Database Systems

Lecture #06

Sang-Wook Kim

Hanyang University

Objectives

◆ To learn relational database design

● ER model to relational model mapping

◆ Relational Database Design by ER-to-Relational Mapping

- Mapping Entity Types

- Mapping Attributes

- Mapping Relationship Types

◆ Case Study

◆ Design a relational database schema

- ● Based on the result of conceptual design

◆ Mapping ER model to relational model

Mapping of Regular Entity Types

◆ Map each regular entity type as a relation

- Called *entity relations*

◆ Attributes

- All simple attributes of the entity type

◆ Primary key

- Choose one from key attributes of the entity type

Mapping of Weak Entity Types

◆ Map each weak entity type as a relation

◆ Attributes

- All simple attributes of the entity type

- Primary key attribute of the owner as a foreign key

◆ Primary key

- Choose one from key attributes of the entity type

- Combine it with the foreign key attribute

◆ Map each multivalued attribute as a new relation

◆ Attributes

- Multivalued attributes
- Key attributes of the entity type relation as FK

◆ Primary key

- Combination of all attributes
  - Multivalued attributes + FK

Mapping of Composite Attributes

◆ Map each composite attribute as a new relation

◆ Attributes

- Simple attributes from the component attributes

- Key attributes of the entity type relation as foreign key

◆ Primary key

- Foreign key

Mapping of Composite Attributes

◆ Alternative approach

    ● Add all simple component attributes to the corresponding relation of the entity type

◆ Identify relations that correspond to entity types participating in the relationship type

◆ Choose one of the relations and include the *PK of the other one* as the FK

   ● Better to choose an entity type with *total participation*

◆ Include simple attributes of 1:1 relationship type as attributes of the relation

◆ Identify relation that represents participating entity type at *N*-side of relationship type

◆ Include *PK of the other entity type* as FK

◆ Include simple attributes of 1:*N* relationship type as attributes of the relation

◆ Map each *M:N* relationship type as an *independent relation*

◆ Attributes

- PK of both participating entity types as FK

- Simple attributes of the relationship type

◆ Primary key

- Combination of all the foreign keys

Mapping of *N*-ary Relationship Types

◆ Map each *N-ary* relationship type as a relation

◆ Attributes

  ● PK of participating entity types as FK

  ● Simple attributes of the relationship type

◆ Primary key

  ● Combination of all foreign keys
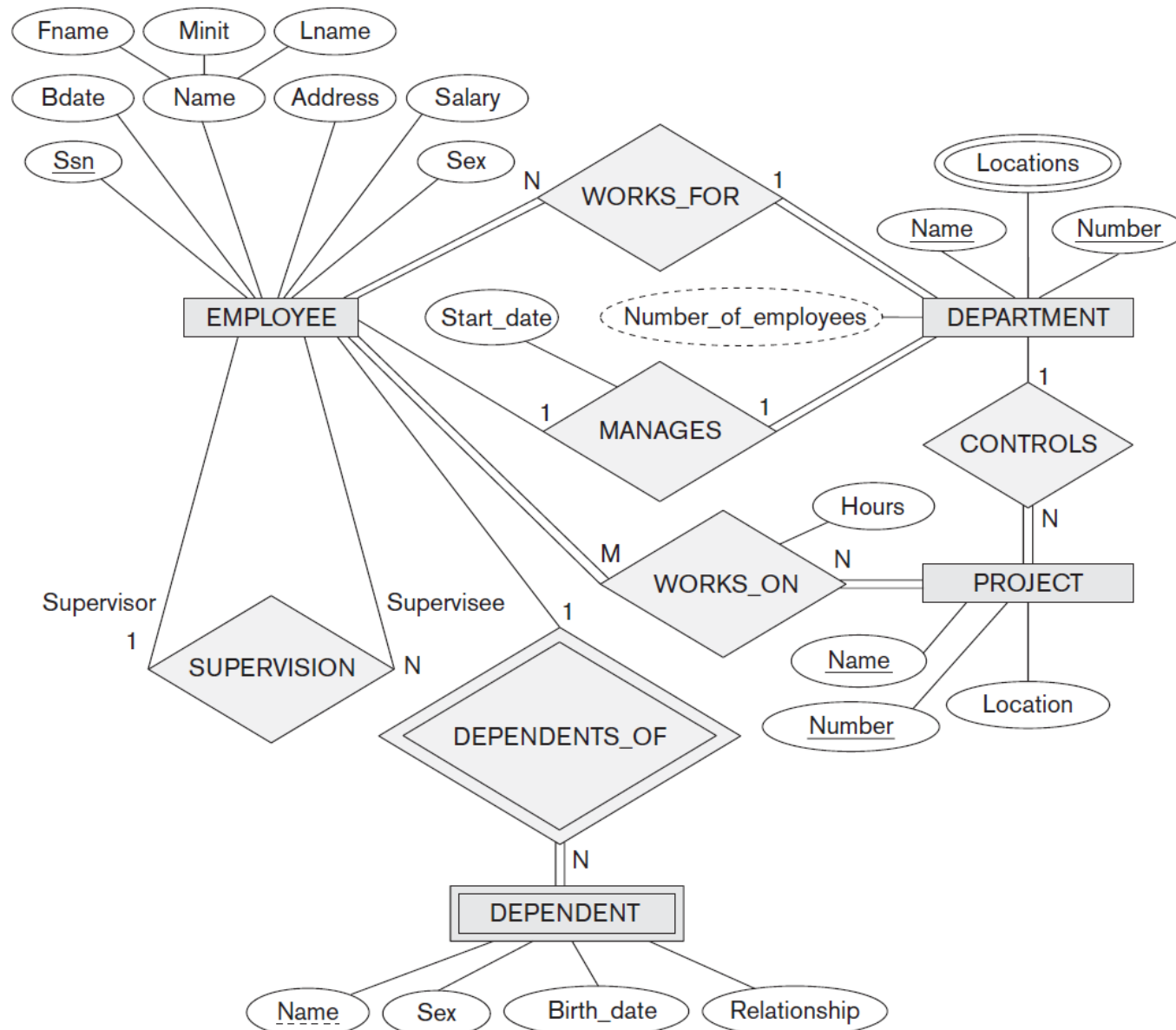
◆ Mapping of regular entity types

- EMPLOYEE

  - Create a new relation EMPLOYEE

  - Include simple attributes

    - **Ssn**, **Bdate**, **Address**, **Sex**, **Salary**

  - Set **<u>Ssn</u>** as PK

◆ Mapping of regular entity types

- DEPARTMENT

  - Create a new relation DEPARTMENT

  - Include simple attributes

    - **Name** as **Dname**, **Number** as **Dnumber**

  - Set **Dnumber** as PK

◆ Mapping of regular entity types

- PROJECT

  ▪ Create a new relation PROJECT

  ▪ Include simple attributes

    - **Name** as **Pname**, **Number** as **Pnumber**, **Location** as **Plocation**

  ▪ Set **<u>Pnumber</u>** as PK

◆ Mapping of weak entity types

- ● DEPENDENT

  - ▪ Create a new relation DEPENDENT

  - ▪ Include simple attributes

    - **Name** as **Dependent_name**, **Sex**, **Birth_date** as **Bdate**, **Relationship**

  - ▪ Include PK of the owner entity type

    - **Ssn** as **Essn**

  - ▪ Set **Essn** + **Dependent_name** as PK

◆ Mapping of multivalued attributes

- **Locations** in DEPARTMENT

  - Create a new relation DEPT_LOCATIONS

  - Include PK of DEPARTMENT as FK

    - **Dnumber**

  - Include multivalued attribute

    - **Locations** as **Dlocation**

  - Set **Dnumber** + **Dlocation** as PK

◆ Mapping of composite attributes

- EMPLOYEE

  ▪ Include simple attributes from the composite attribute **Name**

    - **Fname**, **Minit**, **Lname**

◆ Mapping of binary 1:1 relationship types

- MANAGES

  - DEPARTMENT is *total participation*

  - Include PK of EMPLOYEE as FK of DEPARTMENT

    - **Ssn** as **Mgr_ssn**

  - Include simple attributes of MANAGES

    - **Start_date** as **Mgr_start_date**

◆ Mapping of binary 1:*N* relationship types

- WORKS_FOR

  ▪ EMPLOYEE is *N-side entity type*

  ▪ Include PK of DEPARTMENT as FK of EMPLOYEE

    - **Dnumber** as **Dno**

◆ Mapping of binary 1:*N* relationship types

- CONTROLS

  - PROJECT is *N-side entity type*

  - Include PK of DEPARTMENT as FK of PROJECT

    - **Dnumber** as **Dnum**

◆ Mapping of binary 1:N relationship types

- ● SUPERVISION

    - ▪ Both sides (Supervisor, Supervisee) are EMPLOYEE

    - ▪ Include PK of EMPLOYEE as FK of EMPLOYEE

        - **Ssn** as **Super_ssn**

◆ Mapping of binary *M:N* relationship types

- WORKS_ON

  - Create a *new relation WORKS_ON*

  - Include PK of EMPLOYEE and PROJECT
    - **Ssn** as **Essn**, **Pnumber** as **Pno**

  - Include simple attributes of WORKS_ON
    - **Hours**

  - Set **Essn** + **Pno** as PK

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

DEPT_LOCATIONS

Dnumber	Dlocation

PROJECT

Pname	Pnumber	Plocation	Dnum

WORKS_ON

Essn	Pno	Hours

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
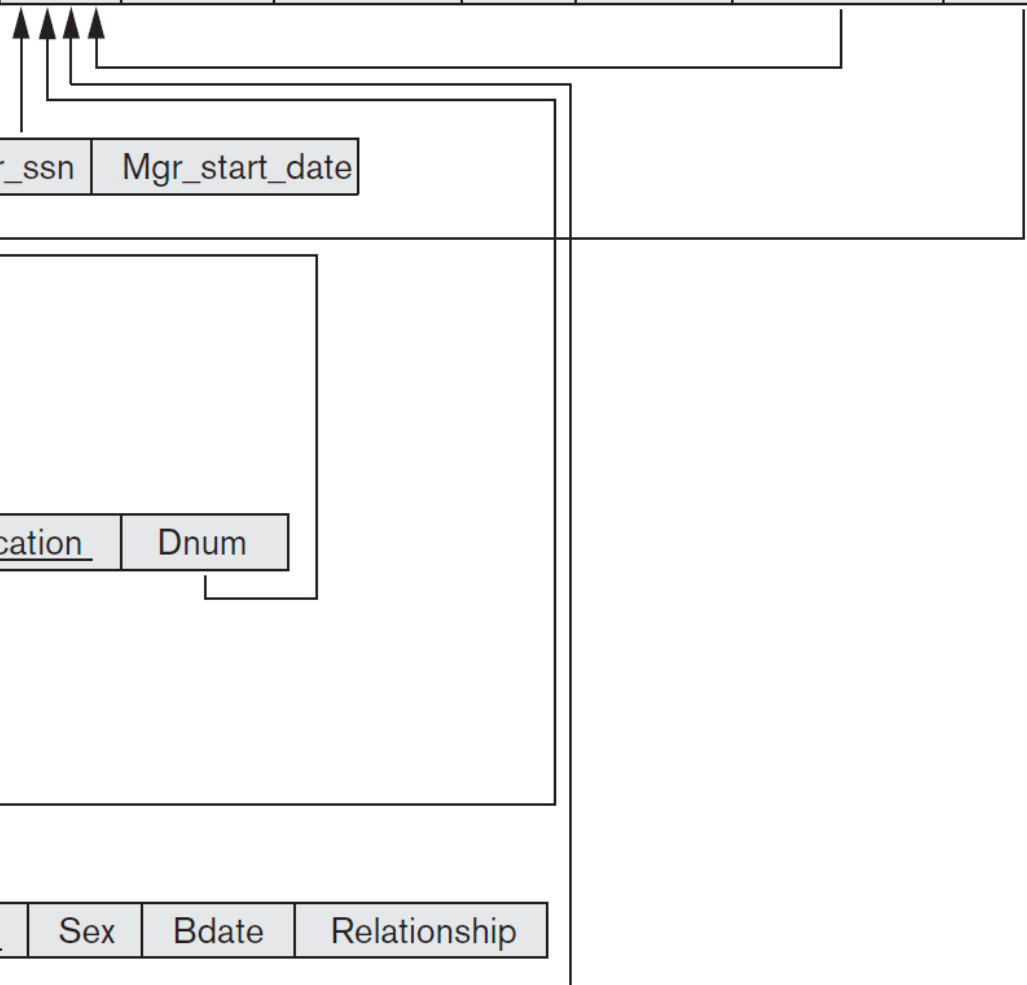
◆ Map conceptual schema design from the ER schema to a relational database schema

- Algorithm for ER-to-relational mapping

- Illustrated by examples from the COMPANY database

References

1. Chen, Peter Pin-Shan. "The entity-relationship model—toward a unified view of data." *ACM Transactions on Database Systems (TODS)* 1.1 (1976): 9-36.

2. Batini, Carlo, Stefano Ceri, and S. Navathe. *Entity Relationship Approach*. Elsevier Science Publishers BV (North Holland), 1989.

Data
And
Knowledge
Engineering

Have a nice day!

Data
And
Knowledge
Engineering