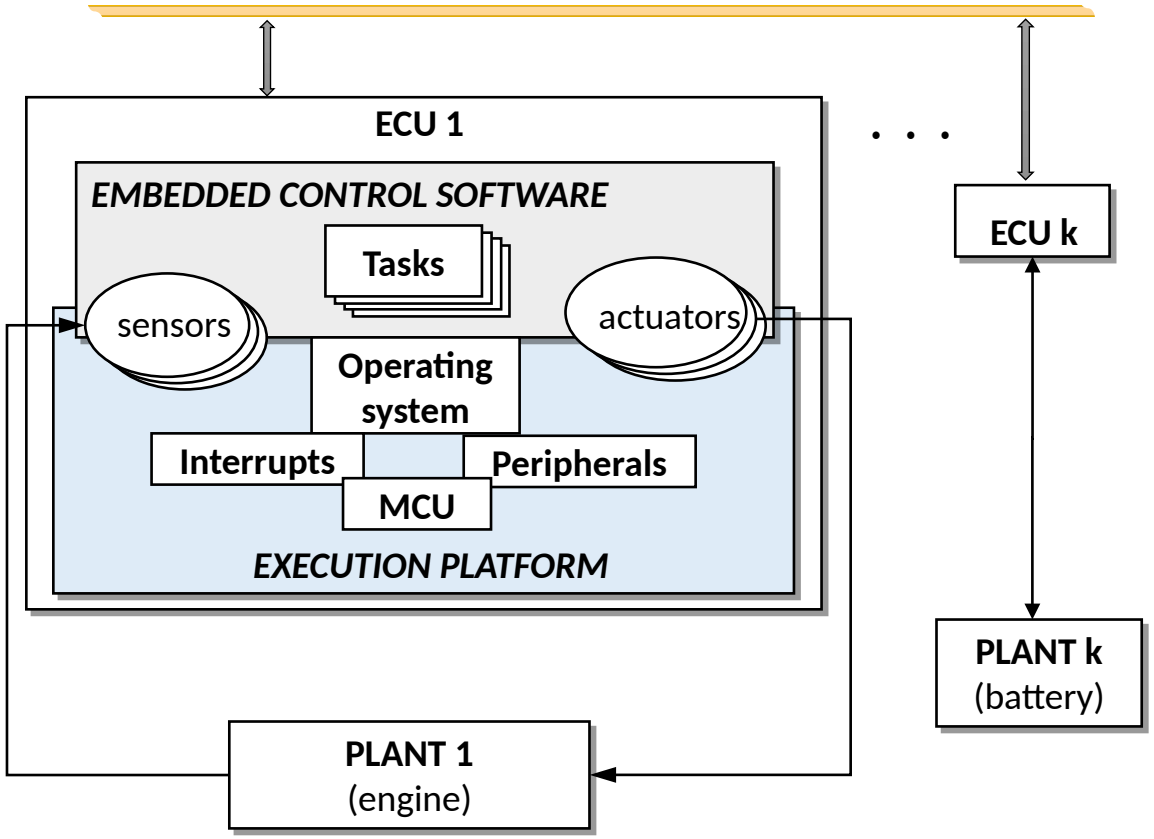
PARIS LODRON UNIVERSITÄT SALZBURG

Modeling and Simulation (M&S) in Embedded Software Development

Stefan Resmerita et al.

Embedded Systems



IEEE Spectrum

Embedded Software

PARIS LODRON UNIVERSITÄT SALZBURG

- Runs on execution platforms with limited computing resources
  - memory and clock speed
- Subject to **real-time requirements**
  - periodic execution, worst-case reaction time, end-to-end latency

- Legacy software: Hand-coded, performance-oriented design

- Modern software: Code-generated, **model-based design**

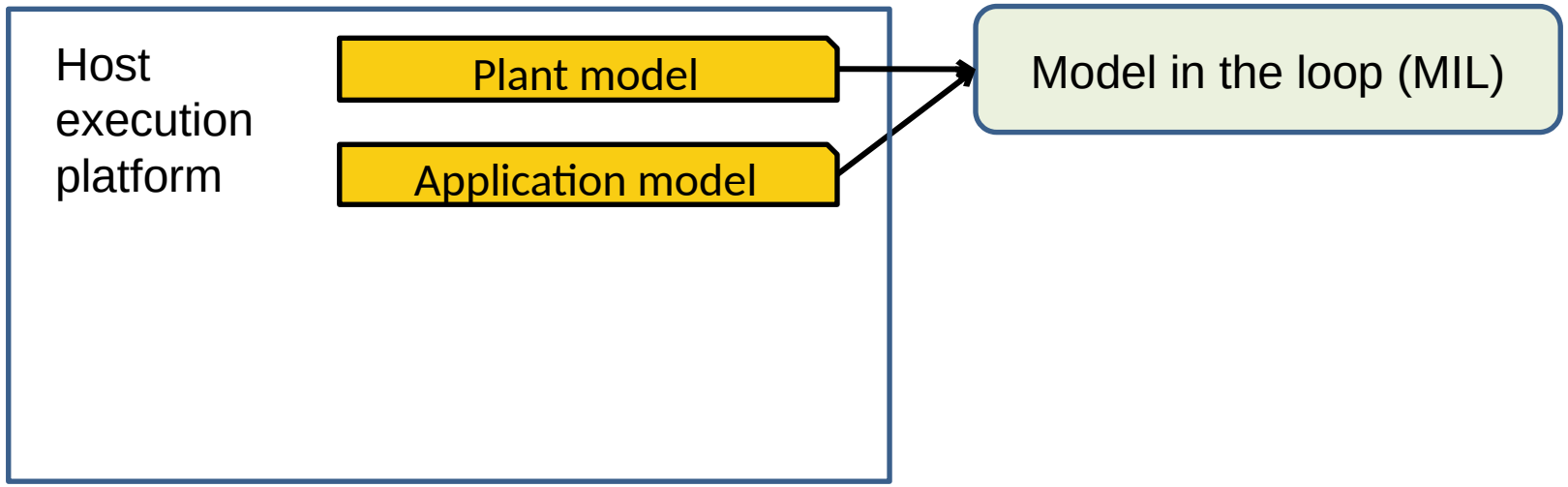
Embedded Software Industry (auto)

PARIS LODRON UNIVERSITÄT SALZBURG

- Functions developed by different teams/companies and delivered as binaries + documentation
- Put together into one system (ECU) by the ECU supplier
- Properties are difficult to verify
- Changes are difficult to manage
- **Validation is done mainly by simulation**

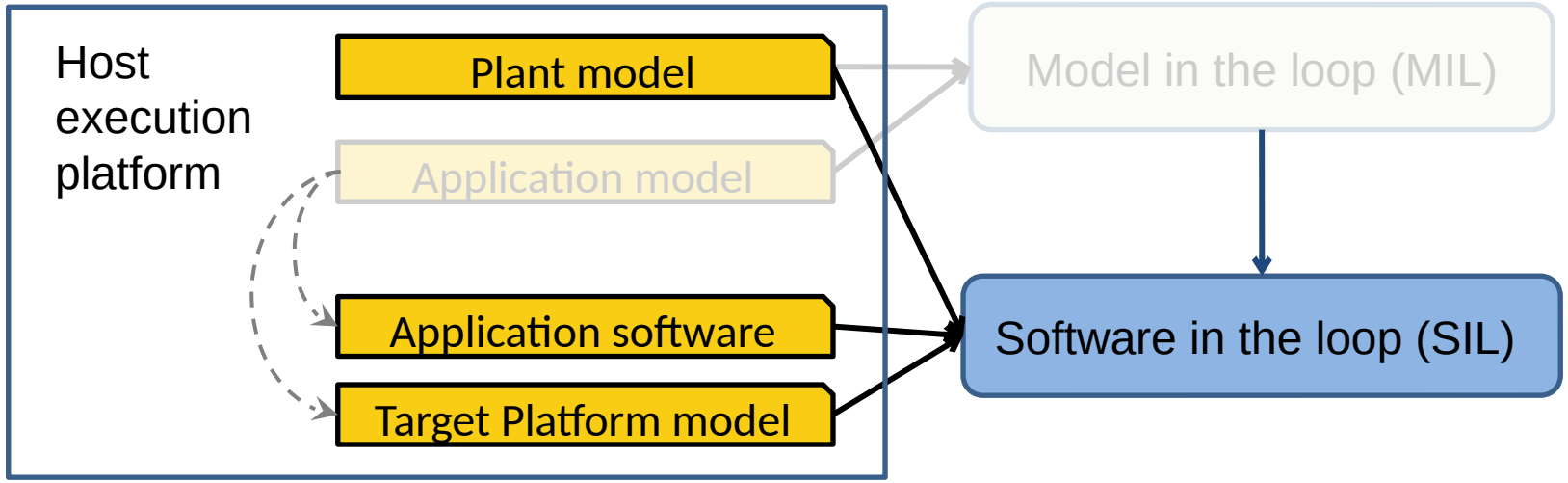
Simulation in Model-based Design

PARIS LODRON UNIVERSITÄT SALZBURG



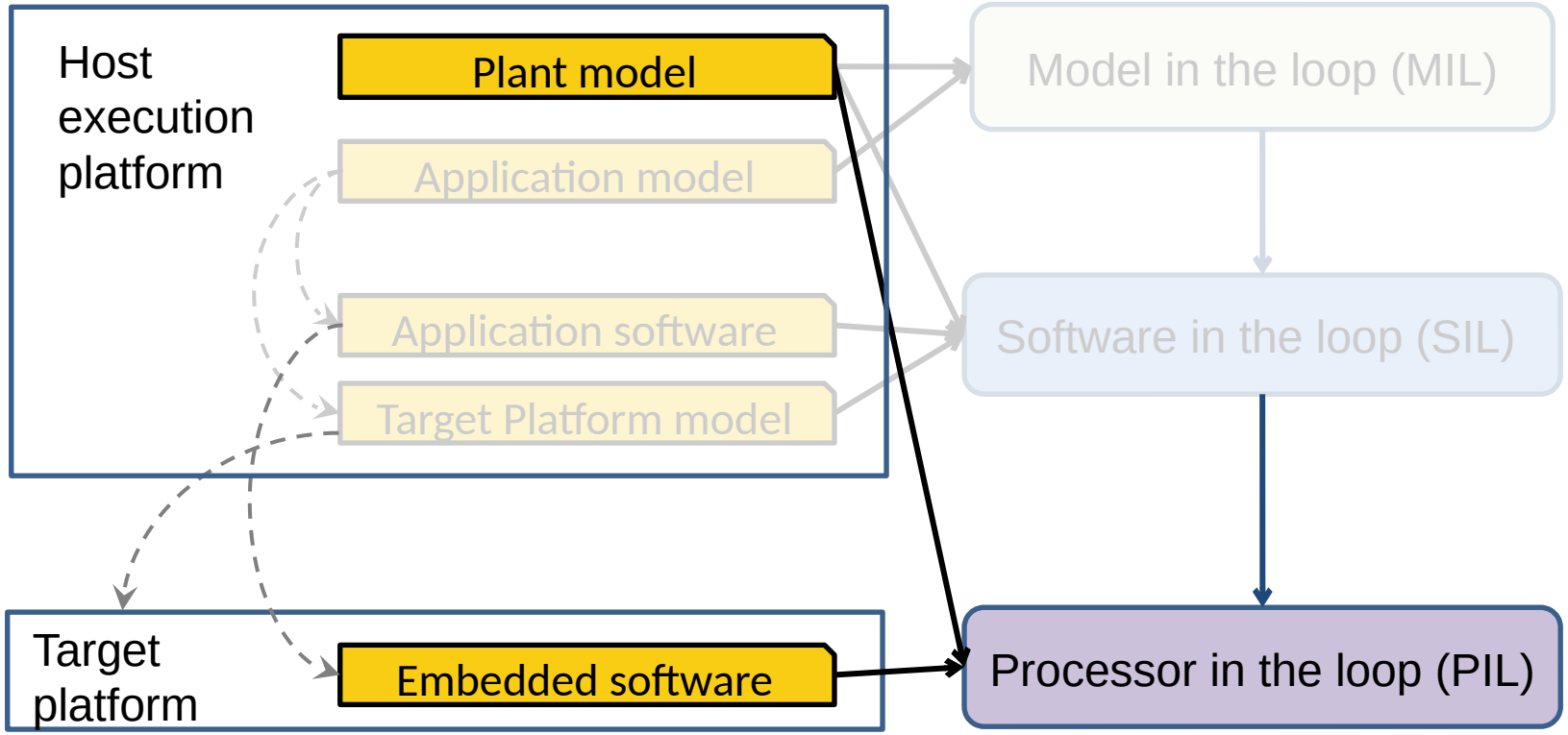
Simulation in Model-based Design

PARIS LODRON UNIVERSITÄT SALZBURG



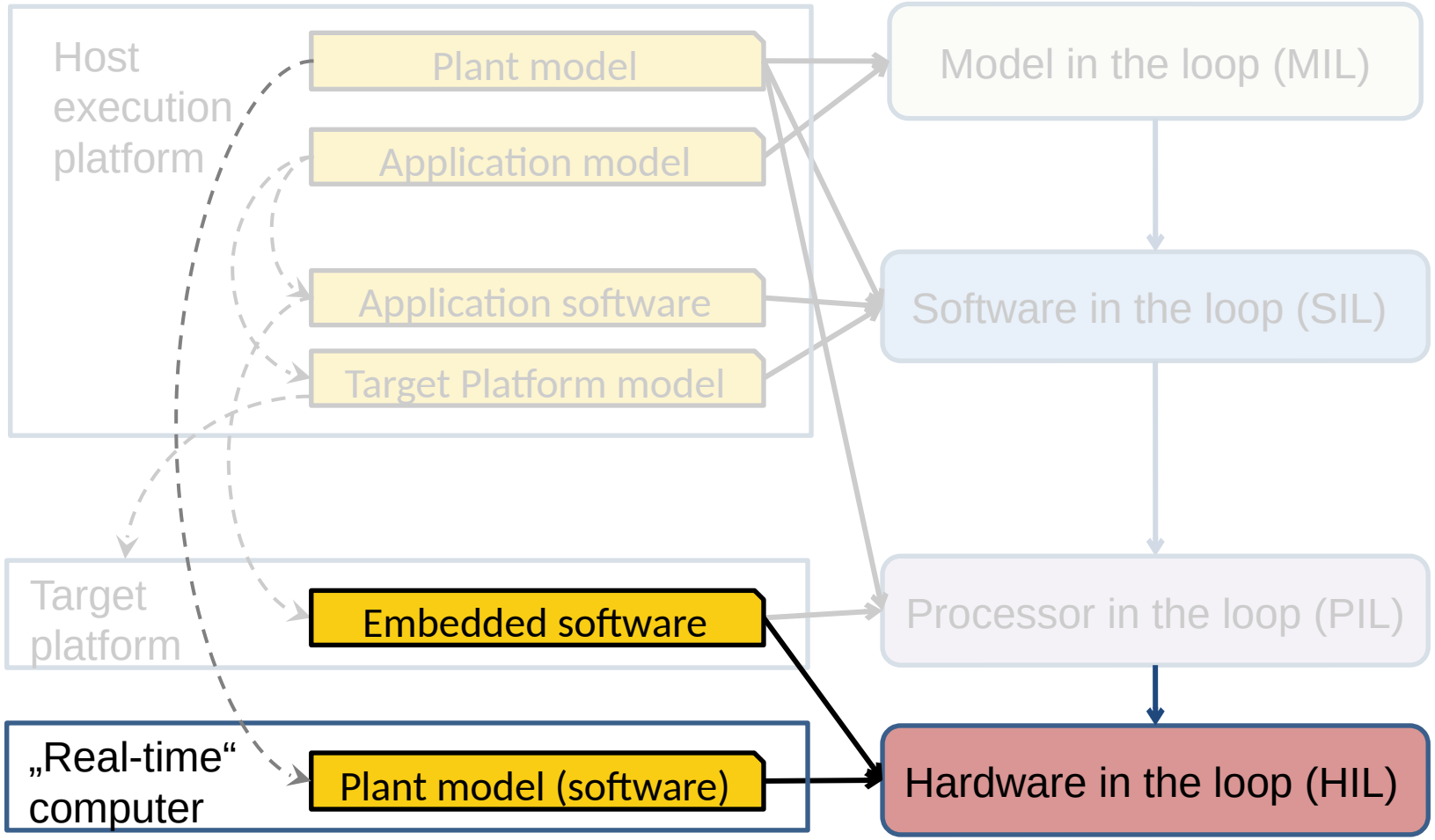
Simulation in Model-based Design

PARIS LODRON UNIVERSITÄT SALZBURG

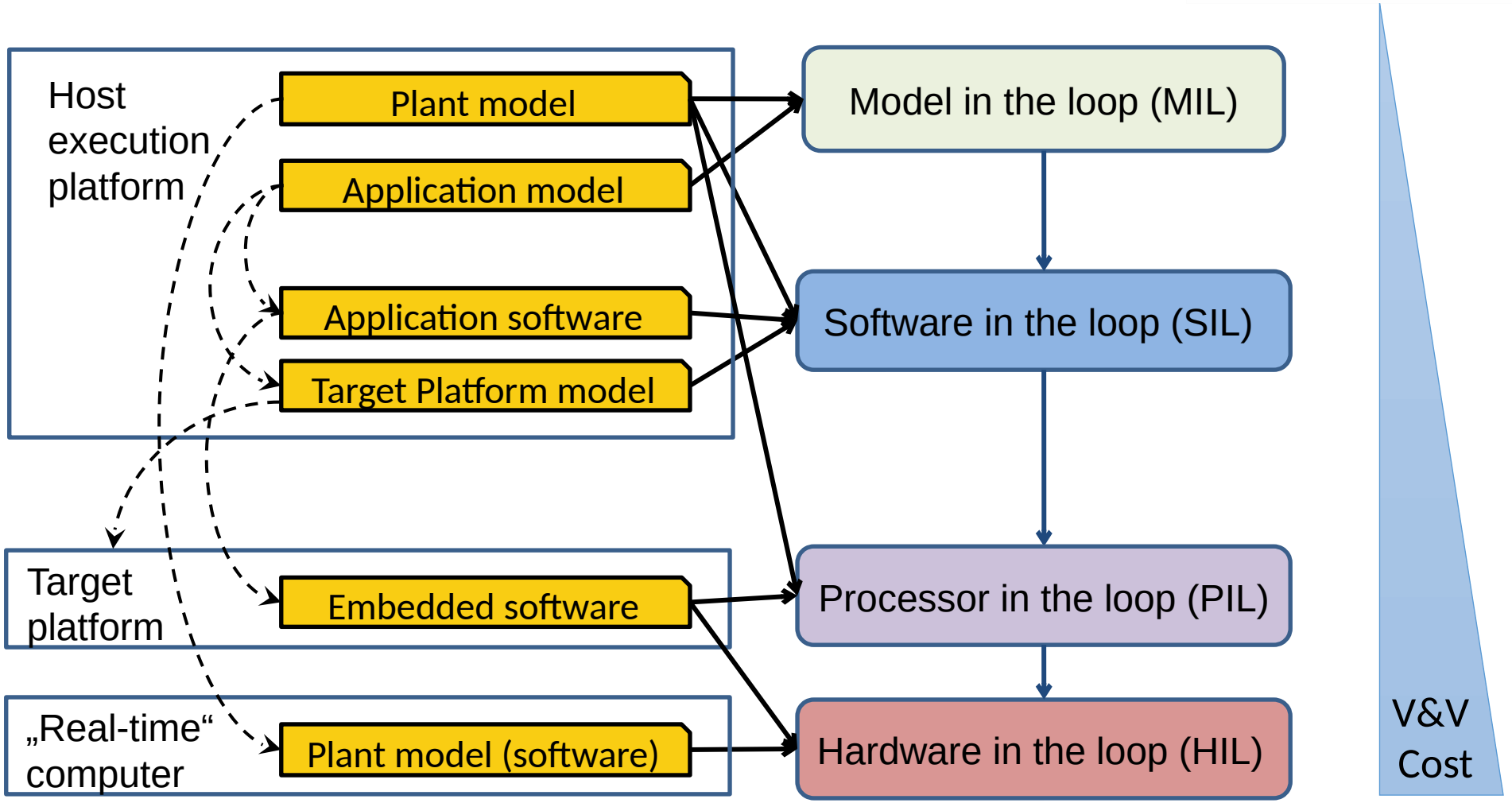


Simulation in Model-based Design

PARIS LODRON UNIVERSITÄT SALZBURG



Simulation in Model-based Design



Validation

PARIS LODRON UNIVERSITÄT SALZBURG

- Functional properties
  - Individual software component (unit testing)
  - Ensemble (integration) of software components
  - Deployment
- Non-Functional properties
  - Time
  - Energy consumption
  - Implementation costs

Validation

PARIS LODRON UNIVERSITÄT SALZBURG

- Functional properties
  - Individual software component (unit testing)
  - Ensemble (integration) of software components
  - Deployment
- Non-Functional properties
  - Time
  - Energy consumption
  - Implementation costs

- **Instrumental in industrial adoption of research results!**

PARIS LODRON UNIVERSITÄT SALZBURG

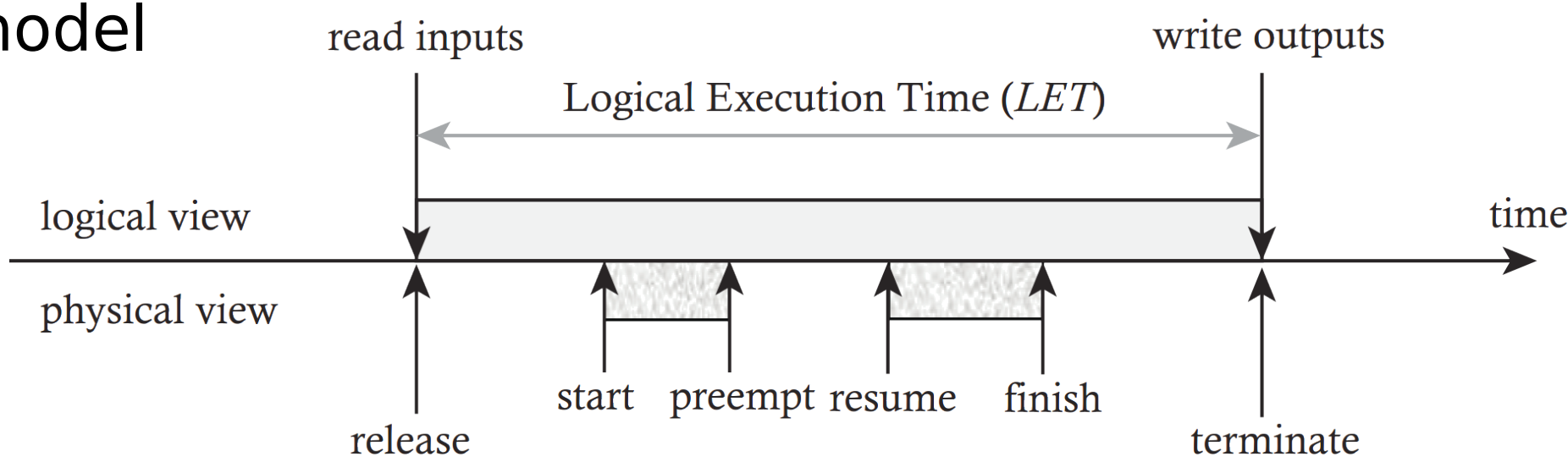
A M&S Success Story

Logical Execution Time: from concept to practice to production (2001-2021)

The Logical Execution Time (LET)

PARIS LODRON UNIVERSITÄT SALZBURG

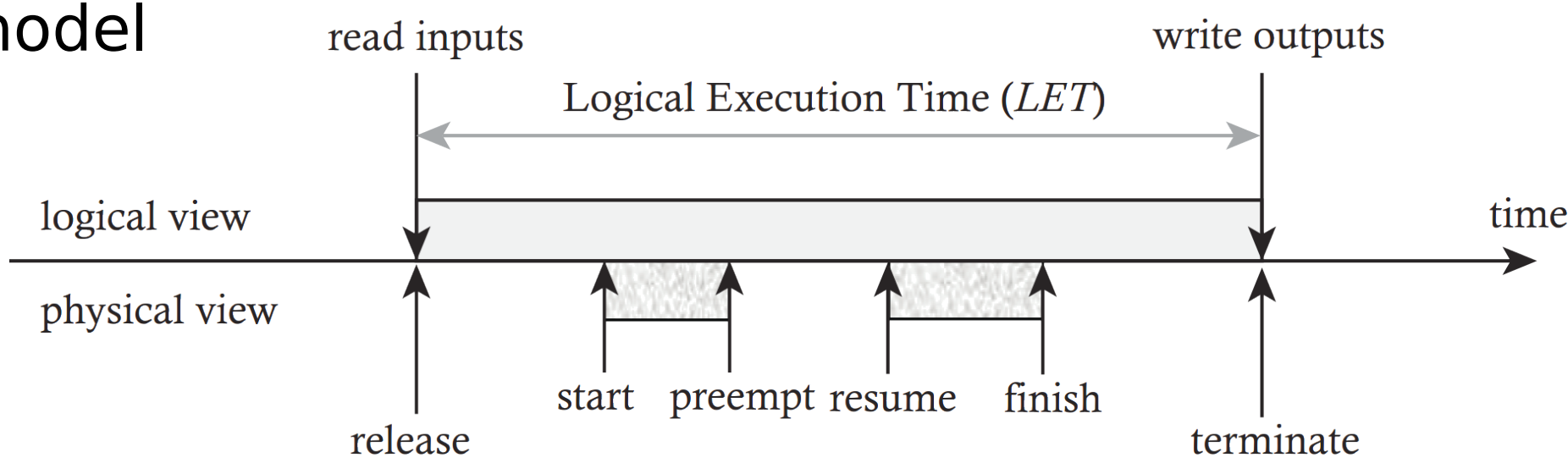
- Behavioral model



The Logical Execution Time (LET)

PARIS LODRON UNIVERSITÄT SALZBURG

- Behavioral model



- Programming model (Giotto, 2001)
  - Separation of concerns: timing from functionality from platform

LET Adoption in Industry?

PARIS LODRON UNIVERSITÄT SALZBURG

Benefits

- Determinism
- Robustness
- Portability
- Correct-by-construction design

Costs

- CPU usage
- Memory
- Increased end-to-end latency

A History of Execution Time

PARIS LODRON UNIVERSITÄT SALZBURG

- 2001: Giotto (Henzinger, Horrowitz, Kirsch)

A History of Execution Time

PARIS LODRON UNIVERSITÄT SALZBURG

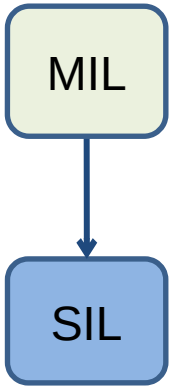
- 2001: Giotto (Henzinger, Horrowitz, Kirsch)
- 2004: TDL (Pree, Templ)
- 2008: *Simulation of LET Models in Simulink and Ptolemy* (Derler, Naderlinger, Pree, Resmerita, Templ)

MIL

A History of Execution Time

PARIS LODRON UNIVERSITÄT SALZBURG

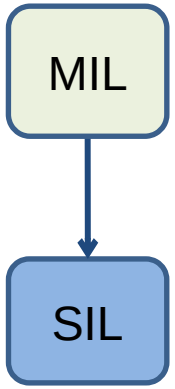
- 2001: Giotto (Henzinger, Horrowitz, Kirsch)
- 2004: TDL (Pree, Templ)
- 2008: *Simulation of LET Models in Simulink and Ptolemy* (Derler, Naderlinger, Pree, Resmerita, Templ)
- 2012: *The Validator Tool Suite* (Resmerita, Pree)



A History of Execution Time (contd.)

PARIS LODRON UNIVERSITÄT SALZBURG

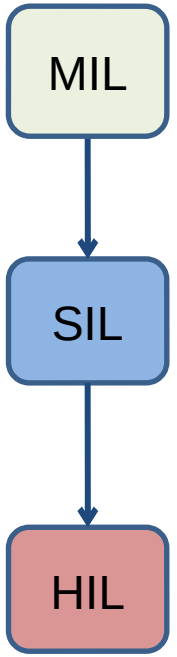
- 2001: Giotto (Henzinger, Horrowitz, Kirsch)
- 2004: TDL (Pree, Templ)
- 2008: *Simulation of LET Models in Simulink and Ptolemy* (Derler, Naderlinger, Pree, Resmerita, Templ)
- 2012: *The Validator Tool Suite* (Resmerita, Pree)
  - 2012: *Coordinating AUTOSAR runnable entities using Giotto - first concepts* (Belau, von Hasseln, Simons)



A History of Execution Time (contd.)

PARIS LODRON UNIVERSITÄT SALZBURG

- 2001: Giotto (Henzinger, Horrowitz, Kirsch)
- 2004: TDL (Pree, Templ)
- 2008: *Simulation of LET Models in Simulink and Ptolemy* (Derler, Naderlinger, Pree, Resmerita, Templ)
- 2012: *The Validator Tool Suite* (Resmerita, Pree)
  - 2012: *Coordinating AUTOSAR runnable entities using Giotto - first concepts* (Belau, von Hasseln, Simons)
- 2015: *Applying Real-time Programming to Legacy Embedded Control Software* (Resmerita, Naderlinger, Huber, Butts, Pree)

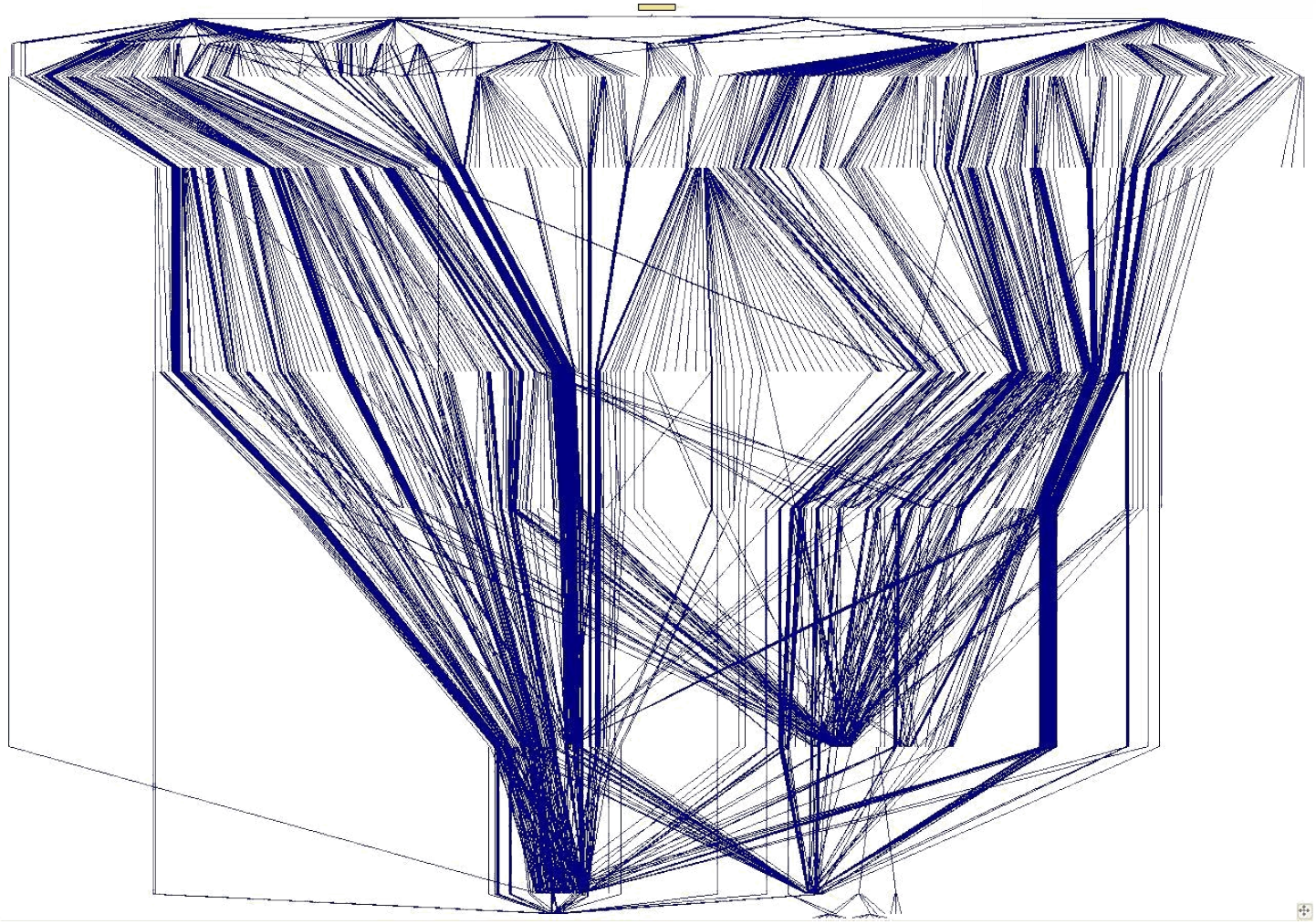


2015: LET for Toyota's Engine Control Software

PARIS LODRON UNIVERSITÄT SALZBURG

- Million+ lines of code
- Event-triggered tasks

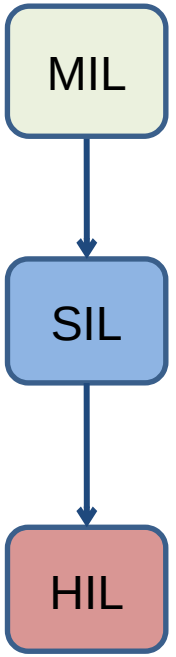
- 8 LET tasks
- 2000+ task ports
- 450 additional buffers
- RAM increase: 0.37%
- ROM increase: 1.3%
- CPU utilization: up 3.4%
- Same I/O behavior
- Increased robustness



A History of Execution Time (contd.)

PARIS LODRON UNIVERSITÄT SALZBURG

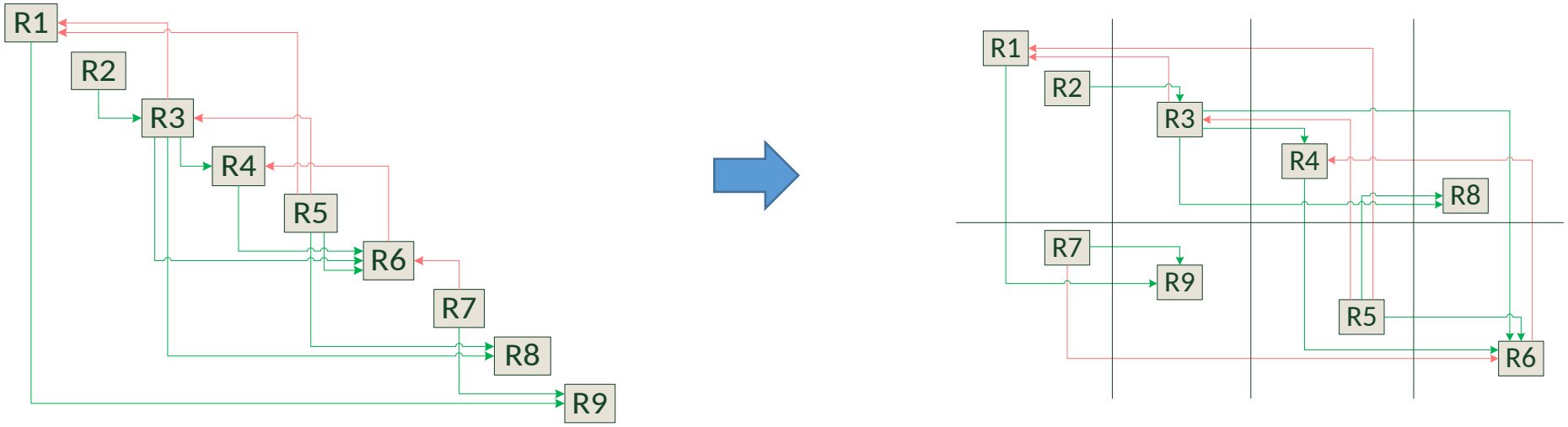
- 2001: Giotto (Henzinger, Horrowitz, Kirsch)
- 2004: TDL (Pree, Templ)
- 2008: *Simulation of LET Models in Simulink and Ptolemy* (Derler, Naderlinger, Pree, Resmerita, Templ)
- 2012: *The Validator Tool Suite* (Resmerita, Pree)
  - 2012: *Coordinating AUTOSAR runnable entities using Giotto - first concepts* (Belau, von Hasseln, Simons)
- 2015: *Applying Real-time Programming to Legacy Embedded Control Software* (Resmerita, Naderlinger, Huber, Butts, Pree)
- 2016: *Towards Parallelizing Legacy Embedded Control Software Using the LET Programming Paradigm* (Hennig, …, Resmerita, Lukesch, Naderlinger)



2016: LET for Daimler's Central Powertrain Control Software (CPC)

PARIS LODRON UNIVERSITÄT SALZBURG

- Migration of CPC-SW from single-core to multi-core with minimal changes in the application
- CPC: One 10ms task, hundreds of top-level functions

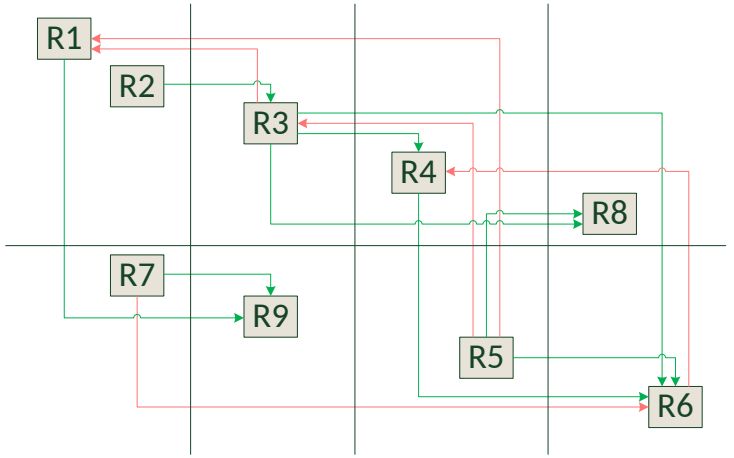


2016: LET for Daimler's Central Powertrain Control Software (CPC)

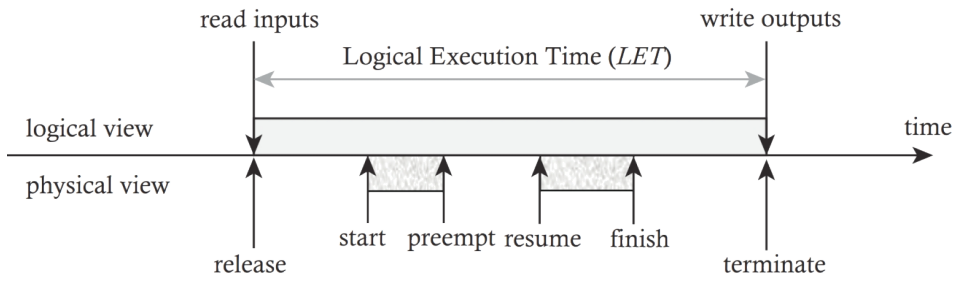
PARIS LODRON UNIVERSITÄT SALZBURG

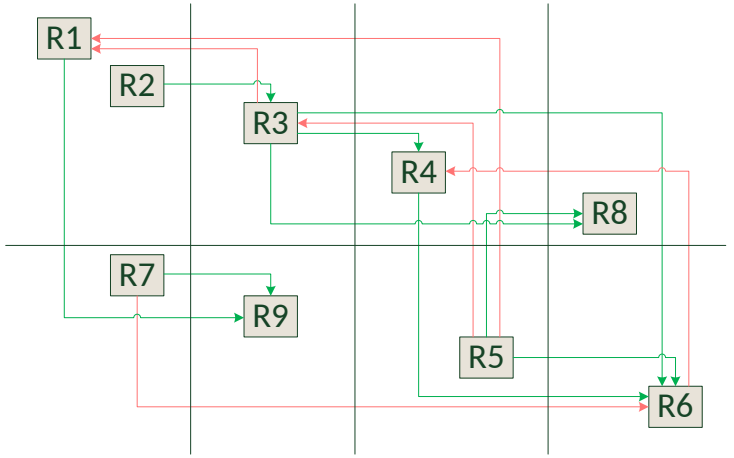
- Migration of CPC-SW from single-core to multi-core with minimal changes in the application
- CPC: One 10ms task, hundreds of top-level functions

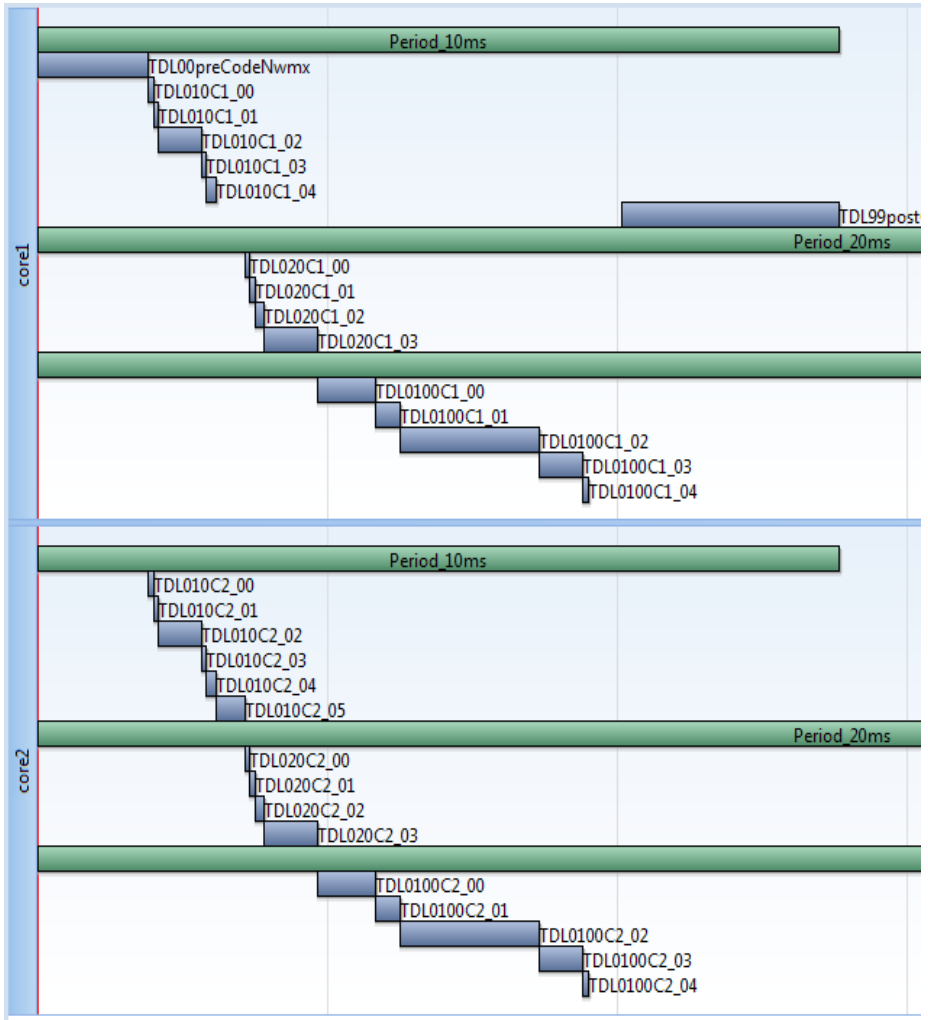
- **Clear, intuitive, standard-supported specification of parallel behavior**
- **Built-in robustness**



2016: LET for Daimler's CPC







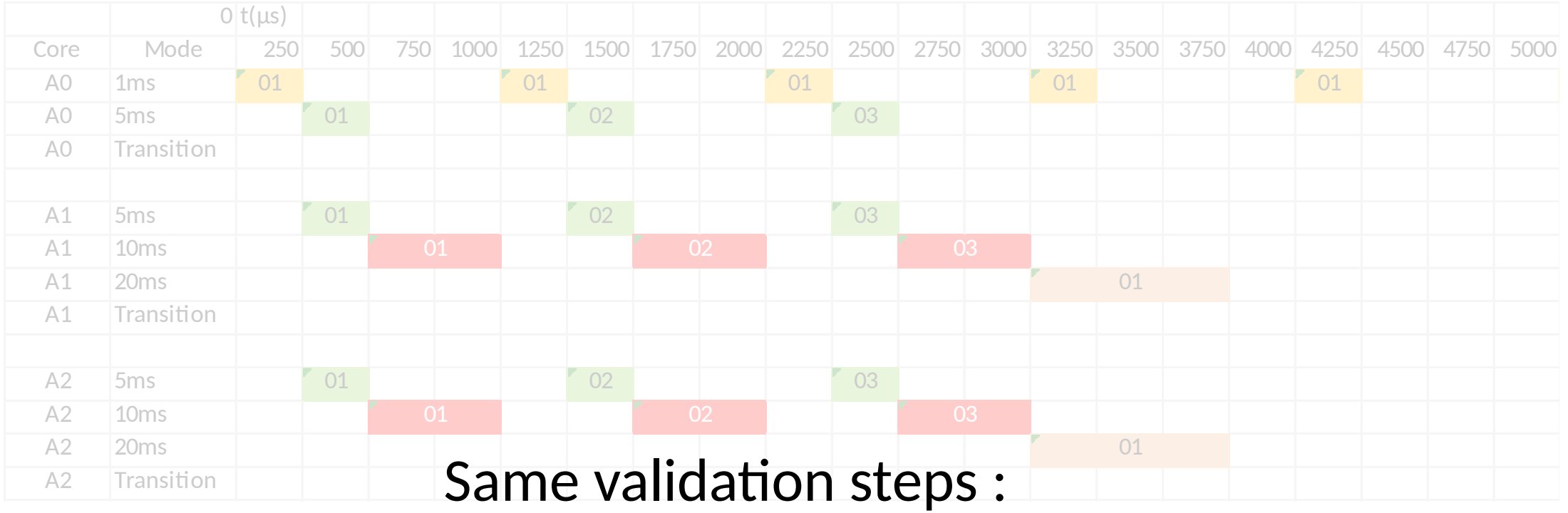
2016: LET @Continental

PARIS LODRON UNIVERSITÄT SALZBURG

	0 t(µs)																			
Core	Mode	250	500	750	1000	1250	1500	1750	2000	2250	2500	2750	3000	3250	3500	3750	4000	4250	4500	4750	5000
A0	1ms	01				01				01				01				01			
A0	5ms		01				02				03										
A0	Transition																				
A1	5ms		01				02				03										
A1	10ms			01				02				03									
A1	20ms													01							
A1	Transition																				
A2	5ms		01				02				03										
A2	10ms			01				02				03									
A2	20ms													01							
A2	Transition																				

2016: LET at Continental

PARIS LODRON UNIVERSITÄT SALZBURG



Same validation steps :

- SIL with the **Validator** approach
- HIL at Continental

LET Inside !

- 2018: LET in industry standard AUTOSAR 4.4
- 2021: LET-based Central Powertrain Software in Mercedes EQA

LET Inside !

- 2018: LET in industry standard AUTOSAR 4.4
- 2021: LET-based Central Powertrain Software in Mercedes EQA
- 2022: Mercedes-Benz: In the Entry Luxury segment, **the EQA is the top-selling all-electric model with 33,100 units**”

„Alle künftigen zentralen Antriebssteuergeräte werden den LET-Ansatz umsetzen“

Dr. Martin Simons, Mercedes-Benz AG

Our Experience

PARIS LODRON UNIVERSITÄT SALZBURG

- Deal with the hard truth
  - No “moving the goal posts”
  - No “example engineering”
  - Not just “proof of concept” (aka quick-hack) implementation

Our Experience

PARIS LODRON UNIVERSITÄT SALZBURG

- Deal with the hard truth
  - No “moving the goal posts”
  - No “example engineering”
  - Not just “proof of concept” (aka quick-hack) implementation
- Deal with IP concerns
  - Work on-site
  - Generalize (structure, architecture)
  - Reproduce off-site

} Modeling

Our Experience

PARIS LODRON UNIVERSITÄT SALZBURG

- Deal with the hard truth
  - No “moving the goal posts”
  - No “example engineering”
  - Not just “proof of concept” (aka quick-hack) implementation
- Deal with IP concerns
  - Work on-site
  - Generalize (structure, architecture)
  - Reproduce off-site

} Modeling

- Convince people
  - Validation (Pictures)
  - Evaluation (Numbers)

} Simulation

PARIS
LODRON
UNIVERSITÄT
SALZBURG

Thank you!