Data Modeling using the Entity-Relationship Model

Fundamental of Database System (Edition 7)

Chapter 3

ER Model (only do Chen notation …

skip the min-max notation)

Database Design Process

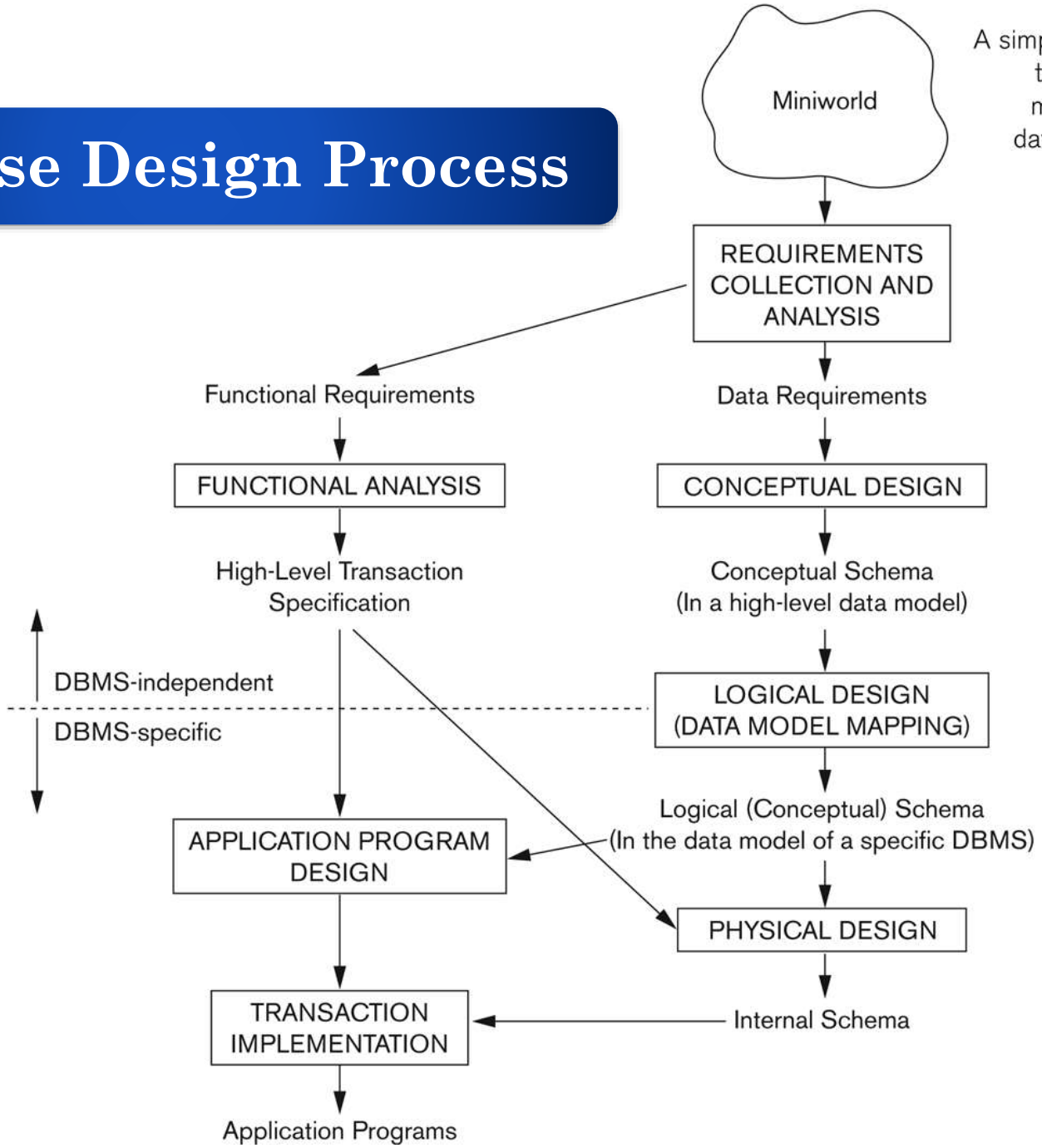


Figure 3.1
A simplified diagram to illustrate the main phases of database design.

ER Model Concepts

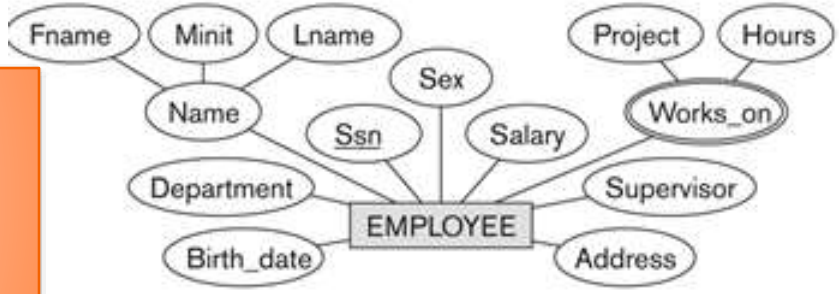
Entities

- specific objects in the mini-world with an independent existence
- E.g. EMPLOYEE John , Research DEPARTMENT

Attributes

- They are properties used to describe an entity.
- Each attribute has a data type
  - integer, string, subrange, enumerated type, …
- Key attribute

Entities with the same basic attributes are grouped into an **Entity type.**



Types of Attributes

Simple

- Each entity has a single atomic value for the attribute.
- For example, SSN.

Composite

- The attribute is composed of several components.
  - **Address**(House#, Street, City, State, Zip, Country),
  - **Name**(FirstName, MiddleName, LastName).

Multi-valued

- An entity may have multiple values for that attribute.
- For example: **PreviousDegrees** of a STUDENT.

Example of a composite attribute

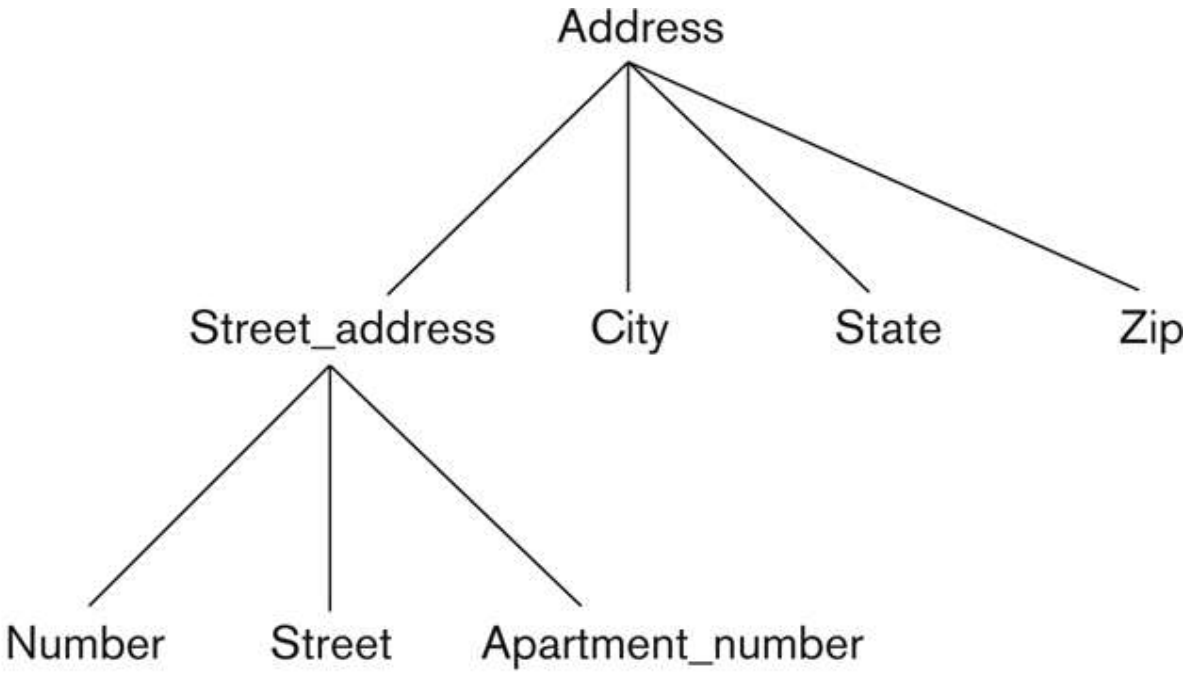


Figure 3.4
A hierarchy of composite attributes.

Types of Attributes

- The composite and multi-valued attributes may be nested arbitrarily to any number of levels,
  - PreviousDegrees of a STUDENT is a composite multi-valued attribute

{PreviousDegrees (College, Year, Degree, Field)}

    - Multiple PreviousDegrees values can exist
    - Each has four subcomponent attributes:
      - College, Year, Degree, Field

Entity Type CAR with two keys and a corresponding Entity Set

(a)

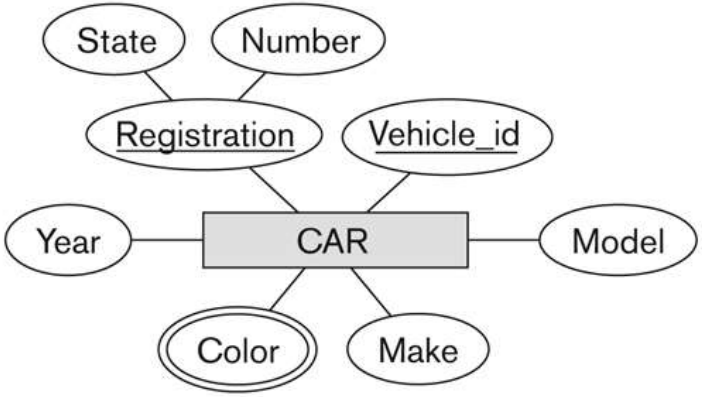


Figure 3.7
The CAR entity type with two key attributes, Registration and Vehicle_id. (a) ER diagram notation. (b) Entity set with three entities.

(b)

CAR
Registration (Number, State), Vehicle_id, Make, Model, Year, {Color}

CAR_1
((ABC 123, TEXAS), TK629, Ford Mustang, convertible, 2004 {red, black})

CAR_2
((ABC 123, NEW YORK), WP9872, Nissan Maxima, 4-door, 2005, {blue})

CAR_3
((VSY 720, TEXAS), TD729, Chrysler LeBaron, 4-door, 2002, {white, blue})

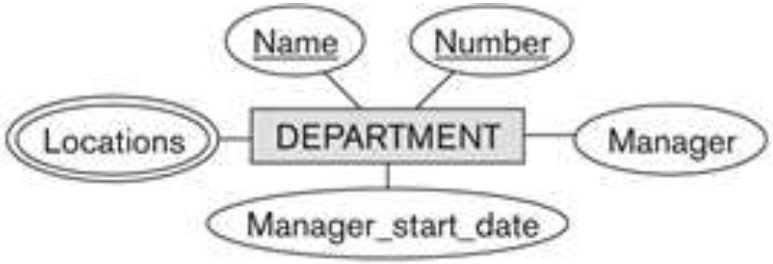
⋮

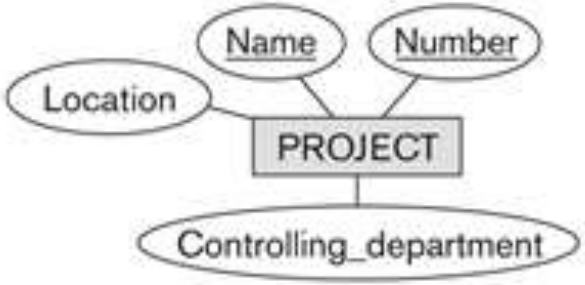
Entity set is the current *state* of the entities that is stored in the database

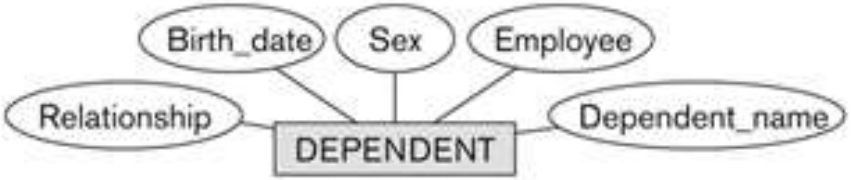
Initial Design of Entity Types

Entity types in the COMPANY database:

- EMPLOYEE
- DEPARTMENT
- PROJECT
- DEPENDENT

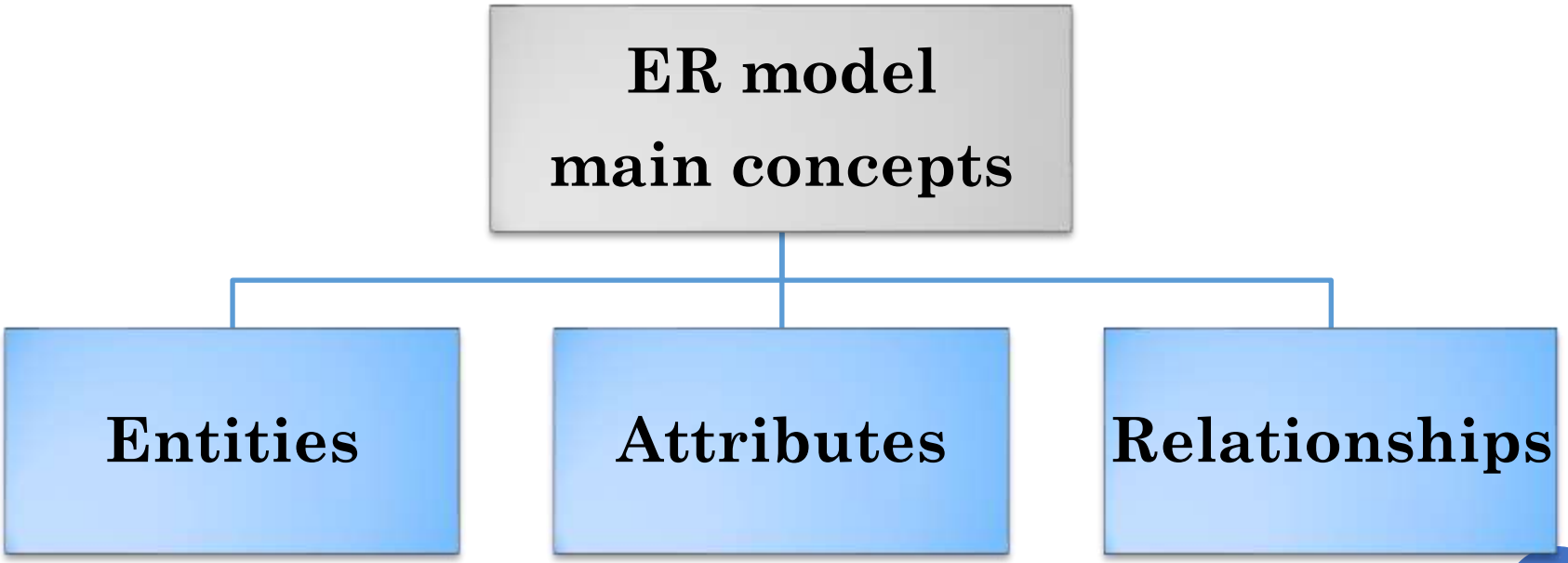






Refining the initial design by introducing relationships

Some aspects in the requirements will be represented as **relationships**



RELATIONSHIPS

A **relationship** relates two or more distinct entities with a specific meaning.

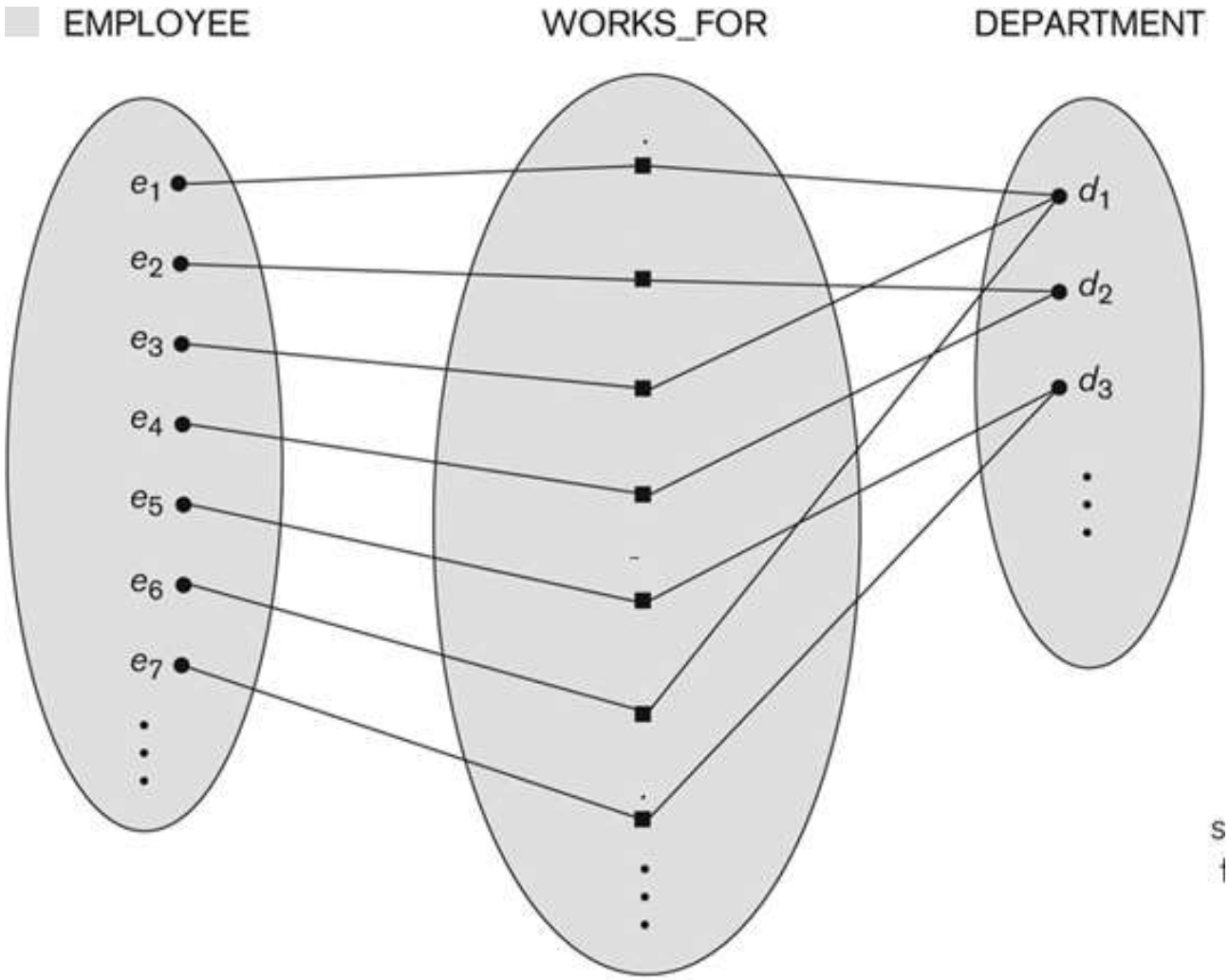
- For example, EMPLOYEE John *works on* the PX PROJECT,
- EMPLOYEE Franklin *manages* the Research DEPARTMENT.

Relationships of the same type are grouped into a **relationship type**.

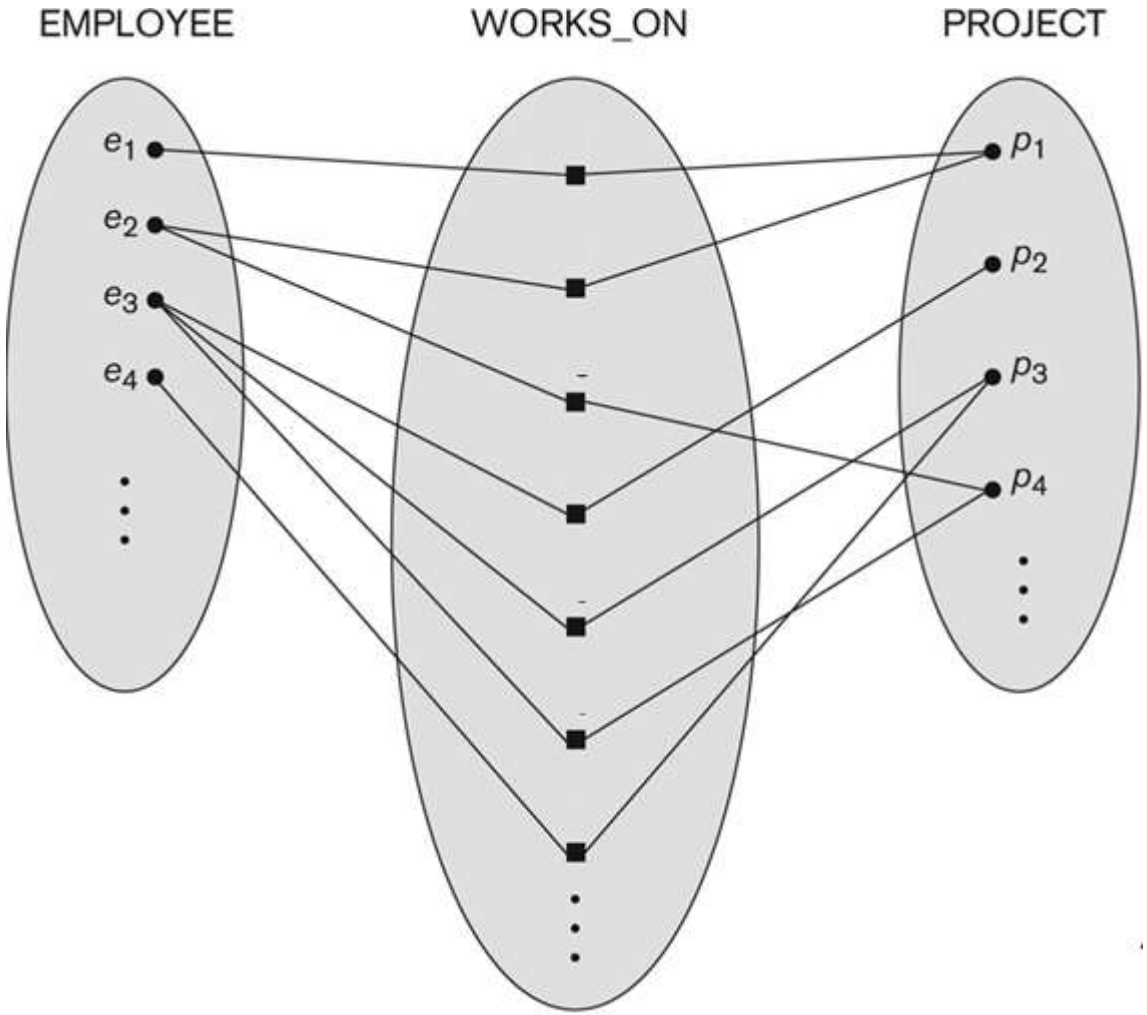
Degree of a relationship type is the no of participating entity types.

- Both MANAGES and WORKS_ON are *binary* relationships.

Relationship instances of the WORKS_FOR N:1



Relationship instances of the M:N WORKS_ON



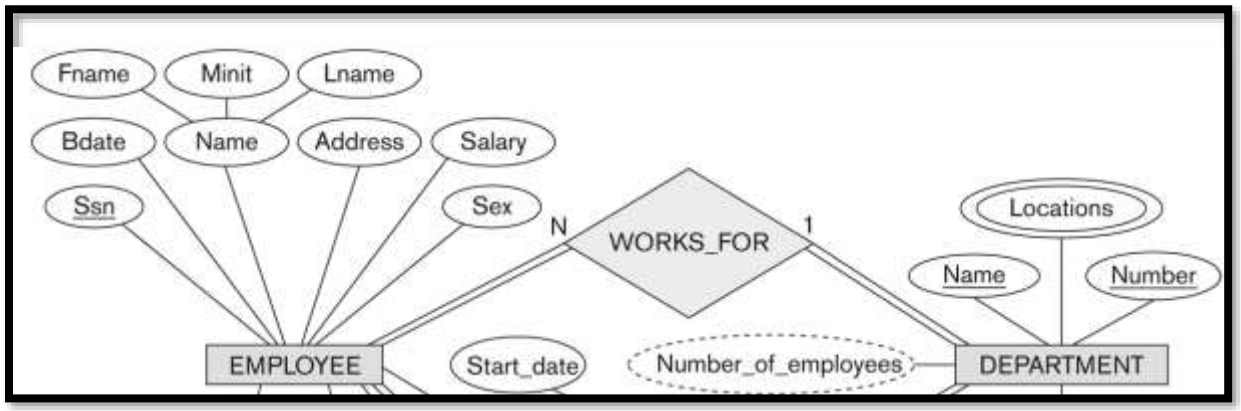
Relationship type vs. relationship set

Relationship Type:

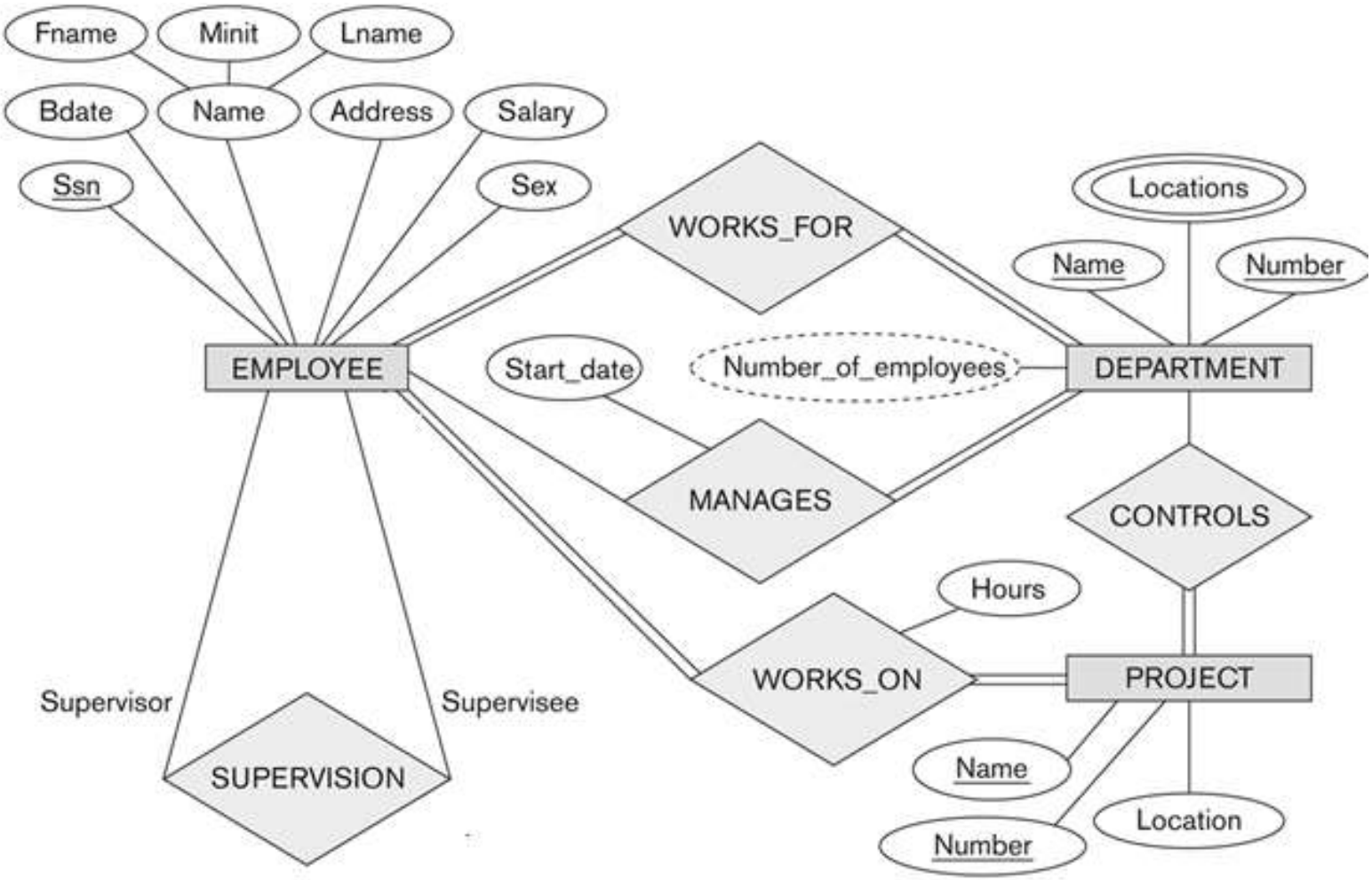
- Is the schema description of a relationship
- Identifies the relationship name and the participating entity types
- Also identifies certain relationship constraints

Relationship Set:

- The current *state* of a relationship type



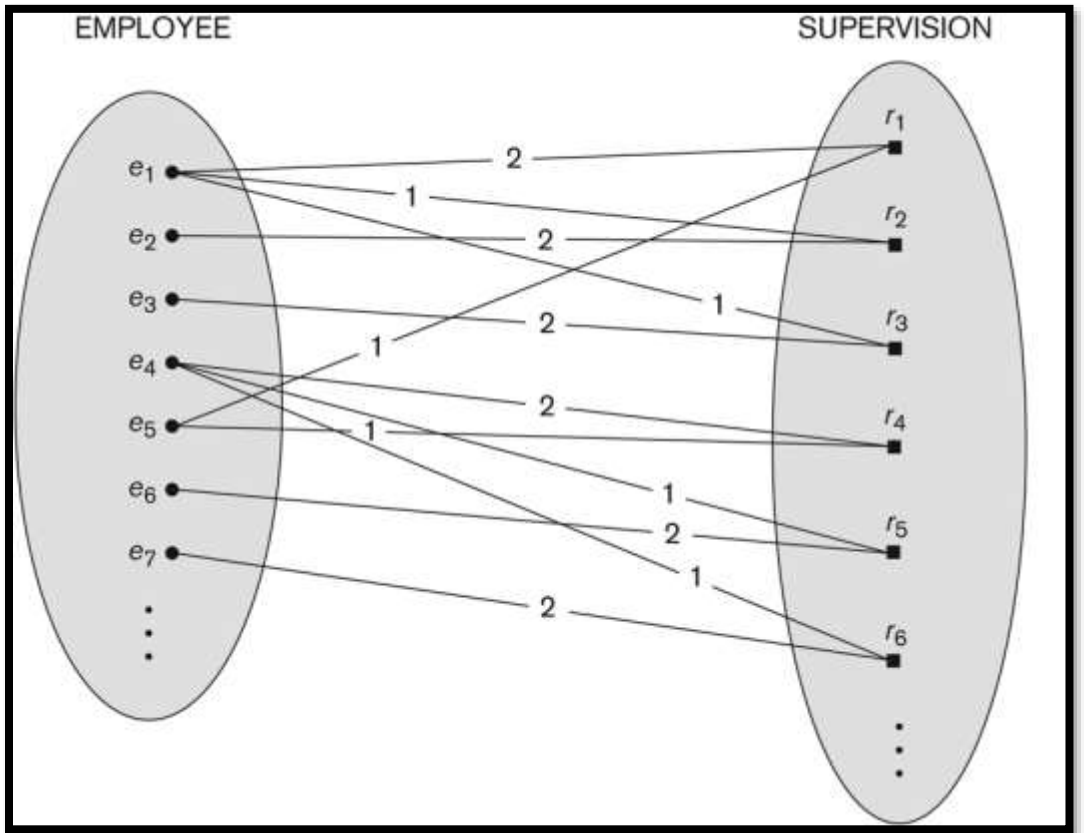
ER DIAGRAM in **Chen Notation**



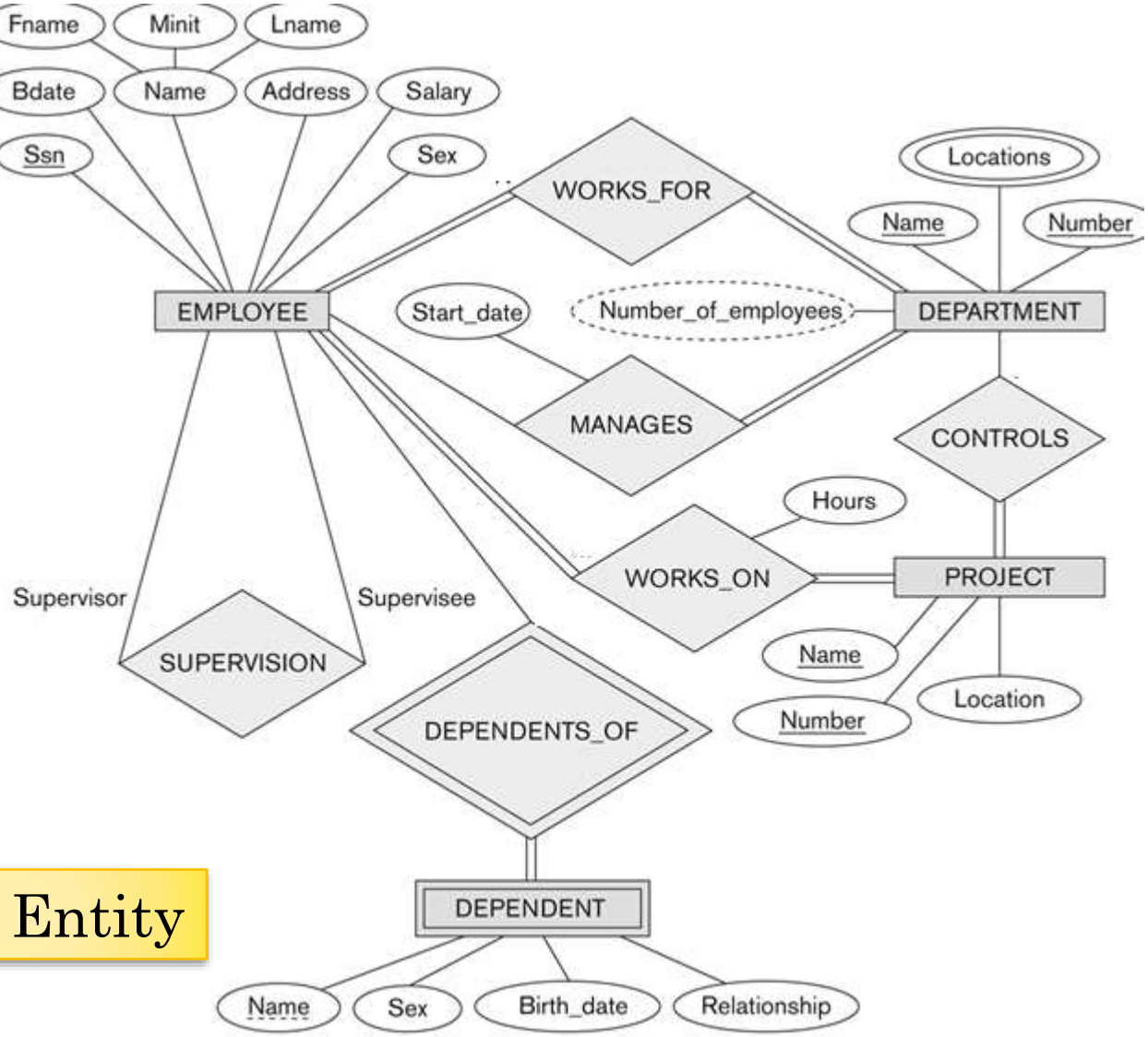
Recursive Relationship

Recursive Relationship Type

- EMPLOYEE participates twice in two distinct roles:
  - supervisor (or boss) role
  - supervisee (or subordinate) role



ER DIAGRAM



Weak Entity

Weak Entity Types

An entity that does not have a key attribute

A weak entity must participate in an identifying relationship type with an owner or identifying entity type

Entities are identified by the combination of:

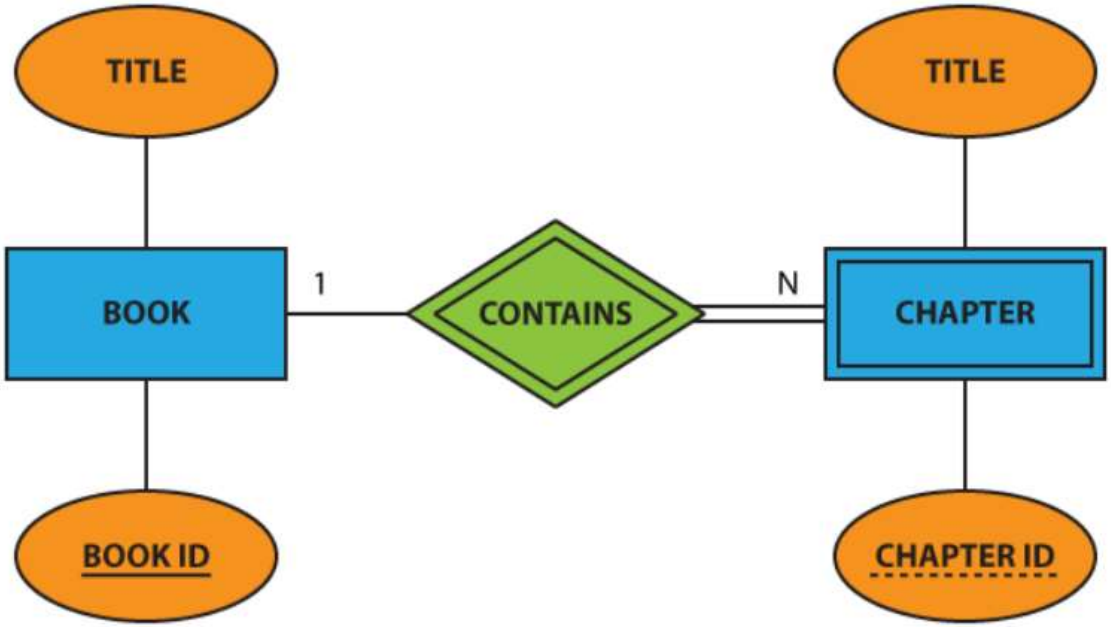
- Partial key of the weak entity type
- The particular entity they are related to in the identifying entity type

Example: A DEPENDENT entity is identified by

- the dependent's first name (*partial key*), *and*
- the specific EMPLOYEE with whom the dependent is related

Weak Entity

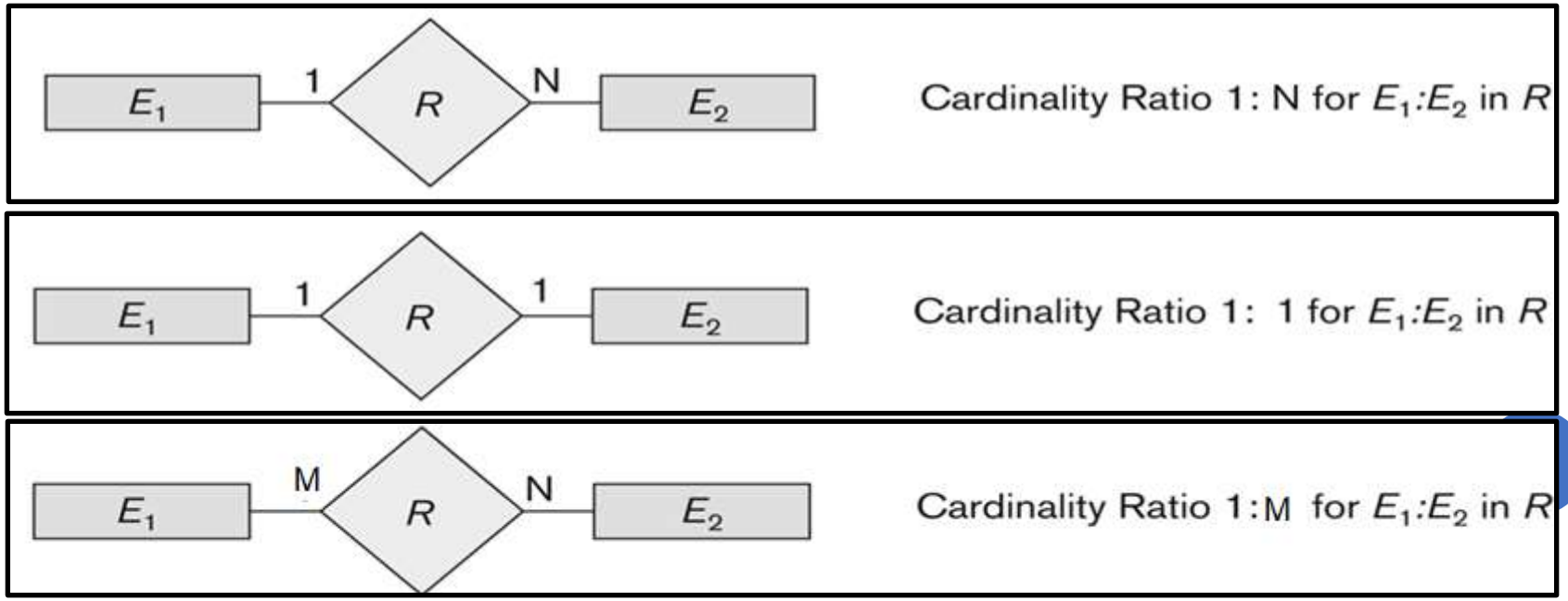
- An entity can be weak => when it does not have a key attribute



Constraints on Relationships

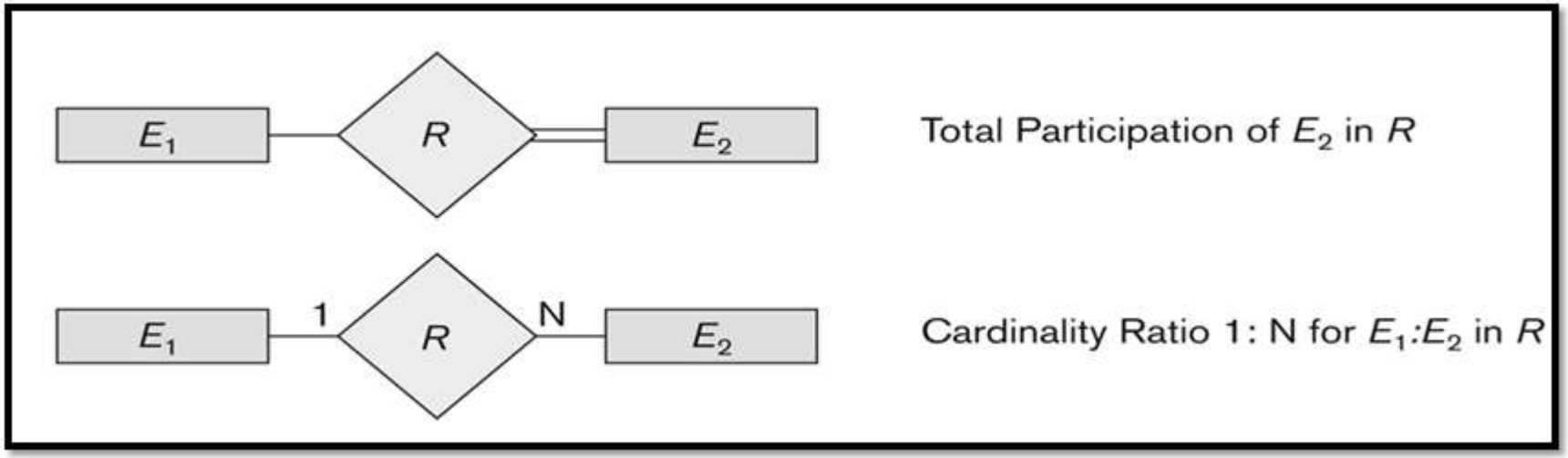
Cardinality Ratio Constraint:
specifies ***maximum*** participation

- One-to-one (1:1)
- One-to-many (1:N) or Many-to-one (N:1)
- Many-to-many (M:N)



RELATIONSHIP CONSTRAINTS

Derived from the knowledge of mini-world constraints



Constraints on Relationships

Cardinality Ratio Constraint

(specifies ***maximum*** participation)

- One-to-one (1:1)
- One-to-many (1:N) or Many-to-one (N:1)
- Many-to-many (M:N)

Existence Dependency Constraint or **Participation constraint**

(specifies ***minimum*** participation)

- zero (optional participation, not existence-dependent)
- one or more (mandatory participation, existence-dependent)

Company ER Model

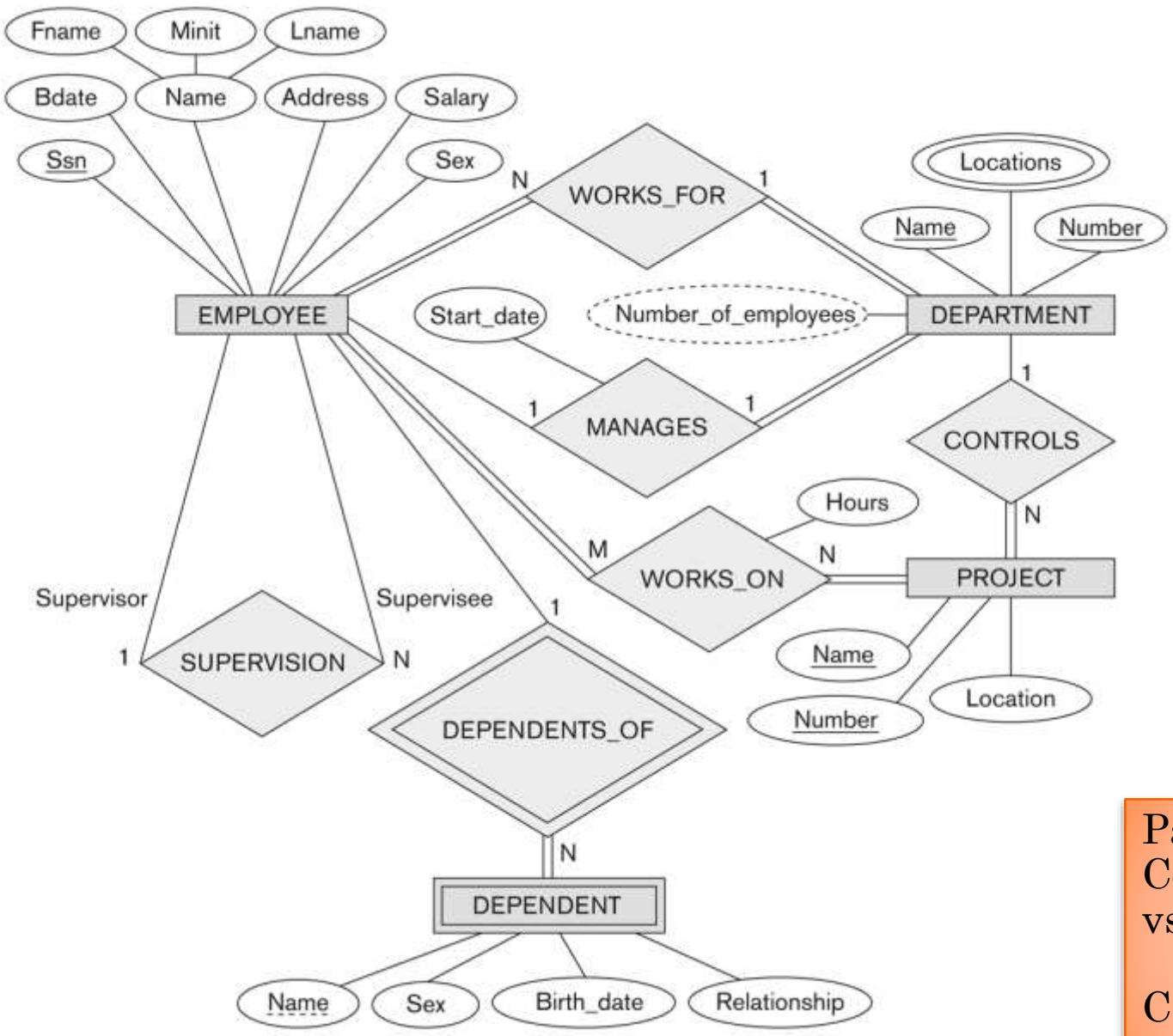


Figure 3.2

Participation Constraint : Total vs partial.

Cardinality Ratio: 1:N, 1:1, N:M

Attributes of Relationship types

A relationship type can have attributes:

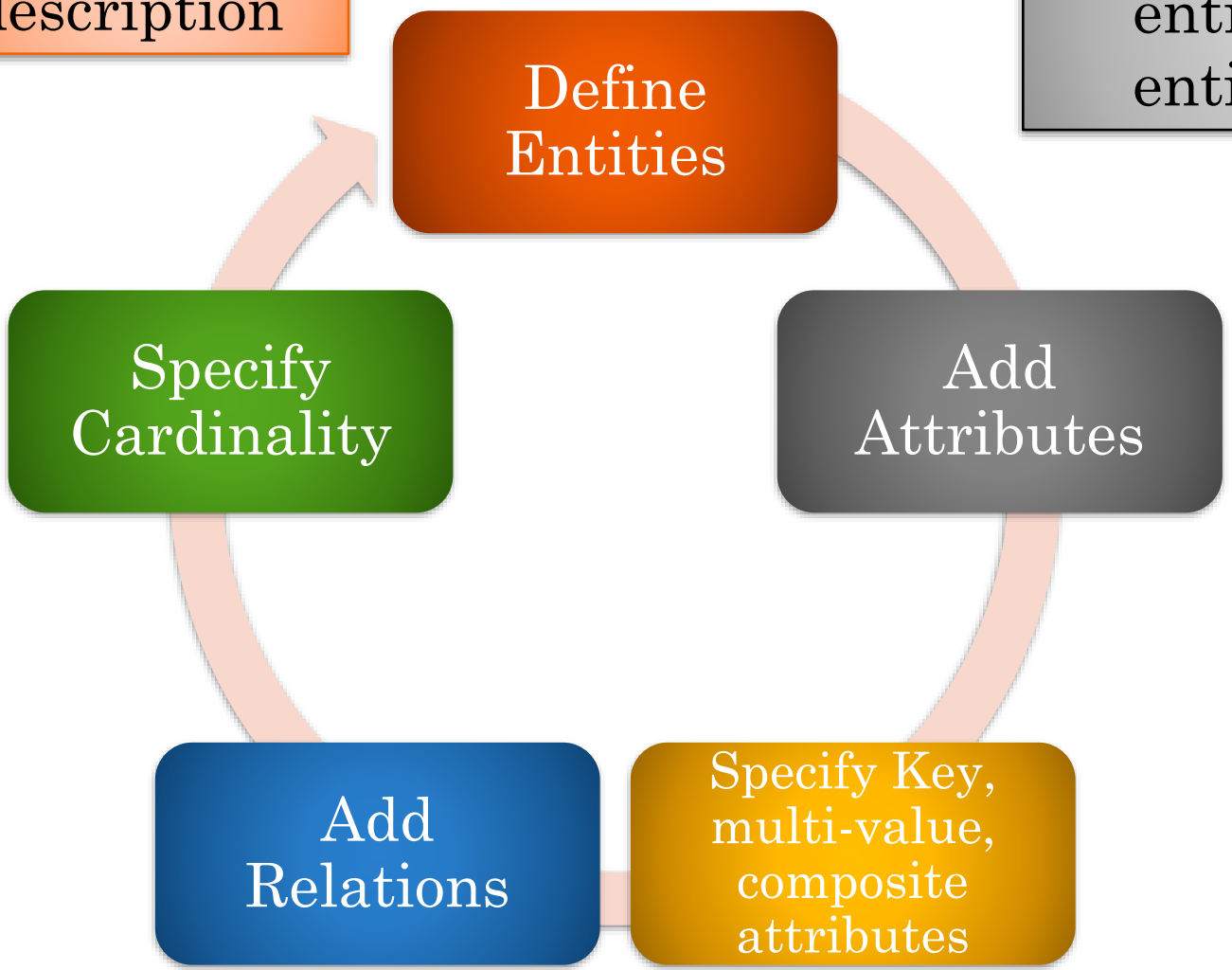
- For example, HoursPerWeek of WORKS_ON
- A value of HoursPerWeek depends on a particular (employee, project) combination

Most relationship attributes are used with M:N relationships

- In 1:N relationships, they can be transferred to the entity type on the N-side of the relationship

Steps to draw an ER diagram

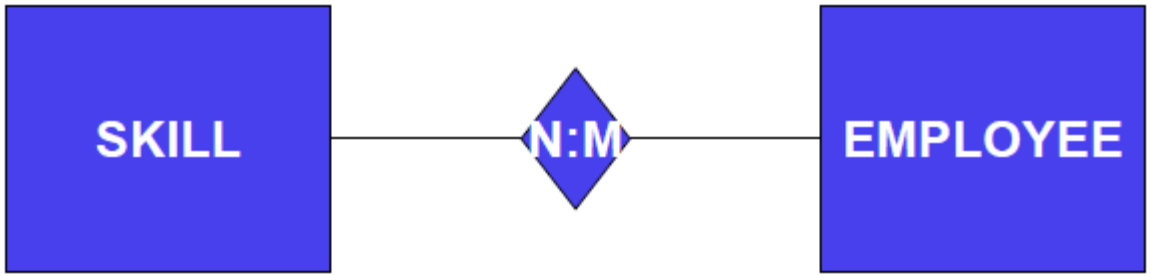
Get problem description



Identify if any entity is a weak entity (key?)

Example – ER-Model

- For each employee we record employee ID, name, SSN, gender, address and phone. An employee may have several skills such as communication, self-motivation, Technical literacy, Data analytics etc. A particular skill may be held by several employees of the company. An employee may have many teammates (current and previous).



Example – ER-Model

- An employee have to serve multiple clients. But a client deals with only one employee to avoid confusions in dealings. We keep track of client's business email, name and address to connect with them.

- A client can place many orders, but each order can be placed by one client only. An order is identified by order number and have a date and timestamp. Before placing an order the client must enter the credit card information. We need to verify client's credit-card information before processing his orders.

Example – ER-Model

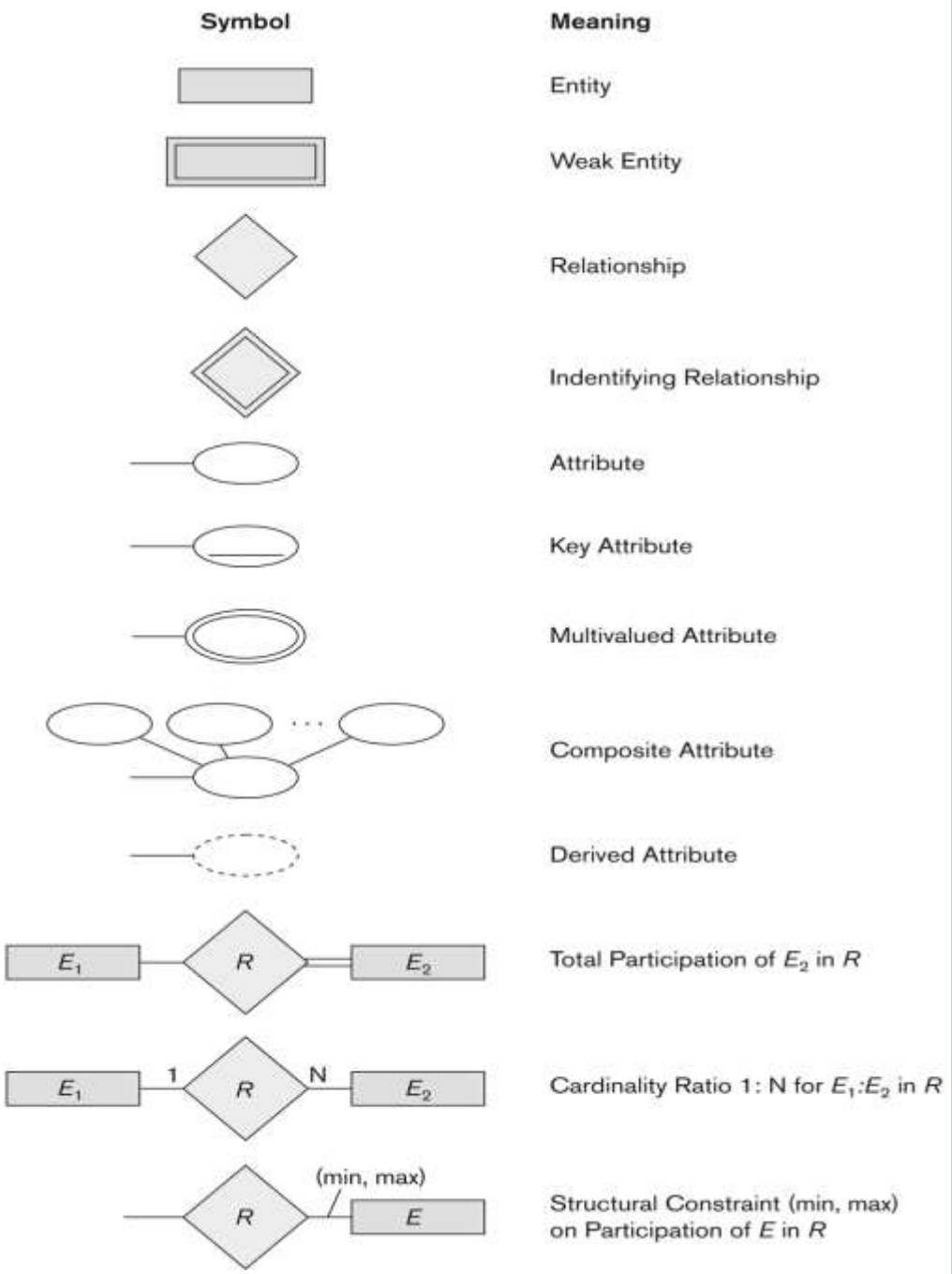
- The company owns multiple buildings around the country. Each building has an id, name, and host many offices for carrying out different company operations and clients' tasks. Each office within a building has a unique number and name.

- Each employee is associated with an office. An office can have multiple employees, so we need to know the total number of employee in each office. All employees working in an office are considered to be colleagues.

- Each employee is allotted a cubicle to work in an office. A cubicle is assigned to one employee only.

Notation for ER diagrams

Figure 3.14
Summary of the notation for ER diagrams.

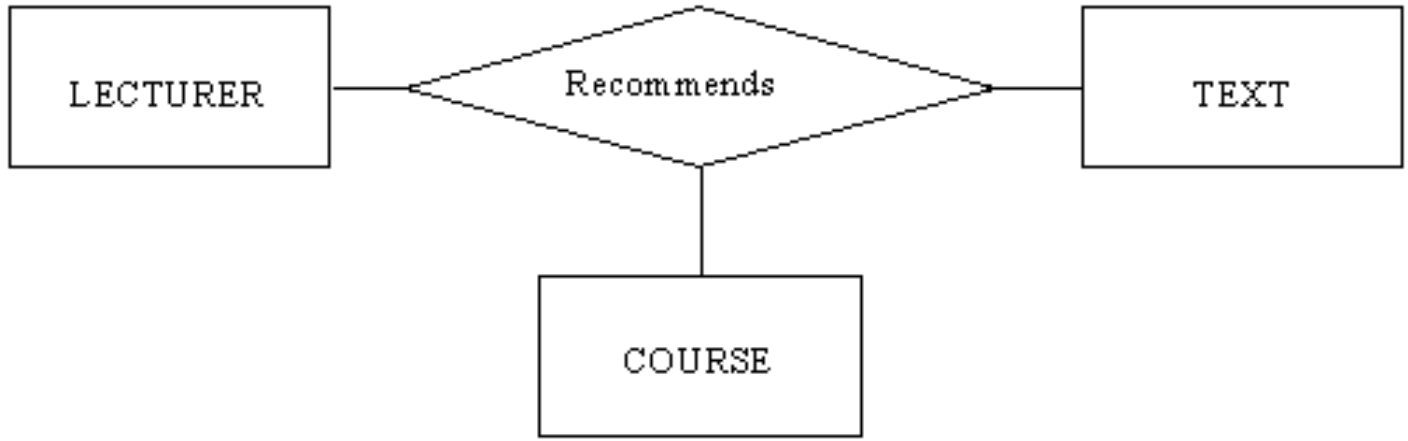


Relationships of Higher Degree

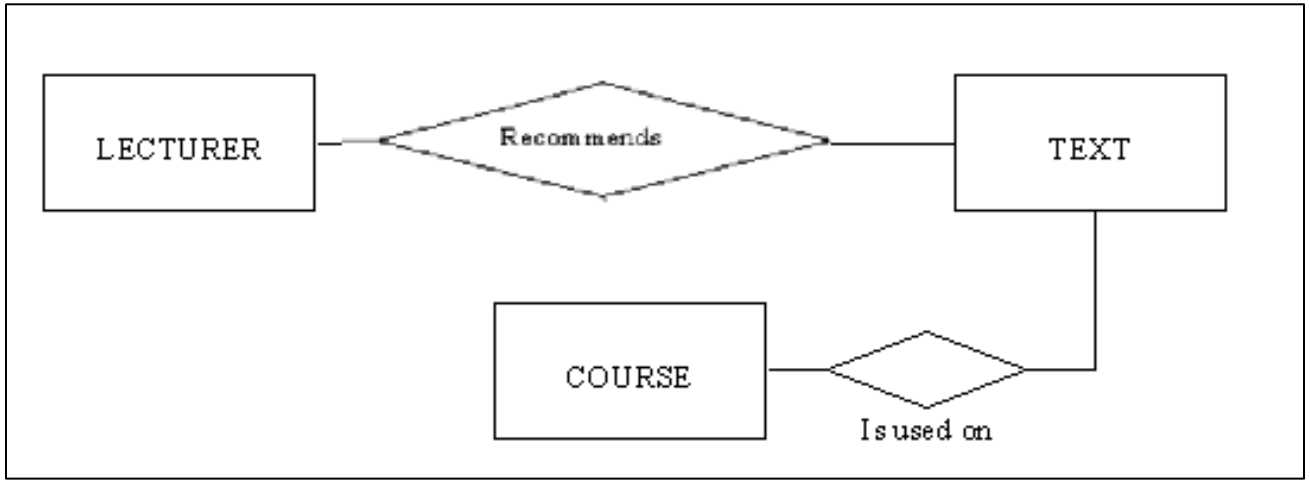
- Relationship types of degree 2 are called binary
- Relationship types of degree 3 are called ternary and of degree n are called n-ary
- Constraints are harder to specify for higher-degree relationships (n > 2) than for binary relationships

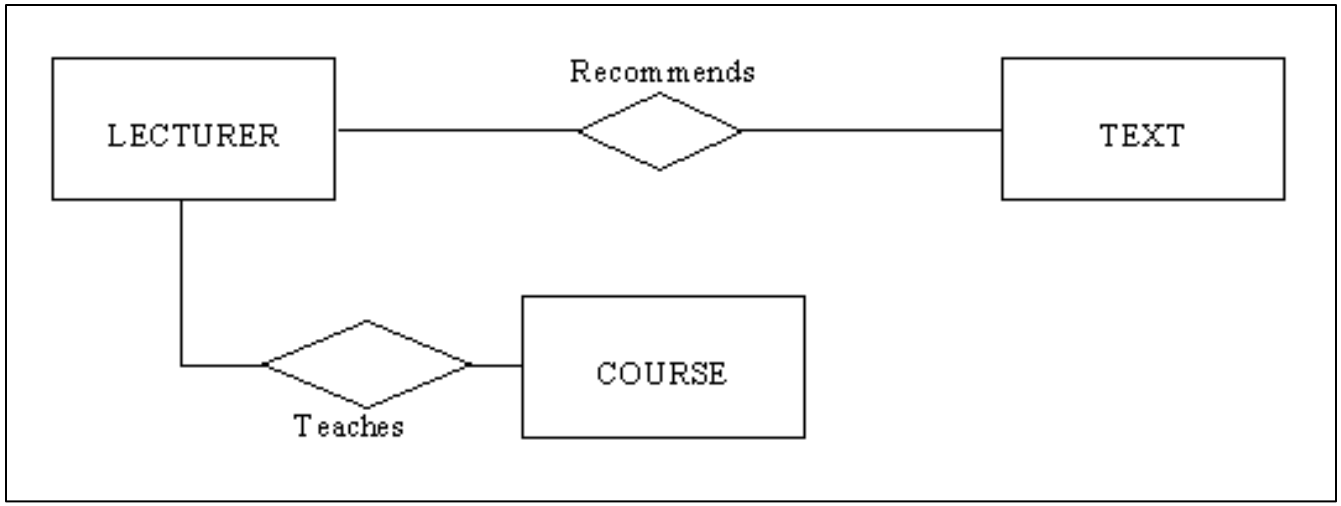
Example of a ternary relationship

(a)

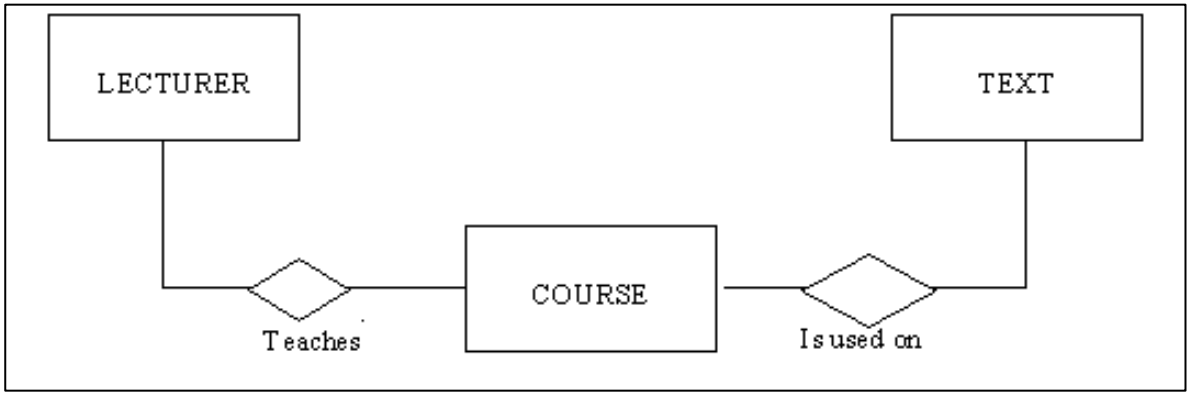


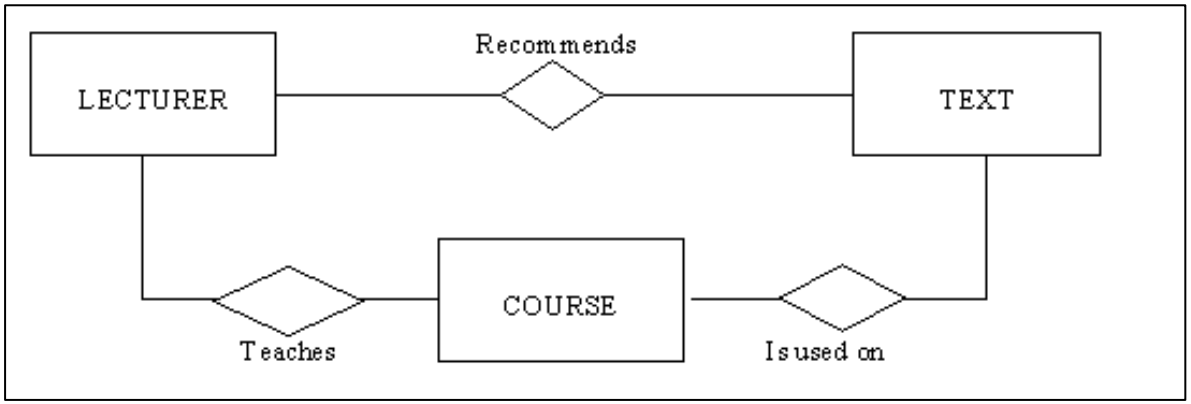
EXAMPLE OF A TERNARY RELATIONSHIP





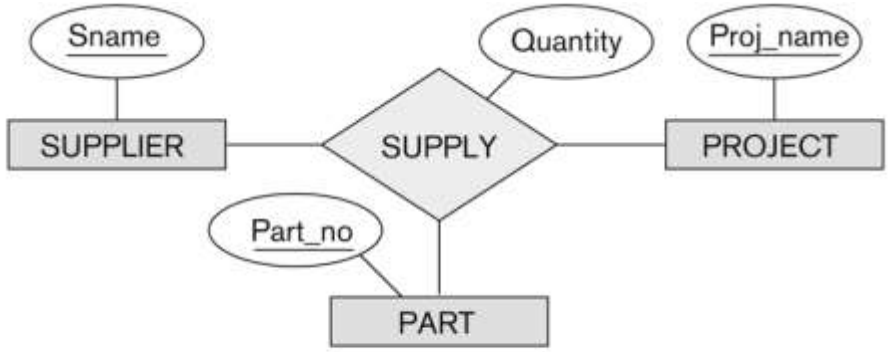
EXAMPLE OF A TERNARY RELATIONSHIP



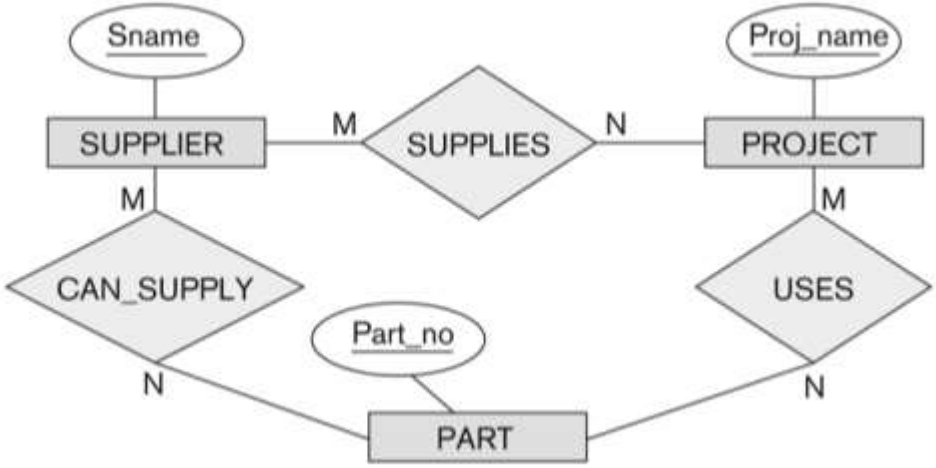


Example 2 of a ternary relationship

(a)

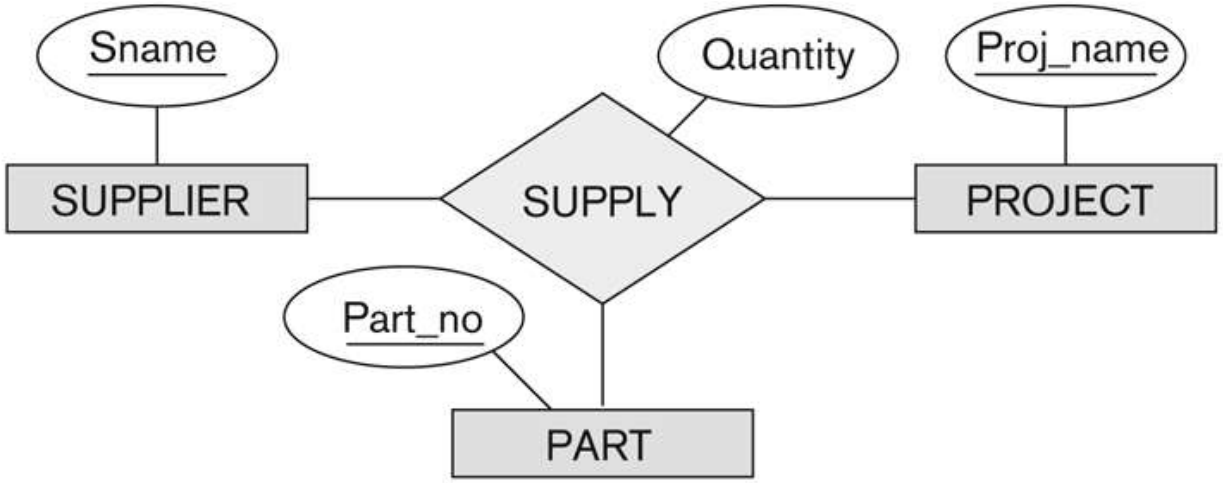


(b)



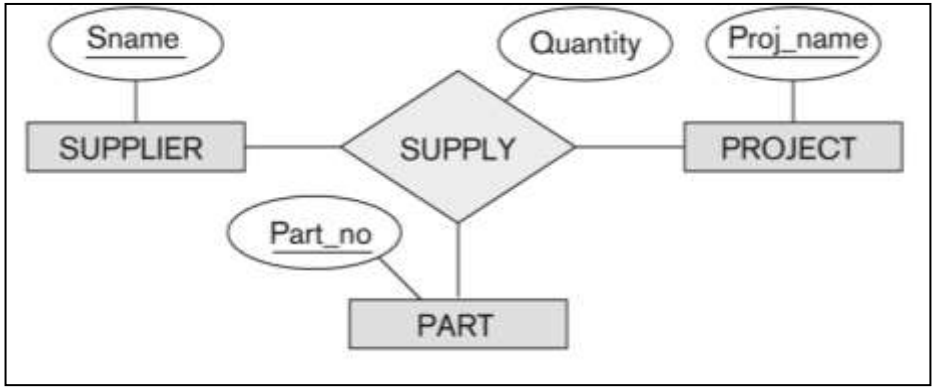
Cardinality for ternary relationship

Constraint: For a particular project-part combination, only one supplier will be used (only one supplier supplies a particular part to a particular project).

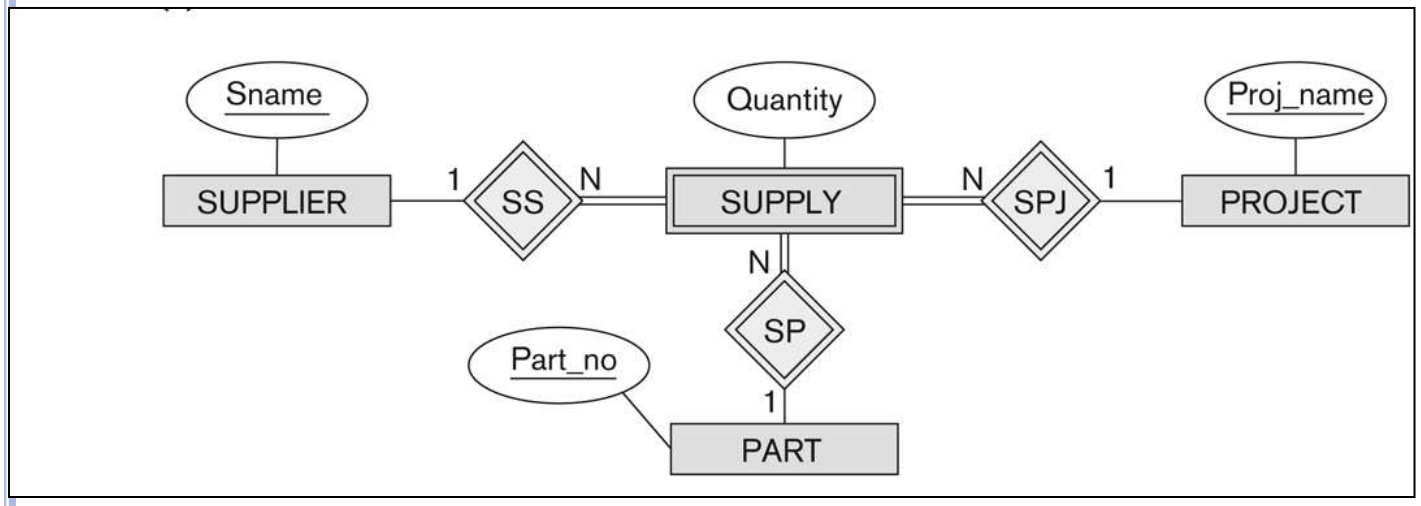


N-ARY RELATIONSHIPS (N > 2)

- Three binary relationships represents different information than a single ternary relationship

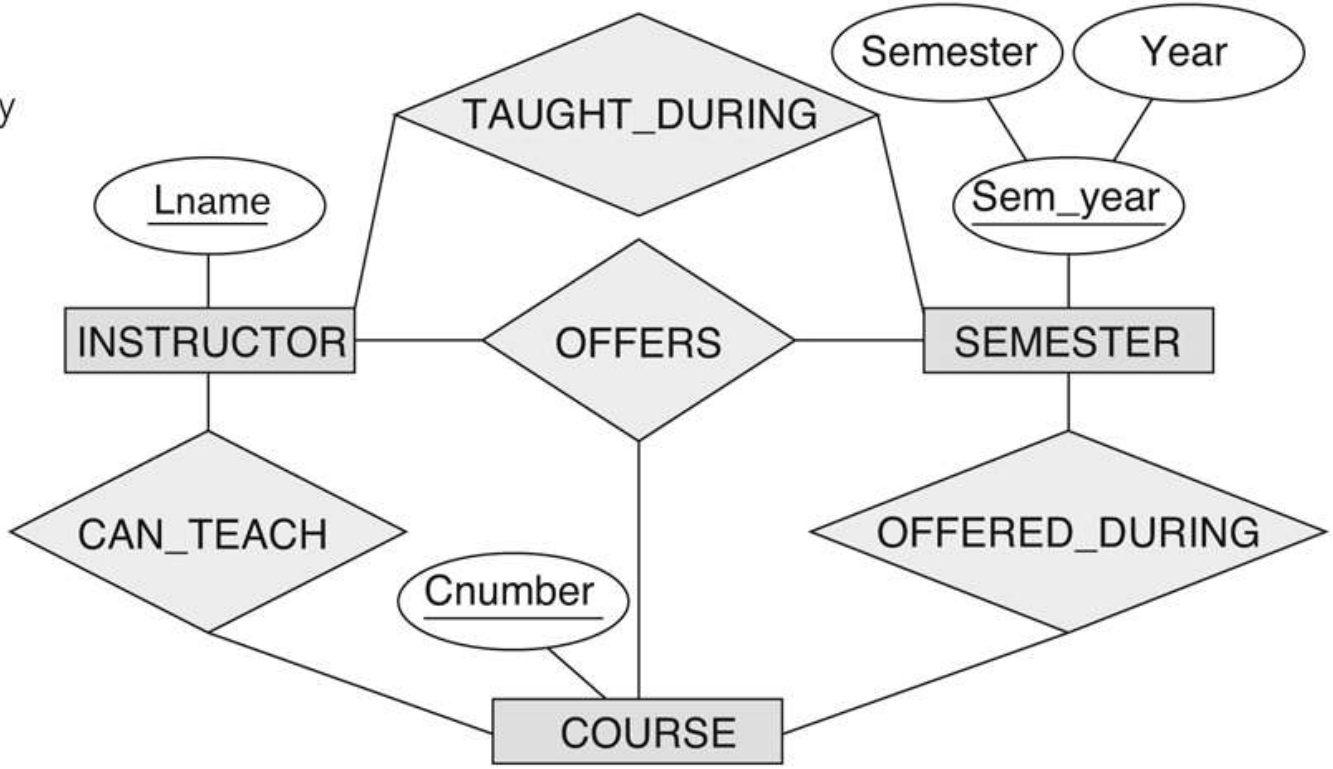


In some cases, a ternary relationship can be represented as a weak



Example 3 of a ternary relationship

Figure 3.18
Another example of ternary versus binary relationship types.



If a particular binary relationship can be derived from a higher-degree relationship at all times, then it is redundant

Activity: ER for Notown Records

Each musician has an SSN, name, address, phone. Poor musicians often share the same address, and no address has more than one phone.

Each instrument that is used in songs recorded at Notown has a name (e.g., guitar, flute) and a musical key (e.g., C, B-flat).

Each album has a title, a copyright date, a format (e.g., CD or MC), and an album identifier.

Each song recorded at Notown has a title and an author.

Each musician may play several instruments, and an instrument may be played by several musicians.

Each album has a number of songs on it, but no song may appear on more than one album.

Each song is performed by one or more musicians and *we keep track of the instrument used by each musician*. A musician may perform a number of songs.

Each album has exactly one musician who acts as its producer. A musician may produce several albums, of course.

Problem: Draw E\R model for the Given university database

a) The university keeps track of each student's name, rollno, SSN, current address and phone, permanent address and phone, birthdate, class (freshman, sophomore, ..., graduate), major department, minor department (if any), and degree program (B.A., B.S., ..., Ph.D.). Some user applications need to refer to the city, state, and zip of the student's permanent address, and to the student's last name. Both SSN and Rollno have unique values for each student.

b) Each department is described by a name, code, office number, office phone, and college. Both name and code have unique values for each department.

c) Each course has a name, description, number, number of semester hours, level, and offering department. The value of course number is unique for each course.

d) Each section has an instructor, semester, year, course, and section number. The section number distinguishes different sections of the same course that are taught during the same semester/year; its values are 1, 2, 3, ...; up to the number of sections taught during each semester.

e) A grade report has a student rollno, section, letter grade, and numeric grade (0, 1, 2, 3, 4 for F, D, C, B, A, respectively).

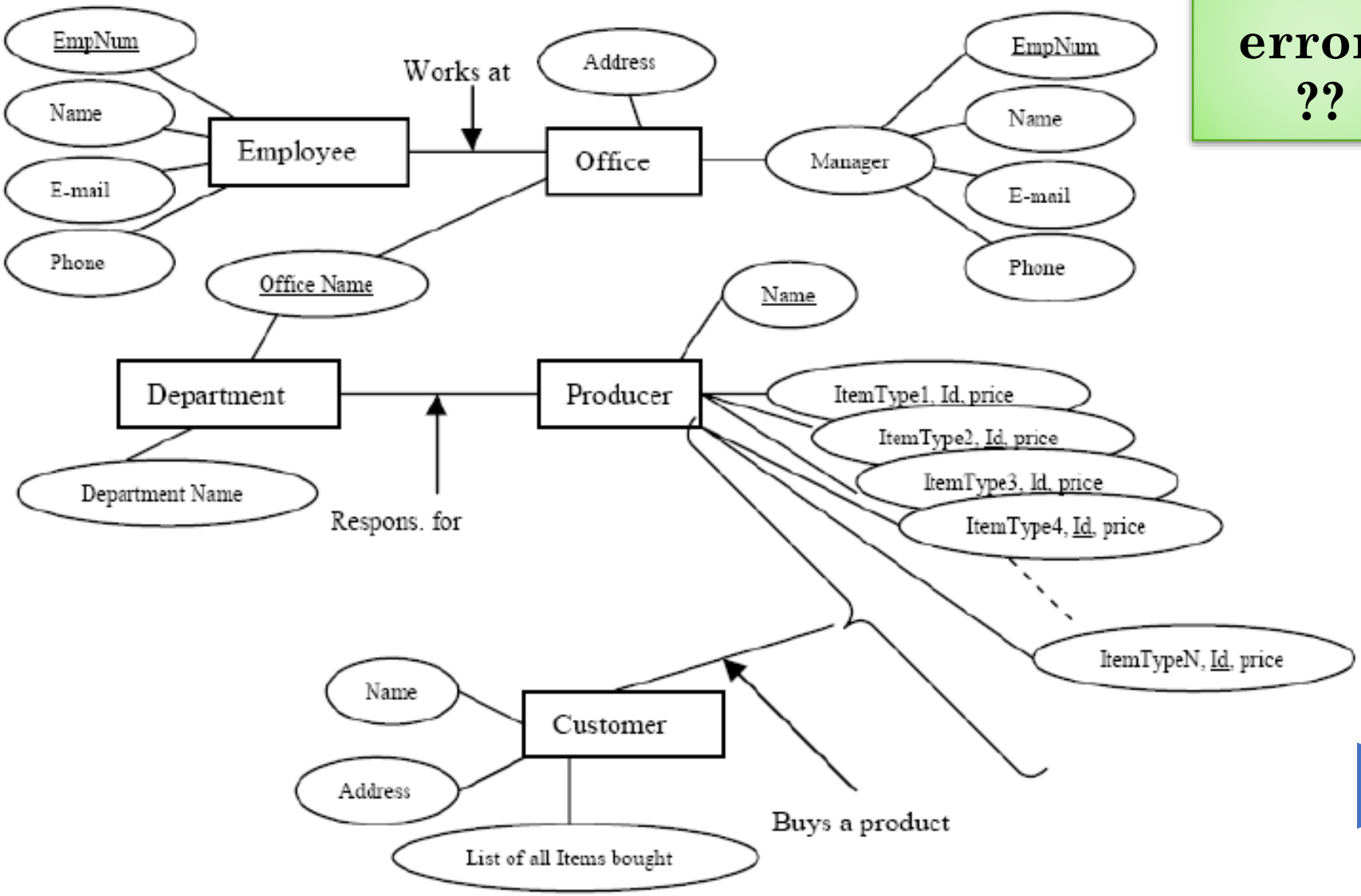
Practice Questions

- Fundamental of Database System (Edition 7)
  - **Example 3.10**
  - **Exercise Questions**
    - **3.16 - 3.19**
    - **3.21 - 3.29**
    - **3.32 – 3.34**
      - **In these questions you are asked to draw ER Model on ERWIN … but for now you just have to do it on paper.**

TEAM WORK IS ALLOWED FOR PRACTICE QUESTIONS

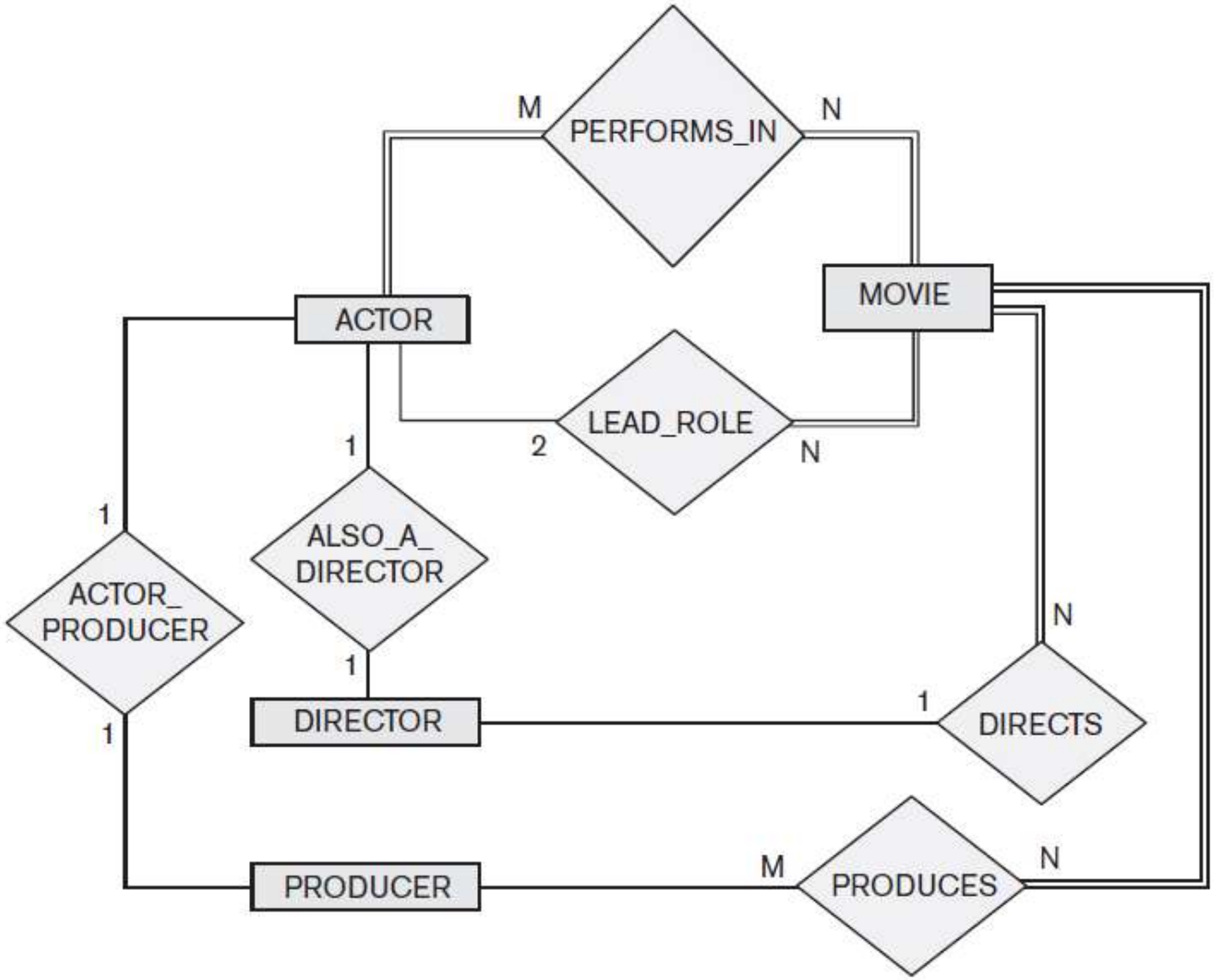
This diagram is supposed to model a company that buy products from producers and sell them to customers.

Find errors ??

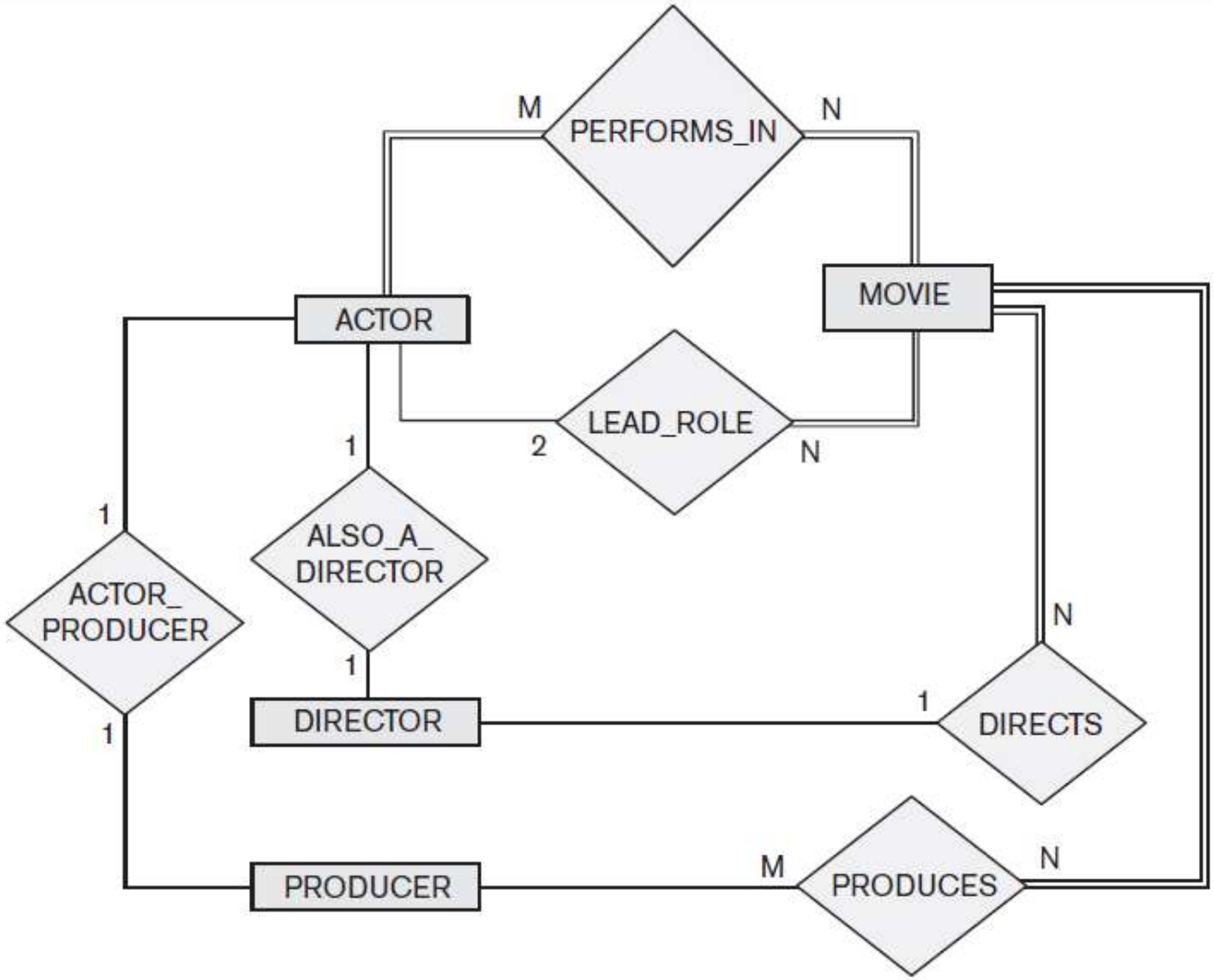


Assume that MOVIES is a populated database. ACTOR is used as a generic term and includes actresses.

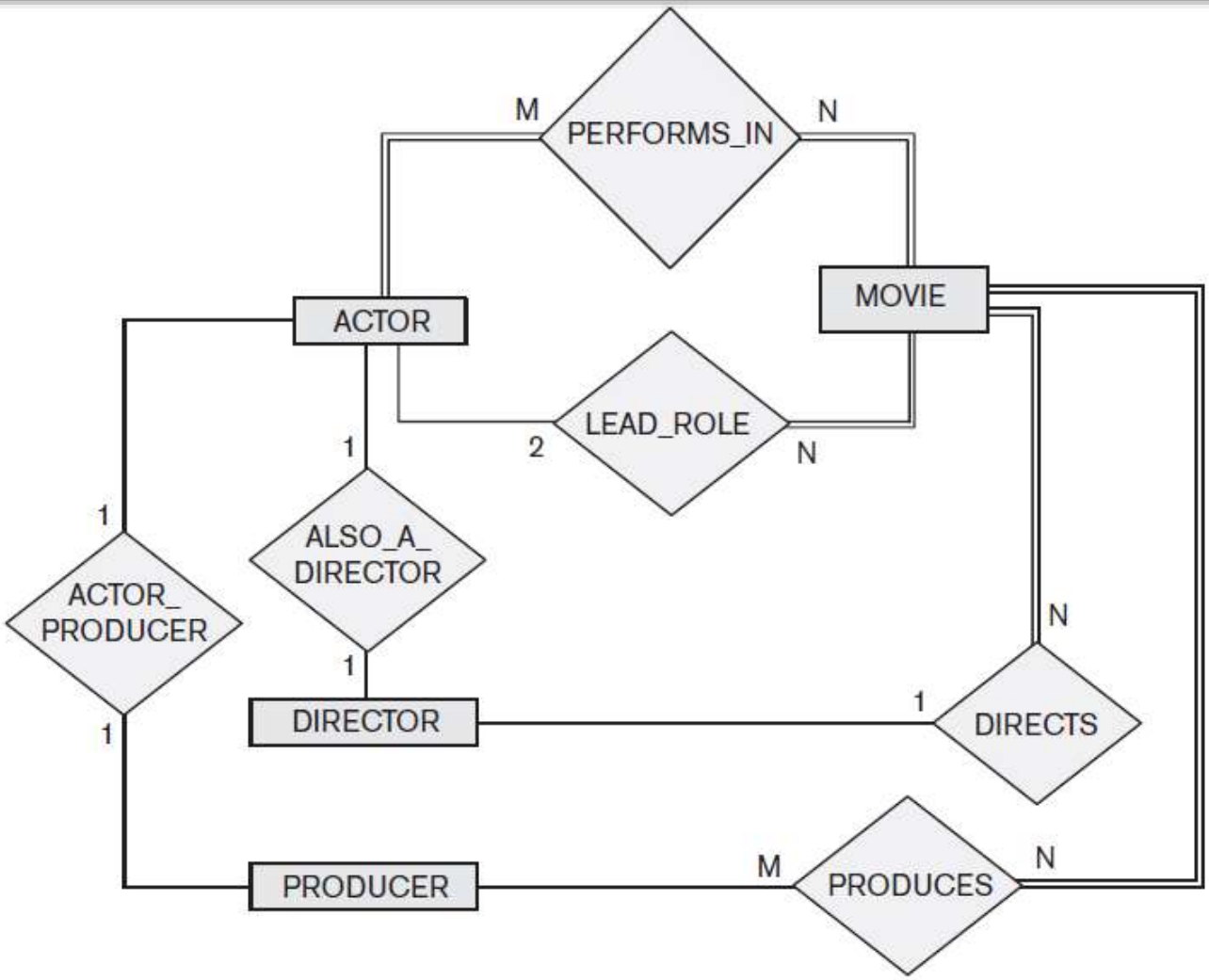
Given the constraints shown in the ER schema, respond to the following statements with *True*, *False*, or *Maybe*.



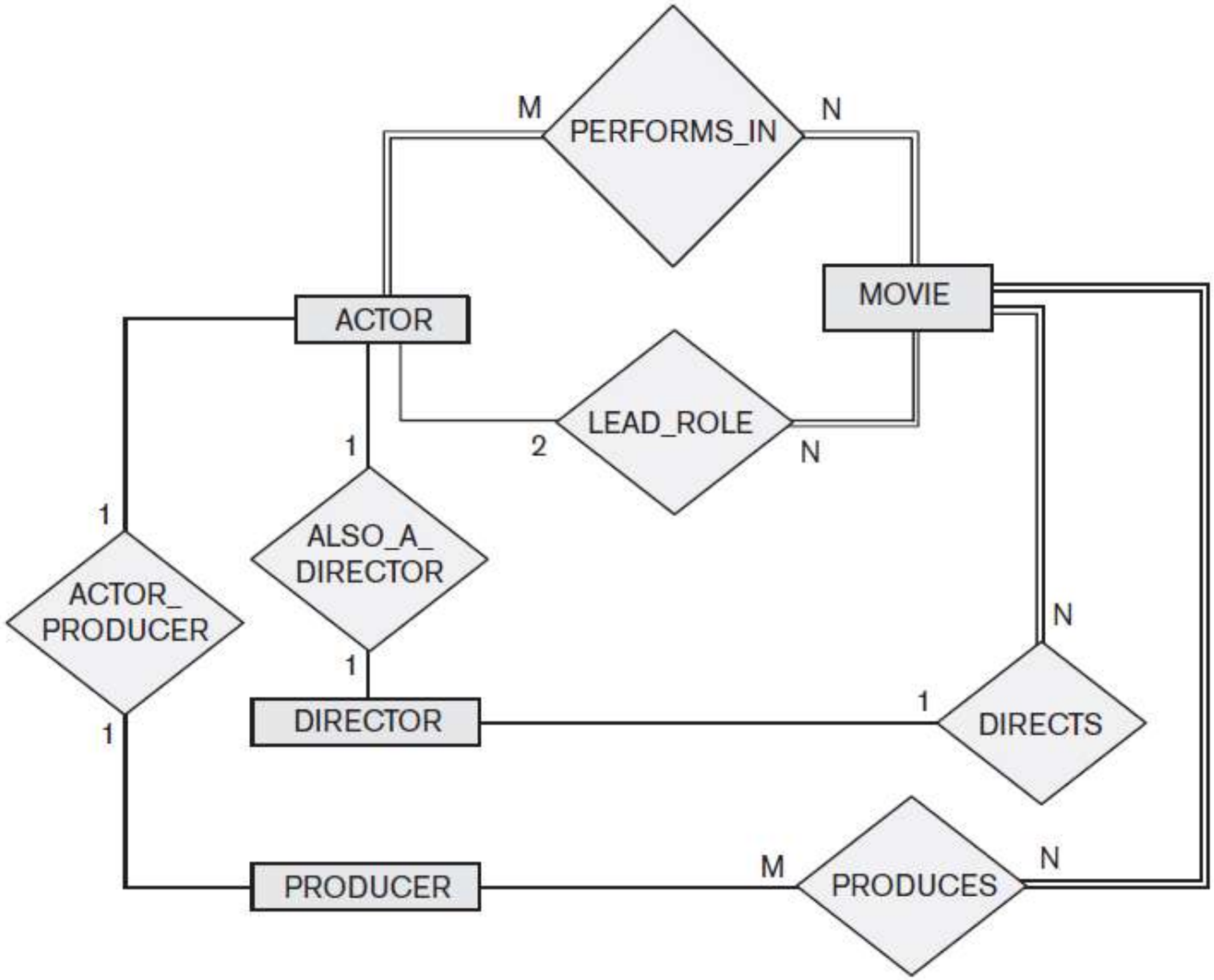
- a. There are no actors in this database that have been in no movies.
- b. There are some actors who have acted in more than ten movies.
- c. Some actors have done a lead role in multiple movies.
- d. A movie can have only a maximum of two lead actors.
- e. Every director has been an actor in some movie.



- f. No producer has ever been an actor.
- g. A producer cannot be an actor in some other movie.
- h. There are movies with more than a dozen actors.
- i. Some producers have been a director as well.
- j. Most movies have one director and one producer.



- **k. Some movies have one director but several producers.**
- **l. There are some actors who have done a lead role, directed a movie, and produced some movie.**
- **m. No movie has a director who also acted in that movie.**



Chapter Summary

- ER Model Concepts: Entities, attributes, relationships
- Constraints in the ER model
- Using ER in step-by-step conceptual schema design for the COMPANY database
- ER Diagrams - Notation
- Alternative Notations – UML class diagrams, others