Enhanced Entity-Relationship (EER) Modeling

Anne Denton

Department of Computer Science
North Dakota State University

Outline

Table of Contents

Enhanced Entity Relationship Modeling

- Allows modeling of inheritance
  - So far we only had associations between entities
  - We decided that we did need to treat aggregation separately
  - Inheritance does have to be treated separately
- Can be mapped to relational model
  - There are multiple alternatives for representing inheritance
  - Specific choices depend on modeling constraints
- Object-relational features could be used but inheritance can also be mapped to the relational model

Elements of the EER Model

- Includes all elements of the ER Model
- Additionally includes class hierarchy using class / subclass relationships
    - Class hierarchies may be the result of a specialization or a generalization
    - May affect modeling constraints
- Categories are class hierarchies without common primary key

Class / subclass relationship

- Represents an "IS-A" relationship
- Members of subclasses represent the same real world entity
- An entity cannot be a member of a subclass without being a member of its superclass
- A subclass entity possesses all attributes as well as relationships of the superclass
- Can be modeled similar to 1-to-1 relationships

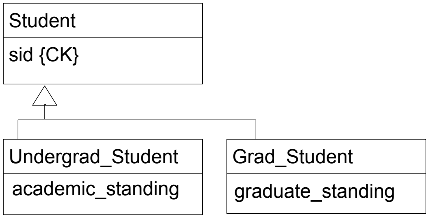
Question 1 (Multiple Answers can be correct)

A class-subclass relationship can be characterized as a

1. Has-a relationship
2. Is-a relationship

Representation in an EER diagram

- A specialization is shown as a triangle
- Allows subclasses to have different attributes, e.g.,
  - Grad_Student may have the attribute graduate_standing (yes/no)
  - Undergraduate_Student would have academic_standing (freshman, sophomore etc.)



- Also allows subclasses to have different relationships

Specialization and Generalization

- A specialization is set of subclasses that is defined on the basis of a distinguishing characteristics
- A generalization is defined by generalizing related entities to have one superclass
  - That means, e.g., that for a generalization all superclass members are in at least one subclass
- For the most part, we will not distinguish between specializations and generalizations

Differences with regard to object-oriented programming

- EER only a modeling technique
  - Object-relational databases have complex types and type inheritance
  - EER is only used at the modeling stage
- Concepts more relaxed since anything is mapped to relational
- Multiple overlapping specializations can apply to the same entity
  - A student may be a Grad_Student or Undergrad_Student
  - In either case he/she may or may not be a Teaching_Assistant

Question 2 (Multiple answers can be correct)

Enhanced Entity Relationship modeling

1. Is a part of working with object-relational databases
2. Allows mapping concepts known from object-oriented programming to standard relational databases

Table of Contents

Disjointness constraint

- A specialization is disjoint if each real world entity can belong to one subclass at most, e.g.,
  - A faculty member at NDSU can be either part-time or full-time but not both
- A specialization is overlapping if each real world entity can belong to more than one subclass, e.g.,
  - A student assistant could be research assistant or teaching assistant or both

Question 3

A library wants to save special information for textbooks and for novels.

1. Overlapping or disjoint specialization?

Completeness constraint

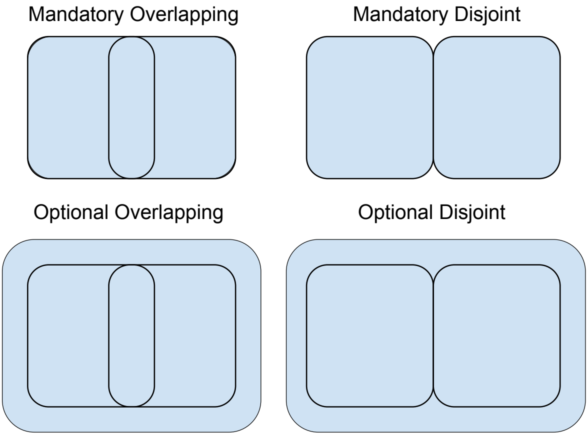
- A specialization is mandatory if each real world entity hast to belong to at least one subclass, e.g.,
  - Every student has to be either an undergraduate or a graduate student (assuming there are no non-degree students)
- A specialization is optional if each real world entity may not belong to any subclass, e.g.,
  - A student does not have to be a student assistant

Question 4

A library wants to save special information for textbooks and for novels.

1. Mandatory or optional specialization?

Any combination of disjointness and completeness constraints is possible



Question 5 (Multiple answers can be correct)

A specialization is disjoint if an entity

1. Can be a member of no more than one subclass
2. Must be a member of at least one subclass
3. Can be a member of more than one subclass
4. Does not have to be a member of any subclass

Question 6 (Multiple answers can be correct)

A specialization is overlapping if an entity

1. Can be a member of no more than one subclass
2. Must be a member of at least one subclass
3. Can be a member of more than one subclass
4. Does not have to be a member of any subclass

Question 7 (Multiple answers can be correct)

A specialization is mandatory if an entity

1. Can be a member of no more than one subclass
2. Must be a member of at least one subclass
3. Can be a member of more than one subclass
4. Does not have to be a member of any subclass

Question 8 (Multiple answers can be correct)

A specialization is optional if an entity

1. Can be a member of no more than one subclass
2. Must be a member of at least one subclass
3. Can be a member of more than one subclass
4. Does not have to be a member of any subclass

Examples

Any combination possible

	Mandatory	Optional
Disjoint	Faculty can be part-time or full-time and nothing else	Library books can be textbooks or novels (assuming no novel is a textbook)
Overlapping	Persons in the NDSU database can be students, employees, or alumni (assuming all have some relationship to NDSU)	A conference gives a special status to members of some professional associations

Defining predicate

- Sometimes an attribute is used to specify the subclass
  - If the specialization is disjoint the defining predicate can be a simple (i.e. single-valued) attribute
  - If the specialization is overlapping the defining predicate has to be multi-valued, modeled as multiple bit-vectors, one for each subclass
- Alternatively, subclass membership may be user-defined

Table of Contents

Mapping Alternatives

Assume that there are N subclasses

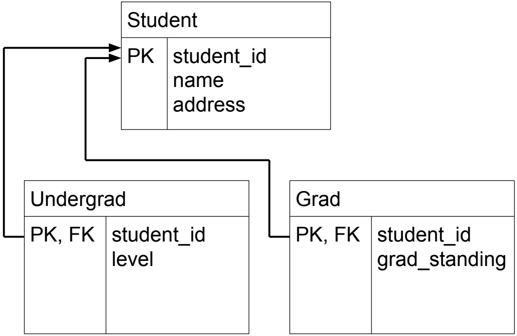
- $N + 1$ tables: One table for each subclass plus one table for the superclass
- N tables: One table for each subclass, which also holds the superclass attributes
- 1 table: One table that holds all attributes

Solution with subclass and superclass tables ($N + 1$ tables)

- The superclass table has the shared attributes and relationships
- Each subclass table have the specific attributes of that subclass
- Does not require a defining predicate
- Combining the information in superclass and subclasses requires a join, i.e. slowest of the alternatives
- Space efficiency trade-off: Foreign keys use some space, but the solution avoids null values
- Works for all constraint combinations

Solution with subclass and superclass tables ($N + 1$ tables): Example

- Notice the similarities to 1-to-1 relationships: The primary key of the superclass also used as foreign key in the subclasses



- Preferable when there are many subclass attributes

Question 9

A specialization or generalization is modeled in a way that is most similar to a

1. 1-to-many relationship
2. Many-to-many relationship
3. 1-to-1 relationship

Solution with only subclass tables (*N* tables)

- The subclass tables have both the shared and the specific attributes
- Does not require a defining predicate
- Combining all records requires an outer union, which is normally faster than a join (An outer union keeps all attributes from both relations and includes null values where necessary)
- Only works for disjoint relations! Otherwise redundancy is introduced
- If the specialization is optional, an additional "none of the above" table has to be added
- Very space efficient (when relations are disjoint)

Solution with only subclass tables (*N* tables): Example

- Student records are in one or the other but cannot be in both

UndergradStudent	
PK	student_id
	name
	address
	level

GradStudent	
PK	student_id
	name
	address
	grad_standing

- Enforcing a uniqueness of the primary key across all relations requires use of a constraint specification language (or an extra table that holds all primary keys)

Solution with a single table

- One table with all attributes
- No need for joins or unions, hence typically fastest
- Problem: Potentially many null values
- Space-tradeoff: No need for foreign keys, but null values use space
- Preferable when there are few subclass attributes
- One defining attribute sufficient for disjoint subclasses
- *N* binary defining attributes needed for overlapping subclasses

Student	
PK	student_id name address is_grad level grad_standing

Question 10 (Multiple answers can be correct)

A relational representation of a specialization that uses a superclass table and one table for each subclass

1. Avoids null values
2. Requires the specialization to be mandatory
3. Requires the specialization to be disjoint
4. Is particularly efficient to query

Question 11 (Multiple answers can be correct)

A relational representation of a specialization that uses one table for each subclass and stores superclass attributes within those

1. Avoids null values
2. Requires the specialization to be mandatory unless modifications to the definition are made
3. Requires the specialization to be disjoint
4. Requires many joins (i.e., queries that combine records based on equality of a foreign key – primary key combination)

Question 12 (Multiple answers can be correct)

A relational representation of a specialization that uses just one table

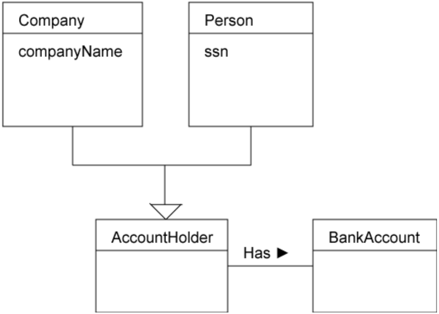
1. Avoids null values
2. Requires the specialization to be mandatory
3. Requires the specialization to be disjoint
4. Is particularly efficient to query

Table of Contents

Categories

- So far we only had a shared subclass of classes that originated from a single superclass
- Now we look at shared subclasses of superclasses that correspond to distinct entity types
- Example: If not every Company and Person is an Account_Holder there is no shared key
- $\Rightarrow$ We have to introduce a surrogate key

- Like a generalization, except that not even identifying information is shared



Question 13 (Multiple answers can be correct)

Categories

1. Are similar to overlapping, optional specializations
2. Are similar to disjoint, mandatory specializations
3. Typically do not share many attributes between categories
4. Are typically well represented through a single table