UML Data Modelling

Carla Teixeira Lopes

Bases de Dados
Mestrado Integrado em Engenharia Informática e Computação, FEUP

Based on Jennifer Widom slides

Agenda

Introduction to Database Design

Classes

Associations

Association Classes

Generalizations

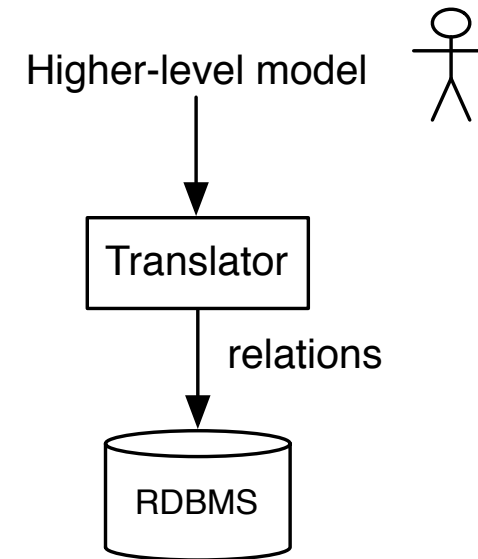Composition & Aggregation

Constraints

Derived Elements

Data modelling

How to represent data for application

Database design model
    Not implemented by system
    Translated into model of DBMS

Higher-level model

Translator

relations

RDBMS

Higher-Level Database Design Models

Entity-Relationship Model (E/R)

Unified Modeling Language (UML)
   Data modeling subset

Both are graphical

Both can be translated to relations automatically
   Or semi-automatically

Key concepts

Classes

Constraints

Associations

Derived Elements

Association Classes

Generalizations

Composition & Aggregation

Classes

Descriptor of a set of objects that share the same properties (semantics, attributes, and relationships)
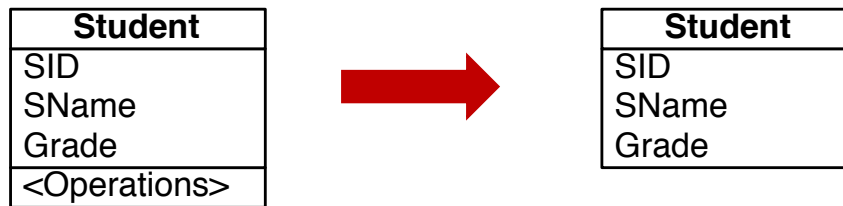
Concrete things
  person, book, car, …

Conceptual things
  class, course, profession, …

Classes

Characterized by name, attributes and operations

  For data modeling: drop operations

Student
SID
SName
Grade
<Operations>

→

Student
SID
SName
Grade

The class name is usually written in the singular, with the first letter in uppercase

Attributes

Attributes are defined in terms of class, while values of the attributes are defined at the instance level

 A student has the attributes: identifier, name and admission grade

 John is a student with the id *123*, name *John Smith* and an admission grade of *180*

A class should not have two attributes with the same name

Attributes can be associated with types

 Not predefined in UML

 Use the ones of the DBMS

Student
SID: integer
SName: string
Grade: integer

In data modelling, we can also specify a primary key

Student
SID pk
SName
Grade

Exercise

Imagine the SIGARRA database.

List 3 classes with attributes that might be in this database.
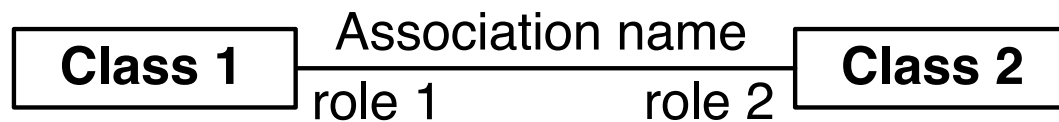
Key concepts

Classes

Associations

Association Classes

Generalizations

Composition & Aggregation

Constraints

Derived Elements

Associations

Relationships between objects of two classes

```
                  Association name
┌──────────┐ ─────────────────────── ┌──────────┐
│ Class 1  │                         │ Class 2  │
└──────────┘ role 1          role 2  └──────────┘
```

As an object is an instance of a class, a **link** is an instance of an association
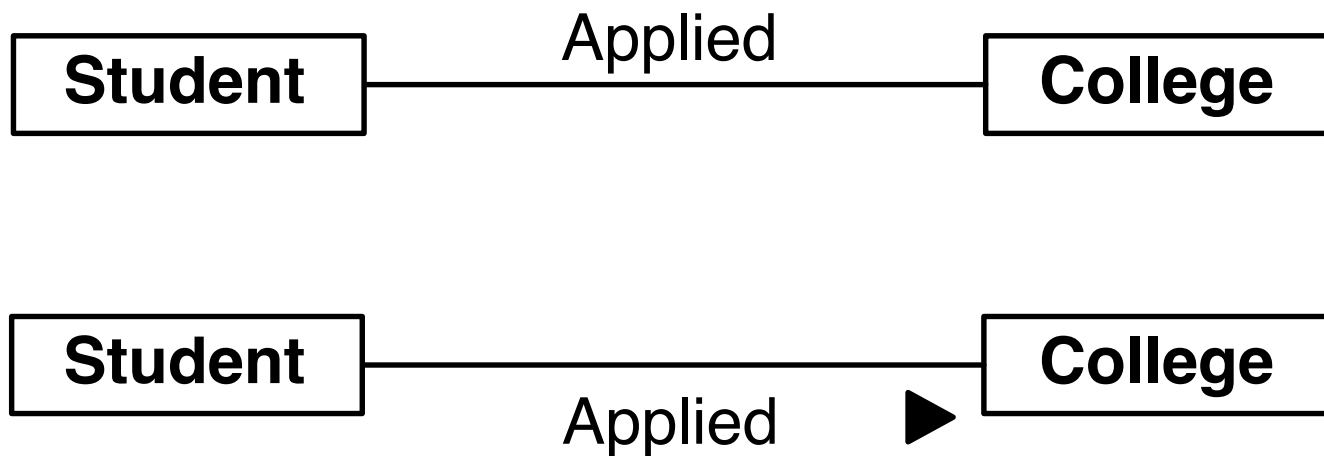
The name is optional

There may be more than one association between the same pair of classes
    Having different names

Associations

Imagine a scenario where students apply to colleges.

What classes would we create? And associations?

Multiplicity of Associations

```
┌──────────┐        Association        ┌──────────┐
│ Class 1  │───────────────────────────│ Class 2  │
└──────────┘                    m..n    └──────────┘
```

Each object of Class 1 is related to at least m and at most n objects of Class 2

A * in place of n stands for *no upper limit*
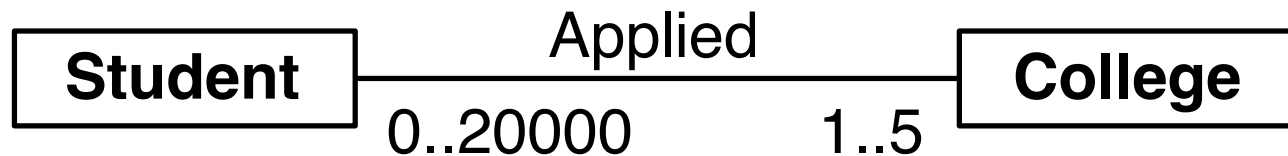
Abbreviations

   * stands for 0..*, that is, no restrictions
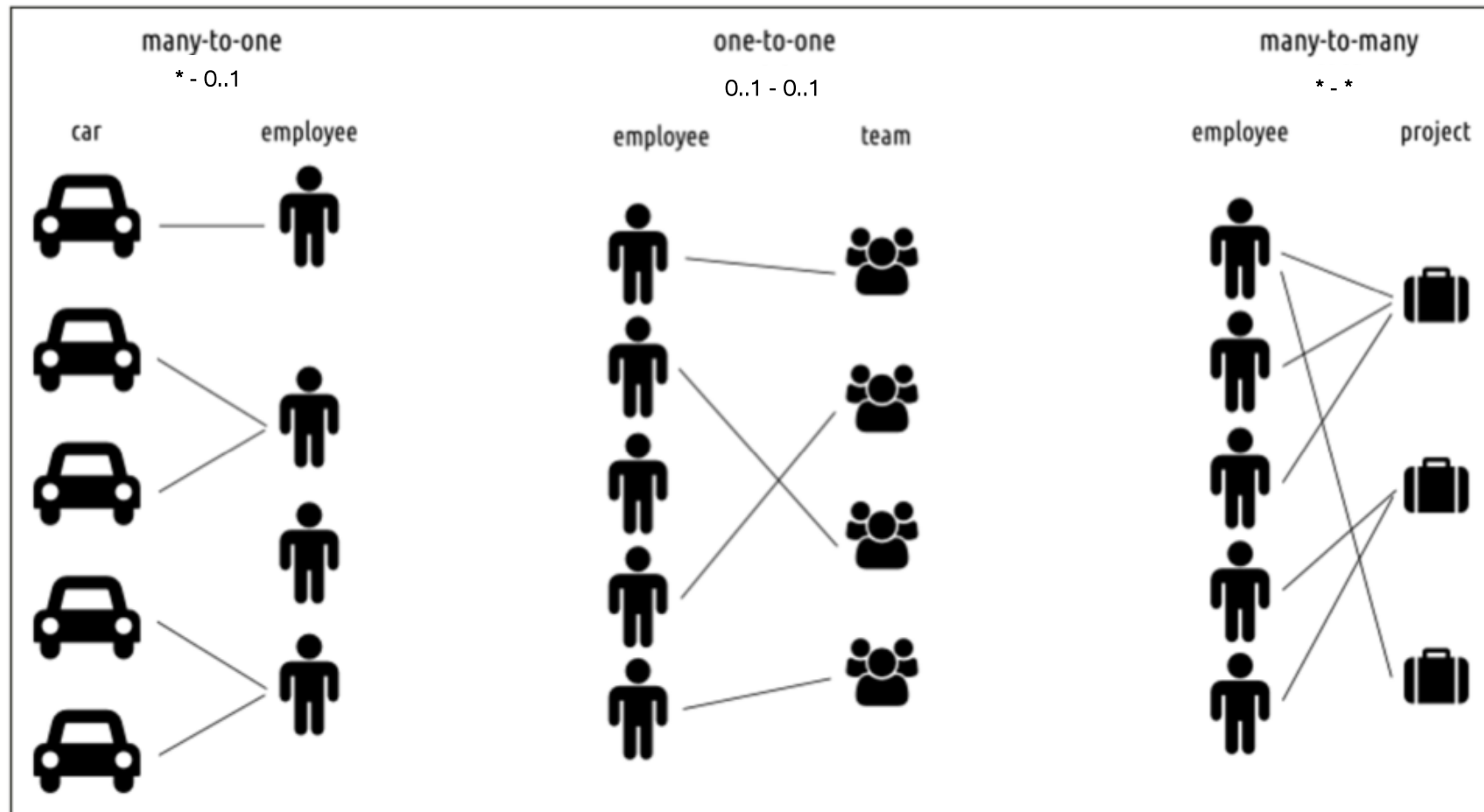   1 stands for 1..1

Default: 1..1

Multiplicity of Associations

Students must apply somewhere and may not apply to more than 5 colleges. No college takes more than 20,000 applications.

Multiplicity of Associations

Credits: André Restivo

Complete Associations

Every object must participate in the association

Complete many-to-one
   1..* - 1..1

Complete one-to-one
   1..1 – 1..1
   Default association

Complete many-to-many
   1..* - 1..*

Kahoot time

Any doubts?

Readings

Jeffrey Ullman, Jennifer Widom, A first course in Database Systems 3rd Edition

    Section 4.7 - Unified Modeling Language