Reduction of ER Model to Relational Model

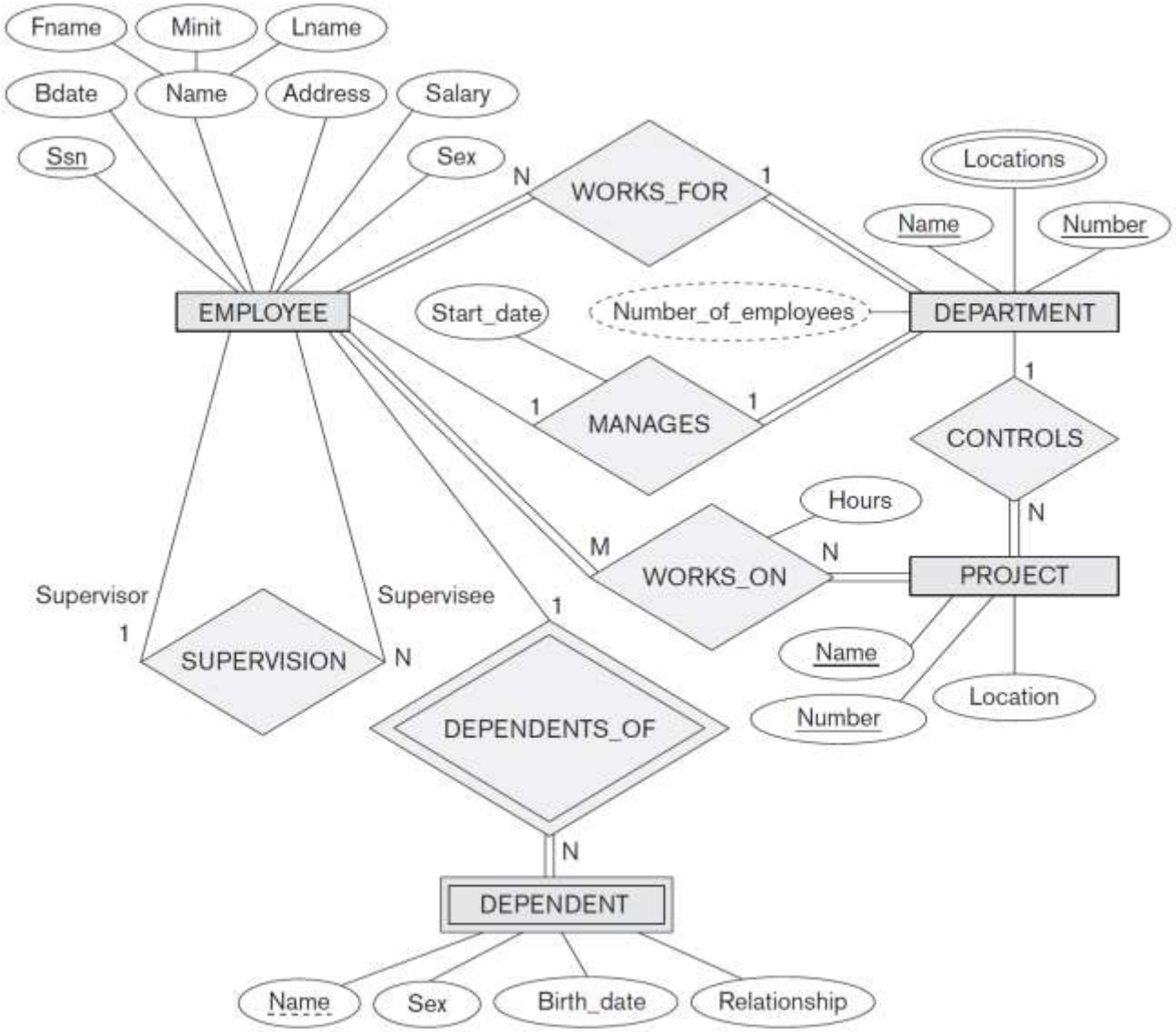
by

Dr. Pratik Roy

ER-to-Relational Mapping Algorithm

- Mapping of Regular Entity Types
- Mapping of Weak Entity Types
- Mapping of Binary 1:1 Relation Types
- Mapping of Binary 1:N Relationship Types
- Mapping of Binary M:N Relationship Types
- Mapping of Multivalued attributes
- Mapping of N-ary Relationship Types
- Options for Mapping Specialization or Generalization
- Mapping of Aggregation

The ER conceptual schema diagram for the COMPANY database



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.
- Include only the simple component attributes of a composite attribute.
- Choose one of the key attributes of *E* as the primary key for *R*.
- If the chosen key of *E* is a composite, then the set of simple attributes that form it will together form the primary key of *R*.

Mapping of Regular Entity Types

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary

DEPARTMENT

Dname	Dnumber

PROJECT

Pname	Pnumber	Plocation

- **Example**
  - 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.

Mapping of Weak Entity Types

- For each weak entity type W in the ER schema with owner entity type E, create a relation R and include all simple attributes (or simple components of composite attributes) of W as attributes of R.
- 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.
- If there is a weak entity type E_2 whose owner is also a weak entity type E_1, then E_1 should be mapped before E_2 to determine its primary key first.

Mapping of Weak Entity Types

❑Example

- Create the relation DEPENDENT 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.

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

Mapping of Binary 1:1 Relationship 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:
  - **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.
    - Include the primary key of the EMPLOYEE relation as foreign key in the DEPARTMENT relation.
    - Include simple attribute Start_date of MANAGES relationship type in DEPARTMENT relation.

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

Mapping of Binary 1:1 Relationship Types

- **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*.
- **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.

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

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

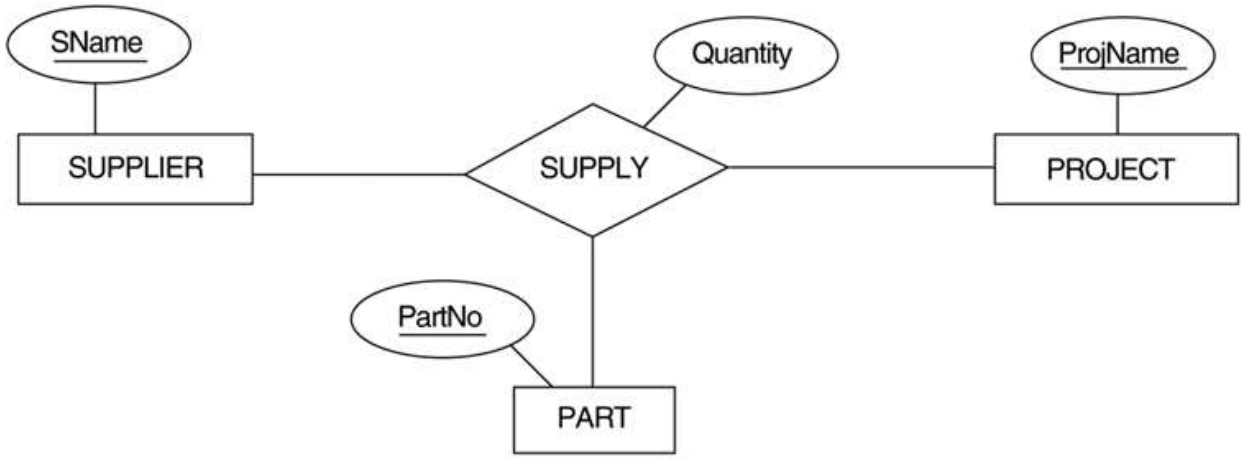
Mapping of Multivalued attributes

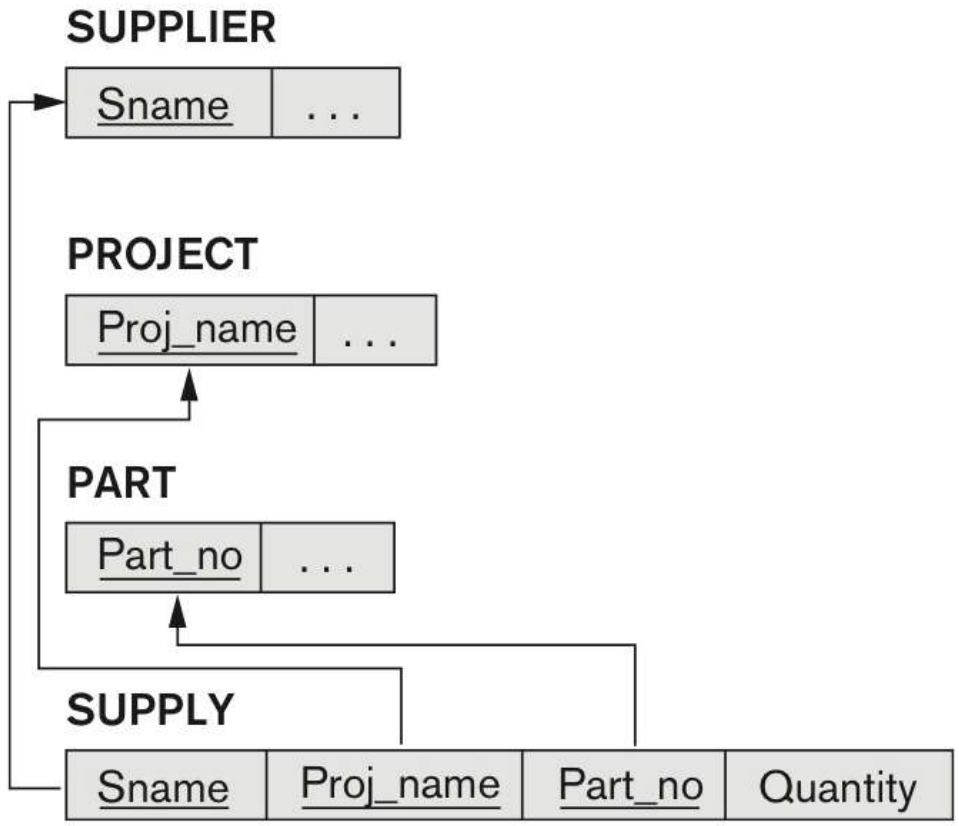
- For each multivalued attribute A, create a new relation R.
- R will include an attribute corresponding to A, plus the primary key attribute K as foreign key in R of the relation that represents the entity type or relationship type that has A as a multivalued attribute.
- The primary key of R is the combination of A and K.
- If the multivalued attribute is composite, we include its simple components.
- **Example**
  - The relation DEPT_LOCATIONS is created.
  - DLOCATION represents the multivalued attribute LOCATIONS of DEPARTMENT, while DNUMBER as foreign key represents primary key of DEPARTMENT relation.
  - Primary key of R is the combination of {DNUMBER, DLOCATION}.

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.
- 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 next slide.
  - This can be mapped to relation SUPPLY shown in the relational schema, whose primary key is combination of three foreign keys {SNAME, PARTNO, PROJNAME}.

Mapping of N-ary Relationship Types

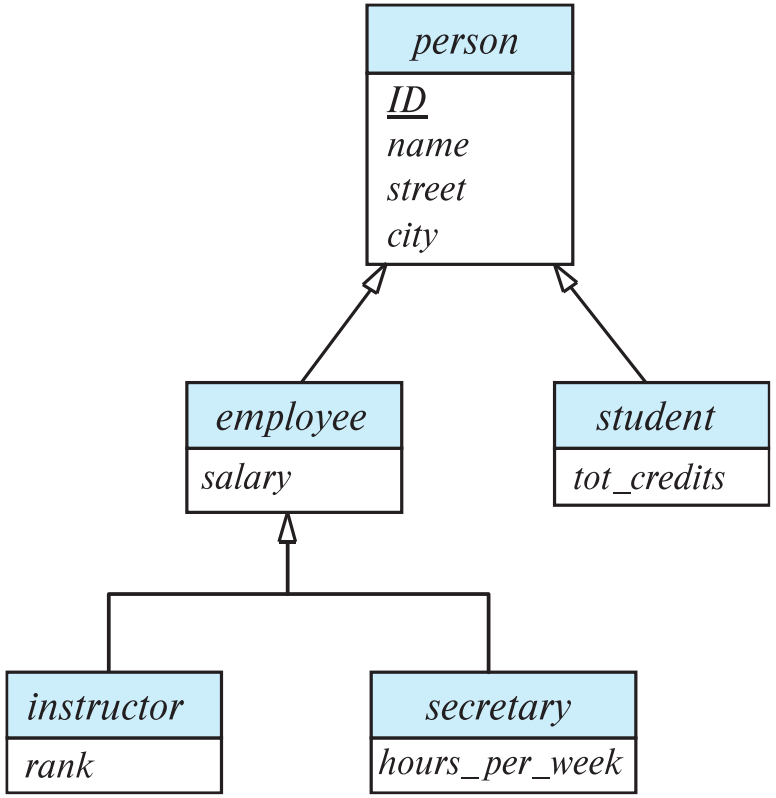




Correspondence between ER and Relational Models

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

Mapping of Specialization/Generalization



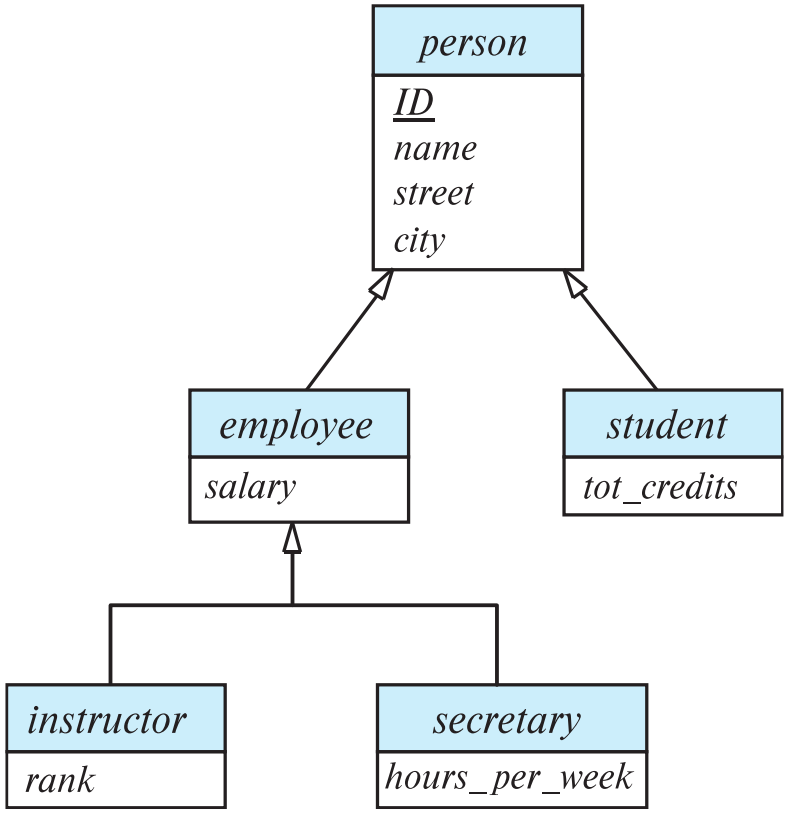
❑Method 1:

- Form a schema for higher-level entity
- Form a schema for each lower-level entity set, include primary key of higher-level entity set and local attributes

schema	attributes
person	ID, name, street, city
student	ID, tot_cred
employee	ID, salary

- Drawback: getting information about, an *employee* requires accessing two relations, one corresponding to low-level schema and one corresponding to high-level schema.

Mapping of Specialization/Generalization



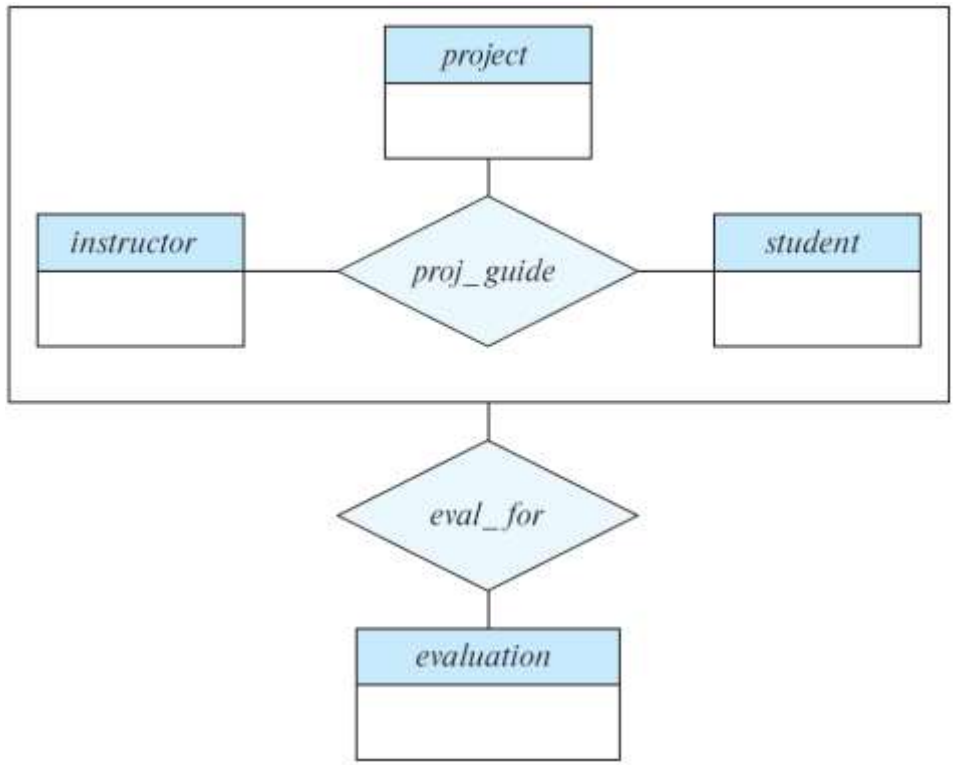
❑Method 2

- Form a schema for each entity set with all local and inherited attributes.

schema	attributes
person	ID, name, street, city
student	ID, name, street, city, tot_cred
employee	ID, name, street, city, salary

- Drawback: *name, street* and *city* may be stored redundantly for people who are both students and employees.

Mapping of Aggregation



- To represent aggregation, create a schema containing
  - Primary key of the aggregated relationship
  - Primary key of the associated entity set
  - Any descriptive attributes
- In our example:
  - Schema eval_for is:

    eval_for (s_ID, project_id, i_ID, evaluation_id)
  - The schema proj_guide is redundant.