

INFS4206/7206 Advanced Topics in Database

2006 Semester 2
Part 1
The Meta-Object Facility (MOF)

Last Week

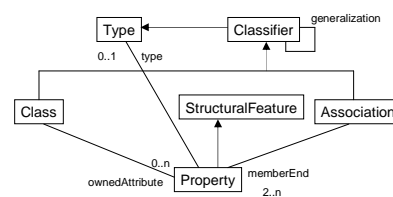


- ❖ **Metamodel** is conceptual model for syntax of a modeling system. Specifies **schema for repository**
- ❖ Repository stores **model instances**. **Constraints** expressed as queries on repository. Repository supports model creation, editing, **rendering**, browsing etc.
- ❖ Metamodel usually has **classes model** and **instances model**

The MOF

- ❖ The MOF is used to develop **OMG metamodels**
- ❖ MOF models can be huge
 - UML (Superstructure) is over 800 pages
- ❖ MOF is a subset of UML
 - UML Infrastructure, only 200 pages
 - MOF, additional 90 pages
- ❖ Two major ways to control size of model
 - Packages
 - Layered model using generalization and abstract classes

Meta-Object Facility (MOF) Classes Model



Infrastructure Classes (Super 47, classifiers 44)

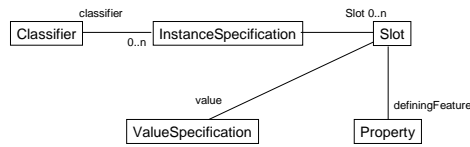
Reading the Classes Diagram

- ❖ *StructuralFeature* is an abstract class
 - Abstract class has instances only by virtue of its (concrete) subclasses
 - *Property* is a concrete class, so has instances
- ❖ Subclasses are not necessarily disjoint
 - An object can be both a class and association
- ❖ Types are machine representations
 - A property has a class as a type
 - A class has an *ownedAttribute* which is the property having the class as its type
 - This is how an association links classes to classes

Reading the Classes Diagram

- ❖ A class in a MOF diagram called a metaclass
- ❖ An association in a MOF diagram called meta-association
- ❖ A classifier has instances by definition
- ❖ What these instances look like is specified by the Instances model
- ❖ Classes model and instance model are both packages

MOF Instances Model



Infra 74

INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

7

How MOF works: metalevels

- ❖ The MOF concept includes metalevels, numbered 3, 2, 1, 0, where the higher level is more abstract
- ❖ Level 3 : MOF a model specification tool
- ❖ Level 2 : Structure of particular modeling tool (STM)
- ❖ Level 1 : Structure of a particular model in a particular tool. Airline passenger STM
- ❖ Level 0 : Structure of instances in a particular model in a particular tool at a particular time. Airline passenger STM as operated by QANTAS at 0900 Thursday 15 September, 2005.

INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

8

How MOF works: metalevels

- ❖ The MOF is its own metamodel. Every structure in the MOF diagrams is an instance of one of the metaclasses or meta-associations in the MOF metamodel, and every metaclass and meta-association in the MOF metamodel is represented in the MOF diagrams.
- ❖ UML is an M2 model. UML includes a State Machines package (Super 593)

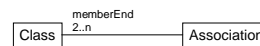
INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

9

How MOF works

Level 3



We know the boxes denote classes, which have instances, and the line denotes relations among classes, represented by tuples

So a valid population of the model (level 2) is an STM model



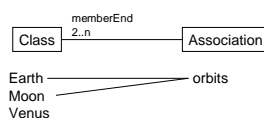
INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

10

How MOF works

But so is



How do we know that *State*, *Transition*, *Event* have instances themselves? That *Earth*, *Moon*, *Venus* don't have instances but *orbits* does?

We know because our natural language understanding of the model has it that way, but the MOF doesn't help with this. We can't use our understanding of how to interpret a MOF model to interpret a MOF model instance. For example the OMG Common Warehouse Metamodel is specified as an M2 model, but could just as well be seen as M1.

INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

11

How MOF works

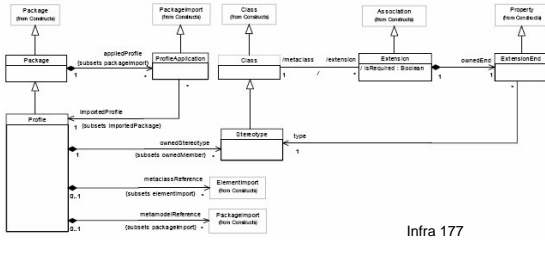
- ❖ By convention, instances (M2) of the MOF element Class (M3) are interpreted as themselves classes, and instances (M2) of the MOF element Association (M3) are interpreted as themselves associations.
- ❖ By convention, instances (M1) of the M2 classes are also interpreted as classes, and instances (M1) of the M2 associations are interpreted as associations.
- ❖ But by convention, instances (M0) of the M1 classes are not classes, nor are instances (M0) of the M1 associations.
- ❖ Class -> State -> WishTravel -> "Bob Colomb"
 - M3 M2 M1 M0

INFS4206/INFS206 part 1 week 2, 19 July 2006

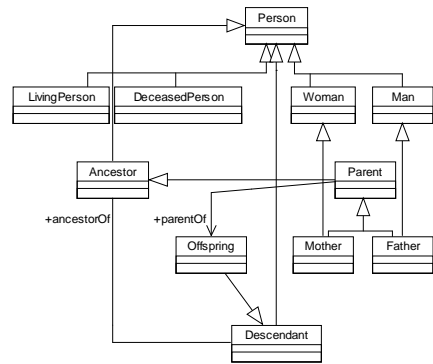
Bob Colomb

12

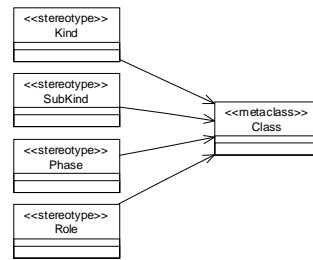
Profiles



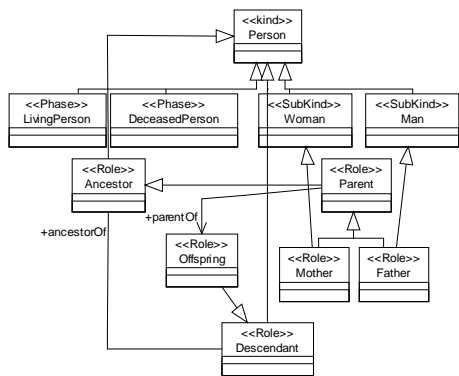
Infra 177



- ❖ Classes diagram is correct, since every instance of every subclass is an instance of all superclasses
- ❖ But not very satisfactory, since Person, DeceasedPerson, Woman and Offspring are profitably thought of as different kinds of object.
- ❖ We can use the profiling mechanism to define a system of submetaclasses of the metaclass Class, so get different kinds of classes



Allowing us to represent the classes diagram as the more informative



Can extend other metaclasses
This extends Property to get
Identifiers as used in ER models

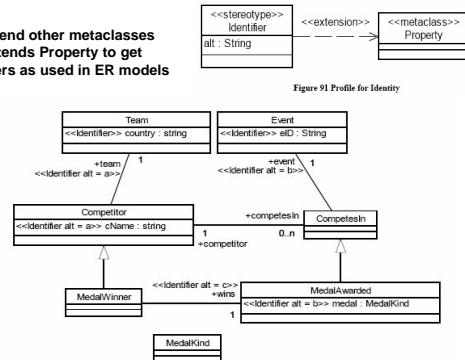
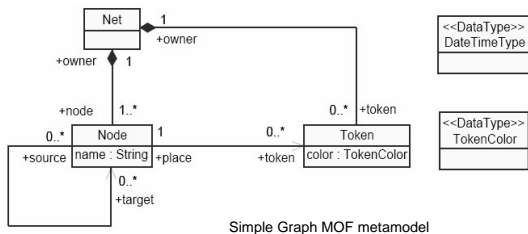


Figure 91 Profile for Identity

Metamodel vs Profile



Simple Graph MOF metamodel

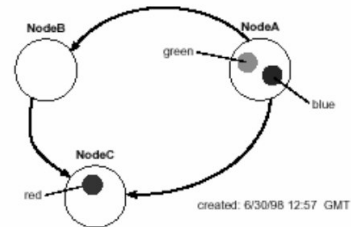
INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

25

Metamodel vs Profile

Metamodel-specific rendering of model instance



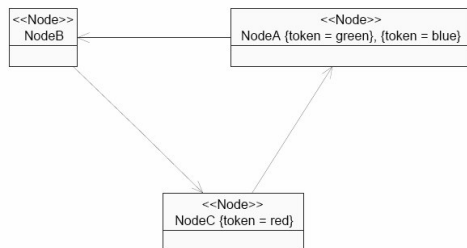
INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

26

Metamodel vs Profile

Instance model using a UML profile for Graphs model



INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

27

Interchange

- ❖ A diagram is a good way for humans to see models, but not a good way to send them from application to application.
- ❖ MOF and UML include an XML markup specification called XML Metadata Interchange (XMI). A model in one tool is rendered as an XML document using the XMI markup, sent to another tool, and then re-rendered in another suitable form.

INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

28

Summary: Key Terms

- ❖ The **MOF** is a subset of UML used to specify metamodels. Uses **layering** and **packages** to organize large models.
- ❖ Alternate way to specify a model is by a specialisation of UML called a **profile**.



INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

29

Resources

- ❖ UML 2.0 Infrastructure
 - ❖ UML 2.0 Superstructure
 - ❖ MOF 2.0
 - ❖ XMI 2.0
 - ❖ ODM Topic Maps Metamodel
 - ❖ Frankel Metamodel vs Profile
- ❖ All on course web site as pdf files. Some are very large.

INFS4206/INFS206 part 1 week 2, 19 July 2006

Bob Colomb

30