ER-to-Relational Mapping

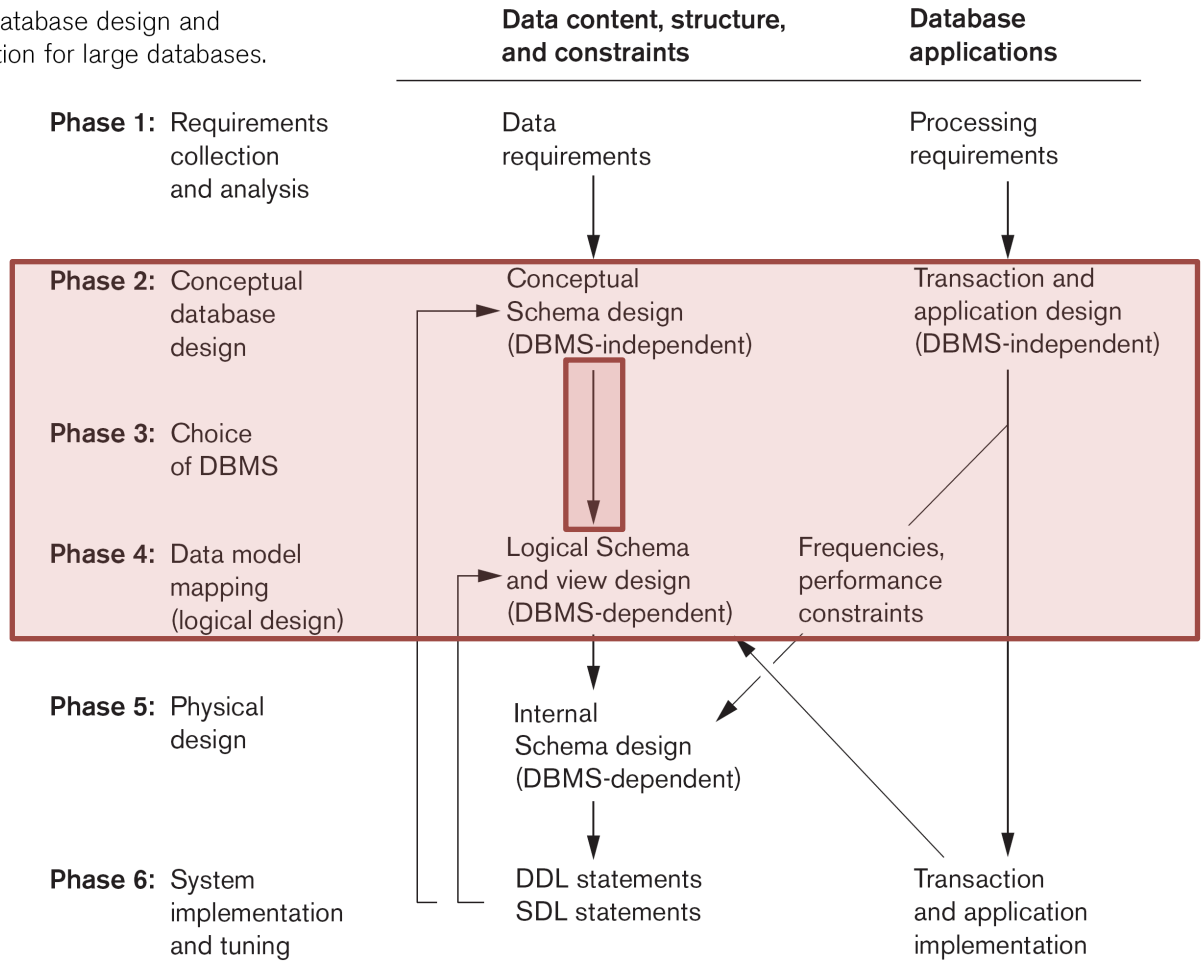
Lecture 4

Outline

1. Context
2. The Algorithm

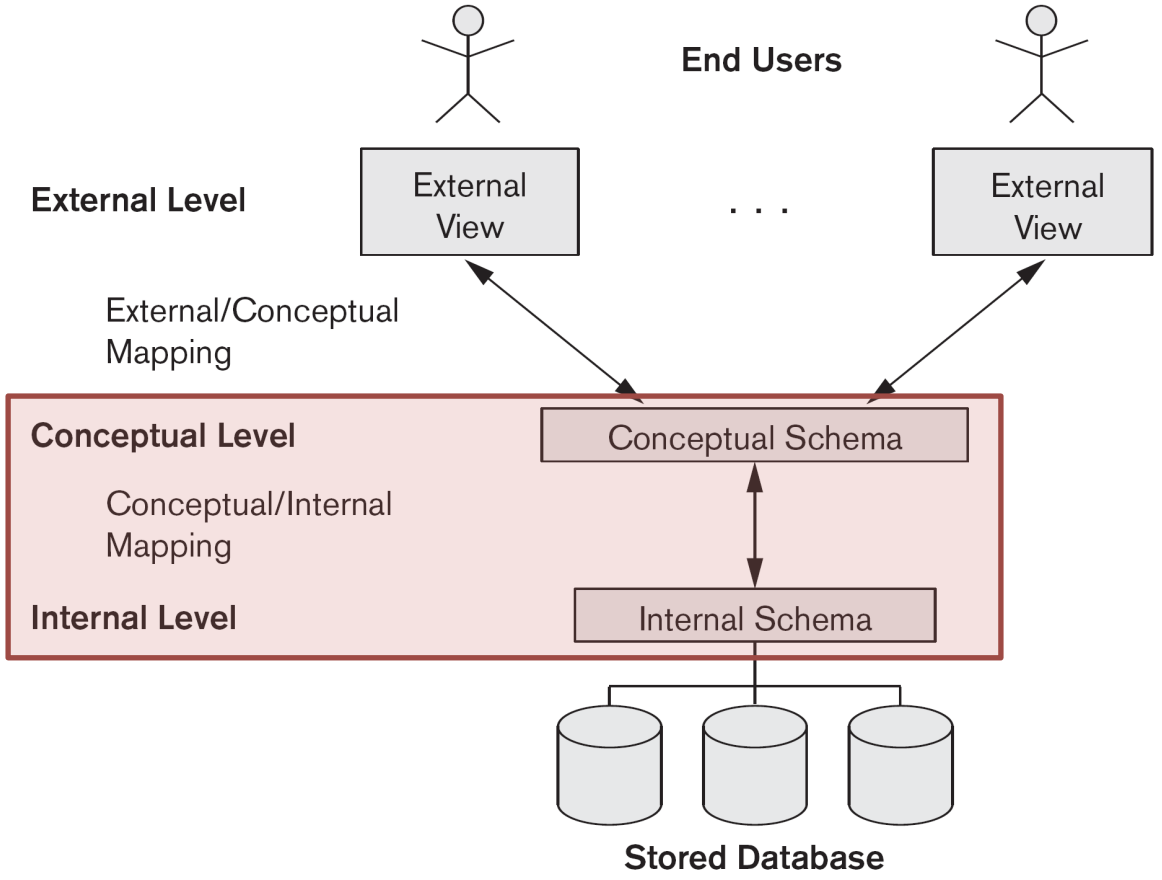
Database Design and Implementation Process

Figure 10.1
Phases of database design and implementation for large databases.



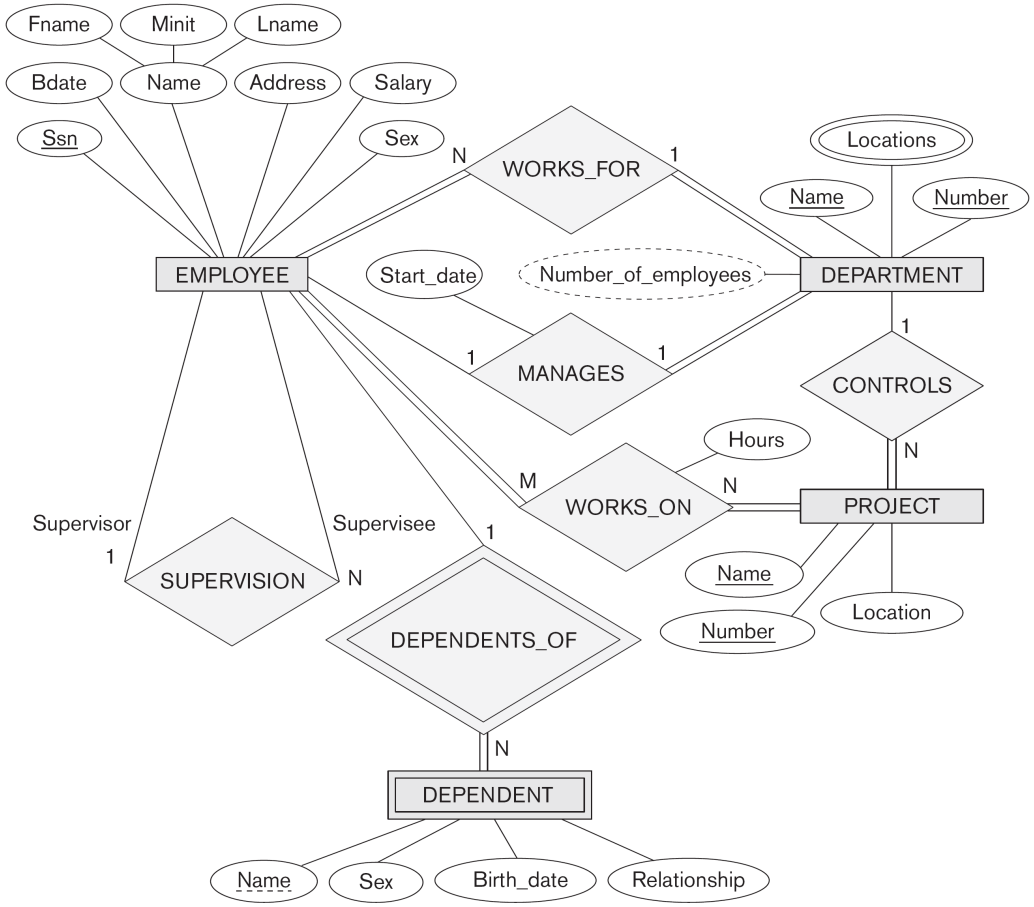
Data Models

Figure 2.2
The three-schema architecture.



Example ERD

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



Resulting Relational Schema

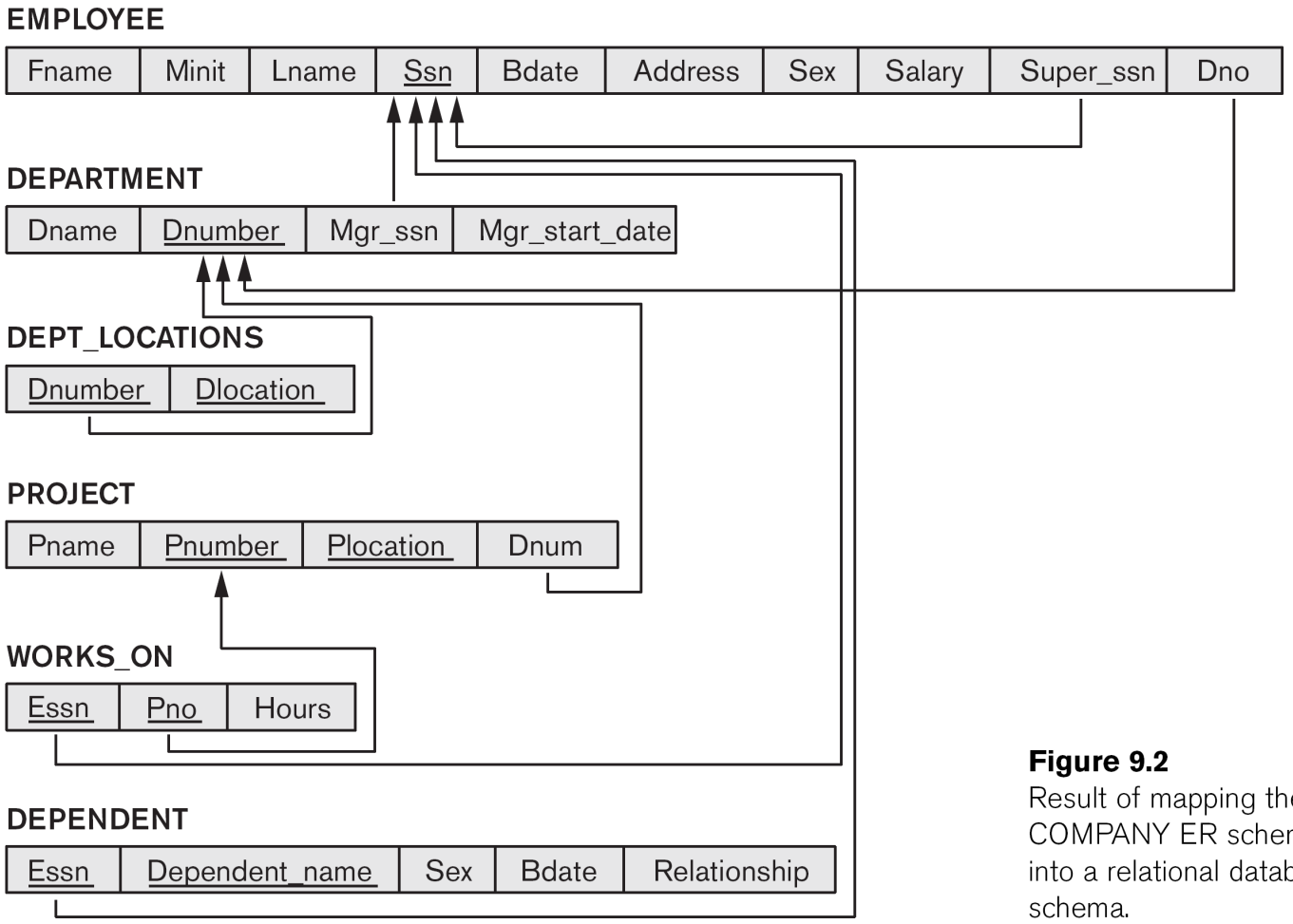


Figure 9.2
Result of mapping the COMPANY ER schema into a relational database schema.

Step 1: Regular Entity Types

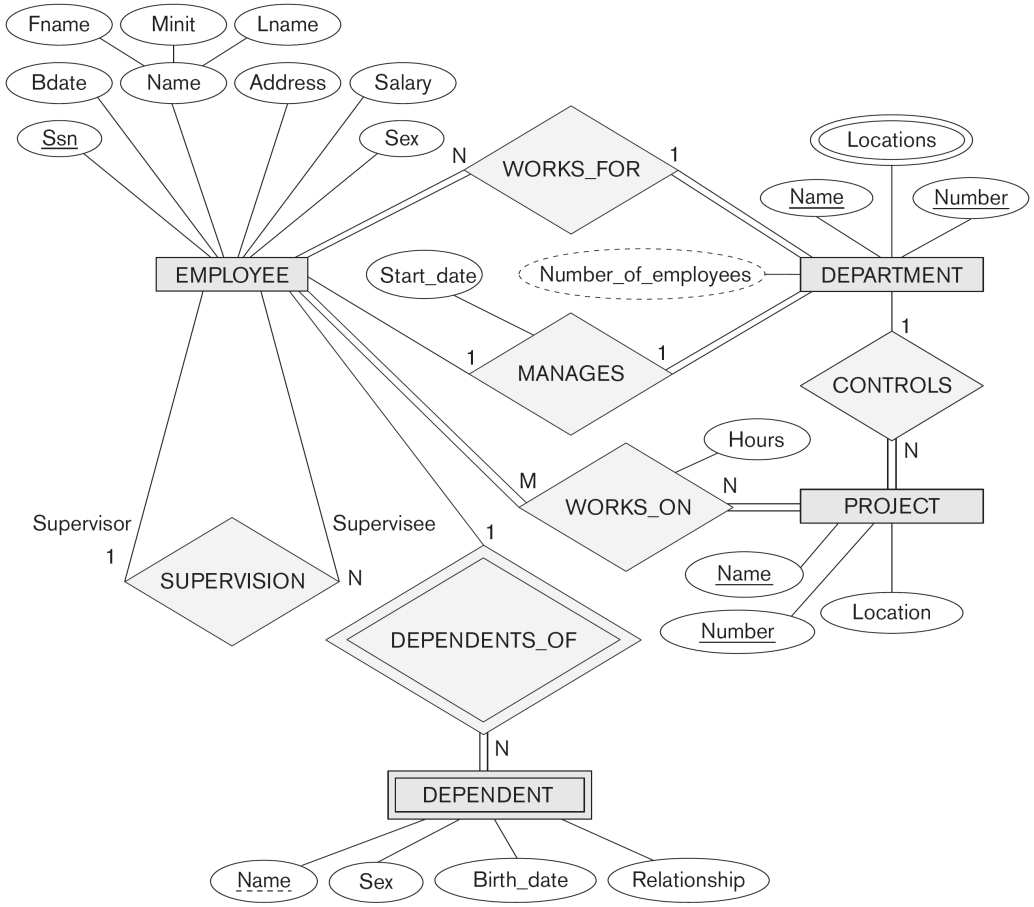
i. For each regular/strong entity type, create a corresponding relation that includes all the simple attributes (includes simple attributes of composite relations)

ii. Choose one of the key attributes as primary
   - If composite, the simple attributes together form the primary key

iii. Any remaining key attributes are kept as secondary unique keys (these will be useful for physical tuning w.r.t. indexing analysis)

Example ERD

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



Step 1 Result

Figure 9.3
Illustration of some mapping steps.
a. *Entity* relations after step 1.

(a)

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary

DEPARTMENT

Dname	Dnumber

PROJECT

Pname	Pnumber	Plocation

Step 2: Weak Entity Types

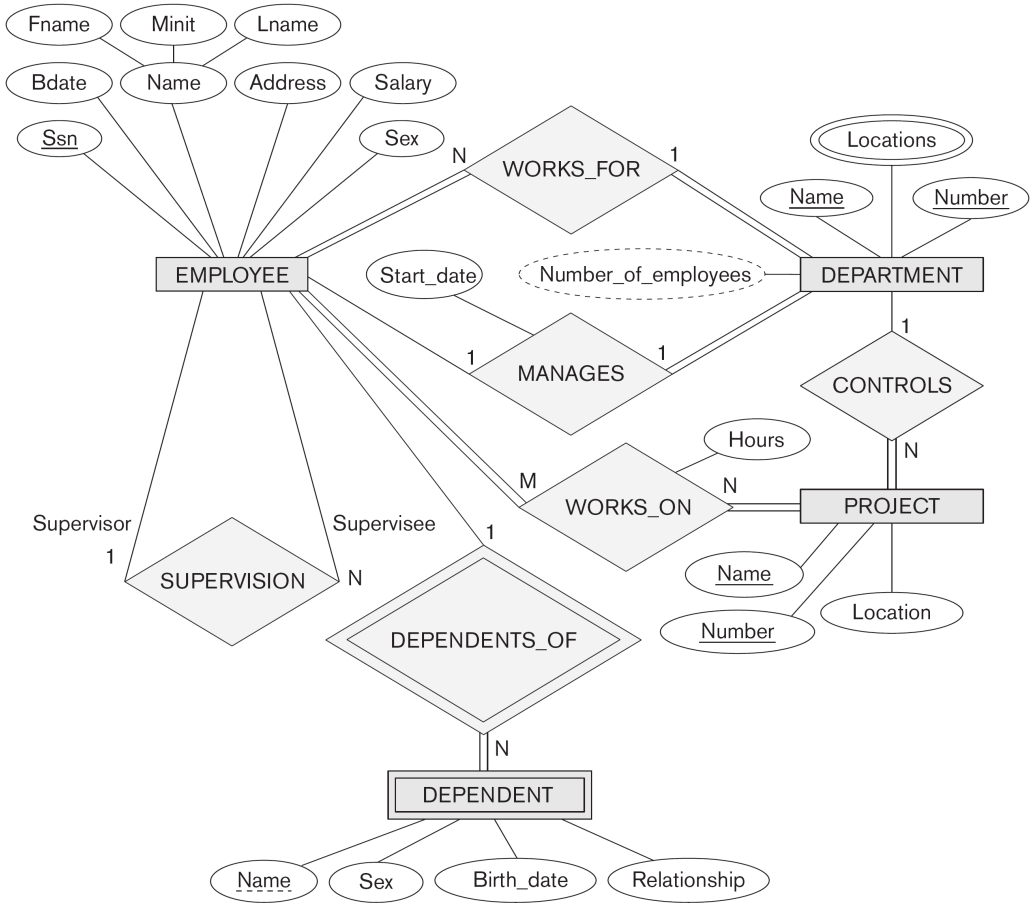
i. For each weak entity type, create a corresponding relation that includes all the simple attributes

ii. Add as a foreign key all of the primary key attribute(s) in the entity corresponding to the owner entity type

iii. The primary key is the combination of all the primary key attributes from the owner and the partial key of the weak entity, if any

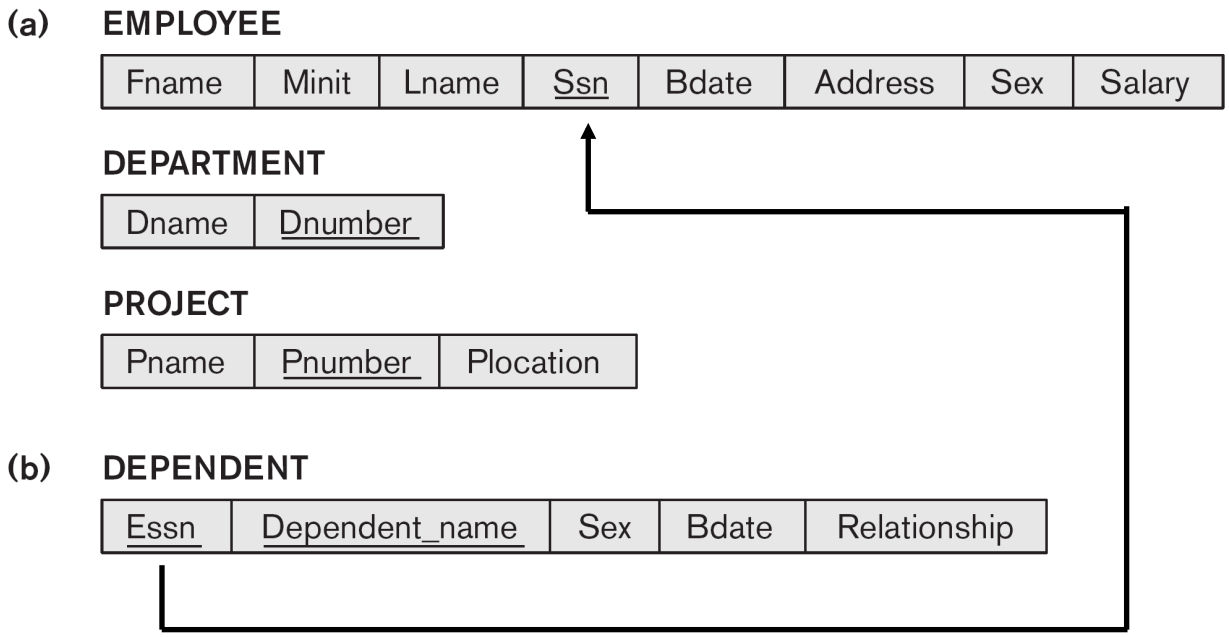
Example ERD

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



Step 2 Result

Figure 9.3
Illustration of some mapping steps.
a. *Entity* relations after step 1.
b. Additional *weak entity* relation after step 2.



Step 3: Mapping Binary 1-to-1

Three approaches

- **Foreign Key**
  - Usually appropriate
- Merged Relation
  - Possible when both participations are total
- Relationship Relation
  - Not discussed

Step 3: Mapping Binary 1-to-1

Foreign Key

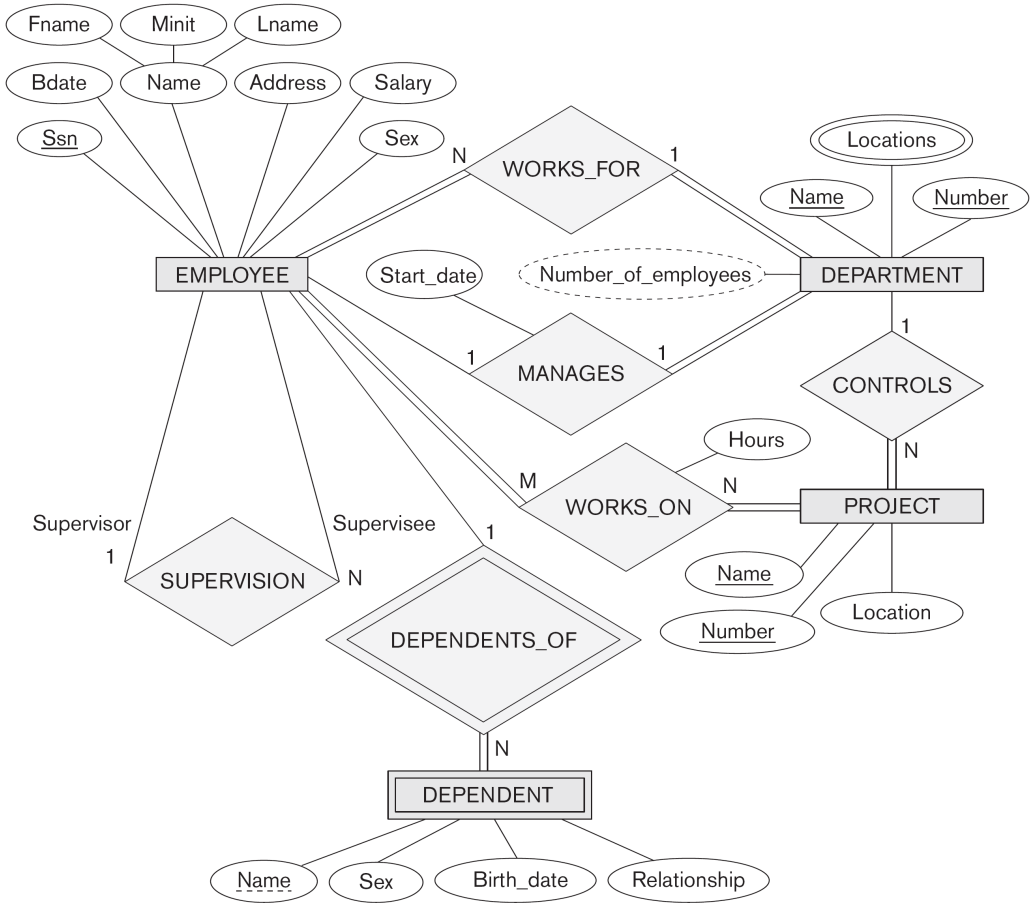
i. Choose one relation as *S*, the other *T*
   - Better if S has total participation (reduces number of NULL values)

ii. Add to *S* all the simple attributes of the relationship

iii. Add as a foreign key in *S* the primary key attributes of *T*

Example ERD

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



Step 2 Result

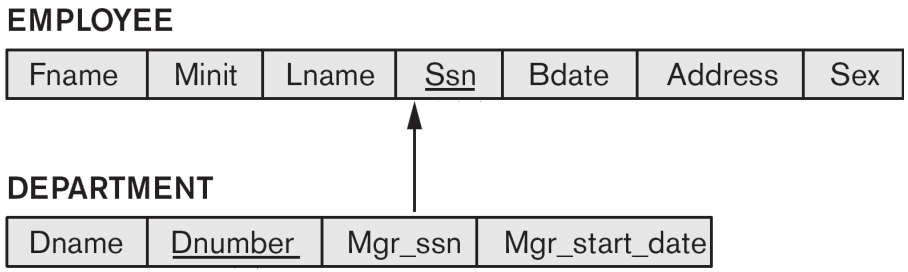
EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary

DEPARTMENT

Dname	Dnumber

Step 3 Result



Step 4: Binary 1-to-N

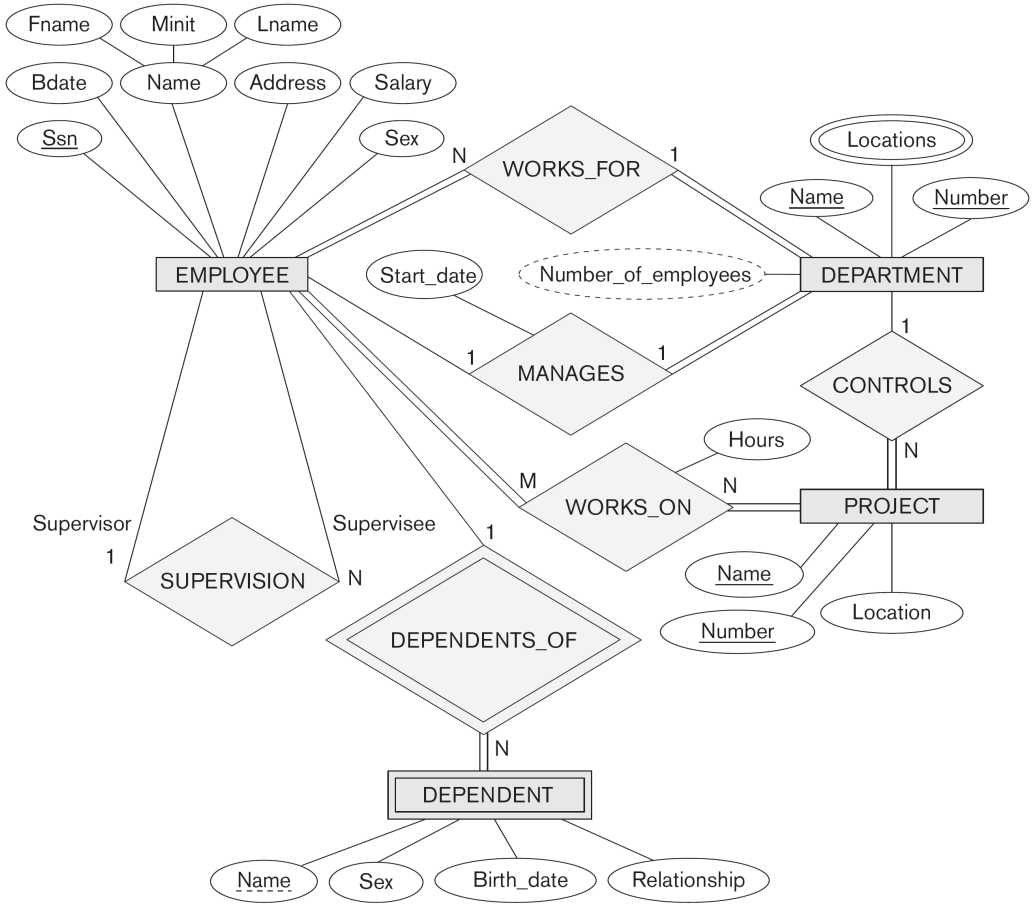
i. Choose the *S* relation as the type at the N-side of the relationship, other is *T*

ii. Add as a foreign key to *S* all of the primary key attribute(s) of *T*

iii. Could also create a relationship relation

Example ERD

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



Step 4 Result

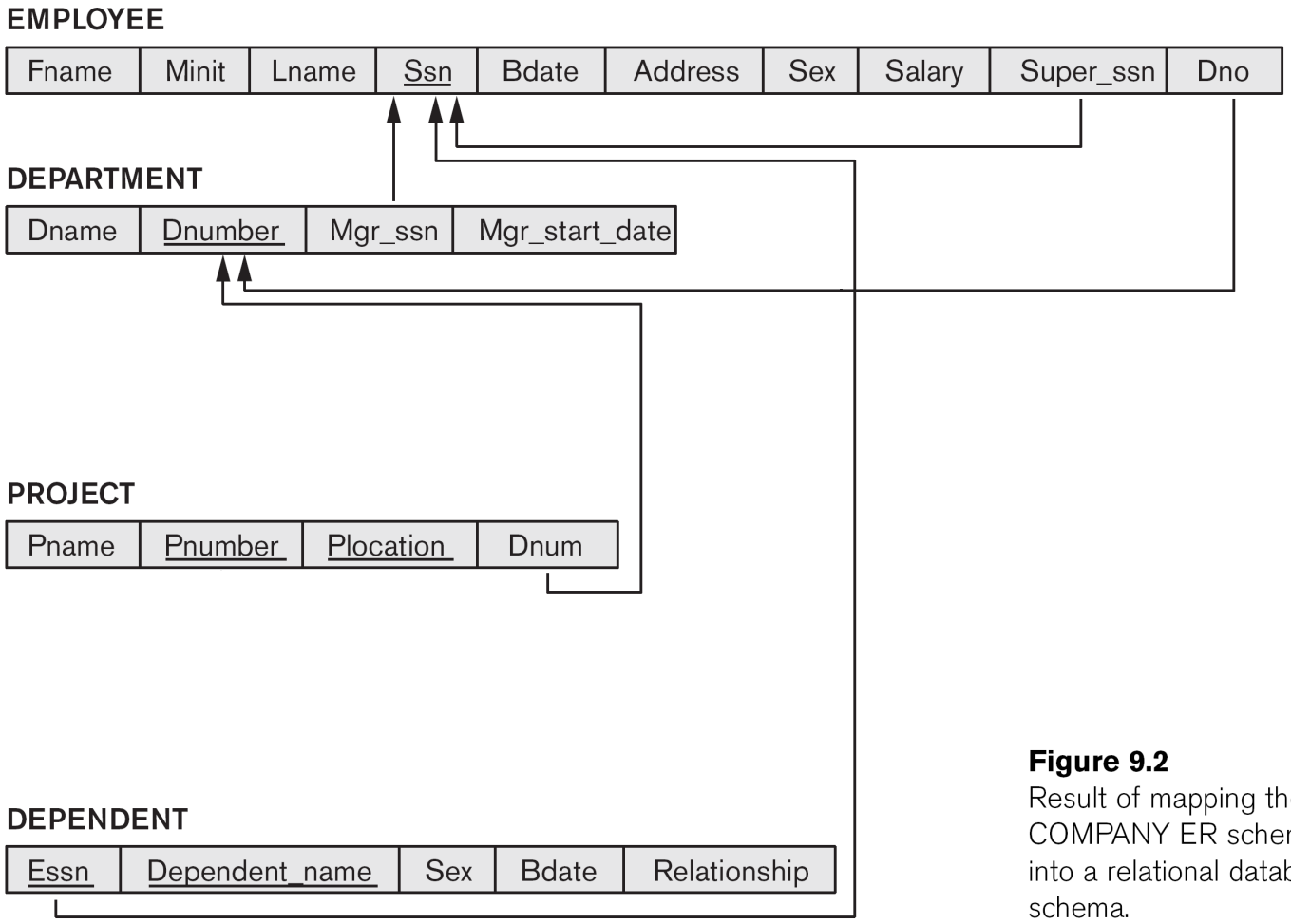


Figure 9.2
Result of mapping the COMPANY ER schema into a relational database schema.

Step 5: Binary M-to-N

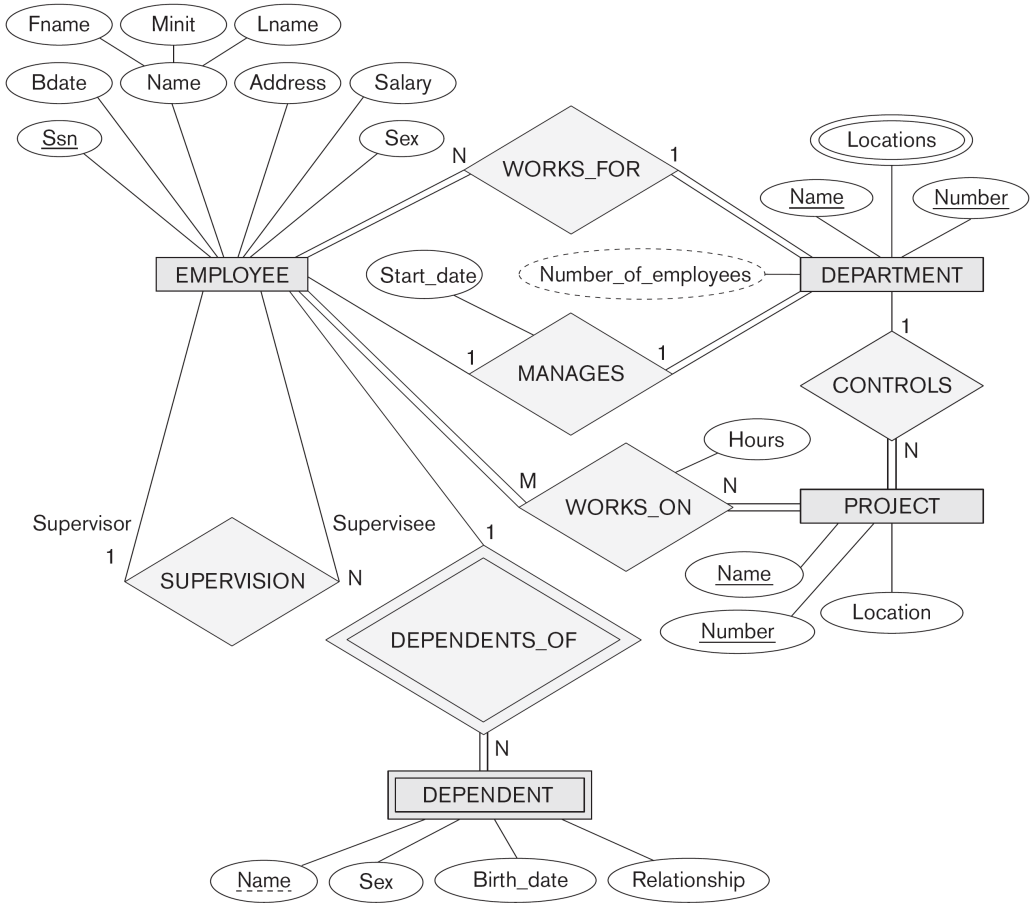
i. Create a new relation *S*

ii. Add as foreign keys the primary keys of both relations; their combination forms the primary key of *S*

iii. Add any simple attributes of the M:N relationship to *S*

Example ERD

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



Step 5 Result

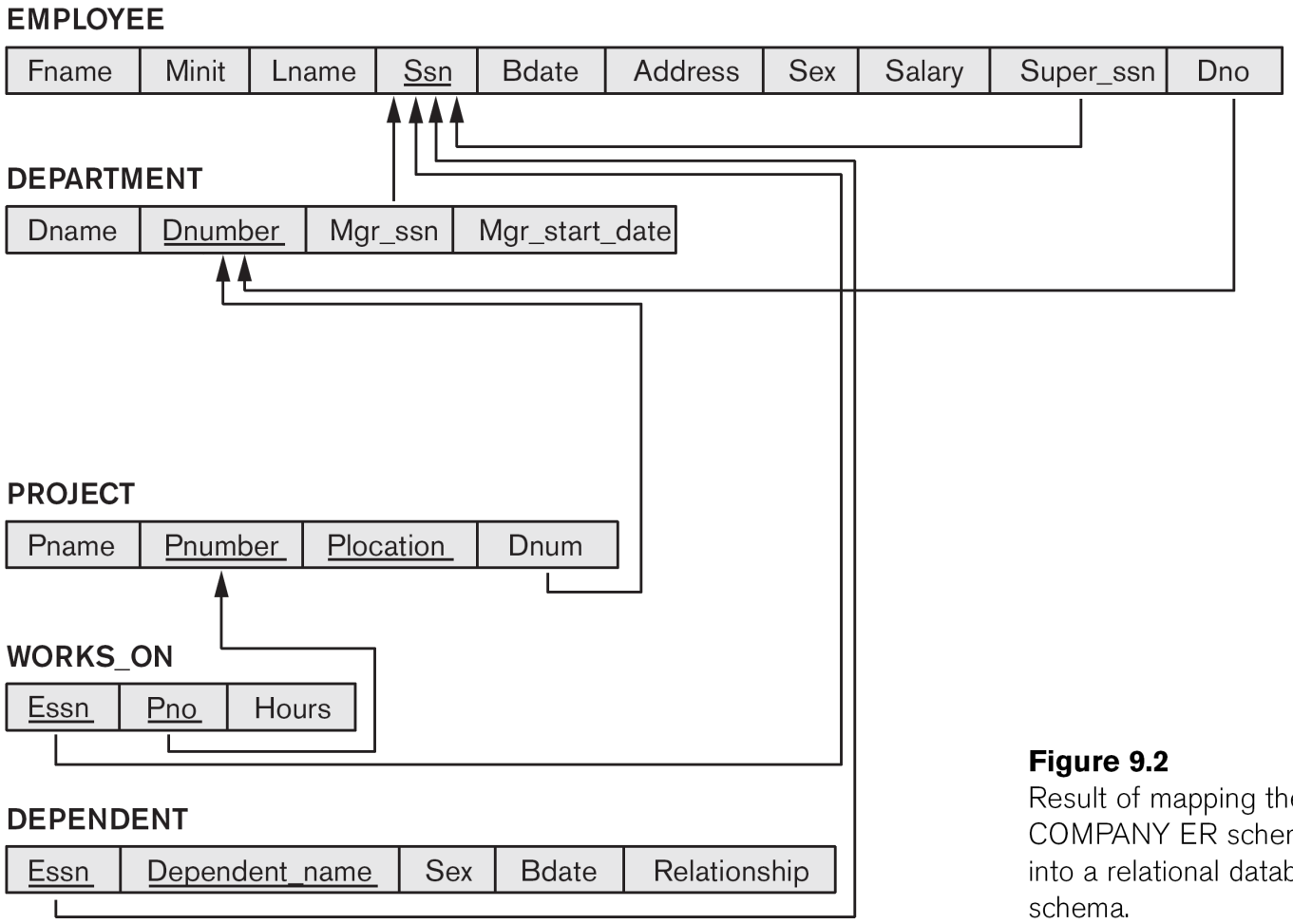


Figure 9.2
Result of mapping the COMPANY ER schema into a relational database schema.

Step 6: Multivalued Attributes

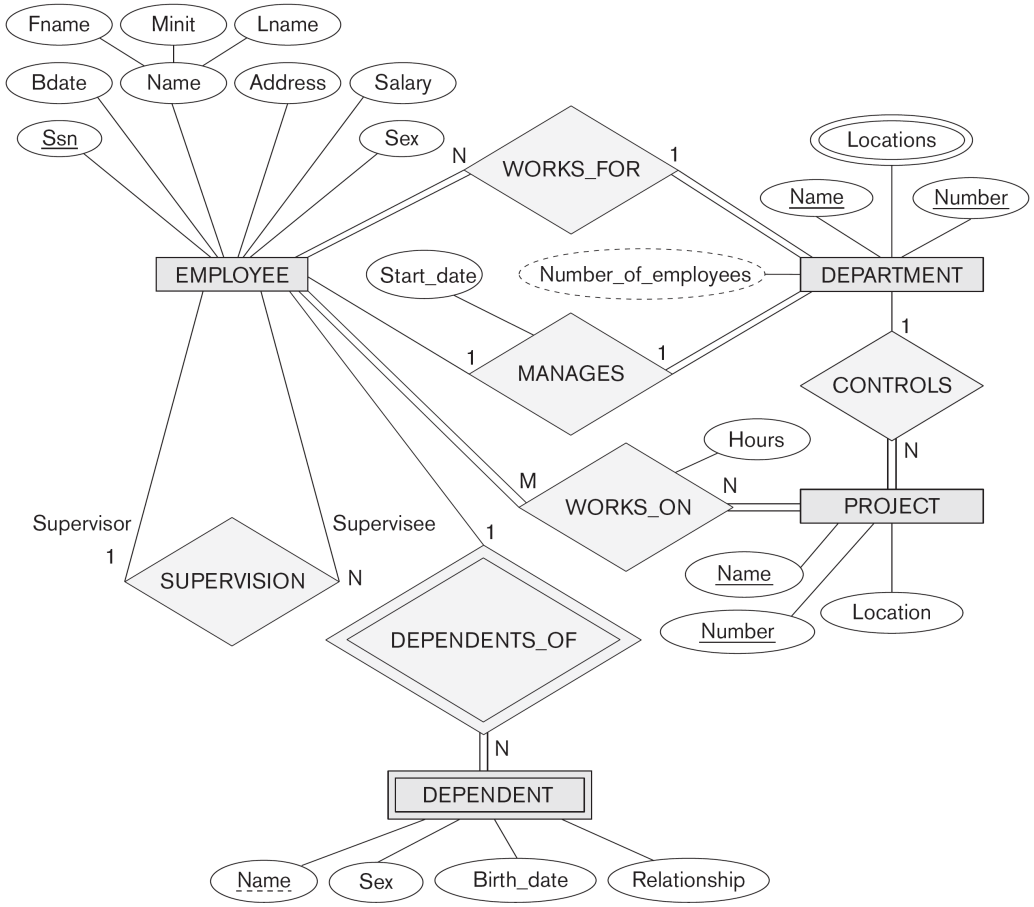
i. Create a new relation *S*

ii. Add as foreign keys the primary keys of the corresponding relation

iii. Add the attribute to *S* (if composite, the simple attributes); the combination of all attributes in *S* forms the primary key

Example ERD

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



Step 6 Result

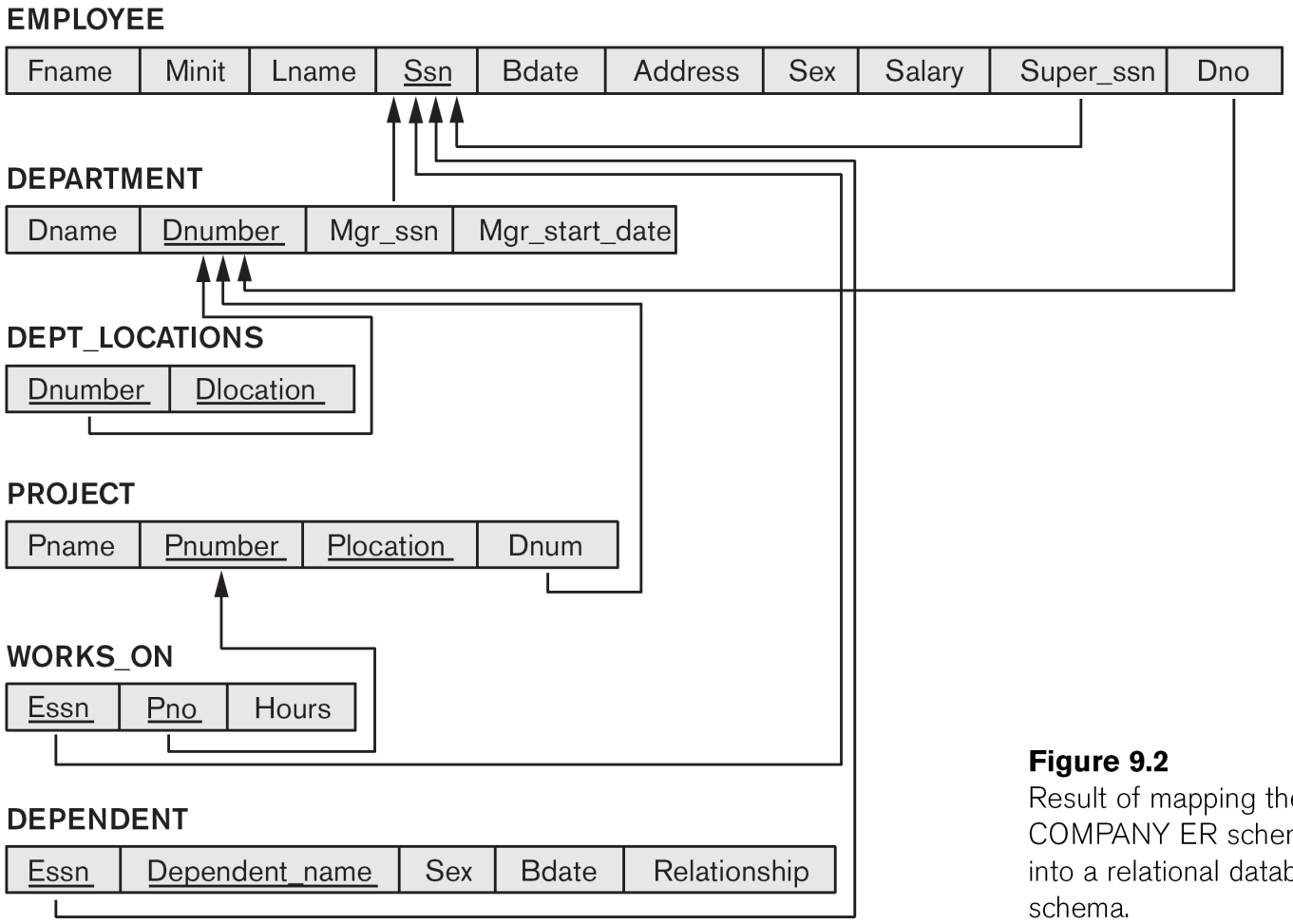


Figure 9.2
Result of mapping the COMPANY ER schema into a relational database schema.

Step 8: Specialization/Generalization

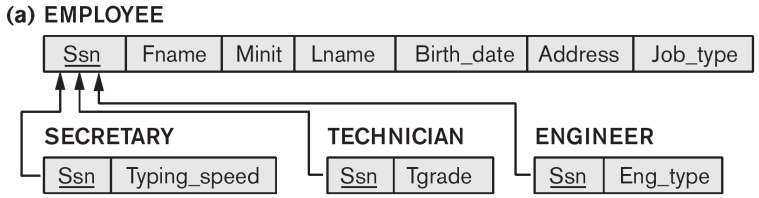
A. Multiple relations – subclass and superclass
   - Always works

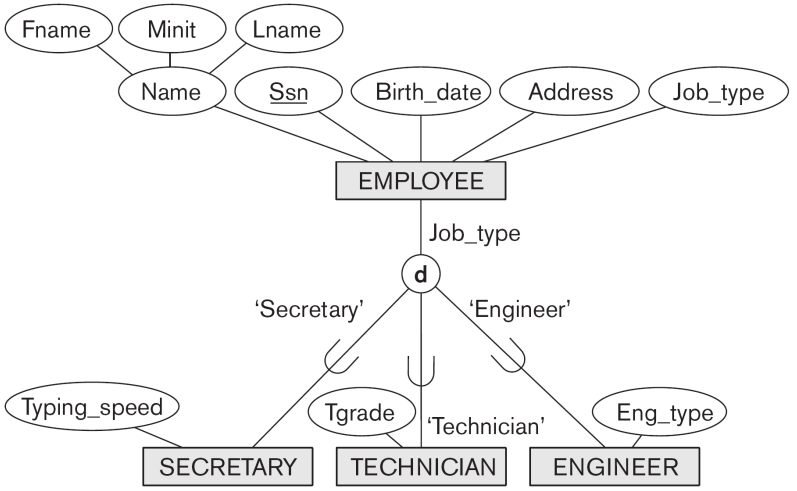
B. Multiple relations – subclass only
   - Should only be used for disjoint

C. Single relation with one type attribute
   - Only for disjoint, can result in many NULLs

D. Single relation with multiple type attributes
   - Better for overlapping, could be disjoint

Specialization/Generalization (A)





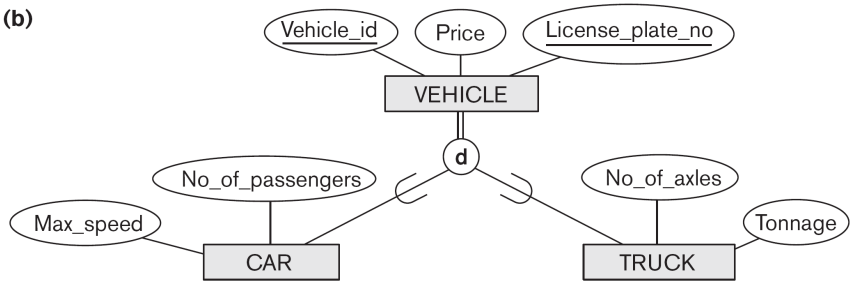
Specialization/Generalization (B)

(b) CAR

Vehicle_id	License_plate_no	Price	Max_speed	No_of_passengers

TRUCK

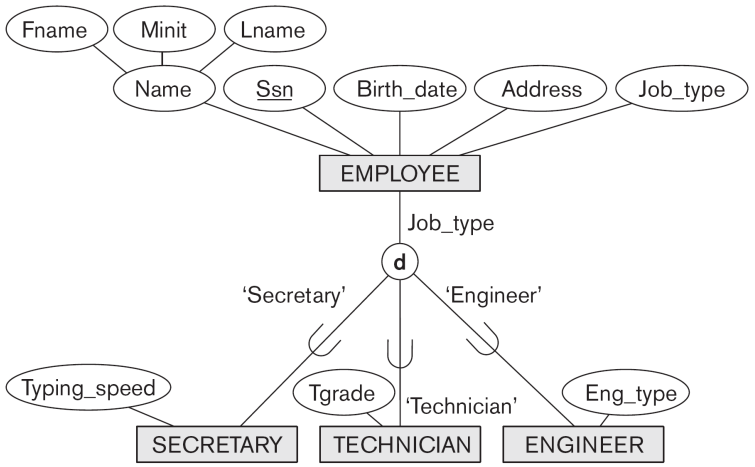
Vehicle_id	License_plate_no	Price	No_of_axles	Tonnage



Specialization/Generalization (C)

(c) EMPLOYEE

Ssn	Fname	Minit	Lname	Birth_date	Address	Job_type	Typing_speed	Tgrade	Eng_type



Specialization/Generalization (D)

(d) PART

Part_no	Description	Mflag	Drawing_no	Manufacture_date	Batch_no	Pflag	Supplier_name	List_price