University of Science, VNU-HCM
Faculty of Information Technology

Static Analysis

Assoc. Prof. TRAN Minh Triet
Department of Software Engineering

TRUONG DAI HOC KHOA HOC TU NHIEN
KHTN
TP. HO CHI MINH

Software Analysis and Design

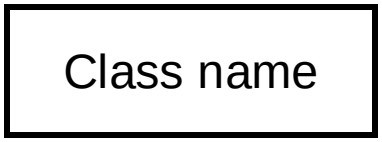
Reference

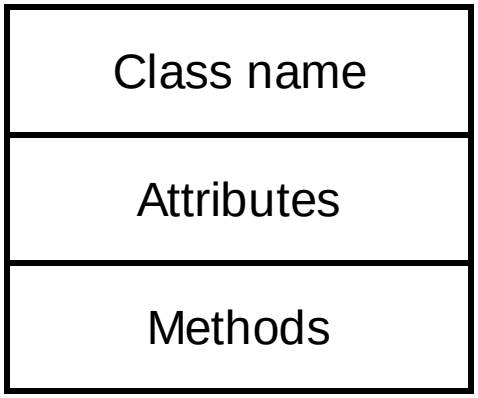
“Mastering Object-Oriented Analysis and Design with UML 2.0”

IBM Software Group

Revision: Object Oriented Programming

Notations

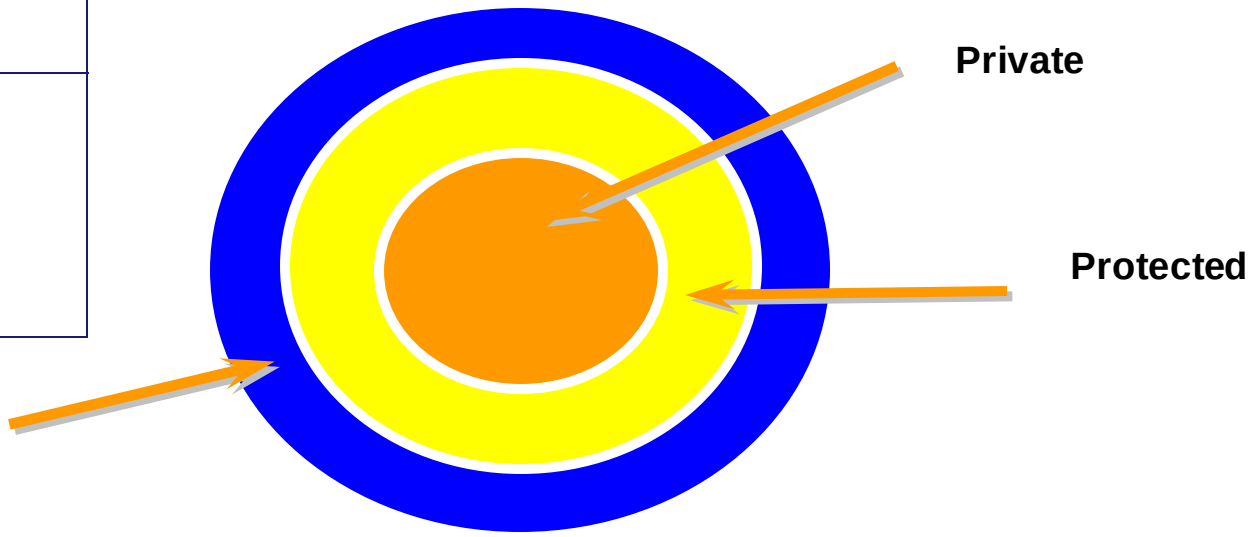




Visibility: Public/Protected/Private

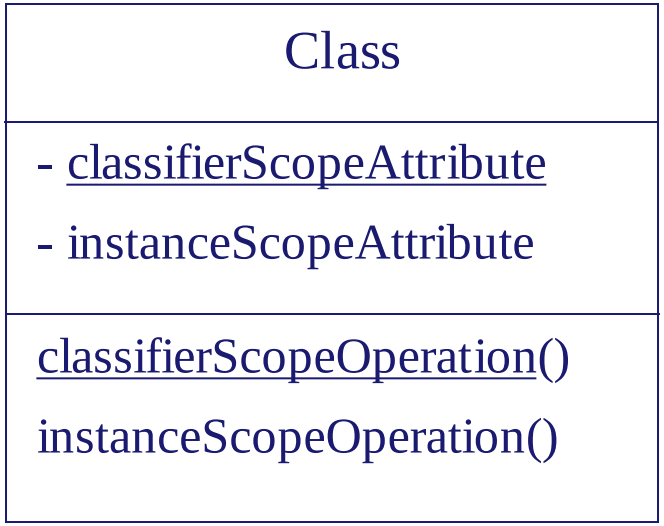
+	public attributes/methods
#	protected attributes/methods
-	Private attributes/methods

Class
- privateAttribute # protectedAttribute
+publicOp() # protectedOp() - privateOp()



Scope

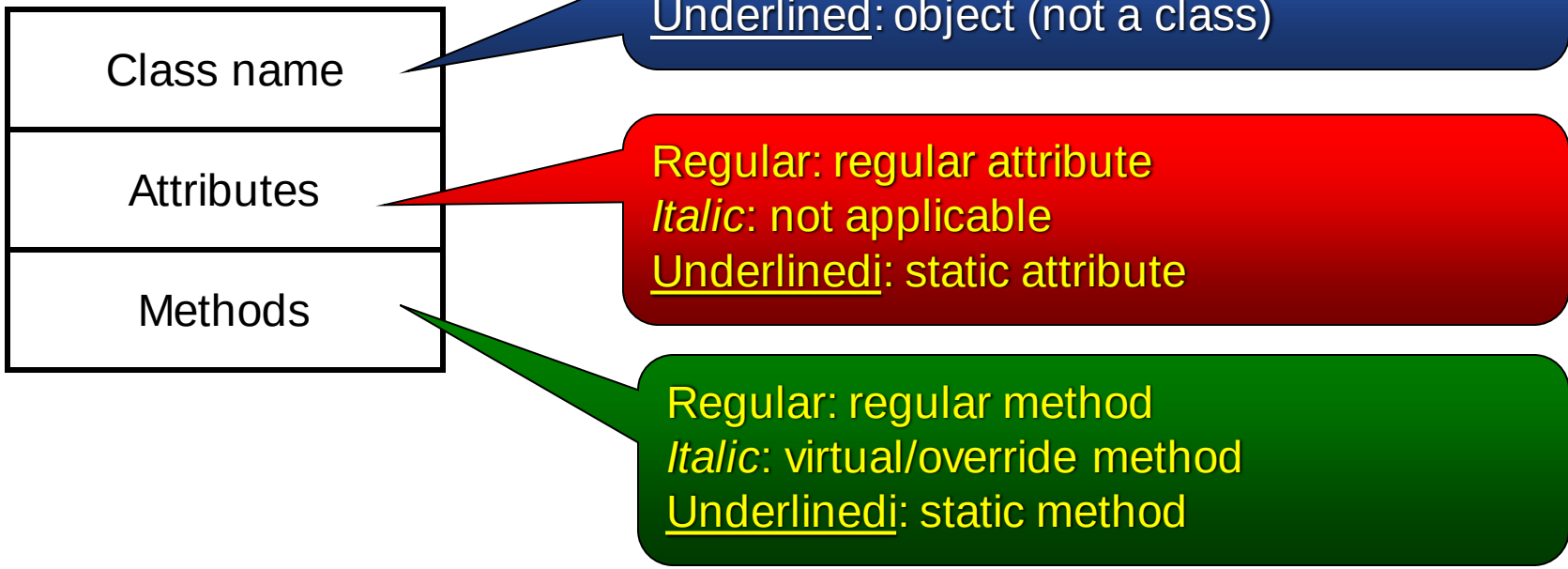
- Number of instances



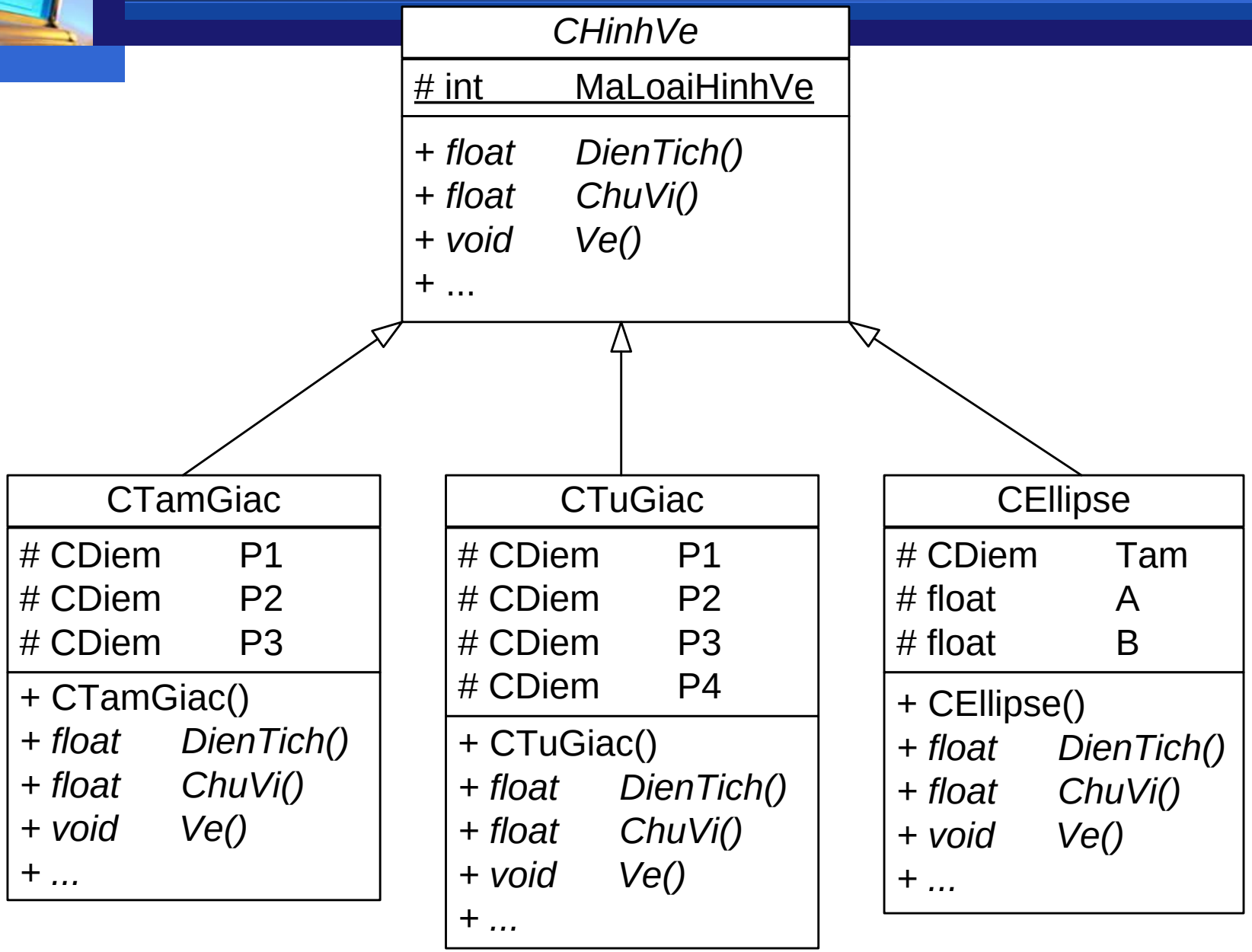
Example

Student
- name - address - studentID - nextAvailID : int
+ addSchedule(theSchedule : Schedule, forSemester : Semester) + getSchedule(forSemester : Semester) : Schedule + hasPrerequisites(forCourseOffering : CourseOffering) : boolean # passed(theCourseOffering : CourseOffering) : boolean + getNextAvailID() : int

Hints

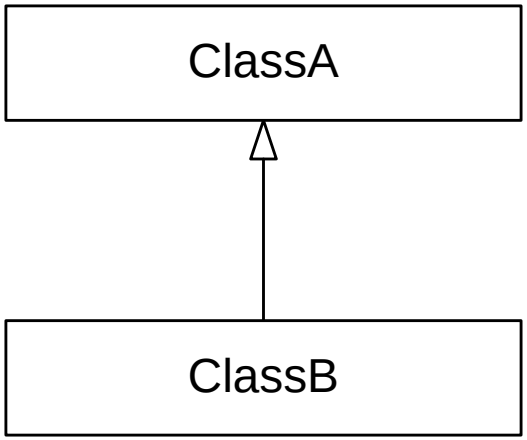

Class name
Attributes
Methods
Regular: regular class
Italic: abstract class/Interface
Underlined: object (not a class)
Regular: regular attribute
Italic: not applicable
Underlinedi: static attribute
Regular: regular method
Italic: virtual/override method
Underlinedi: static method

Example


CHinhVe
int MaLoaiHinhVe
+ float DienTich()
+ float ChuVi()
+ void Ve()
+ ...
CTamGiac
CDiem P1
CDiem P2
CDiem P3
+ CTamGiac()
+ float DienTich()
+ float ChuVi()
+ void Ve()
+ ...
CTuGiac
CDiem P1
CDiem P2
CDiem P3
CDiem P4
+ CTuGiac()
+ float DienTich()
+ float ChuVi()
+ void Ve()
+ ...
CEllipse
CDiem Tam
float A
float B
+ CEllipse()
+ float DienTich()
+ float ChuVi()
+ void Ve()
+ ...

Relationships between classes

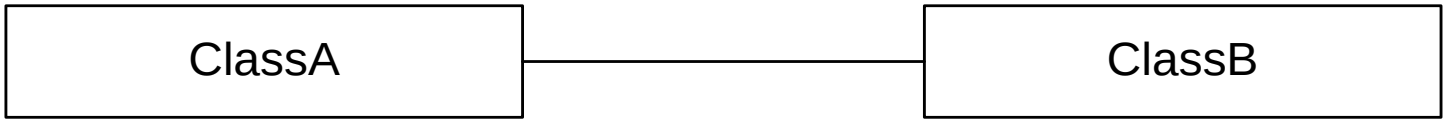
- Generalization



  - ClassB inherits ClassA
  - ClassB is a special case of ClassA
  - ClassA is a general case of ClassB

Relationships between classes

- Association



  - There is an attribute of ClassB in ClassA

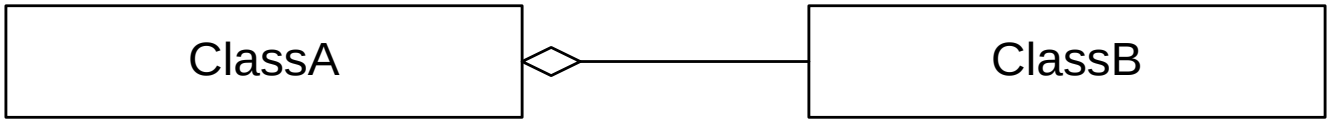
or

  - There is an attribute of ClassA in ClassB

  - The attribute can be represented as a single variable, an array, a pointer, etc
  - Example:?

Relationships between classes

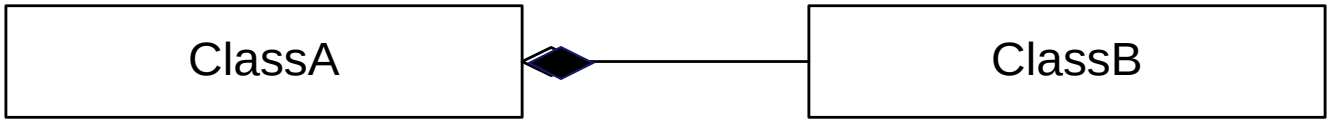
- Aggregation



  - There is an association relationship between ClassA and ClassB
  - More details?
    - ClassB is a part of ClassA
    - ObjectX of ClassA is destroyed then ObjectY of ClassB (belongs to ObjectX) may have a chance to "survive"
  - Example:?

Relationships between classes

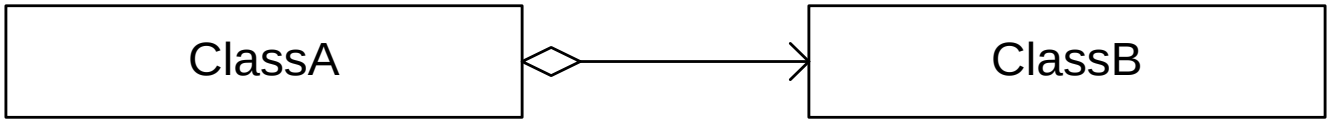
- Composition



  - There is an association relationship between ClassA and ClassB
  - More details?
    - ClassB is a part of ClassA
    - ObjectX of ClassA is destroyed then ObjectY of ClassB (belongs to ObjectX) must be destroyed
  - Example:?

Relationships between classes

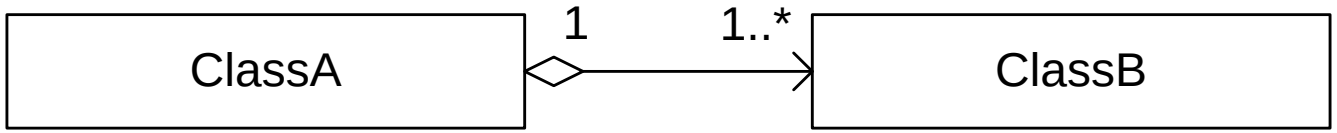
- Direction of a relationship (Association, Aggregation, Composition)



  - Uni-directional relationship : most of the function calls between objects of ClassA and Class B follow the direction of the relationship
  - Undirectional relationship: there is no arrow head of the relationship

Relationships between classes

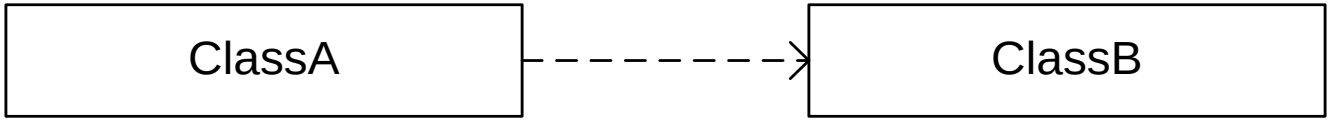
- Multiplicity (Association, Aggregation, Composition)



  - Semantic?
  - Example:
    - 1
    - 2
    - 1..*
    - 0..*
    - *
    - 1, 3, 5..9

Relationships between classes

- Dependency



  - There is no association relationship between ClassA and ClassB
  - ClassA "depends" on ClassB

Input parameter

```
class A
{
    void    F(B   x)
    {
     …
    }
};
```

Returned value

```
class A
{
    B    F()
    {
     …
    }
};
```

Local variable

```
class A
{
    void    F()
    {
        B    x;
    }
};
```

Global variable
Static function
Static attribute

(Analysis) Class Diagram

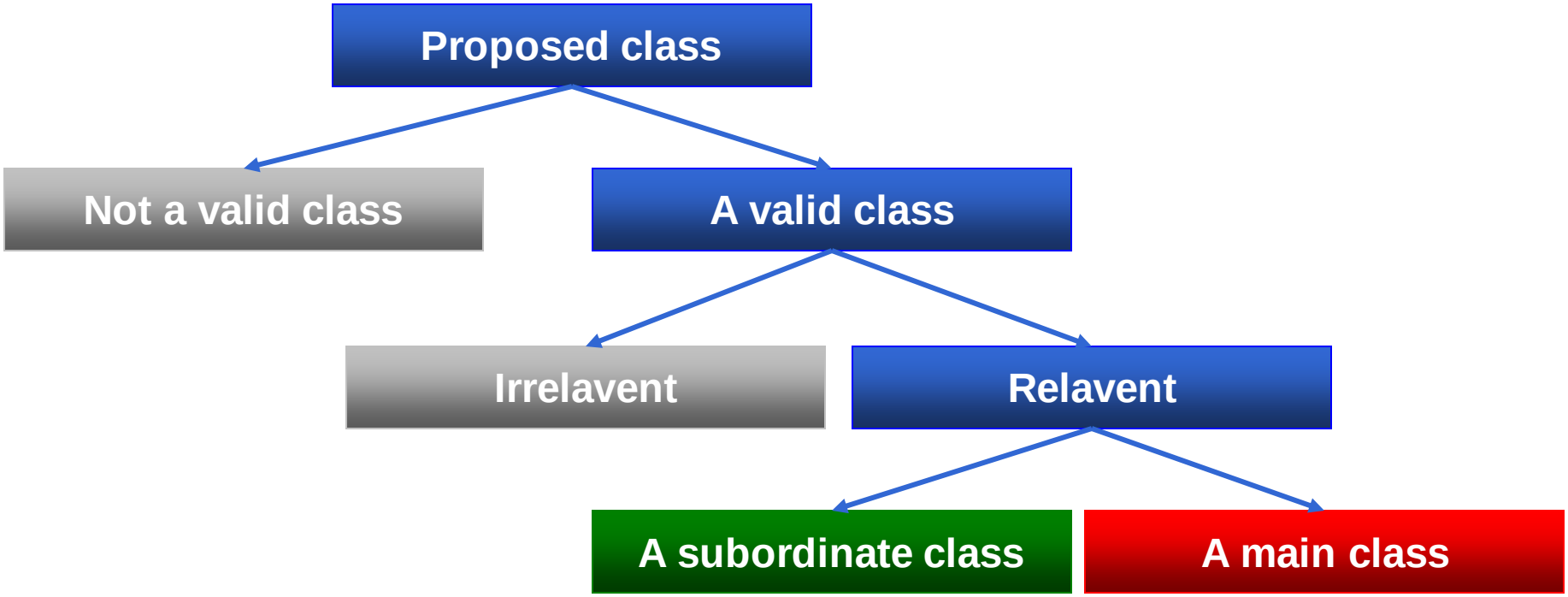
List of class candidates

- How can we recognize a class (and its name)
  - Identifier
  - Life cycle:
  - Relative independence
  - …
- Hints:
  - People
  - Thing
  - Organization
  - Place
  - Time
  - Concept…

- A class name is usually a noun or a noun phrase

List of class candidates

- List of class candidates



Example

- Context: A high school
- Class or not a class?
- List of class candidates:
  - Student
  - Teacher
  - Course
  - Student's parent
  - GPA
  - Department
  - Board of
  - Class
  - Room
  - Academic Year
  - Number of periods
  - Schedule
  - Academic transcript
  - Semester
  - Type of student …

Example

- Relavent/irrelavent?
  - Student Management System:
    - Student, Teacher, Course, Class, Student's Parent, Semester, Academic Year…
  - Teacher Management System :
    - Teacher, Department, Course, Class, Semester, Academic Year …
  - Timetabling System :
    - Teacher, Course, Class, Room, Semester, Academic Year…

Example

- Main class/subordinate class
  - Student Management System:
    - Student, Teacher, Course, Class, Student's Parent, Semester, Academic Year…
  - Teacher Management System :
    - Teacher, Department, Course, Class, Semester, Academic Year …
  - Timetabling System :
    - Teacher, Course, Class, Room, Semester, Academic Year…

List of possible relationships

- Criteria
  - Verb
  - Dependency between objects of the same class or different classes
- Hints:
  - Current/temporary relationship
  - Relationship within the whole life cycle

How to develop a class diagram?

- Step 1: Identify classes, relationships, attributes and methods (responsibility) directly from software requirement
- Step 2:
  - If a class contains an attribute with a complex structure or there are several attributes with strong relationships and a clear semantic, **create** a new subordinate class

How to develop a class diagram?

- Step 3:
  - 3.1. **Multiple** classes with similarities (common

    ➔ **Create** a new abstraction (a new class) as the generalization of such classes.

  - 3.2. **A class** with an classification attribute and the behaviors of instances in that class depend on the value of that attribute

    ➔ **Split** the class into several sub-classes, each of which corresponds to a subgroup of instances having the same value of the classification attribute

How to develop a class diagram?

- Step 4:
  - Modify relationships between classes
- Step 5:
  - Refine the class diagram (personal experience required)