

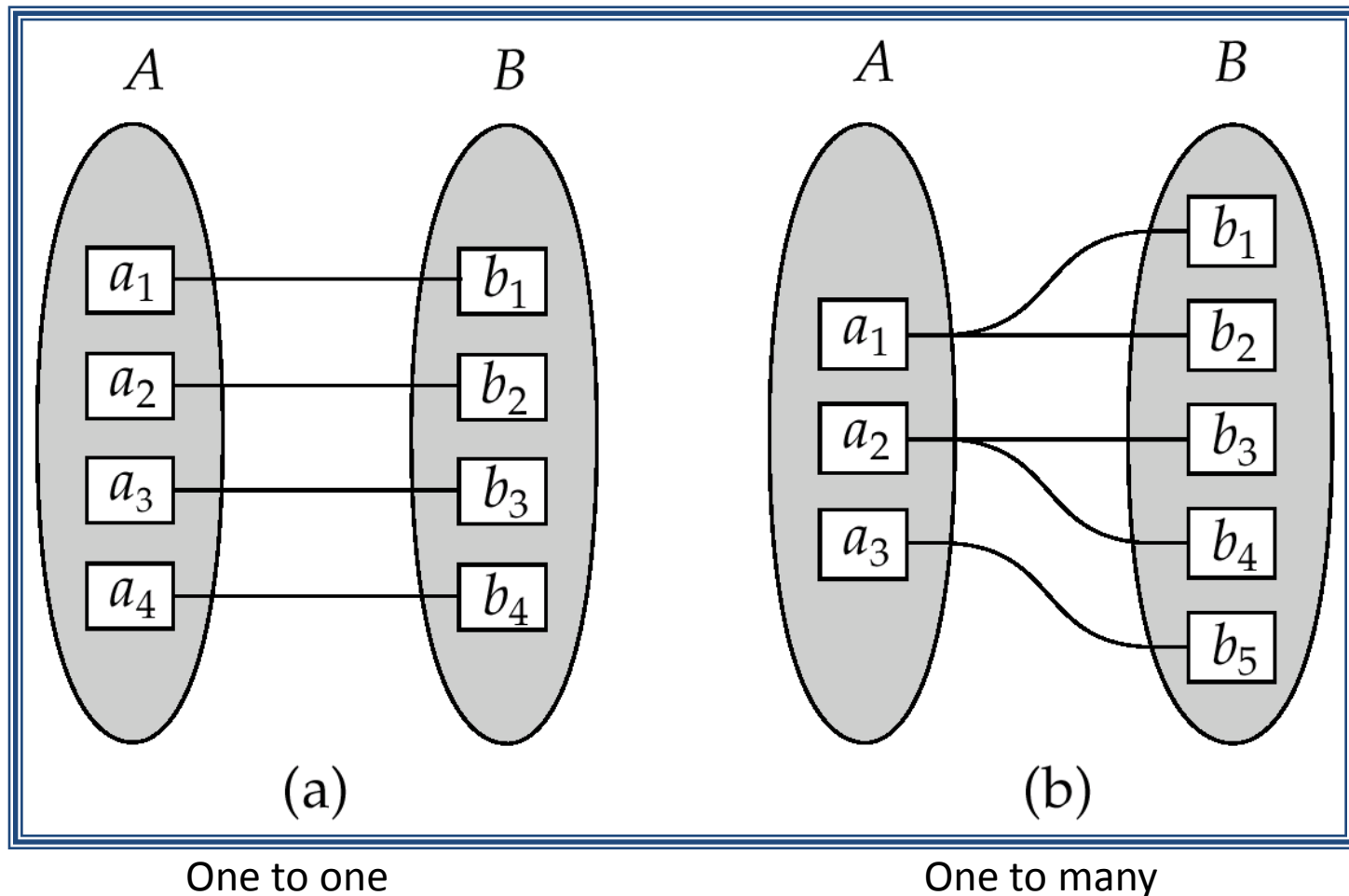
Entity-Relationship Model

1. Entity
2. Attributes
3. Entity Sets
4. Relationship Sets
5. Design Issues
6. Mapping Constraints
7. Weak Entity
8. Keys
9. E-R Diagram
10. Extended E-R Features
11. Design of an E-R Database Schema
12. Reduction of an E-R Schema to Tables

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

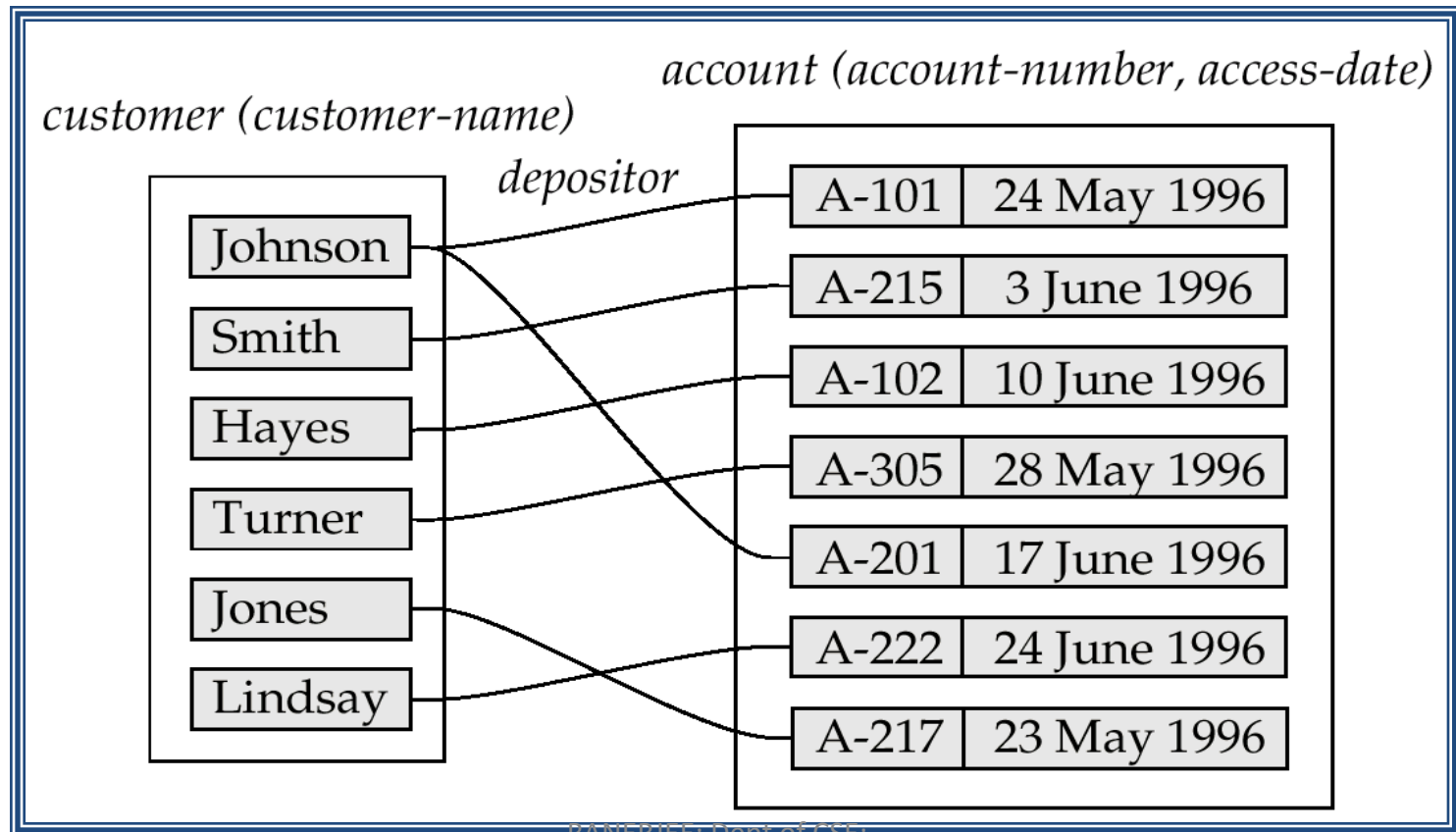
Mapping Cardinalities



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

Mapping Cardinalities affect ER Design

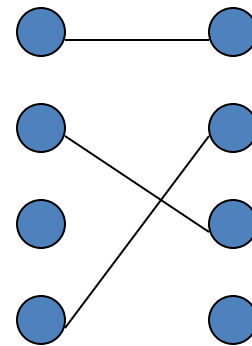
- Can make *access-date* an attribute of account, instead of a relationship attribute, if each account can have only one customer
 - I.e., the relationship from account to customer is many to one, or equivalently, customer to account is one to many



One-One Relationships

- In a *one-one relationship*, each entity of either entity set is related to at most one entity of the other set.
- **Example:** Relationship **Best-seller** between entity sets **Manfs** (manufacturer) and **Beers**.
 - A beer cannot be made by more than one manufacturer, and no manufacturer can have more than one best-seller (assume no ties).

In Pictures:



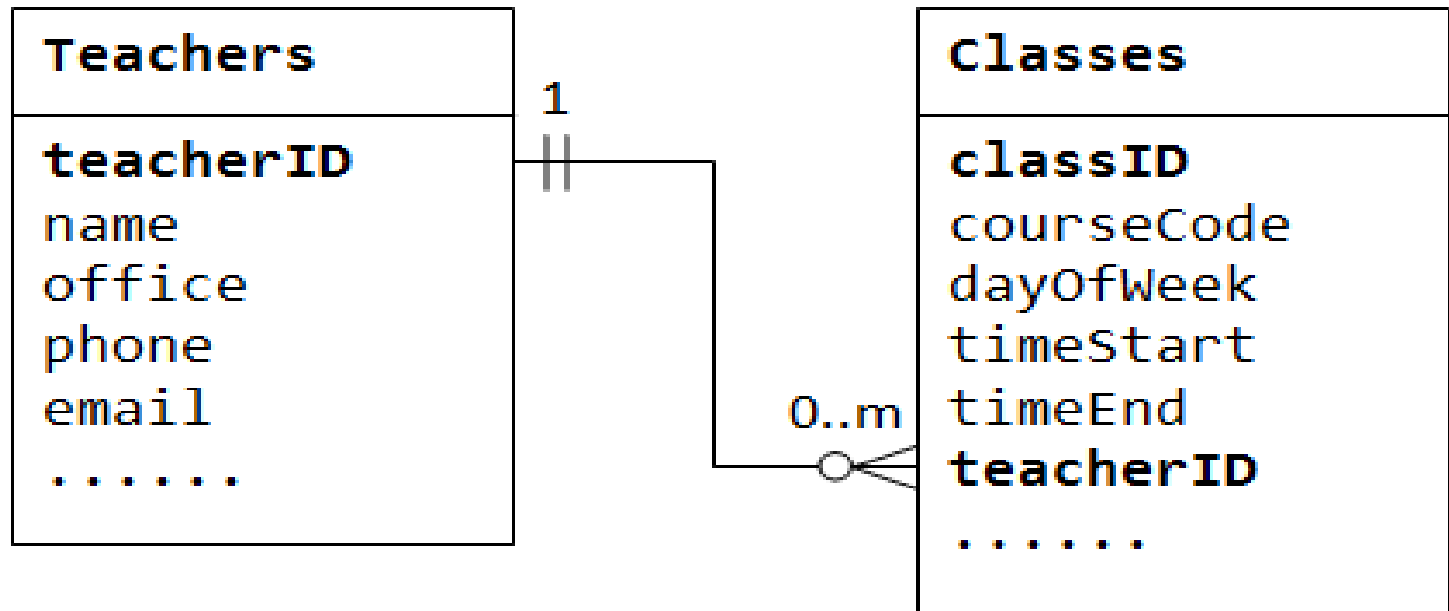
one-one

Example: One-One Relationship

- Consider **Best-seller** between **Manfs** and **Beers**.
- Some beers are not the best-seller of any manufacturer, so a rounded arrow to **Manfs** would be inappropriate.
- But a beer manufacturer has to have a best-seller.

One- Many Relationships

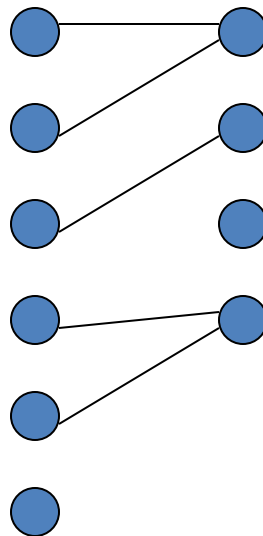
A lecturer may tutor many students, but each student has just one tutor



Many-One Relationships

- Some binary relationships are *many -one* from one entity set to another.
- Each entity of the first set is connected to at most one entity of the second set.
- But an entity of the second set can be connected to zero, one, or many entities of the first set.

In Pictures:

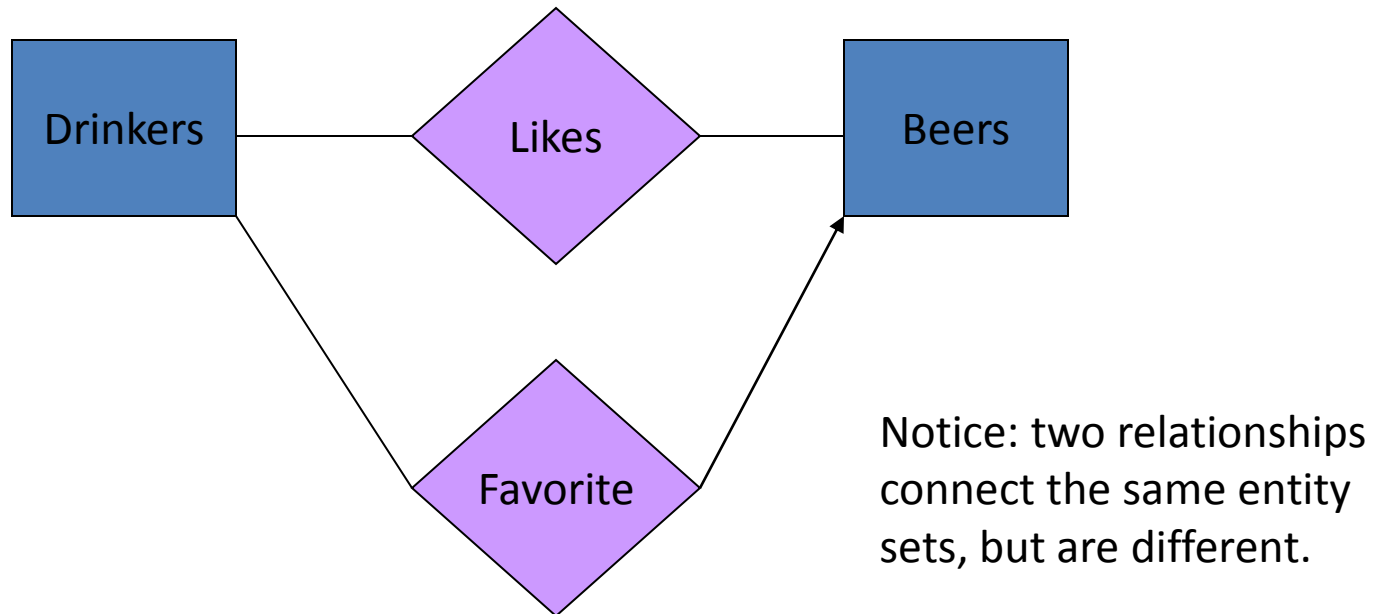


many-one

Example: Many-One Relationship

- Favorite, from Drinkers to Beers is many-one.
- A drinker has at most one favorite beer.
- But a beer can be the favorite of any number of drinkers, including zero.

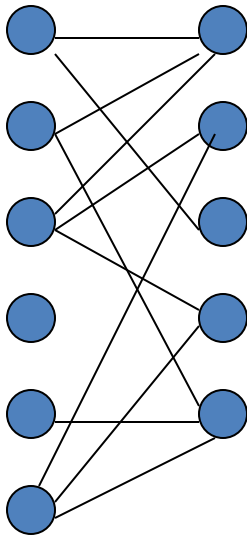
Example Typical Case: Many-One Relationship



Many-Many Relationships

- Focus: **binary** relationships, such as **Sells** between **Bars** and **Beers**.
- In a **many-many relationship**, an entity of either set can be connected to many entities of the other set.
 - E.g., a bar sells many beers; a beer is sold by many bars.

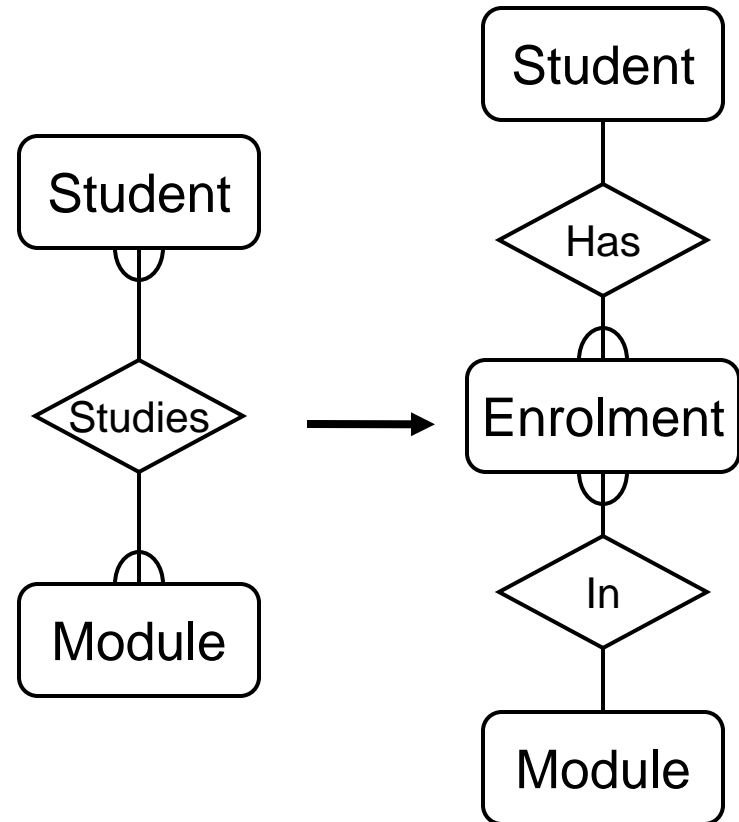
In Pictures:



many-many

Removing M:M Relationships

- Many to many relationships are difficult to represent
- We can split a many to many relationship into two one to many relationships
- An entity represents the M:M relationship

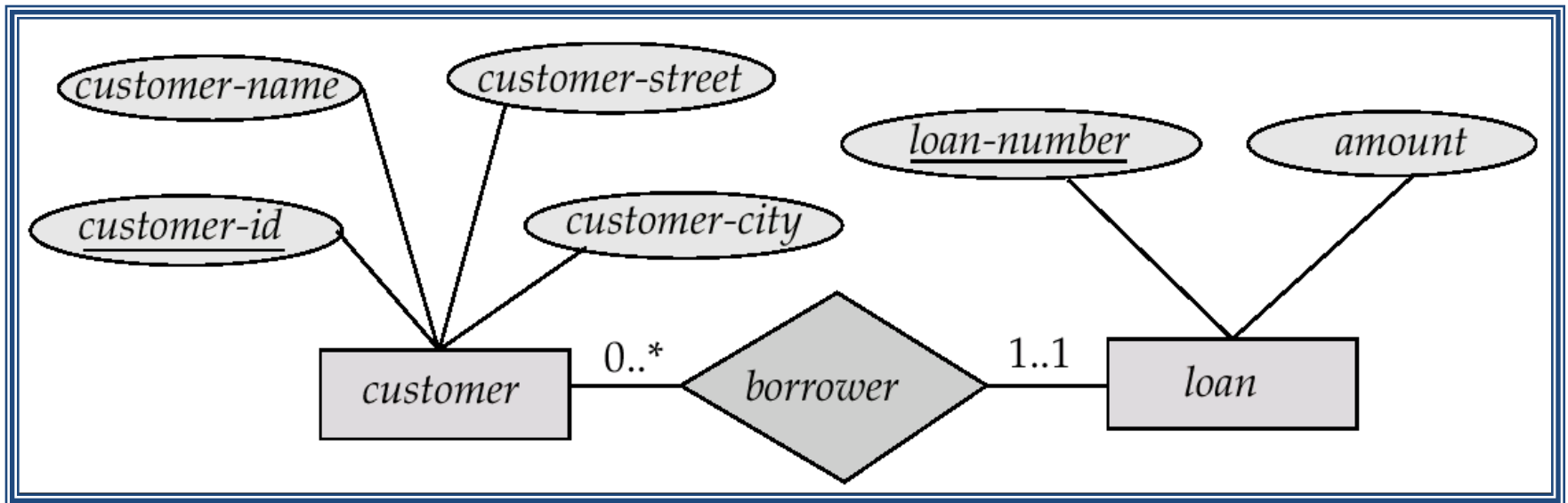


Representing “Multiplicity”

- Show a many-one relationship by an arrow entering the “one” side.
 - **Remember**: Like a functional dependency.
- Show a one-one relationship by arrows entering both entity sets.
- **Rounded arrow** = “exactly one,” i.e., each entity of the first set is related to exactly one entity of the target set.

Alternative Notation for Cardinality Limits

- Cardinality limits can also express participation constraints



Cardinality Constraints

An important feature of entity-relationship schemata is the possibility of specifying *cardinality constraints*.

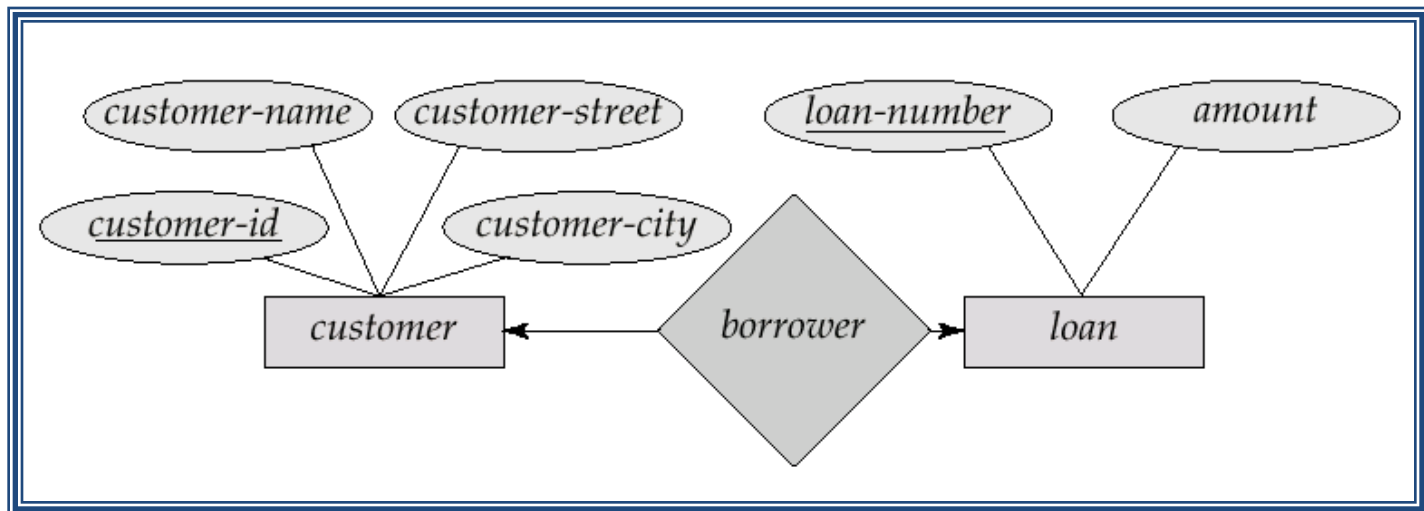
These are instructions of the form "every person has exactly one mother", or "every document must have at least an author (but possibly many)".

These constraints are useful because they allow to maintain the *logical integrity* of the database. In a logically integral database, you can always trust a document to have an author, and we can avoid to check if it is missing.

An important point (often missed) about logical integrity is that *It is a feature of the semantics of the entity-relationship schema, not of the underlying relational database*. It must be expressed abstractly in term of properties of multirelations, and not in term of the particular relational implementation of the entity-relationship schema.

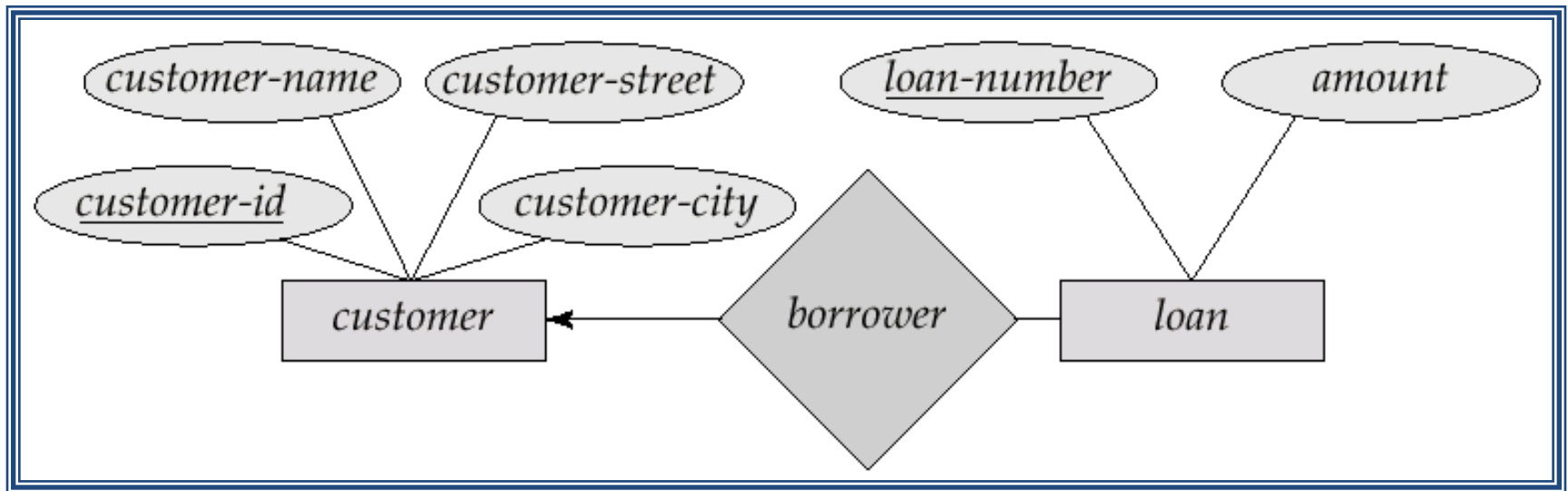
Cardinality constraints restrain the possible multirelation that an instance can assign to the relationship type. For each of the two entity types you can specify a pair of indices between parentheses and separated by colon, as in (1:N). The left character may be 0 or 1, while the right character may be 1, N, or M.

- 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*



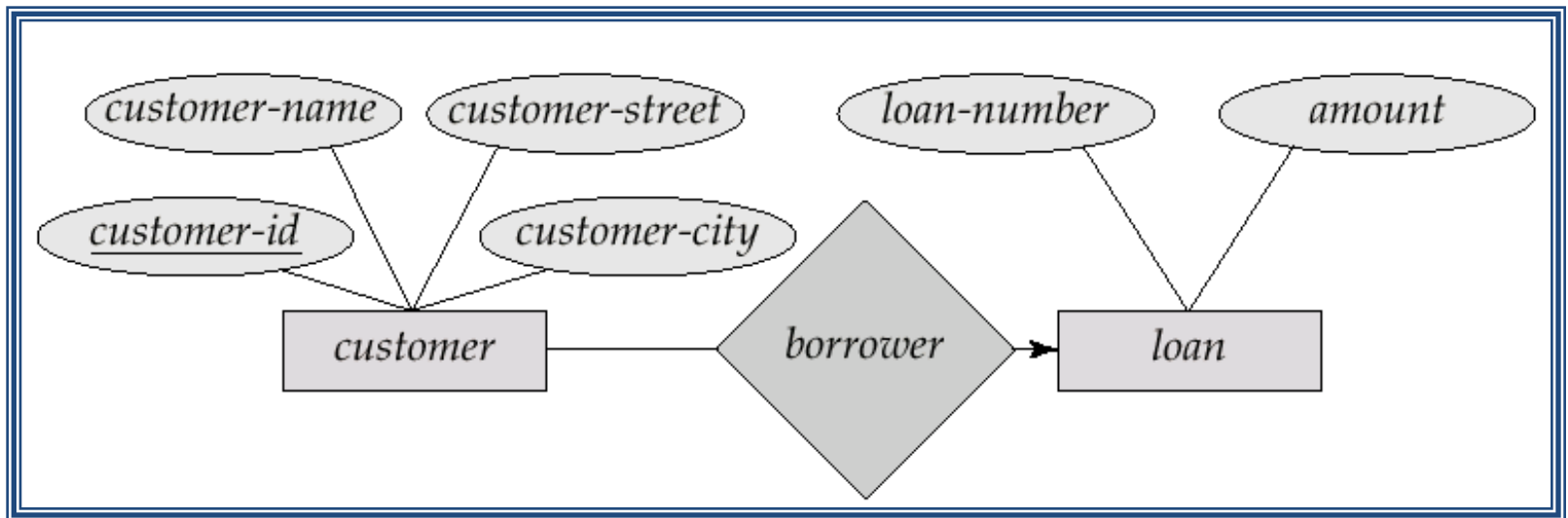
One-To-Many Relationship

- 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*

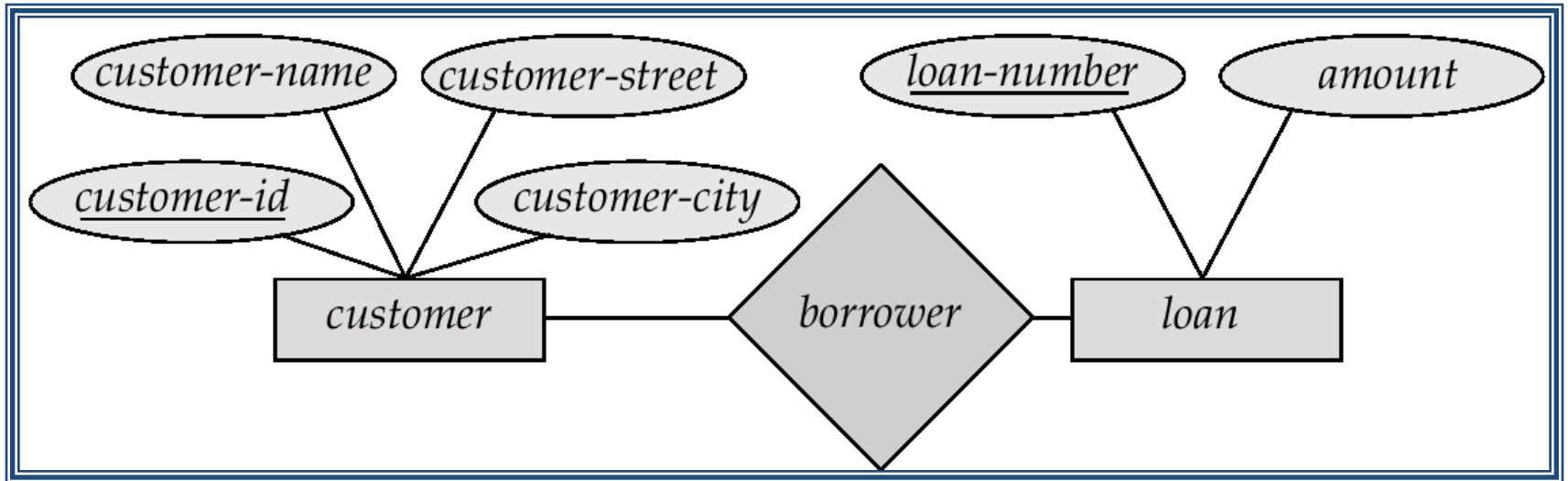


Many-To-One Relationships

- 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