Dealing with Multiple Overlapping Modeling Languages in Complex Model-Driven Development Scenarios

Andreas Seibel

System Analysis and Modeling Group
Hasso Plattner Institute
at the University of Potsdam

andreas.seibel@hpi.uni-potsdam.de

under supervision of
Prof. Dr. Holger Giese

System Analysis and Modeling Group
Hasso Plattner Institute
at the University of Potsdam

holger.giese@hpi.uni-potsdam.de

Agenda

Dealing with Multiple Overlapping Modeling Languages in Complex Model-Driven Development Scenarios

Agenda

- Introduction
- Motivation
- State of the Art
- Our Approach
- Future Work

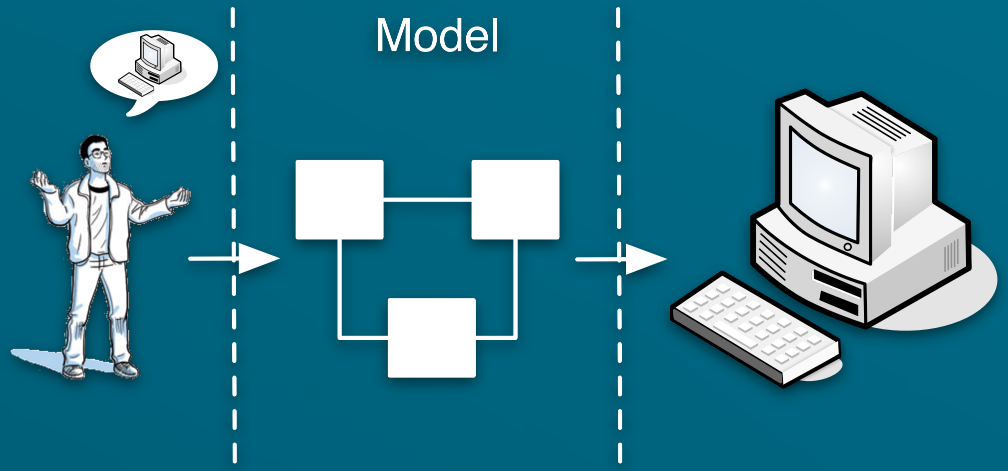
Introduction

Dealing with Multiple Overlapping Modeling Languages in Complex Model-Driven Development Scenarios

What is Model-Driven Development?

„Model-driven development is simply the notion that we can construct a model of a system that we can then transform into the real thing.“

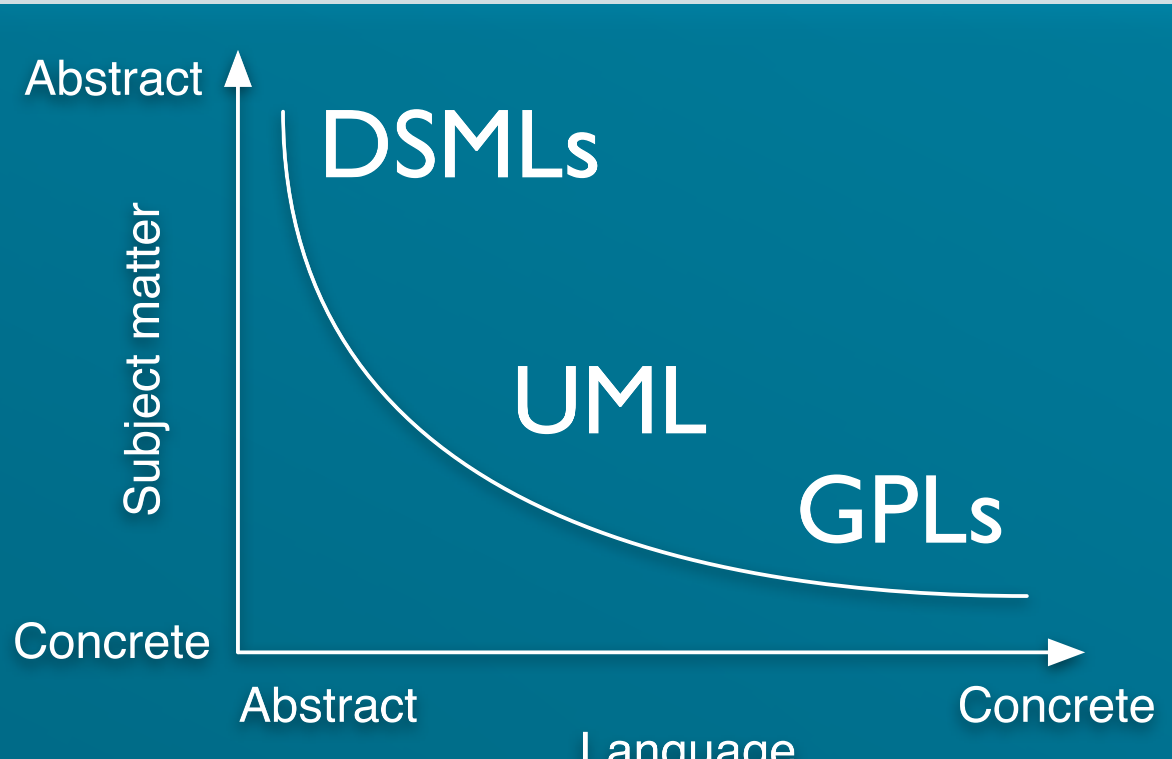
[S.Mellor, 2003]



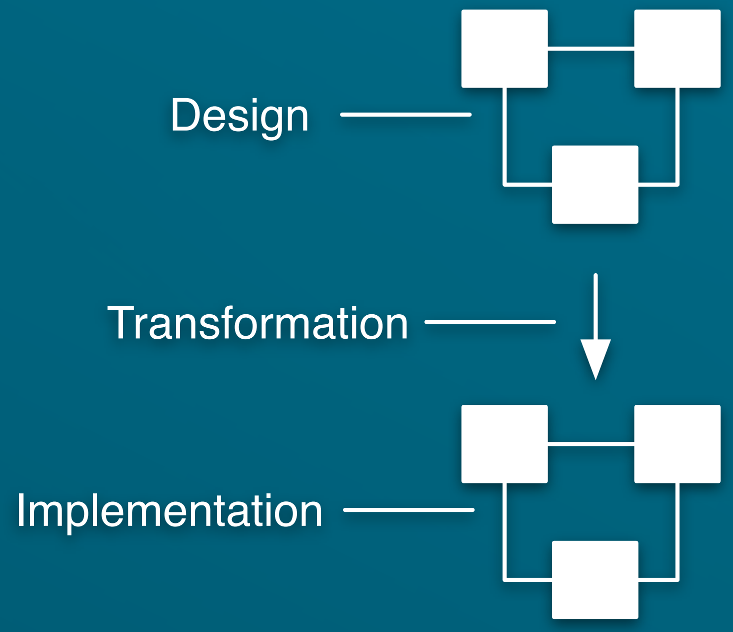
- Does every developer apply model-driven development?
  - Yes and no! Depends on the kinds of models we consider
- We focus on models at higher level of abstraction (above GPLs)

What is Model-Driven Development?

- Level of abstraction
  - The higher the level of abstraction, the more focussed to the problem domain



[S.Mellor, 2003]



- Automation through model operations
  - e.g., employ model transformations to transform models at different levels of abstraction

Motivation

Dealing with Multiple Overlapping Modeling Languages in Complex Model-Driven Development Scenarios

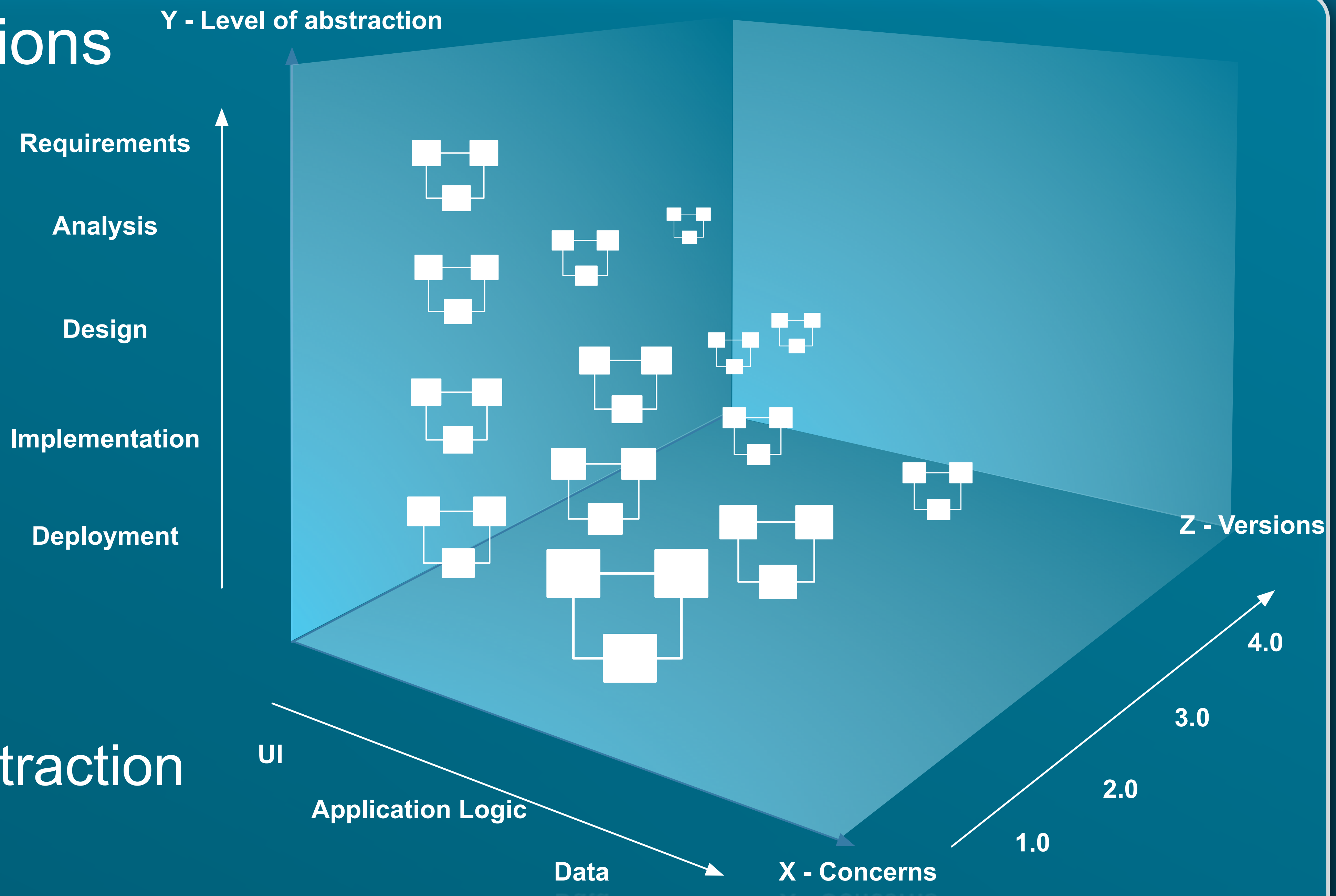
Complexity in MDD: # models

- Three dimensions of models

X - Concerns

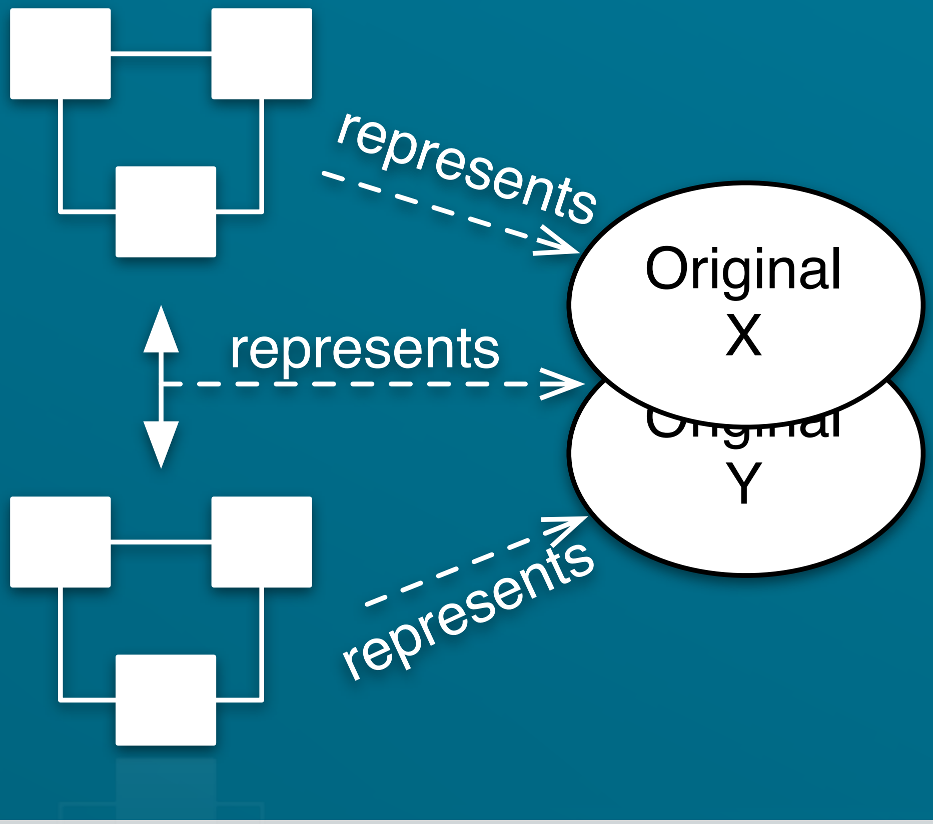
Y - Level of abstraction

Z - Versions



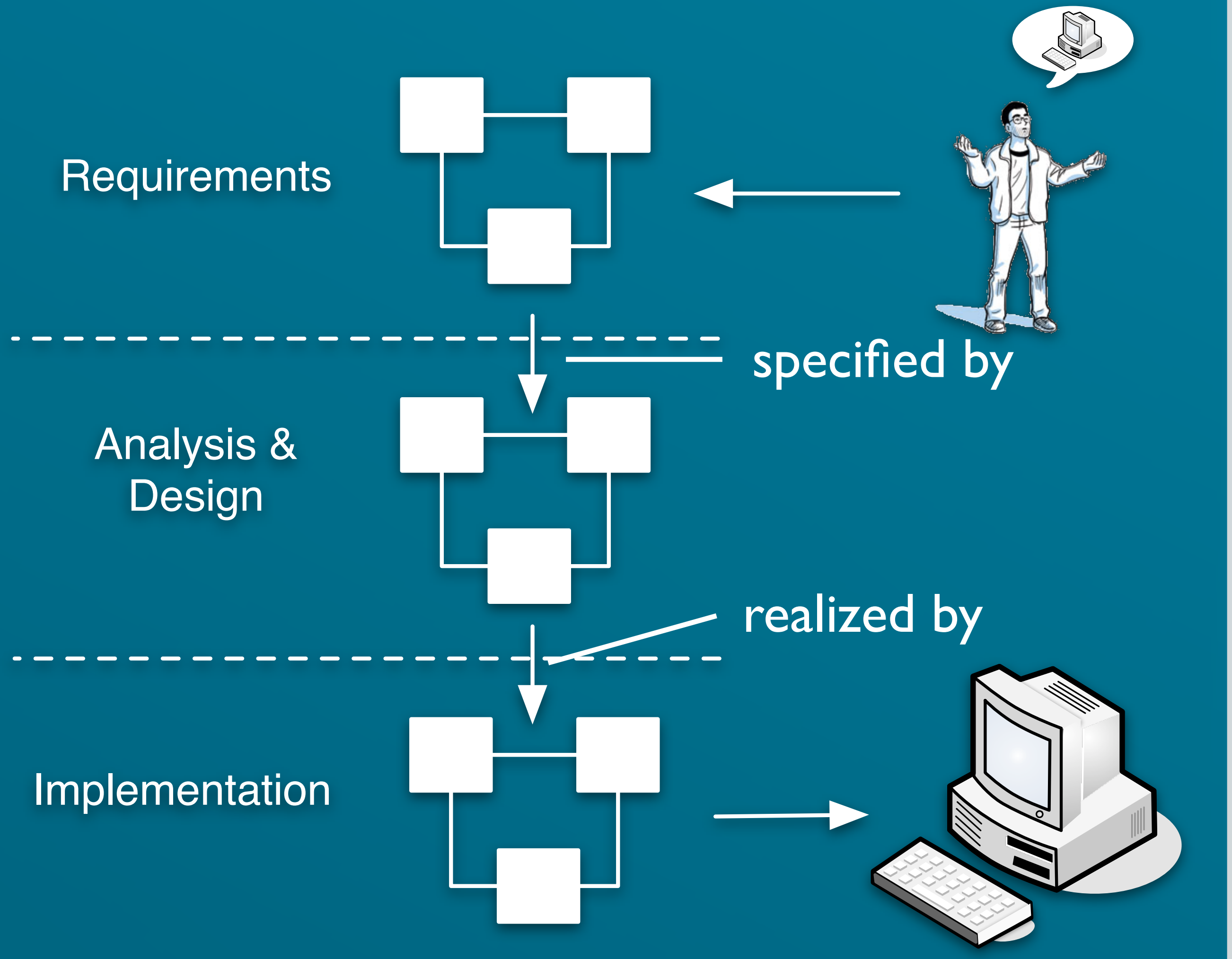
Complexity in MDD: # dependencies

- Are ...
  - … models at different levels of abstraction ...
  - … models representing different concerns …
  - … different version of models ...
- … orthogonal? not necessarily!
- Models usually have overlaps



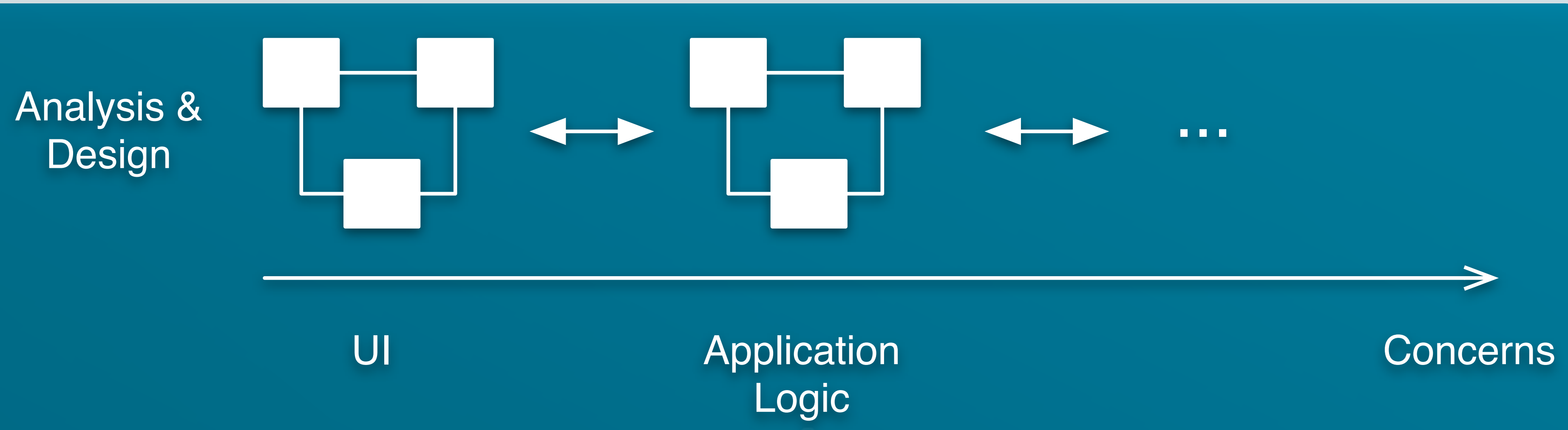
Complexity in MDD: # dependencies

- Vertical dependencies
  - *What?*
    Interrelate different levels of abstraction
  - *Syntax?*
    Dependencies defined between models
  - *Semantic?*
    Application of model transformations (MDA)



Complexity in MDD: # dependencies

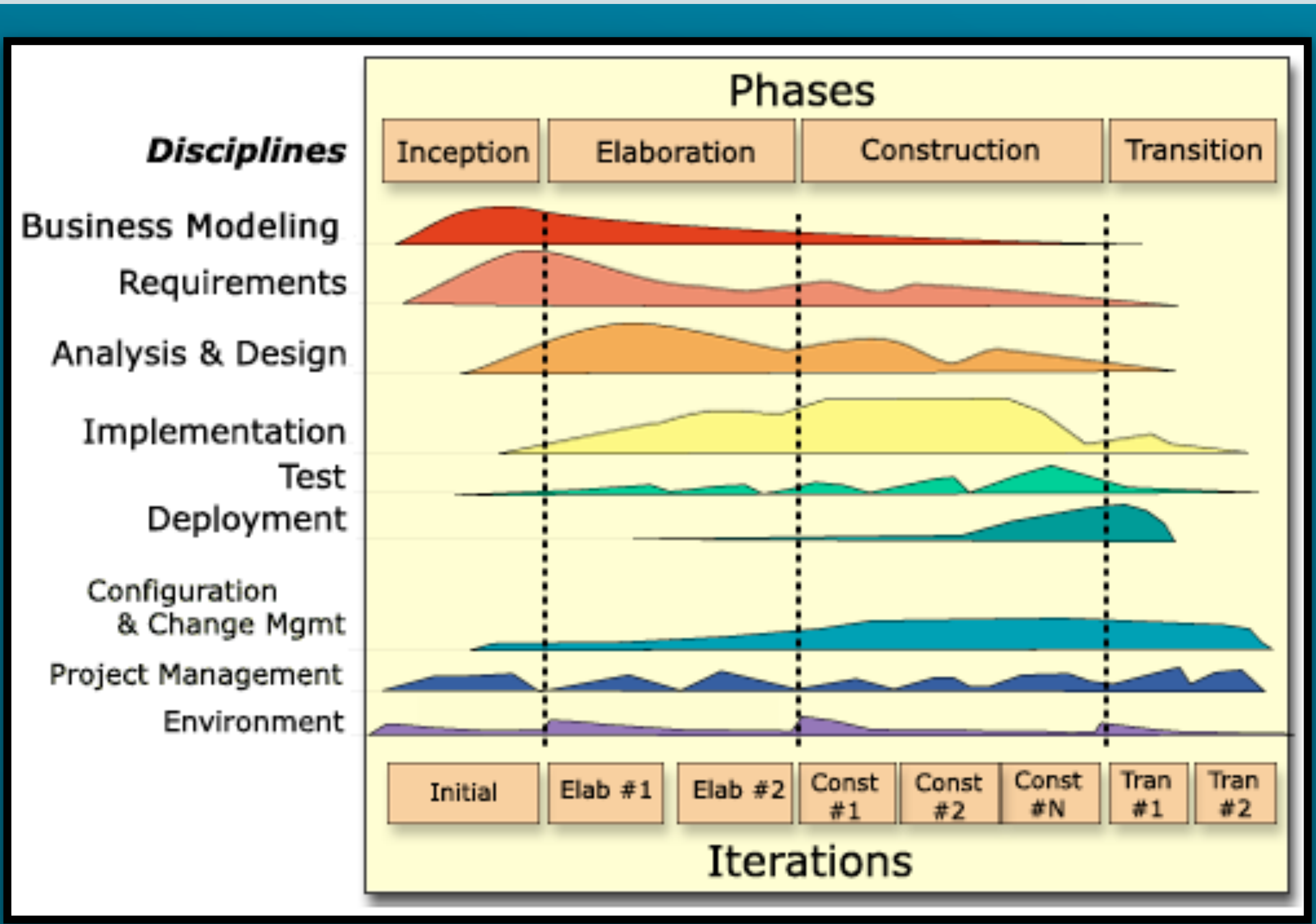
- Horizontal dependencies



  - *What?*
    Interrelate different concerns
  - *Syntax?*
    Dependencies defined between models elements and thus between models
  - *Semantic?*
    Condition between models elements in different models (OCL constraints), etc.

Complexity in MDD: # changes

- An MDD development process is not static but rather a highly dynamic process
- Changes spread over different models

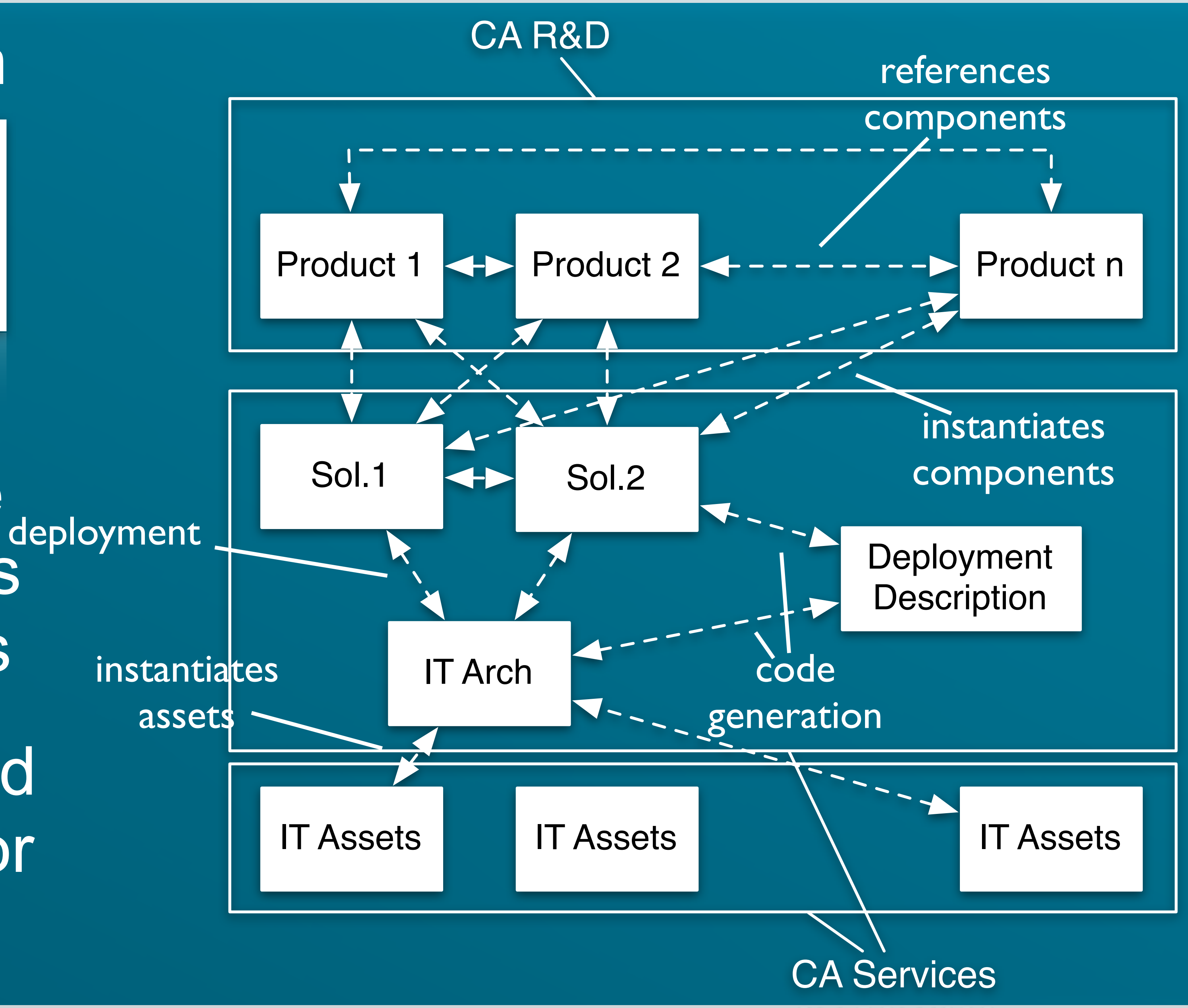


[Jacobson, 1999]

- Changes in one model may have impact on dependent models → *Inconsistencies*

Case Study Scenarios: D-MDA

- In cooperation with CA Labs
- Improve service of deploying software to IT infrastructures of huge companies
- We have developed as set of DSMLs for this task



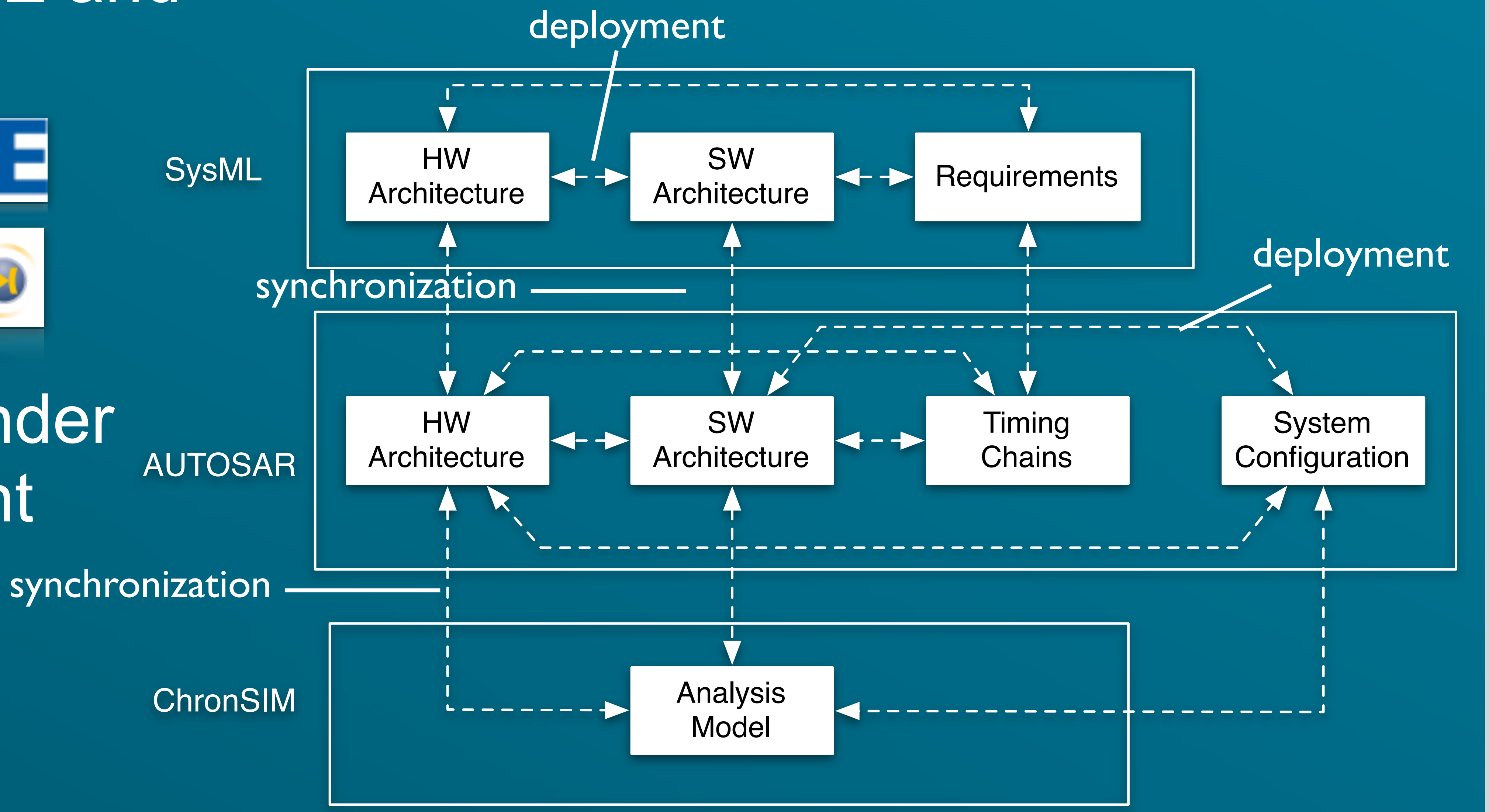
Case Study Scenarios: Embedded Systems Modeling

- Partially in cooperation with dSPACE and INCHRON

dSPACE

INCHRON THINK REAL-TIME

- Currently under development



Motivation

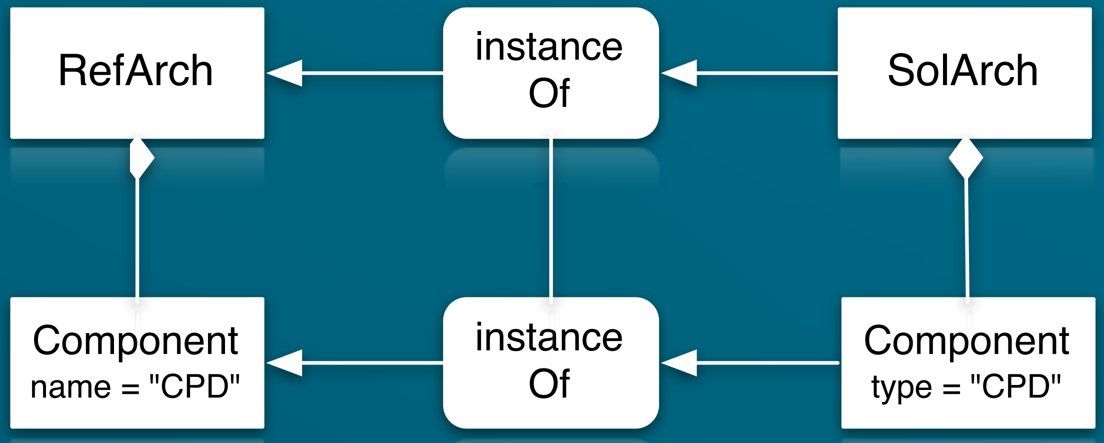
Challenges

- MDD is a complex endeavor
  - Working with many models
  - Models are not orthogonal but have overlaps → dependencies
  - Changes occur frequently and nearly everywhere
    - Inconsistencies may occur
    - Thread to the success of MDD-based projects

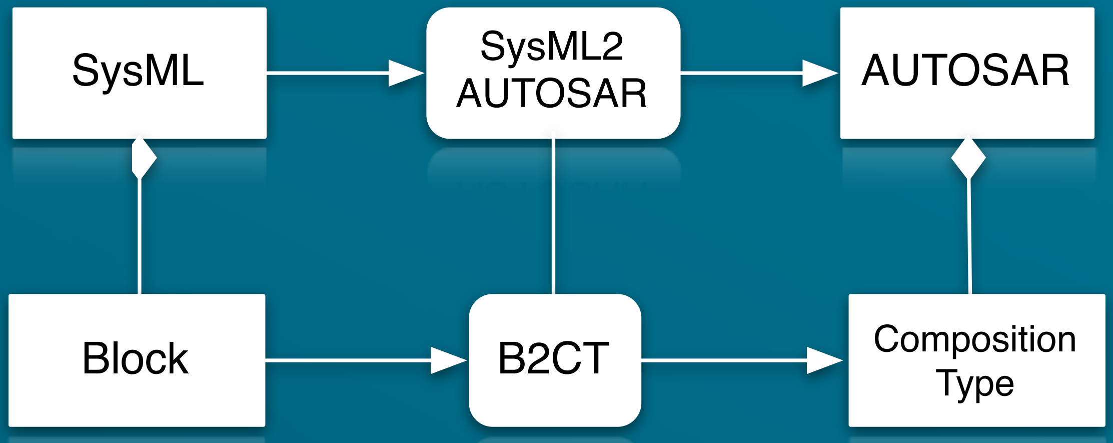
Motivation

What do we need?

- The capability to *explicitly* define dependencies …
  1. … between model elements of different models(microscopic)
  2. … between models (macroscopic)
  3. … between dependencies (combining micro and macro)



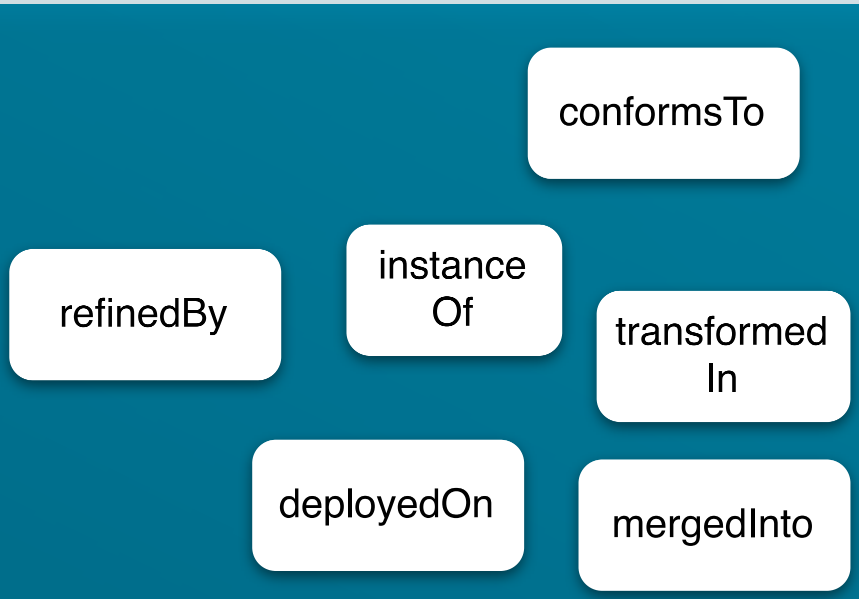
CA case study



dSPACE & INCHRON case study

What do we need?

- Heterogenous dependencies
  - Dependencies should not be restricted in their representation ability



conformsTo

refinedBy

instance Of

transformed In

deployedOn

mergedInto

- Support for automation on a macroscopic/microscopic level
  - Identification of dependencies (*Localization*)
  - Impact analysis
  - (Re-)establishing consistency (*Execution*)

State of the Art

Dealing with Multiple Overlapping Modeling Languages in Complex Model-Driven Development Scenarios

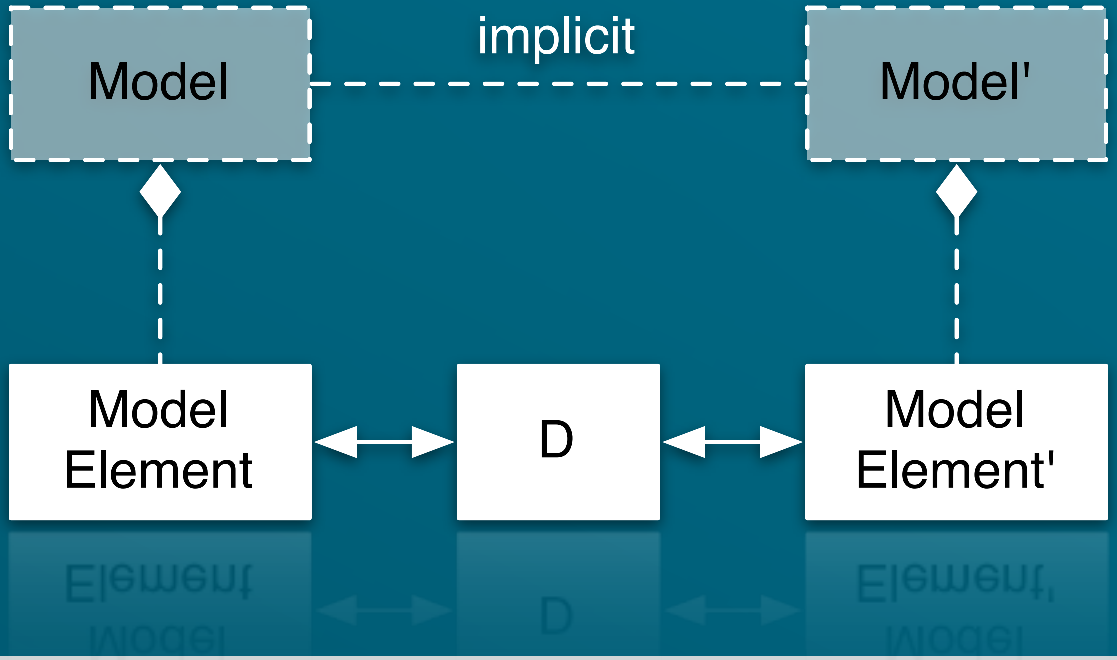
State of the Art

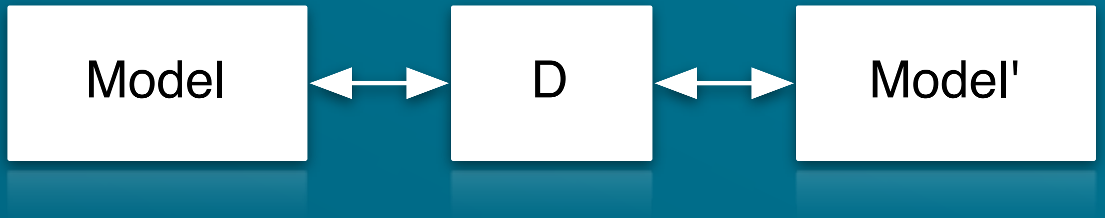
From microscopic to macroscopic

Microscopic approaches ⇔ Macroscopic approaches

- Classical traceability
- Model transformations
- ...

- Megamodeling
- Macromodeling
- Multimodeling
- ...





Direct comparison of approaches

		microscopic	macroscopic	heterogenity	localization	impact analysis	execution
Microscopic Approaches	[N. Drivalos,2008]	x	-	-	-	-	-
	[M.Aleksy,2008]	x	-	x	x	-	-
	[H.Asuncion,2008]	x	-	-	(x)	-	-
	[J.R.Falleri,2006]	x	-	-	x	-	-
	[H.LeDang,2008]	x	-	-	x	-	-
	[B.Vanhooff,2007]	x	-	-	x	-	-
	[N.Aizenbud-Reshef,2005]	x	-	(x)	(x)	(x)	(x)
	[P.Mäder,2008]	x	-	-	(x)	(x)	(x)
	[S.Walderhaug,2006]	x	-	(x)	(x)	(x)	(x)
Macroscopic Approaches	[M.Barbero,2007]	x	x	-	x	-	-
	[R.Salay,2009]	x	x	-	x	x	x
	[M.Del Fabro,2009]		x	-	-	-	(x)
	[M.Fritzsche,2009]	(x)	x	-	(x)	(x)	(x)
	[F.Allilaire,2006]	-	x	-	-	-	(x)
	[J.M.Favre,2005]	-	x	x	-	-	-
	[F.Jouault,2010]	x	x	-	-	-	-
	[P.A.Muller,2009]	-	x	x	-	-	-
	[D.Kolovos,2008]	-	x	x	-	-	x

- not supported x supported (x) mentioned

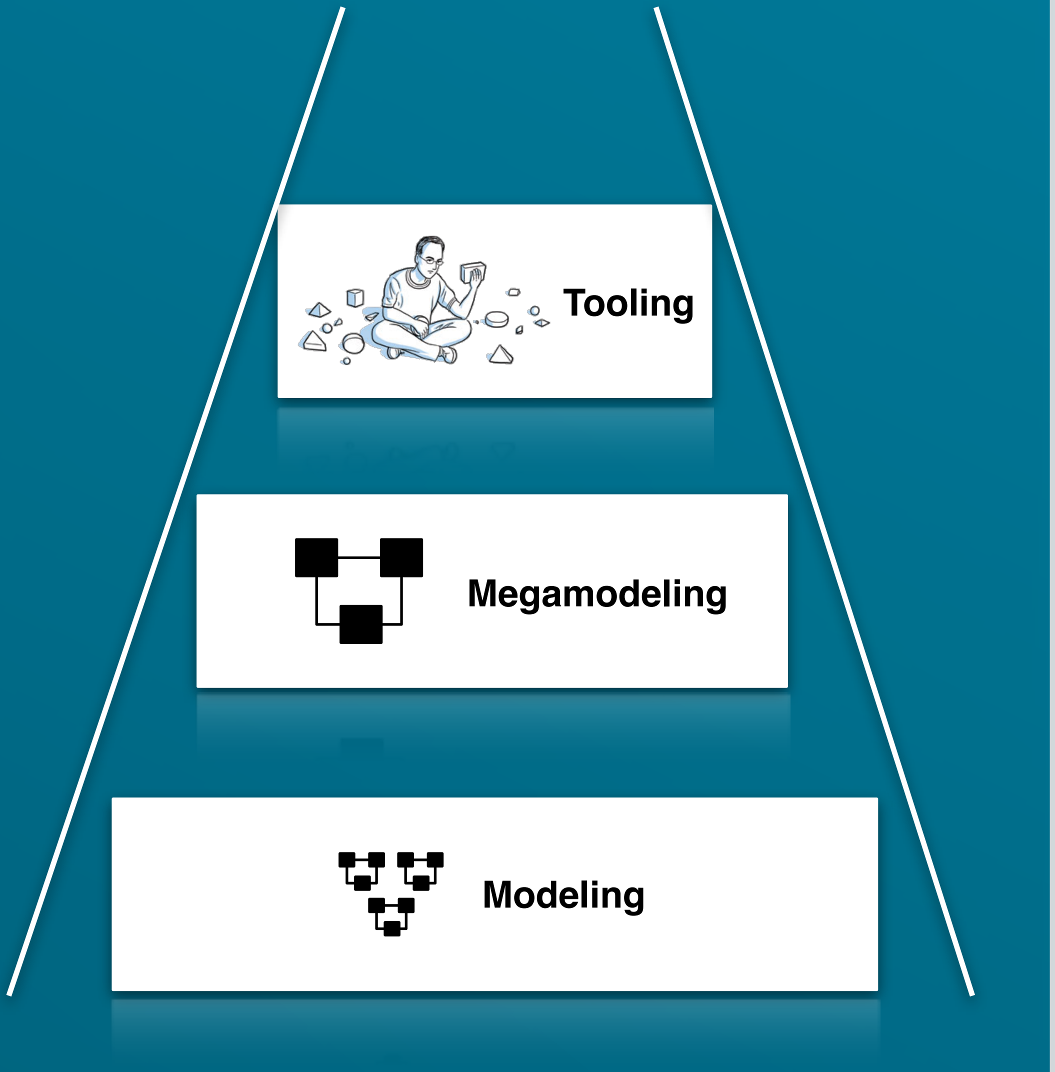
Our Approach

Dealing with Multiple Overlapping Modeling Languages in Complex Model-Driven Development Scenarios

Our Approach

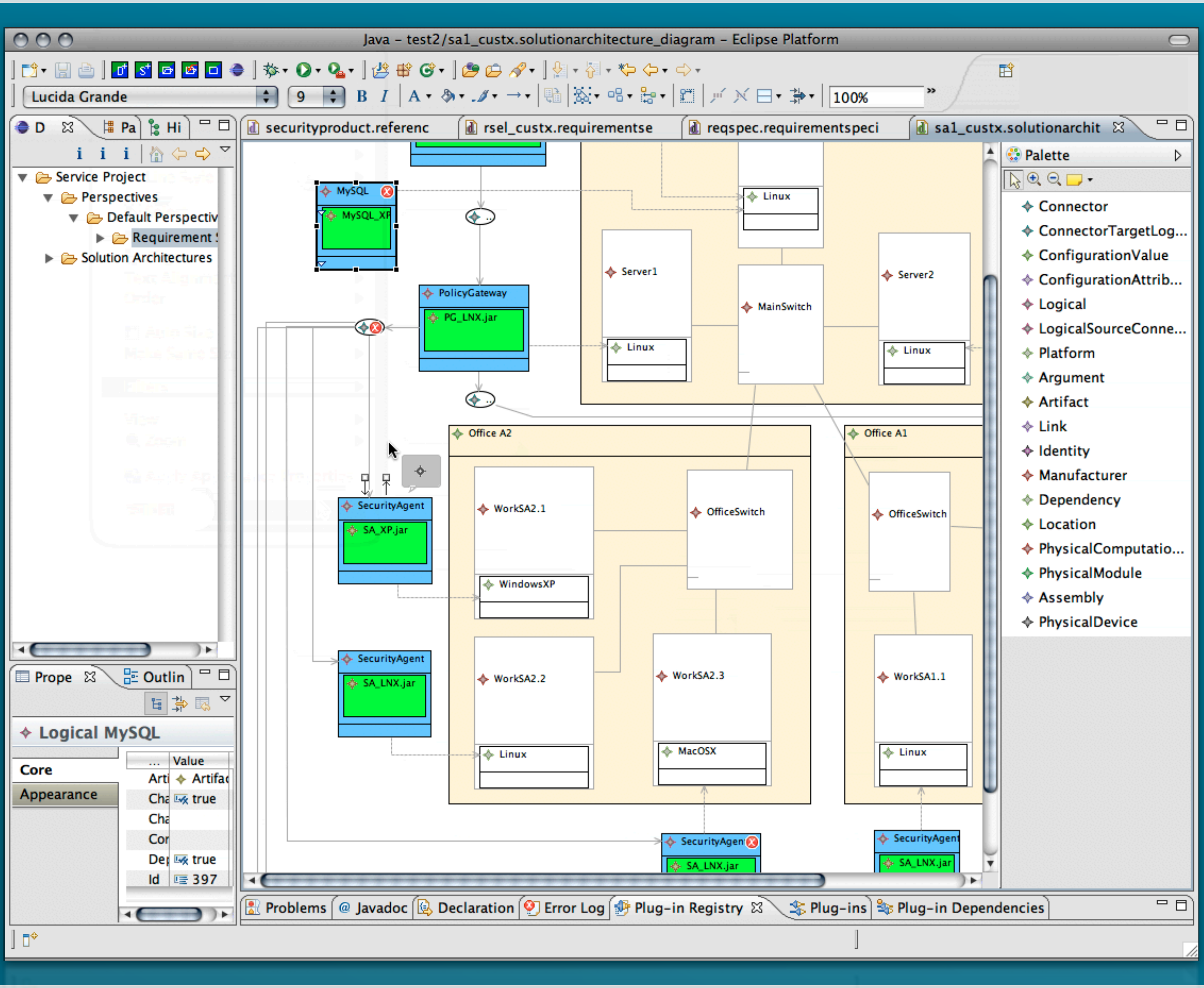
Overview

- Combines microscopic and macroscopic dependencies by extending the notion of megamodels
- **Modeling**:
  not restricted to any specific DSML or even GPML
- **Megamodeling**:
  acts as integration language for DSMLs or even GPML
- **Tooling**:
  supports users using our approach (localization, IA, execution, ...)



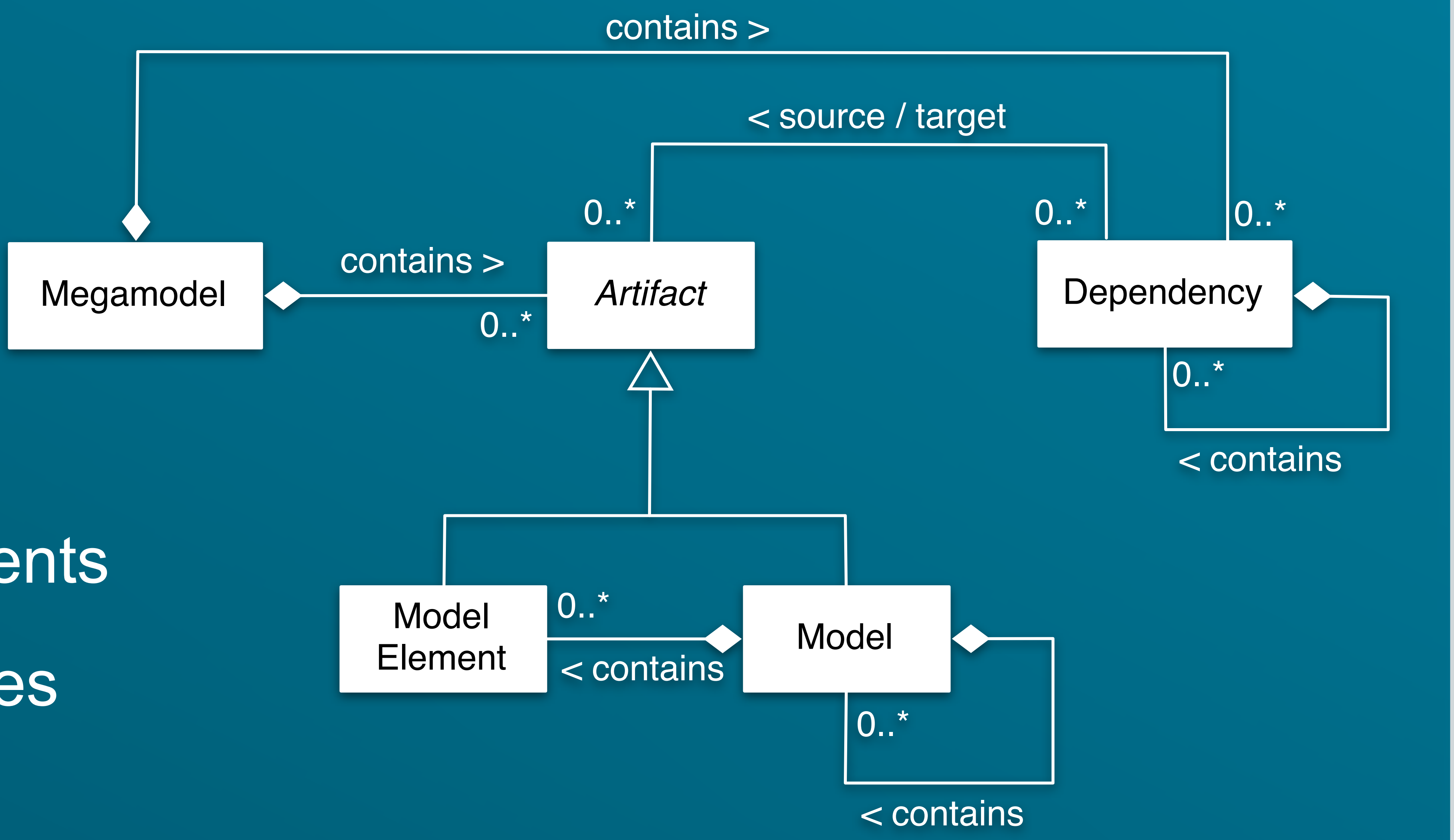
Modeling

- Applying a supported modeling environment (e.g., EMF)
- Make models available in some repository (e.g., workspace in Eclipse) for integration



Megamodeling

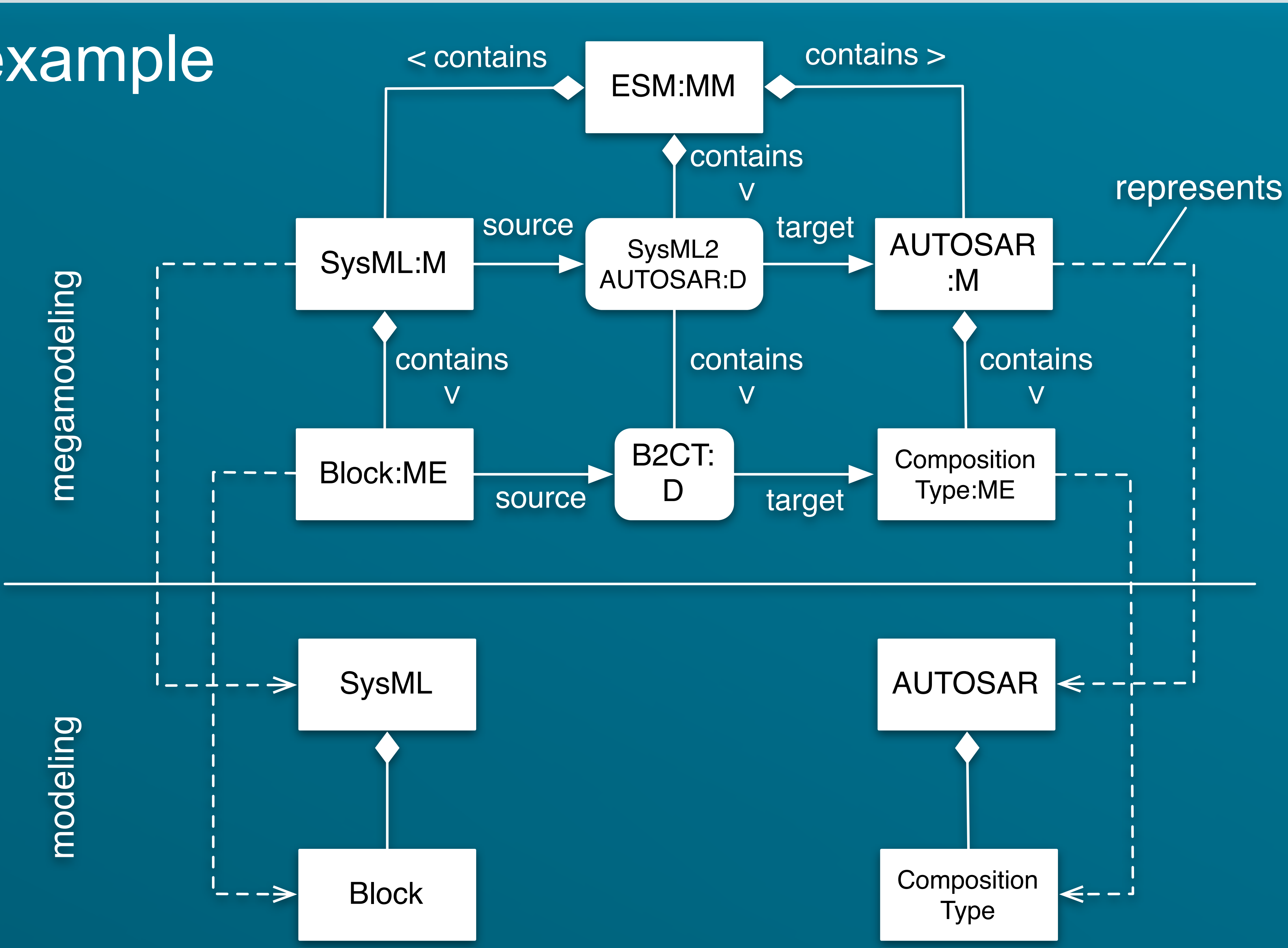
- Metamodel of a megamodel (model of modeling)



- Models
- Model Elements
- Dependencies

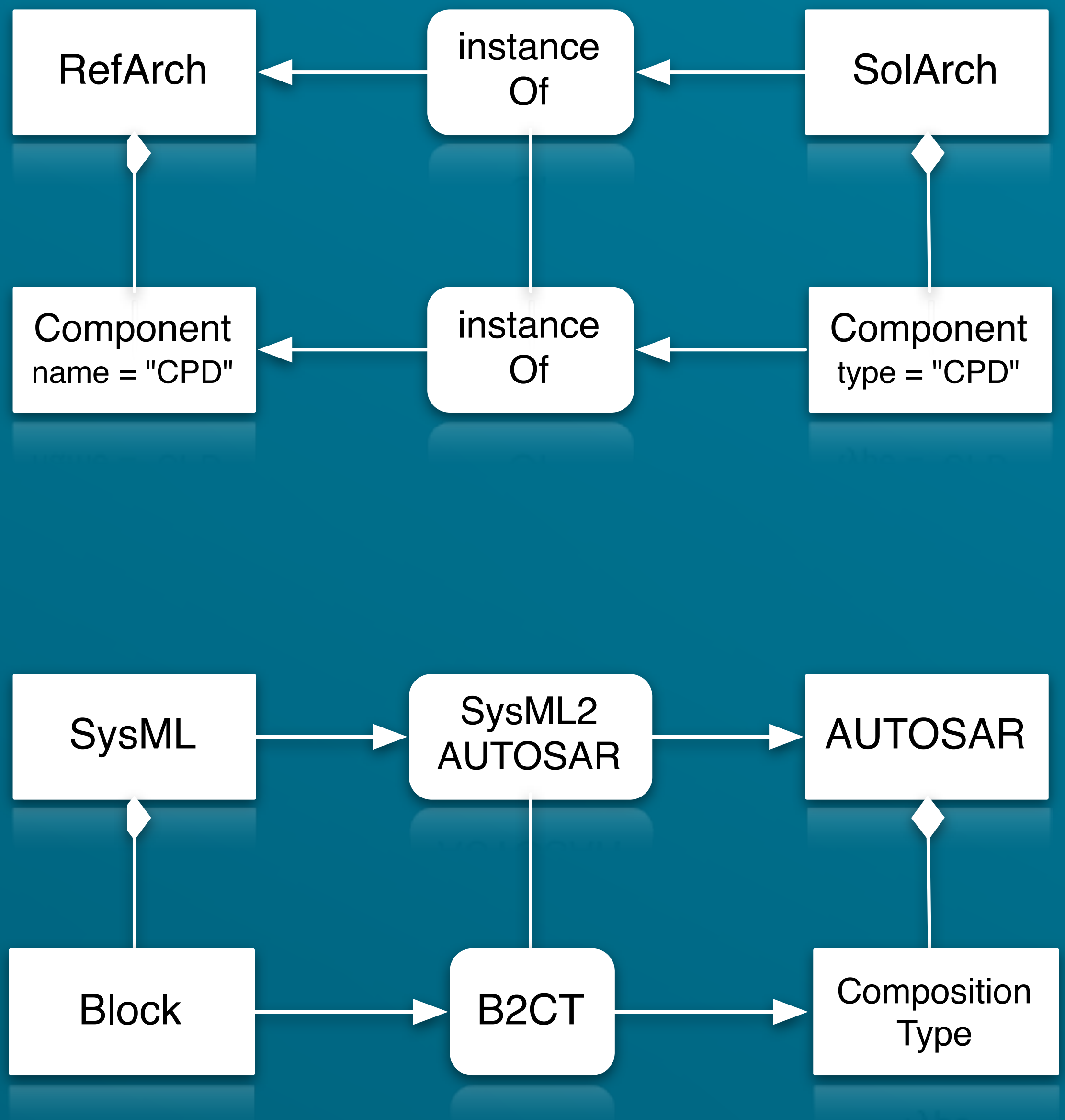
Megamodeling

- Explain by example



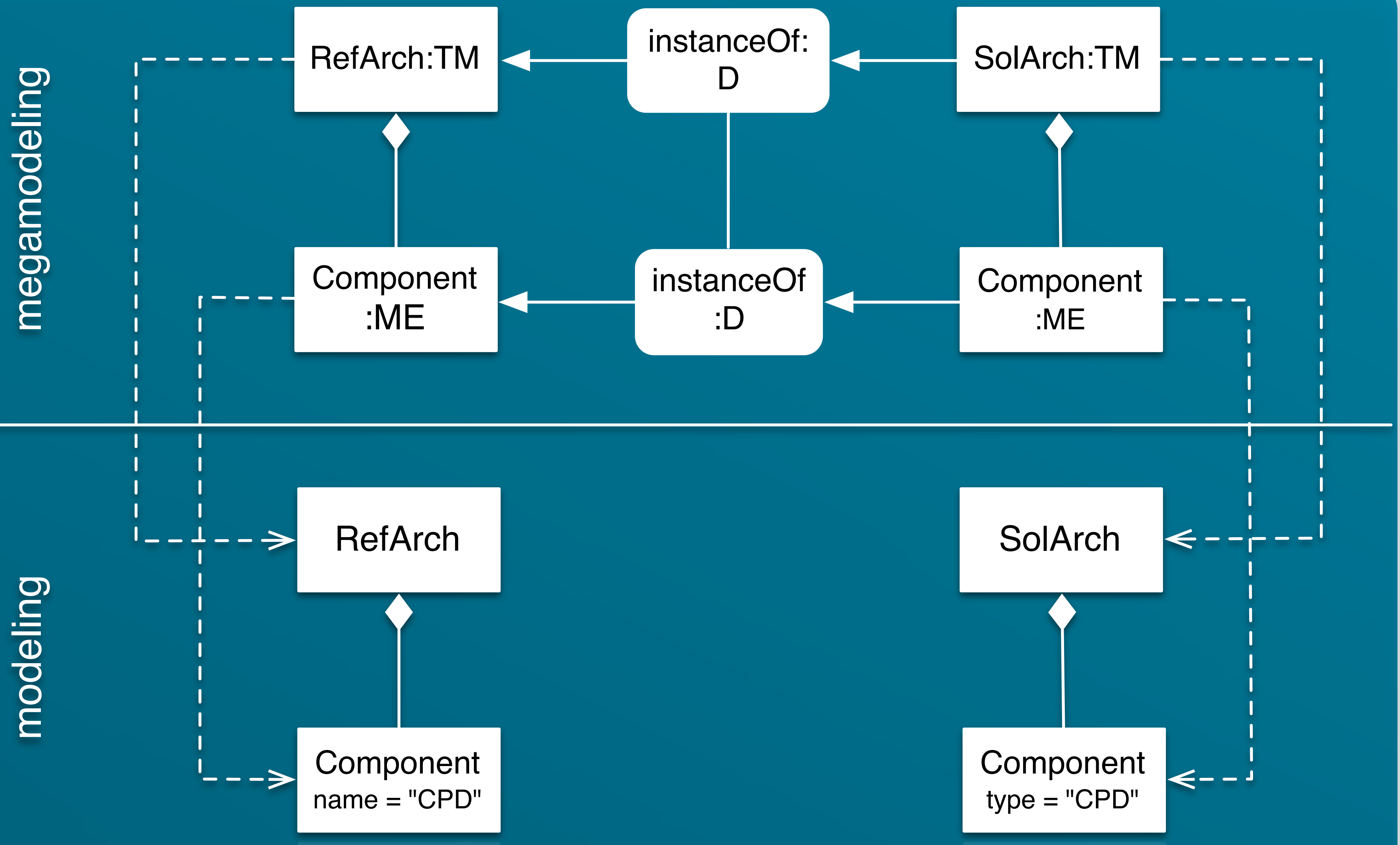
Megamodeling

- A megamodel can be considered as ...
  - … a static model
  - … a dynamic model
    - dependencies are interpreted as executable units
    - (Re-)establishing consistency
- Heterogenous dependencies



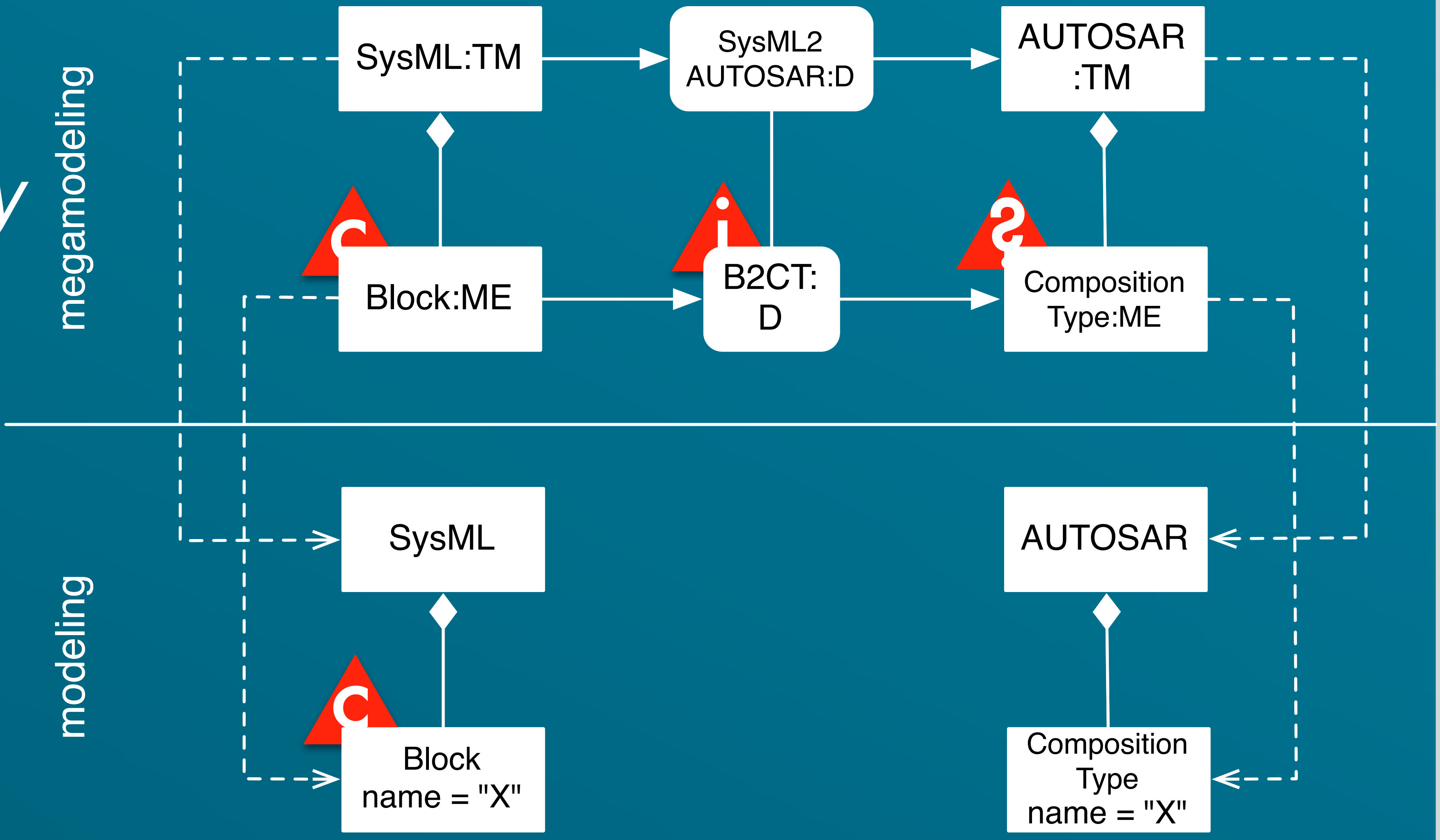
Tooling: Localization

- Localization supports identification of dependencies
- Applied on the whole megamodel



Tooling: Impact Analysis

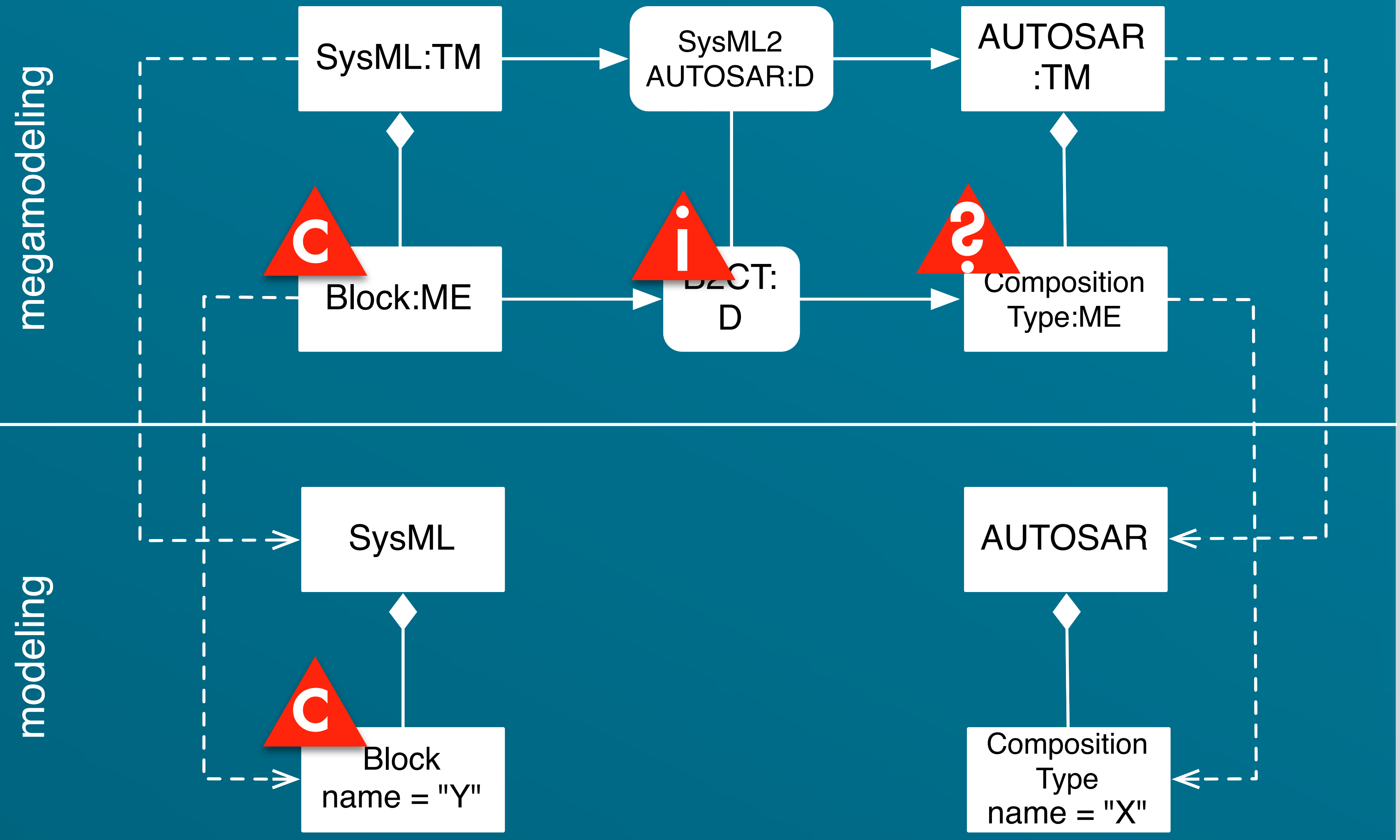
- Whenever models change, existing dependencies *may* get invalidated



- Output of invalidated dependencies are at least potentially inconsistent!

Tooling: Execution

- Potential inconstencies should be resolved
- In this example, dependency executed by a synchronization
- Execution is obtained on the whole megamodel



Future Work

Dealing with Multiple Overlapping Modeling Languages in Complex Model-Driven Development Scenarios

Future Work

Research perspective?

Not mentioned in this talk?

- Can be used as a language for horizontal and vertical composition of heterogenous model operations
- Localization and executation are capable of interpreting the megamodel for this task

What have to be done?

- Working on a complete formal definition of the approach
- Research:
  - Megamodels and versioning
  - Megamodels and runtime systems
  - Megamodels and patterns
  - ...

Future Work

Take away

- Facilitate *megamodels* to support MDD by ...
  - ... integrating *heterogenous models*
  - … explicitly capturing *heterogenous dependencies*
    - between model elements (microscopic)
    - between models (macroscopic)
    - between dependencies
  - … providing *automation* on a macroscopic/microscopic level

References

[N.Drivalos,2008] Nicholas Drivalos, Richard F. Paige, Kiran Fernandes and Dimitrios S. Kolovos. **Towards Rigorously Defined Model-to-Model Traceability**. In ECMDA-TW'08: Proceedings of 4th Workshop on Traceability, Berlin, Germany, SINTEF, 9-12 June 2008

[M.Aleksy,2008] Markus Aleksy, Tobias Hildenbrand, Claudia Obergfell and Michael Schwind. **A Pragmatic Approach to Traceability in Model-Driven Development**. In PRIMIUM, 2008.

[H.Asuncion,2008] Hazeline U. Asuncion. **Towards practical software traceability**. In ICSE Companion '08: Companion of the 30th international conference on Software engineering, Pages 1023--1026, ACM, New York, NY, USA , 2008.

[J.R.Falleri,2006] J. R. Falleri, M. Huchard and C. Nebut. **Towards a Traceability Framework for Model Transformations in Kermeta**. In Tor Neple, Jon Oldevik and Jan Aagedal editors, ECMDA-TW'06: Proceedings of 2rd Workshop on Traceability, Bilbao, Spain, SINTEF, 10-13 July 2006.

[H.LeDang,2008] Hung Le Dang, Hubert Dubois and Sébastien Gérard. **Towards a traceability model in a MARTE-based methodology for real-time embedded systems**. In Innovations in Systems and Software Engineering, Vol. 4(3):189--193, October 2008.

[B.Vanhooff,2007] Bert Vanhooff, Stefan Van Baelen, Wouter Joosen and Yolande Berbers. **Traceability as Input for Model Transformations**. In Jon Oldevik, G\oran K. Olsen and Tor Neple editors, ECMDA-TW'07: Proceedings of 3rd Workshop on Traceability, Haifa, Israel, Pages 37--46, SINTEF, 11-15 June 2007.

[N.Aizenbud-Reshef,2005] Netta Aizenbud-Reshef, Richard F. Paige, Julia Rubin, Yael Shaham-Gafni and Dimitrios S. Kolovos. **Operational Semantics for Traceability**. In ECMDA-TW'05: Proceedings of 1st Workshop on Traceability, Nurnberg, Germany, Pages 7--14, SINTEF, 7-10 November 2005.

[P.Mäder,2008] Patrick Mäder, Orlena Gotel and Ilka Philippow. **Enabling Automated Traceability Maintenance by Recognizing Development Activities Applied to Models**. In ASE '08: Proceedings of the 2008 23rd IEEE/ACM International Conference on Automated Software Engineering, Pages 49--58, IEEE Computer Society, Washington, DC, USA , 2008.

References

[S.Walderhaug,2006] Staale Walderhaug, Ulrik Johansen, Erlend Stav and Jan Aagedal. **Towards a Generic Solution for Traceability in MDD**. In Tor Neple, Jon Oldevik and Jan Aagedal editors, ECMDA-TW'06: ECMDA Traceability Workshop, Bilbao (Spain), SINTEF, 10-13 July 2006.

[M.Barbero,2007] Mikael Barbero, Marcos Didonet Fabro and Jean Bézivin. **Traceability and Provenance Issues in Global Model Management**. In Jon Oldevik, G\oran K. Olsen and Tor Neple editors, ECMDA-TW'07: Proceedings of 3rd Workshop on Traceability, Haifa, Israel, Pages 47--55, SINTEF, Haifa, Israel, June 2007.

[R.Salay,2009] Rick Salay, John Mylopoulos and Steve Easterbrook. **Using Macromodels to Manage Collections of Related Models.** In 21st International Conference, CAiSE 2009, Amsterdam, The Netherlands, Vol. 5565/2009:141-155 of Lecture Notes in Computer Science (LNCS), Springer-Verlag, 8-12 June 2009.

[M.Del Fabro,2009] Marcos Didonet Fabro, Patrick Albert, Jean Bézivin and Frédéric Jouault. **Achieving Rule Interoperability Using Chains of Model Transformations**. In ICMT '09: Proceedings of the 2nd International Conference on Theory and Practice of Model Transformations, Pages 249--259, Springer-Verlag, Berlin, Heidelberg , 2009.

[M.Fritzsche,2009] Mathias Fritzsche, Hugo Brunelière, Bert Vanhooff, Yolande Berbers, Frédéric Jouault and Wasif Gilani. **Applying Megamodelling to Model Driven Performance Engineering**. In ECBS '09: Proceedings of the 2009 16th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems, Pages 244--253, IEEE Computer Society, Washington, DC, USA , 2009.

[J.M.Favre,2005] Jean-Marie Favre. **Megamodelling and Etymology**. In James R. Cordy, Ralf Lämmel and Andreas Winter editors, Transformation Techniques in Software Engineering, Vol. 05161 of Dagstuhl Seminar Proceedings, Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2005.

[F.Allilaire,2006] Freddy Allilaire, Jean Bézivin, Hugo Brunelière and Frédéric Jouault. **Global Model Management In Eclipse GMT/AM3**. In Proceedings of the Eclipse Technology eXchange workshop (eTX) at ECOOP'06, 2006.

References

[F.Jouault,2010] Frédéric Jouault, Bert Vanhooff, Hugo Brunelière, Guillaume Doux, Yolande Berbers and Jean Bézivin. **Inter-DSL coordination support by combining megamodeling and model weaving**. In SAC '10: Proceedings of the 2010 ACM Symposium on Applied Computing, Pages 2011--2018, ACM, New York, NY, USA , 2010.

[P.A.Muller,2009] Pierre-Alain Muller, Frédéric Fondement and Benoît Baudry. **Modeling Modeling**. In 12th International Conference on Model Driven Engineering Languages and Systems (MoDELS), Denver, CO, USA, Vol. 5795:2-16 of LNCS, Springer, 2009.

[D.Kolovos,2008] Dimitrios S. Kolovos, Richard Paige and Fiona Polack. **A Framework for Composing Modular and Interoperable Model Management Tasks**. In MDTPI workshop, EC-MDA, Berlin, Germany, June 2008.