CS502052 / Chapter 05

Designing Databases

Phuc H. Duong, *M.Sc.*

phuc@it.tdt.edu.vn

EER Model

Outline

1. Subclasses, Super-classes, and Inheritance
2. Specialization and Generalization
3. Constraints
4. UNIVERSITY EERD
5. Multiple vs. Single Inheritance

EER Model (1)

- **Enhanced ER (EER) model**
  - Created to design more accurate database schemas
  - Reflect the data properties and constraints more precisely
  - More complex requirements than traditional applications

EER Model (2)

- EER includes:
  - **Subclasses** and **super-classes**
  - **Specialization** and **generalization**
  - **Category** or **union type**
  - **Attribute** and **relationship inheritance**

EER Model (3)

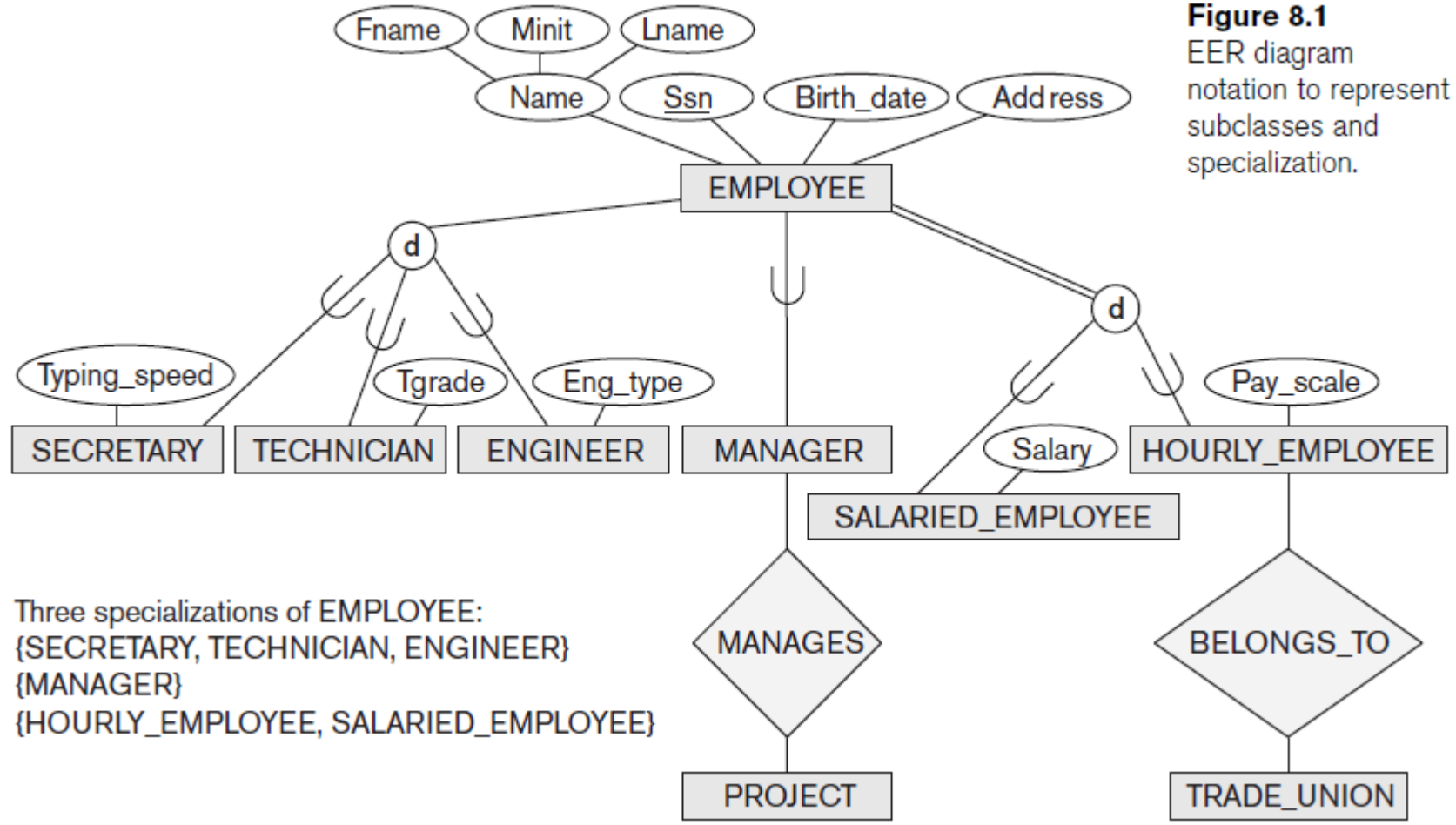


Figure 8.1
EER diagram notation to represent subclasses and specialization.

Specialization (1)

- **Specialization**
  - Process of defining a set of subclasses of an entity type
  - Defined on the basis of some distinguishing characteristic of the entities in the superclass
- **Subclass can define:**
  - Specific attributes
  - Specific relationship types

Specialization (2)

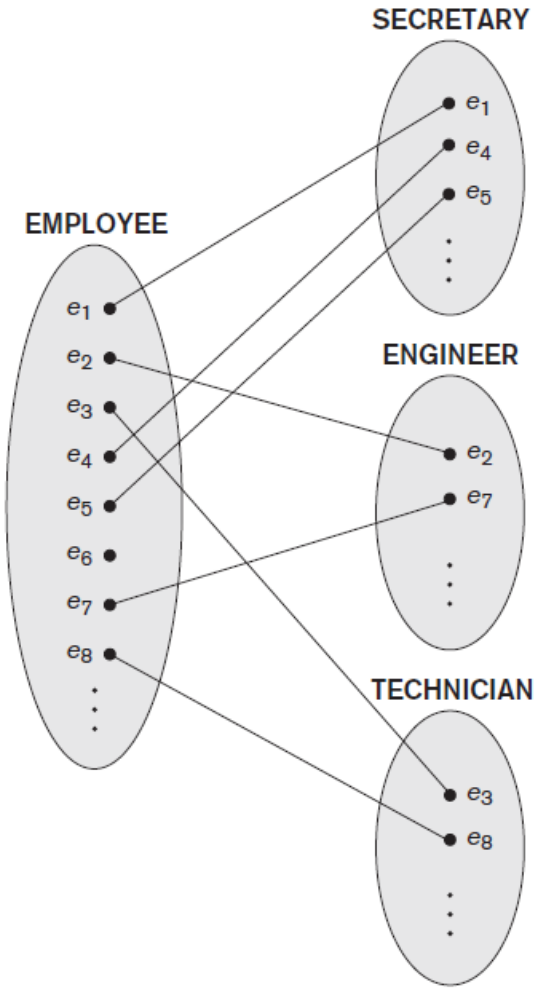


Figure 8.2
Instances of a specialization.

Specialization (3)

- Certain attributes of subclasses may apply to some but not all entities of the superclass
- Some relationship types may be participated in only by members of the subclass

Generalization

- Reverse process of abstraction
- **Generalize** into a single **superclass**
  - Original entity types are special subclasses
- **Generalization**
  - Process of defining a generalized entity type from the given entity types

Constraints (1)

- Constraints that apply to a single specialization or a single generalization
- Differences between specialization/generalization lattices and hierarchies

Constraints (2)

- **Dis-jointness constraint**
  - Specifies that the subclasses of the specialization must be disjoint
- **Completeness** (or **total-ness**) constraint
  - May be **total** or **partial**
- Dis-jointness and completeness constraints are independent

Constraints (3)

- **Completeness** (or **total-ness**) constraint
  - An occurrence of a Superclass must also have a corresponding Subclass occurrence
  - Represent as **double line**

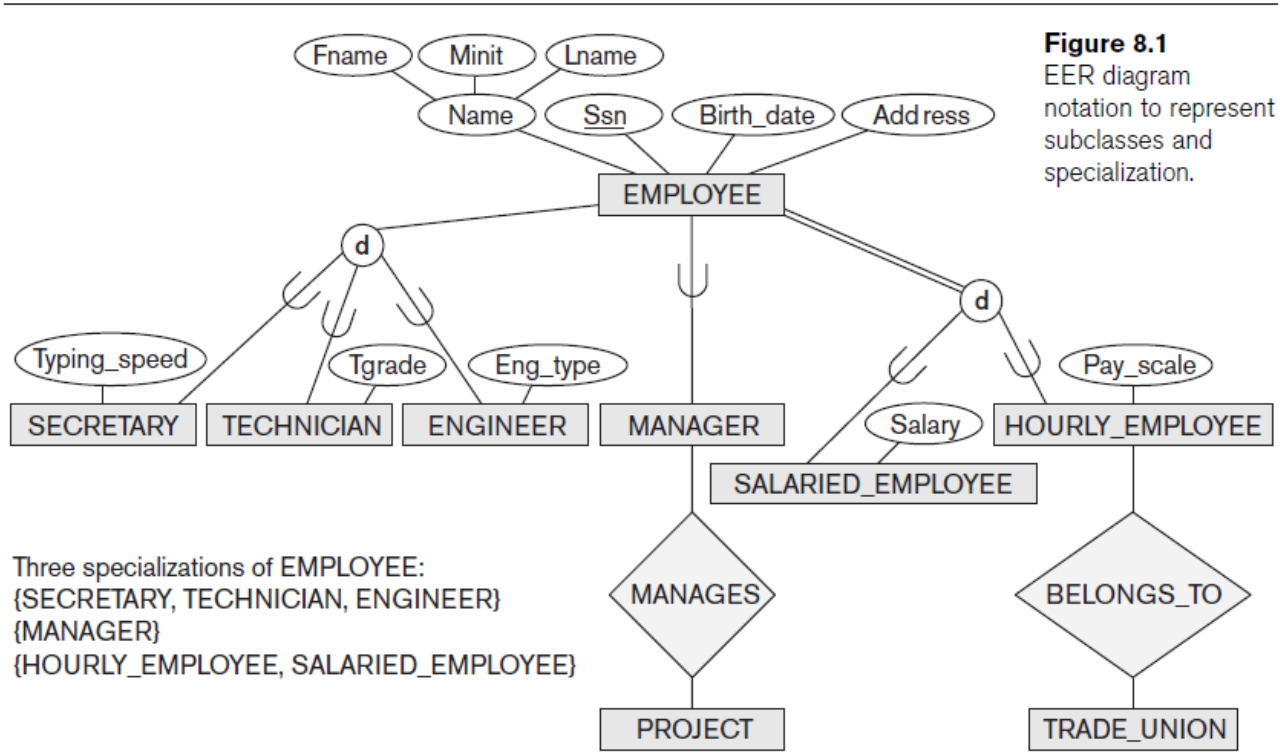


Figure 8.1
EER diagram notation to represent subclasses and specialization.

Constraints (4)

- Business rule states that an occurrence of a Superclass may also have more than one Subclass occurrence
  - **D**is-jointness constraint → NO
  - **O**verlap constraint → YES

Constraints (5)

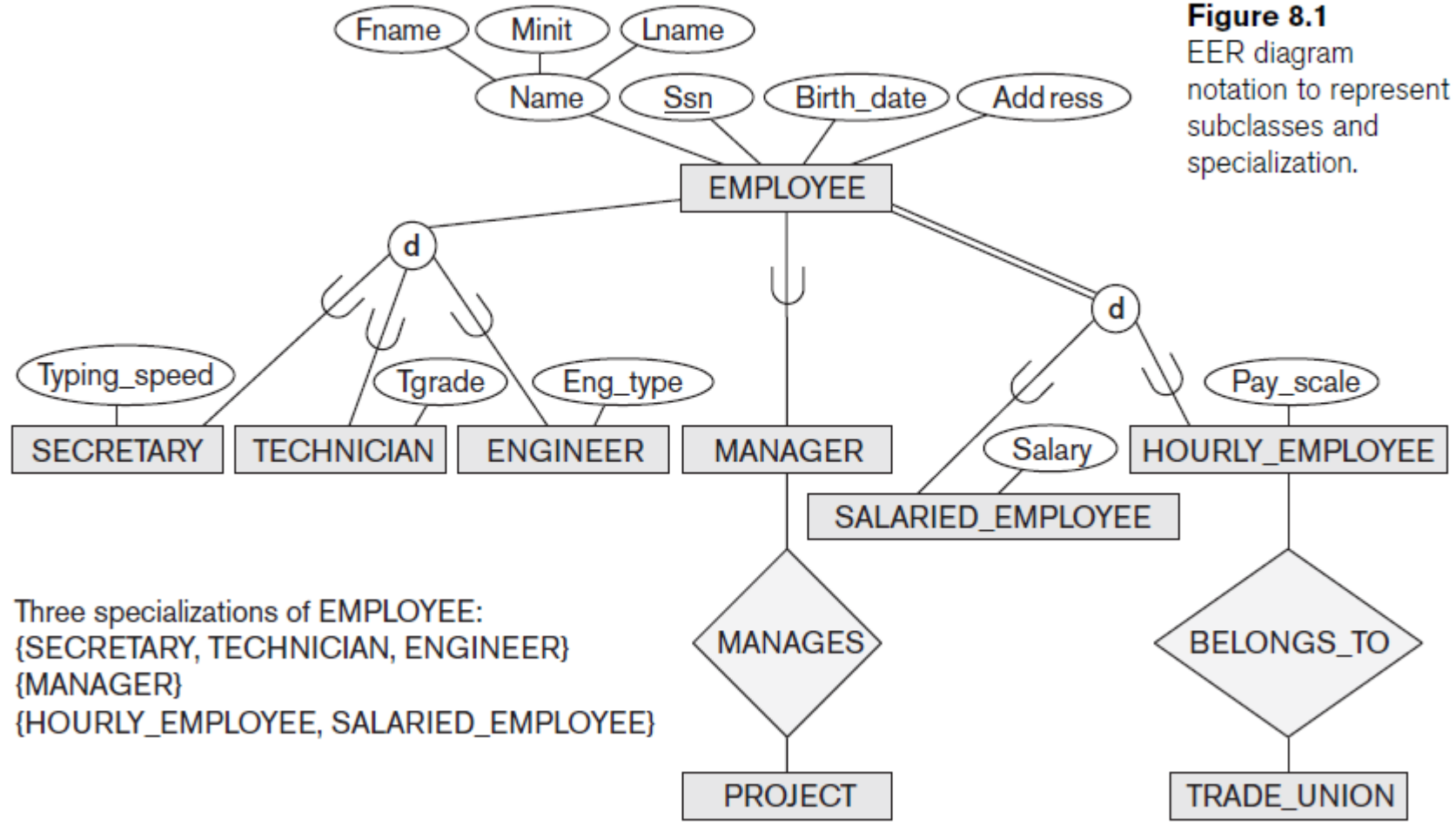
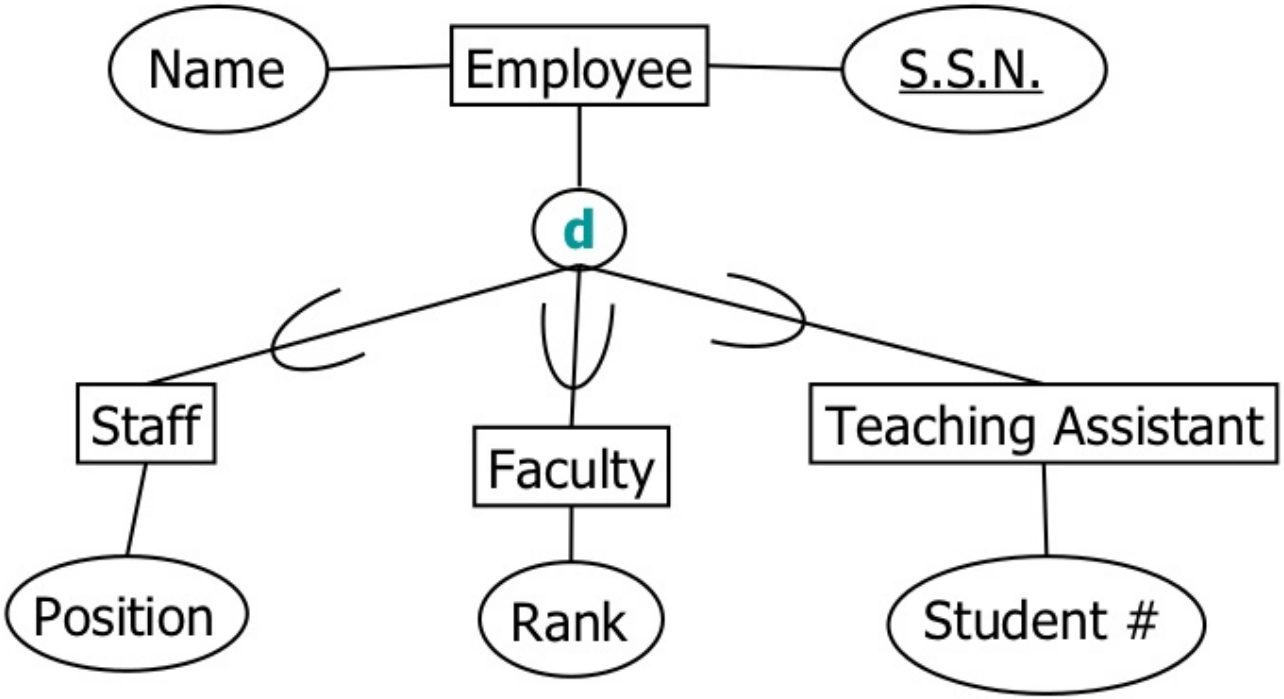
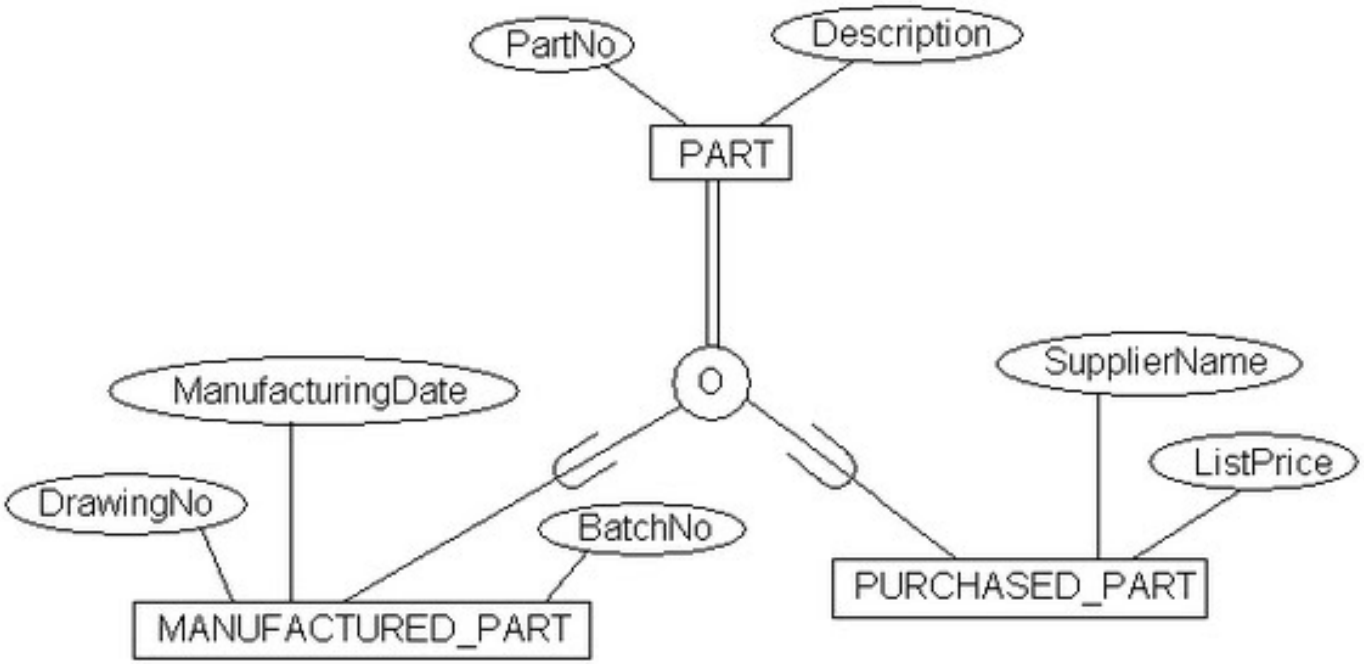


Figure 8.1
EER diagram notation to represent subclasses and specialization.

Constraints (6)



Constraints (7)



UNIVERSITY EERD

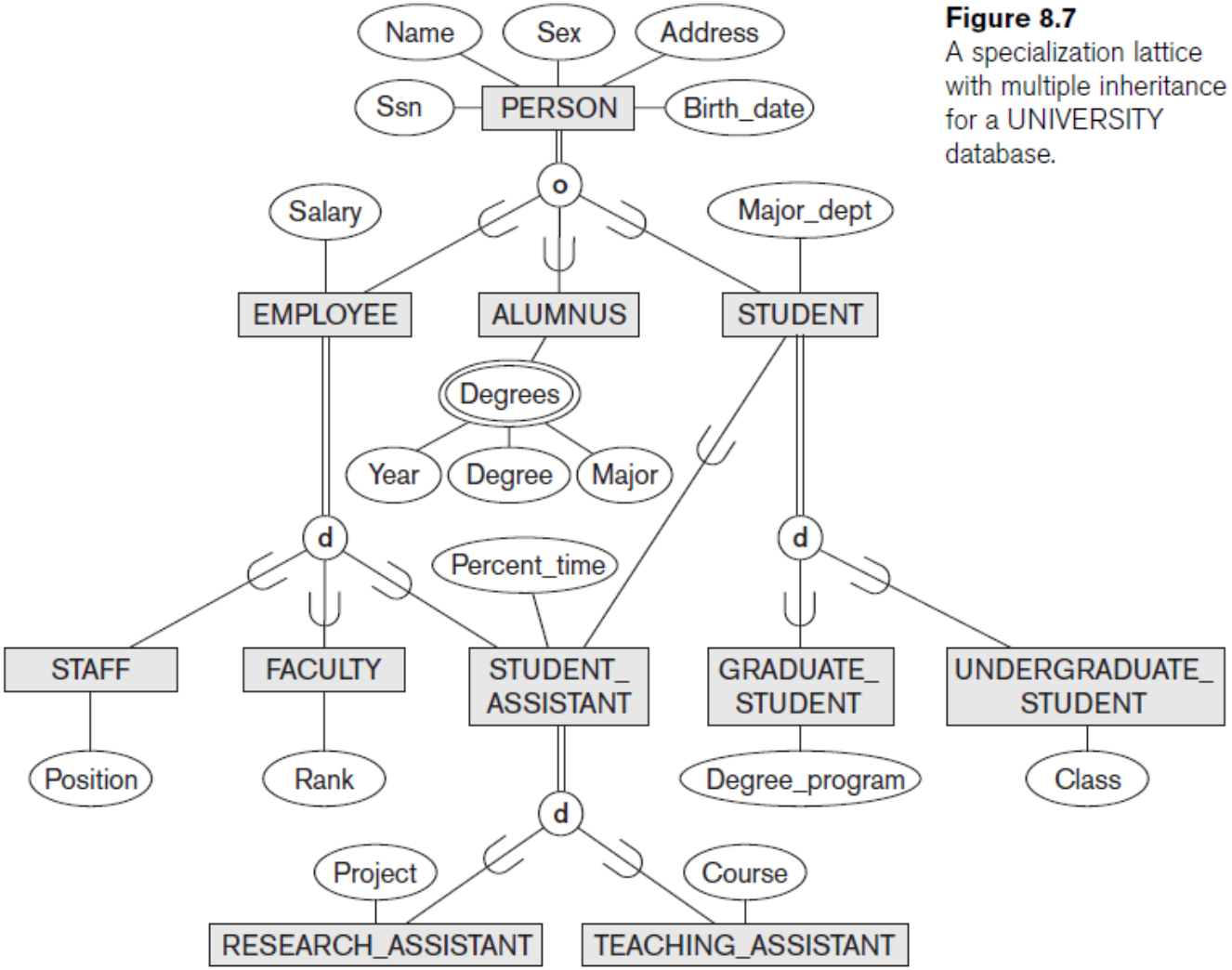


Figure 8.7
A specialization lattice with multiple inheritance for a UNIVERSITY database.

Multiple vs. Single Inheritance

- **Multiple inheritance**
  - Subclass with more than one superclass
  - If attribute (or relationship) originating in the same superclass inherited more than once via different paths in lattice
  - Included only once in shared subclass
- **Single inheritance**
  - Some models and languages limited to single inheritance

Relational Database Design by ER-to-Relational Mapping

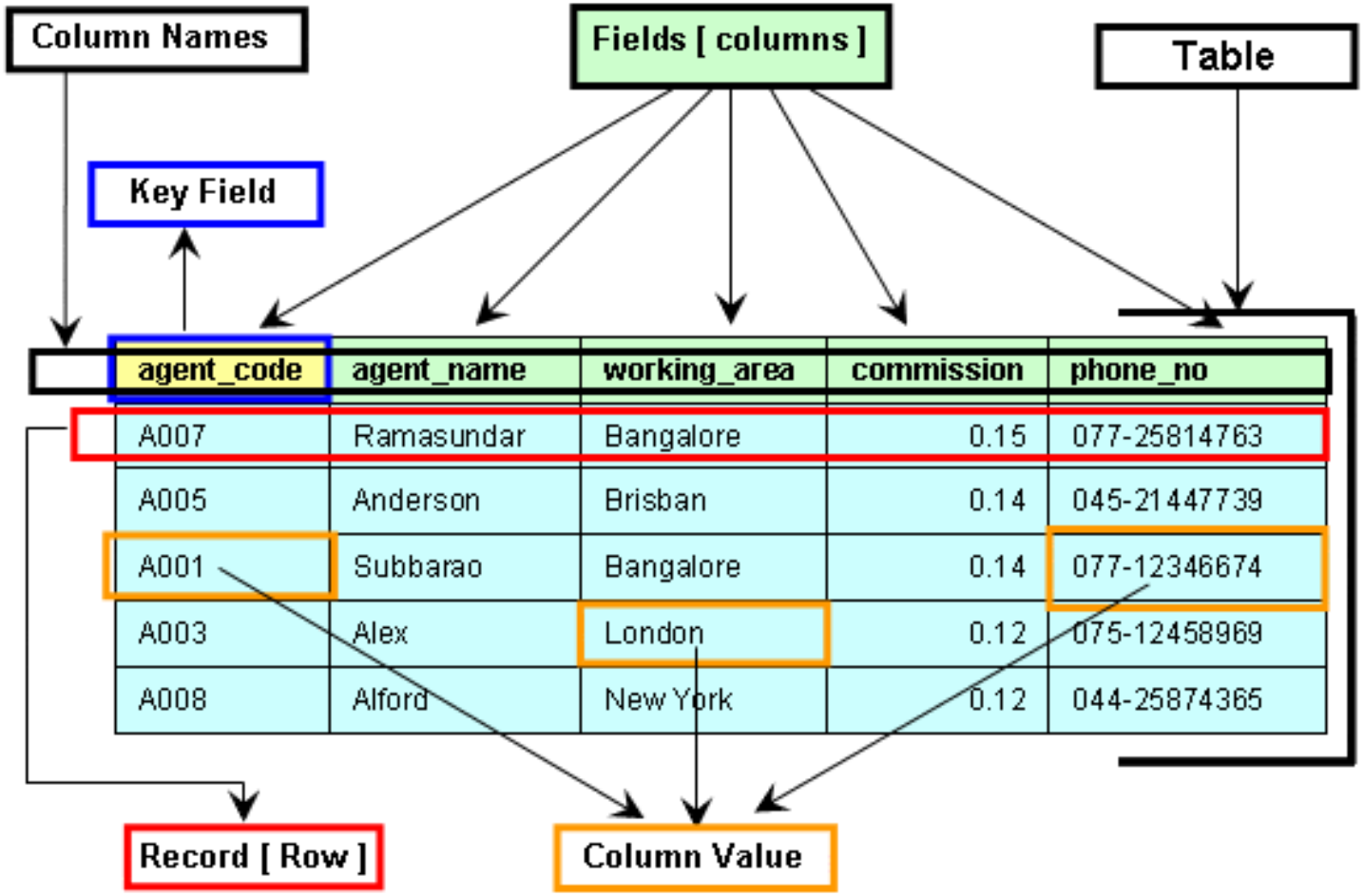
Outline

1. Relational Database Design Based on ERD
2. Relational Database Design Based on EERD

Before Begin – Creating ERD

- Một công ty mua và bán một trong các loại đồ cũ như đồ gỗ, quần áo, đồ sứ, đá quý. Mỗi mặt hàng có mã mặt hàng, đặc tả về mặt hàng, giá đặt hàng, tình trạng.
- Trong số khách hàng của công ty có khách hàng chỉ mua hàng, có khách hàng chỉ bán, có khách hàng vừa mua vừa bán với công ty. Thông tin về khách hàng cần lưu là mã, tên và địa chỉ.
- Khi công ty bán hàng cho khách hàng, thông tin cần lưu là tiền hoa hồng, giá bán thật sự, thuế bán hàng và ngày bán. Thuế bằng không là trường hợp được miễn thuế. Khi mua hàng của khách hàng, công ty lưu thông tin về giá mua, ngày mua, tình trạng hàng khi mua.

What is a database table?



What is a foreign key? (1)

Customer

FirstName	LastName	CustID
Elaine	Stevens	101
Mary	Dittman	102
Skip	Stevenson	103
Drew	Lakeman	104
Eva	Plummer	105

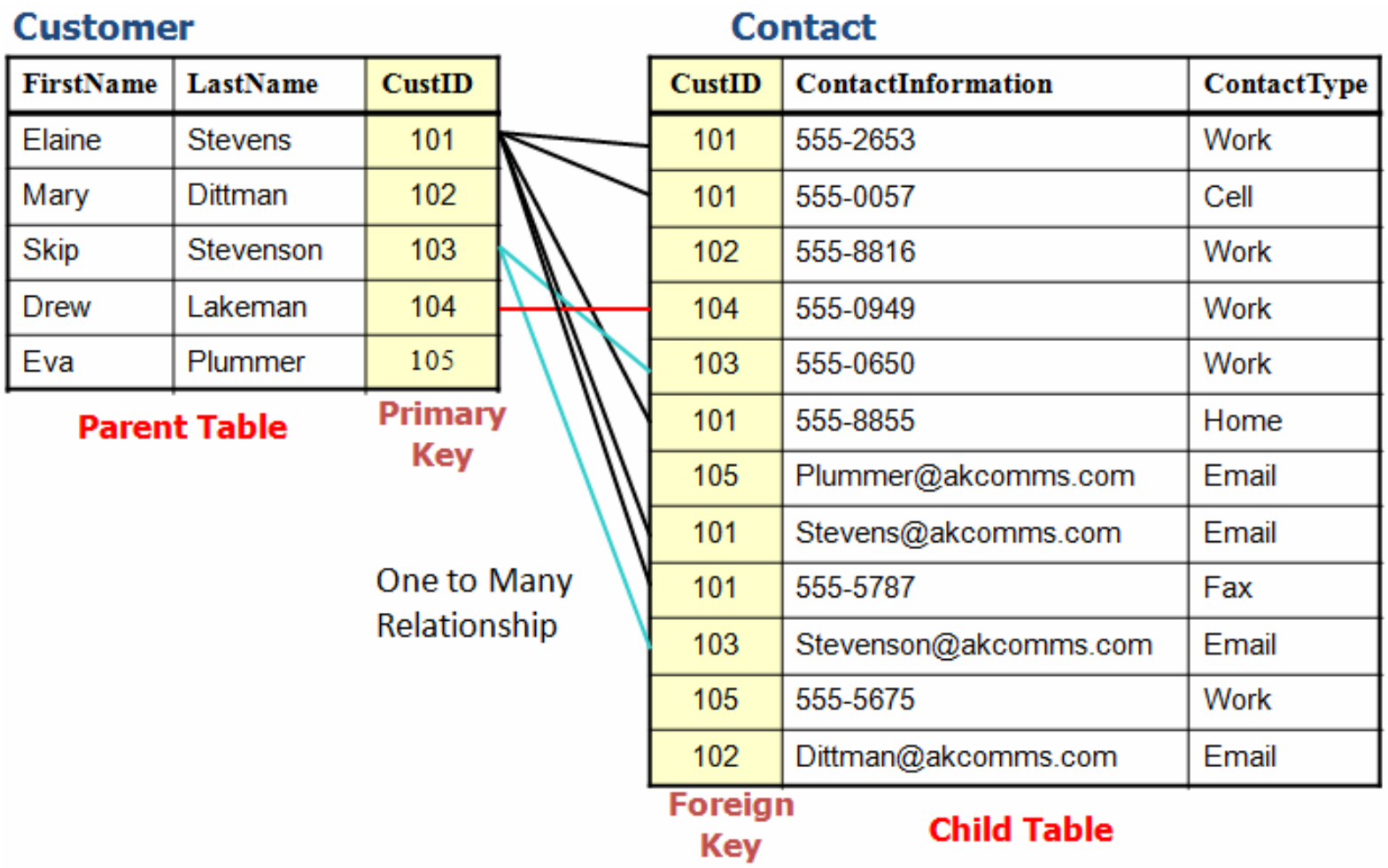
Contact

CustID	ContactInformation	ContactType
101	555-2653	Work
101	555-0057	Cell
102	555-8816	Work
104	555-0949	Work
103	555-0650	Work
101	555-8855	Home
105	Plummer@akcomms.com	Email
101	Stevens@akcomms.com	Email
101	555-5787	Fax
103	Stevenson@akcomms.com	Email
105	555-5675	Work
102	Dittman@akcomms.com	Email

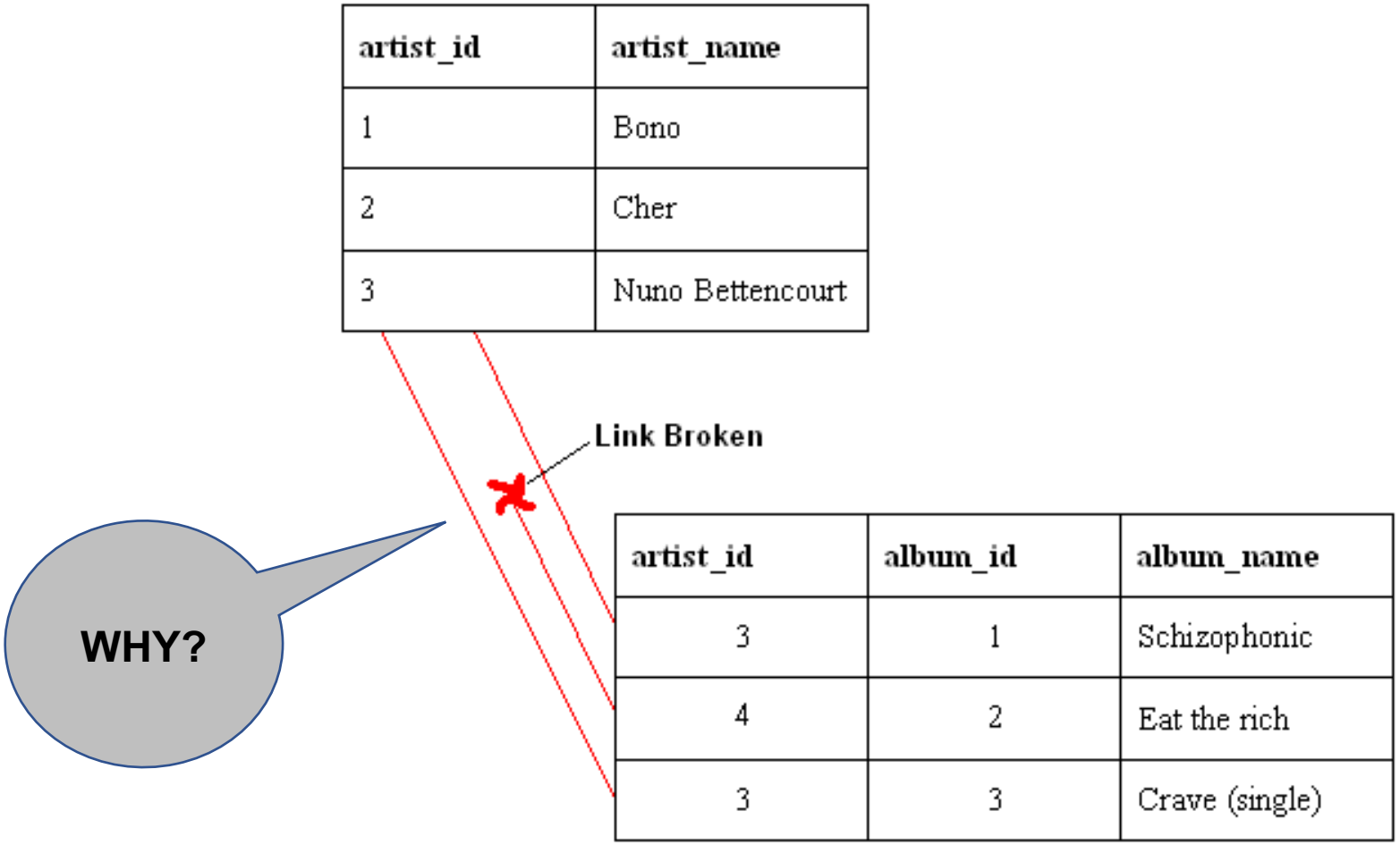
What is a foreign key? (2)

- A foreign key is a field (or a set of fields) in a table that uniquely identifies a row of another table
- The table in which the foreign key is defined to refer to the primary key in the parent table
- Foreign key constraint is how data integrity is enforced between two tables (update, delete, insert)
- The link between the primary key and the foreign key will always be maintained, so your database will never be left with orphaned records

What is a foreign key? (3)



What is a foreign key? (4)

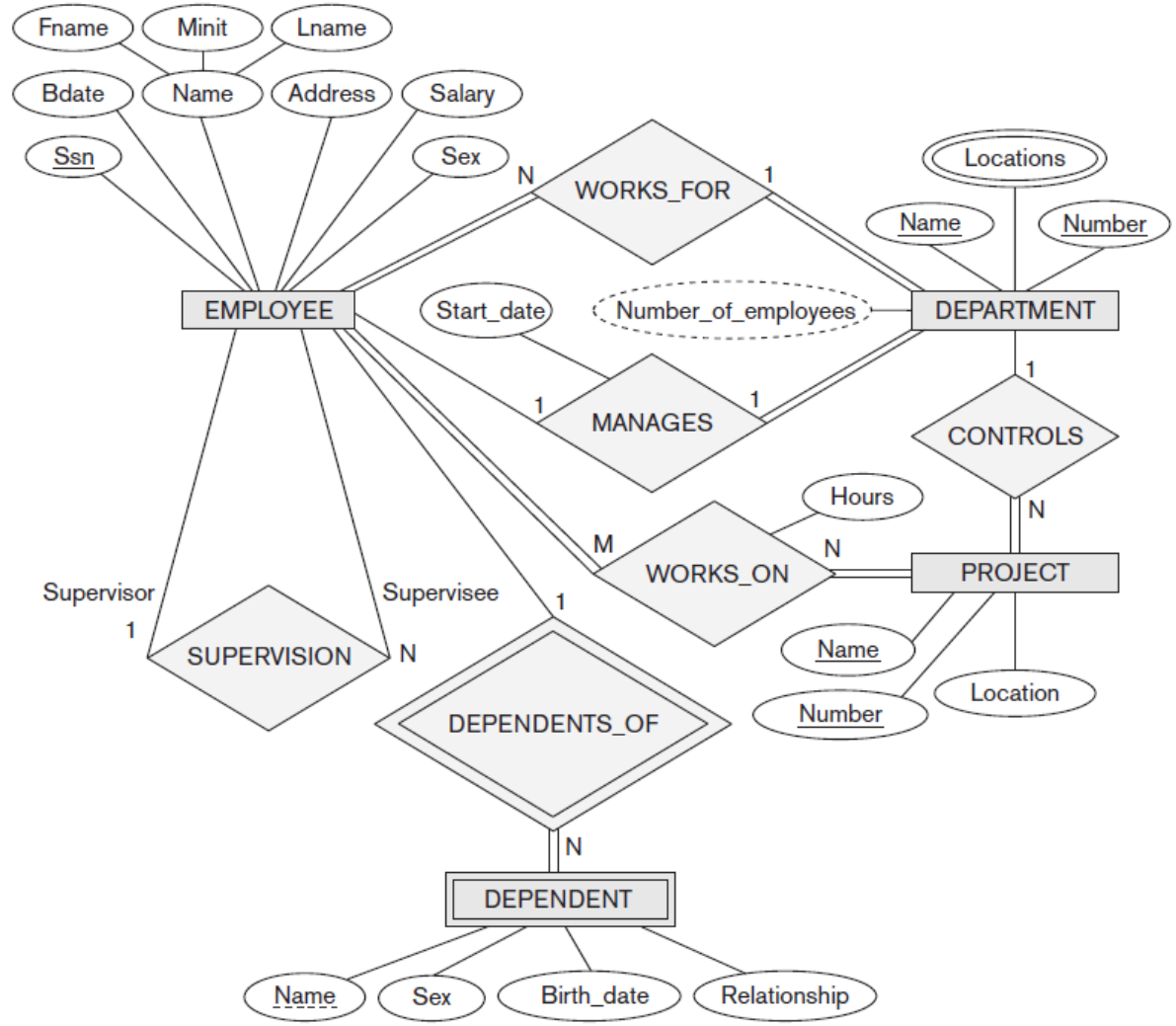


Relational DB Design (1)

- Design a relational database schema
  - Based on a conceptual schema design
- **7-step** algorithm to convert the basic ER model constructs into relations
- Additional steps for EER model

Relational DB Design (2)

Figure 9.1
The ER conceptual schema diagram for the COMPANY database.



Relational DB Design (3)

- COMPANY database example
  - Assume that the mapping will create tables with simple single-valued attributes
- **Step 1: *Mapping of Regular Entity Types***
  - For each regular entity type, create a relation (table) *R* that includes all the simple attributes of *E*

Relational DB Design (4)

- **Step 2:** ***Mapping of Weak Entity Types***
  - For each weak entity type, create a relation (table) *R* and include all simple attributes of the entity type
  - Include primary key attribute of owner as foreign key attributes of *R*

Relational DB Design (5)

Figure 9.3
Illustration of some mapping steps.
a. *Entity* relations after step 1.
b. Additional *weak entity* relation after step 2.
c. *Relationship* relation after step 5.
d. Relation representing multivalued attribute after step 6.

(a)

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary

DEPARTMENT

Dname	Dnumber

PROJECT

Pname	Pnumber	Plocation

(b)

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

Relational DB Design (6)

- **Step 3:** ***Mapping of Binary* 1:1 *Relationship Types***
  - For each binary 1:1 relationship type
    - Identify relations that correspond to entity types participating in *R*
  - Possible approaches:
    - **Foreign key approach** *(recommended)*
    - **Merged relationship approach**
    - **Cross-reference or relationship relation approach**

Relational DB Design (7)

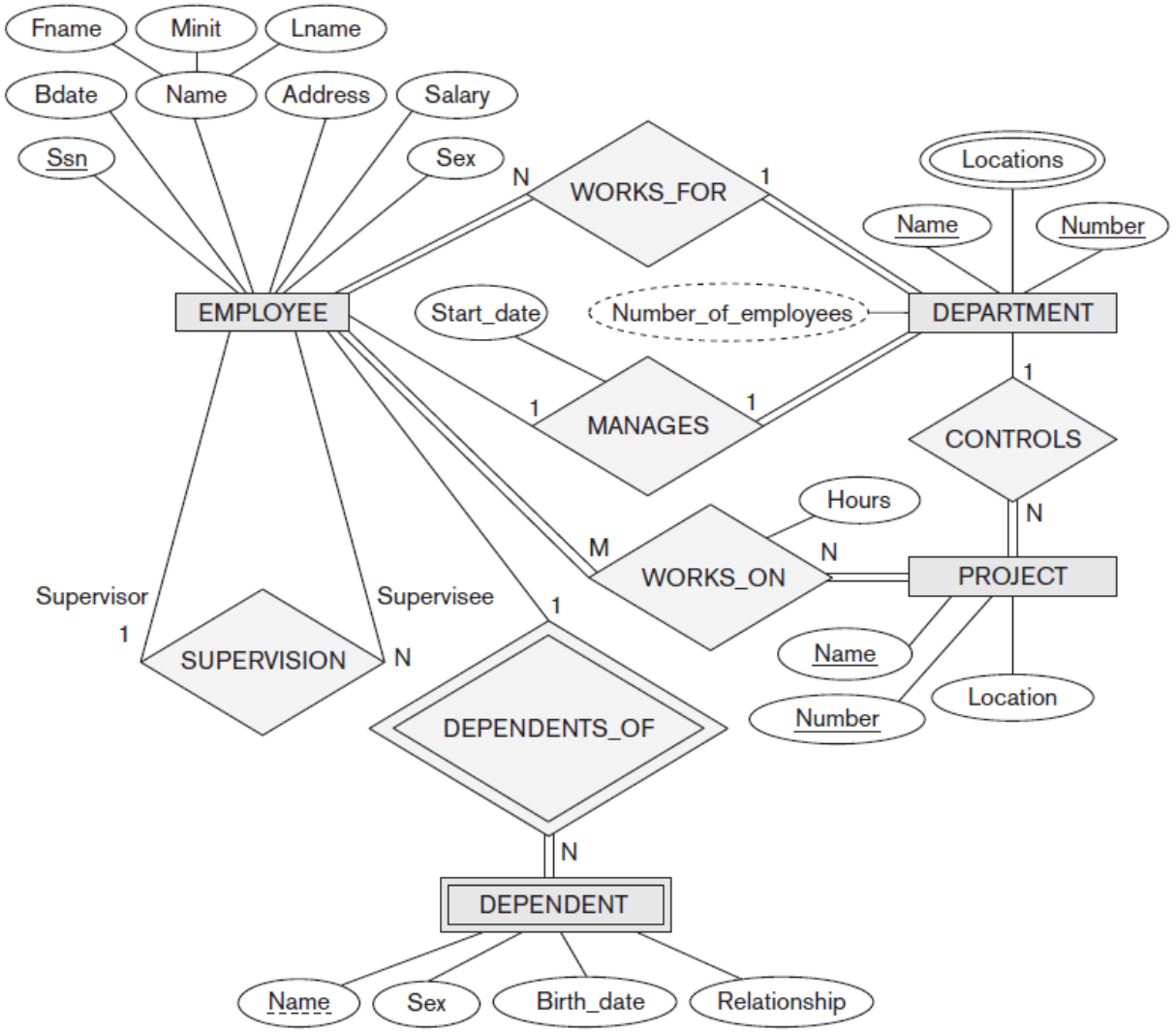
- **Step 4:** ***Mapping of Binary* 1:N *Relationship Types***
  - For each regular binary 1:*N* relationship type
    - Identify relation that represents participating entity type at *N*-side of relationship type
    - Include primary key of 1-side to *N*-side

Relational DB Design (8)

- **Step 5:** ***Mapping of Binary*** **M:N** ***Relationship Types***
  - For each binary *M*:*N* relationship type
    - Create a new relation (table) *S*
    - Include primary key of participating entity types as foreign key attributes in *S*
    - Include any simple attributes of *M*:*N* relationship type

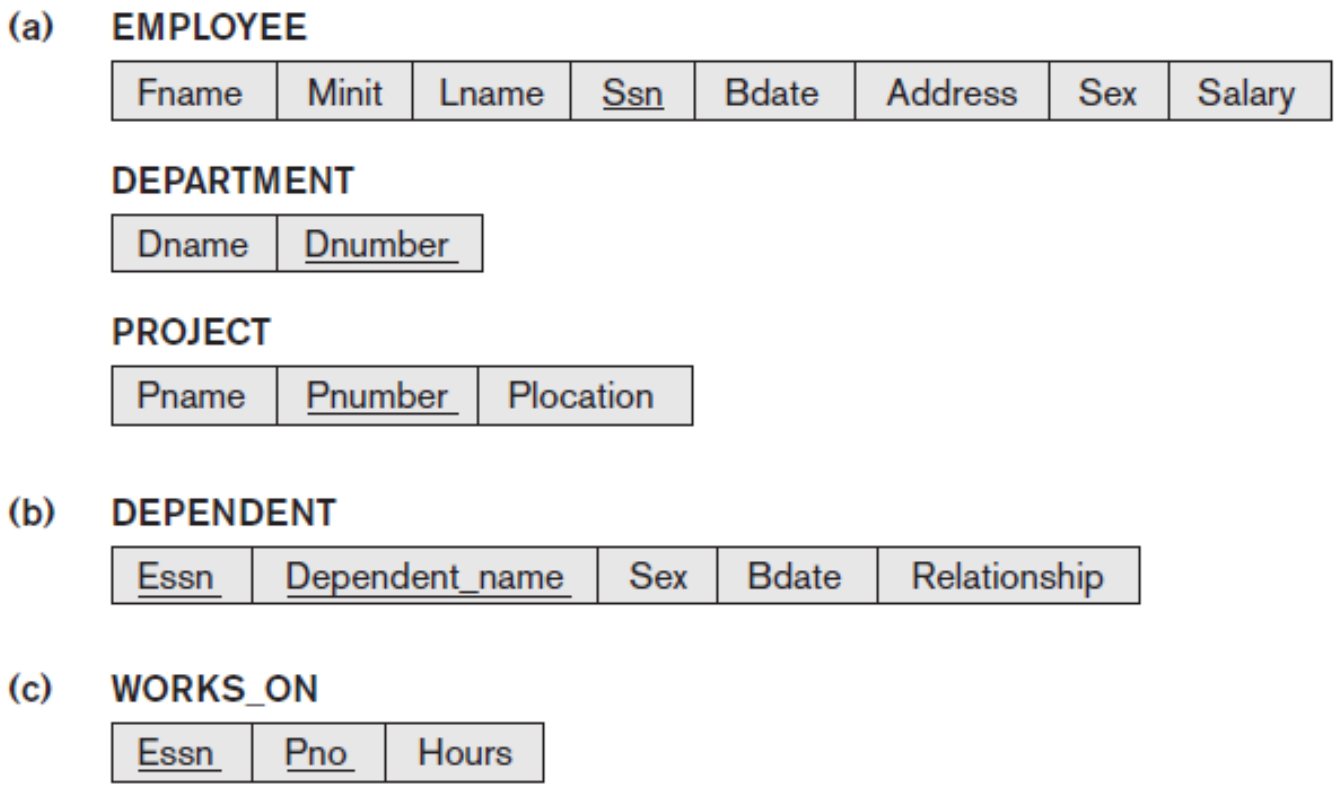
Relational DB Design (9)

Figure 9.1
The ER conceptual schema diagram for the COMPANY database.



Relational DB Design (10)

Figure 9.3
Illustration of some mapping steps.
a. *Entity* relations after step 1.
b. Additional *weak entity* relation after step 2.
c. *Relationship* relation after step 5.
d. Relation representing multivalued attribute after step 6.



Relational DB Design (11)

- A not good design! (Why?)

Employee

SSN	***Name***	***Dept_ID***
1	AB	D1
2	CD	null
3	EF	D2
4	GH	null
5	IJ	null

Department

Dept_ID	***Name***
D1	HR
D2	R&D

Relational DB Design (12)

- A good design! (Can you spot the difference?)

Employee

SSN	***Name***
1	AB
2	CD
3	EF
4	GH
5	IJ

Department

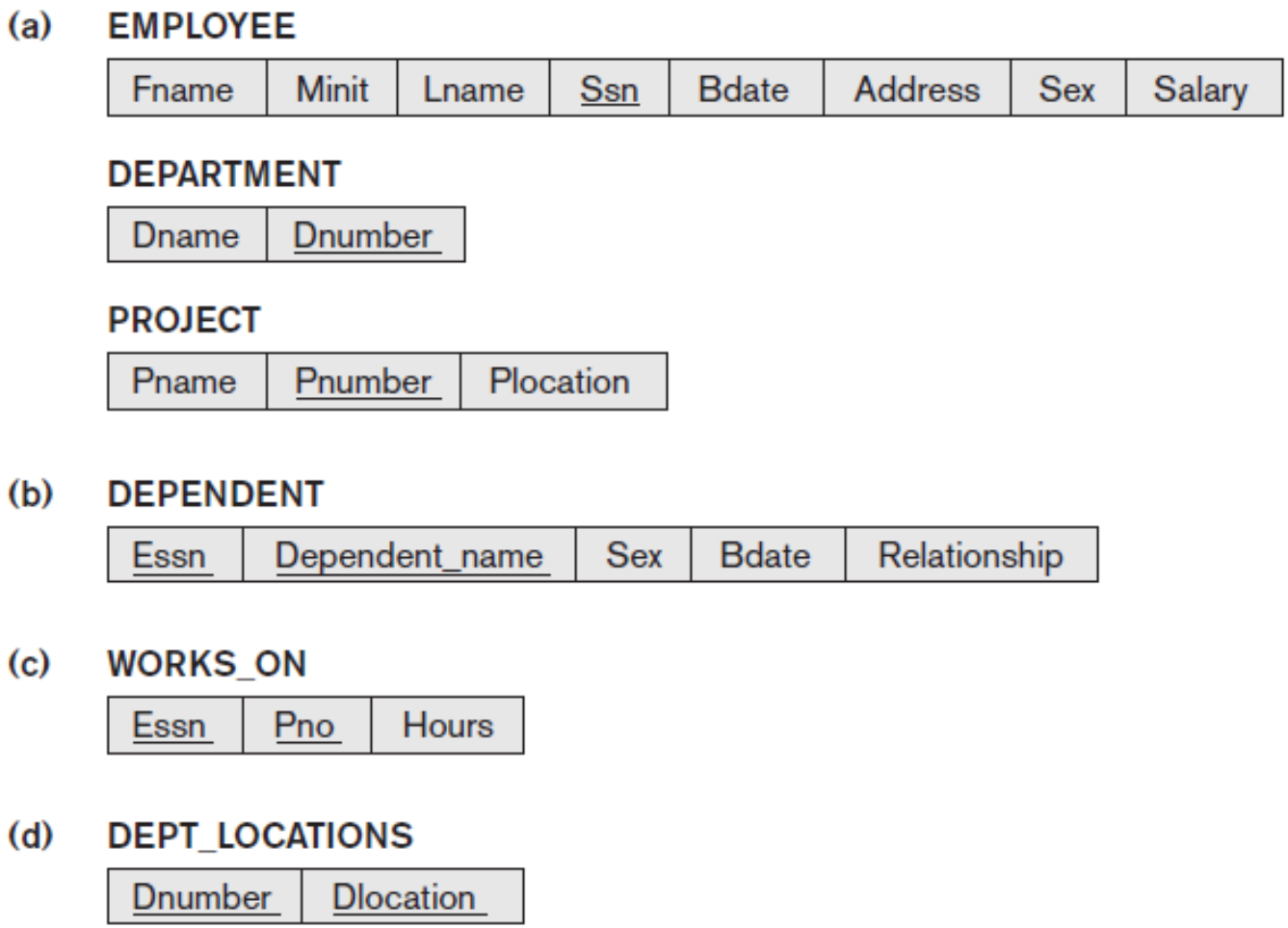
Dept_ID	***Name***	***SSN***
D1	HR	1
D2	R&D	3

Relational DB Design (13)

- **Step 6:** ***Mapping of Multivalued Attributes***
  - For each multivalued attribute
    - Create a new relation (table) *R*
    - Primary key of *R* is the combination of *A* and *K*
    - If the multivalued attribute is composite, include its simple components

Relational DB Design (14)

Figure 9.3
Illustration of some mapping steps.
a. *Entity* relations after step 1.
b. Additional *weak entity* relation after step 2.
c. *Relationship* relation after step 5.
d. Relation representing multivalued attribute after step 6.



Relational DB Design (15)

- **Step 7:** ***Mapping of N-ary Relationship Types***
  - For each *n*-ary relationship type *R*
    - Create a new relation *S* to represent *R*
    - Include primary keys of participating entity types as foreign keys
    - Include any simple attributes as attributes

Relational DB Design (11)

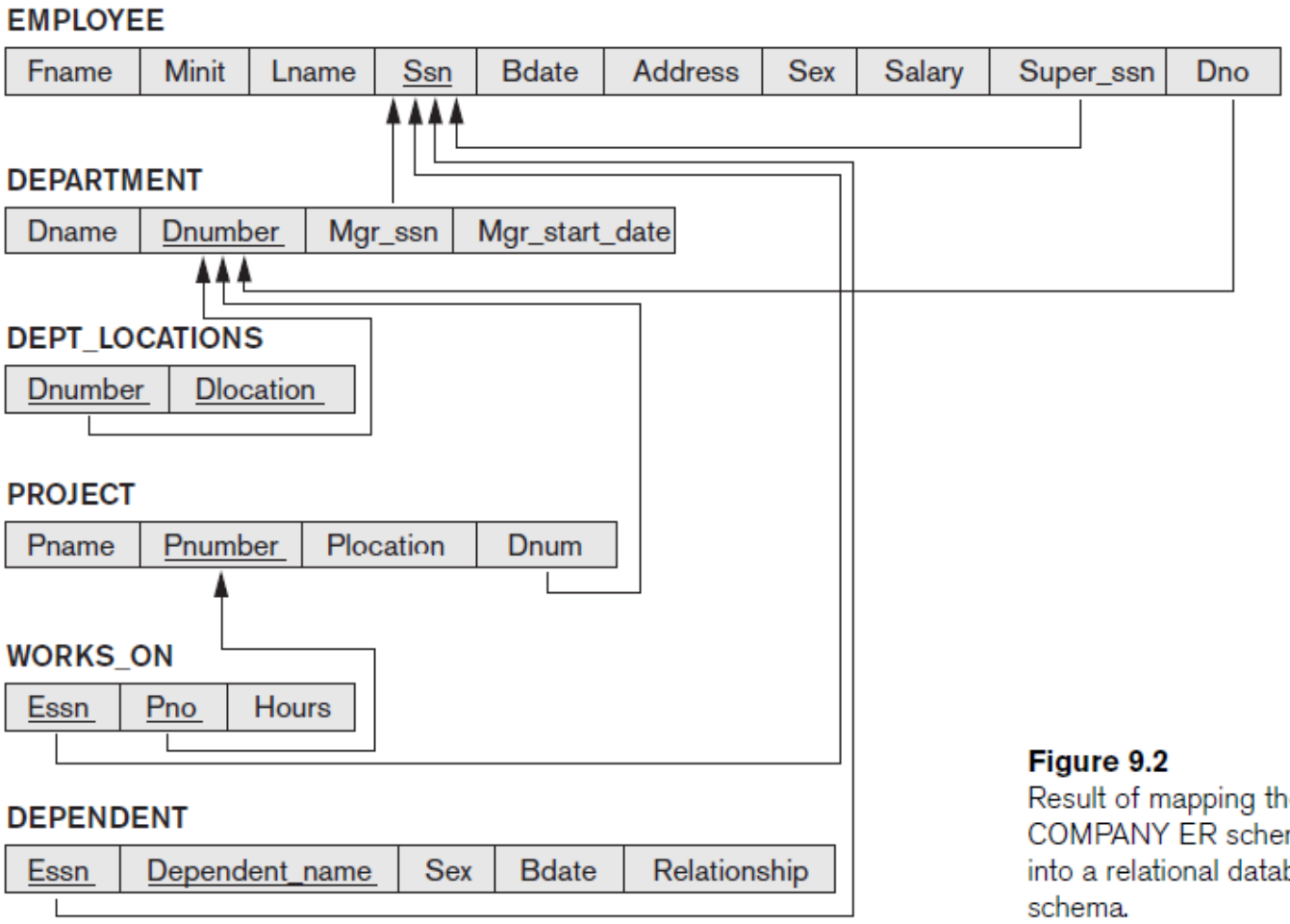


Figure 9.2
Result of mapping the COMPANY ER schema into a relational database schema.

Summary

Table 9.1 Correspondence between ER and Relational Models

ER MODEL	RELATIONAL MODEL
Entity type	*Entity* relation
1:1 or 1:N relationship type	Foreign key (or *relationship* relation)
M:N relationship type	*Relationship* relation and *two* foreign keys
n-ary relationship type	*Relationship* relation and *n* foreign keys
Simple attribute	Attribute
Composite attribute	Set of simple component attributes
Multivalued attribute	Relation and foreign key
Value set	Domain
Key attribute	Primary (or secondary) key

Outline

1. Relational Database Design Based on ERD
2. Relational Database Design Based on EERD

Relational DB Design Based on EERD (1)

- **Step 8:** ***Options for Mapping Specialization or Generalization***
  - **Option 8A:** Multiple relations - superclass and subclasses
    - For any specialization (total or partial, disjoint or overlapping)
  - **Option 8B:** Multiple relations - subclass relations only
    - Subclasses are total
    - Specialization has disjointedness constraint

Relational DB Design Based on EERD (2)

- **Option 8C:** Single relation with one type attribute
  - Type or discriminating attribute indicates subclass of tuple
  - Subclasses are disjoint
    - Potential for generating many NULL values if many specific attributes exist in the subclasses
- **Option 8D:** Single relation with multiple type attributes
  - Subclasses are overlapping
  - Will also work for a disjoint specialization

Relational DB Design Based on EERD (3)

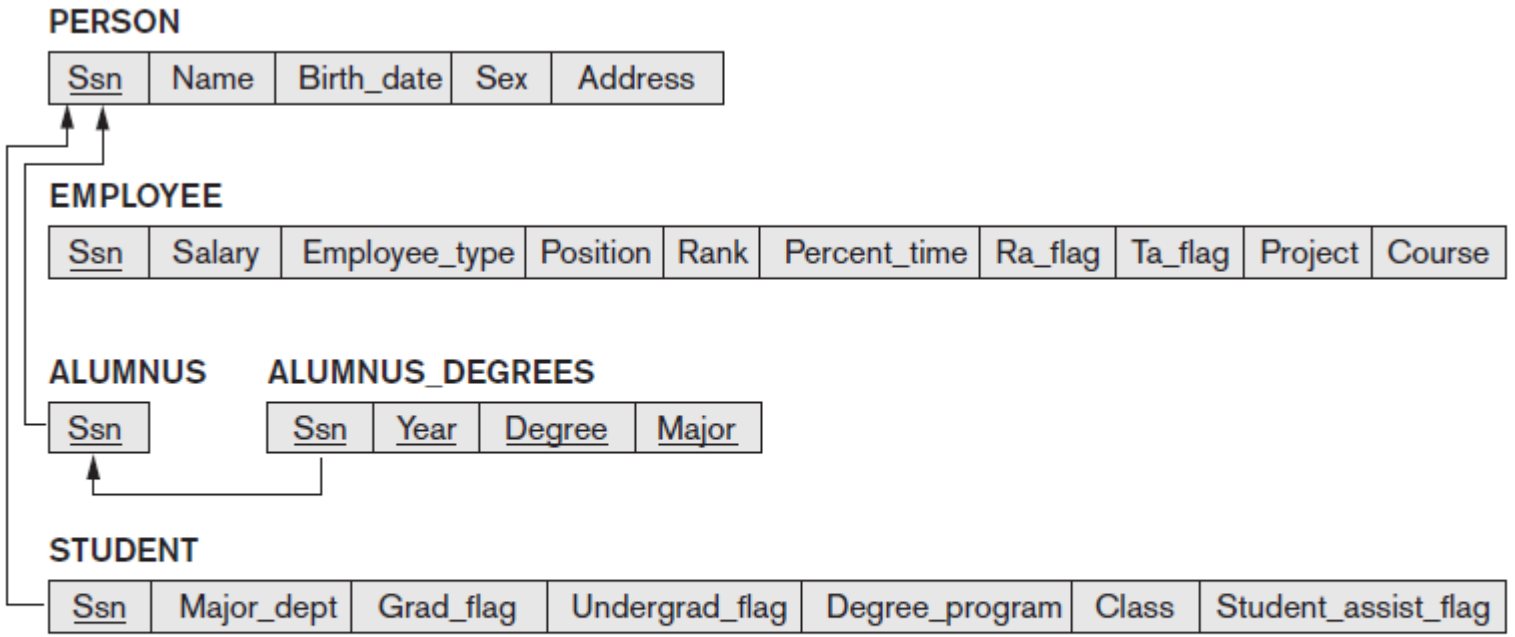


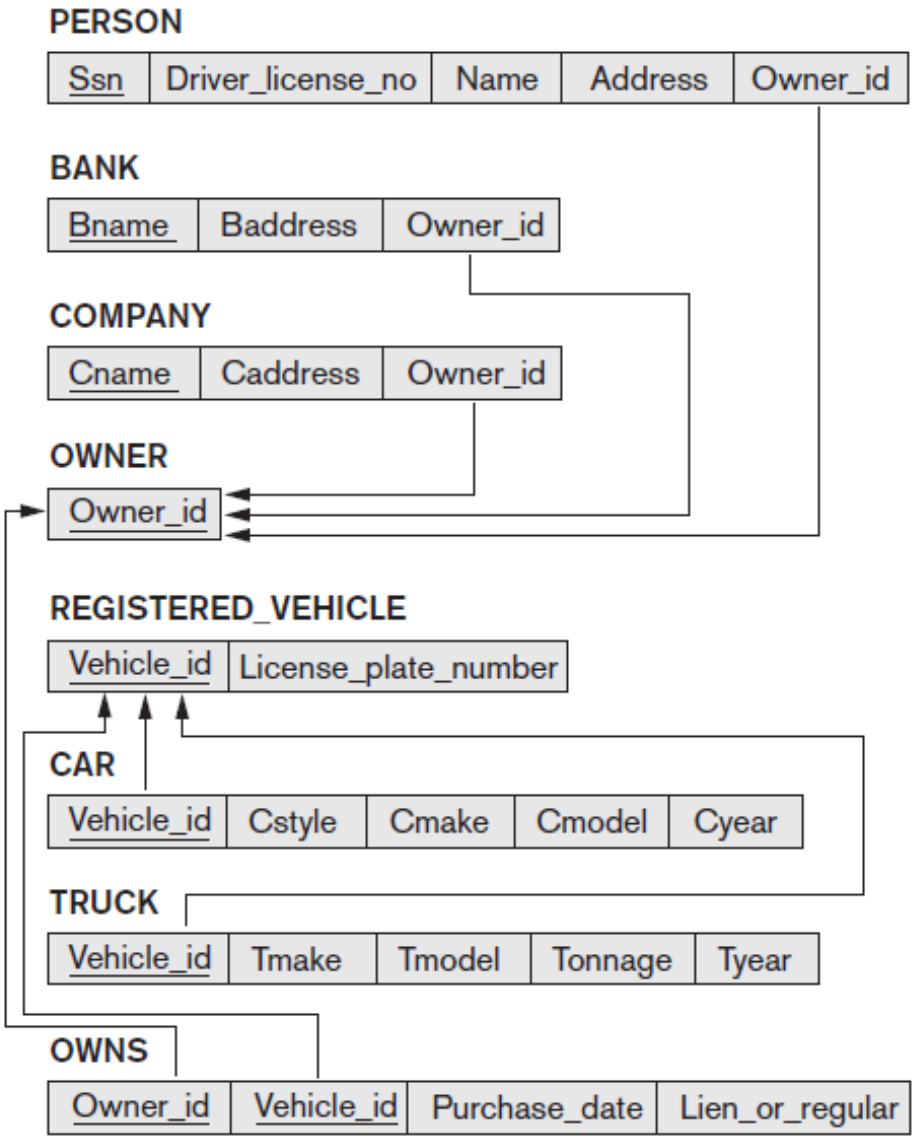
Figure 9.6
Mapping the EER specialization lattice in Figure 8.8 using multiple options.

Relational DB Design Based on EERD (4)

- **Step 9:** ***Mapping of Union Types (Categories)***
  - Defining superclasses have different keys
  - Specify a new key attribute
  - Surrogate key (can be though as candidate key)

Relational DB Design Based on EERD (5)

Figure 9.7
Mapping the EER categories (union types) in Figure 8.8 to relations.



END OF CHAPTER