Lecture 2
The Entity-Relationship Model

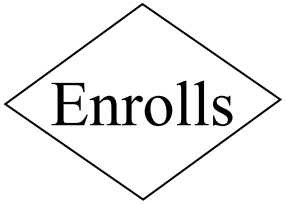
Main Phases of Database Design

- **Requirements document and analysis**: a specification of user requirements
- **Conceptual design**: *ER Model* is used at this stage, which is then translated to a relational schema
- **Schema refinement**: *(Normalization)* Check relational schema for redundancies and related anomalies
- **Physical database design and tuning**: Consider typical workloads and further refine the database design

ER Diagrams -- The Basics

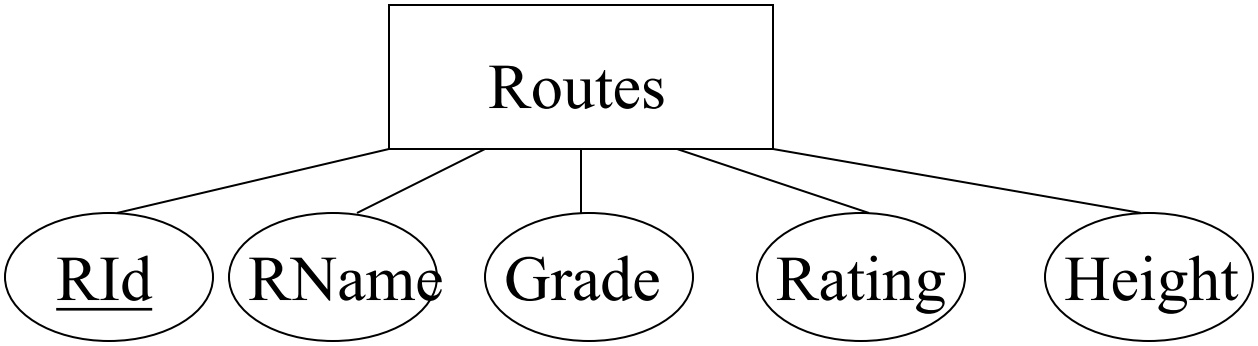
- **Entity**: an objects that exists and is distinguishable from other objects, e.g. Student, Course, Route, Climber
- **Entity set**: a set of entities of the same type that share the same properties or attributes
  - Drawn as rectangles: 

- **Attribute**: descriptive properties possessed by each member of an entity set, e.g., Name, Age, Height
  - Drawn as ovals: 

- **Relationship**: an association among several entities, e.g. a student enrolls in a course, a climber climbs a route, etc.
- **Relationship set**: a set of relationships with the same type
  - Drawn as diamonds: 

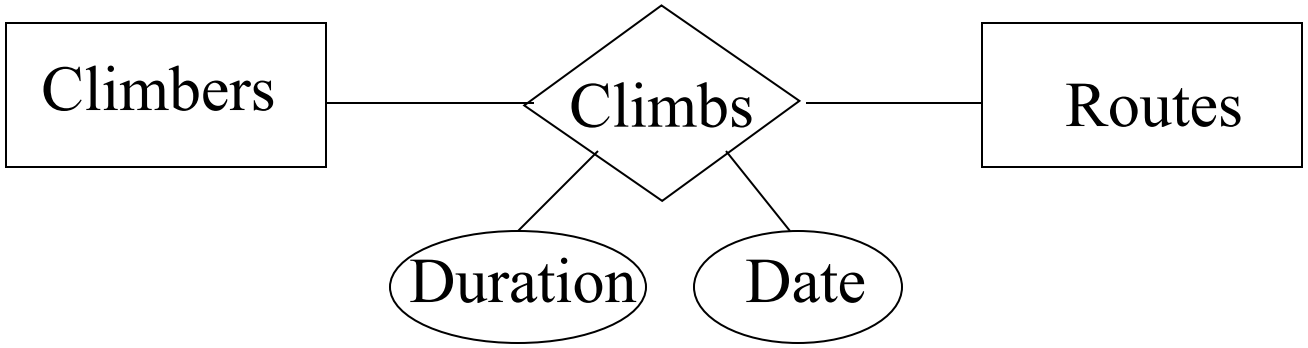
Drawing Entity Sets

- The term “entity” and “entity set” are often confused. Boxes represent *sets* of entities
- To draw an entity set we connect it with its attributes and indicate the key by underlining it:



Drawing Relationships

- Relationship sets are represented by diamonds
- They connect the entities they relate, and may additionally have attributes



What does rectangle represent in ER diagram?

- An entity set
- An entity
- A relationship
- A relationship set

Total Results: 0

What does rectangle represent in ER diagram?

An entity set

An entity

A relationship

A relationship set

What does rectangle represent in ER diagram?

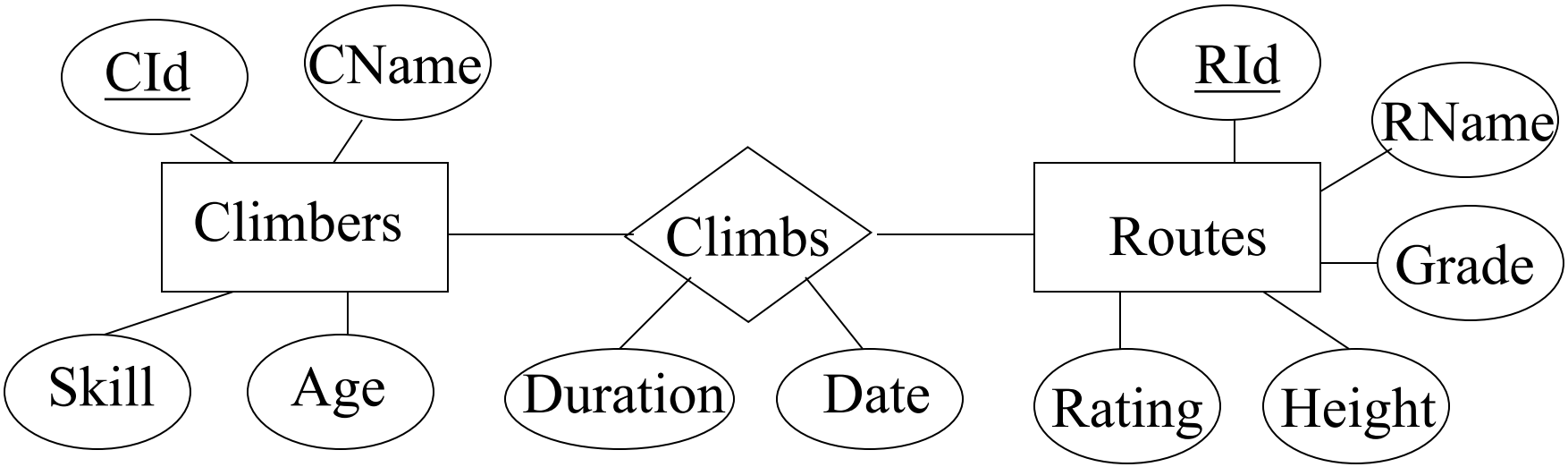
An entity set

An entity

A relationship

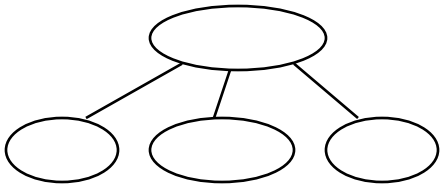
A relationship set

The Whole Diagram

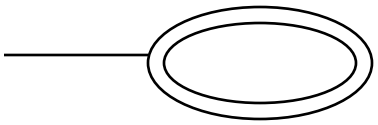


Types of Attributes

- **Composite***:* Divided into smaller subparts which represent more basic attributes with independent meanings

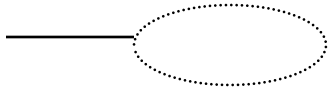


- **Atomic***:* Not divisible
- **Single-valued***:* A single value for a particular entity (e.g., age of a person)
- **Multi-valued***:* Attribute with a set of values for the same entity (e.g., degrees); A person may have 2 or more degrees



Types of Attributes

- **Stored**: DOB
- **Derived**: Value of an attribute can be derived from other attributes (e.g., age from DOB)



- **Null**: When a value is not applicable for an attribute of a particular entity (e.g., apt#) or values exists but is missing (e.g., weight of a person) or it is not known whether the attribute value exists (e.g., phone#)

Key Attribute

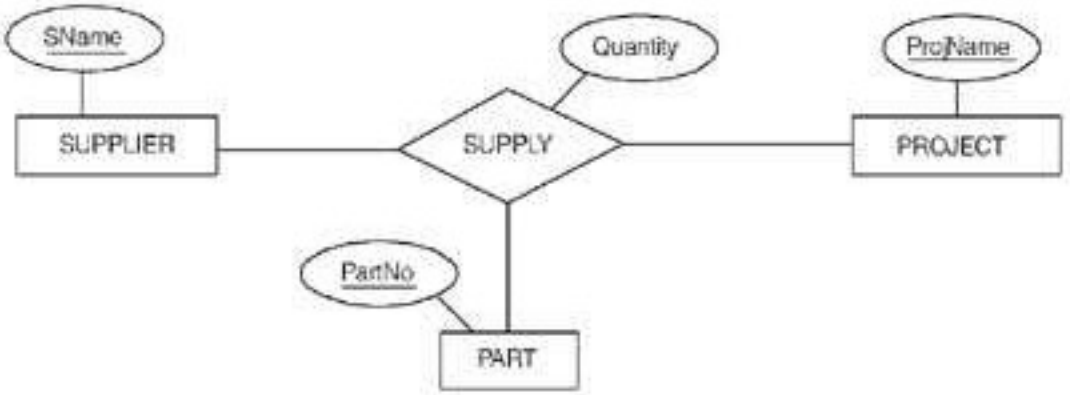
- An attribute of an entity set for which each entity must have a unique value
  - SSN of EMPLOYEE
  - StudentID of STUDENT
- A key attribute may be composite
  - VehicleTagNumber is a key of the CAR entity set with components (Number, State)

Types of Keys

- **Superkey**: one or more attributes that allow us to identify an entity in an entity set
  - Name attribute of EMPLOYEE is not a superkey
- **Candidate key**: a minimal superkey that uniquely identifies an entity
  - Suppose combination of Name and Address is sufficient to distinguish among members of the EMPLOYEE entity set
  - SSN and {Name, Address} are candidate keys
- **Primary key**: candidate key chosen by DBA to identify entities of an entity set (e.g., SSN)

Degree of Relationship

- **Degree of relationship:** # of participating entity sets in a relationship
- **Binary:** Degree two (e.g., Employee works for Department)
- **Ternary:** Degree three



- **N-ary**

Cardinality Ratio

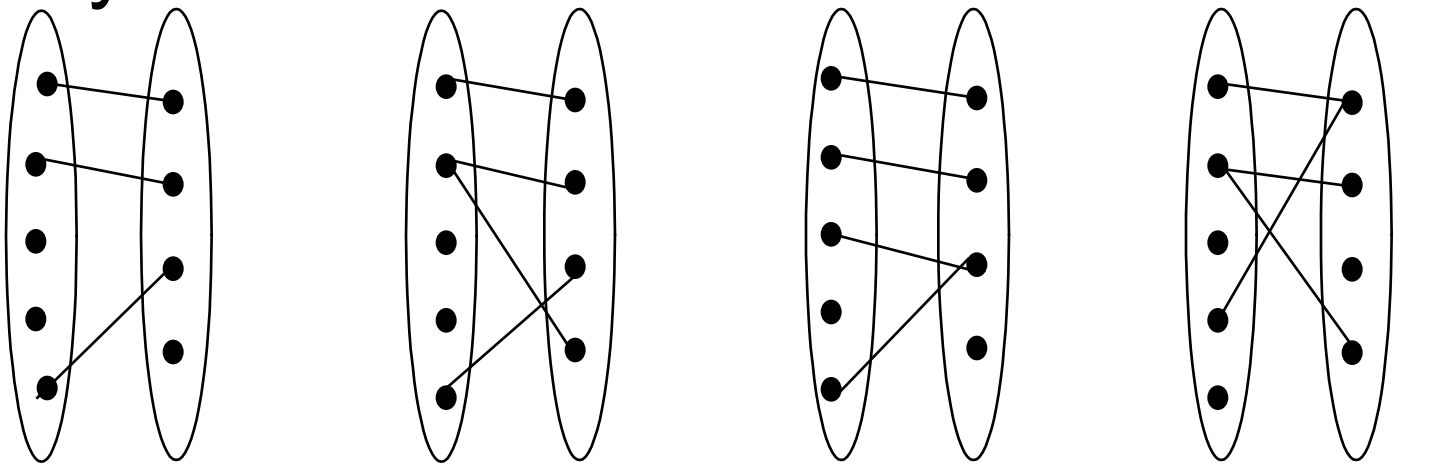
- Expresses the # of entities to which another entity can be associated via a relationship set
  - 1:1—Employee manages department
  - 1:N—Mother having children
  - N:1—Children having mothers
  - M:N—Students enrolled courses

Binary Relationships

- **1:1**: 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
- **1:N**: An entity in A is associated with any number of entities in B. An entity in B, however, can be associated with at most one entity in A
- **M:N**: An entity in A is associated with any number of entities in B and an entity in B is associated with any number of entities in A

Binary Relationships

- Relationships involving two entity sets (binary relationships) can be classified as 1-to-1, 1-to-Many (Many-to-1) or Many-to-Many



1-to-1

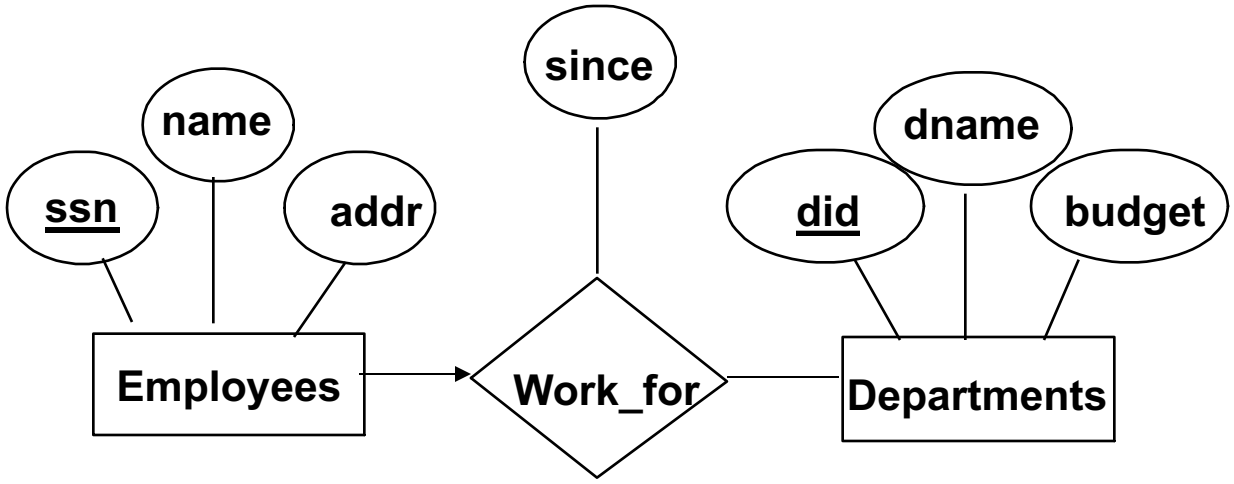
1-to Many

Many-to-1

Many-to-Many

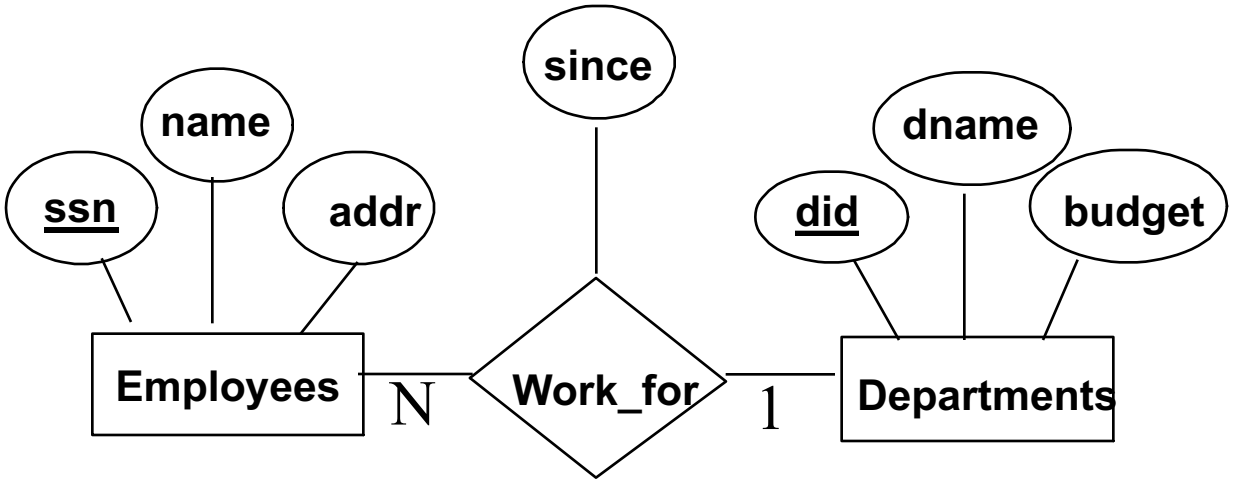
Modeling Many-to-1 Relationships

- A many-to-1 relationship is modeled by placing an arrow in the many $\rightarrow$ one direction or N:1
- An employee can work for exactly one department but a department can have any number of employees



Modeling Many-to-1 Relationships

- Explicit Notation:

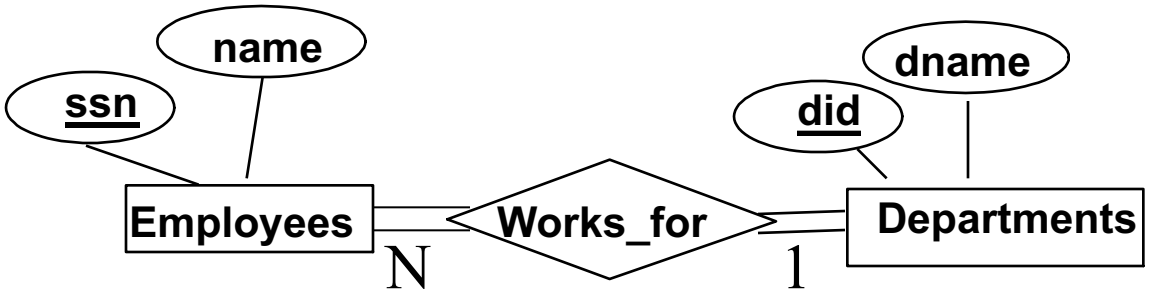


Participation

- Specifies whether the existence of an entity depends on its being related to another entity via the relationship set
- Two types
  - Total
  - Partial

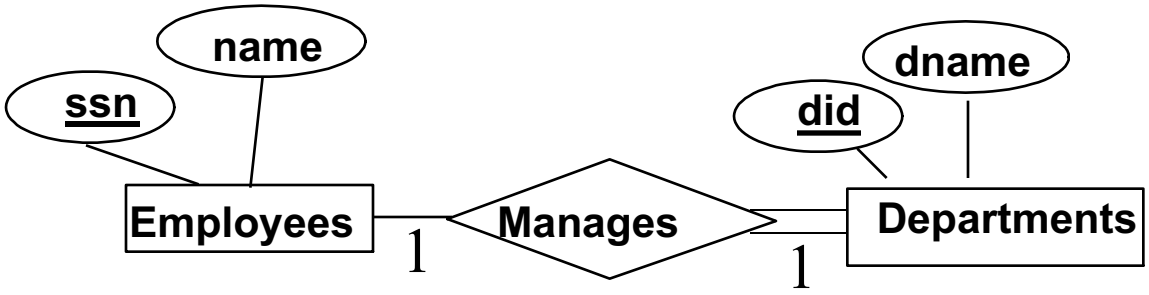
Total Participation

- Every employee must work for a department, then every employee in Employees entities must be related to a departments entity via Works_For relationship (represented by double line)



Partial Participation

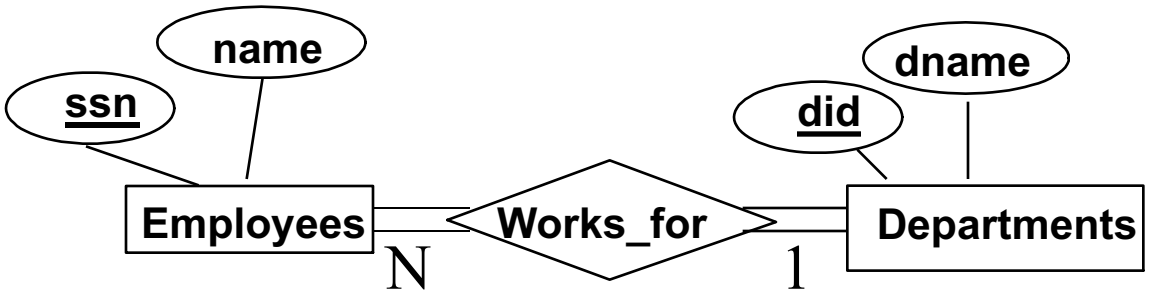
- Some employee entities are related to a department entity via Manages but not necessarily all



- **Structural constraints:** cardinality ratio and participation

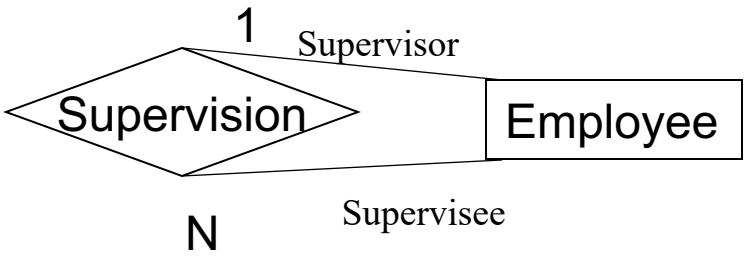
Role

- **Role:** Function that an entity plays in a relationship is called its role
- **Implicit Role:** All the participating entity sets in a relationship set are distinct
  - Works_For relationship set: Employee plays the role of employee or worker and Department plays the role of department or employer



Recursive Relationships

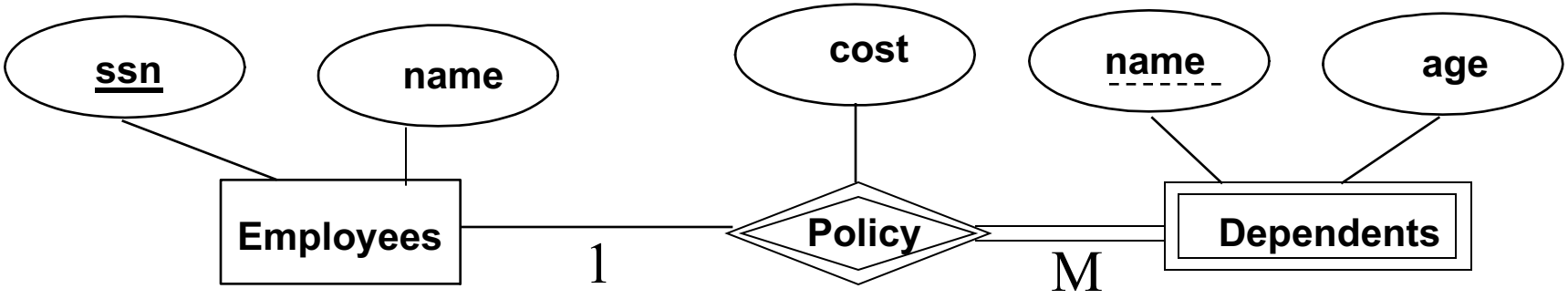
- **Explicit Role**: all the participating entity sets in a relationship are not distinct



- **Recursive relationship:** Same entity set participates more than once in a relationship in different roles
  - Supervision relationship relates an employee to a supervisor, where both employee and supervisor entities are the members of the same Employee entity set

Weak Entities

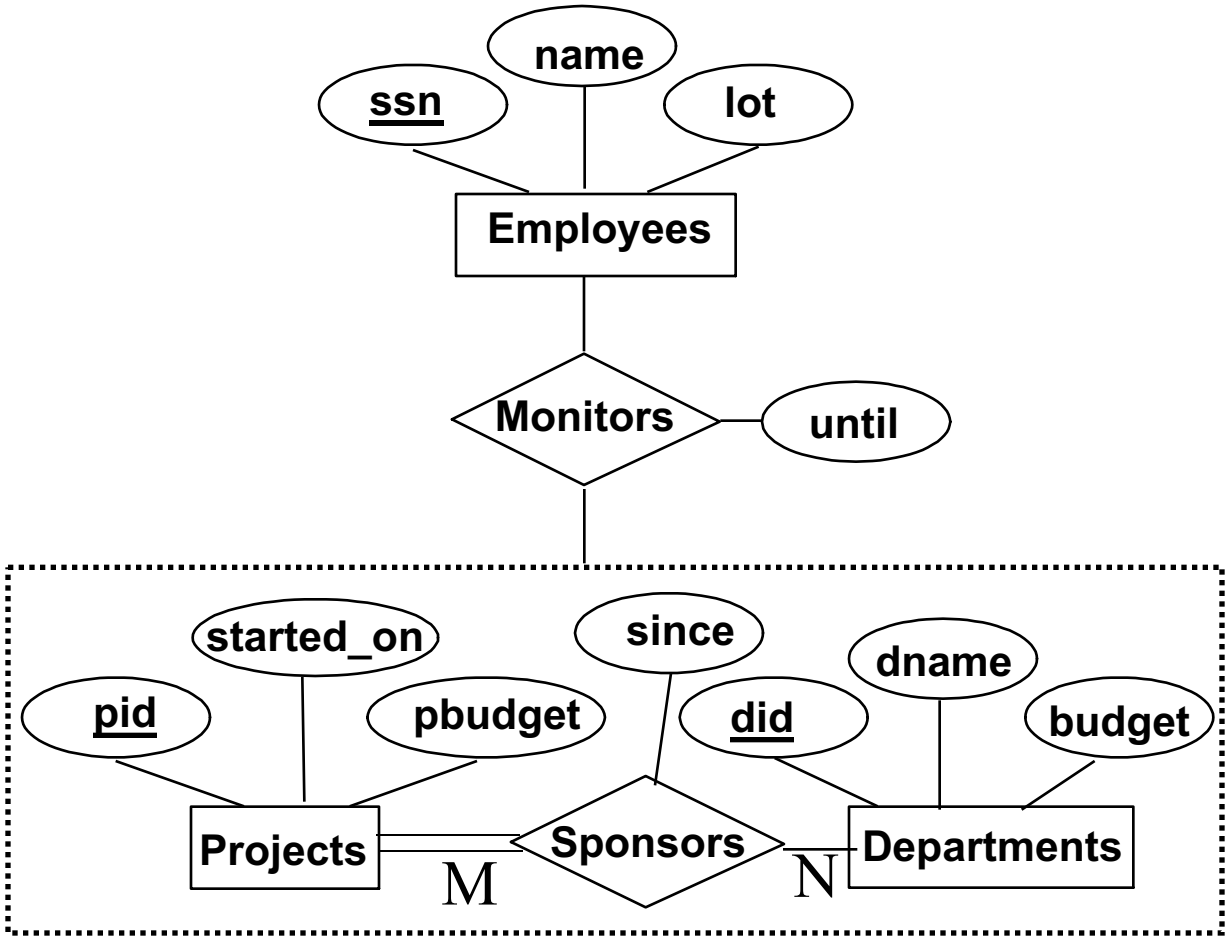
- **Weak entity set** does not have enough attributes to form a primary key
- A **weak entity** can be identified uniquely only by considering the primary key of another (*owner*) entity
  - Owner entity set and weak entity set must participate in a one-to-many relationship set (1 owner, many weak entities)
  - Weak entity set must have total participation in this identifying relationship set



Aggregation

- Used when we have to model a relationship between a collection of entities and relationships
- Allows us to treat a relationship set as an entity set for purposes of participation in (other) relationships
- Why aggregation?
  - Need to express relationship among relationships
- Why not ternary relationship?
  - Several distinct relationships
  - Each with its own attributes

Aggregation Example

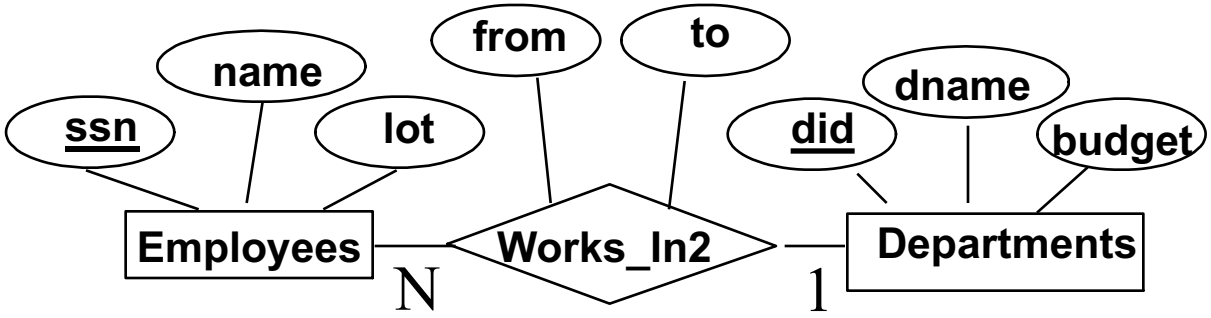


Conceptual Design Using the ER Model

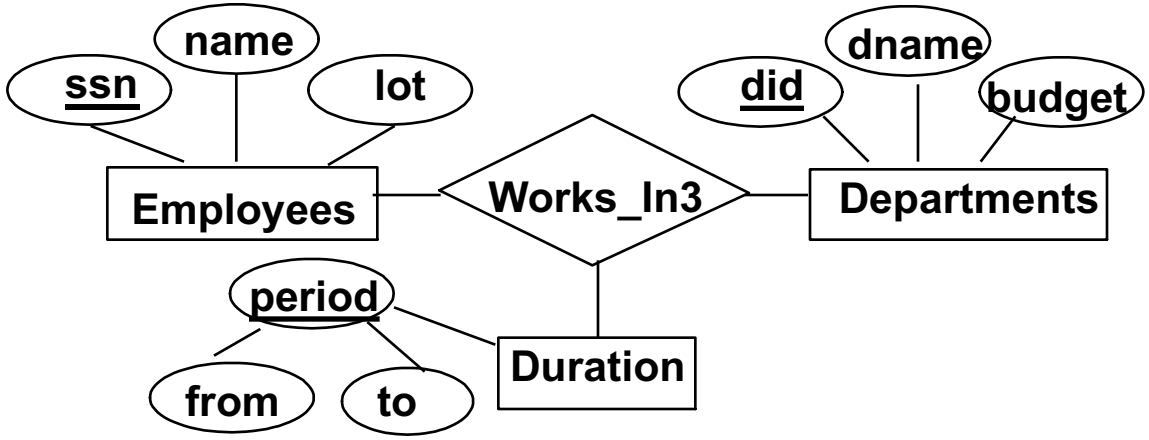
- **Design choices**:
  - Should a concept be modeled as an entity or an attribute?
  - Should a concept be modeled as an entity or a relationship?
  - Identifying relationships: Binary or ternary? Aggregation?

Entity vs. Attribute

- Works_In2 does not allow an employee to work in a department for two or more periods

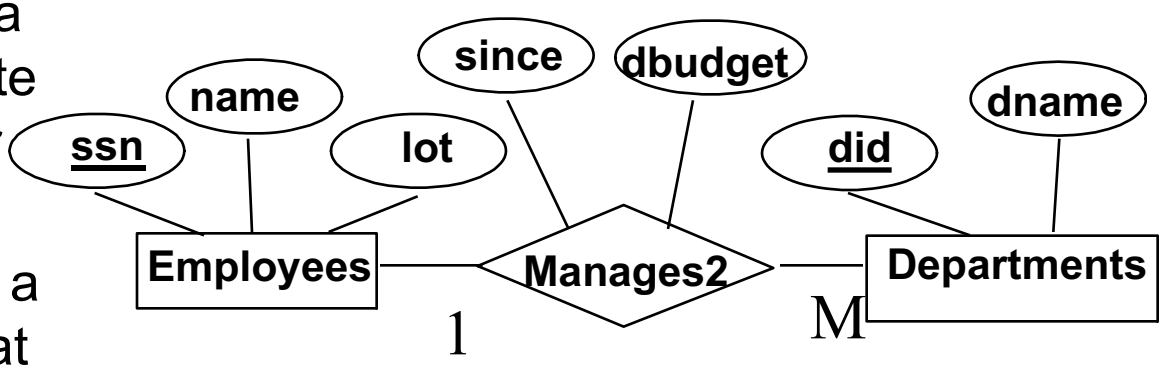


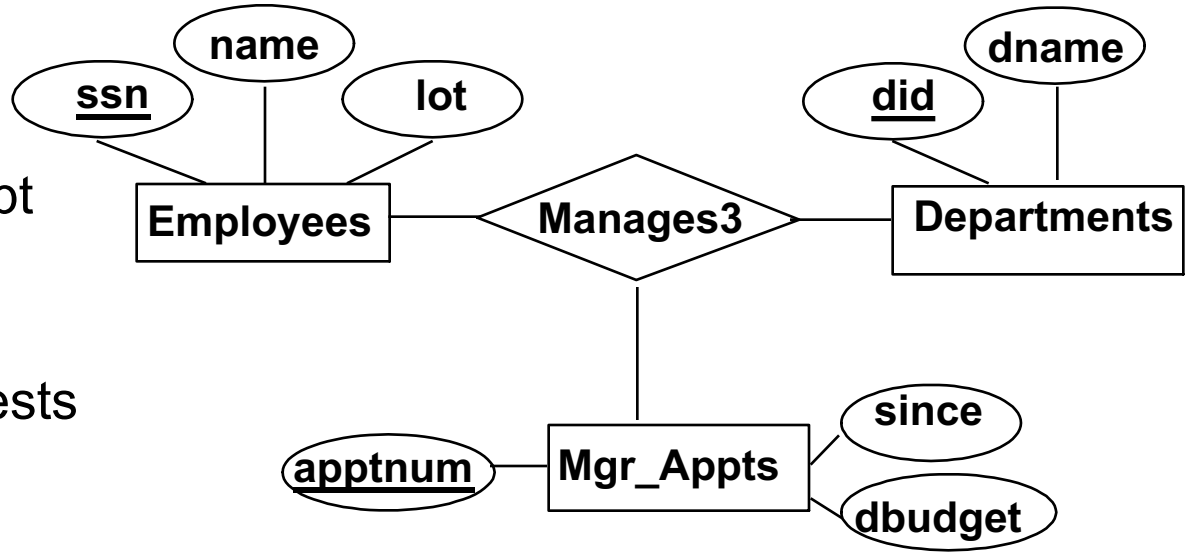
- We want to record several periods for each employee



Entity vs. Relationship

- First ER diagram OK if a manager gets a separate discretionary budget for each dept.
- What if a manager gets a discretionary budget that covers *all* managed depts?
  - **Redundancy** of *dbudget,* which is stored for each dept managed by the manager
  - **Misleading**: suggests *dbudget* tied to managed dept.



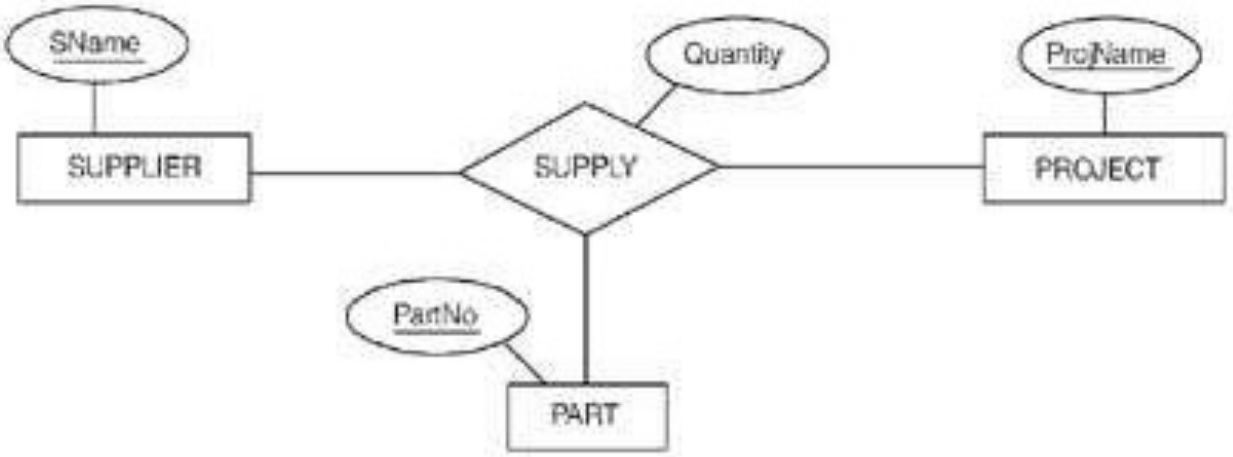


Binary vs. Ternary Relationships

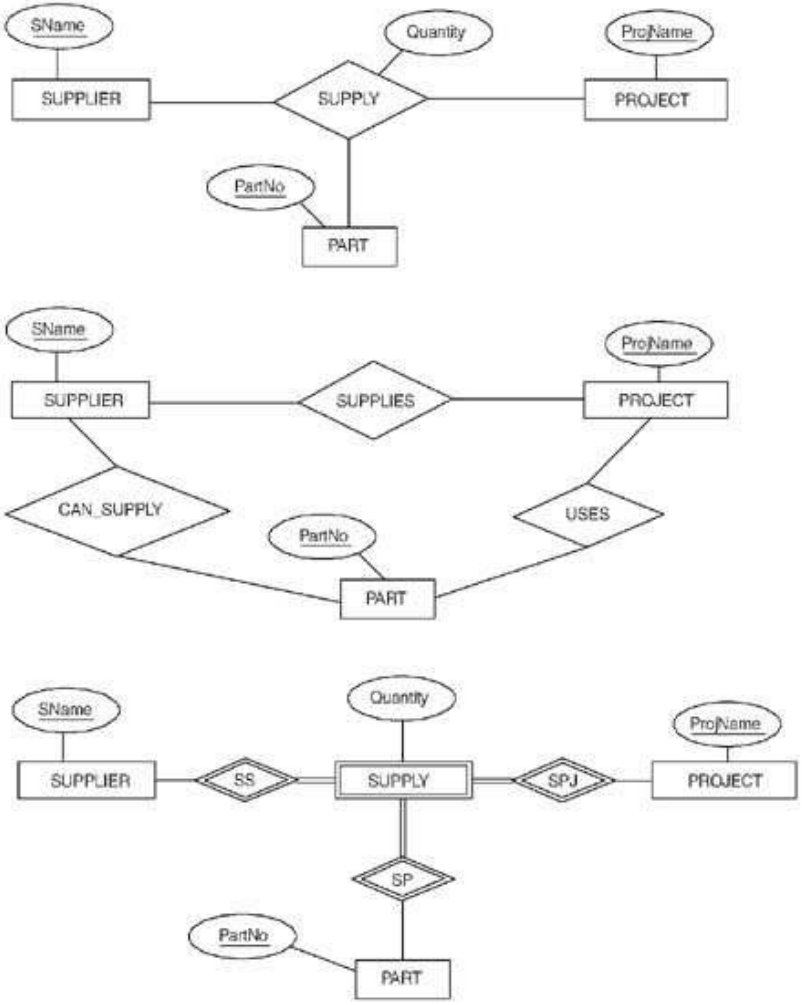
- Ternary relationship represents more information than do 3 binary relationships
- In general, an n-ary relationship is **not** equivalent to n binary relationships

Binary vs. Ternary Relationships

- A ternary relation **Supply** relates entity sets **Part**, **Project** and **Supplier**, and has descriptive attribute Quantity. No combination of binary relationships is an adequate substitute



Binary vs. Ternary Relationships



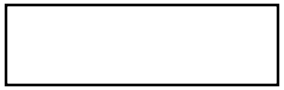
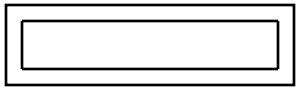
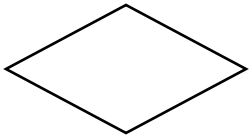
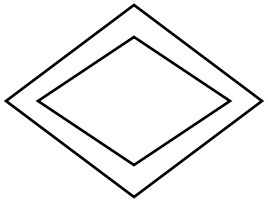
Summary of Conceptual Design

- *Conceptual design* follows *requirements analysis*
  - Yields a high-level description of data to be stored
- ER model popular for conceptual design
  - Constructs are expressive, close to the way people think about their applications
- Basic constructs: *entities*, *relationships*, and *attributes* (of entities and relationships)
- Some additional constructs: *weak entities*, and *aggregation*
- There are many variations on ER model

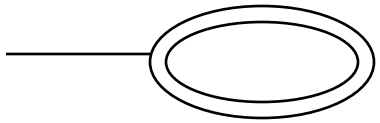
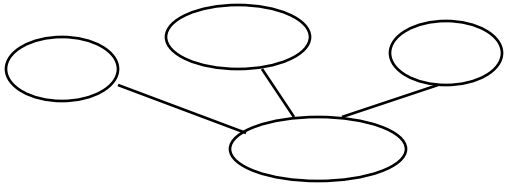
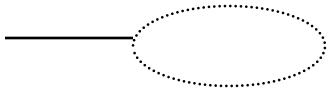
Summary of ER

- ER design is *subjective*. There are often many ways to model a given scenario! Analyzing alternatives can be tricky, especially for a large enterprise. Common choices include:
  - Entity vs. attribute, entity vs. relationship, binary or n-ary relationship, and whether or not to use aggregation
- Ensuring good database design: resulting relational schema should be analyzed and refined further. (FD information and normalization techniques)

Summary of ER (Contd.)

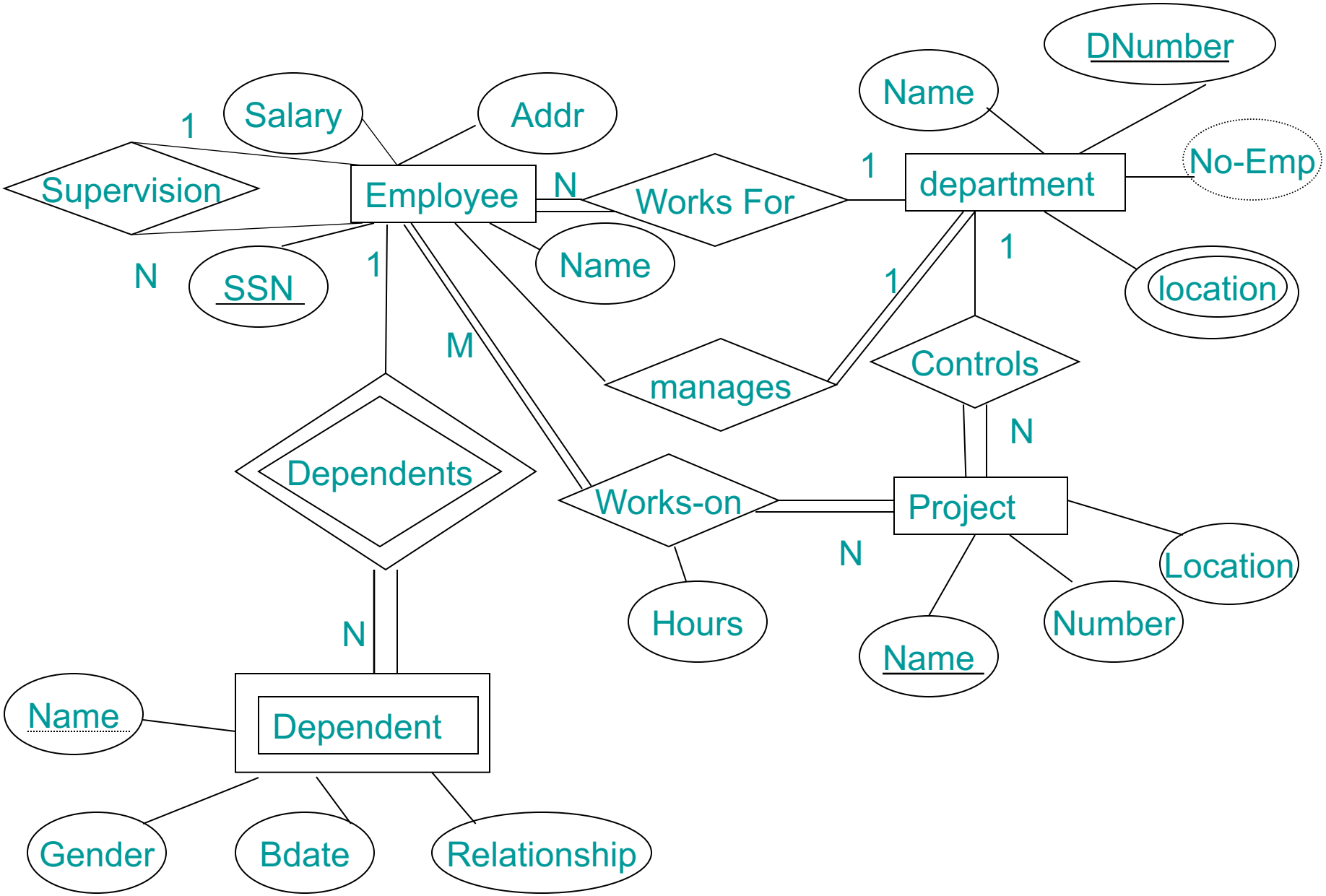
Symbol	Meaning
	Entity Set
	Weak Entity Set
	Relationship Set
	Identifying Relationship Set
	Attribute
	Key Attribute

Summary of ER (Contd.)

Symbol	Meaning
	Multi-valued attribute
	Composite Attribute
	Derived Attribute

Summary of ER (Contd.)

Symbol	Meaning
	Total Participation of E2 in R
	Cardinality Ratio 1:N For E1:E2 in R


DNumber
Name
Salary
Addr
1
Supervision
Employee
N
Works For
1
department
No-Emp
N
SSN
1
Name
1
1
location
M
manages
Controls
N
Dependents
Works-on
Project
N
Location
Hours
Number
Name
N
Name
Dependent
Gender
Bdate
Relationship

Exercise 1

- Employee: salary, address, SSN, name
- Department: name, dnumber, location (multiple)
- Each employee works for one department
- One department has many employees
- One employee might manage one department
- One department is managed by one employee
- One employee might have many dependents
- Dependents: name, age
- A dependent can’t exist without an employee entity

Exercise 2

- Student: SSN, name
- Instructor: SSN, name
- Class: course#, dept, enrollment, time, place
- One student takes many classes
- One class is taken by many students
- Record the grade for each student and class
- One instructor teaches many classes
- One class is taught by only one instructor