CS275 – Intro to Databases

ER Diagrams- *Chap. 7*

Using High-Level Conceptual Data Models for Database Design

- **Requirements collection and analysis**
  - Database designers interview prospective database users to understand and document data requirements
  - Result: **data requirements**
  - **Functional requirements** of the application

Using High-Level Conceptual Data Models (cont'd.)

- **Conceptual schema**
  - Conceptual design
  - Description of data requirements
  - Includes detailed descriptions of the entity types, relationships, and constraints
  - Transformed from high-level data model into implementation data model

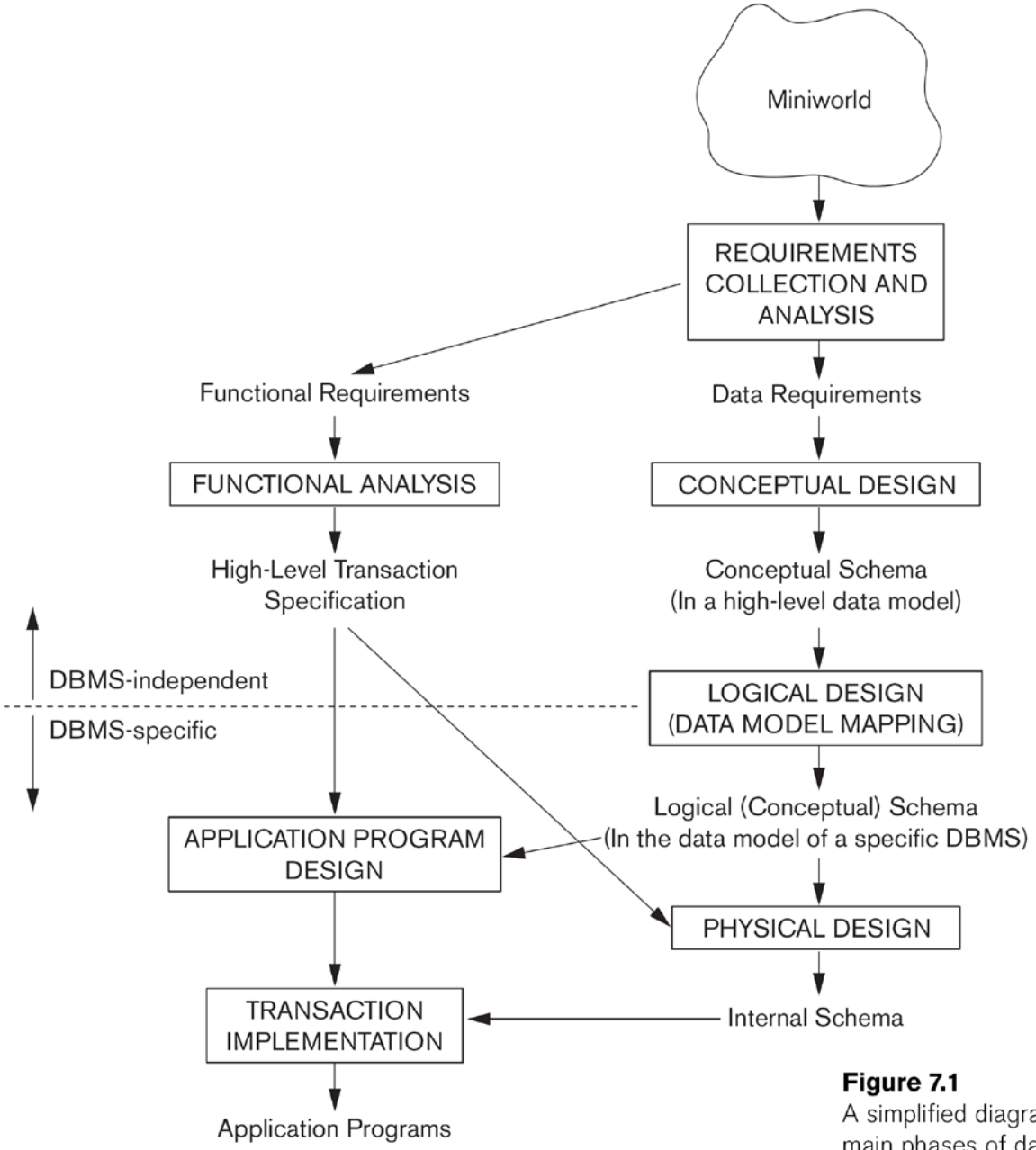


Figure 7.1
A simplified diagram to illustrate the main phases of database design.

Entity-Relationship Model (ER Model)

- ER Model is a popular high-level conceptual data model
  - It is also frequently used for the conceptual design of database applications
  - Many database design tools employ its concepts
- ER diagrams are diagrammatic notations associated with ER models

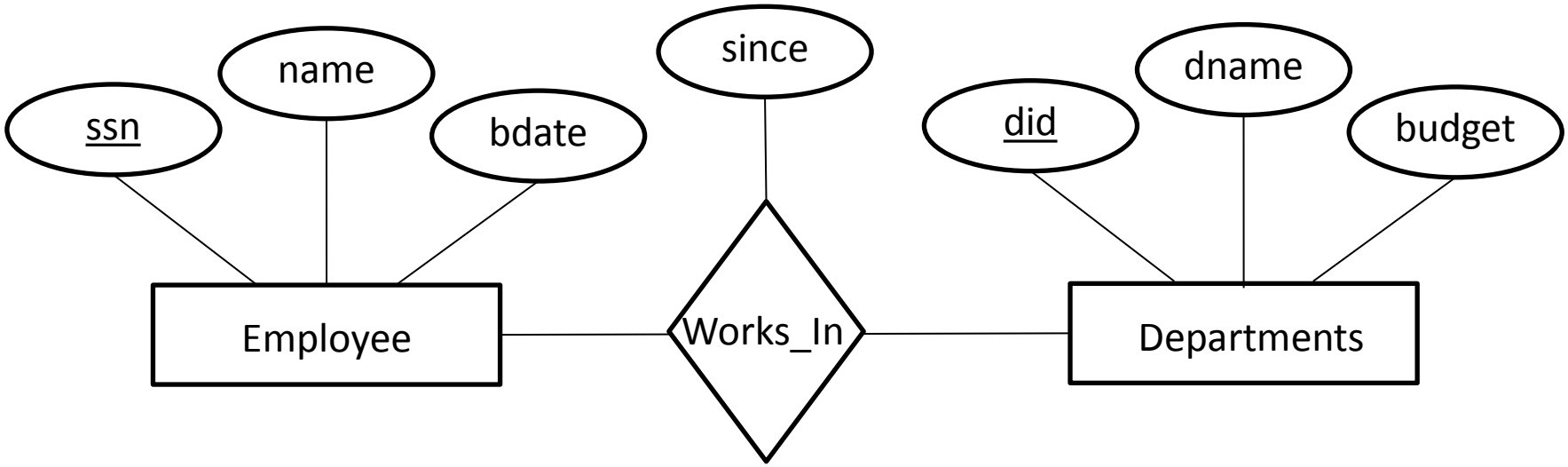
Entity-Relationship Model (ER Model)

- An **entity** is a thing in the real world with an independent existence
- **Attributes** of an entity describe its properties
  - **Composite** versus **simple** (atomic) attributes
  - **Single-valued** versus **multivalued** attributes
  - **Stored** versus **derived** attributes
  - **NULL** values
  - **Complex** attributes- ()composite,{}multivalued

- **Relationships**

Relationships

- An association b/w at least two entities
  - Descriptive attributes
  - Instance



Ex. COMPANY Data Requirements

- The company is organized into departments. Each dept. has a unique name, unique number and a particular employee who manages the dept. We keep track of the start date when that employee began managing the dept. A dept. many have several locations.
- A dept. controls a number of projects, each of which has a unique name, unique number, and a single location.
- We store each employee's name, ssn, address, salary, gender, and birth date. An employee is assigned to one dept., but may work on several projects, which are not necessarily controlled by the same dept. We keep track of the current number of hours per week that an employee works on each project. We also keep track of the direct supervisor of each employee (who is another employee).
- We want to keep track of the dependents of each employee for insurance purposes. We keep each dependent's first name, gender, birth date, and relationship to employee.

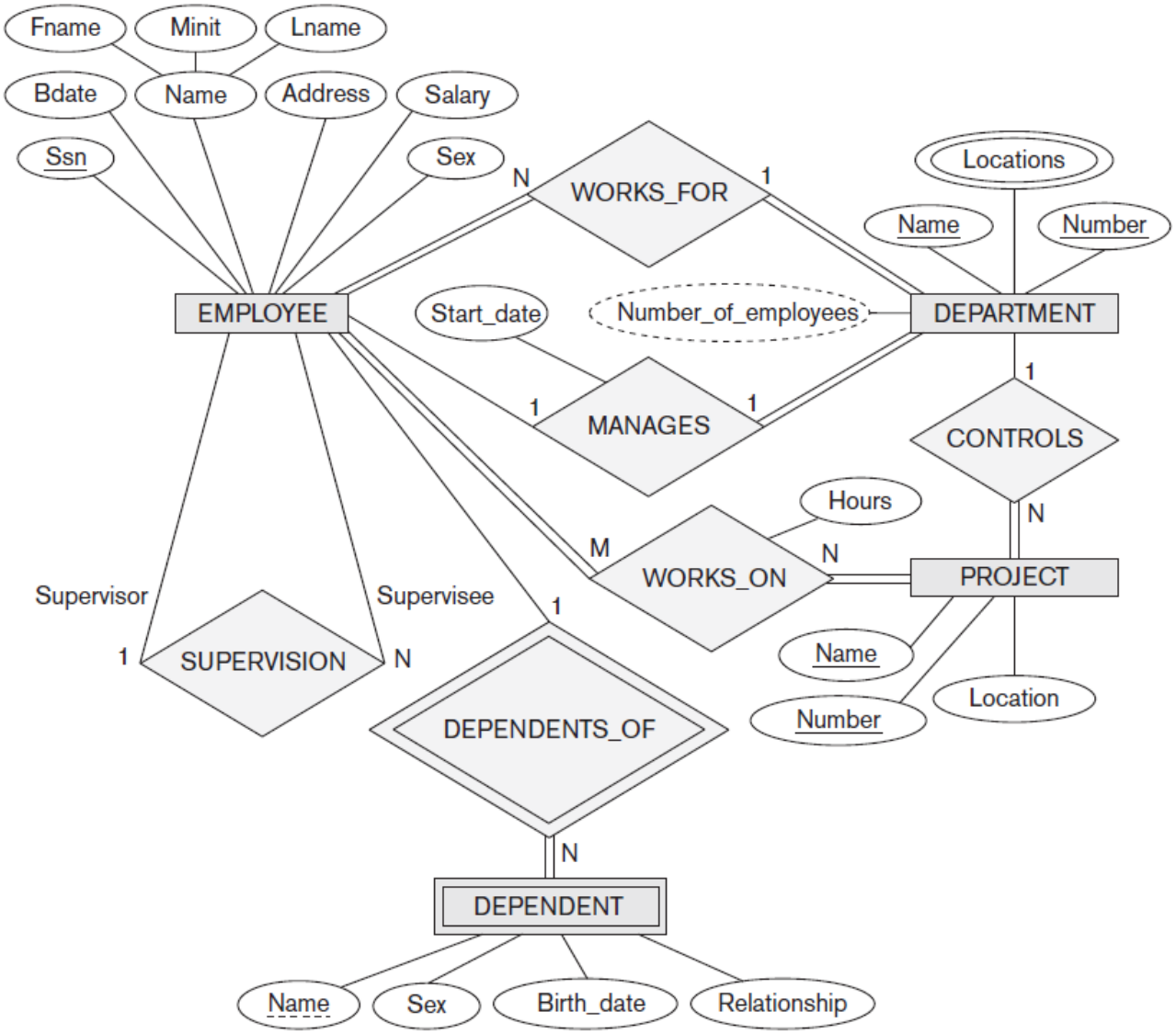


Figure 7.2
An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter and is summarized in Figure 7.14.

ER Diagrams, Conventions, & Design

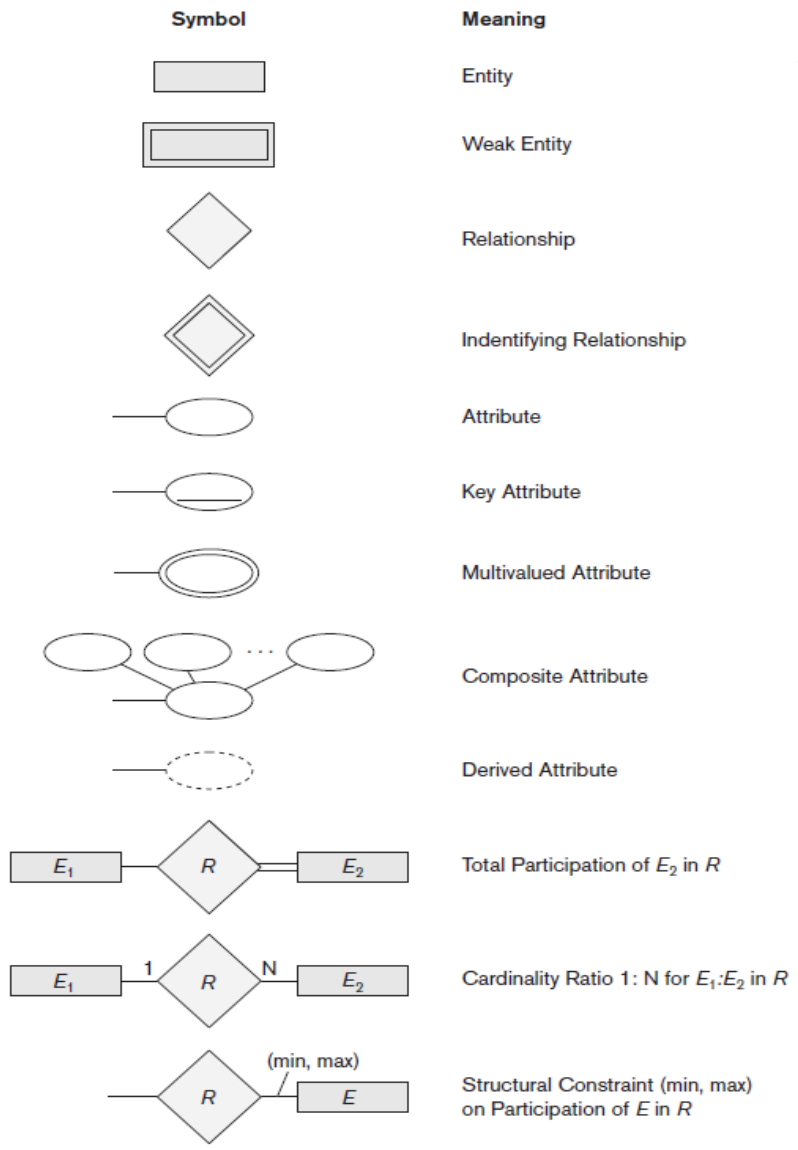
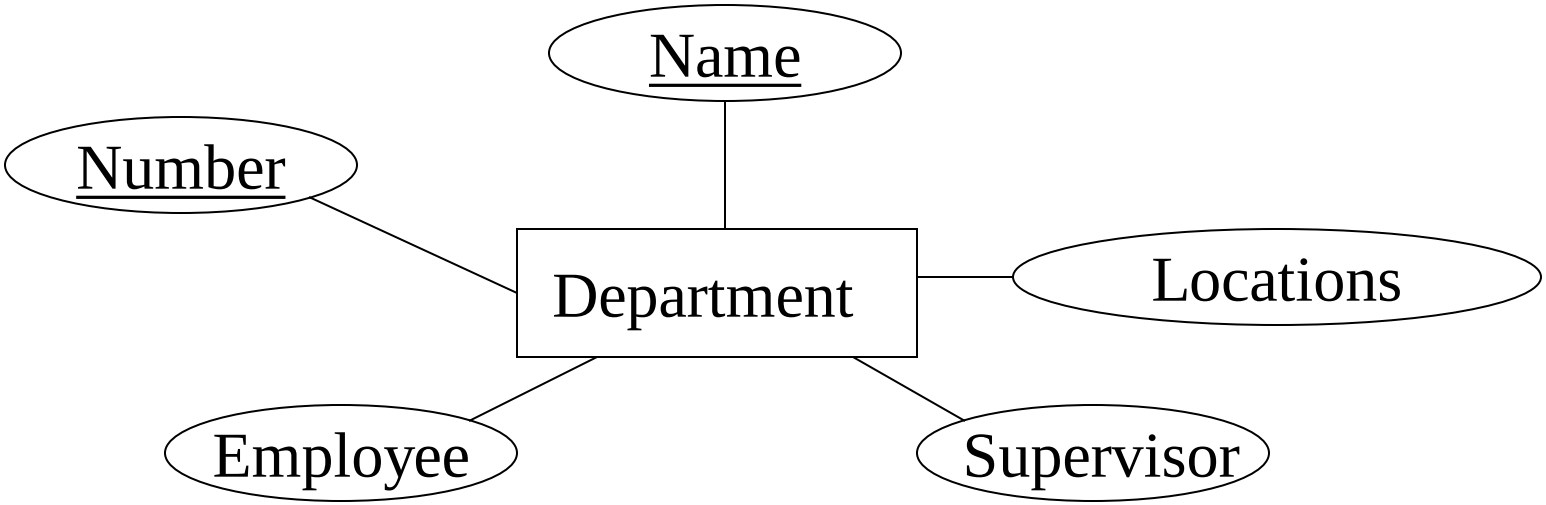


Figure 7.14
Summary of the notation for ER diagrams.

Key Attributes

- An entity type is represented in ER diagrams as a rectangle box
- Attribute names are enclosed in ovals
- Key attributes are shown with underlines



Initial Conceptual Design of the COMPANY Database

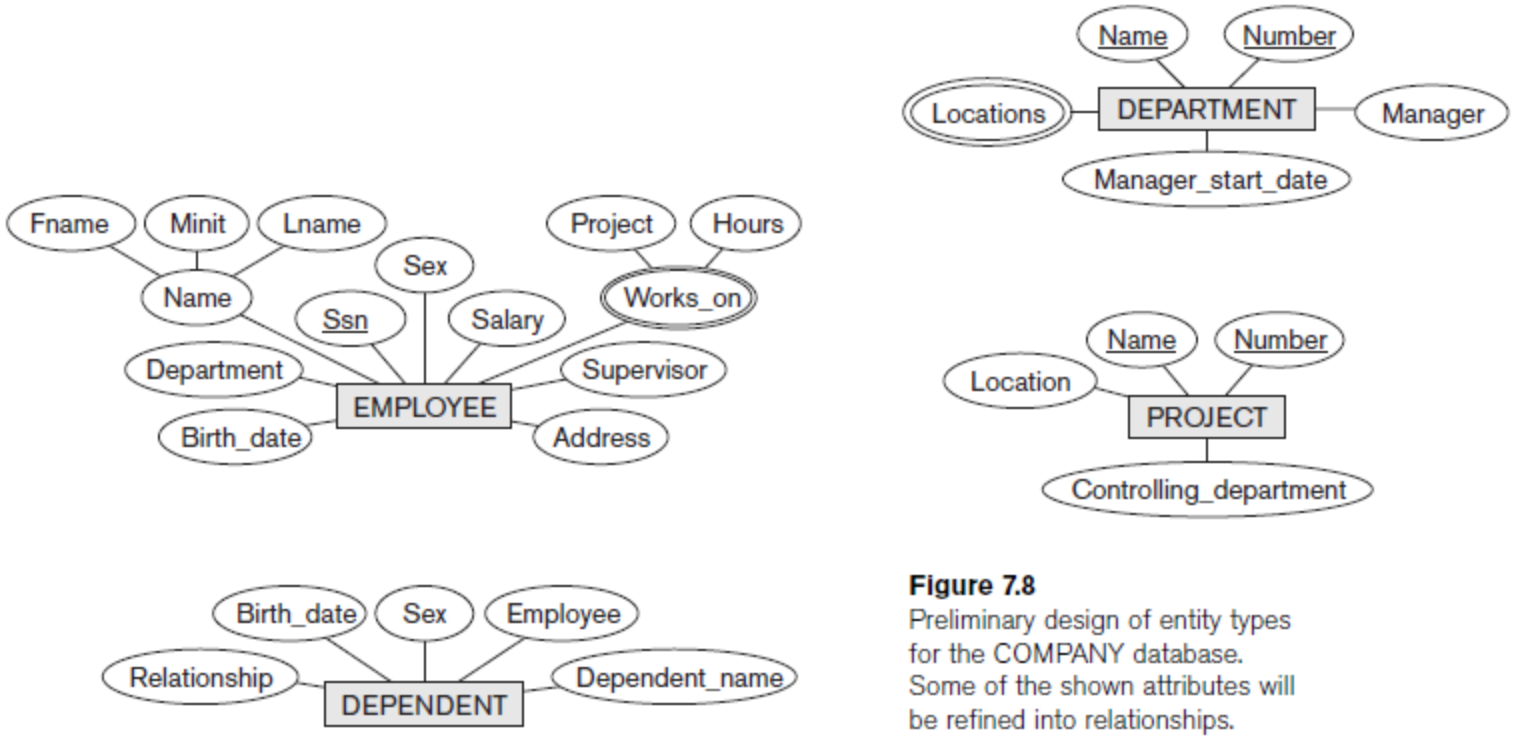
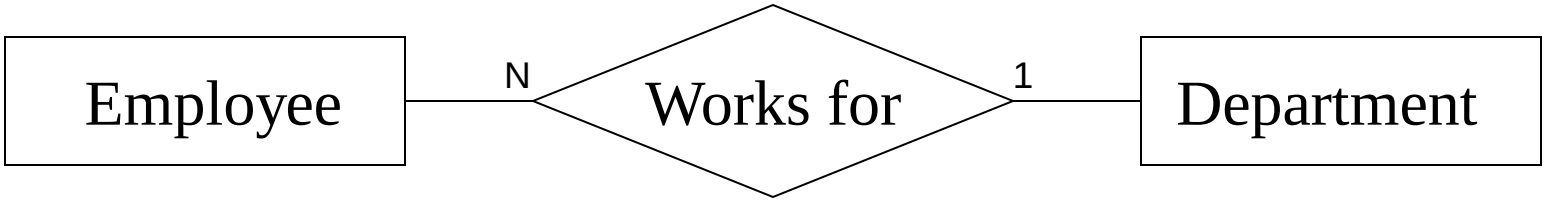


Figure 7.8
Preliminary design of entity types for the COMPANY database. Some of the shown attributes will be refined into relationships.

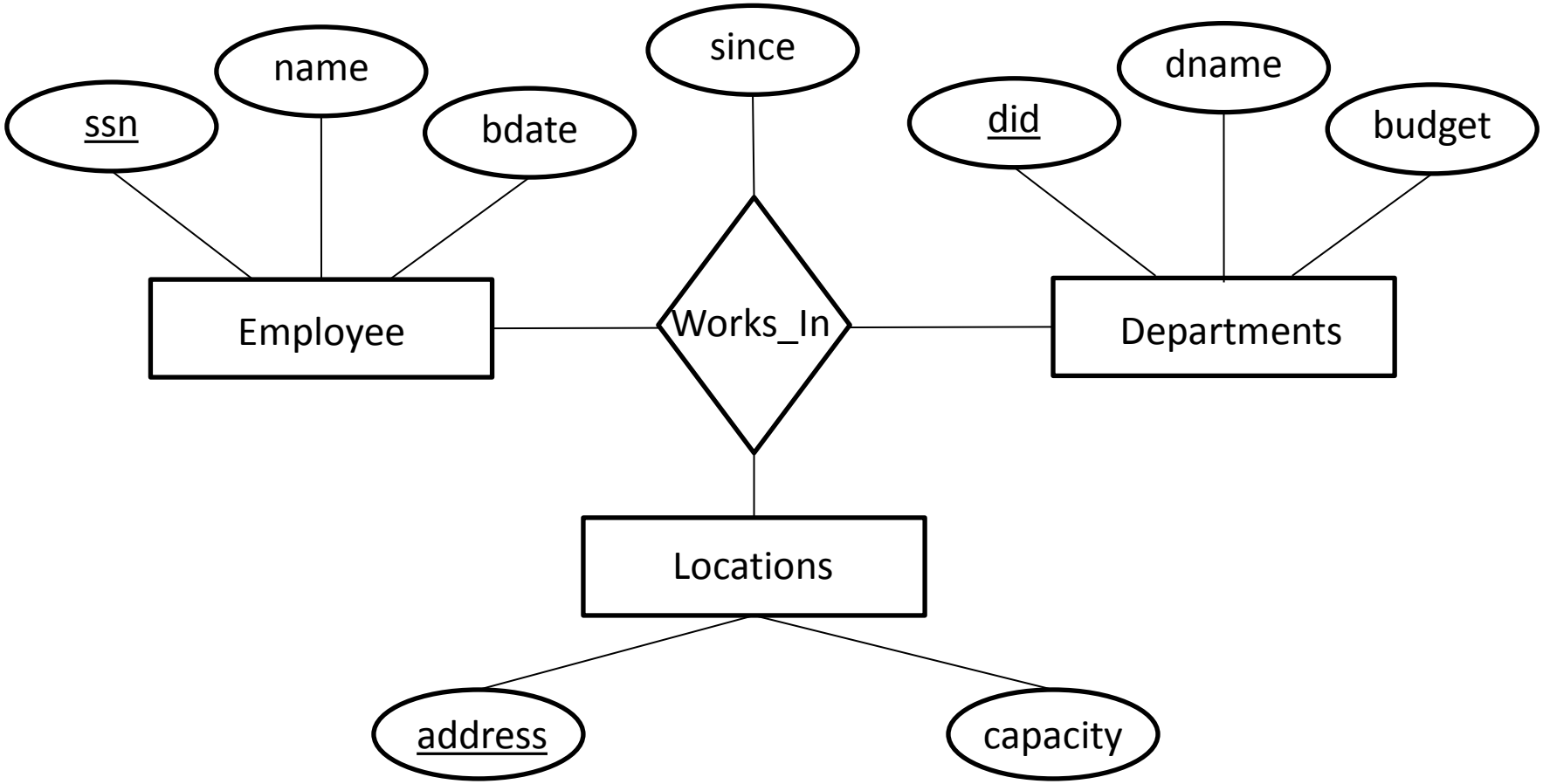
Entity-Relationship Model (ER Model)

- A relationship type R among n entity types E1, …, En defined a relationship set
  - Each individual instance is a relationship instance
  - Displayed as diamond-shaped boxes in ER diagrams
    - The degree: number of participating entities
    - Cardinality ratios for binary relationship: 1:N, M:N, 1:1



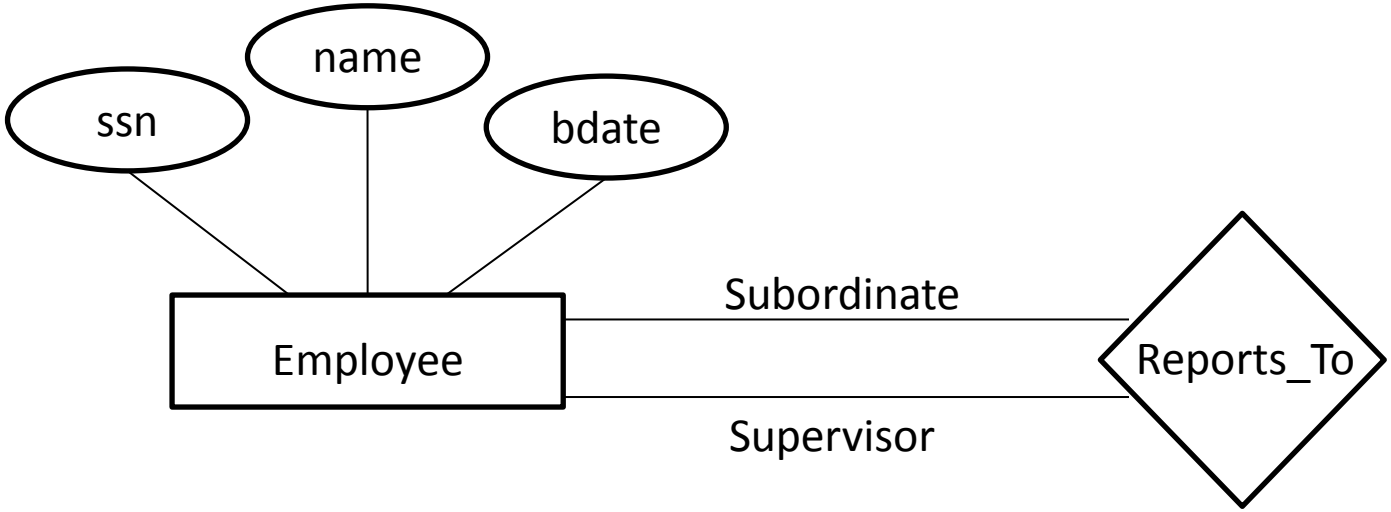
Entity-Relationship Model (ER Model)

- Ternary Relationship Set



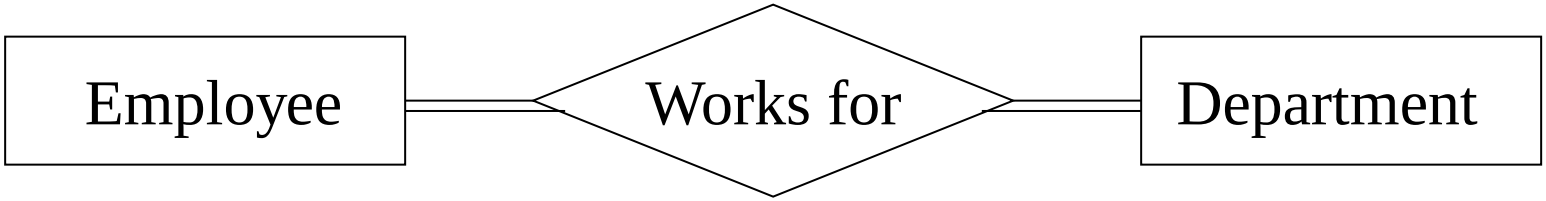
Entity-Relationship Model (ER Model)

- Roles Relationship Set
  - Roles indicator



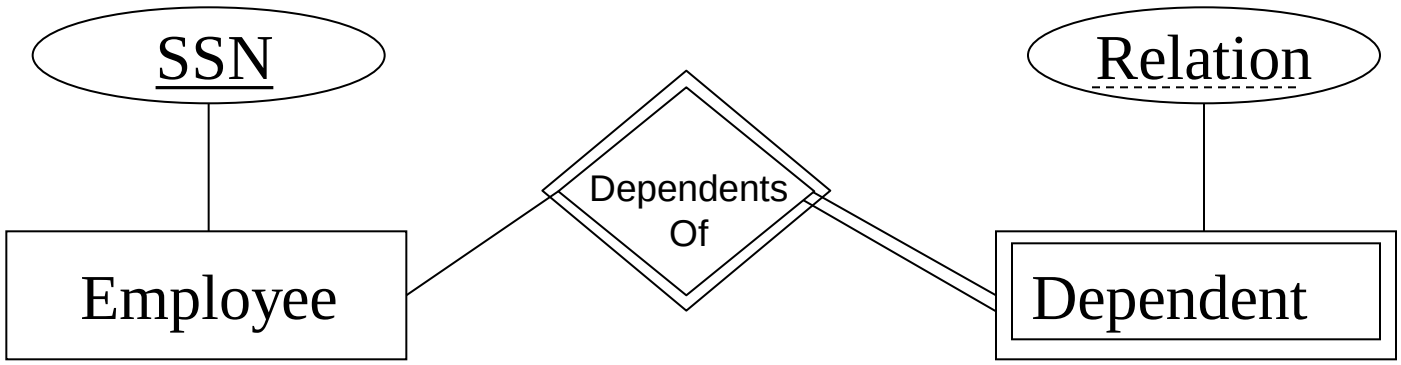
Entity-Relationship Model (ER Model)

- Participation constraints
  - Total: every employee must work for a department (employee-department-work_for)
  - Partial: not every employee manages a department (employee-department-manage)

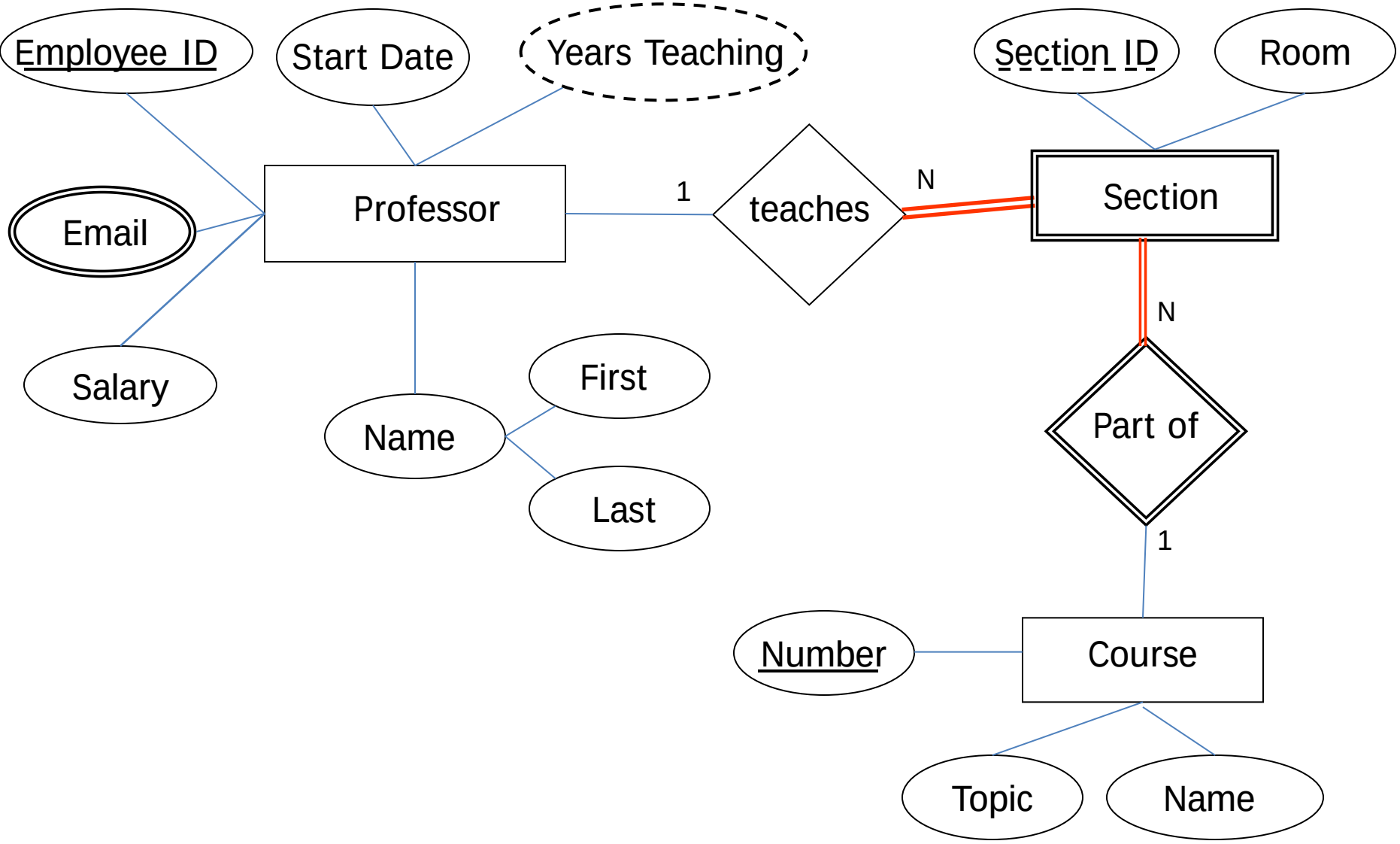


Entity-Relationship Model (ER Model)

- Weak entity types
  - Cannot be identified without an owner entity
  - Has a partial key to identify a record once the owner's key is given
  - Weak entity types can sometimes be represented as complex attributes (multi-valued, composite)



ER Diagrams


Employee ID
Start Date
Years Teaching
Section ID
Room
Email
Professor
1
teaches
N
Section
Salary
First
Name
Last
N
Part of
1
Number
Course
Topic
Name

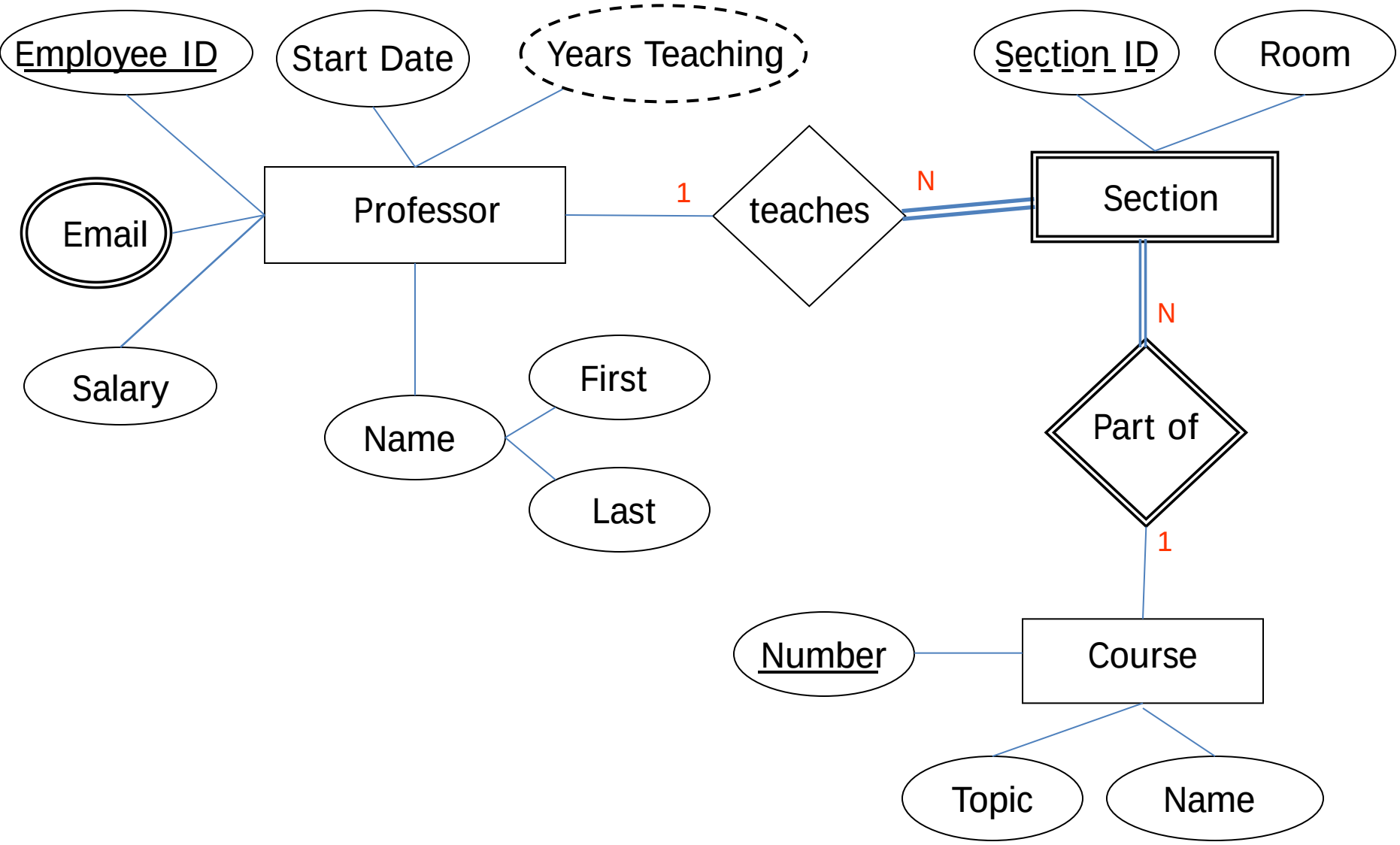
Attributes of Relationship Types

- Attributes of 1:1 or 1:N relationship types can be migrated to one entity type
- For a 1:N relationship type
  - Relationship attribute can be migrated only to entity type on N-side of relationship
- For M:N relationship types
  - Some attributes may be determined by combination of participating entities
  - Must be specified as relationship attributes

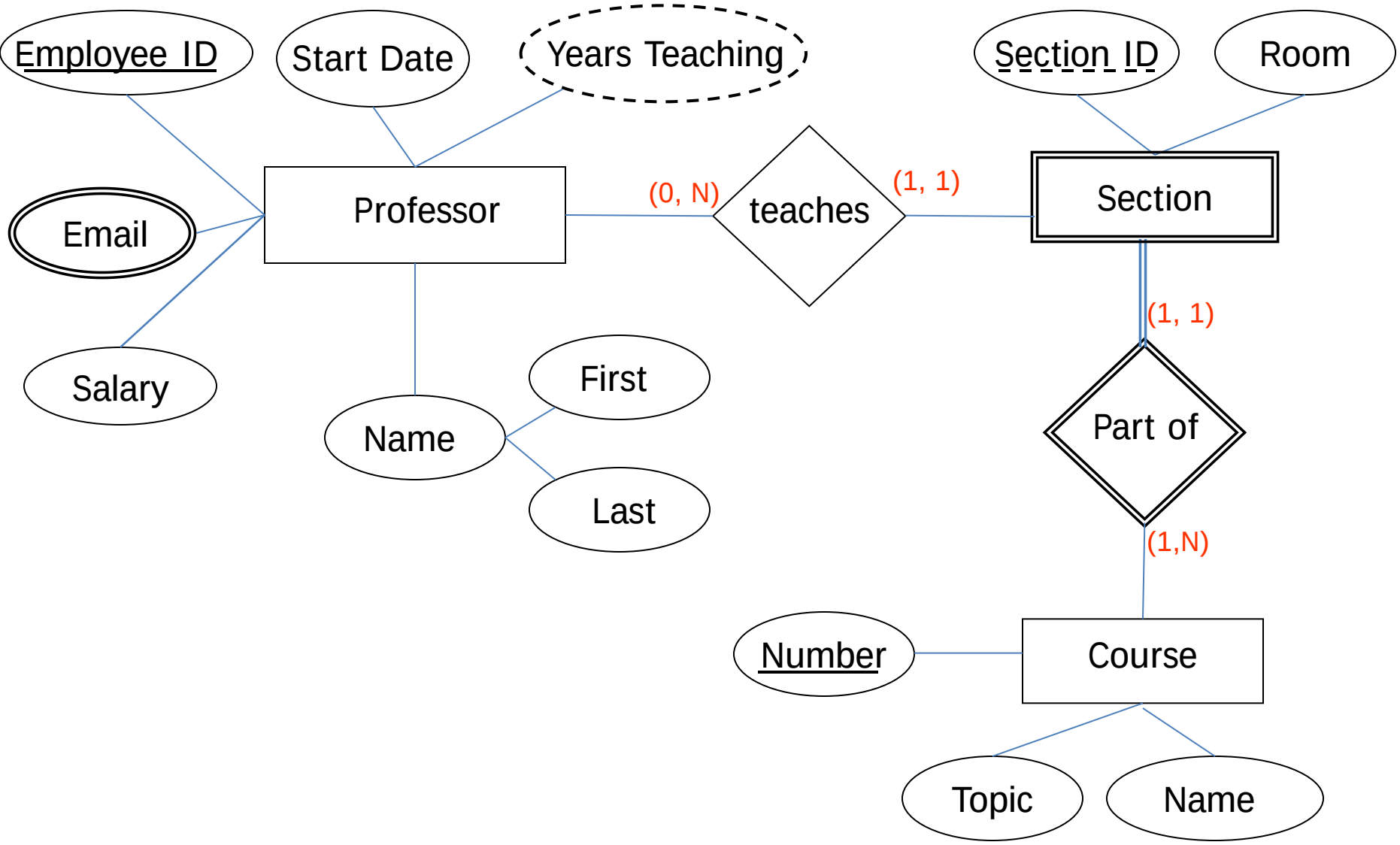
ER Modeling

- Another way of specifying cardinality and participation
  - (min, max)

ER Diagrams


Employee ID
Start Date
Years Teaching
Section ID
Room
Email
Professor
1
teaches
N
Section
Salary
First
Name
Last
N
Part of
1
Number
Course
Topic
Name

ER Diagrams


Employee ID
Start Date
Years Teaching
Section ID
Room
Professor
(0, N)
teaches
(1, 1)
Section
Email
Salary
First
Name
Last
(1, 1)
Part of
(1,N)
Number
Course
Topic
Name

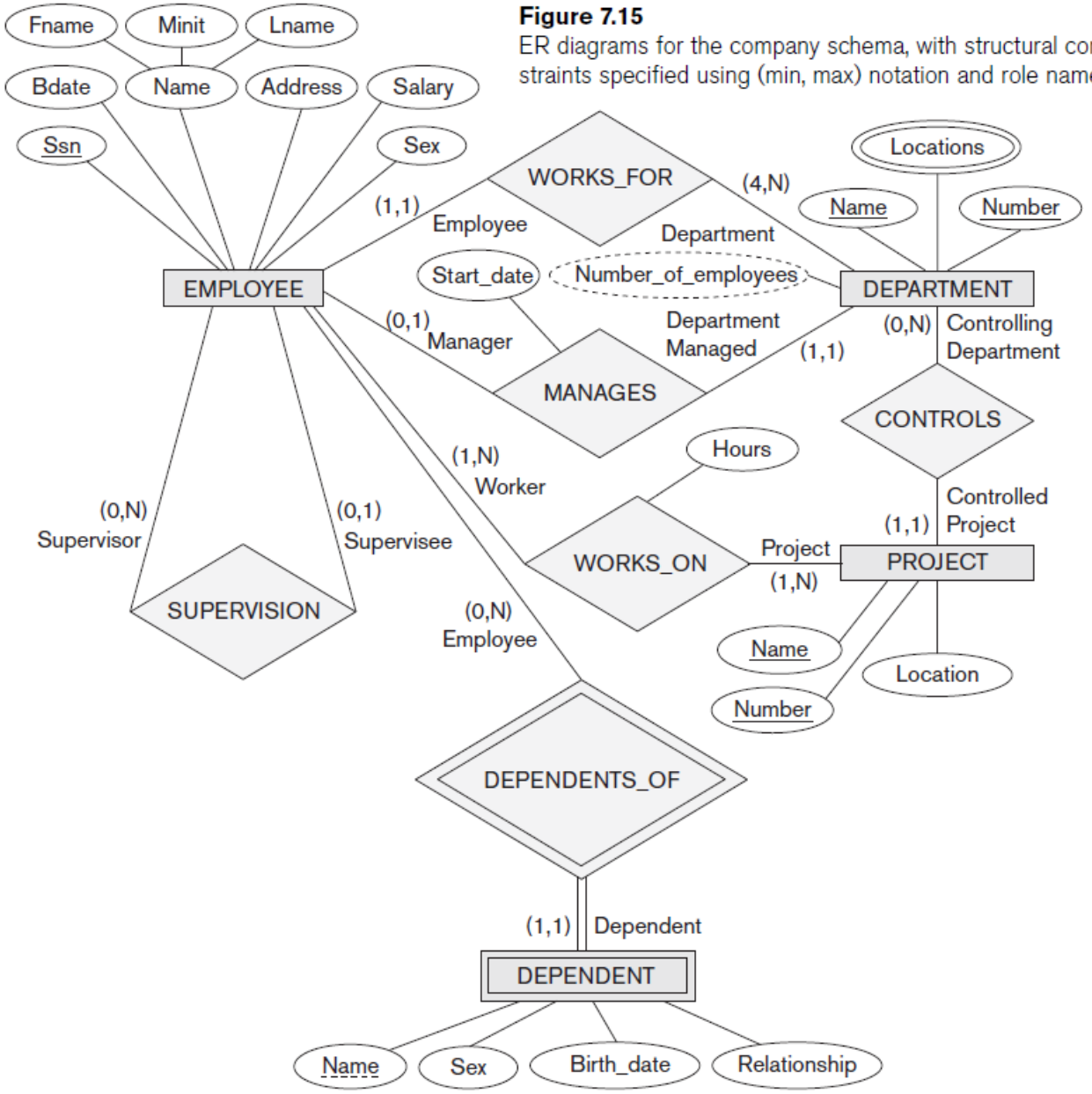


Figure 7.15
ER diagrams for the company schema, with structural constraints specified using (min, max) notation and role names.