ENTITY RELAIONAL MODEL

Mirunalini.P

SSNCE

March 29, 2022

Table of Contents

Session Objective

- Understand ER model concepts
- Understand ER diagram notations

Session Outcome

At the end of this session, participants will be able to

- To Understand ER model concepts
- To Understand ER diagram notations
- Able to draw ER diagram for any schema

ER Model

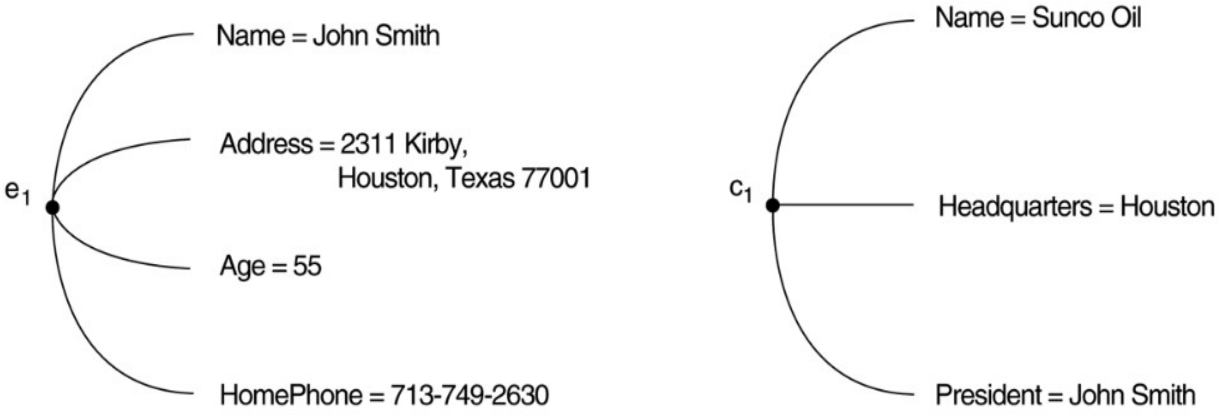
- **Entity-Relational (ER)** model is a popular high level conceptual data model and is used for **conceptual design** of database applications
- E-R facilitate **database design** by representing overall logical structure of a database.
- E-R data model employs three basic notions :
  - **Entity sets**
  - **Relationship sets**
  - **Attributes**
- The E-R model is usually expressed as an **E-R diagram**

Entities and Attributes

- **Entity:** An entity is a thing or an object in the miniworld that are represented in databases.
- Eg: Employee, Department.
- **Attributes:** Properties used to describe the entity.
- Eg: EMPLOYEE Entity has four attributes Name, Address, Age, Phone.
- **Value:** Each entity has a value for each of its attributes.
- Eg: Name='JONES', Address='Harrington Road',Age=40,Phone=01-44-3470727

Entity and Attributes

Two entities Employee and Company and their attributes



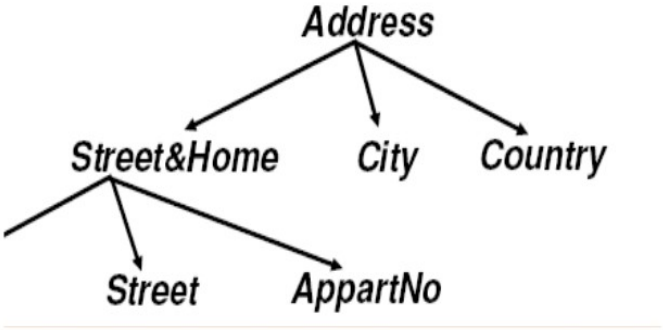
Types of Attributes

Attributes can be

- Composite / Simple (atomic)
- Single-valued / Multivalued
- Stored / Derived
- Key / Nonkey

Types of Attributes

- **Simple:** Attribute that has a **single atomic value or not divisible**.
- Example: SSN or age.
- **Composite:** The attribute that can be **divided into smaller subparts**.
- Example: Address (Apt_No, House_No, Street, City, State, ZipCode, Country) .



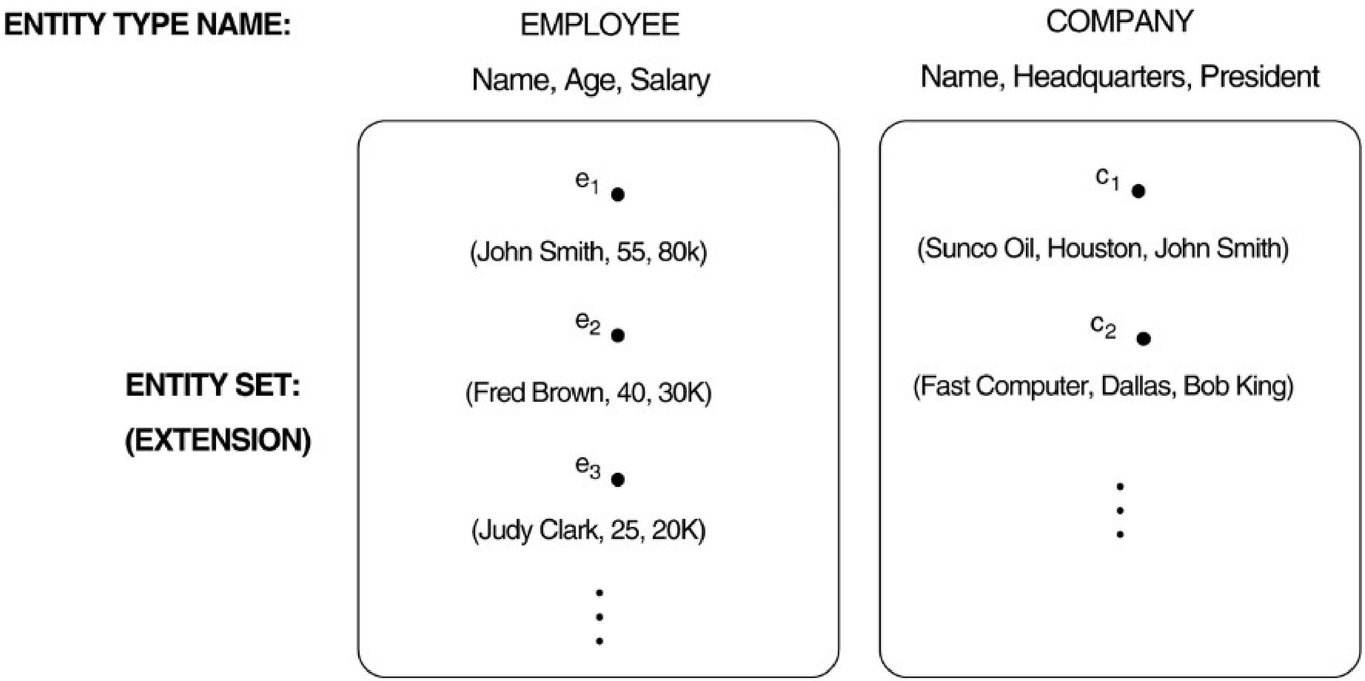
Types of Attributes

- **Single Valued:** Attributes with **single value** for a particular entity
- Ex: Age of a student is a single valued attribute.
- **Multi-valued :**An entity may have **multiple values** for an attribute.It may have upper and lower bounds on the number of values.
- Example: Color of a CAR Denoted as Color
- **Stored vs Derived attribute :** The values for this type of attribute are **derived from the values of other related attributes or entities**.
- Age can be determined from the current date and the value of person's BirthDate .
- Hence the **age attribute is derived attribute** and **BirthDate is called stored attribute**

Entity Types and Entity Set

- **Entity Type**
  - It is a **collection of entities** that share common properties or characteristics.
  - Each entity type is described by name and its attributes.
  - An entity type is represented as a **rectangular box enclosing the entity type name**
  - Example: the EMPLOYEE entity type or the PROJECT entity type.
- **Entity Set:**
  - The **collection of all entities of a particular entity type in the DB at any point** in time
  - Entity set uses the same name as the entity type.

Entity Types and Entity Set



ENTITY TYPE NAME:

EMPLOYEE

Name, Age, Salary

COMPANY

Name, Headquarters, President

ENTITY SET:

(EXTENSION)

e_1

(John Smith, 55, 80k)

e_2

(Fred Brown, 40, 30K)

e_3

(Judy Clark, 25, 20K)

c_1

(Sunco Oil, Houston, John Smith)

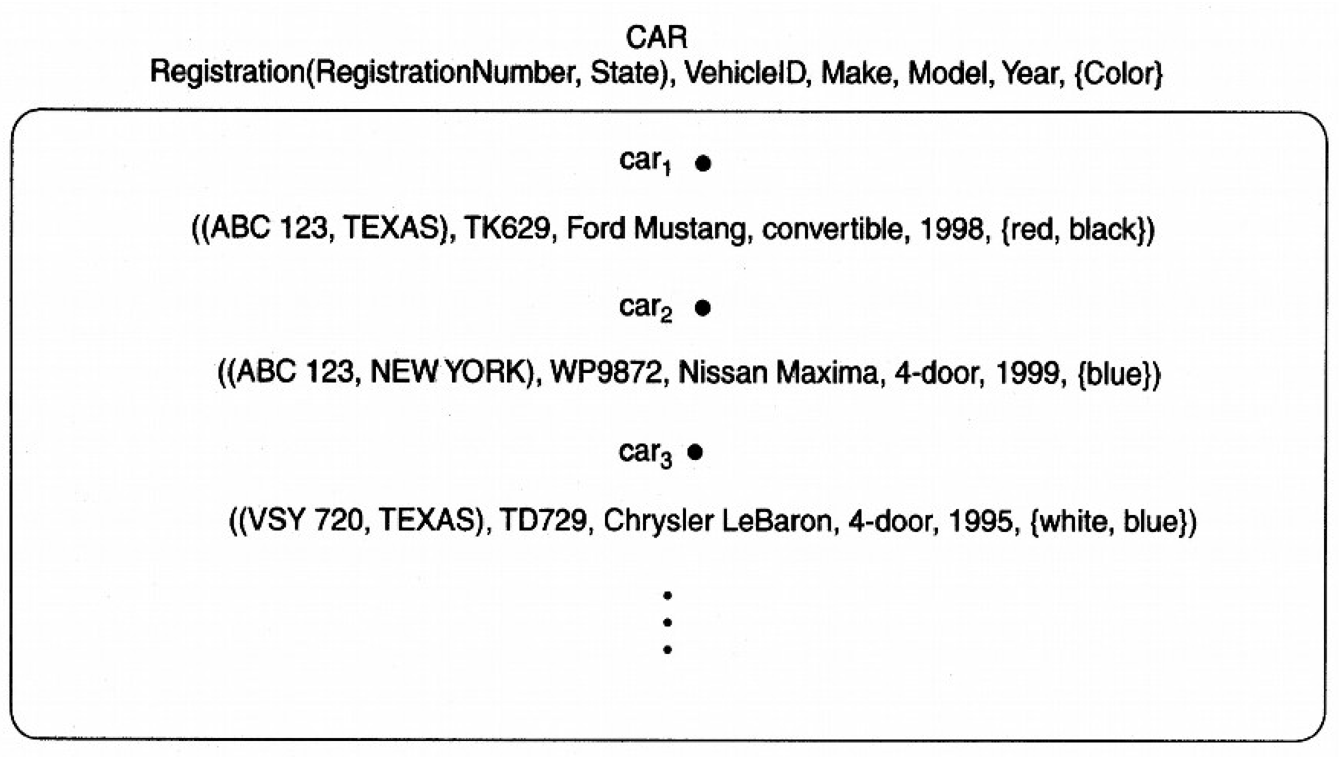
c_2

(Fast Computer, Dallas, Bob King)

Keys

- An attribute whose values that **uniquely identifies** individual instances of an entity type is key attribute
- Eg: SSN of EMPLOYEE.
- A key attribute may be composite.
- Eg: VehicleTagNumber is a key of the CAR entity type with components (Number, State).
- The Key attribute is **underlined inside the oval** in ER Diagram.
- A weak entity type is one **without any key attributes** of its own.

Entity Types and Entity Set



CAR

Registration(RegistrationNumber, State), VehicleID, Make, Model, Year, {Color}

car_1 ●

((ABC 123, TEXAS), TK629, Ford Mustang, convertible, 1998, {red, black})

car_2 ●

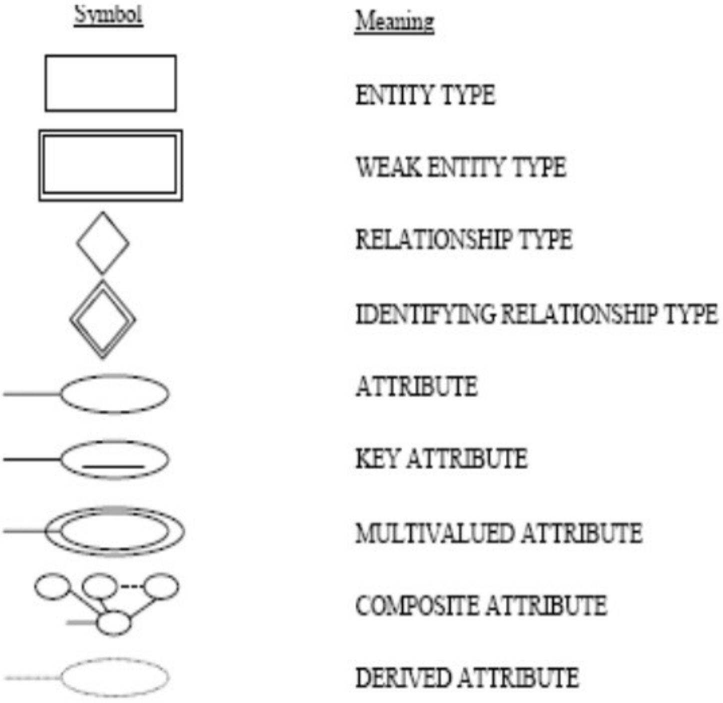
((ABC 123, NEW YORK), WP9872, Nissan Maxima, 4-door, 1999, {blue})

car_3 ●

((VSY 720, TEXAS), TD729, Chrysler LeBaron, 4-door, 1995, {white, blue})

⋮

ER Diagram Notation



Symbol

Meaning

ENTITY TYPE

WEAK ENTITY TYPE

RELATIONSHIP TYPE

IDENTIFYING RELATIONSHIP TYPE

ATTRIBUTE

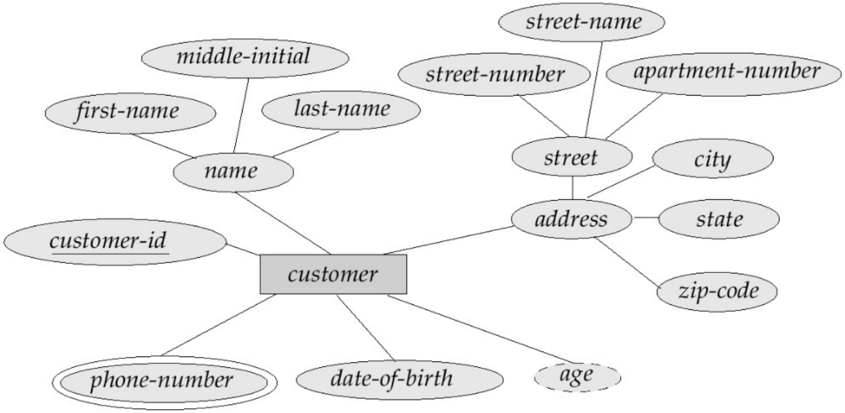
KEY ATTRIBUTE

MULTIVALUED ATTRIBUTE

COMPOSITE ATTRIBUTE

DERIVED ATTRIBUTE

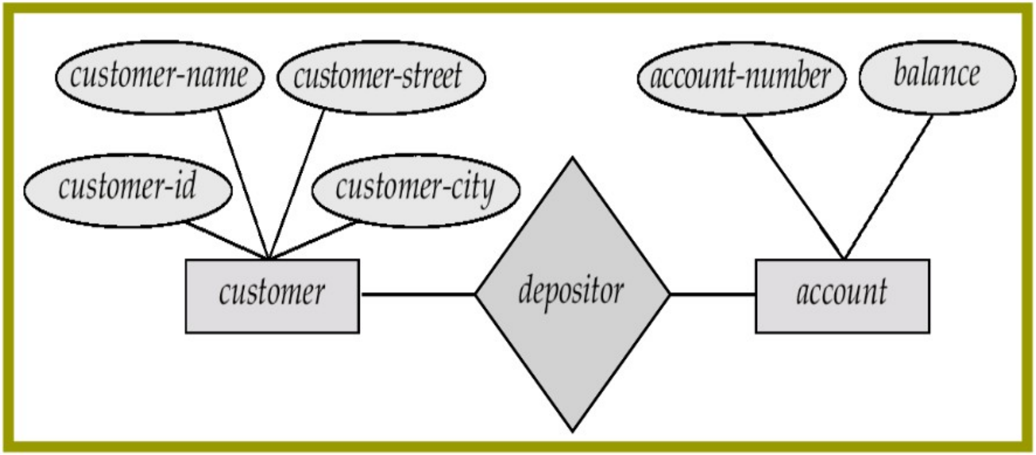
ER Diagram -Example



RelationshipTypes, sets and Instances

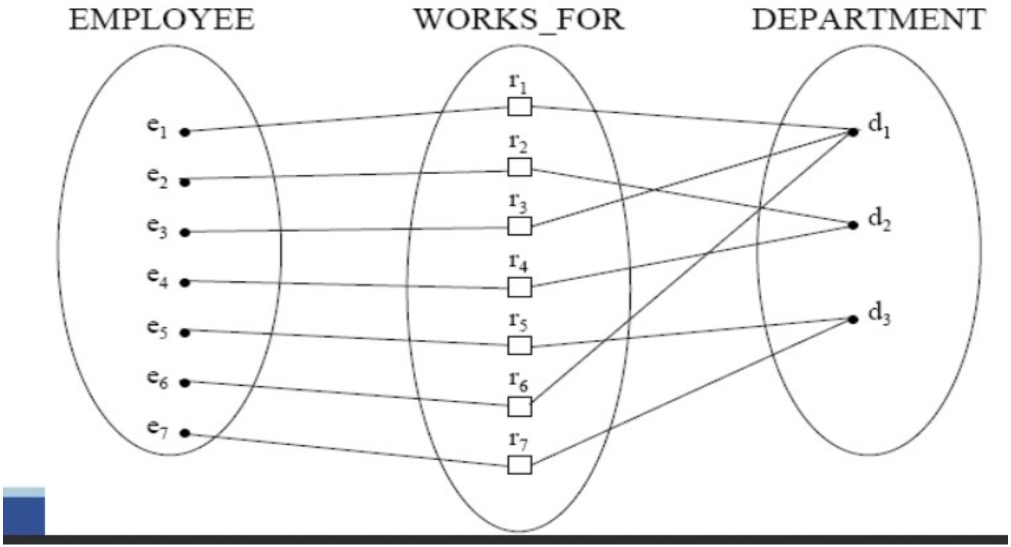
- An attribute of one entity type refers to another entity type, some relationship exists.
- A relationship **relates two or more distinct entities** with a specific meaning.
- Relationship Type defines **meaningful association** or a relationship set between (or among) entities from the entity types.
- The relationship set R is a set of relationship instances r_i
- Example: The Attribute Manager of DEPARTMENT refers to an employee who manages the department
- It is denoted by a **diamond symbol** in E-R Diagram.
- It is connected by straight lines to the rectangular boxes representing the participating entity types.

Example - Realtionship Type

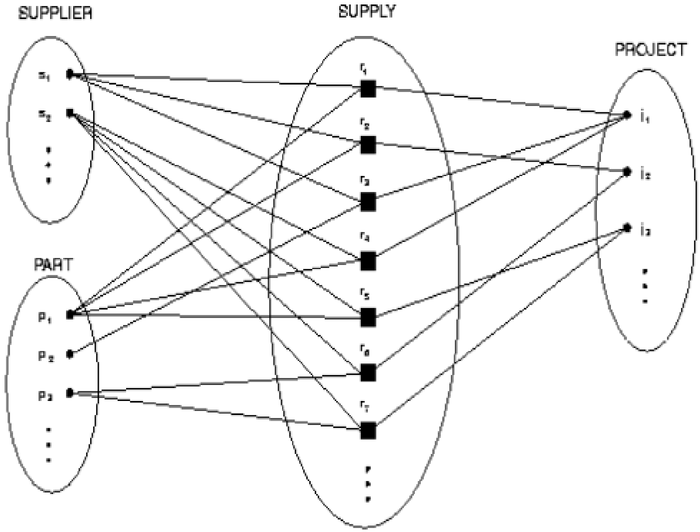


Relationship Degree

- The **degree** of a relationship type is the **number of participating entity types**.
- Example: WORKS_FOR realtionship is degree of two.
- A relationship type of degree two is called **binary**, and three is called **ternary**



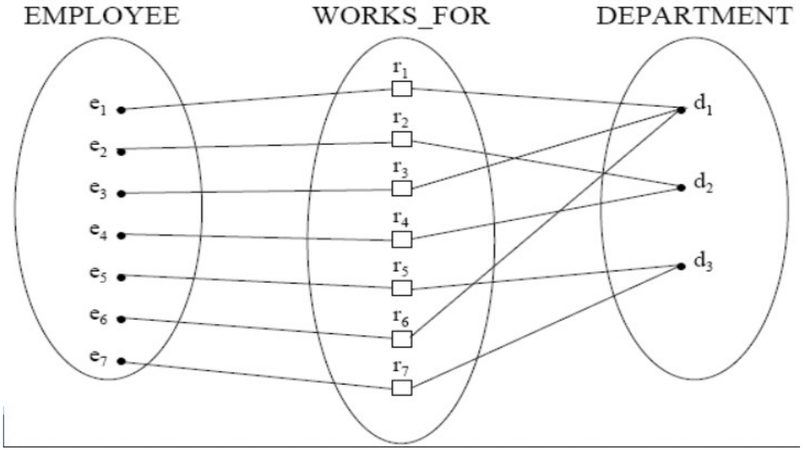
Relationship Degree - Ternary



Each realtionship instance r_i associates three entities a supplier s_i, part p_i and project j_i

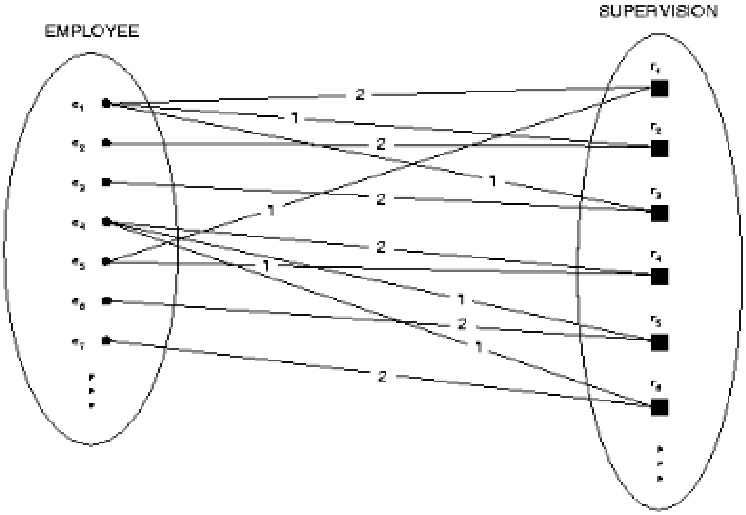
Role Names

- Role name signifies the **role that a participating entity** from the entity type plays in each relationship instance.
- Example : The works_for relationship type, EMPLOYEE plays the role of employee or worker and DEPARTMENT plays the role of department or employer.



Recursive Relationship

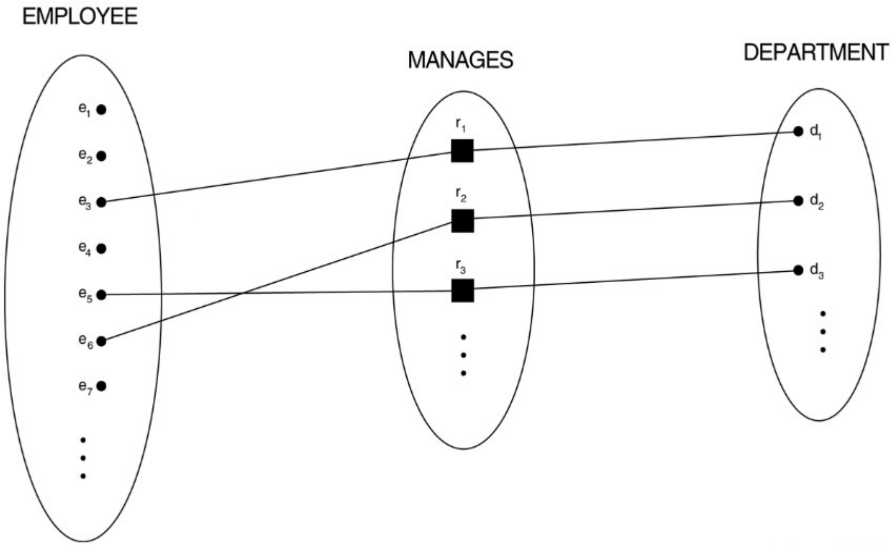
- Recursive relationship type: Same entity type participate in the relationship type **more than once in different roles**.
- Example: SUPERVISION relationships between EMPLOYEE (in role of supervisor or boss) and EMPLOYEE (in role of subordinate or worker).



Constraints on Relationship Types

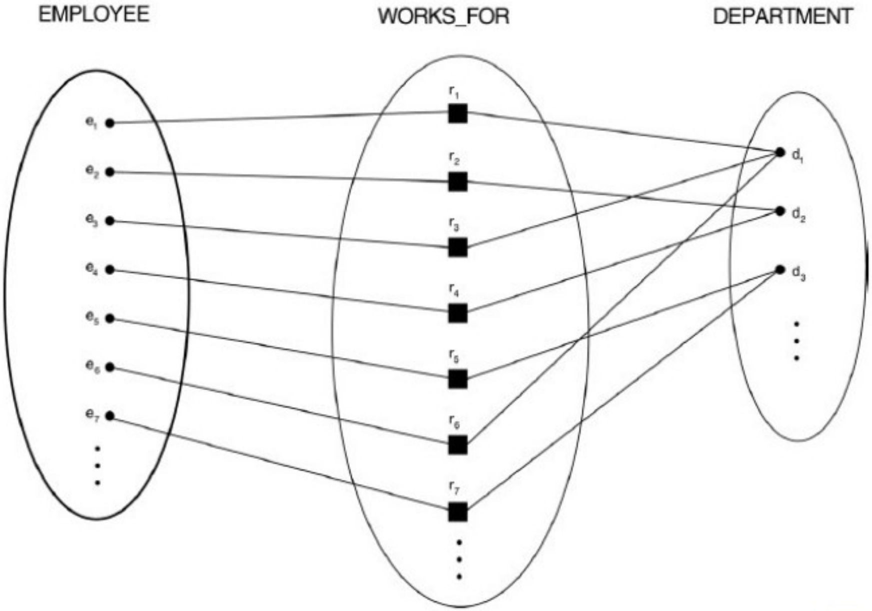
- Relationship types usually have certain constraints that limit the possible combinations of entities.
- These constraints are determined from the situation from the relationship that represent.
- Two main types of relationship constraints
  - **Cardinality Ratio**
  - **Participation**
- Cardinality Ratio of binary relationship specifies the **number of relationship instances** that an entity can participate in.
- The Cardinality Ratio for binary relationship types are **1:1,1:N,N:1 AND M:N**
- Shown By Placing Appropriate Number On The Link.

1: 1 Relationship Types

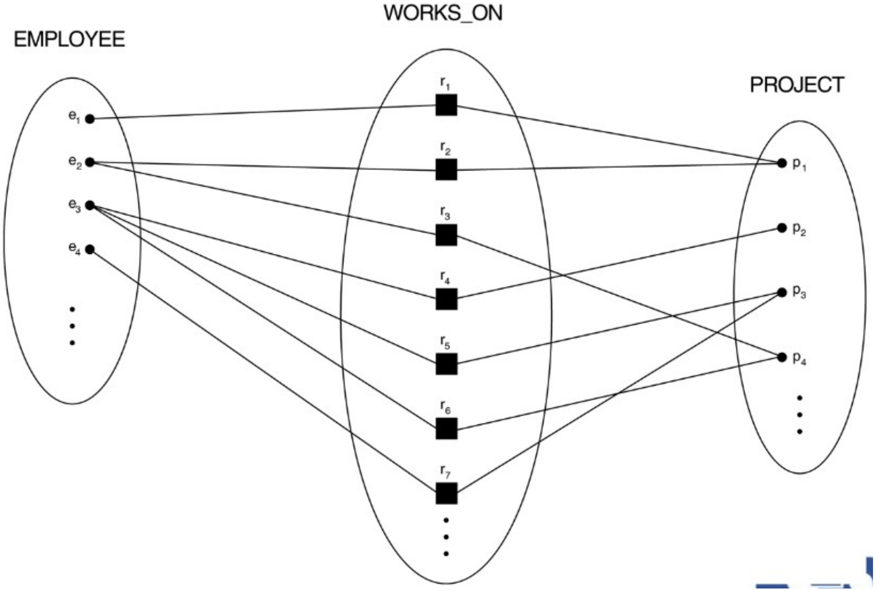


1: N Relationship Types

- The 1:N relationship : WORKS_FOR



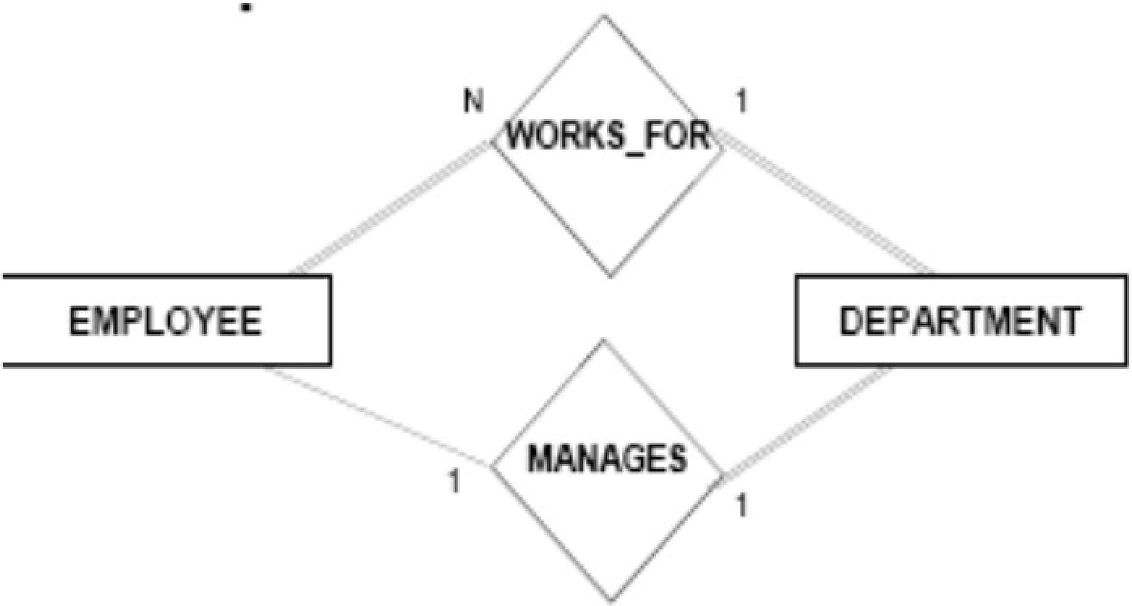
M: N Relationship Types



Participation Constraints on Relationship Types

- Determines **whether all or only some entity** occurrences participate in a relationship
- Two types of participation constraints
  - **Total participation**
  - **Partial participation**
- total = every entity needs to be 'related'
- e.g. EMPLOYEE works_in DEPARTMENT
- Total participation displayed as double line in ER Diagram
- partial = some entities are involved
- e.g. EMPLOYEE manages DEPARTMENT
- Partial participation displayed as single line in ER Diagram

Example for participation



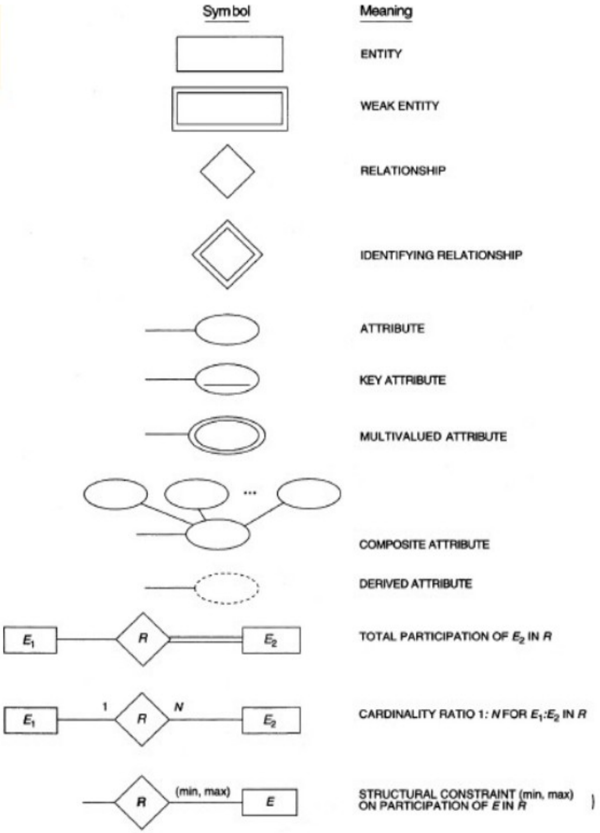
Note: double lines represent total participation

Weak Entity Type

- A weak entity type is one **without any key attributes** of its own.
- Entity types having key attribute $--->$ strong entity types
- Entities belonging to a weak entity type are **identified** by being related to another entity type called **identifying owner**
- The relationship type that relates a weak entity type to its owner is called **identifying relationshipof weak entity type**
- A weak entity type has **total participation** constraint w.r.t. its identifying relationship

ER Notations

ER Notations



Symbol

Meaning

ENTITY

WEAK ENTITY

RELATIONSHIP

IDENTIFYING RELATIONSHIP

ATTRIBUTE

KEY ATTRIBUTE

MULTIVALUED ATTRIBUTE

COMPOSITE ATTRIBUTE

DERIVED ATTRIBUTE

TOTAL PARTICIPATION OF E_2 IN R

CARDINALITY RATIO 1: N FOR E_1:E_2 IN R

STRUCTURAL CONSTRAINT (min, max) ON PARTICIPATION OF E IN R)

Company Database

The company is organized into DEPARTMENTs:

- Each department has a name, number,locations, an employee who manages the department and Manager_start_date. Locations is the only multivalued attribute. Also specify that both Name and Number are (separate) key attributes because each was specified to be unique.
- Each department controls a number of PROJECTs. Each project has a name, number and is located at a single location and controlling_department. Both Name and Number are (separate) key attributes because each was specified to be unique.

Company Database

- Store each EMPLOYEE's social security number, address, salary, sex, and birthdate. Name may be composite attributes. Each employee works for one department but may work on several projects. We keep track of the number of hours per week that an employee currently works on each project. We also keep track of the direct supervisor of each employee.
- Each employee may have a number of DEPENDENTs. For each dependent, we keep track of their name, sex, birthdate, and relationship to employee.

ER Notations

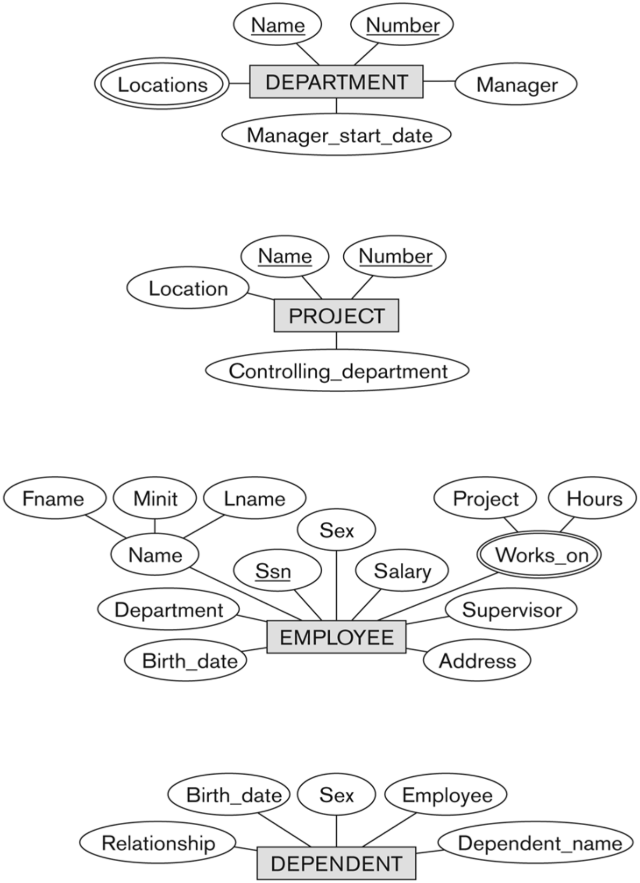


Figure 3.8
Preliminary design of entity types for the COMPANY database. Some of the shown attributes will be refined into relationships.

Refining the COMPANY database schema by introducing relationships

- By examining the requirements, six relationship types are identified
- All are binary relationships(degree 2)
- Listed below with their participating entity types:
  - **WORKS_FOR** (between EMPLOYEE, DEPARTMENT)
  - **MANAGES** (also between EMPLOYEE, DEPARTMENT)
  - **CONTROLS** (between DEPARTMENT, PROJECT)
  - **WORKS_ON** (between EMPLOYEE, PROJECT)
  - **SUPERVISION** (between EMPLOYEE (as subordinate), EMPLOYEE (as supervisor))
  - **DEPENDENTS_OF** (between EMPLOYEE, DEPENDENT)

Discussion on Relationship Types

Some attributes from the initial entity types are refined into relationships:

- Manager of DEPARTMENT – > MANAGES
- Works_on of EMPLOYEE – > WORKS_ON
- Department of EMPLOYEE – > WORKS_FOR

More than one relationship type can exist between the same participating entity types

- MANAGES and WORKS_FOR are distinct relationship types between EMPLOYEE and DEPARTMENT

EMPLOYEE participates twice in two distinct roles:

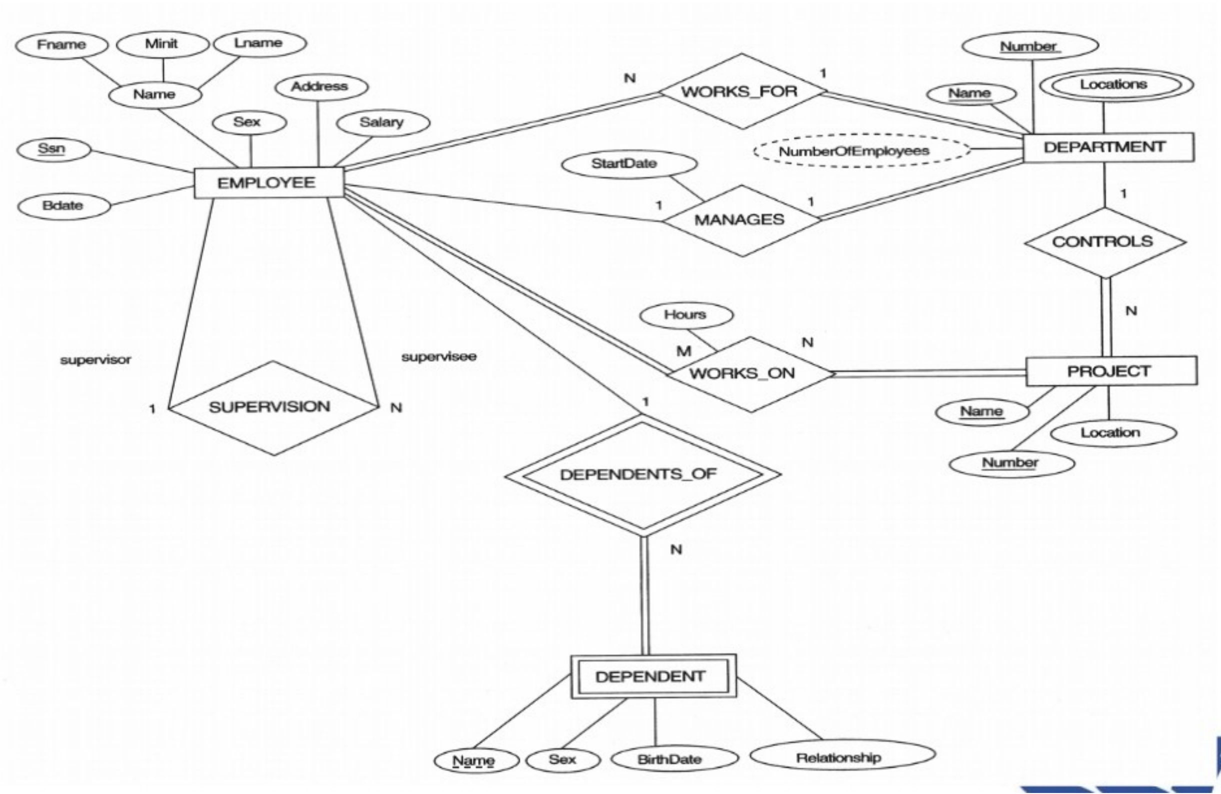
- Supervisor (or boss) role
- Supervisee (or subordinate) role

Discussion on Relationship Types

Weak Entity type

- A DEPENDENT entity is identified by the dependent's first name, and the specific EMPLOYEE with whom the dependent is related
- Name of DEPENDENT is the partial key
- DEPENDENT is a weak entity type
- EMPLOYEE is its identifying entity type via the identifying relationship type DEPENDENT_OF

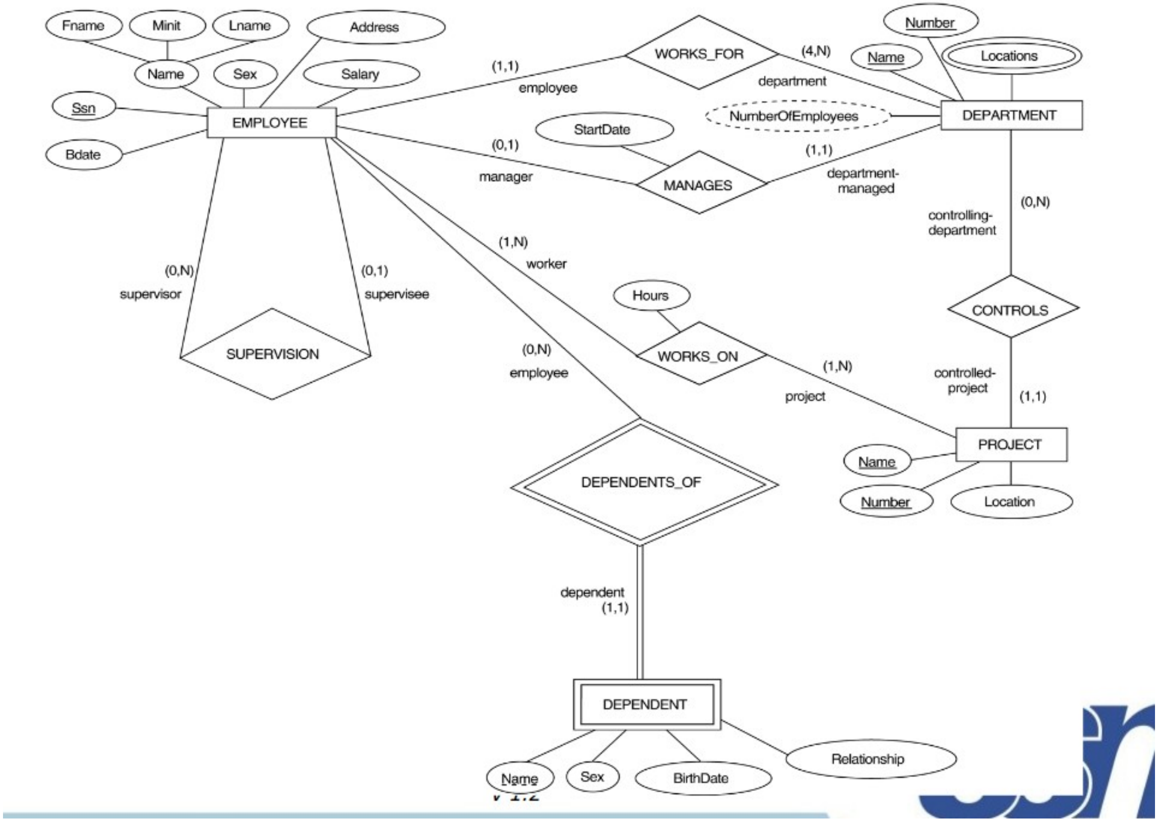
ER DIAGRAM


Fname
Minit
Lname
Name
Address
Sex
Salary
Ssn
Bdate
EMPLOYEE
N
WORKS_FOR
1
Number
Name
Locations
DEPARTMENT
NumberOfEmployees
StartDate
1
MANAGES
1
1
CONTROLS
N
supervisor
supervisee
1
SUPERVISION
N
Hours
M
N
WORKS_ON
PROJECT
Name
Location
Number
1
DEPENDENTS_OF
N
DEPENDENT
Name
Sex
BirthDate
Relationship

Structural Constraints

- Alternative notation for specifying structural constraints on relations.
- Involves associating pair of integer numbers (min,max) with each participation.
    - $0 <= min <= max$ and $max >= 1$
    - $Min = 0$ partial participation
    - $Min > 0$ total participation

Alternative notation of ER DIAGRAM



Reference

- Fundamentals of Database systems 7^{th} Edition by Ramez Elmasri.
- Oracle 9i SQL