COSC344
Database Theory and Applications

Lecture 2
Database Design

Learning Objectives of Lecture 2

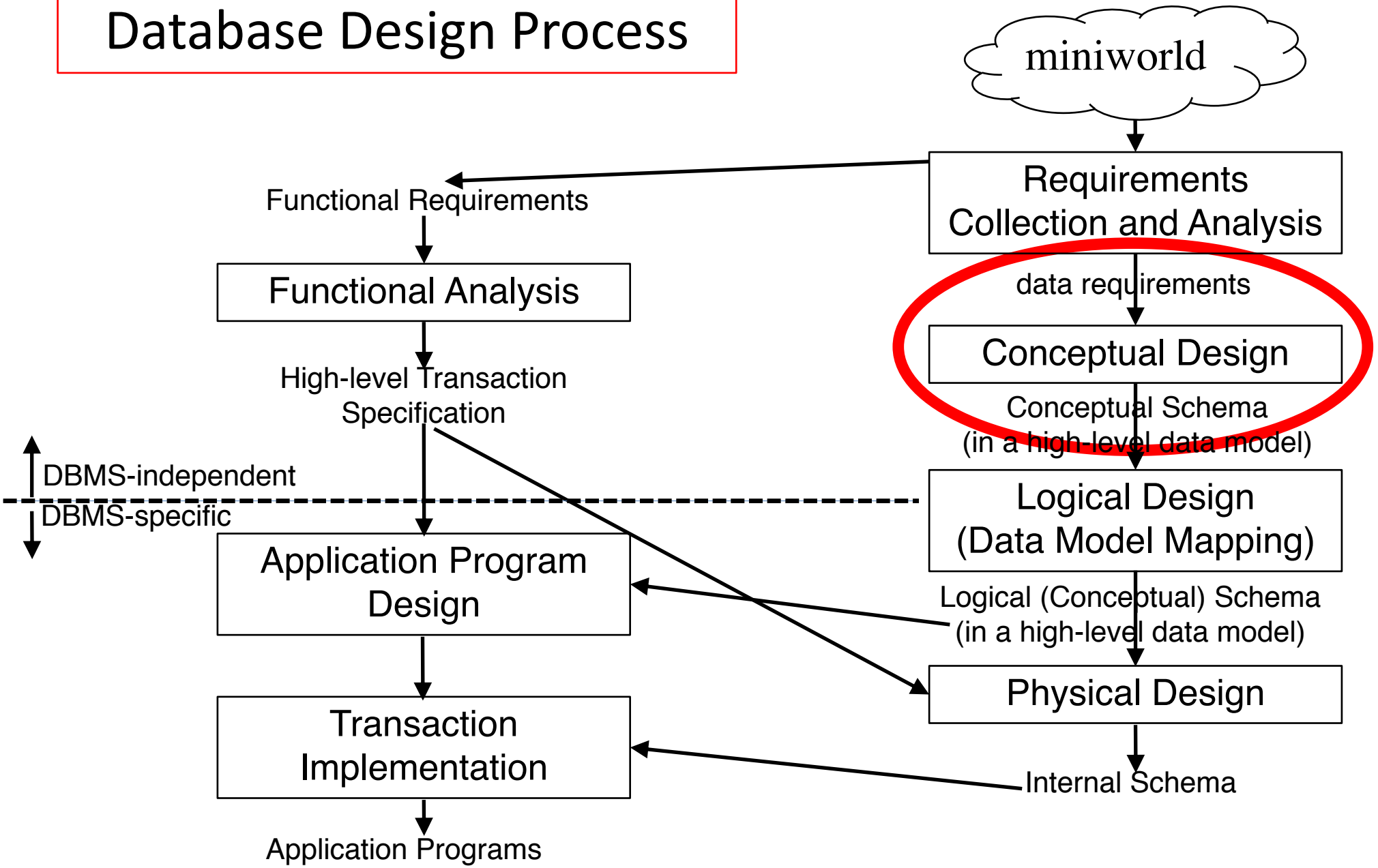
You should

- Understand the components in an ER diagram
  - Entity and Attribute
  - Relationship (cardinality ratio, participation constraints)
  - Weak Entity
- Be able to model a mini-world using ER diagram

Source

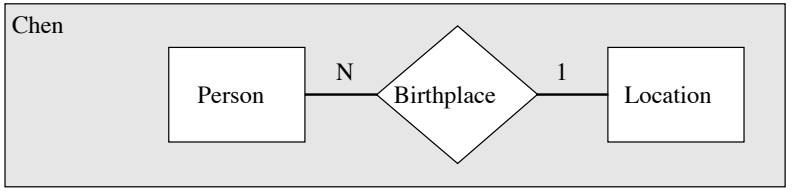
- Textbook: Chapter 3.1 – 3.7

Database Design Process

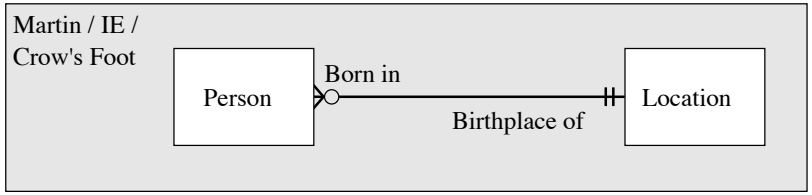


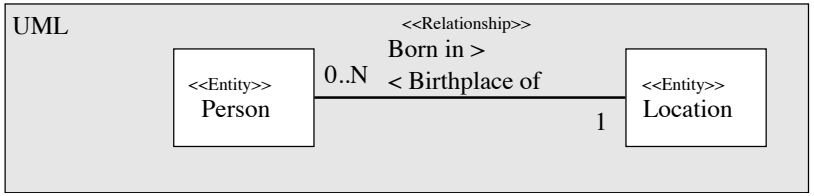
Entity – Relationship Model (ER-model)

- ER-Model is a popular high-level conceptual data model in database modelling.
- The diagrammatic notation associated with the ER model is called the **ER diagram**



- ER diagram notation approach
  - Chen's notation
  - Crow's Foot Notation
  - UML notation

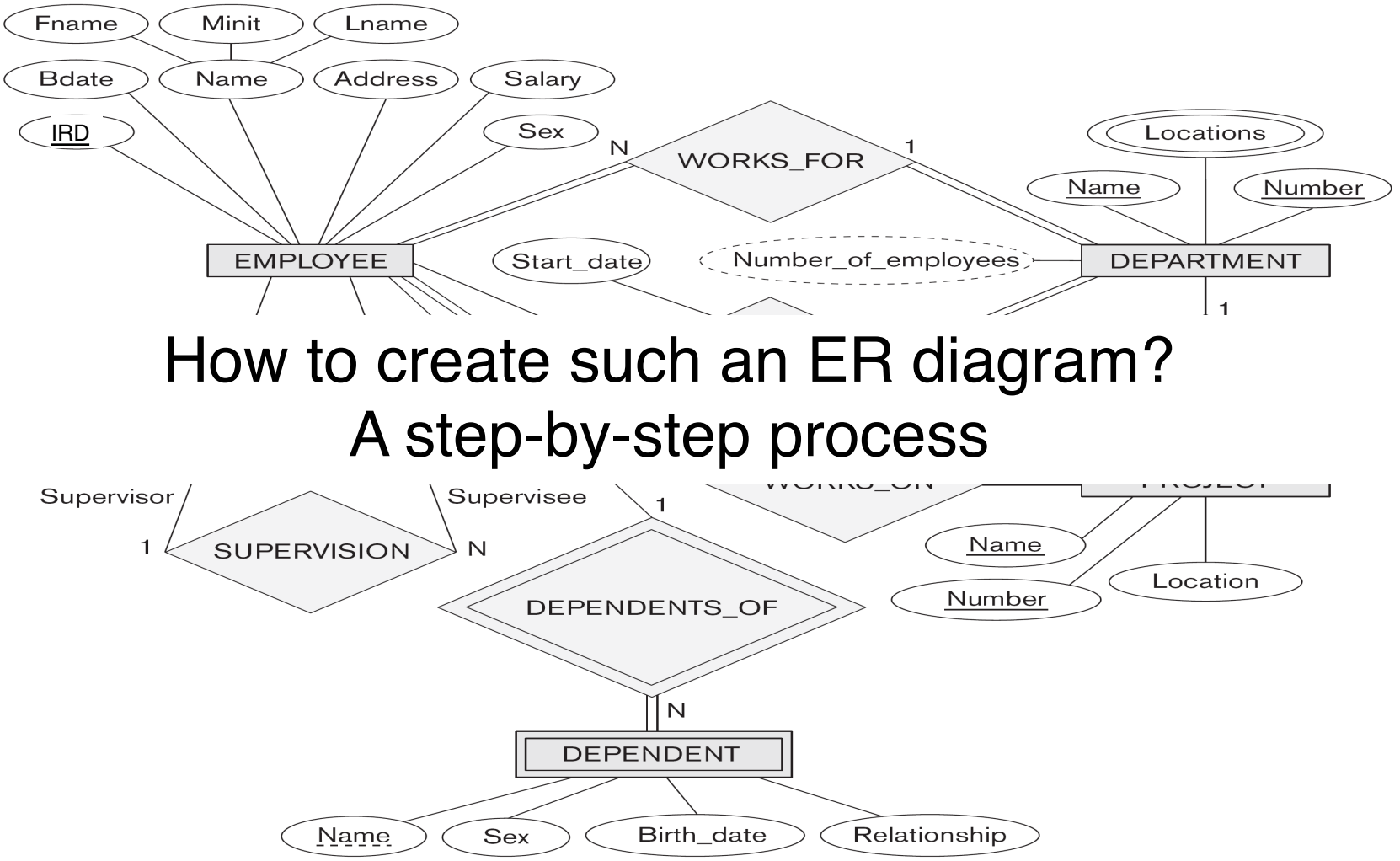




An Example Application

- Departments
  - Unique name & number
  - A particular employee manages a department
  - Start date when the manager began managing
  - Department may have several locations
- Projects
  - Unique name & number
  - Single location
  - Controlled by a department
- Employees
  - Name
  - IRD
  - Address
  - Salary, gender & birth date
  - Works for one department, but may work on several projects
  - Track number of hours per week that an employee works on each project
  - Track the direct supervisor of each employee
- Dependents
  - First name
  - gender & birth date
  - Relationship to employee

ER Diagram for Example Application



How to create such an ER diagram?

A step-by-step process

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.

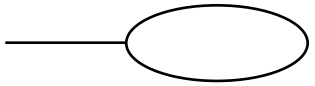
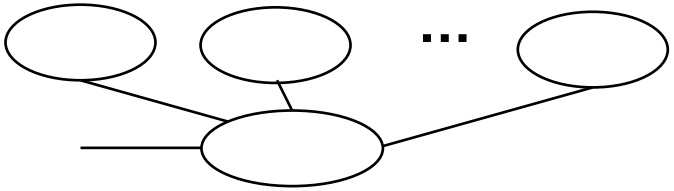
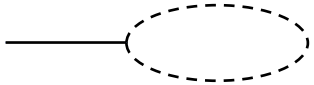
Entities and Attributes

- Entity: a *thing* in real word with **independent existence**
  - **Physical** existence (person, car, house, etc)
  - **Conceptual** existence (job, flight, etc)
- Attribute: the particular properties of an entity
  - Simple (atomic)
  - Composite (can be further divided into subparts)
  - Single-valued
  - Multi-valued
  - Stored
  - Derived

Entity Types, Entity Sets, Keys & Domains

- **Entity type**: a collection of entities with the same attributes
- Key attribute
  - **Unique** value
  - A simple or a composite attribute
  - An entity type can have more than one key attribute
- Domains
  - A value set associated with each attribute, which specifies the set of values that may be assigned to that attribute for each individual entity
  - e.g., domain for age attribute: 0-120

ER-Diagram Notation (Entity & Attribute)

Symbol	Meaning
	Entity Type
	Attribute
	Key Attribute
	Multivalued Attribute
	Composite Attribute
	Derived attribute

Initial Conceptual Design(COMPANY Database)

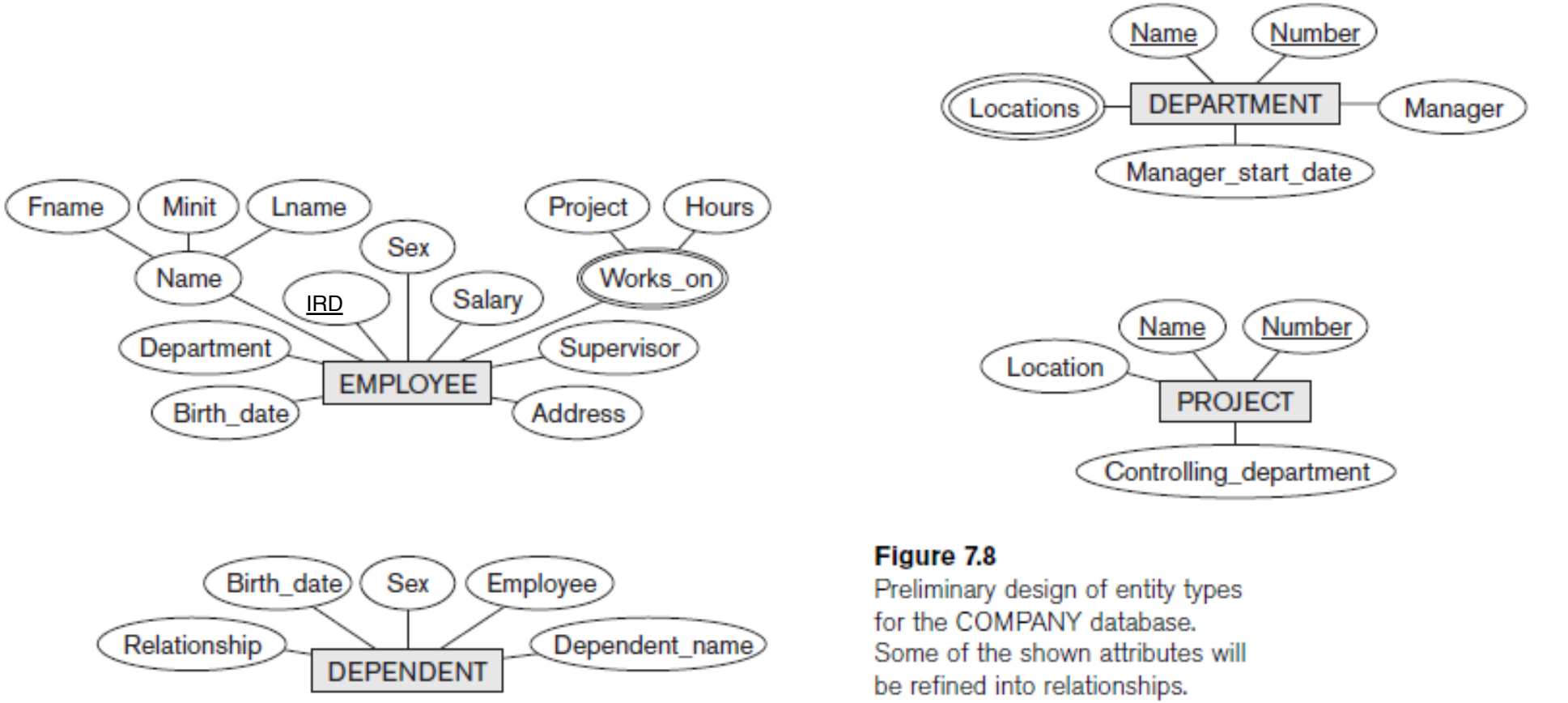


Figure 7.8
Preliminary design of entity types for the COMPANY database. Some of the shown attributes will be refined into relationships.

Relationships and Relationship Types

- A *relationship* exists whenever an attribute of one entity type **refers to** another entity type.

```
Each employee works for one department
```

- In the ER model, references (between entity types) should be represented as relationships, not attributes
  - A *relationship type R* among *n* entity types $E_1, E_2, \ldots, E_n$ defines a set of associations - or a relationship set - among entities from these types.
  - Each item in *R* is called a *relationship instance.*
  - Each of the entity types $E_1, E_2, \ldots, E_n$ is said to participate in the relationship type *R*.

Relationship Example

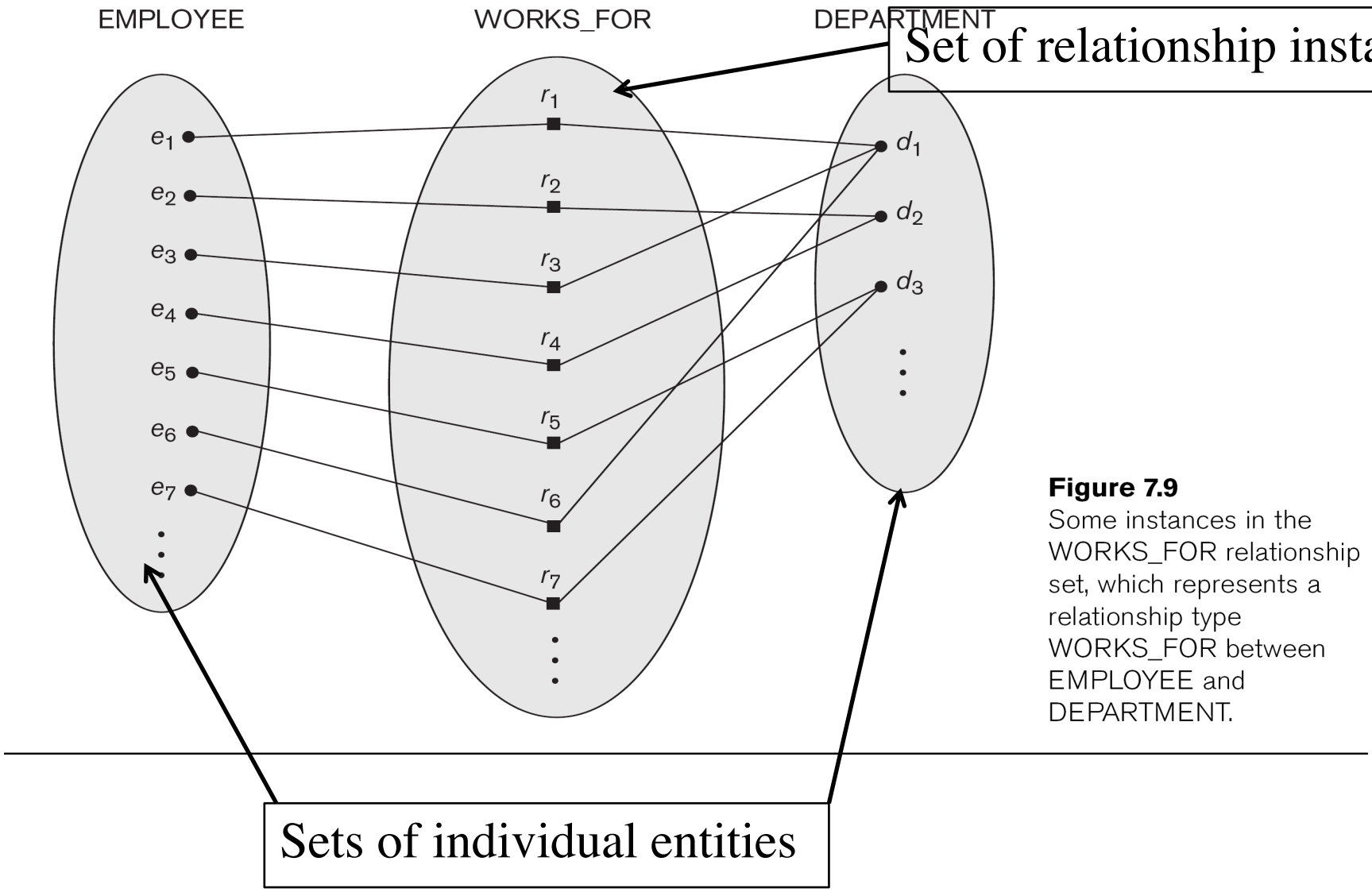


Figure 7.9
Some instances in the WORKS_FOR relationship set, which represents a relationship type WORKS_FOR between EMPLOYEE and DEPARTMENT.

Relationship Degree

- Degree of a relationship type: number of participating entity types
  - Binary (degree of two)
  - Ternary (one of degree three)

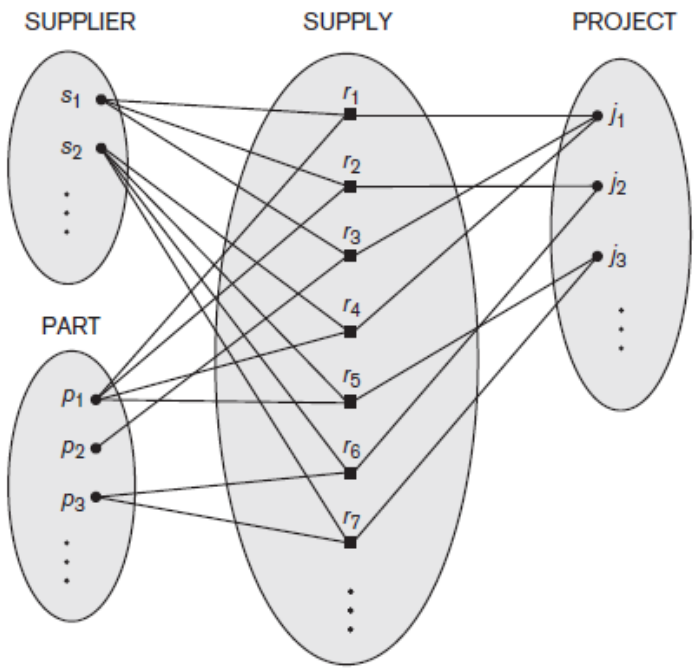


Figure 7.10
Some relationship instances in the SUPPLY ternary relationship set.

- Relationships as attributes
  - *Department* as an attribute of EMPLOYEE

X

Role Names

- Each entity type that participates in a relationship type plays a particular role in the relationship.
- Role Name
  - Signifies the role of the participating entity
  - Helps to explain what the relationship means
  - Not necessary where participating entity types are distinct
  - **Recursive** relationship: same entity type participates more than once

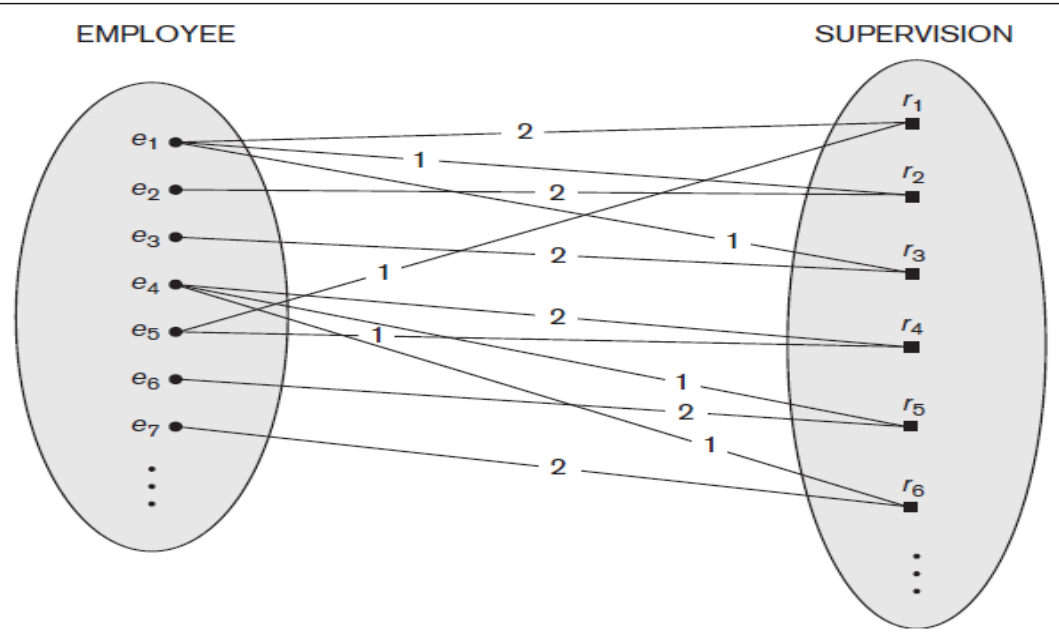


Figure 7.11
A recursive relationship SUPERVISION between EMPLOYEE in the *supervisor* role (1) and EMPLOYEE in the *subordinate* role (2).

Cardinality ratios for binary relationships

- Cardinality specifies the **maximum** number of relationship instances that an entity can participate in
  - 1:1
  - 1:N
  - M:N

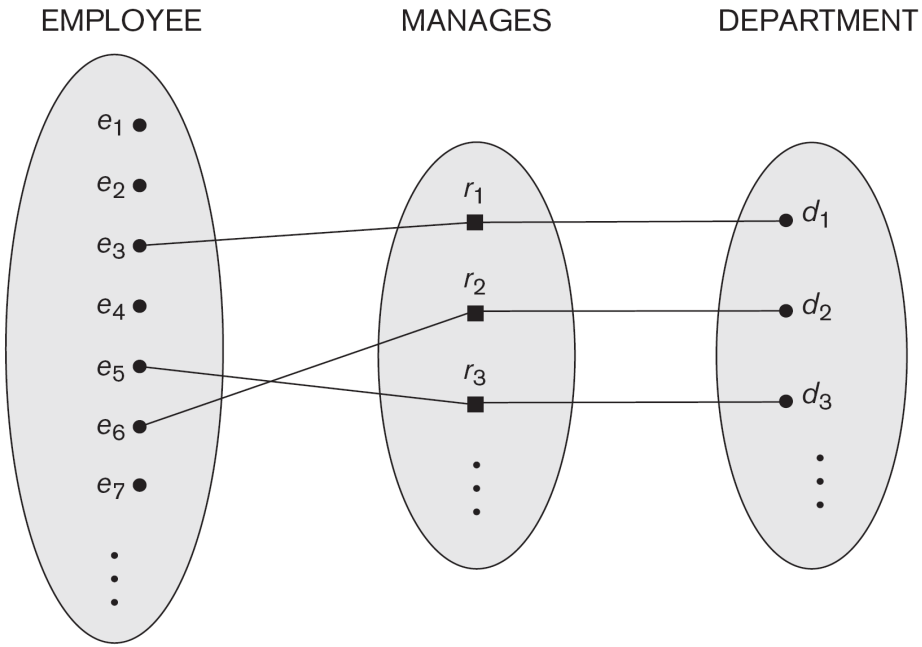


Figure 7.12
A 1:1 relationship, MANAGES.

[9]N stands for *any number* of related entities (zero or more).

Example 1:N Relationship

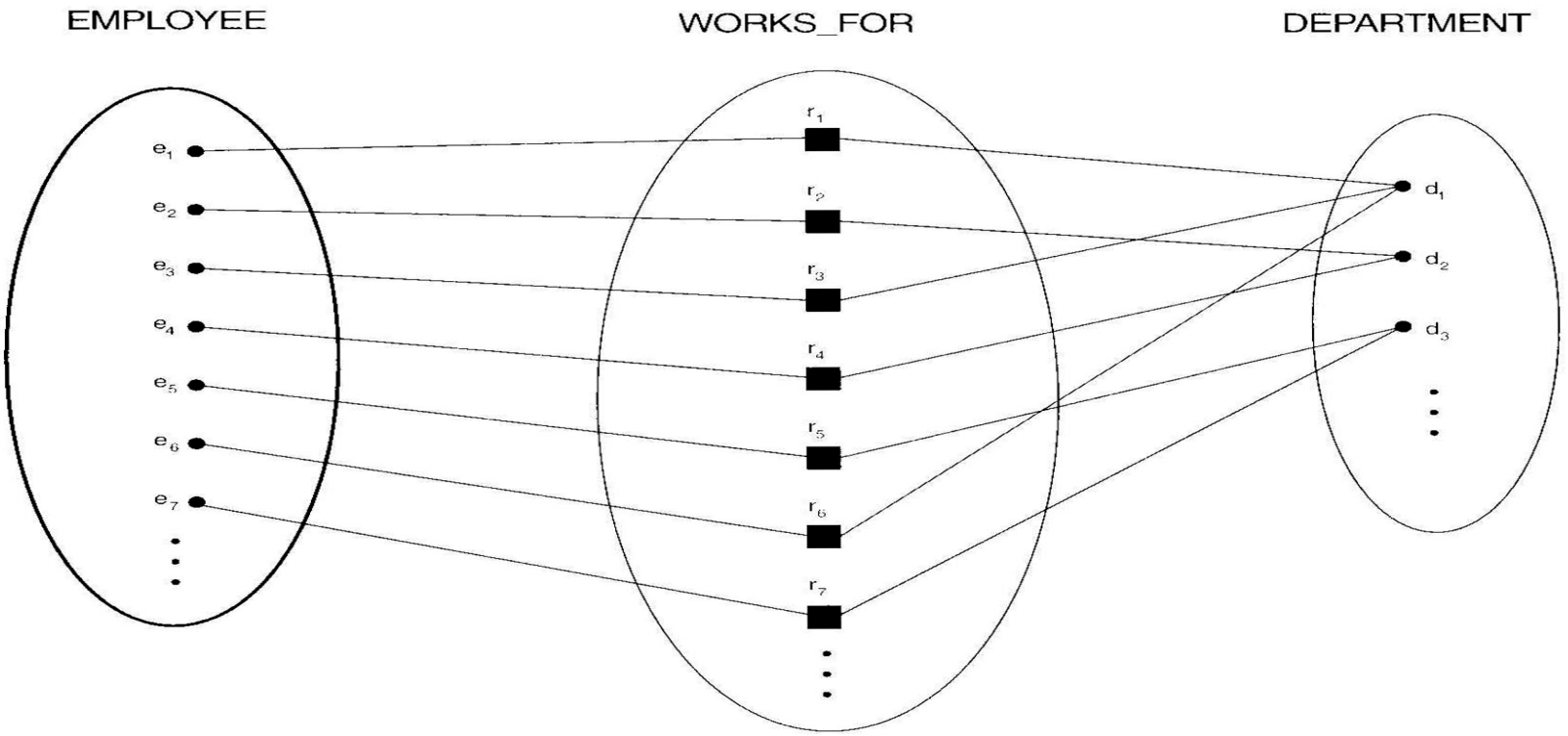


Figure 3.9 Some instances of the WORKS_FOR relationship between EMPLOYEE and DEPARTMENT.

Example M:N Relationship

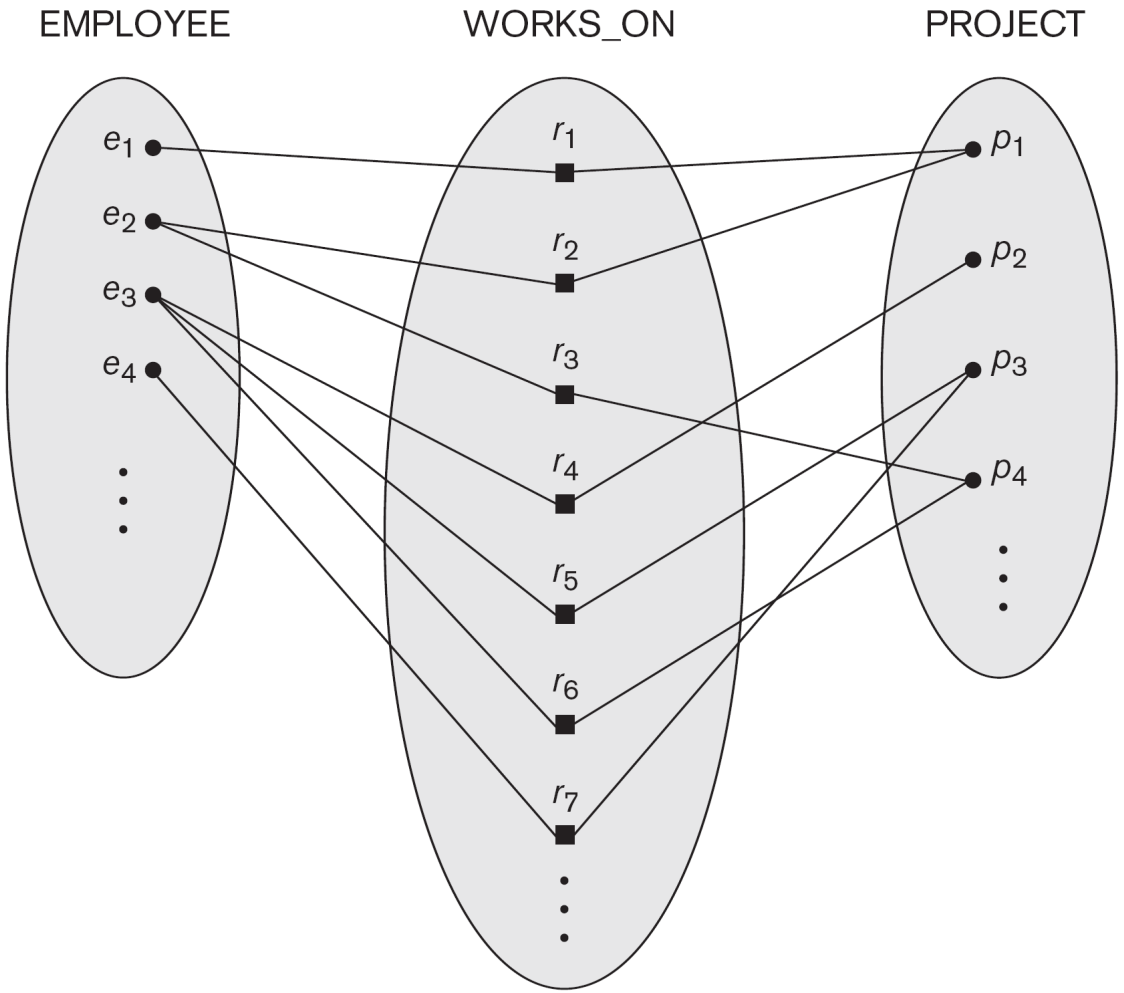


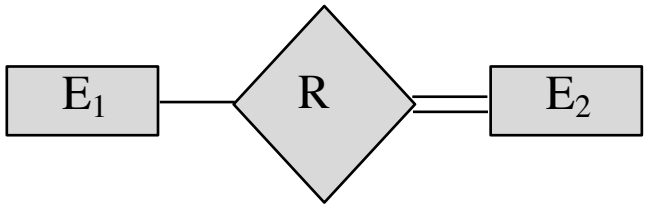
Figure 7.13
An M:N relationship, WORKS_ON.

Participation Constraints

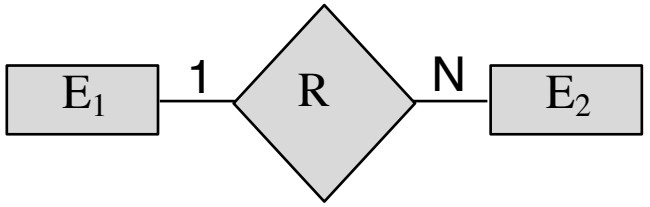
- Participation constraints
  - Specifies whether the existence of an entity depends on its being related to another entity via the relationship type
  - Specifies the *minimum* number of relationship instances that each entity can participate in (Minimum Cardinality Constraint)
- Types of Participation constraints
  - **Total** (Existence dependency)
    - If every employee must work for a department, then an employee entity can exist only if it participates in a WORKS_FOR relationship.
    - Every entity in the "total set" of employee entities must be related to a department entity by the WORKS_FOR relationship.
    - **In ER diagram, it is shown as a double line**
  - **Partial**
    - Every employee is not expected to manage a department.
    - A "part of the set" of employees are related to a department by the MANAGES relationship.
    - **In ER diagram, it is shown as a single line**

ER-Diagram Notation (Relationship)

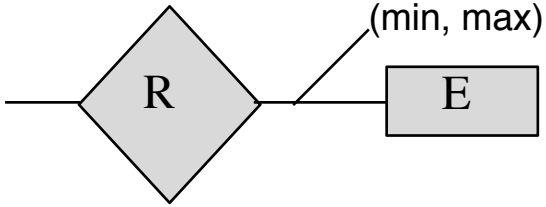
Symbol



Total Participation of E_2 in R



Cardinality Ratio 1:N for E_1:E_2 in R



Structural Constraint (min, max) on Participation of E in R

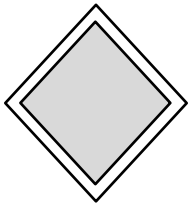
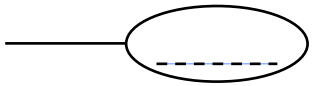
Attributes of Relationship Types

- Relationship types can have attributes, similar to entity types
- In some cases, attributes of relationship types can be migrated to participating entity types
  - 1:1 relationship type
    - attributes can be migrated to one of the participating entity types
    - e.g., *Start_date* attribute of the MANAGES relationship
  - 1:N relationship type
    - attributes can be migrated only to the entity type on the N-side of the relationship
    - e.g., *Start_date* attribute of the WORKS_FOR relationship
  - M:N relationship type
    - some attributes may be determined by the combination of participating entities, not by any single entity. **These attributes must be specified as relationship attributes**.
    - the *hours* attribute of the WORKS_FOR relationship

Weak Entity Types

- Weak entity types
  - do not have key attributes of their own.
  - Identified by being related to specific entities from another entity type in combination with one of their attributes values.
  - the other type is called the **identifying or owner entity type**, and the relationship is called the **identifying relationship**
  - Has a **total participation** constraint with respect to its owning entity type
  - e.g. the DEPENDENT entity type
- Partial key
  - A set of attributes that can uniquely identify the weak entities that are related to the same owner entity
  - A weak entity normally has a **partial key**

ER-Diagram Notation (Weak Entity)

Symbol	Meaning
	Weak entity
	Identifying relationship
	Partial key attribute

ER Diagram Revisited

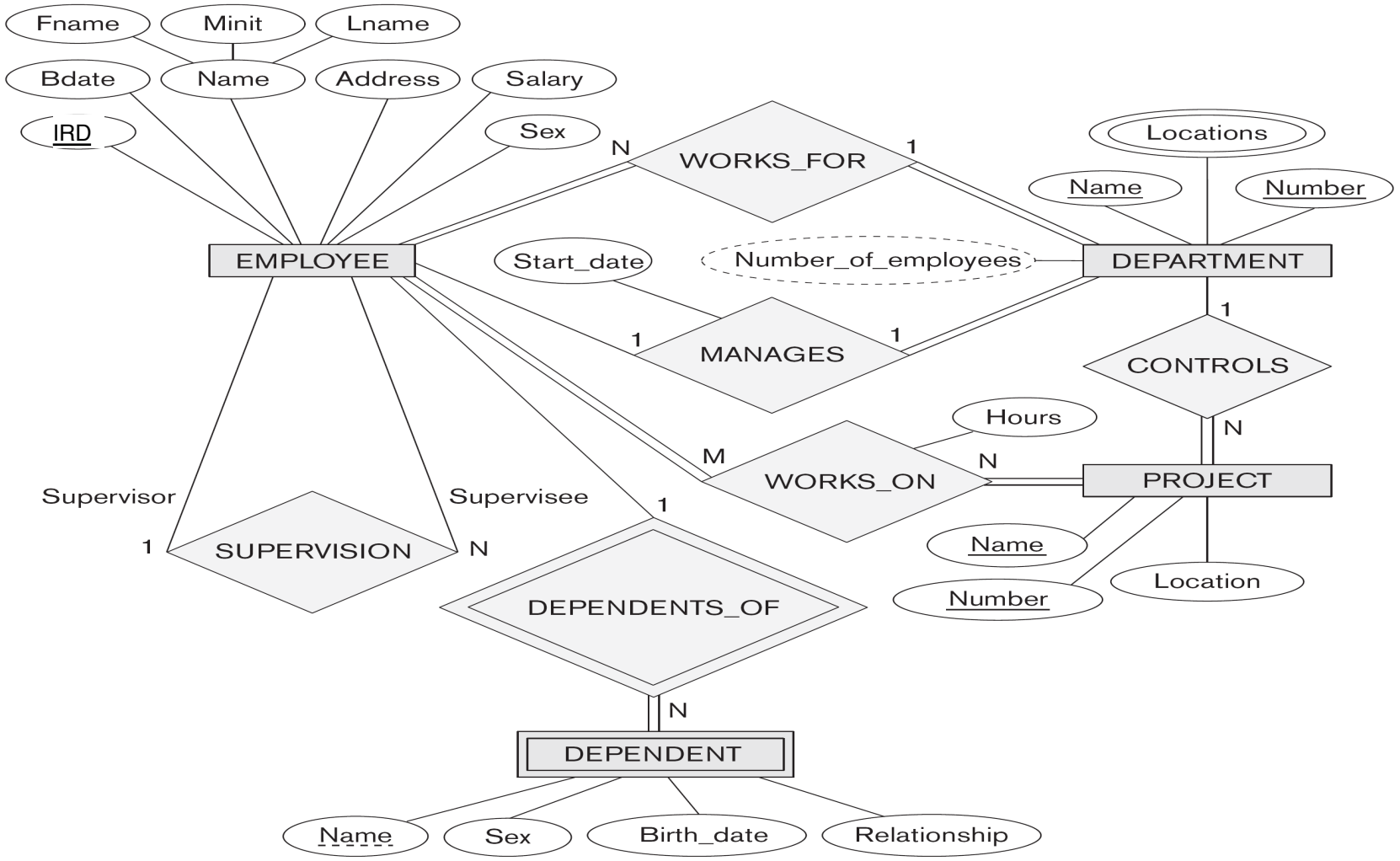


Figure 7.2
An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter and is summarized in Figure 7.14.

ER Diagram Revisited (continued)

- MANAGES
  - 1:1 between EMPLOYEE & DEPARTMENT
  - EMPLOYEE participation is partial
  - DEPARTMENT participation is total
  - StartDate attribute
- WORKS_FOR
  - 1:N between DEPARTMENT & EMPLOYEE
  - Both participations are total
- CONTROLS
  - 1:N between DEPARTMENT & PROJECT
  - PROJECT participation is total
  - DEPARTMENT participation is partial
- SUPERVISION
  - 1:N between EMPLOYEE (in supervisor role) & EMPLOYEE (in supervisee role)
  - Both participations determined to be partial after users indicate not every employee has a supervisor

ER Diagram Revisited (continued)

- WORKS_ON
  - M:N between EMPLOYEE & PROJECT
  - Both participations are total
  - Hours attribute

- DEPENDENTS_OF
  - 1:N between EMPLOYEE & DEPENDENT
  - EMPLOYEE participation is partial
  - DEPENDENT participation is total

- *Which gives us the completed ER diagram*

Summary

- Definitions
  - Entity, attribute, types of attributes, entity type, entity set, domain
  - relationship, relationship type, relationship degree, cardinality, role, participation
  - weak entity type, identifying relationship
- ER diagram

- Miniworld to ER diagram