

**OBJECT-ORIENTED
MODELLING (OOM) -
32536**

MODULE - 9

**UML's extensibility mechanisms
(POOA – Chapter 5)**

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

1

MODULE

UML's Extensibility Mechanism

Stereotypes; Notes;

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

2

Extending UML

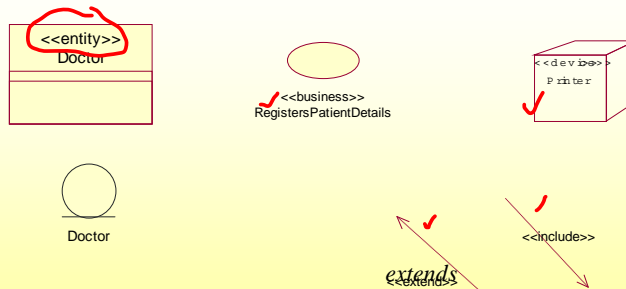
- UML provides three mechanisms to extend itself further
 - Stereotypes - Mechanism to classify anything and everything in UML, by far the most popular
 - Tagged Values and Constraints
 - (May not be used during Analysis)
- Notes:
 - Can be used to Extend diagrams and add further Value to UML diagrams

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

3

Stereotypes

 guillemet



"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

4

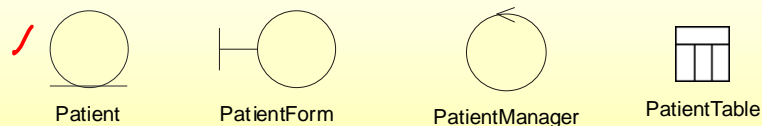
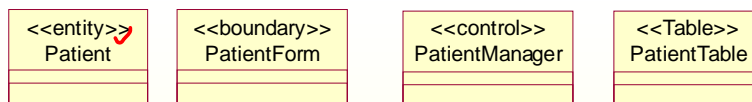
Stereotypes...

- Stereotypes are 'types' of types
 - For Example: Some Use Cases deal with Interactions, Others with Business Logic
 - Use Cases dealing with Business Logic may be categorized as <<Entity>> types of use cases or <<Entity>> Stereotypes
- << >> is the notation for stereotypes
- Each UML artifact Has only one Stereotype

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

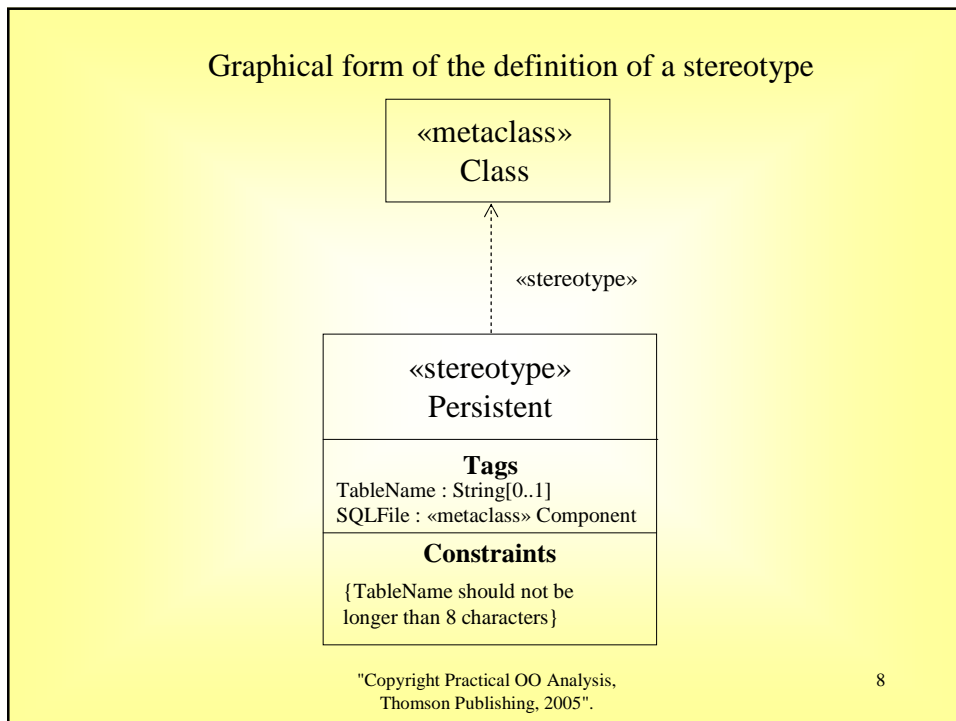
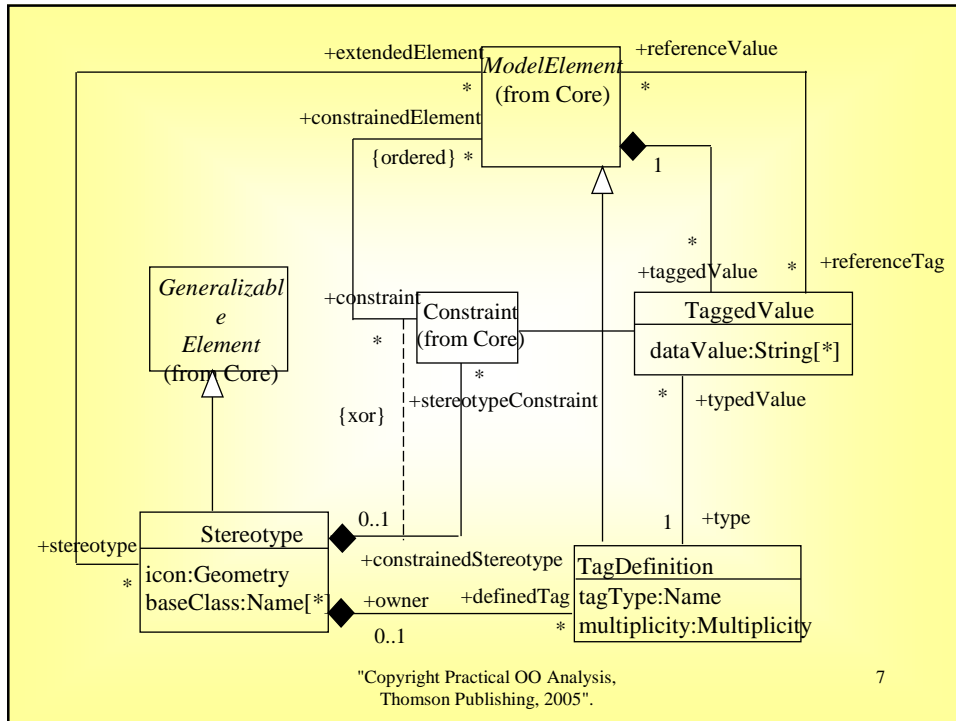
5

Some Stereo typed Classes



"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

6



Tabular form for the Architectural Element stereotype (copyright OMG, 2001)

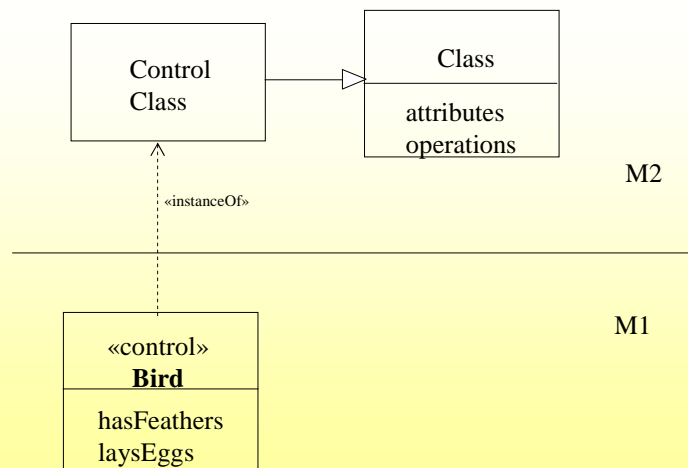
Stereotype	Base Class	Parent	Tags	Constraints	Description
Architectural Element	Generalizable Element	N/A	N/A	N/A	A generic stereotype that is the parent of all other stereotypes used for architectural modelling
Capsule	Class	Architectural Element	isDynamic	self.isActive = true	Indicates a class that is used to model the structural components of an architecture specification.

"Copyright Practical OO Analysis, Thomson Publishing, 2005".

9

Example

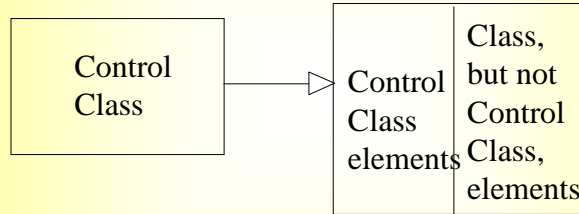
A pseudo or virtual metasubtype (often user-defined) e.g. ControlClass



"Copyright Practical OO Analysis, Thomson Publishing, 2005".

10

Can be viewed as a subset, often as a partition,
of the main UML metaclass, here Class.
The partition name is the «stereotype» label,
here «control»

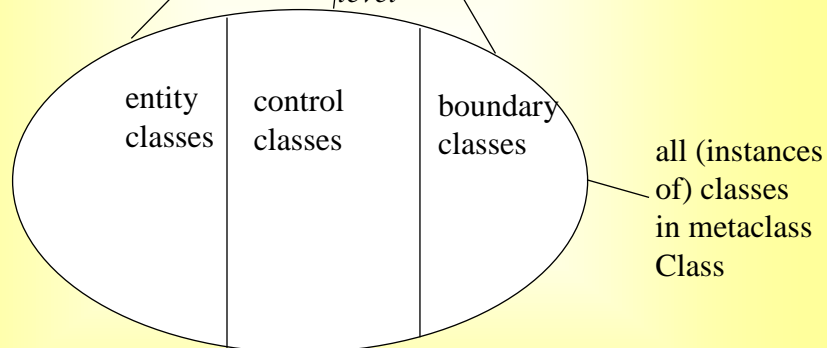


"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

11

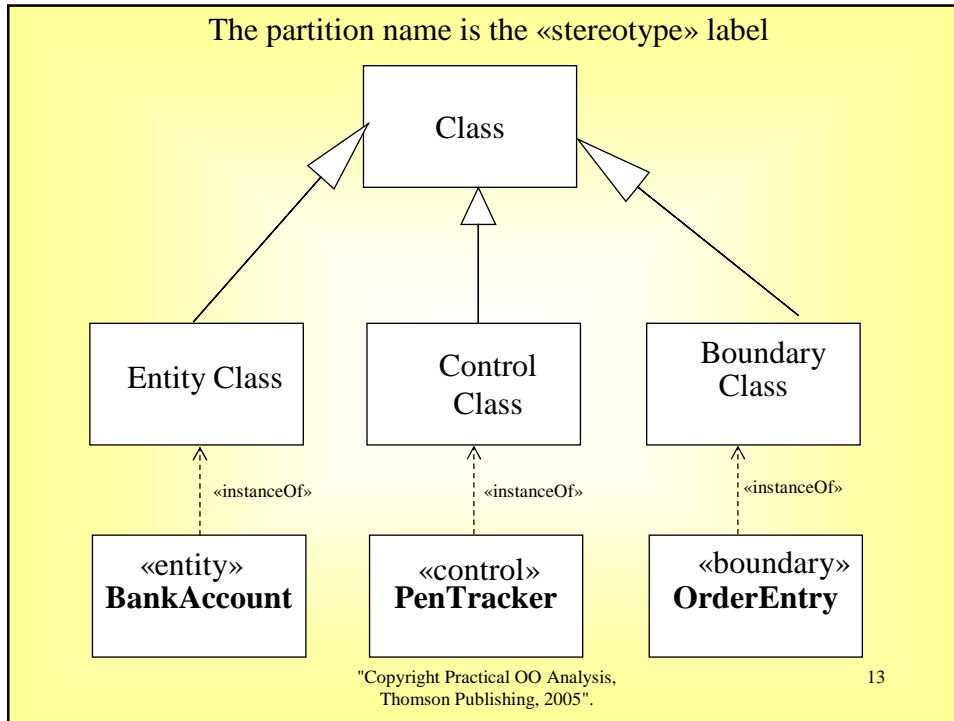
Each M2 level partition can be represented using a stereotype

3 stereotypes form a partition
of the extension *at the M2*
level



"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

12



4) Stereotypes applied to Comment

- responsibility

5) Stereotypes applied to Constraint

- invariant
- precondition
- postcondition

There are, of course, other metatypes that use the stereotype notation while not being stereotypes themselves e.g. «interface», «extend», «include»

UML2 stereotype mechanism is significantly changed.

(The intent in UML2 metamodel for stereotypes is the same as UML 1.4

However, it is still not yet precisely defined in both versions)

Constraints

- A semantic condition or restriction expressed in text (e.g. informal English, OCL)
- An assertion, non-executable
- Place inside braces { } + possibly connect by dashed line
- May be in a Note icon
- Can be used to express pre- and post-conditions of operation and for class invariants
- BUT note that constraints often lead to inter-dependencies between classes, hence thwarting reusability

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

17

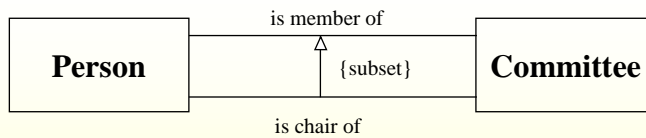
Predefined Constraints

- {xor}
- For Generalization
- {overlapping}
 - {disjoint}
 - {complete}
 - {incomplete}

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

18

Example



{subset} constraint on Dependency

or, in OCL:

Person

```
self.committeeMembers->includes(self.committeeChair)
```

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

19

Tagged Values

- Extends properties of model element but not its instances i.e. not the same as a class attribute - think of as metadata
- Tagged value is a pair: tagName + value

{tagName = value}

or just {tagValue}

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

20

Predefined Tagged Values

- tagName=persistence
tagValue = {transitory}, {persistent}
- tagName=semantics
- tagName=usage (Activity Graph)
tagValue = {uses} {modifies}
- tag for profile packages
tagValue = {applicable Subset}

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

21

Other useful keywords

{abstract}

{leaf}

Other commonly used, but not predefined, tagged values

{readonly} {private} {obsolete}

{Version=1.4} {Author = ABC}

as enumeration

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

22

Example

BankAccount
{abstract, author=BH-S}

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

23

Example constraints

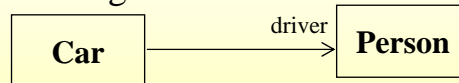
For Person class, `person.age > 21`

Set operations

`person.address->size`

Rolenames of associations (or else class names)

used for navigation



gives context
or focus

Car

or

Car

`self.driver`

`self.person` ↑

specific instance

of context"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

24

More complex examples

Car

```
self.driver->forAll(d|d.age>=21)
```

ensures all drivers are aged at least 21

or

Car

```
self.driver->forAll(age>=21)
```

To find all such drivers

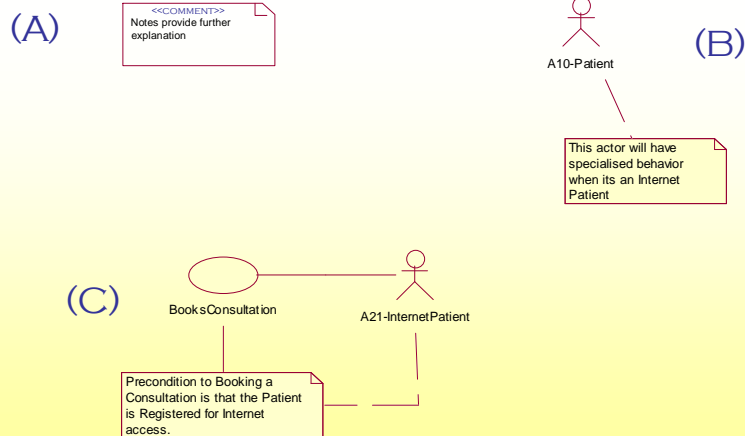
Car

```
self.driver->select(age>=21)
```

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

25

Notes



"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

26

NOTES: Invaluable in Enhancing Quality of UML diagrams



Notes can be drawn on ALL diagrams; They SHOULD be added to ALL diagrams to Enhance Readability;
Notes help provide Additional Explanations on Diagrams;
Notes are represented by “Dog ear Rectangle”;
They are then linked to any other ‘things’ on the diagram.
Notes may also be stereotyped, if required.

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

27

Conclusions

- Discussed UML’s extensibility mechanisms that allow UML to be customised for use in specific projects.
- These mechanisms included stereotypes, constraints, tagged values and notes.

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

28



PROJECT WORK : **(DURING TUTORIALS IN THE LAB);**

**Follow the Project Work Requirements
given at the End of the Chapter;
Discuss Project Work with Team
Members and Tutor;
Carry Out the Project Work**

"Copyright Practical OO Analysis,
Thomson Publishing, 2005".

29