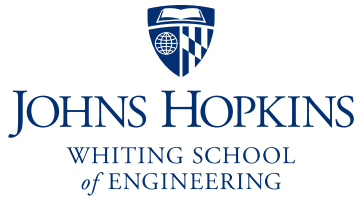
Johns Hopkins Engineering

Principles of Database Systems

Module #4

Enhanced ER (EER) Modeling



Enhanced ER Modeling Introduction

- Additional semantic data modeling with additional symbols in order to describe more complex business requirements.
- There is no standard for EERD.
- Commercial database design tools do not support EERD.

Enhanced ER Modeling Introduction (cont.)

- Most EERDs can be represented and implemented with regular ERDs with relational mappings.

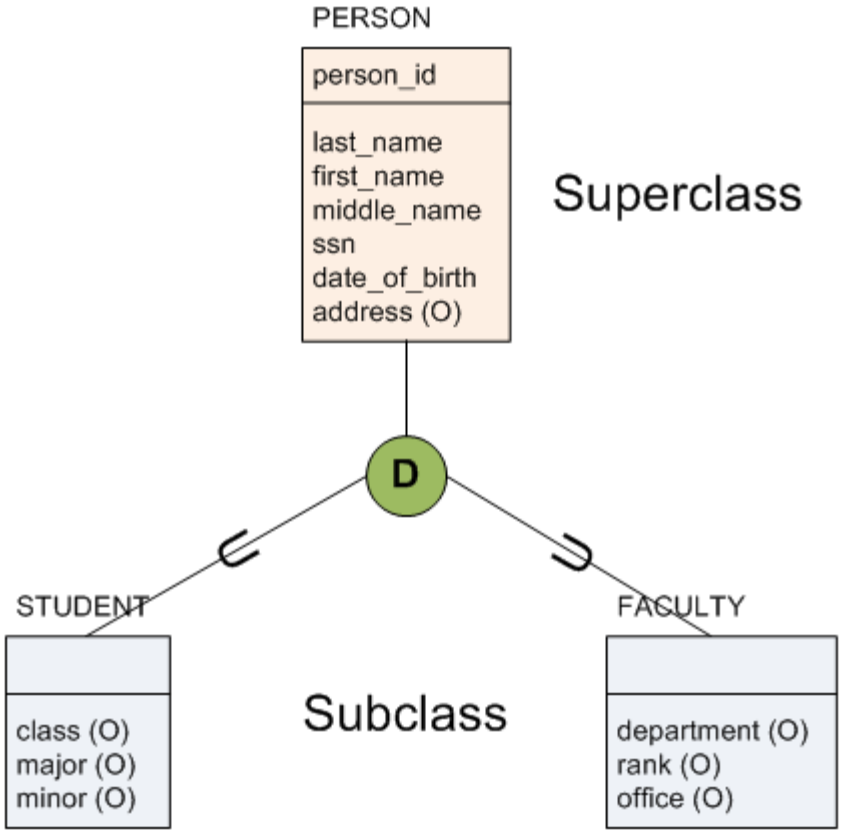
- There are challenges to implementing EERD.
  - Database design tools do not support EERD.
  - There are multiple ways to implement EERD.
    - Efficiency for implementation
    - Performance
    - Business rules

Superclasses and Subclasses

- Superclass
  - An entity type requires representation in a data model.
  - It includes one or more distinct subgroupings of its occurrences, which require representation in a different data model.

Example: The University ERD
PERSON entity contains common attributes for a person as a superclass.
FACULTY entity contains special attributes for a faculty instance as a subclass.
STUDENT entity contains special attributes for a student instance and a subclass.

Superclasses and Subclasses (cont.)



Superclass
- PERSON

Subclass
- STUDENT
- FACULTY

1:1 relationship between superclass and subclass

Superclasses and Subclasses (cont.)

- Superclass
  - Distinct subgroupings of an entity type require representation in a data model.
  - An entity instance cannot exist in the subclass without a corresponding entity instance existing in its superclass.
  - The subclass has additional attributes of interest.
  - Potential subclasses represent variations of a similar concept.
  - All potential subclasses have the same attribute that can be factored out or expressed in the superclass.

Superclasses and Subclasses (cont.)

- Subclass
  - Each member of a subclass is also a member of the superclass. In other words, the entity in the subclass is the same entity in the superclass, but with a distinct role.
  - The relationship between the superclass and the subclass is **1:1** and is called superclass/subclass relationship.
  - Not every entity (instance or member) of a superclass needs be an entity of a subclass.

Superclasses and Subclasses (cont.)

- Examples

  Example 1: An OWNER entity type may be classified as ORGANIZATION or PERSON. Each entity type has distinct attributes.

  Example 2: An EMPLOYEE entity type may be classified as MANAGER, ENGINEER, SALEPERSON, TECHNICIAN, and SECRETARY. Each entity type has distinct attributes.

Superclasses and Subclasses (cont.)

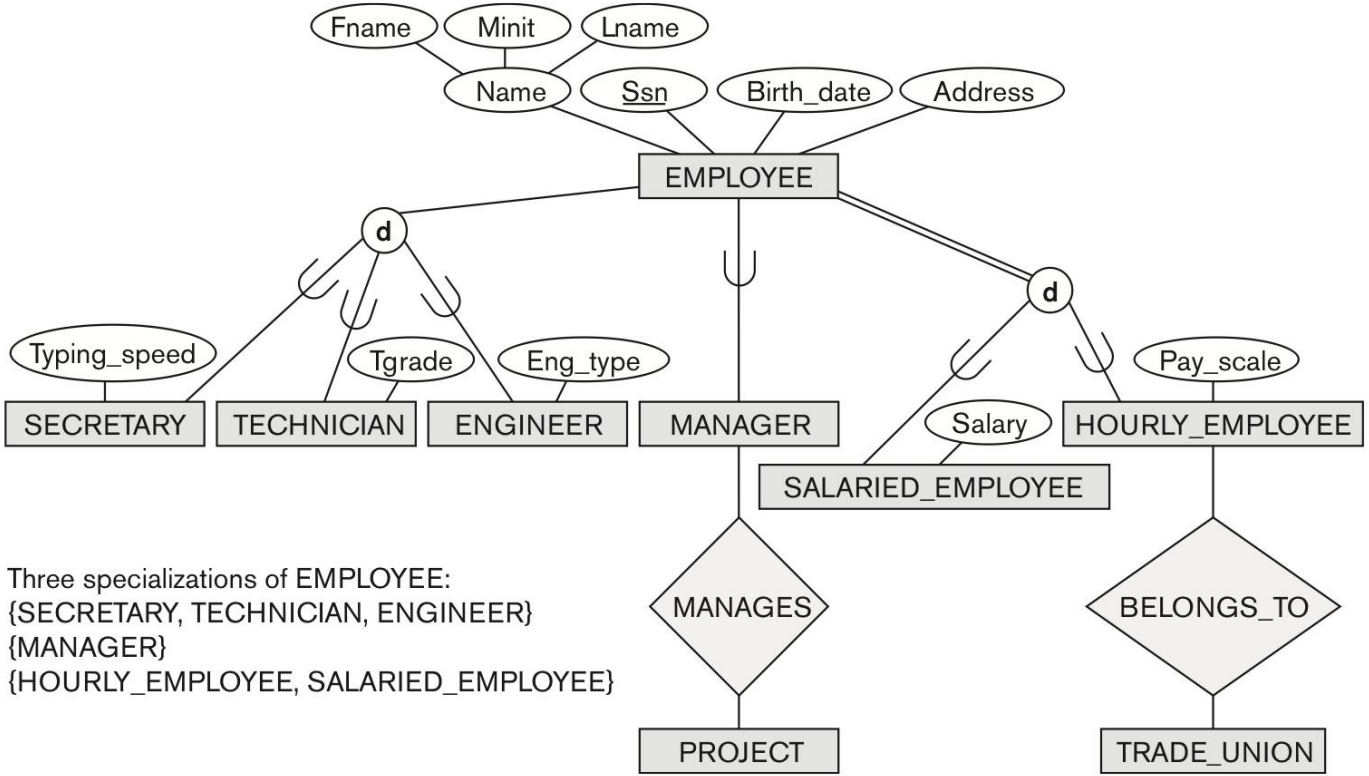


Figure 4.1 EER diagram notation to represent subclasses and specialization.

Superclasses and Subclasses (cont.)

- A superclass may include more than one subclass (e.g., an employee is an engineer and has full-time status.)

- In Object-Oriented Concept:
  - A subclass inherits all the attributes of its superclass (type inheritance).
  - A superclass/subclass relationship is often called an **IS-A** relationship or hierarchy. For example, a Manager IS-A member of Staff or Employee.

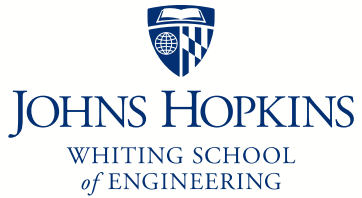
- Comparison between EER and Object-Oriented (OO) designs.

Johns Hopkins Engineering

Principles of Database Systems

Module 4 / Lecture 2

Enhanced ER (EER) Modeling



Specialization and Generalization

- Specialization: The process of maximizing the differences between members of an entity by identifying their unique characteristics.

- Specialization is a top-down approach to defining a set of superclasses and their related subclasses. Like a process of defining a supertype and (a one or more) subtypes with proper relationships.

Specialization and Generalization (cont.)

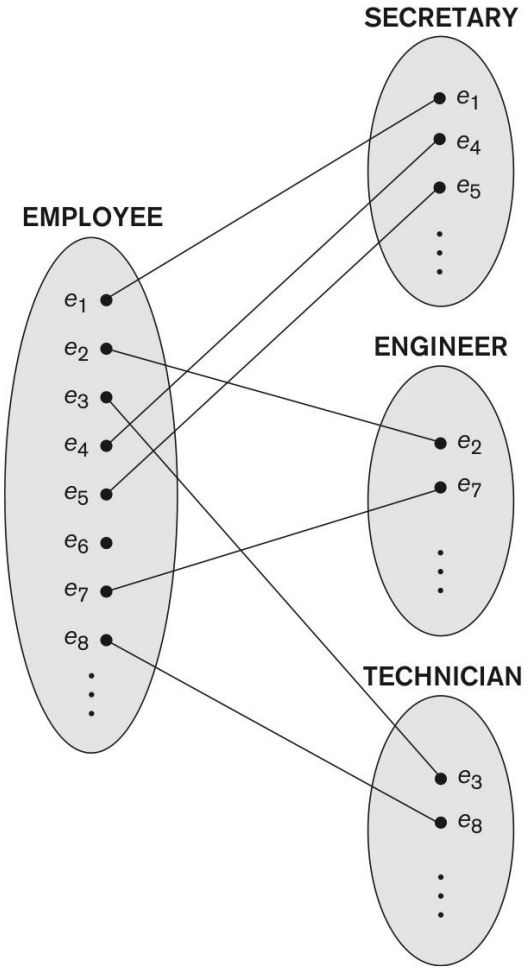


Figure 4.2
Instances of a specialization.

Specialization, and Generalization (cont.)

- Generalization: The process of minimizing the differences between entities by identifying their common characteristics.
- Generalization is a bottom-up approach, which results in the identification of a generalized superclass from the original entity types. The same process can be used to define supertype and subtypes and their relationships.

Specialization and Generalization (cont.)

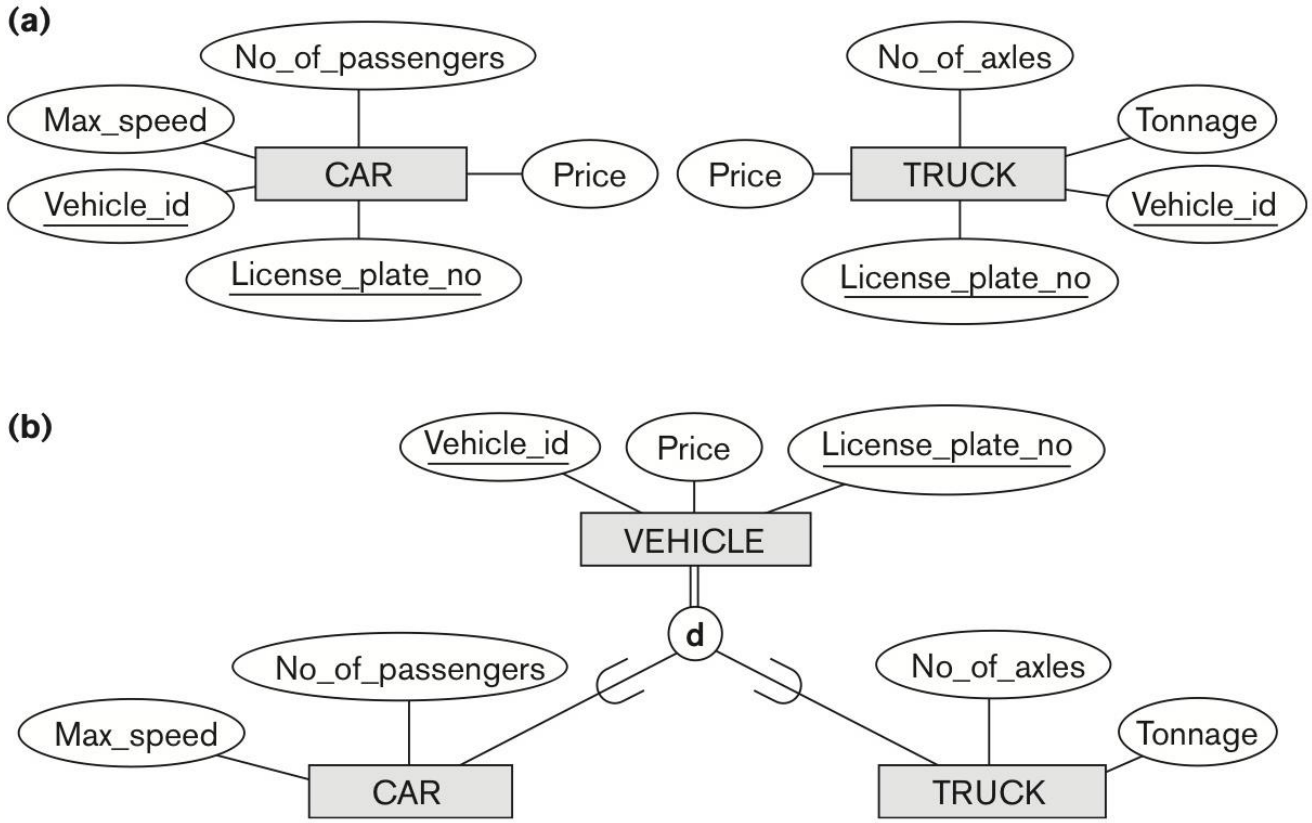


Figure 4.3
Generalization.
(a) Two entity types, CAR and TRUCK.
(b) Generalizing CAR and TRUCK into the superclass VEHICLE.

Constraints and Characteristics of Specialization and Generalization

- Determine a condition on the value of an attribute of the superclass to identify the participations on its subclasses.

- Disjointness constraint: the subclasses of specialization must disjoint. An entity can be a member of, at most, one of the subclasses of the specialization. The **d** in a circle stands for disjoint.

Constraints and Characteristics of Specialization and Generalization (cont.)

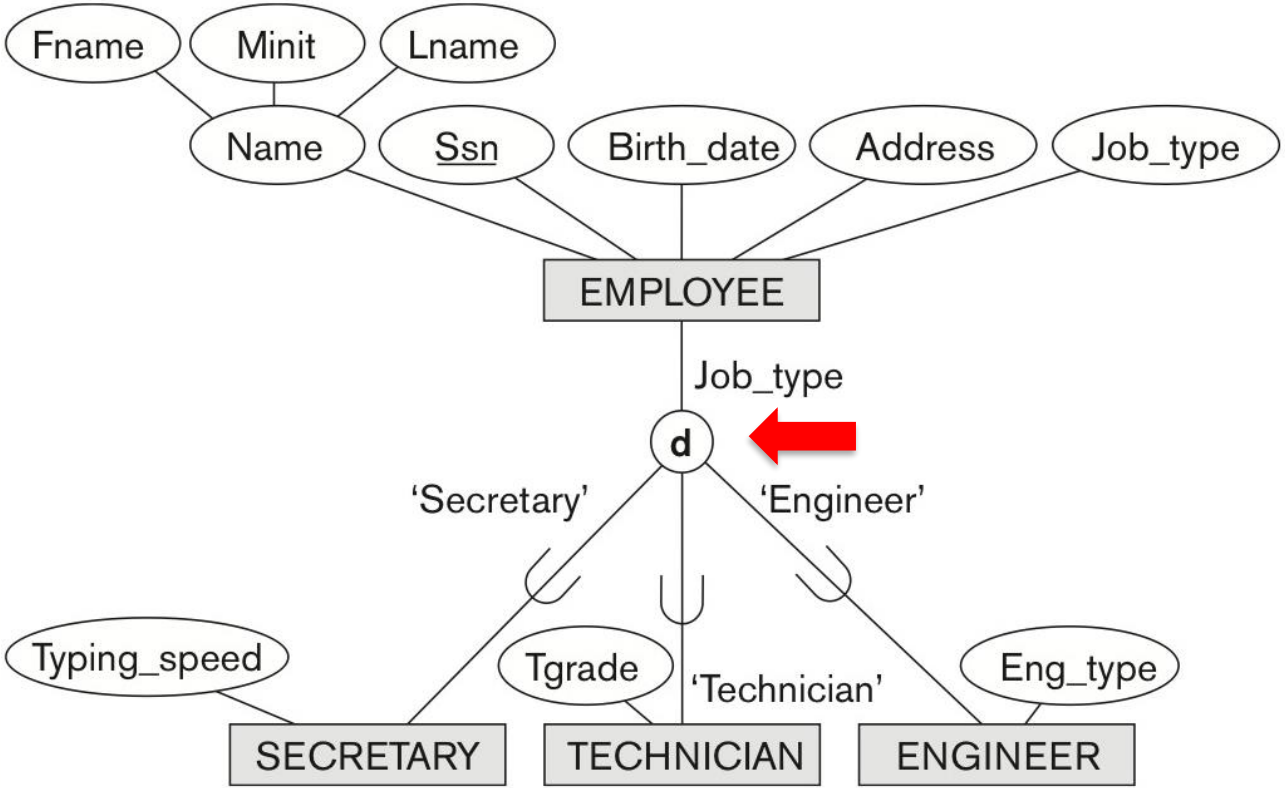


Figure 4.4 EER diagram notation for an attribute-defined specialization on Job_type.

Constraints and Characteristics of Specialization and Generalization (cont.)

- Overlap: If the subclasses are not constrained to be disjoint, their sets of entities may overlap. The entity in a superclass may have a member in more than one subclass of specialization. An **o** in a circle stands for overlap.

Constraints and Characteristics of Specialization and Generalization (cont.)

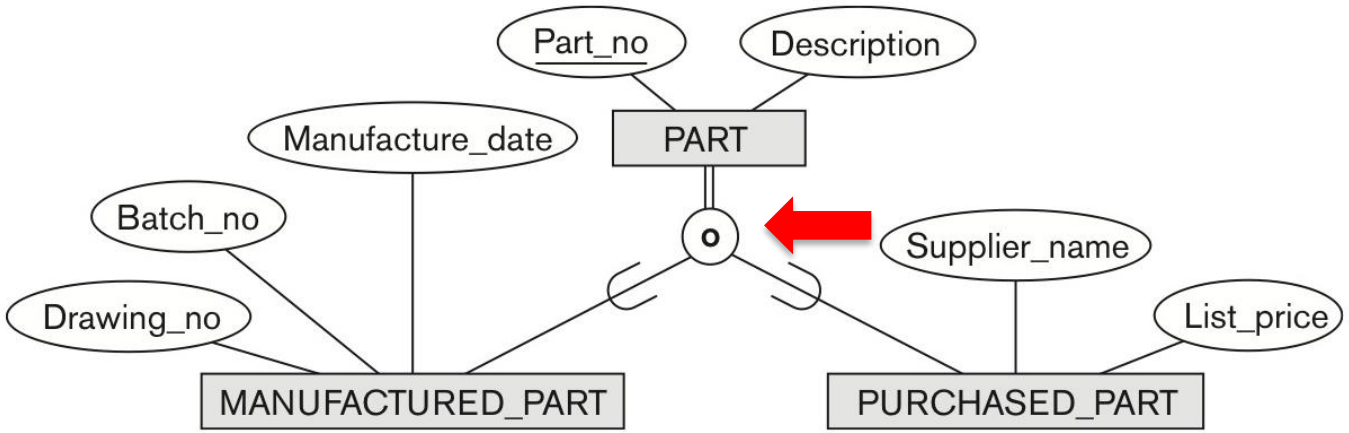


Figure 4.5 EER diagram notation for an overlapping (nondisjoint) specialization.

Constraints and Characteristics of Specialization and Generalization (cont.)

- Completeness constraint: A total specialization constraint specifies every entity in the superclass must be a member of at least one subclass in the specialization.

  Example: An EMPLOYEE entity has to be either an HOURLY_EMPLOYEE or SALARY_EMPLOYEE entity types. An entity can be a member of, at most, one of the subclasses of the specialization. A *double line connects the superclass to the circle*.

Constraints and Characteristics of Specialization and Generalization (cont.)

- A partial specialization allows an entity not to belong to any of the subclasses.

  Example: An employee entity may not be a secretary, technician, or engineer.

- A specialization hierarchy, has the constraint that every subclass participates as a subclass in only one class/subclass relationship.

Constraints and Characteristics of Specialization and Generalization (cont.)

- A specialization lattice allows a subclass to have more than one class/subclass relationship.

  Example: an Engineering_Manager is manager, engineer, and full-time employee, a union of three superclasses

Constraints and Characteristics of Specialization and Generalization (cont.)

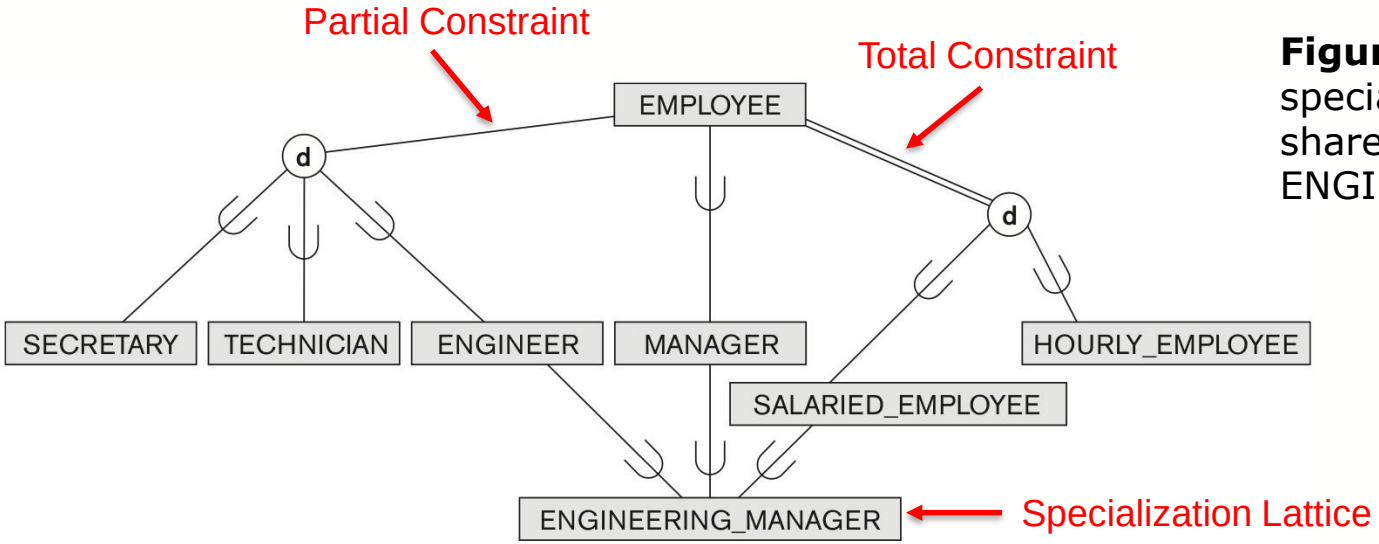


Figure 4.6 A specialization lattice with shared subclass ENGINEERING_MANAGER.

Constraints and Characteristics of Specialization and Generalization (cont.)

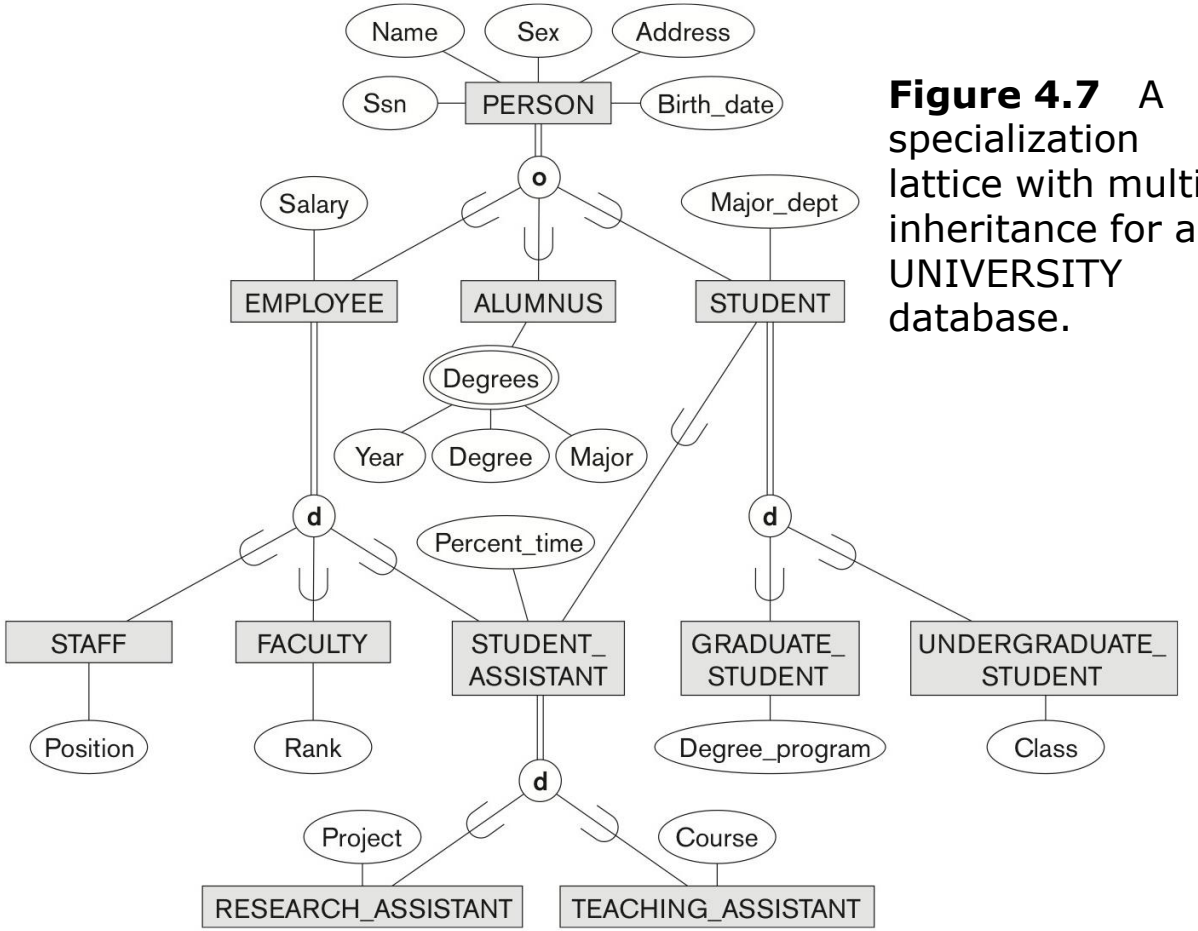


Figure 4.7 A specialization lattice with multiple inheritance for a UNIVERSITY database.

ohns Hopkins Engineering

Principles of Database Systems

Module 4 / Lecture 3

Enhanced ER (EER) Modeling

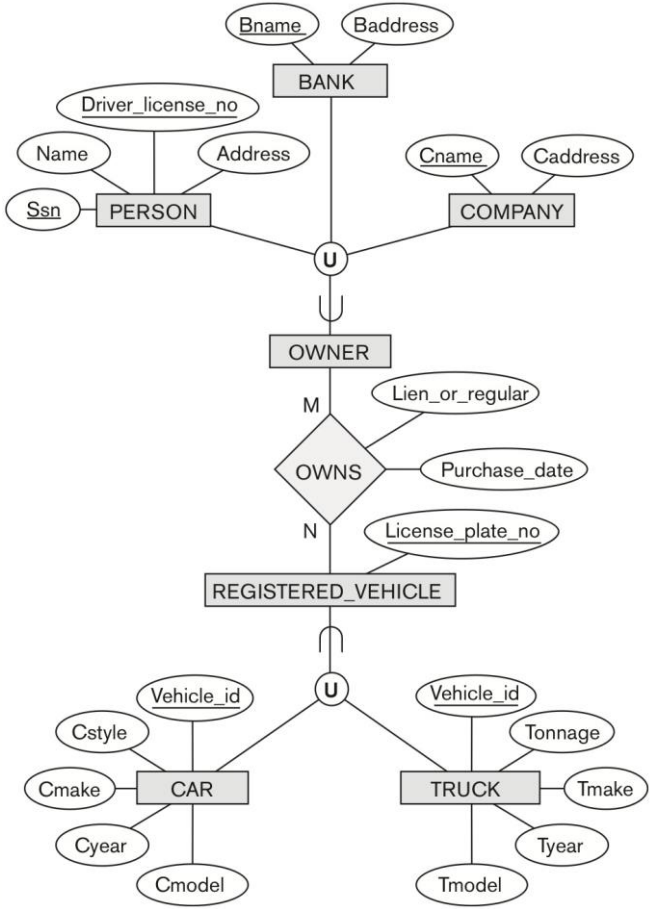
JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Modeling with Union Types Using Categories

- A union type subclass is a collection of objects that is a union of a distinct entity types.
- An instance of a subclass links to only one instance of one entity type.

Example:
An owner of a registered vehicle can be a PERSON, a BANK, or a COMPANY.
A registered vehicle can be a CAR or a TRUCK.

Modeling with Union Types Using Categories (cont.)



Superclass:
PERSON, BANK, COMPANY
Subclass:
OWNER

Figure 4.8 Two categories (union types): OWNER and REGISTERED_VEHICLE

Superclass:
CAR, TRUCK
Subclass:
REGISTERED_VEHICLE

An Example: UNIVERISTY EERD Using Chen's Notation

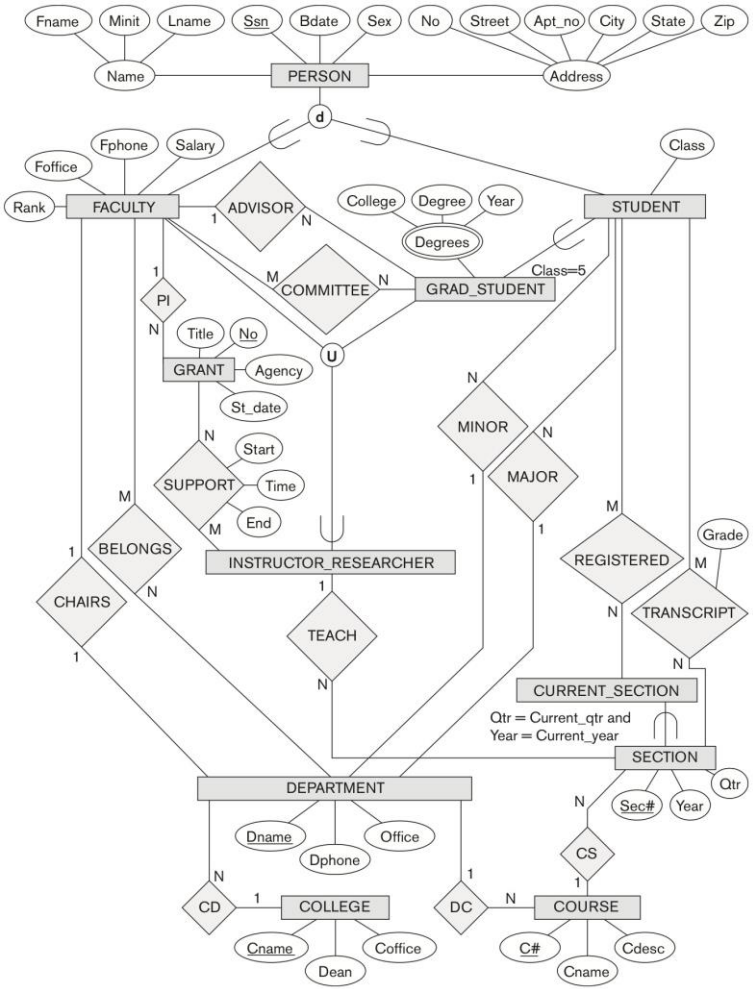
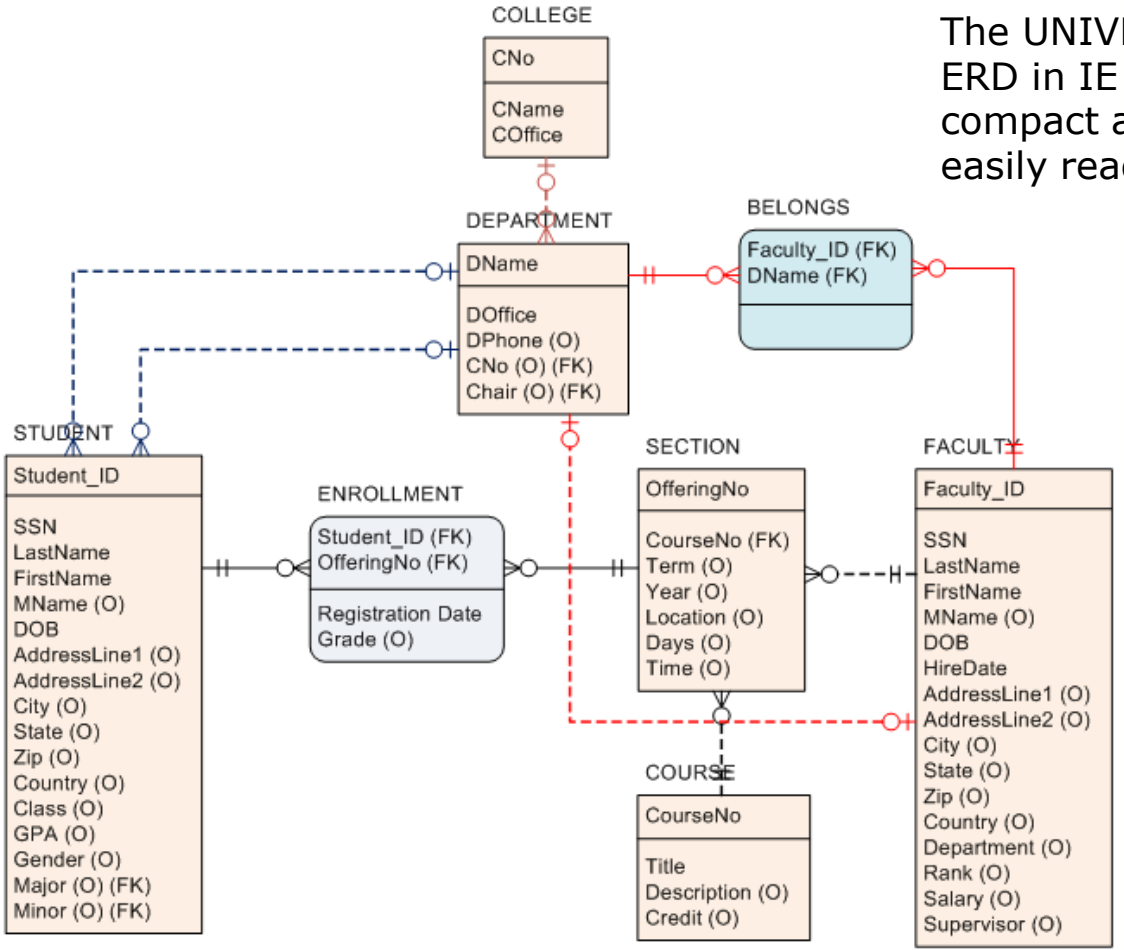


Figure 4.9 An EER conceptual schema for a different UNIVERSITY database.

Chen's notation vs. IE notation

EERD and ERD compare and contrast

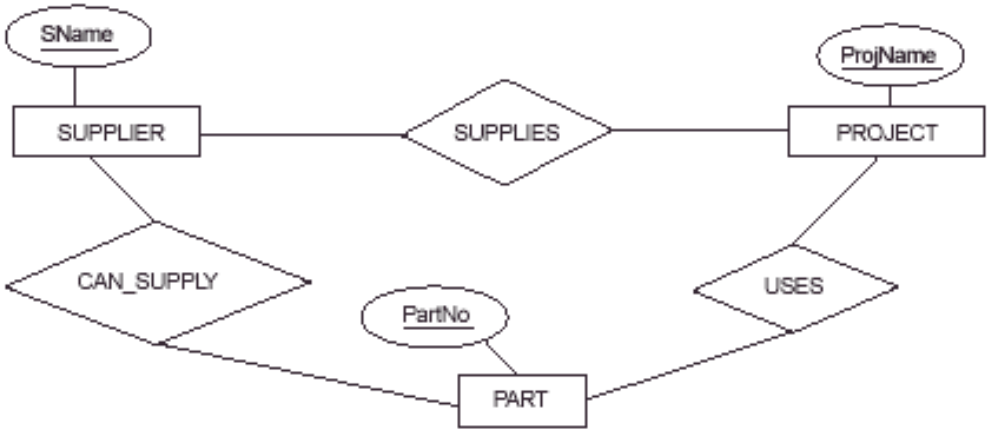
An Example: UNIVERISTY ERD Using IE Notation



The UNIVERISTY ERD in IE is compact and easily read.

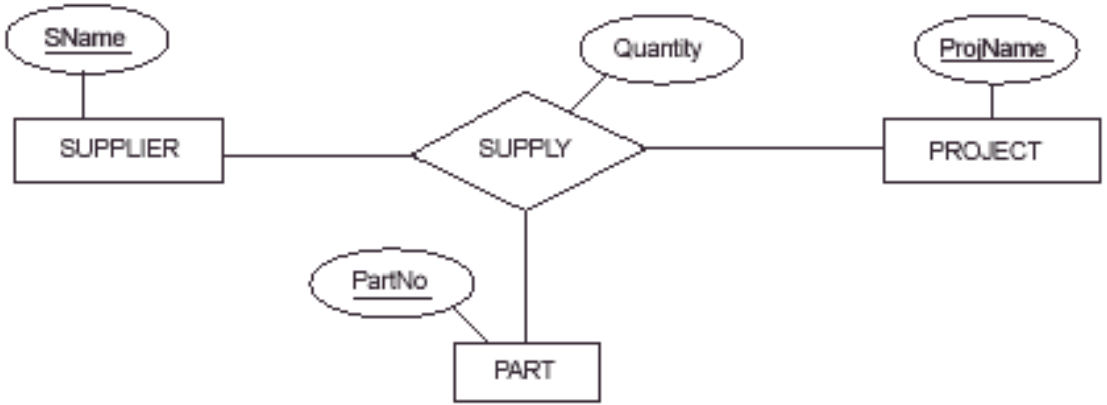
Relationship Types with Higher Degrees

- The relationships among SUPPLIER, PROJECT, and PART:
  - Three binary M:N relationships



Relationship Types with Higher Degrees (cont.)

- The relationships among SUPPLIER, PROJECT, and PART:
  - One ternary relationship

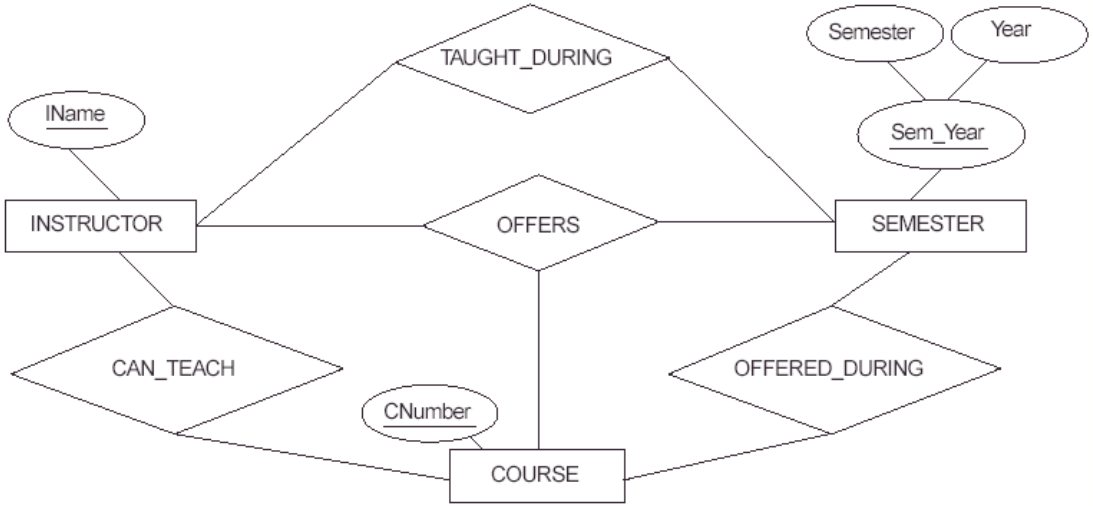


Relationship Types with Higher Degrees (cont.)

- Three binary M:N relationships vs. a ternary relationship:
  - Which one you use depends on the business requirements.
- Different relationships represent different information.
  - Do not create all relationships if business requirements don’t specify them.

Relationship Types with Higher Degrees (cont.)

- The relationships among INSTRUCTOR, COURSE, and SEMESTER:
  - Three binary relationships or a ternary relationship.



Johns Hopkins Engineering

Principles of Database Systems

Module 4 / Lecture 4

Enhanced ER (EER) Modeling

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Unified Modeling Language Class Diagram

- Unified Modeling Language (UML)
  - A set of graphical notations are used for business modeling and software design.
  - Class diagrams include classes, object features (attributes and methods) and associations among classes.
    - Class: A class defines both the structural attributes and behavioral features (methods or operations).
    - Class Association: A relationship between classes.
      - Binary: Simple association between two classes
      - N-ary: An association between multiple classes

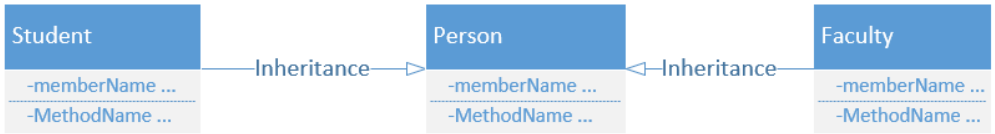
Unified Modeling Language Class Diagram (cont.)

- Class Association
  - A family of links like uni-directiononal, bi-directiononal, and aggregation
  - One line connects two classes depicting an association

  - Multiplicity: ***0*** (no instance); ***0..1*** (no instance or one instance);***1*** or ***1..1*** (exactly one instance); ***** or ***0..**** (zero or more instances); and ***1..**** (one or more instances.)

Unified Modeling Language Class Diagram (cont.)

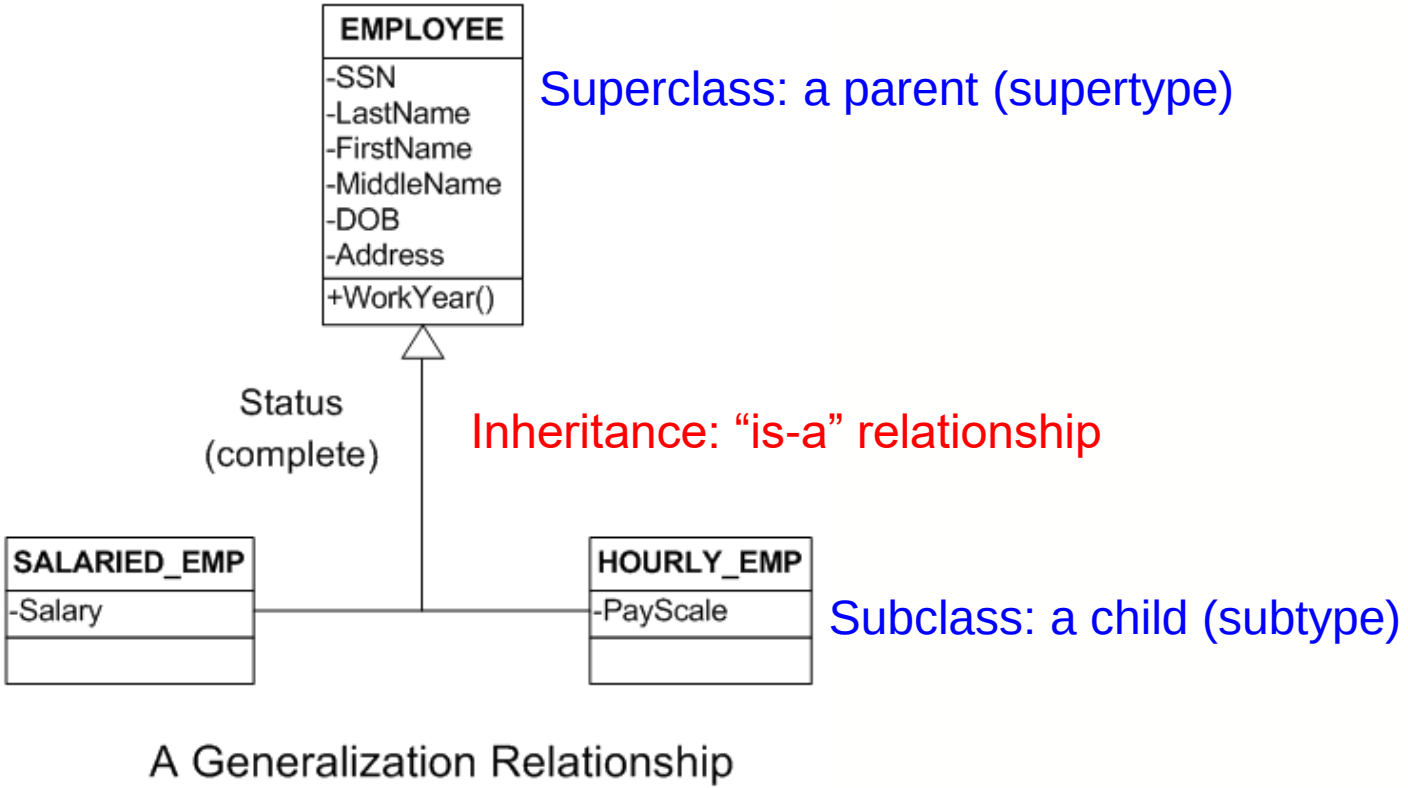
- Inheritance: “is-a” relationship



- Aggregation: “has-a” relationship

- Composition: “is-composed-of” relationship

Unified Modeling Language Class Diagram (cont.)



A Generalization Relationship

Unified Modeling Language Class Diagram (cont.)

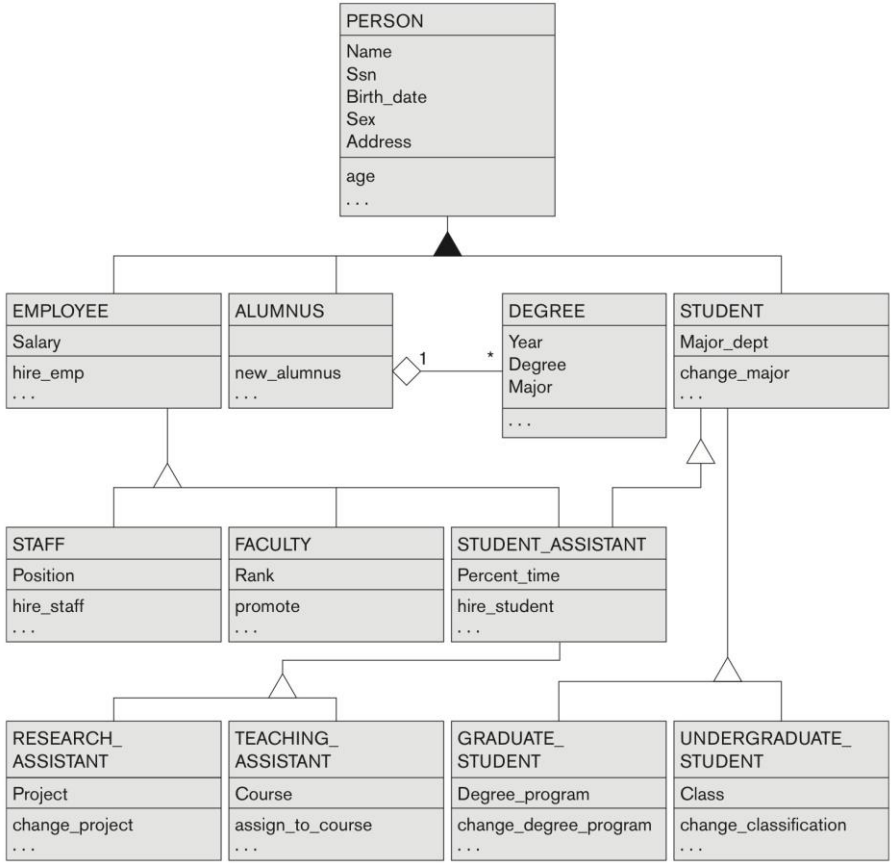


Figure 4.10 A UML class diagram corresponding to the EER diagram in Figure 4.7, illustrating UML notation for specialization / generalization.

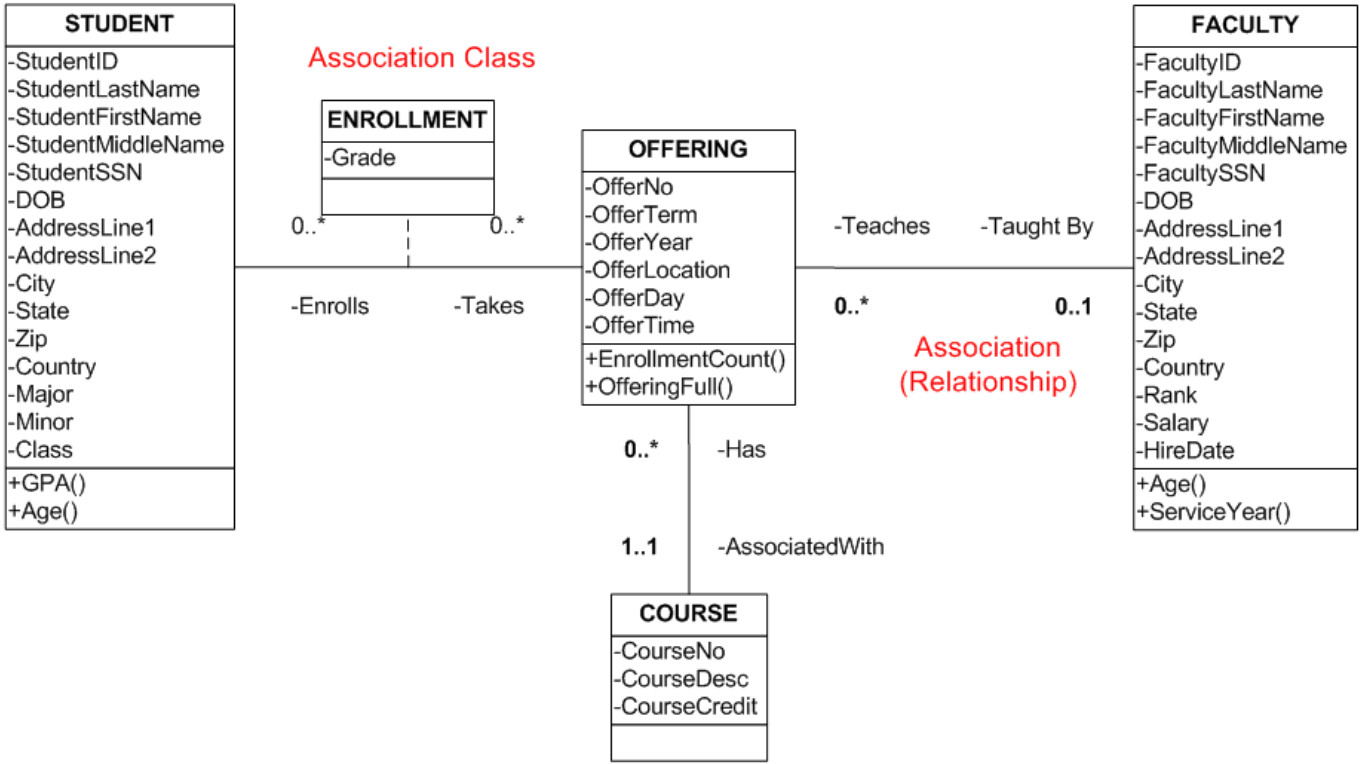
Aggregation:
ALUMNUS and DEGREE

Inheritance with overlapping constraint:
PERSON vs. EMPLOYEE, ALUMUS, STUDENT

Inheritance with disjoint constraints:
EMPLOYEE vs. STAFF, FACULTY, STUDENT_ASSISTANT

STUDENT vs. GRADUATE_STUDENT, UNDERGRADUATE_STUDENT

Unified Modeling Language Class Diagram (cont.)



More Information for OO Analysis and Design

- Most object-oriented applications involve storing the data associated with objects through data retrieval and data manipulation in databases (which will most likely be ‘relational’ or ‘object-relational’.)

- CS 605.704 Object-Oriented Analysis & Design covers the principals of OO approaches to modeling software requirements and design.

EERD and ERD Conclusions

- EERD improves conceptual database design with rich notations and constraints.
  - Superclass and subclass
  - Total or partial
  - Disjoint or overlapping
- Traditional data modeling for ERDs still dominates database design.
- Database design starts/completes first before the application development.