# Logical Database Design
# Mapping ERD to Relational - Part 2

Dr. Jeevani Goonetillake

UCSC

# Map Unary Relationships

Procedure depends on both the degree of the relationships and the cardinalities of the relationships.

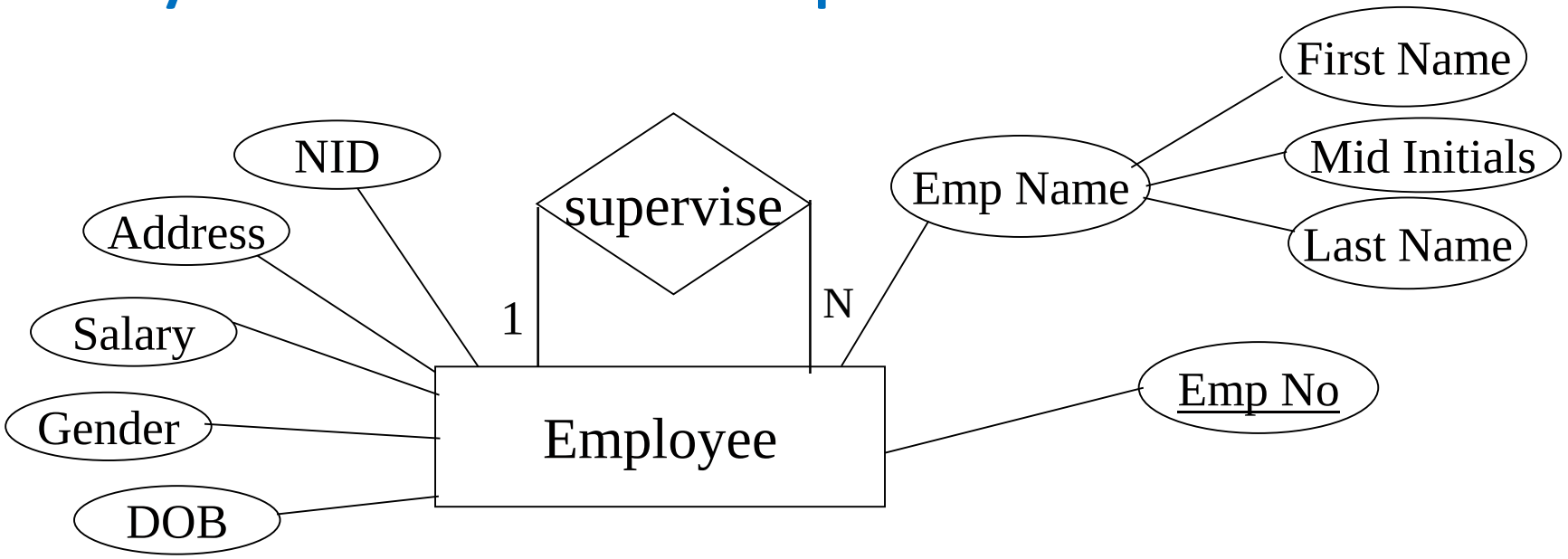
- Map Unary One-to-Many Relationships
- Map Unary Many-to-Many Relationships
- Map Unary One-to-One Relationships

# Map Unary 1:N or 1:1 Relationships

The same rule is adopted in mapping unary relationships with cardinality ratios 1:1 and 1:N.

- Create a relation for the entity type.
- Include PK of the entity as a foreign key within the same relation with any attributes associated with the relationship.

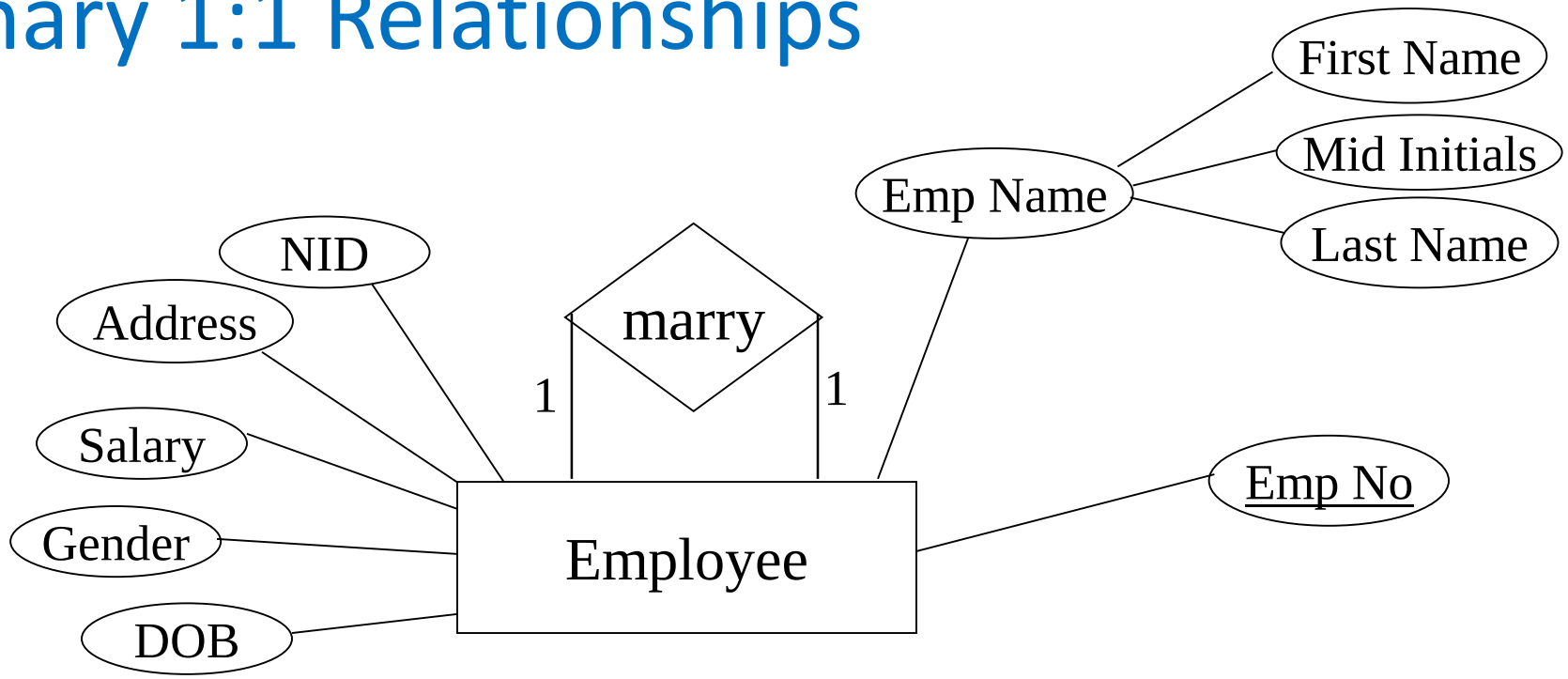
# Map Unary 1:N Relationships




Employee(Emp_No, NID, Address, Salary, Gender, DOB, First_Name, Mid_Initials, Last_Name, Dept_No, *Supervisor*)

FK/Null

# Map Unary 1:1 Relationships




Employee(Emp_No, NID, Address, Salary, Gender, DOB, First_Name, Mid_Initials, Last_Name, Dept_No, *Marry*)

FK/Null

# Map Unary M:N Relationships

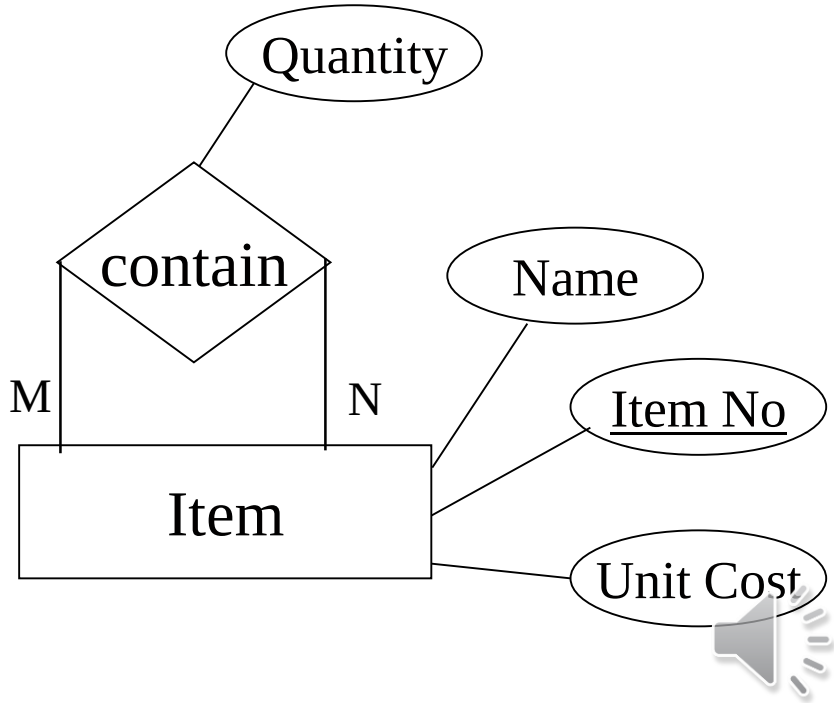
- Create a relation for the entity type
- Create new relation and include PK of the entity type as FK twice. These attributes may become the PK (composite)
- Include any attributes of the relationship to the new relation.

Item(Item_No, Name, Unit_Cost)

FK/NN

*Component(Item_No, Component_No, Quantity)*

FK/NN



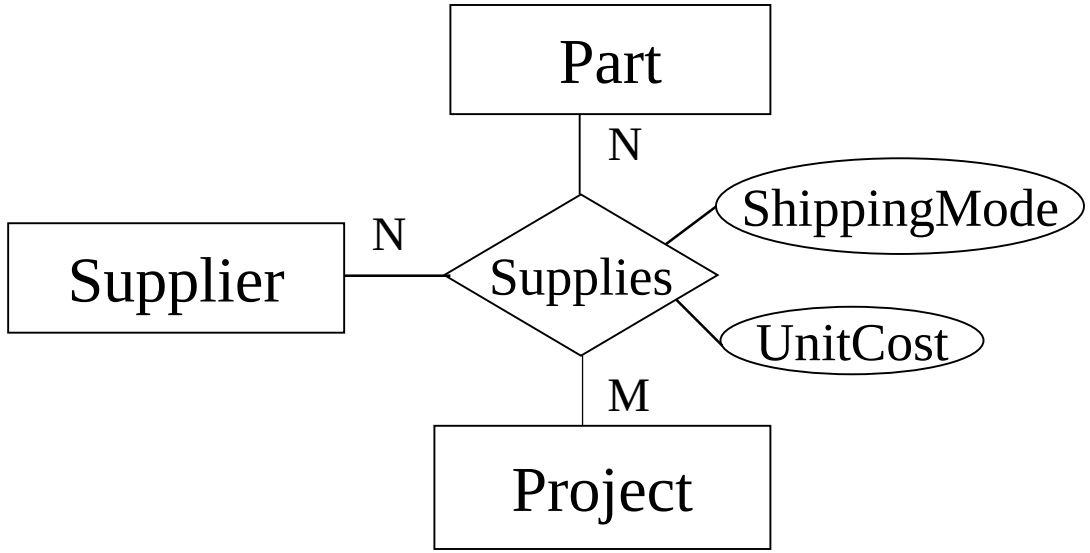
# Map Complex Relationship Types

Create relation to represent relationship and include any attributes that are part of the relationship.

- Post copy of primary key attribute(s) of entities that participate in the complex relationship into new relation, to act as foreign keys.

- Any foreign keys that represent a ‘many’ relationship generally will also form the primary key of new relation, possibly in combination with some of the attributes of the relationship.

# Map Complex Relationship Types




Supplier(supplier_no, ….)

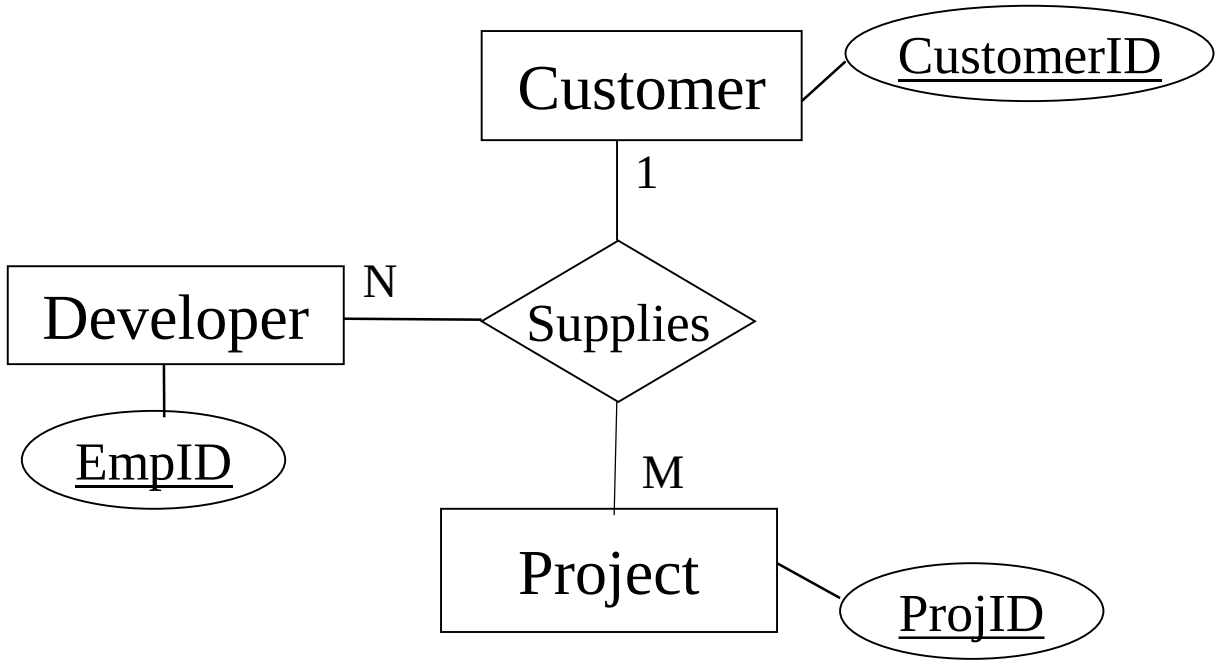
Project(project_no, ….)

Part(part_no, ….)

Supply(*supplier_no*, project_*no*, *part_no*,Shipping_mode, Unit_Cost)

# Map Complex Relationship Types

When ternary relationships are expressed in a relational model, each of the entity type with a "many" cardinality indicator becomes part of the primary key.




Customer(CustomerID, ….)
Project(ProjID, ….)
Developer(EmpID, ….)
Supplies(EmpID, ProjID,CustomerID)

# Mapping Associative Entity Types

- Follow similar steps to mapping an M:N relationship
- Three relations are created, one for each of the two participating entity types and the third for the associative entity.
- The relation formed is called the associative relation.



# Identifier Not Assigned

- The default primary key for the associative relation consists of the two primary key attributes from the other two relations.

- These attributes are then foreign keys that reference the other two relations.



# Identifier Assigned

Sometimes an identifier (called a surrogate identifier or key) is assigned to the associative entity type on the ER diagram. There are 2 possible reasons:

i) The associative entity type has a natural identifier that is familiar to end users.

ii) The default identifier (consisting of identifiers for each of the participating entity types) may not uniquely identify instances of the associative entity.

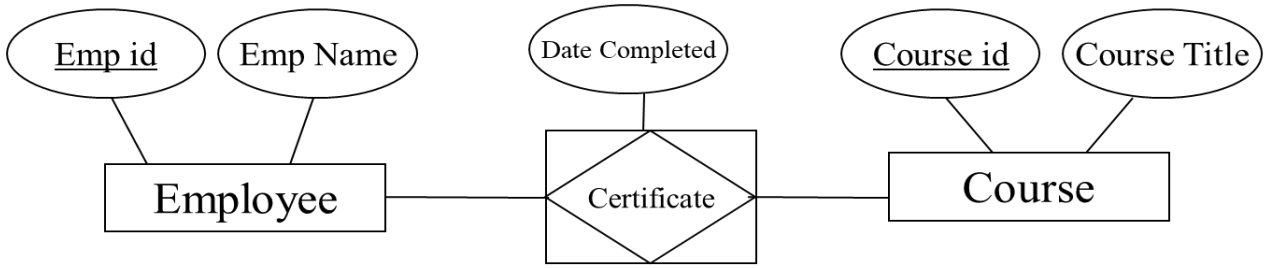


# Identifier Assigned

- A new associative relation is created to represent the associative entity.
- However, the primary key for this relation is the identifier assigned on the ER diagram.
- The primary keys for the two participating entity types are then included as foreign keys in the associative relation.



# Mapping Associative Entity Types




## Identifier Not Assigned

Employee(Emp_id, ….)
Course(Course_id, ….)
Certificate(**Emp_id, Course_id**,DateCompleted)

## Identifier Assigned

Employee(Emp_id, ….)
Course(Course_id, ….)
Certificate(**Certificate_id**, Emp_id, Course_id, DateCompleted)