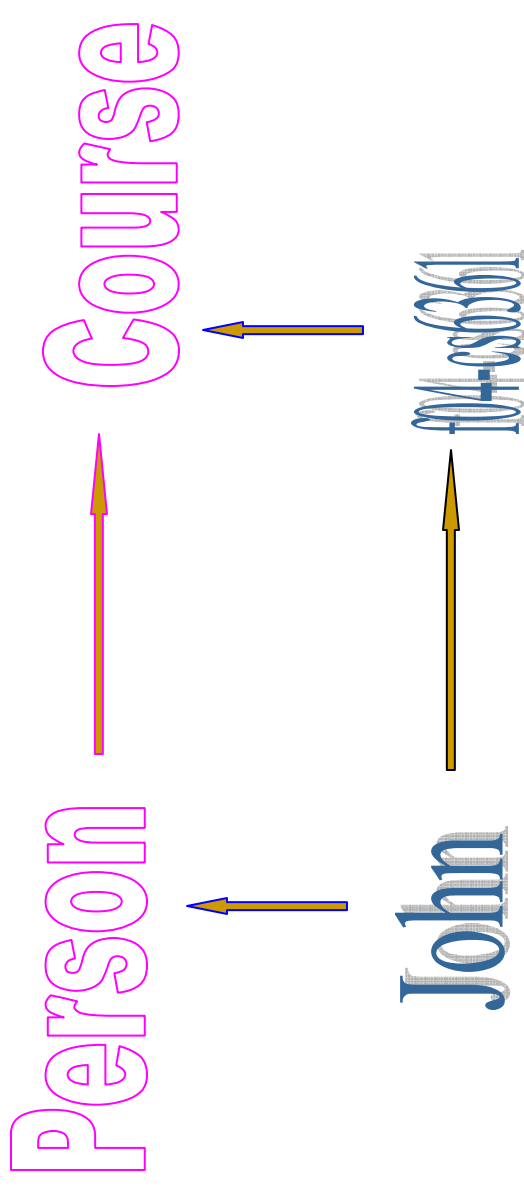
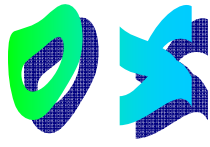

Metamodel

Global view

Some sources

- Object Modeling with UML: Advanced Modeling, Karin Palmkvist, Bran Selic, and Jos Warner, March 2000
 - [UML 2.0 Superstructure Specification](#), **OMG**, 2004.
-

Review: Instances and Classes



Instances: tokens, specific things
Classes: collections of instances

Review: Instances and Classes



Why?

Person



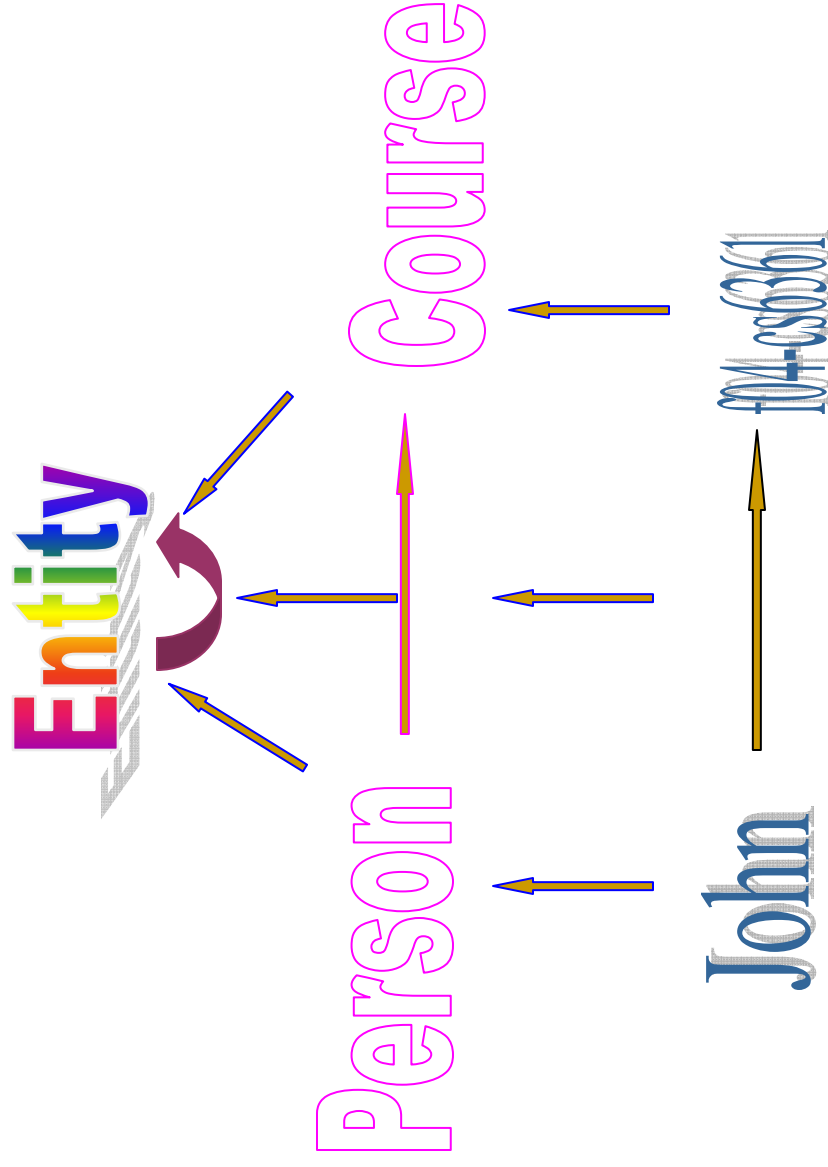
John



104-es6661

Instances: tokens, specific things
Classes: collections of instances

Models and Metamodels



Specific models: tokens, specific things, and relationships

Generic models: collections of instances, and relationships

Metamodels: ontology, ...and epistemology, of generic models

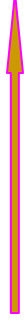
Models and Metamodels



Why?



Person



Course



John

104-csb661

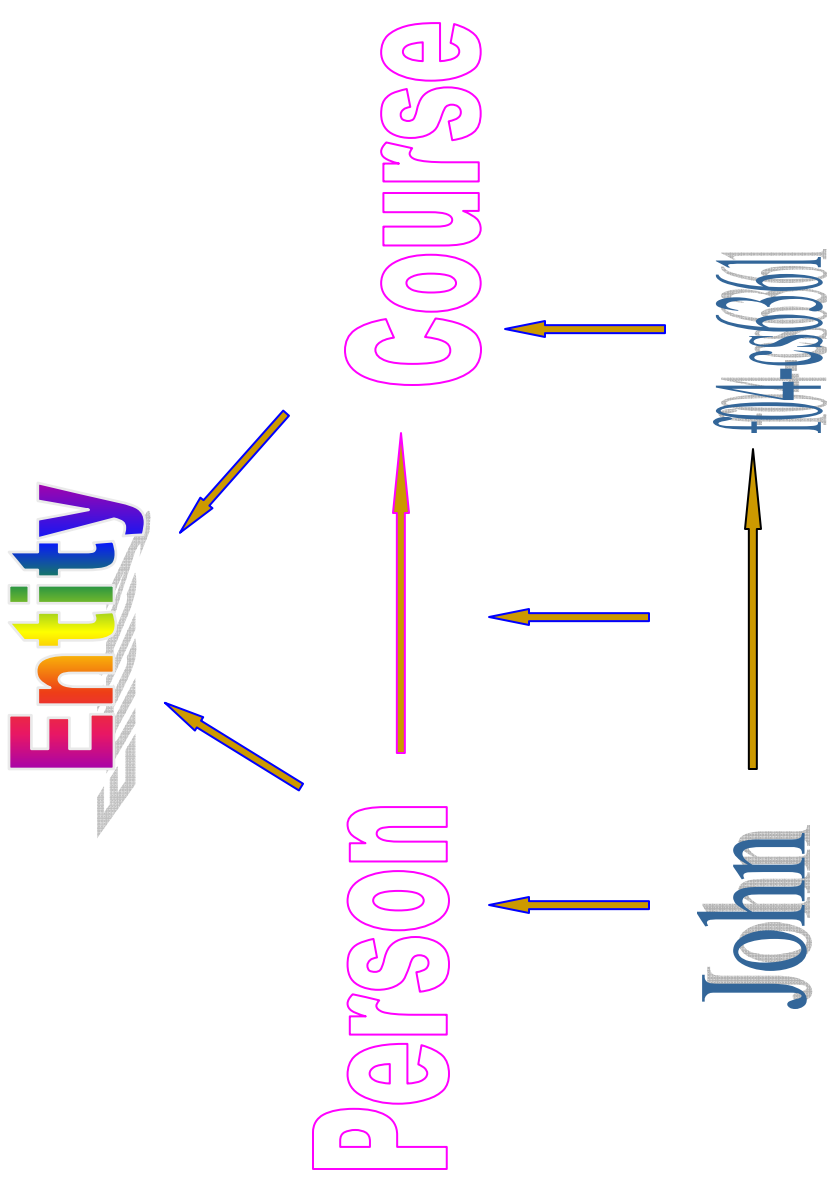


- Specific models: tokens, specific things, and relationships
 - Generic models: collections of instances, and relationships
 - Metamodels: ontology, ...and epistemology, of generic models
-

Models and Metamodels



Why?



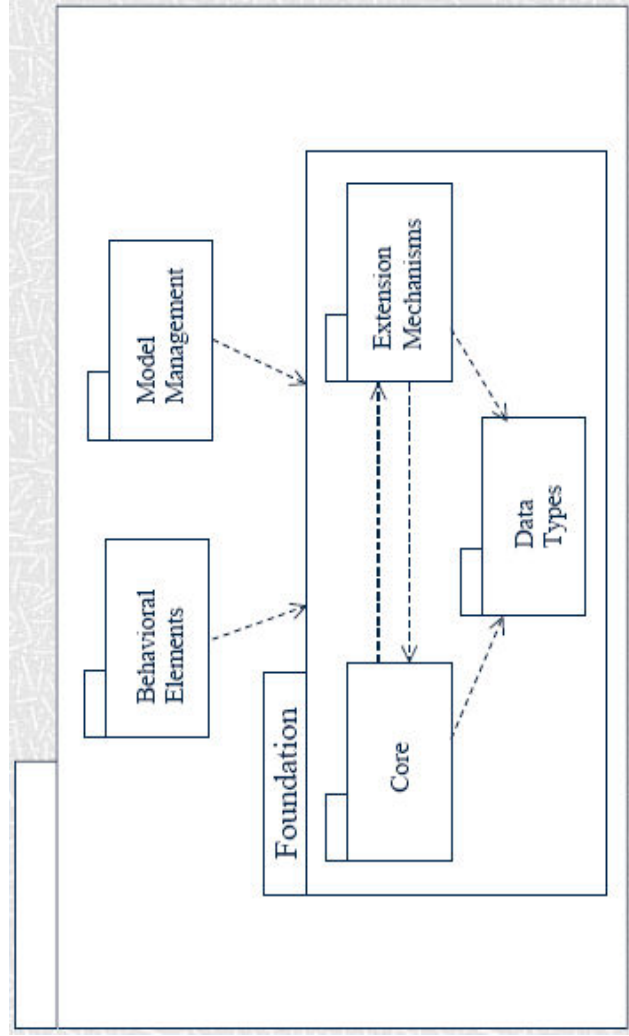
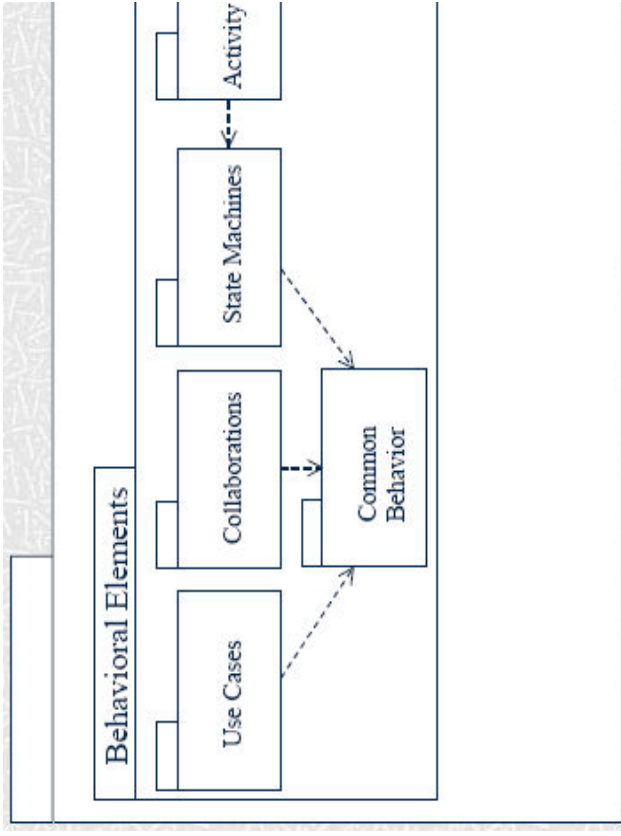
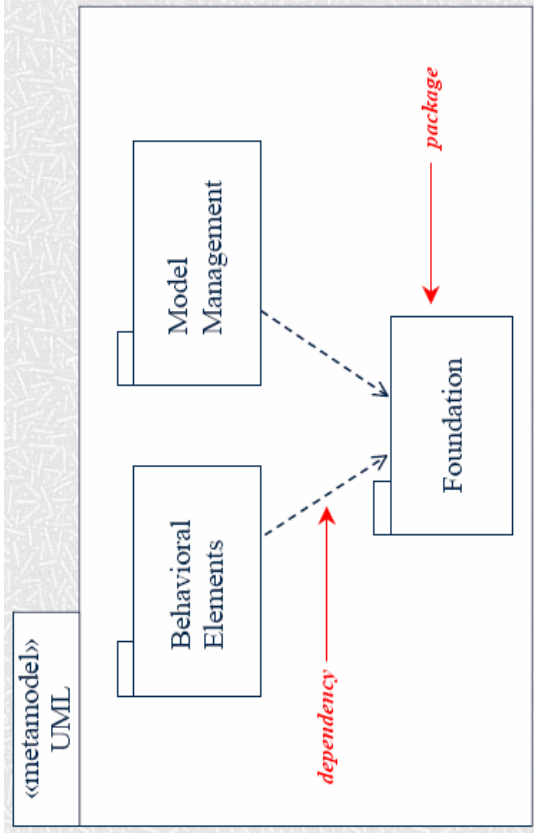
- Specific models: tokens, specific things, and relationships
- Generic models: collections of instances, and relationships
- Metamodels: ontology, and epistemology, of generic models

What is meta-modeling?

<http://en.wikipedia.org/wiki/Meta-modeling>

- ❖ Generally, the analysis, construction and development of the frames, rules, constraints, models and theories applicable and useful for the modeling in a predefined class of problems.
- ❖ In computer science and related disciplines, the construction of a collection of "concepts" (things, terms, etc.) within a certain domain.
- ❖ A model is an abstraction of phenomena in the real world, and a metamodel is yet another abstraction, highlighting properties of the model itself.
- ❖ A model should conform to its metamodel like a program conforms to the grammar of the programming language in which it is written.

*“A valid metamodel is an **ontology**,
but not all ontology are modeled explicitly as **metamodels**” [Söderström2002].*



Meta-model

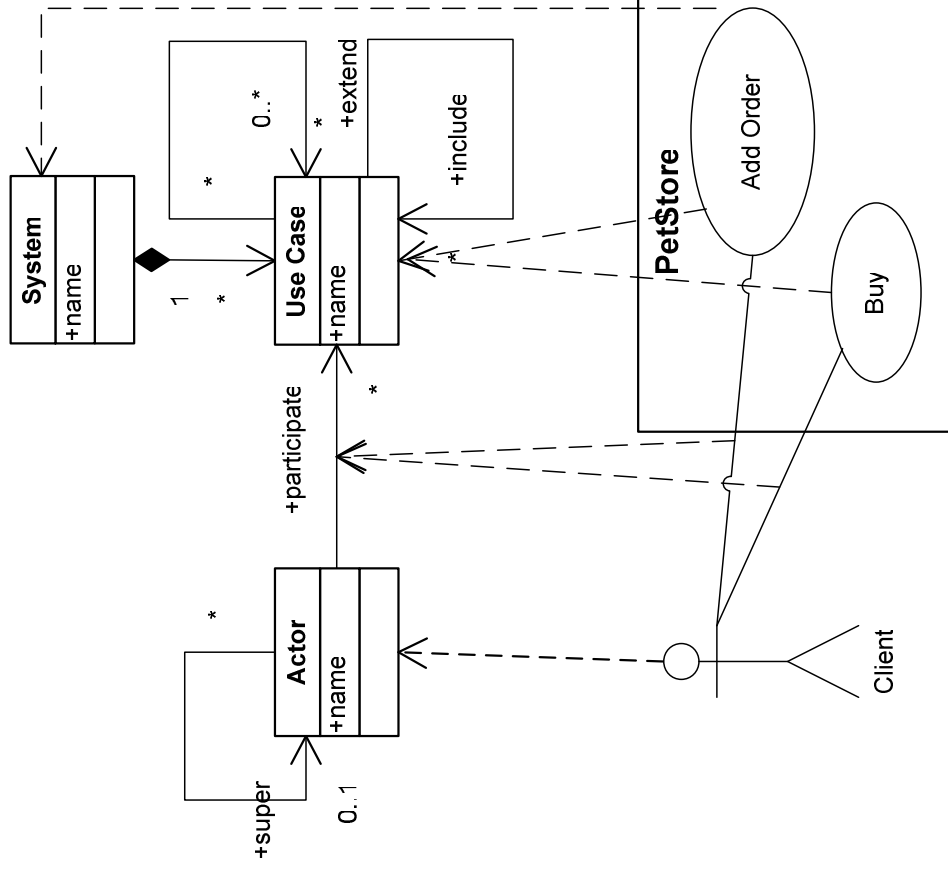
Capitalization

A meta-model defines concepts and their relationships thanks to a class diagram.

A meta-model only defines structure (no semantic).

A model is an instance of a meta-model if it respects the structure defined by the meta-model.

The UML meta-model defines the structure that all UML models must have.



An Example: UML metamodel in the 4 Layer

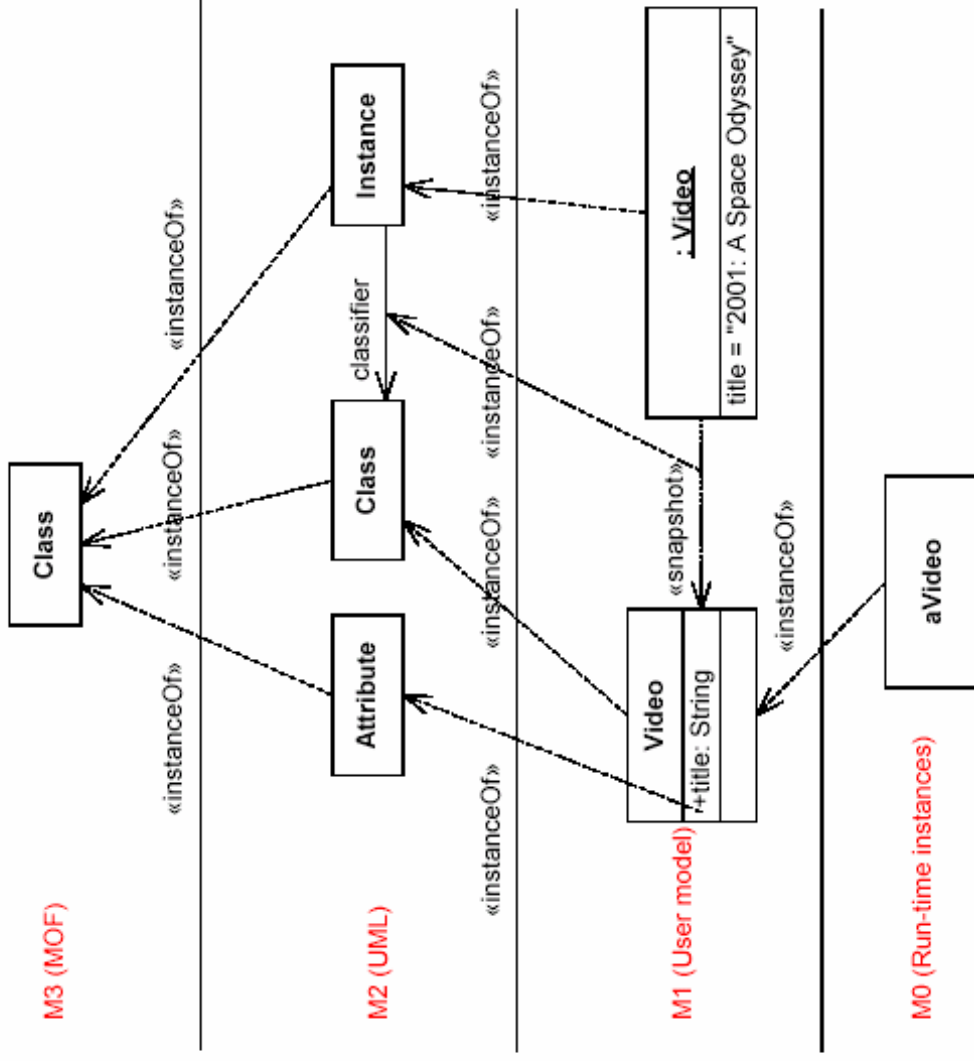
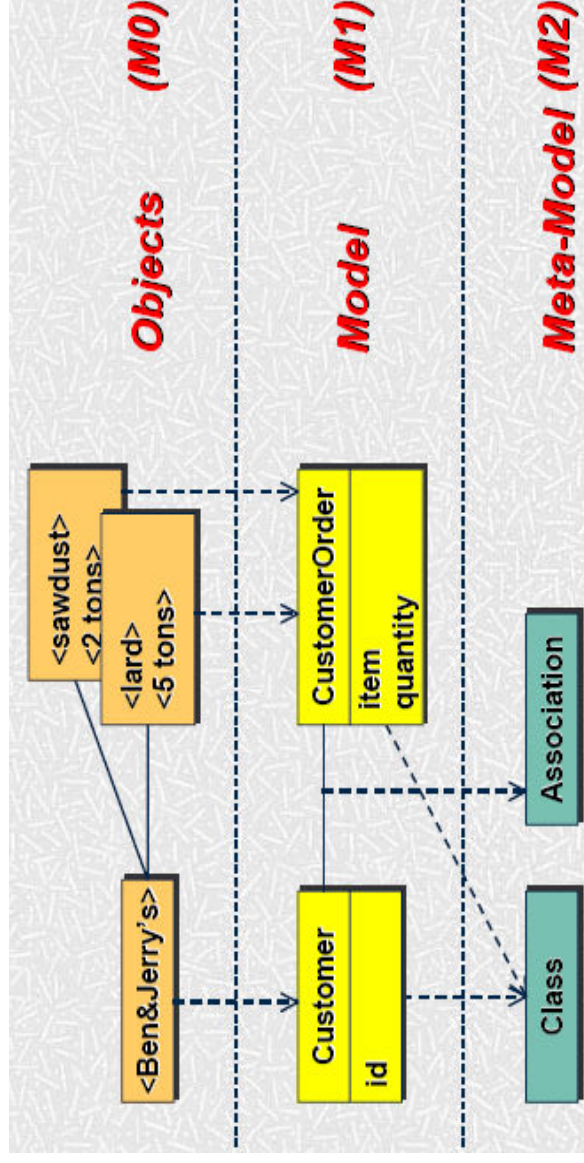


Figure 8 - An example of the four-layer metamodel hierarchy

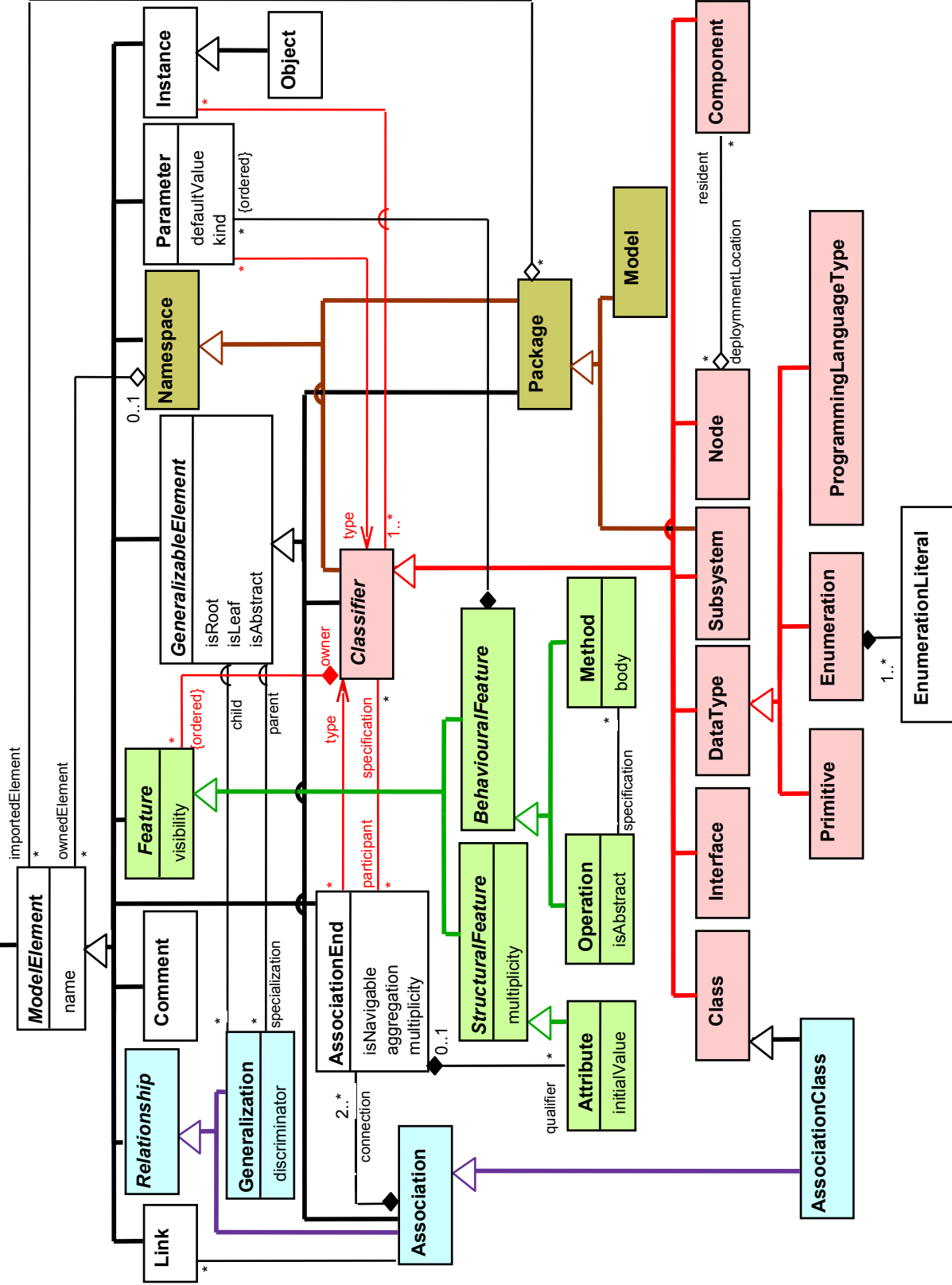
Another Example: UML metamodel in the 4 Layer



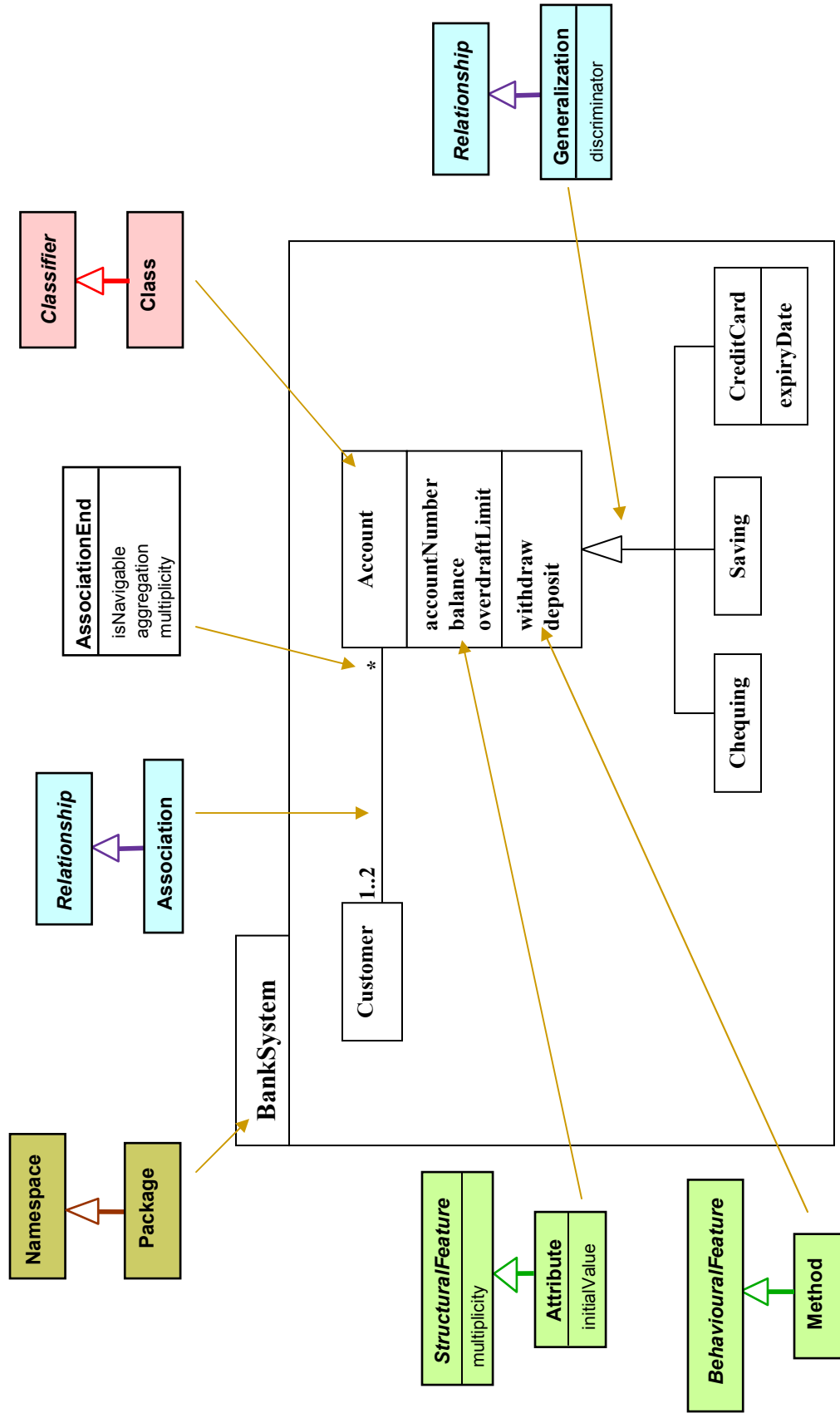
UML Meta Model

Metaclasses used in class, package, component and deployment diagrams

www.site.uottawa.ca/ftppub/courses/Winter/seg3310/coursenotes/Lecture7-8-Metamodeling.ppt

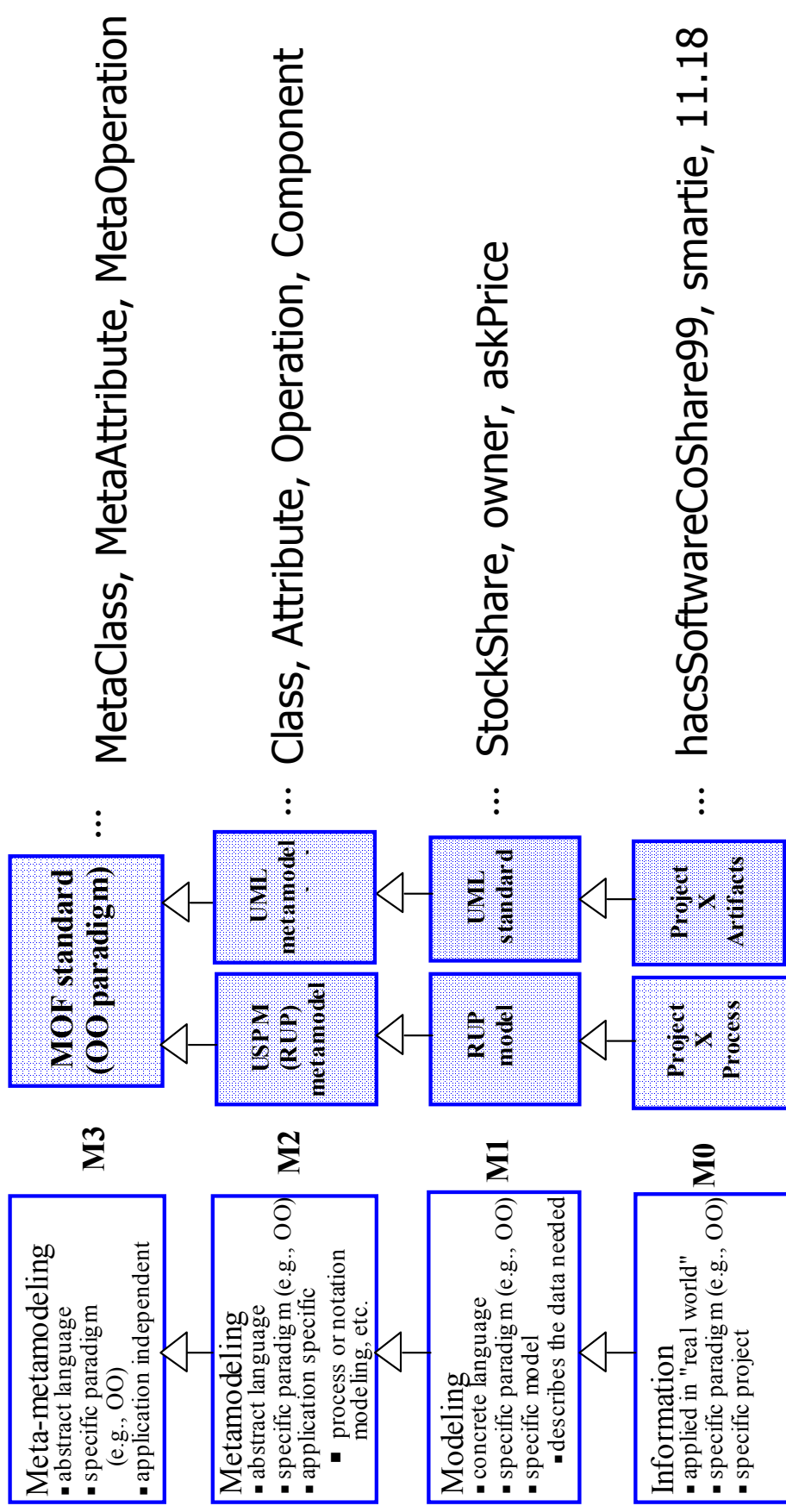


Mapping of UML Models to Metamodel Elements (Example)



Points to Ponder

- Can we create a new version of UML, in consideration of the following?

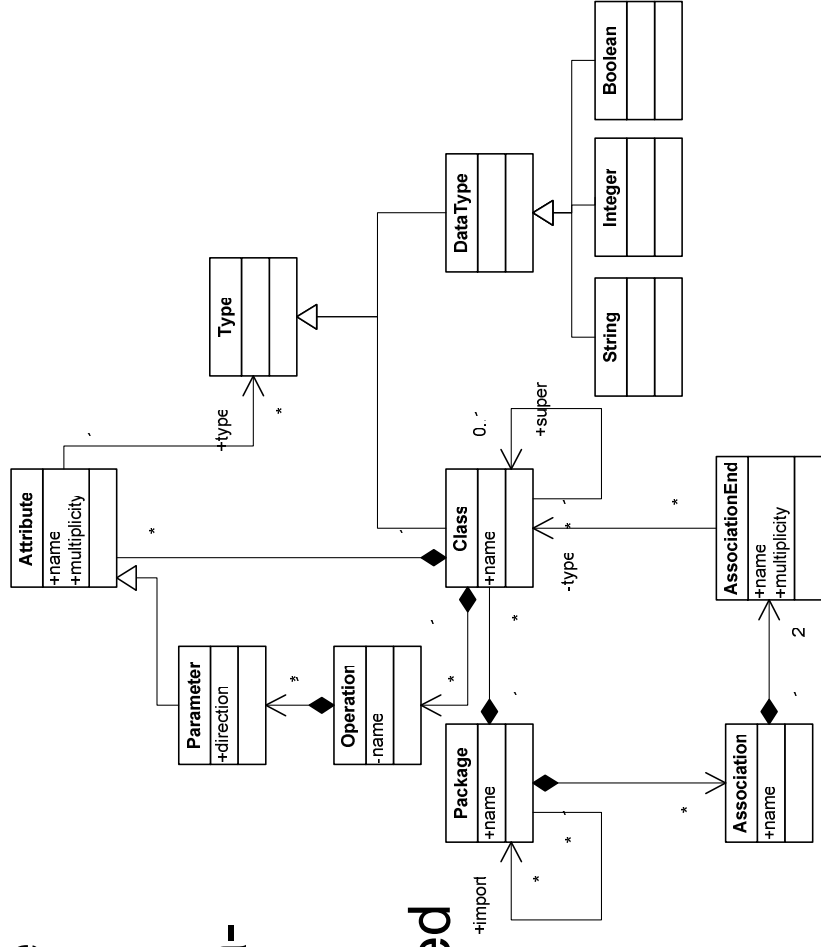


Metamodeling
Architecture

Examples

Meta-meta-model

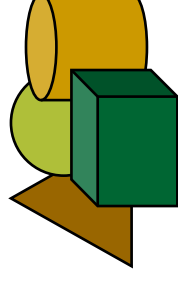
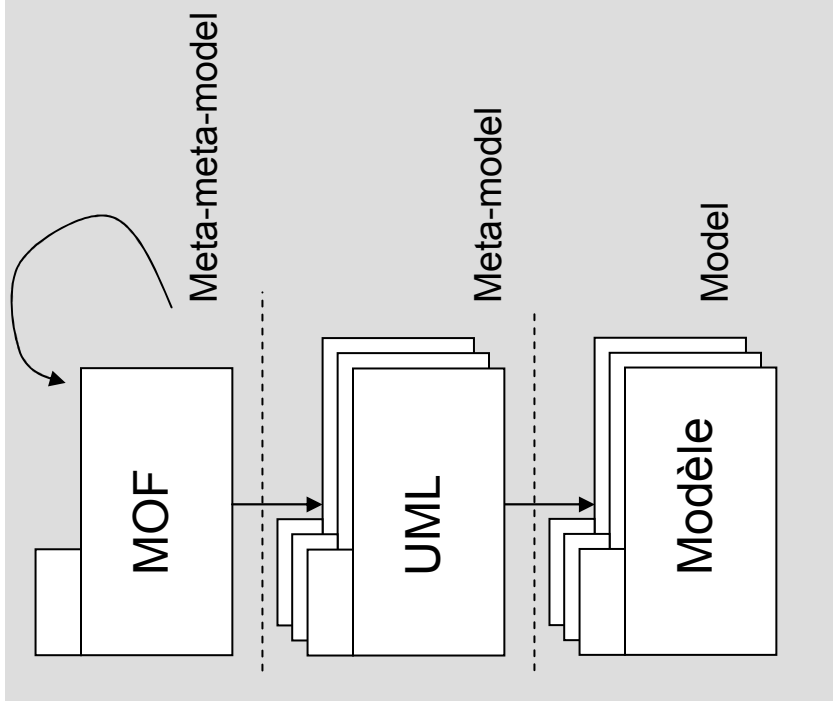
- MOF defines the language for defining meta-models
- MOF concepts are meta-class, meta-attribute, meta-association, etc.
- MOF concepts and their relationships can be defined by a class diagram. This diagram is also a meta-model (called the meta-meta-model)
- The meta-meta-model is self defined.



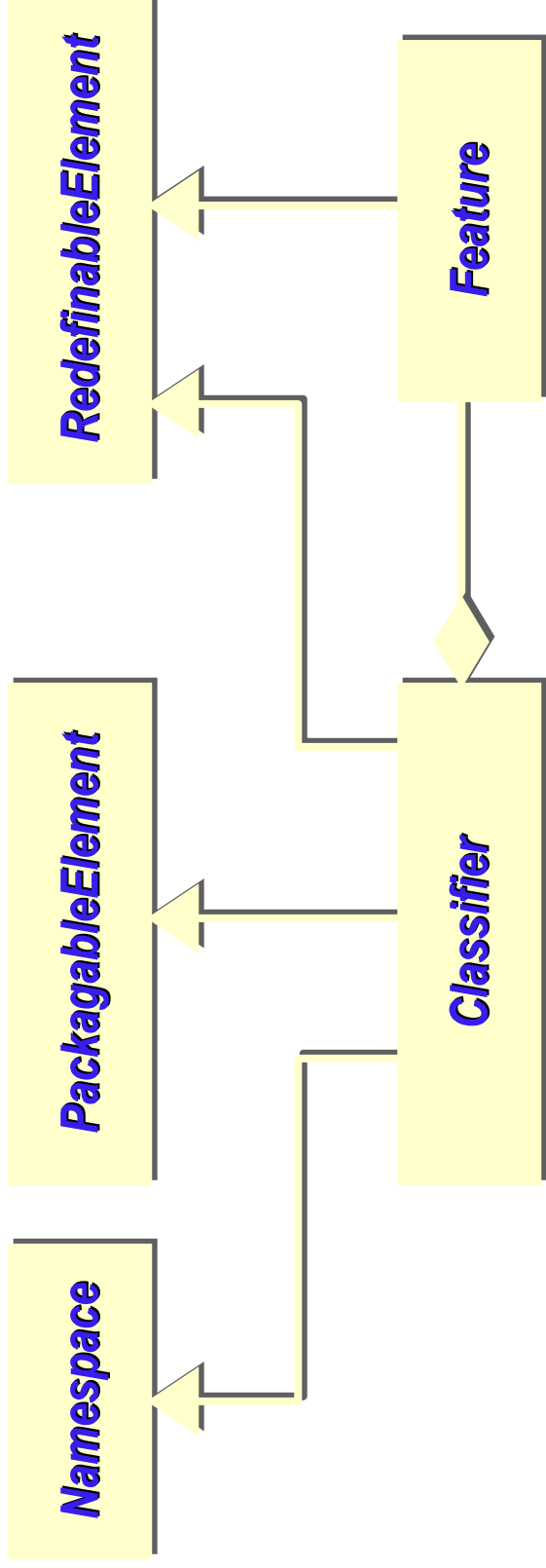
Meta-layers

Capitalization

- M_{n+1} defines the structure of M_n
- M_{n+1} is not an abstraction of M_n
- Meta-layer relationships are similar to grammar-layer relationships (BNF, or XML Schema)

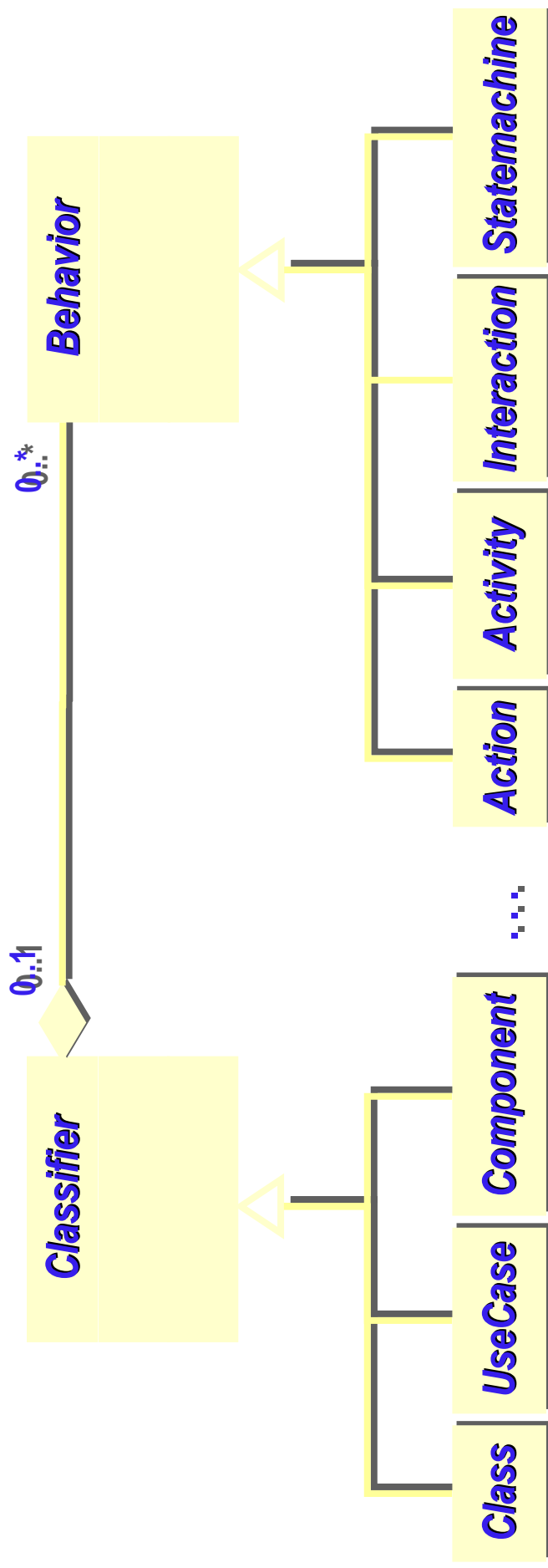


Infrastructure: Consolidation of Concepts

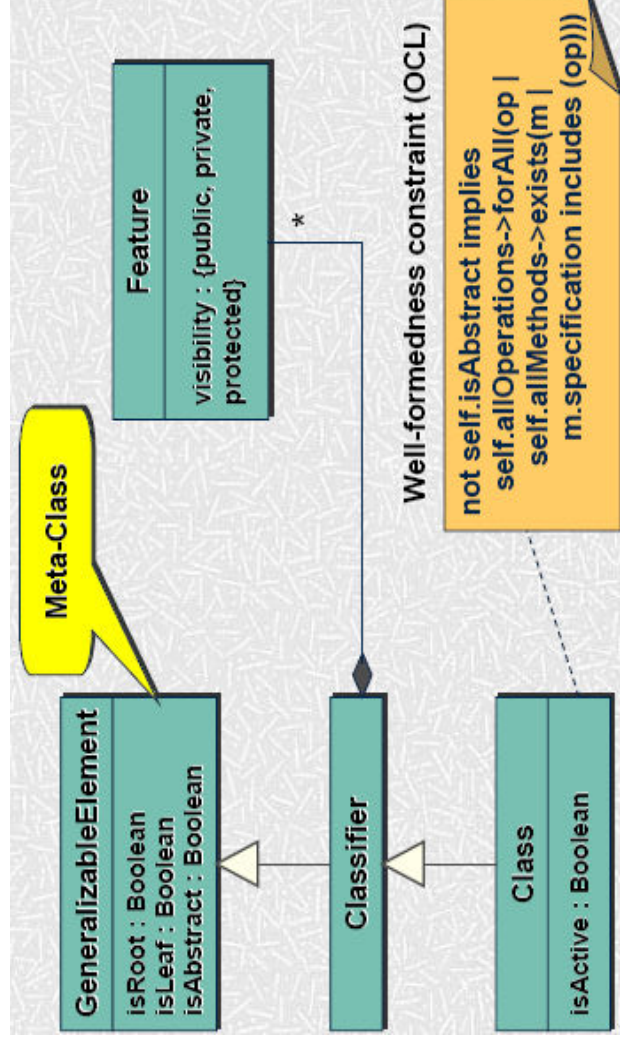


Infrastructure: Behavior Harmonization

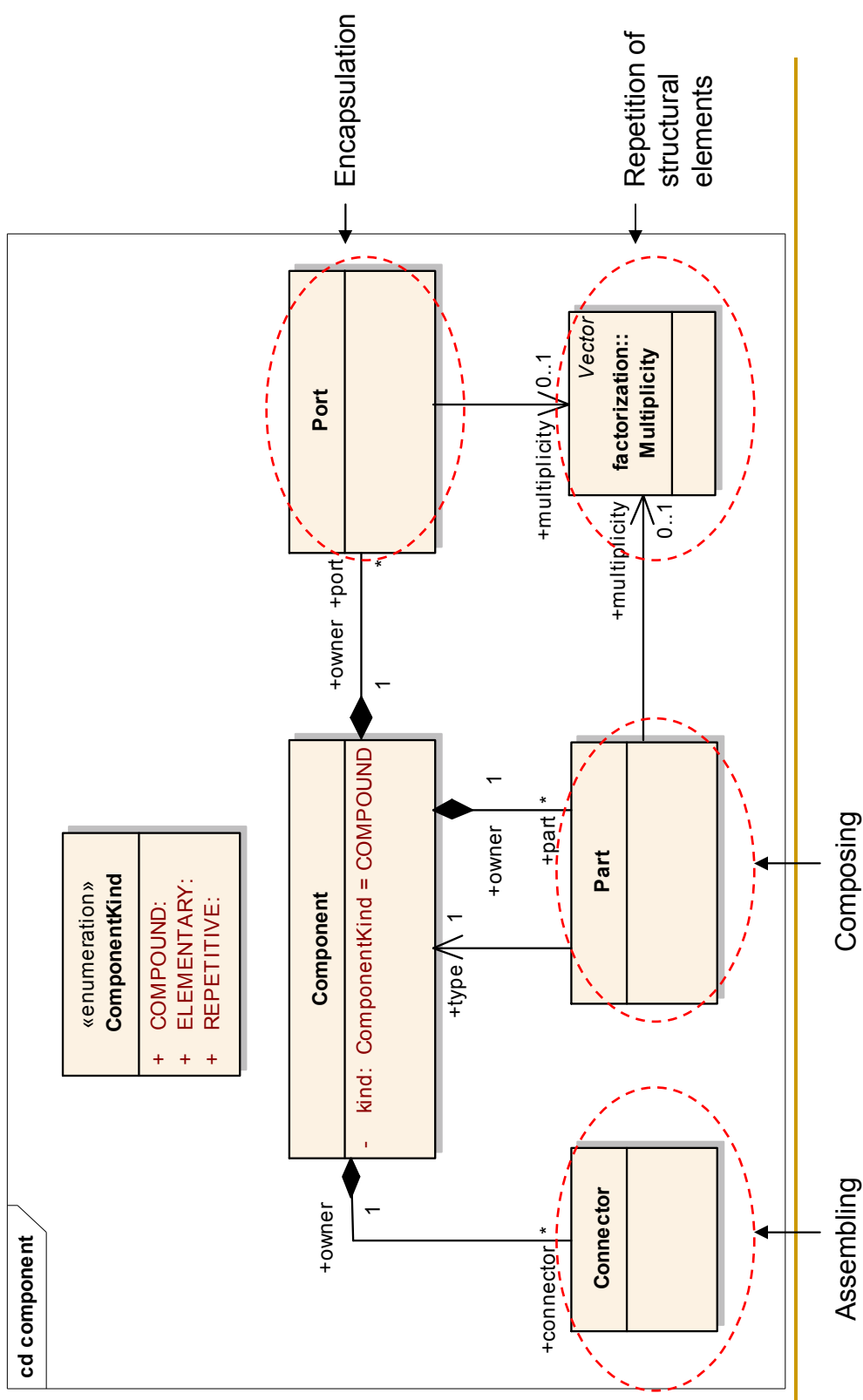
- Common semantic base for all behaviors
 - Choice of behavioral formalism driven by application needs



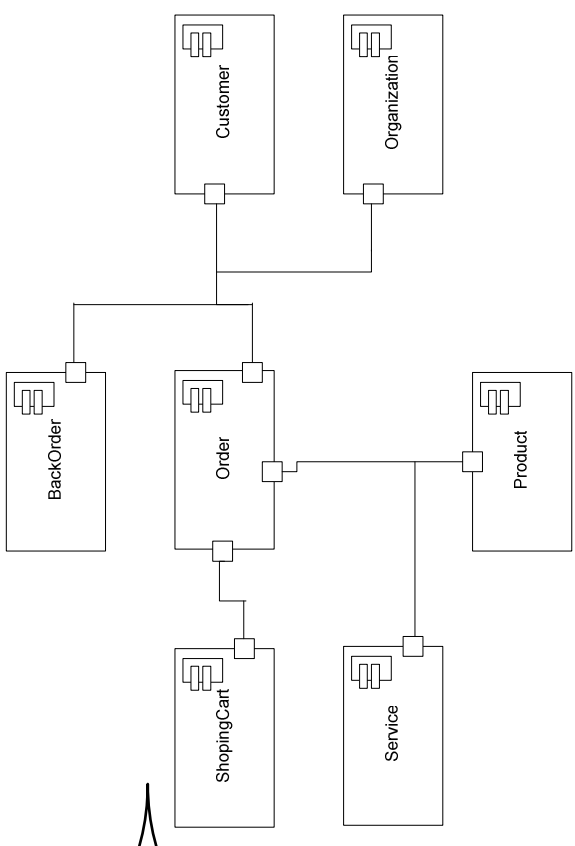
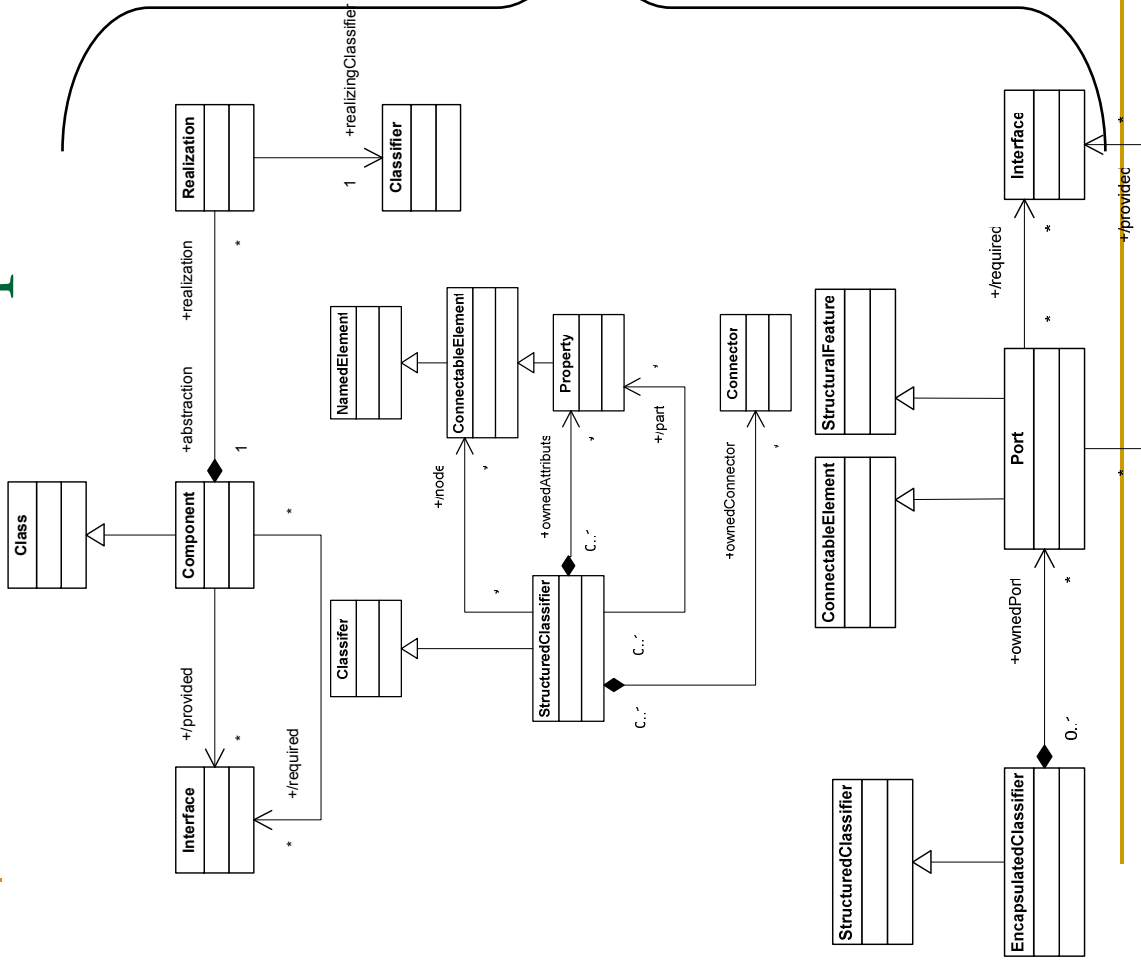
Classifier & Class



Common component paradigm



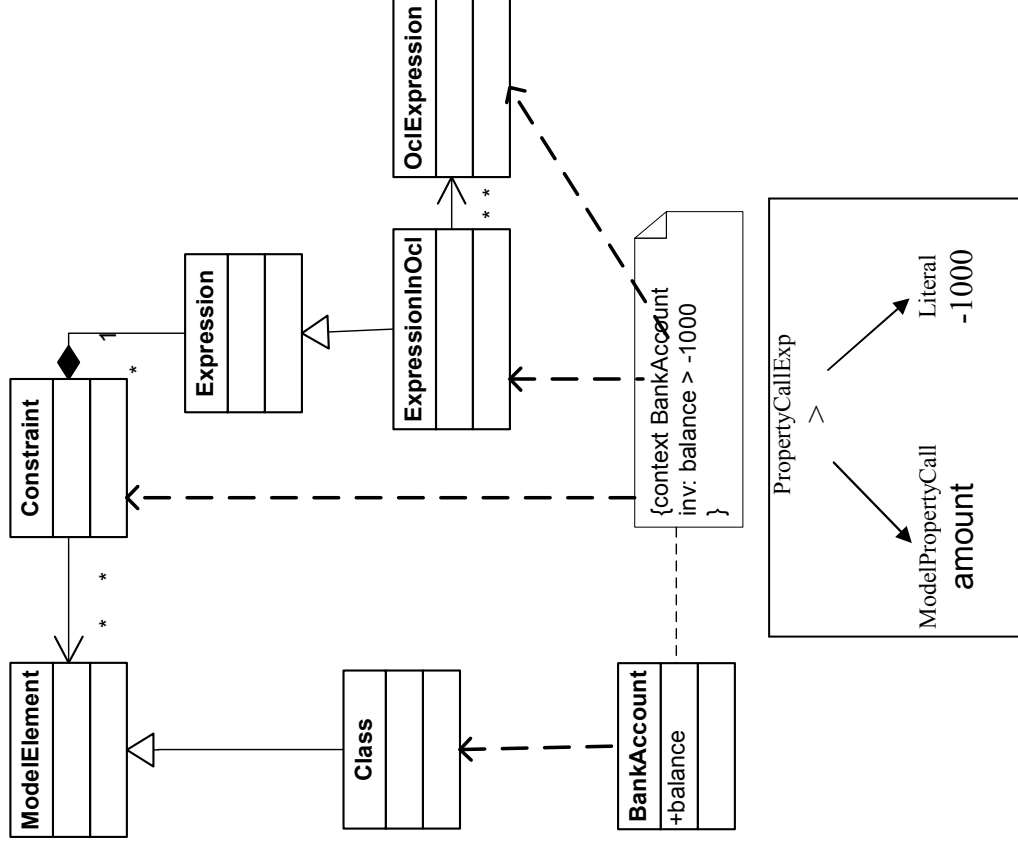
UML2.0 Component



Object Constraint Language

Capitalization

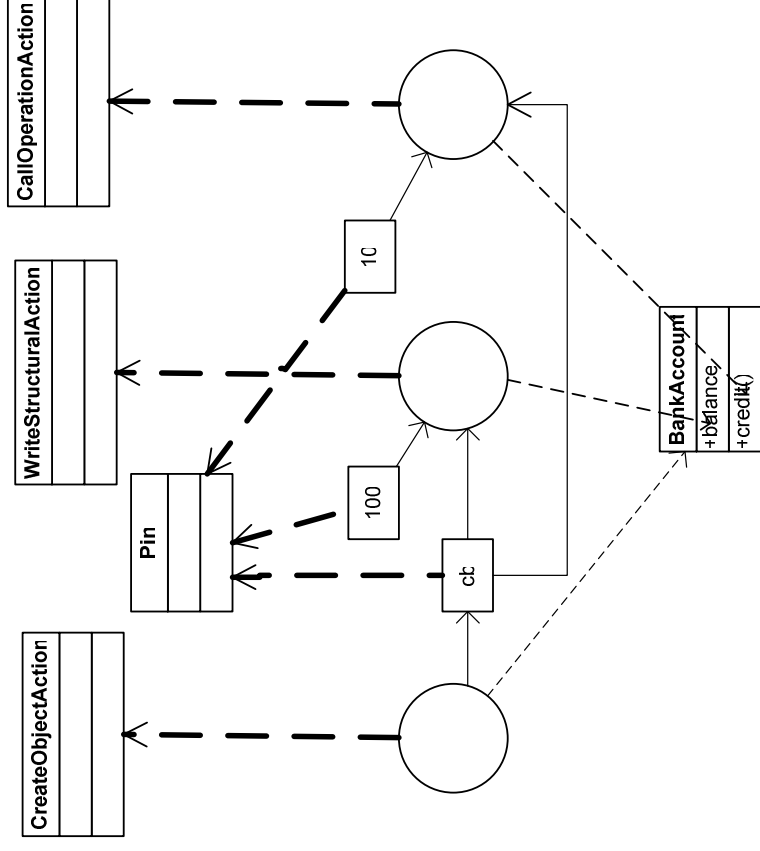
- OCL defines the structure of models expressing constraints
 - Invariant, Pre-post conditions
- OCL is a meta-model instance of the MOF
- OCL is highly coupled with UML
- The OCL semantic is defined with models (operation without side effect)
- OCL defined a concrete syntax



Action Semantics

Capitalization

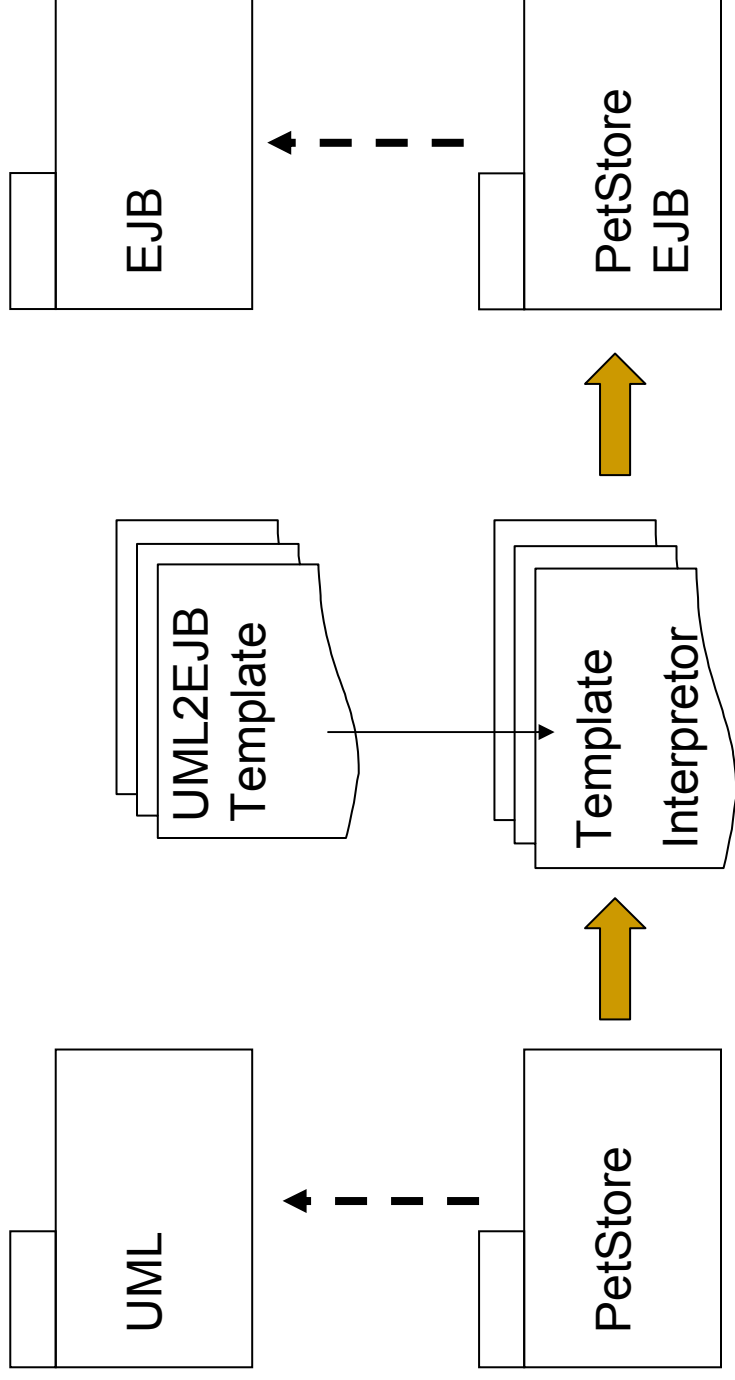
- AS defines the structure of models expressing sequences of actions
- AS was a meta-model and is now completely integrated in UML2.0
- AS has no concrete syntax (UML diagram)
- The semantic of AS is not formally defined (an RFP is published)

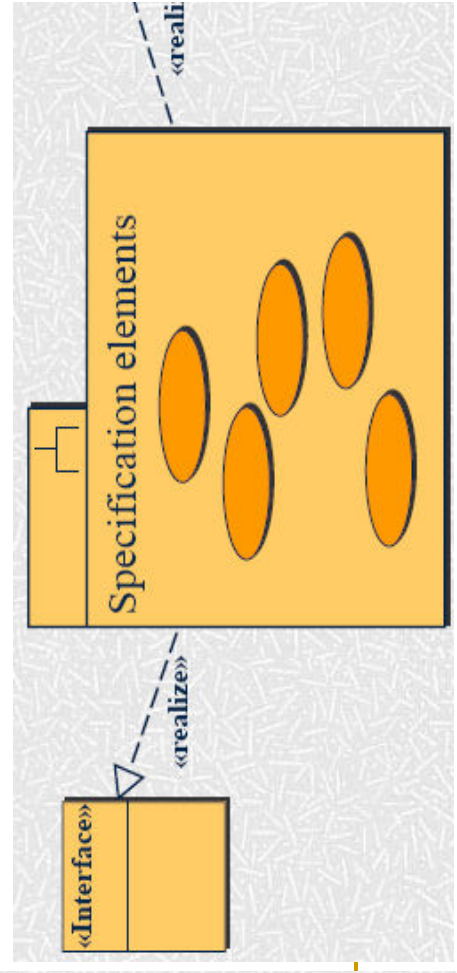
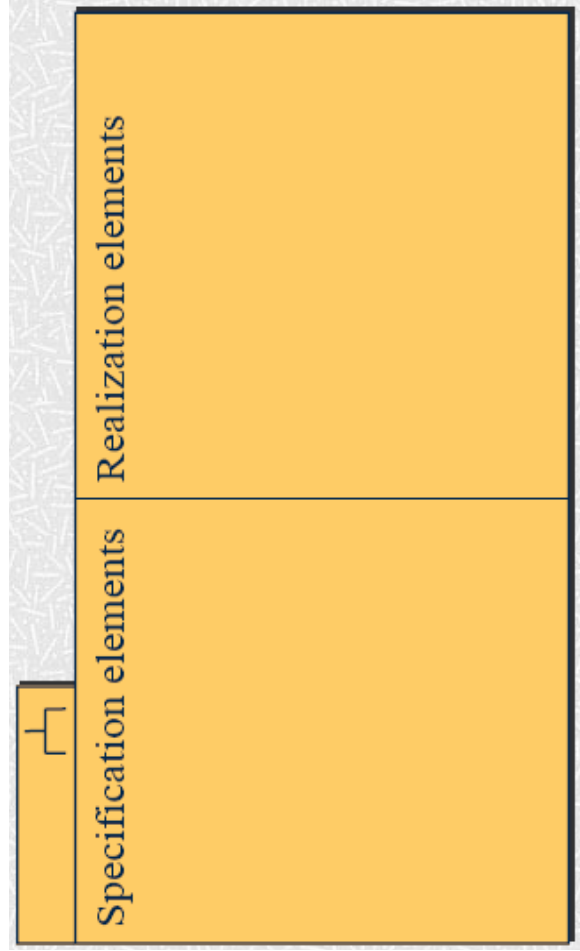
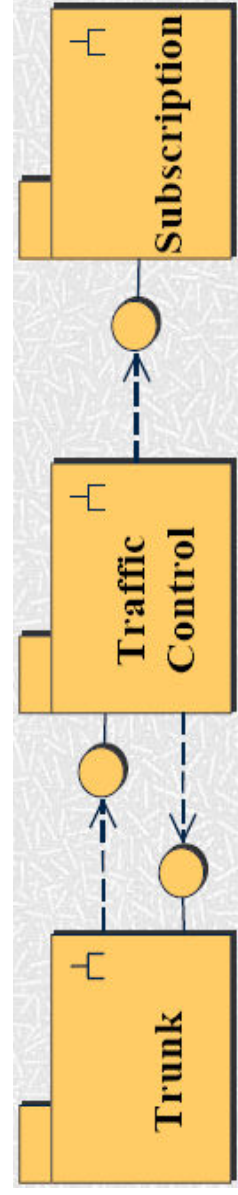
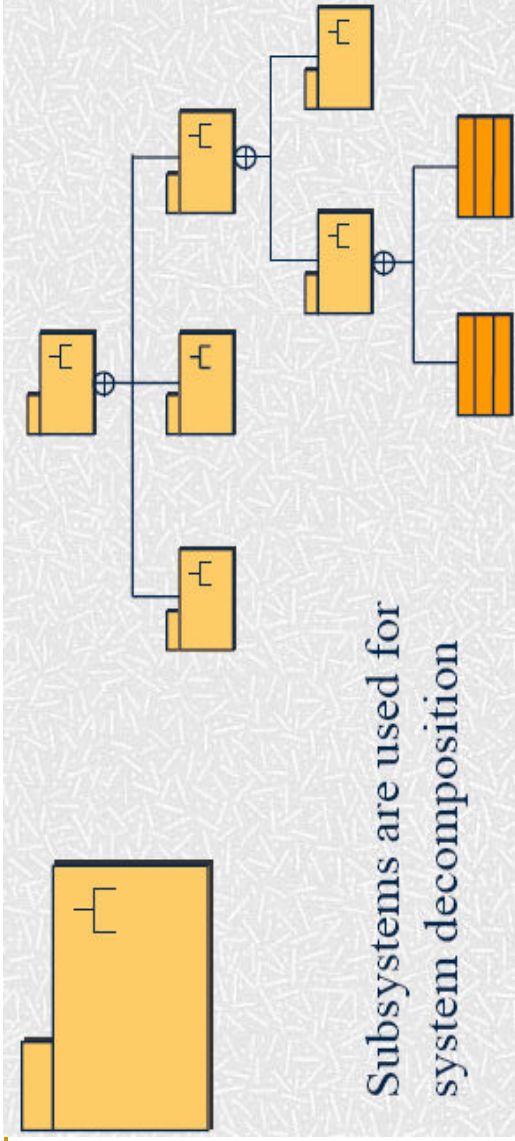


Template

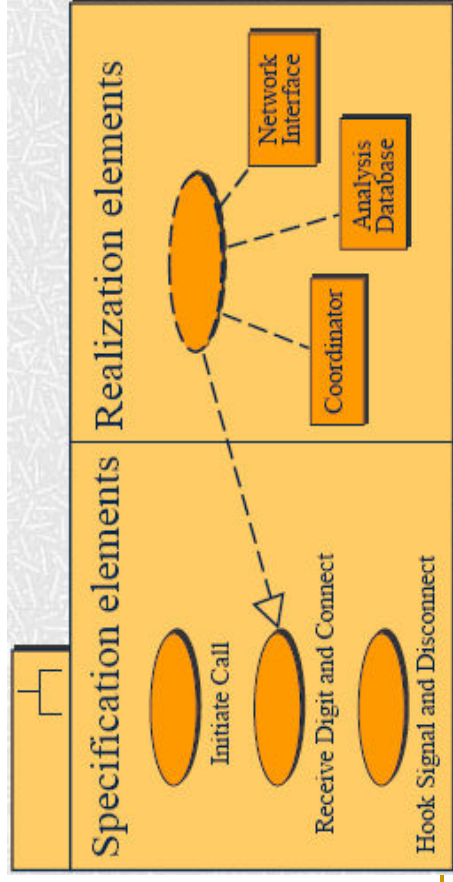
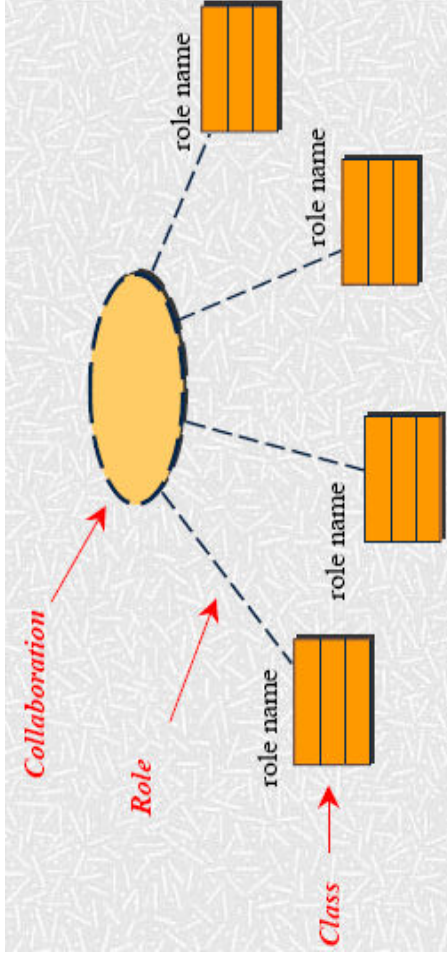
Productivity

- Model transformation is a template written in a dedicated language

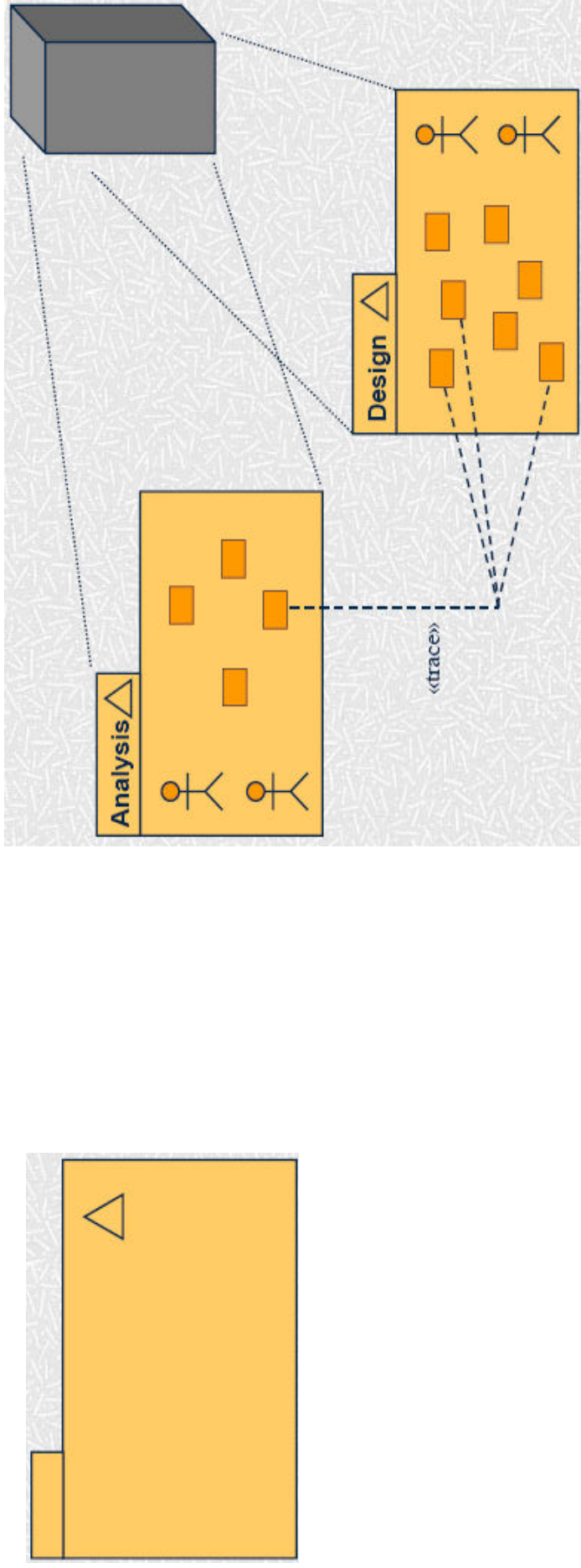




Collaboration



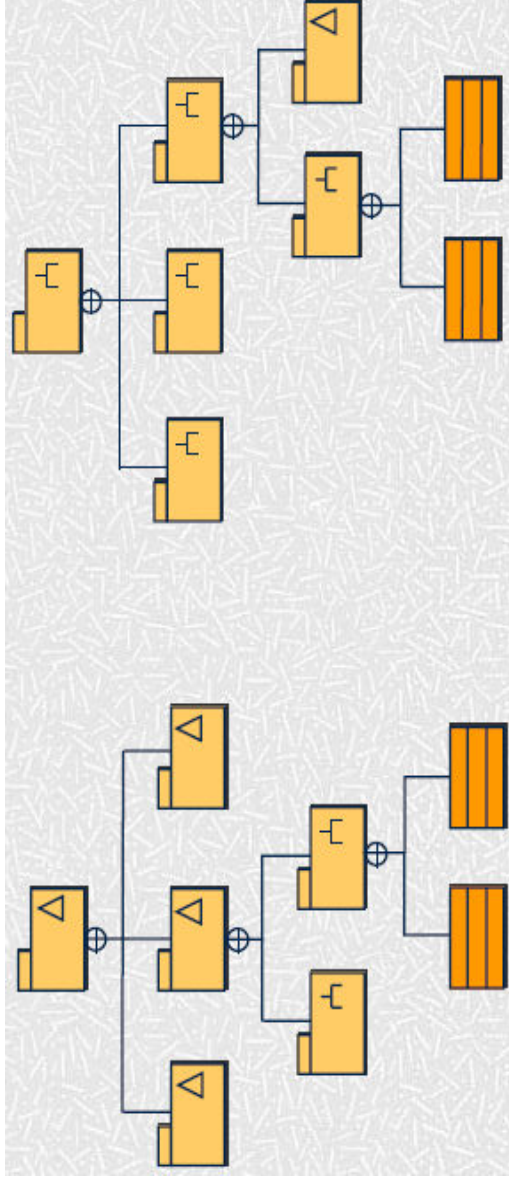
Models & Traceability



A model is an abstraction of a system, specifying the system from a certain viewpoint and at a certain level of abstraction

Model & System Hierarchy

- Models and subsystems can be combined in a hierarchy:



Stereotype Notation

Stereotype Notation

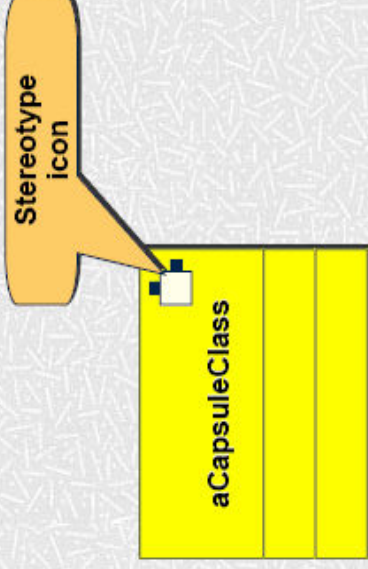
- Several choices



(a) with guillemets



(c) iconified form



(b) with icon