

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

شروع اللہ کے پاک نام سے جو بڑا مہربان نہایت رحم والا ہے





Dr. Abid Sohail Bhutta

abidbhutta@cuilahore.edu.pk

**Department of Computer
Science,**

**COMSATS University , Lahore
Campus**



Database Systems



Lecture 18

Enhanced Entity Relationship Diagram (EERD) and Inheritance in Schema Modeling (Part 2)



Recall Lecture 17

- Enhanced Entity Relationship Diagram
 - Classification
 - Aggregation



Today's Lecture

- EER (Enhanced Entity Relationship)
 - Generalization
 - Specialization

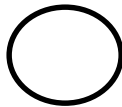

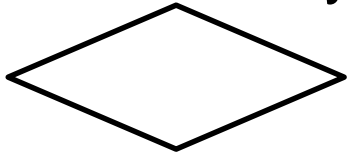


What is an EER Model?

- Enhanced Entity Relationship (EER) – Data Modeling
- EER shows complex relationships between objects in a database (multimedia, geographical).
- Concepts of subclasses and superclasses, specialization and generalizations.
- Put OOD Modeling concepts in ER diagram to form an EER model



Model Shapes

- When you have more than one subclass based on the same defining attribute (*JobType*), use

- To show class/subclass relationships, use

 - Used for relationships between entity types
- To show relationship between two different entity types, use


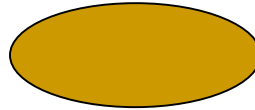


Model Shapes (Cont.)

- Entities,



- Attributes



- We classify entities according to their type:

- Strong Entity



Student



Lecture

- Weak Entity: Entity sets that are not uniquely identified by their attributes
 - A weak entity set has an "identifying relationship" with an entity set that is the "identifying owner" of the weak entity set



Dependents



Subclasses and Superclasses of Entity Types

EMPLOYEE
ENGINEER,



superclass
EMPLOYEE



SECRETARY,
MANAGER

subclass of entity type

superclass/subclass relationship
class/subclass
EMPLOYEE/ENGINEER

Each member of subclass is a member of its superclass

- Not all elements in superclass need to be in subclass



Type inheritance

- Subclass entities inherit attributes/relationships of superclass entity
- Own specific (local) attributes/relationships
 - Avoid describing similar concepts more than once
 - Add more semantic information to design



Specialisation

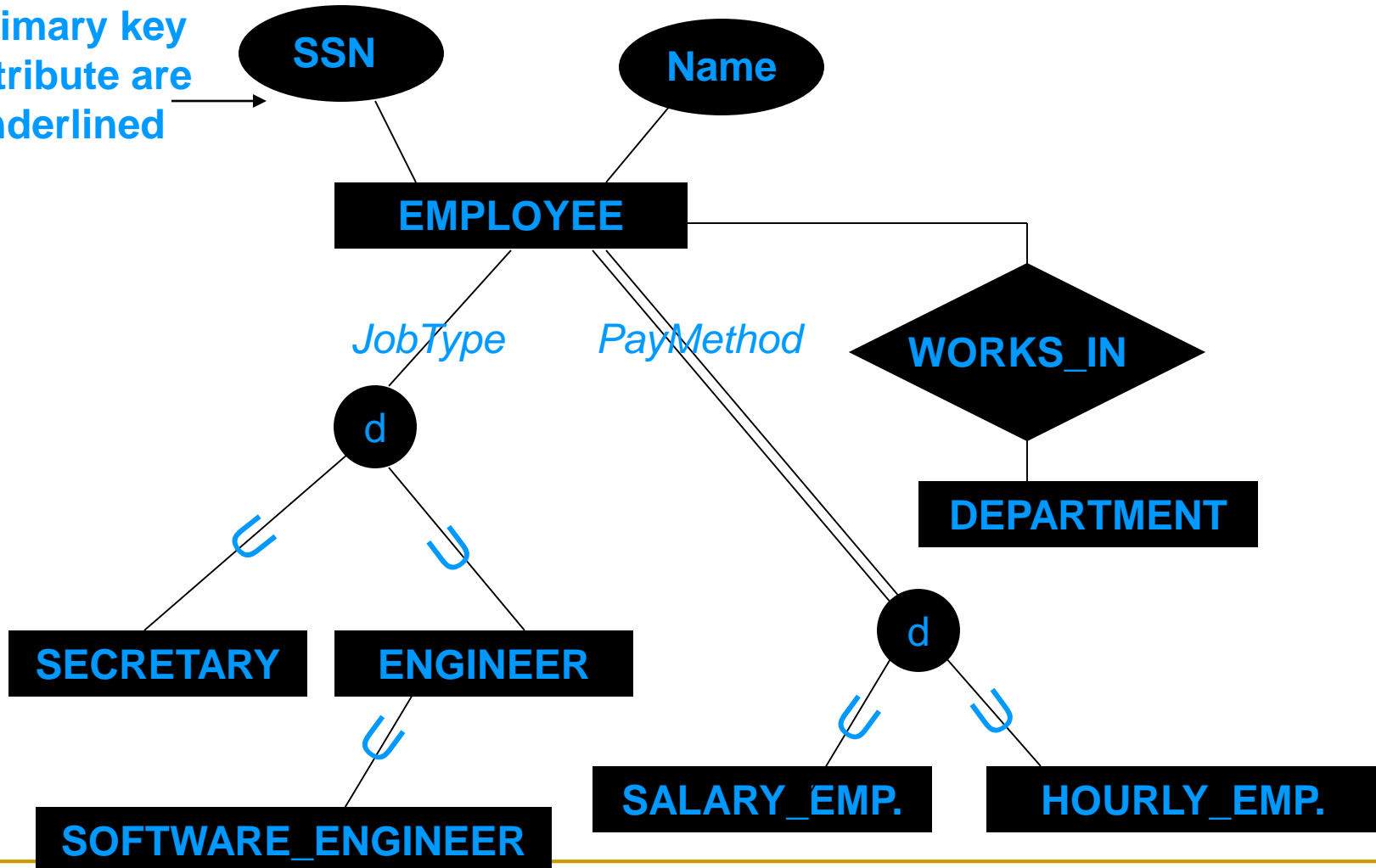
Subgrouping into subclasses (top-down approach)

- Example: EMPLOYEE -> SECRETARY
MANAGER, etc.
- **Inheritance** – Inherit attributes and relationships from superclass (Name, Birthdate, etc.)
- Subclasses may have unique attributes
 - SECRETARY has Typing Speed attribute, MANAGER has Business Unit Managed, etc.



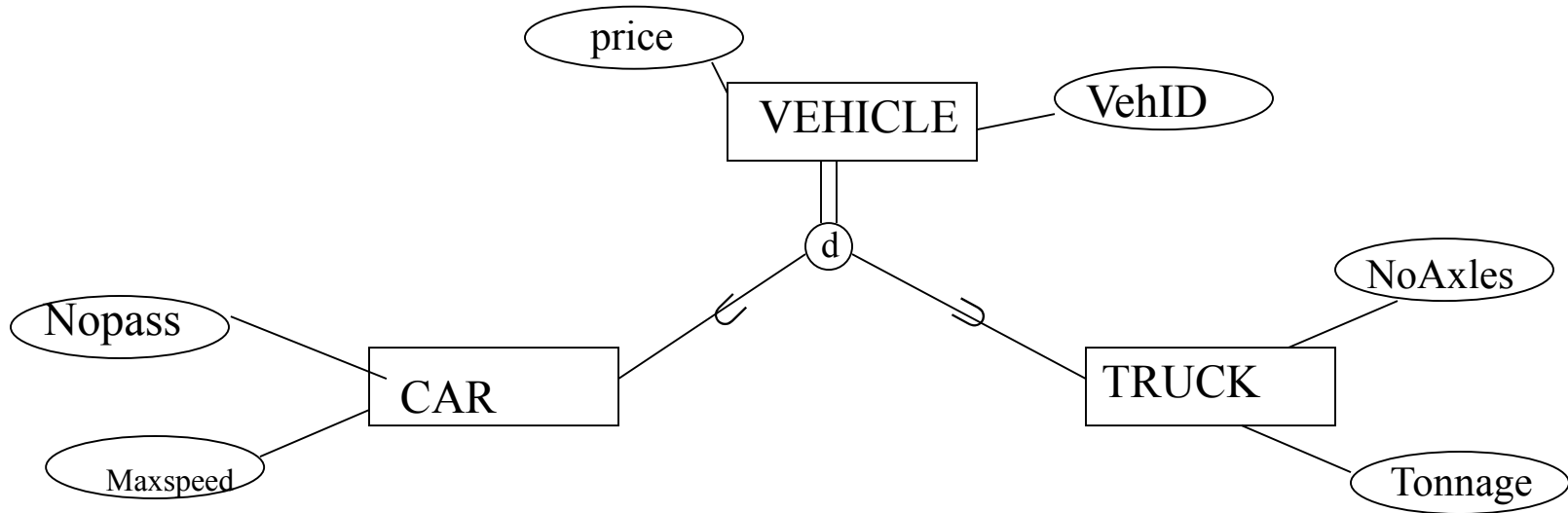
Specialisation (cont.)

Primary key
attribute are
underlined







Generalisation

- Reverse processes of defining subclasses (bottom-up approach)
- Bring together common attributes in entities
 - Example: CAR (with attributes color, Nopass, max speed) and Truck (with attributes price, NoAxles) can be generalised into VEHICLE (with attributes color and price).



Constraints (1)

- **Disjoint** – an entity can be a member of at most one subclass of a specialization 
- **Overlap** – an entity may belong to more than one subclass of a specialization 
- **Total specialisation** – each entity of a superclass belongs to some subclass of a specialisation 
- **Partial specialization** – each entity of a superclass does not have to belong to some subclass of a specialisation 

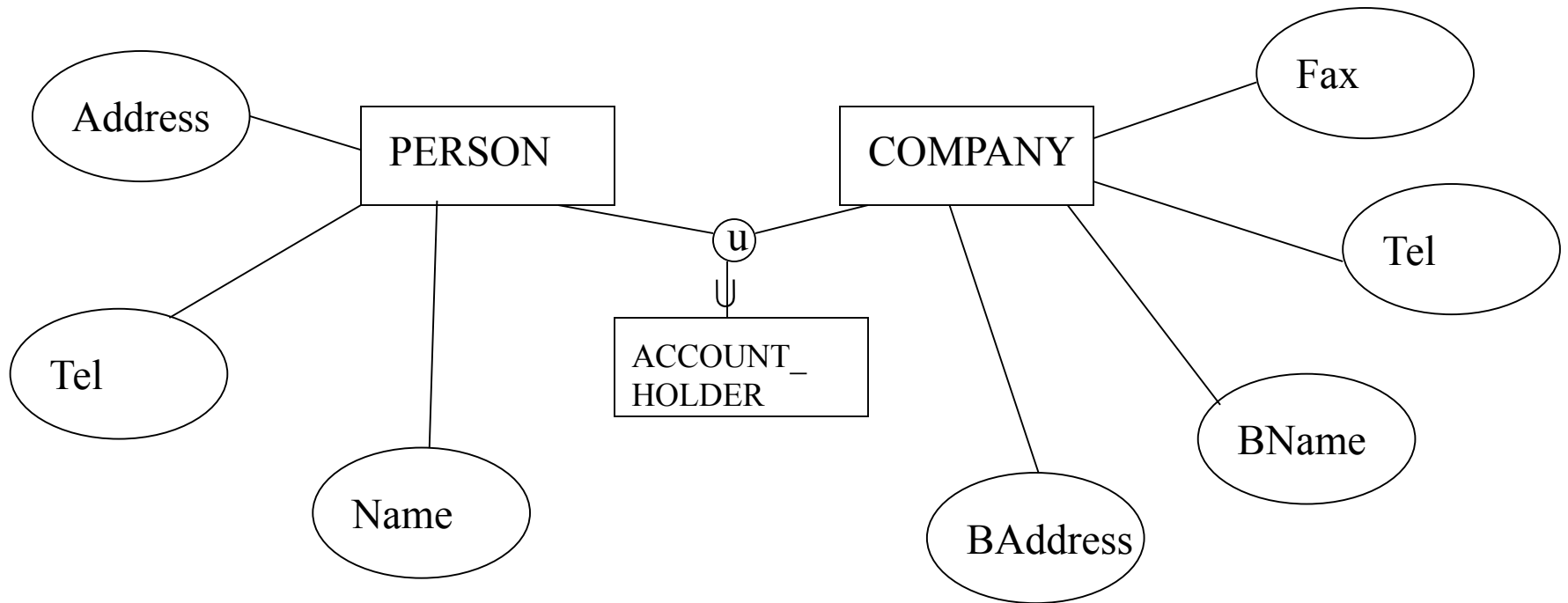


Constraints on Specialization and Generalization (2)

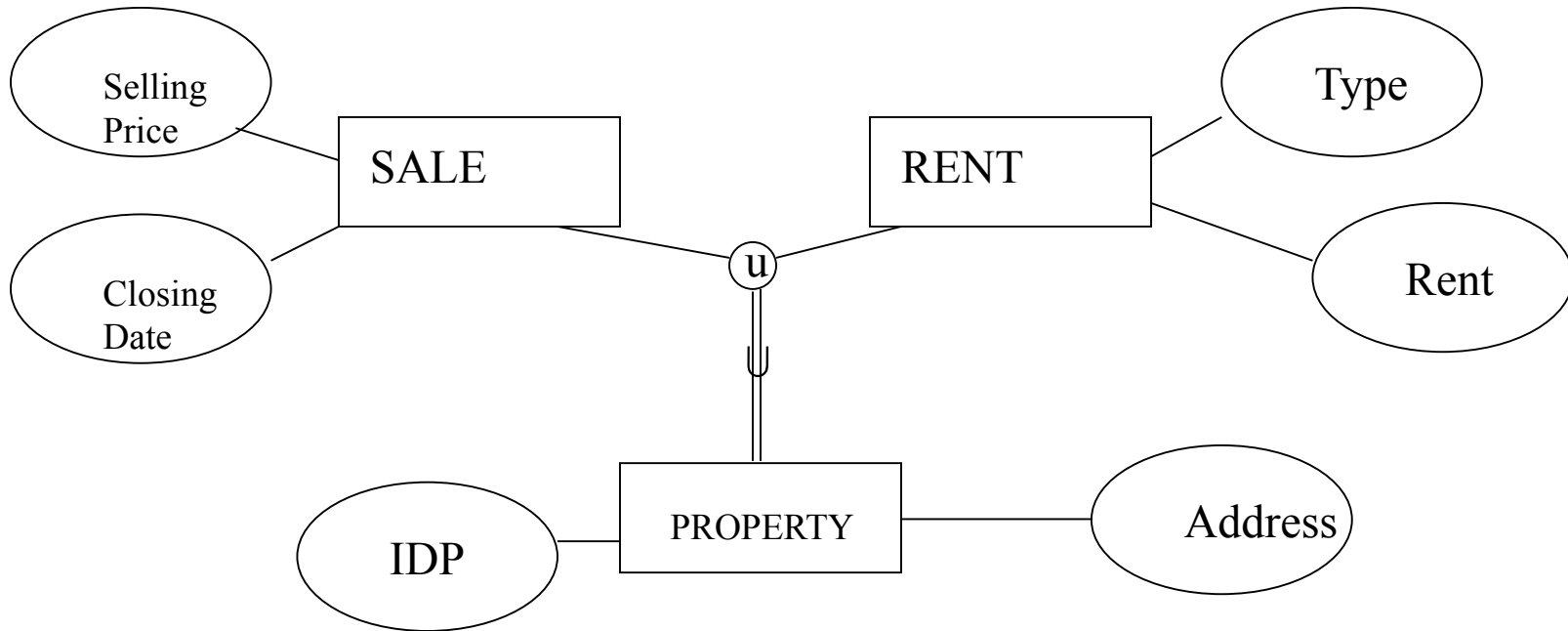
- Hence, we have four types of specialization/generalization:
 - Disjoint, total
 - Disjoint, partial
 - Overlapping, total
 - Overlapping, partial
- Note: Generalization usually is total because the superclass is derived from the subclasses.



Partial category



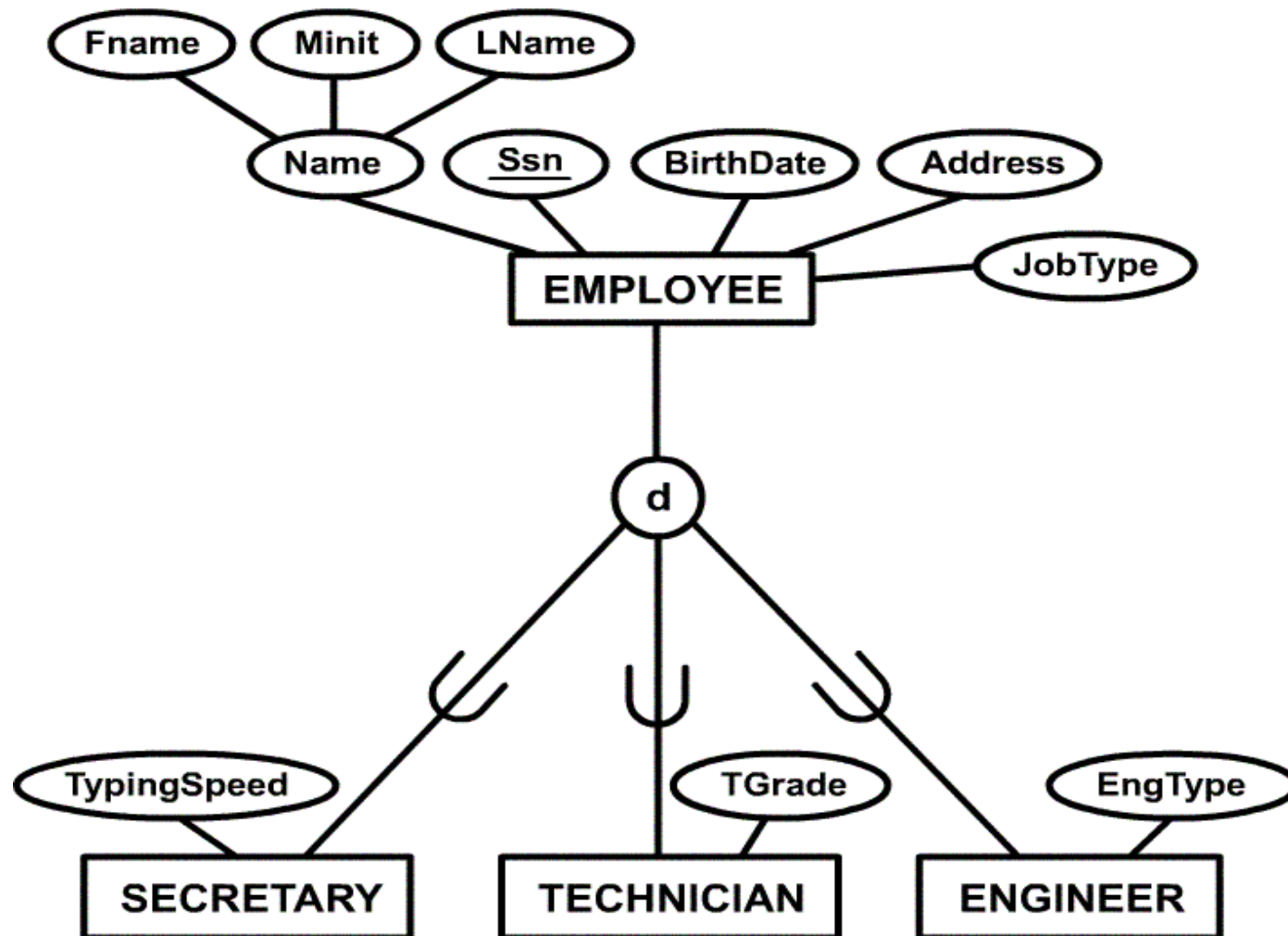
Total category



- Every occurrence of superclass must appear in the category
- Better represented as **specialisation** (share many attributes and keys)

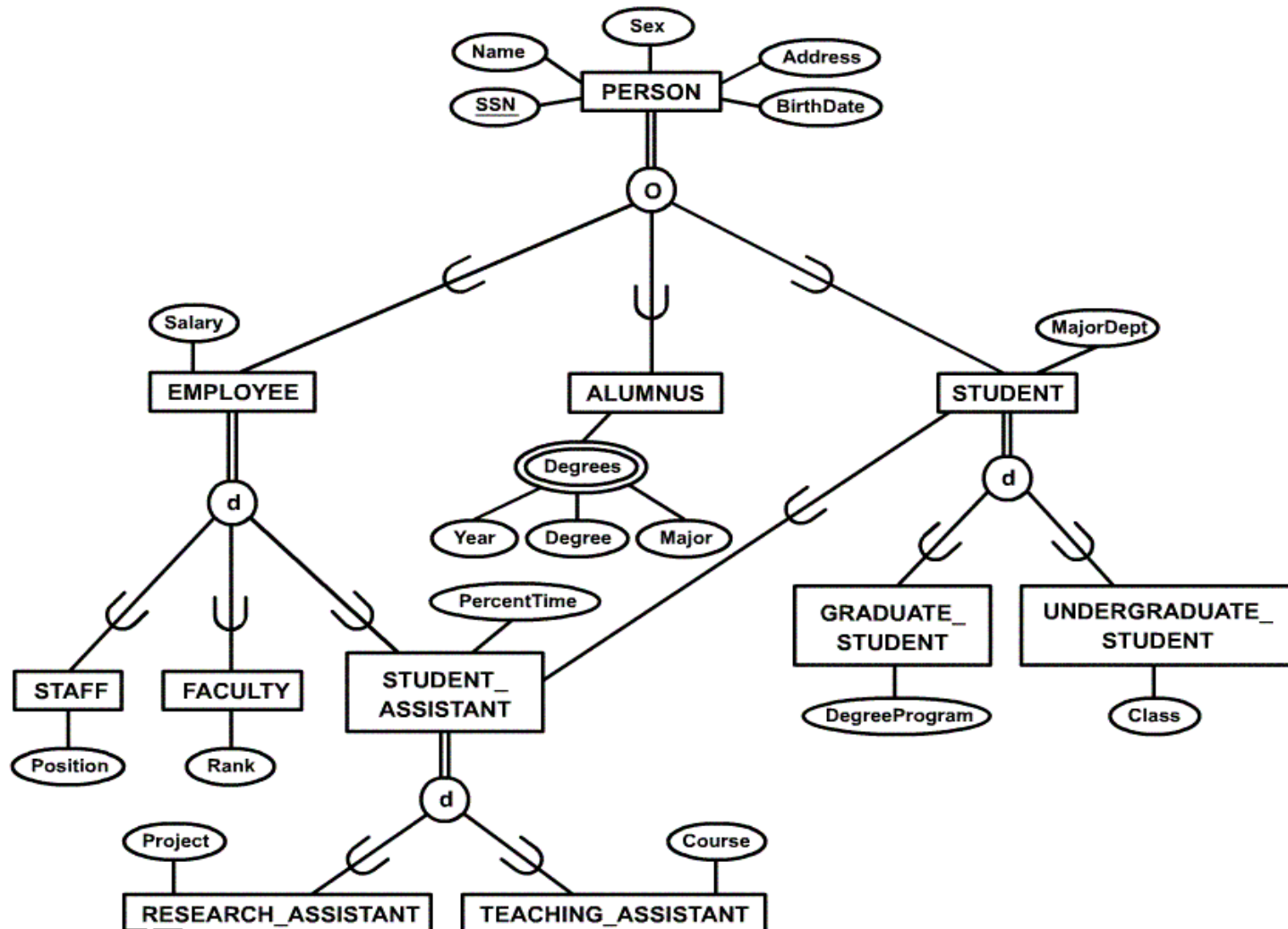


Example of disjoint partial Specialization



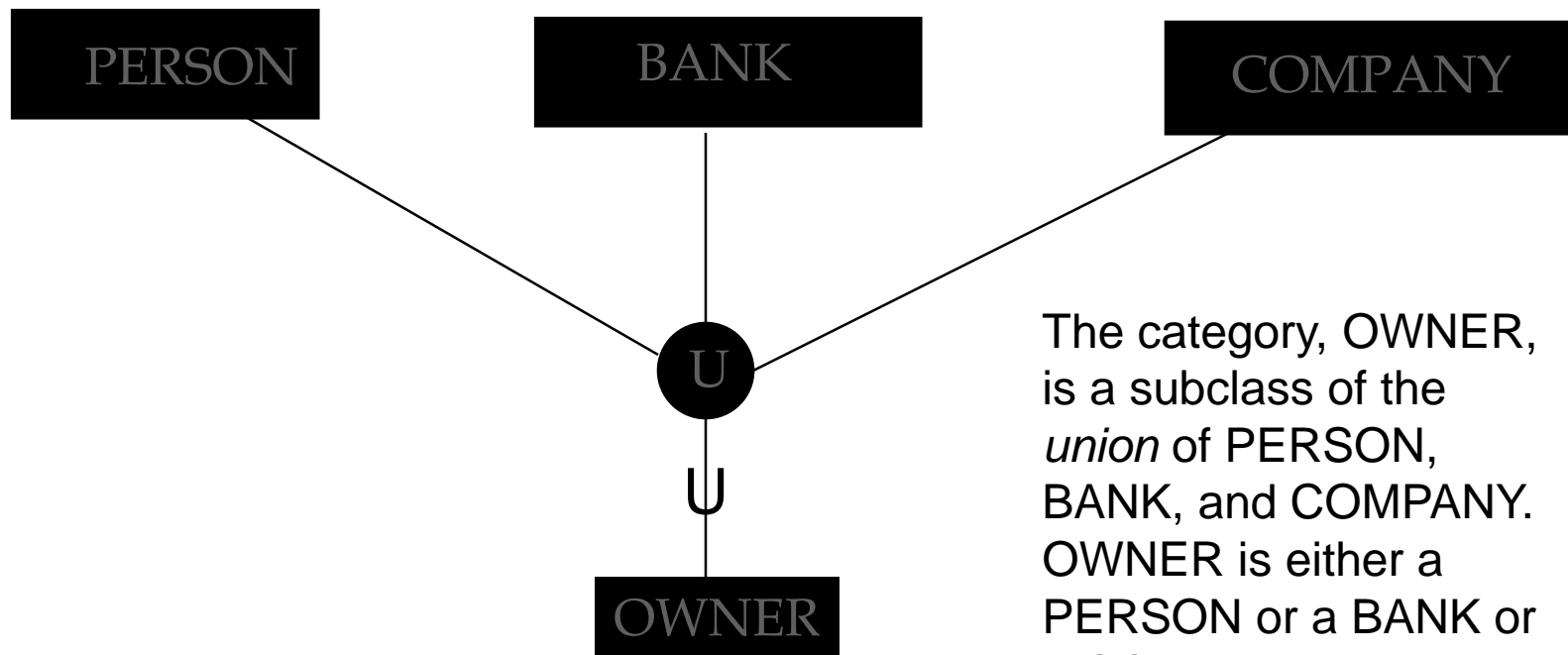
Specialization / Generalization Lattice

Example (UNIVERSITY)



Union Types/Categories

- Models a class/subclass with more than one superclass of *distinct* entity types. Attribute inheritance is selective.



The category, OWNER, is a subclass of the *union* of PERSON, BANK, and COMPANY. OWNER is either a PERSON or a BANK or a COMPANY

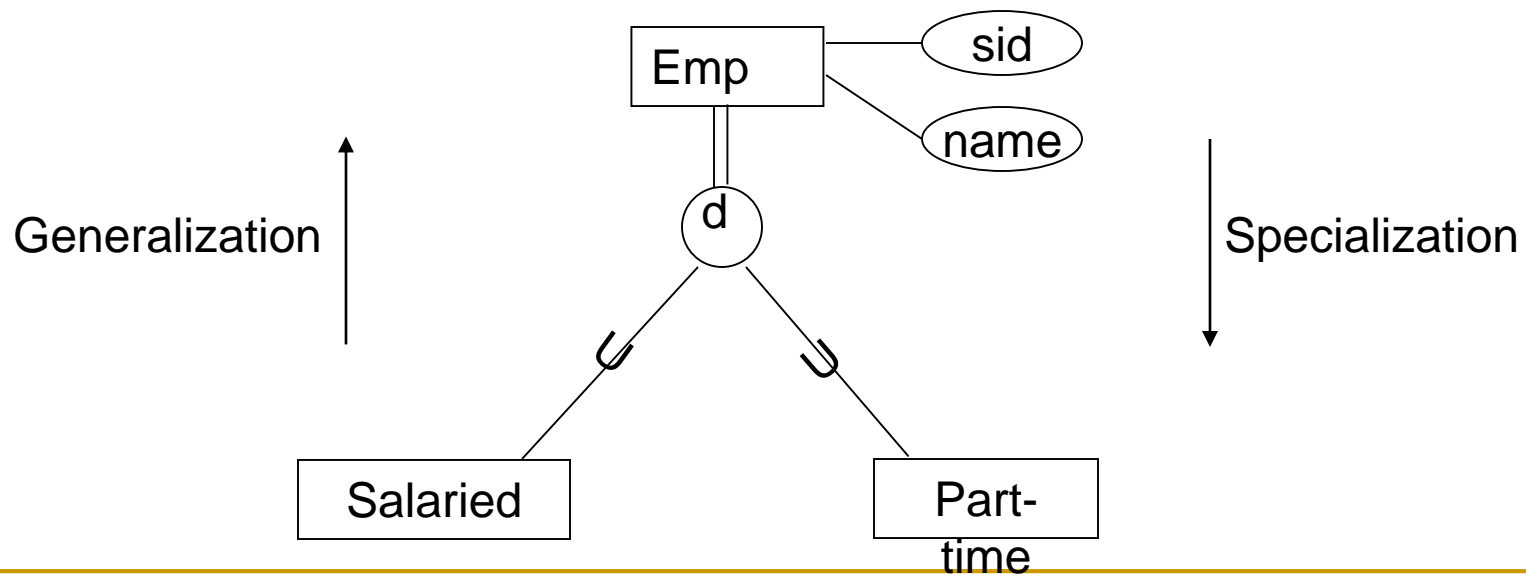


More understanding of generalization/specialization

- A design may require all members of an entity-set to be specialized. For example, an employee MUST be a member of either a Salaried or Part-time. Use double lines to dictate this constraint

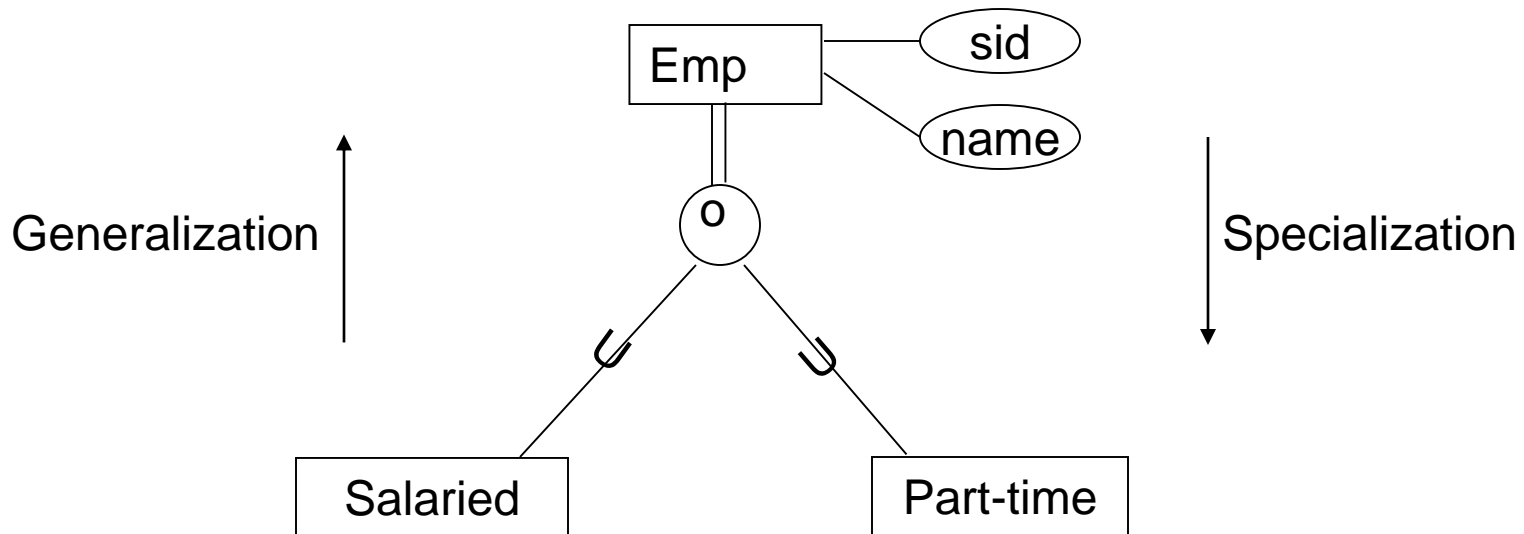


GENERALIZATION AND SPECIALIZATION



GENERALIZATION AND SPECIALIZATION

- One may allow the specialized entity sets to overlap. For example, an entity might be both a Salaried and Part-time. “o” stands for Overlap when specializing.



Conceptual Object Modeling

- Object Modeling Methodologies
 - in software engineering: UML, OMT, ...
 - specify *operations* applied to objects (entities)
 - class = entity type (attributes) + operations
- Object Modeling Concepts (vs. EER)
 - class: entity type, object: entity
 - association (link): relationship (rel. instance)
 - multiplicity: relationship constraint



generalization / specialization

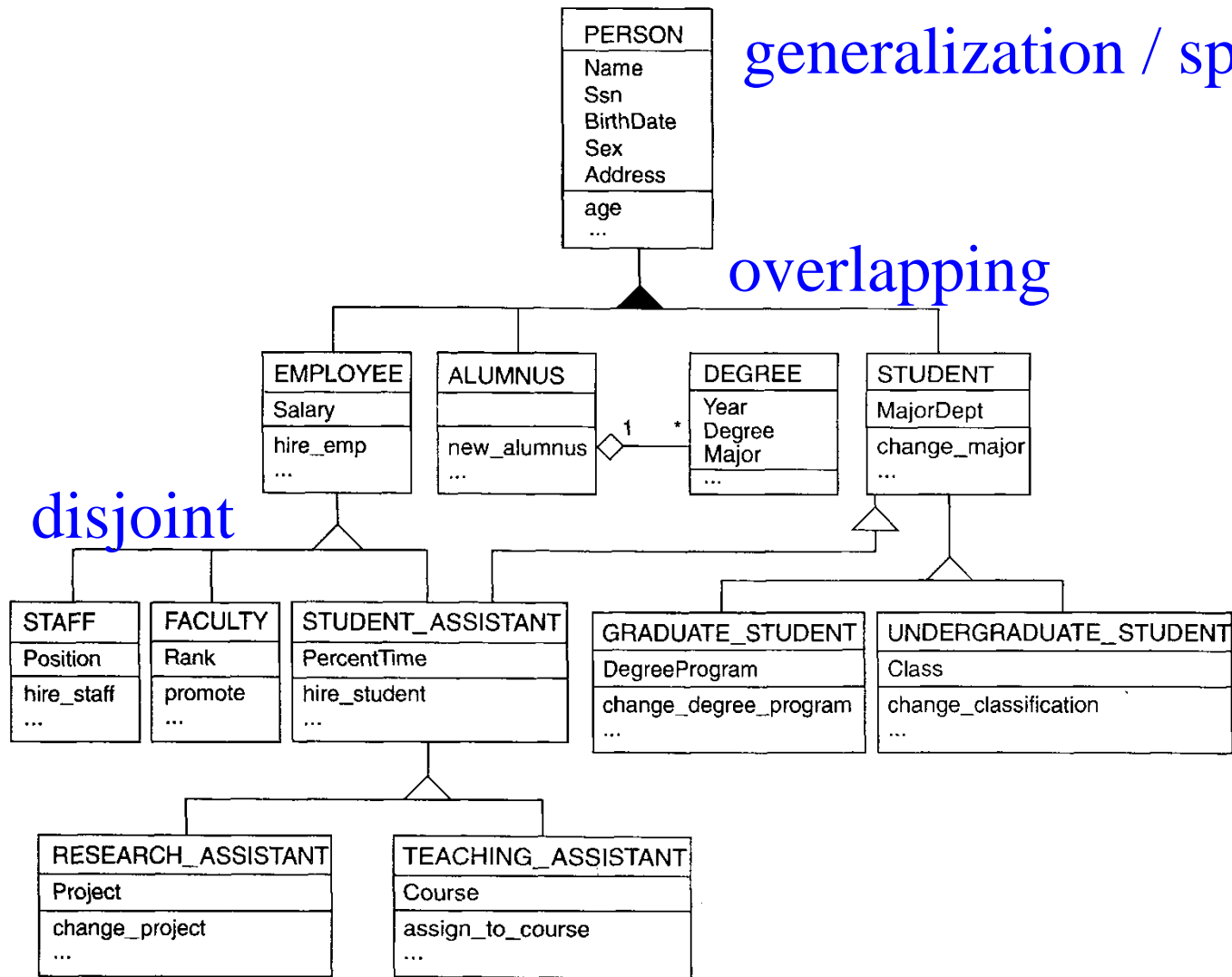


Figure 4.12 Specialization/generalization notation in UML shown by a class diagram corresponding to the EER diagram in Figure 4.7.

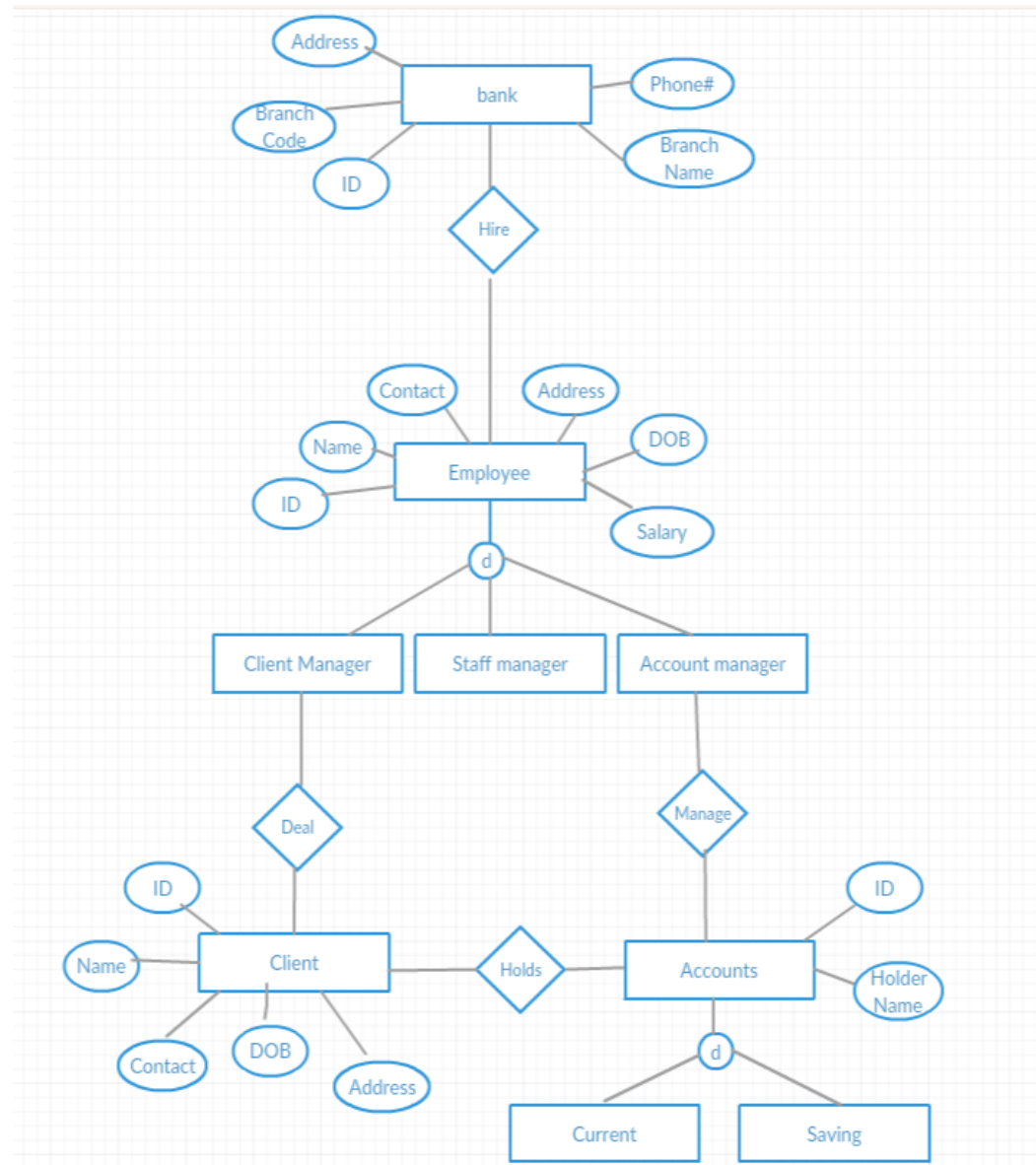


Practical Assignment -1

- In a public sector bank different types of employees like client managers , staff managers and account managers are working. The client managers are managing the details of every client for Customer relation management while account managers holds the accounts information for the both current and saving account types.



Solution



Home Assignment

- Draw an EER of CUONLINE



In Next Lecture

- Database Term Project
- Schema refinement / Normalization



Thanks

