

Metamodeling

- ❖ Why metamodels?
- ❖ Models and Metamodels
- ❖ Examples of metamodels

Questions

- ❖ How can we tell if our specification is correct, syntactically and semantically, wrt. the particular notation being used?
- ❖ How do we define the syntax and semantics of a notation?
- ❖ How do we compare the expressive power of different notations?
- ❖ How do we extend a notation in a consistent manner?

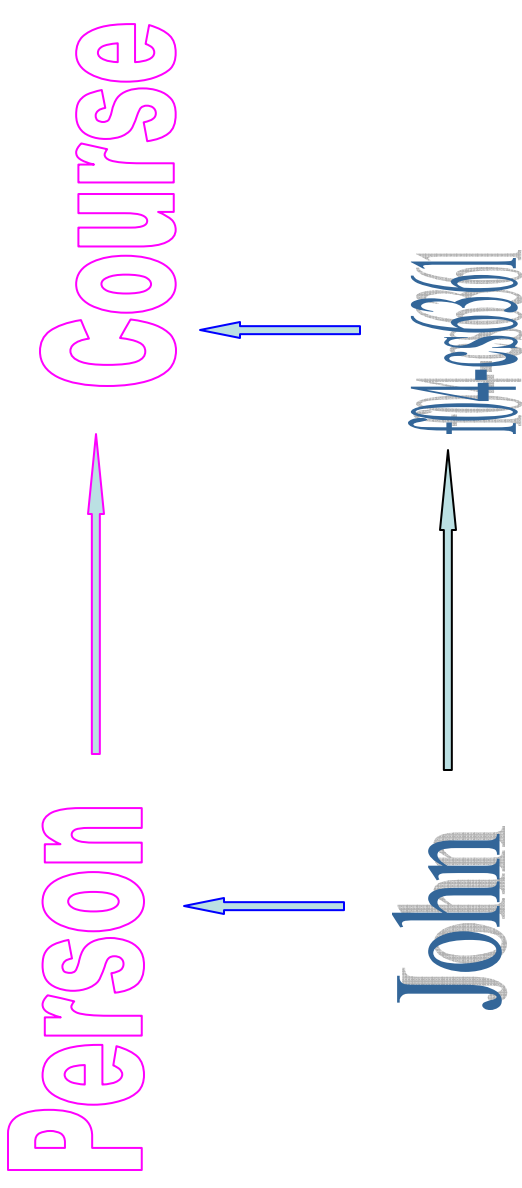
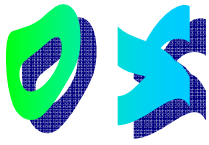
What is meta-modeling?

<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].

Review: Instances and Classes



Instances: tokens, specific things
Classes: collections of instances

Review: Instances and Classes

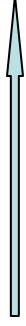


Why?

Person



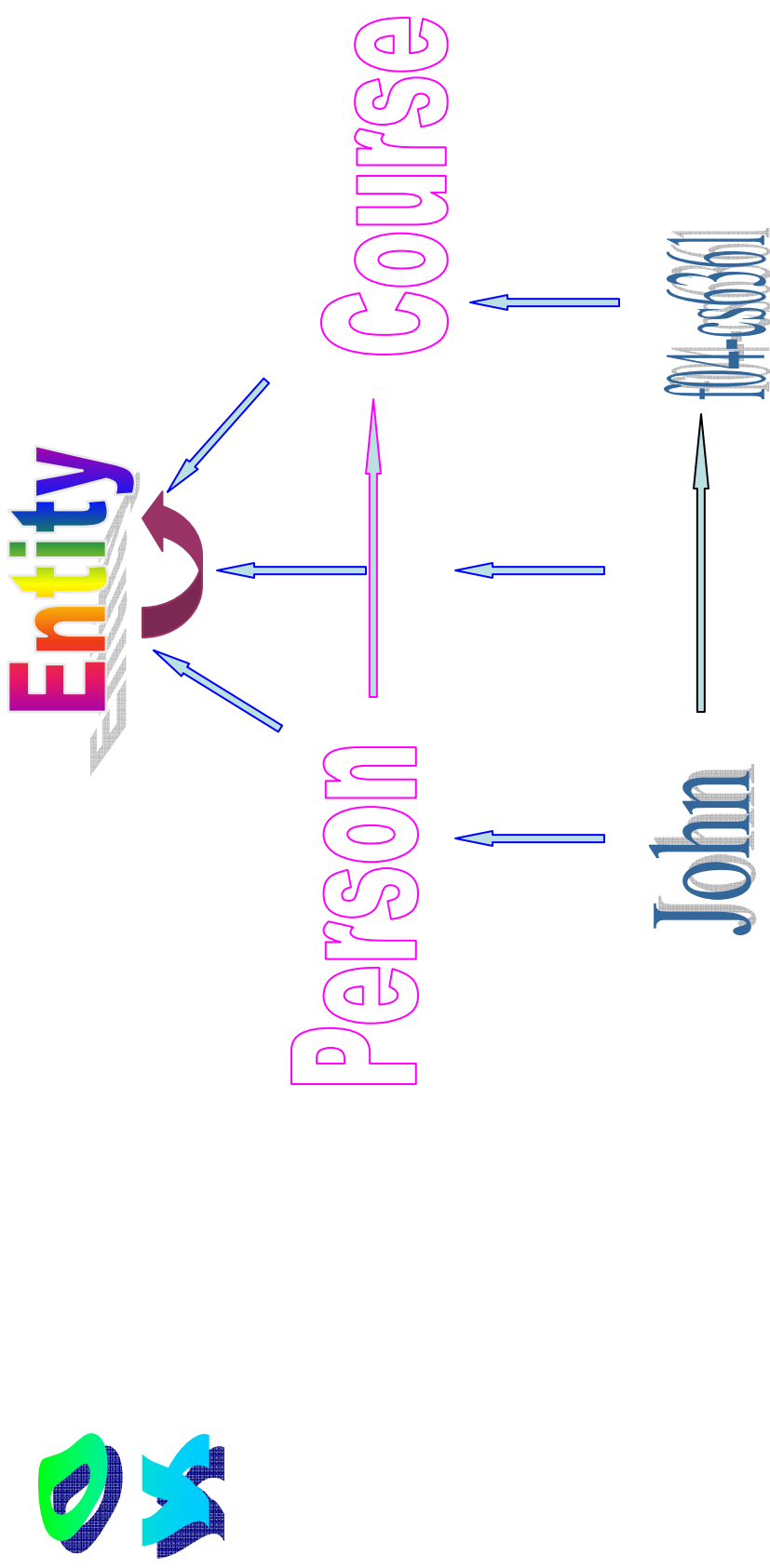
John



104-cs6361

Instances: tokens, specific things
Classes: collections of instances

Models and Metamodels



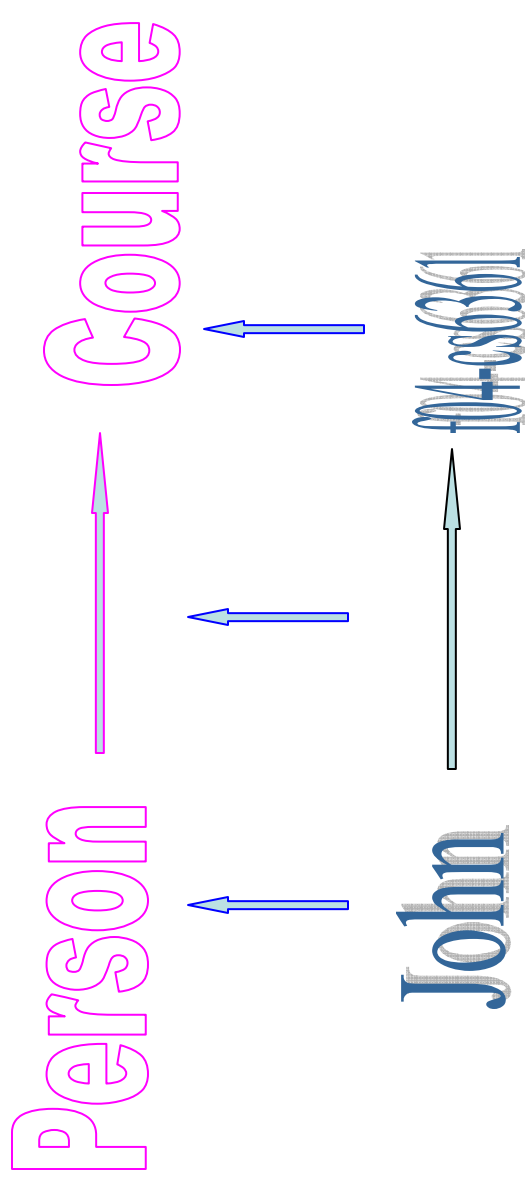
- Specific models: tokens, specific things, and relationships
- Generic models: collections of instances, and relationships
- Metamodels: ontology, ...and epistemology, of generic models

Lawrence Chung

Models and Metamodels



Why?



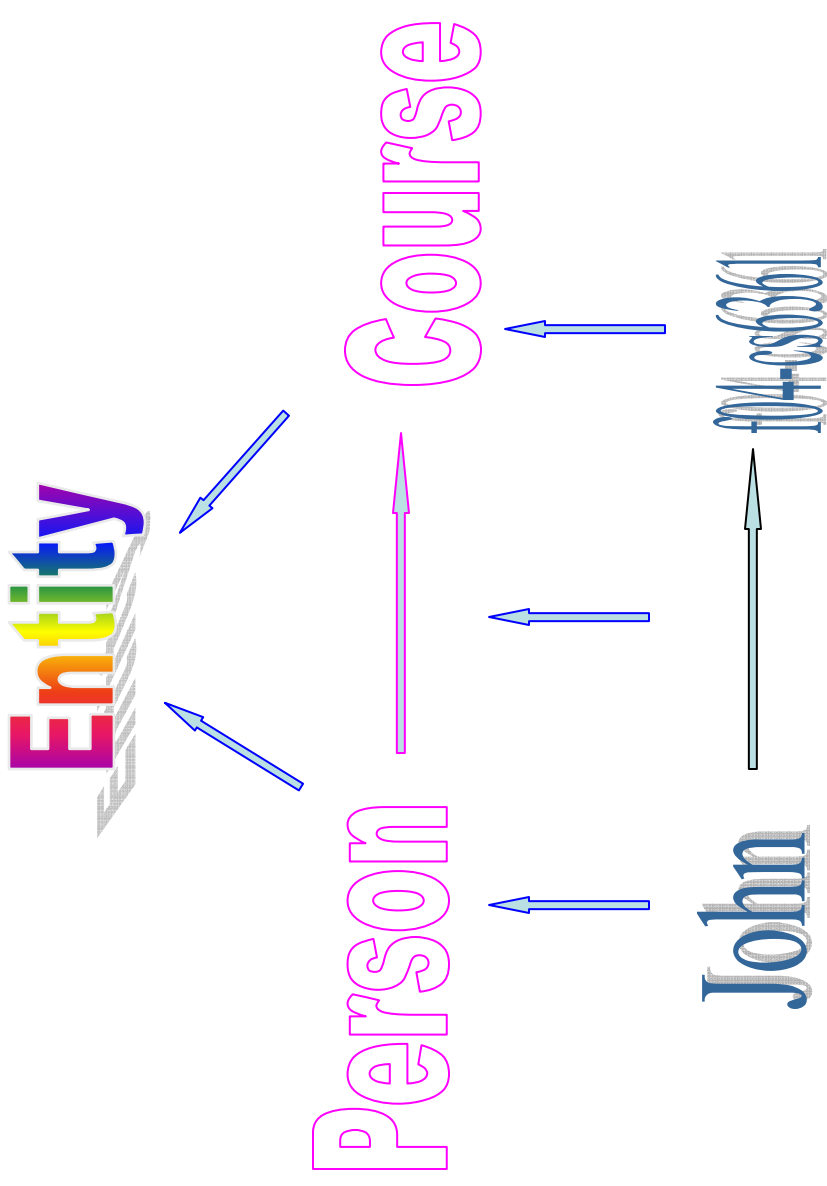
- Specific models: tokens, specific things, and relationships
- Generic models: collections of instances, and relationships
- Metamodels: ontology, ...and epistemology, of generic models

Lawrence Chung

Models and Metamodels



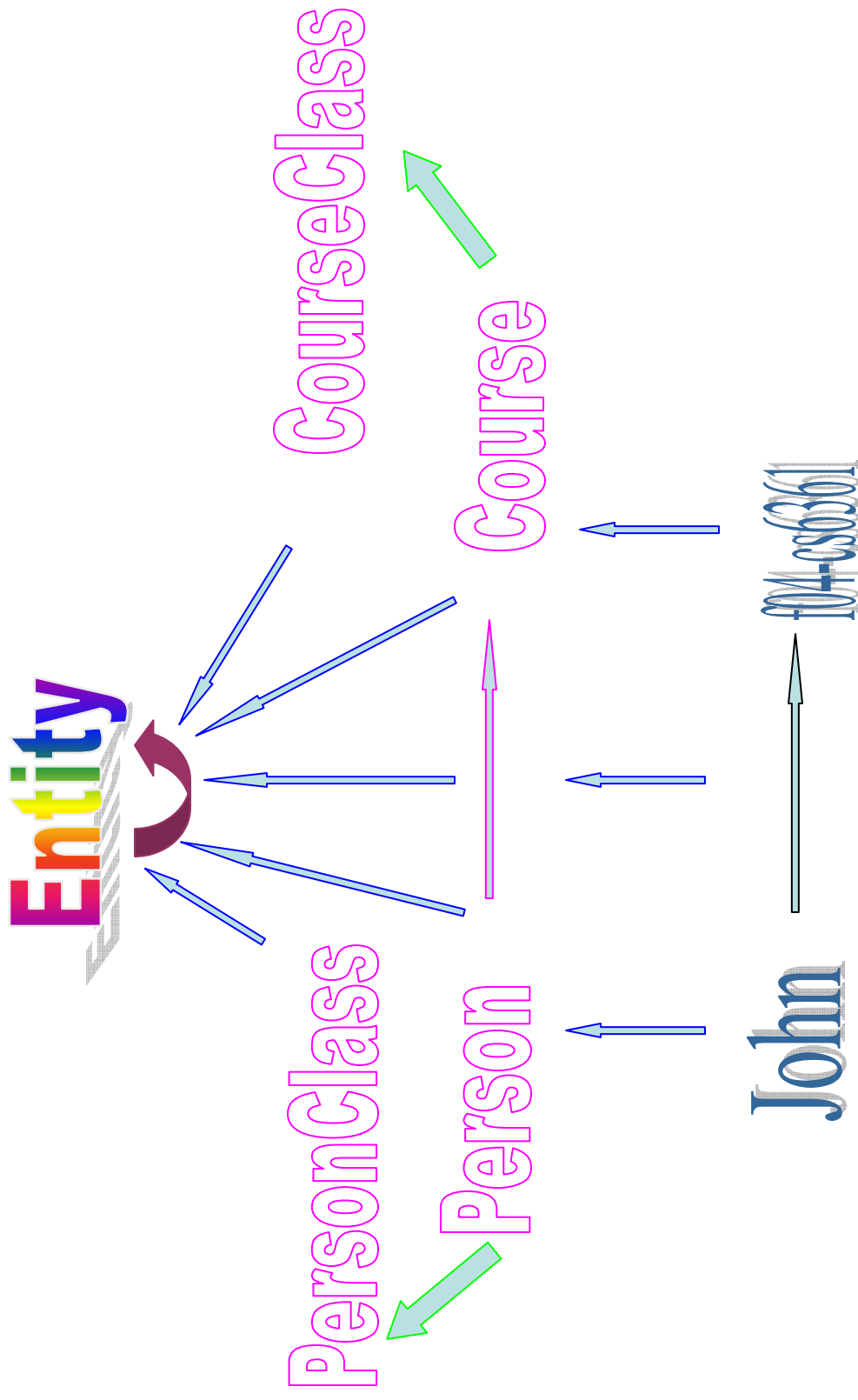
Why?



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

Lawrence Chung

Models, Metamodels & Metaclasses



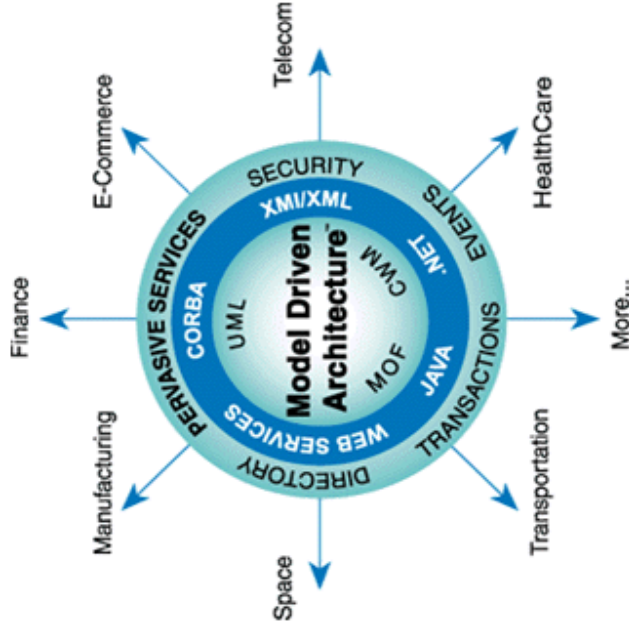
"The Architecture of Choice for a Changing World®"

OMG Model Driven Architecture

<http://www.omg.org/mda/>

How Systems Will Be Built

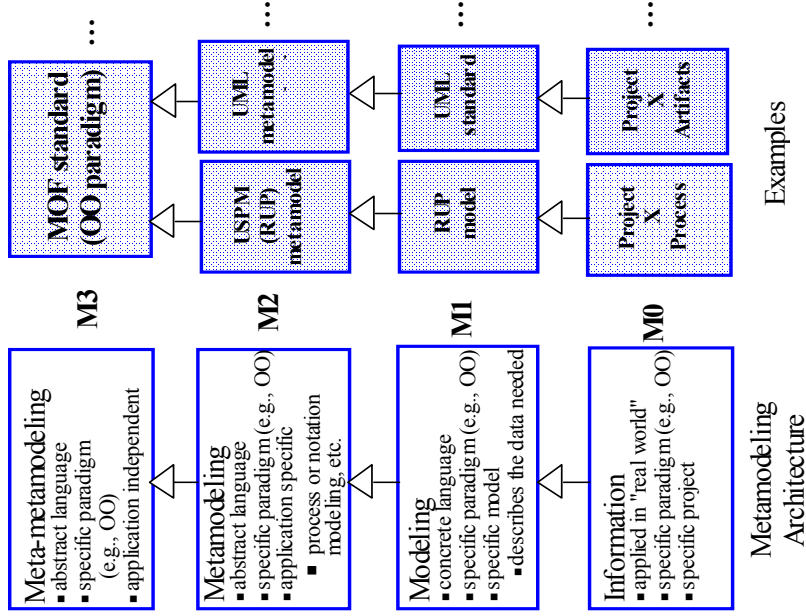
MDA[®] provides an open, vendor-neutral approach to the challenge of business and technology change. Based firmly upon OMG's established standards*, MDA aims to separate business or application logic from underlying platform technology. Platform-independent applications built using MDA and associated standards can be realized on a range of open and proprietary platforms, including CORBA[®], J2EE, .NET, and Web Services or other Web-based platforms. Fully-specified platform-independent models (including behavior) can enable intellectual property to move away from technology-specific code, helping to insulate business applications from technology evolution, and further enable interoperability. In addition, business applications, freed from technology specifics, will be more able to evolve at the different pace of business evolution.



Key standards that make up the MDA suite of standards include Unified Modeling Language (UML); Meta-Object Facility (MOF); XML Meta-Data Interchange (XMI); and Common Warehouse Meta-model (CWM).

OMG Four Layer Metadata Architecture

[Cooper and Chung]



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

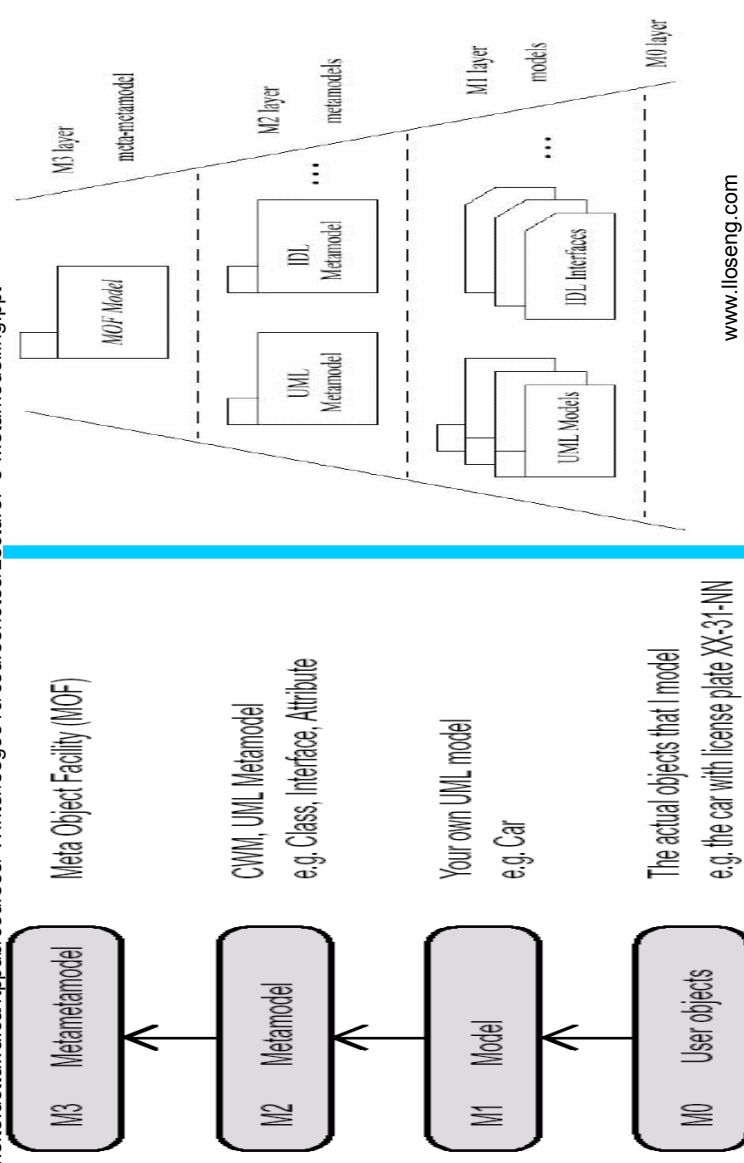


Figure 1 OMG 4-layer architecture

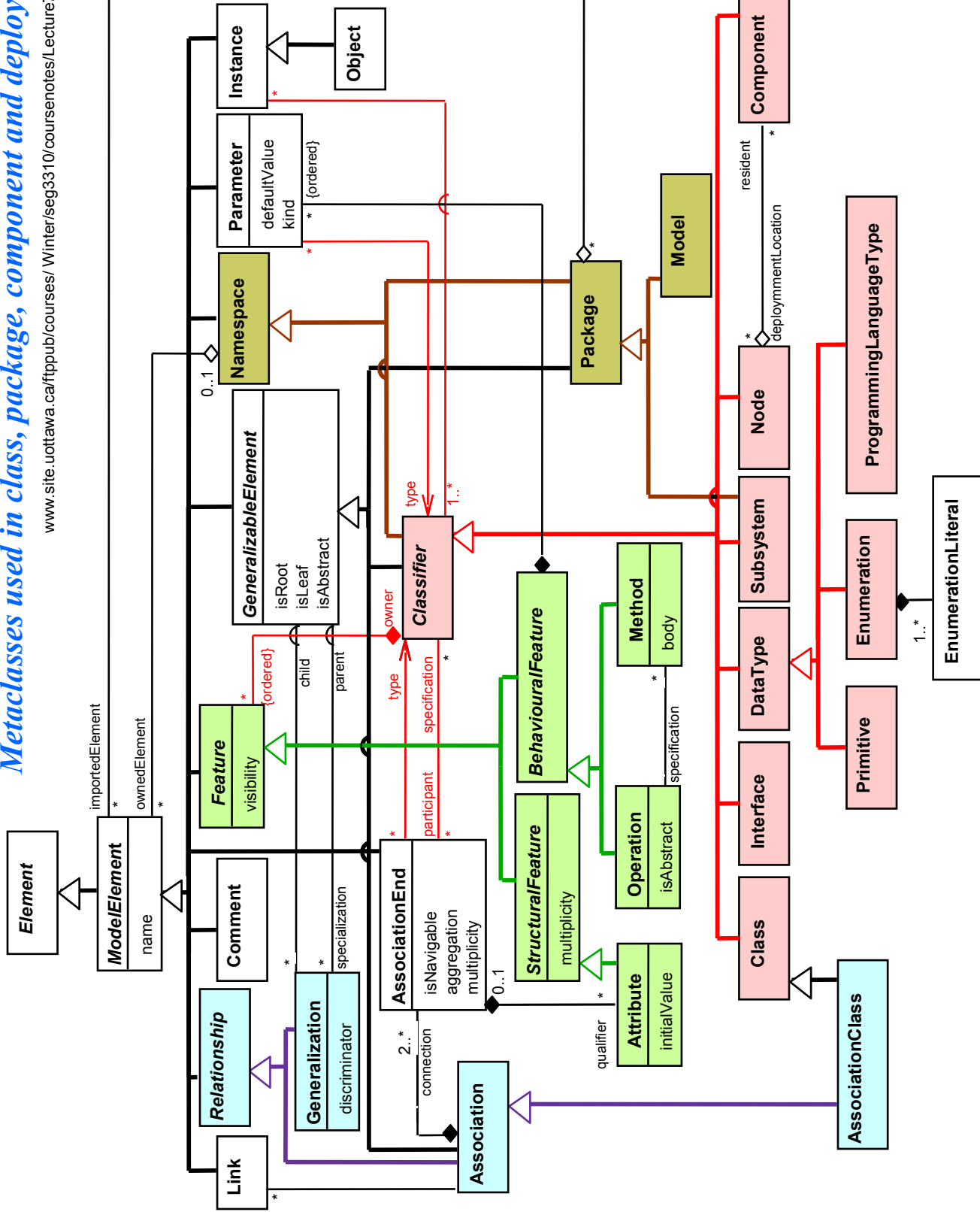
UML Metamodel Diagrams

http://www.utdallas.edu/~chung/OOAD_SUMMER04/UML11_Metamodel_Diagrams.pdf

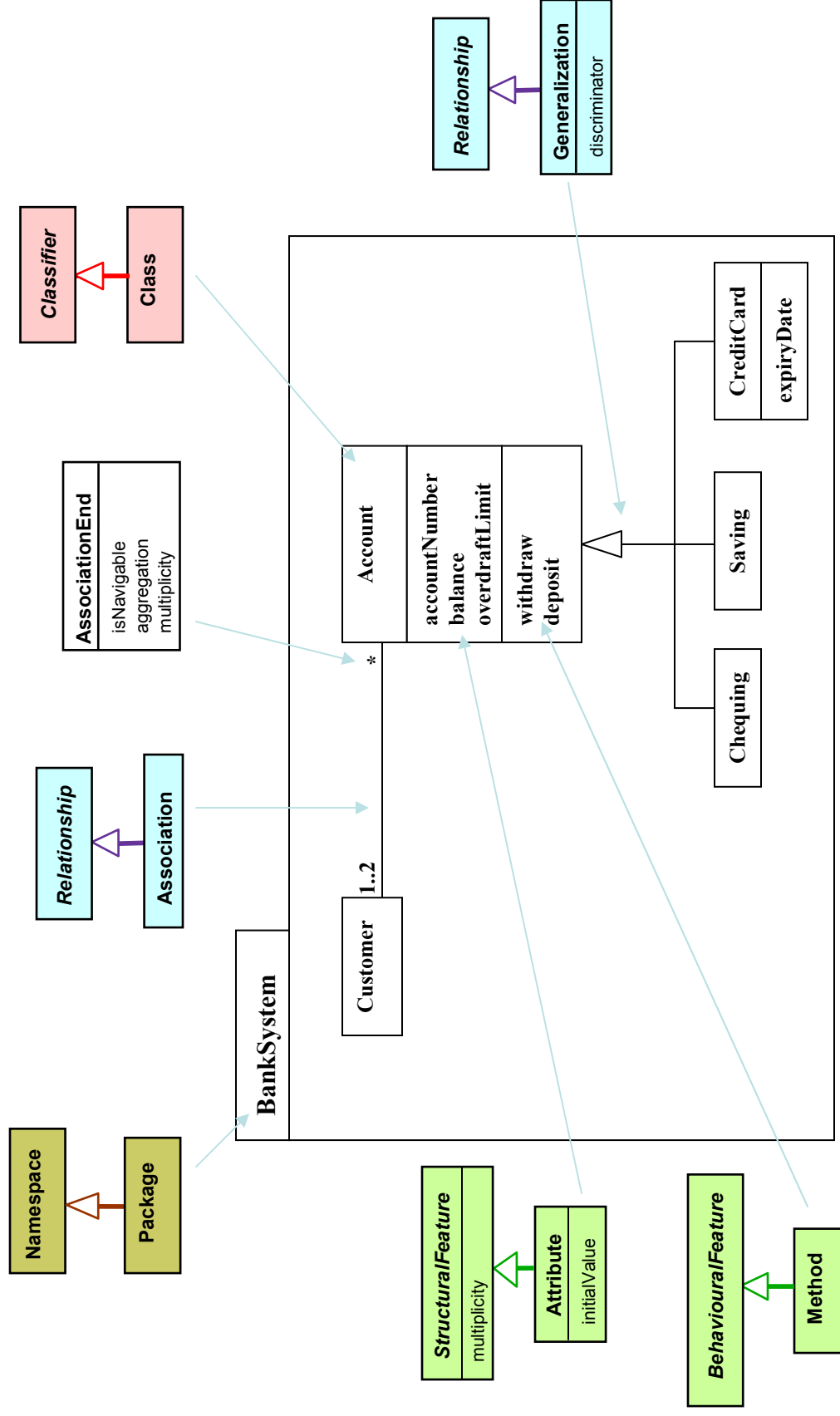
UML Meta Model

Metaclasses used in class, package, component and deployment diagrams

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

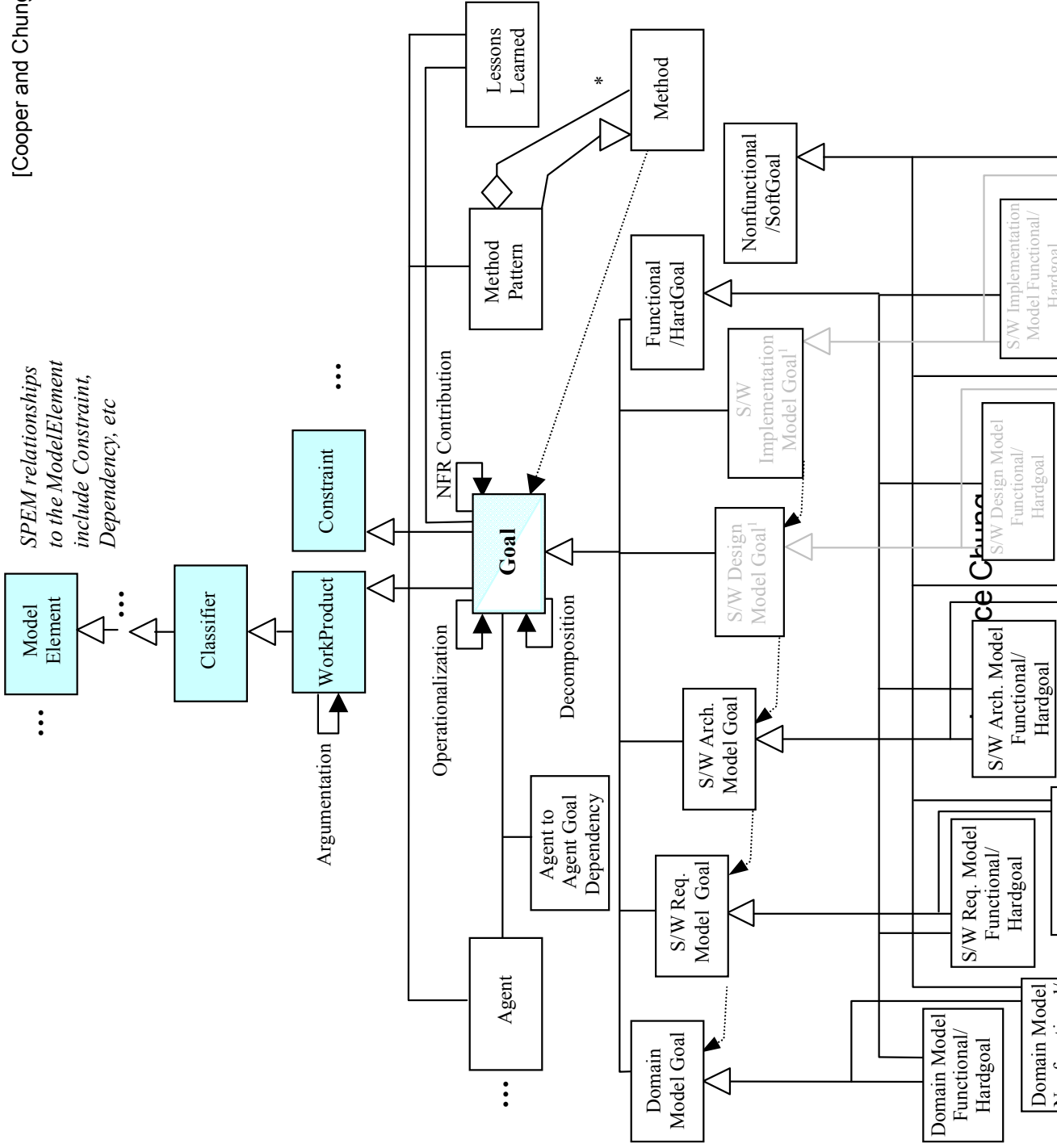


Mapping of UML Models to Metamodel Elements (Example)



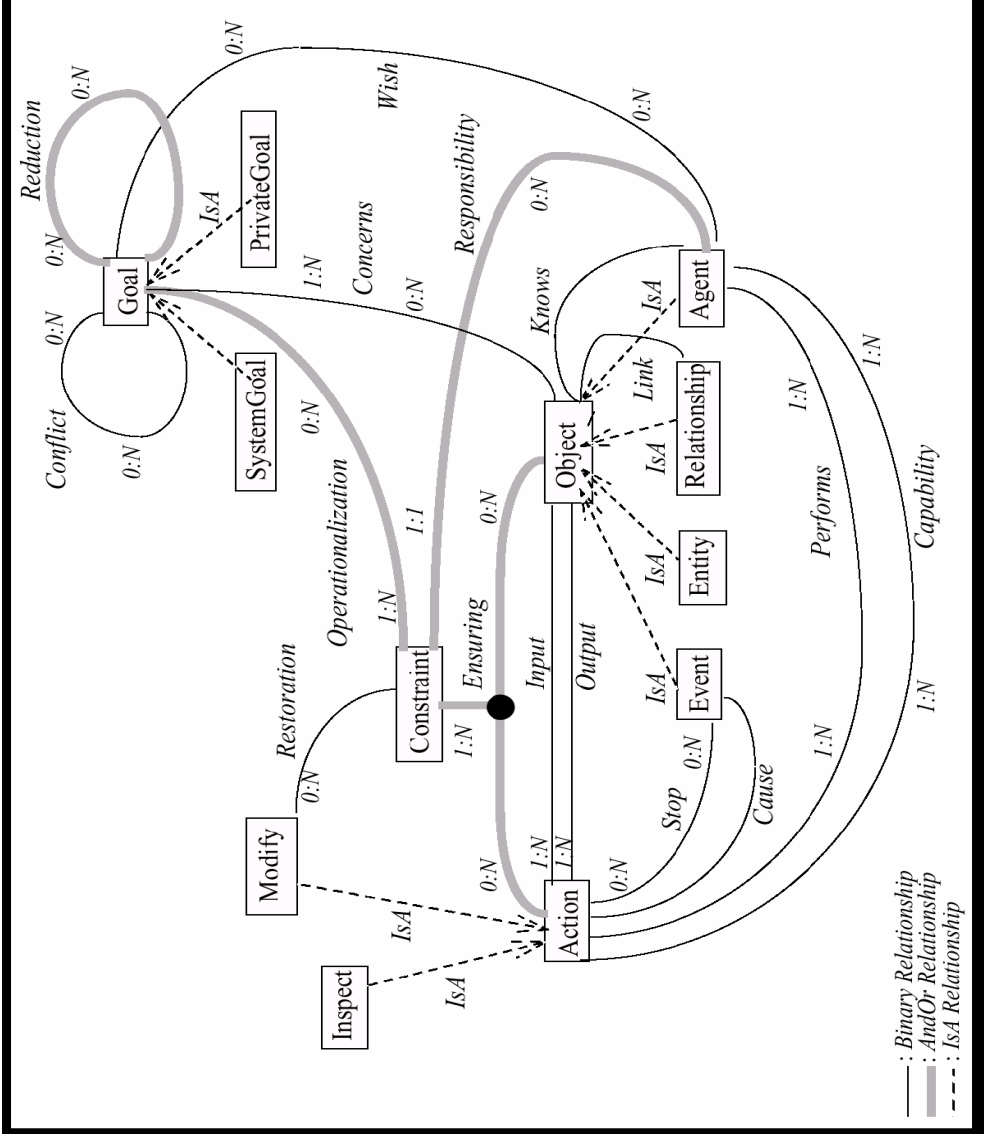
Extending UML Metamodel Diagrams

[Cooper and Chung]



KAOS Metamodel

[See module on requirements elicitation]

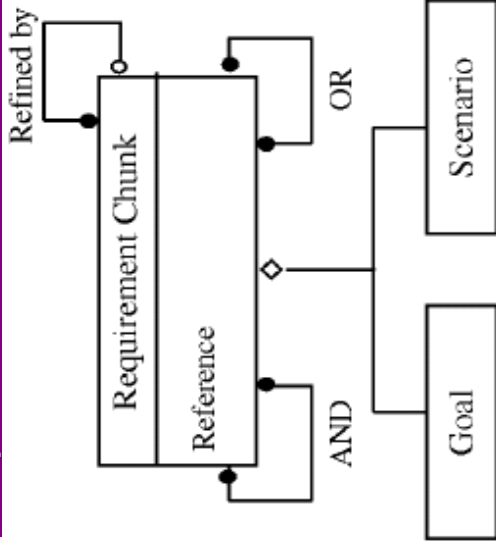


Goals & Scenarios

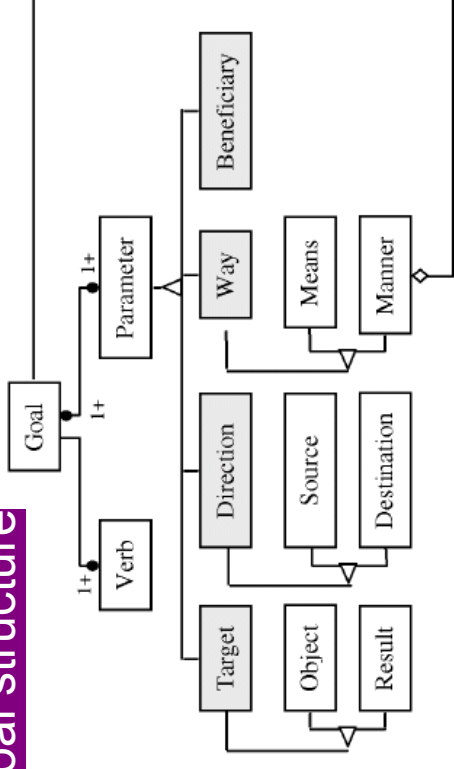
[See module on scenario analysis]

C. Rolland, C. Souveyet, and C. Ben Achour, "Guiding Goal Modeling Using Scenarios," IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 24, NO. 12, Dec. 1998. pp. 1055-1071.

Requirements chunk model



Goal structure



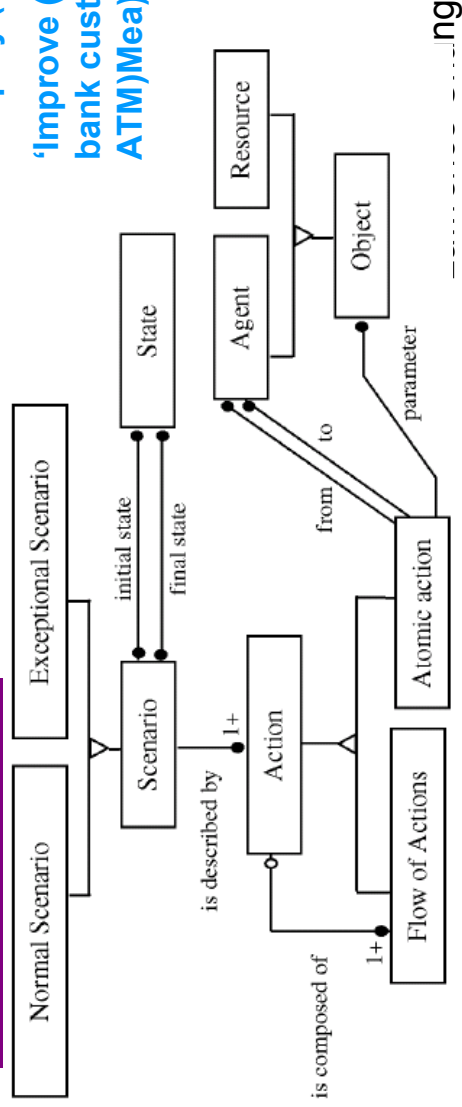
'Take (the receipt)Obj (from the printer)So';

'Read (the validity date of card)Obj (in the card chip)So';

'Display (the error message)Obj (to the customer)Dest';

'Improve (our services)Obj (by providing (cash)Obj (to our bank customers)Dest (from account)So(with a card based ATM)Mea)Man',

Scenario structure



Points to Ponder

- ❖ What is the ontology of SADT?
- ❖ What is the ontology of RML/Telos?
- ❖ What is the metamodel for SADT?
- ❖ What is the metamodel for RML/Telos
- ❖ How do you extend the UML metamodel to incorporate SADT?
- ❖ How do you extend the UML metamodel to accommodate RML/Telos
- ❖ How do you extend the UML metamodel to accommodate NFRs?