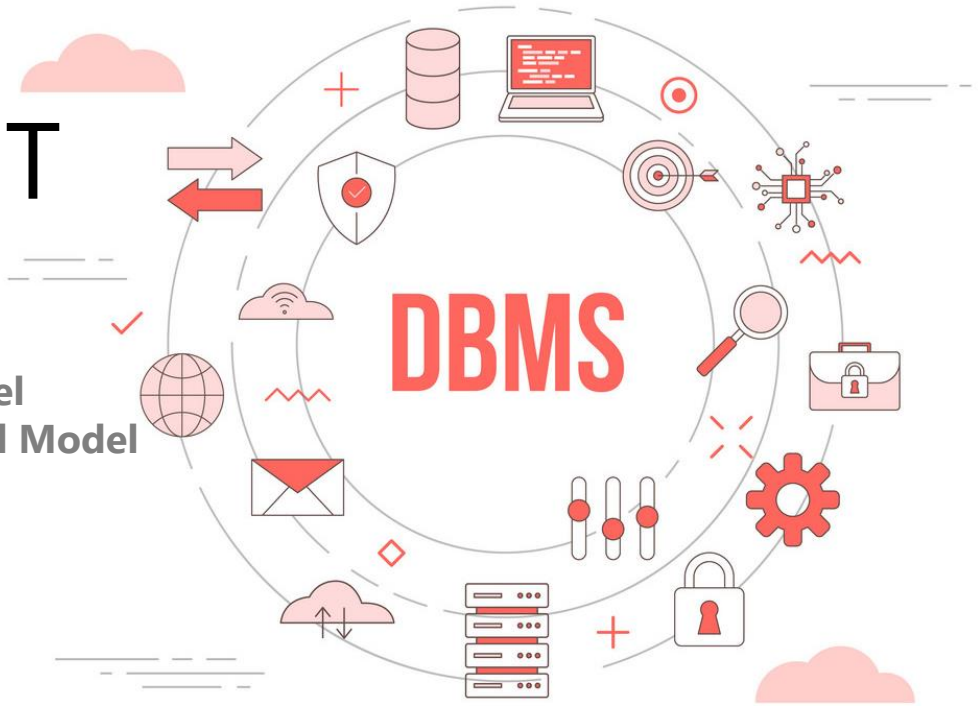
DATABASE MANAGEMENT SYSTEMS

Conversion of ER model to Relational Model



Dr. Jay Sarraf

School of Computer Engineering

KIIT Deemed to be University

Overview

The notations may be used to represent the database, and these notations can be condensed into a set of tables.
A group of relational schemas can be used to depict a database that adheres to an ER diagram schema. Both the ER model and the Relational data model are logical, abstract representations of businesses in the actual world.

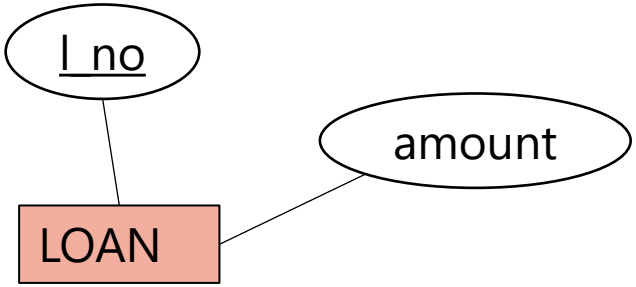
ER diagram to table conversion considerations include the following:

- Entity type becomes a table.
- All single-valued attribute becomes a column for the table.
- A key attribute of the entity type represented by the primary key.
- The multivalued attribute is represented by a separate table.
- Composite attribute represented by components.
- Derived attributes are not considered in the table.

Conversion

1. Representation of Strong Entity sets

A schema with the same properties can be created from a strong entity set. The resultant schema's main key is the entity set's primary key.

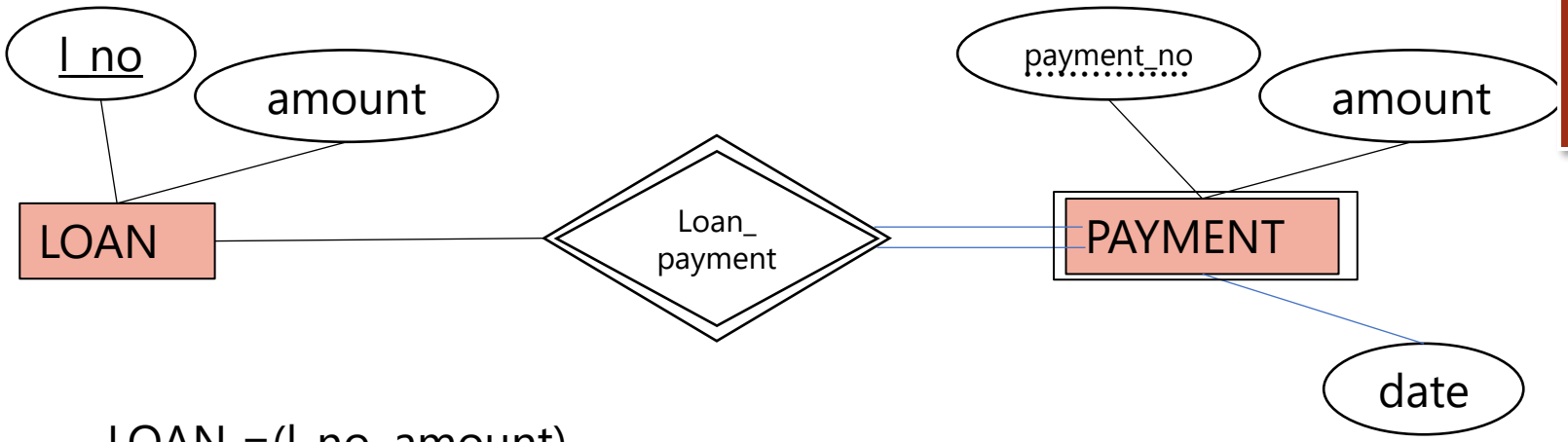


LOAN =(l_no, amount)

Conversion

2. Representation of Weak Entity sets

A table with a column for the main key of the identifying strong entity set is created from a weak entity set. The collection of foreign keys and partial keys creates the primary key.



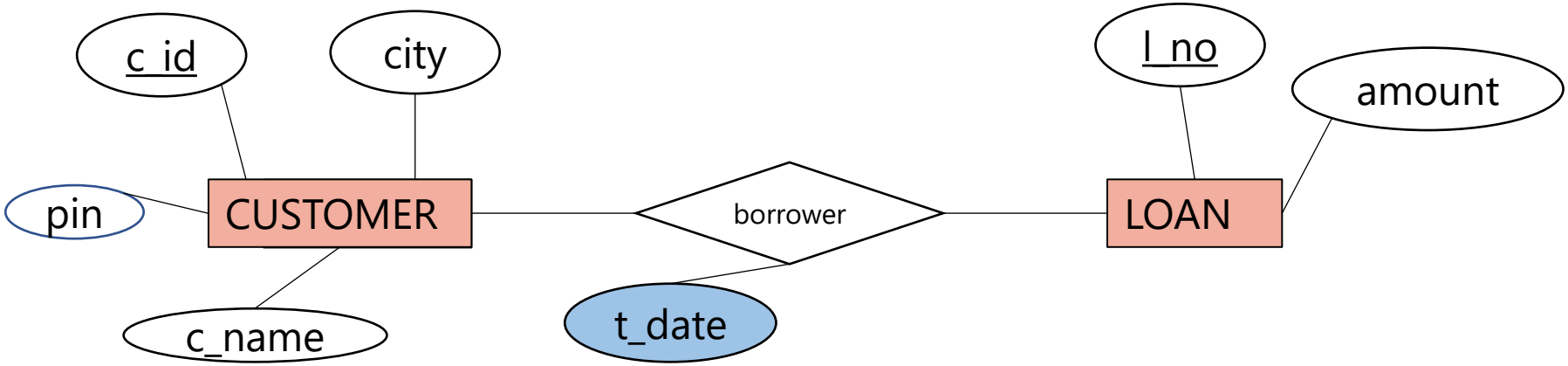
LOAN =(l_no, amount)
PAYMENT =(loan_no, payment_no, payment_date, payment_amt)

Conversion

3. Representation of Relationship sets

Binary M:N

The main key of the connection is formed by joining the primary key characteristics from the participating entity sets.



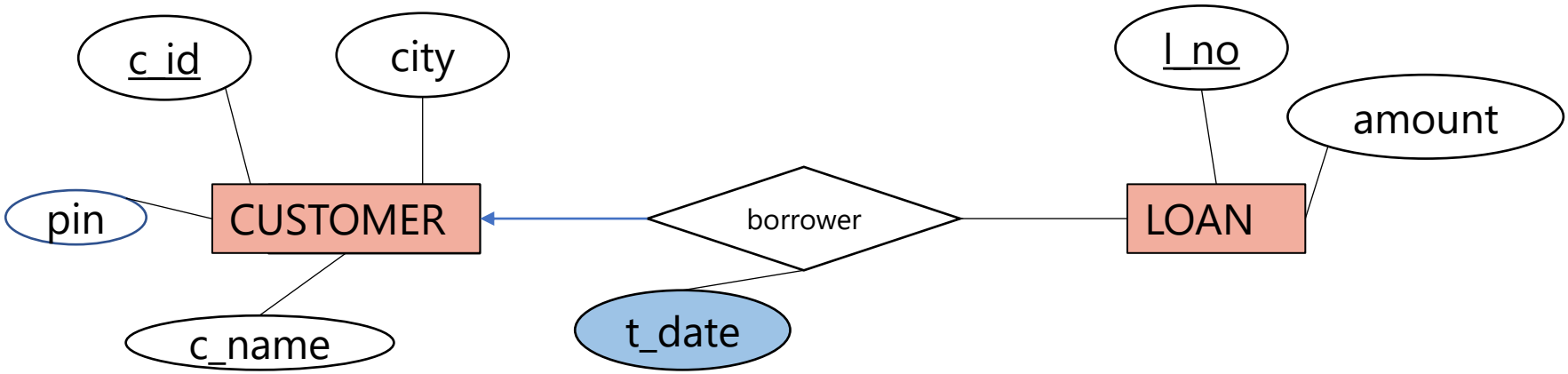
Customer = (cid, c_name, pin, city)
Loan = (loan_no, amount)
Borrow = (cid, loan_no, t_date)

Conversion

3. Representation of Relationship sets

Binary M:1/1:M

Create two tables: one for the entity set at 1 side and the other for the entity set at M side. Include the descriptive characteristics in both tables as well as a reference to the primary key of 1 side in the entity set at M side.

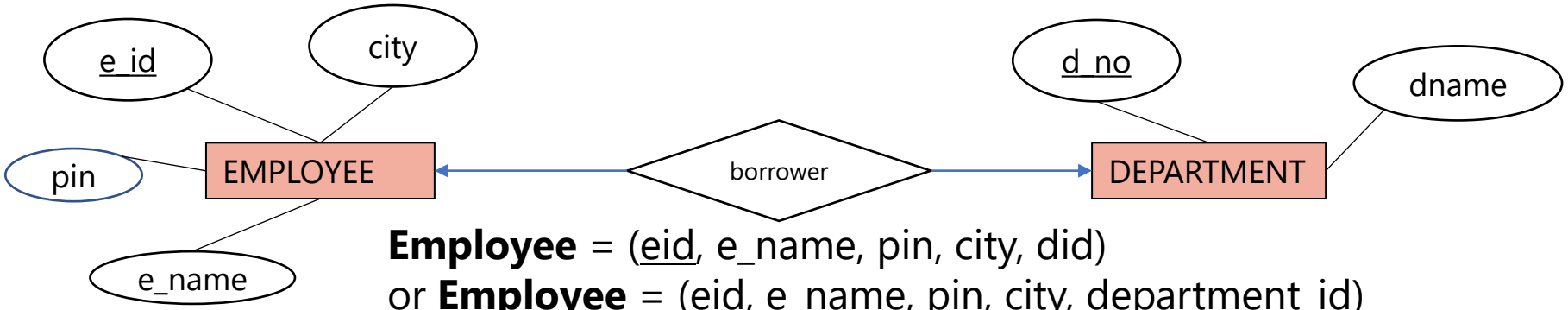


Customer = (cid, c_name, pin, city)
Loan = (loan_no, amount, t_date, cid)
Or **Loan** = (loan_no, amount, t_date , borrowing_cid)

Conversion

3. Representation of Relationship sets

Binary 1:1
Build two tables. Either side may be picked to represent the numerous sides in this situation. So, while additional characteristics may be added to any of the tables corresponding to the two entity sets, they cannot be added simultaneously.



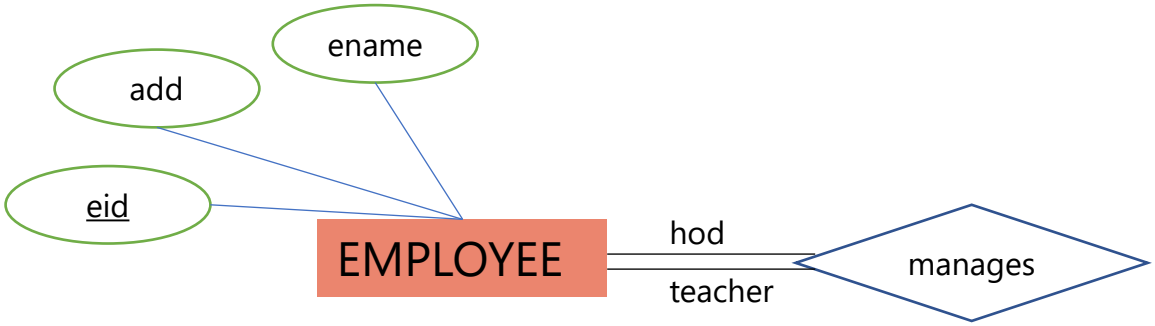
Employee = (eid, e_name, pin, city, did)
or **Employee** = (eid, e_name, pin, city, department_id)
Department = (d_no, dname)

If department set considered as many side:
Employee = (eid, e_name, pin, city)
Department = (d_no, dname,eid)

Conversion

4. Representation of Recursive Relationship sets

For a recursive relationship two tables will be constructed; one for entity set and one for relationship set.



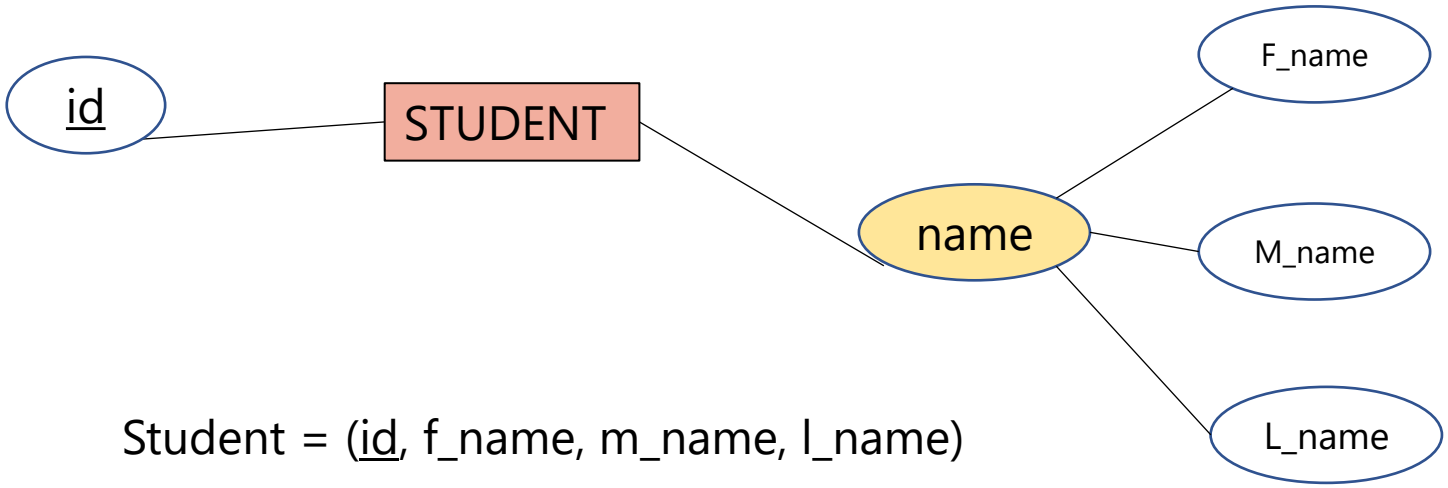
Employee = (eid, ename, address)
Manages= (hodid, teacherid)

This ER diagram can also be represented by using a single relation schema.

Employee = (eid, ename, address, hodid)

5. Representation of Composite attributes

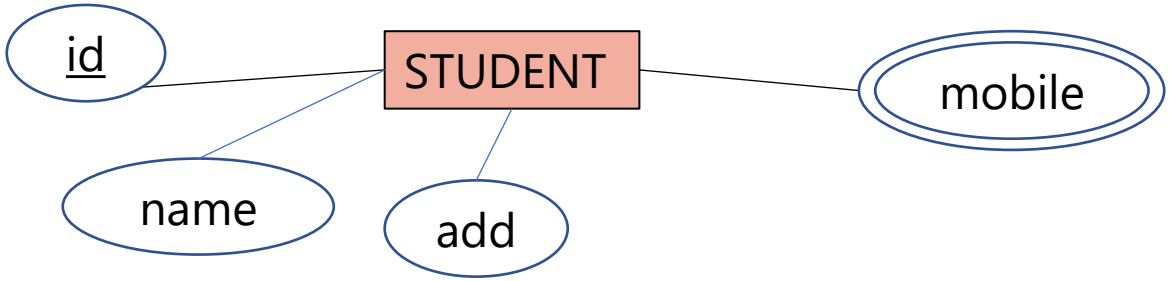
By generating a distinct property for each of its components, the composite attributes are flattened down.



Student = (id, f_name, m_name, l_name)

6. Representation of Multi-valued attributes

A distinct schema E M is used to express a multi-valued attribute M of an entity set E. (primary key of E,M).

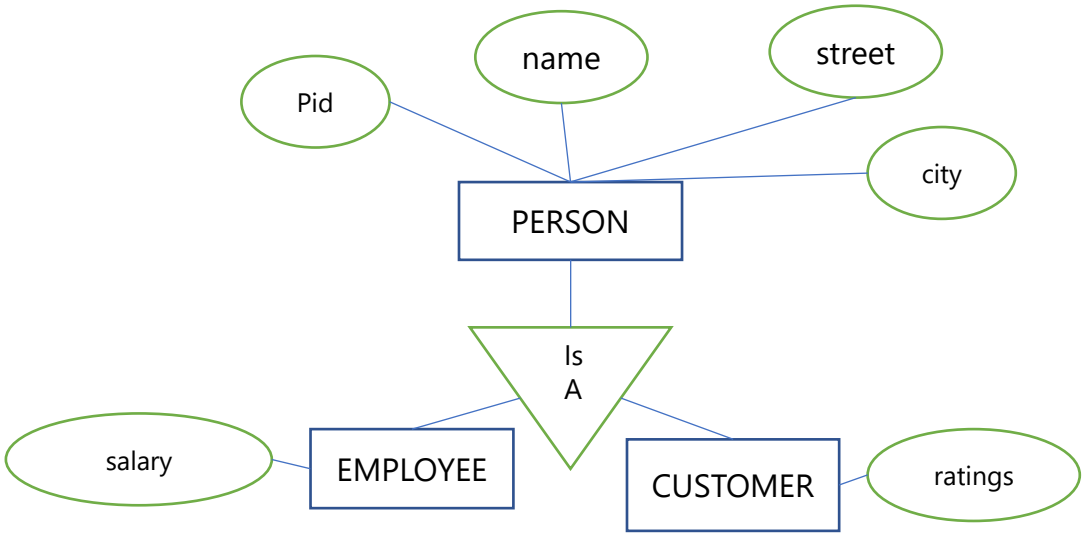


Student = (id, name, add)
Student_mobile = (id, mobile)

Conversion

7. Representation of Generalization/Specialization

One schema will be created for the generalized entity set and one for each of the specialized entity sets in the event of an ER diagram relating to generalization and specialization.



Person = (<u>pid</u>, name, street, city)
Employee = (pid, salary)
Customer = (pid, credit_rating)

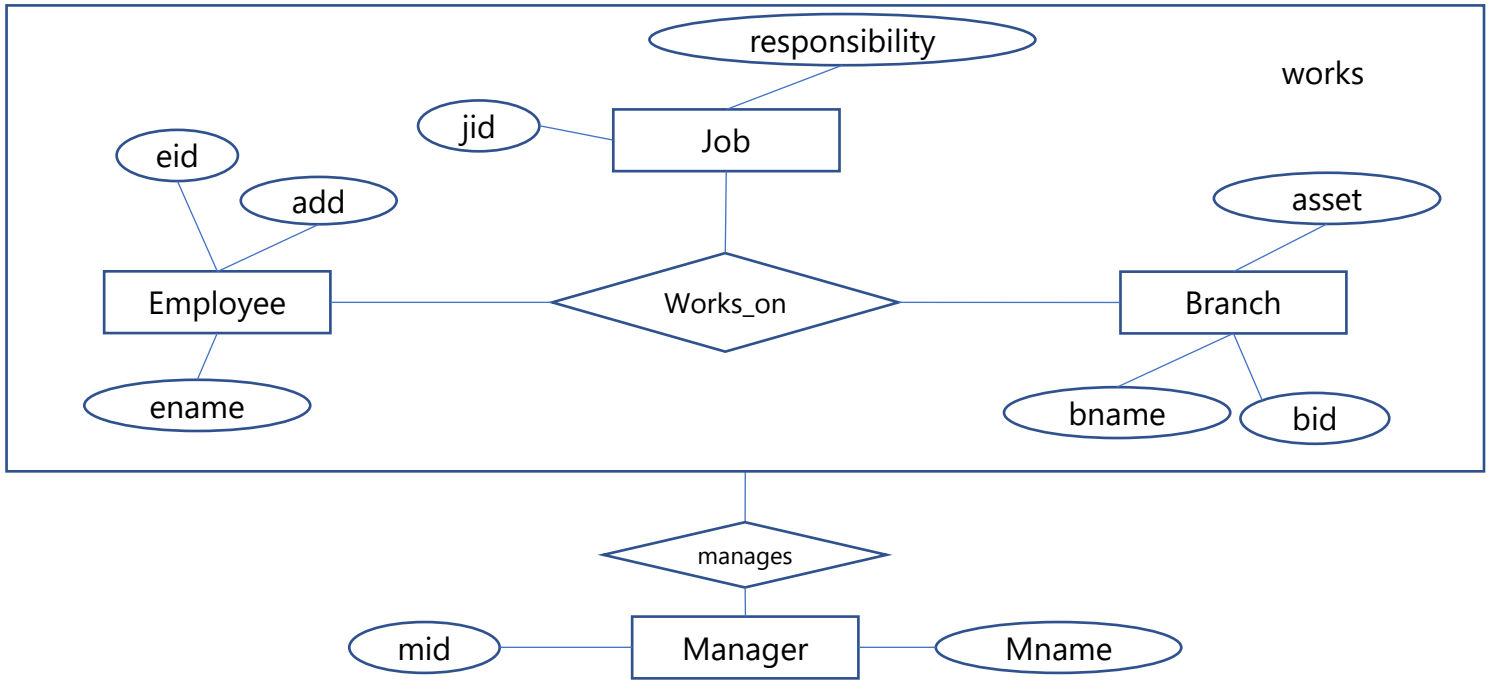
7. Representation of Generalization/Specialization

Schemas are only built for specialized entity sets when generalization/specialization is a disjointness scenario.

Employee = (employee_id, name, address, salary)
Customer = (customer_id, name, address, credit_rating)

8. Representation of Aggregation

Create a schema with the primary key of the aggregated connection, primary key of the related entity set, and descriptive characteristics to indicate aggregation (if any)



Conversion

8. Representation of Aggregation

Create a schema with the primary key of the aggregated connection, primary key of the related entity set, and descriptive characteristics to indicate aggregation (if any)

Employee = (eid, ename, address)
Branch = (bid, bname, asset)
Job = (jid, position, responsibility)
Works_on = (eid, bid, jid)
Manager = (mid, mgrname)
Manages = (eid, bid, jid, mid)