Chapter 6: Entity-Relationship Model

Chapter 6: Entity-Relationship Model

- Design Process
- Modeling
- Constraints
- E-R Diagram
- Design Issues
- Weak Entity Sets
- Extended E-R Features
- Design of the Bank Database
- Reduction to Relation Schemas
- Database Design
- UML

Modeling

- A *database* can be modeled as:
  - a collection of entities,
  - relationship among entities.
- An **entity** is an object that exists and is distinguishable from other objects.
  - Example: specific person, company, event, plant
- Entities have *attributes*
  - Example: people have *names* and *addresses*
- An **entity set** is a set of entities of the same type that share the same properties.
  - Example: set of all persons, companies, trees, holidays

Entity Sets *customer* and *loan*

customer_id	customer_name	customer_street	customer_city		loan_number	amount
321-12-3123	Jones	Main	Harrison		L-17	1000
019-28-3746	Smith	North	Rye		L-23	2000
677-89-9011	Hayes	Main	Harrison		L-15	1500
555-55-5555	Jackson	Dupont	Woodside		L-14	1500
244-66-8800	Curry	North	Rye		L-19	500
963-96-3963	Williams	Nassau	Princeton		L-11	900
335-57-7991	Adams	Spring	Pittsfield		L-16	1300

customer   *loan*

Relationship Sets

■ A **relationship** is an association among several entities

Example:

Hayes   *depositor*   A-102
customer entity   relationship set   *account* entity

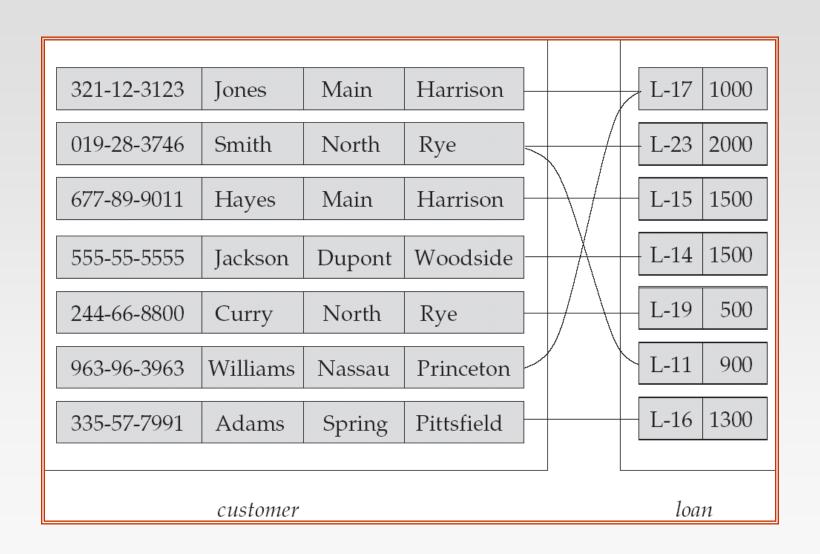■ A **relationship set** is a mathematical relation among $n \geq 2$ entities, each taken from entity sets

$$\{(e_1, e_2, \dots e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

where $(e_1, e_2, \dots, e_n)$ is a relationship
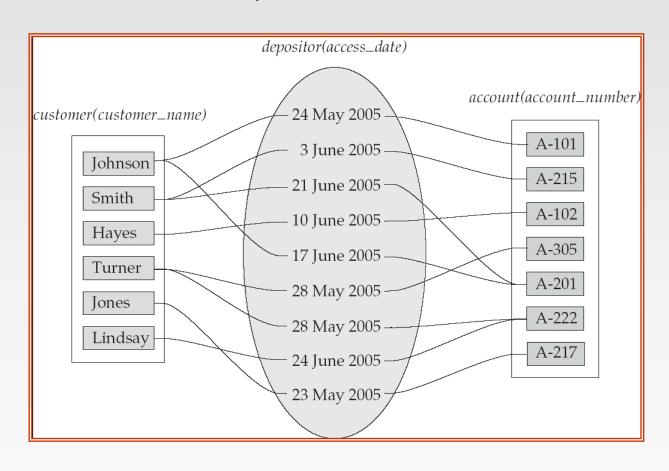
● Example:

(Hayes, A-102) $\in$ *depositor*

Relationship Set *borrower*

Relationship Sets (Cont.)

- An **attribute** can also be property of a relationship set.
- For instance, the *depositor* relationship set between entity sets *customer* and *account* may have the attribute *access-date*
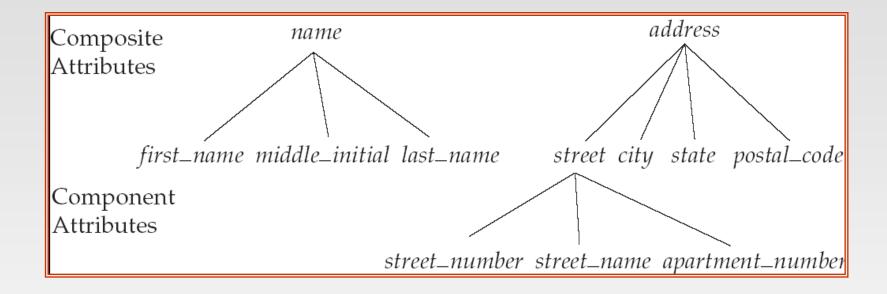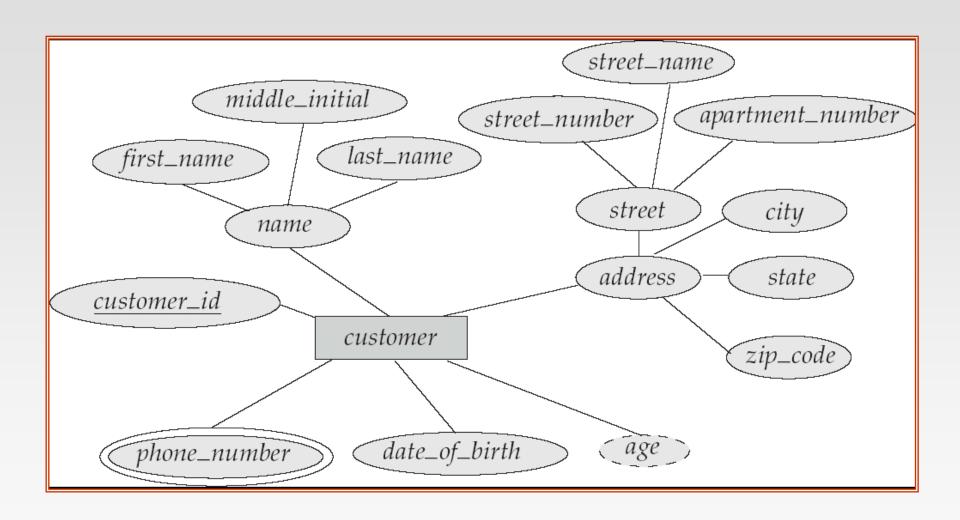
Attributes

■ An entity is represented by a set of attributes, that is descriptive properties possessed by all members of an entity set.
   Example:

$$customer = (customer\_id, customer\_name,$$
$$customer\_street, customer\_city)$$
$$loan = (loan\_number, amount)$$

■ **Domain** – the set of permitted values for each attribute

■ Attribute types:

- *Simple* and *composite* attributes.

- *Single-valued* and *multi-valued* attributes

   ‣ Example: multivalued attribute: *phone_numbers*

- *Derived* attributes

   ‣ Can be computed from other attributes

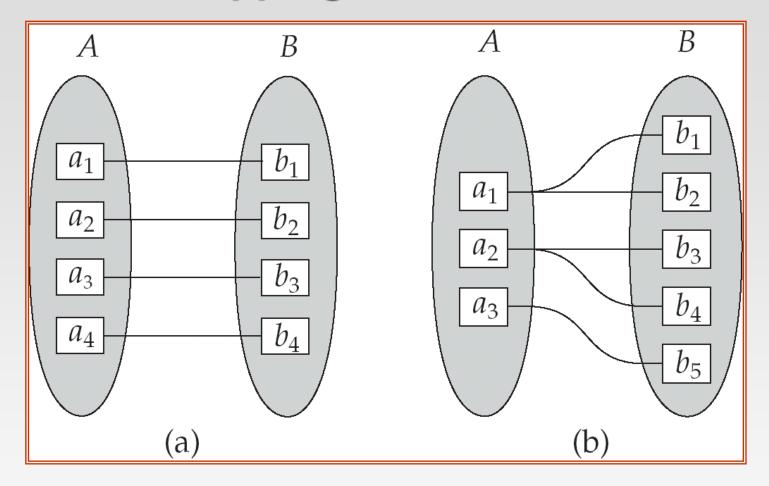   ‣ Example: age, given date_of_birth

Composite Attributes

E-R Diagram With Composite, Multivalued, and Derived Attributes

Mapping Cardinalities

- Express the number of entities to which another entity can be associated via a relationship set.

- Most useful in describing binary relationship sets.

- For a binary relationship set the mapping cardinality must be one of the following types:
  - One to one
  - One to many
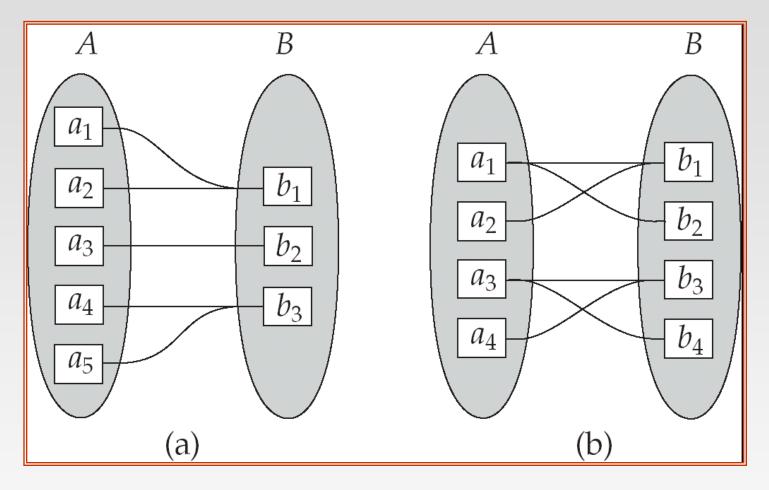 - Many to one
  - Many to many

Mapping Cardinalities

One to one   One to many

Note: Some elements in *A* and *B* may not be mapped to any elements in the other set
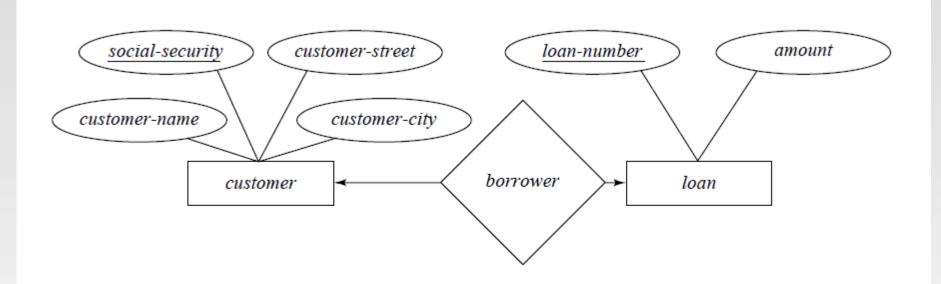
Mapping Cardinalities

Many to one

Many to many

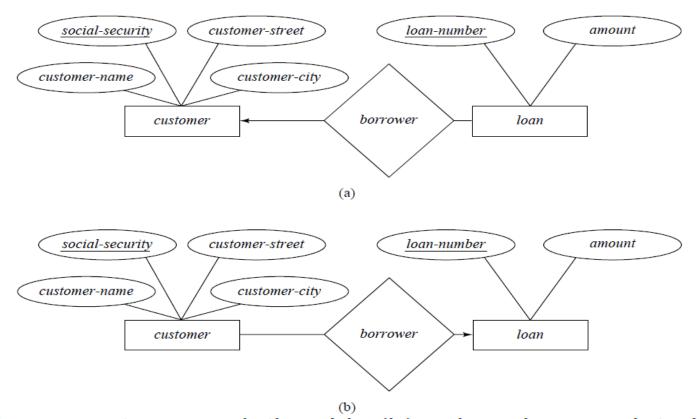Note: Some elements in A and B may not be mapped to any elements in the other set

One-To-One Relationship

- A customer is associated with at most one loan via the relationship *borrower*

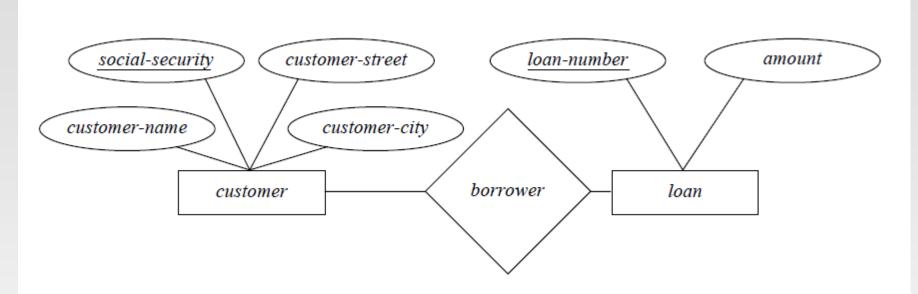- A loan is associated with at most one customer via *borrower*

One-To-Many and Many-to-One Relationships

- In the one-to-many relationship (a), a loan is associated with at most one customer via *borrower*; a customer is associated with several (including 0) loans via *borrower*
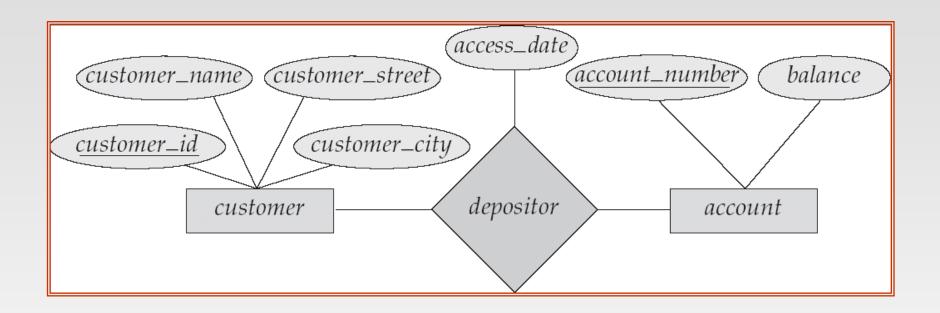

(a)

(b)

- In the many-to-one relationship (b), a loan is associated with several (including 0) customers via *borrower*; a customer is associated with at most one loan via *borrower*

Many-To-Many Relationship

- A customer is associated with several (possibly 0) loans via borrower

- A loan is associated with several (possibly 0) customers via borrower
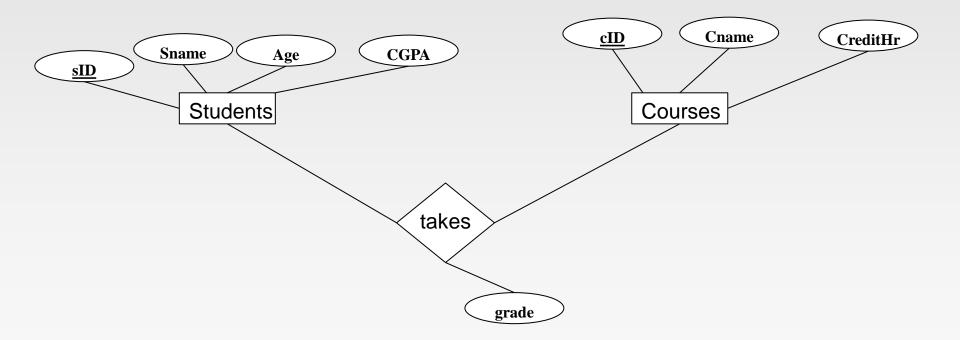
Relationship Sets with Attributes

E-R Diagrams

- An ER diagram is useful in mapping the real world things into a conceptual schema.

- It has following major components:

- ☐ (Rectangle) : represent entity set.

- ⬭ (Ellipse) : represent attribute
  - ⬭ (Double ellipses) : represent multivalued attribute.
  - ⬭ (Dashed ellipse) : denote derived attributes.
  - ⬭ : Underline indicates primary key attributes

- ◇ (Diamond) : represent relationship set.

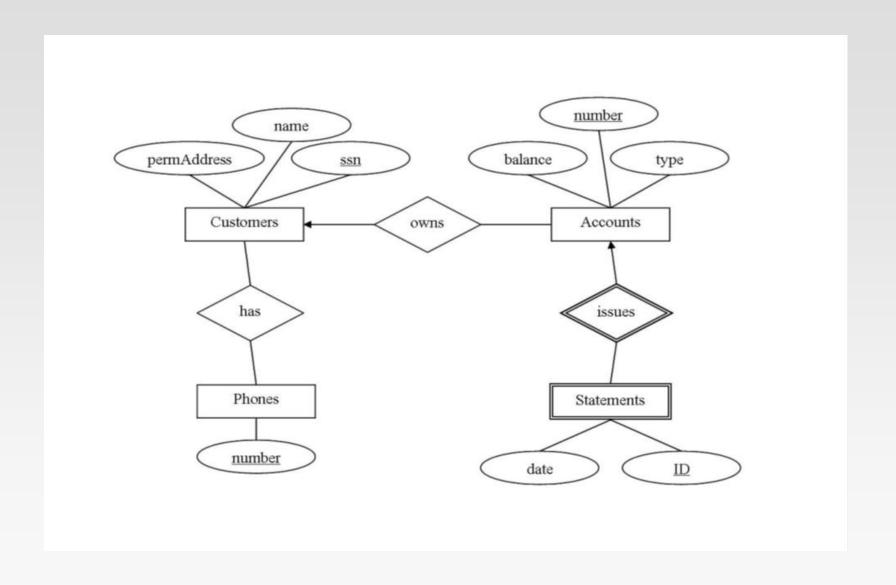- ——— (Line) : link attributes to entity sets and entity sets to relationship sets.

Example 1

➤ Draw an ER diagram to model the application with the following assumptions:

1) Each student has an ID, name, age and CGPA
2) Each course has an ID, name, Credit hours
3) Students in courses receive a grade.

Example 2

➢ Draw an ER diagram to model the application with the following assumptions:

1) Each customer has a name, a permanent address, and a social security number.
2) Each customer can have multiple phone numbers, and the same phone number may be shared by multiple customers.
3) A customer can own multiple accounts, but each account is owned by a single customer.
4) Each account has an account number, a type (such as saving, checking, etc), and a balance.
5) The bank issues an account statement for each account and mails it to its account owner every month. As time goes on, there will be multiple statements of the same account.
6) Each statement has an issued date and a statement ID. All the statements of the same account have different statement IDs, but two different accounts could have statements with the same statement ID. For example, it is possible that account A has a statement with ID `123', while account B has another statement with the same ID `123'.

ER example

Classification of Entity sets

■ There are two types of entity sets. They are-
- Strong Entity Set
- Weak Entity Set

Strong Entity Set
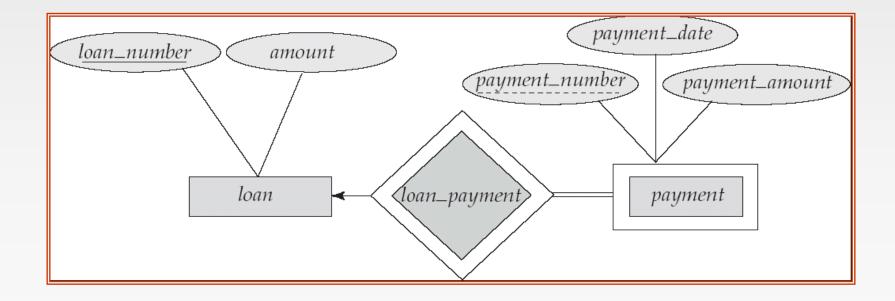
- Its existence doesn't depend on other entity.
- It has a primary key.

Weak Entity Sets

- An entity set that does not have a primary key is referred to as a **weak entity set**.

- The existence of a weak entity set depends on the existence of a **identifying entity set**

  - it must relate to the identifying entity set via a total, one-to-many relationship set from the identifying to the weak entity set

  - Identifying relationship depicted using a double diamond

- The **discriminator** (*or partial key)* of a weak entity set is the set of attributes that distinguishes among all the entities of a weak entity set.

- The primary key of a weak entity set is formed by the primary key of the strong entity set on which the weak entity set is existence dependent, plus the weak entity set's discriminator.

Weak Entity Sets (Cont.)

- We depict a weak entity set by double rectangles.

- We underline the discriminator of a weak entity set with a dashed line.

- payment_number – discriminator of the *payment* entity set

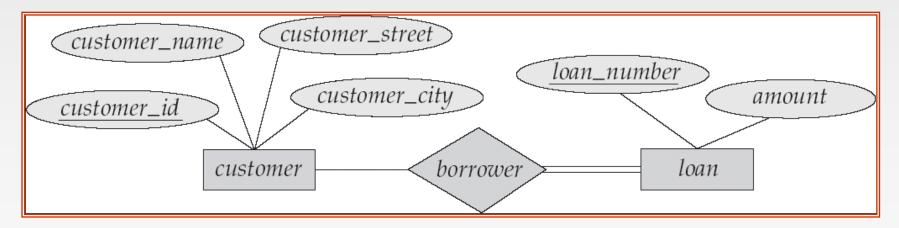- Primary key for *payment* – (*loan_number, payment_number*)

More Weak Entity Set Examples

- In a university, a *course* is a strong entity and a *course_offering* can be modeled as a weak entity

- The discriminator of *course_offering* would be *semester* (including year) and *section_number* (if there is more than one section)

- If we model *course_offering* as a strong entity we would model *course_number* as an attribute.

  Then the relationship with *course* would be implicit in the *course_number* attribute
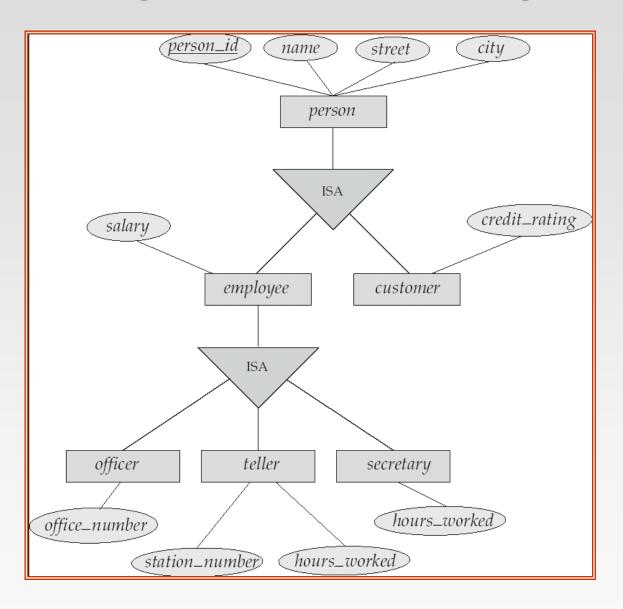
Participation of an Entity Set in a Relationship Set

- **Total participation** (*indicated by double line*): every entity in the entity set participates in at least one relationship in the relationship set
  - E.g. participation of loan in borrower is total
    - every loan must have a customer associated to it via borrower
- **Partial participation**: some entities may not participate in any relationship in the relationship set
  - Example: participation of customer in borrower is partial

Extended E-R Features: Specialization

- Top-down design process;

- We create subgroups within an entity set that are distinctive from other entities in the set -*this process is known as specialization.*

- These subgroups become lower-level entity sets and may have attributes or participate in relationships that do not apply to the higher-level entity set.

- A *triangle* component labeled ISA (ISA) is used to show specialization.

- ISA stands for "is a"(E.g. *customer* "is a" *person*).

- **Attribute inheritance** – a lower-level entity set inherits all the attributes and relationship participation of the higher-level entity set to which it is linked.

- The *ISA relationship* is also known as "*superclass-subclass*" relationship.
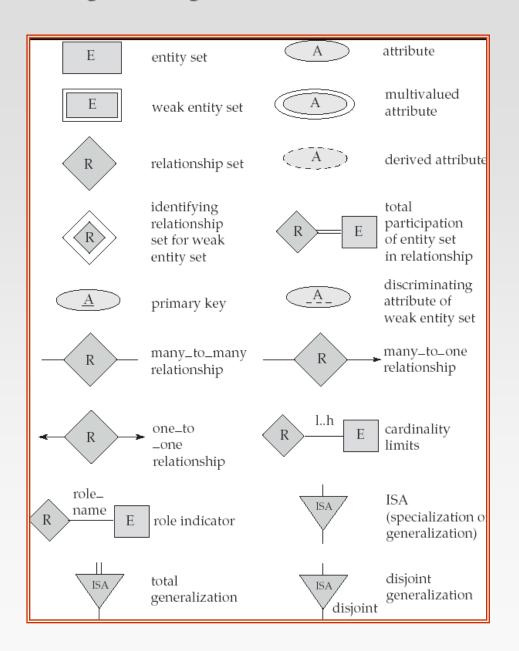
Specialization Example

Extended ER Features: Generalization

- **A bottom-up design process**

- Here, multiple entity sets are combined into a higher-level entity set on the basis of common features - *this process is known as generalization* .

- Specialization and generalization are simple inversions of each other and they are represented in an E-R diagram in the same way.

- The terms specialization and generalization are used interchangeably.

Summary of Symbols Used in E-R Notation

Summary of Symbols (Cont.)