# THE ENTITY-RELATIONSHIP (ER) MODEL

CHAPTER 7 (6/E)

CHAPTER 3 (5/E)

# CHAPTER 7 OUTLINE

- Using High-Level, Conceptual Data Models for Database Design
- **Entity-Relationship (ER) model**
  - Popular high-level conceptual data model
- **ER diagrams**
  - Diagrammatic notation associated with the ER model

# STEPS IN DATABASE DESIGN

- **Requirements collection and analysis**
  - DB designers interview prospective DB users to understand and document data requirements
    - Data requirements
    - Functional requirements of the principal applications
- **Conceptual** or **logical DB design**
  - Description of data requirements
    - Detailed descriptions of components and constraints
    - Transformed into implementation data model
  - Result: DB schema in implementation data model of DBMS
- **Physical DB design**
  - Internal storage structures, file organizations, indexes, access paths, and physical design parameters for the DB files
- **External** or **view design**

# A SAMPLE DATABASE APPLICATION

- Requirements gathered for COMPANY
  - Employees, departments, and projects
  - Company is organized into departments
  - Department controls several projects
  - Employee: require each employee's name, Social Security number, address, salary, sex (gender), and birth date
  - Keep track of the dependents of each employee

# ER MODEL OVERVIEW

- ER model describes data in terms of:
  - **Entities** and **entity sets**
    - Objects
  - **Relationships** and **relationship sets**
    - Connections between objects
  - **Attributes**
    - Properties that characterize or describe entities or relationships

# ENTITIES AND ATTRIBUTES EXAMPLE

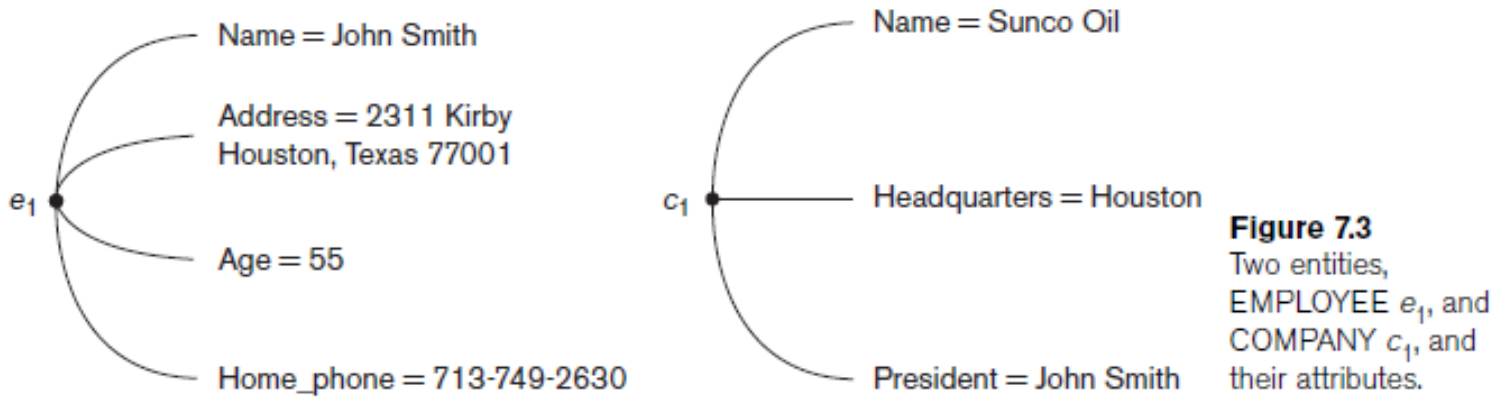



**Figure 7.3**
Two entities, EMPLOYEE $e_1$, and COMPANY $c_1$, and their attributes.

# ENTITY SETS

- **Entity type** or **set**
  - Collection (or set) of similar entities that have the same attributes

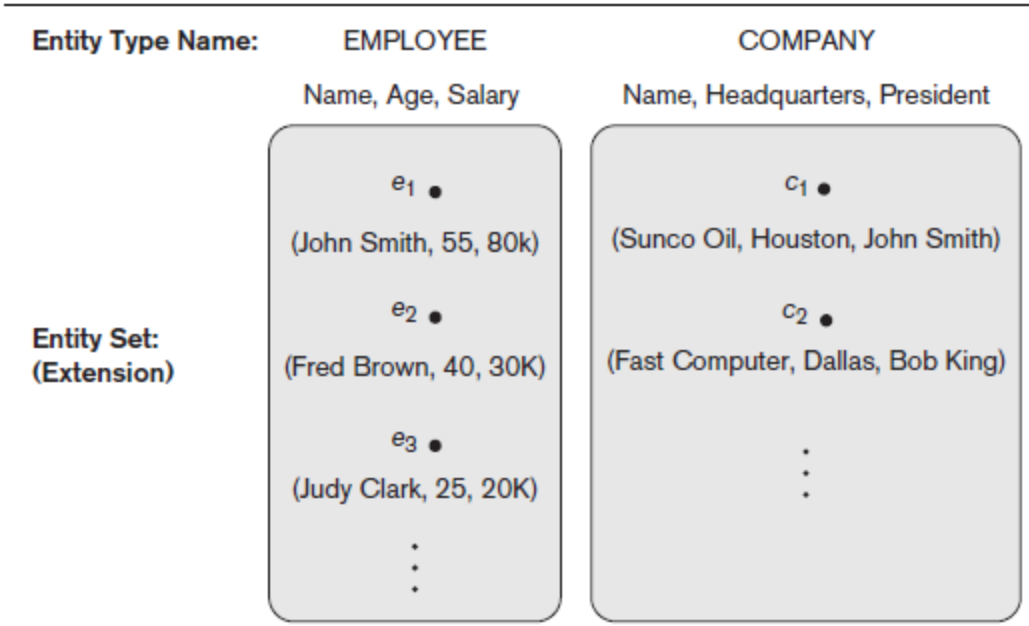



**Figure 7.6**
Two entity types, EMPLOYEE and COMPANY, and some member entities of each.

- ER model defines *entity sets,* not individual entities
- But entity sets described in terms of their attributes

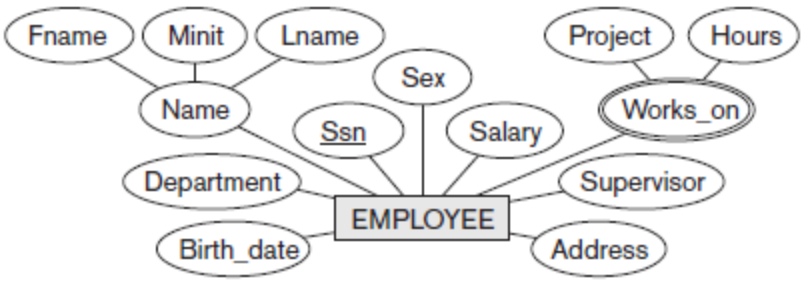
# CATEGORIES OF ATTRIBUTES

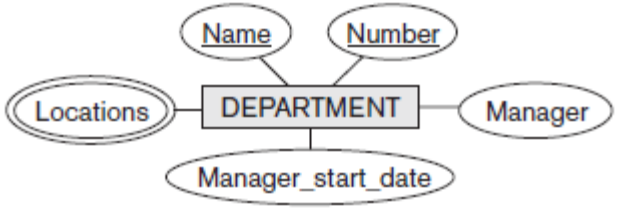
- **Simple** (atomic) vs. **composite** attributes
- **Single-valued** vs. **multivalued** attributes
- **Stored** vs. **derived** attributes

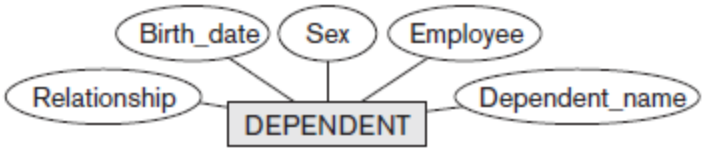
- **Key** or **unique** attributes
  - Attribute values constrained to be distinct for individual entities in entity set

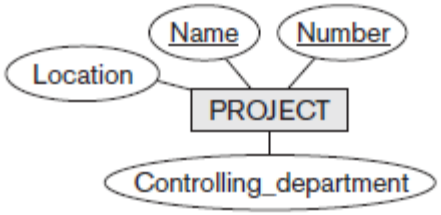
# INITIAL ER DIAGRAM FOR COMPANY

- Four entity types
- Most attributes are simple, single-valued, and stored
  - Works_on and Locations are multivalued
  - Employee's Name is composite
- Employee has one key, department and project have two keys, dependent has none



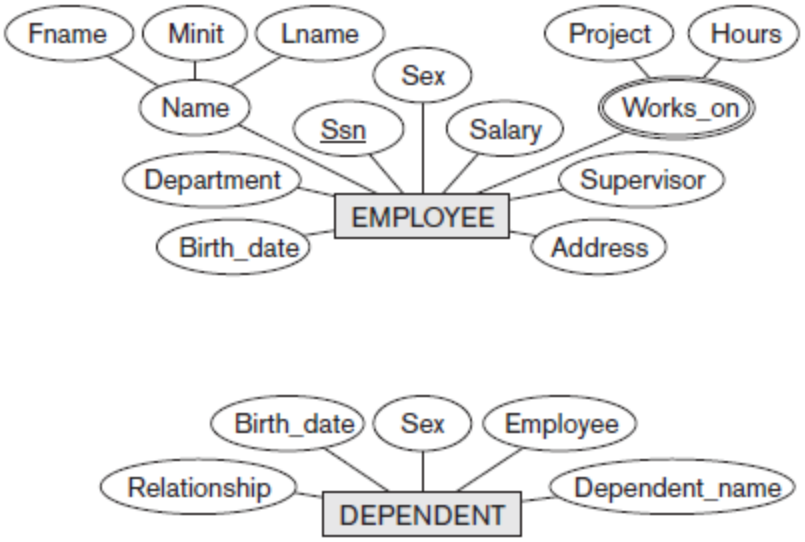



# WEAK ENTITY TYPES

- Entity types that do not have key attributes of their own
  - Identified by their relationship to specific entities from another entity type

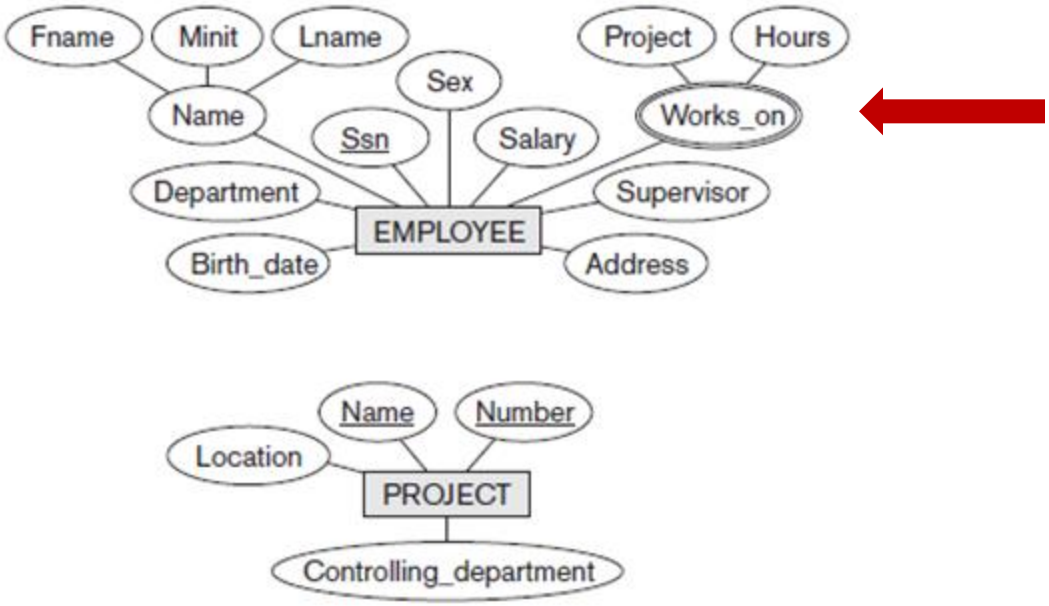



- Dependent is meaningless in COMPANY DB independently of Employee
- Identified by relationship to Employee
- Dependent_name discriminates one dependent from other dependents for the same employee

- **Identifying relationship**
  - Relates a weak entity type to its owner

# RELATIONSHIPS IN GENERAL

- **Relationship**
  - Interaction between entities
  - Indicator: an attribute of one entity refers to another entity
    - Represent such references as relationships not attributes



# RELATIONSHIPS

- **Relationship**
  - Interaction between entities
  - Indicator: an attribute of one entity refers to another entity
    - Represent such references as relationships not attributes
- **Relationship type** *R* among *n* entity types $E_1$, $E_2$, ..., $E_n$
  - Defines a set of associations among entities from these entity types
- **Relationship instance** $r_i$
  - Each $r_i$ associates *n* individual entities ($e_1$, $e_2$, ..., $e_n$)
  - Each entity $e_j$ in $r_i$ is a member of entity set $E_j$
  - Relationships uniquely identified by keys of participating entities
- **Degree** of a relationship type
  - Number of participating entity types
  - e.g., binary, ternary

# RELATIONSHIPS & RELATIONSHIP SETS

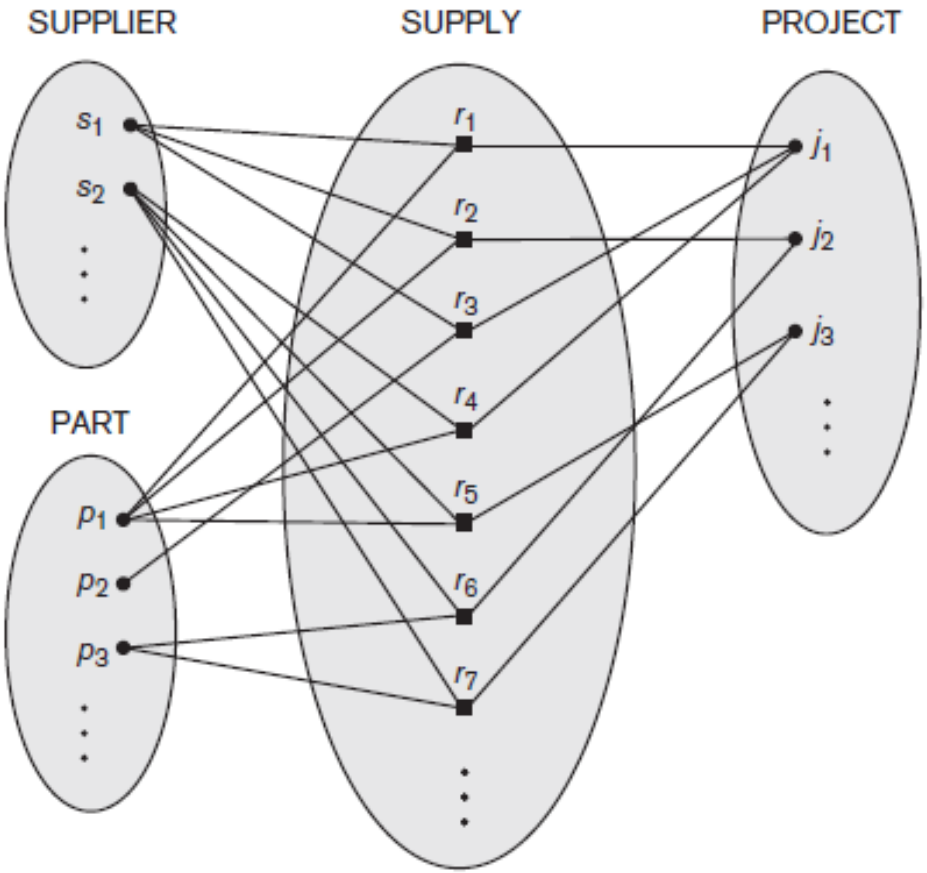
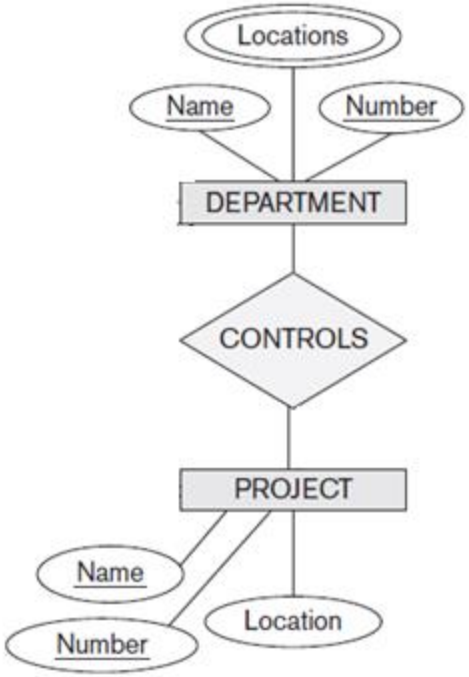



**Figure 7.10**
Some relationship instances in the SUPPLY ternary relationship set.

# DIAGRAMMING RELATIONSHIP TYPE

- Diamond for relationship type
- Connected to each participating entity type
  - Could be binary, ternary, or higher degree
- *Remember*:
  - Represents a *set* of entities of each type, some of which are related to entities of the other type(s)
  - Some entities might participate in several relationships
  - Some entities might not participate in the relationship at all



# RELATIONSHIPS WITH REPEATED ENTITY SETS

- Some relationships involve multiple entities from the same entity set
  - e.g., spouse (two persons), games (two teams)
  - e.g., recursive relationships, such as supervises (two employees)
- **Role name**
  - Signifies role that participating entity plays in relationship instance
  - Required when entity type participates multiple times in a relationship

# USING ROLE NAMES

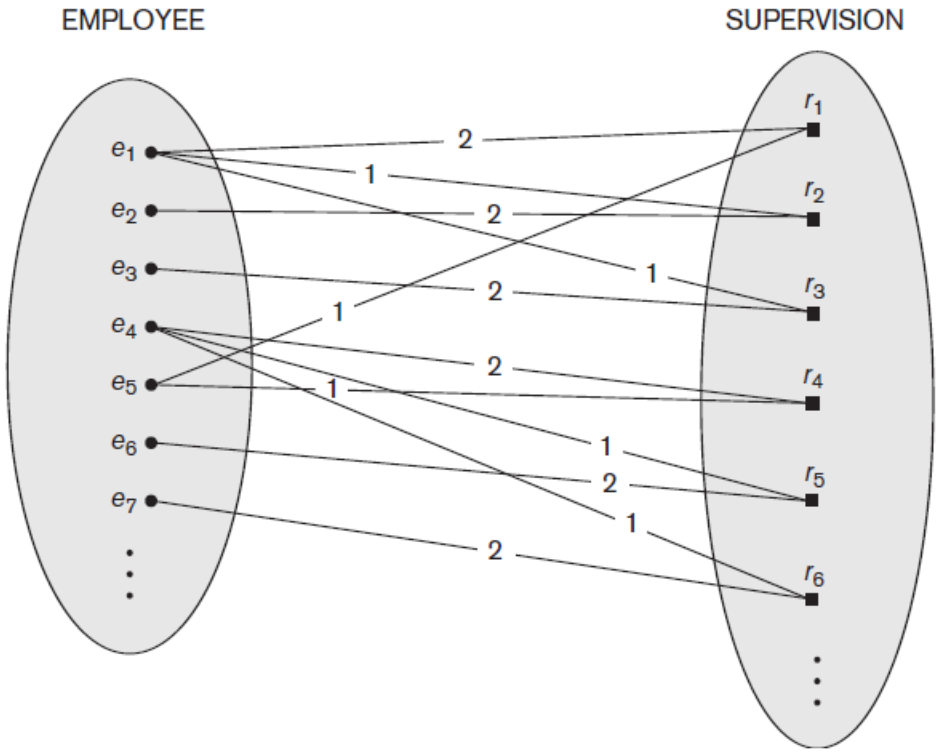
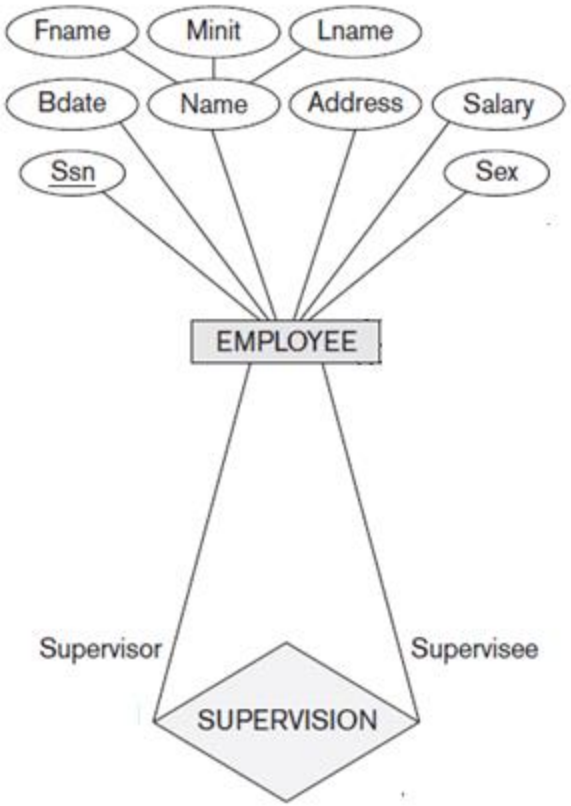



**Figure 7.11**
A recursive relationship SUPERVISION between EMPLOYEE in the *supervisor* role (1) and EMPLOYEE in the *subordinate* role (2).



# RELATIONSHIP CONSTRAINTS

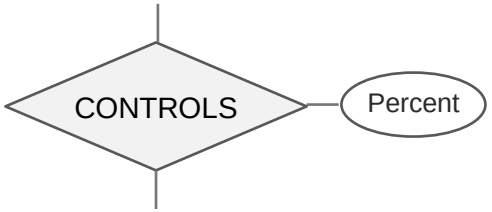
- **Cardinality ratio**
  - Specifies maximum number of relationship instances in which each entity can participate
  - Types 1:1, 1:N, or M:N
- **Participation constraint**
  - Specifies whether existence of entity depends on its being related to another entity
  - Types: **total** and **partial**
  - Thus minimum number of relationship instances in which entities can participate: thus1 for total participation, 0 for partial
  - Diagrammatically, use a double line from relationship type to entity type

---

- **Alternative: Structural constraint**
  - Generalization: specifying any min and max participation
    - Replaces cardinality ratio  numerals and single/double line notation
  - Associate a pair of integer numbers (min, max) with each participation of an entity type *E* in a relationship type *R*, where $0 \leq \text{min} \leq \text{max}$ and $\text{max} \geq 1$
  - max=N $\Rightarrow$ finite, but unbounded

# RELATIONSHIP ATTRIBUTES

- Relationship types can also have attributes
  - Property that depends on both/all participating entities
  - Example: Percentage of control that department has on a project




- Attributes of 1:1 or 1:N relationship types can be migrated to one of the participating entity types
  - For a 1:N relationship type, relationship attribute can be migrated only to entity type on N-side of relationship
  - Attributes on M:N relationship types must be specified as relationship attributes

# SUMMARY OF ER DIAGRAM SYMBOLS

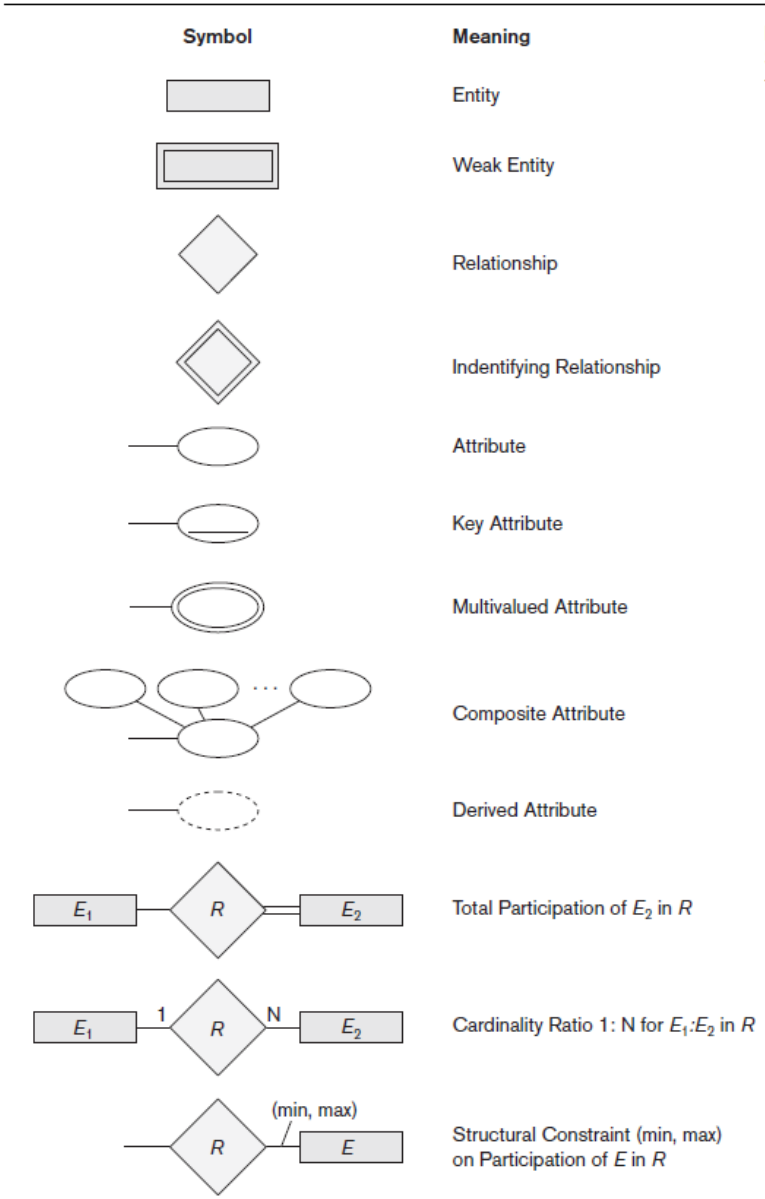
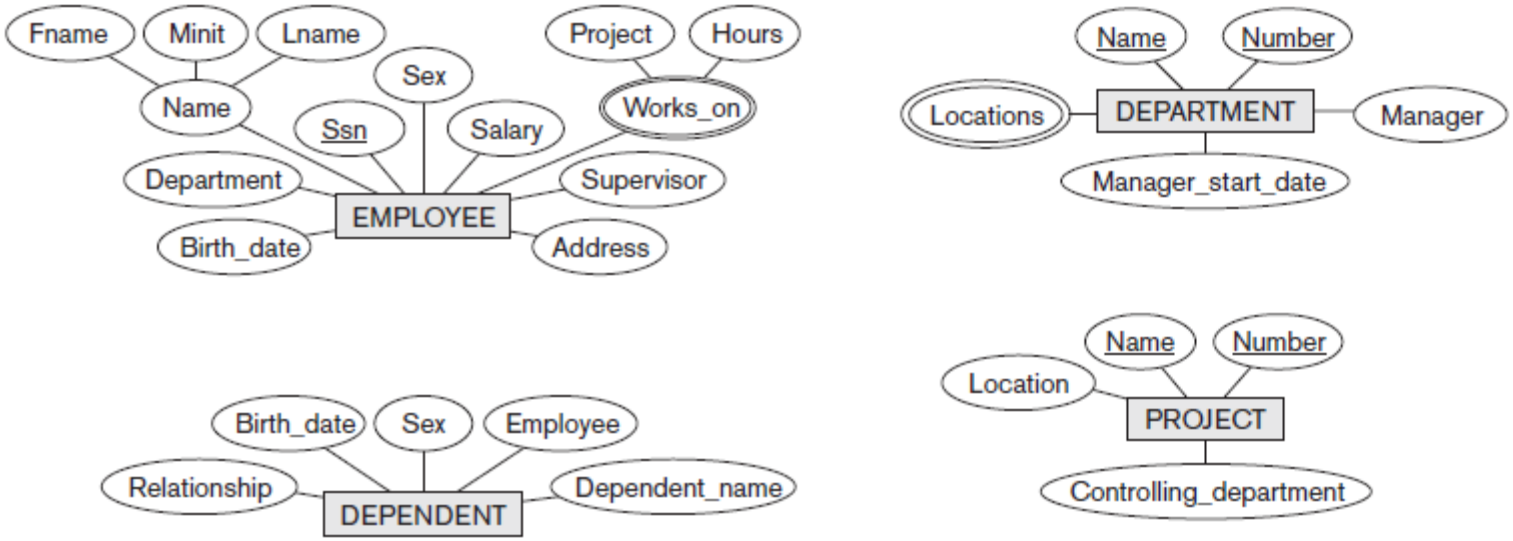



**Figure 7.14**
Summary of the notation for ER diagrams.

# REFINING EXAMPLE ER DESIGN

- Recall preliminary ER design




- Change attributes that reference entity types into relationship types
  - Weak entities use **identifying relationship**
- Determine cardinality ratio and participation constraints for each relationship type
  - Weak entity type always has structural constraint of (1,1) participation in identifying relationship

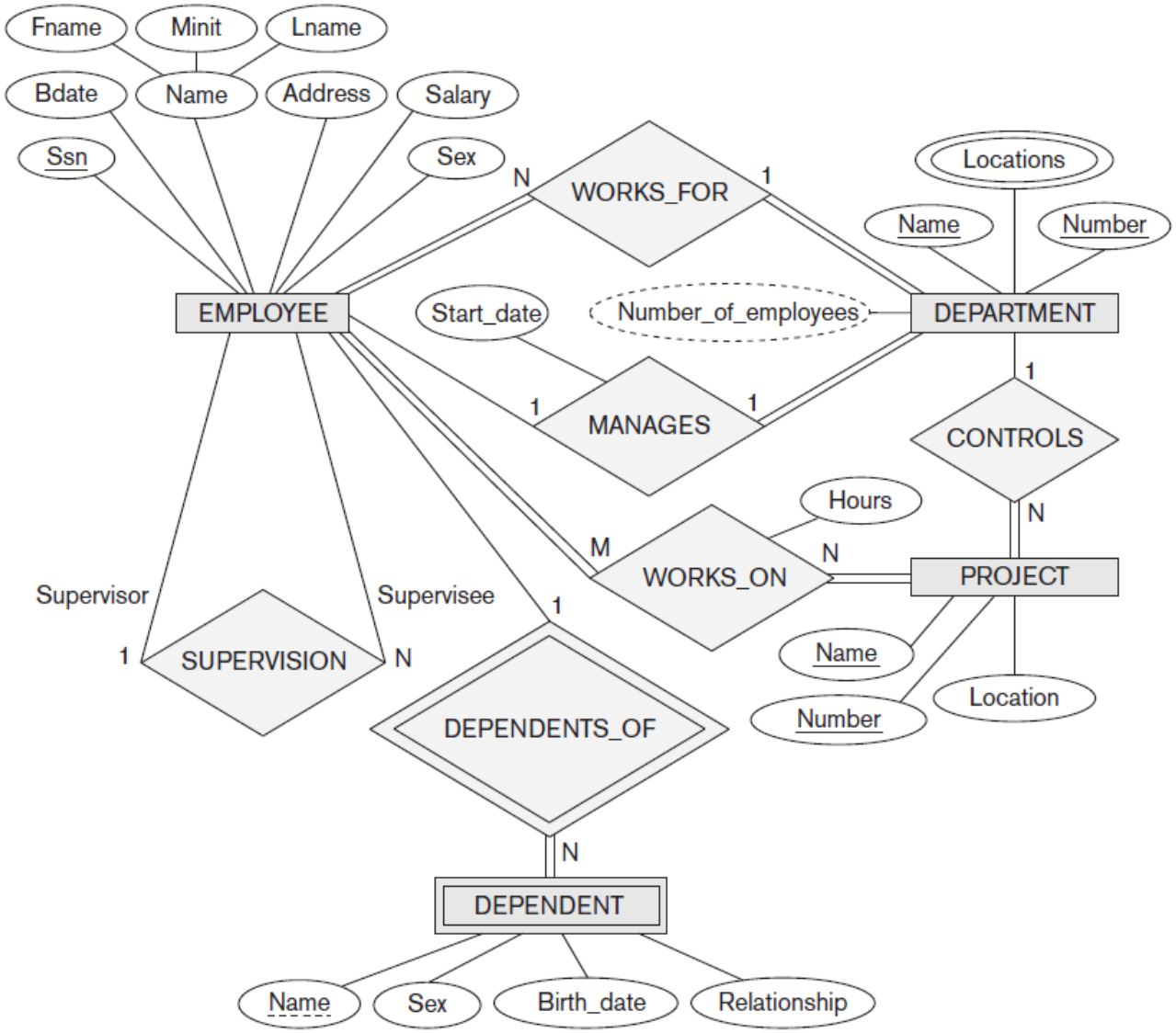



**Figure 7.2**
An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter and is summarized in Figure 7.14.

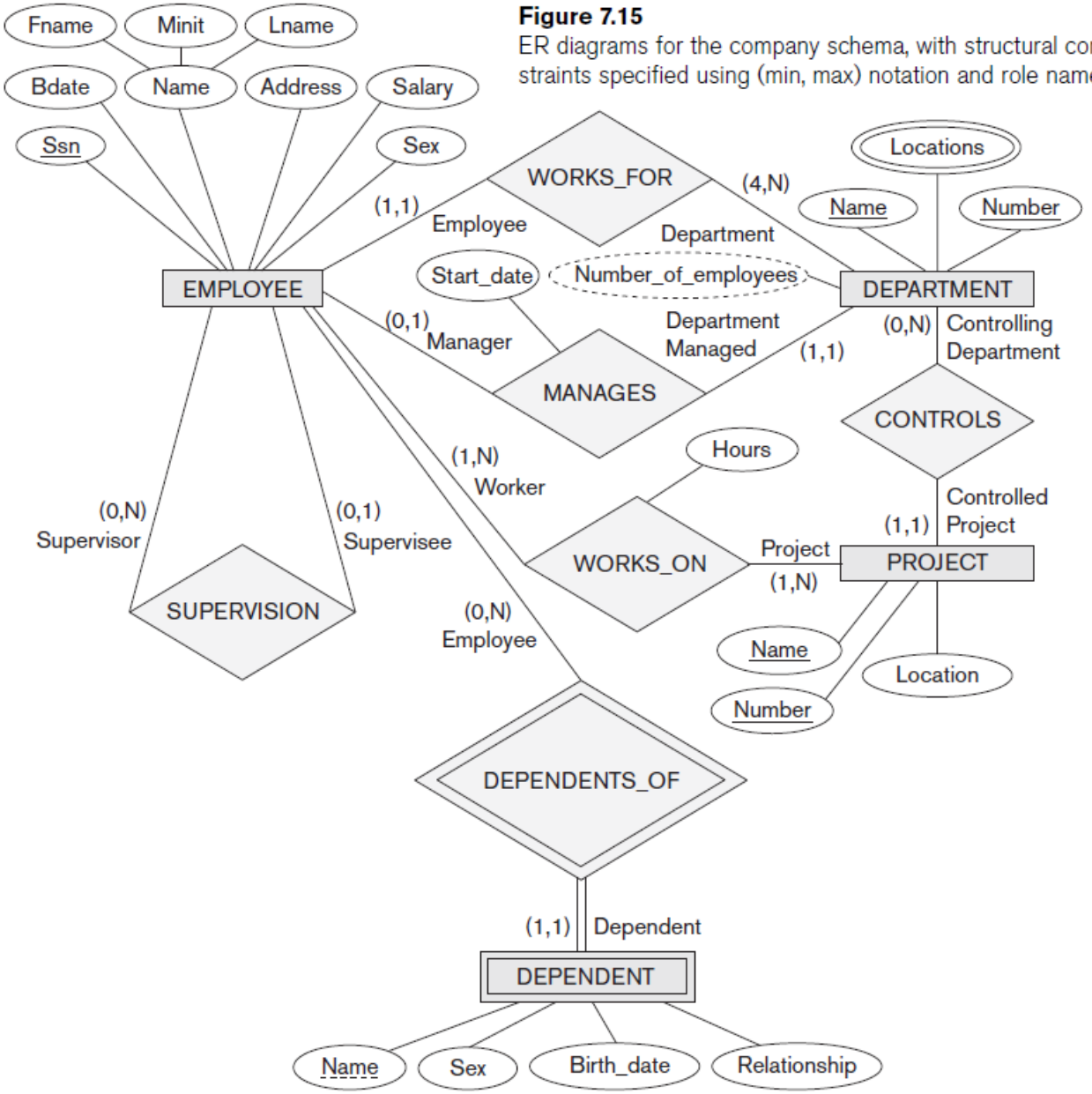



**Figure 7.15**
ER diagrams for the company schema, with structural constraints specified using (min, max) notation and role names.

# APPROPRIATE ER MODEL DESIGN

- Choose names that convey meanings attached to various constructs.
- Nouns give rise to entity type names
- Verbs indicate names of relationship types
  - Choose binary relationship names to make ER diagram readable from left to right and from top to bottom
- Review all attributes
  - Refine into a relationship if attribute references an entity type
  - Attribute that exists in several entity types may be better modelled as an independent entity type
- Entities that *must* participate in a relationship with another entity type and with cardinality constraint of 1 might be better modelled as weak entity

# REVIEW HIGH-DEGREE RELATIONSHIPS

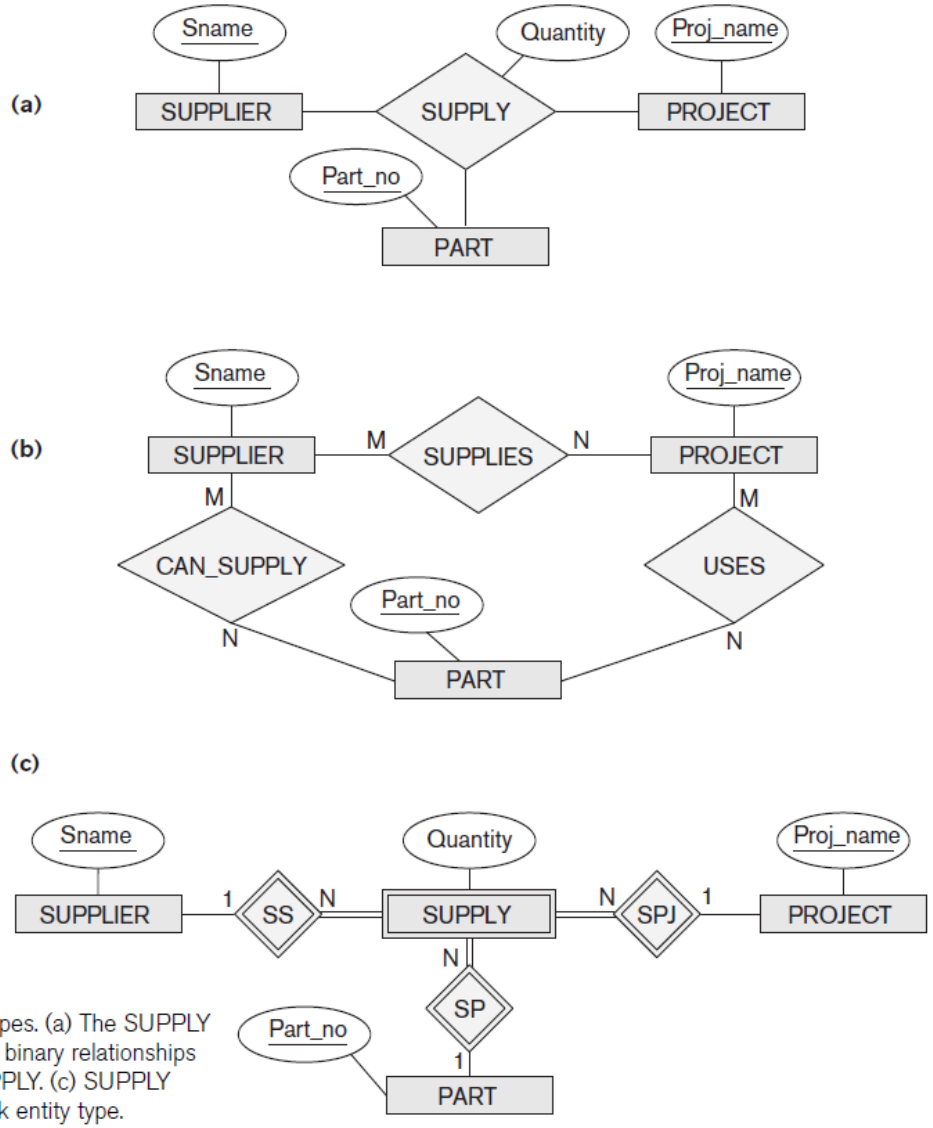



**Figure 7.17**
Ternary relationship types. (a) The SUPPLY relationship. (b) Three binary relationships not equivalent to SUPPLY. (c) SUPPLY represented as a weak entity type.

# SUMMARY

- Components of the Entity-Relationship Model
  - Entity Types, Entity Sets
  - Weak Entity Types
  - Relationship Types, Relationship Sets, Roles
  - Attributes, Attribute Classification, Keys
  - Structural Constraints
- ER diagrams represent ER models
- Appropriate ER design