Entity-Relationship Model

Entity – Relationship Diagram

By the end of this session you will be able to

- ***Skilfully illustrate with the help of an entity – relationship diagram the entities and their relationships involved in a given system***
- ***Properly apply the concepts of aggregation, specialisation and generalisation to an ER Diagram.***

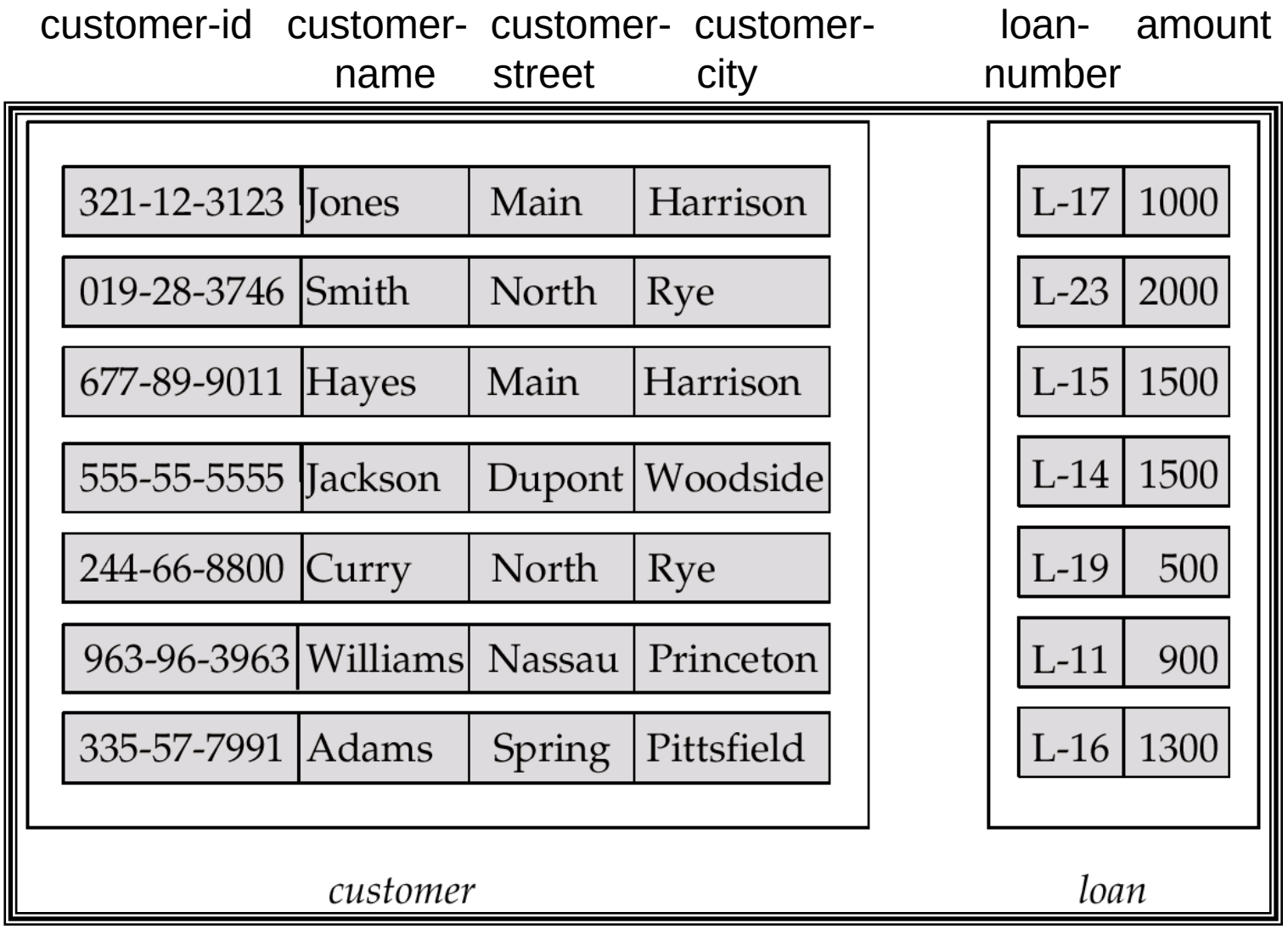
Agenda

- ***Entity Sets***
- ***Attributes***
- ***Relationship Sets***
- ***Mapping Cardinalities***
- ***Roles***
- ***Keys***
- ***Weak Entity Sets***
- ***Specialization***
- ***Generalization***
- ***Aggregation***

Entity Sets

- 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

customer-id	customer-name	customer-street	customer-city
321-12-3123	Jones	Main	Harrison
019-28-3746	Smith	North	Rye
677-89-9011	Hayes	Main	Harrison
555-55-5555	Jackson	Dupont	Woodside
244-66-8800	Curry	North	Rye
963-96-3963	Williams	Nassau	Princeton
335-57-7991	Adams	Spring	Pittsfield

customer

loan-number | amount

loan-number	amount
L-17	1000
L-23	2000
L-15	1500
L-14	1500
L-19	500
L-11	900
L-16	1300

loan

Attributes

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

  Example:

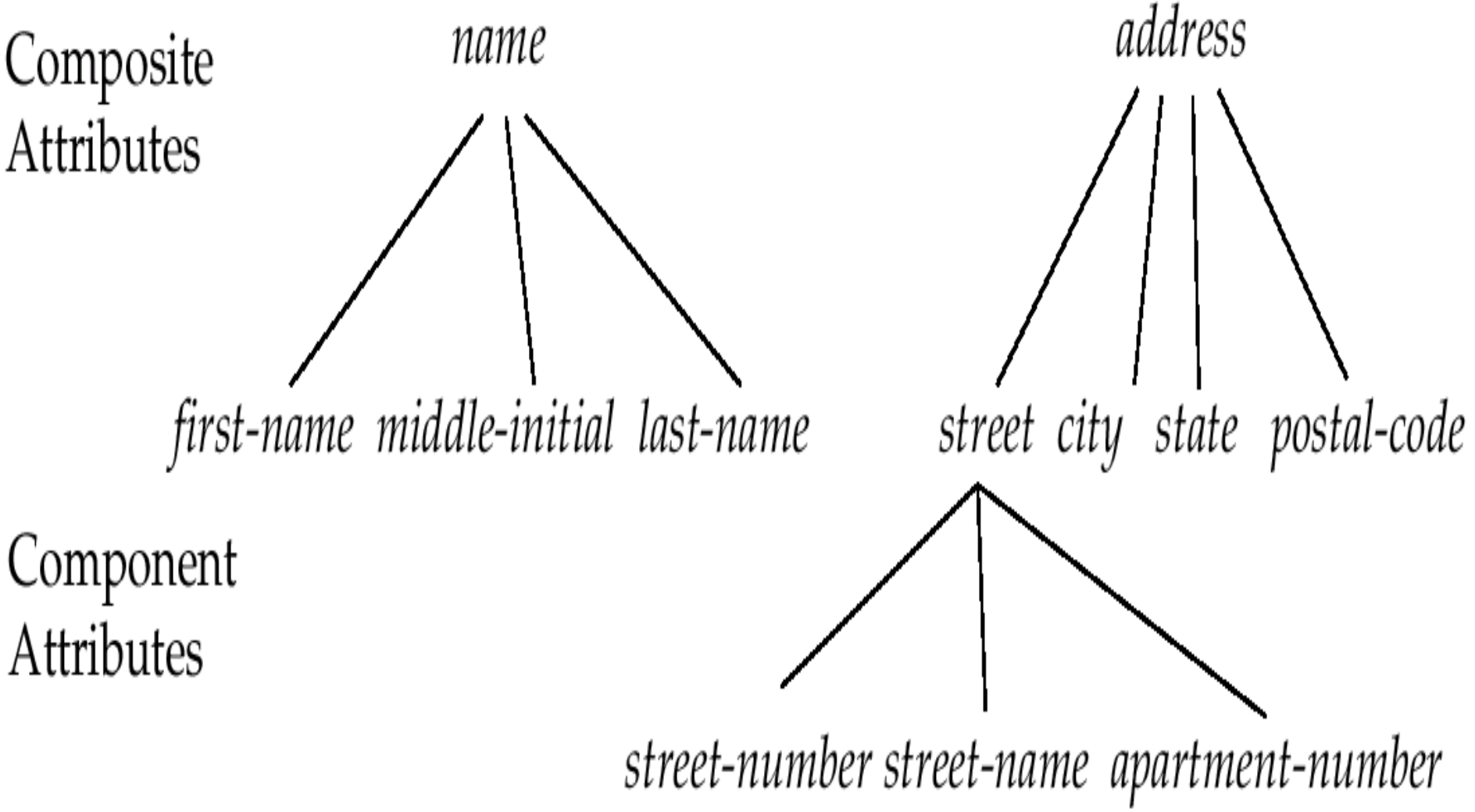
  customer = (customer-id, customer-name, customer-street, customer-city)
  loan = (loan-number, amount)

- *Domain* – the set of permitted values for each attribute
- Attribute types:
  - *Simple* and *composite* attributes.
  - *Single-valued* and *multi-valued* attributes
    - E.g. multivalued attribute: *phone-numbers*
  - *Derived* attributes
    - Can be computed from other attributes
    - E.g. *age*, given date of birth

Types of Attributes

- **Simple**
  - Each entity has a single atomic value for the attribute. For example, SSN or Sex.
- **Composite**
  - The attribute may be composed of several components. For example, Address (Apt#, House#, Street, City, State, ZipCode, Country) or Name (FirstName, MiddleName, LastName). Composition may form a hierarchy where some components are themselves composite.
- **Multi-valued**
  - An entity may have multiple values for that attribute. For example, Color of a CAR or Previous Degrees of a STUDENT. Denoted as {Color} or {PreviousDegrees}.

Composite Attributes



Relationship Sets

- A relationship is an association among several entities

  Example:

  Hayes — *customer* entity; *depositor* — relationship; A-102 — *account* entity

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

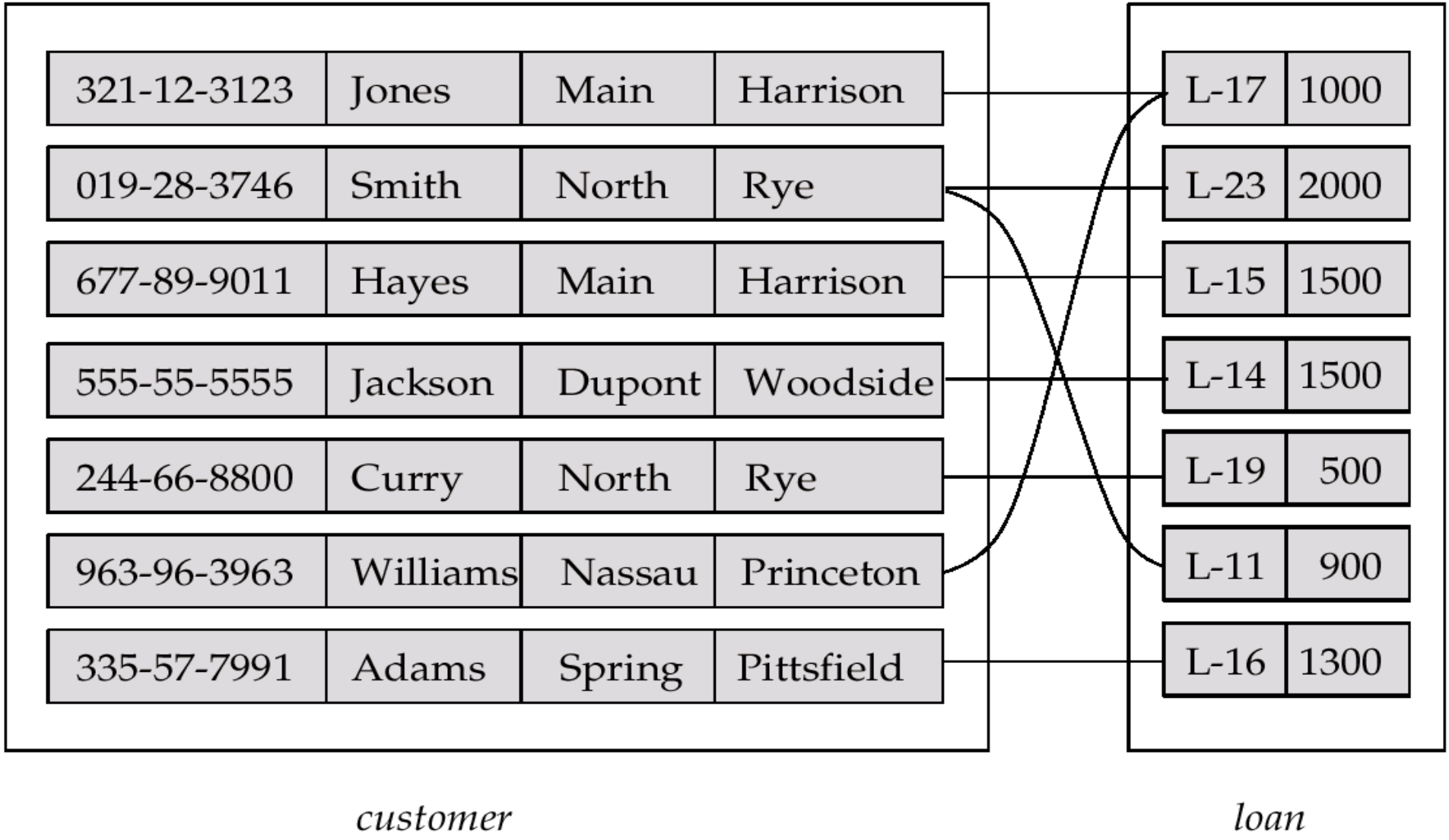
$$\{(e_1, e_2, \ldots e_n) \mid e_1 \in E_1, e_2 \in E_2, \ldots, e_n \in E_n\}$$

  where $(e_1, e_2, \ldots, e_n)$ is a relationship, E1, E2, .. En are entity sets

  – Example:

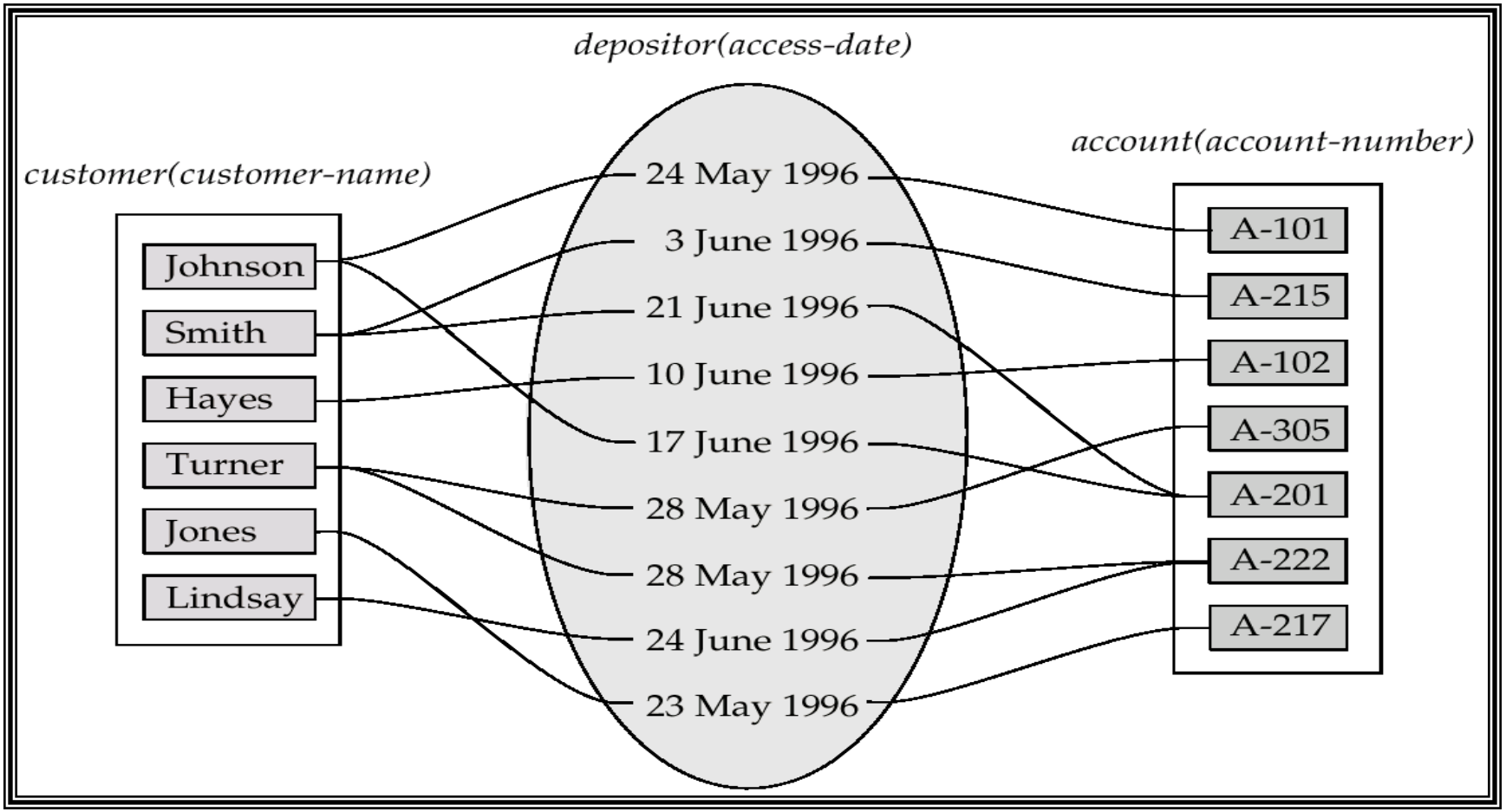
  (Hayes, A-102) $\in$ *depositor*

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*



Degree of a Relationship Set

- Refers to number of entity sets that participate in a relationship set.
- Relationship sets that involve two entity sets are *binary* (or degree two). Generally, most relationship sets in a database system are binary.
- Relationship sets may involve more than two entity sets.

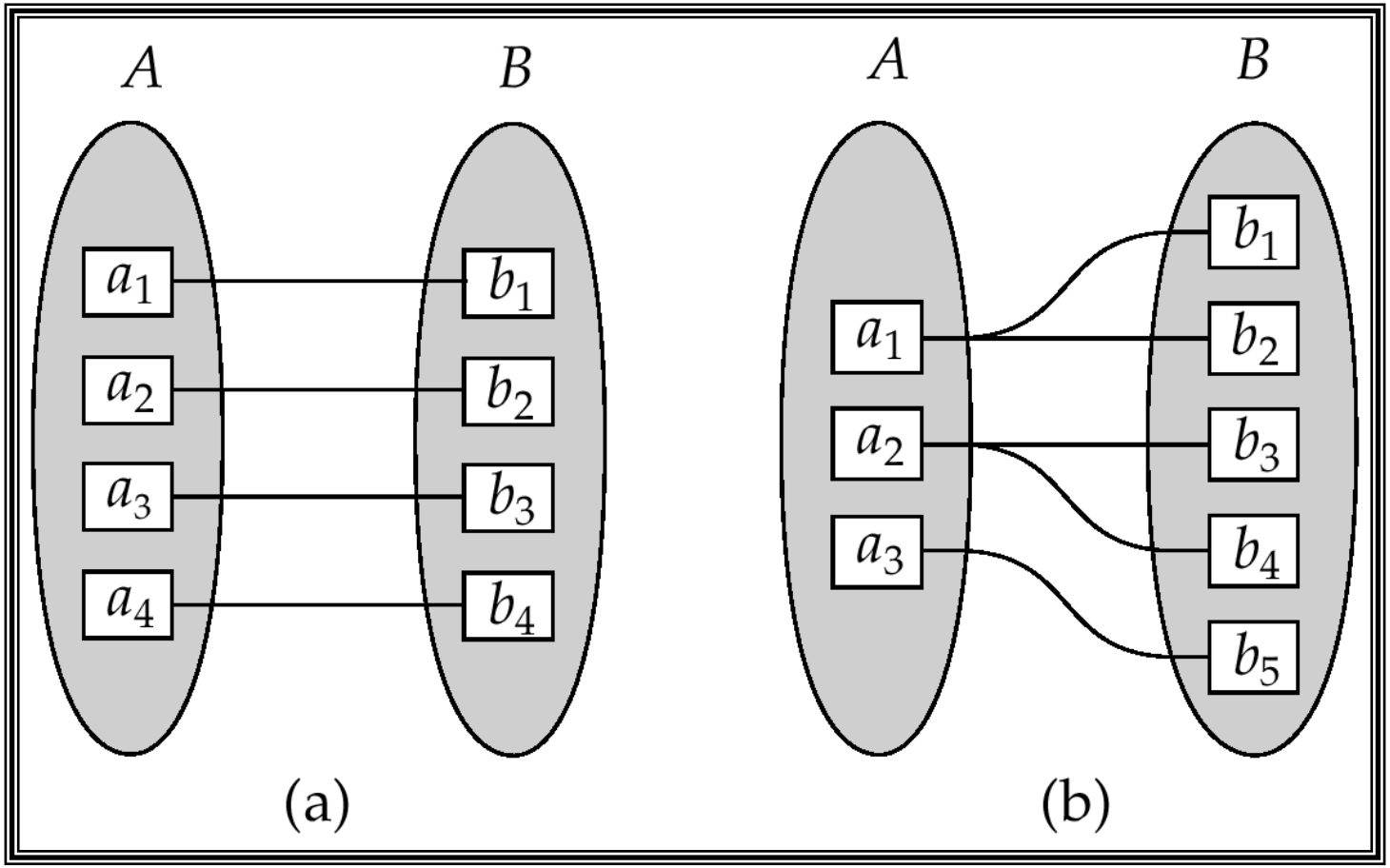
  E.g. Suppose employees of a bank may have jobs (responsibilities) at multiple branches, with different jobs at different branches. Then there is a ternary relationship set between entity sets *employee, job and branch*

- Relationships between more than two entity sets are rare. Most relationships are binary.

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

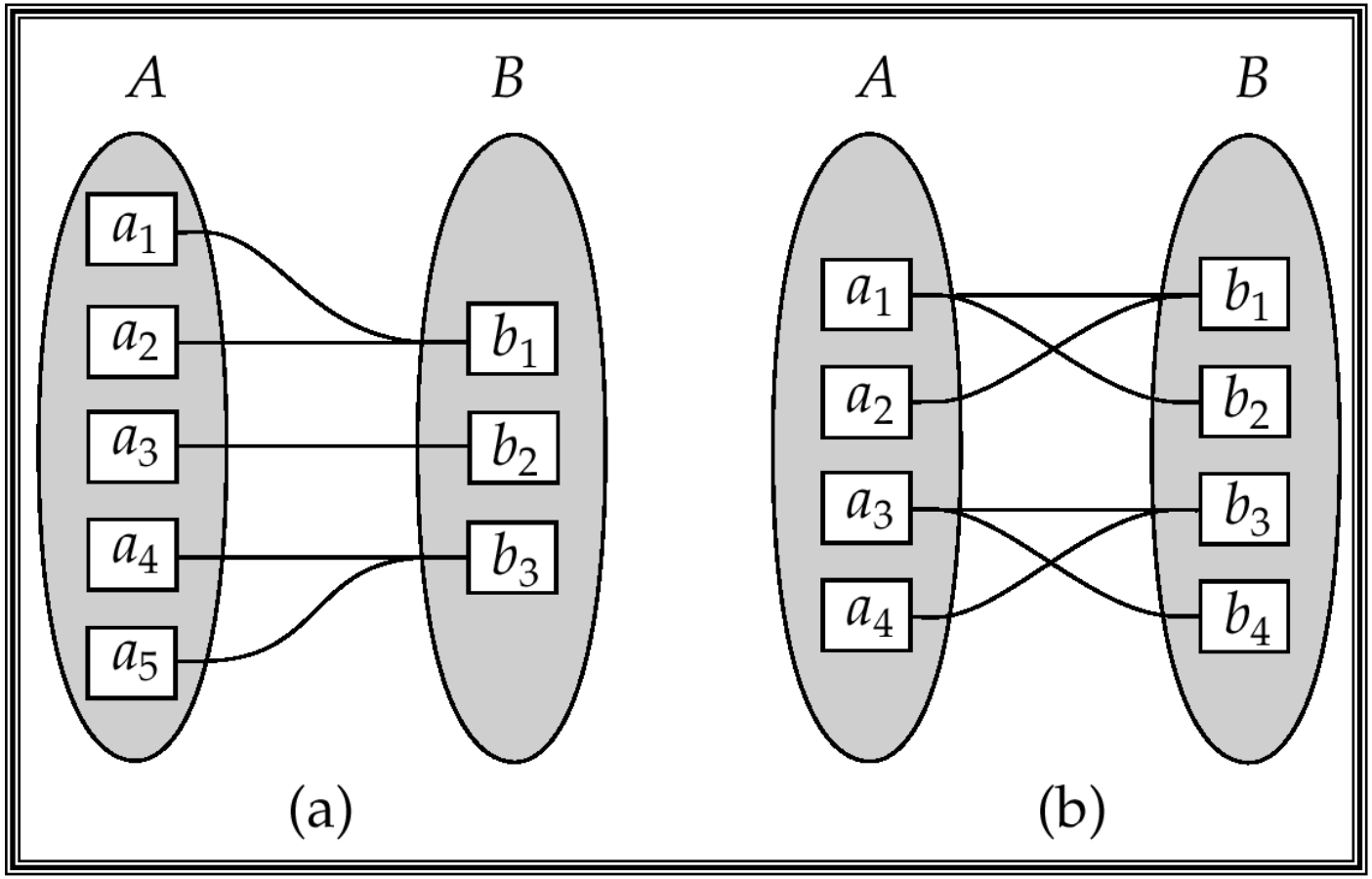


One to one

One to many

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

Mapping Cardinalities

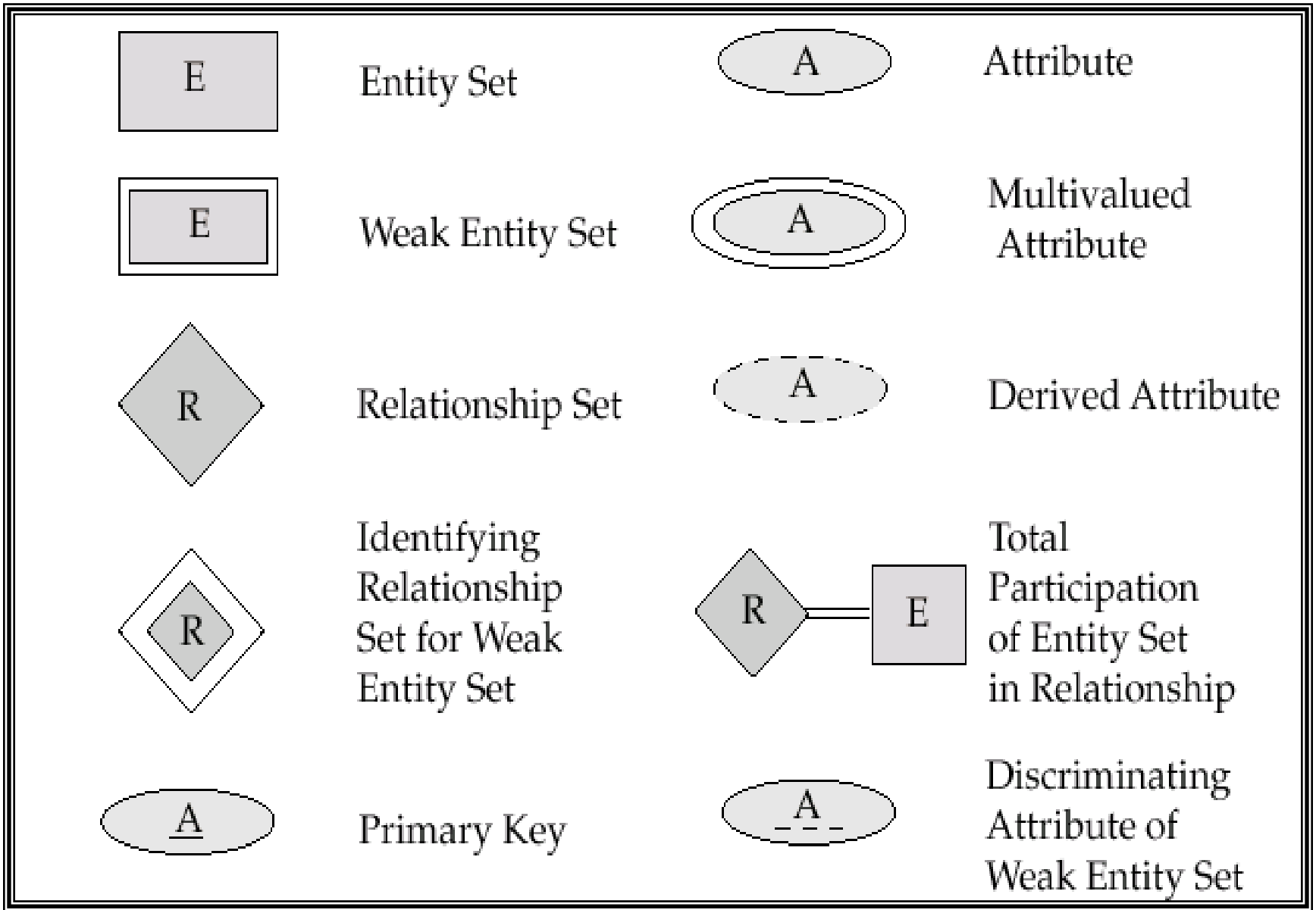


Many to one

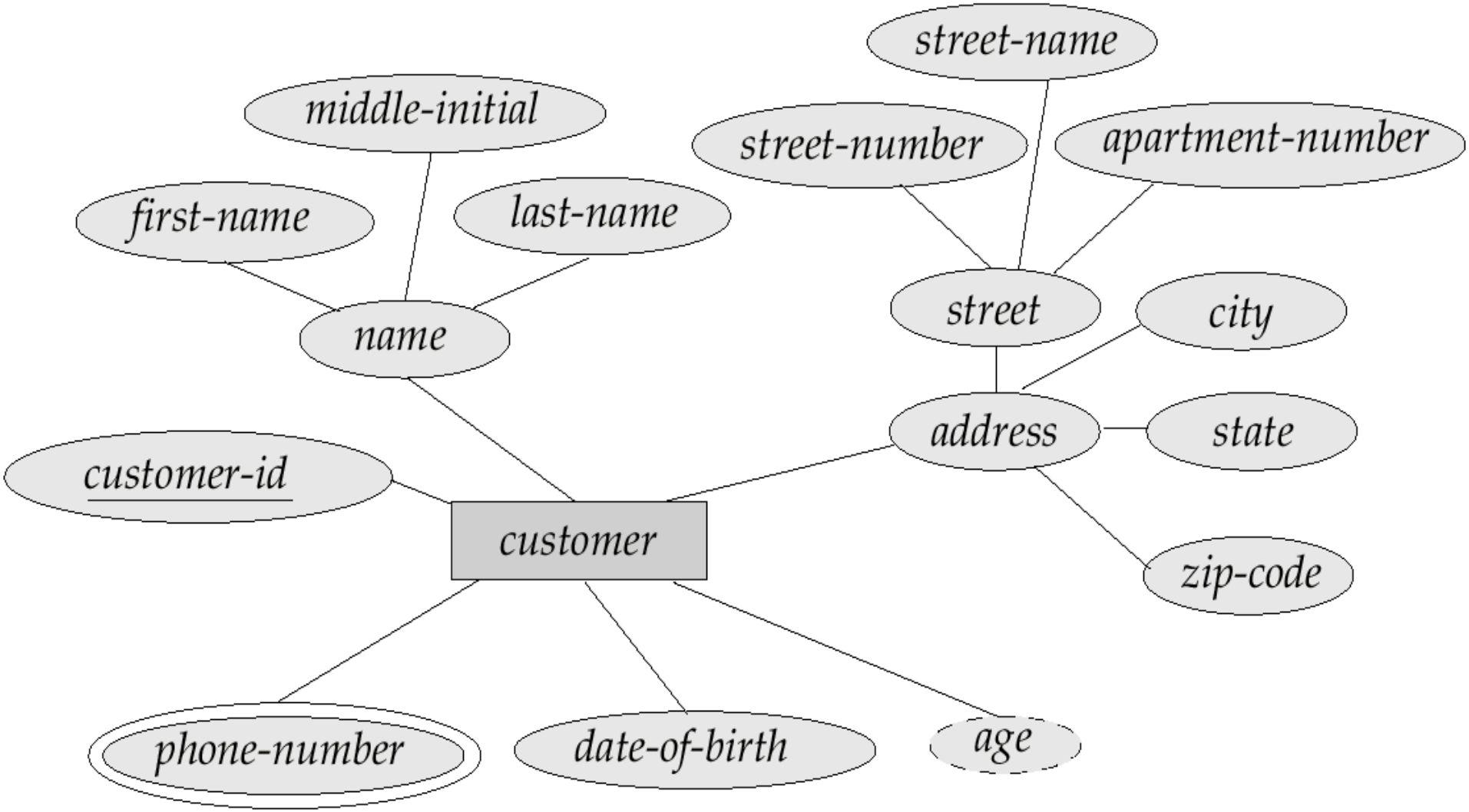
Many to many

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

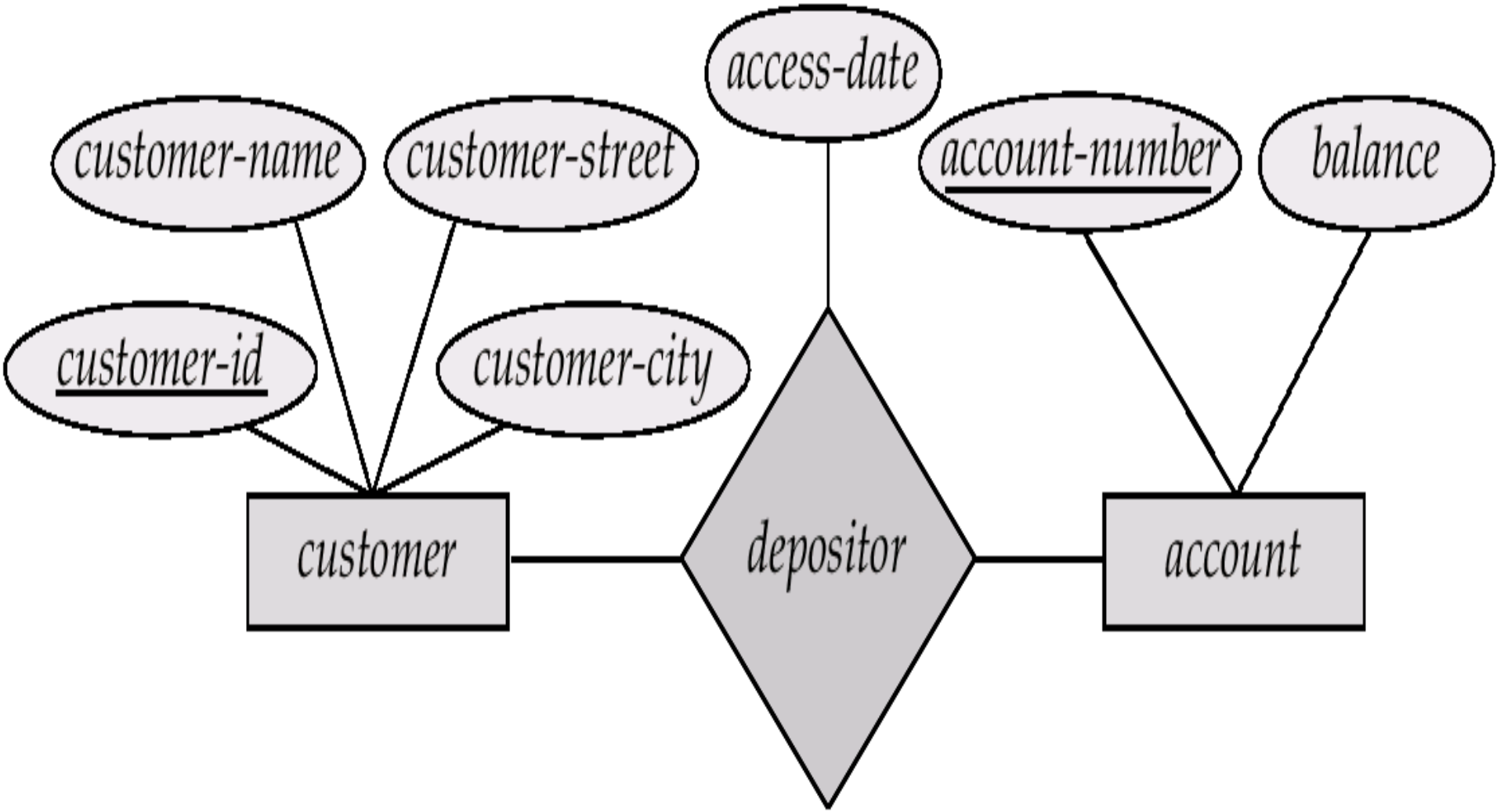
Summary of Symbols Used in E-R Notation



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

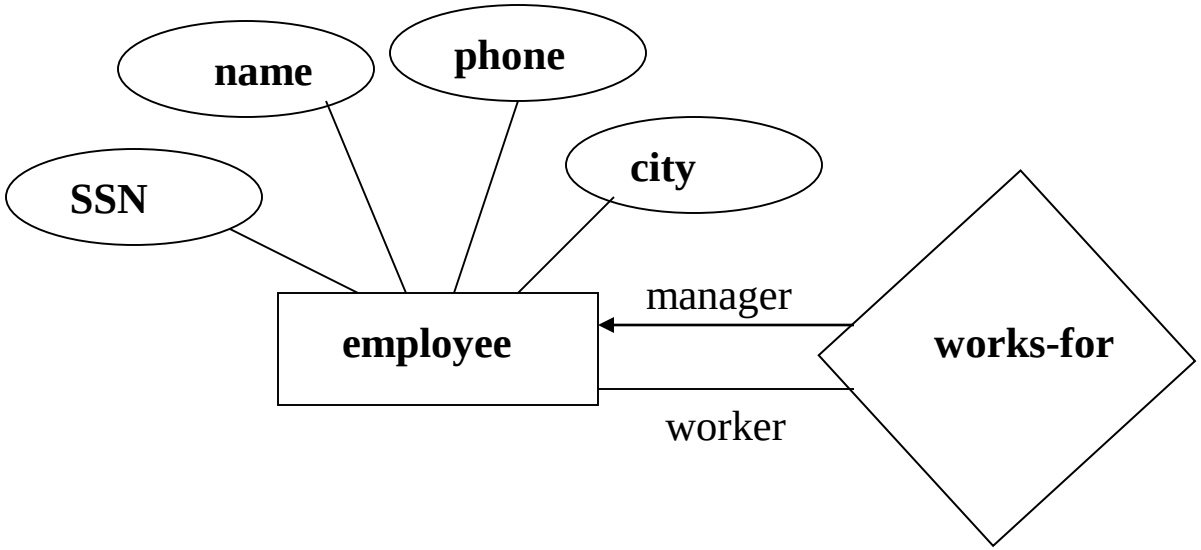


Relationship Sets with Attributes



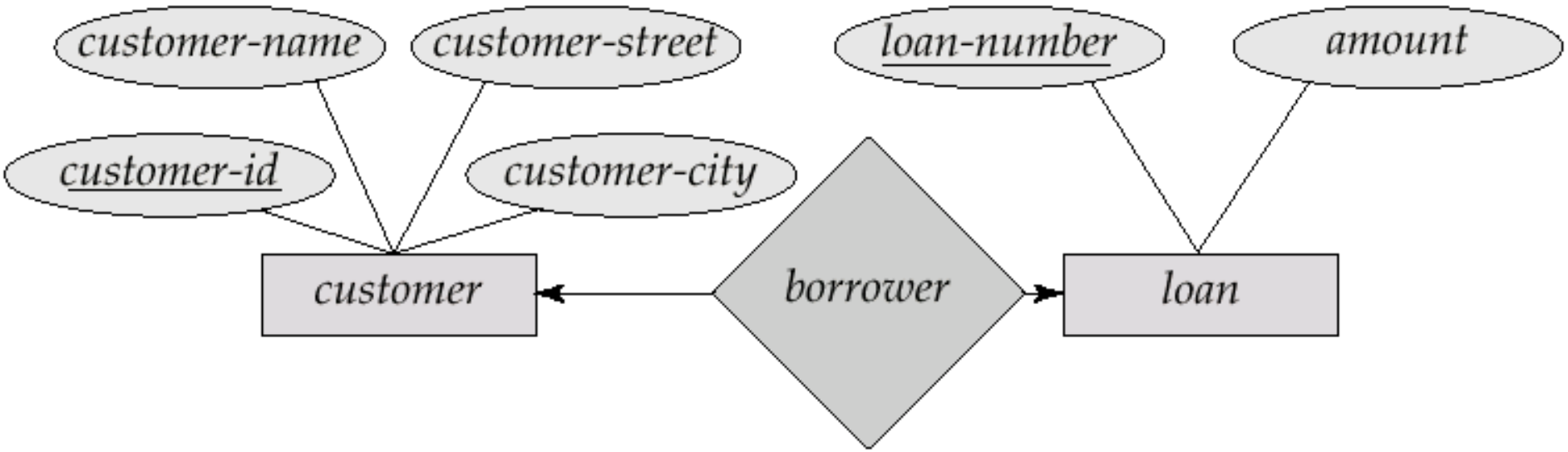
Roles

- Entity sets of a relationship need not be distinct
- Roles are indicated in E-R diagrams by labeling the lines that connect diamonds to rectangles.
- Role labels are optional, and are used to clarify semantics of the relationship
  - E.g. The labels “manager” and “worker” are called roles; they specify how employee entities interact via the works-for relationship set.



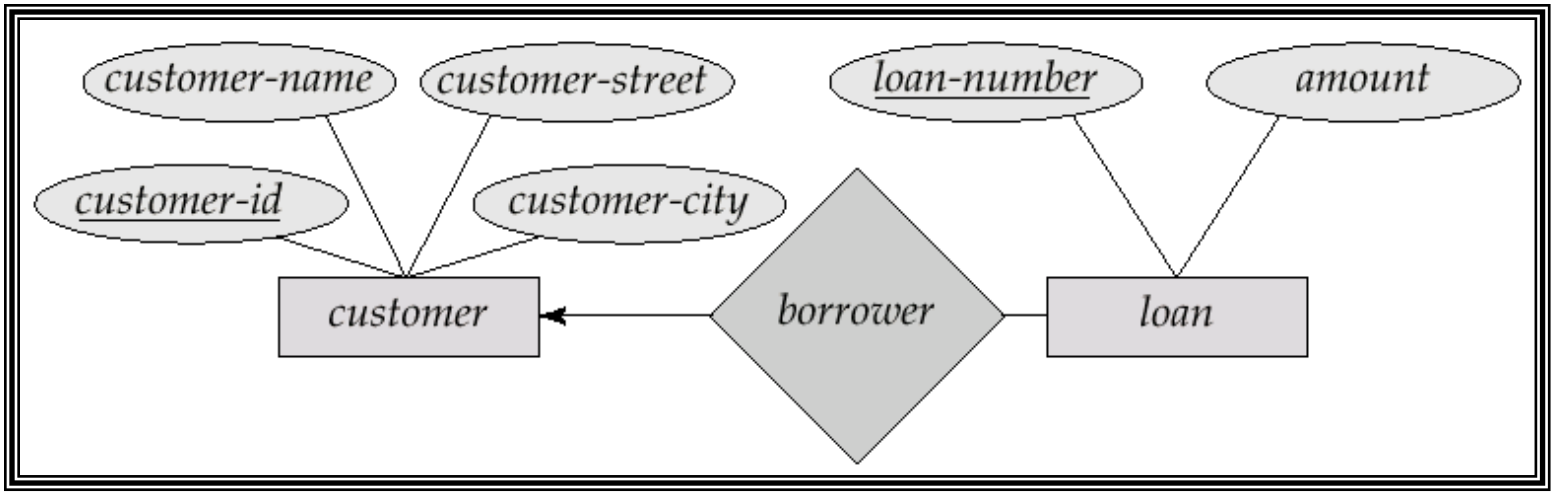
Cardinality Constraints

- We express cardinality constraints by drawing either a directed line ($\rightarrow$), signifying "one," or an undirected line (—), signifying "many," between the relationship set and the entity set.
- E.g.: 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*



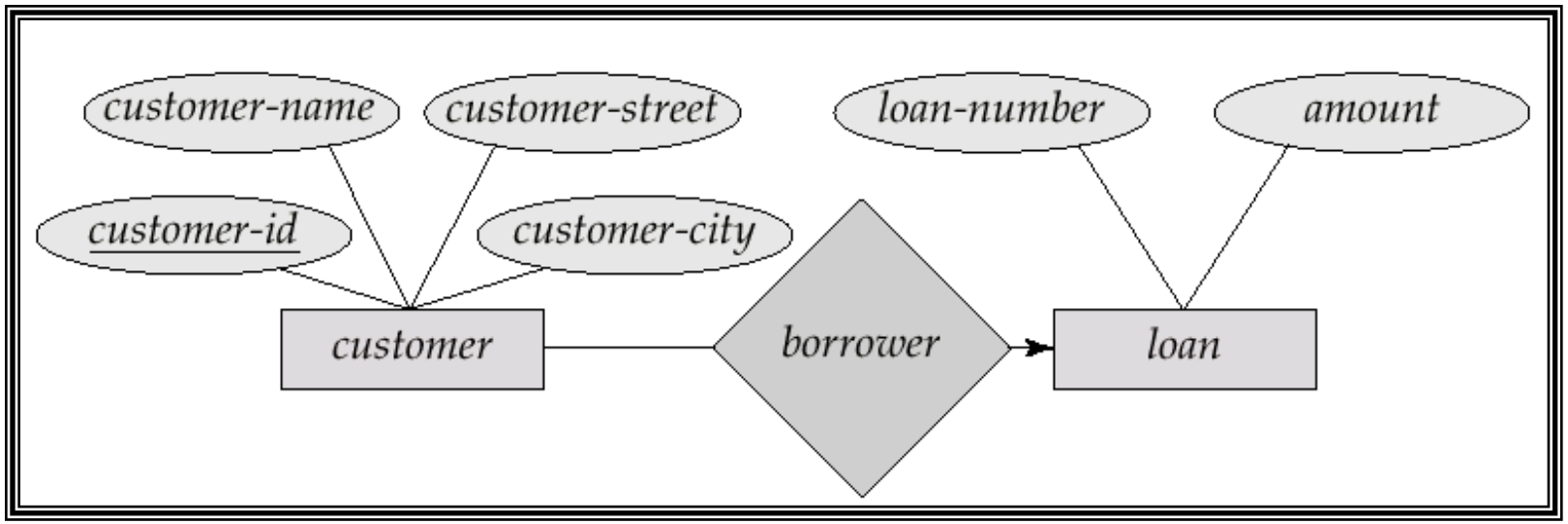
One-To-Many Relationship

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

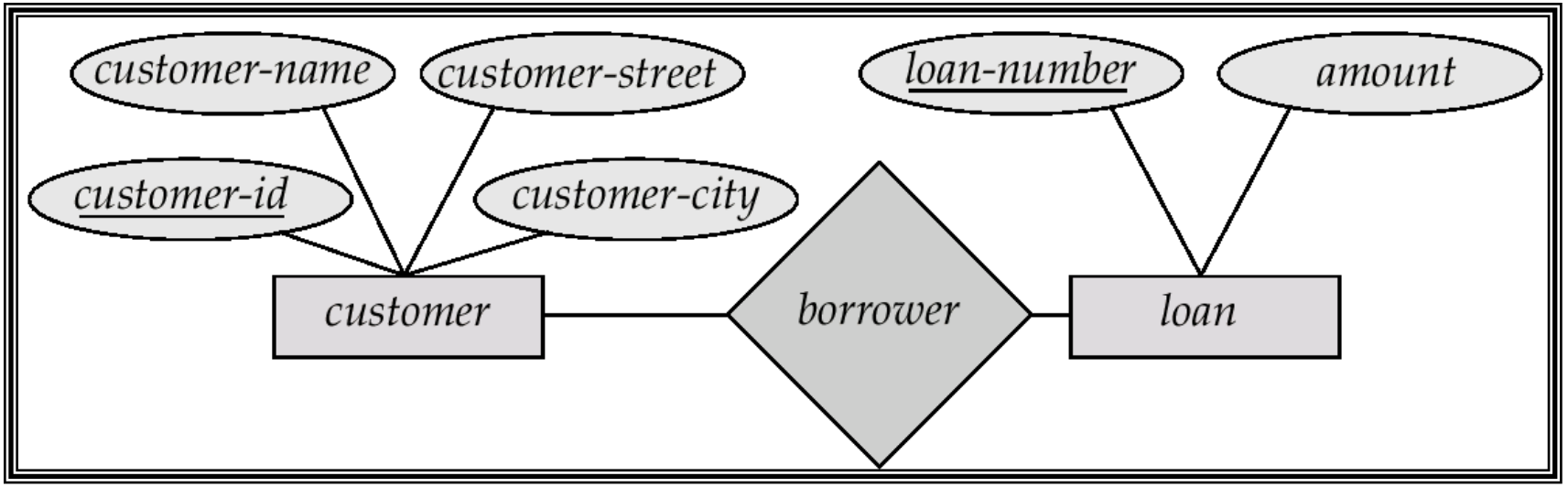


Many-To-One Relationships

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



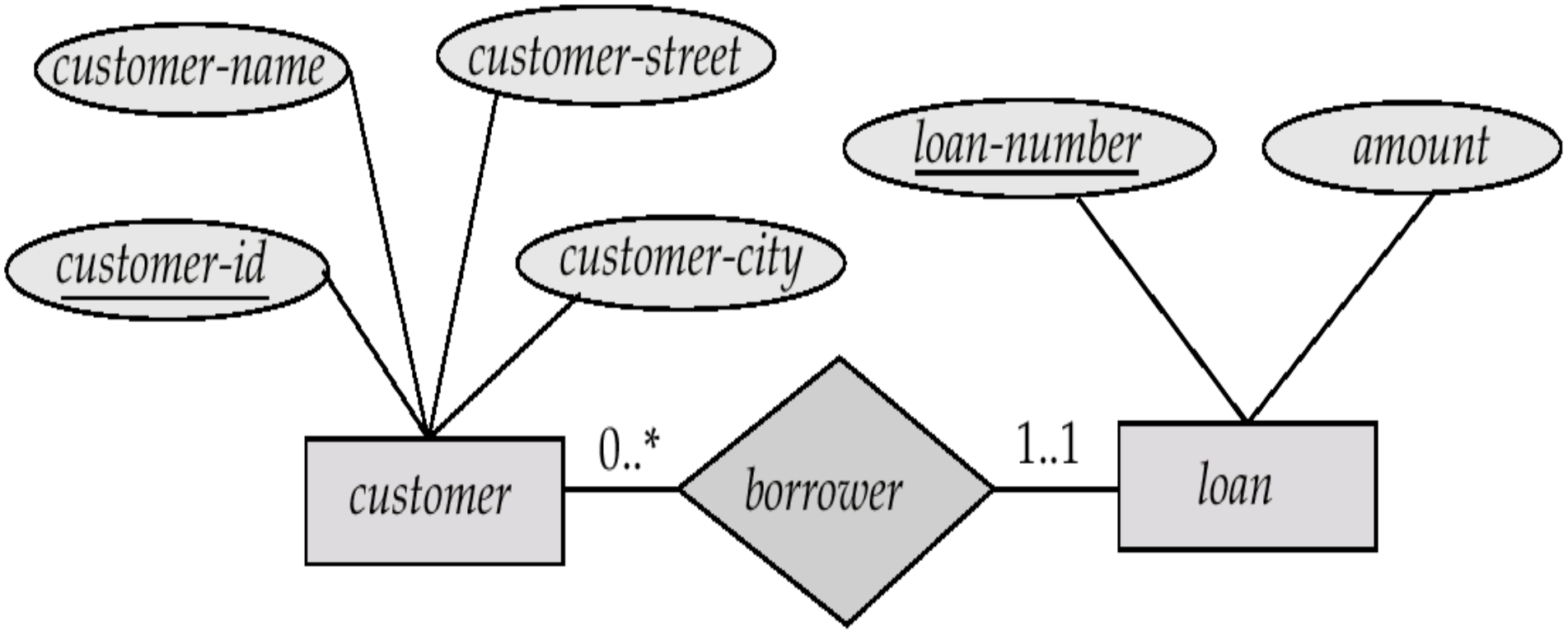
Many-To-Many Relationship



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

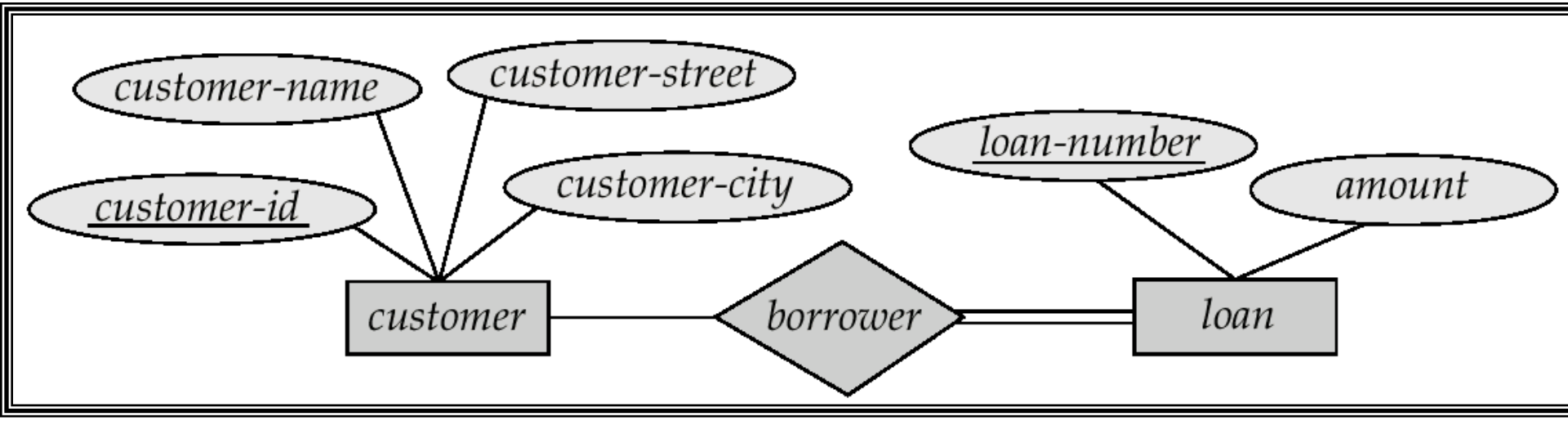
Alternative Notation for Cardinality Limits

- Cardinality limits can also express participation constraints



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
  - E.g. participation of *customer* in *borrower* is partial



Another Example

- If every entity in a entity set has at least one relation in a relationship set then it is known as “total” participation.
- Ex: Employee entity set , salary-paid relationship set, salary-code entity set
- If only some of the entities have relations in a relationship set then it is known as “partial” participation.
- Ex: Employee entity set, on-leave relationship set, dates entity set

Keys

- A *super key* of an entity set is a set of one or more attributes whose values uniquely determine each entity.

e.g a combination of *customer-name* and *SSN* is a super key for the entity customer

- A *candidate key* of an entity set is a minimal super key
  - *SSN* is candidate key of *customer*
  - *account-number* is candidate key of *account*
- Although several candidate keys may exist, one of the candidate keys is selected to be the *primary key*.

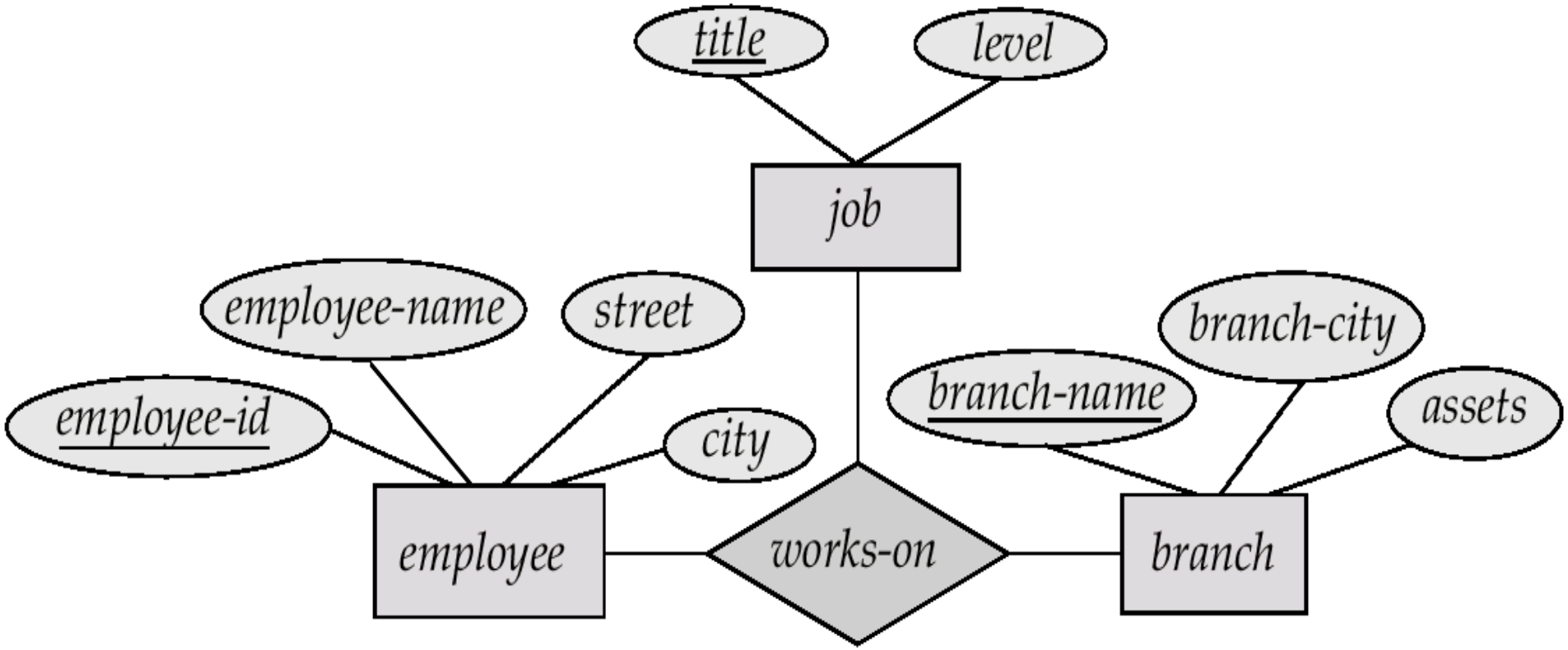
Primary key of a relationship set

- The structure of the primary key for the relationship set depends on the mapping cardinality of the relationship set.
- If the relationship set **R** is **many**-to-**many**, then the primary key of **R** consists of the union of the primary keys of **both** entity sets.
- If the relationship set **R** is **many**-to-**one**, then the primary key of **R** is the primary keys of the entity set which is on the **many** side.
- If the relationship set **R** is **one**-to-**one**, then the primary key of **R** can be **either** of the two primary keys.

Relationships of Higher Degree

- Relationship types of degree 2 are called **binary**
- Relationship types of degree 3 are called **ternary** and of degree n are called **n-ary**
- In general, an n-ary relationship *is not* equivalent to n binary relationships

E-R Diagram with a Ternary Relationship

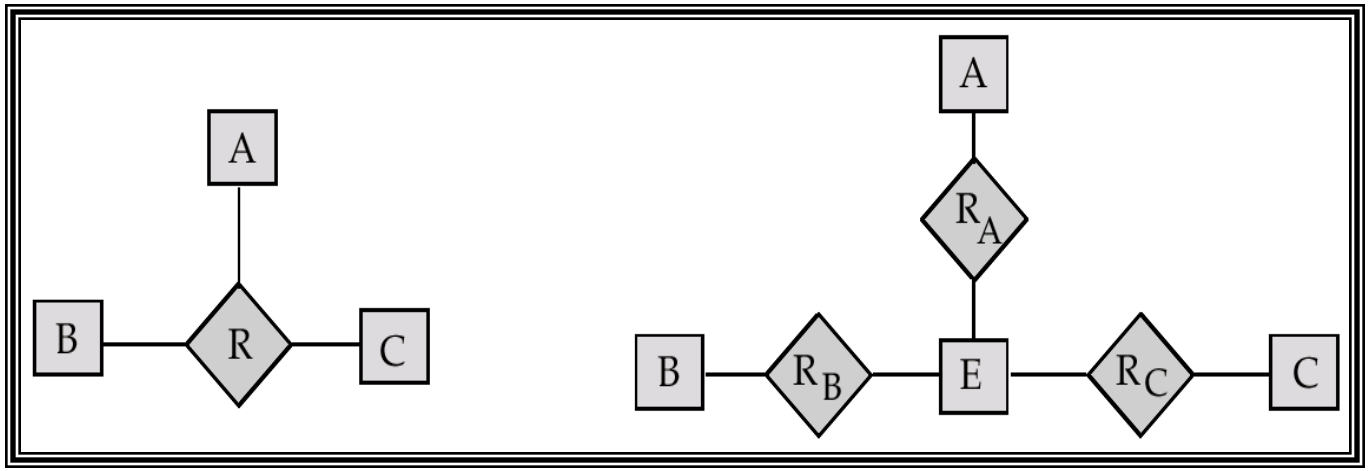


Cardinality Constraints on Ternary Relationship

- We allow at most one arrow out of a ternary (or greater degree) relationship to indicate a cardinality constraint
- E.g. an arrow from *works-on* to *job* indicates each employee works on at most one job at any branch.
- If there is more than one arrow, there are two ways of defining the meaning.
  - E.g a ternary relationship *R* between *A*, *B* and *C* with arrows to *B* and *C* could mean
    1. each *A* entity is associated with a unique entity from *B* and *C* or
    2. each pair of entities from (*A, B*) is associated with a unique *C* entity, and
    3. each pair (*A, C*) is associated with a unique *B*
  - Each alternative has been used in different formalisms
  - To avoid confusion we outlaw more than one arrow

Converting Non-Binary Relationships to Binary Form

- In general, any non-binary relationship can be represented using binary relationships by creating an artificial entity set.
  - Replace *R* between entity sets A, B and C by an entity set *E*, and three relationship sets:
    1. R_A, relating *E* and *A*
    2. R_B, relating *E* and *B*
    3. R_C, relating *E* and *C*
  - Create a special identifying attribute for *E*
  - Add any attributes of *R* to *E*
  -



Weak Entity Sets

- An entity set that has a primary key is referred to as a *strong entity set*.
- 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 an *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.

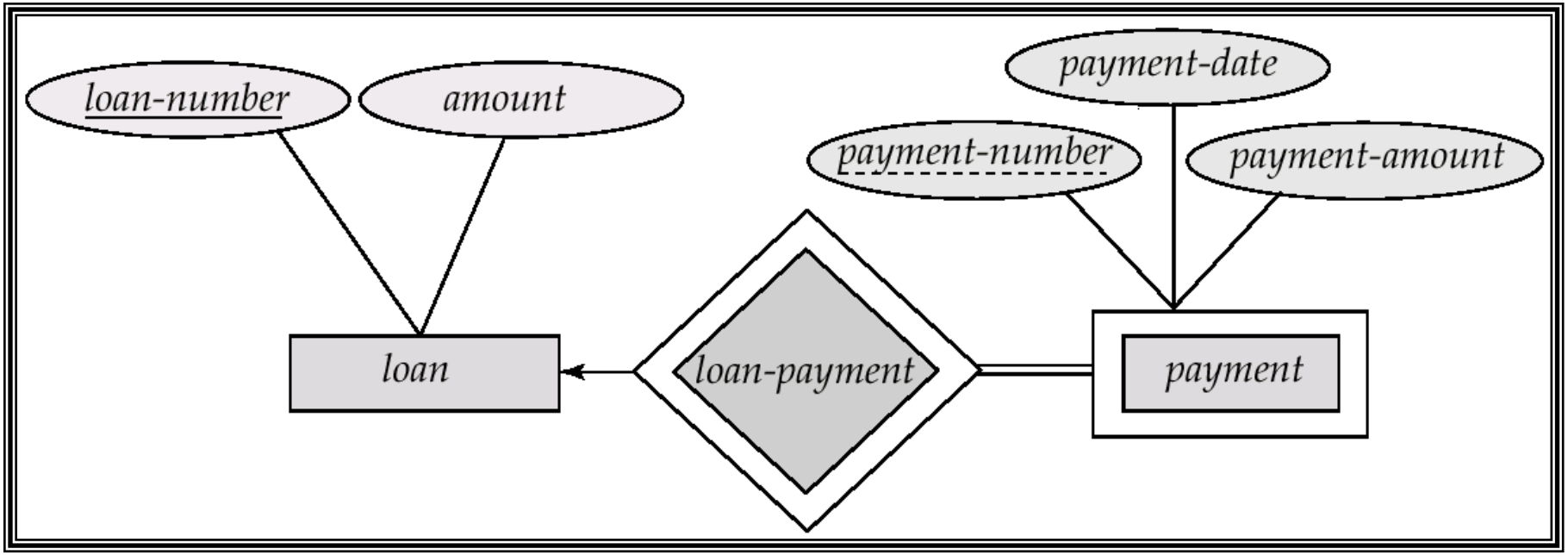
Weak Entity Sets (Cont.)

- 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

(although each payment entity is distinct, payments for different loans may share the same payment number)

- Primary key for *payment* – (*loan-number, payment-number*)

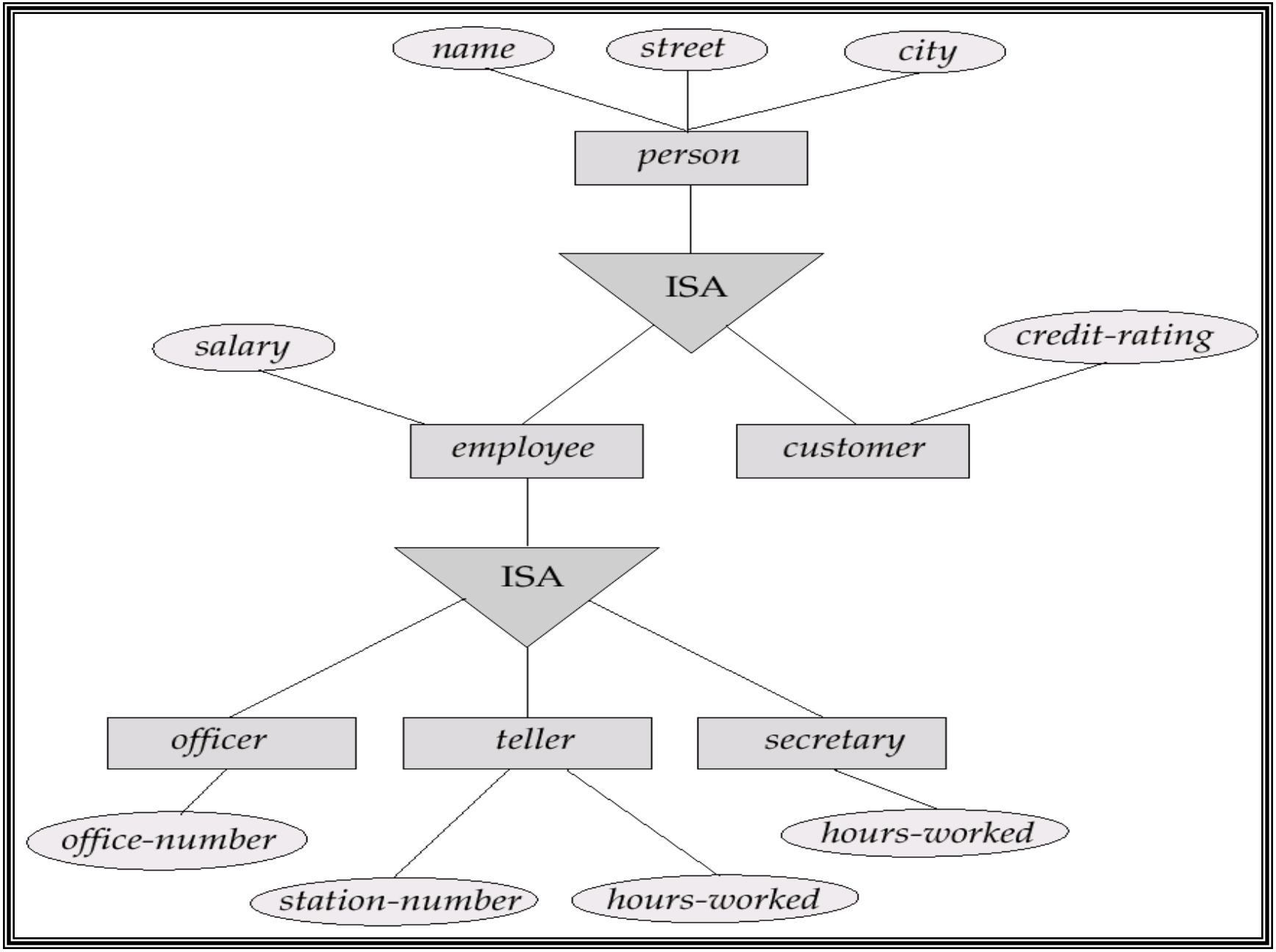


More Weak Entity Set Examples

- In a university, a *course* is a strong entity and a *course-offering* can be modeled as a weak entity.
  - *Course = (course-number, name, description)*
  - *Course-offering = (semester, section-number, instructor)*

- The discriminator of *course-offering* would be *semester* (including year) and *section-number* (if there is more than one section).

Specialization & Generalization Example



Specialization

- Top-down design process; we designate subgroupings within an entity set that are distinctive from other entities in the set.
- These subgroupings become lower-level entity sets that have attributes or participate in relationships that do not apply to the higher-level entity set.
- Depicted by a *triangle* component labeled ISA (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.

Generalization

- A bottom-up design process – combine a number of entity sets that share the same features into a higher-level entity set.
- Specialization and generalization are simple inversions of each other; they are represented in an E-R diagram in the same way.
- The terms specialization and generalization are used interchangeably.

Specialization and Generalization

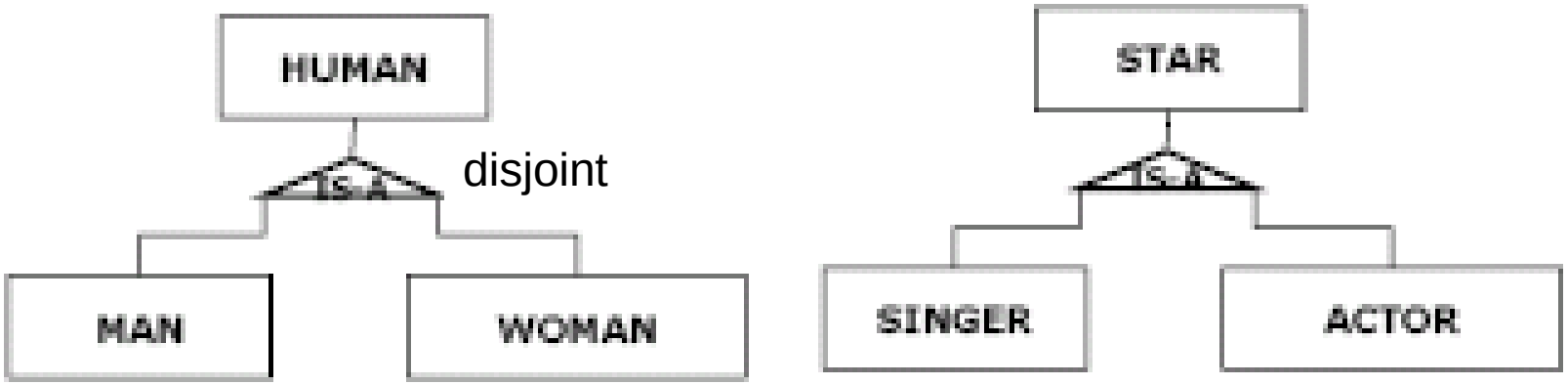
- Can have multiple specializations of an entity set based on different features.

E.g. *permanent-employee* vs. *temporary-employee*, in addition to *officer* vs. *secretary* vs. *teller*

- Each particular employee would be
  - a member of one of *permanent-employee* or *temporary-employee*,
  - and also a member of one of *officer*, *secretary*, or *teller*
- The ISA relationship also referred to as **superclass - subclass** relationship

Design Constraints on a Specialization/Generalization

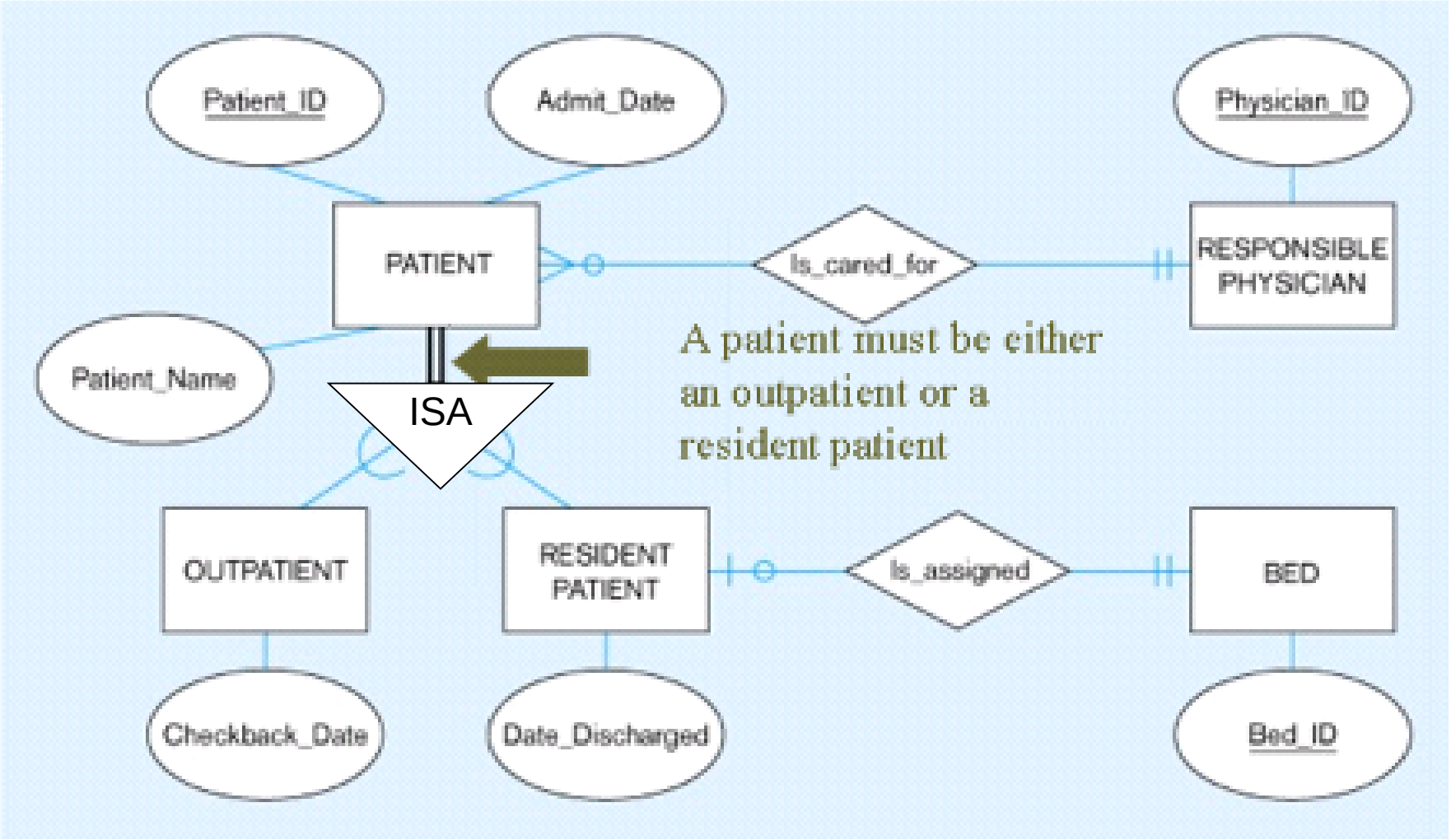
- Constraint on whether or not entities may belong to more than one lower-level entity set within a single generalization.
- **Disjoint**
  - an entity can belong to only one lower-level entity type
  - Noted in E-R diagram by writing *disjoint* next to the ISA triangle
  - Example: (HUMAN: MAN and WOMAN)
- **Overlapping**
  - an entity can belong to more than one lower-level entity type
  - Example (STAR: SINGER and ACTOR)



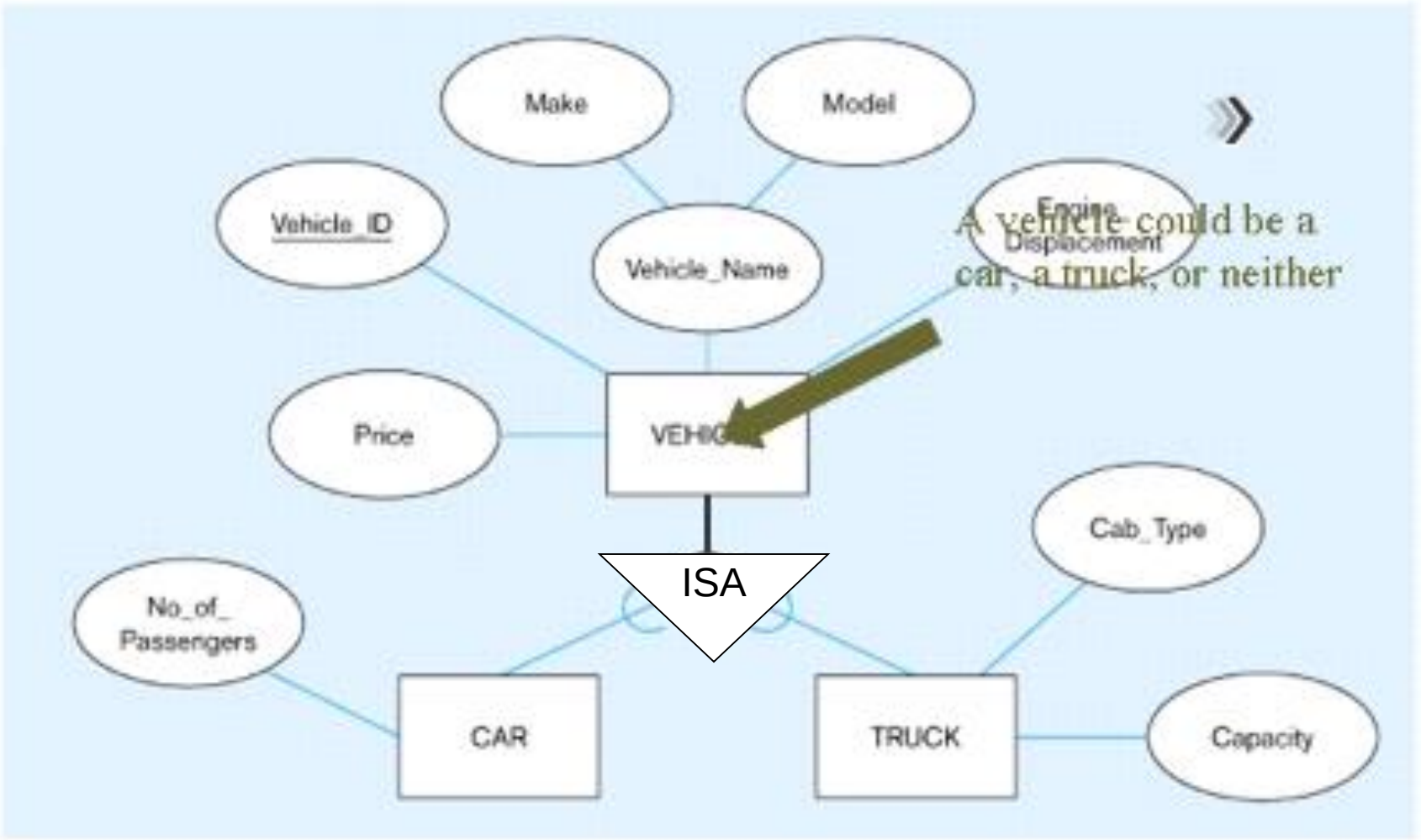
Design Constraints on a Specialization/Generalization

- Completeness constraint - specifies whether or not an entity in the higher-level entity set must belong to at least one of the lower-level entity sets within a generalization.
  - **total**: an entity must belong to one of the lower-level entity sets
  - **partial**: an entity need not belong to one of the lower-level entity sets

Total completeness

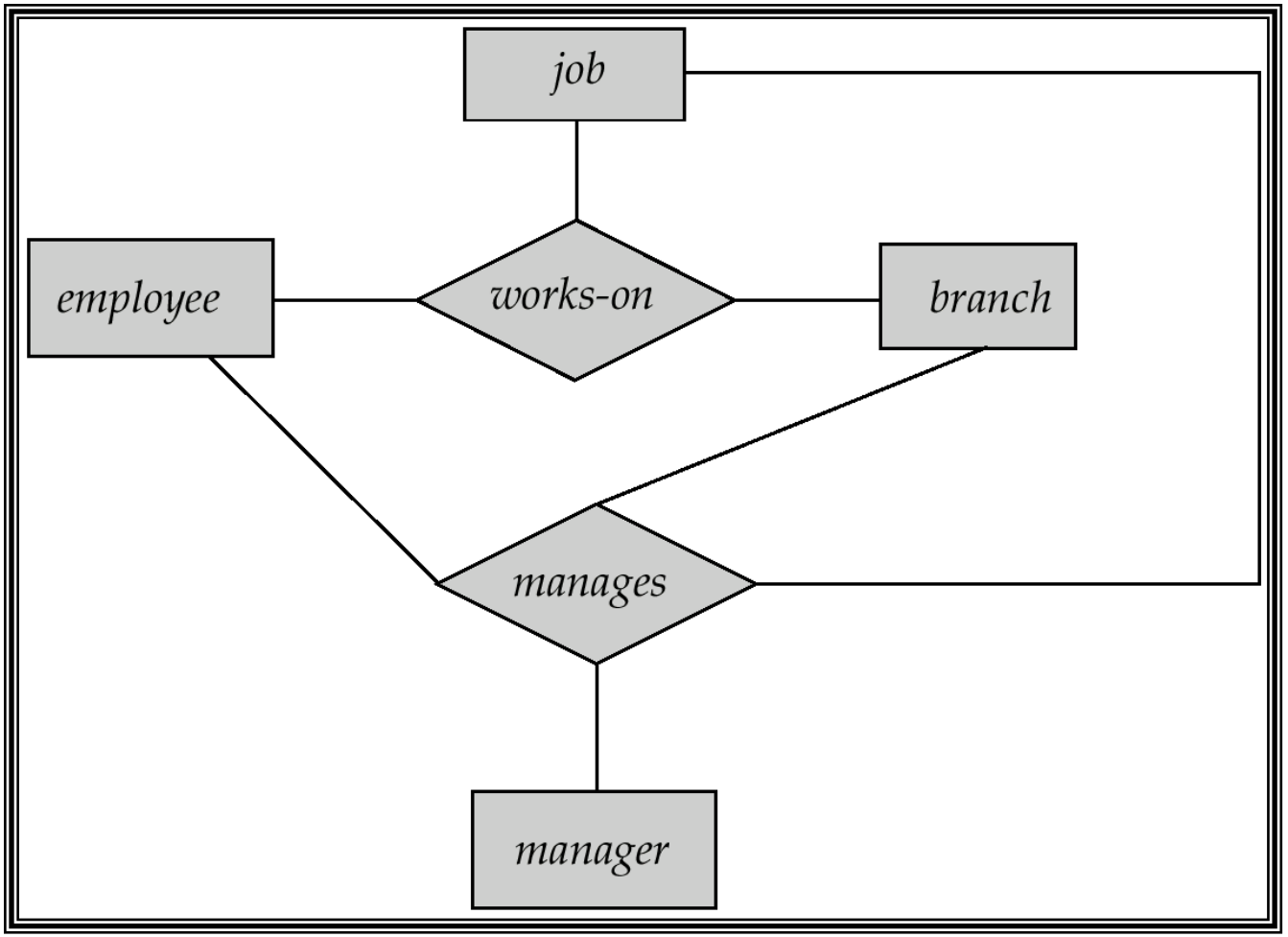


Partial completeness



Aggregation

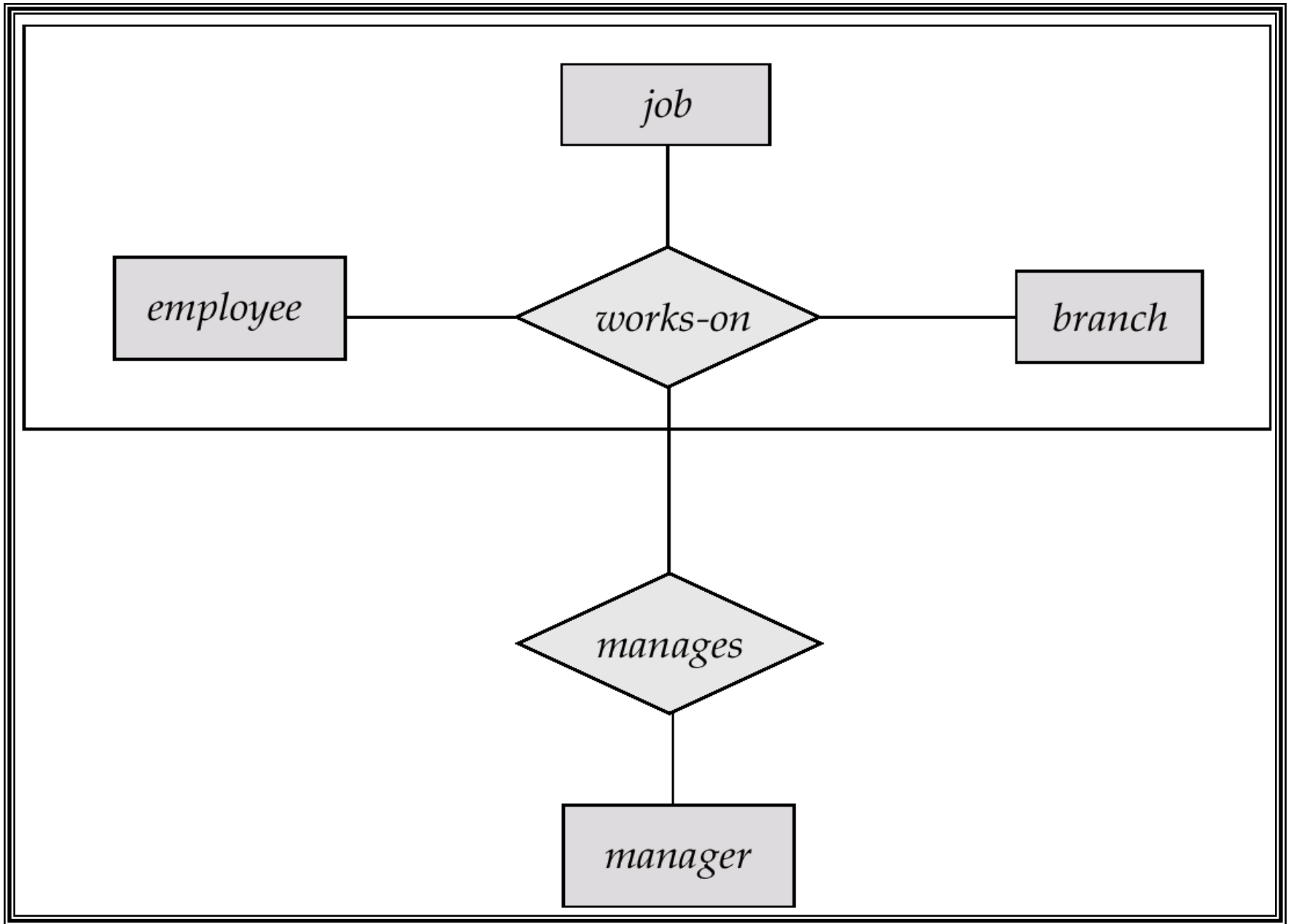
Suppose we want to record managers for tasks performed by an employee at a branch



Aggregation (Cont.)

- Relationship sets *works-on* and *manages* represent overlapping information
  - Every *manages* relationship corresponds to a *works-on* relationship
  - However, some *works-on* relationships may not correspond to any *manages* relationships
    - So we can't discard the *works-on* relationship
- Eliminate this redundancy via *aggregation*
  - Treat relationship as an abstract entity
  - Allows relationships between relationships
  - Abstraction of relationship into new entity
- Without introducing redundancy, the following diagram represents:
  - An employee works on a particular job at a particular branch
  - An employee, branch, job combination may have an associated manager

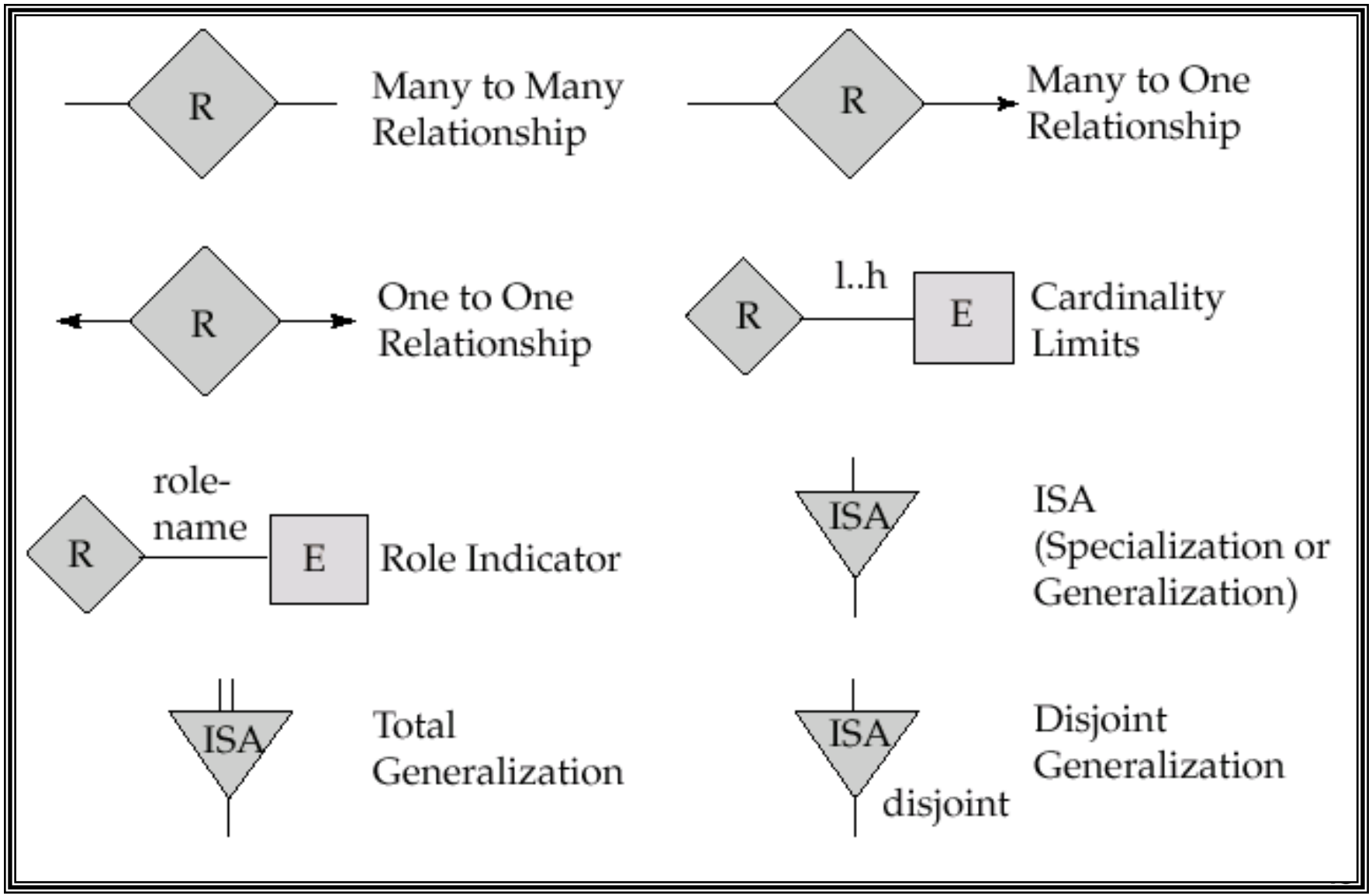
E-R Diagram With Aggregation



E-R Design Decisions

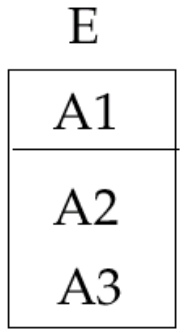
- The use of an attribute or entity set to represent an object.
- Whether a real-world concept is best expressed by an entity set or a relationship set.
- The use of a ternary relationship versus a pair of binary relationships.
- The use of a strong or weak entity set.
- The use of specialization/generalization – contributes to modularity in the design.
- The use of aggregation – can treat the aggregate entity set as a single unit without concern for the details of its internal structure.

Summary of Symbols

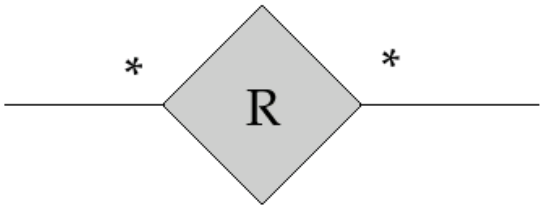


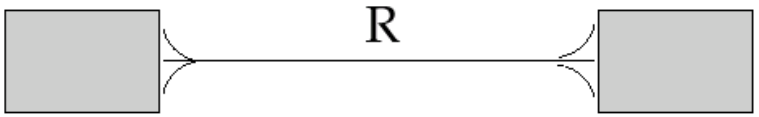
Alternative E-R Notations

Entity set E with attributes A1, A2, A3 and primary key A1

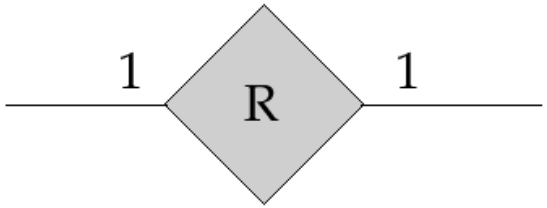


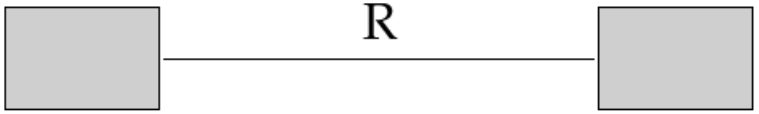
Many to Many Relationship



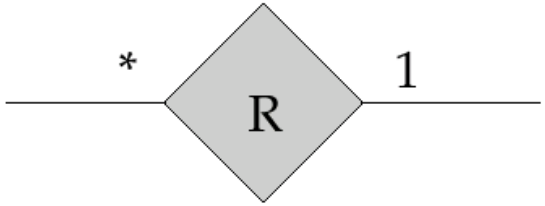


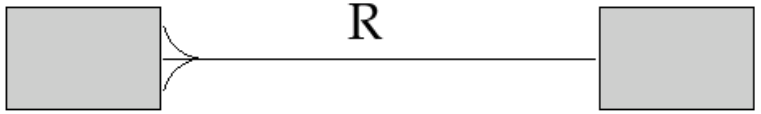
One to One Relationship





Many to One Relationship





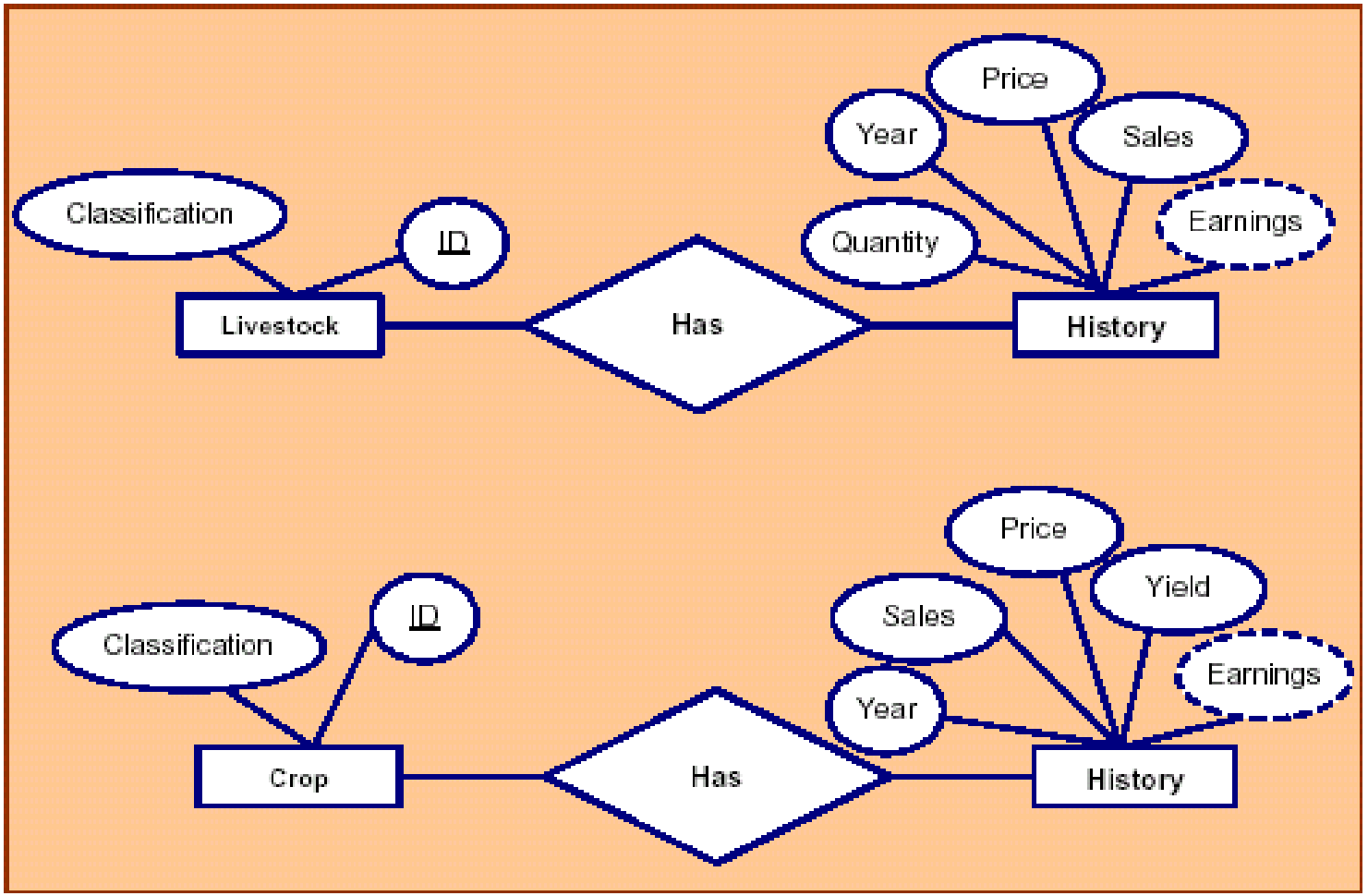
Question 1

John's family owns and operates a 640-acre farm for several generations. Since the farm business is growing, John is thinking to build a database that would make easier the management of the activities in the farm. He is considering the following requirements for the database:

1. For each livestock classification group (for example: caw, horse etc.), John keeps track of the following: identification number, classification, total number of livestock per classification group (for example: number of caws, number of horses etc.).
2. For each crop the following information is recorded: crop identification number and classification.
3. John has recorded the yield of each crop classification group during the last ten years. The records consist of the year, yield, sales, price of the crop and the amount of money earned.
4. John has recorded the yield of each livestock classification group during the last ten years. The records consist of the following historical data: the year, (historical) selling price per head, number of livestock in the end of the year, number of livestock sold during one-year period, and the total amount of money earned.

Draw an E-R diagram for this application. Specify the key attribute of each entity type.

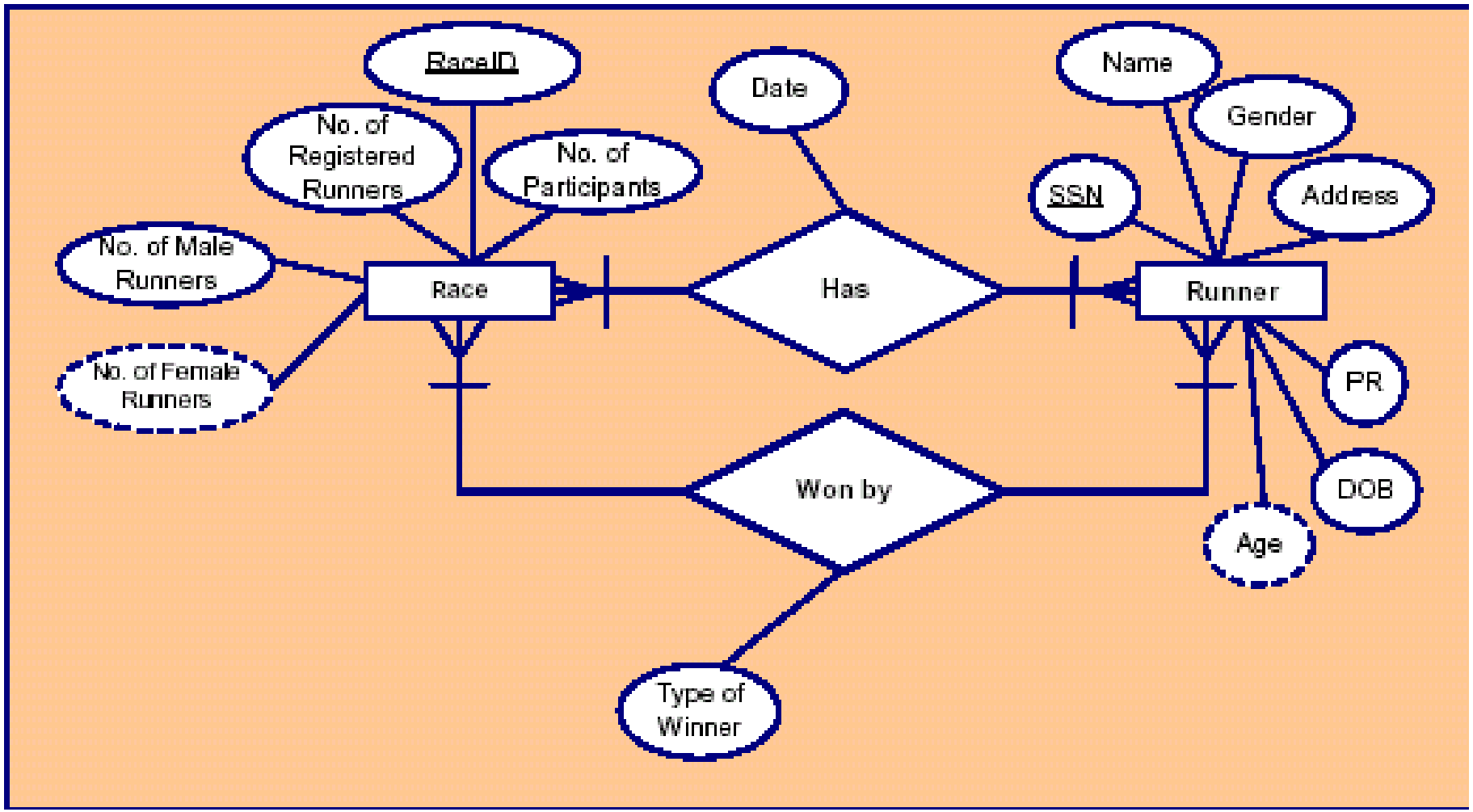
Question 1


Price
Year
Sales
Classification
ID
Earnings
Quantity
Livestock
Has
History
Price
Yield
Sales
Classification
ID
Earnings
Year
Crop
Has
History

Question 2

- The annual Bolder Boulder is one of America's top 10 Km races. The race is held each Memorial Day in Boulder, Colorado. This race attracts world-class runners as well as casual joggers. The race has grown to approximately 20,000 runners. The race is a point-to- point race beginning at the Bank of Boulder at the northeast corner of the city winding throughout the city streets, ending near to the town center in the University of Colorado's football stadium. The organizers record the following information for each race: the date of the race, total number of runners registered for the race (on-line pre-registration is possible), the actual number of participants, the number of female runners, the number of male runners, the name of the man winner, the name of the woman winner, the name of the man master (runner of age more than 40) winner, the name of the woman master winner. In addition, the following information about each participant is recorded as well: social security number, name, birthday, gender, address, age and certified personal record (PR) running times for a 10 Km race.
- **Draw an E-R diagram for this application. Specify the key attribute of each entity type.**

Question 2

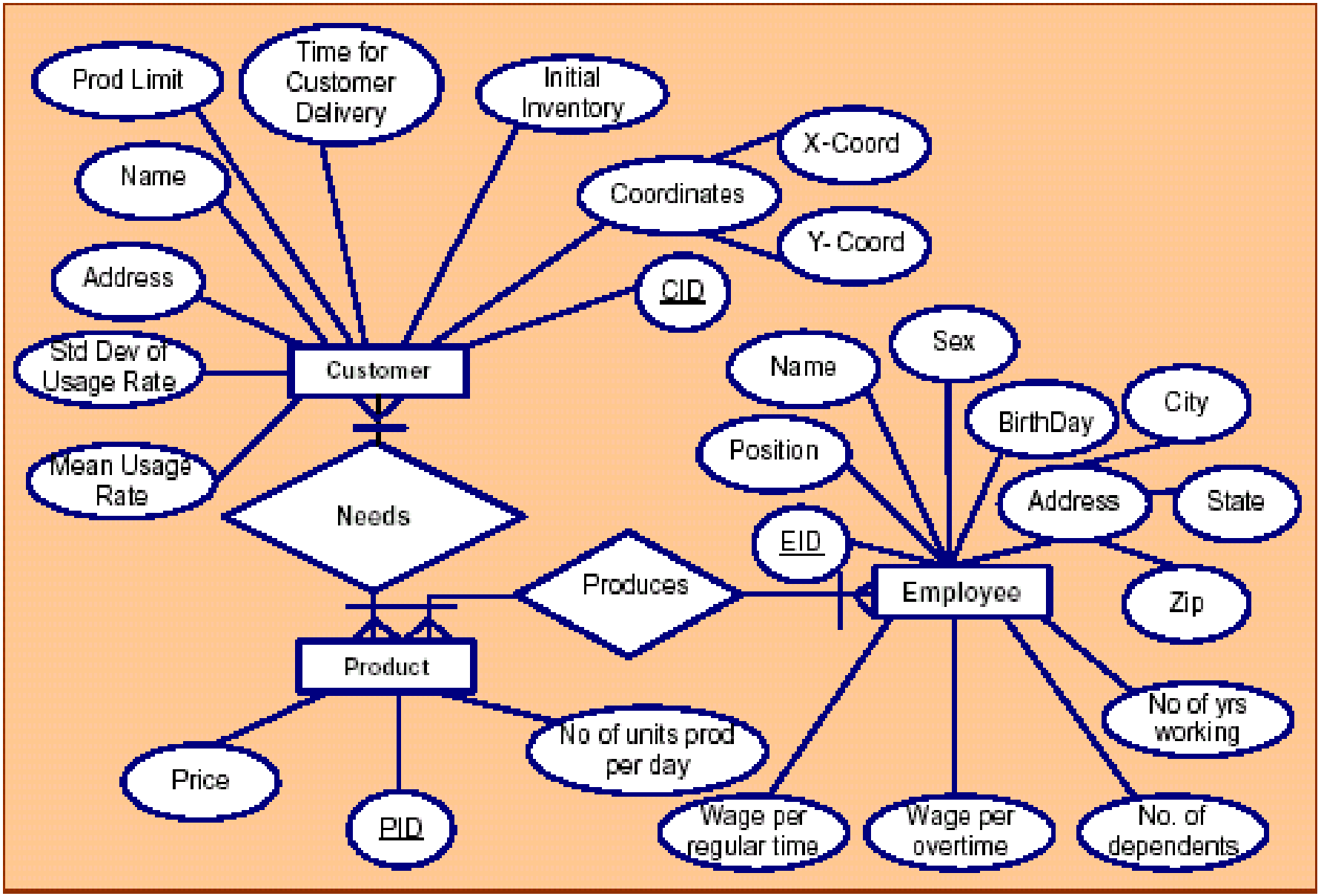

RaceID
Date
Name
Gender
No. of Registered Runners
No. of Participants
SSN
Address
No. of Male Runners
Race
Has
Runner
No. of Female Runners
PR
Won by
DOB
Age
Type of Winner

Question 3

- Coca Cola Co. in Atlanta, Georgia produces a wide range of products that are delivered to the customers once a week. The Coca Cola Co. keeps the information about the employees, products, and customers in a database. The database considers the following set of tables:
- The company keeps the following information about each customer: customer identification number, name, address, X (longitude) and Y (latitude) coordinates of their location, amount of time (fraction of an hour) required to make a stop at a customer, type of product that is used by a customer, mean rate at which customer uses product per day in a week, standard deviation of this usage rate, the limit on how much inventory of a product can be held at a customer, initial inventory of product at a customer.
- Each employee has an employee identification number, name, address (the address consists of: city, state and zip code), sex, birthday, position in the company, wage earned per hour of regular timework, wage earned per hour of overtime work, number of dependents, and number of years working for Coca Cola Co.
- Each product has a product identification number, price and number of units produced per day.

Draw an E-R diagram for the Coca Cola Co and add the minimum and maximum relationship cardinalities. Identify the: (a) key attributes for each entity (b) composite attributes.

Question 3

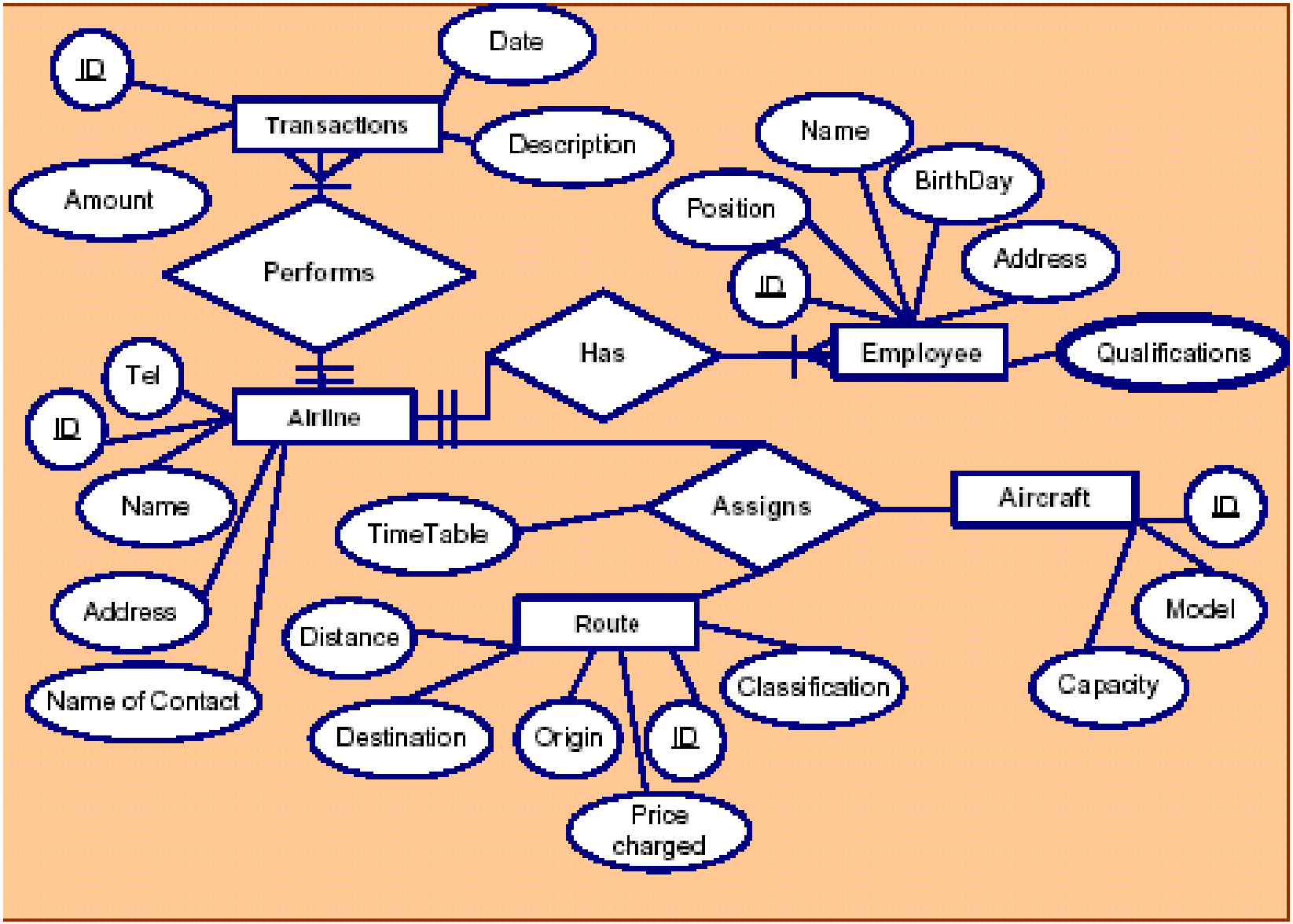

Prod Limit
Time for Customer Delivery
Initial Inventory
Name
X-Coord
Coordinates
Y- Coord
Address
CID
Std Dev of Usage Rate
Customer
Name
Sex
BirthDay
City
Position
Mean Usage Rate
Needs
Address
State
EID
Produces
Employee
Zip
Product
No of yrs working
No of units prod per day
Price
PID
Wage per regular time
Wage per overtime
No. of dependents

Question 4

- Major airlines companies that provide passenger services in Taiwan are: UniAir, TransAsia Airways, Far Eastern Transport, Great China Airlines etc. Taiwan's Federal Aviation Administration (TFAA) keeps a database with lots of information on all airlines. This information is made accessible to all airlines in Taiwan with the intention of helping the companies assess their competitive position in the domestic market. The information kept consists of:
- Each airline has an identification number, name and address, name of the contact person and telephone number.
- For each aircraft an aircraft identification number, capacity, and model is recorded.
- Each employee has an employee identification number, name, address, birthday, sex, position with the company, and qualification.
- Each route has a route identification number, origin, destination, classification (into domestic or international route), distance of the route, and price charged per passenger.
- Each airline keeps information about their buy/sell transactions (for example selling an airplane ticket is a sell transaction, paying for maintenance is a buy transaction). Each transaction has a transaction identification number, date, description, and amount of money paid/received.

Draw an E-R diagram for the database presented above. Make sure to identify the associative entity (entities) and provide corresponding key attributes (attributes).

Question 4


ID
Date
Transactions
Description
Amount
Performs
Name
BirthDay
Position
Address
ID
Has
Employee
Qualifications
Tel
ID
Airline
Name
Address
Name of Contact
TimeTable
Assigns
Aircraft
ID
Model
Capacity
Distance
Route
Classification
Destination
Origin
ID
Price charged

Question 5

- **Consider the following information pertaining to '*Video World*' Store which deals with the rental of videos.**
- **The store maintains a record of all the videos that they have. The store has quite a number of copies of any given video especially when it is first released.**
- **Each video has the following information: title, release date, duration (in minutes), the category it belongs to and whether or not it is to be displayed in the 'New release' section. Each category is allocated a unique number and examples of the category name are: Action, Drama, Comedy, Kids, etc.**
- **The proprietors also like keep a list of major actors and/or actresses that are in any one video so that if a customer asks for any videos that have their favourite star they can provide them with a list of videos they have.**
- **Each video copy has its own unique barcode. In addition the copy number and the status of the copy is always known. A copy can have a status of 'in' or 'out', where 'out' means that it is being rented by a customer. And hence unavailable and 'in' means that it is available**
- **Customers are given a video card with their own unique number. A customer is classified as an employee or a regular customer. The card number as well as the surname, first name, street, city, state, postcode, and telephone details are recorded for regular customers and the card number and employee number are recorded for employees. ('Video World' has asked that you use the EmployeeId from an existing Personnel / Payroll system and so you do not need to worry about what other details are stored for each employee in that system).**
- **For each video rented to a customer the following information is kept: the date rented, the date due back, the date it was returned, the days it was late, and the fee charged for the rental.**
- **The proprietors of '*Video World*' Store have approached you and asked if you could design a database to help them manage their business using the information given above. They would like you to create a conceptual model of their business information requirements. The model should be in the form of an Entity-relationship diagram.**

Question 6

- Electronic commerce is one of the most used terms in the business world. Electronic commerce is the buying and selling of goods and services on the Internet. One of the most popular products for e-commerce is compact disc. This problem describes the database of a CD warehouse. The database will be used by the customers and company's employees. Assume that customers have access to company's web site and they are able to open an account by providing their social security number, name, address, and music preferences. Every time a customer orders a product, the system updates the customer's account (the amount of money due is updated as well the name and amount of product ordered is recorded). For each employee the following information is recorded: an employee identification number, name, address, birthday, and the title of the position with the company. The products for the CD warehouse are the albums. The database keeps the following information about each album: an album identification number, name, group name, release date, musical category, name of the vocalist, names of the other band members, and the number of CDs on stock. The database keeps information about suppliers: a supplier identification number, address, name of the company, and name of the contact person.

- **Draw an E-R diagram for the CD warehouse database. Identify the relationship cardinalities and the key attribute of each entity.**