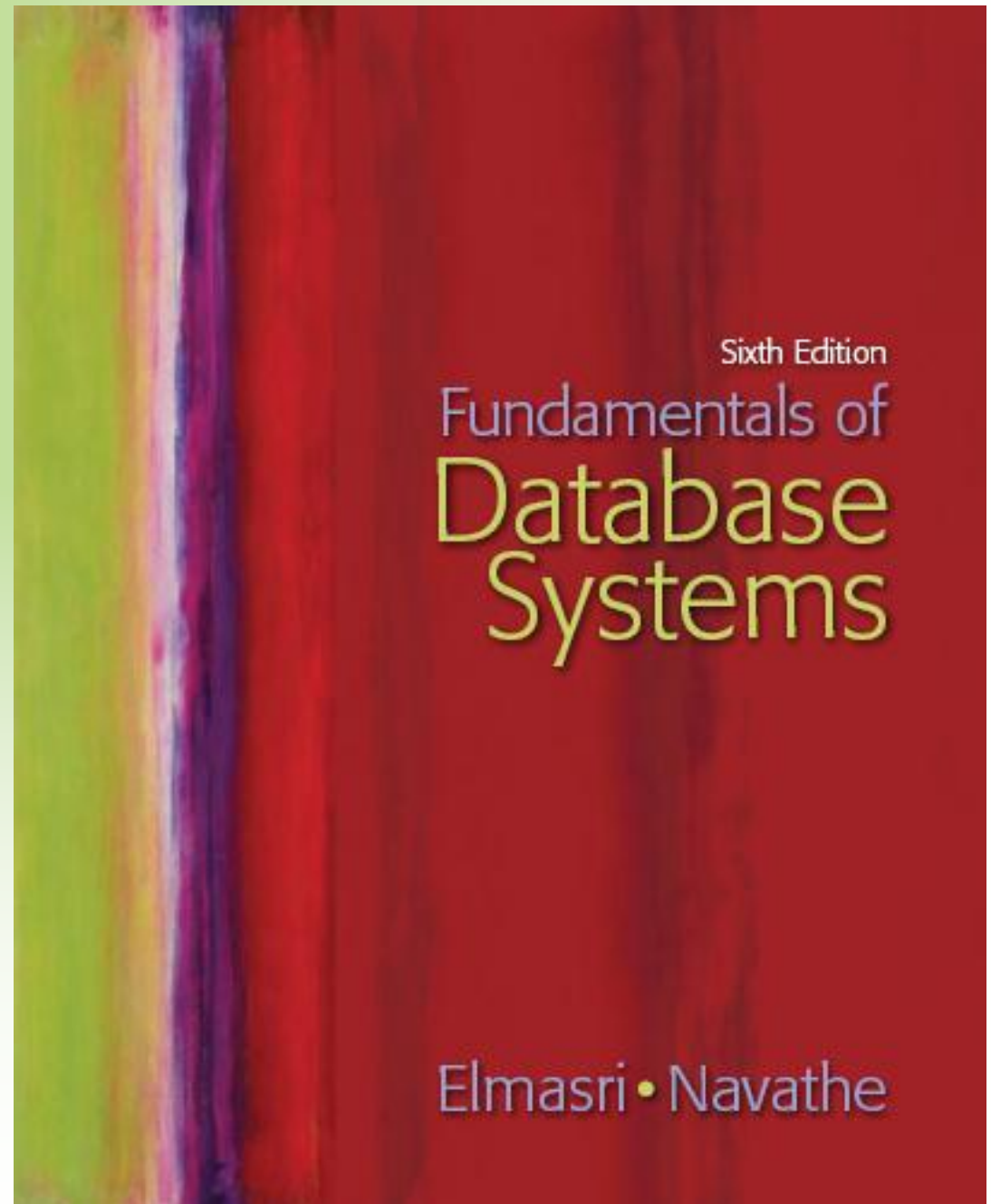


Chapter 8

The Enhanced Entity- Relationship (EER) Model



Addison-Wesley
is an imprint of

PEARSON

Chapter 8 Outline

- Subclasses, Superclasses, and Inheritance
- Specialization and Generalization
- Constraints and Characteristics of Specialization and Generalization Hierarchies
- Modeling of UNION Types Using Categories

Chapter 8 Outline (cont'd.)

- A Sample UNIVERSITY EER Schema, Design Choices, and Formal Definitions
- Example of Other Notation: Representing Specialization and Generalization in UML Class Diagrams
- Data Abstraction, Knowledge Representation, and Ontology Concepts

The Enhanced Entity-Relationship (EER) Model

- **Enhanced ER (EER) model**
 - Created to design more accurate database schemas
 - Reflect the data properties and constraints more precisely
 - More complex requirements than traditional applications

Subclasses, Superclasses, and Inheritance

- EER model includes all modeling concepts of the ER model
- In addition, EER includes:
 - **Subclasses and superclasses**
 - **Specialization and generalization**
 - **Category or union type**
 - **Attribute and relationship inheritance**

Subclasses, Superclasses, and Inheritance (cont'd.)

- **Enhanced ER or EER diagrams**
 - Diagrammatic technique for displaying these concepts in an EER schema
- **Subtype or subclass** of an entity type
 - Subgroupings of entities that are meaningful
 - Represented explicitly because of their significance to the database application

Subclasses, Superclasses, and Inheritance (cont'd.)

- Terms for relationship between a superclass and any one of its subclasses
 - **Superclass/subclass**
 - **Supertype/subtype**
 - **Class/subclass** relationship
- **Type inheritance**
 - Subclass entity inherits all attributes and relationships of superclass

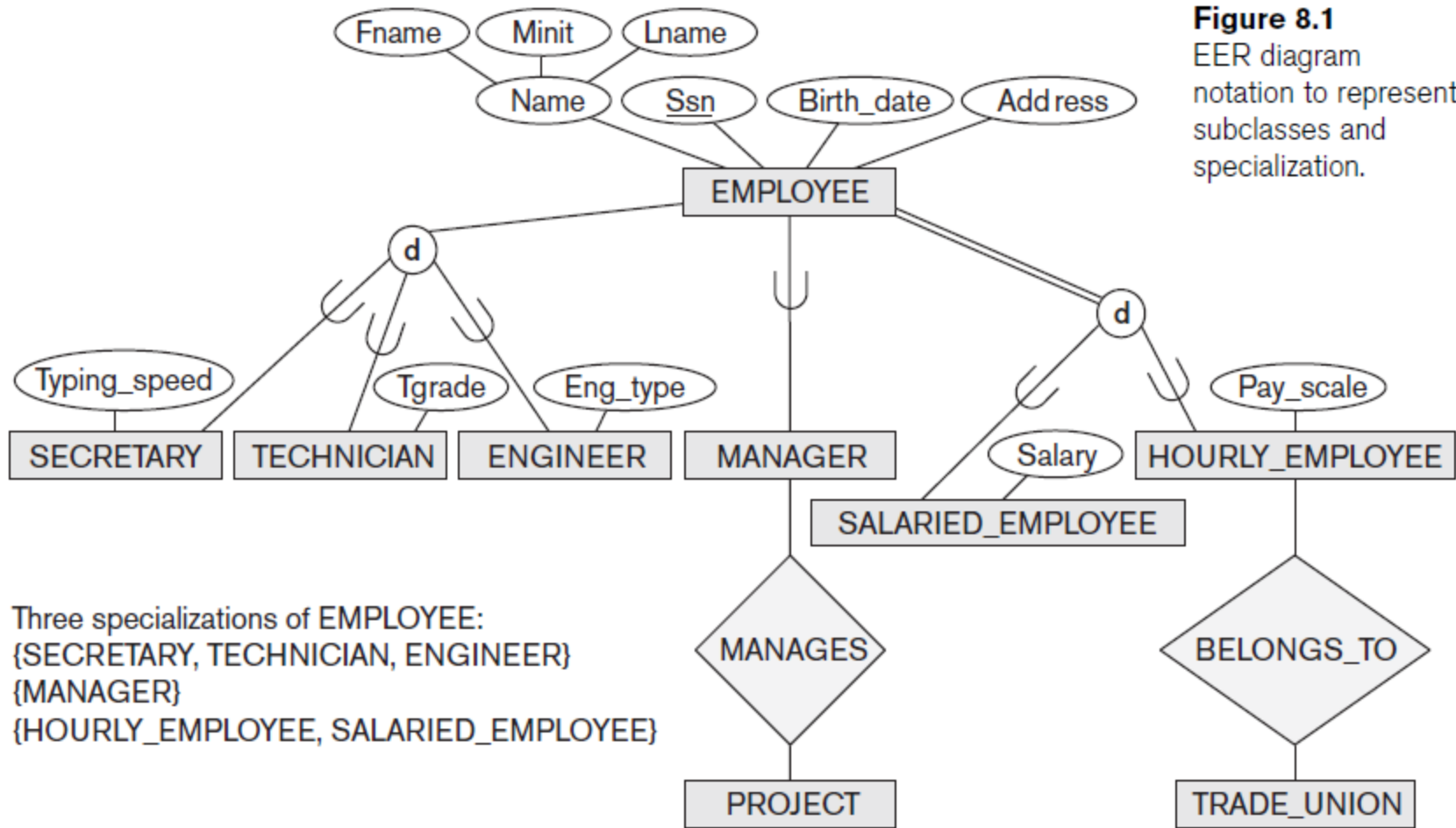


Figure 8.1
EER diagram notation to represent subclasses and specialization.

Three specializations of EMPLOYEE:
 {SECRETARY, TECHNICIAN, ENGINEER}
 {MANAGER}
 {HOURLY_EMPLOYEE, SALARIED_EMPLOYEE}

Specialization and Generalization

- **Specialization**
 - Process of defining a set of subclasses of an entity type
 - Defined on the basis of some distinguishing characteristic of the entities in the superclass
- Subclass can define:
 - **Specific attributes**
 - **Specific relationship types**

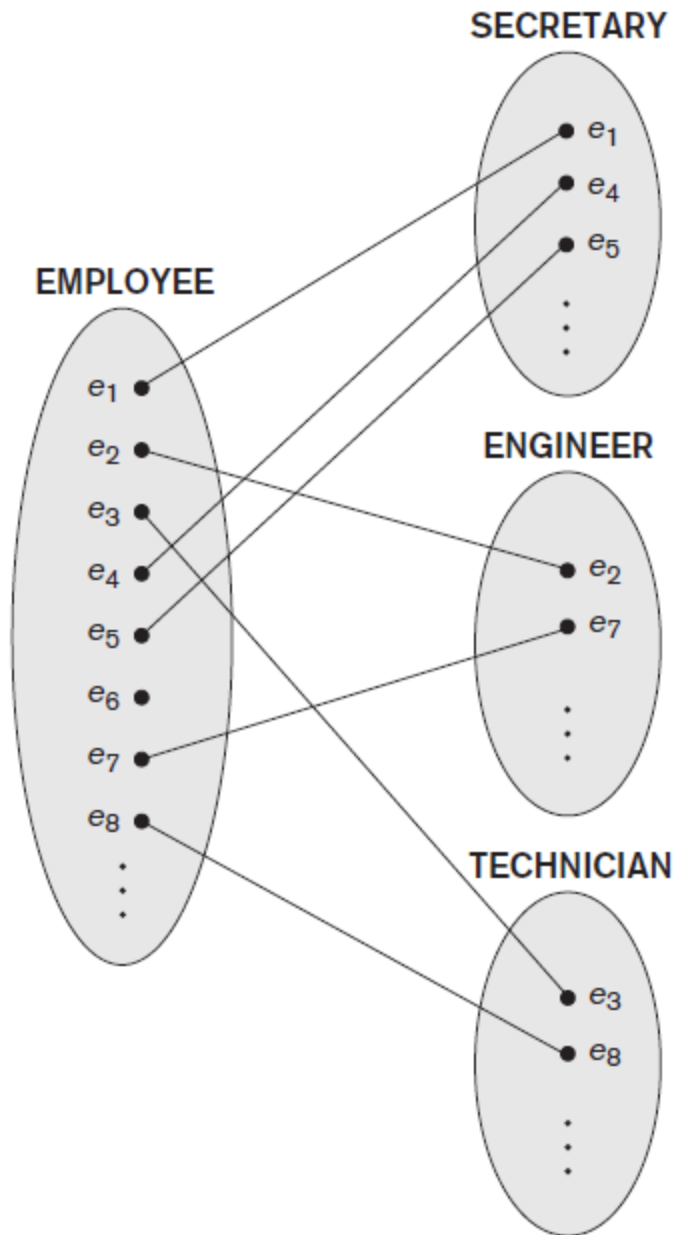


Figure 8.2
Instances of a specialization.

Specialization and Generalization (cont'd.)

- Certain attributes may apply to some but not all entities of the superclass
- Some relationship types may be participated in only by members of the subclass

Generalization

- Reverse process of abstraction
- **Generalize** into a single **superclass**
 - Original entity types are special subclasses
- **Generalization**
 - Process of defining a generalized entity type from the given entity types

Constraints and Characteristics of Specialization and Generalization Hierarchies

- Constraints that apply to a single specialization or a single generalization
- Differences between specialization/generalization lattices and hierarchies

Constraints on Specialization and Generalization

- May be several or one subclass
- Determine entity subtype:
 - **Predicate-defined (or condition-defined) subclasses**
 - **Attribute-defined specialization**
 - **User-defined**

Constraints on Specialization and Generalization (cont'd.)

- **Disjointness constraint**
 - Specifies that the subclasses of the specialization must be disjoint
- **Completeness (or totalness) constraint**
 - May be **total** or **partial**
- Disjointness and completeness constraints are independent

Specialization and Generalization Hierarchies and Lattices

- **Specialization hierarchy**
 - Every subclass participates as a subclass in only one class/subclass relationship
 - Results in a **tree structure** or **strict hierarchy**
- **Specialization lattice**
 - Subclass can be a subclass in more than one class/subclass relationship

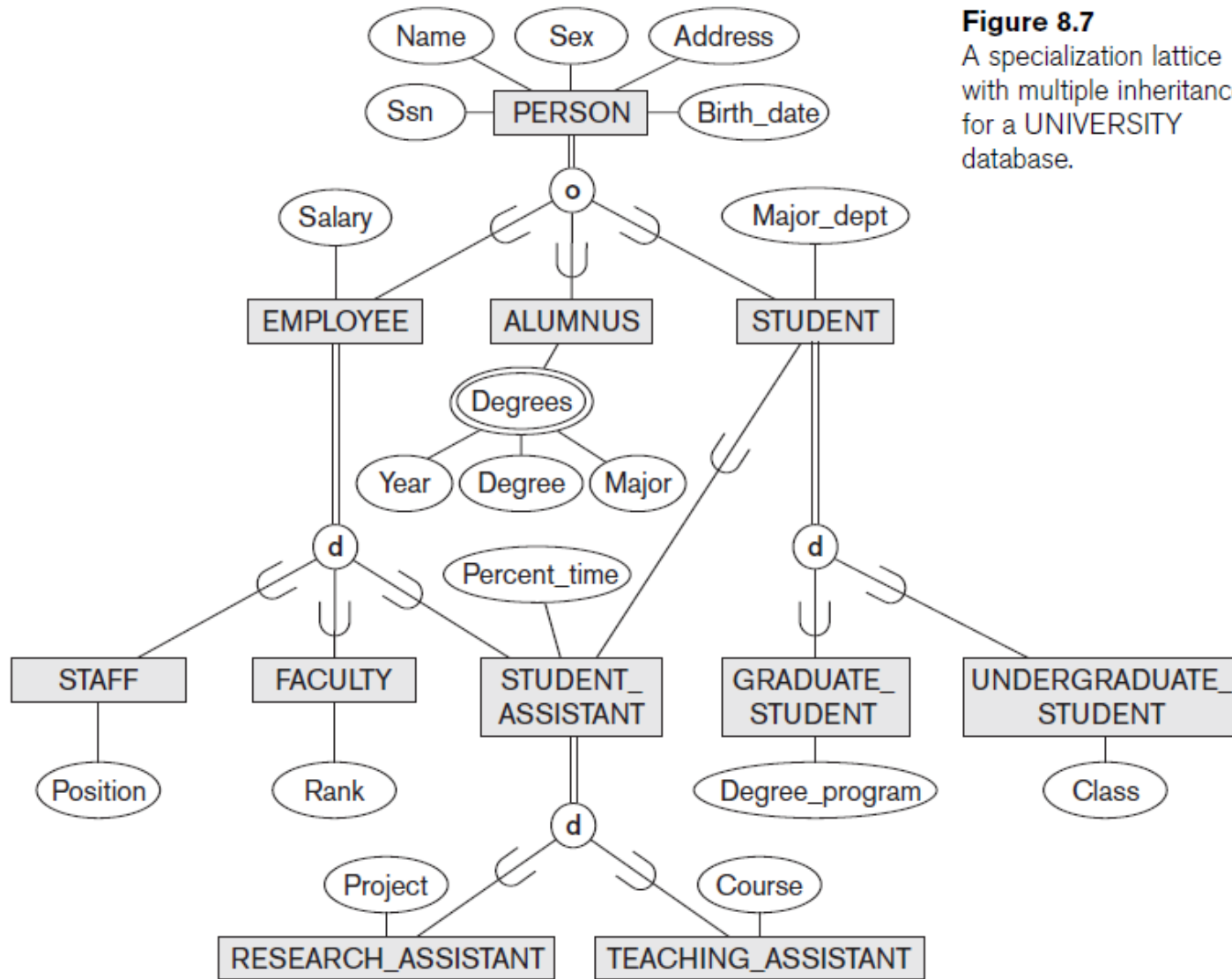


Figure 8.7
A specialization lattice with multiple inheritance for a UNIVERSITY database.

Specialization and Generalization Hierarchies and Lattices (cont'd.)

■ **Multiple inheritance**

- Subclass with more than one superclass
- If attribute (or relationship) originating in the same superclass inherited more than once via different paths in lattice
 - Included only once in shared subclass

■ **Single inheritance**

- Some models and languages limited to single inheritance

Utilizing Specialization and Generalization in Refining Conceptual Schemas

- Specialization process
 - Start with entity type then define subclasses by successive specialization
 - **Top-down conceptual refinement process**
- **Bottom-up conceptual synthesis**
 - Involves generalization rather than specialization

Modeling of UNION Types Using Categories

- **Union type** or a **category**
 - Represents a single superclass/subclass relationship with more than one superclass
 - Subclass represents a collection of objects that is a subset of the UNION of distinct entity types
 - Attribute inheritance works more selectively
 - Category can be **total** or **partial**
- Some modeling methodologies do not have union types

A Sample UNIVERSITY EER Schema, Design Choices, and Formal Definitions

- The UNIVERSITY Database Example
 - UNIVERSITY database
 - Students and their majors
 - Transcripts, and registration
 - University's course offerings

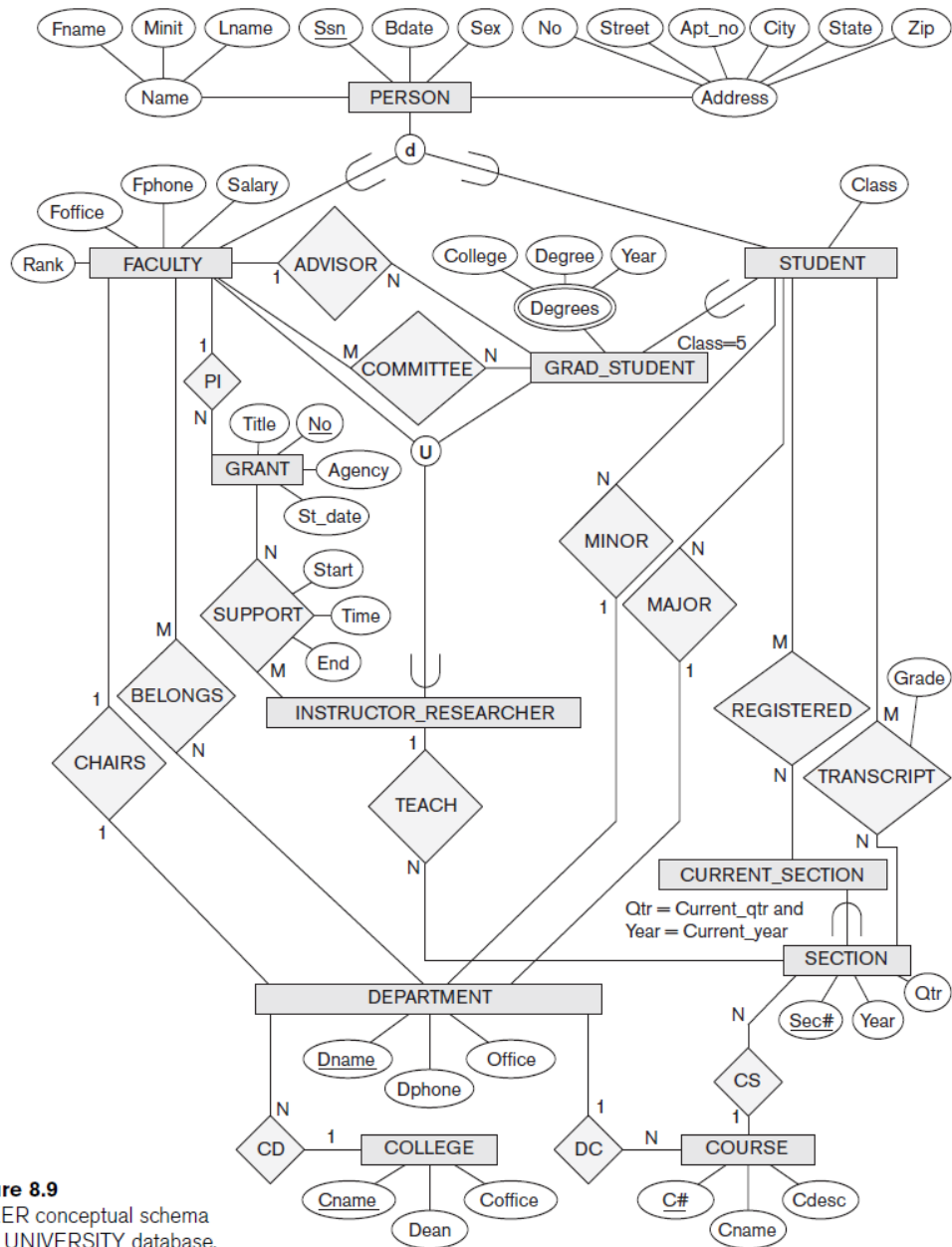


Figure 8.9
An EER conceptual schema
for a UNIVERSITY database.

Design Choices for Specialization/Generalization

- Many specializations and subclasses can be defined to make the conceptual model accurate
- If subclass has few specific attributes and no specific relationships
 - Can be merged into the superclass

Design Choices for Specialization/Generalization (cont'd.)

- If all the subclasses of a specialization/generalization have few specific attributes and no specific relationships
 - Can be merged into the superclass
 - Replace with one or more type attributes that specify the subclass or subclasses that each entity belongs to

Design Choices for Specialization/Generalization (cont'd.)

- Union types and categories should generally be avoided
- Choice of disjoint/overlapping and total/partial constraints on specialization/generalization
 - Driven by rules in miniworld being modeled

Formal Definitions for the EER Model Concepts

■ **Class**

- Set or collection of entities
- Includes any of the EER schema constructs of group entities

■ **Subclass**

- Class whose entities must always be a subset of the entities in another class

■ **Specialization**

- Set of subclasses that have same superclass

Formal Definitions for the EER Model Concepts (cont'd.)

- **Generalization**
 - Generalized entity type or superclass
- **Predicate-defined**
 - Predicate on the attributes of C is used to specify which entities in C are members of S
- **User-defined**
 - Subclass that is not defined by a predicate

Formal Definitions for the EER Model Concepts (cont'd.)

■ **Category**

- Class that is a subset of the union of n defining superclasses

■ **Relationship type**

- Any class can participate in a relationship

Example of Other Notation

- Representing specialization and generalization in UML class diagrams
 - Basic notation
 - See Figure 8.10
 - Base class
 - Root superclass
 - Leaf classes
 - Subclasses (leaf nodes)

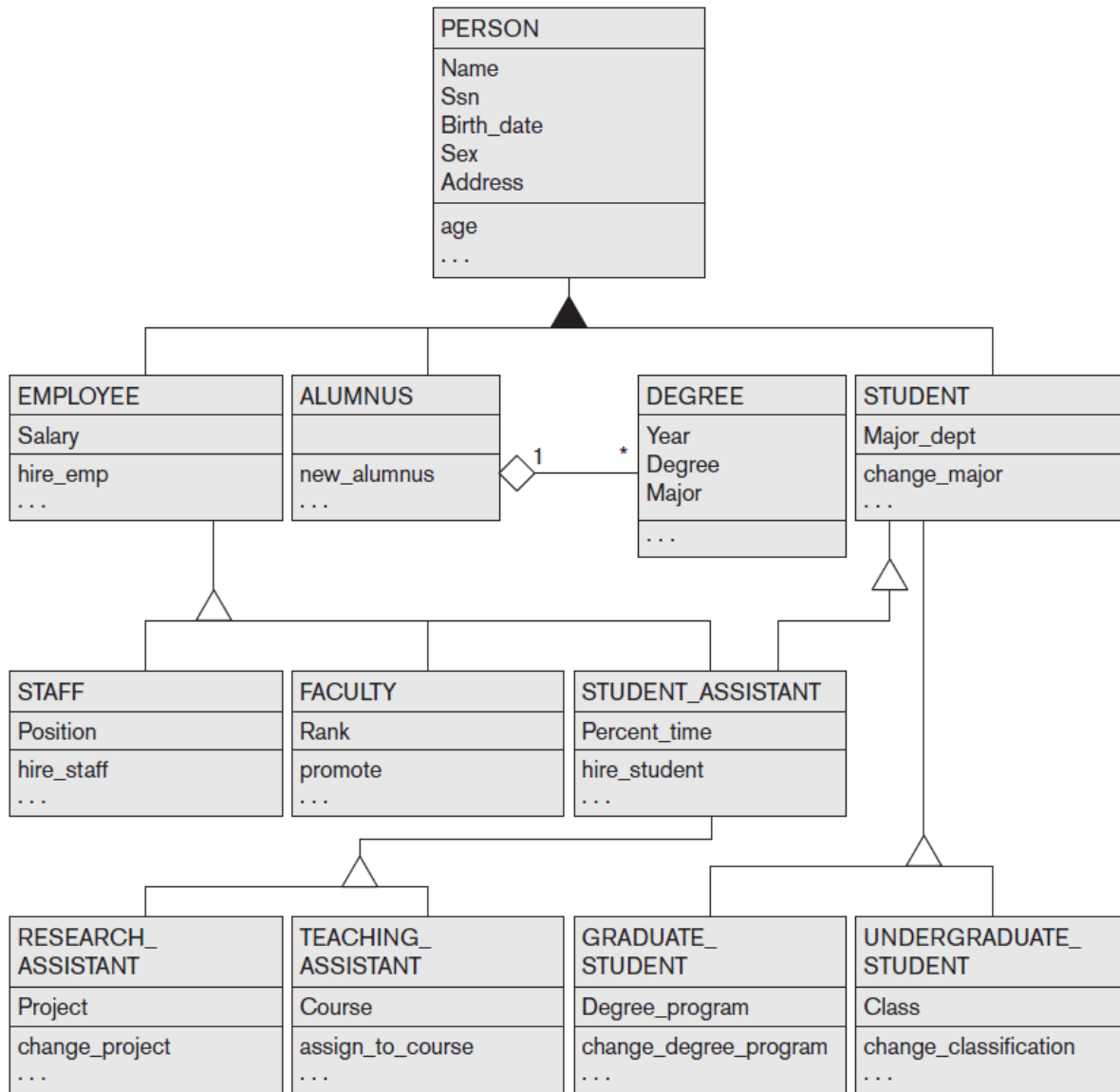


Figure 8.10
 A UML class diagram corresponding to the EER diagram in Figure 8.7,
 illustrating UML notation for specialization/generalization.

Data Abstraction, Knowledge Representation, and Ontology Concepts

- Goal of **knowledge representation (KR)** techniques
 - Accurately model some **domain of knowledge**
 - Create an **ontology** that describes the concepts of the domain and how these concepts are interrelated
- Goals of KR are similar to those of semantic data models
 - Important similarities and differences

Classification and Instantiation

■ Classification

- Systematically assigning similar objects/entities to object classes/entity types

■ Instantiation

- Inverse of classification
- Generation and specific examination of distinct objects of a class

Classification and Instantiation (cont'd.)

- **Exception objects**
 - Differ in some respects from other objects of class
 - KR schemes allow such **class properties**
- One class can be an instance of another class (called a meta-class)
 - Cannot be represented directly in EER model

Identification

- Abstraction process
- Classes and objects are made uniquely identifiable by means of some **identifier**
- Needed at two levels
 - To distinguish among database objects and classes
 - To identify database objects and to relate them to their real-world counterparts

Specialization and Generalization

- **Specialization**

- Classify a class of objects into more specialized subclasses

- **Generalization**

- Generalize several classes into a higher-level abstract class
- Includes the objects in all these classes

Aggregation and Association

■ Aggregation

- Abstraction concept for building composite objects from their component objects

■ Association

- Associate objects from several independent classes

■ Main structural distinction

- When an association instance is deleted
 - Participating objects may continue to exist

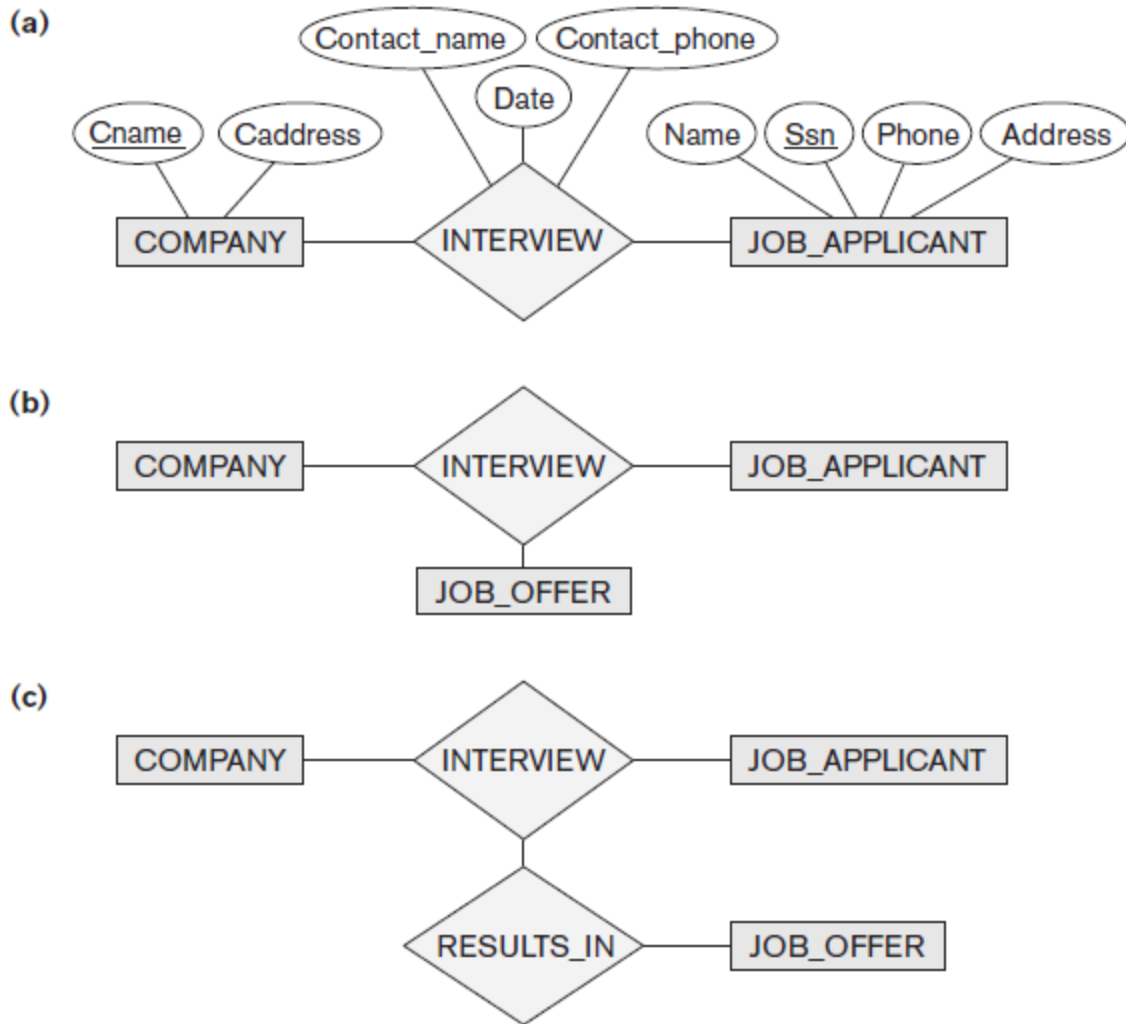


Figure 8.11

Aggregation. (a) The relationship type **INTERVIEW**. (b) Including **JOB_OFFER** in a ternary relationship type (incorrect). (c) Having the **RESULTS_IN** relationship participate in other relationships (not allowed in ER). (d) Using aggregation and a composite (molecular) object (generally not allowed in ER but allowed by some modeling tools). (e) Correct representation in ER.

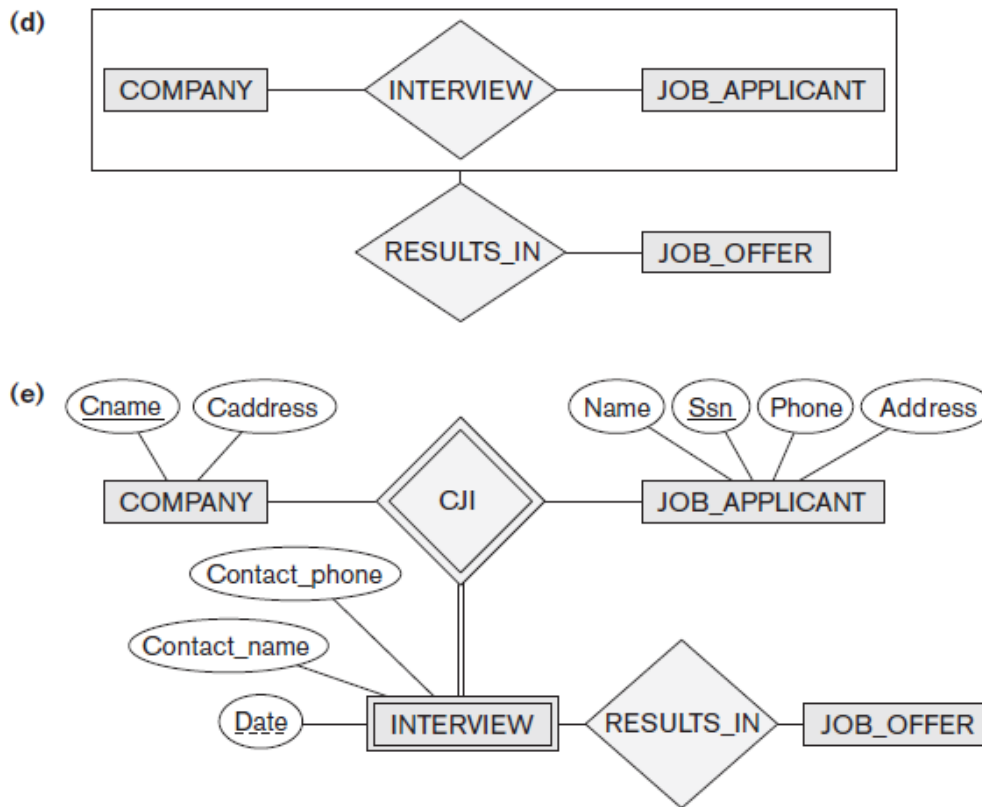


Figure 8.11

Aggregation. (a) The relationship type INTERVIEW. (b) Including JOB_OFFER in a ternary relationship type (incorrect). (c) Having the RESULTS_IN relationship participate in other relationships (not allowed in ER). (d) Using aggregation and a composite (molecular) object (generally not allowed in ER but allowed by some modeling tools). (e) Correct representation in ER.

Ontologies and the Semantic Web

- Documents contain less structure than database information does
- **Semantic Web**
 - Allow meaningful information exchange and search among machines
- **Ontology**
 - Specification of a **conceptualization**
- **Specification**
 - Language and vocabulary terms used to specify conceptualization

Summary

- Enhanced ER or EER model
 - Extensions to ER model that improve its representational capabilities
 - Subclass and its superclass
 - Category or union type
- Notation and terminology of UML for representing specialization and generalization