

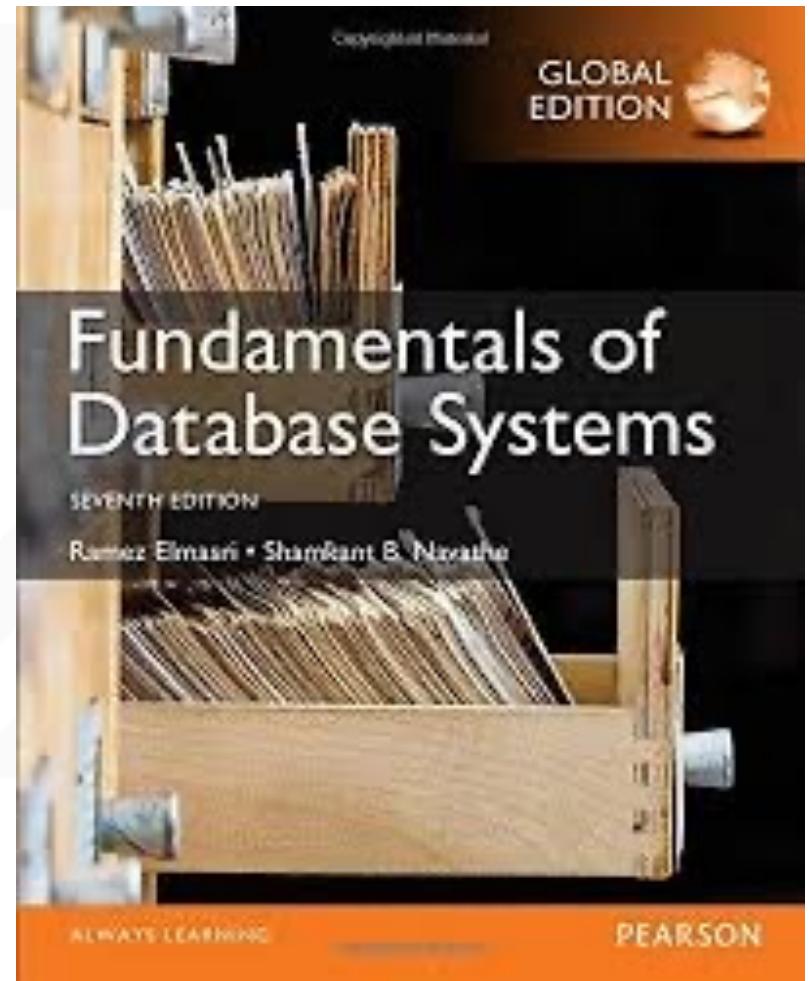
Database Systems

Program in Computer Engineering
School of Engineering

King Mongkut's Institute of Technology Ladkrabang

Text

- Ramez Elmasri and Shamkant B. Navathe.
“Fundamentals of Database Systems”
7th Edition., Pearson, 2017

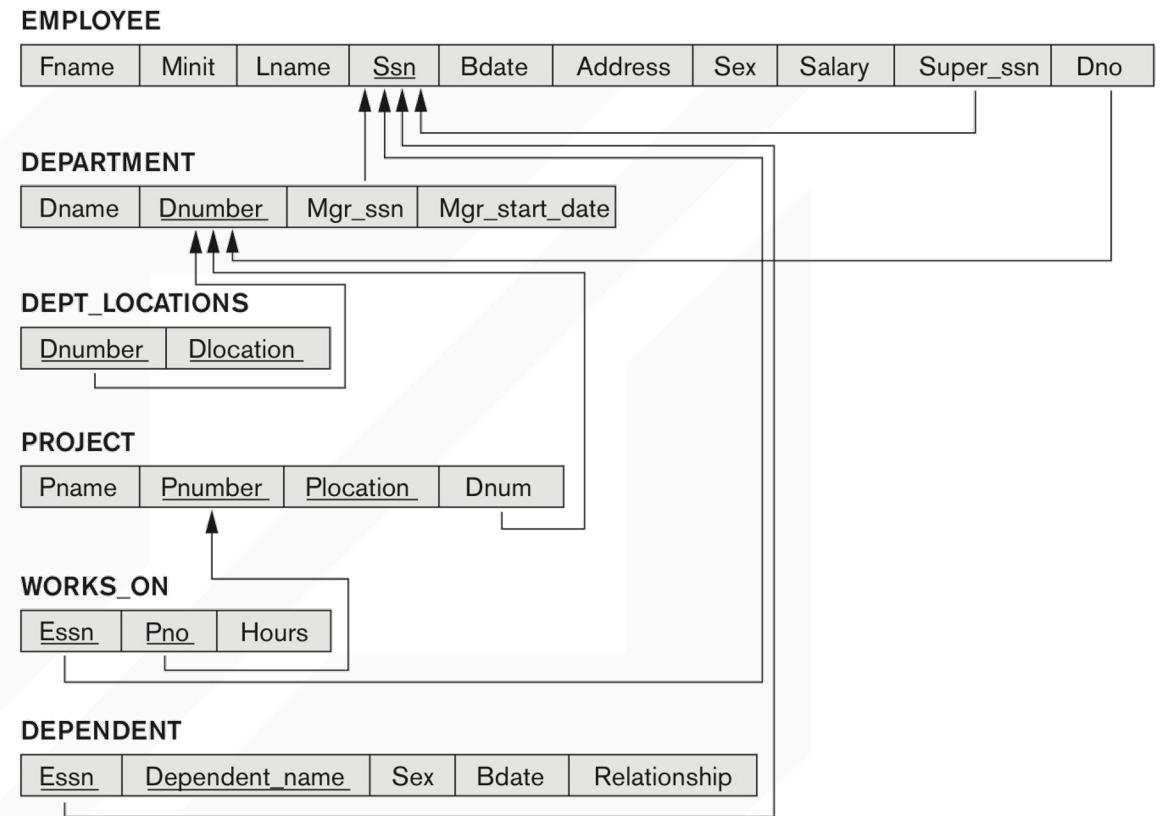
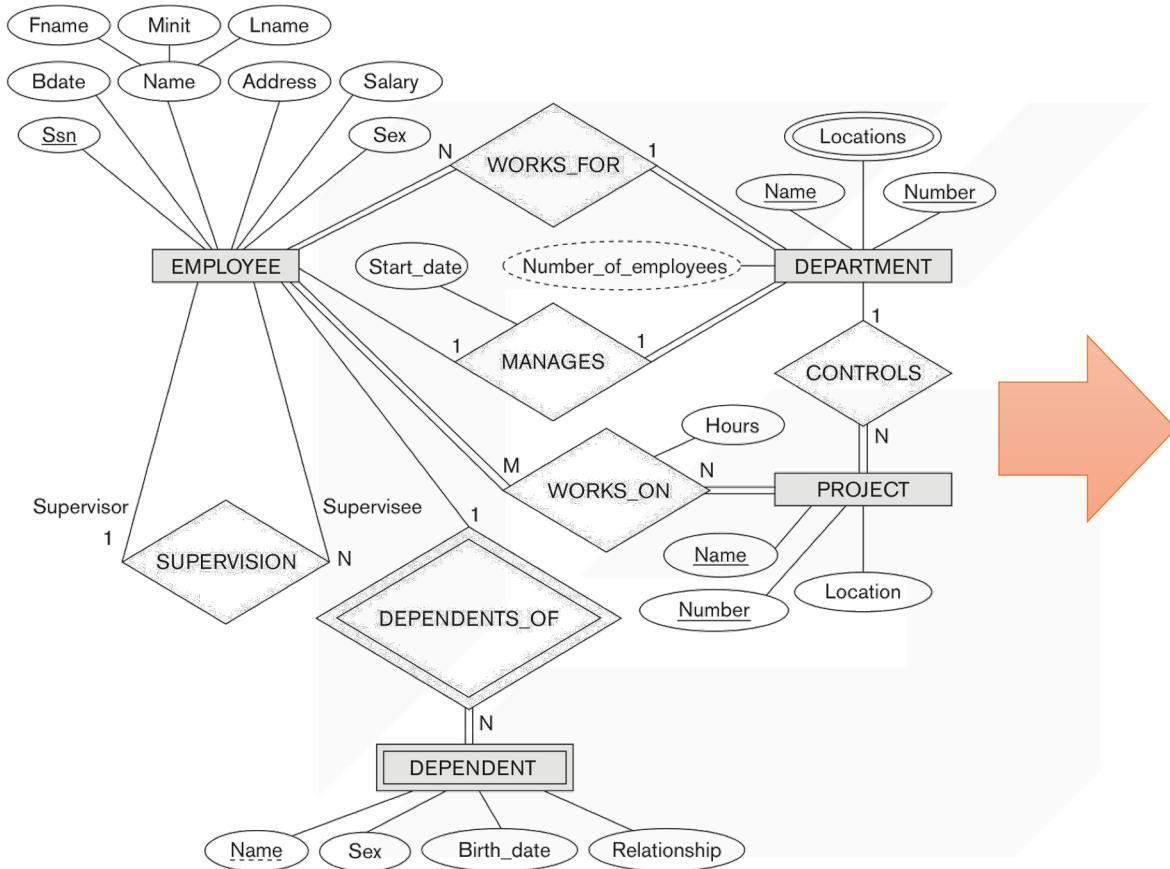


Chapter 9 (Partial)

Relational Database Design by ER to Relational Mapping

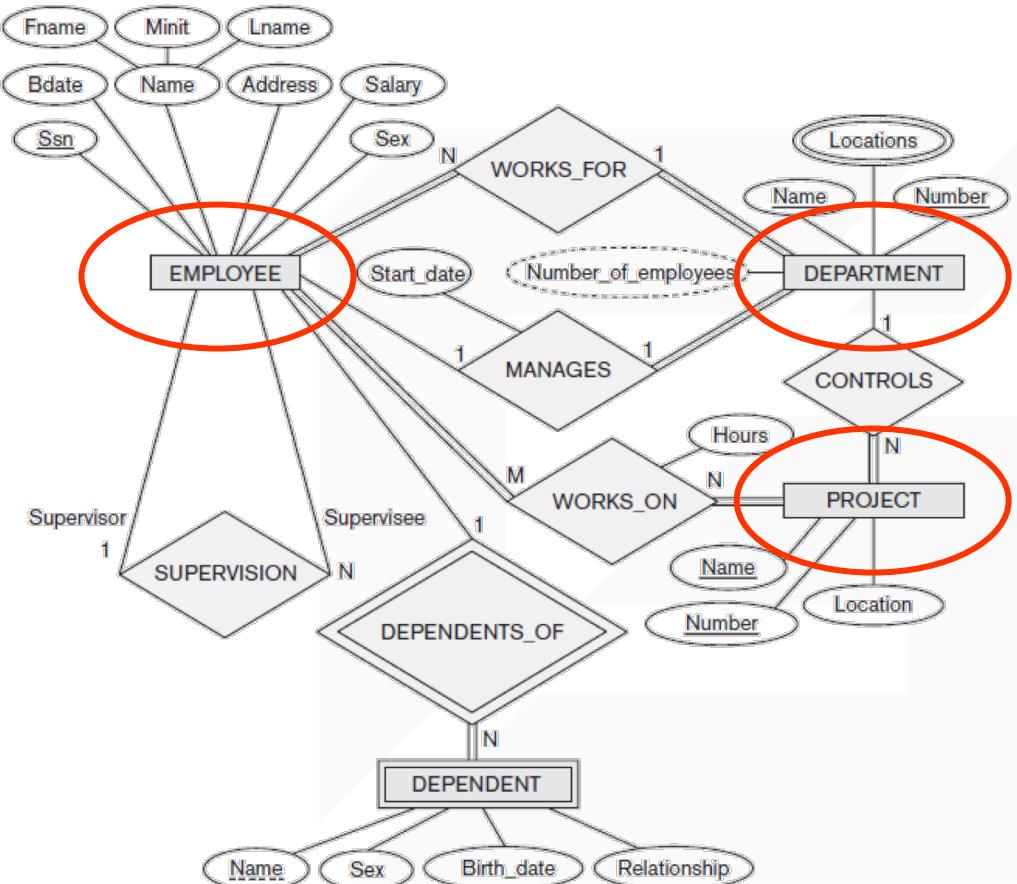
Goals during Mapping

- Preserve all information (that includes all attributes)
- Maintain the constraints to the extent possible (Relational Model cannot preserve all constraints
 - E.g., max cardinality ratio such as 1:10 in ER
- Minimize null values
 - The mapping procedure described has been implemented in many commercial tools.



ER-to-Relational Mapping Algorithm

- Step 1: Mapping of Regular Entity Types.
 - For each **regular (strong)** entity type E in the ER schema, create a relation R that includes all the simple attributes of E.
 - Choose one of the key attributes of E as the primary key for R.
 - If the chosen key of E is composite, the set of simple attributes that form it will together form the primary key of R.



EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary
-------	-------	-------	-----	-------	---------	-----	--------

DEPARTMENT

Dname	Dnumber
-------	---------

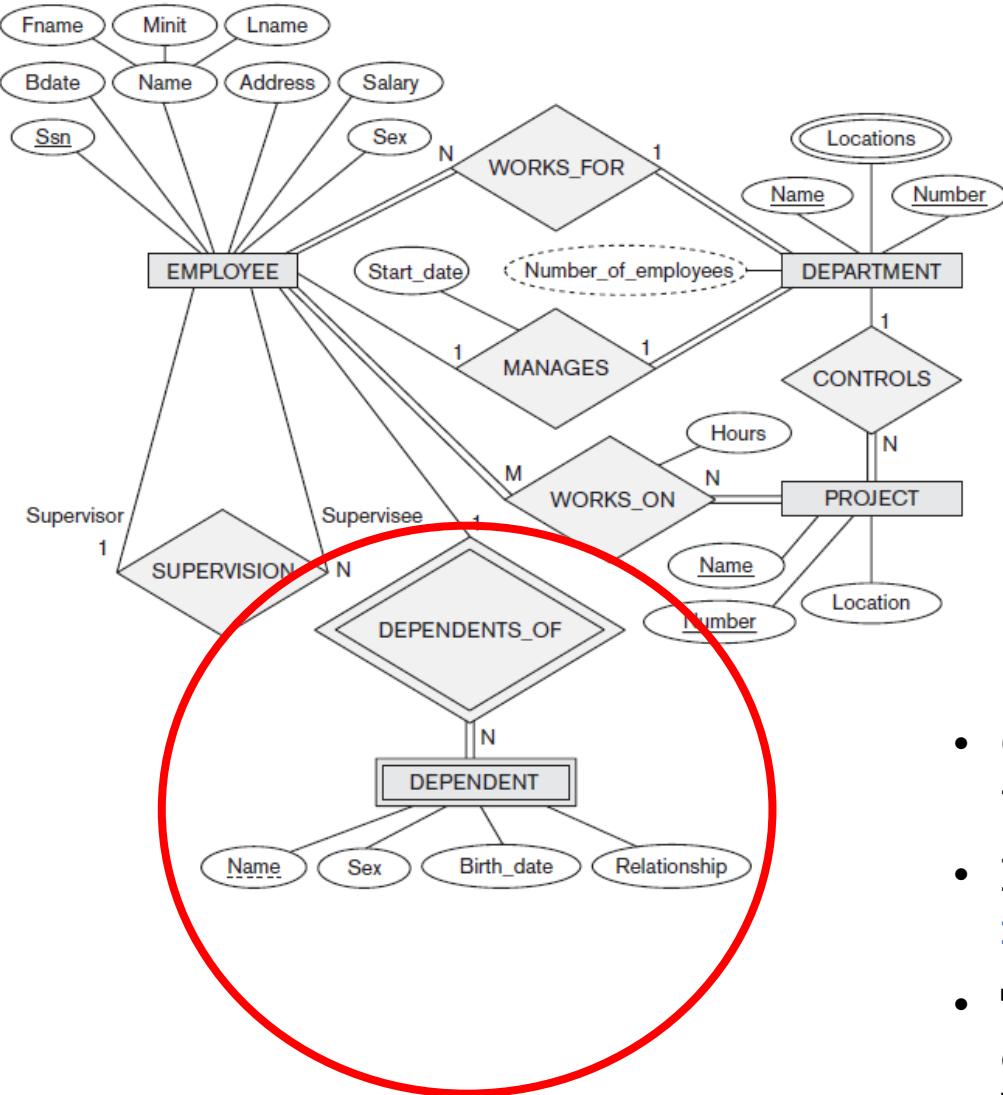
PROJECT

Pname	Pnumber	Plocation
-------	---------	-----------

- We create the relations **EMPLOYEE**, **DEPARTMENT**, and **PROJECT** in the relational schema corresponding to the regular entities in the ER diagram.
- **SSN**, **DNUMBER**, and **PNUMBER** are the primary keys for the relations **EMPLOYEE**, **DEPARTMENT**, and **PROJECT** as shown.

• Step 2: Mapping of Weak Entity Types

- For each **weak entity type W** in the ER schema **with owner entity type E**, **create a relation R & include all simple attributes** (or simple components of composite attributes) of W as attributes of R.
- Also, **include as foreign key attributes** of R the primary key attribute(s) of the relation(s) that correspond to the owner entity type(s).
- The primary key of R is the combination of the primary key(s) of the owner(s) and the partial key of the weak entity type W, if any.



EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary
-------	-------	-------	------------	-------	---------	-----	--------

DEPARTMENT

Dname	<u>Dnumber</u>
-------	----------------

PROJECT

Pname	<u>Pnumber</u>	Plocation
-------	----------------	-----------

DEPENDENT

<u>Essn</u>	<u>Dependent_name</u>	Sex	Bdate	Relationship
-------------	-----------------------	-----	-------	--------------

- Create the relation **DEPENDENT** in this step to correspond to the weak entity type **DEPENDENT**.
- Include the primary key **SSN** of the **EMPLOYEE** relation as a **foreign key attribute** of **DEPENDENT** (renamed to **ESSN**).
- The primary key of the **DEPENDENT** relation is the combination {**ESSN**, **DEPENDENT_NAME**} because **DEPENDENT_NAME** is the partial key of **DEPENDENT**.

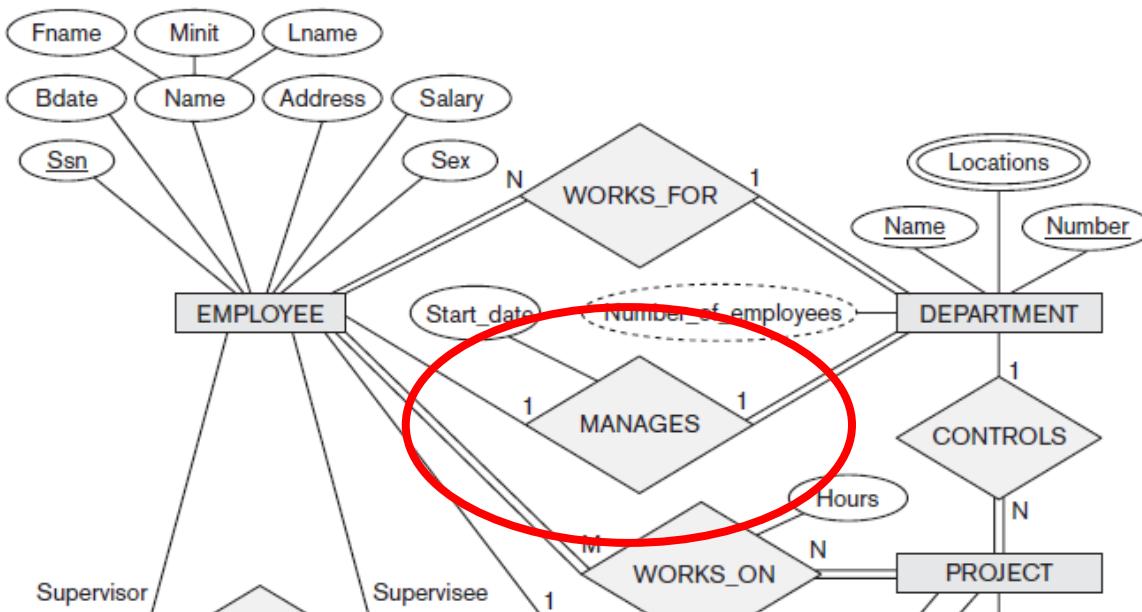
- Step 3: Mapping of Binary 1:1 Relation Types
 - For each binary 1:1 relationship type R in the ER schema, identify the relations S and T that correspond to the entity types participating in R.

- There are three possible approaches:

1. Foreign Key (2 relations) approach:

Choose one of the relations-say S-and include a foreign key in S the primary key of T. It is better to choose an entity type with total participation in R in the role of S.

- Example: 1:1 relation MANAGES is mapped by choosing the participating entity type DEPARTMENT to serve in the role of S, because its participation in the MANAGES relationship type is total.



EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary
-------	-------	-------	------------	-------	---------	-----	--------

DEPARTMENT

Dname	<u>Dnumber</u>	<u>Mgr_ssn</u>	<u>Mgr_start_date</u>
-------	----------------	----------------	-----------------------

A red arrow points from the circled attribute Ssn in the EMPLOYEE table to the circled attribute Mgr_ssn in the DEPARTMENT table, indicating they are both foreign keys referencing the primary key Dnumber.

2. **Merged relation (1 relation) option:**

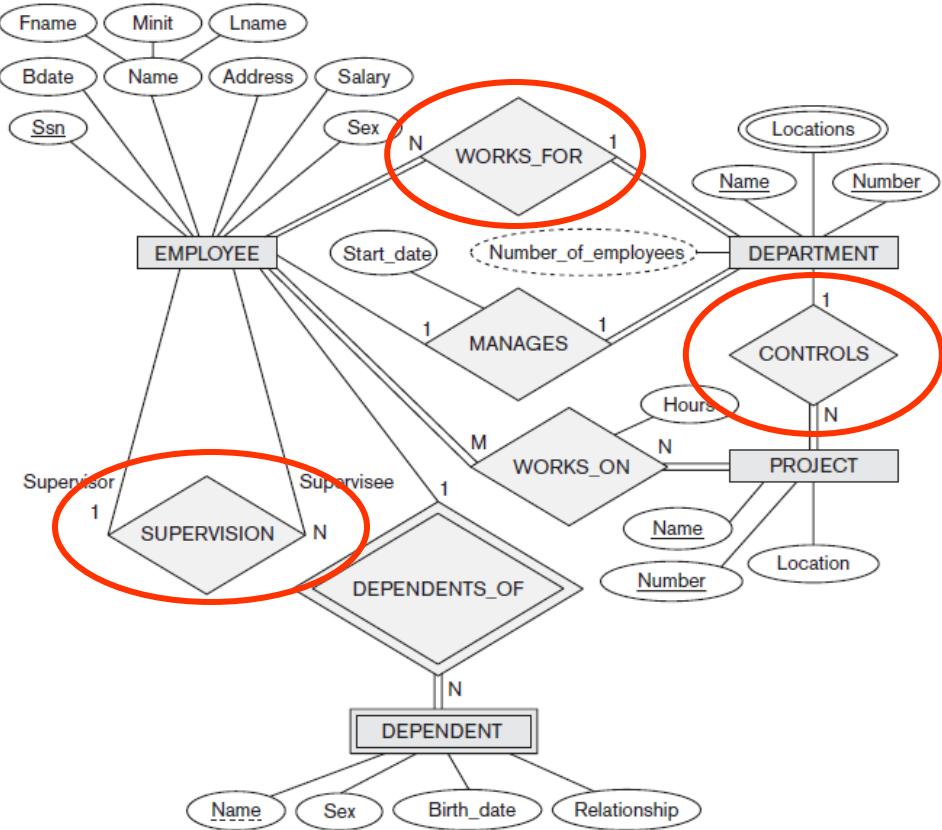
An alternate mapping of a 1:1 relationship type is possible by merging the two entity types and the relationship into a single relation.

This may be appropriate when both participations are total.

3. **Cross-reference or relationship relation (3 relations) option:**

The third alternative is to set up a third relation R for the purpose of cross-referencing the primary keys of the two relations S and T representing the entity types.

- **Step 4: Mapping of Binary 1:N Relationship Types.**
 - For each regular binary 1:N relationship type R, identify the relation S that represent the participating entity type at the N-side of the relationship type.
 - Include as foreign key in S the primary key of the relation T that represents the other entity type participating in R.
 - Include any simple attributes of the 1:N relation type as attributes of S.

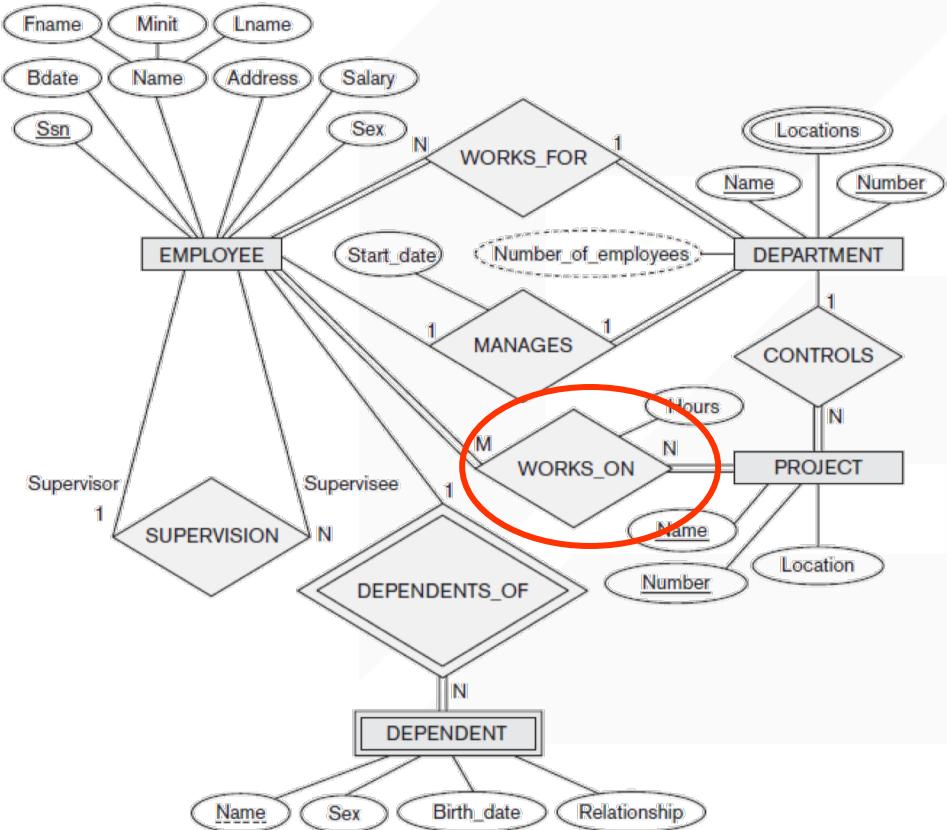


PROJECT				DEPARTMENT			
Pname	<u>Pnumber</u>	Plocation	Dnum	Dname	<u>Dnumber</u>		
EMPLOYEE							
Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary

- Example: 1:N relationship types **WORKS_FOR**, **CONTROLS**, and **SUPERVISION** in the figure.
- For **WORKS_FOR** we include the primary key **DNUMBER** of the **DEPARTMENT** relation as foreign key in the **EMPLOYEE** relation and call it **DNO**.

• Step 5: Mapping of Binary M:N Relationship Types.

- For each regular binary M:N relationship type R,
create a new relation S to represent R. This is a relationship relation.
 - Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types;
their combination will form the primary key of S.
 - Also include any simple attributes of the M:N relationship type (or simple components of composite attributes) as attributes of S.



EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	DNO
-------	-------	-------	------------	-------	---------	-----	--------	-----------	-----

WORKS_ON

<u>Essn</u>	Pno	Hours
-------------	-----	-------

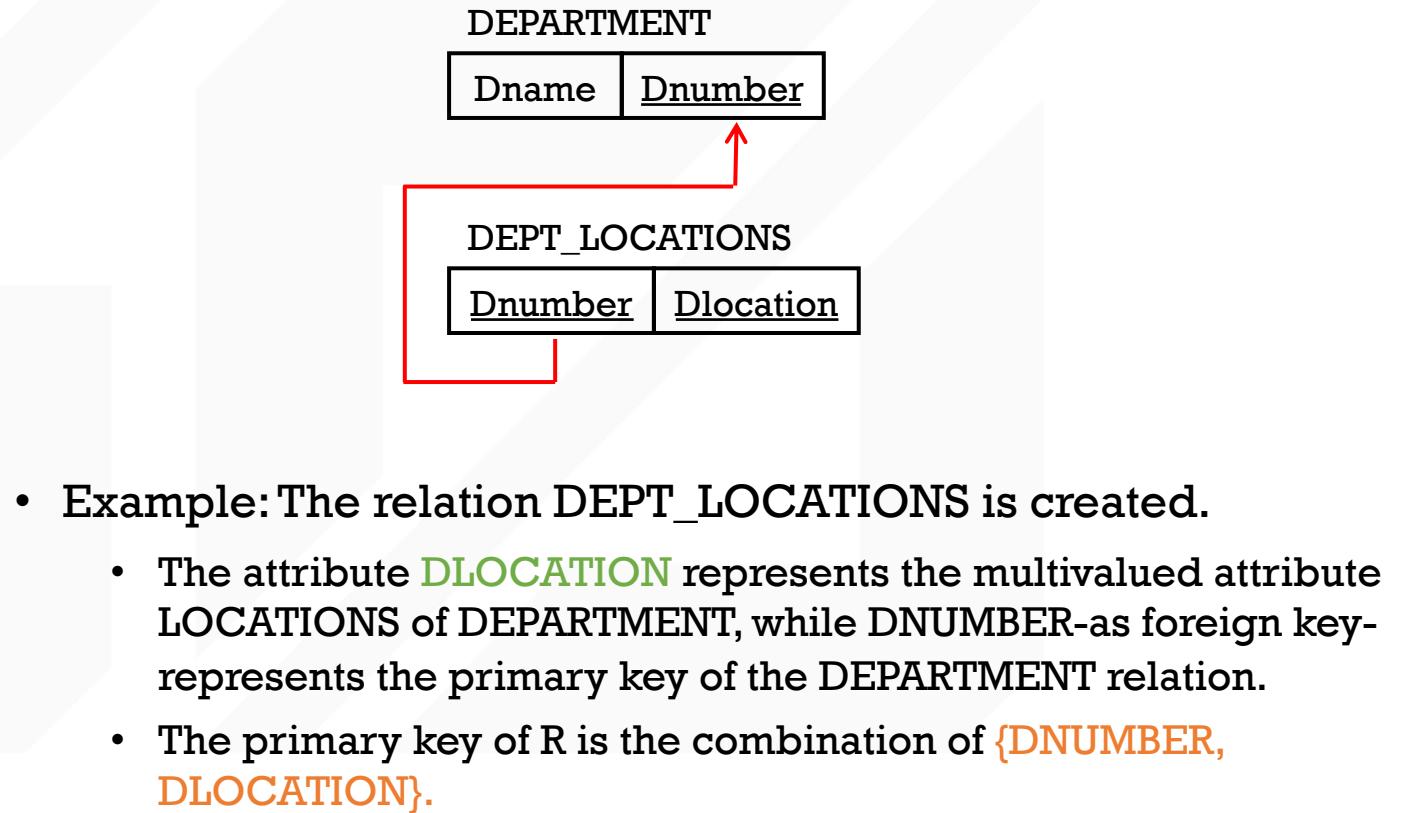
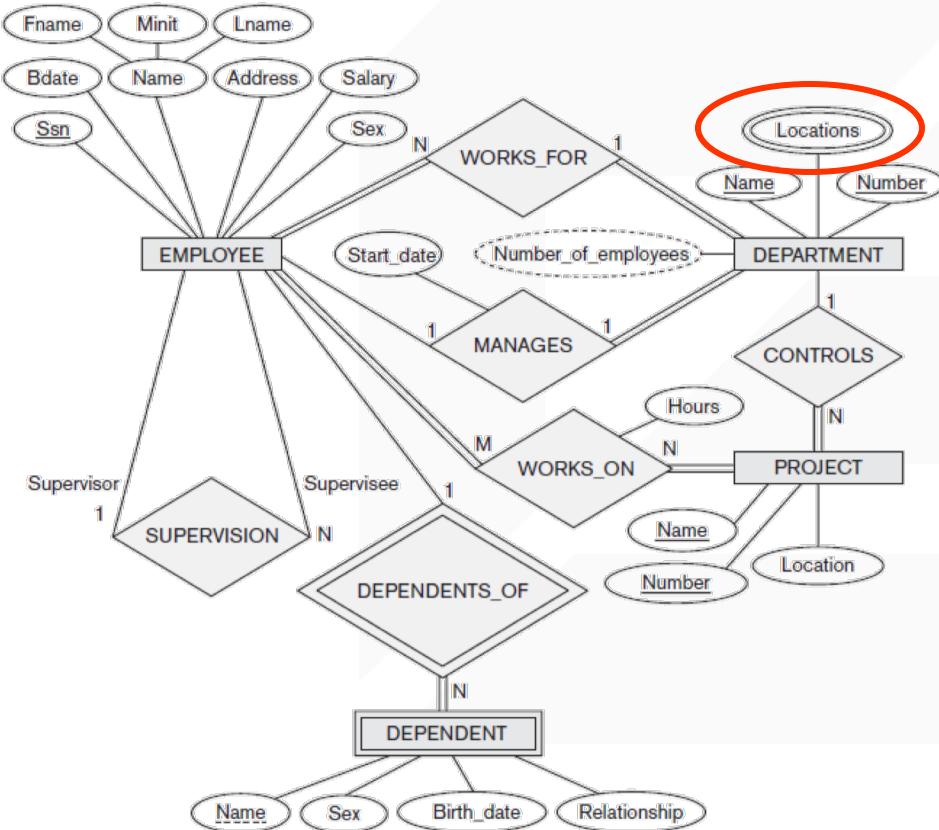
PROJECT

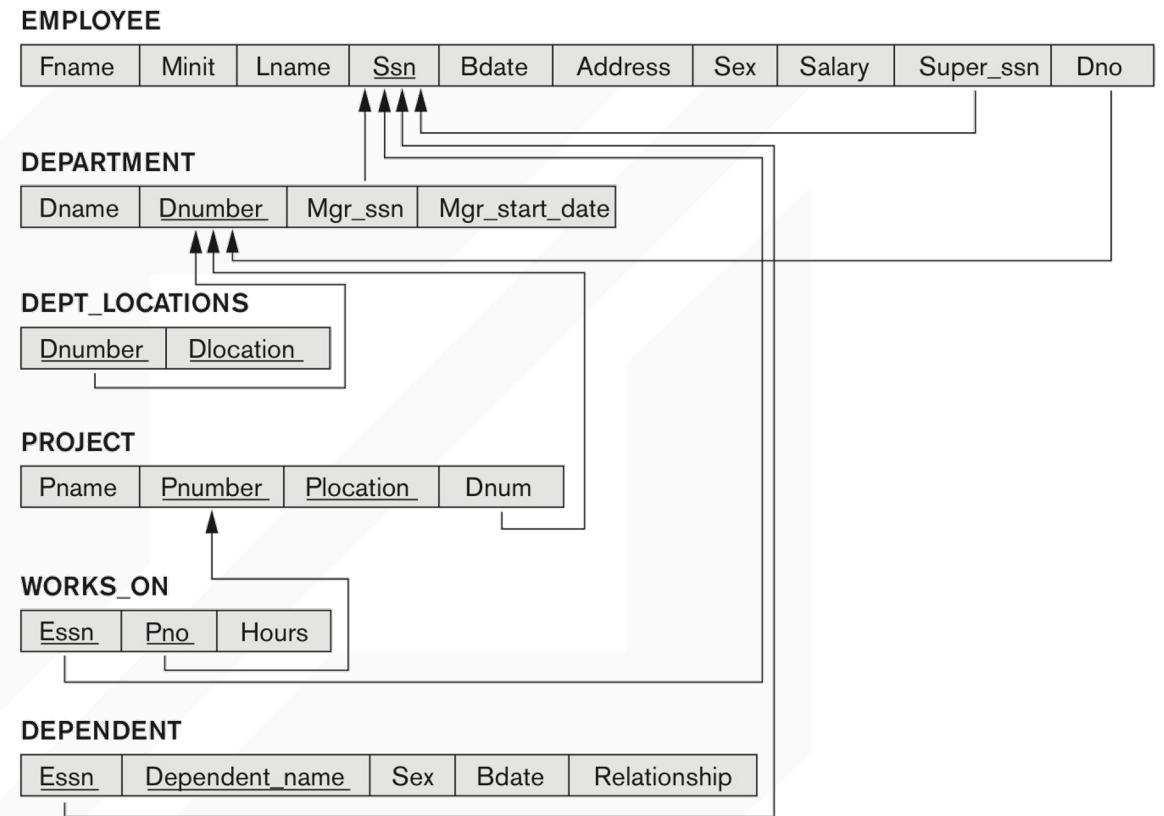
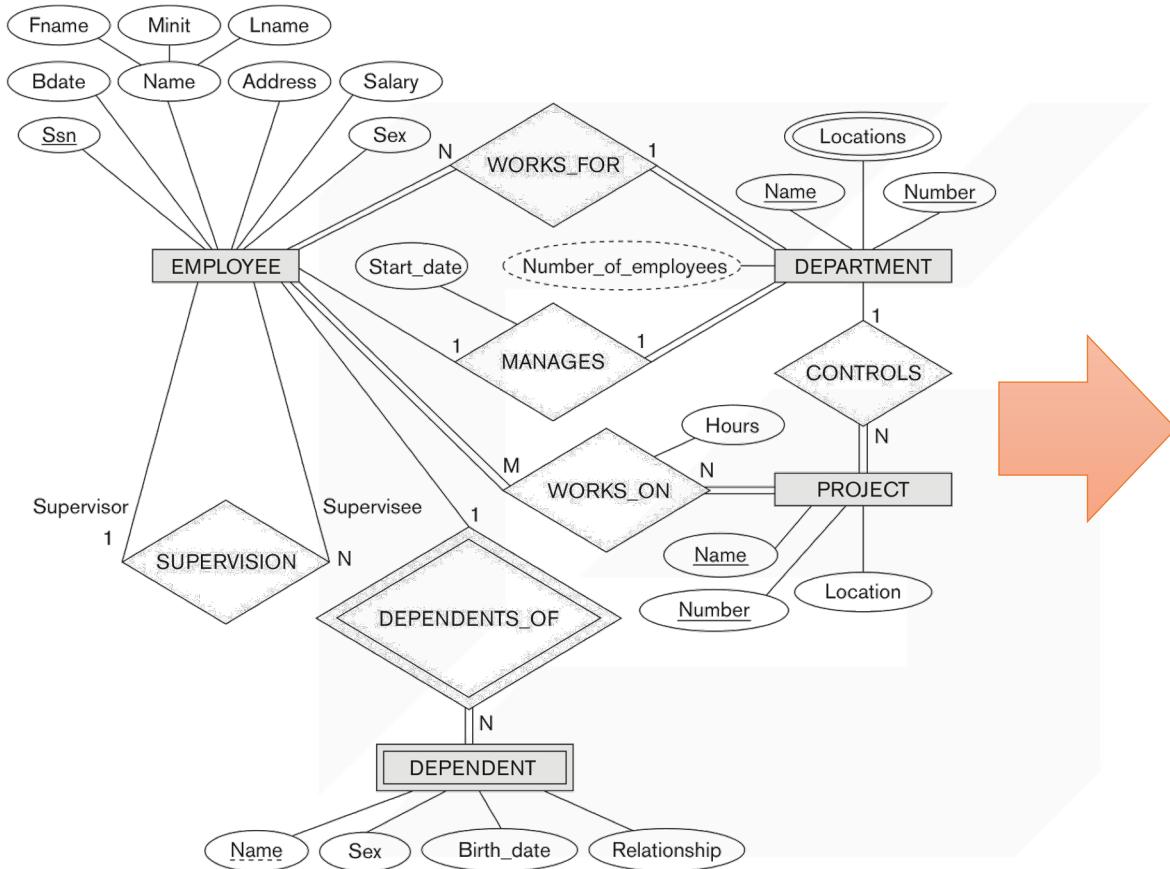
Pname	<u>Pnumber</u>	Plocation	Dnum
-------	----------------	-----------	------

- Example: The M:N relationship type **WORKS_ON** from the ER diagram is mapped by creating a relation **WORKS_ON** in the relational database schema.
 - The **primary keys of the PROJECT and EMPLOYEE relations** are included as foreign keys in **WORKS_ON** and renamed **PNO** and **ESSN**, respectively.
 - Attribute **HOURS** in **WORKS_ON** represents the **HOURS** attribute of the relation type. The primary key of the **WORKS_ON** relation is the combination of the foreign key attributes {**ESSN, PNO**}.

• **Step 6: Mapping of Multivalued attributes.**

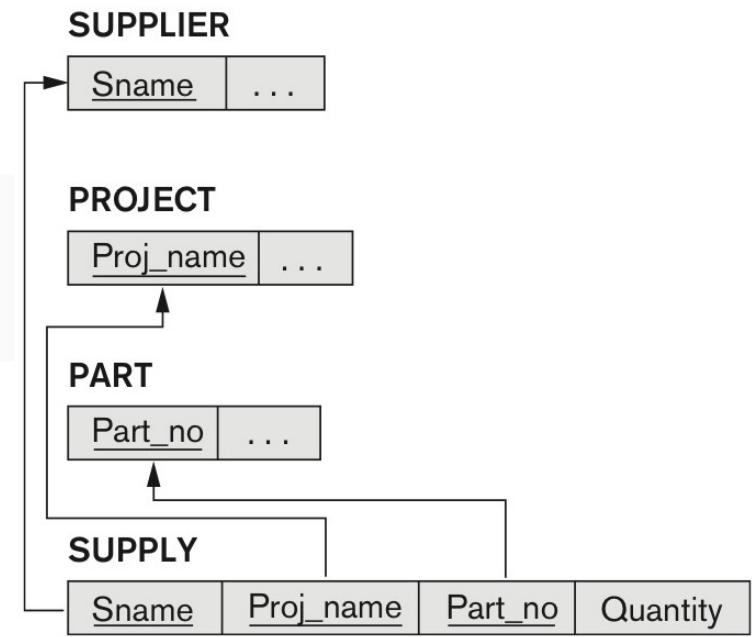
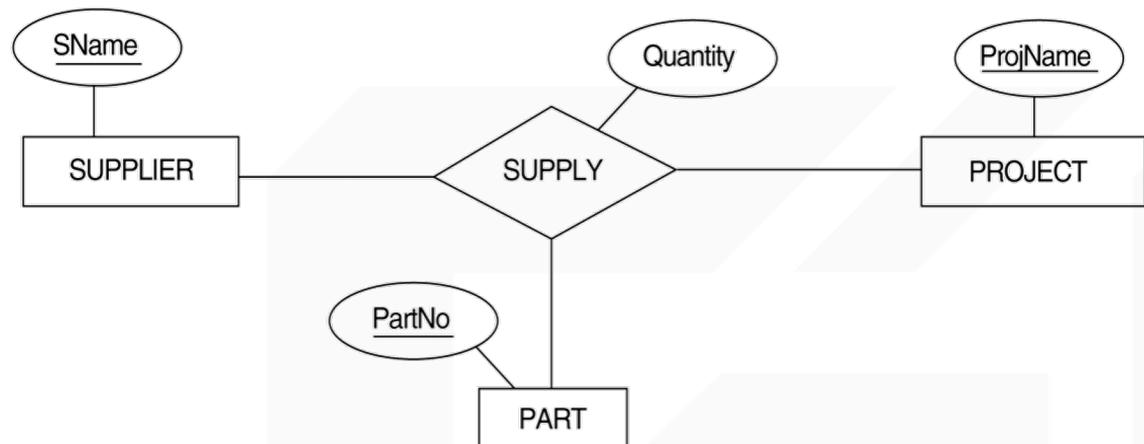
- For each **multivalued attribute A**, create a new relation R.
 - This relation R will include an attribute corresponding to A, plus the primary key attribute K-as a foreign key in R-of the relation that represents the entity type of relationship type that has A as an attribute.
 - The primary key of R is the combination of A and K. If the multivalued attribute is composite, we include its simple components.





• Step 7: Mapping of N-ary Relationship Types.

- For each **n-ary relationship type** R, where $n > 2$,
create a new relationship S to represent R.
 - Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types.
 - Also include any simple attributes of the n-ary relationship type (or simple components of composite attributes) as attributes of S.



- Example: The relationship type SUPPLY in the ER.
- This can be mapped to the relation SUPPLY shown in the relational schema, whose primary key is the combination of the three foreign keys {SNAME, PARTNO, PROJNAME}.

Table 9.1 Correspondence between ER and Relational Models

ER MODEL	RELATIONAL MODEL
Entity type	<i>Entity</i> relation
1:1 or 1:N relationship type	Foreign key (or <i>relationship</i> relation)
M:N relationship type	<i>Relationship</i> relation and <i>two</i> foreign keys
<i>n</i> -ary relationship type	<i>Relationship</i> relation and <i>n</i> 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

