Lab 02- Exploring Enhanced Entity-Relationship Models

Vishakha Maruti Sonmore (CSULB ID: 032188141)

College of Business, California State University, Long Beach

Course Number: I S 680 Sec02 11140 Database Management Systems

Instructor: Dr. Amini, Mostafa

Due date: 14- Feb-2024

Week 3 – Lab #2 – Exploring Enhanced Entity-Relationship Models

Table of Contents

Exploring the Enhanced Entity Relationship (EERD) Models

Welcome to your second lab in IS380! In this lab we will expand our experience working with Entity Relationship Diagrams by exploring a new set of relationships that we can create with them: Supertype/Subtype relationships.

We can think of supertype and subtype relationships as general and specific categories respectively for a particular database entity (i.e. "*thing, person, place, unit, object or any item about which the data should be captured*").

By practicing EERD models you will gain the capability of creating more complex models that represent general categories and subcategories within an entity. You will also be able to denote details about how these entities relate to each other including whether one entity instance of the supertype must belong to a subtype or not, and if it does belong to a subtype, can it belong to more than one at a time.

The lab will first introduce the main concepts to concern for the development of effective Enhanced Entity Relationship Diagrams (EERDs) and will then provide 2 problems to solve. The first problem will give a brief discussion of the development of a model for a non-profit organization, while the 2nd one will give you the opportunity to examine a technology company's offerings (think of a company like DELL).

Lab Goals

- SLO1- To infer and Interpret supertype and subtype relationships from business rules.
- SLO2- To Design effective data models with supertype and subtype notations including the proper design of its properties.
- SLO3- To read and understand the key design considerations behind the completeness and disjoint constraints.
- SLO4- To recognize how to use subtype discriminators and when to use entity clusters to summarize models.

What You Will Need to Begin

- Visit draw.io (https://www.draw.io) to gain access to a modeling tool either through its web version or by downloading a copy into your own computer.
- Please complete the modeling exercises in draw.io and copy your models (with your name visible) into this document. Screenshots are an acceptable choice for copying the models.
- Once completed, submit the exercise (this document) into a corresponding DropBox folder within BeachBoard.

Part 1 – Introduction, & summary of Key Concepts for Exploring EERD Models

NOTE: This lab builds on lab #1 by assuming you can develop basic Entity Relationship Diagrams (specifically conceptual models as developed in class). If you have doubts, please do not hesitate to check Lab #1 for instructions on how to carry out the model or ask your instructor for assistant.

Between this section and the following, we will go through each component of the Enhanced Entity Relationship Diagram. We will first review each concept of interest in preparation for carrying out exercises on our own. These are:

1. Main Concepts:

a. Enhanced Entity Relationship Diagrams (EERD):

i. Supertypes

ii. Subtypes

iii. Generalization

iv. Specialization

b. Completeness Constraint:

- Total Specialization Rule (An instance of the supertype must also belong to either of the subtypes; Use double line)
- Partial Specialization Rule (An instance of an entity could belong to either of the subtypes of none of them; Use single line)

c. Disjointness Constraints:

- Disjoint Rule (Instance of Supertype could be **only ONE** of the subtypes; write a "d" within circle)
- Overlap Rule (Instance of Supertype could be more than one subtype; write an "o" within circle)

d. **Subtype Discriminator** (attribute of the supertype that describes to which subtype an instance of the entity belongs)

- If Disjoint: a simple attribute is created with different categories for each subtype
- If Overlapping: a composite attribute is created with subparts indicating whether or not an instance of the entity belongs to each separate subtype)

e. **Entity Clusters** (If model gets too complex, we can group it into "clusters" by replacing them with an abstract entity type)

f. **Packaged Models** (the search for best practices means that rather than build our models entirely from the ground up, we can acquire data models from vendors)

- Universal data model (generic model template that can be reused with ease in multiple scenarios)
- Industry-specific data model (predefined models for specific scenarios)

Part 2 – Creating your own EERD.

Exercise 1:

To strengthen our modeling skills, we will now explore step by step a single problem:

A non-profit organization depends on several different types of persons to function effectively. The organization is interested in the following attributes for all these persons: SSN, Name, Address, City/Zip Code, and Cellphone.

There are 3 main types of persons of interest: Employees, Volunteers, and Donors. Employees have only a Date Hired attribute, and volunteers have a Skill attribute. On the other hand, Donors have a unique relationship ("Donates") with the entity type "Item".

A Donor must have donated one or more items, and an item may have no donors or one or more donors.

There are persons other than employees, volunteers, and donors who are of interest to the organization so that a person need not belong to any of these three groups. On the other hand, at a given time a person may belong to two or more of these groups (e.g., employee and donor).

Let us start by repeating the process with carried out during our first lab and:

Q1. Please identify the entities, their attributes, and their relationships as suggested in the problem above:

Entity	Attributes	Relationship to (add name to it)
Person	SSN, Name, Address, City/Zip Code, and Cellphone.	Person (Supertype) has four subtype Employee, Volunteer, Donor, Other_Person
Employees	Date Hired	Associate with supertype Person
Volunteers	Skill	Associate with supertype Person
Donors	Doner_ID	• Donates has unique relationship with item • Donor must have donated one or more items
Item	Quantity	• item may have no donors OR • one or more donors.
Other_Person	OtherPersonID, AffiliationDate	Associate with supertype Person

Q2. Are there any supertype/subtype relationships present? If so, what entities would be involved? Which would be the supertype, and which would be the subtype?

Yes, there is a supertype/subtype relationship present.

Entities involved:

- ***Supertype***: Person
- ***Subtypes***: Employee, Volunteer, Donor, Other_Person

In this scenario, "Person" is the supertype, and Employee, Volunteer, Donor, and Other_Person are subtypes.

Q3. Now consider the completeness constraint. In the supertype/subtype relationship identified above a total specialization rule or a partial specialization rule? What does this mean for this relationship? (Refer to part 1 or the class lecture for a refresher on this)

In the supertype/subtype relationship identified, there is a total specialization rule.

This means that every person must belong to one of the subtypes (Employee, Volunteer, Donor, Other_Person). There are no individuals in the system who are just a general "Person" without falling into one of these specific subtype categories. Every person is fully specialized as either an employee, volunteer, donor, or other person.

The relationship between Person and its subtypes (Employee, Volunteer, Donor, Other_Person) *follows a total specialization rule rather than a partial specialization rule.*

Here's why:

Total specialization (total inclusion constraint): This means that every entity instance in the supertype must be a member of at least one subtype. In the case of Person, every person must be classified as either an Employee, Volunteer, Donor, or Other_Person. There are no individuals in the system who are just "Persons" without falling into one of these specific subtype categories.

Partial specialization, on the other hand, would allow for instances of the supertype to exist without being members of any subtype. This is not the case in the scenario described. Every person is expected to belong to one of the specified subtypes, including Donor. Therefore, it follows a total specialization rule rather than a partial specialization rule.

Q4. Now consider the disjointness Constraints. Is this a disjoint rule or an overlapping rule? What does this mean for this relationship? (Refer to part 1 or the class lecture for a refresher on this)

Considering few scenarios to further explore the disjointness and overlapping constraints:

Overlapping Scenario 1:

- A person can be both an Employee and a Volunteer simultaneously.
- A person can be a Donor and also volunteer for events.

Overlapping Scenario 2:

- An Employee may also be a Donor and contribute to the organization financially.
- A Volunteer can be a Donor and donate items to the organization.

Disjoint Scenario:

- A person can be either an Employee, a Volunteer, or a Donor, but not more than one at the same time.
- An Employee is specifically hired for a position within the organization and may not concurrently serve as a Volunteer or a Donor.

Considering these scenarios, *it appears that the overlapping constraint is more fitting* for the described relationships. Individuals can have multiple roles simultaneously (Employee, Volunteer, Donor), suggesting that the subtypes can overlap.

This implies that a person may belong to more than one subtype, allowing for flexibility in modeling individuals with diverse roles within the organization.

Q5. Now let us consider the subtype discriminator. How would you describe the subtype discriminator for this supertype/subtype relationship? Is it a simple attribute, or a composite attribute?

In the context of the supertype/subtype relationship described, a *composite attribute* could serve as the subtype discriminator.

The composite attribute would encapsulate multiple simple attributes to provide a detailed representation of the various subtypes that a person could belong to within the organization.

For instance, the composite attribute could include the following components:

Composite Attribute: **Role**

- IsEmployee (Boolean flag indicating whether the person is an employee)

- IsVolunteer (Boolean flag indicating whether the person is a volunteer)

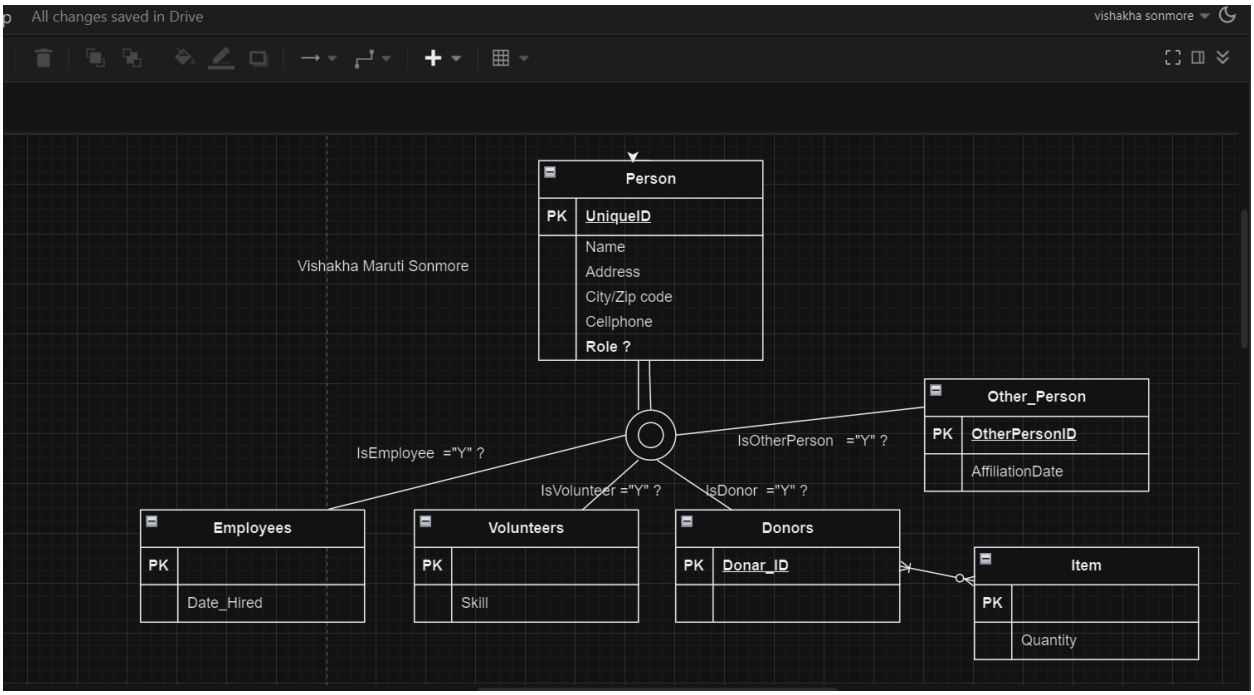
- IsDonor (Boolean flag indicating whether the person is a donor)

- IsOtherPerson (Boolean flag indicating whether the person is classified as "Other_Person")

Each boolean flag within the composite attribute would represent a different subtype that the person could be associated with. The presence of a "true" value for a specific flag would denote the person's membership in that particular subtype.

Utilizing a composite attribute as the subtype discriminator enables a comprehensive portrayal of the relationships, particularly in scenarios where individuals can assume multiple roles concurrently within the organizational context.

Q6. Finally, please use the information above and create a corresponding EERD model. Take a screenshot, and post it back under this question:



Exercise 2:

To strengthen our modeling skills, we will now explore step by step a single problem:

A technology company provides offerings to its customers. Offerings are of two separate types: products and services. Offerings are identified by an offering ID and an attribute of description. In addition, products are described by product name, standard price, and date of first release; services are described by name of the company's unit responsible for the service and conditions of service.

There are repair, maintenance, and other types of services. A repair service has a cost and is the repair of some product; a maintenance service has an hourly rate. Fortunately, some products never require repair. However, there are many potential repair services for a product.

A customer may purchase an offering, and the company needs to keep track of when the offering was purchased and the contact person for that offering with the customer. Unfortunately, not all offerings are purchased. Customers are identified by customer ID and have descriptive data of name, address, and phone number.

When a service is performed, that service is billed to some customer. Because some customers purchase offerings for their clients, a customer may be billed for services he or she did not purchase, as well as for ones that were purchased. When a customer is billed for a service (although some may never require a service of any type), the company needs to keep track of the date the service was performed, the date the bill is due, and the amount due.

Once more, let us start by repeating the process with carried out during our first lab and:

Q7. Please identify the entities, their attributes, and their relationships as suggested in the problem above:

Entity	**Attributes**	**Relationship to (add name to it)**
Offerings	Offering ID Description	Customers Product Services
Products	Product ID Product name Standard Price Date of first release	Offerings
Services	Name of company's unit Condition of service	Offerings Repair Service

		Maintenance Customer
Repair Service	Cost Repair of some products	Services
Maintenance	Hourly Rate	Services
Customer	Customer_ID Name Address Phone Number	Offerings Services

Q8. Are there any supertype/subtype relationships present? If so, what entities would be involved? Which would be the supertype, and which would be the subtype?

Entity	**Attributes**	**Relationship to (add name to it)**	**Subtype or Supertype? (N/A if not part of Subtype/supertype relationship)**
Offerings	Offering ID Description	Product Service	Supertype
Product	Product ID Product name Standard Price Date of first release	Offerings	Subtype
Services		Name of company's unit Condition of service	Supertype
Repair and Maintenance	Cost Hourly rate	Service	Subtype

Q9. Now consider the completeness constraint. In the supertype/subtype relationship identified above a total specialization rule or a partial specialization rule? What does this mean for this/these relationship(s)? (Refer to part 1 or the class lecture for a refresher on this)

In the supertype/subtype relationship identified above (Offerings as the supertype with subtypes Product and Service), it is a *total specialization rule*.

Total Specialization Rule: Every instance of the supertype must be a member of at least one subtype.

In this context, every offering must be classified as either a product or a service (total specialization for the Offering supertype). This means that there are no instances of the supertype "Offerings" that exist outside the specified subtypes "Product" or "Service." Every offering provided by the technology company must fall into one of these two subtypes, adhering to the total specialization rule.

Q10. Now consider the disjointness Constraints. Is this a disjoint rule or an overlapping rule? What does that mean for this/these relationship (s)? (Refer to part 1 or the class lecture for a refresher on this)

Given the nature of offerings as products or services, *it's likely that a disjoint rule applies*. This means that an offering can be either a product or a service but not both at the same time. Explanation given below

Disjointness and overlapping rules refer to how the subtypes relate to each other.

- *Disjoint Rule:* This rule states that an instance of the supertype can belong to only one of its subtypes, but not to more than one at the same time.
- *Overlapping Rule:* This rule allows instances of the supertype to belong to more than one subtype simultaneously.

For the Offerings supertype with subtypes Product and Service:

- *Disjoint Rule:* It means that an instance of an offering cannot simultaneously be both a product and a service. In other words, an offering must belong exclusively to either the Product subtype or the Service subtype.
- *Overlapping Rule:* This would imply that an instance of the supertype could belong to both the Product and Service subtypes simultaneously, which isn't applicable in this scenario.

And hence, *it's likely that a* ***disjoint rule applies.***

Q11. Now let us consider the subtype discriminator. How would you describe the subtype discriminator for this supertype/subtype relationship? Is it a simple attribute, or a composite attribute?

In the context of the Offerings supertype with subtypes Product and Service, the subtype discriminator is likely to be a ***simple attribute.***

In this case, the subtype discriminator could be a single attribute in the Offerings entity that indicates whether an offering is a product or a service. This attribute, often referred to as a "Type" or "Category," serves as a simple discriminator to classify instances into the appropriate subtype.

For example: Offerings, Offering ID, Description, Type (or Category) [This attribute serves as the subtype discriminator]

If the value of the "Type" attribute is indicative of whether the offering is a product or a service, then it would be considered a simple subtype discriminator. If, however, multiple attributes are required to determine the subtype, it would be a composite subtype discriminator.

Q12. Finally, please use the information above and create a corresponding EERD model. Take a screenshot, and post it back under this question: