

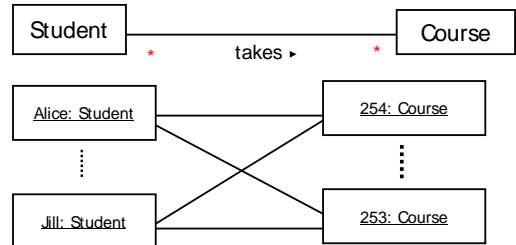
UML and Classes, Objects and Relationships [2]

Defining Domain Models Using Class Diagrams

1

Association - Multiplicity

- A Student can take many Courses and many Students can be enrolled in one Course.



2

Notes

- One class can be relate to another in a
 - One-to-one
 - One-to-many
 - One-to-one or more
 - One-to-zero or one
 - One-to-a bounded interval (one-to-two through twenty)
 - One-to-exactly n
 - One-to-a set of choices (one-to-five or eight)

3

Notes

- Multiplicity can be expressed as,
 - Exactly one - 1
 - Zero or one - 0..1
 - Many - 0..* or *
 - One or more - 1..*
 - Exact Number - e.g. 3..4 or 6
 - Or a complex relationship – e.g. 0..1, 3..4, 6..* would mean any number of objects other than 2 or 5

4

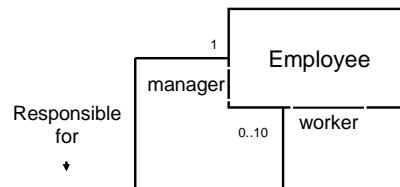
Association - Self

- An association that connects a class to itself is called a self association.

5

Association - Self

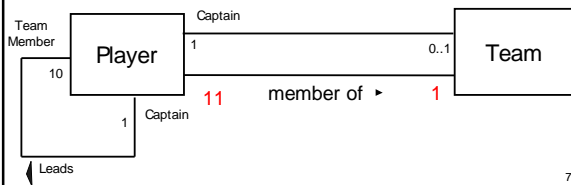
- A Company has Employees.
- A single manager is responsible for up to 10 workers.



6

Association - Multiplicity

- A cricket team has 11 players. One of them is the captain.
- A player can play only for one Team.
- The captain leads the team members.



7

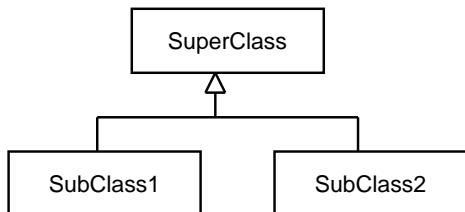
Class Relationships

- Association
- Generalization
- Realization
- Dependency

8

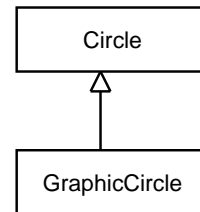
Generalization (Inheritance)

- Child class is a special case of the parent class



9

Generalization (Inheritance) e.g.



10

Inheritance - Implementation

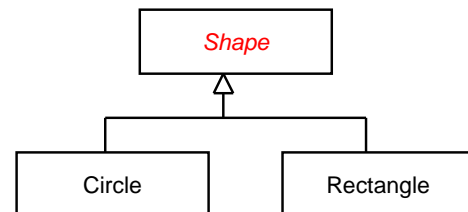
```

public class Circle {
}

public class GraphicCircle extends Circle {
}
  
```

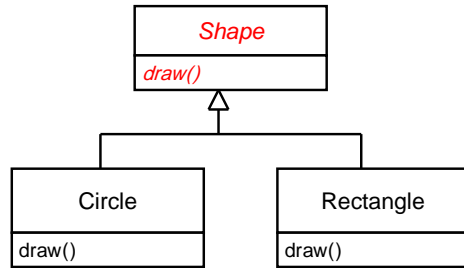
11

Abstract Class



12

Abstract Methods (Operations)



13

Abstract class and method Implementation

```

public abstract class Shape {
    public abstract draw(); //declare
    without implementation
    .....
}

public class Circle {
    public draw(){
        .....
    }
    .....
}
    
```

14

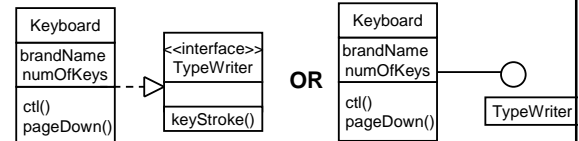
Class Relationships

- Association
- Generalization
- *Realization*
- Dependency

15

Realization- Interface

- Interface is a set of operation the class carries out



16

Realization - Implementation

```

public interface TypeWriter {
    void keyStroke()
}

public class KeyBoard implements TypeWriter {
    public void keyStroke(){
        .....
    }
}
    
```

17

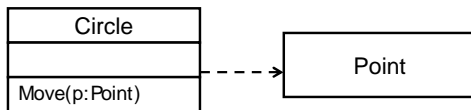
Class Relationships

- Association
- Generalization
- Realization
- *Dependency*

18

Dependency

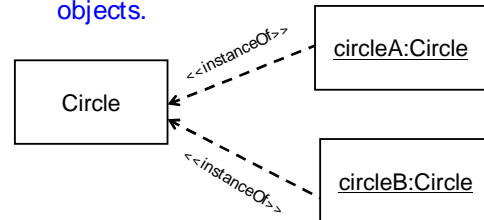
- Change in specification of one class can change the other class. This can happen when one class is using another class.



19

Dependency cont

- Dependency relationship can be used to show relationships between classes and objects.



20

Class Diagrams

- The UML class diagram consists of several *Classes*, connected with *Relationships*.

21

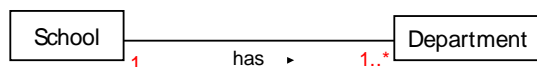
Class Diagram - Example

- Draw a class diagram for a information modeling system for a school.
 - School has one or more Departments.
 - Department offers one or more Subjects.
 - A particular subject will be offered by only one department.
 - Department has instructors and instructors can work for one or more departments.
 - Student can enrol in upto 5 subjects in a School.
 - Instructors can teach upto 3 subjects.
 - The same subject can be taught by different instructors.
 - Students can be enrolled in more than one school.

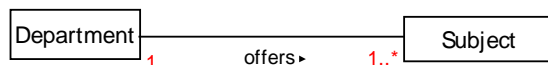
22

Class Diagram - Example

- School has one or more Departments.



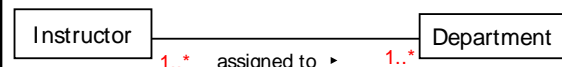
- Department offers one or more Subjects.
- A particular subject will be offered by only one department.



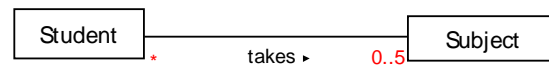
23

Class Diagram - Example

- Department has Instructors and instructors can work for one or more departments.



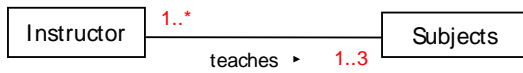
- Student can enrol in upto 5 Subjects.



24

Class Diagram - Example

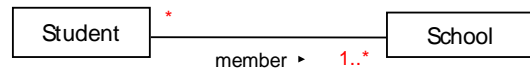
- Instructors can teach up to 3 subjects.
- The same subject can be taught by different instructors.



25

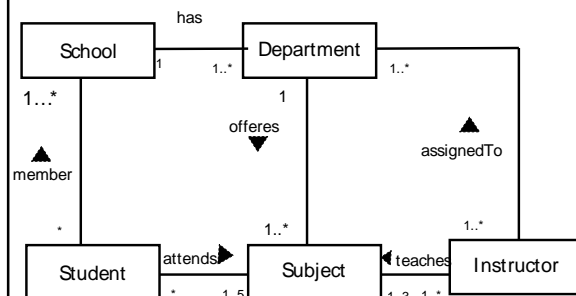
Class Diagram - Example

- Students can be enrolled in more than one school.



26

Class Diagram Example



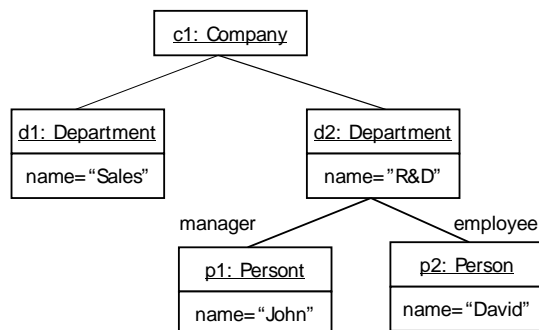
27

Object Diagram

- Object Diagram shows the relationship between objects.
- Unlike classes objects have a state.

28

Object Diagram - Example



29