The Australian National University, School of Computing
COMP2400/6240 (Relational Databases)
Semester 2, 2023

Lab 4, Week 5

ER Modeling (Solutions)

In this week's lab, we will use the Entity-Relationship (ER) modeling approach to conduct the conceptual design for a database application.

1 ER Modeling Question

Consider the following data requirements for a university project database:

- Professors have an SSN, a name, an age, a rank, and a research specialty.
- Projects have a project number, a sponsor name, a starting date, an ending date, and a budget.
- Graduate students have an SSN, a name, an age, and a degree program (M.S. or Ph.D.)
- Each project is managed by one professor (known as the project's principal investigator).
- Each project is worked on by one or more professors (known as the project's co-investigators).
- Professors can manage and/or work on multiple projects.
- Each project is worked on by one or more graduate students (known as the project's research assistants).
- When graduate students work on a project, a professor must supervise their work on the project. Graduate students can work on multiple projects, in which case they will have a (potentially different) supervisor for each one.
- Departments have a department number, name, and a main office.

- Departments have a professor (known as the chairman) who runs the department.
- Professors work in one or more departments, and for each department that they work in, a time percentage is associated with their job.
- Graduate students have one major department in which they are working on their degree.
- Each graduate student may have another (usually more senior) graduate student (known as a student advisor) who advises them on what courses to take.

You need to draw an ER diagram to capture the data requirements (if possible) through the following exercises (1)-(5):

Suggested solution:

(1) Identify the entity types.

- PROFESSOR
- PROJECT
- DEPARTMENT
- GRADUATE

(2) Identify the relationship types.

- WORK_IN: PROFESSOR - PROJECT
- MANAGES: PROFESSOR - PROJECT
- SUPERVISES: PROFESSOR - PROJECT – GRADUATE (**a ternary relationship type**[1])
- WORKS: PROFESSOR - DEPARTMENT
- RUNS: PROFESSOR - DEPARTMENT
- MAJOR: DEPARTMENT – GRADUATE
- ADVISES: GRADUATE – GRADUATE (**a recursive relationship type in which two participating entities are of the same type but have different role names**)

[1] In general, there is no need to specify the cardinality ratios for a ternary relationship type.

(3)-(5) We skip attributes and constraints for simplicity. Please refer to the provided ER diagram for details, i.e., see the figure below.

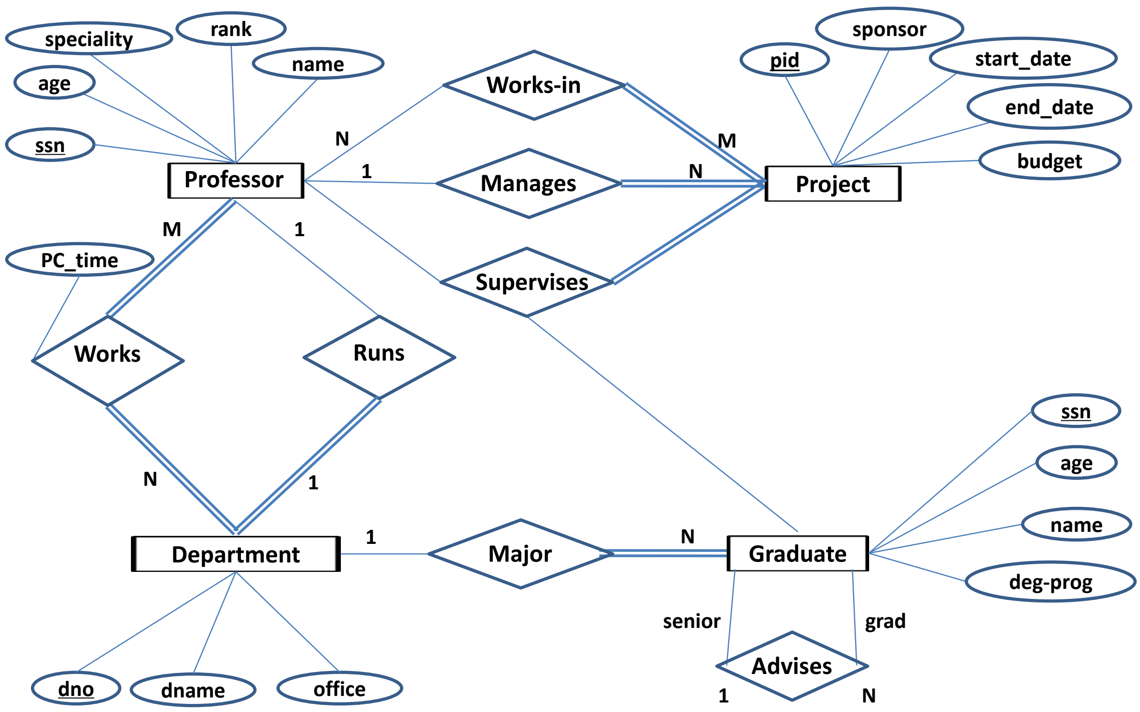

Figure 1: Sample ER diagram

Remarks:

- Identify cardinality ratios (additional assumptions are indicated in bold):
  - Each professor can work on multiple projects and each project is worked on by one or more professors: PROFESSOR - N – WORK_IN - M - PROJECT.
  - Each project is managed by one professor, and one professor can manage many projects: PROFESSOR - 1 – MANAGES - N - PROJECT.
  - Each department has a professor (known as the chairman) who runs the department, **and each professor can chair at most one department**: PROFESSOR – 1 - RUNS - 1 – DEPARTMENT
  - Professors work in one or more departments, **and each department may have several professors**: PROFESSOR – M - WORKS - N – DEPARTMENT.

– Graduate students have one major department in which they are working on their degree, and each department can offer many majors to graduate students: DEPARTMENT – 1 - MAJOR - N – GRADUATE.

– Each graduate student has another, more senior graduate student as a advisor and a graduate student may advise many other graduate students: GRADUATE – 1 (senior) - ADVISES - N (grad)– GRADUATE.

2 Design Choice Questions

The following questions relate to several important design choices you may often come across when designing an ER model.

2.1 Entity versus Attribute

(1) *Consider the database application described in Section 1. Suppose that we want to record addresses of graduate students. One graduate student may have* more than one *address. We also want to capture* the structure of an address *in terms of city, state, country, and postcode. In this case, should the concept address be modeled as an attribute or an entity?*

Suggested solution:

Because more than one address may be recorded for a graduate student and the structure of an address also needs to be captured, address should be modelled as an entity in this case.

2.2 Entity versus Relationship

(2) *Consider the database application described in Section 1. Suppose that we want to further model the information about a discretionary budget given to professors who manage projects. How would you model this in the following two different cases?*

- Case 1: Suppose that each professor is given a discretionary budget for *each* project they manage, and the budgets may vary in different projects for the same professor.
- Case 2: Suppose that each professor is given a discretionary budget for *all* projects they manage, regardless of how many projects the professor manages.

Suggested solution:

- For Case 1, we may model it by a relationship between two entities PROFESSOR and PROJECT, and such a relationship may have budget as its attribute.
- For Case 2, we may add budget as an attribute of the entity PROFESSOR.

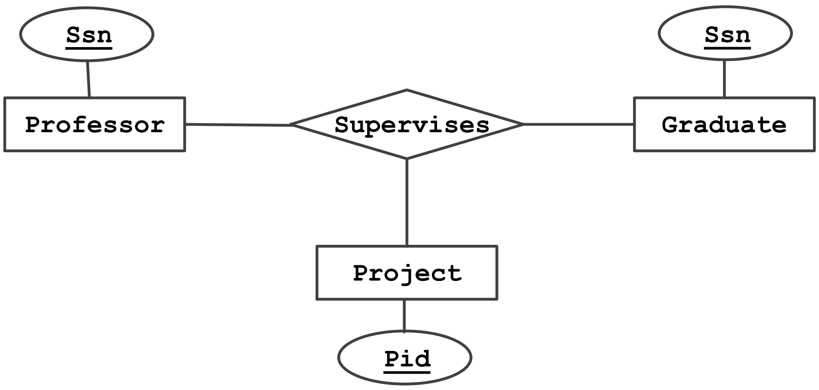

Figure 2: Ternary relationship

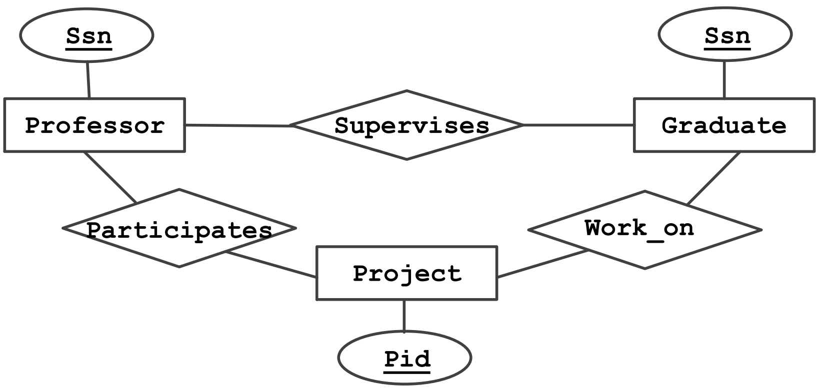

Figure 3: Binary relationship

2.3 Binary versus Ternary Relationships

(3) *Consider the database application described in Section 1, and two ER models presented in Figure 2 and Figure 3. For the relationships between professors, projects and* *graduate students, which of the ER models do you prefer to choose? Explain the difference between the data requirements captured by these two ER models.*

Suggested solution:

In this case, a ternary relationship among PROFESSOR, PROJECT and GRADUATE is preferred. Because each project is worked on by one or more professors, a professor can work on multiple projects, each project is worked on by one or more graduate students, and a graduate student can work on multiple projects, the binary relationships depicted in Figure 3 cannot precisely specify the relationship "*when graduate students work on a project, a professor must supervise their work on the project*".

Moreover, a ternary relationship can capture specific relationships among PROFESSOR, PROJECT and GRADUATE which may not be captured by three binary relationships. For example, we can project the following ternary relationship SUPERVISION into three binary relationships as shown below.

SUPERVISION (TERNARY)		
Professor	Project	Graduate
X	1	A
Y	1	B
X	2	B
Y	2	A

PARTICIPATES (BINARY)	
Professor	Project
X	1
Y	1
X	2
Y	2

SUPERVISES (BINARY)	
Professor	Graduate
X	A
Y	B
X	B
Y	A

WORKS_ON (BINARY)	
Project	Graduate
1	A
1	B
2	B
2	A

The ternary relationship can specify the following two senarios:

- Professor X supervises Graduate A on Project 1 (refering to the first row in the ternary relationship SUPERVISION)
- Professor X does not supervise Graduate A on Project 2 (no such row in the ternary relationship SUPERVISION)

However, the above binary relationships cannot model/distinguish these two senarios.