

# Entwicklung von Modellierungswerkzeugen mit MOFLON



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT

## Developing Modeling Tools with MOFLON

**Informatik 2009 – Workshop  
„Methodische Entwicklung von Modellierungswerkzeugen“  
Keynote Talk**



ES Real-Time Systems Lab

Prof. Dr. rer. nat. Andy Schürr

Dept. of Electrical Engineering and Information Technology

Dept. of Computer Science (adjunct Professor)

[www.es.tu-darmstadt.de](http://www.es.tu-darmstadt.de)

**Felix Klar**

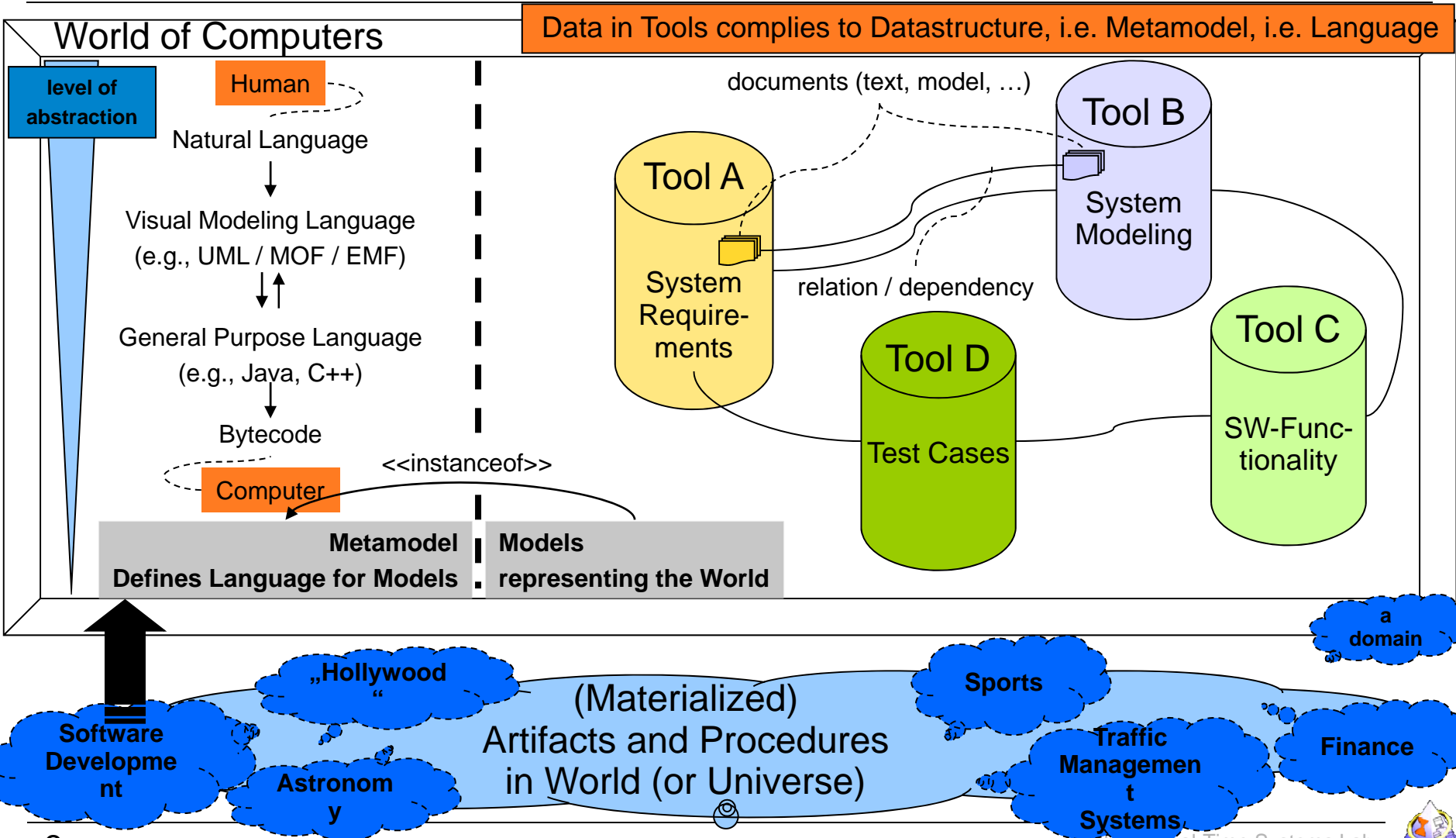
[Felix.Klar@es.tu-darmstadt.de](mailto:Felix.Klar@es.tu-darmstadt.de)

01.10.2009

# Outline

- Metamodels are Languages – A Motivation
- Metamodeling – Examples
- MOFLON – Scenarios
- Demo (Tool Integration Scenario – TiE-DEA)
- MOFLON – Architecture
- Related Work
- Current and Future Activities at Real-Time Systems Lab

# Metamodeling – Overview and Motivation



# Metamodeling – Goals

## Constraints

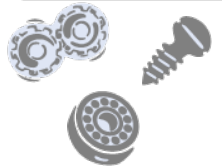
- Constraints for detailed definition of language
- Definition of erroneous states
- Rules to comply with special design guidelines



## Metamodel

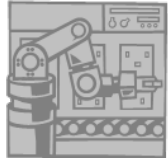
- (Meta-)Modeling of language constructs
- Definition of language structure
- Domain specific semantics

- Transformationen to repair erroneous models
- Conversion of incompatible models into design compliant models
- Automatic adaptation to design guidelines



## Abstract Syntax

## Transformation



## Model

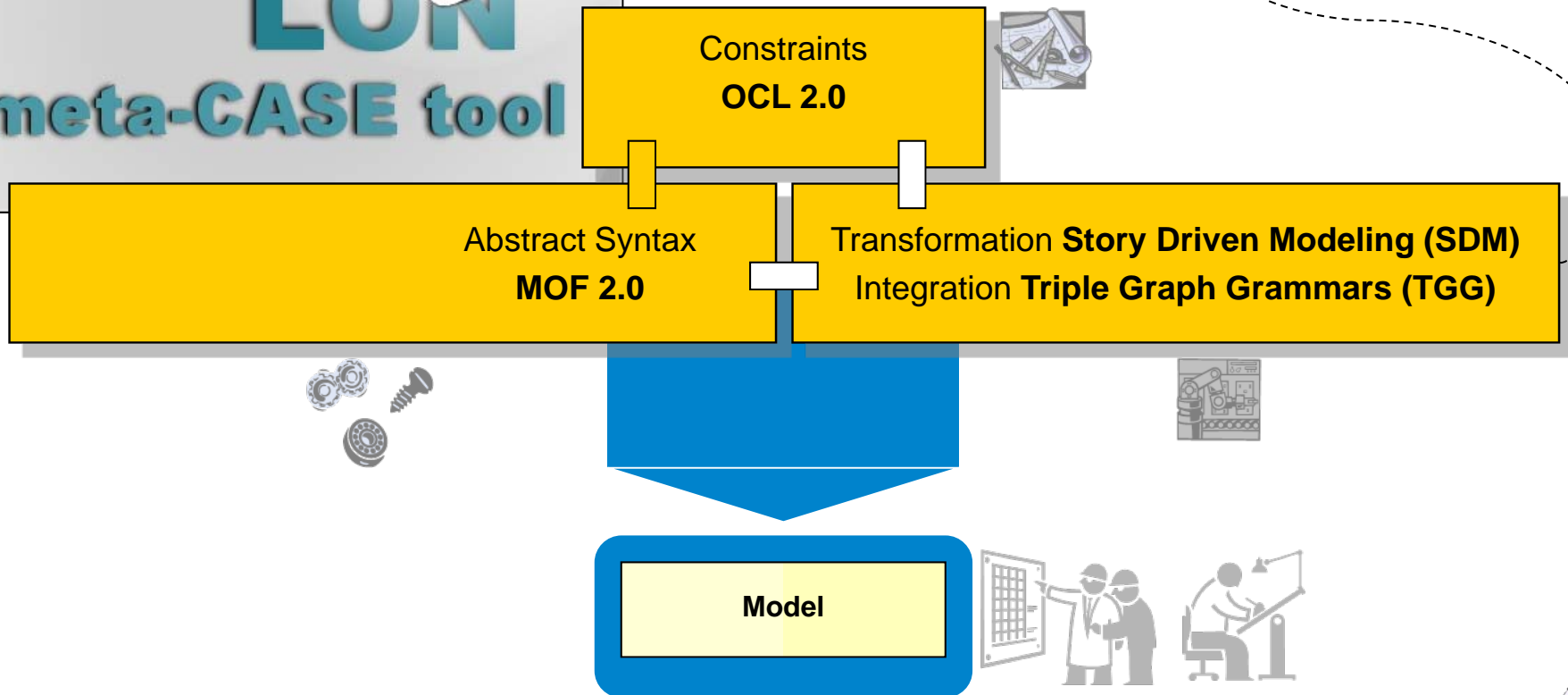


# A Solution

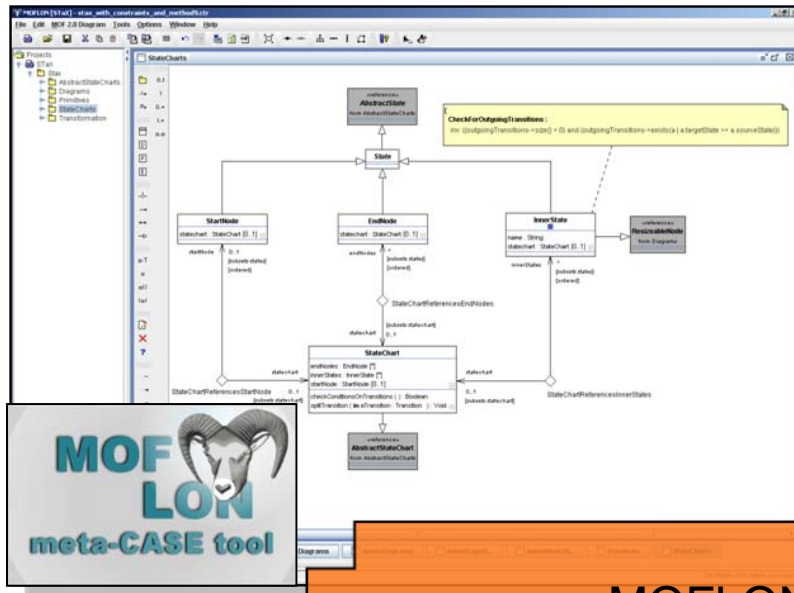
**MOFLON**  
meta-CASE tool



SDM = Pattern-based Transformation Language  
TGG = Bi-directional Transformation Language with TraceLinks

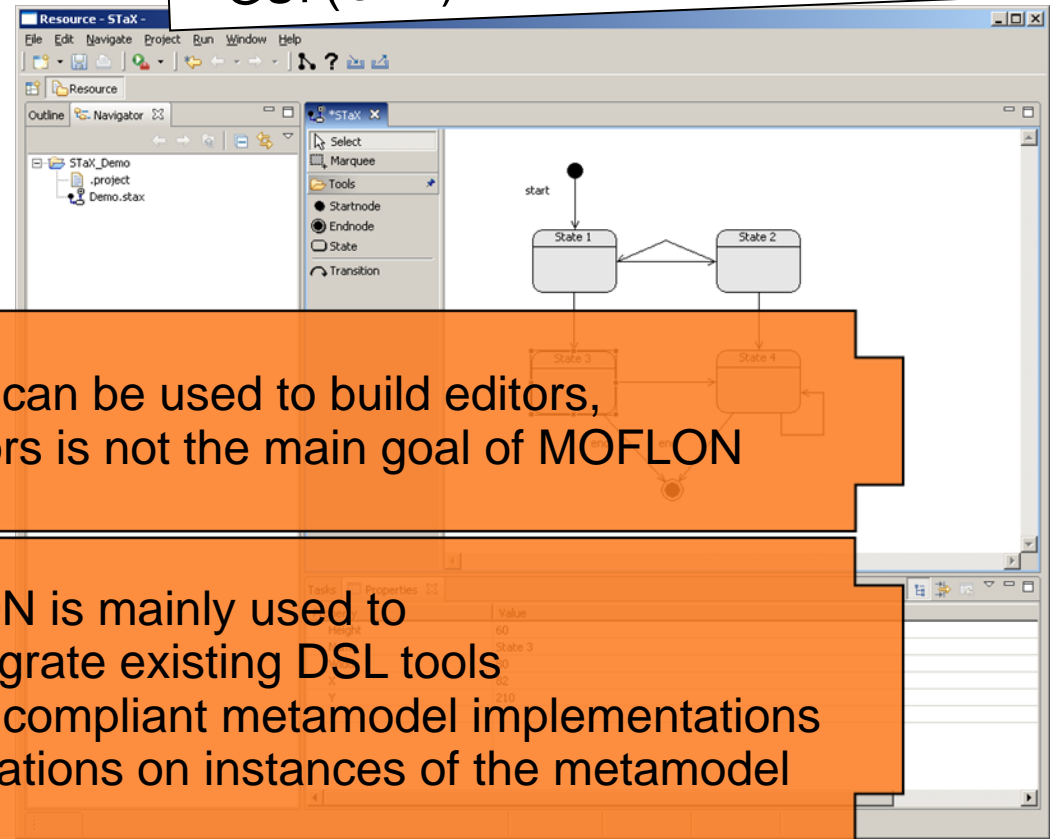


# Case Study – Statechart Editor (STaX)



Editor:

- data structure (MOFLON repository)
- GUI (GEF)



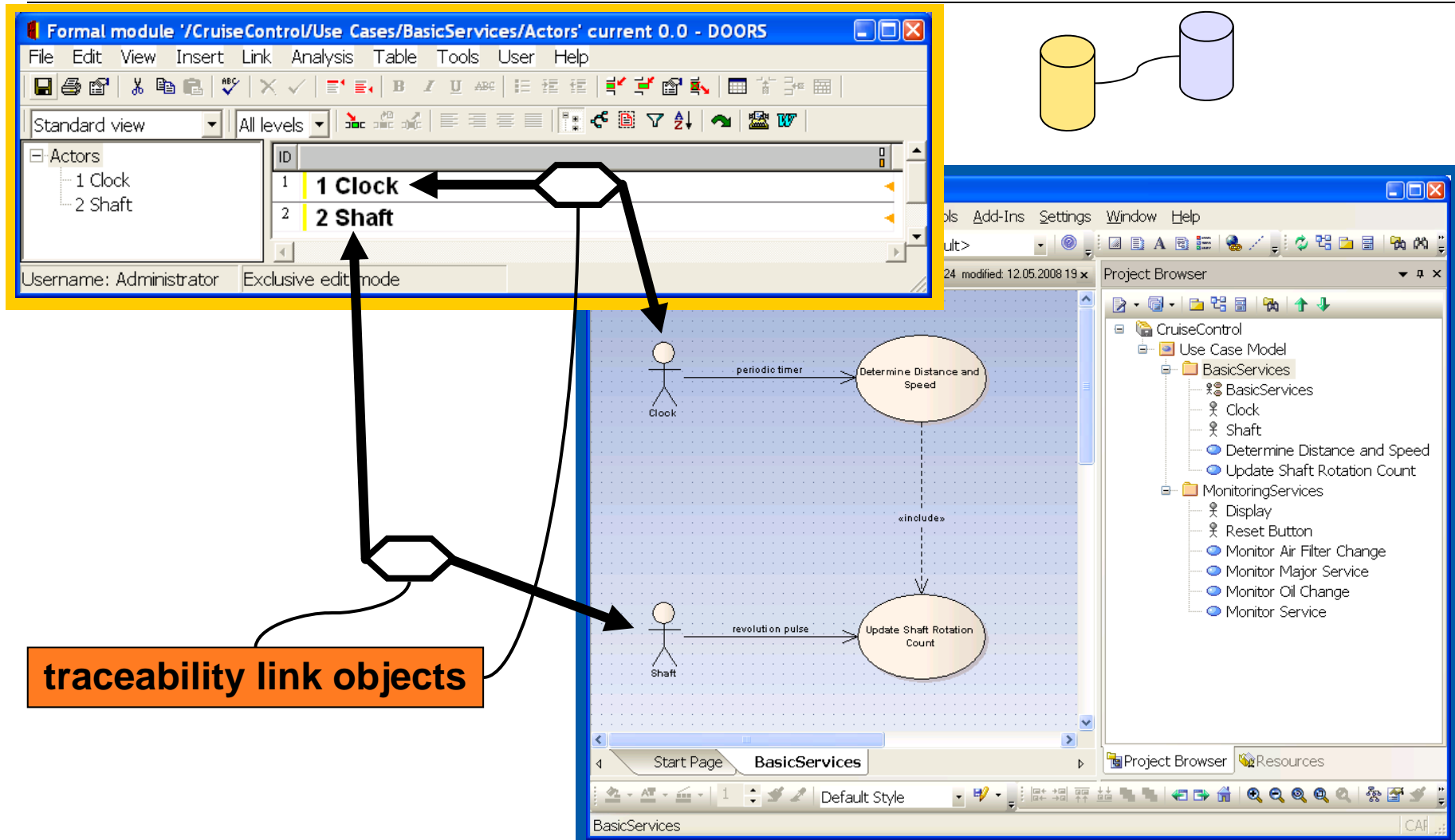
MOFLON can be used to build editors,  
but building editors is not the main goal of MOFLON

MOFLON is mainly used to

- integrate existing DSL tools
- generate standard compliant metamodel implementations
- specify transformations on instances of the metamodel



# Tool Integration Example – DOORS / EA



# Outline

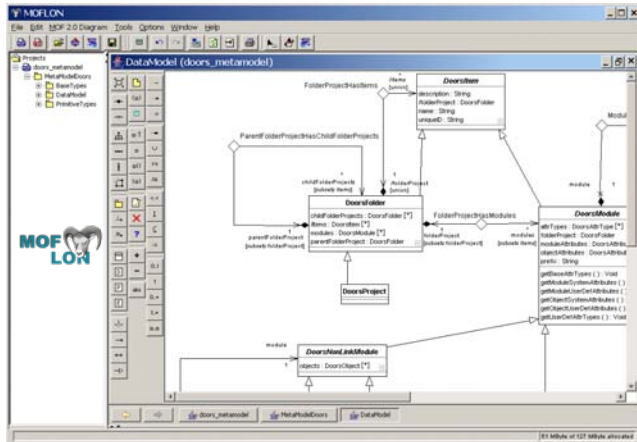
- Metamodels are Languages – A Motivation
- Metamodeling – Examples
- MOFLON – Scenarios
- **Demo (Tool Integration Scenario – TiE-DEA)**
- MOFLON – Architecture
- Related Work
- Current and Future Activities at Real-Time Systems Lab





# Demo – Tool Integration Scenario (DOORS - EA)

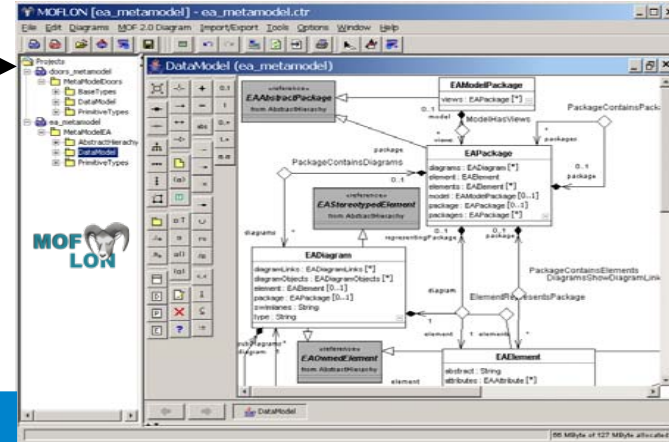
DOORS Metamodel



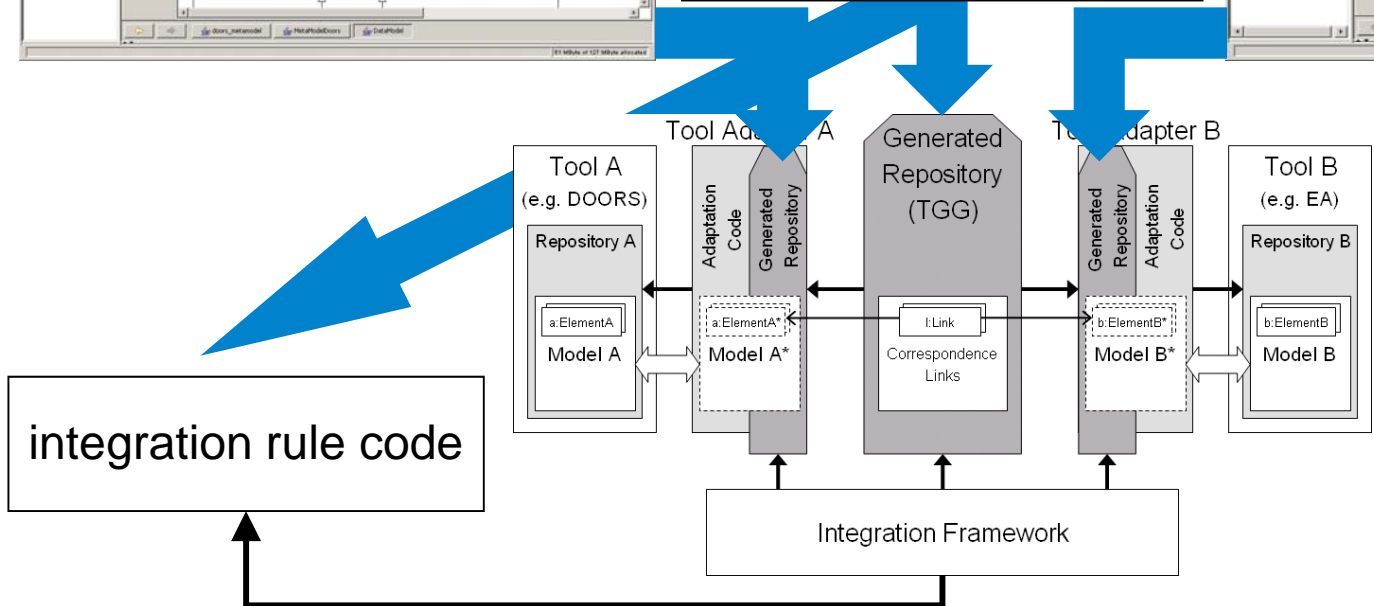
TGGs relate



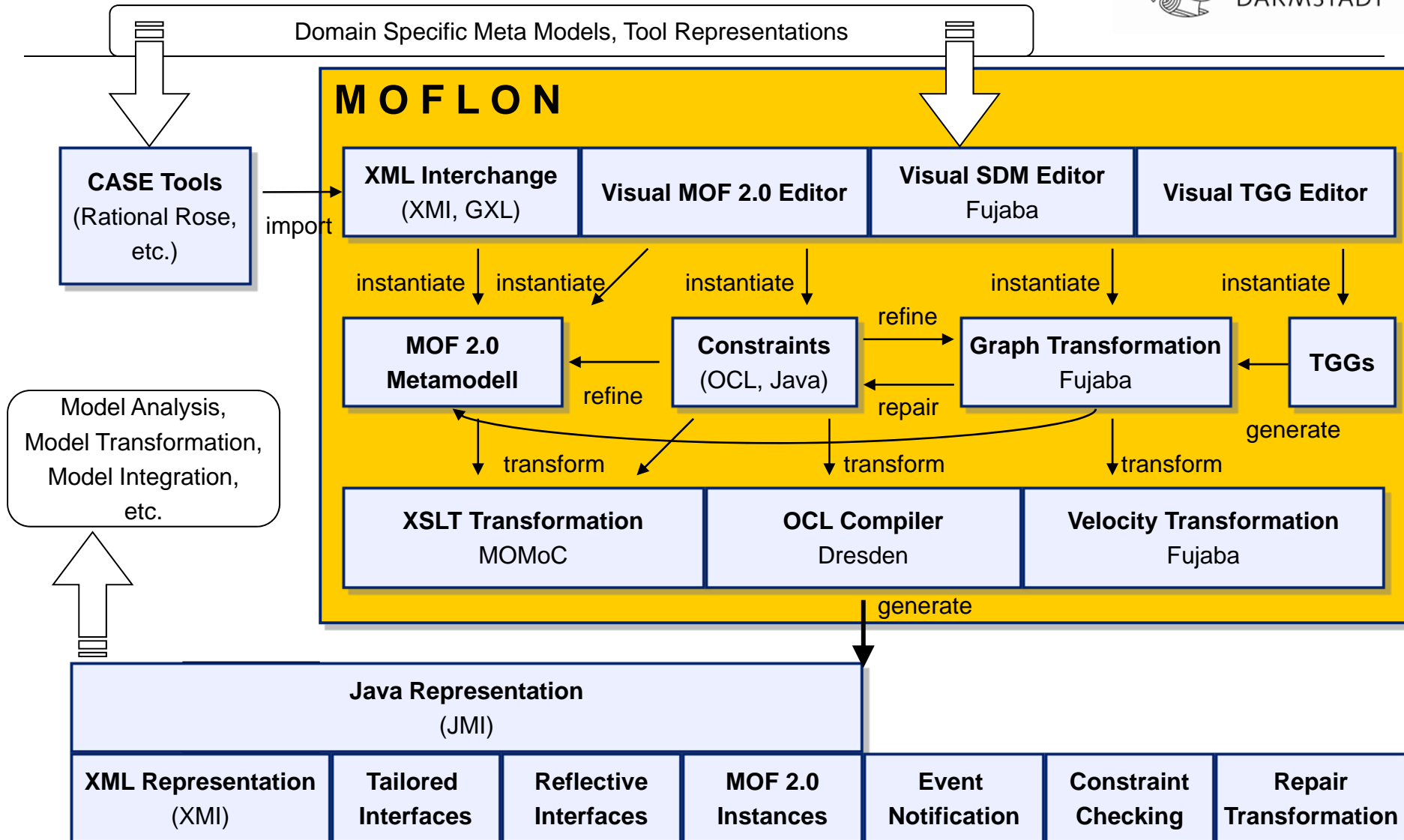
EA Metamodel



MOFLON generates



# MOFLON – Architecture

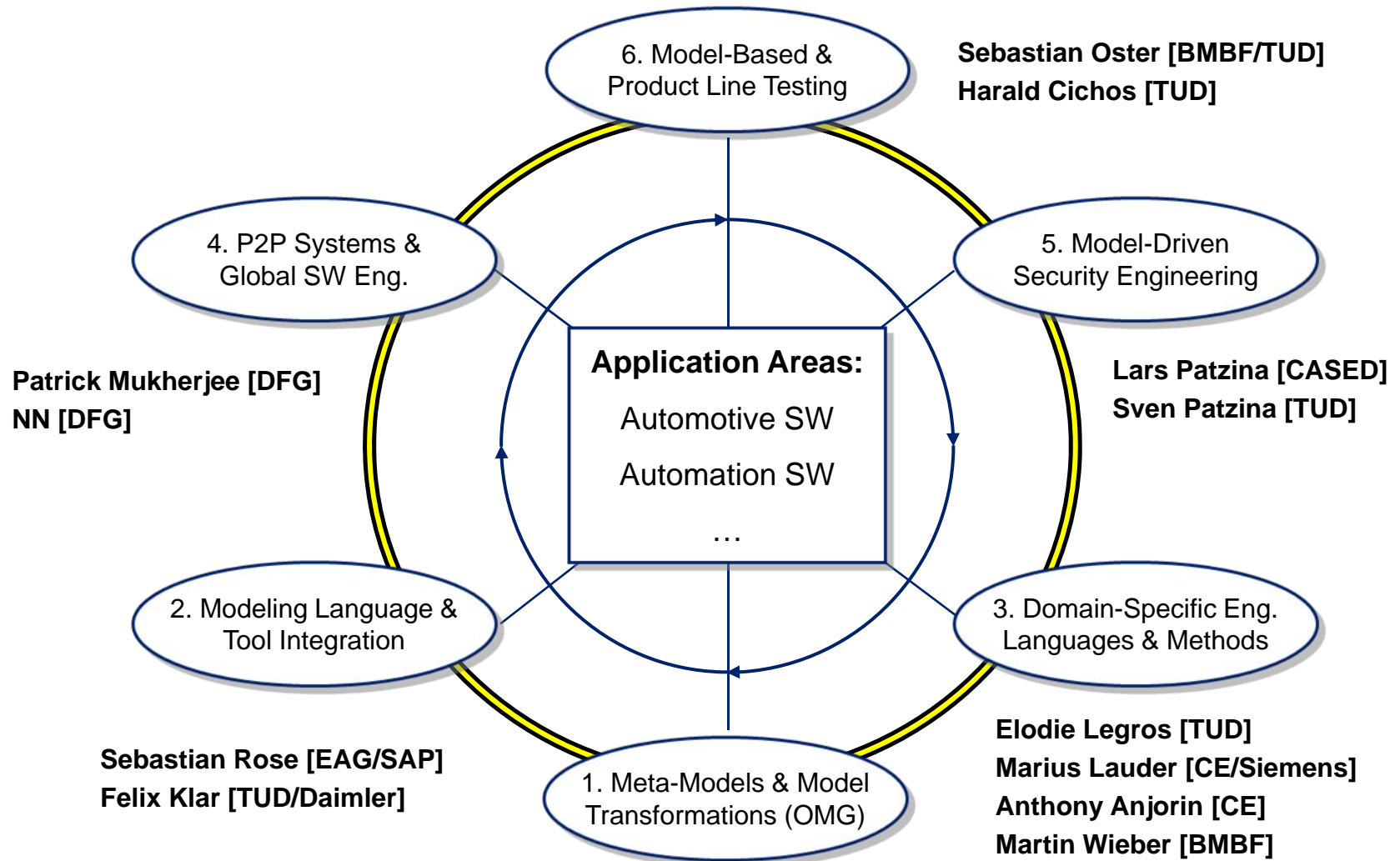


# Related Approaches

standards	approaches based on graph-/modeltransformation					classic meta-CASE approaches				text based approaches					
	MOF, OCL, QVT	Fujaba & MOFLON	Progres & TGG	GME & TGG	EMF & GREAT	EMF & Tefkat	AToM <sup>3</sup>	Microsoft DSL MetaEdit+	EMF & DSL	Pounamu	EBNF & TXL	DiaGen	SQL	XML	
Abstract syntax	+	+	+	+	o	o	o	+	+	o	+	+	+	o	+
Concrete syntax	--	--	--	+	+	--	+	+	+	+	+	+	--	--	--
Static semantics	+	+	o	+	+	+	o	o	--	+	o	+	o	o	--
Dynamic semantics	+	+	+	+	+	+	+	o	o	--	--	--	+	--	o
Model analysis	+	+	+	+	+	o	+	o	--	+	--	o	+	o	+
Model transformation	+	+	+	+	+	+	+	o	--	--	--	o	+	o	+
Model integration	+	+	+	+	o	+	--	--	--	--	--	--	o	--	o
Acceptability	+	+	o	--	o	+	--	+	--	o	+	o	o	+	+
Scaleability	+	+	--	o	--	o	--	o	--	--	--	--	--	--	o
Tool availability	--	o	o	+	+	+	+	+	o	o	+	+	+	+	o
Expressiveness	+	+	o	+	+	o	o	o	o	o	o	o	+	o	o

from Amelunxen, Königs, Rötschke, and Schürr,  
**„MOSL: Composing a Visual Language for a Metamodeling Framework“**  
 in IEEE Symposium on Visual Languages and Human-Centric Computing (VLHCC 2006),  
 September, 2006, 81-84

# Model-Driven Software Development at Real-Time Systems Lab



# Further reading

- **A. Königs, A. Schürr: "Tool Integration with Triple Graph Grammars - A Survey", in: R. Heckel (ed.), Proceedings of the SegraVis School on Foundations of Visual Modelling Techniques, Amsterdam: Elsevier Science Publ., 2006; Electronic Notes in Theoretical Computer Science, Vol. 148, 113-150.**
- **F. Klar, S. Rose, A. Schürr: "TiE - A Tool Integration Environment", Proceedings of the 5th ECMDA Traceability Workshop, 2009; CTIT Workshop Proceedings, Vol. WP09-09, 39-48**
- **F. Klar, S. Rose, A. Schürr: "A Meta-Model-Driven Tool Integration Development Process", Proceedings of the 2nd International United Information Systems Conference, 2008; Lecture Notes in Business Information Processing, 201-212.**
- **C. Amelunxen, A. Königs, T. Rötschke, A. Schürr: "MOFLON: A Standard-Compliant Metamodeling Framework with Graph Transformations", in: A. Rensink, J. Warmer (eds.), Model Driven Architecture - Foundations and Applications: Second European Conference, Heidelberg: Springer Verlag, 2006; Lecture Notes in Computer Science (LNCS), Vol. 4066, Springer Verlag, 361-375.**
- **A. Königs: "Model Integration and Transformation - A Triple Graph Grammar-based QVT Implementation", Technische Universität Darmstadt, Phd Thesis, 2009.**

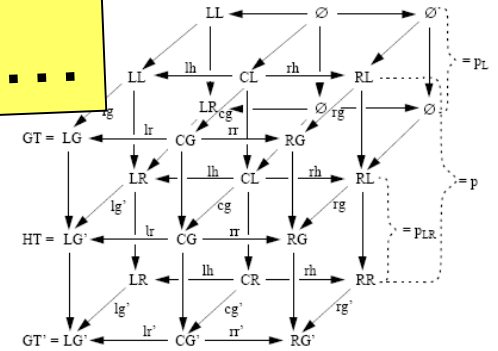


# Time for questions and discussion

Thank you for your attention...



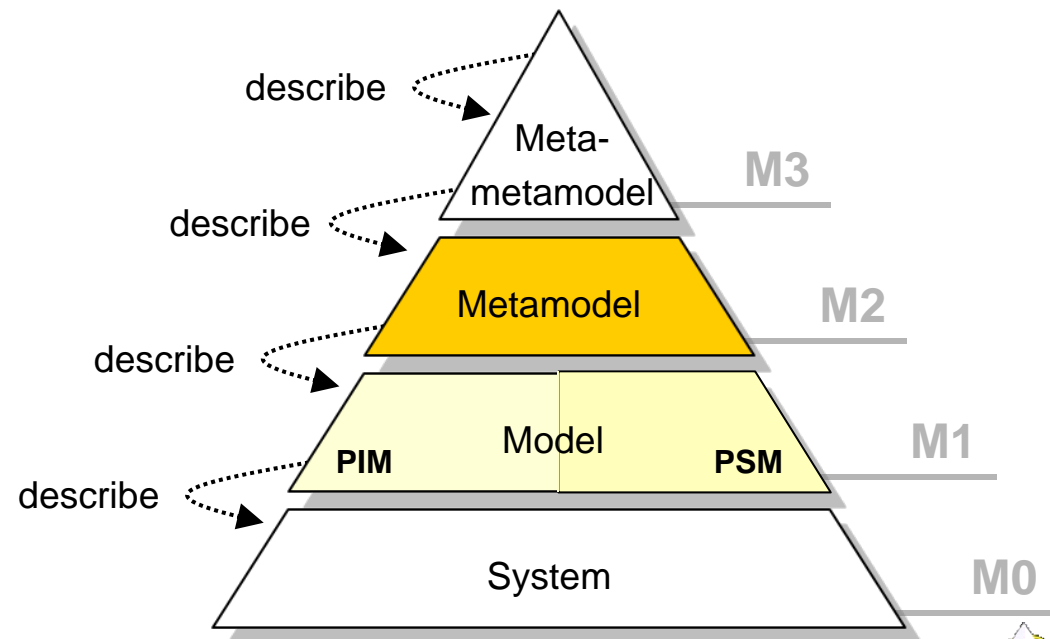
<http://www.moflon.org>



# Backup Slides

# Motivation

- Models are widely used in engineering disciplines
- Need for tool support that enables model-editing
- Domain experts want domain specific languages (DSL)  
→ domain specific models
- do not build model editors from scratch each time  
→ reuse functionality  
→ use meta-information



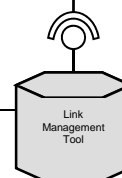
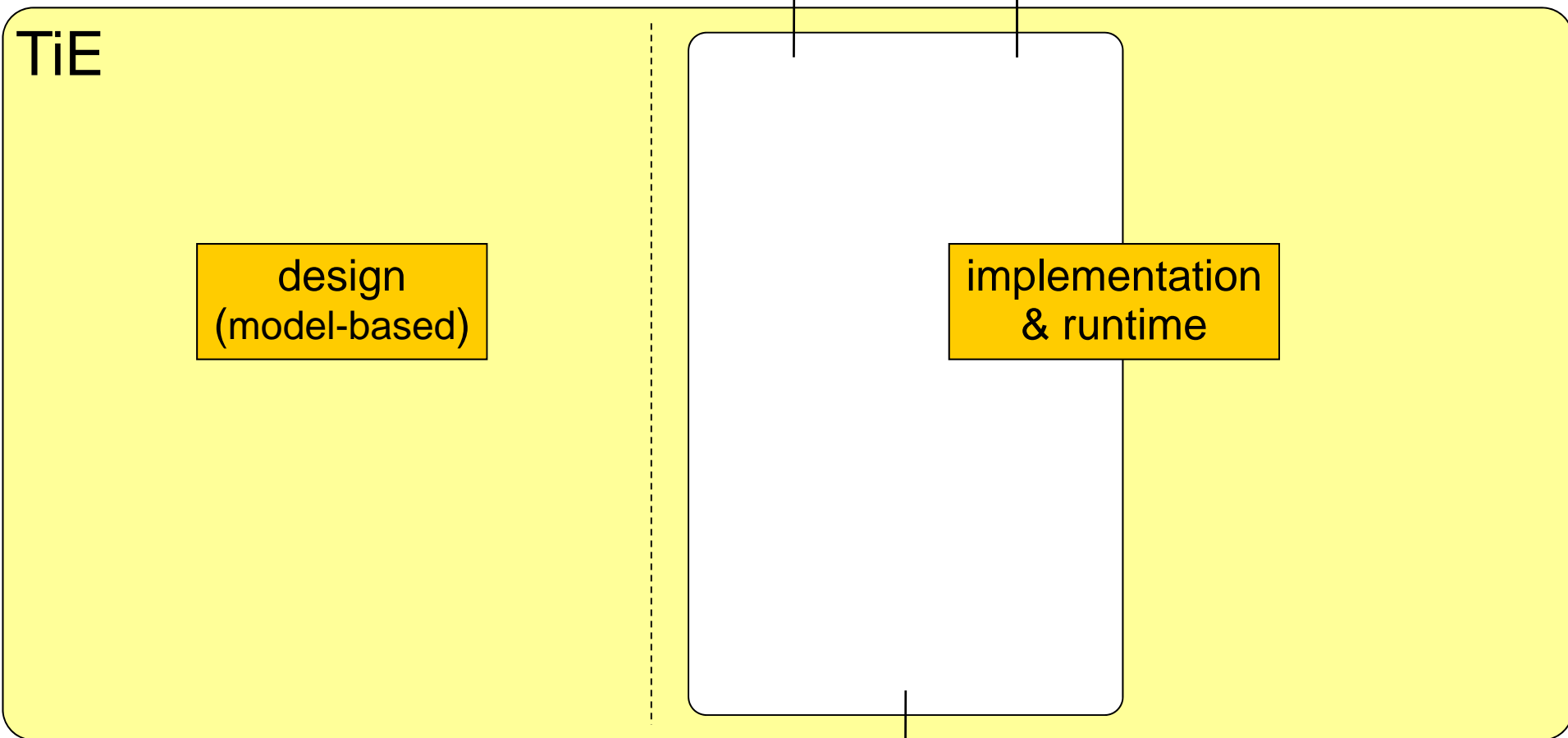
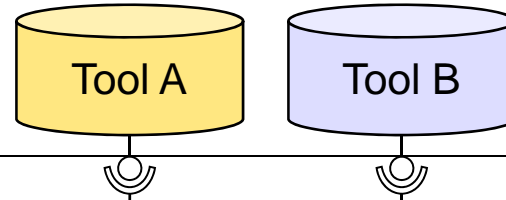


# MOFLON – Main Features

- MOF2.0 editor (draw metamodels that comply to MOF2.0 standard)  
→ build Domain Specific Languages (DSLs)
- based on the CASE-tool framework Fujaba
- possibility to extend MOFLON by own plugins
- interoperability (import / export)
- transform metamodel instances with model transformations (SDM, TGG)
- generate code (JMI-compliant) from DSLs
- instantiate models of the DSL (= repositories)
- basic editing support for generated repositories
- Standard compliance!

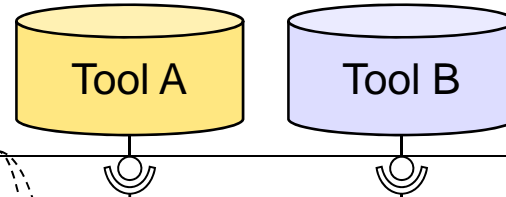


# TiE – Architecture

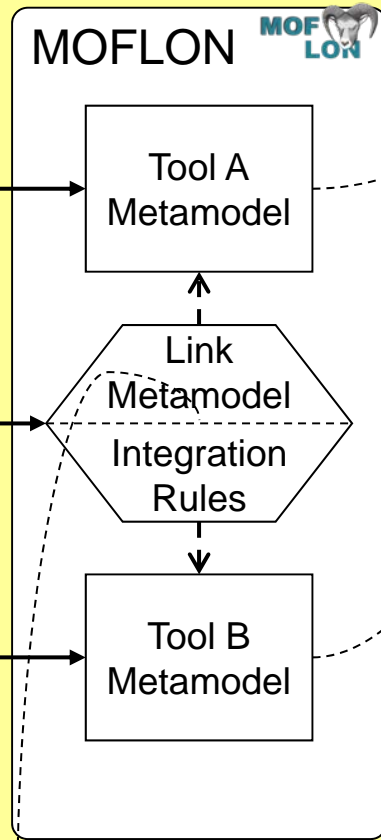


# TiE – Architecture

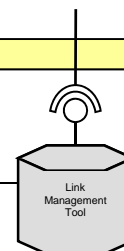
tool's data structure as metamodel



## TiE

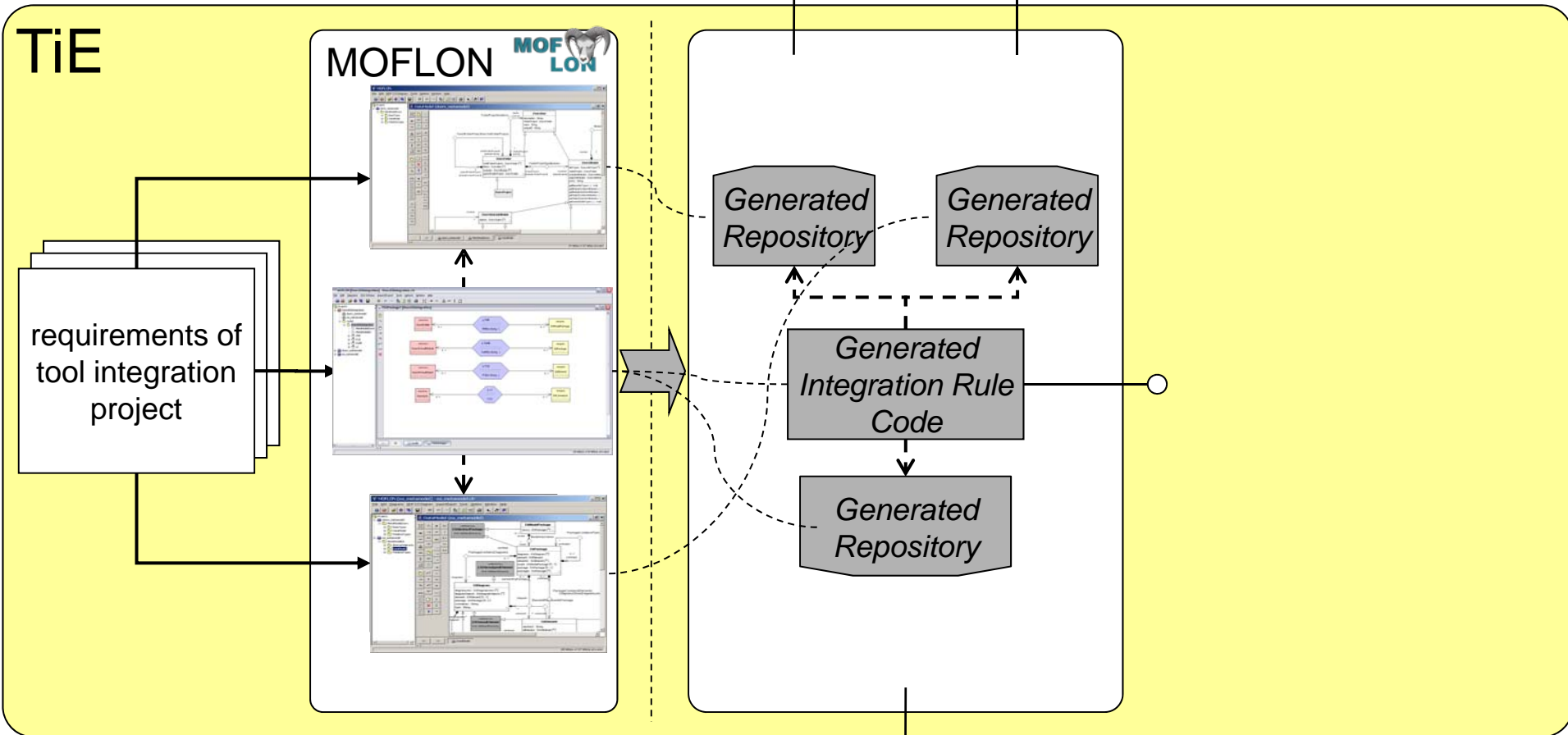
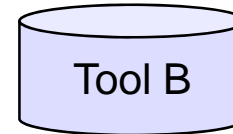


Triple Graph Grammars relate

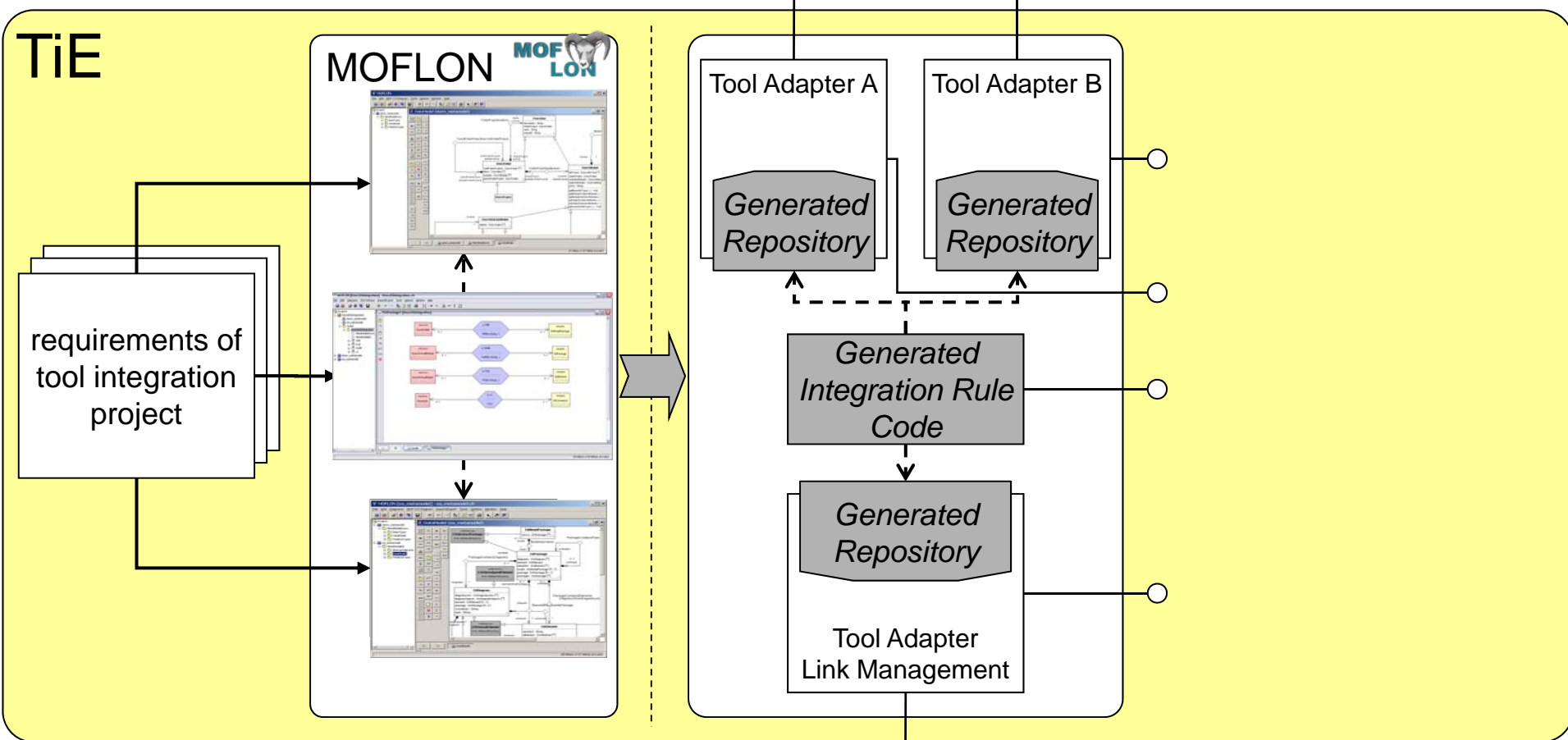
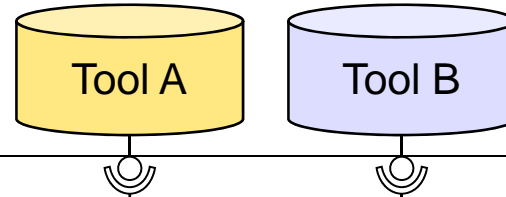


# TiE – Architecture

MOFLON generates



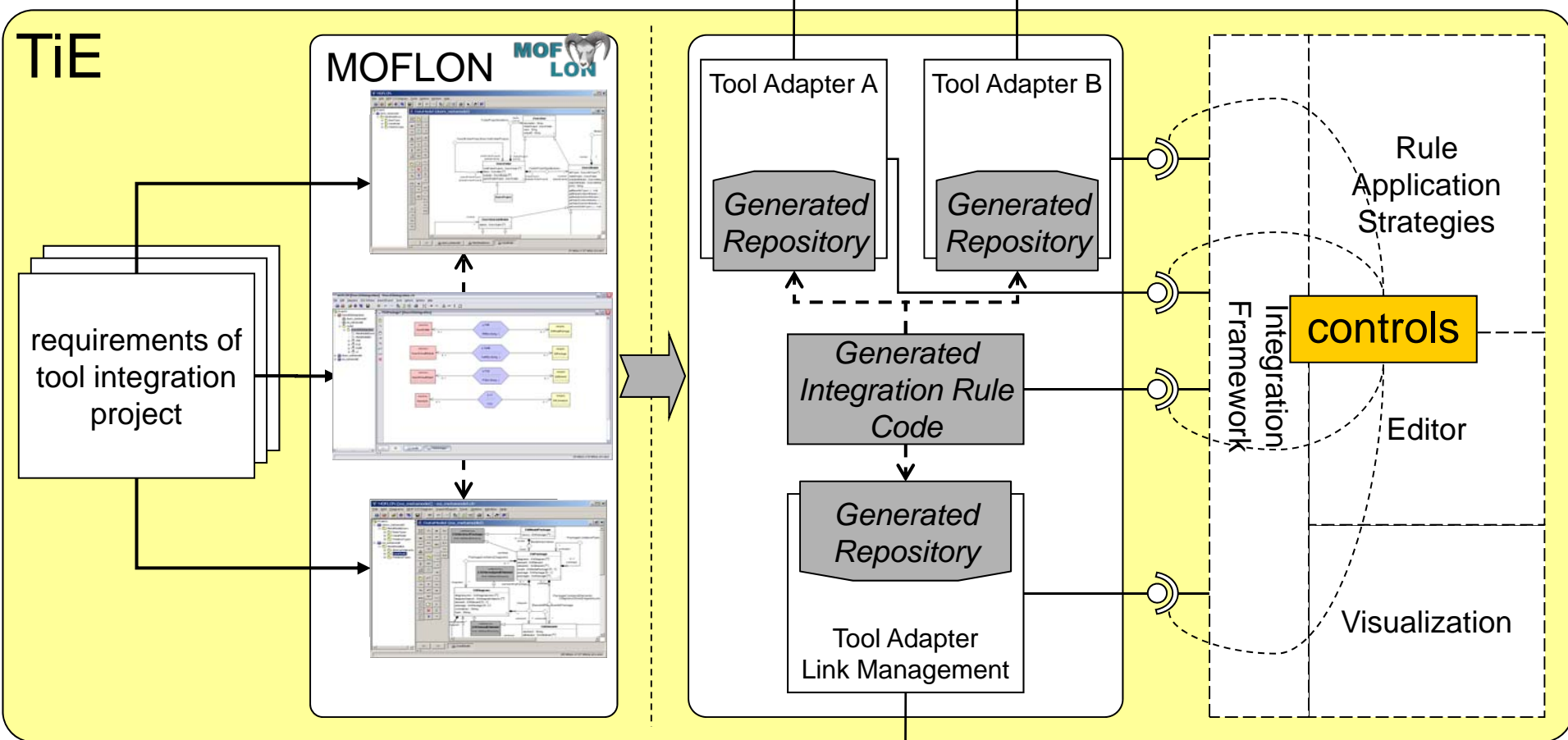
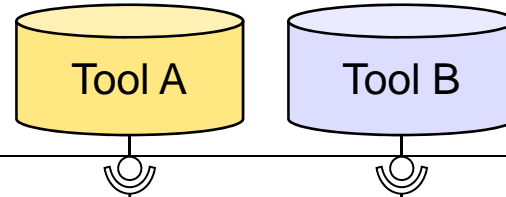
# TiE – Architecture



implement tool adapters (handwritten code)

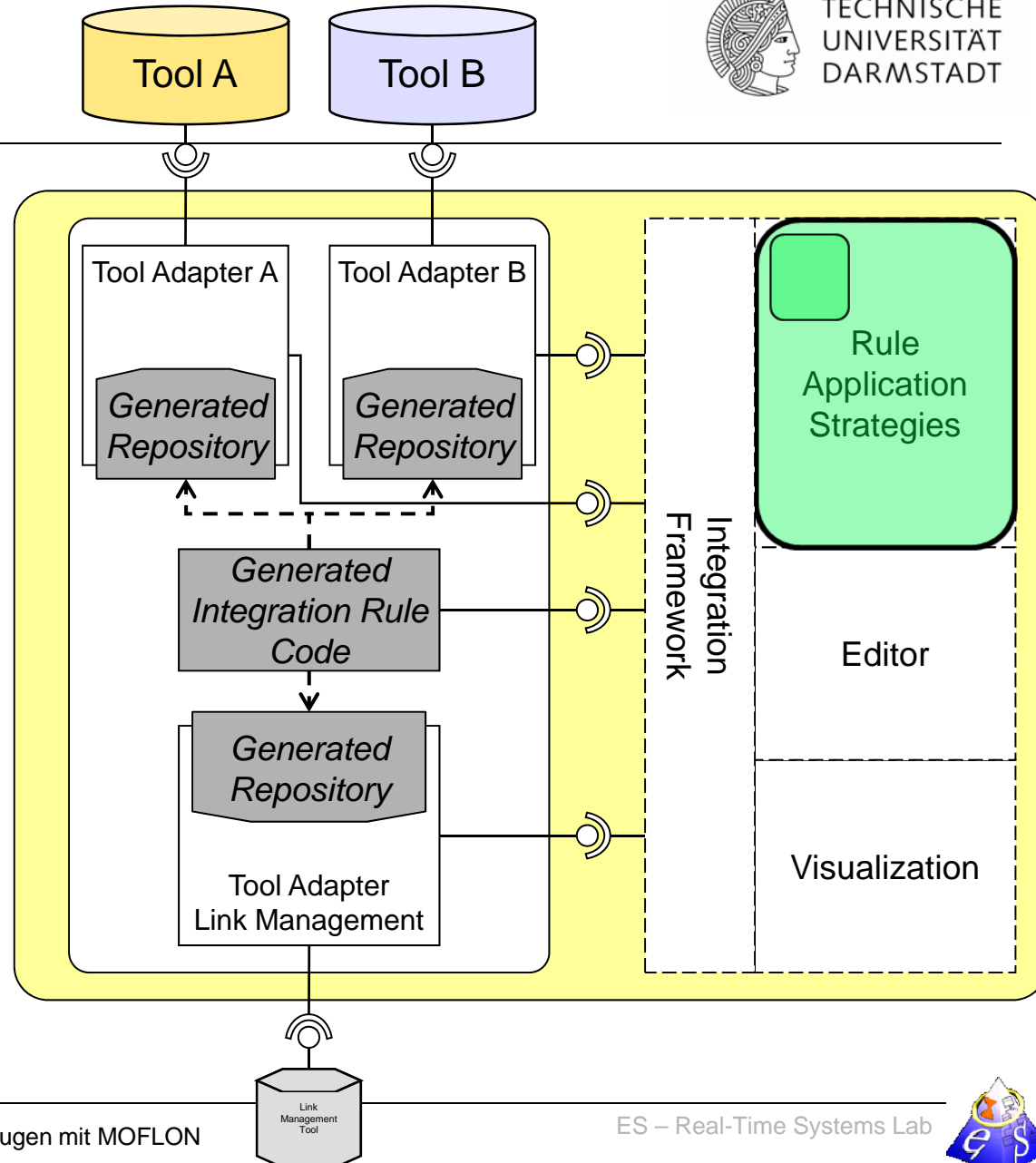
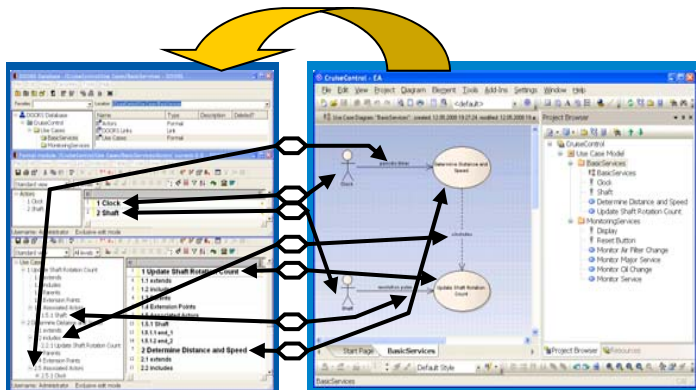


# TiE – Architecture

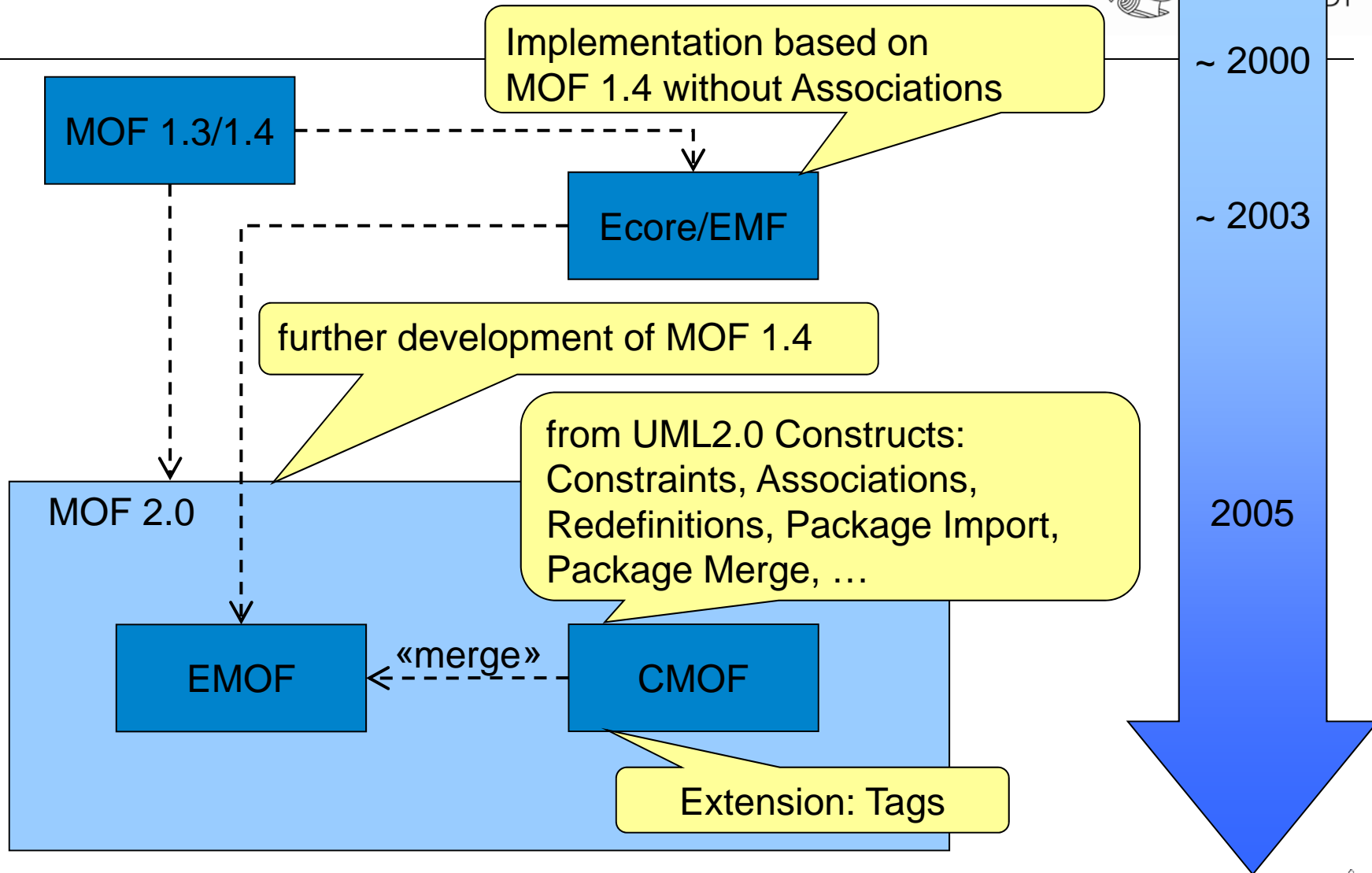


# TiE – Runtime

- get element from tool
- invoke integration rule
  - find match
  - transform
  - establish link
- process next element



# Meta-Metamodels – A History





# Concepts in MOF 1.4, EMOF, and CMOF

	EMOF 2.0	MOF 1.4	CMOF 2.0
Primitive Type, Enumeration	✓	✓	✓
Class, Property, Operation	✓	✓	✓
- Classifier, Feature	✗	✓	✓
Packages	✓	✓	✓
- Namespaces	✗	✓	✓
- Inherit/Cluster/Merge	✗	✓	✓
Reference	✓	✓	✓
Association	✗	✓	✓
- Subsets, Redefinitions	✗	✗	✓

# Code Generation

	<b>EMOF 2.0/ Ecore</b>	<b>MOF 1.4</b>	<b>CMOF 2.0</b>
Implementation	EMF	MOIN	MOFLON
Interface Standard	<i>none</i>	JMI	JMI
Serialization	XMI	XMI	XMI
Metaobject Access	✓	✓	✓
Refl. Access on Properties	✓	✓	✓
Refl. Call of Methods	<b>x</b> *	✓	✓
All Instances of Classifier	<b>x</b>	✓	✓
delete() on Links/Elements	<b>x</b> *	✓	✓

\* static-Method in EcoreEvaluationEnvironment/EcoreUtil