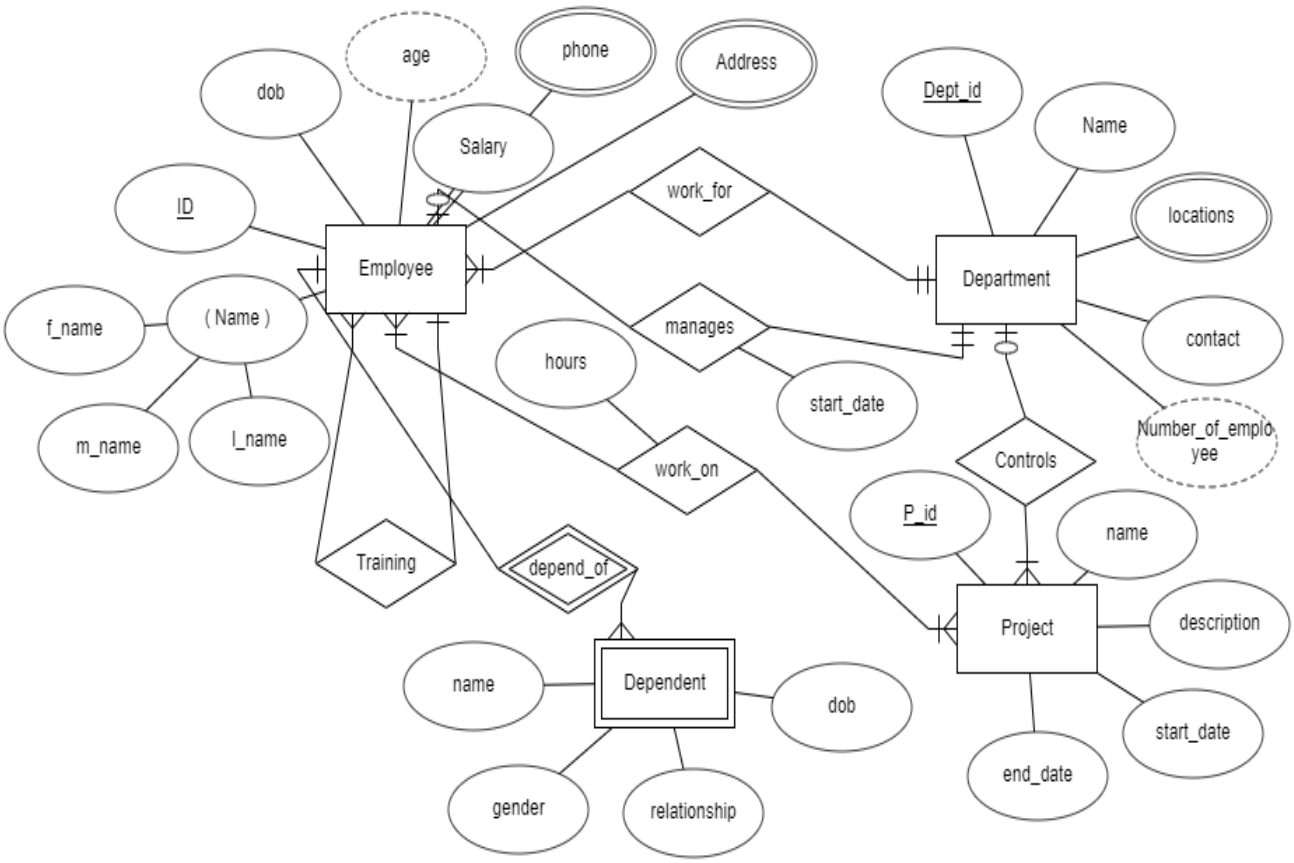
Database Design

- **Constructing ER Diagram**
- **Reduction to Relation Schema/ ERD to Table**



Reduction to Relation Schema

- Representation of Strong Entity Sets with Simple Attributes
- *Representation of Strong Entity Sets with Complex /Multivalued Attributes*
- *Representation of Weak Entity Sets*
- Representation of Relationship Sets
  - *One to one Relationships*
  - *One to many Relationships*
  - *Many to Many Relationships*
  - *Recursive Relationships*
- Attributes of Relationships

Rules for Reduction:

Entity and Attributes:

- **Regular/Strong entity:** Create a relation and indicate it's Primary Key (***Figure 1***)
- **Multivalued and Composite Attributes**: make a separate relation with an attribute of the same name and a PK of the entity that it's the attribute of as the FK of this relation. Primary key is the FK itself. (***Figure 1***)

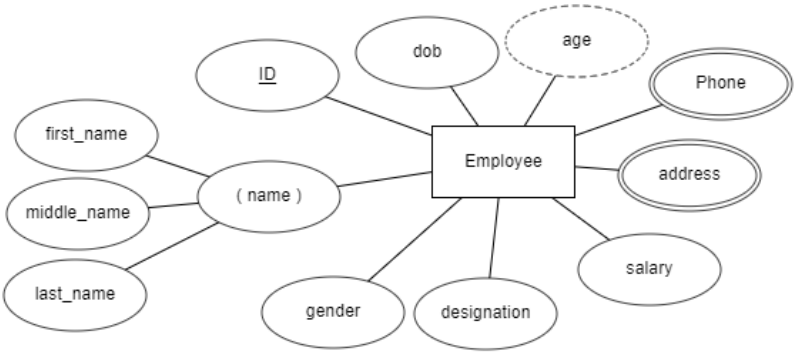


Figure 1: Entities and Attributes

Schemas

- Employee(emp_id, first_name, middle_name, last_name, designation, email, gender, dob, salary)
- Phone (*emp_id*, *phone_no*)
- Address (*emp_id*, *address*, upazilla, district)

Weak Entity:

- Create a relation with it's name and add it's own attributes and the PK of the entity it's dependent on as a FK . The PK of weak entity would be a mixture of FK and it's partial key. *These primary keys are called composite primary key* ***[formed a PK with multiple attributes]****.[****Figure 2****]*

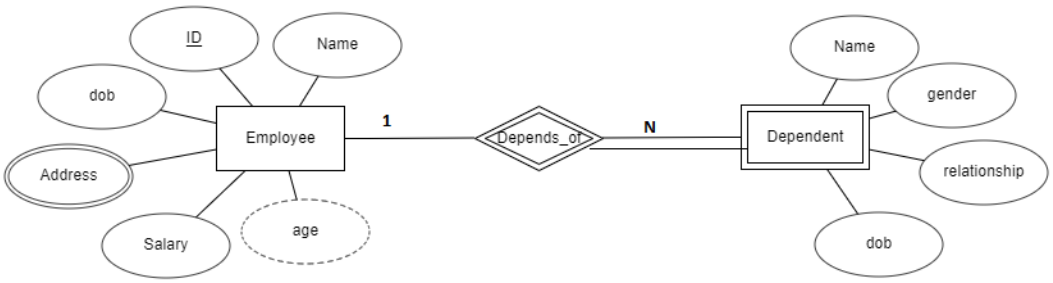


Figure 2: Weak Entity

Schema:

Employee(emp_id, name, dob, salary)

Dependent(***emp_id***, *name*, dob, gender, relationship)

1:1 Relationship:

- Include the PK of the entity ***that optionally participates*** in the relationship as the FK of the ***total Participating*** entity's relation table. [***Figure 3***]
- If both entities have ***Total Participation*** the a relation with the relationship name and include all of the entities as attributes. Choose anyone as primary key.

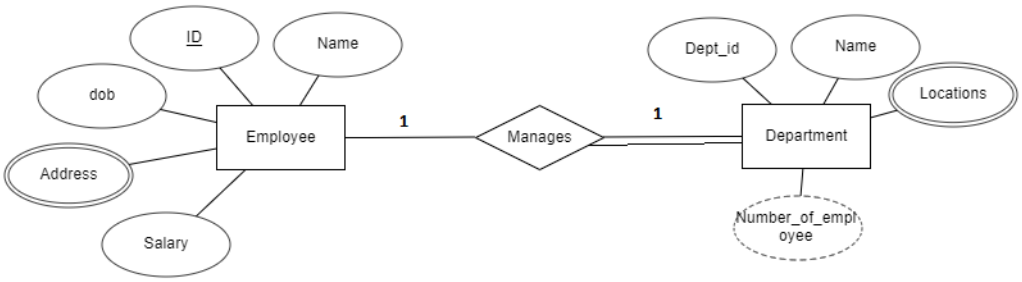


Figure 3: 1:1 Relationship

Schemas:

Employee(ID, name, dob, Salary)
Address(*emp_id*, home_name, street, district)
Department(Dept_id, name, ***manager_id***)
Locations(*dept_id*, address)

1:N Relationship:

For this type of relationship include the PK of the entity that's on the 1 side as the FK of the entity on the N side.[***Figure 4***]

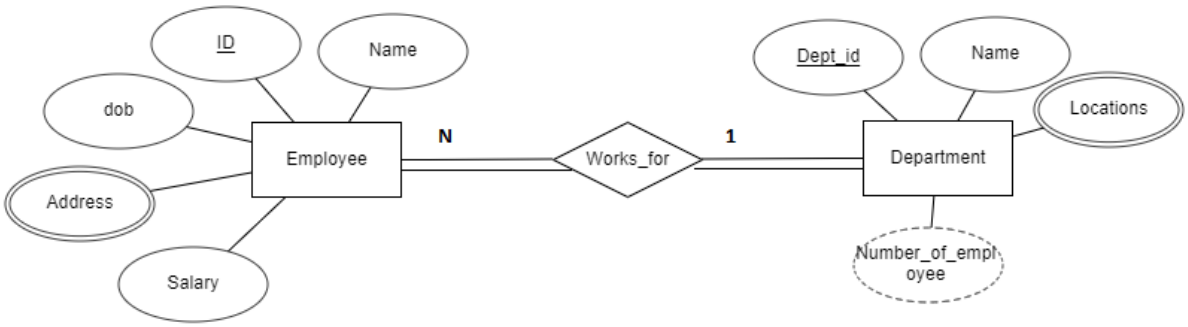


Figure 4: 1:N Relationship

Schemas:

Employee(ID, name, dob, Salary, ***dept_id***)

Address(*emp_id*, home_name, street, district)

Department(Dept_id, name, *locations*)

M:N Relationship:

For these type of relationship make another relation with the relationship's name and include its own attributes (if have any) and the PK of both entities as the FK. The combo of the both FKs will make the primary key. [***Figure 5***]

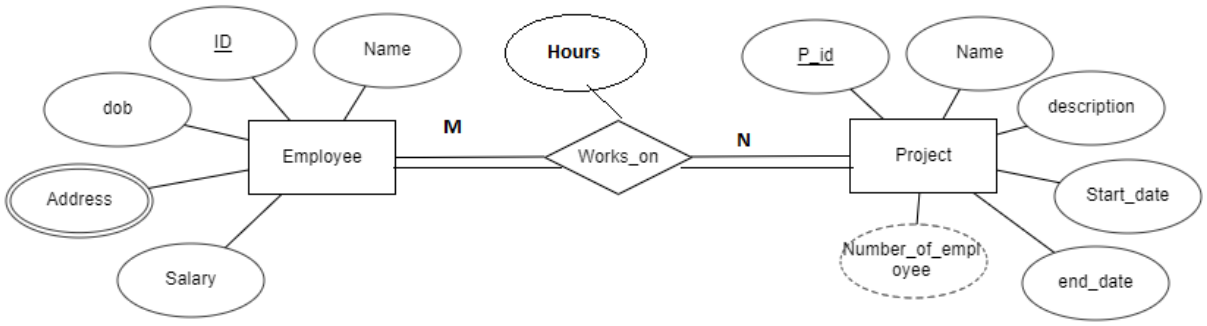


Figure 5: M-N Relationship

Schemas:

Employee(ID, name, dob, Salary)

Project (p_id, name, description, start_date, end_date)

Works_on(*emp_id, p_id*, hours)

Recursive Relationship:

Include a new FK that is actually the PK of the entity itself but with a different name. [***Figure 6***]

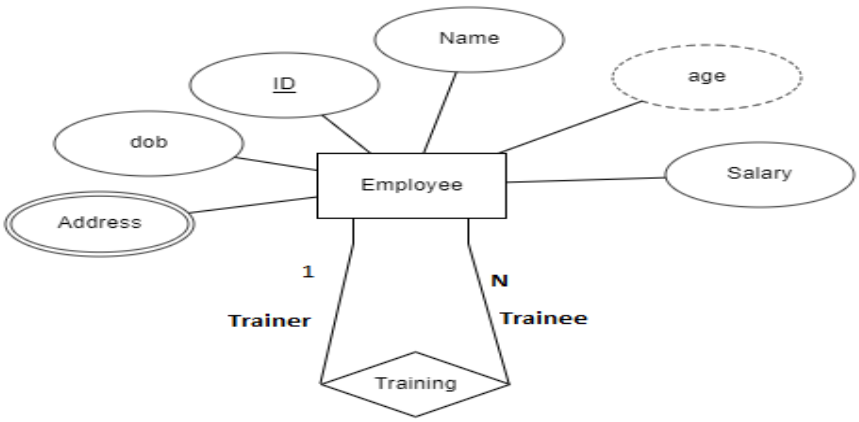


Figure 6: Recursive Relationship

Schemas:

Employee(ID, name, dob, Salary, ***trainer_id***)

Attributes of Relationships:

- For many to many relationship add the attributes to the new relation
- For other relationships, add the attribute to the ***N relationship entity side*** (1-N) or ***total participation side*** (1-1) [***Figure 7***]

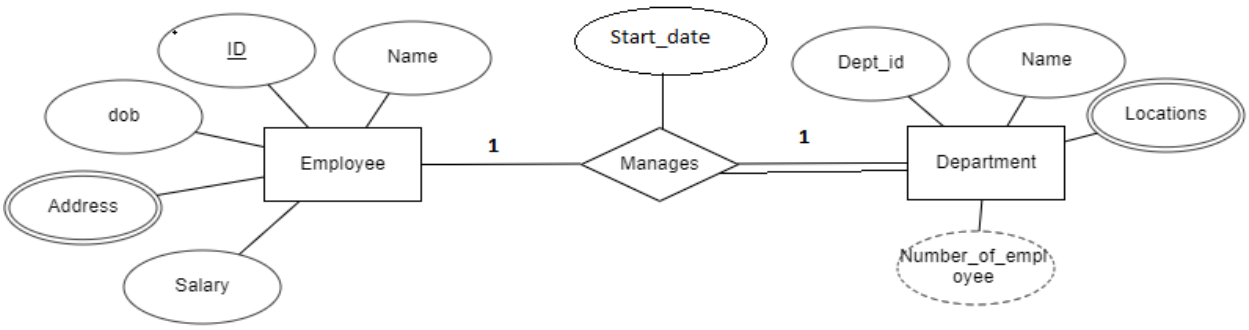
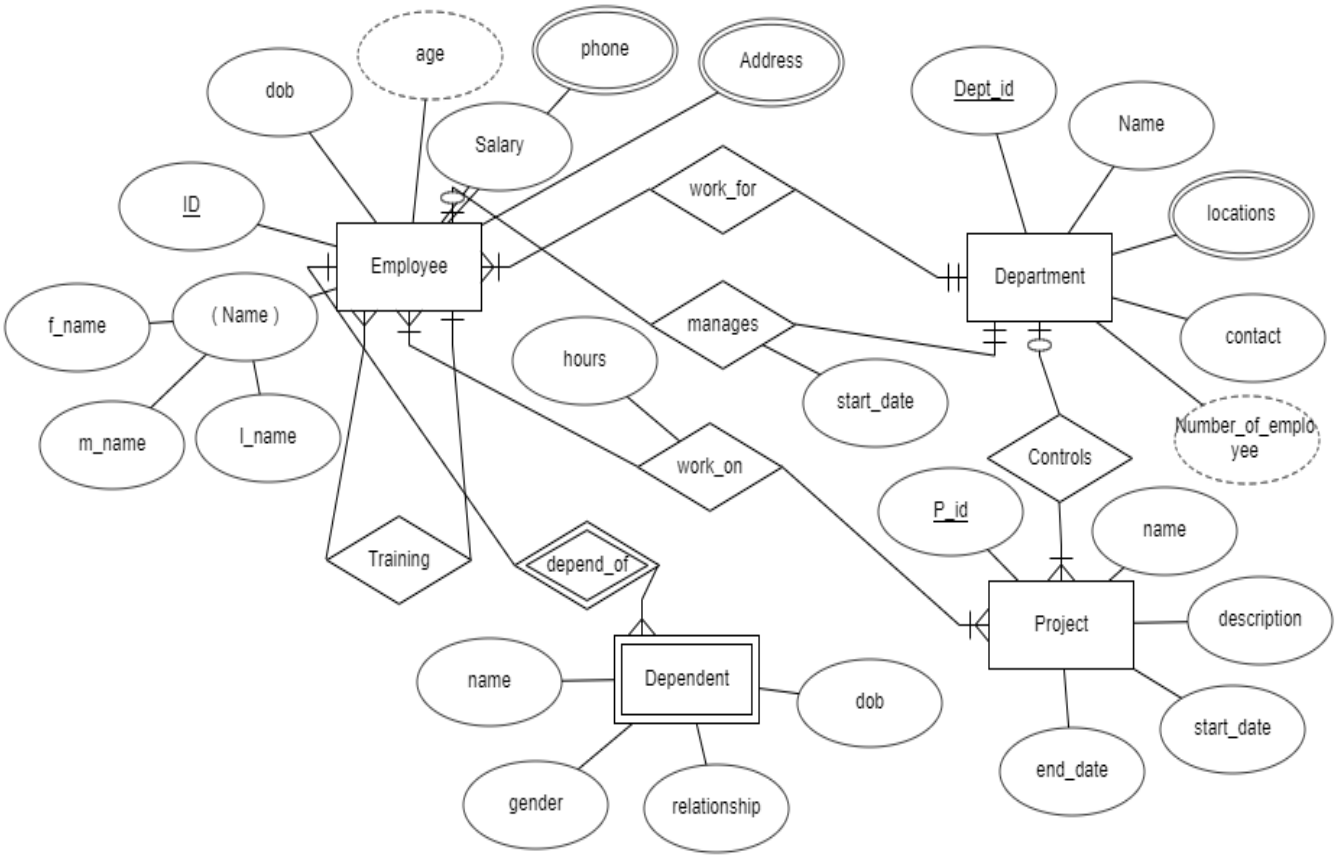


Figure 7: Attributes of Relationship

Schemas:

Employee(ID, name, dob, Salary)
Department(Dept_id, name)
Manager(Manager_id, *employee_id, dept_id*, joining_date)

Full ERD fro Human Resource Database



Relations/Tables:

Employee(emp_id, f_name, m_name, l_name, salary, dob, *trainer_id, dept_id,*)

Employee_Phone(*emp_id*, phone_no)

Employee_Address(*emp_id*, home_name, street, district, division)

Dependents(*emp_id*, name, gender, dob, relationship)

Department(dept_id, name, contact)

Manager(Manager_id, *employee_id, dept_id*, joining_date, end_date)
Dept_locations(*dept_id*, location)

Project(p_id, name, description, start_date, end_date, dept_id)

Work_on(*emp_id, p_id*, hours)

Schema Definition: