



Chapter 2

Entity Relationship Modeling

Department: Computer

Course: DBMS

Faculty: Sana Shaikh

Learning Objectives:

- **Quick Recap**
- Constraints on Relationship
 - Cardinality of a Relationship
 - Relationship Participation
- Attributes on Relationship
- Weak Entity
- How to Evaluate a Data Model?
- Solving few examples

Learning Outcomes:

Students should be able to:

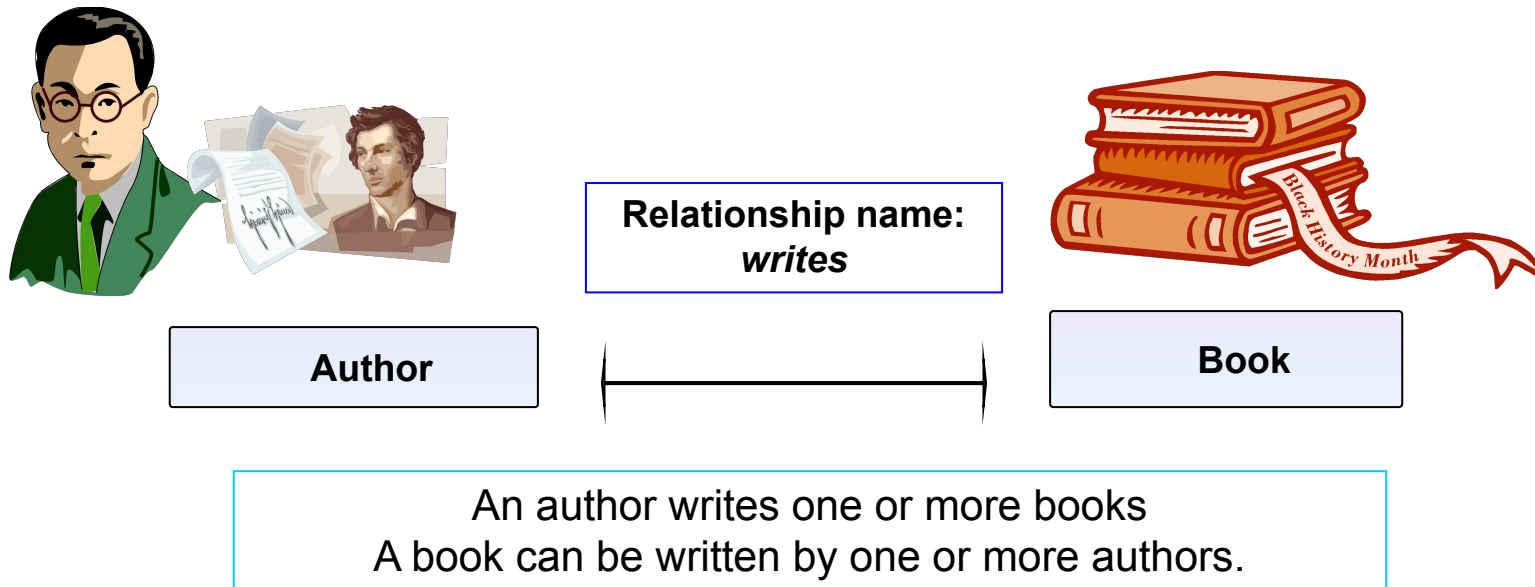
- Understand the need of Attributes on Relationship
- Apply constraints on relationships for any real world problems
- Differentiate between weak and strong Entities.
- Evaluate data model
- Develop a basic Entity Relationship Model with the appropriate entities, attributes, relationships, connectivity, and cardinality using Chen notation

Quick Recap

- Entity Types and Entity Sets
- Domain of Attributes
- Types of Attributes
- Key attribute
- NULL Values
- Degree of a Relationship

Relationships

- Associations between instances of one or more entity types that is of interest
- Given a name that describes its function.
 - relationship name is an active or a passive verb.



Degree of Relationships

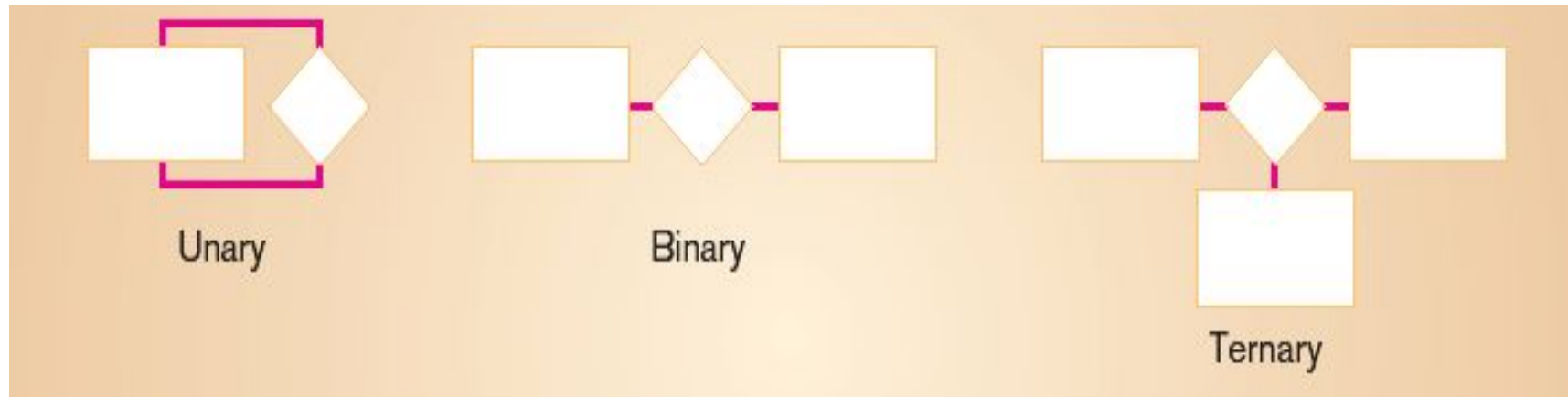
Degree: number of entity types that participate in a relationship

Three cases

Unary: between two instances of one entity type

Binary: between the instances of two entity types

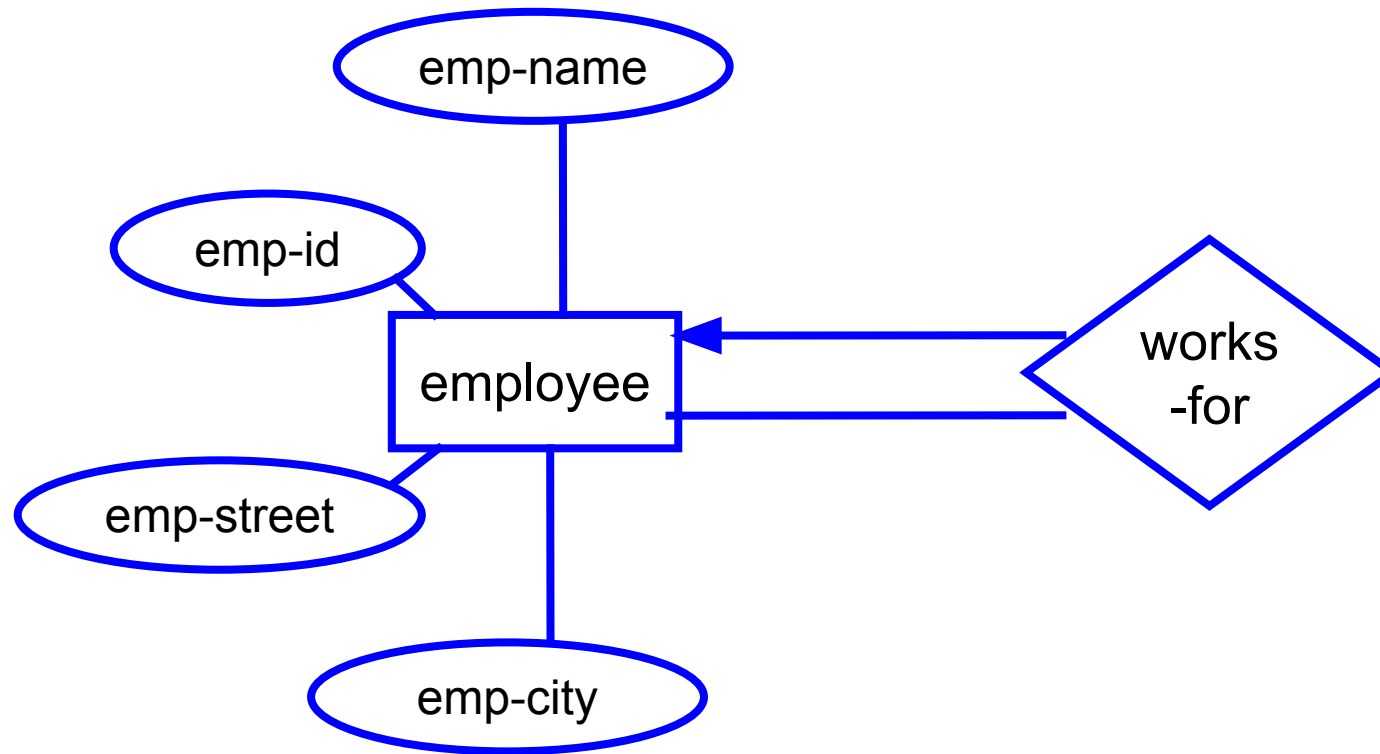
Ternary: among the instances of three entity types



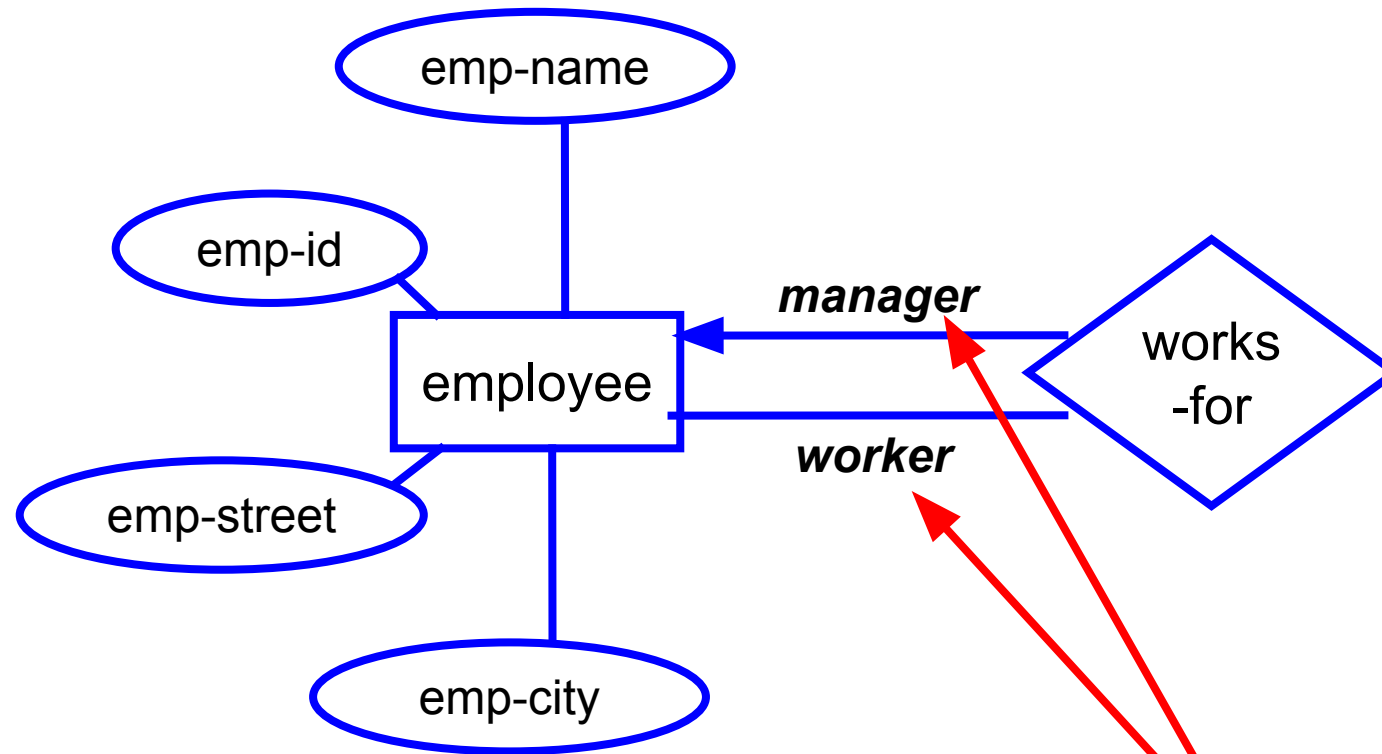
Unary Relationship

- Sometimes a relationship associates an entity set to itself
- also known as Recursive relationships

Unary Relationships



Unary Relationships

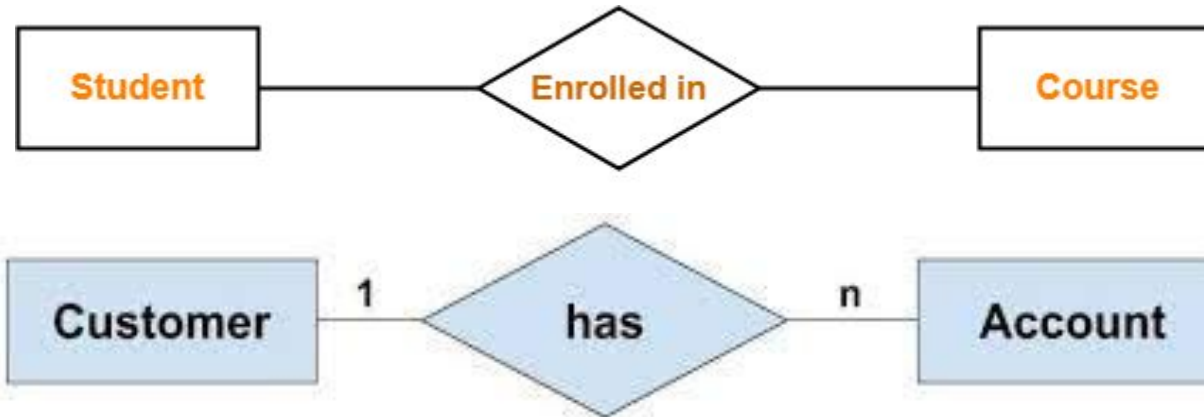


Must be declared with roles
known as Role names

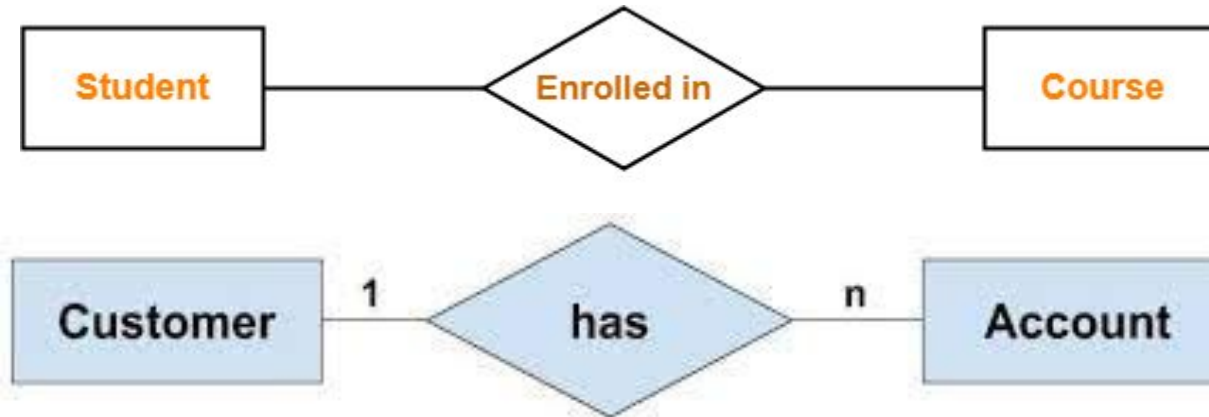
Binary Relationship



Binary Relationship

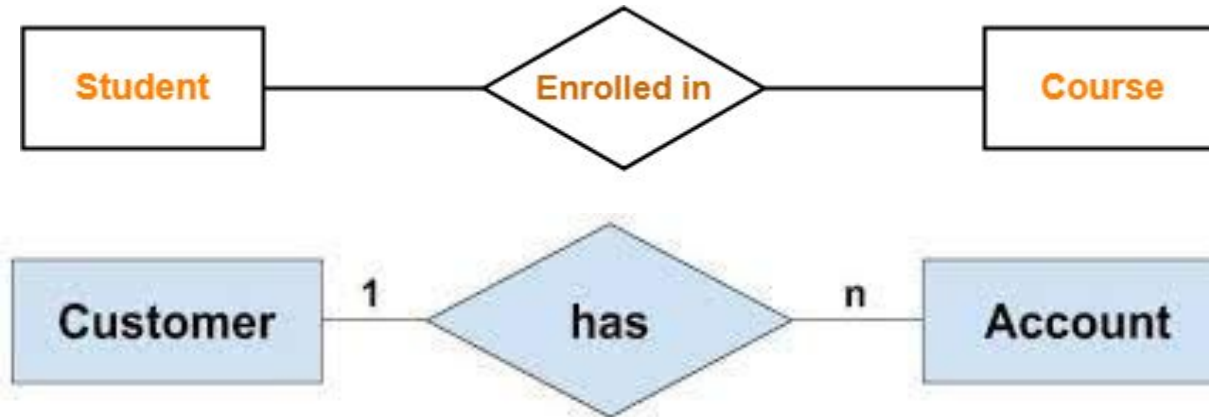


Binary Relationship

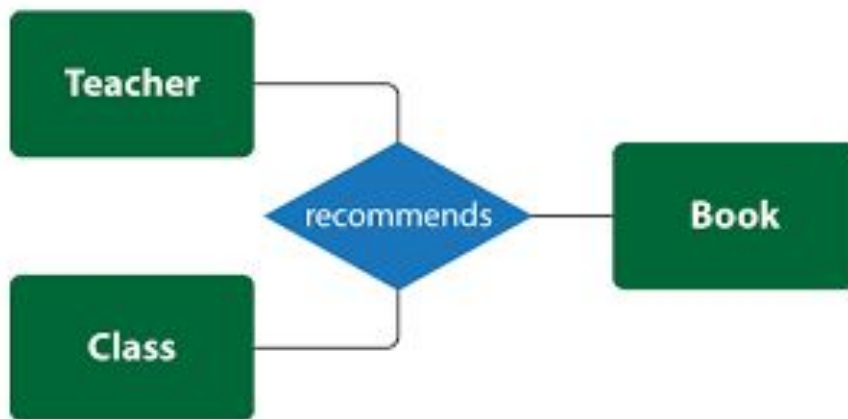


Ternary Relationship

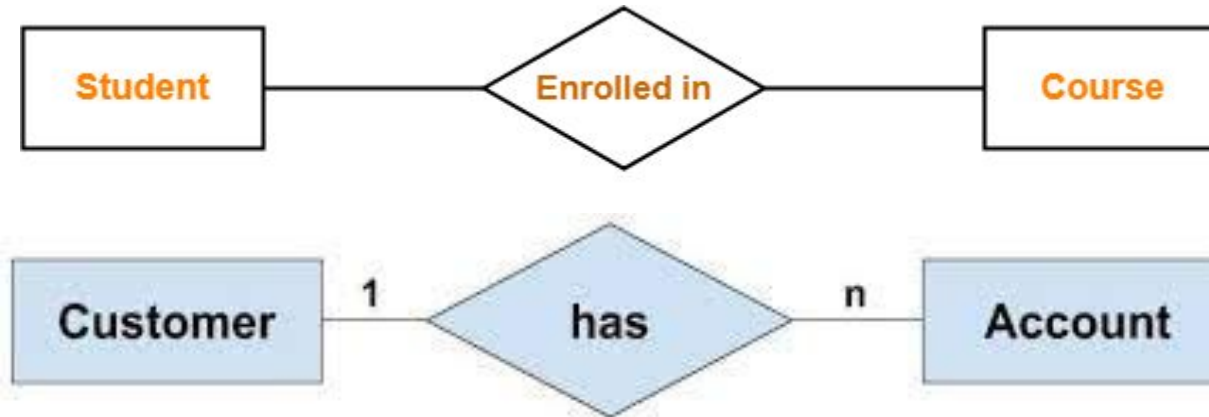
Binary Relationship



Ternary Relationship



Binary Relationship



Ternary Relationship



Kinds of Constraints

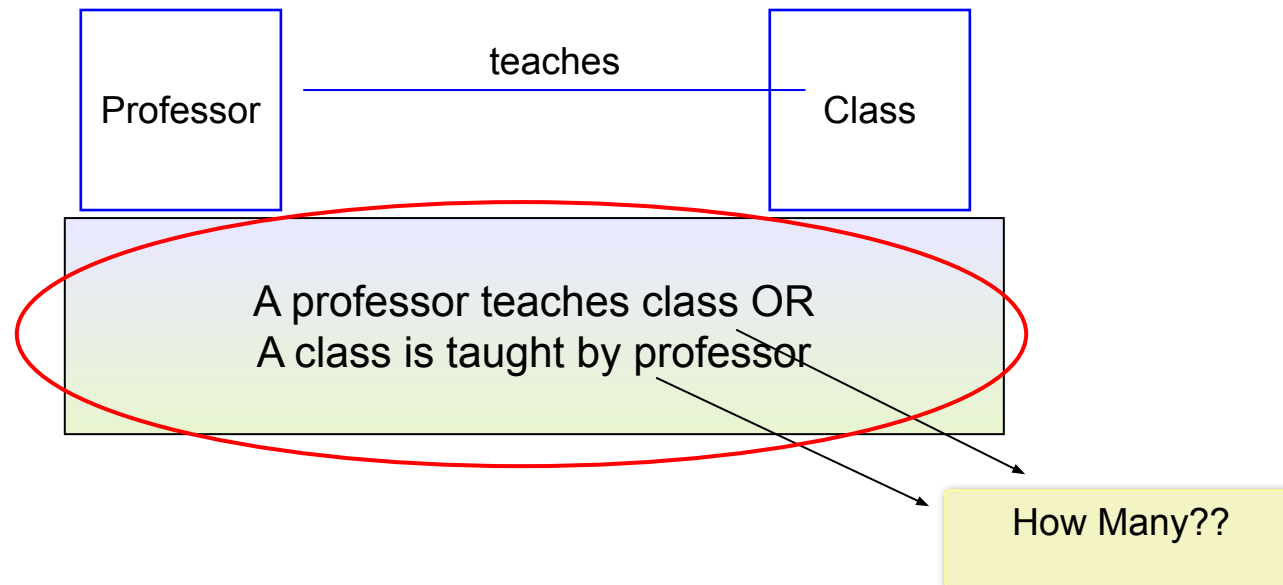
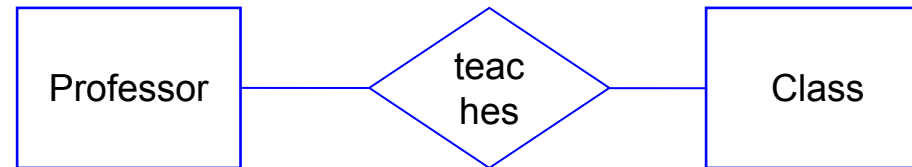
What kind of constraints can be defined in the ER Model?

- Cardinality Constraints
- Participation Constraints

Together called “Structural Constraints”

Constraints are represented by specific notation in the ER diagram

Cardinality and Connectivity



Cardinality and Connectivity


Relationships can be classified as either

one – to – one

one – to – many

many – to – one

many – to – many

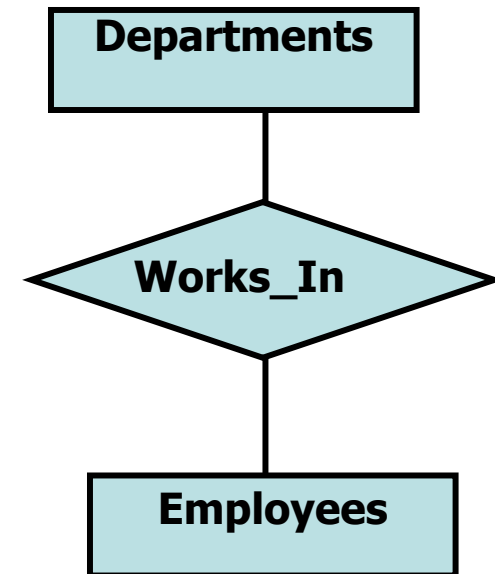


Connectivity

Cardinality : minimum and maximum number of instances of Entity B that can (or must be) associated with each instance of entity A.

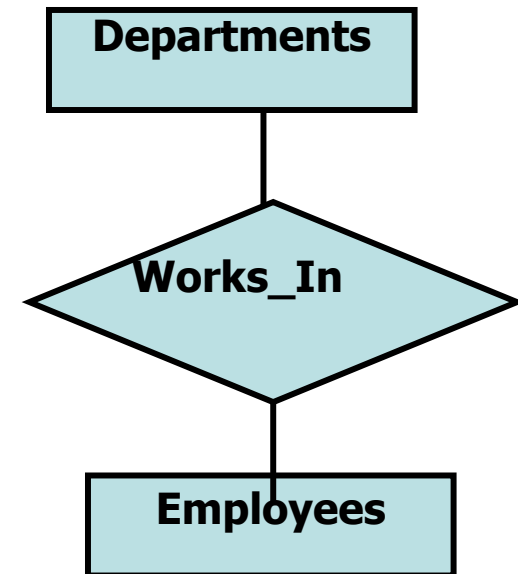
Possible Cardinality Ratios

- The “Cardinality Ratio” for a binary relationship specifies the number of relationship instances that an entity can participate in



Possible Cardinality Ratios

- The “Cardinality Ratio” for a binary relationship specifies the number of relationship instances that an entity can participate in
 - Works-In is a binary relationship
 - Participating entities are DEPARTMENT : EMPLOYEE
 - *One* department can have *Many* employees

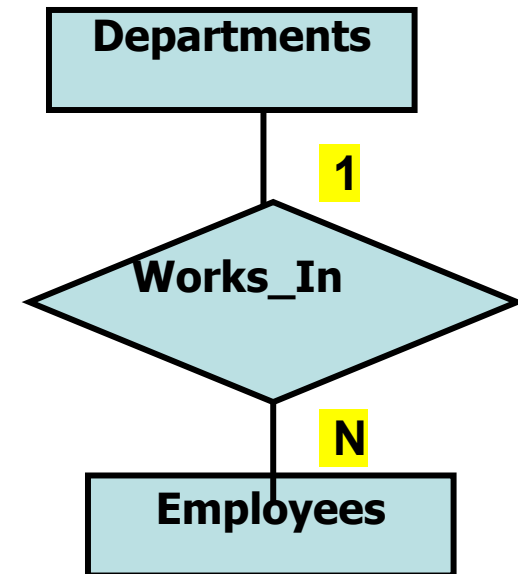


Possible Cardinality Ratios

- The “Cardinality Ratio” for a binary relationship specifies the number of relationship instances that an entity can participate in

- Works-In is a binary relationship
- Participating entities are DEPARTMENT : EMPLOYEE
- *One* department can have *Many* employees

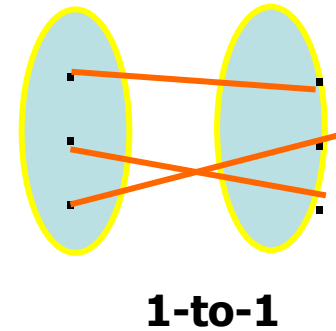
Cardinality Ratio is 1 : N



Possible Cardinality Ratios

□ 1-to-1 (1 : 1)

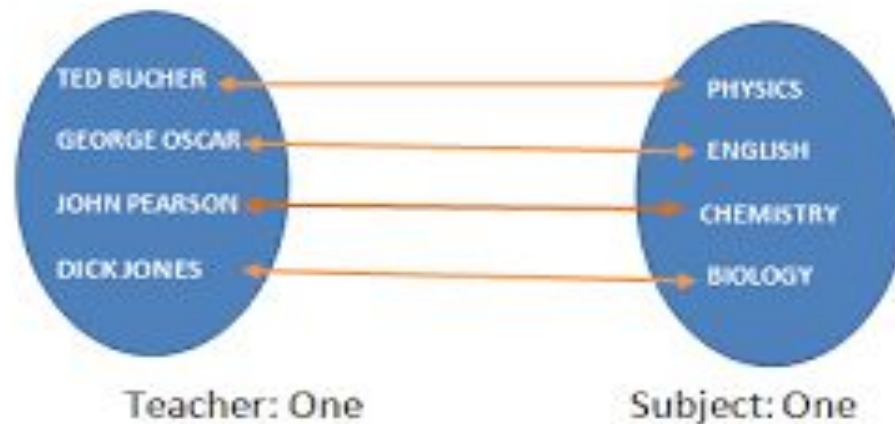
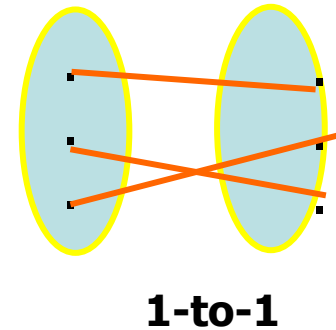
- Both entities can participate in only **one** relationship instance



Possible Cardinality Ratios

□ 1-to-1 (1 : 1)

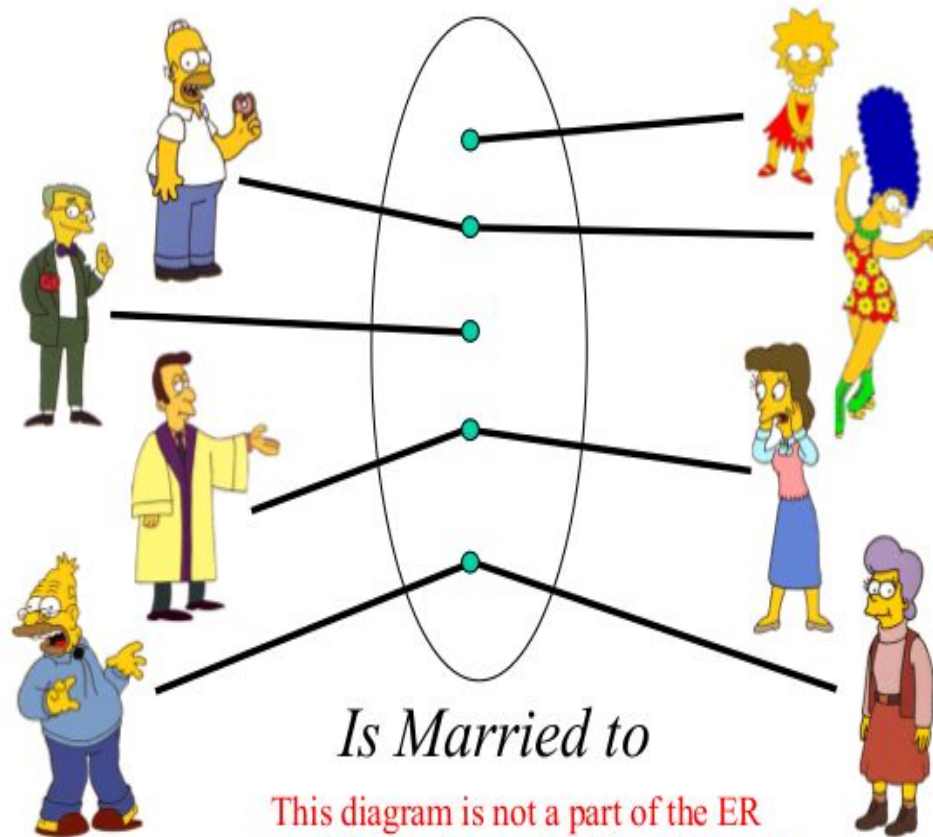
- Both entities can participate in only **one** relationship instance



Example

- One-to-one:** An entity in A is associated with at most one entity in B, and an entity in B is associated with at most one entity in A.

A man may be married to at most one woman, and a woman may be married to at most one man (both men and women can be unmarried)

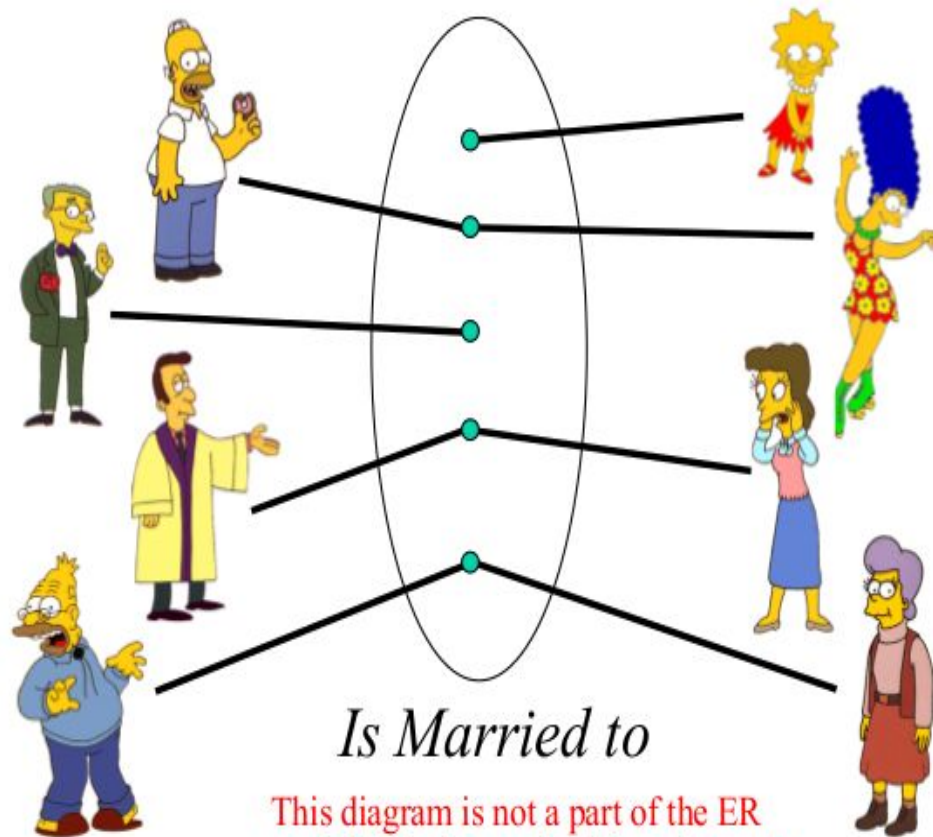


This diagram is not a part of the ER model! It is just an intuitive picture to explain a concept

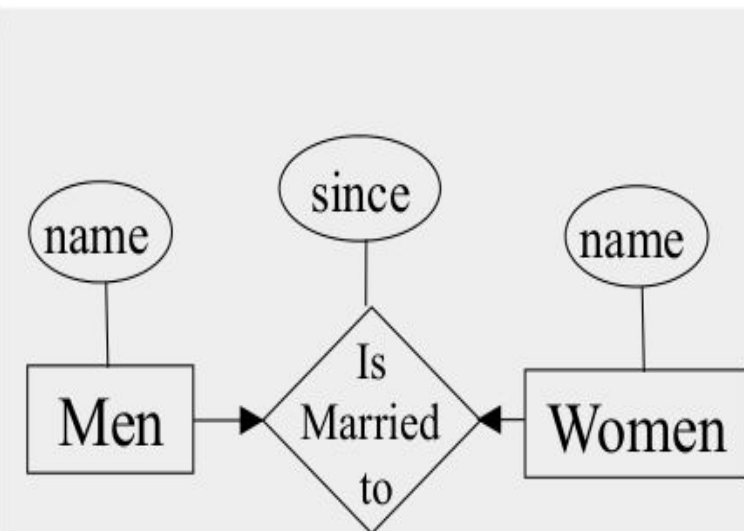
Example

- **One-to-one:** An entity in A is associated with at most one entity in B, and an entity in B is associated with at most one entity in A.

A man may be married to at most one woman, and a woman may be married to at most one man (both men and women can be unmarried)

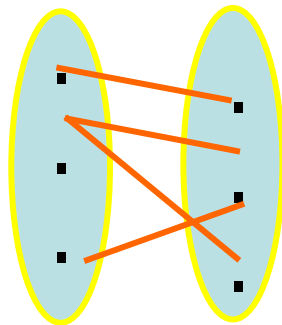


This diagram is not a part of the ER model! It is just an intuitive picture to explain a concept

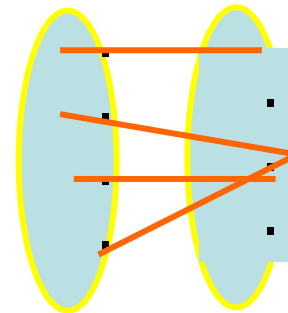


Possible Cardinality Ratios

- 1-to-Many, Many-to-1
(1 : N, N : 1)
 - One entity can participate in **many** relationship instances



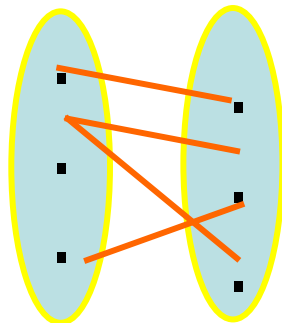
1- to - Many.



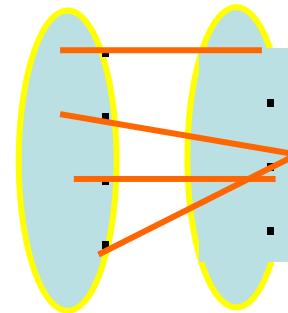
Many - to - 1

Possible Cardinality Ratios

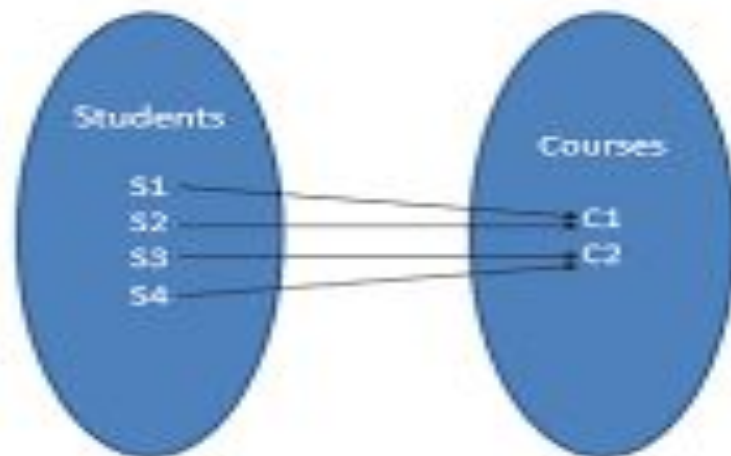
- 1-to-Many, Many-to-1
(1 : N, N : 1)
 - One entity can participate in **many** relationship instances



1- to - Many.

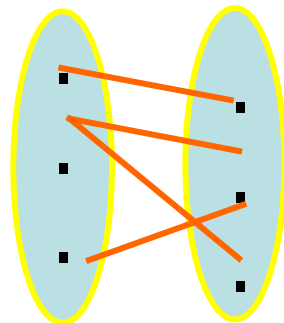


Many - to - 1

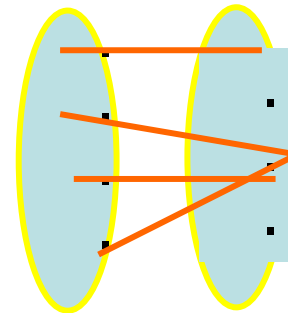


Possible Cardinality Ratios

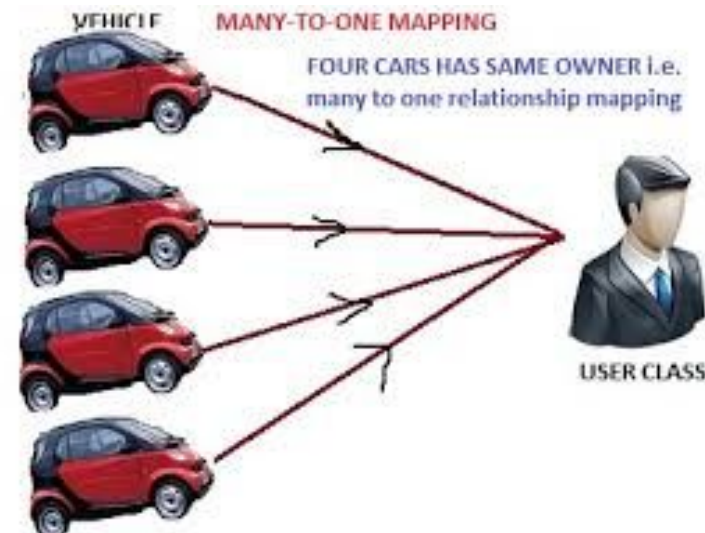
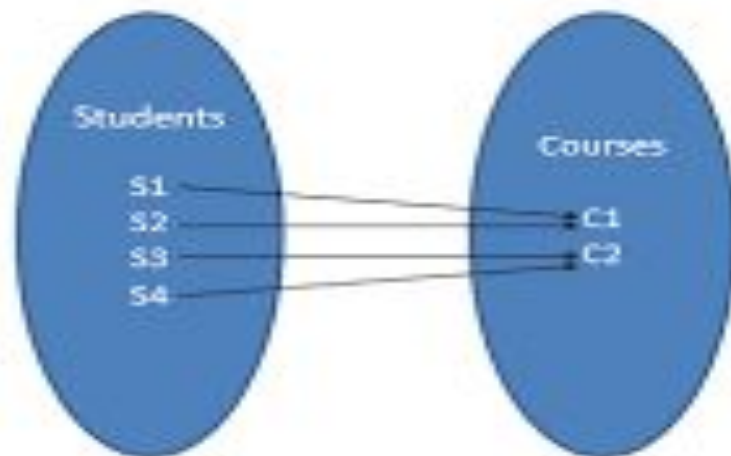
- 1-to-Many, Many-to-1
(1 : N, N : 1)
 - One entity can participate in **many** relationship instances



1- to - Many.



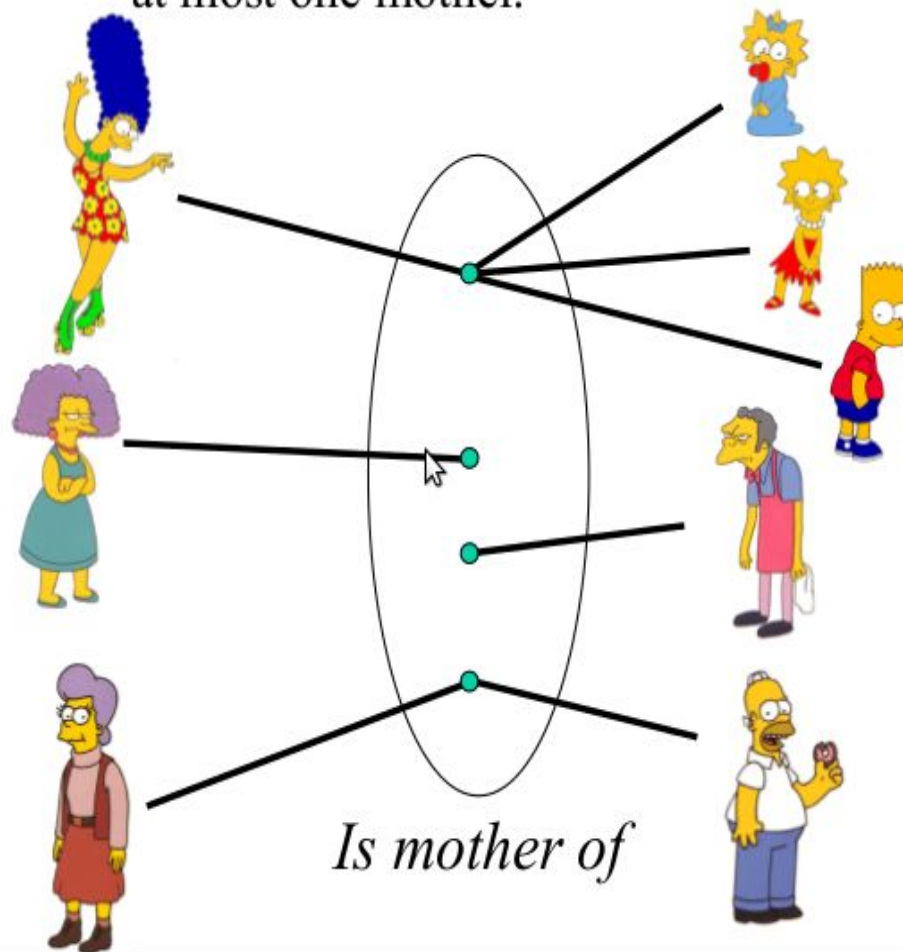
Many - to - 1



Example

•**One-to-many:** An entity in A is associated with any number in B. An entity in B is associated with at most one entity in A.

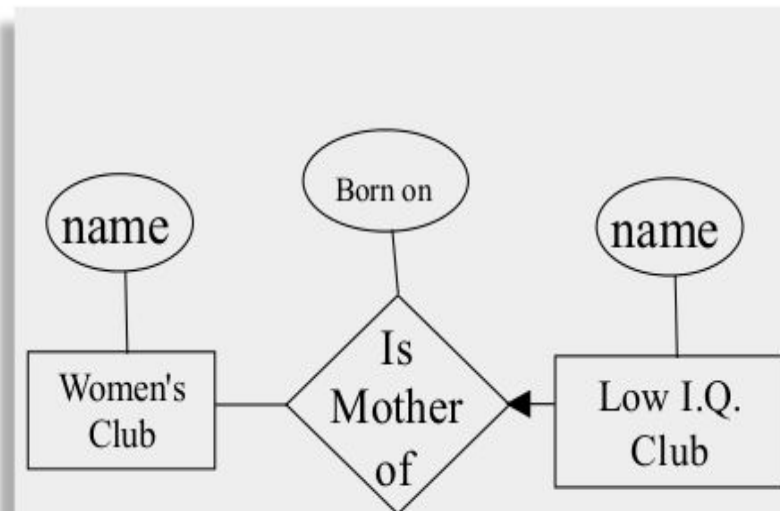
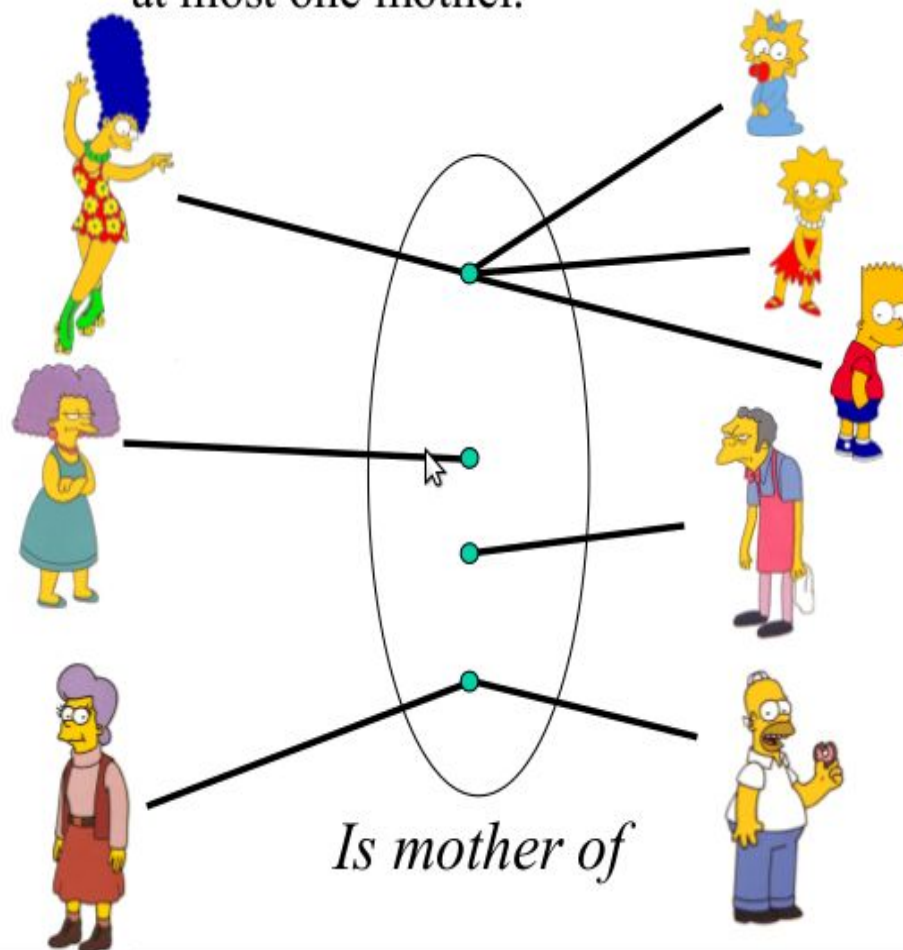
A woman may be the mother of many (or no) children. A person may have at most one mother.



Example

•**One-to-many:** An entity in A is associated with any number in B. An entity in B is associated with at most one entity in A.

A woman may be the mother of many (or no) children. A person may have at most one mother.

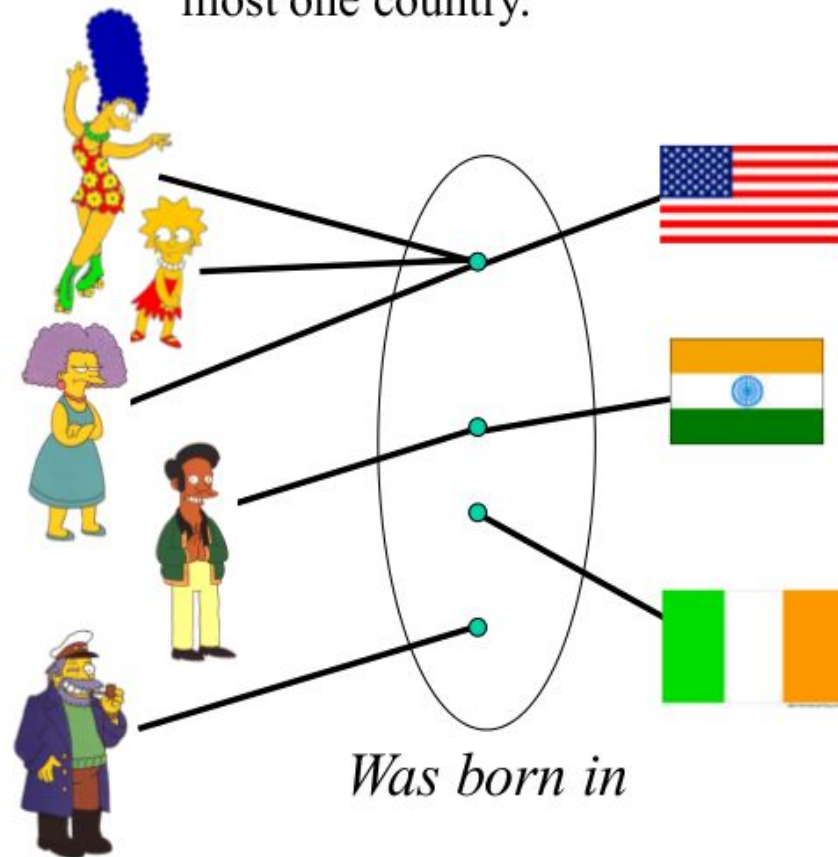


Note that this example is not saying that Moe does not have a mother, since we know as a biological fact that everyone has a mother. It is simply the case that Moe's mom is not a member of the Women's club.

Example

- **Many-to-one:** An entity in A is associated with at most one entity in B. An entity in B is associated with any number in A.

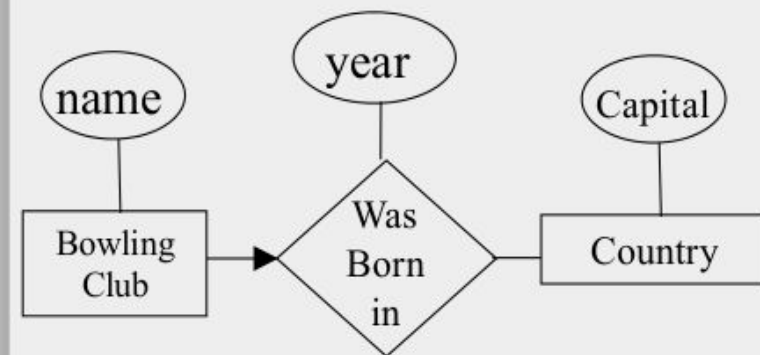
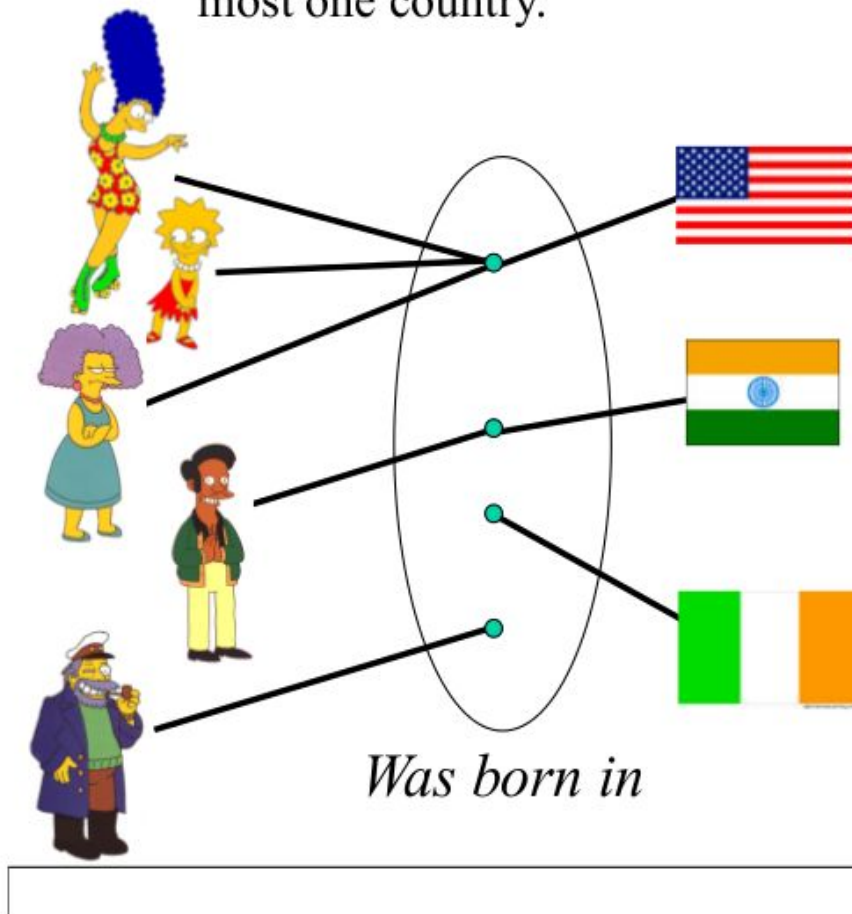
Many people can be born in any country, but any individual is born in at most one country.



Example

- **Many-to-one:** An entity in A is associated with at most one entity in B. An entity in B is associated with any number in A.

Many people can be born in any country, but any individual is born in at most one country.

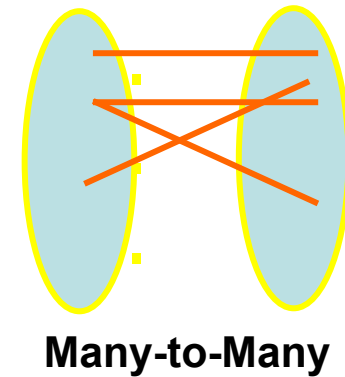


Note that we are not saying that the Sea Captain was not born in any country, he almost certainly was, we just don't know which country, or it is not in our Country entity set.

Also note that we are not saying that no one was born in Ireland, it is just that no one in the Bowling Club was.

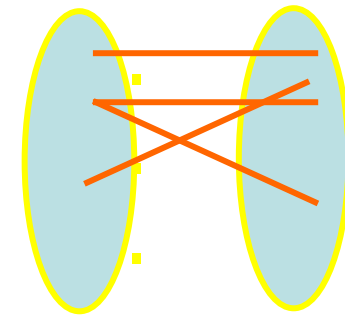
Possible Cardinality Ratios

- Many-to-Many (N: M)
 - Both entities can participate in **many** relationship instance

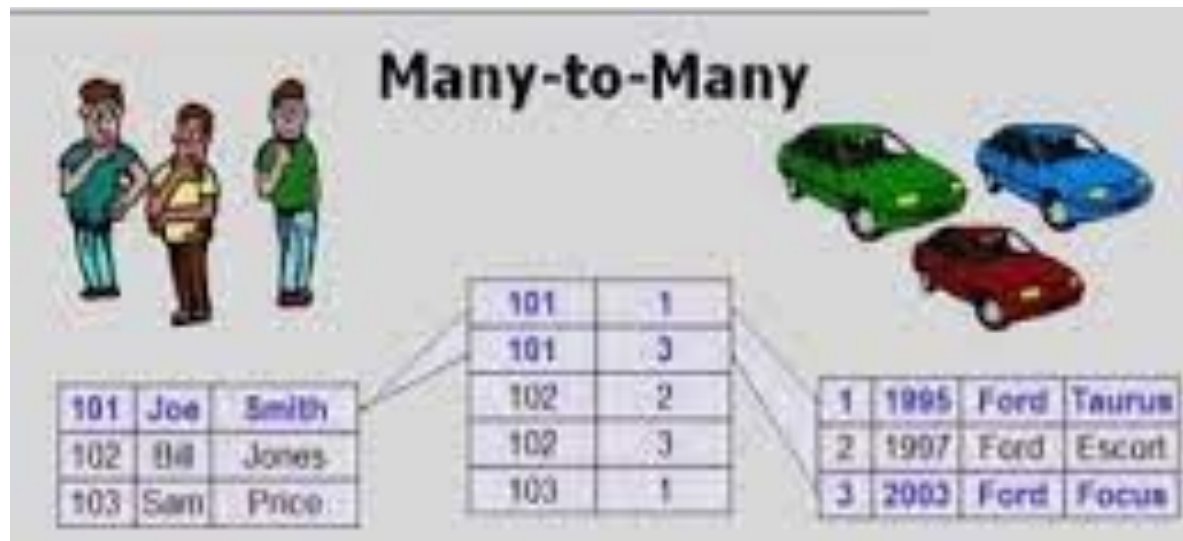


Possible Cardinality Ratios

- Many-to-Many (N: M)
 - Both entities can participate in **many** relationship instance

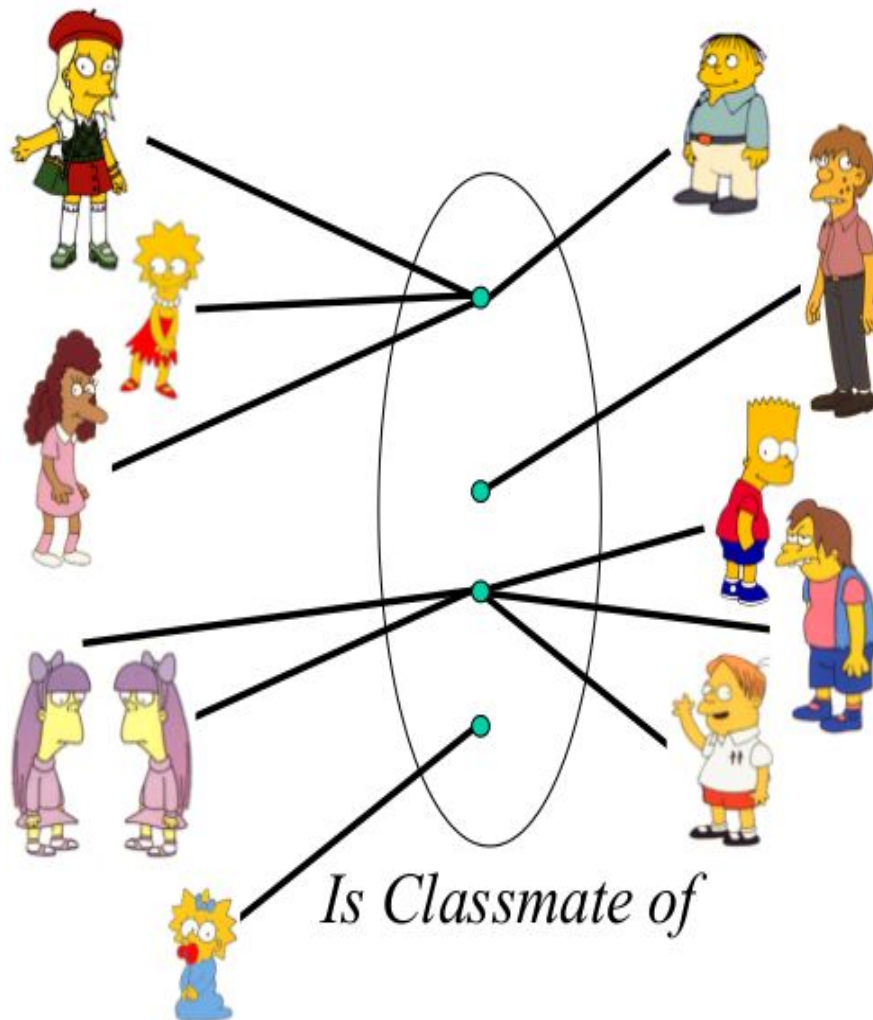


Many-to-Many



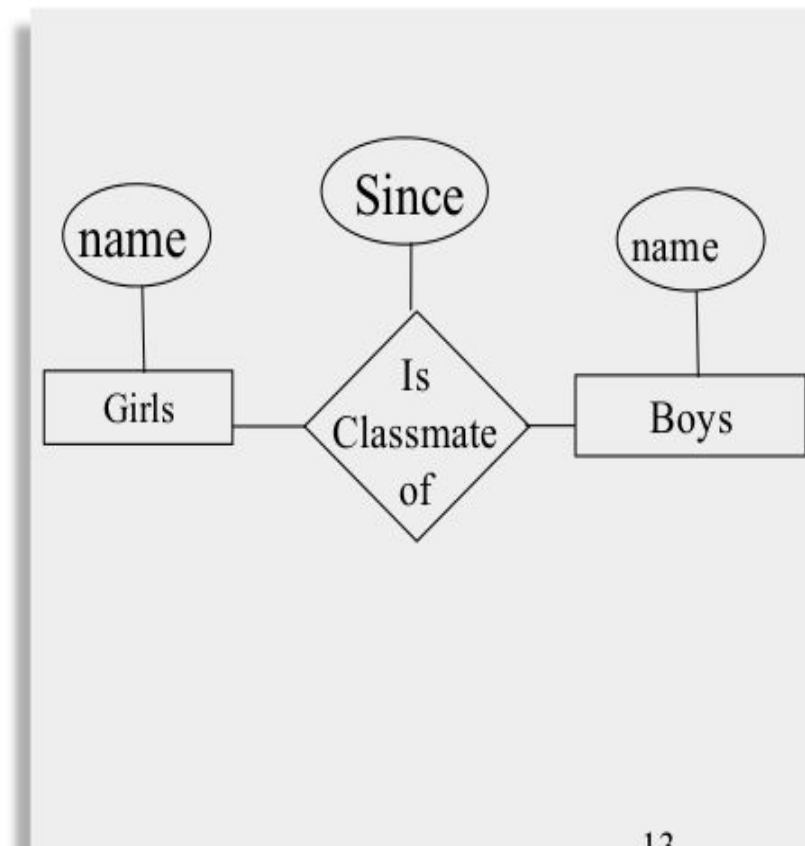
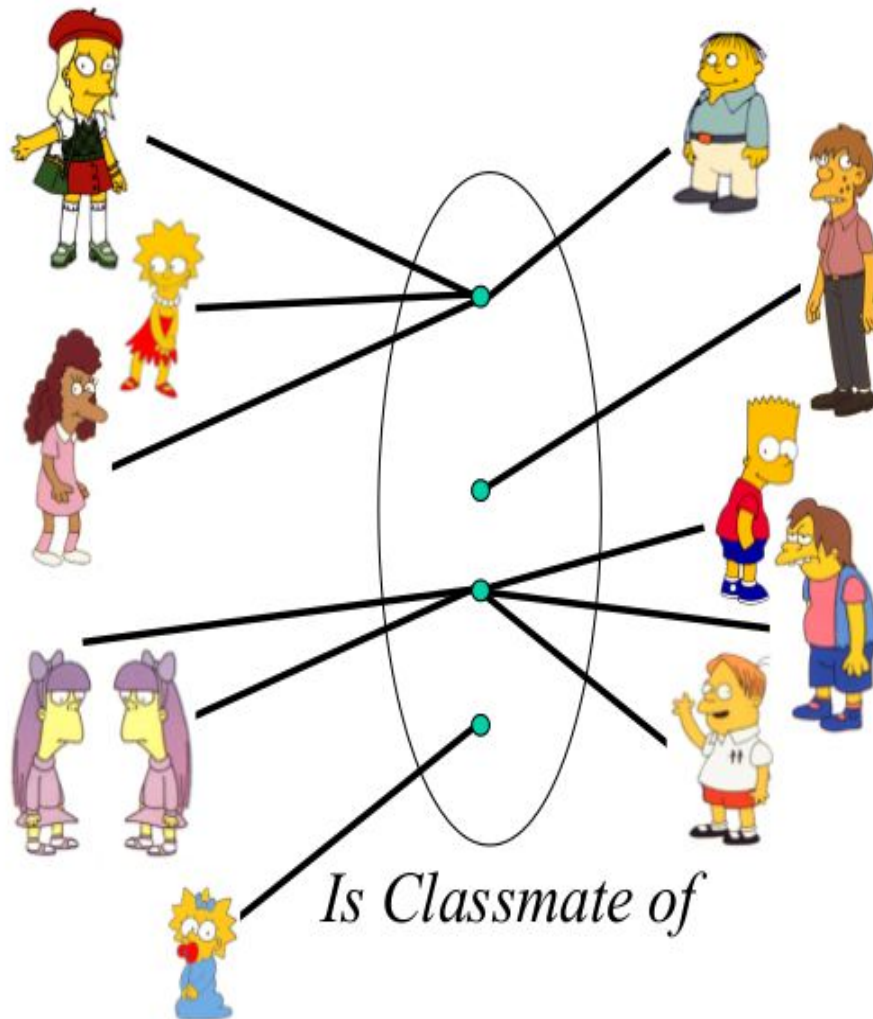
Example

- **Many-to-many:** Entities in A and B are associated with any number from each other.



Example

- **Many-to-many:** Entities in A and B are associated with any number from each other.

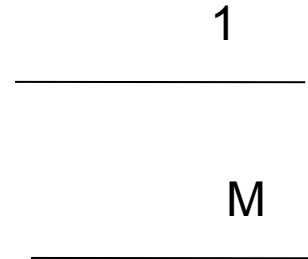


Connectivity

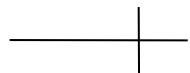
Chen Model

1 to represent one.

M to represent many



Crow's Foot

 One

 many

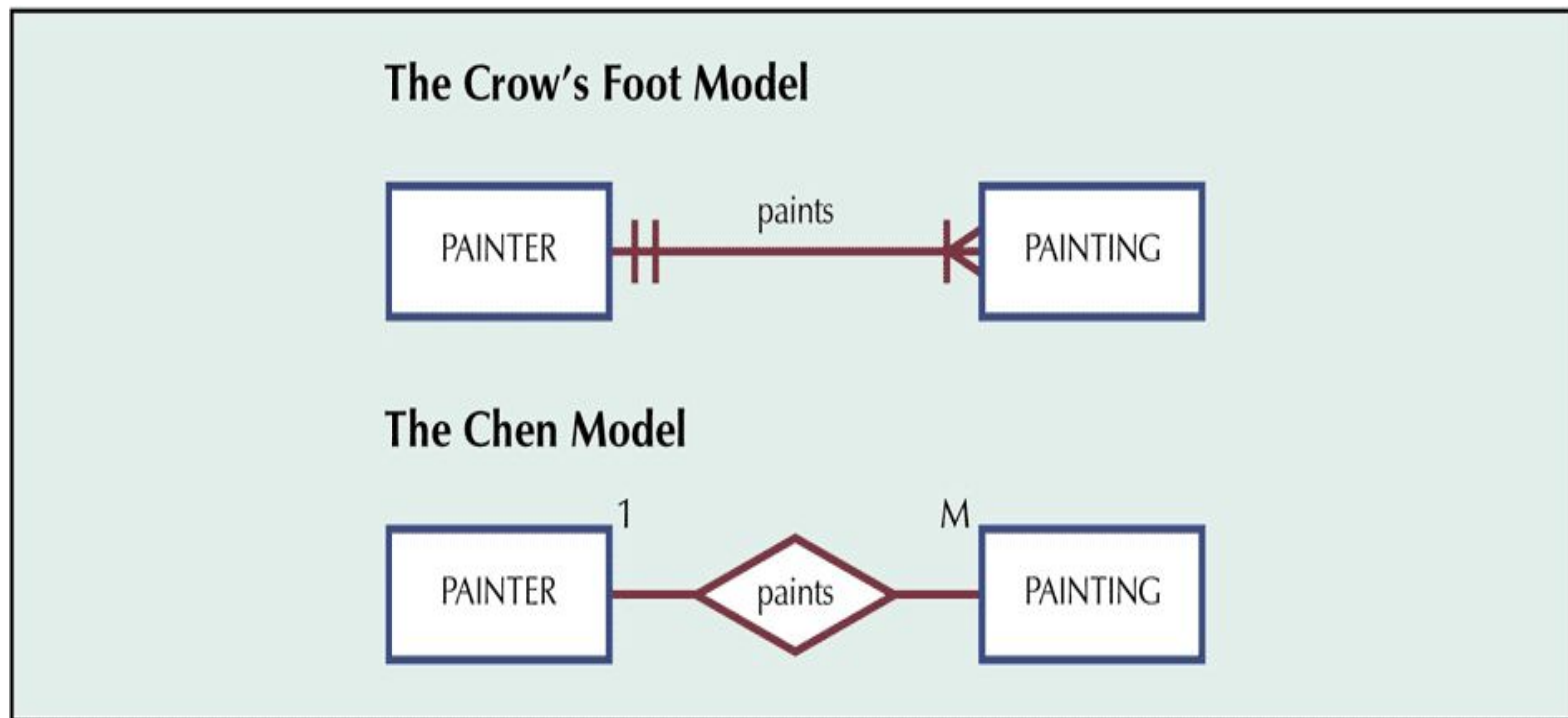
 One or many

 Mandatory one , means (1,1)

Binary Relationships

1:M relationship

Relational modeling ideal



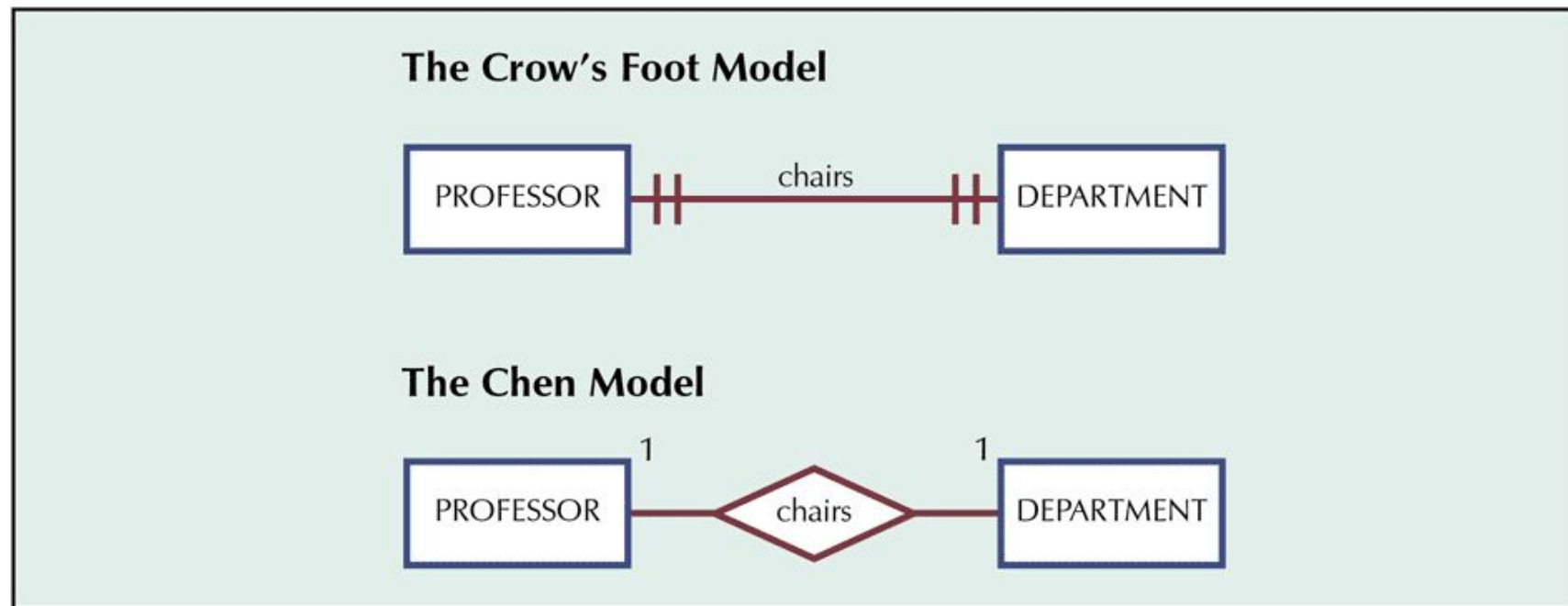
The 1: M relationship between PAINTER and PAINTING

Binary Relationships

1:1 relationship

Should be rare in any relational database design

A single entity instance in one entity class is related to a single entity instance in another entity class

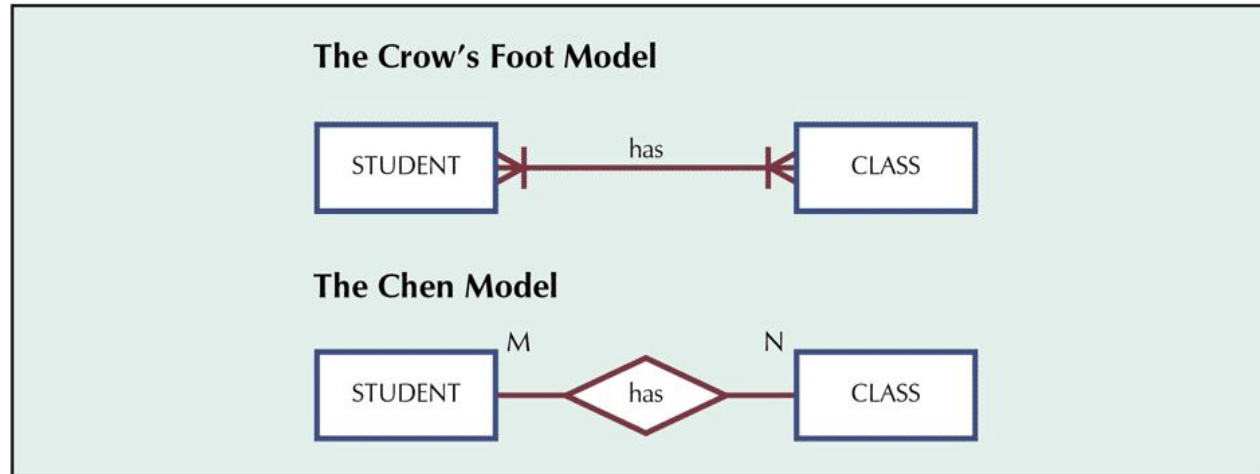


The 1:1 Relationship Between PROFESSOR and DEPARTMENT

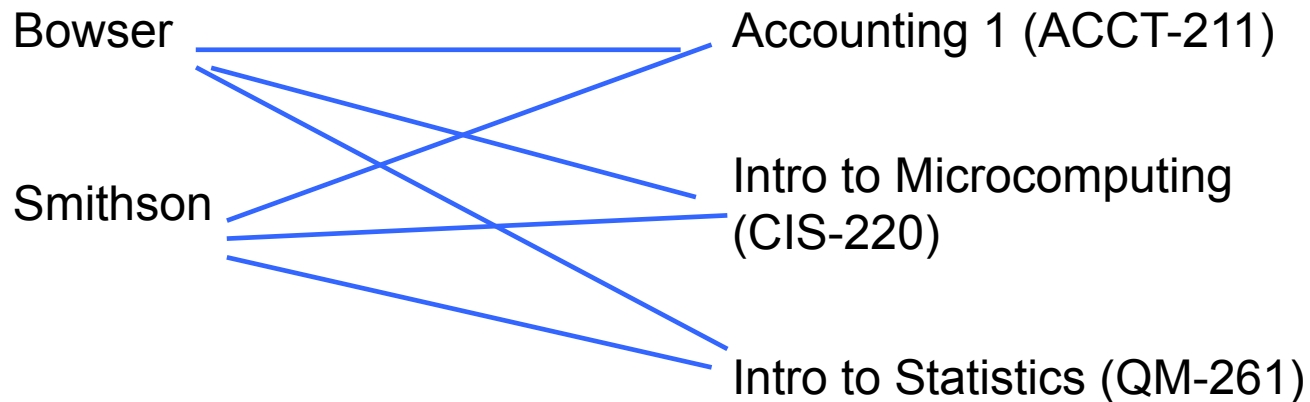
Binary Relationships

M:N relationships

Must be avoided because they lead to *data redundancies*.

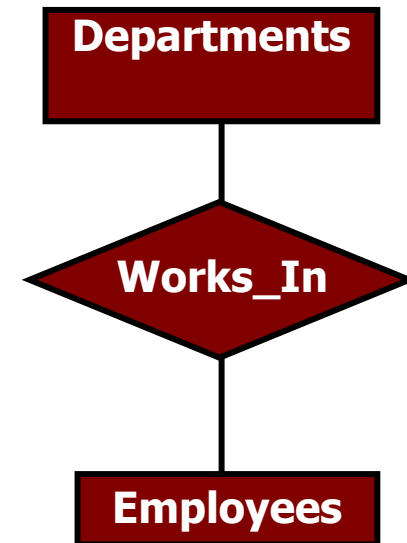


The M:N Relationship Between STUDENT and CLASS



Existence Dependency/ Participation Constraint

- Existence dependency indicates whether the existence of an entity depends on its relationship to another entity via the relationship type
- - Every employee must work for a department - EMPLOYEE is existentially dependent on DEPARTMENT via the Works In relationship type



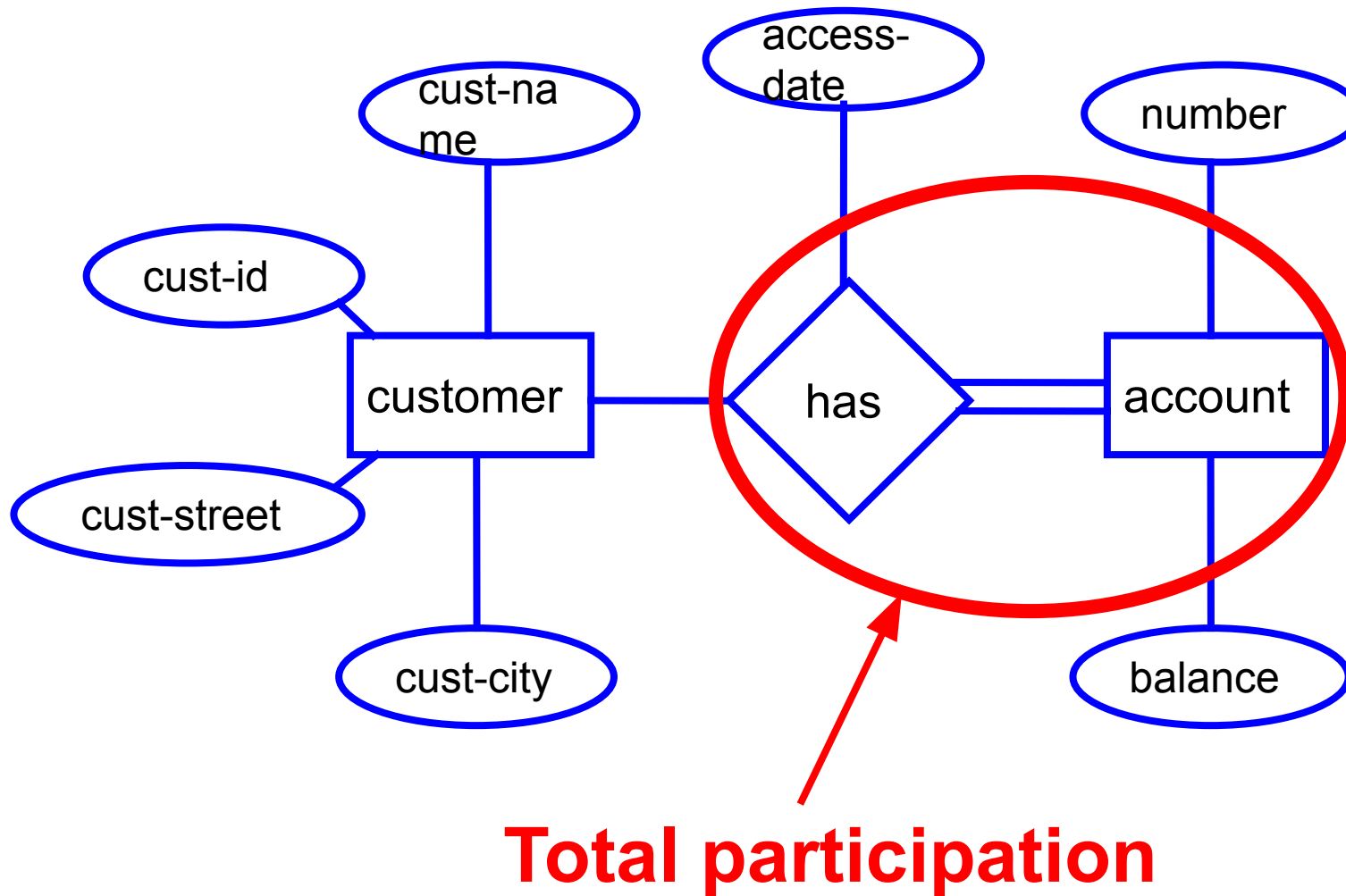
Kinds of participating constraints

- TOTAL Participation (Existence Dependency)
Constraint : Every employee must work for a department
- PARTIAL Participation
Constraint : Not every employee is a manager

Participation Constraint

- Given an entity set E , and a relationship R it participates in:
 - If every entity in E participates in at least one relationship in R , it is total participation
 - partial otherwise

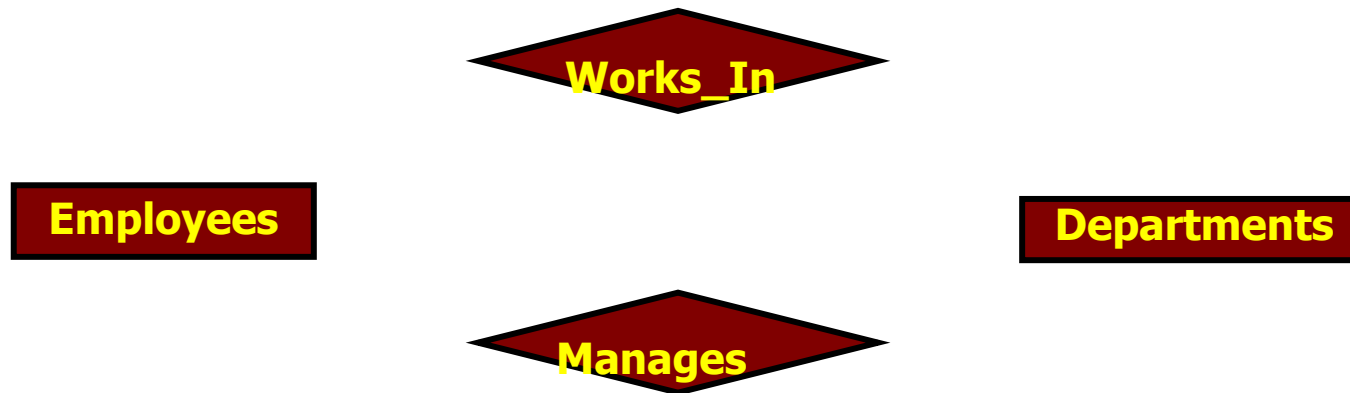
Participation Constraint



Representing Participation

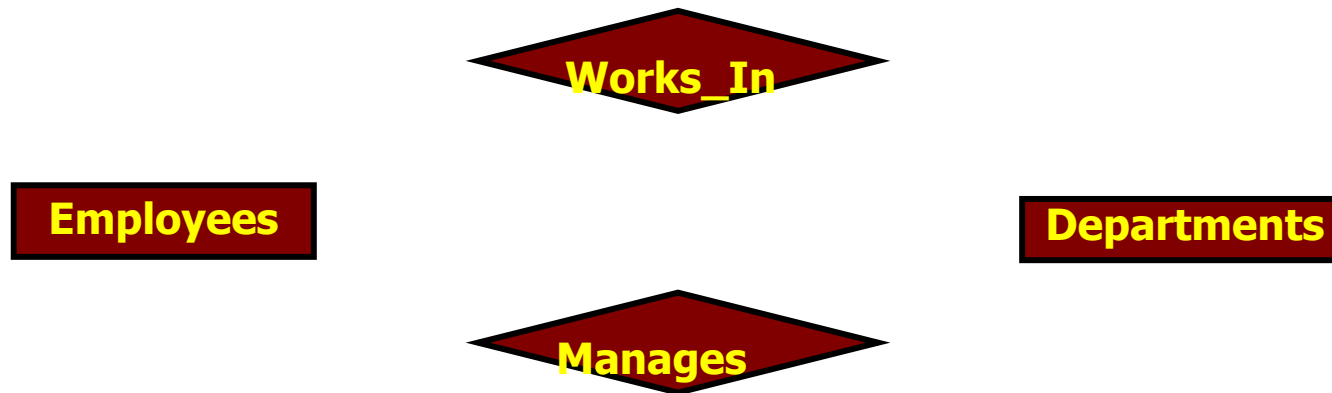
Every employee must work for a department
Every department must have a manager
Every department must have employees
Not every employee is a manager

Representing Participation



Every employee must work for a department
Every department must have a manager
Every department must have employees
Not every employee is a manager

Representing Participation



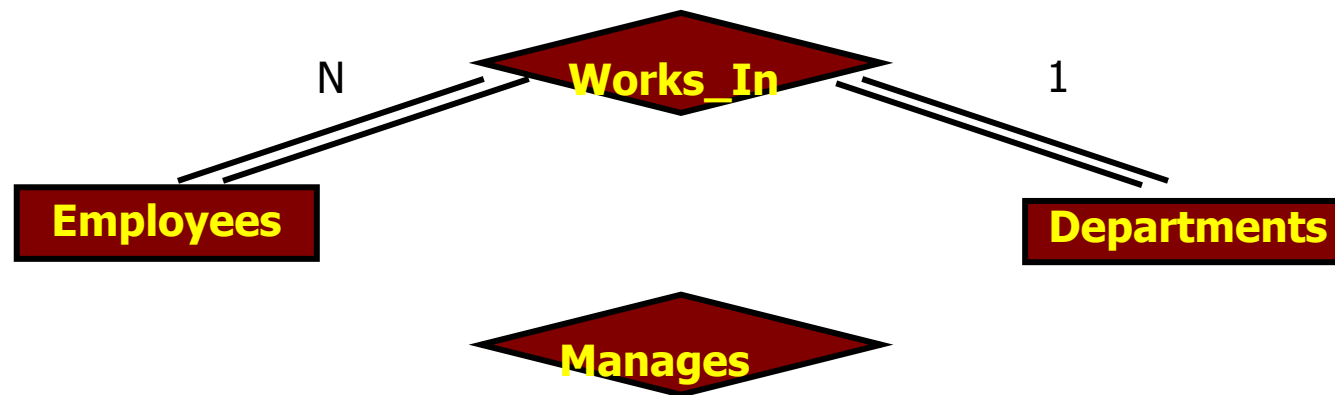
Every employee must work for a department

Every department must have a manager

Every department must have employees

Not every employee is a manager

Representing Participation



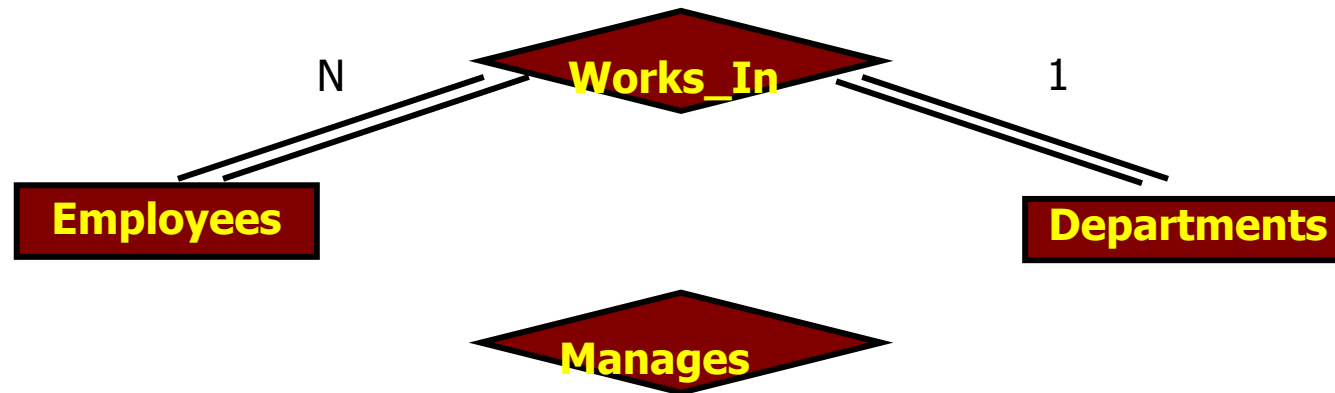
Every employee must work for a department

Every department must have a manager

Every department must have employees

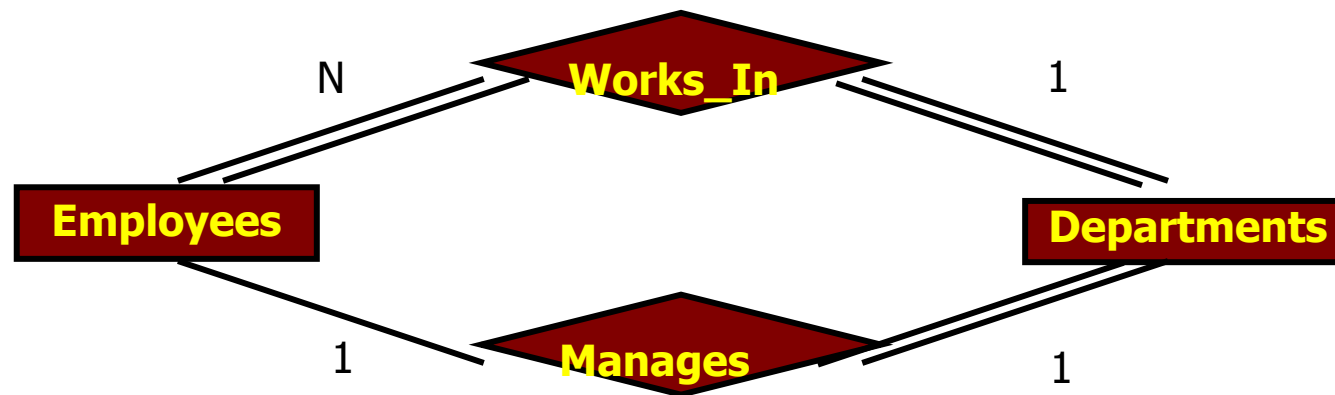
Not every employee is a manager

Representing Participation



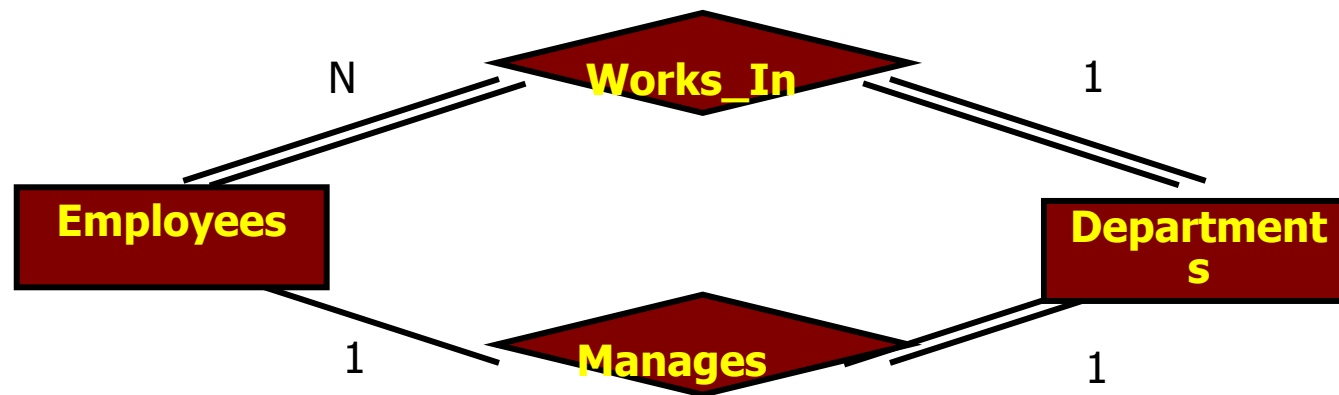
Every employee must work for a department
Every department must have a manager
Every department must have employees
Not every employee is a manager

Representing Participation



Every employee must work for a department
Every department must have a manager
Every department must have employees
Not every employee is a manager

Representing Participation

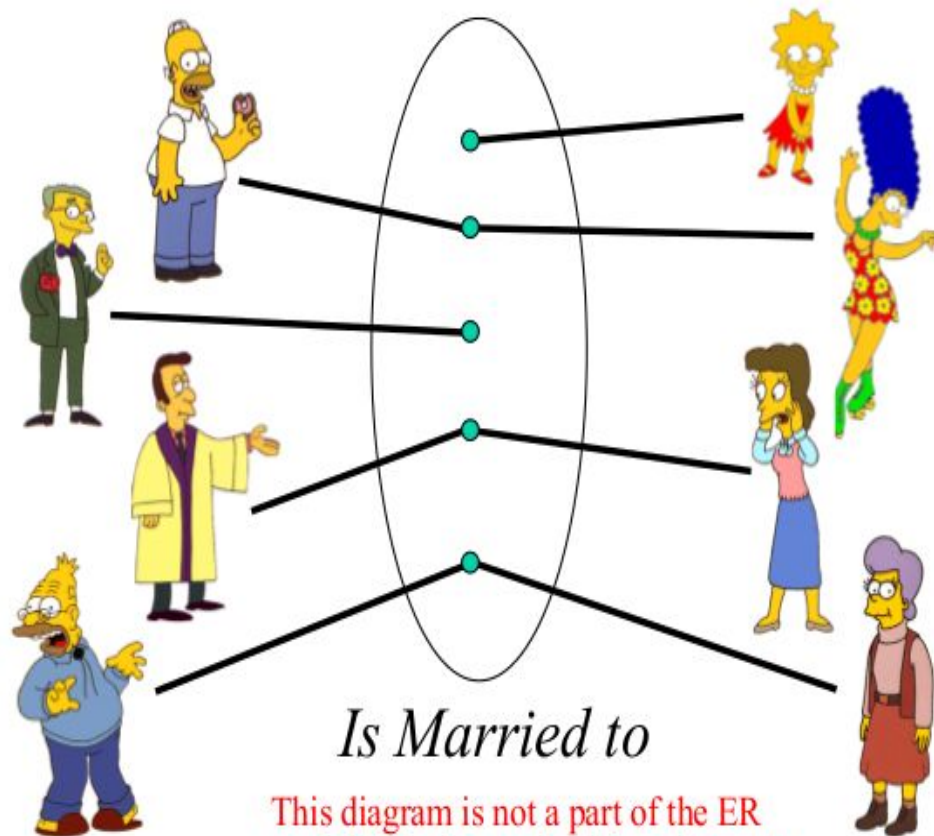


Every employee must work for a department
Every department must have a manager
Every department must have employees
Not every employee is a manager

Attributes of Relationship Types

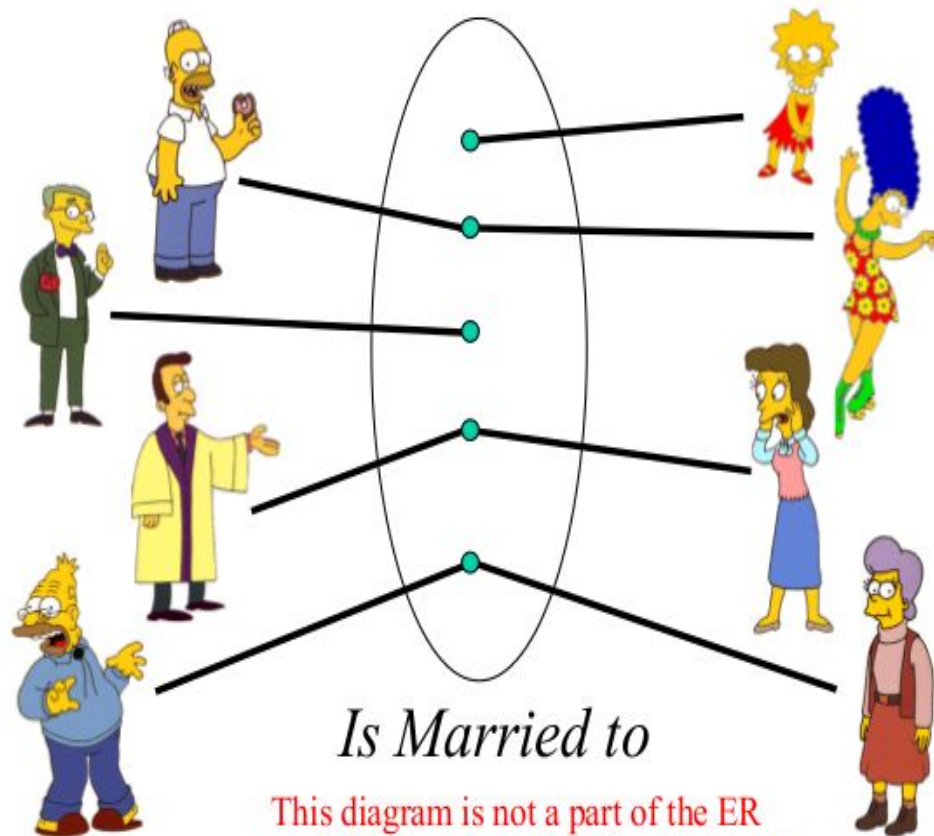
- Relationship types can also have attributes
- Example: To record since when Men is married to a Women
- Draw ER diagram for above example.

Attributes of Relationship Types

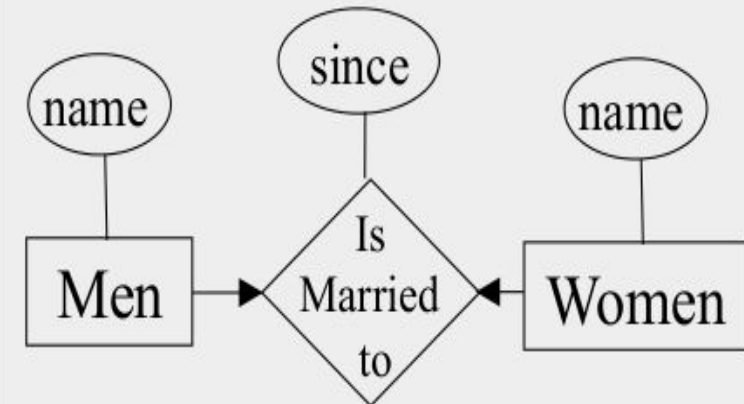


This diagram is not a part of the ER model! It is just an intuitive picture to explain a concept

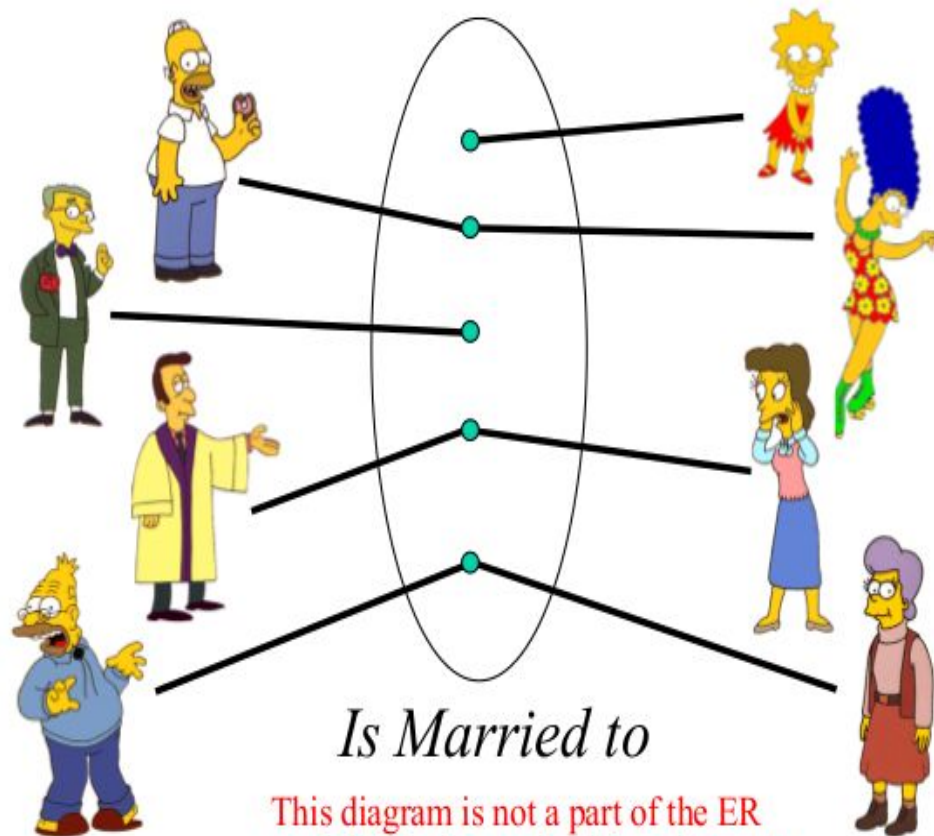
Attributes of Relationship Types



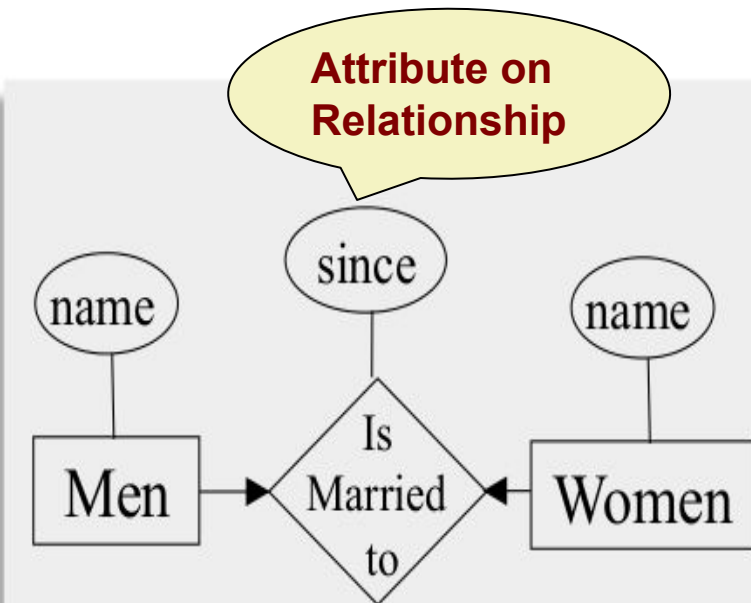
This diagram is not a part of the ER model! It is just an intuitive picture to explain a concept



Attributes of Relationship Types



This diagram is not a part of the ER model! It is just an intuitive picture to explain a concept



Weak Entity Types

An entity set without enough attributes to have a primary key

E.g. Transaction Entity

Attributes:

transaction-number, transaction-date,
transaction-amount, transaction-type

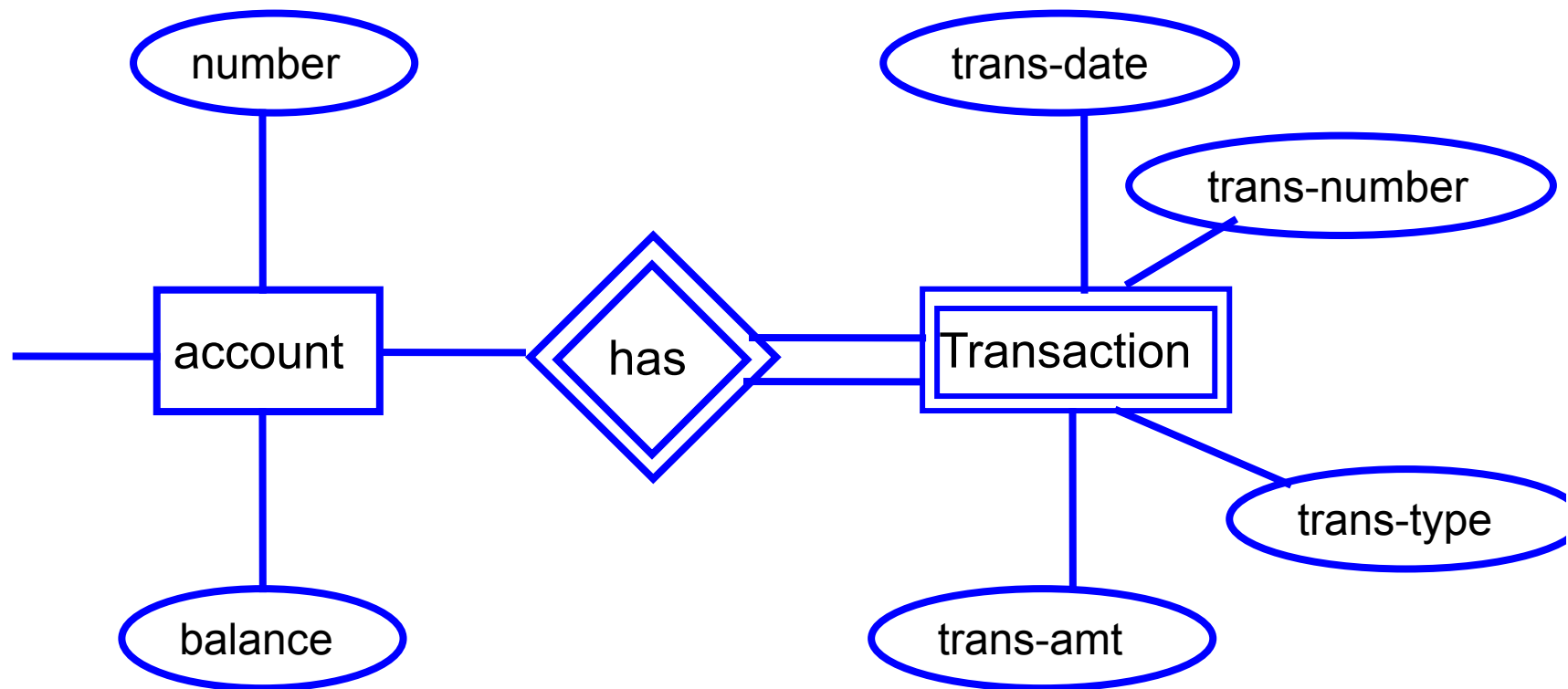
transaction-number: may not be unique across
accounts

Weak Entity Types

- A weak entity set must be associated with an identifying or owner entity set
- Account is the owner entity set for Transaction
- Weak Entity Type always has a total participation constraint with respect to its identifying relationship

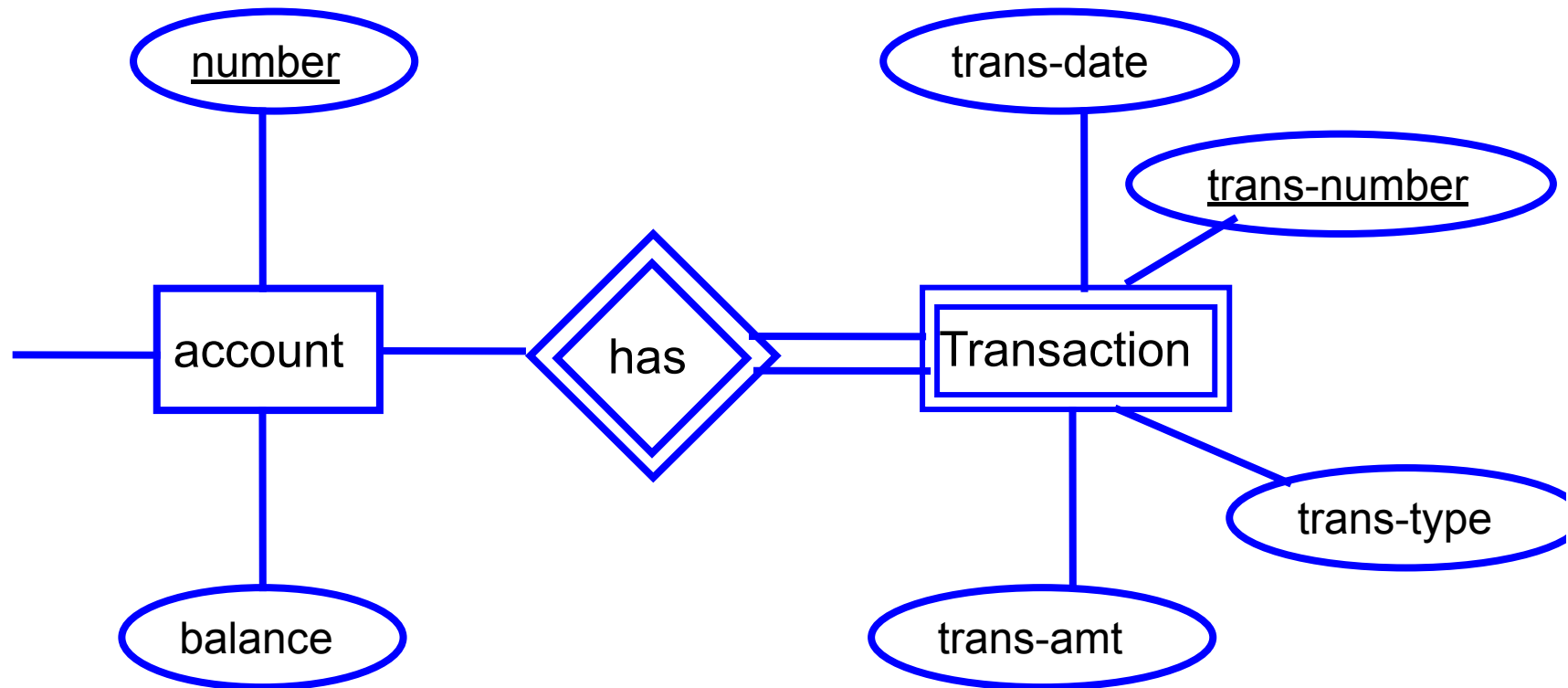
Weak Entity Types

Still need to be able to distinguish between different weak entities associated with the same strong entity



Weak Entity Types

Discriminator: A set of attributes that can be used for that



Differences between Strong Entity and Weak Entity.

Sr. No.	Key	Strong Entity	Weak Entity
1	Key		
2	Dependency		
3	Represented by		
4	Relationship Representation		
5	Participation		

Differences between Strong Entity and Weak Entity.

Sr. No.	Key	Strong Entity	Weak Entity
1	Key	Strong entity always have one primary key.	Weak entity have a foreign key referencing primary key of strong entity.
2	Dependency		
3	Represented by		
4	Relationship Representation		
5	Participation		

Differences between Strong Entity and Weak Entity.

Sr. No.	Key	Strong Entity	Weak Entity
1	Key	Strong entity always have one primary key.	Weak entity have a foreign key referencing primary key of strong entity.
2	Dependency	Strong entity is independent of other entities.	Weak entity is dependent on strong entity.
3	Represented by		
4	Relationship Representation		
5	Participation		

Differences between Strong Entity and Weak Entity.

Sr. No.	Key	Strong Entity	Weak Entity
1	Key	Strong entity always have one primary key.	Weak entity have a foreign key referencing primary key of strong entity.
2	Dependency	Strong entity is independent of other entities.	Weak entity is dependent on strong entity.
3	Represented by	A strong entity is represented by single rectangle.	A weak entity is represented by double rectangle.
4	Relationship Representation		
5	Participation		

Differences between Strong Entity and Weak Entity.

Sr. No.	Key	Strong Entity	Weak Entity
1	Key	Strong entity always have one primary key.	Weak entity have a foreign key referencing primary key of strong entity.
2	Dependency	Strong entity is independent of other entities.	Weak entity is dependent on strong entity.
3	Represented by	A strong entity is represented by single rectangle.	A weak entity is represented by double rectangle.
4	Relationship Representation	Relationship between two strong entities is represented by single diamond.	Relationship between a strong and weak entity is represented by double diamond.
5	Participation		

Differences between Strong Entity and Weak Entity.

Sr. No.	Key	Strong Entity	Weak Entity
1	Key	Strong entity always have one primary key.	Weak entity have a foreign key referencing primary key of strong entity.
2	Dependency	Strong entity is independent of other entities.	Weak entity is dependent on strong entity.
3	Represented by	A strong entity is represented by single rectangle.	A weak entity is represented by double rectangle.
4	Relationship Representation	Relationship between two strong entities is represented by single diamond.	Relationship between a strong and weak entity is represented by double diamond.
5	Participation	Strong entity may or may not participate in entity relationships.	Weak entity always participates in entity relationships.

How to Evaluate a Data Model?

A good data model has the following:

- Accuracy and completeness
- Non redundancy
- Enforcement of business rules
- Data Reusability
- Stability and Flexibility
- Communication Effectiveness
- Simplicity