Student Online Teaching Advice Notice

The materials and content presented within this session are intended solely for use in a context of teaching and learning at Trinity.

Any session recorded for subsequent review is made available solely for the purpose of enhancing student learning.

Students should not edit or modify the recording in any way, nor disseminate it for use outside of a context of teaching and learning at Trinity.

Please be mindful of your physical environment and conscious of what may be captured by the device camera and microphone during videoconferencing calls.

Recorded materials will be handled in compliance with Trinity's statutory duties under the Universities Act, 1997 and in accordance with the University's policies and procedures.

Further information on data protection and best practice when using videoconferencing software is available at https://www.tcd.ie/info_compliance/data-protection/.

University of Dublin
Trinity College

CSU22041: Information Management I
Information Modeling
Using
The Unified Modelling Language (UML)

… the **art of communication** of the design of information..

2020-2021

Gaye Stephens gaye.stephens@tcd.ie

University of Dublin
Trinity College

UML Class Diagrams

Objects and Classes

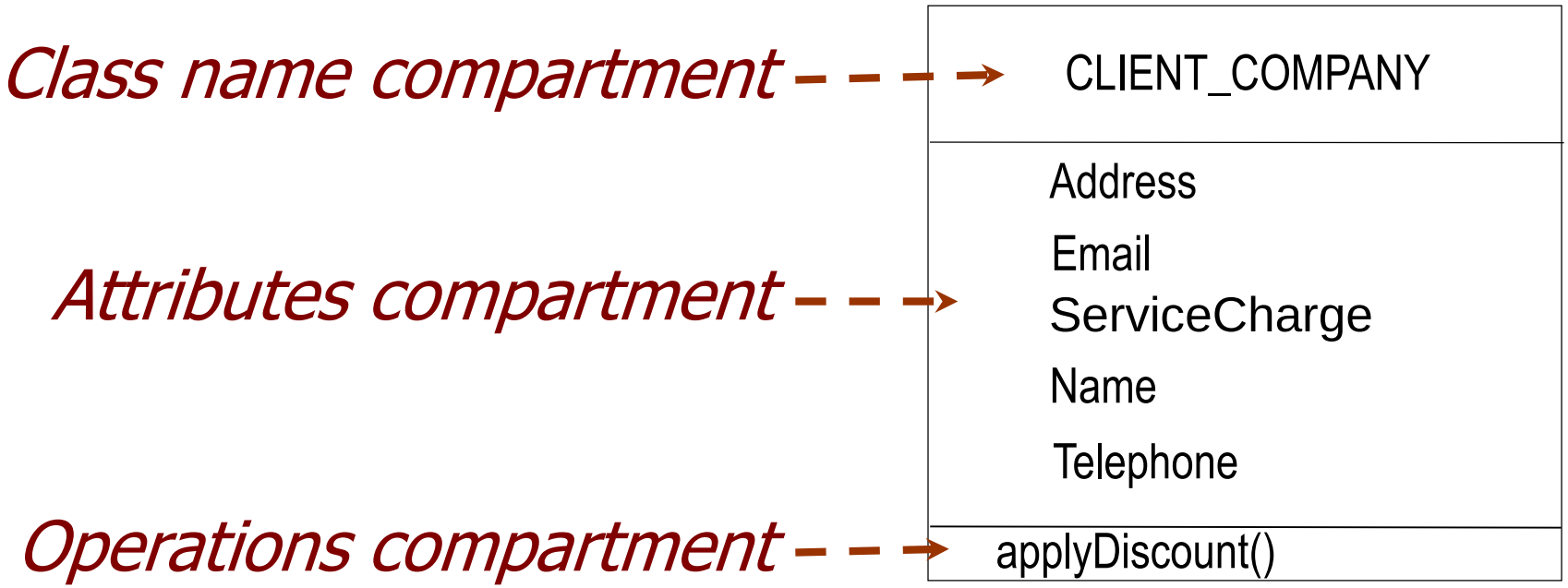
We categorise **objects** into **Classes**. We say that objects are *instances* of classes.

UML Class Diagrams: Purpose

- Used throughout the development process
- Describe, in a visual form, the static structure of system at a certain level of abstraction
- Features of classes: attributes, operations, associations
- Behavioral and data management responsibilities of classes

- Class Diagrams **Do not** show the functional requirements of a system (Use use case models for this)
- Class Diagrams **Do not** show how classes interact at run time (Use interaction diagrams for this)

Class Diagram: Class Symbol



In addition to specific operations each **class** will probably have implied behaviors which are represented as operations e.g. 1) Create new instances 2) Update data or attributes 3) Delete instances 4) Display information

Objects can be persistent or transitory and have state.

Attributes

Attribute: a named property of a class that describes a range of values that instances of the property may hold

Attribute type: Either UML predefined types, model types, or programming language types

Each attribute has one *value* for each **object**

- At a given moment, an object of a class will have specific values for every one of it's class attributes- This is its state.

Attributes

Syntax of an attribute in the UML:

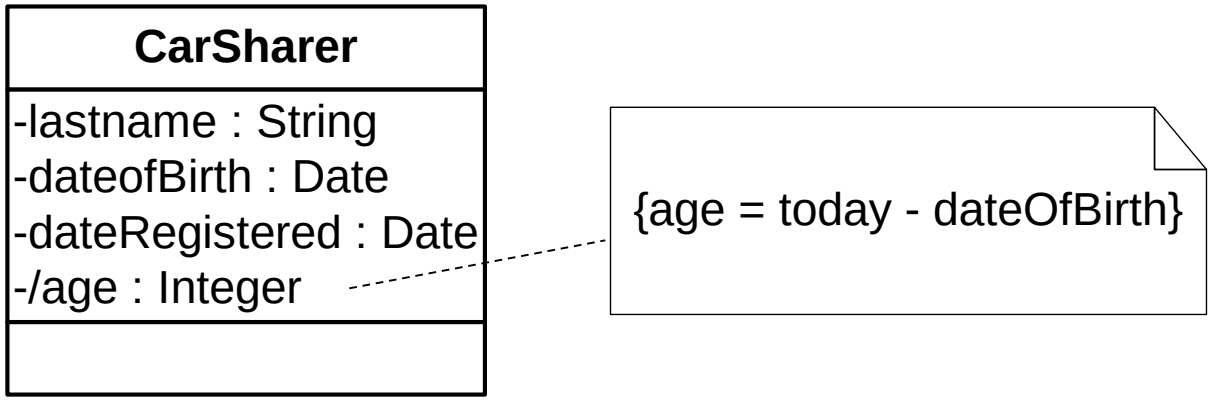
```
[visibility] name [multiplicity] [:type]
[= initial-value] [{property-string}]
```

Visibility indicators +Public -Private

Examples

```
origin                          // name only
+ origin                        // visibility and name
origin : Point                  // name and type
name [0..1] : String            // name, multiplicity, and type
origin : Point = (0,0)          // name, type, and initial value
id : Integer {readOnly}         // name, type, and property
```

Derived attribute



•Derived attributes are used to specify attributes whose value is the result of a computation, based on other attribute values: attribute name is preceded by a “/”.

Operations

Operation: the implementation of a service that you can request on any object of the class

- An abstraction of something you can do to an information entity and that is shared by all instances of that entity

A class may have any number of operations or no operation at all

Are listed in an additional box underneath the attribute box using a specific syntax

```
name (arg1 : type, arg2  : type ...) : return type
```

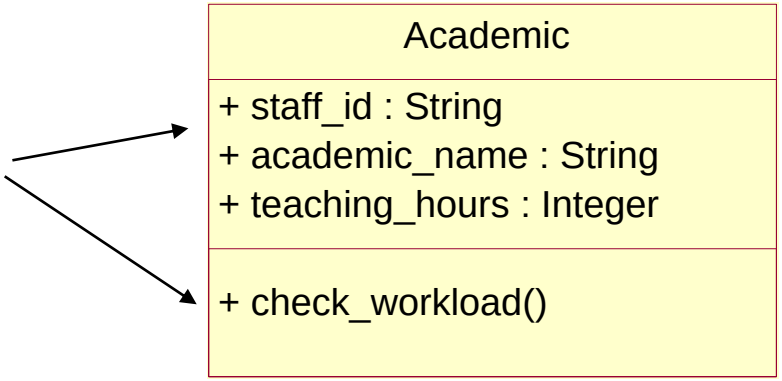
Operations

Bank Account
Account_number Account_name
Check_balance() Debit_account(amount:int) Credit_account(amount:int)

((Visibility indicators

+ public

- private))



How to start Information Modelling: Nouns => Potential classes and attributes

The LearnAlot University offers a number of degrees to under graduate and post graduate students who may be fulltime or parttime.. The educational structure of the university consists of schools. Schools contain several departments. While a single school administers each degree, the degree may include courses from other schools. In fact the university prides itself on the freedom of choice given to students in selecting courses towards their degrees.
Each university degree has a number of compulsory courses and a number of elective courses. Each course is at a given level and has a credit point value. ……

…. A student's proposed program of study is entered in the online enrolment system. The system checks the program's consistency, checks if courses are open and reports any problems…….

Relevant?
Potential Attributes?
Fuzzy?
Irrelevant?

Refinement

Relevant Classes

- Manifestly of interest within problem domain of system, potential record

Potential Attributes

- Representing aspects of an identified record => fields

Irrelevant Classes

- Outside the interest of problem domain of system

Fuzzy Classes

- Cannot confidently classify as irrelevant or relevant **yet**

Operations

- Representing actions related to a record

Roles

- Representing an actor of the system

LearnAlot Class Discovery

Irrelevant

structure

needs

system

Relevant

degree

school

department

course

course offering

timetable

enrolment

instructions

exam results

academic

Fuzzy

- study program
- elective course
- compulsory course

Attributes

- level
- credit point

Operations

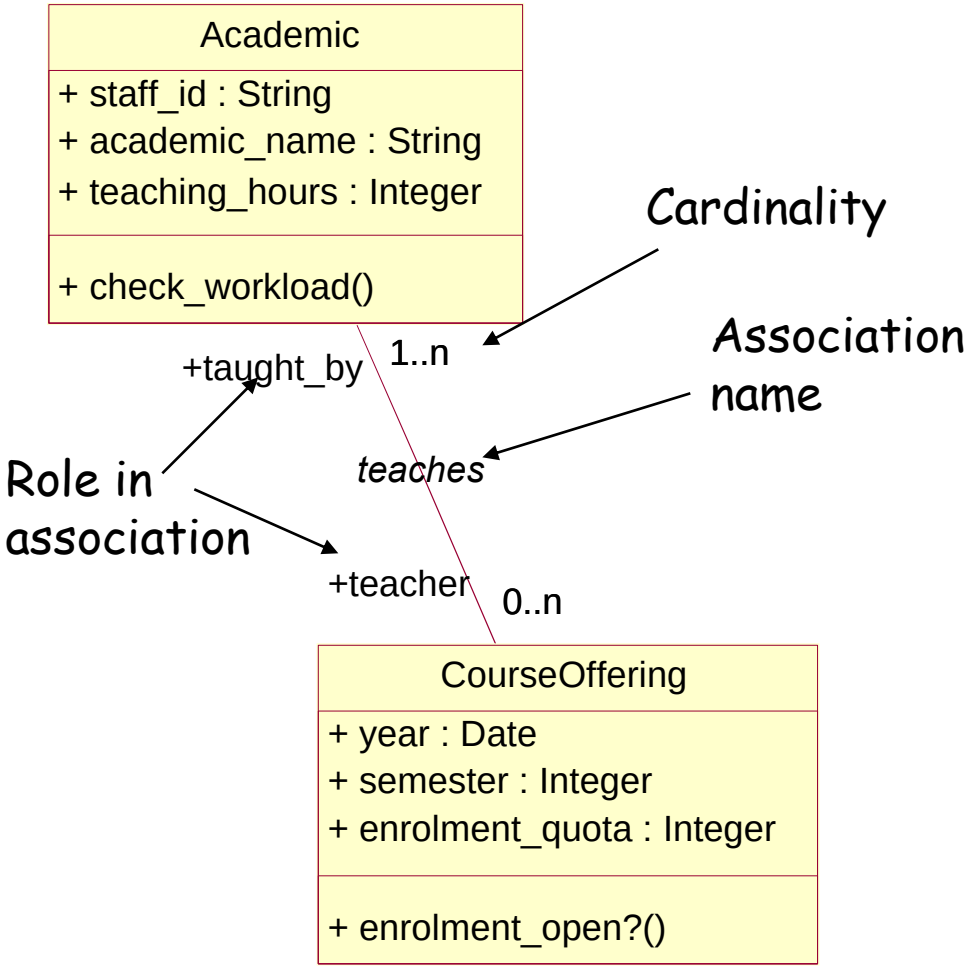
- approval
- open

Roles

- registrar
- student
  - Parttime/Fulltime
  - Under/Post grad
- delegate

Describing relationships: Associations and roles

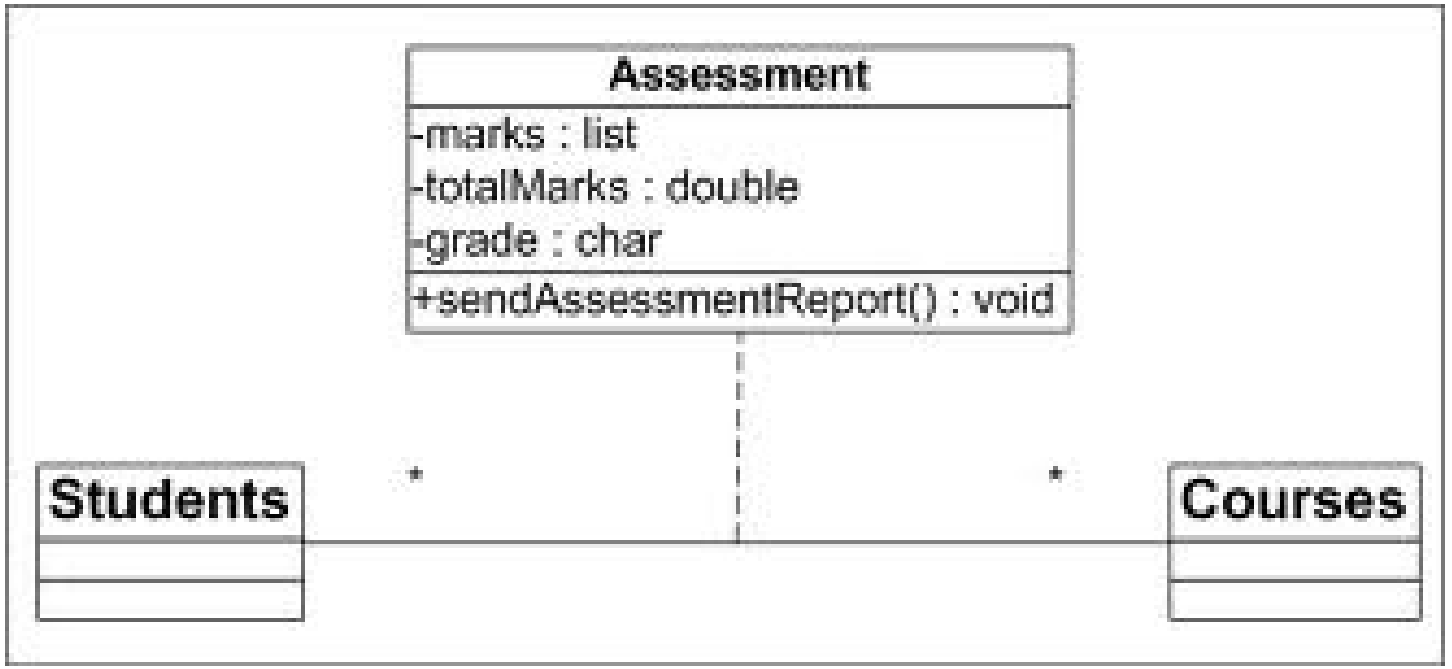
Express the relationships between the problem domain classes using **associations**



CourseOffering is taught_by one or more academics
Academic is teacher_of zero or more course offerings

Association Class

Used to model information in an association

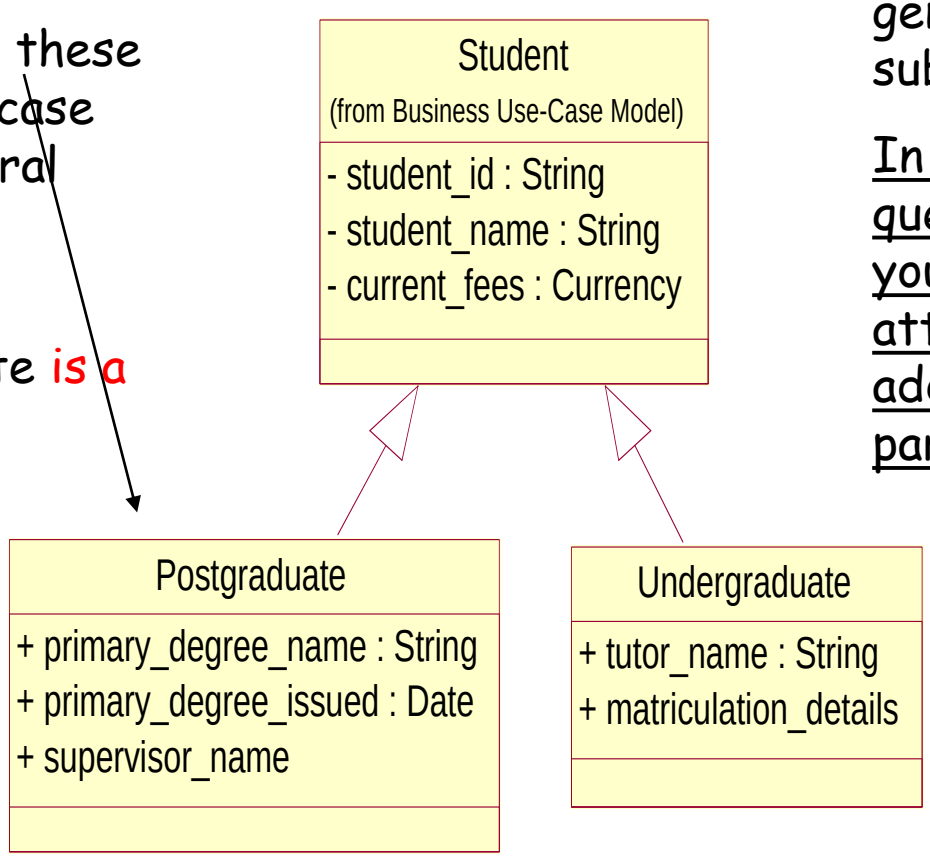


Dotted line from the relevant association indicates the association class

Class association - generalisation

e.g. Each of these is a special case of the general class

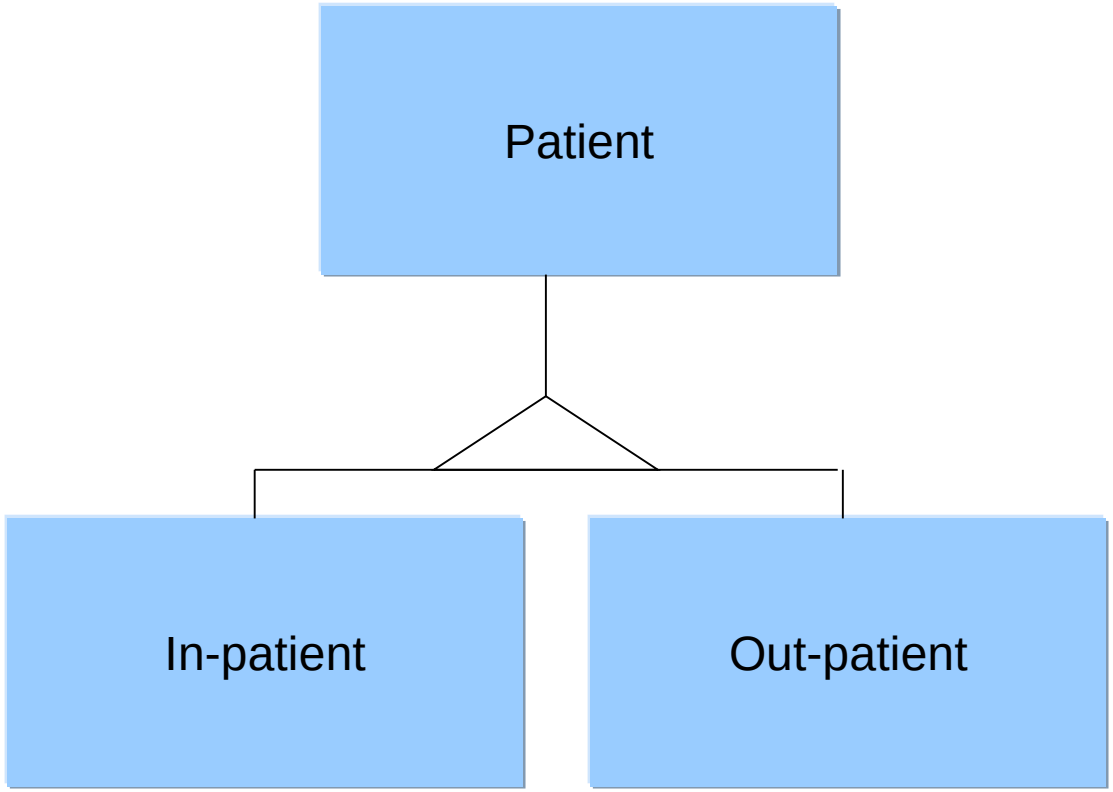
"Every Postgraduate is a Student"



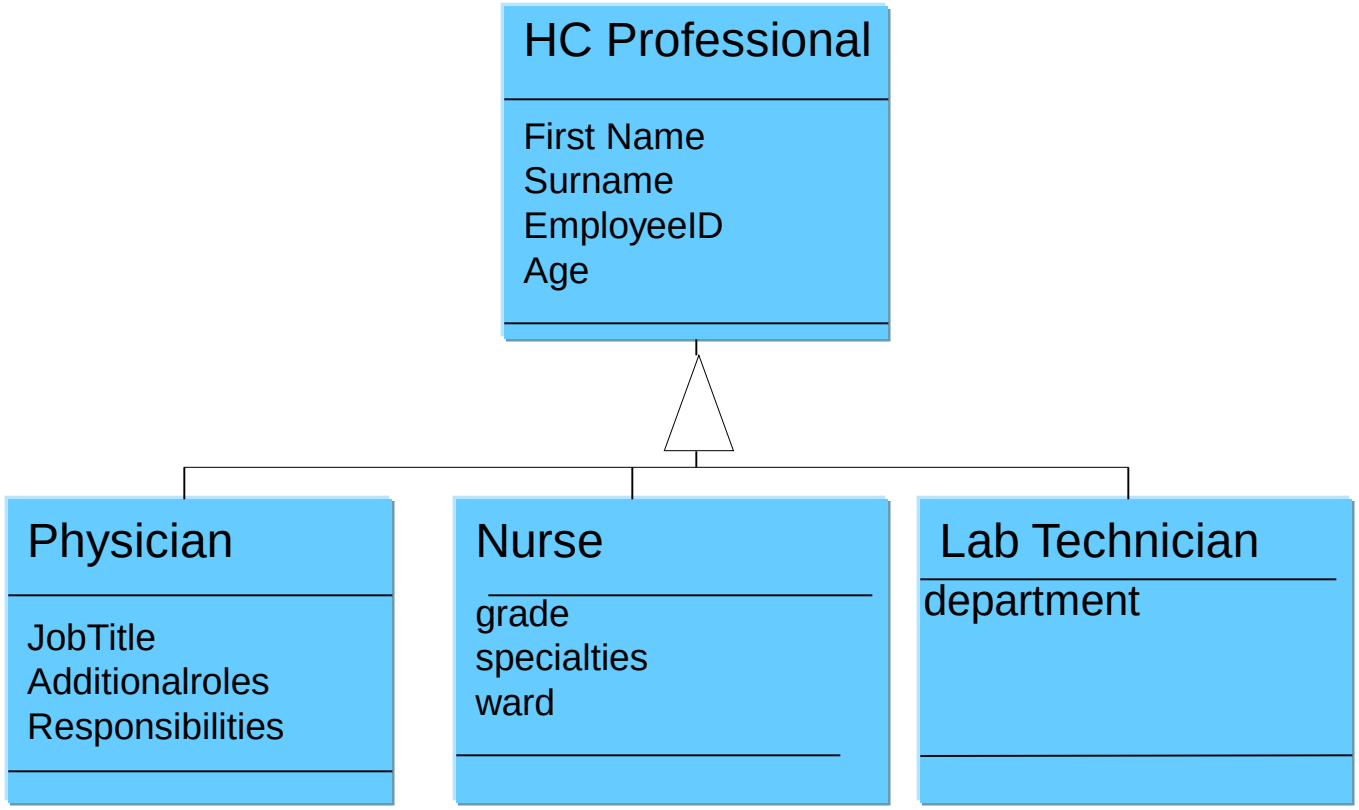
Define sub-classes of the general abstraction –like Java sub-classing

In Information Modelling; key question you should ask yourself is what new attributes does a subclass add that differentiates from parent class?

Class association – generalisation, Another Example

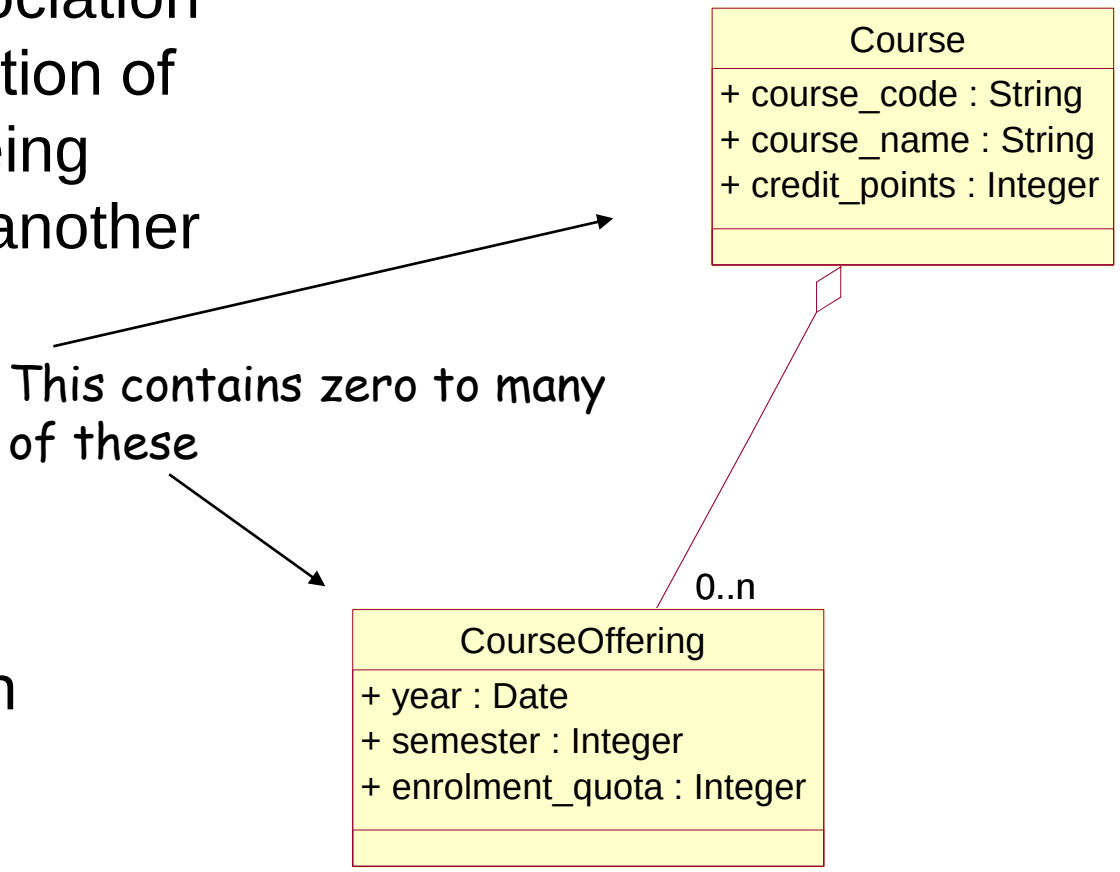


Class association – generalisation, Another Example



Class association - aggregation

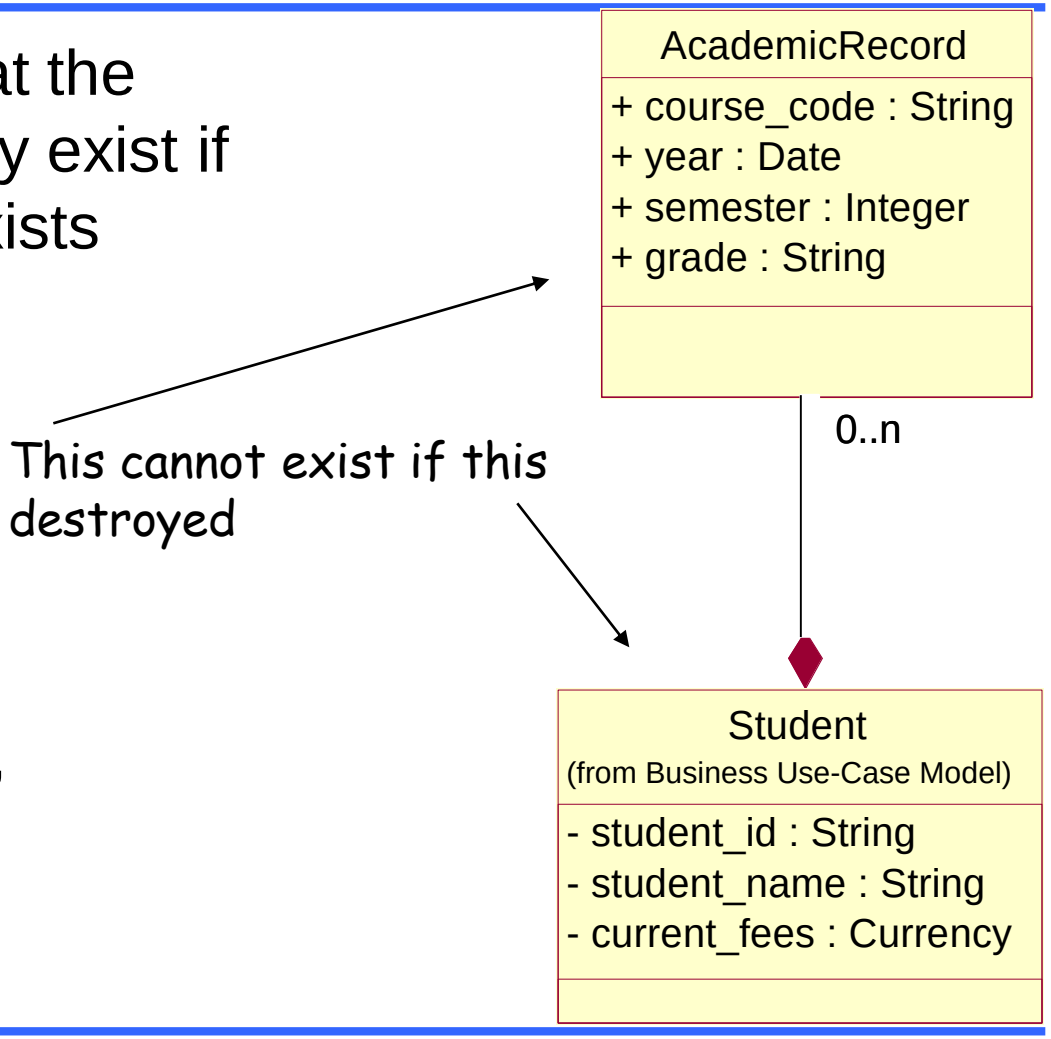
A stronger form of association where there's some notion of objects of one class being "made-up of" those of another



Known as "Aggregation by Reference"

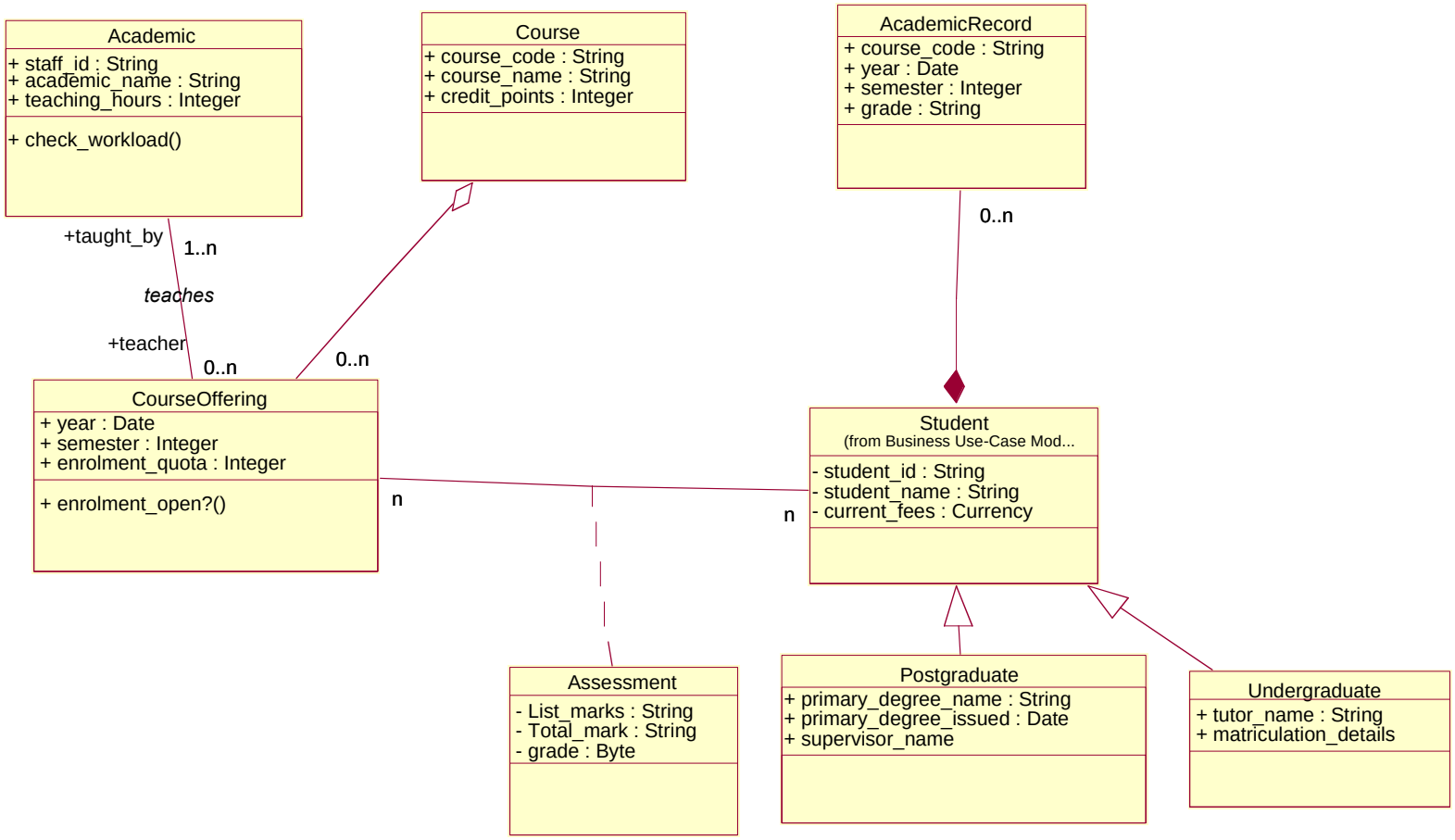
Class association - composition

Aggregation except that the subset classes can only exist if the composed class exists



Known as
“Aggregation by Value”

Putting it all together… the Information Model starts to emerge



Extract from Assignment Sheet Relating to Class Diagrams

A UML Class diagram comprising: at least 15 classes with each class having at least 2 data attributes (with types), Associations to be named and include role and cardinality information, No more than 2 subclass or aggregations, description of design decisions made.

CS2041: Class Diagram Exercise
draw a UML Class diagram including associations, cardinalities, any derived attributes etc.
for a Theatre Ticket Booking System

Customers may have many reservations

Each reservation is made by one customer through a box office

Reservations are of two kinds – subscription series and individual

Each reservation is associated with a ticket or tickets

Each ticket is either associated with a subscription series reservation or an individual reservation but not both

A subscription series comprises at least 3 but not more than 6 tickets

Each ticket or subscription must be paid for

Customers can pay by credit card or cash

Tickets are issued from a kiosk

Every performance has many tickets available each with a unique seat number.

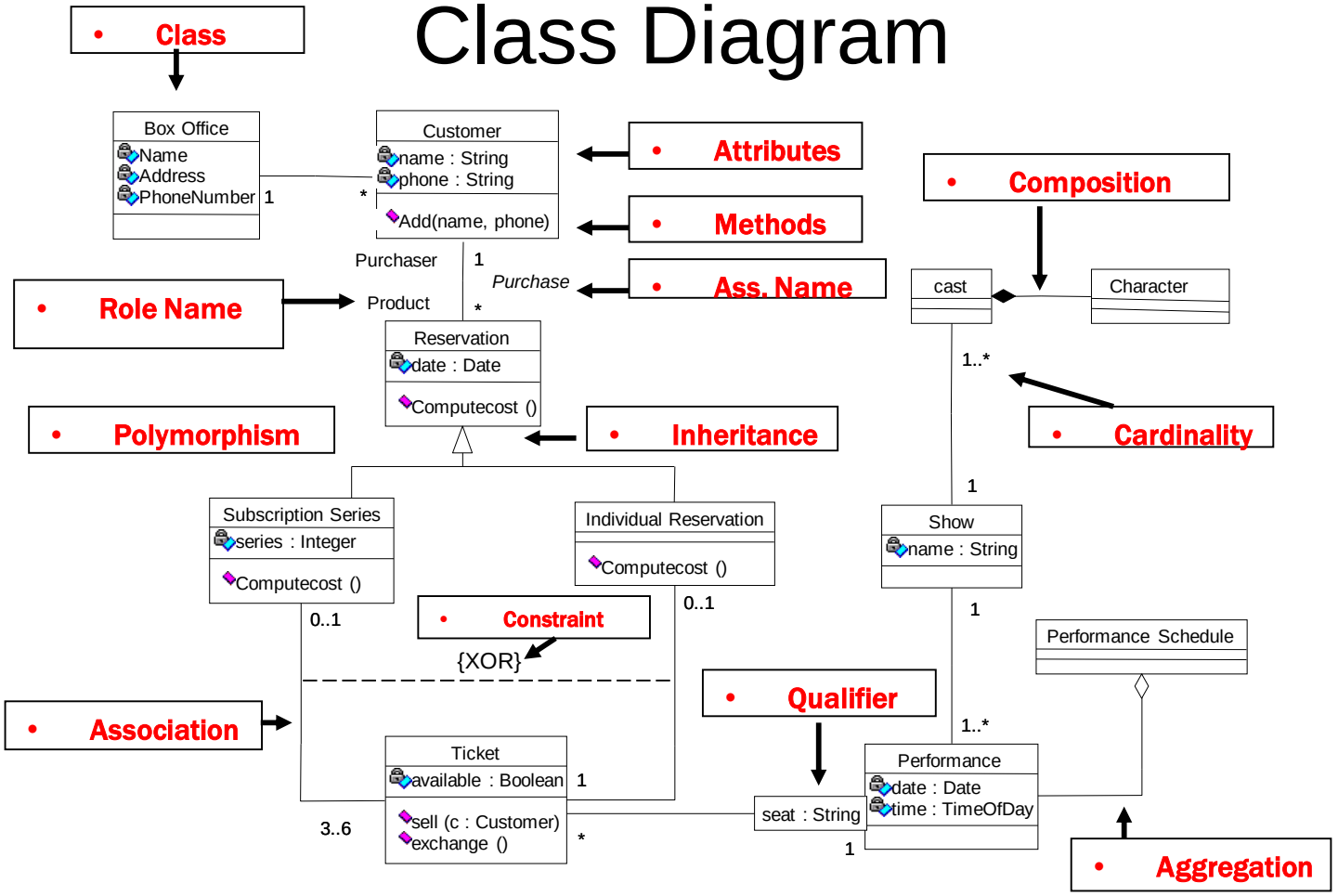
A performance can be identified by a show, date and time.

A performance schedule is a list of performances for a particular show.

A cast and a reserve cast is associated with each show

A cast is composed of a group of actors

Class Diagram



Design Pattern Book

Design Patterns: Elements of Reusable Object-Oriented Software
By Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides

Published 1994

That's All Folks Thank You for Listening

"Please don't ask me to remind you to do anything else for awhile."