CONSTRAINTS & ENTITIES

Database Management System

Sujan Tamrakar

Constraints ~ rule

- Contents of db must conform/follow/obey
- Condition that a solution to a problem must satisfy

- Data integrity:
  - Integrity constraints ensure that changes made to db don't result in a loss of data consistency
  - Guards against accidental damage to db & maintains correctness of db

Classification of Integrity constraints

1. Type Constraint
   - Legal values for given type
   - Ex: constraints which asserts minimum value for balance in an account ex. min_value = 1000

2. Attribute Constraint
   - Ensures that specified attribute is of a specified type
   - Is a part of the definition of attribute
   - Ex: ‘Account’ entity

Acct_no	Char(10)
Branch_Name	Char(50)
Balance	Integer

Classification of Integrity constraints

3. Relvar Constraint
   - Constraint on an individual relvar
   - Ex: suppliers in Kathmandu must have status 20

Supplier1	Pokhara	30
Supplier2	Butwal	10
Supplier3	Kathmandu	20

4. Database Constraint
   - Constraint that relates two or more distinct relvars
     - Ex: 'Customer' = cus_name, cus_add, cus_dob, acc_no
     - 'Account' = acc_no, branch_name, balance
     - Finding Customer names of Pokhara having balance > 5000

Classification of Integrity constraints

5. Domain Constraint
   - Specifies set of all possible values that may be associated with an attribute
   - May disallow use of null values for particular attributes
   - Ex: constraint that ensures the hourly wage is greater than 4.

6. Relationship Constraint
   - Relationship types usually have certain constraints that limit the possible combinations of entities that may participate in corresponding relationship set
   - Ex: if a company has a rule that each employee must work for exactly one department; then we would need to describe this in schema.
   - Following different types:

Classification of Integrity constraints

i. Mapping cardinalities (Cardinality ratio)
   - Specifies the maximum number of relationship instances that an entity can participate in.
   - Possible cardinality ratios for binary relationship types:
     - 1:1, 1:N, N:1, M:N

ii. Participation Constraints
   - Specifies whether the existence of an entity depends on its being related to another entity via the relationship type
   - Specifies minimum number of relationship instances that each entity can participate in & is sometimes called Minimum Cardinality Constraint.
   - Types:
     - a. Total b. Partial

Classification of Integrity constraints

a. Total

- If every entity participates in at least one relationship instance a.k.a. existence dependency
- Ex: if a company rule says every employee must involve in at least one works for relationship with department

b. Partial

- Some or part of set of employee entities are related to some department entity via manager but not necessarily all.

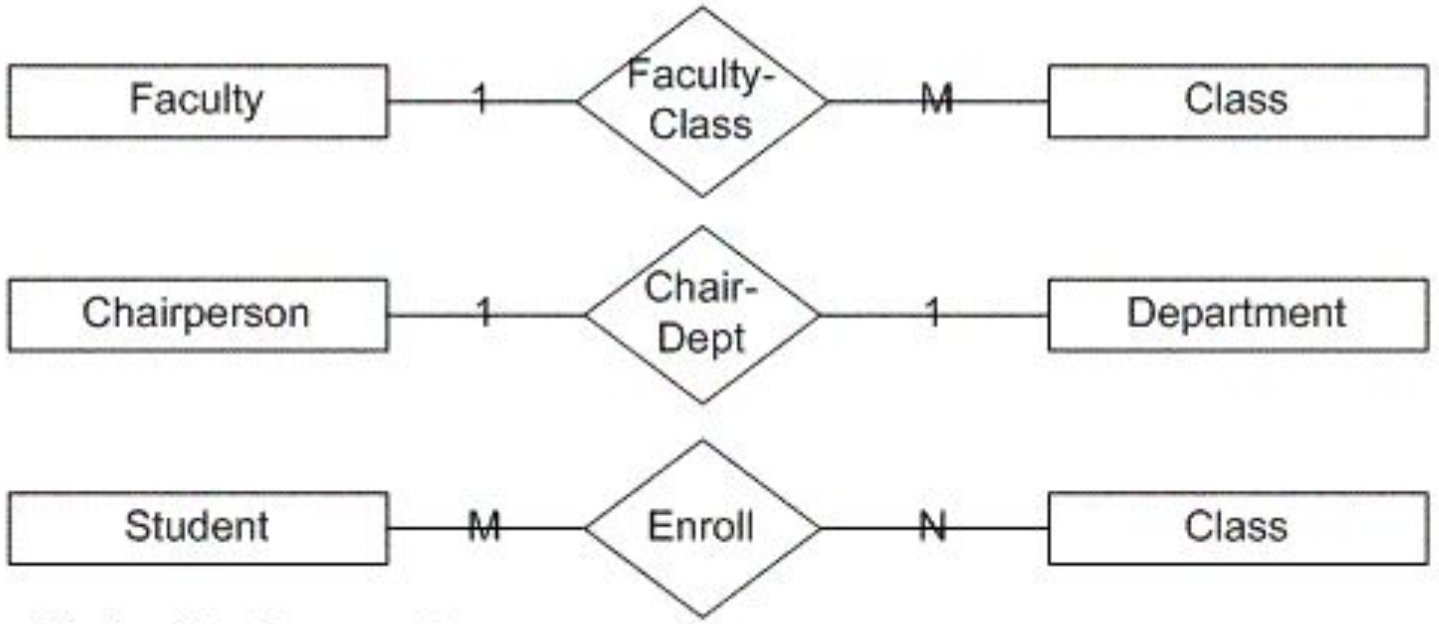
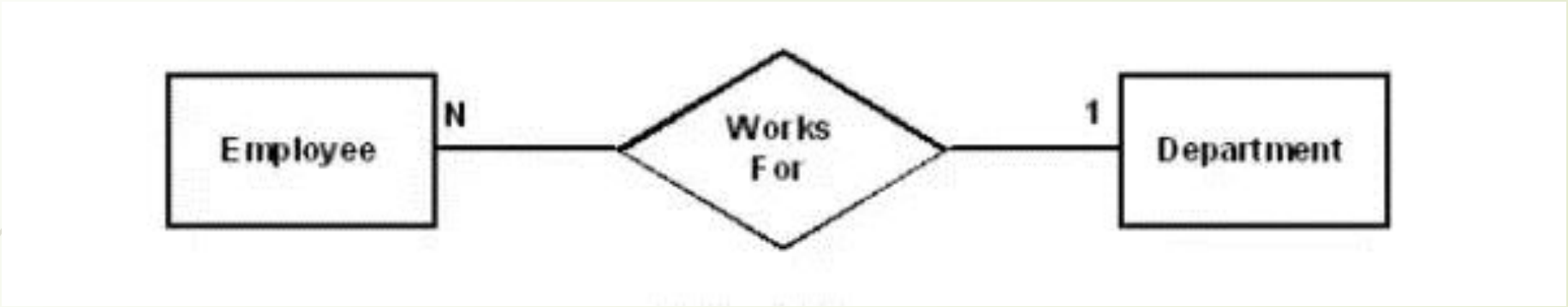


Figure : One to One relationship example

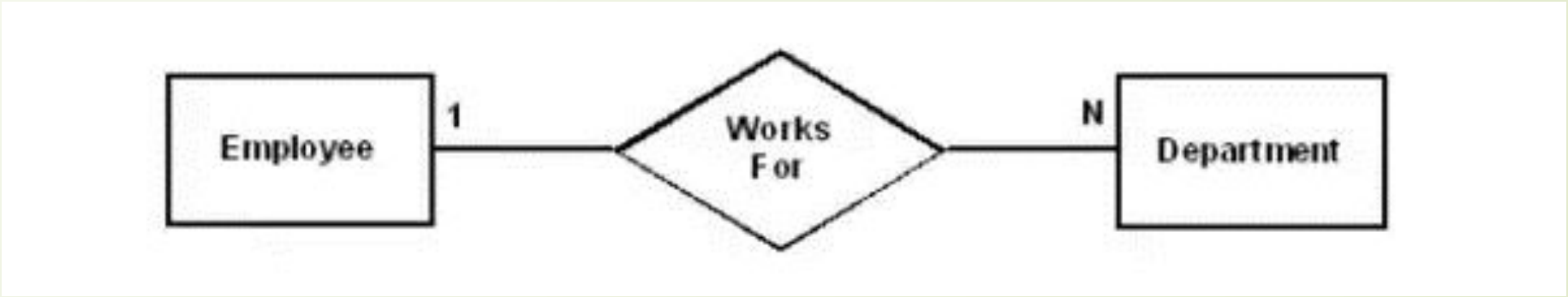
A one to may relationship - one manager manages many employees, but each employee only has one manager, so it is a one to many (1:n) relationship

Figure : One to Many relationship example



One department can have many employees.

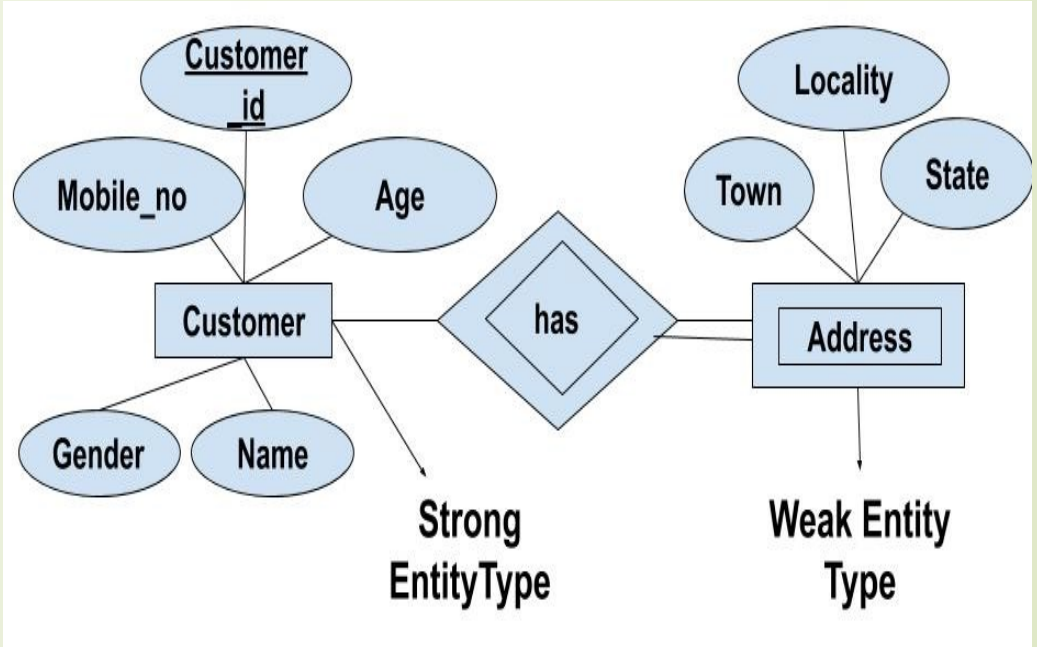
One employee works for many department.



Entity Types

a. Weak Entity type

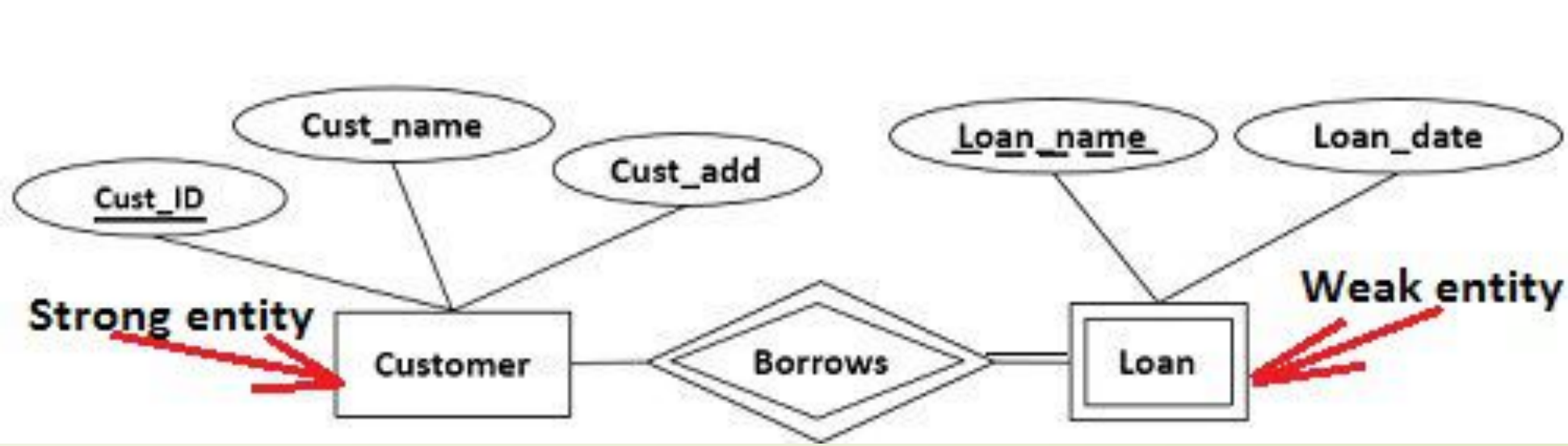
- Entities that do not have key attributes of their own
- An entity set may not have sufficient attributes to form a PK, such entity is termed as Weak Entity Set
- Entities of weak entity type are identified by being related to specific entities from another entity type (a.k.a. identifying/owner entity type) in combination with one of their attribute values.



Entity Types

b. Strong Entity type

- An entity set that has a PK
- Unique data or record in a table (no repetition of data)
- Easy to operate (retrieve, update)



Entity only diagram

A. Customer entity with cus_id, name (fname, mname, lname), address (city, state, zip, street (st_no, st_name, st_blockno), dob, age, phone_no)

Relationship between ER Model & UML Class diagrams

- UML (Unified Modeling Language) provides a graphical means of modeling various components of a software system.
- In software design, UML methodology is being used extensively and has many types of diagrams for various software design purposes
- Class diagram component of UML is based on ER diagram (alternative notation)
- Class is displayed as a box with class name at top section, attributes at middle section & operations/methods at last section.
- Operations are not defined in ER diagram

Relationship between ER Model & UML Class diagrams

- Relationship types are called Associations in UML & relationship instances are links.
- Min-Max notation / Cardinality Ratio is used to specify relationship constraints, which are called Multiplicities in UML
- UML specifies Aggregation & Association as Relationship but in ER Model they are represented as Relationship only.
- UML distinguishes between unidirectional and bidirectional associations but in ER Model no direction / arrow is specified.

Notations, ER diagram, Conceptual schema

Diagrams.

1. Customer deposits on Account
2. Doctor examines Patient
3. Doctor prescribes Medication
4. Librarian issues Book

Schema v/s Instances

Thank you!