

5. Лекция

Category	Empty
Files	Empty
Created	May 28, 2023 11:22 AM
Reminder	Empty
Status	Open
URL	Empty
Updated	May 28, 2023 11:22 AM

ERD Notation

ERD Cardinality

—	One
—<	Many
—	One (and only one)
—○	Zero or one
—<	One or many
—○<	Zero or many

The domain model class diagram

Class

- type of classification used to describe a collection of objects

Domain class

- classes that describe objects in the problem domain

Class Diagram

- A UML diagram that shows classes with attributes and associations (plus methods if it models software classes)

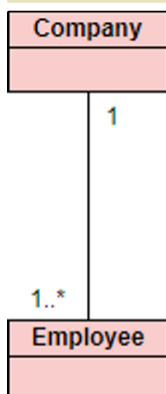
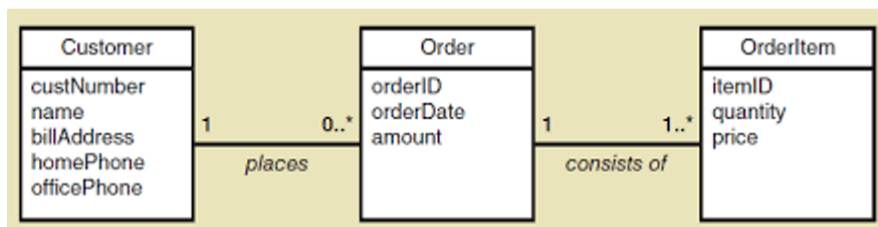
Domain model class diagram

- a class diagram that only includes classes from the problem domain, not software classes so no method
- Тези множества същности, които разглеждахме досега, имат само характеристики/атрибути, а нямат методи като софтуерните класове

UML Domain class notation



Simple domain model class diagram



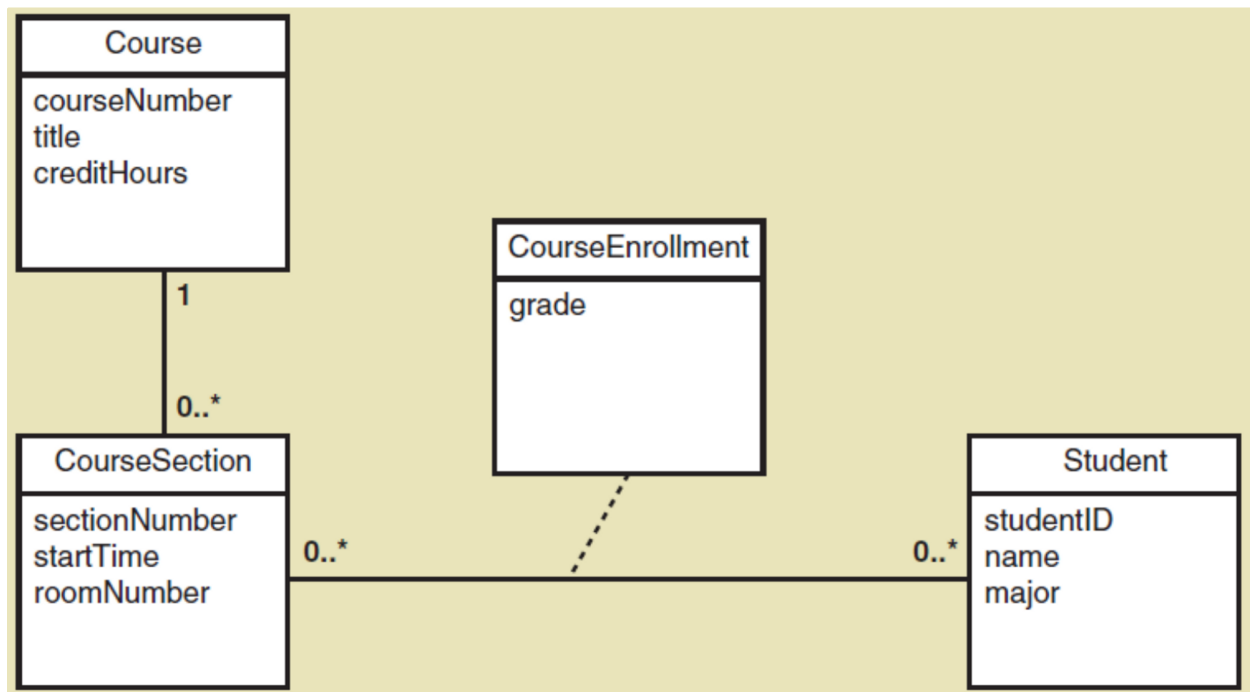
Multiplicities examples:

- | | |
|------|----------------------------------|
| 1 | Exactly one, no more and no less |
| 0..1 | Zero or one |
| * | Many |
| 0..* | Zero or many |
| 1..* | One or many |

<https://www.google.com/url?sa=i&url=https%3A%2F%2Fcourses.acs.uwinnipeg.ca%2F2913-001%2F8-DomainModeling.pdf&psig=AOvVaw1BxKXkPRinJv9pdHS7XO6L&ust=1679491899725000&source=images&cd=vfe&ved=0CBEQjhXqFwoTCMCGi4OR7f0CFQAAAAAdAAAAABAE>

Асоциативни класове

Ако имаме връзка много към много използваме асоциативни класове, за да запазим атрибутите на връзката



Generalization/Specialization Relationships

Generalization/Specialization

- A hierarchical relationship where subordinate classes are special types of the superior classes. Often called an Inheritance Hierarchy

Superclass

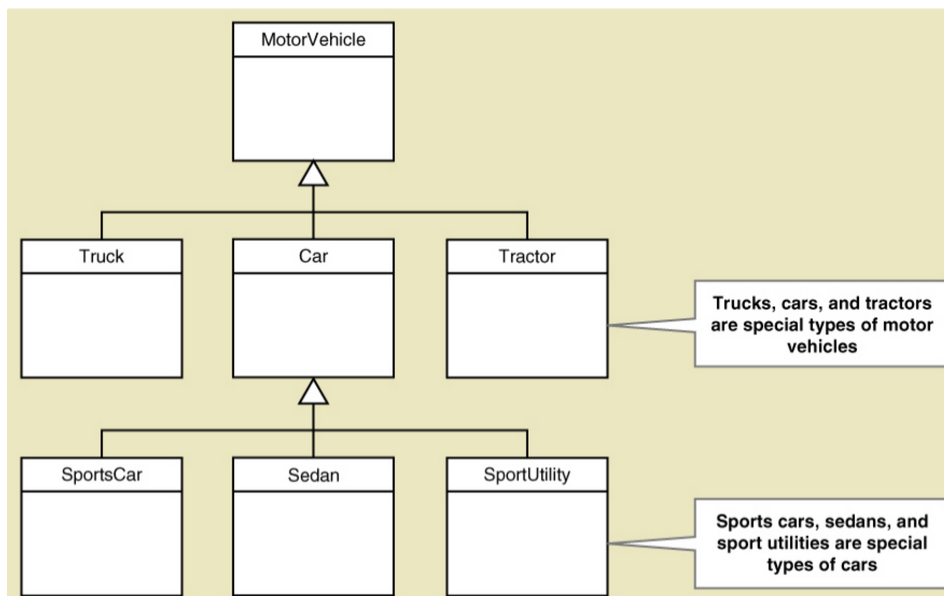
- the superior or more general class in a generalization/specialization hierarchy

Subclass

- the subordinate or more specialized class in a generalization/specialization hierarchy

Inheritance

- the concept that subclasses classes inherit characteristics of the more general superclass

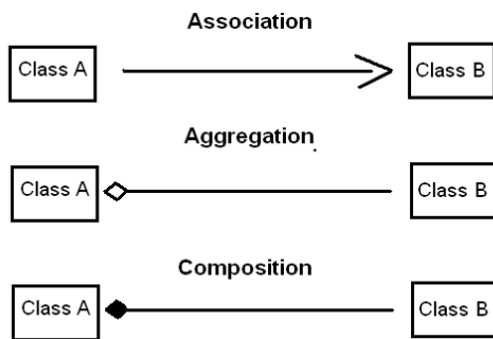


Whole-part relationship

Имаме възможност да посочим, че някакъв обект е част от друг обект

Тази връзка има две различни измерения

1. **Aggregation** - Когато имаме нещо цяло и то се състои от отделни компоненти, но тези компоненти могат да съществуват самостоятелно - примерно компютърна конфигурация с принтер
2. **Composition** - Примерно ръката на човек не може да съществува самостоятелно, друг пример е стая в хотел, няма как да имаме просто стая



UML Relationships

1. Association Relationships - като връзките в
2. Whole-part relationships
 - a. Aggregation
 - b. Composition
3. Generalizations/specialization Relationships
 - a. Inheritance

UML Diagrams

1. Structure diagrams

Описваме структурата на информацията - свързана с E/R модела

Къде е инфомрацията, структурата на самото приложение
2. Behavior Diagram
 - a. Use case diagram
 - b. Activity diagram - стъпка по стъпка какво правят ползвателите
 - c. Interaction diagram - диаграми на взаимодействие - описват интерфейса, комуникацията...