Creating UML Class Diagrams

Software Design (40007) – 2023/2024

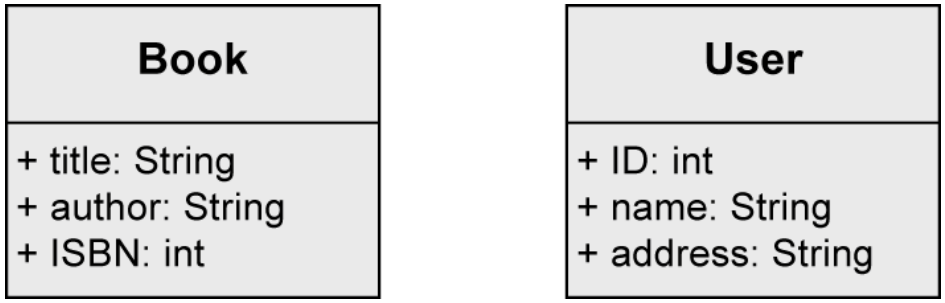
Justus Bogner
Ivano Malavolta

Software and Sustainability research group (S2)
Department of Computer Science, Faculty of Sciences

Creating a class diagram

- Usually not possible to extract classes, attributes, and associations from natural language automatically
- Guidelines
  - **Nouns** often indicate classes, but can also be attributes
  - **Adjectives** often indicate attribute values
  - **Verbs** often indicate operations or relationships

- Example: the library management system stores users with their unique ID, name, and address as well as books with their title, author, and ISBN number. Ann Foster wants to use the library.



Question: What about Ann Foster?

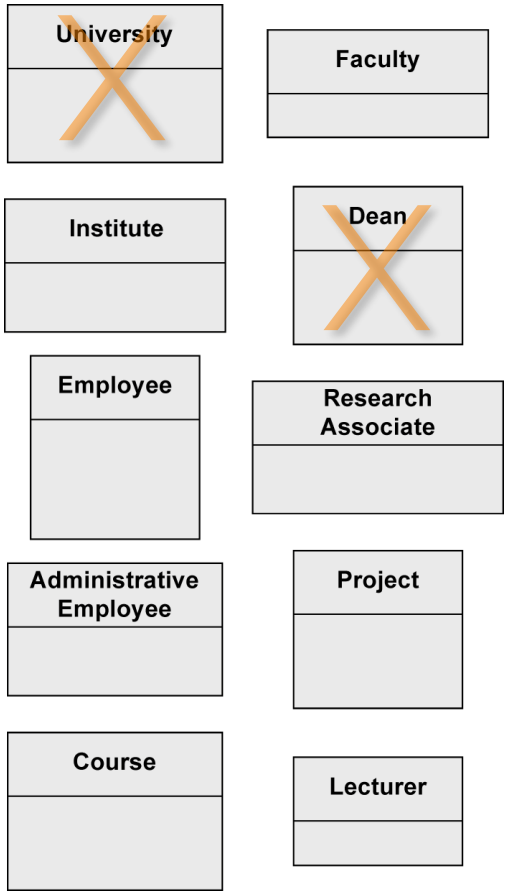
Example – University Information System

- A university consists of multiple faculties, which are composed of various institutes.
- Each faculty and each institute has a name.
- An address is known for each institute.
- Each faculty is led by a dean, who is an employee of the university.
- The total number of employees is known.
- Employees have a social security number, a name, and an email address. There is a distinction between research and administrative personnel.
- Research associates are assigned to at least one institute. The field of study of each research associate is known. Furthermore, research associates can be involved in projects for a certain number of hours, and the name, starting date, and end date of the projects are known.
- Some research associates hold courses. Then they are called lecturers.
- Courses have a unique number (ID), a name, and a weekly duration in hours.

Step 1: identifying classes

- A university consists of multiple faculties, which are composed of various institutes.
- Each faculty and each institute has a name.
- An address is known for each institute.
- Each faculty is led by a dean, who is an employee of the university.
- The total number of employees is known.
- Employees have a social security number, a name, and an email address. There is a distinction between research and administrative personnel.
- Research associates are assigned to at least one institute. The field of study of each research associate is known. Furthermore, research associates can be involved in projects for a certain number of hours, and the name, starting date, and end date of the projects are known.
- Some research associates hold courses. Then they are called lecturers.
- Courses have a unique number (ID), a name, and a weekly duration in hours.

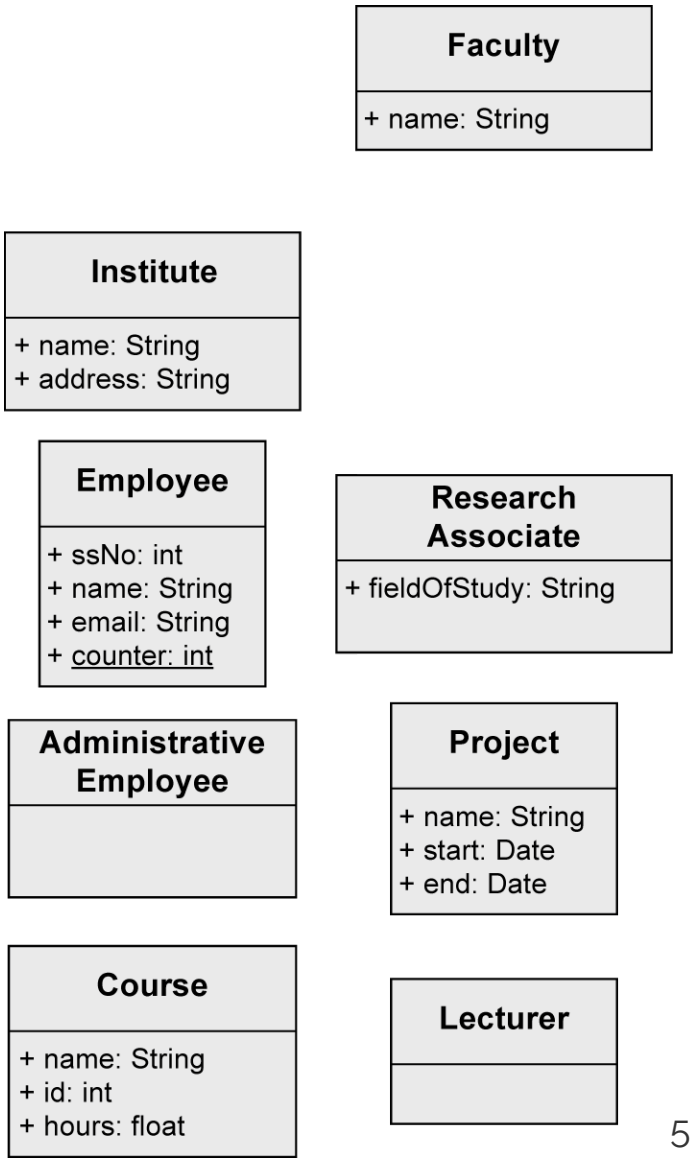
We model the system „University“



Dean has no further attributes than any other employee

Step 2: identifying attributes

- A university consists of multiple faculties which are composed of various institutes.
- Each faculty and each institute has a name.
- An address is known for each institute.
- Each faculty is led by a dean, who is an employee of the university.
- The total number of employees is known.
- Employees have a social security number, a name, and an email address. There is a distinction between research and administrative personnel.
- Research associates are assigned to at least one institute. The field of study of each research associate is known. Furthermore, research associates can be involved in projects for a certain number of hours, and the name, starting date, and end date of the projects are known.
- Some research associates hold courses. Then they are called lecturers.
- Courses have a unique number (ID), a name, and a weekly duration in hours.

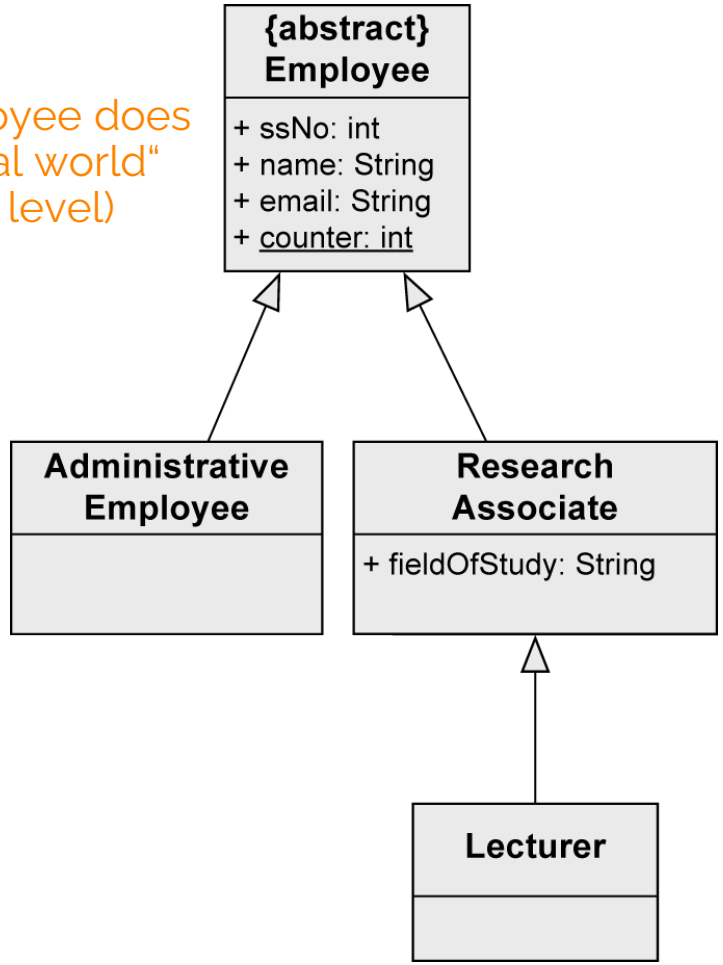


Step 3: identifying relationships (1/6)

Three kinds of relationships:

- Association
- Generalization
- Aggregation

Abstract, i.e., employee does not exist in the „real world“ (i.e., at the instance level)

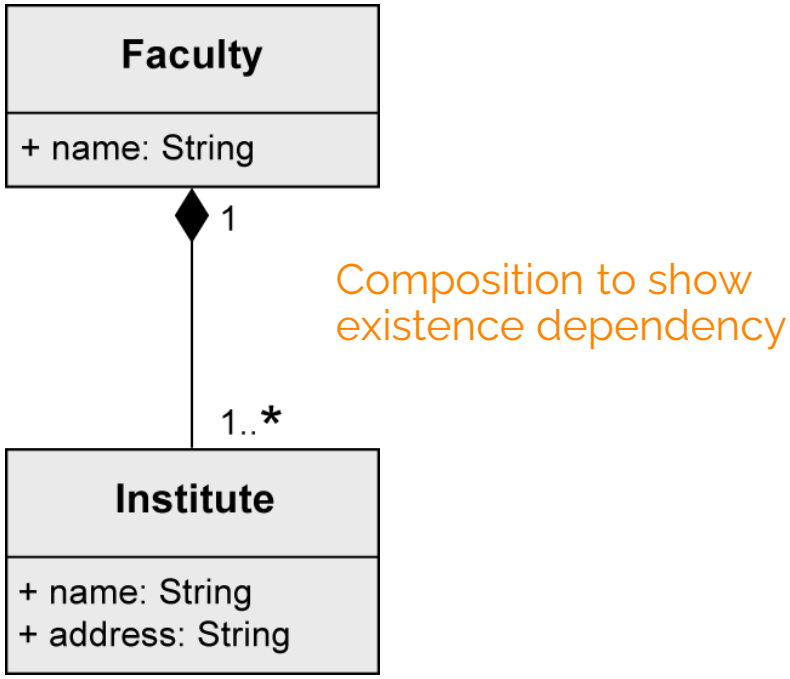


Indication of a generalization

- *“There is a distinction between research and administrative personnel.”*
- *“Some research associates hold courses. Then they are called lecturers.”*

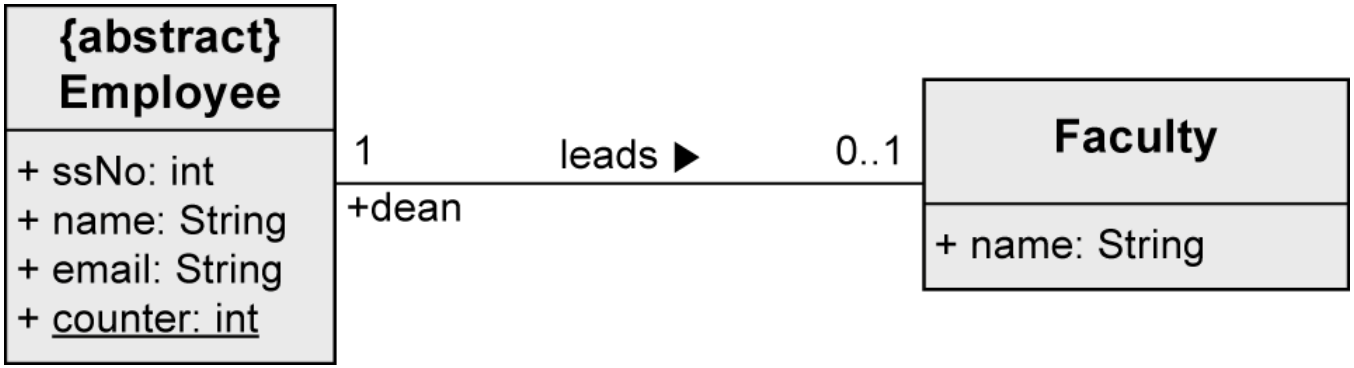
Step 3: identifying relationships (2/6)

"A university consists of multiple faculties which are composed of various institutes."



Step 3: identifying relationships (3/6)

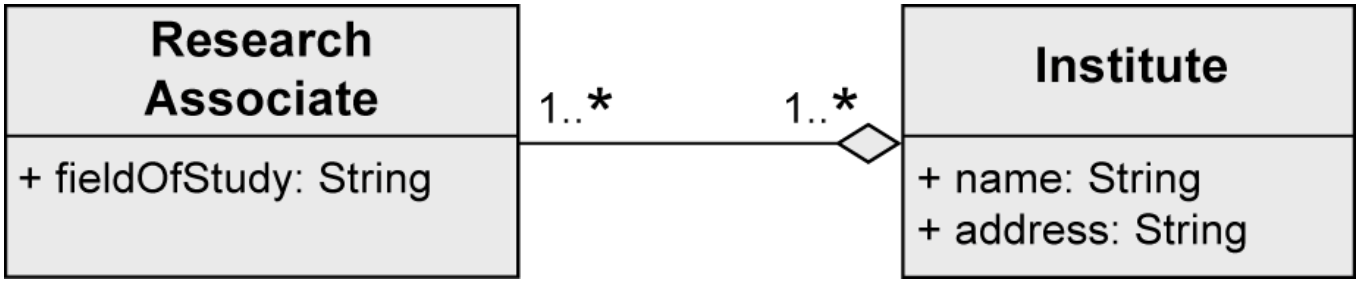
"Each faculty is led by a dean, who is an employee of the university"



In the `leads` relationship, the `Employee` takes the role of a dean.

Step 3: identifying relationships (4/6)

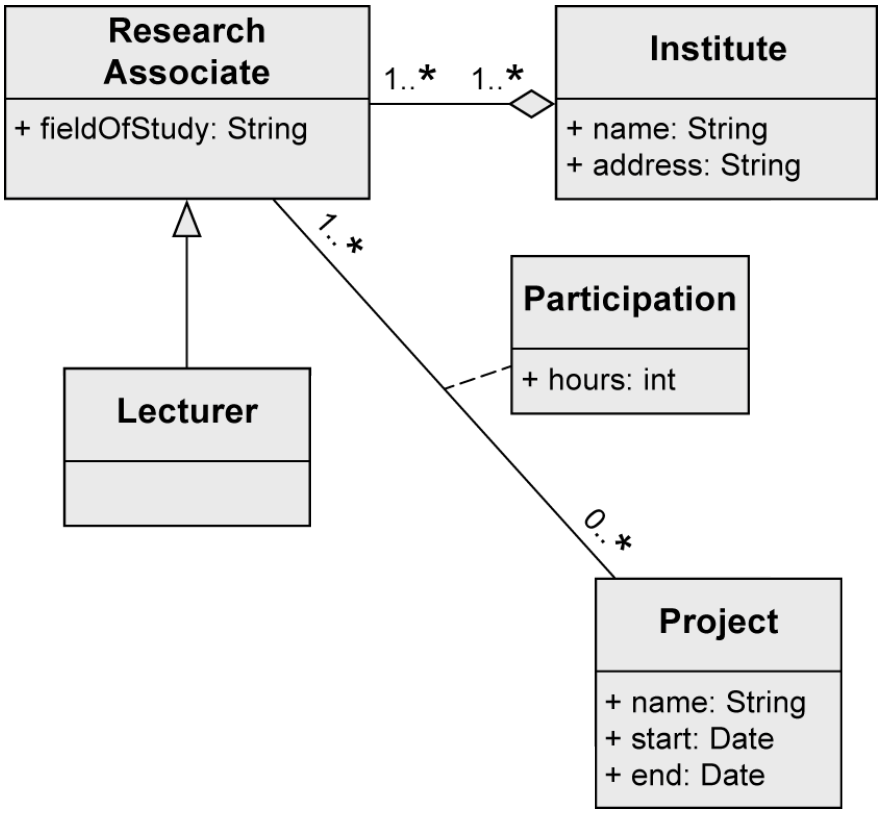
"Research associates are assigned to at least one institute."



Aggregation to show that `ResearchAssociates` are part of an `Institute`, but there is no existence dependency

Step 3: identifying relationships (5/6)

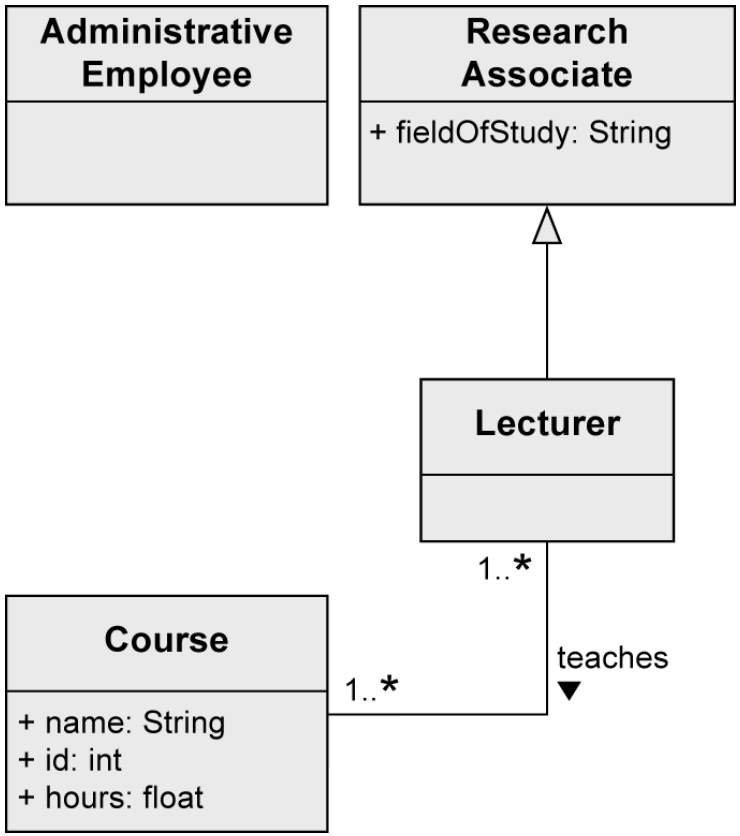
"Furthermore, research associates can be involved in projects for a certain number of hours."



Association class enables to store the number of hours for every `Project` of every `ResearchAssociate`

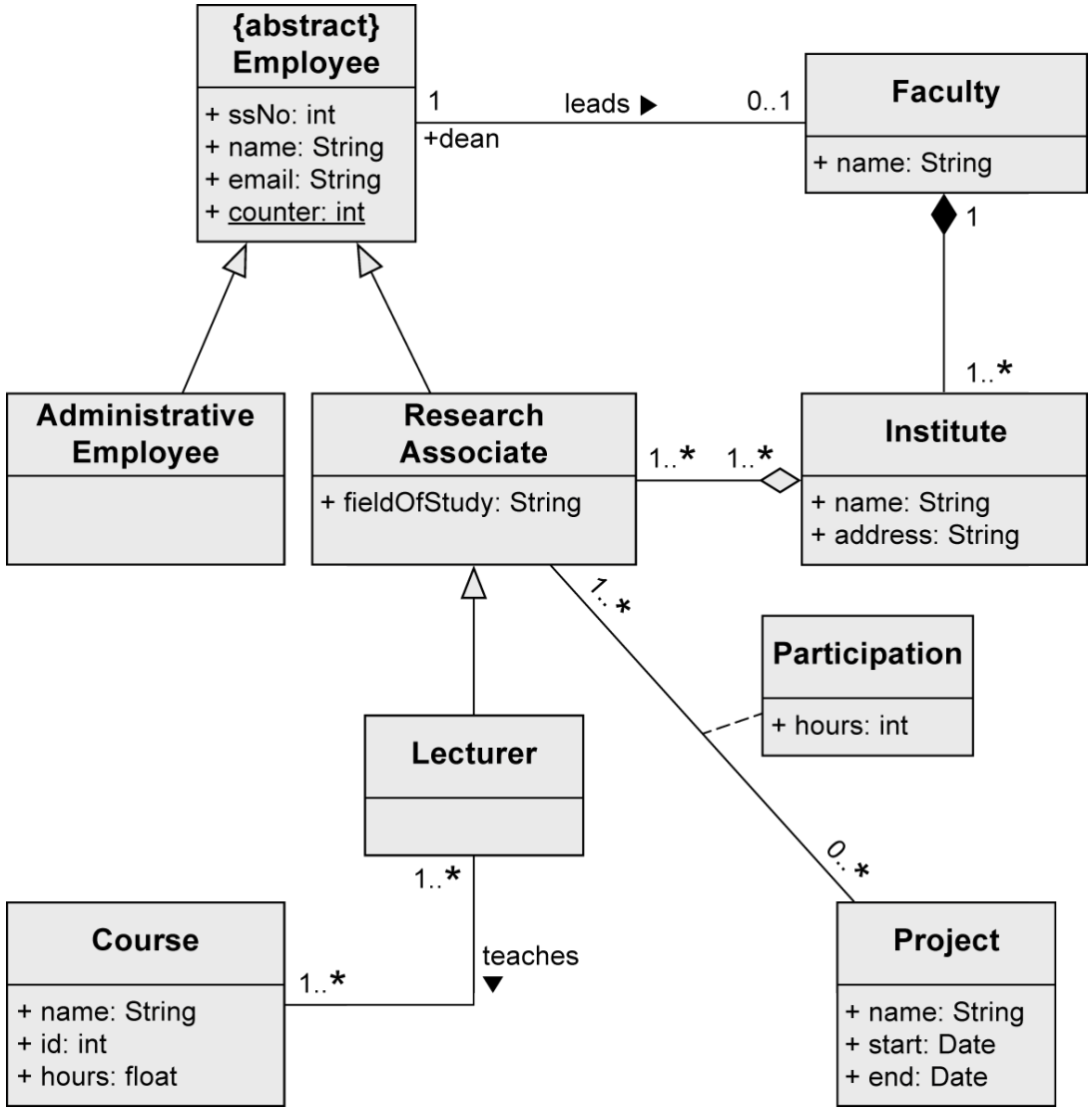
Step 3: identifying relationships (6/6)

"Some research associates hold courses. Then they are called lecturers."

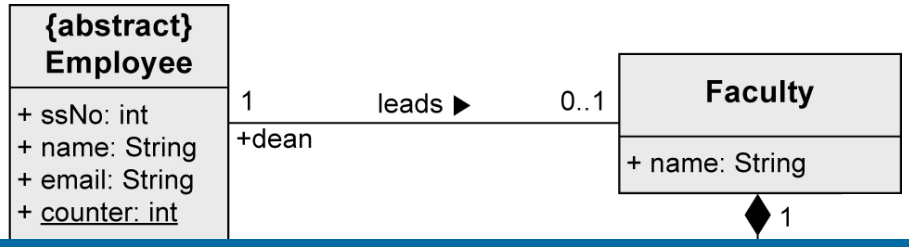


- `Lecturer` inherits all characteristics, associations, and aggregations from `ResearchAssociate`
- `Lecturer` has a `teaches` association with `Course`

Complete class diagram



Complete class diagram



REFLECTION

- There is rarely a single correct solution
- Several acceptable solutions with pros and cons
- Your design decisions depend on many factors, like:
  - Intent
  - Consumer
  - Operational profile of the system
  - …
- Make sure that your solution is consistent with these factors and the system goals