On the Use of Ontologies in the Software Process

Uwe Aßmann

Technische Universität Dresden

Institut für Software- und Multimediatechnik

uwe.assmann@inf.tu-dresden.de

Suppose you were Mr Bernhard...

Suppose you were Mr Bernhard...

How to build the product data of the next version of the Daimler-Chrysler CLS?

What is an Ontology?

What is a *model*?

- An abstract representation of something
- Here: a partial type-based specification

What is a *taxonomy*?

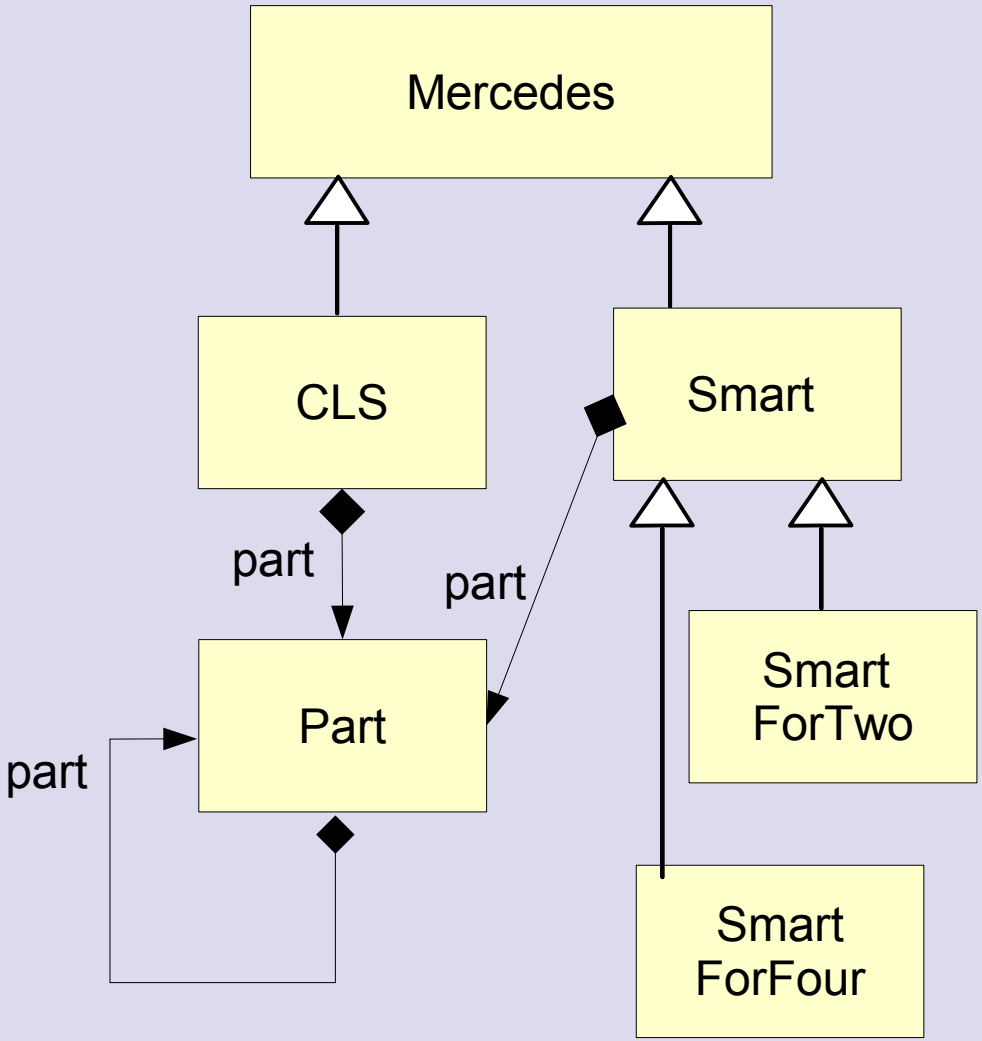
- An inheritance hierarchy or lattice of concepts or types

What is an *ontology*?

- A taxonomy
- Plus relations between the concepts
- Plus integrity constraints, i.e., conditions about the object space (declarative)
- ***standardized in a domain***

- **For an ontology, you need the consent of many users**

Car Ontologies



Invariant:

- Forall p in Smart.part*:
  if p in CLS.part*
  then mustBeCheckedSpecial(p)

What is an Ontology?

A standardized, logic-based type model of an application domain

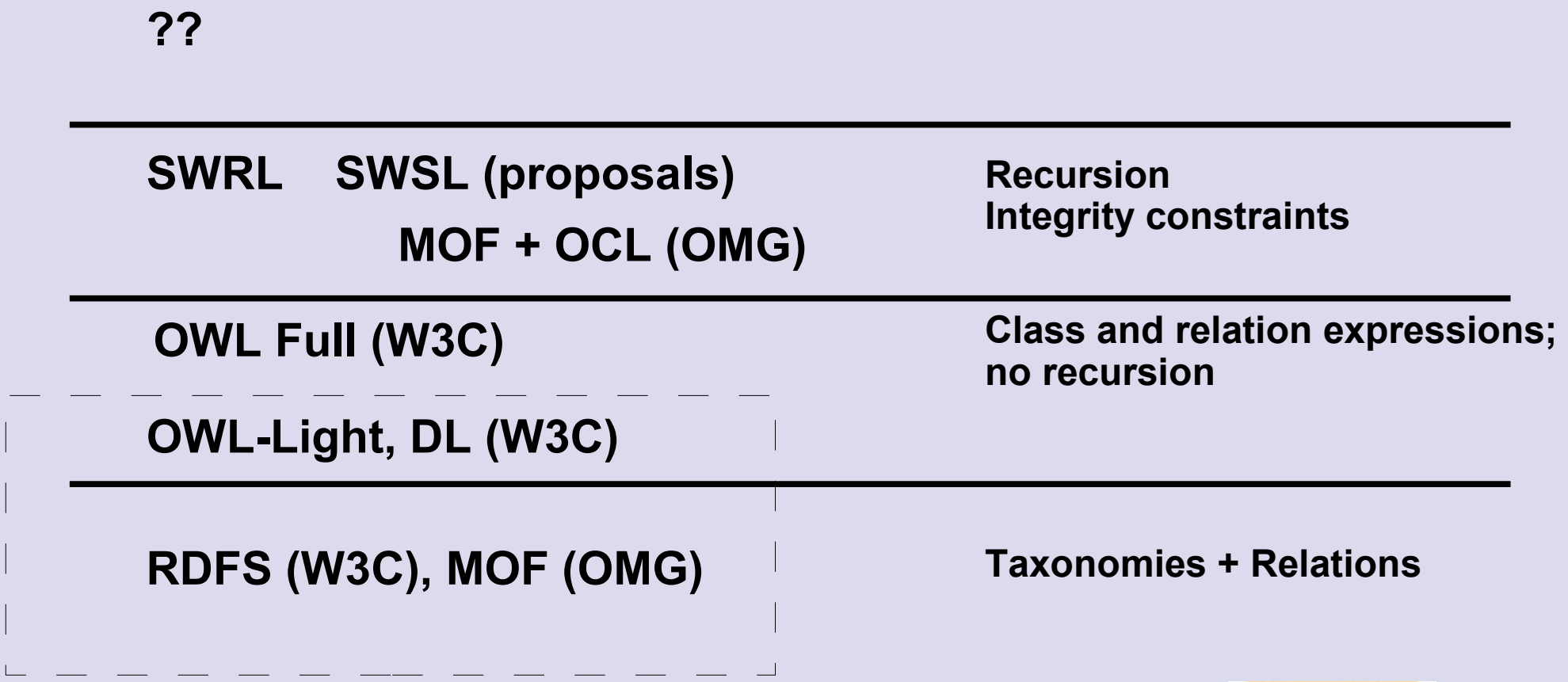
How to use it in our software??

- Bioinformatics
- Medicine
- Embedded systems
- Business software
- Car industry (Daimler CLS)
- http://www.daml.org/ontologies/class.html ....

Ontology Languages

A language hierarchy is intended

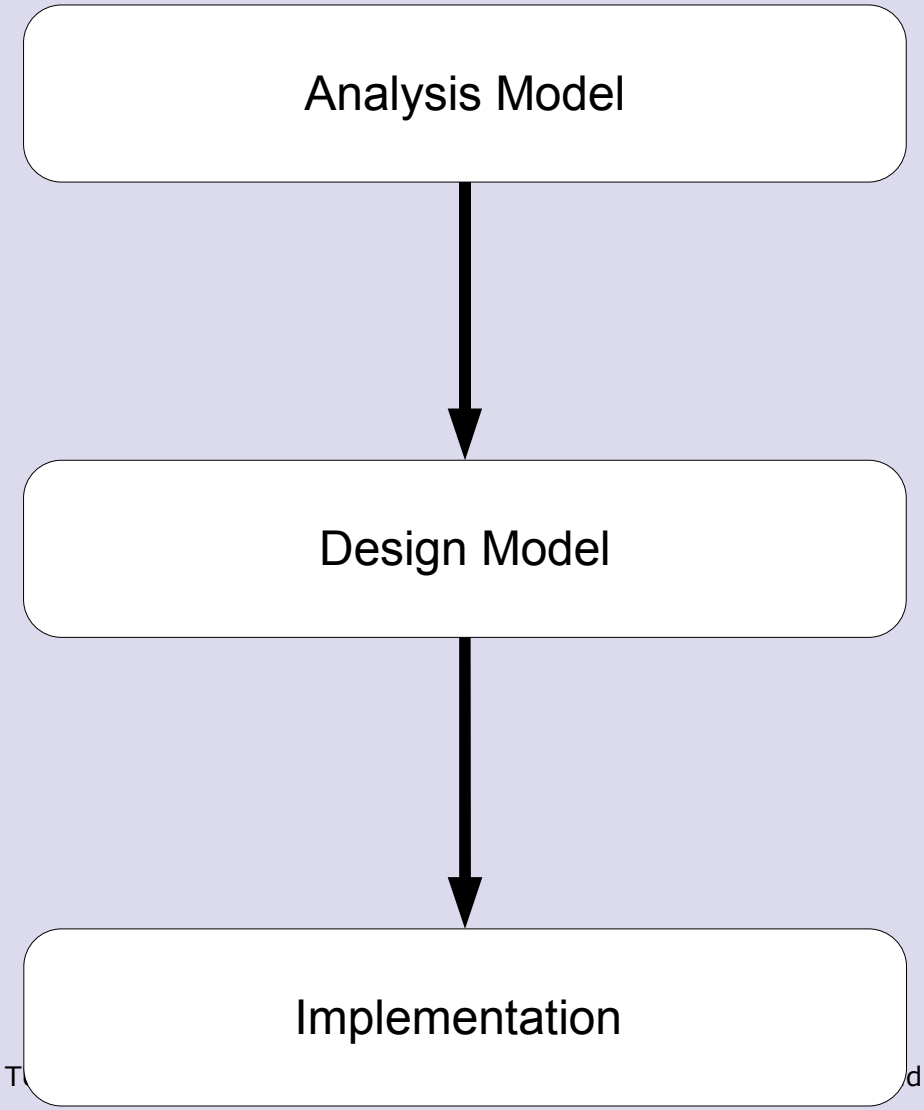
Top Level: „Prolog“ with types and inheritance lattices



How to Use Ontologies in the Software Process

Standard Object-Oriented Software Development

A la Booch/Rumbaugh/Jacobsen (3 amigos)

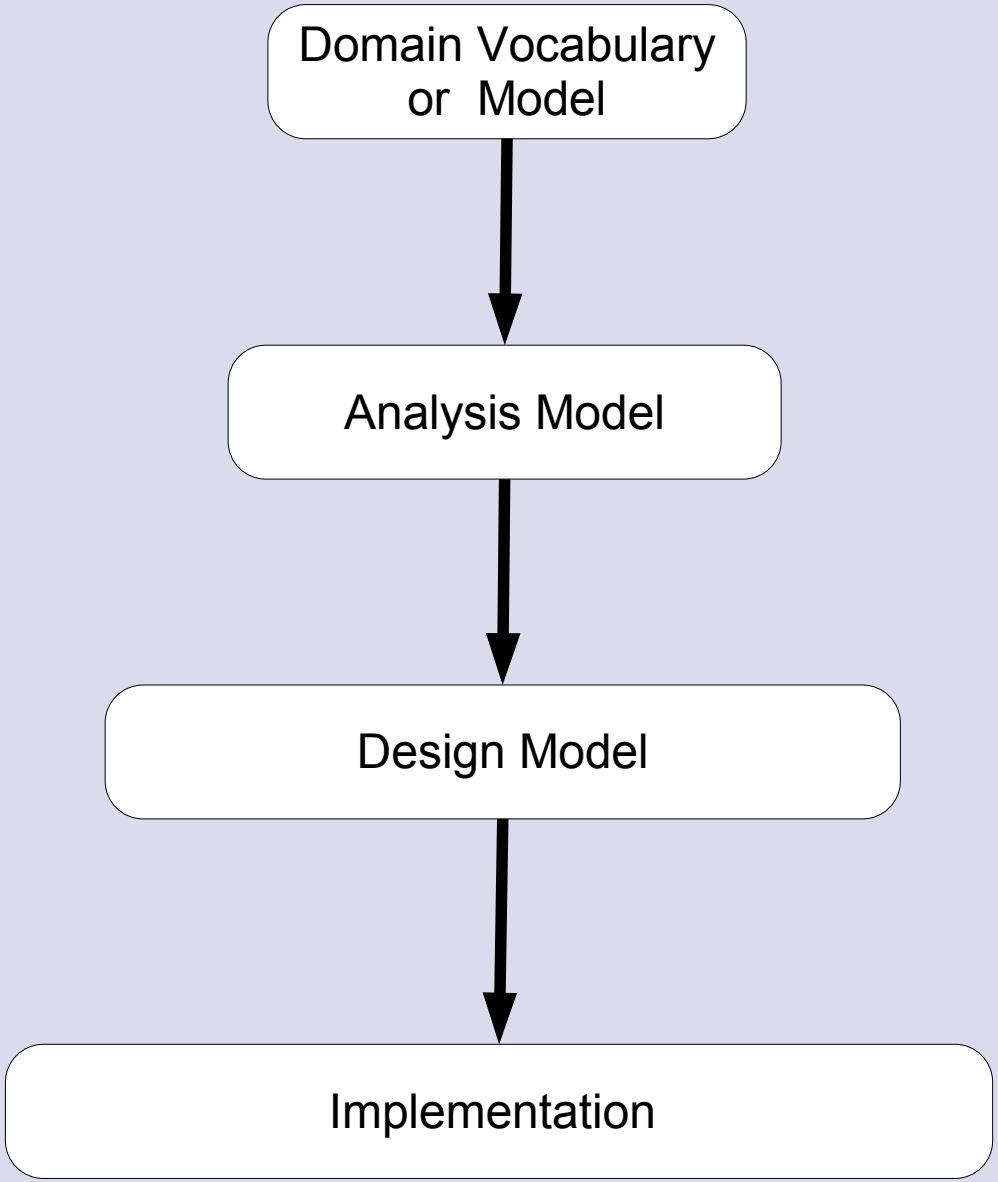


Domain vocabulary. Specifications for requirements of a system

Additions for the technical design of the system

Adding implementations of all parts

Refinement 1: With Domain Analysis



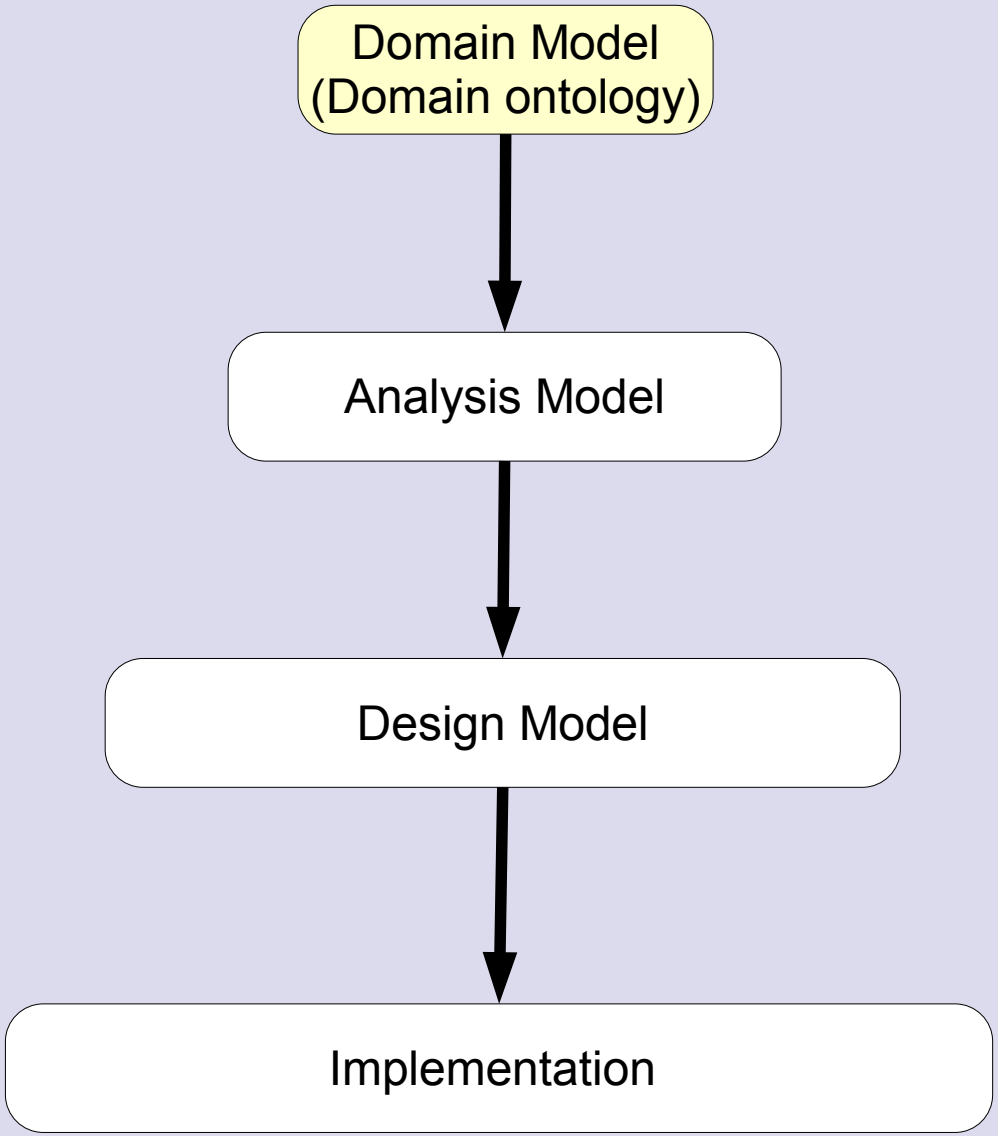
Domain vocabulary, perhaps with relations

Specifications for requirements of a system

Additions for the technical design of the system

Adding implementations of all parts

Refinement 2: With Standardized Ontology



Domain vocabulary
with relations, integrity constraints,
and rules

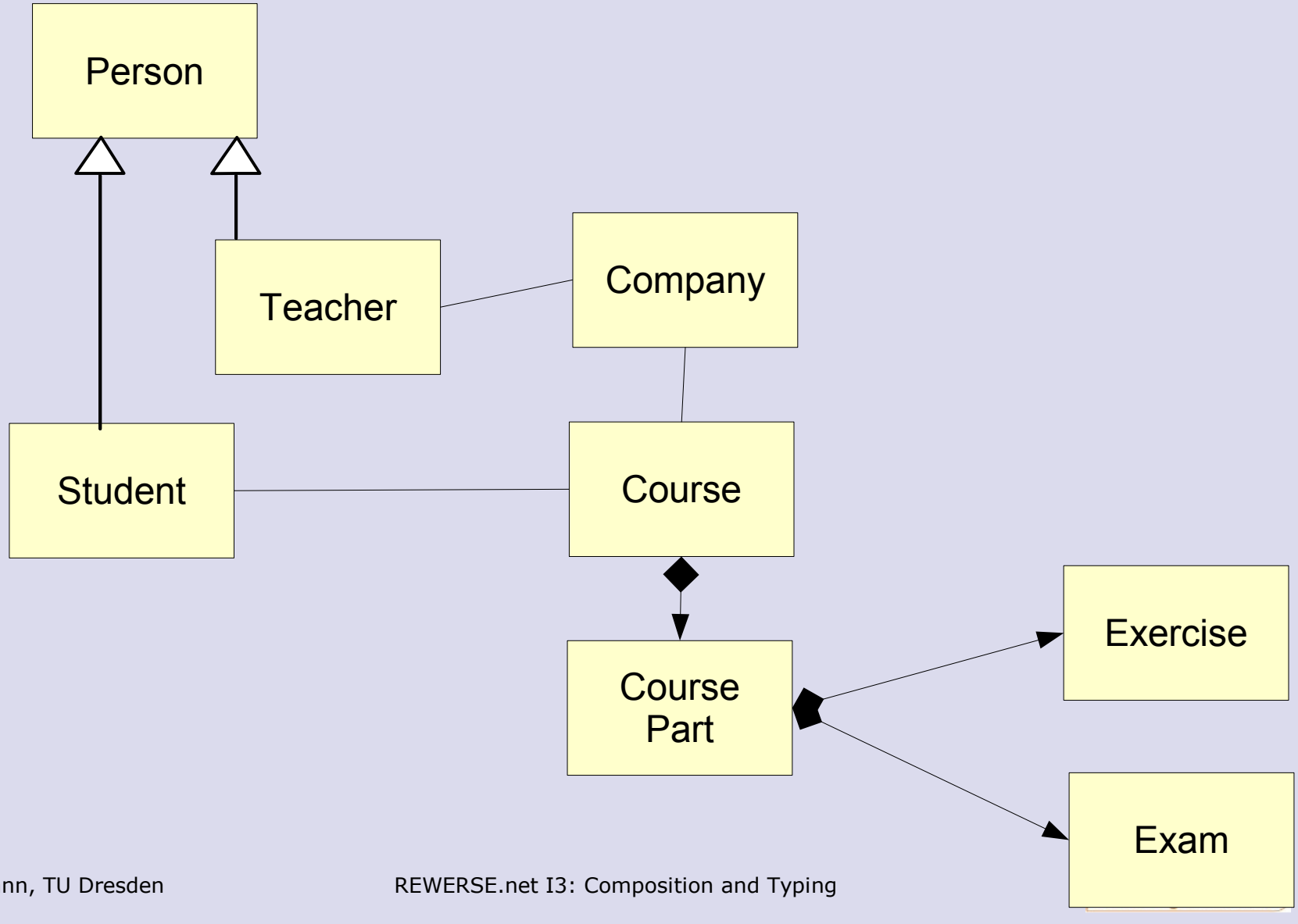
Specifications
for requirements of a
system

Additions for the
technical design of the
system

Adding implementations
of all parts

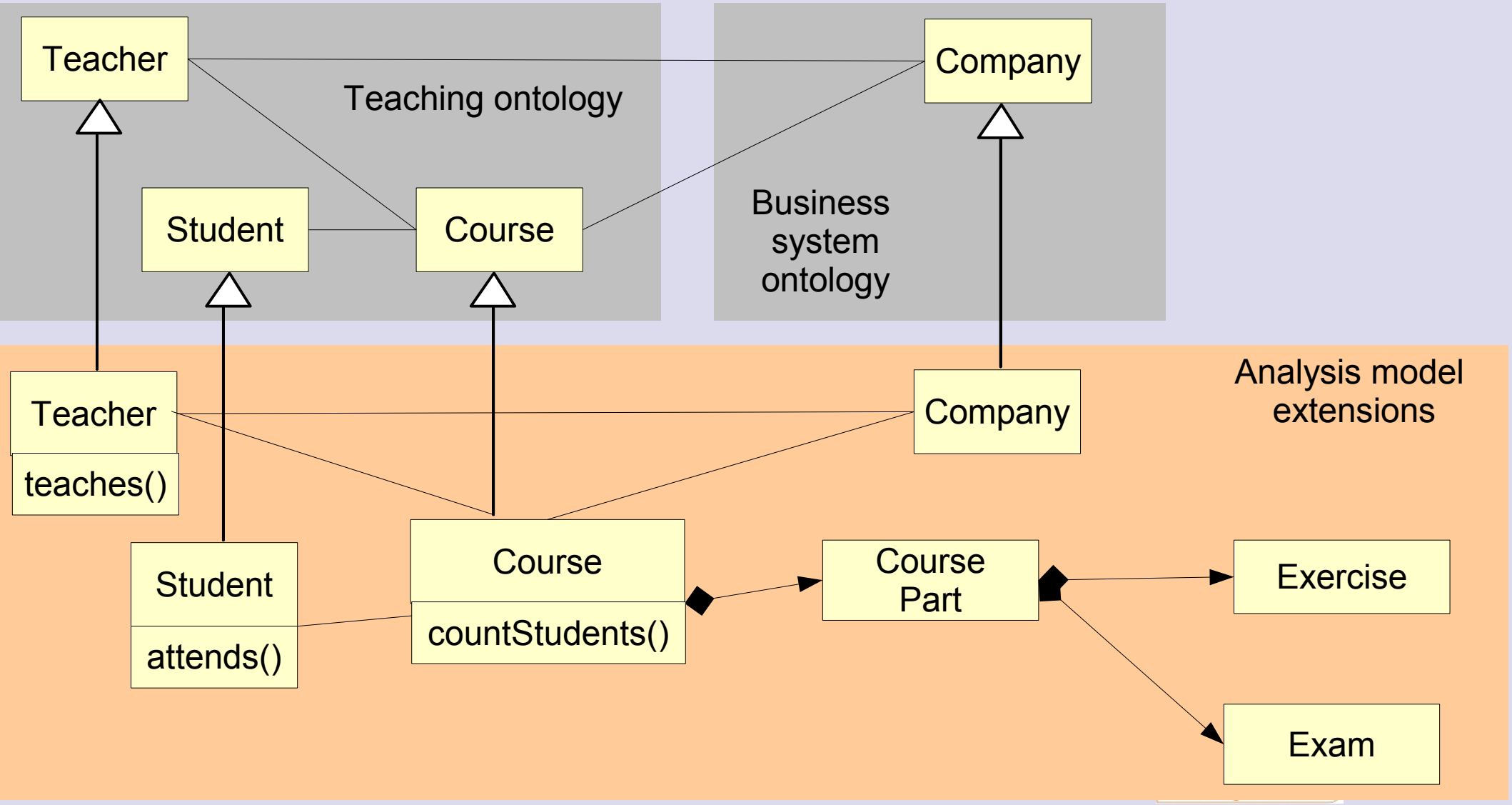
Today: A Simple Domain Model for a Course Management System

Usually self-made



Tomorrow: Deriving Models from the Domain Ontologies

Standardized domain ontologies are reused



Tomorrow

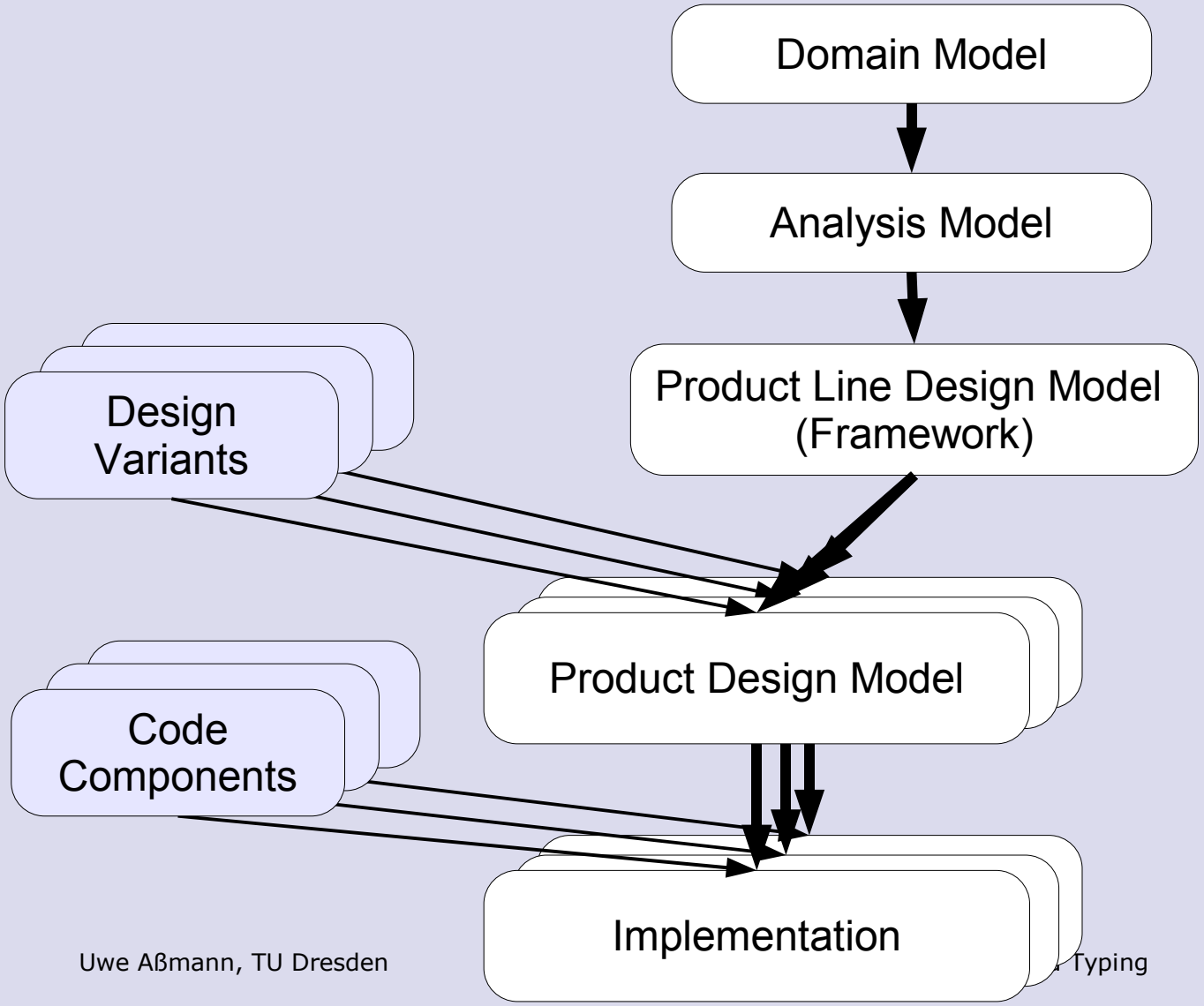
A software product has to be developed around one or several domain ontologies

Because these application experts will use ontologies to express their domain knowledge.

Product-Line Development

Today: Development of Product Lines

Commonality-Variability Analyis of the domain [DraCo]



Domain vocabulary of the product family

Specifications for requirements of a product family

Additions for the technical design of the product family (commonalities)

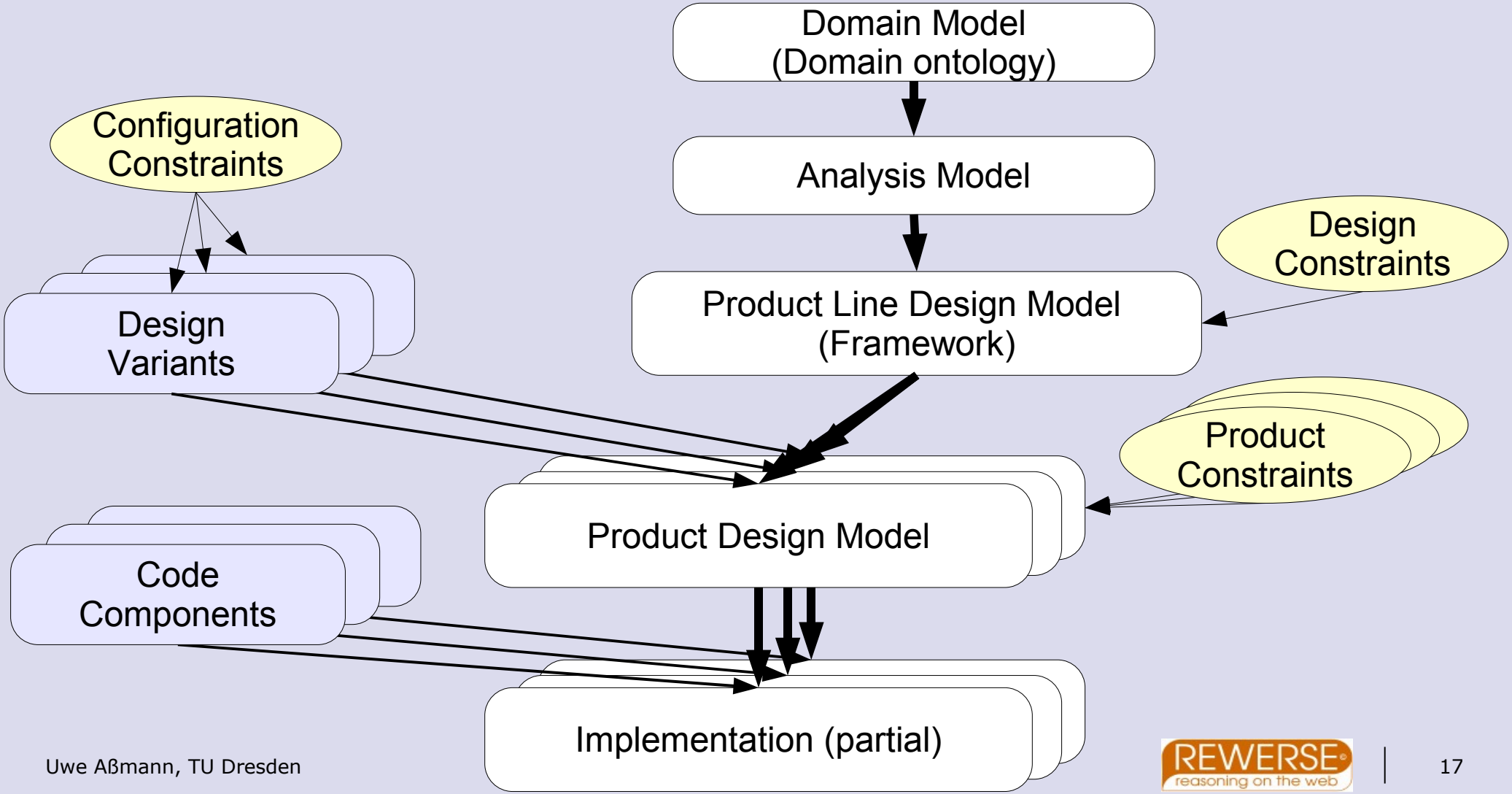
Additions for the technical design of the product (product variants)

Adding implementations of all parts

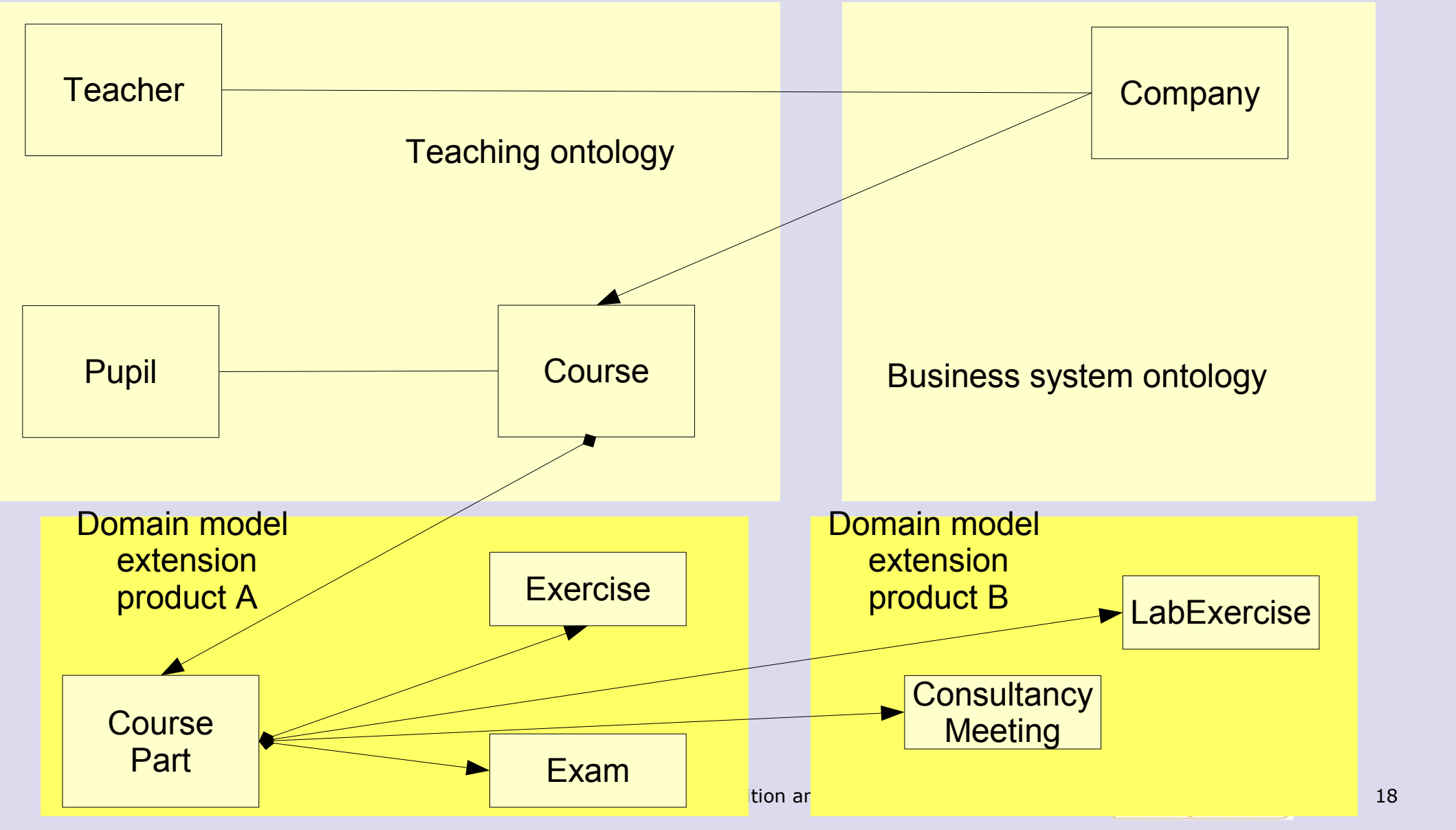
Tomorrow: Product Line Development with Ontologies

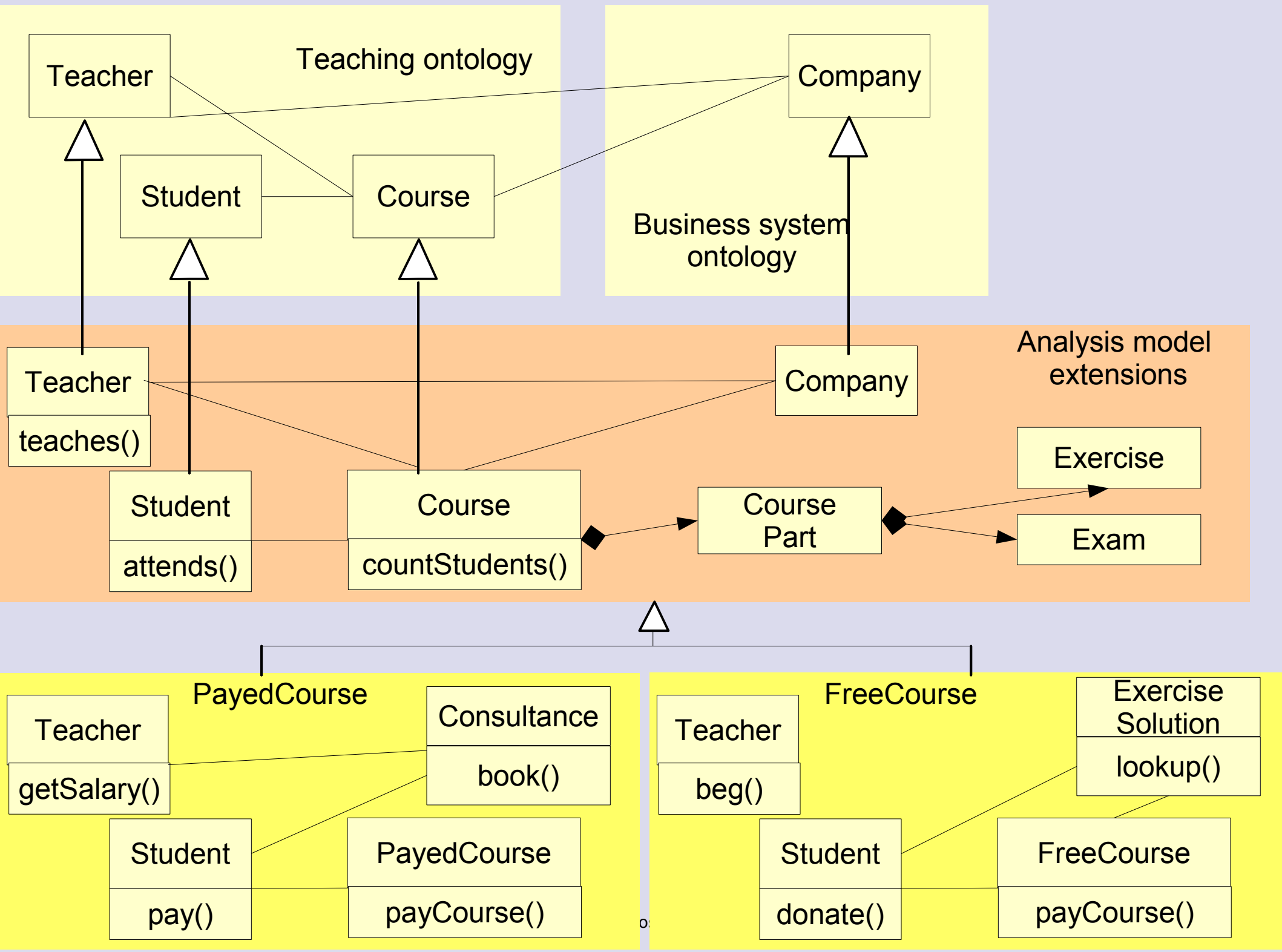
Ontologies can express the terminology of a product line

- The domain, design, the configuration, and product constraints



The Start of a Course Management System Product Line




Teacher
Teaching ontology
Company
Student
Course
Business system ontology
Teacher
teaches()
Company
Analysis model extensions
Exercise
Student
attends()
Course
countStudents()
Course Part
Exam
PayedCourse
FreeCourse
Teacher
getSalary()
Consultance
book()
Teacher
beg()
Exercise Solution
lookup()
Student
pay()
PayedCourse
payCourse()
Student
donate()
FreeCourse
payCourse()

Product-Line Development of the Future

No product line without product line ontologies

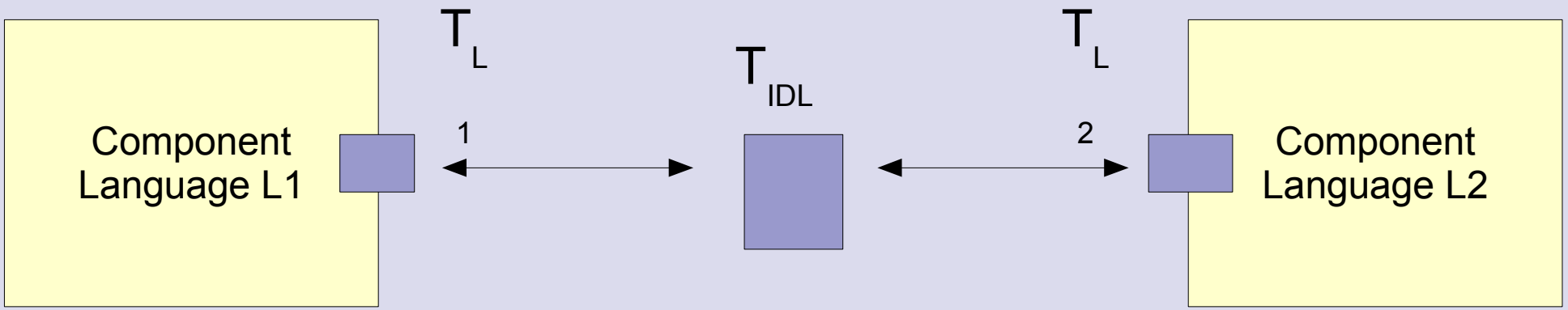
Product-line ontologies will be declarative, logic-based models of the different aspects of a product line.

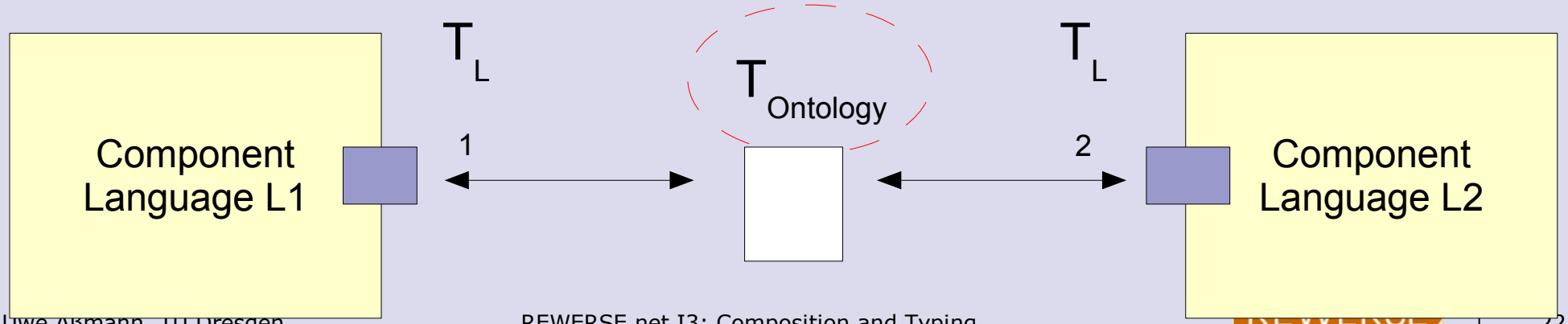
- Standardized for product line community
  - Users, vendors, OEMs, etc.
- Big frameworks have thousands of users

Another Application: Service-oriented Architectures (SOA)

How to Improve on CORBA

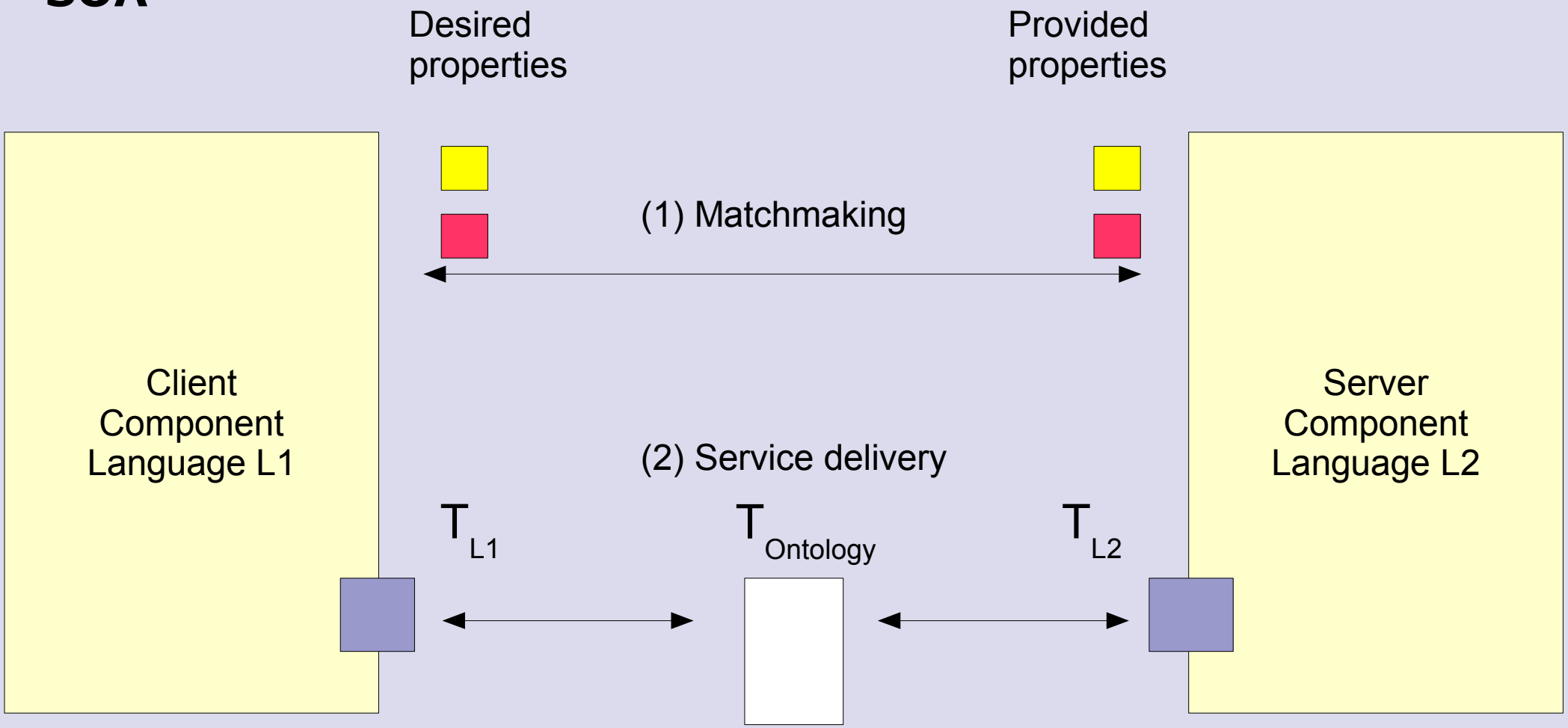
Better understanding, if types come from standardized ontologies





Matchmaking in a SOA

Ontologies are indispensible for good matchmaking in a SOA



The REWERSE Network of Excellence

... reasoning on the web ...

The REWERSE Network www.rewerse.net

Develop rule-based languages for reasoning on the web

6th framework Network of Excellence

- 2003-2007, > 5 Mio Euro
- Coordinator TU München (Francois Bry)
- 27 partners

Working Groups

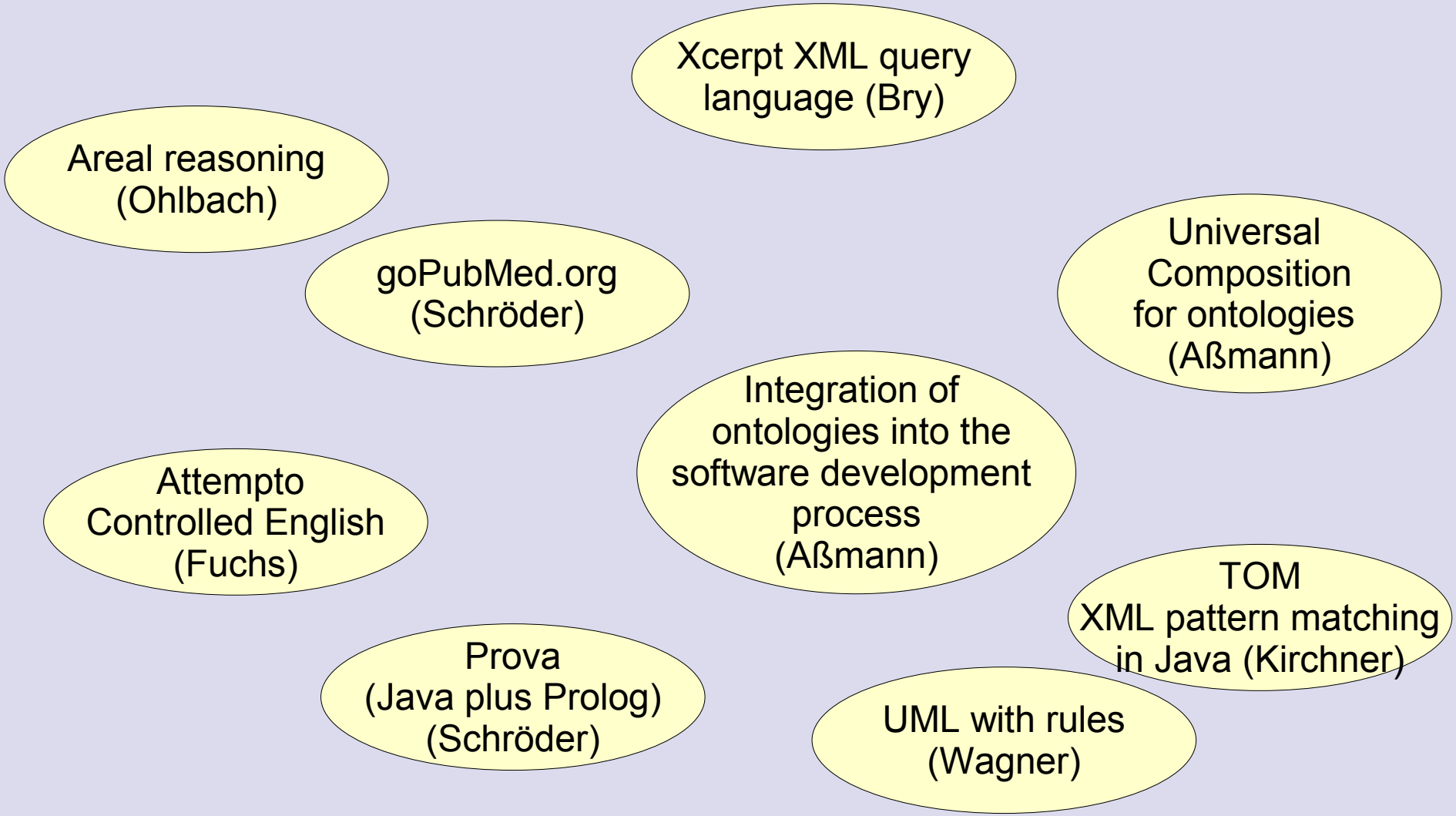
5 technical working groups

- I1: Rule markup languages (Wagner, Cottbus)
- I2: Policy specification, composition, and conformance (Bonatti, Naples)
- I3: Composition and Typing (Aßmann, Dresden)
- I4: Reasoning-aware querying (Bry, Munich)
- I5: Evolution and reactivity (Alferes, Lisboa)

3 application groups

- A1: Web-based decision support for event, temporal, geographical data (Ohlbach, Munich)
- A2: Bioinformatics web (Schröder, Dresden)
- A3: Personalized information systems (Henze, Hannover)
- ET: Education and training (Maluszynski, Linköping)
- TTA: Technology transfer and awareness (Geisler, webXcerpt)

Some Activities So Far



Working Group I3: Composition and Typing

Develop a composition technology for ontologies

- Based on invasive software composition and type systems described by metamodels
- Transfer the Java-based technology to ontologies
- Develop a *universal* ontology composition method for many ontology languages

Develop a typing technology for web-query languages

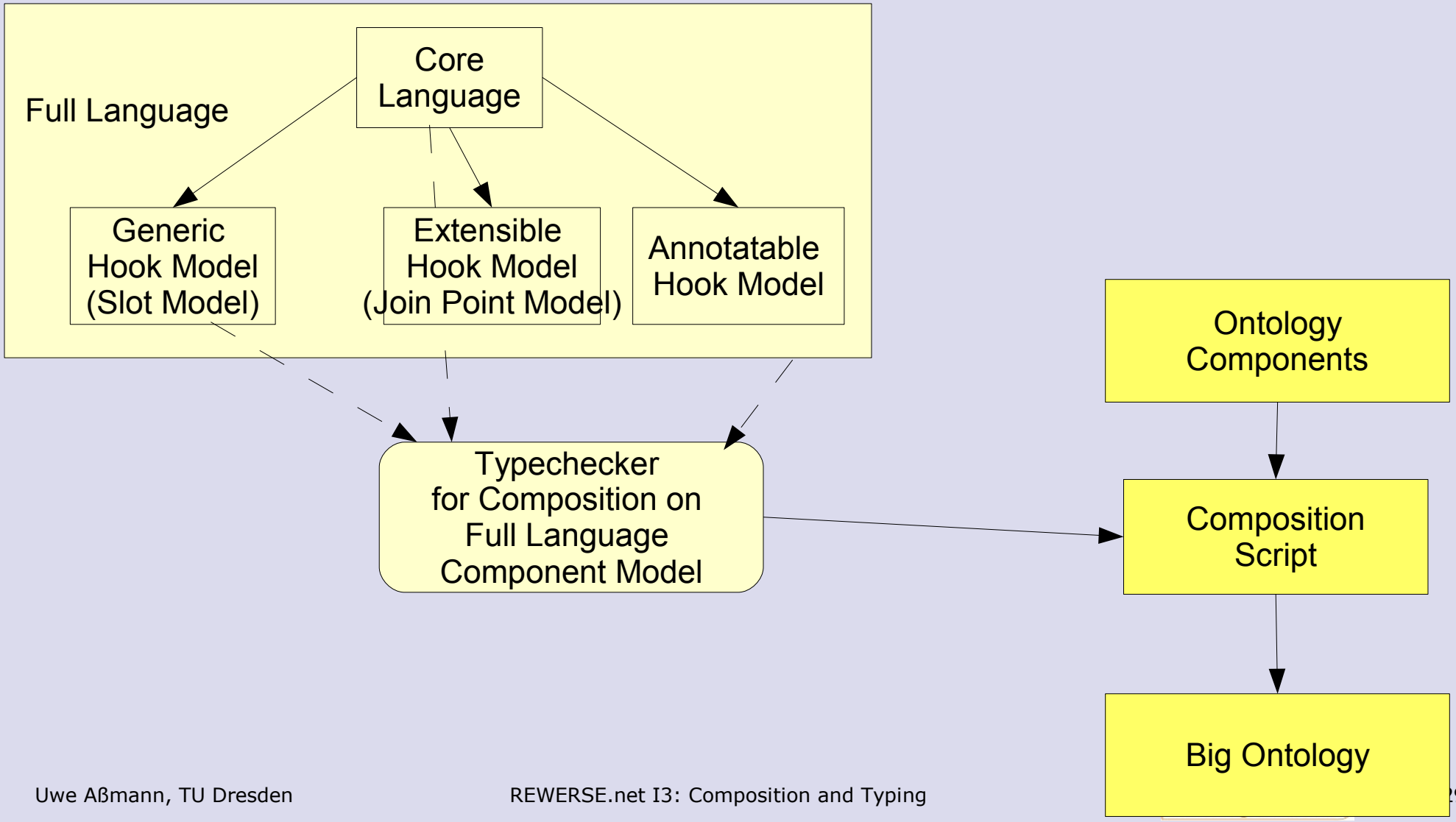
- Use it for composition of ontologies

Some Questions:

- How to master large ontologies (Geneontology.org)?
- How to reuse ontologies?
- How to use ontologies in applications?
- How to use typing technologies to improve

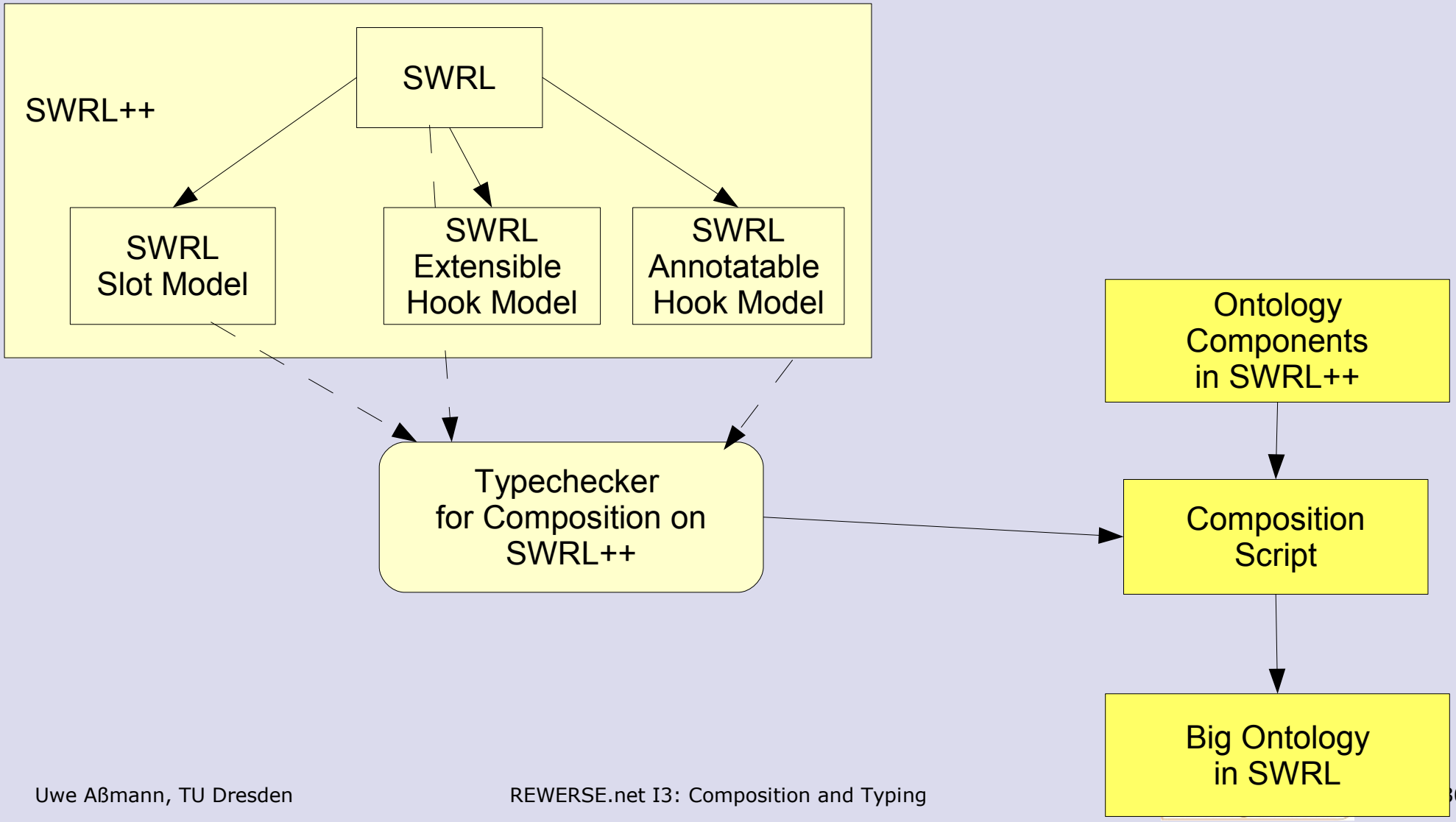
I3: Derive a Component Model from a Core Ontology Language

From a core metamodel, derive a component model and type checker for compositions



I3: Derive a Component Model for SWRL

From a core metamodel, derive a component model and type checker for compositions



Many More Questions

How to describe the business model of a big company?

How to describe a product line ontology of SAP?

- SAP complete bases its software architecture on *semantic services*
- Many application domains
- Many specific notions in the framework

How to build tax ontologies?

How to build the ontology of the Porsche 911?

How to make tools interoperate with ontologies?

Research Questions for Typing Community

How to compose ontologies of different application domains?

- Ontology mapping
- Ontology merging
- Ontology collaboration

Treating polymorphism in ontologies

Developing efficient ontology reasoners

- How to check the product data of a car manufacturer supplier chain?

Identifying more decidable subsets of FOL in the spirit of DL and Datalog

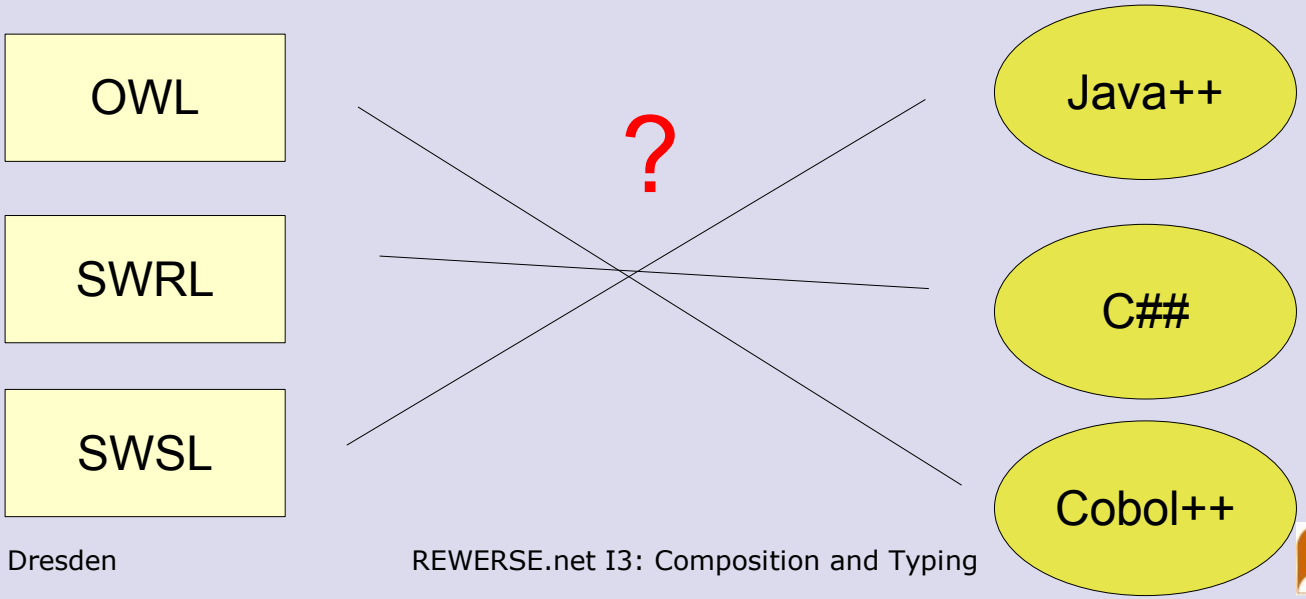
- What lies between DL and Datalog? (DLP)
- How can the efficient algorithms for set constraints and control flow analysis be taken over to ontology processing?

The Big Question

Every software product has to take in domain ontologies

Any serious programming language of the future needs to *understand* ontology languages

- Compilers and tool chains need to read ontologies



The typing community and ontology community need each other

How to Link This with MDA ?

1st European Conf on MDA, Foundations and Applications

- www.ecama-fa.org
- Nov 7-10, 2005, Nuremberg

UML/Models

- http://www.umlconference.org/
- Oct 2-7, 2005, Jamaica

PPSWR 2005 Reasoning on the web

- http://contraintes.inria.fr/~fages/PPSWR05/

REWERSE Summer School in Malta

July 25-29, 2005

www.reasoningweb.org

Book LNCS 3564

- Uwe Aßmann. Reuse in semantic applications. In Norbert Eisinger and Jan Maluszynski, editors, Summer School of REWERSE (Reasoning on the Semantic Web), Lecture Notes in Computer Science 3564, July 2005. Springer.

Book „Invasive Software Composition" Springer 2003.

- Composition of Java fragments.

The End

http://www.w3.org/2001/sw/

www.rewerse.net

http://www-st.inf.tu-dresden.de

www.ecmda-fa.org

www.reasoningweb.org