



The Entity-Relationship Model

- Chapter 2

Database Models

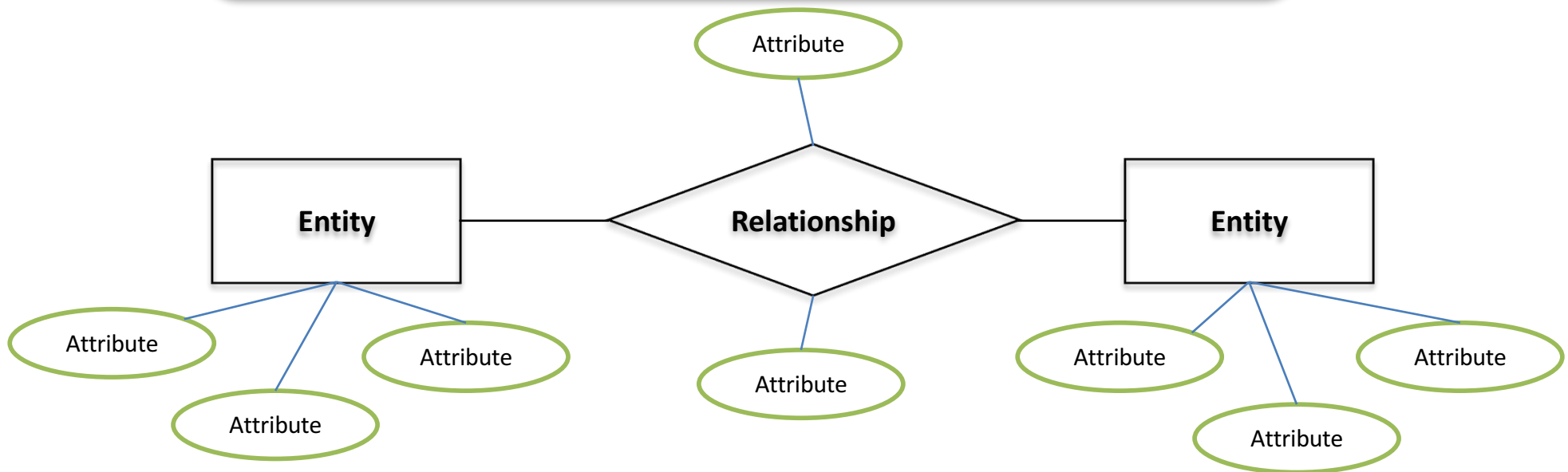
Database model: determines in which manner data can be stored, organized and manipulated in a database system

Common Models:

- **Relational Model (implemented in Databases)**
- **Entity Relationship Model (useful for discussing with end-users)**
- Network Model
- Object Model
-

ER Model

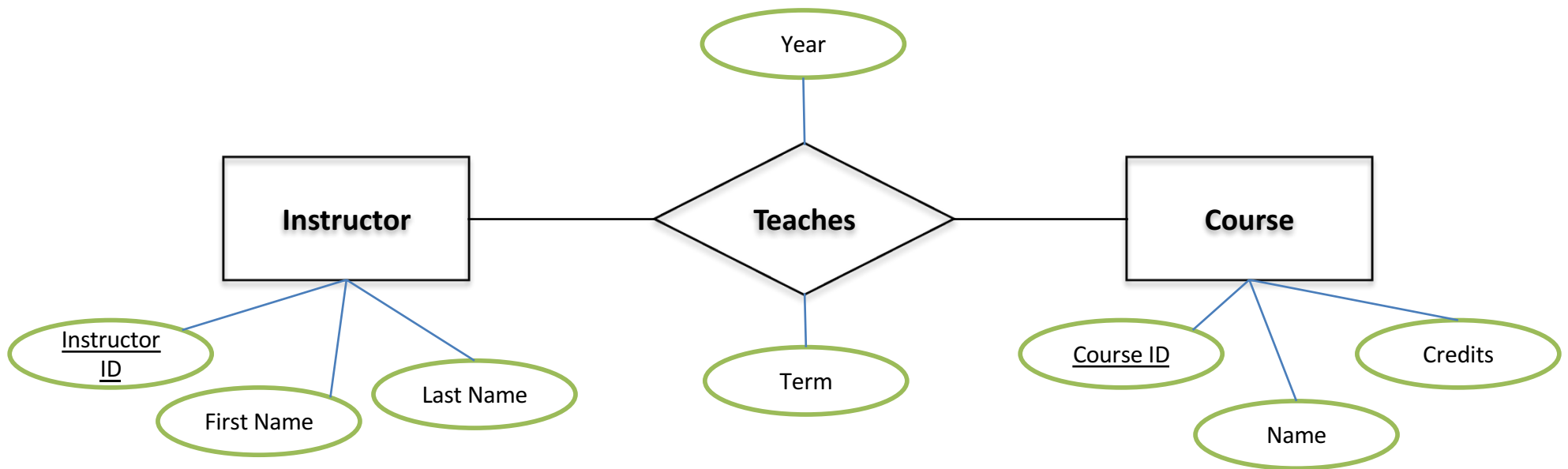
Describes the data as a set of **entities** and **relationships**



This diagram is called the **Entity-Relationship (ER) Diagram**

Entity Relationship Diagram

Example



Relational Database

Entities and (most) relationships are translated into Tables

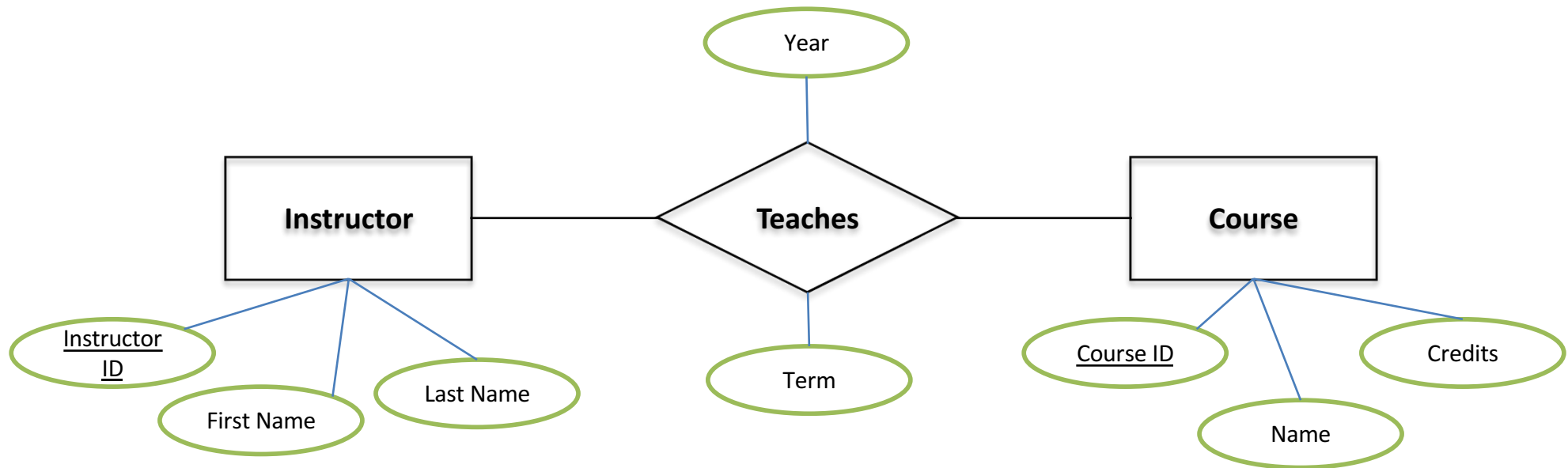
Columns (Fields) (Attributes)

The diagram shows a table with three columns and five rows. A blue bracket above the columns is labeled 'Columns (Fields) (Attributes)'. A blue bracket to the left of the rows is labeled 'Rows (Records) (Tuples)'. The table has a header row with green backgrounds and five data rows with light green backgrounds. Each data row contains five dots representing data values.

Attribute 1	Attribute 2	Attribute 3
.....
.....
.....
.....
.....

Entity or Relationship

Translation from ER Diagrams to Relations



Instructor ID	First Name	Last Name
394953	John	Smith
454544	Sara	King
439849	Alex	Dee
....
....

Instructor ID	Course ID	Year	Term
454544	E302	2009	F
394953	C210	2010	W
439849	M184	2010	F
....
....

Course ID	Name	Credits
M184	Calculus	3
C210	Physics	4
E302	Algorithms	4
....
....

Database Design using ER modeling

Database Design

- Our goal is to avoid the really bad mistakes and design clean and easily understood databases
- Database design starts with a picture (E-R diagram)

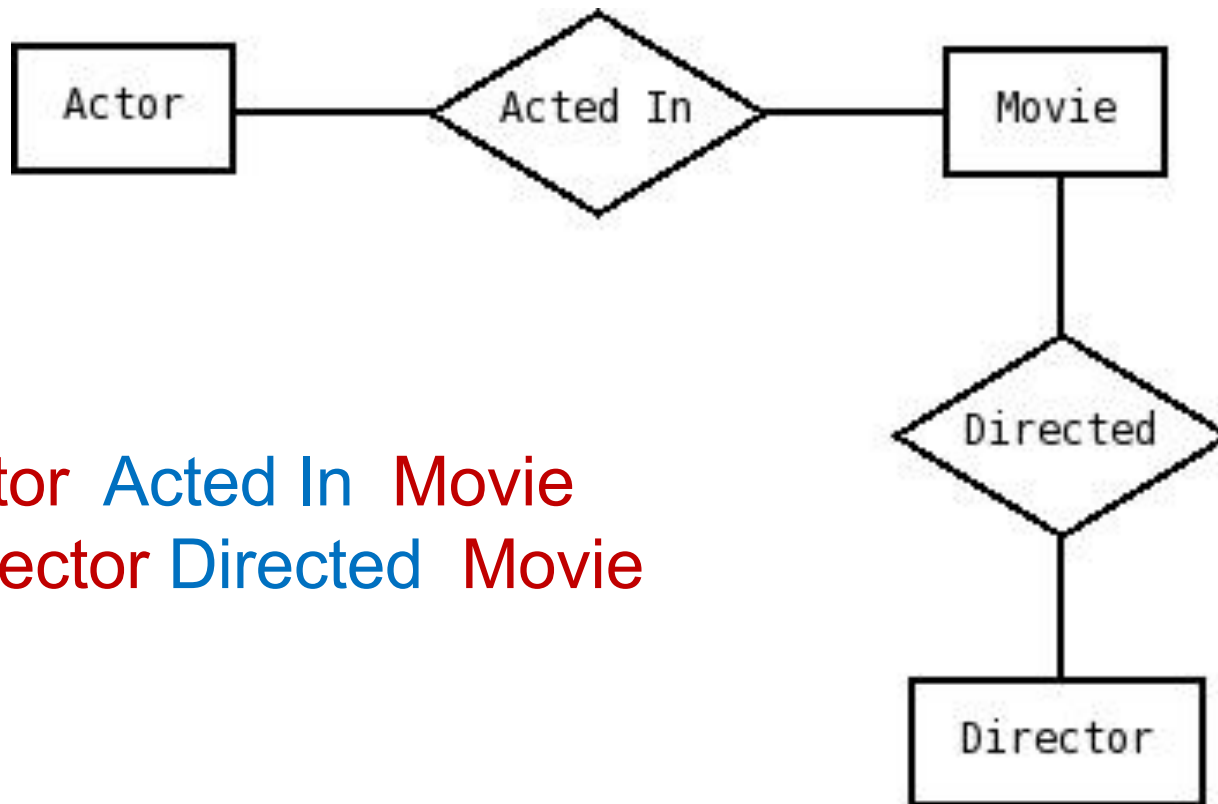
ER Diagram



Entities: 'Actor' and 'Movie' (represented by boxes)

Relationship: 'Acted In' (represented by diamonds)

ER Diagram



Actor Acted In Movie
Director Directed Movie

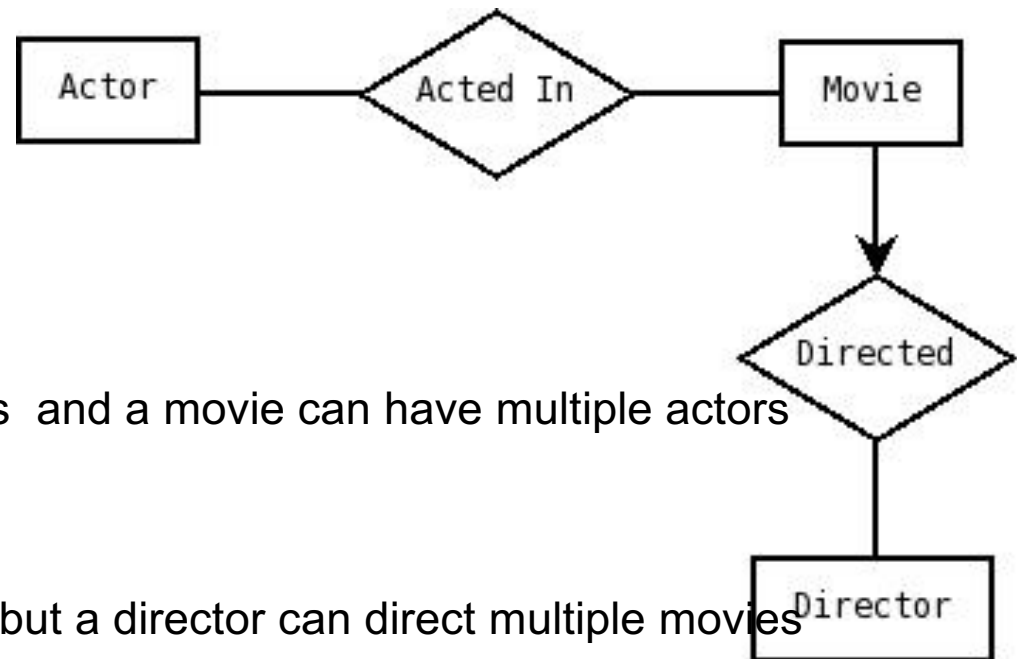
ER Diagram

Relationship Types

- Many-to-Many
- One-to-Many (or Many-to-one)
- One-to-One

ER Diagram

What are the relationship types here?



Acted In: Many-to-Many

An actor can act in multiple movies and a movie can have multiple actors

Directed: One-to-Many

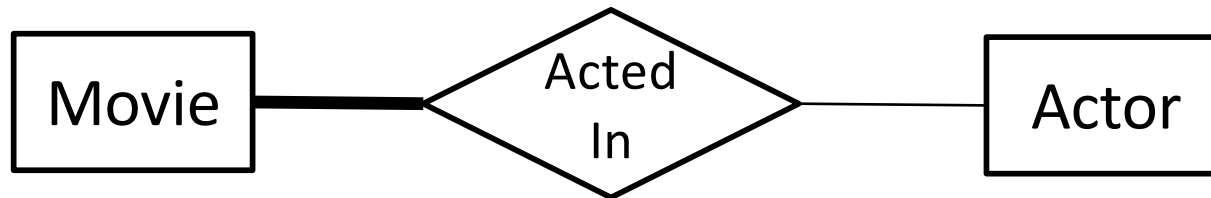
A movie has at most **one** director, but a director can direct multiple movies

Notice: Think of the arrow in the ER diagram as saying that a movie can *map* to at most one director

The arrow indicates that there is a **Key Constraint** on Directed relationship: a movie in the relationship must be unique.

ER Diagram

Participation Constraints



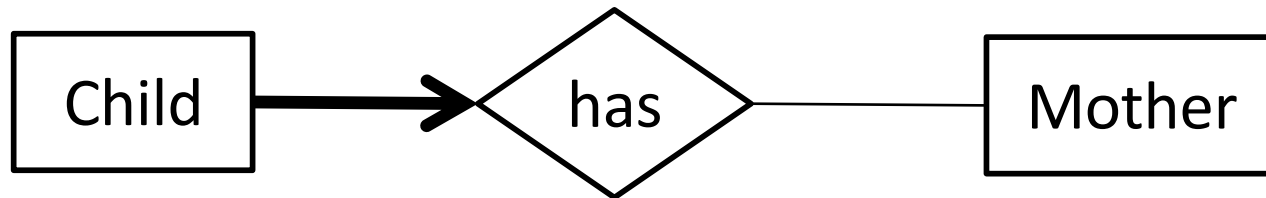
Bold line means participation constraint – every Movie entity must participate in a relationship with an Actor.

In other words, every movie must have *at least* one actor.

On the other hand, an actor can participate in 0 or more movies. With the above design, there can be actors who have not yet acted in movies.

ER Diagram

Combining Participation and Key Constraints



Key-Constraint:

A child has at most one mother but a mother may have several children

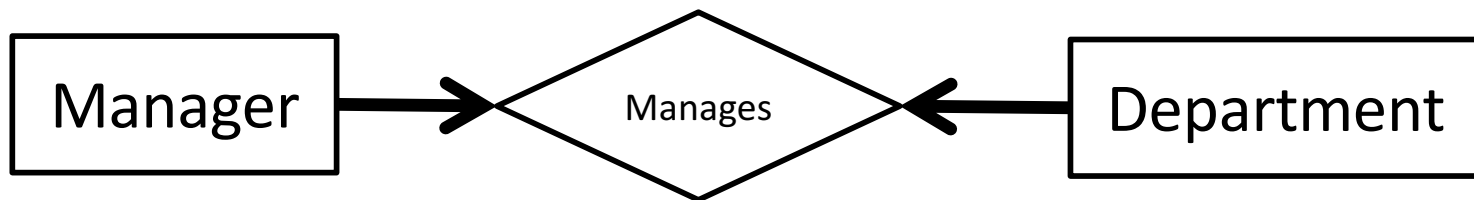
Participation constraint:

Each child must have at least one mother

Net Result: Every child has exactly one mother.

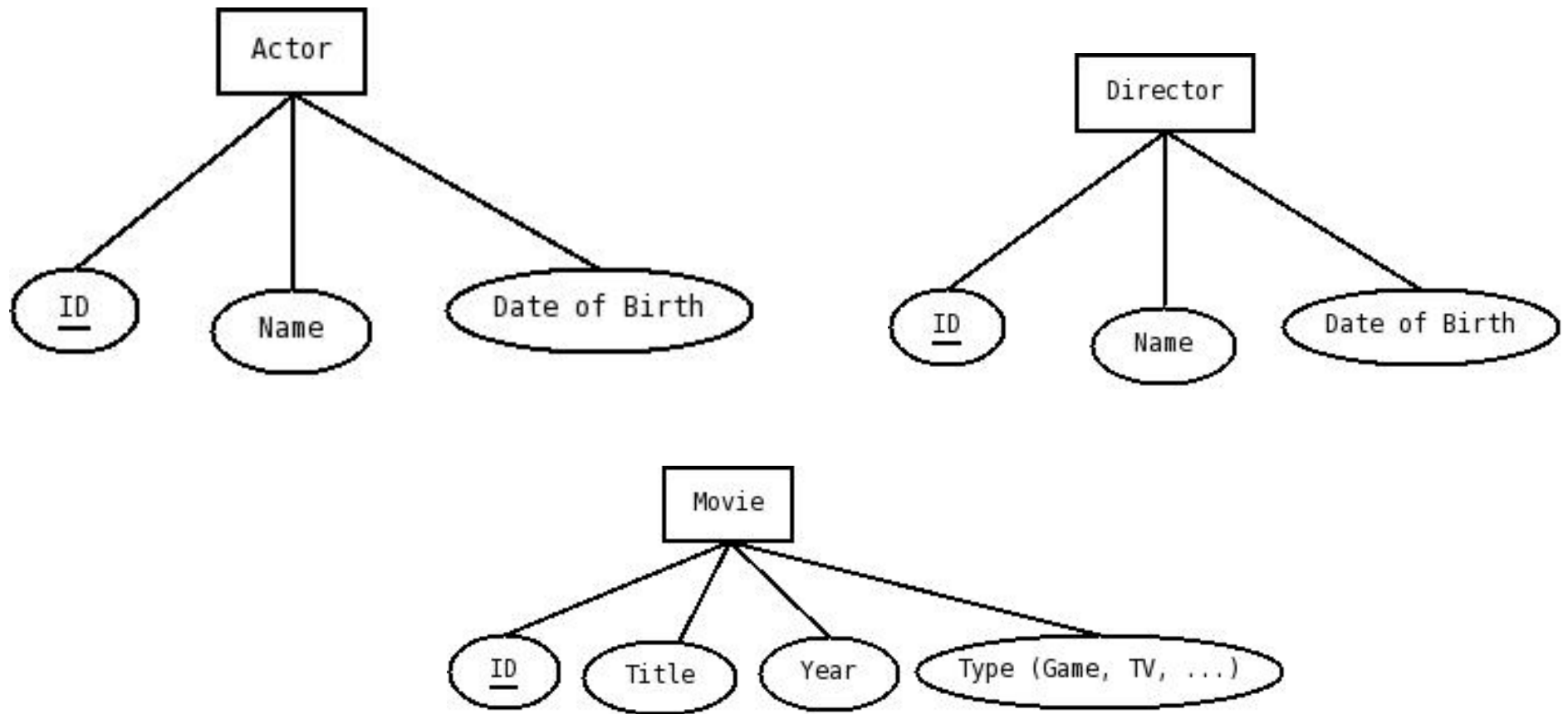
ER Diagram

What is the relationship type here?



ER Diagram

Attributes: Represented by ovals



Keys

Key: a set of one or more attributes that has unique value for each record

- Students in a student database have multiple potential keys (they must be unique to be a key):
 - Student ID
 - Login name
 - SSN
 - (Name, address)
- These are called **candidate** keys

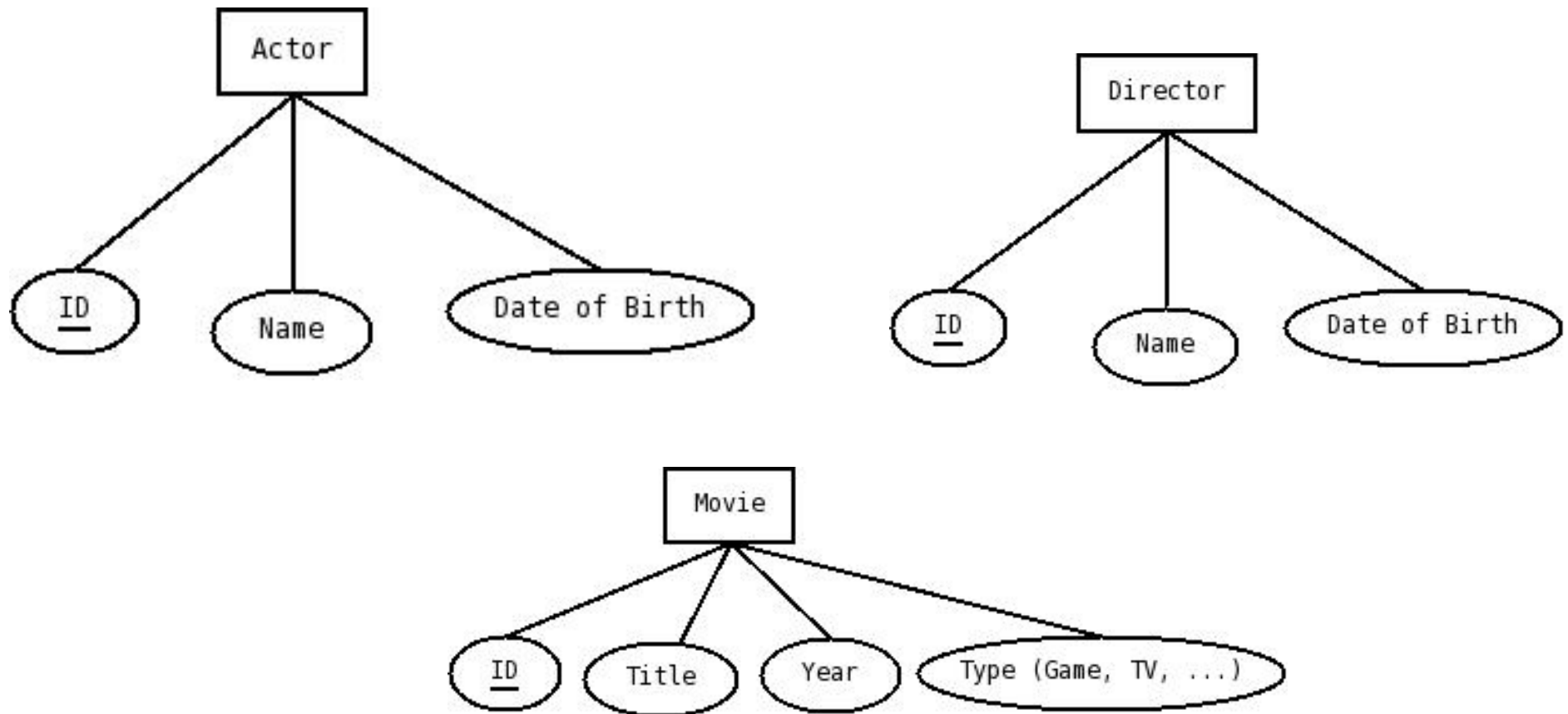
Primary Key

Primary Key: One of the candidate keys

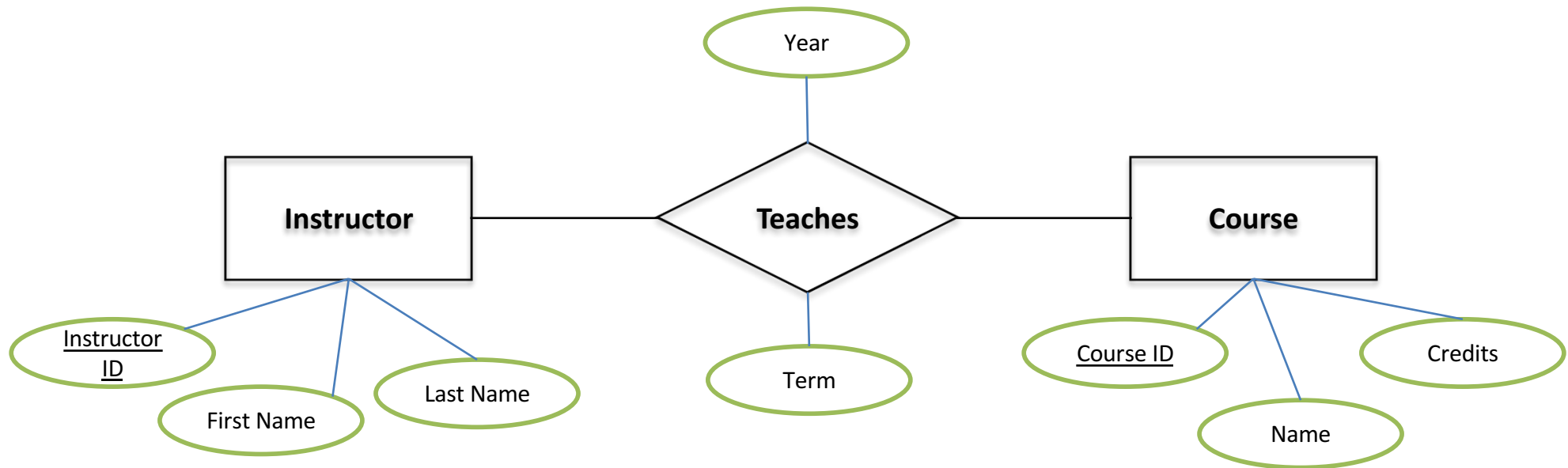
- As a database designer, you will choose one of the candidate keys as the **primary** key to identify a row in a table. This key is cross-referenced in other tables to represent relationships
- For students, **Student ID** is a good primary key
- The primary key attribute(s) is (are) underlined in the ER diagram

Primary keys

Often, each Entity is assigned a unique **ID**, which serves as a primary key.



What are the Primary Keys for Instructor? Course? Teaches?



Instructor ID	First Name	Last Name
394953	John	Smith
454544	Sara	King
439849	Alex	Dee
....
....

Instructor ID	Course ID	Year	Term
454544	E302	2009	F
394953	C210	2010	W
439849	M184	2010	F
....
....

Course ID	Name	Credits
M184	Calculus	3
C210	Physics	4
E302	Algorithms	4
....
....



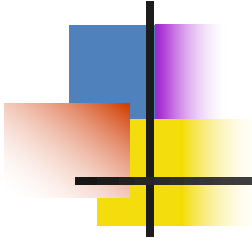
Practice ER Modeling Problem

- Citizens vote on initiatives at polling stations
- Polling stations have a location and size.
 - Citizen info: SSN, name, bday (SSN is unique)
 - Initiatives info: Id, name, description (Id is unique)
- Citizens are represented by an elected official, who is also a citizen
- Exercise:
 - identify entities (objects), relationships between entities
 - Attach attributes
 - Keys: something that uniquely identify an attribute



Practice ER Modeling Problem

- First step: Identify the **entities**.
- **Citizens** vote on **initiatives** at **polling stations**
- **Polling stations** have a location and size.
 - Citizen info: SSN, name, bday (SSN is unique)
 - Initiatives info: Id, name, description (Id is unique)
- **Citizens** are represented by an elected official, who is also a **citizen**
- Exercise:
 - Identify entities (objects), attach attributes
 - Identify or add key for each attribute
 - Identify relationships



Entities

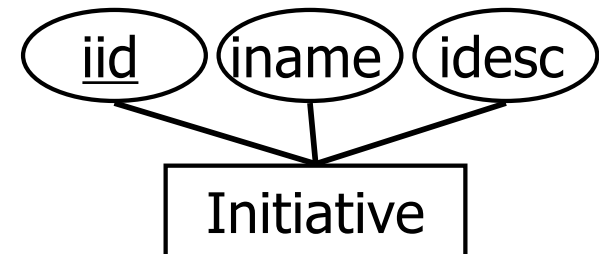
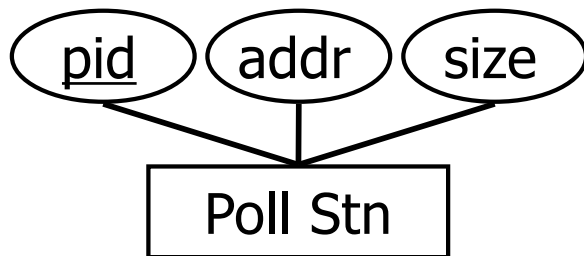
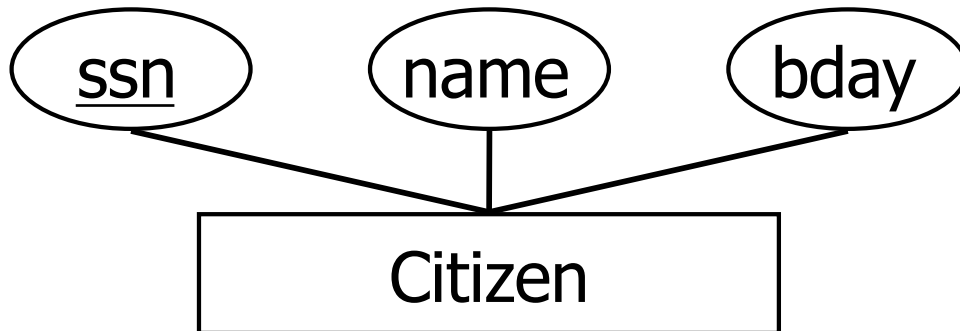
Citizen

Initiative

Poll Stn

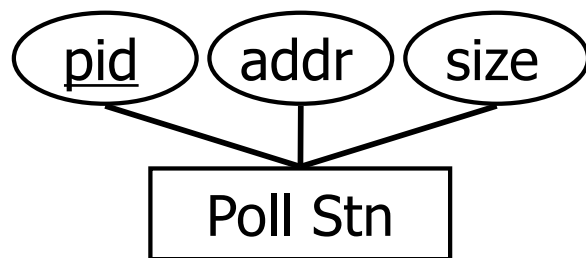
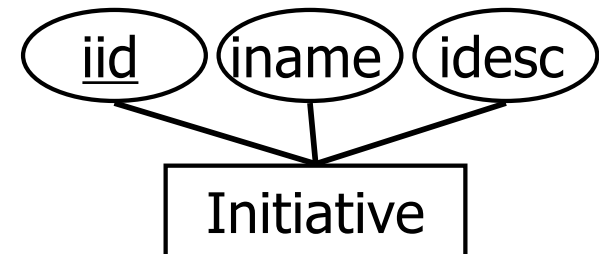
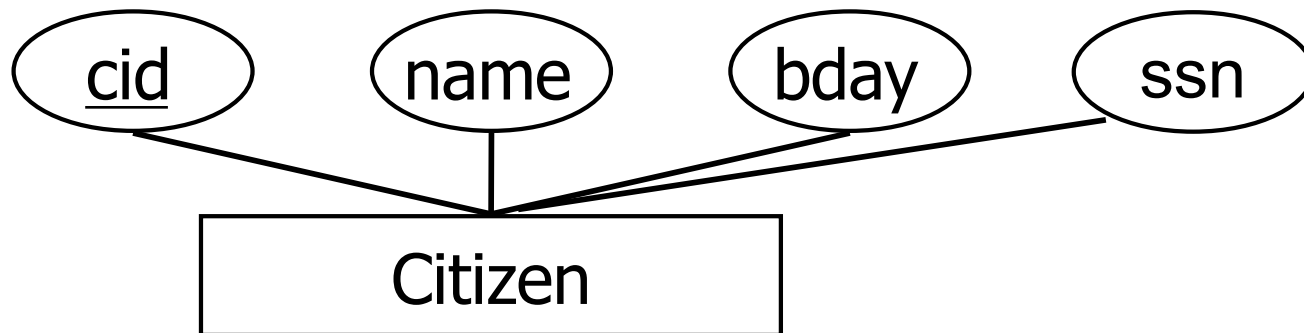


Add attributes and underline keys



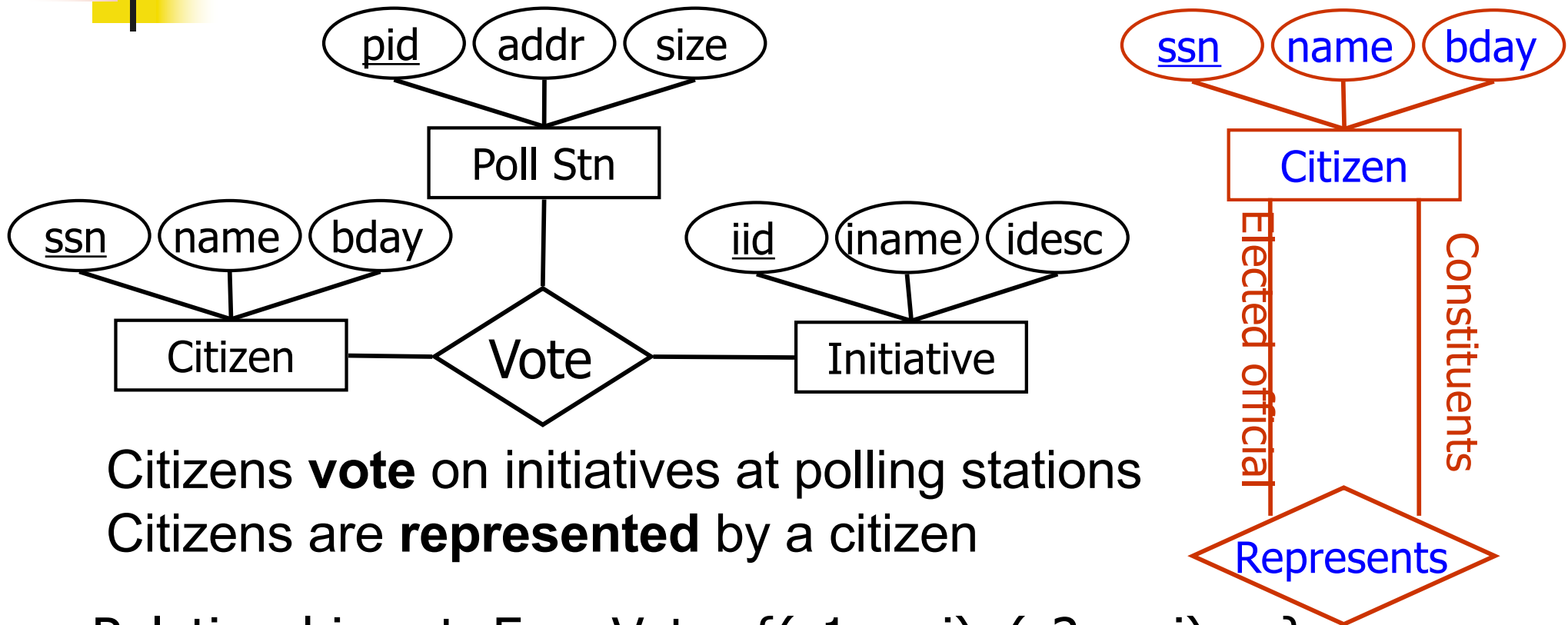


Alternate key for Citizen



Pros and cons?

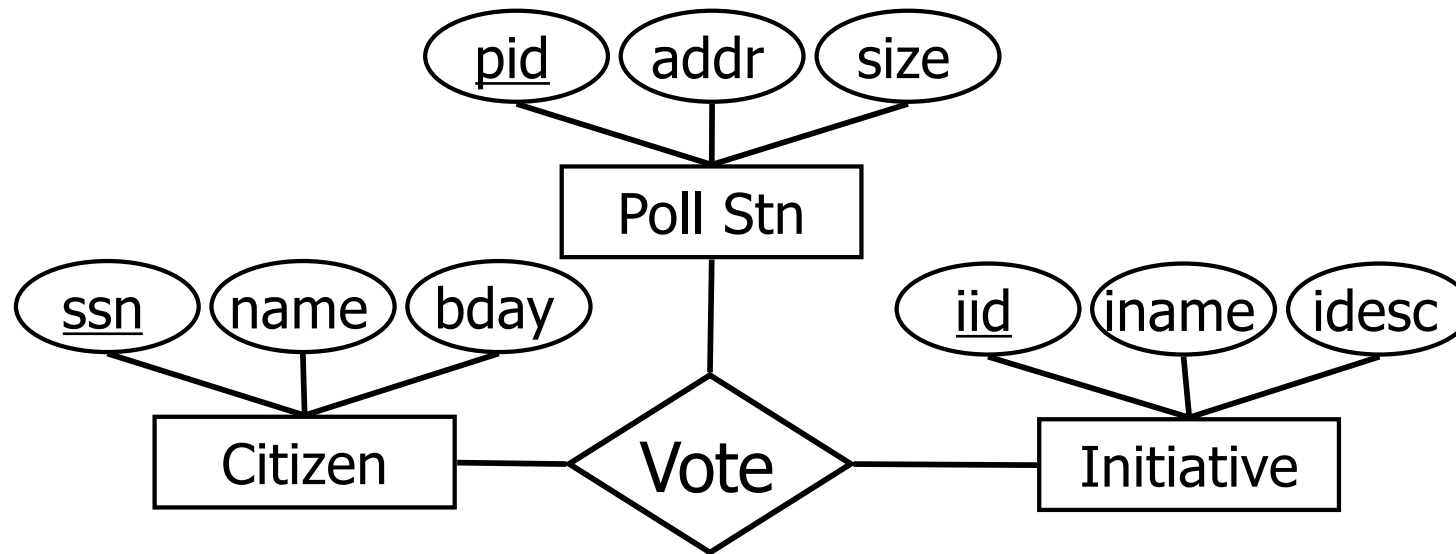
Add in the relationships



Citizens **vote** on initiatives at polling stations
 Citizens are **represented** by a citizen

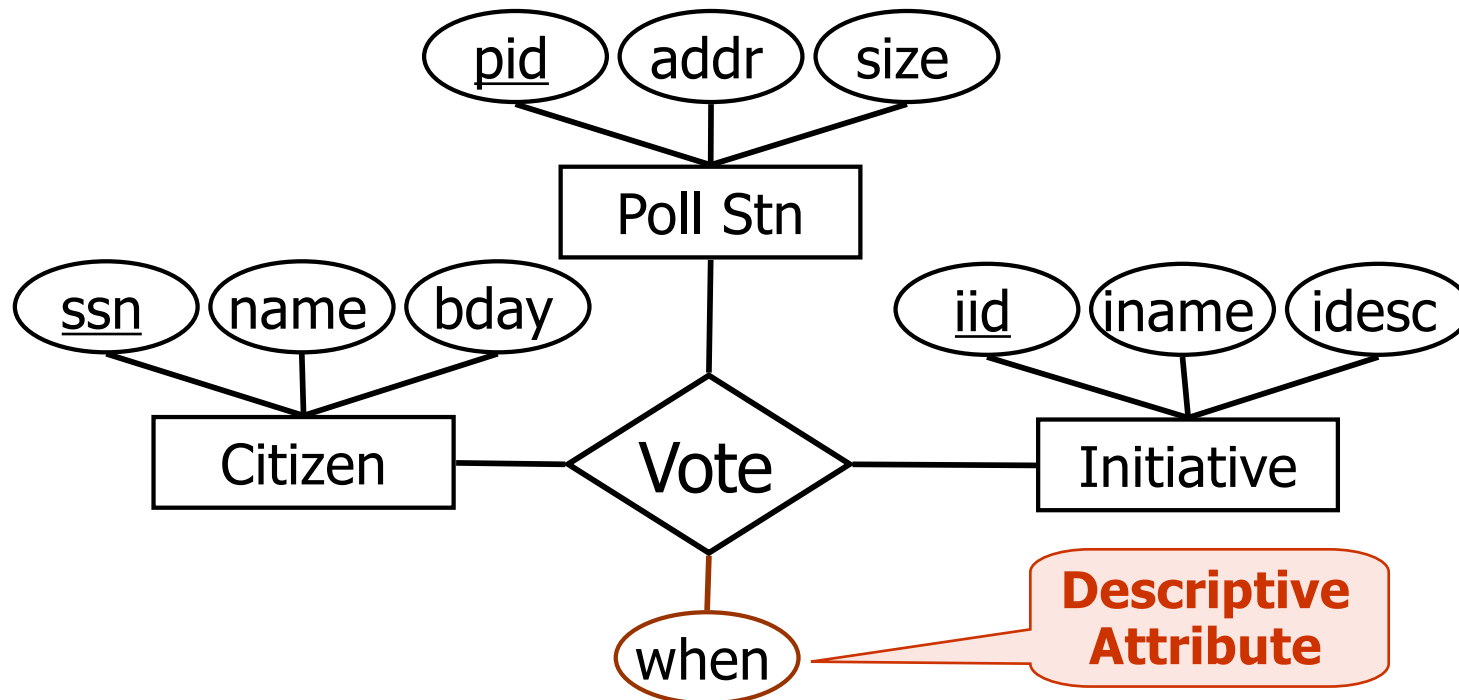
- Relationship set: E.g., Vote: $\{(c1, p, i), (c2, p, i), \dots\}$
 $\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$
- Entity sets:
 - Collection of entity instances. E.g., set of citizens.

Problem



- Suppose we want to also record *when* a Citizen voted.
- How should we represent that?

Attributes on Relationships

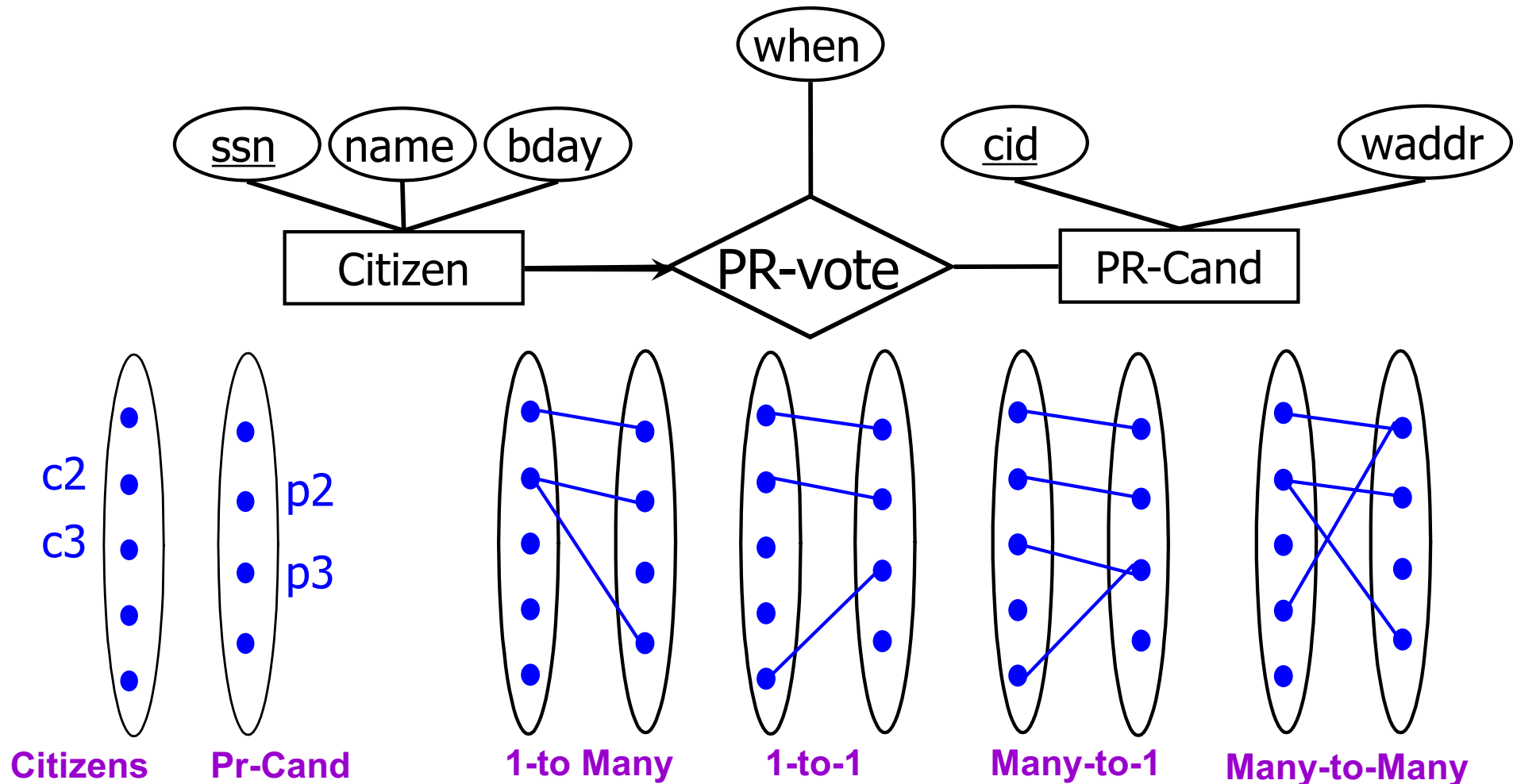


Additional requirement

- A citizen also votes for a candidate. He can vote at most once for a candidate, who is also a citizen and has a work address (waddr) on a particular date.

Key Constraints

- **Key Constraint** : Each citizen votes at most once



Relationship Type from
Choices on the right?



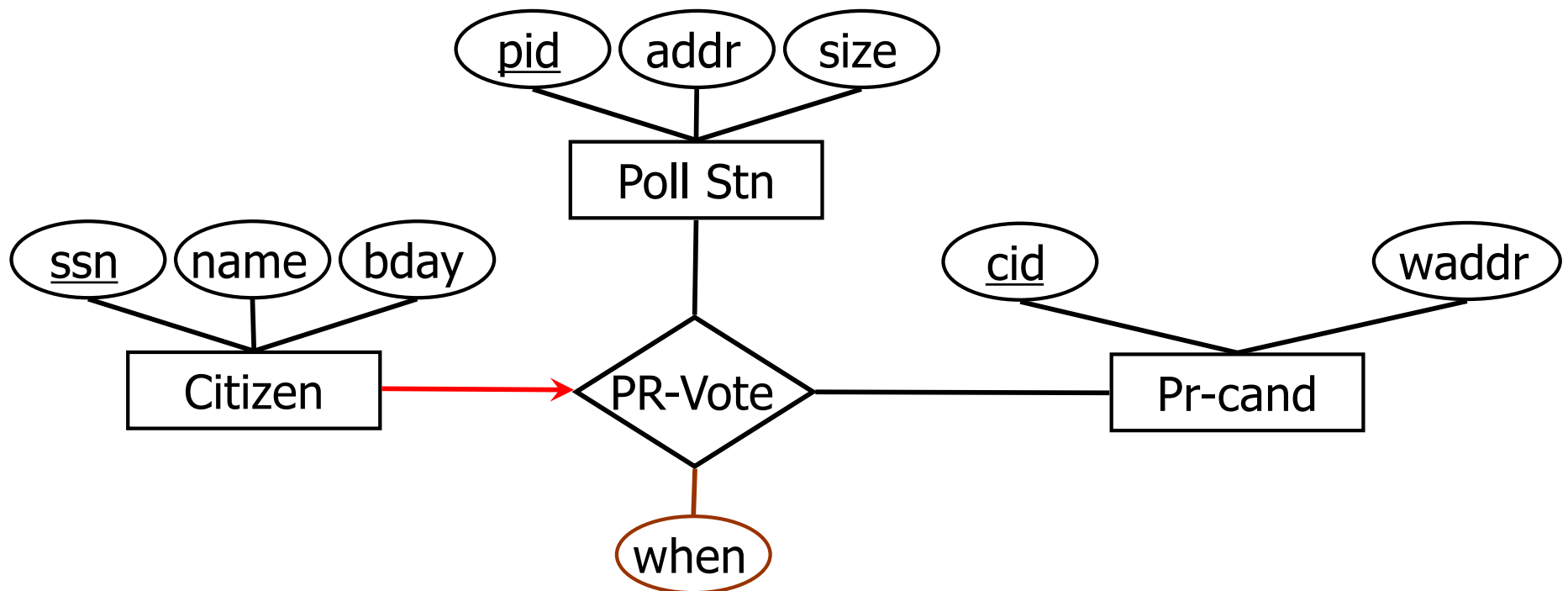
Key Constraints: Generalize

Each voter votes at most once (for one candidate)
and at a polling station.

A given citizen can appear at most once in (citizen,
polling station, candidate) sets

Key Constraints: Generalize

Each voter votes at most once (for one candidate)
and at a polling location.



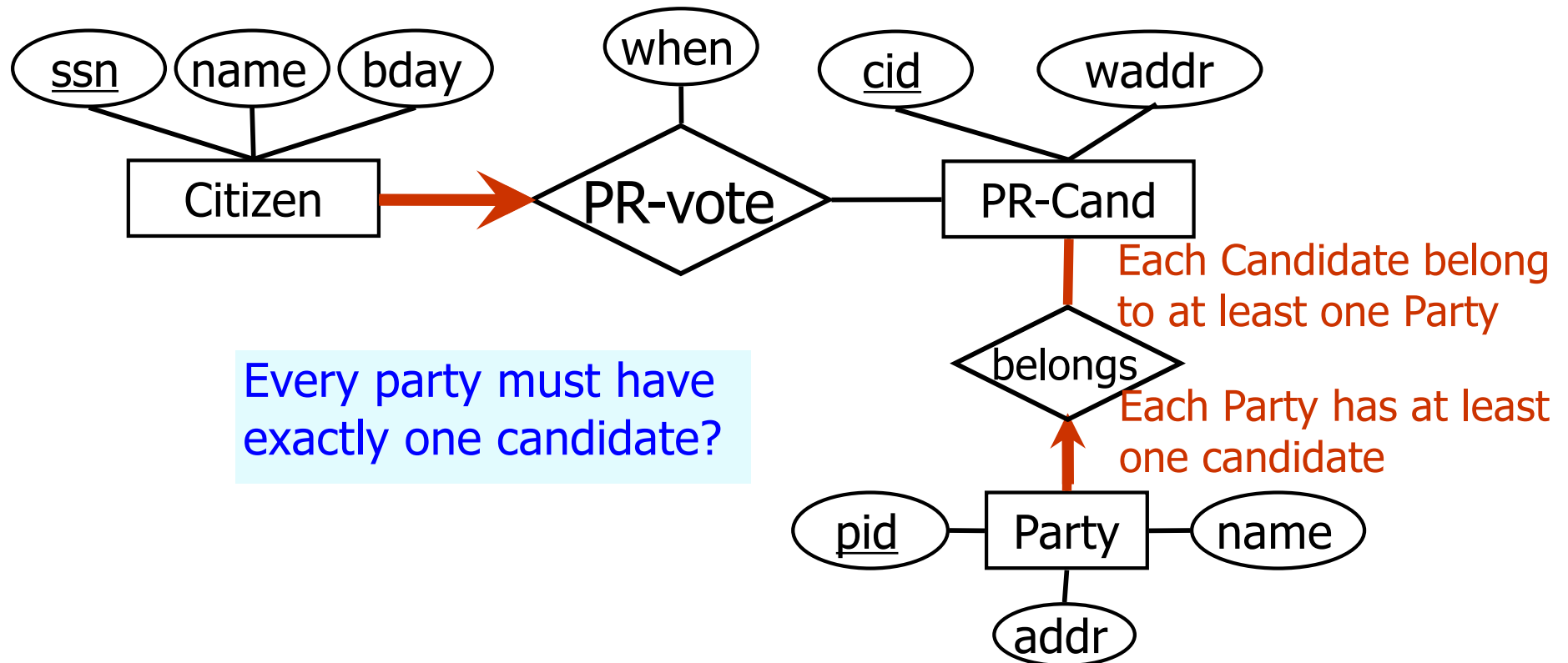
A given citizen can appear at most once in (citizen, polling station, candidate) sets

Additional requirements

- Every citizen MUST vote for a candidate
- Each candidate belongs to a party. Parties have a name and address
- Each party has exactly one candidate

Participation Constraints

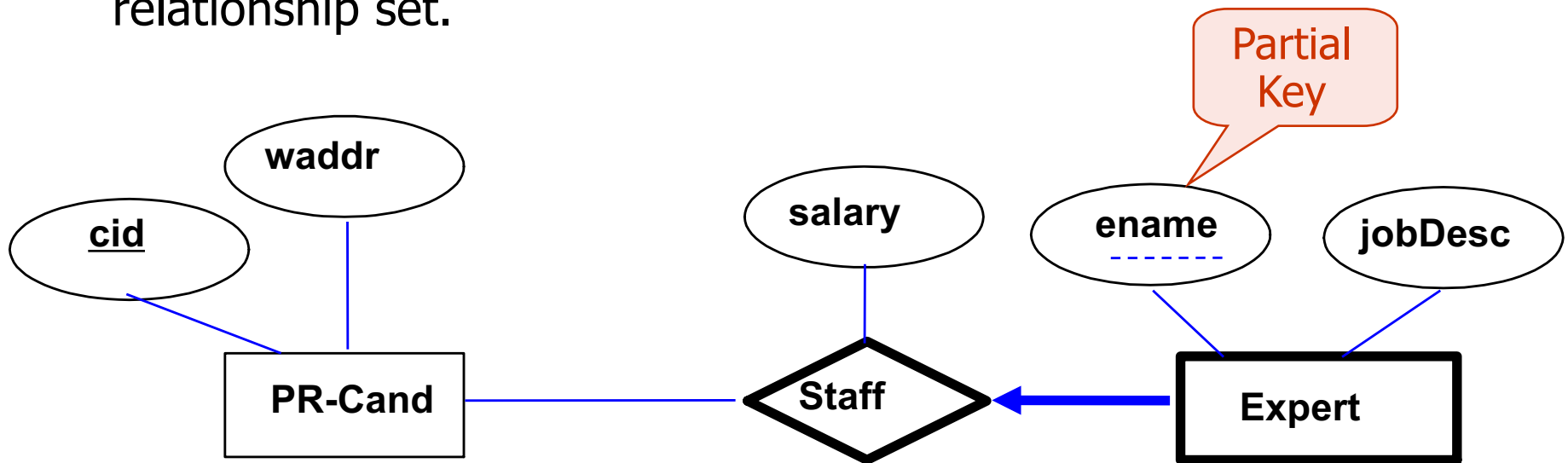
- Q: Must every citizen vote?
 - If so, this is a **participation constraint** : Every citizen must participate, i.e., **total participation** required by all citizens (vs. partial).



Weak Entities

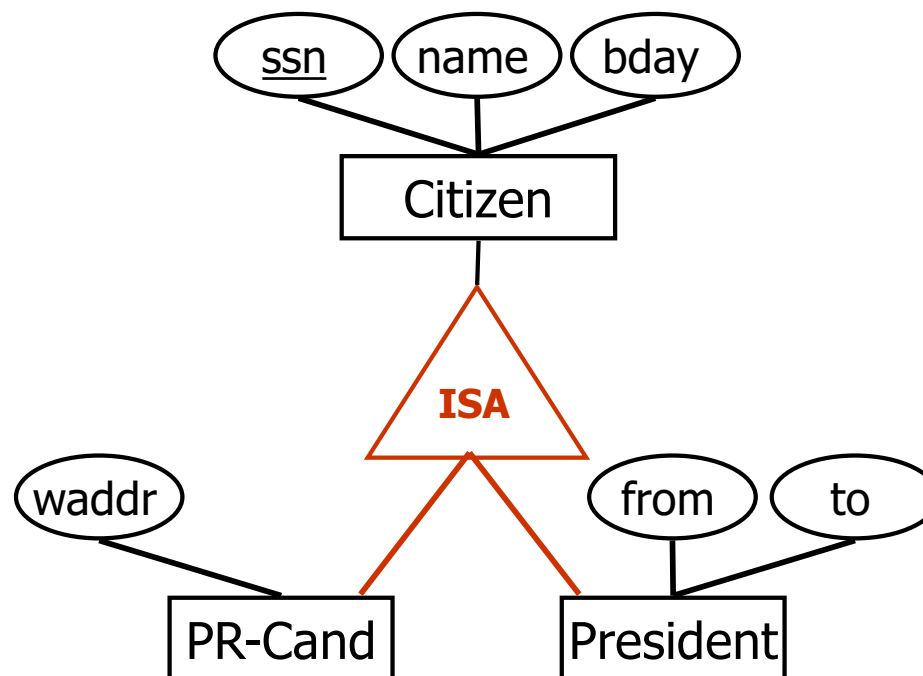
Example: Candidates have experts on their staff, identified by their name.

- Model experts as **weak entities**. Expert names are not globally unique. To identify an expert, we need candidate's ID + expert's name. Rules:
 - Weak entity has a single owner (one-to-many relationship)
 - Weak entity **must** have total participation in the above **identifying** relationship set.



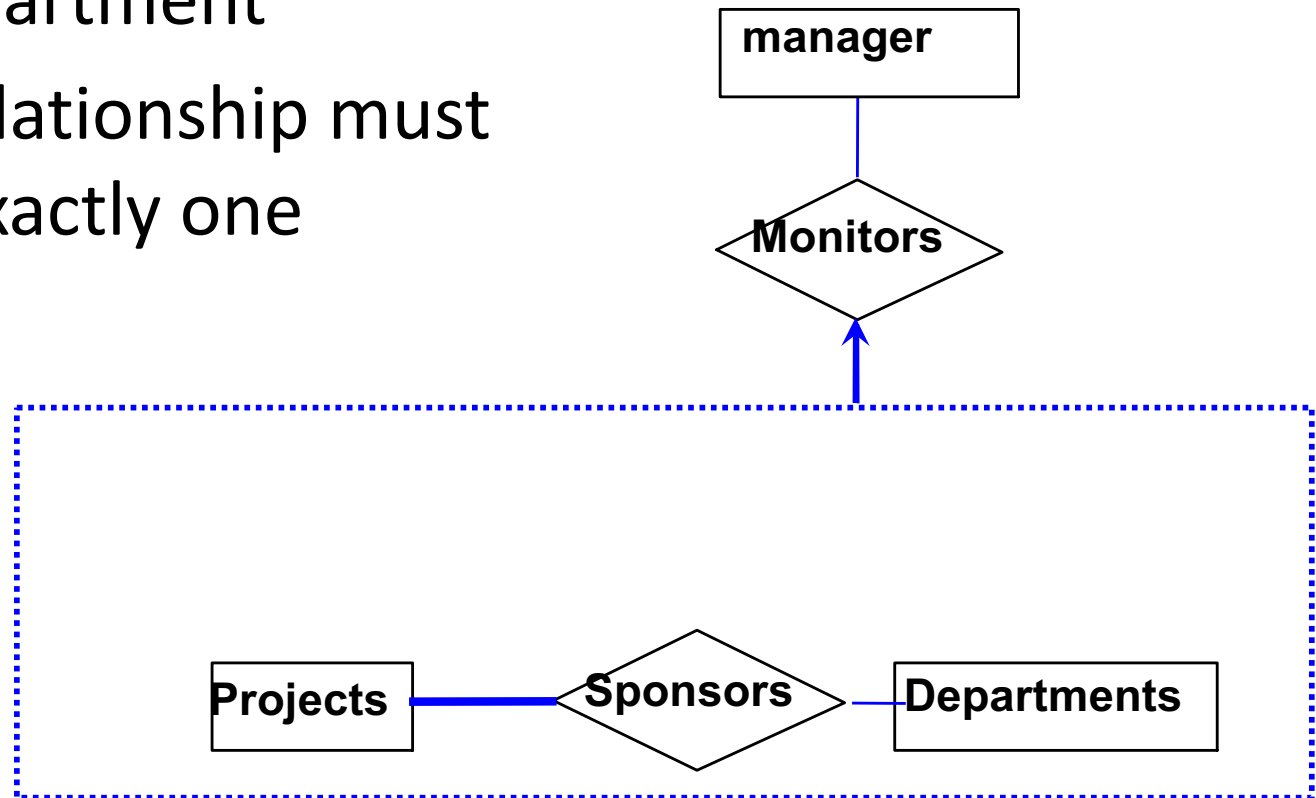
ISA ('is a') Hierarchies

- ✓ As in C++ attributes are inherited.
- ✓ If A **ISA** B, every A entity is also a B entity
- ✓ Specialize superclass (top-down design)
- ✓ Generalize subclasses (bottom-up design)
- ✓ Can be multi-level



Relationship with relationships

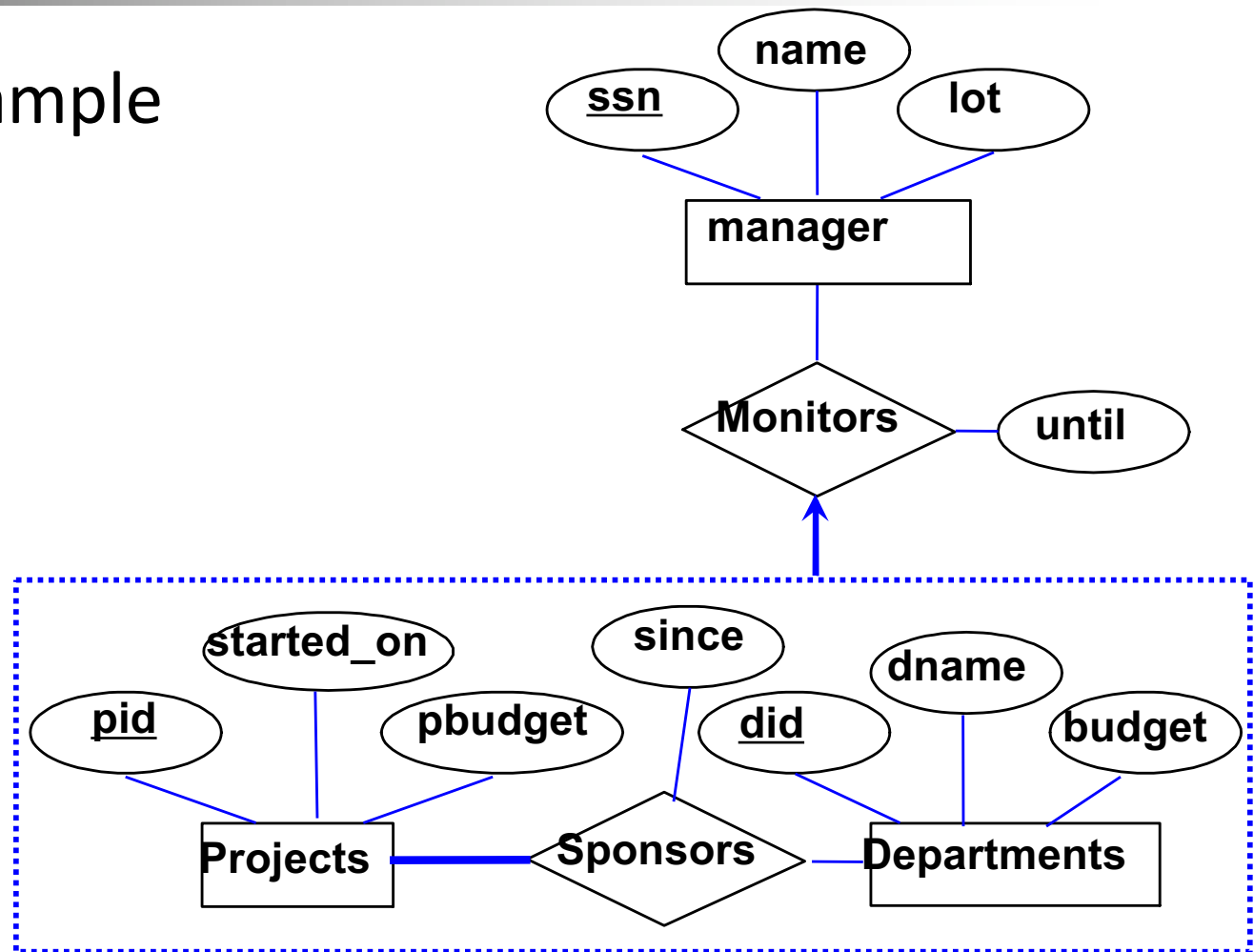
- Each Project must be sponsored by at least one Department
- Each sponsoring relationship must be monitored by exactly one manager



Treating a relationship as an entity for another relationship is called **aggregation**


Aggregation

- More complete example



Conceptual Design Using the ER Model

- Design choices:
 - Should a concept be modeled as an entity or an attribute?
 - Should a concept be modeled as an entity or a relationship?
 - Identifying relationships: Binary or ternary?
Aggregation?

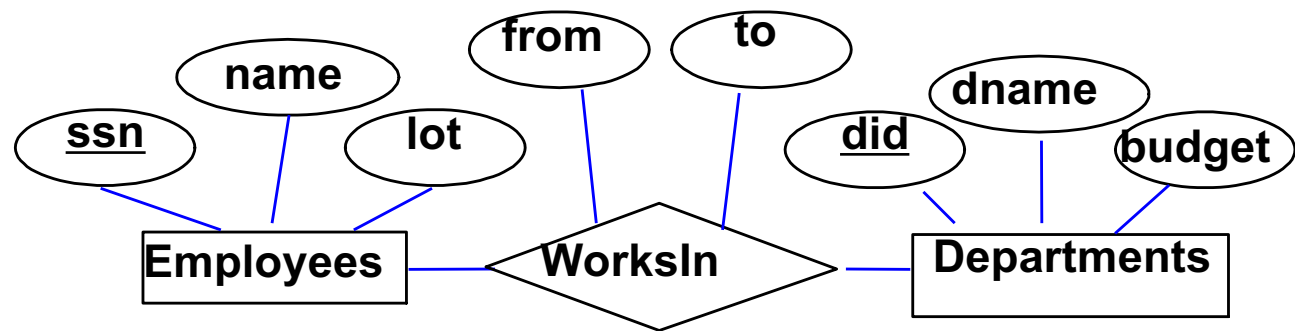


Entity vs. Attribute

- In the voting example, each party had an address
- Should address be an attribute of Party or an entity (connected to Party by a relationship)?
 - Several addresses per Party => address must be an entity
 - If the structure of address (city, street, etc.) is important, address must be modeled as an entity.

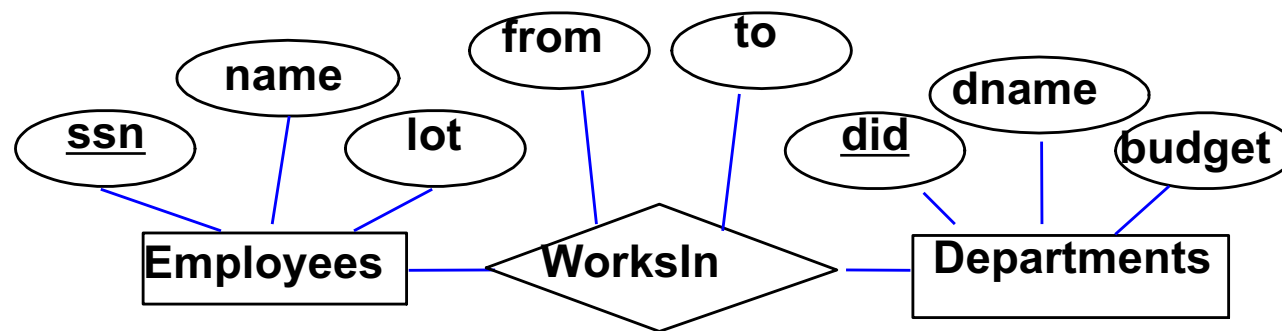
Entity vs. Attribute

- Can employee work in a given dept. for two or more periods?



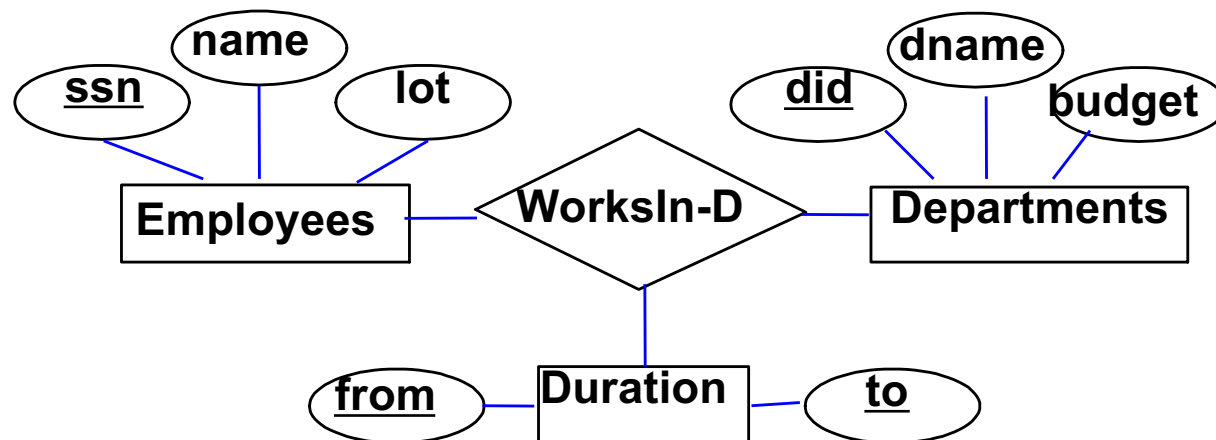
Entity vs. Attribute

- How to allow an employee to work in a department for multiple periods?



Problem is that WorksIn is defined by (Employees, Department):
Set semantics, not multi-set.

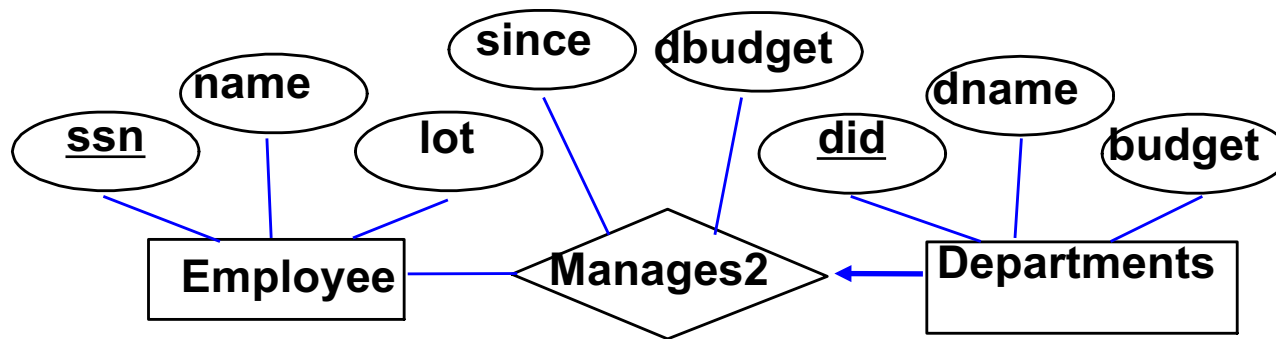
Solution



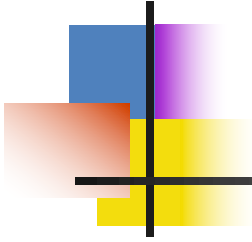
Now, the relationship is 3-way:
(Employee, Department, Duration)

Entity vs. Relationship

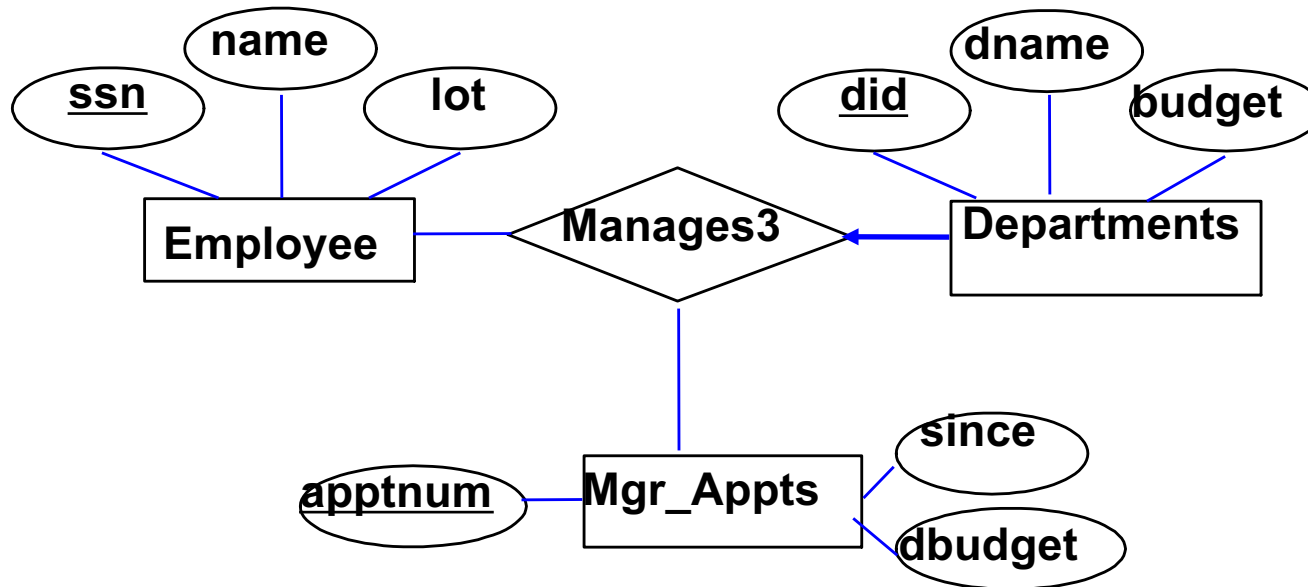
- In this, Manager (also an employee) gets a separate discretionary budget for each dept.



- What if we want Manager to get a discretionary budget that covers **all** managed depts?
 - **Redundancy** of dbudget, in Manages2.
 - **Misleading**: suggests dbudget tied to managed dept.



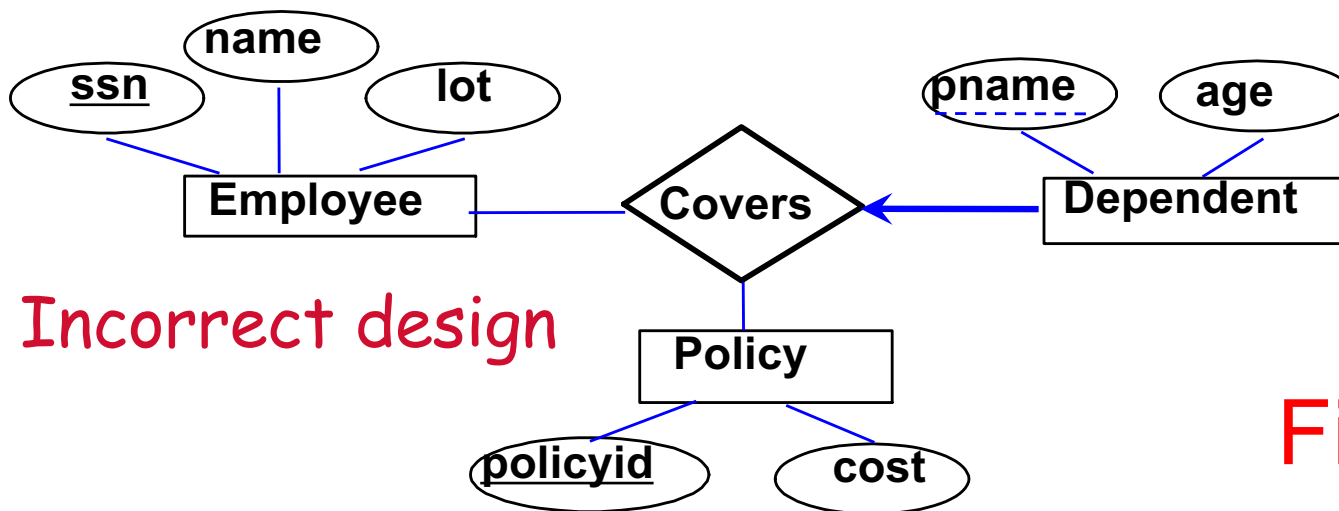
One Solution



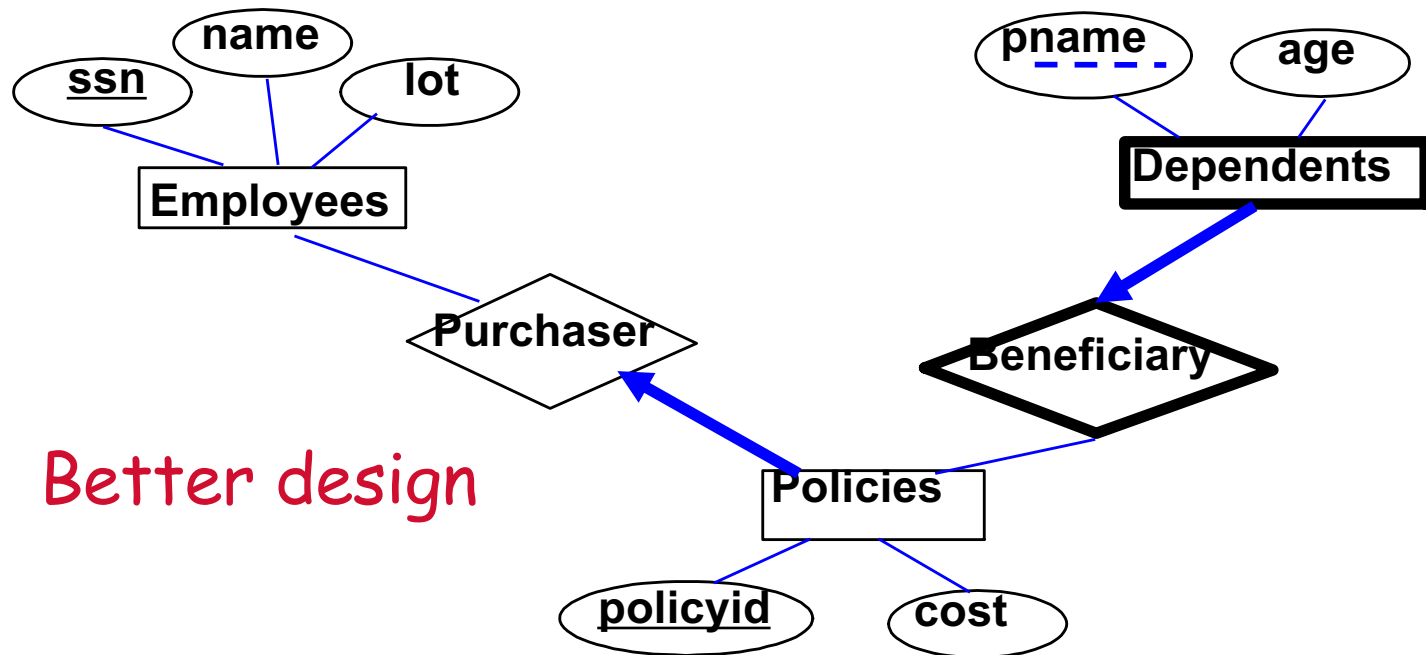
There is another way to do this. Left as an exercise.
(Think ISA hierarchy)

Incorrect Design Example

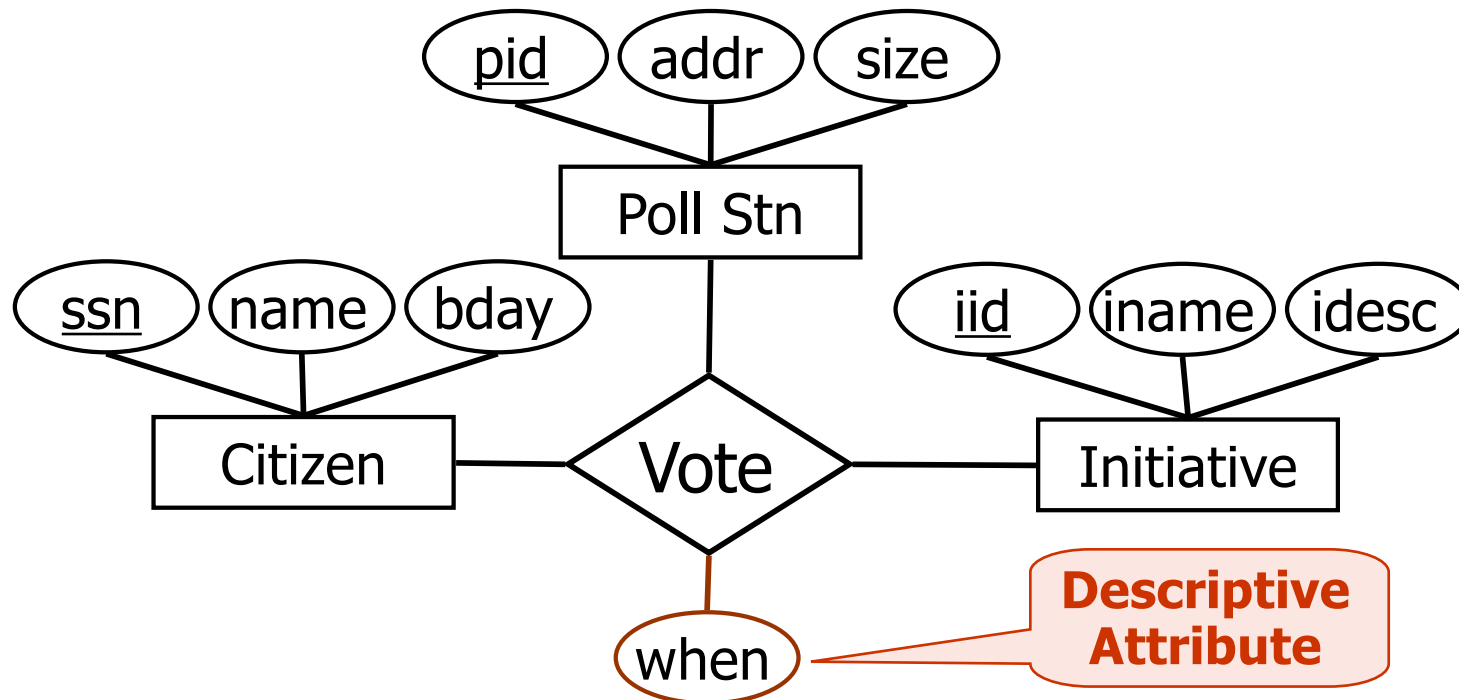
- An insurance policy cannot be owned by multiple employees
- Every policy must be owned by an employee
- Dependents is a weak entity set, identified by the conjunction of pname with policyid.



Better Design

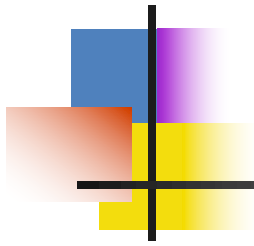


Revisiting Citizen-Initiative Example



Questions to think about

- Does the ER diagram allow a citizen to vote for multiple initiatives?
 - If so, how can the design be changed for the answer to become No?
- Does the ER diagram allow a citizen to vote for the same initiative at multiple polling stations? (i.e., cast two votes for an initiative)
 - If so, how can the design be changed for the answer to become No, while allowing vote on multiple initiatives? And what if we wanted to require those votes by a particular citizen to be on at most one polling station?



Summary of Conceptual Design

- High-level description of data to be stored
- ER model popular for conceptual design
 - Constructs are expressive and natural
 - Basic constructs: entities, relationships, and attributes (of entities and relationships).
 - Additional constructs: weak entities, ISA, aggregation.
 - Integrity constraints: key constraints and participation constraints.
- Note: There are many variations on ER model
- ER designing is subjective!