Ludwig
Maximilians
Universität
München
LMU

What is a Multi-Modelling Language?

Martin Wirsing
in cooperation with
Artur Boronat, Alexander Knapp, Jose Meseguer

Urbana-Champaign, IFIP WG 1.3, 2008

Motivation

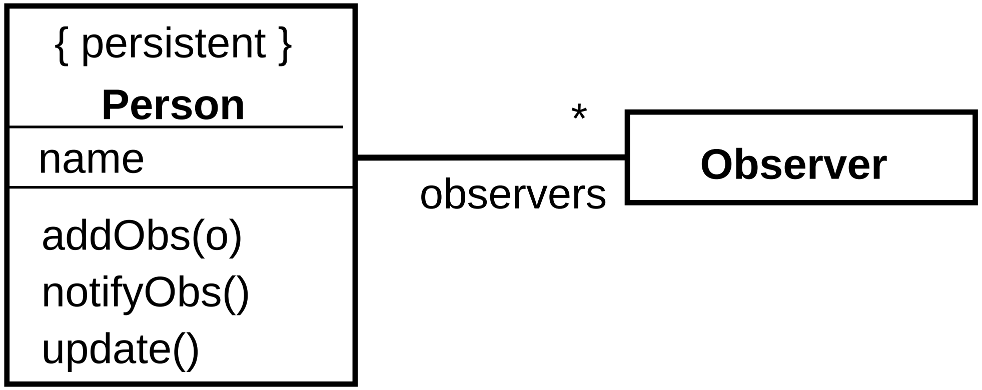
- In practice, several modelling languages are used at the same time.
  - UML, XML, BPMN, ER, SQL, …

- **Example scenarios**
  - Modelling with UML for an existing SQL-database schema
  - Deriving a UML-class diagram from an SQL-database
  - Business-process modelling with BPMN and design with BPEL

- What is called for: **A “multi-modelling” language**
  - **Consistency** of heterogeneous models
  - **Transformations** between different modelling languages
  - **Correctness** of transformations
  - **Systematic combination** of modelling languages

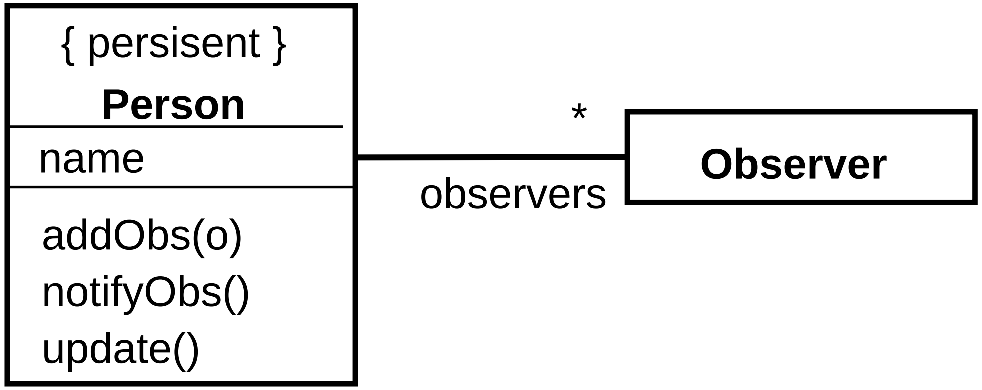
Example: UML+DB Design

(Part of) UML GUI design



Example: UML+DB Design

(Part of) UML GUI design



{ persistent }

Person

name

addObs(o)
notifyObs()
update()

*

observers

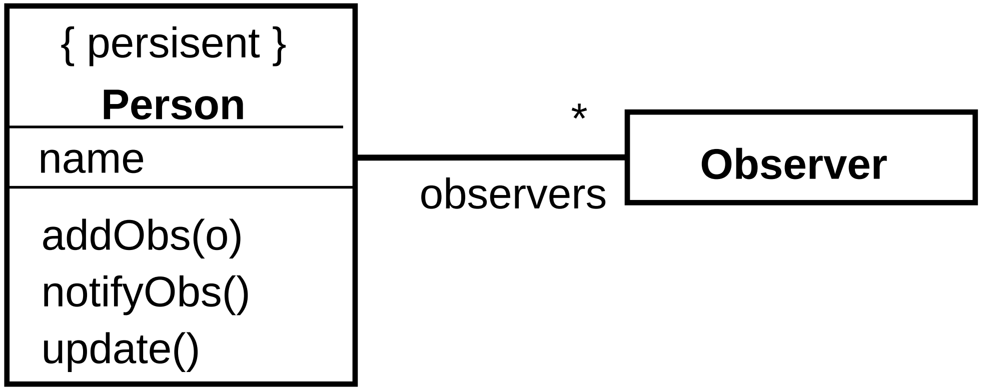
Observer

Model transformation

Person	pkey	name

Example: UML+DB Design

(Part of) UML GUI design


{ persisent }
Person
name
addObs(o)
notifyObs()
update()
*
observers
Observer

Model transformation

Person	pkey	name

⊆

Company	ckey	name

Person	pkey	name

ComPers	ckey	pkey

(Part of) database schema

Related Work

- **”System model approach”**
  - Broy: Stream processing functions 1981 & UML system model, 2006-2007
  - Gurevich: ASM, 1988 & Börger et al.: ASM semantics for UML ,2000-2004
  - Meseguer: Rewriting Logic, 1992 & W, Knapp: OO SW Enginerg, 1996
- **”Model-driven architecture approach”**
  - Poernomo: Type theory for MOF, 2007
  - **Boronat, Meseguer: Algebraic Semantics for MOF, 2008**
  - SENSORIA: Model transformations for services, 2005-2008
- **”Heterogeneous semantics and development approach”**
  - Goguen, Burstall: Institutions, 1984
  - W, Knapp: View Consistency, 2004
  - Mossakowski: Heterogeneous Institutions, Habilitation Thesis, 2005
  - **Cengarle, Knapp, Tarlecki, W: Heterogeneous UML Semantics**, 2008

➢ **Natural (logic) semantics for multi-modelling languages**

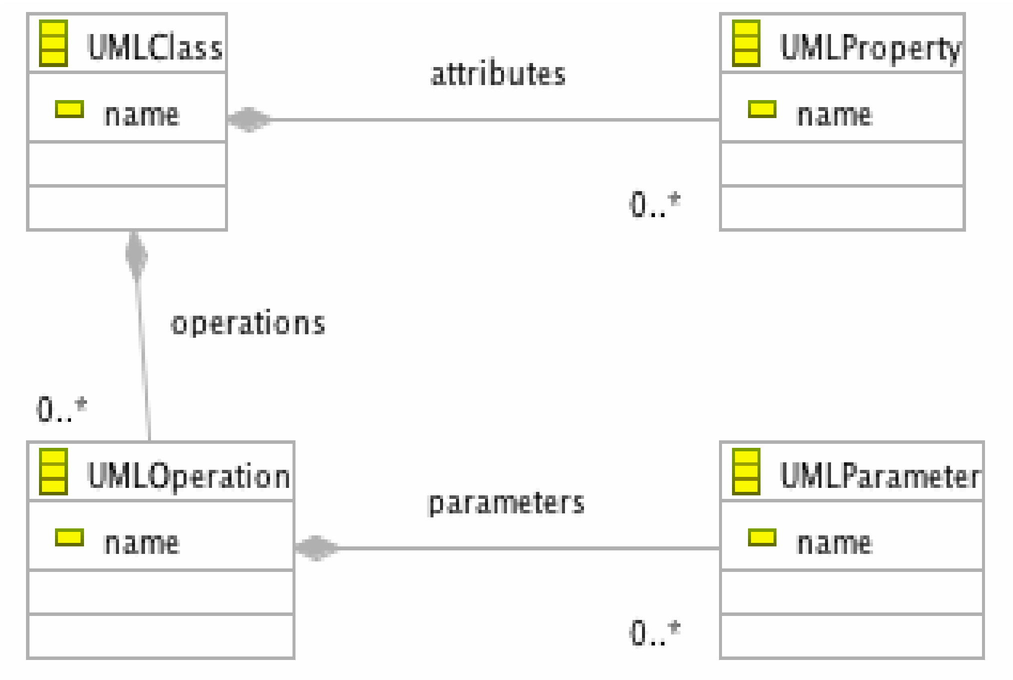
Approach

- **Combine model-driven architecture and heterogeneous semantics**
  - **Metamodels** for precise language descriptions
    - 1) Executable **metamodel specification** in Maude
  - **Heterogeneous semantics for heterogeneous models**
    - 3) **Semantic map** from metamodel spec to appropriate institution
    - ➢ Heterogeneous institutions as semantics for families of metamodels
  - **Semantics-preserving model transformations**
    - 2) **Model transformation** based on metamodel specification
    - 4) **Semantic connection** via Institution morphisms and comorphisms

➢ new, **semantically well-founded notion of a multi-modelling language**

➢ new notion of **semantic correctness for model transformations**

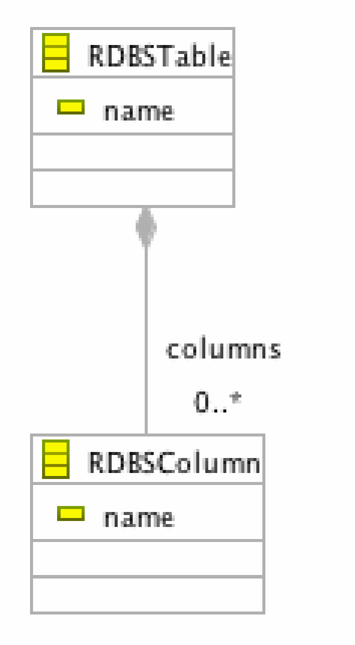
1) The UML Metamodel Specified in Maude



```
mod UML is
  including MOMENT2-GT .

  sorts UMLClass   UMLProperty
    UMLOperation  UMLParameter .
    . . .

    op name:`_ :
      String -> Property .

    op attributes:`_ :
      List{Oid} -> Property .
    . . .
endm
```

MOMENT2-GT [Boronat] obtains the Maude specification

1) The Relational Metamodel Specified in Maude

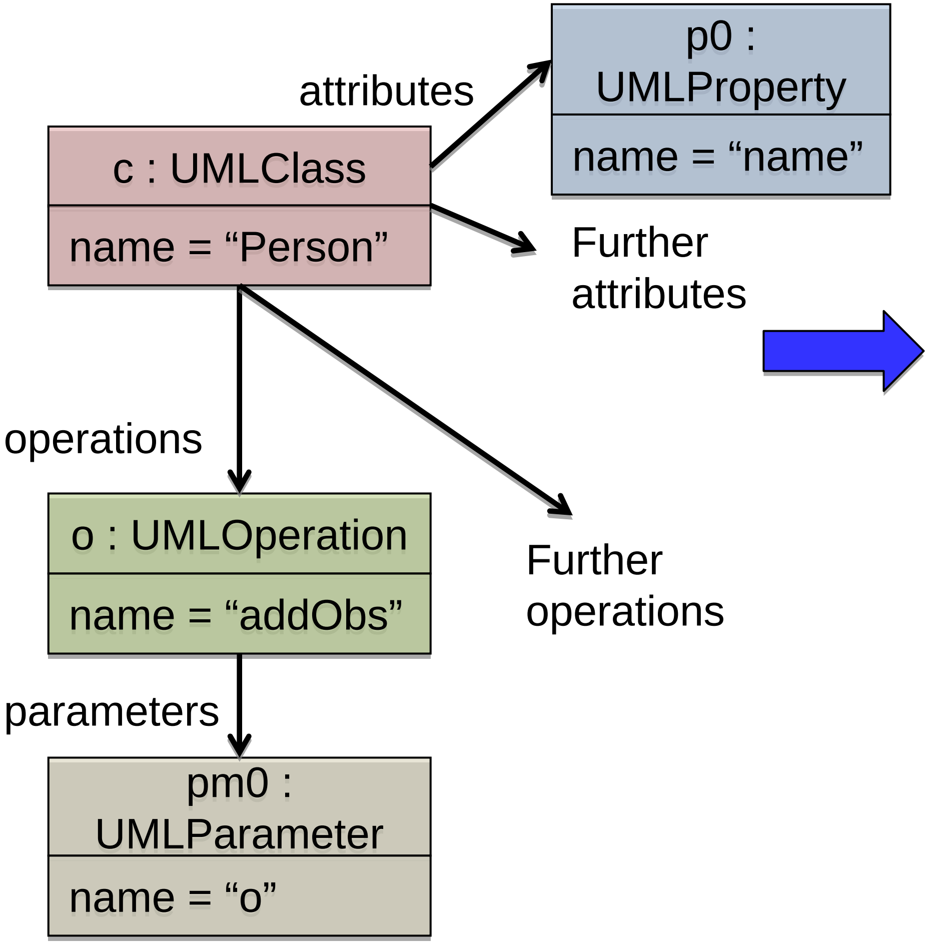


MOMENT2-GT obtains the Maude specification

```
mod RDBS is
    including MOMENT2-GT .

    sorts
      RDBSTable   RDBSColumn .
    ...
    op name:`_ :
      String -> Property .
    op columns:`_ :
      List{Oid} -> Property .
endm
```

1) Data Representations of UML Models

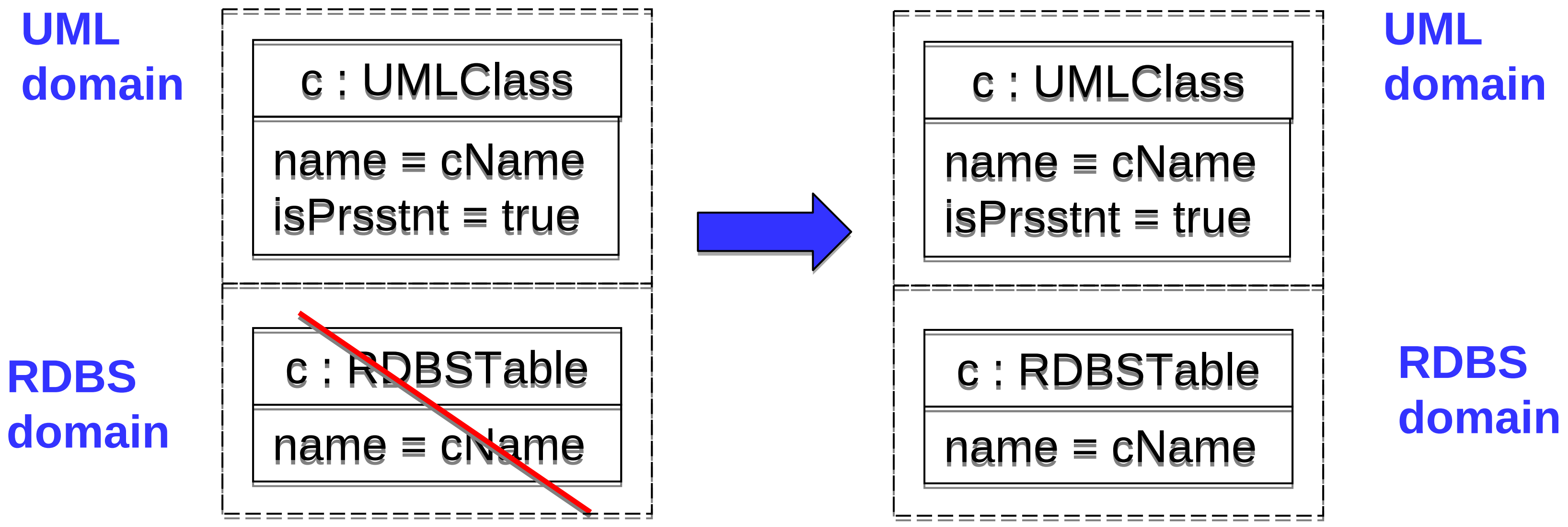
Data representation of source UML model (object diagram)



Isomorphic term-based data representation computed by MOMENT2-GT

```
<<
  < 'c : UMLClass |
       name = “Person",
       attributes = ’p0,
       operations = ’o >

< ’p0 : UMLProperty |
       name = “name" >

< ’o : UMLOperation |
      name = “addObs",
     parameters = ’pm0 >

< ’pm0 : UMLParameter |
     name = “o" >
. . .
>>
```

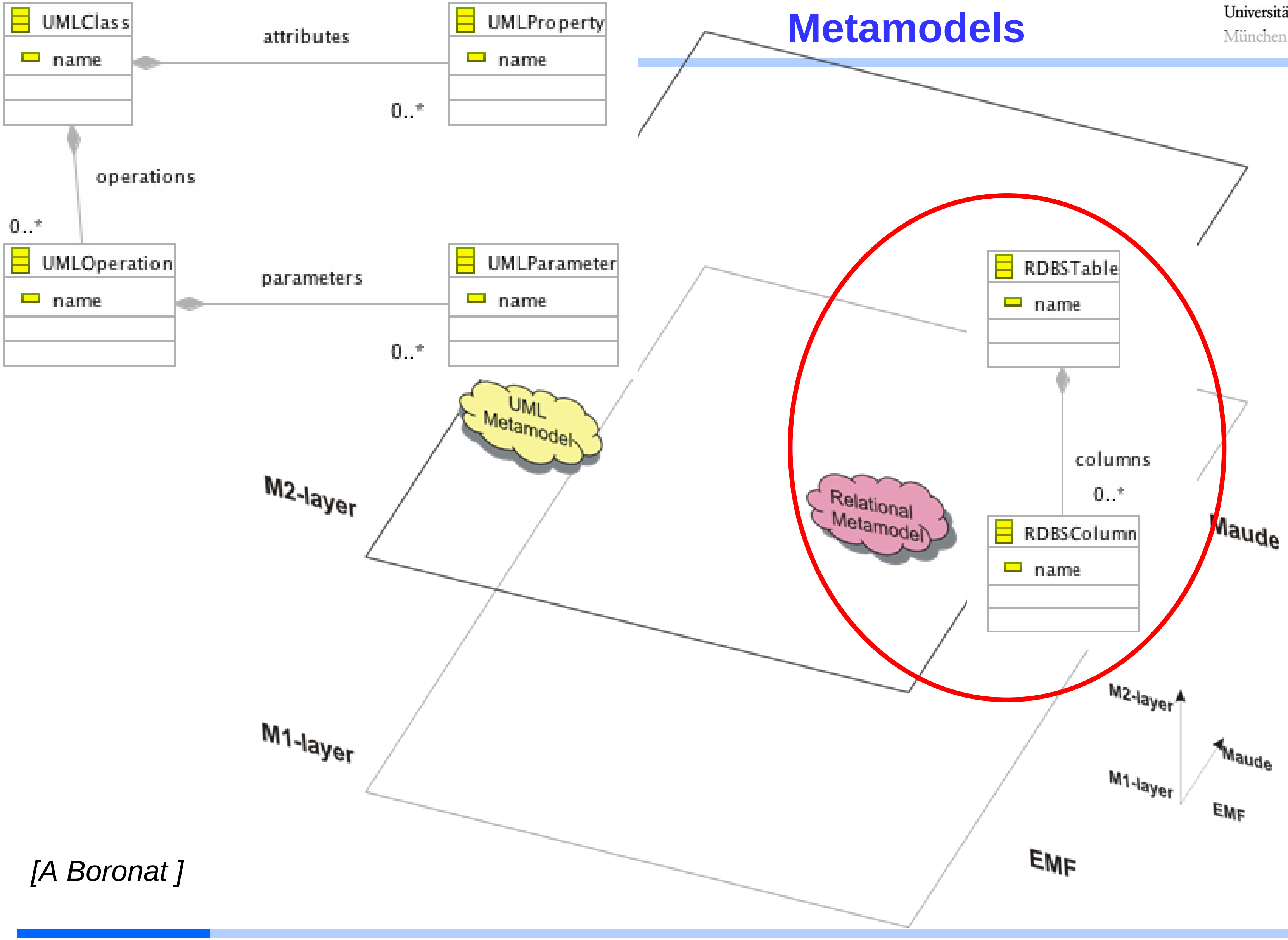
2) Model transformation
Production rule: Class-->Table



```
eq 'Class-->Table {
    lhs 'uml {
        'c : 'UMLClass {
            'name = 'cName:String
            ‘isPrsstnt = true}};
    lhs 'rdbs ;
    nac 'rdbs 'noTable {
        't : 'RDBSTable {
            'name = 'cName:String}};
```

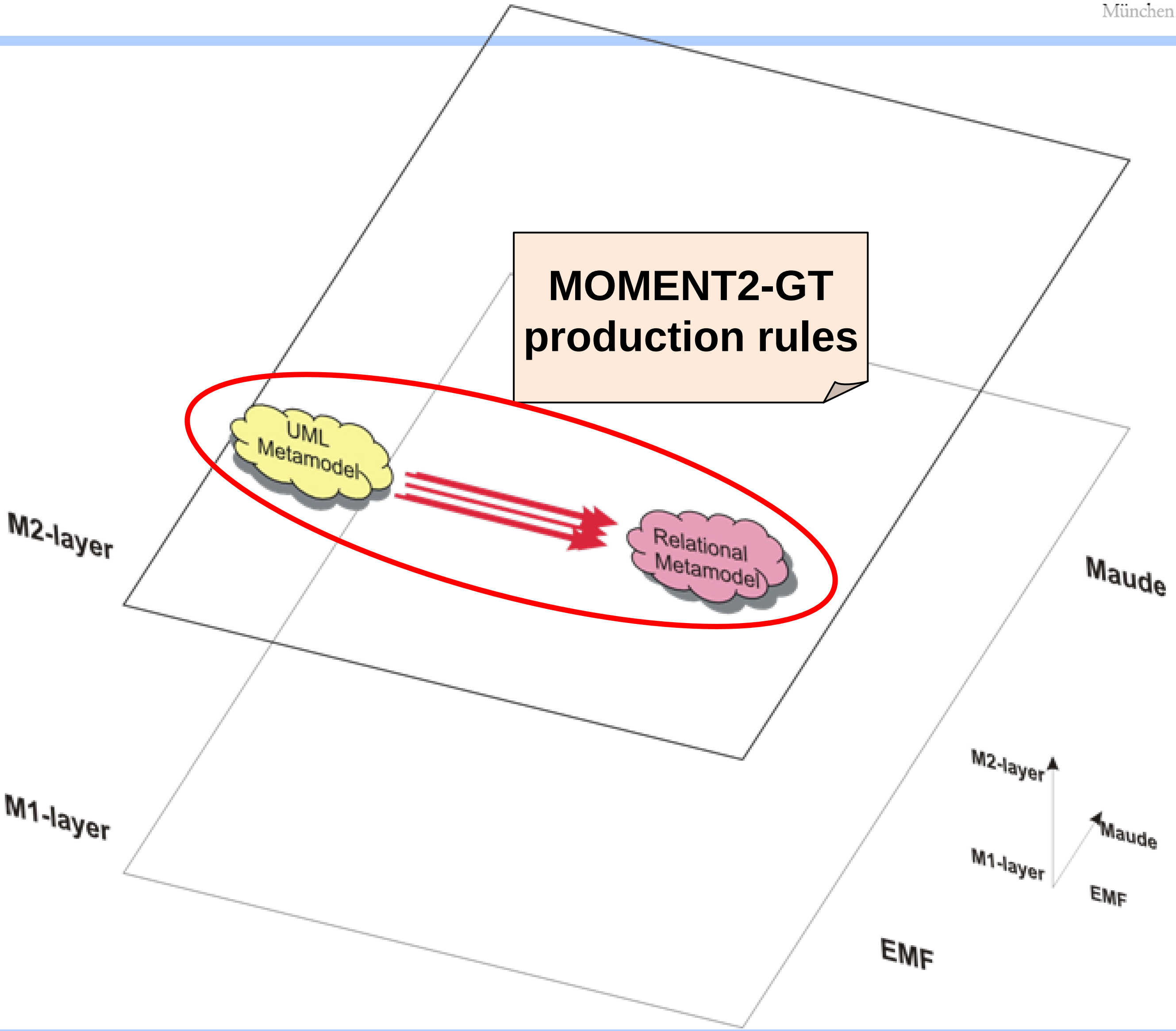
```
... rhs 'uml {
        'c : 'UMLClass {
            'name = 'cName:String
            ‘isPrsstnt = true}};
    rhs 'rdbs {
        't : 'RDBSTable {
            'name = 'cName:String}};}
```

Model Transformations in MOMENT2-GT: Metamodels



[A Boronat]

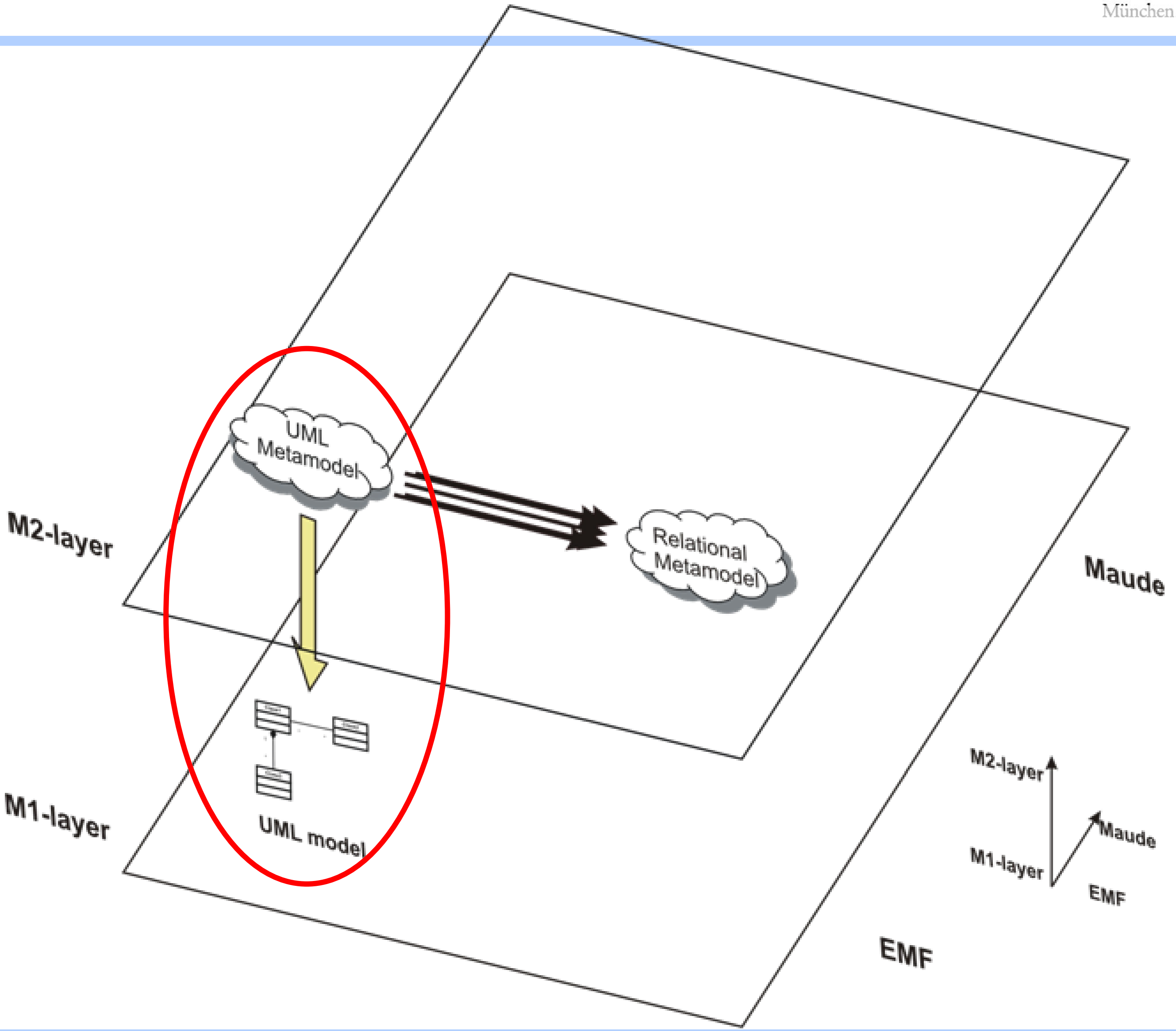
2) Model Transformations in MOMENT2-GT: Production Rules



MOMENT2-GT production rules

[A Boronat]

Model Transformations in MOMENT2-GT: UML Model as Metamodel Instance



[A Boronat]

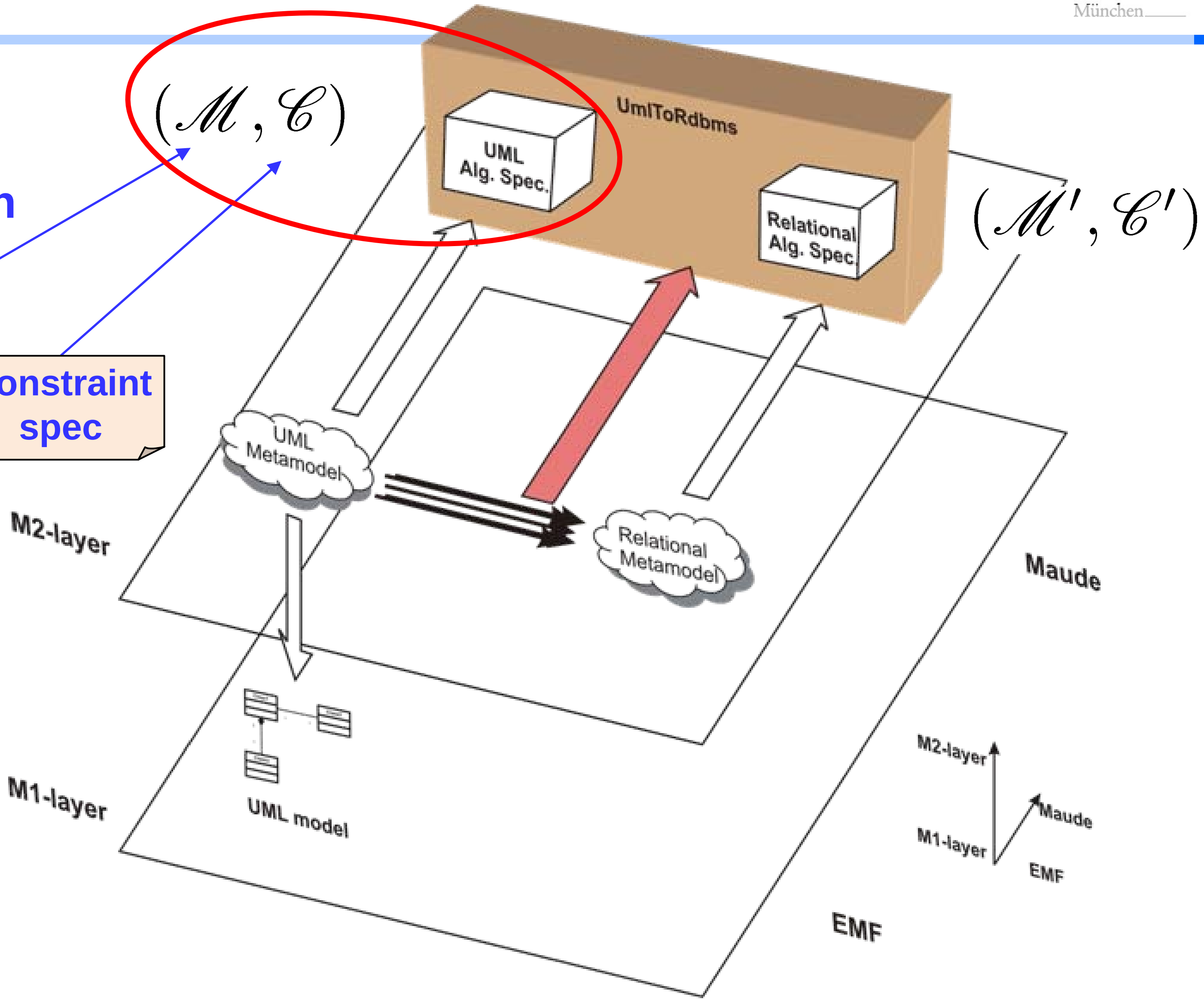
Recap 1)
Metamodel Specifications (in MOMENT2-GT):

Algebraic Metamodel specification

Metamodel spec

Constraint spec

$(\mathscr{M}, \mathscr{C})$



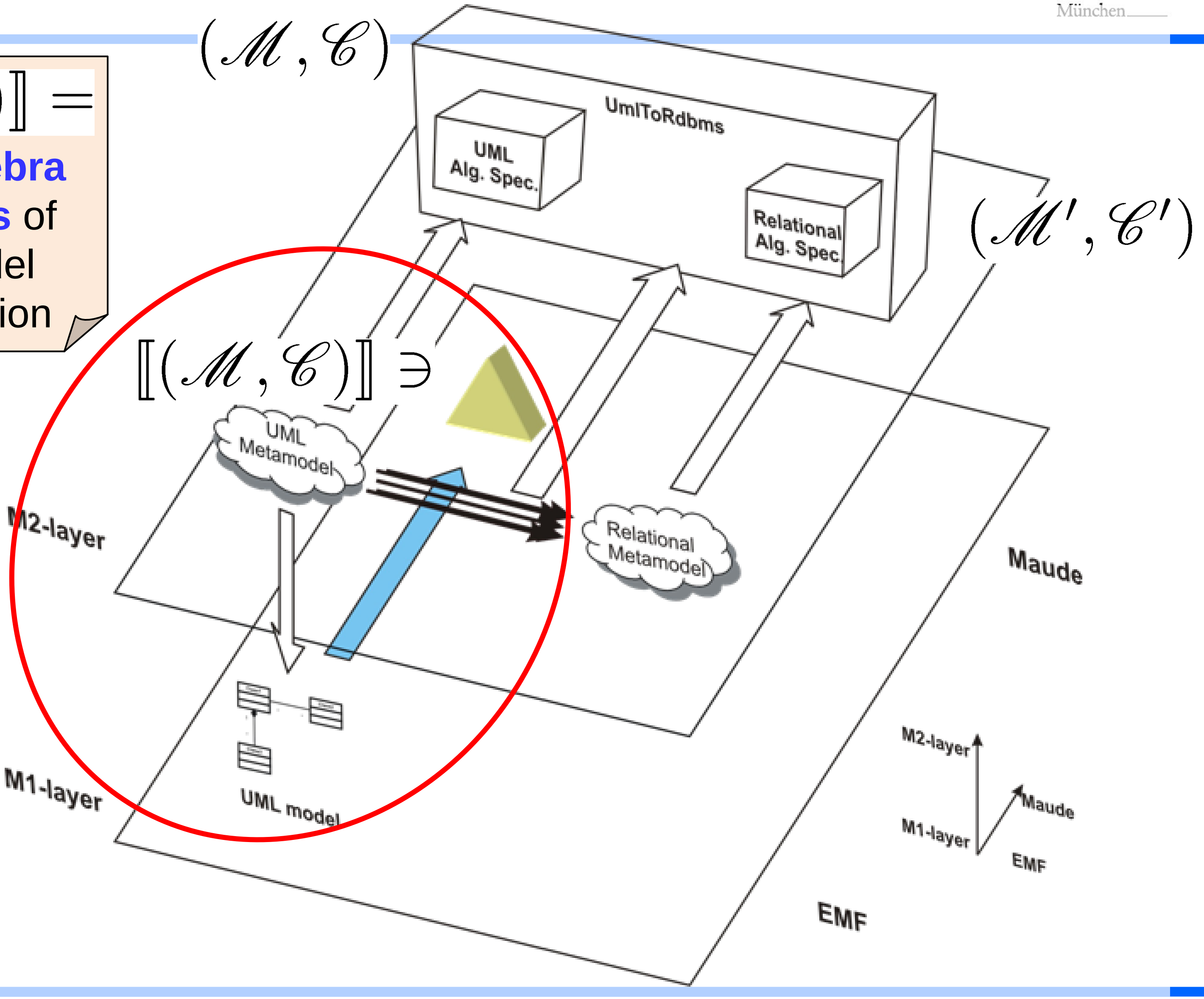
$(\mathscr{M}', \mathscr{C}')$

[A Boronat]

Recap 1)
Metamodel Instances (MOMENT2-GT):

$[\![(\mathscr{M},\mathscr{C})]\!] =$

Initial algebra semantics of metamodel specification



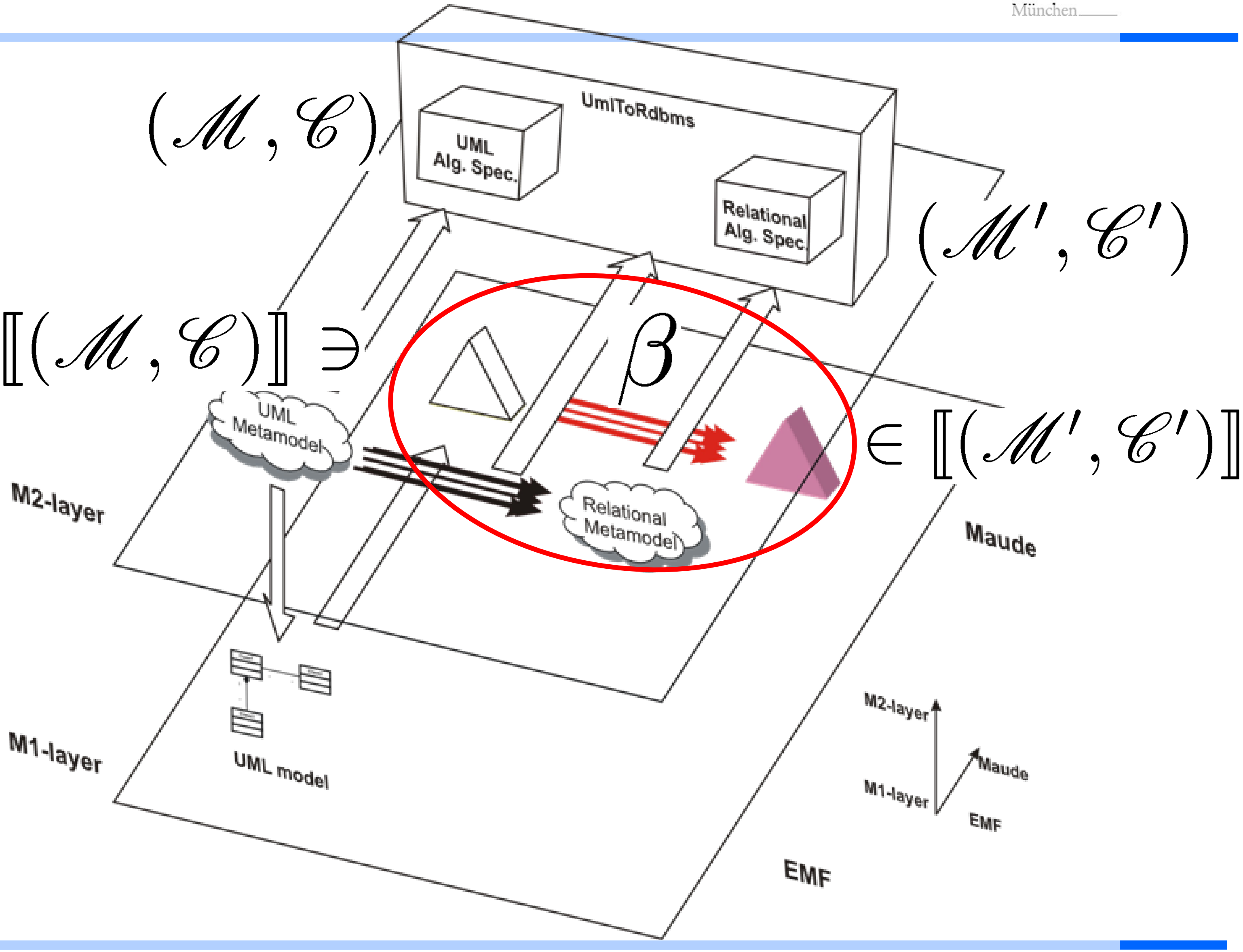
$(\mathscr{M},\mathscr{C})$

$(\mathscr{M}',\mathscr{C}')$

$[\![(\mathscr{M},\mathscr{C})]\!] \ni$

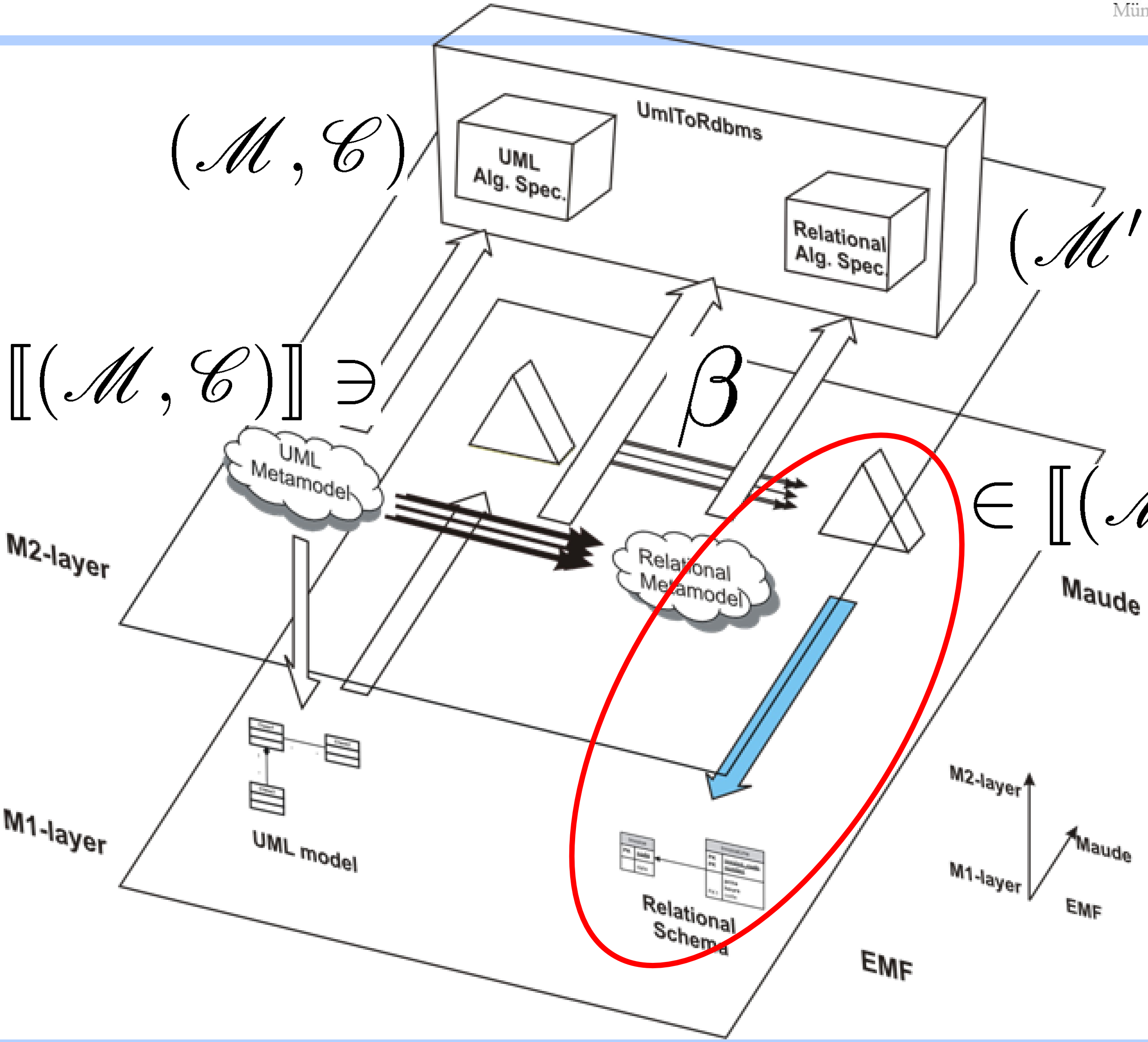
[A Boronat]

Recap 2) Model Transformation (in MOMENT2-GT):



[A Boronat]

Recap 2) Model Transformation (in MOMENT2-GT): Display of Resulting Model



$(\mathscr{M},\mathscr{C})$

$(\mathscr{M}',\mathscr{C}')$

$[\![(\mathscr{M},\mathscr{C})]\!] \ni$

β

$\in [\![(\mathscr{M}',\mathscr{C}')]\!]$

[A Boronat]

3) Institutions: Institutions and their Morphisms

- **Institution** $I = (\mathrm{Sign}^I, \mathrm{Sen}^I, \mathrm{Mod}^I, \models^I)$
- **Institution morphism**

$$\mu : I \to I'$$

$$\mu^{Sign} : \mathrm{Sign}^I \longrightarrow \mathrm{Sign}^{I'}$$

$$\mu^{Sen} : \mathrm{Sen}^I \longleftarrow \mathrm{Sen}^{I'}$$

$$\mu^{Mod} : \mathrm{Mod}^I \longrightarrow \mathrm{Mod}^{I'}$$

+ satisfaction condition

“Richer I over poorer I' “

Institution comorphism

$$\rho : I \to I'$$

$$\rho^{Sign} : \mathrm{Sign}^I \longrightarrow \mathrm{Sign}^{I'}$$

$$\rho^{Sen} : \mathrm{Sen}^I \longrightarrow \mathrm{Sen}^{I'}$$

$$\rho^{Mod} : \mathrm{Mod}^I \longleftarrow \mathrm{Mod}^{I'}$$

+ satisfaction condition

“Embed poor I into rich I' “

[cf. Mossakowski 2005]

3) Institutions: An Institution for UML-Class Diagrams

- **Signatures**
  - Sorts: class names, basic types
  - Operations: typed attributes and methods
  - Relations: association names with corresponding association ends.

- **Sentences**
  - declare multiplicities for associations
  - Example:

$$\mathit{association}(\mathsf{obs}, \mathsf{persons} : \mathsf{Person} : 0..1, \mathsf{observers} : \mathsf{Observer} : 0..\star)$$

- **Structures**
  - of a class diagram signature are sets of object states consisting of:
    - Sets of created object identifiers
    - Functions interpreting attributes and methods
    - Relations interpreting associations.

[CKTW 2008]

3) Institutions: An Institution for Simple Database Schemata

- **Signatures**
  - Sorts: Table names, basic types
  - Operations: typed columns for tables, update operations
  - Operations for primary and foreign keys
- **Sentences**
  - Declare properties of RDB such as PrimaryKey, ForeignKey, NonNull, IsUnique as well as multiplicity constraints
  - Example: PrimaryKey(Person, pkey)
- **Structures**
  - of a database signature consist of
    - relations interpreting tables with elements of basic types
    - functions querying columns, update operations
  - satisfying the axioms for the special properties PrimaryKey, …

3) Institutions: Institutional Semantics

Let $(\mathscr{M}, \mathscr{C})$ be a meta-model specification

- The **institutional semantics** is given by
  - an **institution** I and
  - a **semantic mapping**

$$\sigma : [\![(\mathscr{M}, \mathscr{C})]\!] \to \mathrm{Th}^{I}$$

connecting each SE-model with a theory in I

3) Institutions: Semantic Connection

- Let
  $(I, \sigma : [\![(\mathscr{M}, \mathscr{C})]\!] \to \mathrm{Th}^{I})$ and $(I', \sigma' : [\![(\mathscr{M}', \mathscr{C}')]\!] \to \mathrm{Th}^{I'})$
  be two institutional semantics.

- A **semantic connection**

$$I \overset{\mu}{\to} I_0 \overset{\rho}{\to} I'$$

consists of

  - Mediating institution I_0
  - Institution morphism $\mu : I \to I_0$
  - Institution comorphism $\rho : I_0 \to I'$

3) Semantic Connections: Poor-Man’s-Class Diagrams as Mediating Institution

$$\mathrm{CD} \xrightarrow{\mu} \mathrm{PM\text{-}CD} \xrightarrow{\rho} \mathrm{RDBS}$$

- **Poor Man’s Class Diagrams Institution:**
  - As class diagram institution, but without operations
- **Institution morphism from CDs to poor man’s CDs**
  - Forgetting operations
  - Identity on sentences
- **Institution co-morphism from poor man’s CDs to database schemata**
  - Class to table; association end to table with foreign keys
  - Attributes to columns
  - Multiplicity constraints expressed by cardinality constraints

4) Semantics preserving model transformation

$$\begin{tikzcd}
\llbracket(\mathscr{M},\mathscr{C})\rrbracket \arrow[rr, "\beta"] \arrow[d, "\sigma"'] & & \llbracket(\mathscr{M}',\mathscr{C}')\rrbracket \arrow[d, "\sigma'"] \\
\mathrm{Th}^{I} \arrow[dr, "\mu"'] & & \mathrm{Th}^{I'} \\
& \mathrm{Th}^{I_0} \arrow[ur, "\rho"'] &
\end{tikzcd}$$

- A **model transformation** β is called **semantics preserving** (w.r.t. the semantic connection $(\mu,\ \rho)$) if for any SE-model M the models of the ρ-reduct of $\beta(M)$ are models of the μ-image of M :

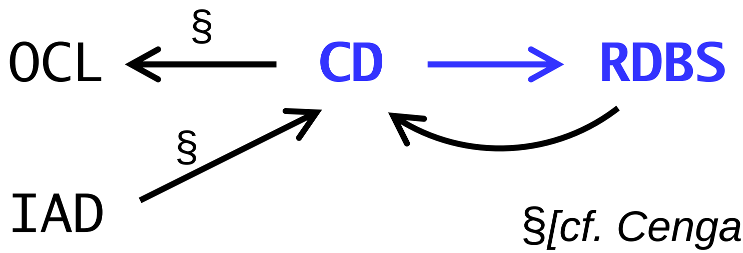
$$\rho^{\mathrm{Mod}}(\mathrm{Mod}(\sigma'(\beta(M)))) \subseteq \mu^{\mathrm{Mod}}(\mathrm{Mod}(\sigma(M)))$$

What is a multi-modelling language?

- **A multi-modelling language is given by**
  - A **family of metamodel specifications** $(\mathscr{M}_l, \mathscr{C}_l)_{l \in L}$
  - **Institutional semantics** $\sigma_l : [\![(\mathscr{M}_l, \mathscr{C}_l)]\!] \to \mathrm{Th}^{I_l}$
  - A **net of semantics preserving model transformations**

$$\beta_{lm} : [\![(\mathscr{M}_l, \mathscr{C}_l)]\!] \to [\![(\mathscr{M}_m, \mathscr{C}_m)]\!]$$

- **Example:** RDBS+UML



OCL

§

CD

RDBS

§

IAD

(interaction diagrams)

§*[cf. Cengarle, Knapp, Tarlecki, W UMFest 08]*

Concluding Remarks

- **New, semantically well-founded notion of a multi-modelling language**
  - Algebraic semantics of meta-models
  - Heterogeneous institutional semantics
- **New notion of semantic correctness for model transformations**
  - Semantics preserving model transformations based on institutions
- **Composition operators**
  - Grothendieck construction
  - Development graphs (HETS)
- **Proof-theoretical correctness conditions**
  - General logics
- **Tool support**