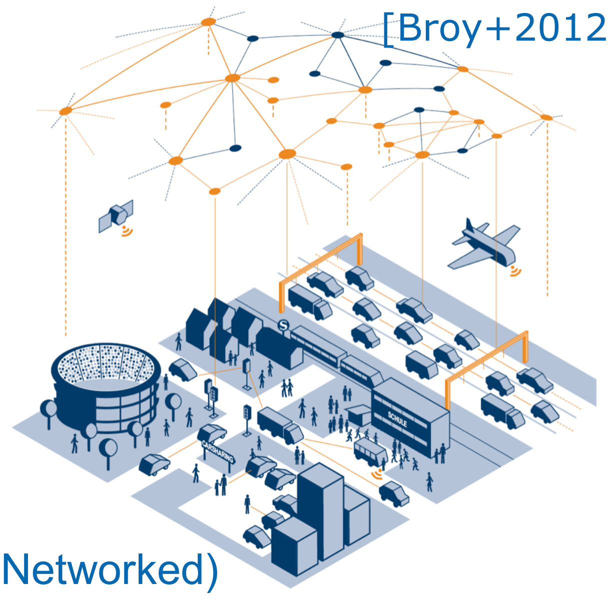
Multi-Paradigm Modeling for Cyber-Physical Systems: Implications for Multidirectional Transformations and Synchronizations

Dagstuhl Seminar 18491 on Multidirectional Transformations and Synchronisations. December 2–7, 2018.

Holger Giese
System Analysis & Modeling Group,
Hasso Plattner Institute, University of Potsdam, Germany
holger.giese@hpi.uni-potsdam.de

1. Cyber-Physical Systems & Integration

[Broy+2012]

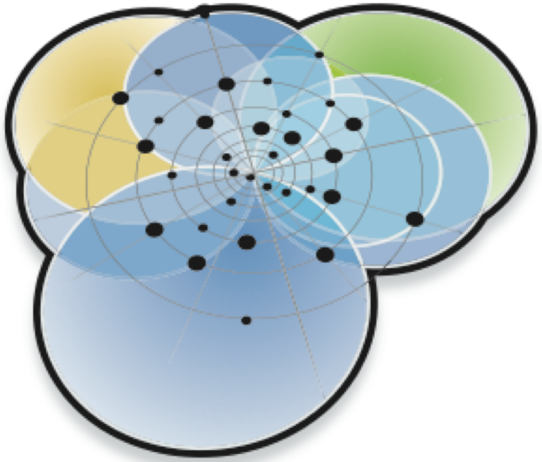


(Networked)
Cyber-Physical Systems

Internet of Things

Smart City

[Northrop+2006]



Ultra-Large-Scale Systems

http://oceanservice.noaa.gov/news/weeklynews/nov13/ioos-awards.html

Smart Home

Smart Factory -
E.g. Industry 4.0

E-Health

Smart Logistic

Micro Grids

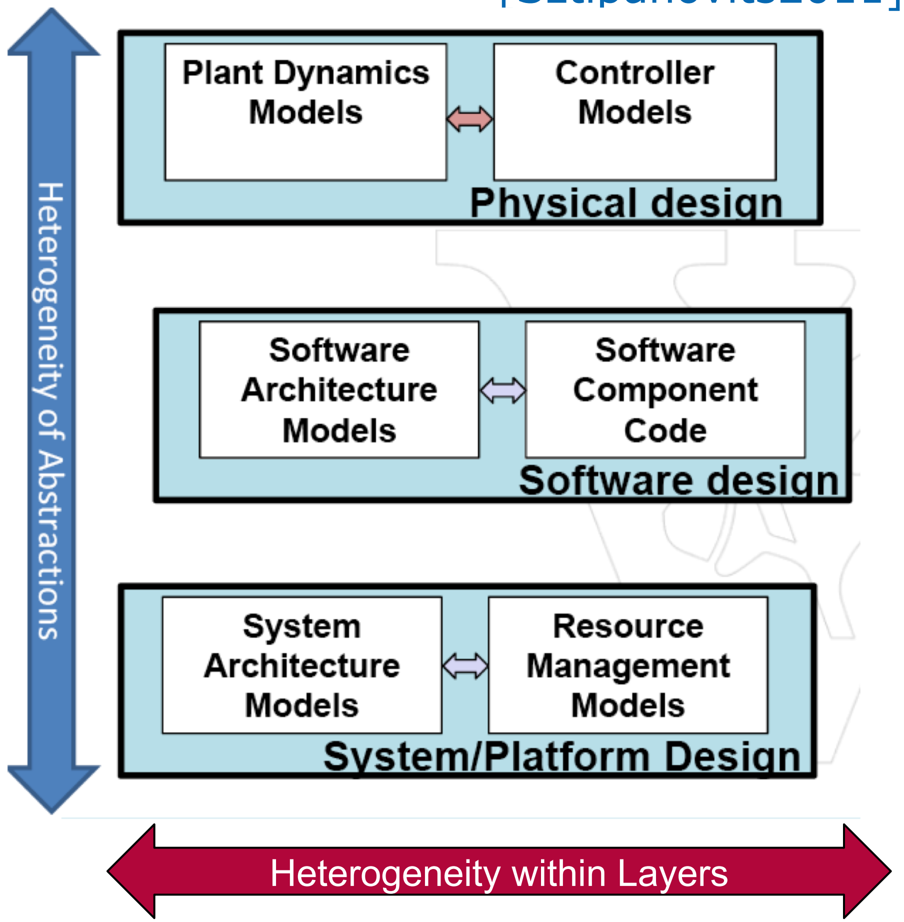
COST Action IC1404 for Multi-Paradigm Modeling for Cyber-Physical Systems (MPM4CPS)

Ambient
Assisted Living

Challenge: Integrate Models of Computation

- Problem to integrate models within one layer as different **models of computation** are employed
- Leaky abstractions are caused by lack of composability across system layers. Consequences:
  - intractable interactions
  - unpredictable system level behavior
  - full-system verification does not scale

[Sztipanovits2011]

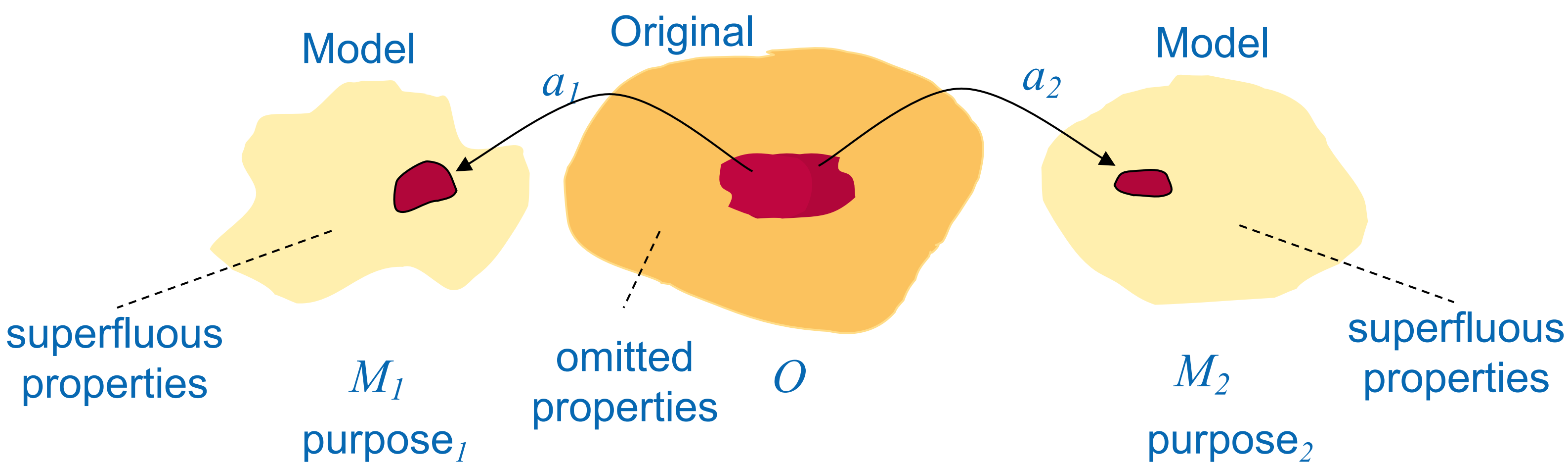


Integration has to cover multiple layers and their **paradigms**

2. Multiple Models and ...
(1) Multiple Models ...

HPI Hasso Plattner Institut

Each **model** M_j is an abstract representations of of a part or multiple parts of an existing or envisioned original used for a specific purpose.

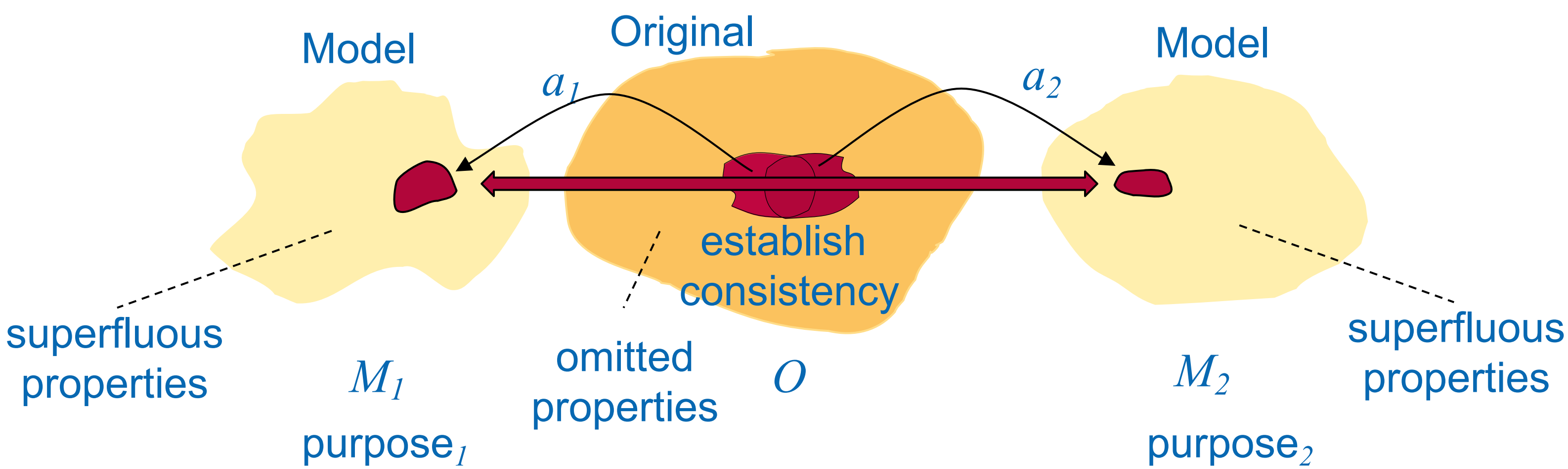


Benefit: For purpose$_j$ we replace the original O by a suitable model M_j that does not contain any irrelevant information (**reduced complexity**!)

Drawback: Does an original O consistent with both models M_1 and M_2 really exist (**consistency**)? – simple existence is often not enough!

How to Handle Multiple Models?

Try for each purposes to find a model M_j that replace the original O, does not contain any irrelevant information (reduced complexity!), and **integrate** the models systematically to establish consistency.



Key questions:

- How many models are helpful (tradeoff benefits vs. integration effort)?
- When and how is integration happen for these models?

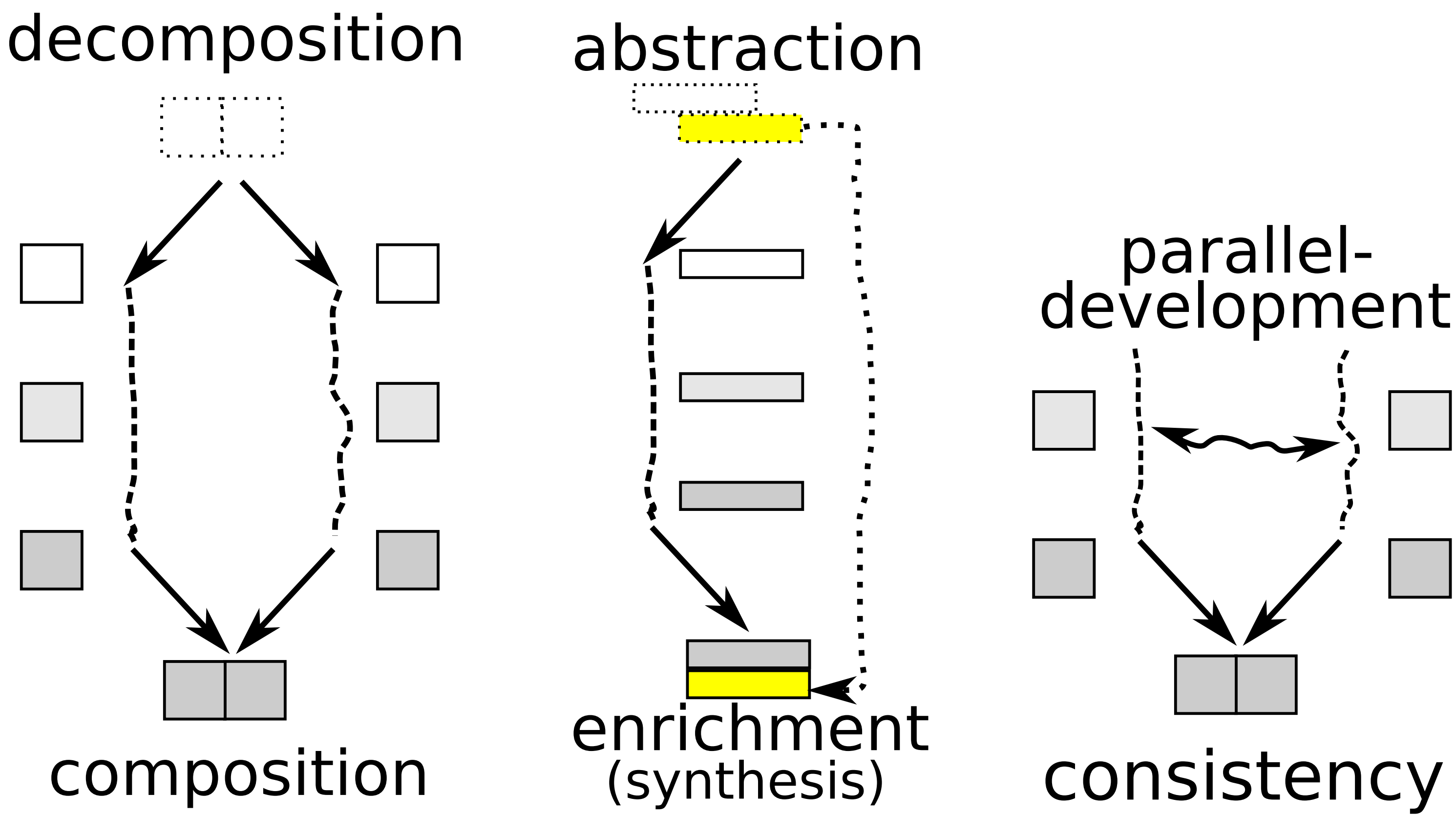
(2) Integration: When & How

Warning: We use a less restricted notion of integration than many others ...

6

Fundamental Techniques for Integration:

[Giese+2011]



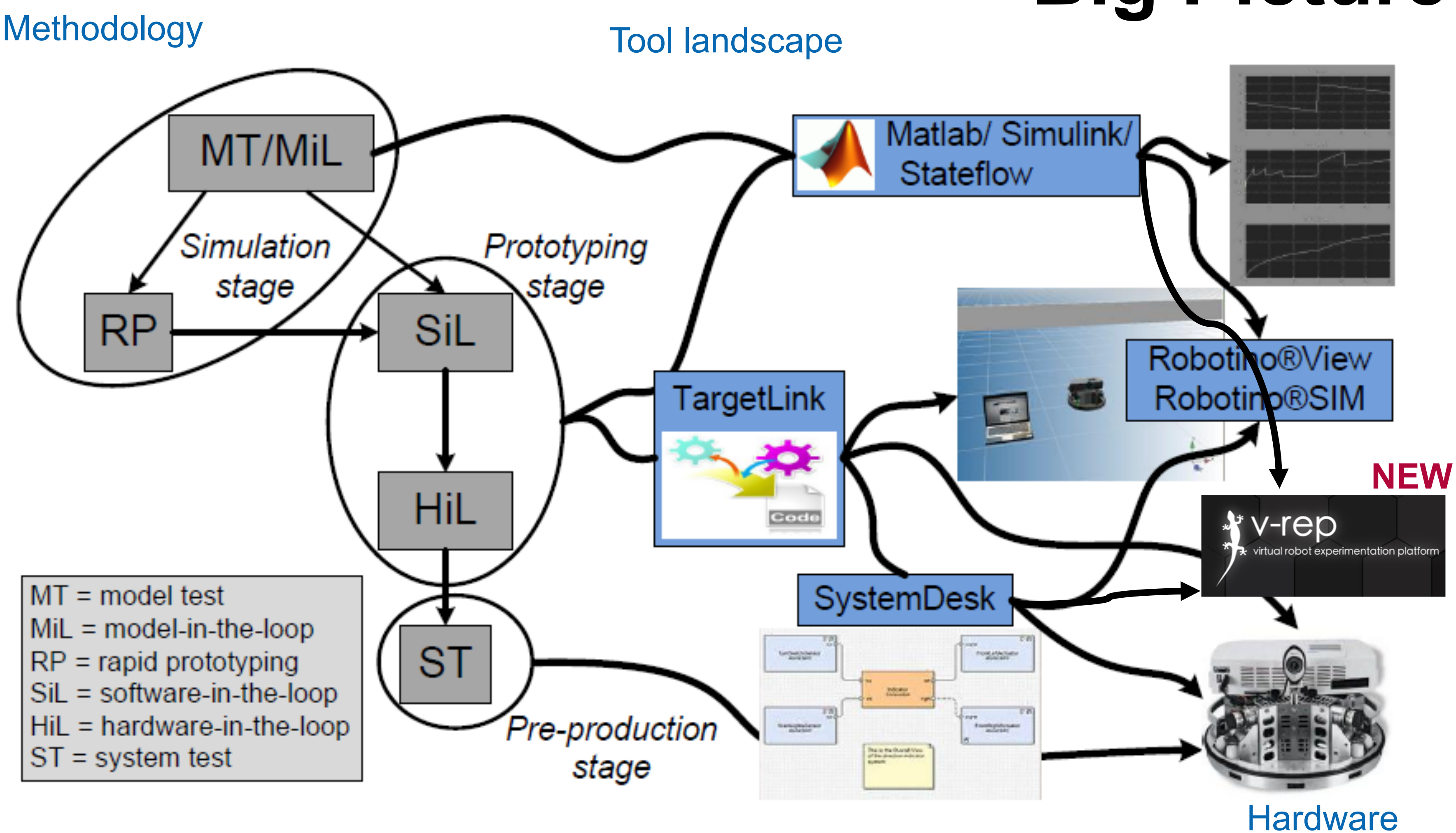
(a) composition (b) abstraction (c) consistency

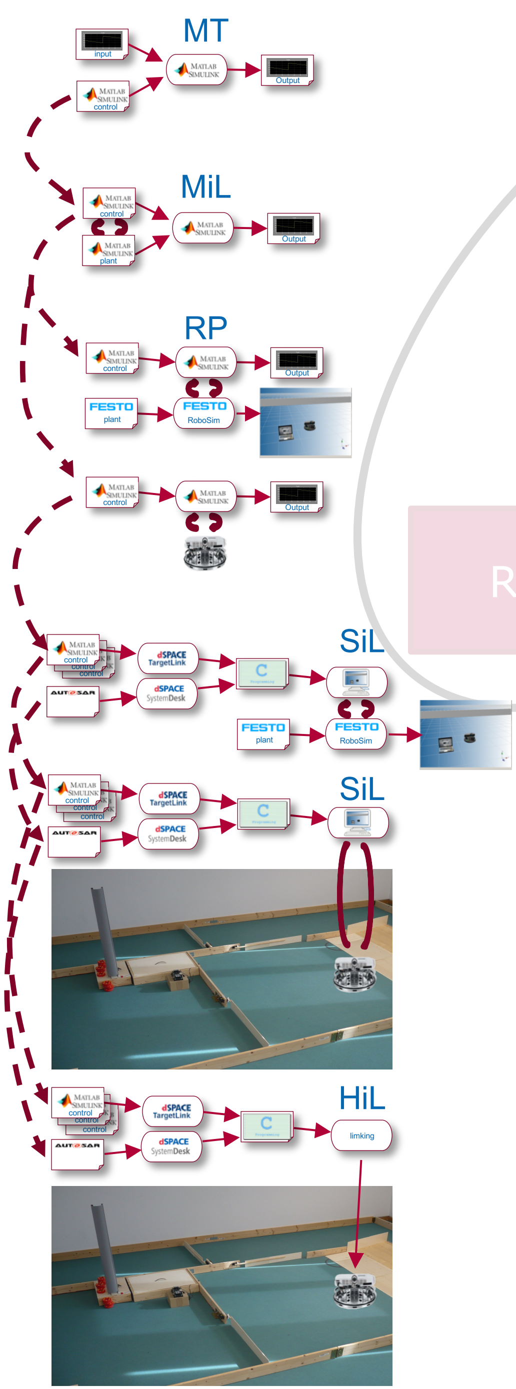
Level of Integration

- **Representation-level**: integration efforts only guarantee that a joint representation is reached
- **Syntax-level**: integration efforts lead to correct syntax
- **Semantics-level**: integration efforts lead to compatibility at the level of the semantics

3. CPSLab & Integration:

Big Picture

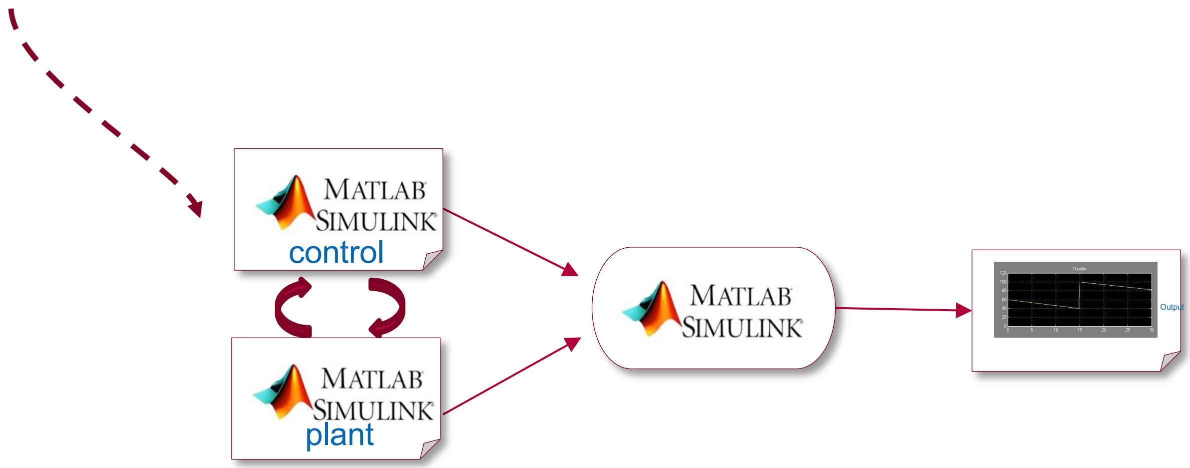


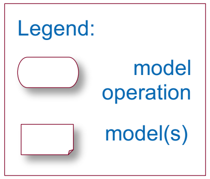


Legend:
- tool
- model

- Vertical **enrichment** of functional models (consistency manually)
- Horizontal **integration** of functional and plant models
- Horizontal **integration** of multiple functional models, an architecture model, and a plant model
- Vertical **enrichment** of multiple functional models, an architecture model, and a plant model (to realize functions while meeting resource constraints)

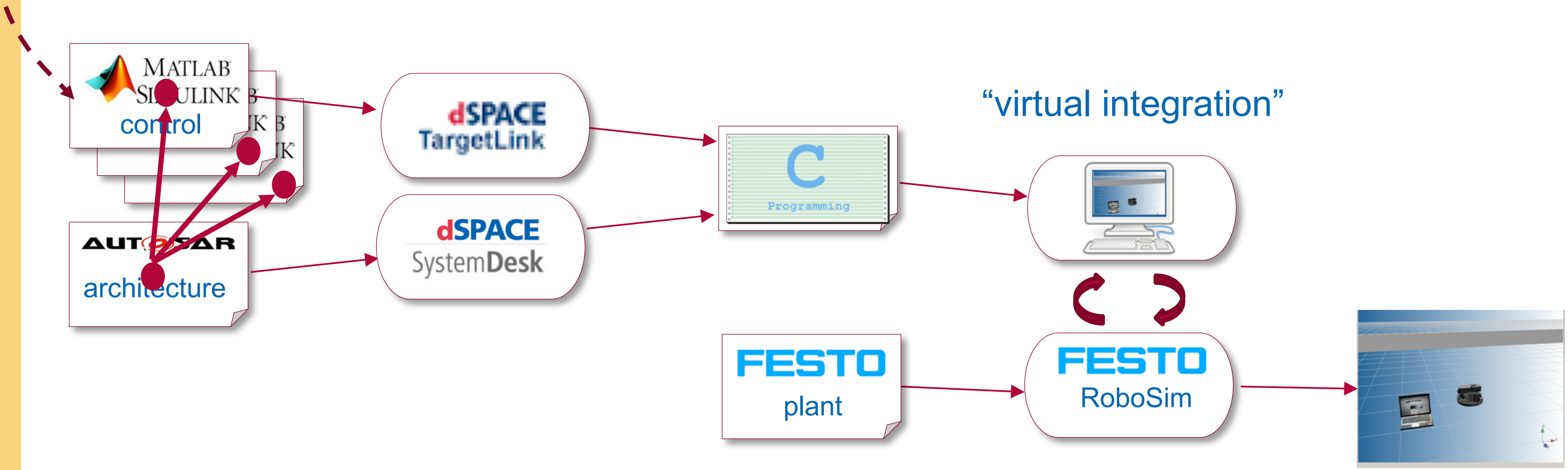
Model in the Loop (MiL)





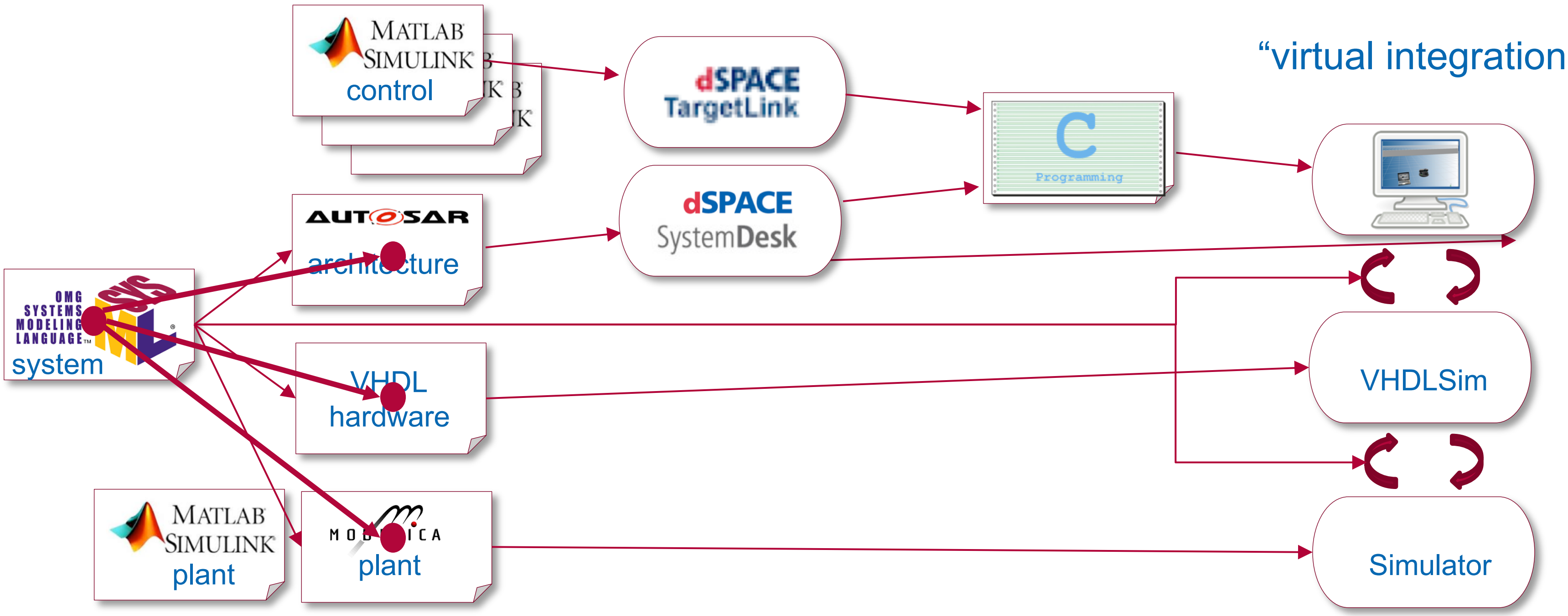
- Layer: Abstract Control Algorithm + Idealized Plant
- Domain: Control/Software + Physics
- Multi-Paradigm: Yes, if control is discrete
- Cyber-Physical system: Yes, as control is cyber world and plant is from the physical world
- Integration: Decomposition & Composition + parallel development; semantics-level

Scenario: Complex Horizontal Integration



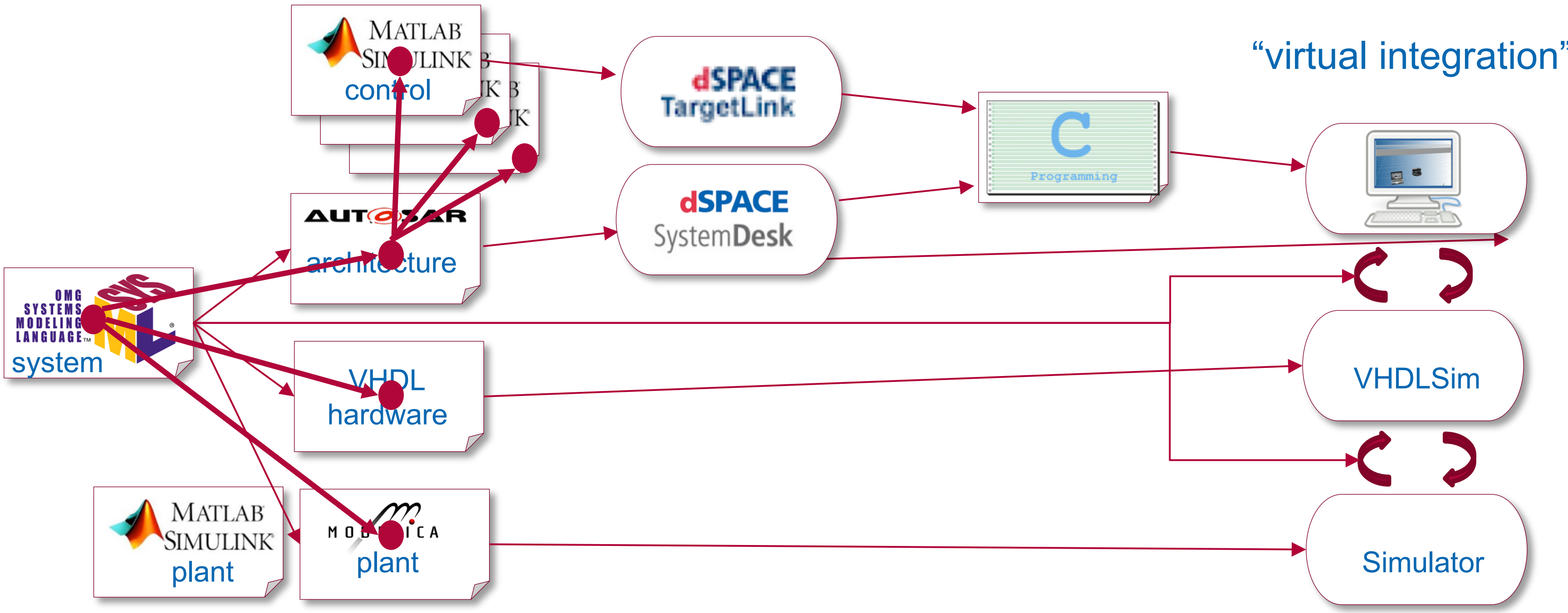
- Horizontal combination of multiple functional models by the architecture via the generated software (integration by composition for functions, integration by abstraction for OS)
- Downwards propagation can be expected, but must be managed
- Upwards propagation is usually forbidden (suppressed)
- Horizontal propagation is therefore also forbidden (suppressed)

Scenario: More Complex Horizontal Integration



- Horizontal combination of multiple specific structures (Autosar: software; VHDL: hardware, Matlab/Modelica: plant) via a generic structure (SysML)
- Downwards propagation can be expected, but must be managed
- Upwards propagation is usually forbidden (suppressed)
- Horizontal propagation is therefore also forbidden (suppressed)

Scenario: More Complex Horizontal Integration



- Vertical **decomposition** via a generic system structure (SysML) containing multiple specific structures (Matlab: control; Autosar: software; VHDL: hardware, Matlab/Modelica: plant; ...)
- Consistency between models and in the models interact, which may lead to transitive propagation/conflicts

4. Needs for Integration

Observations:

- A **horizontal composition** is often **mainly** done to establish consistency at the **semantics-level** to ensure that the different models fit together (“virtual integration”). Keep syntax-level consistency throughout the development for a **horizontal composition** of n models (a **multidirectional transformation or synchronization**) is not really an issue.

Implications:

- We can help as **semantics-level checks** for the **horizontal composition** of n models requires **syntax-level consistency** as prerequisite!

Needs for Integration

Observations:

- Often propagation between multiple models (**multidirectional transformation or synchronization**) was not wanted/permitted.

Implications:

- Need for concepts to manage permission to do only changes as permitted (interfaces?)
- To unleash the full potential of **multidirectional transformation or synchronization** we have to study the context (processes, activities, … = mega models / paradigm) and identify how processes and activities can be improved.

Needs for Integration

Observations:

- The overlapping parts are linked to not overlapping parts and therefore **conflicts** may also result w.r.t. not overlapping parts.

Implications:

- For the overlapping parts we cannot expect to achieve more than has been achieved for the **merging of multiple versions** and also related finding may be relevant to us (limits for merging, living with inconsistencies, ...). => **semantics-level** likely not feasible

5. Conclusion & Outlook

For Future MPM4CPS with self-adaptation we get:
- Runtime model sync.
- Executable Runtime Mega Models organizing the sync. and other model operations
- ...

- **Multiple models** and their **integra** developing complex systems
- In case of **cyber-physical systems** it holds:
  - models employ **different paradigms** specific for their **layer**
  - **Integration** of the models is of **paramount importance**

- **Current integration challenges:**
  - Build cost-effectively tools to integrate the models at the **semantics-level** (not only syntax-level) for a "virtual integration" to also support analysis of **emergent properties**
  - **Multidirectional transformation and synchronization** may establish **syntax-level consistency** throughout the development to enable **automated semantics-level integration checks**

Bibliography (1/3)

[Broman+2012] David Broman, Edward A. Lee, Stavros Tripakis and Martin Torngren. Viewpoints, Formalisms, Languages, and Tools for Cyber-physical Systems. In Proceedings of the 6th International Workshop on Multi-Paradigm Modeling, Pages 49--54, ACM, New York, NY, USA, 2012.

[Brooks+2008] Christopher Brooks, Chihhong Cheng, Thomas Huining Feng, Edward A. Lee and Reinhard von Hanxleden. Model Engineering using Multimodeling. In 1st International Workshop on Model Co-Evolution and Consistency Management (MCCM '08), September 2008.

[Broy+2012] Manfred Broy, MaríaVictoria Cengarle and Eva Geisberger. Cyber-Physical Systems: Imminent Challenges. In Radu Calinescu and David Garlan editors, Large-Scale Complex IT Systems. Development, Operation and Management, Vol. 7539:1-28 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2012.

[Becker+2006] Basil Becker, Dirk Beyer, Holger Giese, Florian Klein and Daniela Schilling. Symbolic Invariant Verification for Systems with Dynamic Structural Adaptation. In Proc. of the 28th International Conference on Software Engineering (ICSE), Shanghai, China, ACM Press, 2006.

[Becker&Giese2008] Basil Becker and Holger Giese. On Safe Service-Oriented Real-Time Coordination for Autonomous Vehicles. In In Proc. of 11th International Symposium on Object/component/service-oriented Real-time distributed Computing (ISORC), Pages 203--210, IEEE Computer Society Press, 5-7 May 2008.

[Becker&Giese2012] Basil Becker and Holger Giese. Cyber-Physical Systems with Dynamic Structure: Towards Modeling and Verification of Inductive Invariants. Technical report, 64, Hasso Plattner Institute at the University of Potsdam, Germany, 2012.

[Burmester+2008] Sven Burmester, Holger Giese, Eckehard Münch, Oliver Oberschelp, Florian Klein and Peter Scheideler. Tool Support for the Design of Self-Optimizing Mechatronic Multi-Agent Systems. In International Journal on Software Tools for Technology Transfer (STTT), Vol. 10(3):207-222, Springer Verlag, June 2008.

[Giese+2010] Holger Giese, Stefan Neumann and Stephan Hildebrandt. Model Synchronization at Work: Keeping SysML and AUTOSAR Models Consistent. In Gregor Engels, Claus Lewerentz, Wilhelm Schäfer, Andy Schürr and B. Westfechtel editors, Graph Transformations and Model Driven Enginering - Essays Dedicated to Manfred Nagl on the Occasion of his 65th Birthday, Vol. 5765:555-579 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2010.

Bibliography (2/3)

Hasso Plattner Institut

[Giese+2011] Holger Giese, Stefan Henkler and Martin Hirsch. A multi-paradigm approach supporting the modular execution of reconfigurable hybrid systems. In Transactions of the Society for Modeling and Simulation International, SIMULATION, Vol. 87(9):775-808, 2011.

[Giese+2015] Holger Giese, Thomas Vogel and Sebastian Wätzoldt. Towards Smart Systems of Systems. In Mehdi Dastani and Marjan Sirjani editors, Proceedings of the 6th International Conference on Fundamentals of Software Engineering (FSEN '15), Vol. 9392:1--29 of Lecture Notes in Computer Science (LNCS), Springer, 2015.

[Giese&Schäfer2013] Holger Giese and Wilhelm Schäfer. Model-Driven Development of Safe Self-Optimizing Mechatronic Systems with MechatronicUML. In Javier Camara, Rogério de Lemos, Carlo Ghezzi and AntÃ³nia Lopes editors, Assurances for Self-Adaptive Systems, Vol. 7740:152-186 of Lecture Notes in Computer Science (LNCS), Springer, January 2013.

[Ghezzi2012| Carlo Ghezzi. Evolution, Adaptation, and the Quest for Incrementality. In Radu Calinescu and David Garlan editors, Large-Scale Complex IT Systems. Development, Operation and Management, Vol. 7539:369-379 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2012.

[Krause&Giese2012] Christian Krause and Holger Giese. Probabilistic Graph Transformation Systems. In Proceedings of Intern. Conf. on Graph Transformation (ICGT' 12), Vol. 7562:311-325 of Lecture Notes in Computer Science, Springer-Verlag, 2012.

[Maier1998] Mark W. Maier. Architecting principles for systems-of-systems. In Systems Engineering, Vol. 1(4):267--284, John Wiley & Sons, Inc., 1998.

[Maximova2018] Maria Maximova, Holger Giese and Christian Krause. Probabilistic Timed Graph Transformation Systems. In Journal of Logical and Algebraic Methods in Programming, Vol. 101:110 - 131, 2018.

[Northrop+2006] Northrop, Linda, et al. Ultra-Large-Scale Systems: The Software Challenge of the Future. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2006.

Bibliography (3/3)

[Pereira+2013] Eloi Pereira, Christoph M. Kirsch, Raja Sengupta and Jo~ao Borges de Sousa. Bigactors - A Model for Structure-aware Computation. In ACM/IEEE 4th International Conference on Cyber-Physical Systems, Pages 199--208, ACM/IEEE, Philadelphia, PA, USA, 2013.

[Sztipanovits2011] Janos Sztipanovits with Ted Bapty, Gabor Karsai and Sandeep Neema. MODEL-INTEGRATION AND CYBER PHYSICAL SYSTEMS: A SEMANTICS PERSPECTIVE. FM 2011, Limerick, Ireland. 22 June 2011

[Sztipanovits+2012] Janos Sztipanovits, Xenofon Koutsoukos, Gabor Karsai, Nicholas Kottenstette, Panos Antsaklis, Vineet Gupta, B. Goodwine, J. Baras and Shige Wang. Toward a Science of Cyber-Physical System Integration. In Proceedings of the IEEE, Vol. 100(1):29-44, January 2012.

[Valerdi+2008] Ricardo Valerdi, Elliot Axelband, Thomas Baehren, Barry Boehm, Dave Dorenbos, Scott Jackson, Azad Madni, Gerald Nadler, Paul Robitaille and Stan Settles. A research agenda for systems of systems architecting. In International Journal of System of Systems Engineering, Vol. 1(1-2):171--188, 2008.

[Vogel+2009] Thomas Vogel, Stefan Neumann, Stephan Hildebrandt, Holger Giese and Basil Becker: Model-Driven Architectural Monitoring and Adaptation for Autonomic Systems. In: Proc. of the 6th International Conference on Autonomic Computing and Communications (ICAC'09), Barcelona, Spain, ACM (15-19 June 2009)

[Vogel+2010] Thomas Vogel and Stefan Neumann and Stephan Hildebrandt and Holger Giese and Basil Becker. Incremental Model Synchronization for Efficient Run-Time Monitoring. In Sudipto Ghosh, ed., Models in Software Engineering, Workshops and Symposia at MODELS 2009, Denver, CO, USA, October 4-9, 2009, Reports and Revised Selected Papers, vol. 6002 of Lecture Notes in Computer Science (LNCS), pages 124-139. Springer-Verlag, 4 2010.

[Vogel&Giese2012] Thomas Vogel and Holger Giese. A Language for Feedback Loops in Self-Adaptive Systems: Executable Runtime Megamodels. In Proceedings of the 7th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS 2012), pages 129-138, 6 2012. IEEE Computer Society.