Database Systems
C.Ranichandra

Module I

Introduction to databases

Database Environment

Database Architecture

Data Models

ER Modelling

EER Modelling

What is?

- Data : Facts (Actual Information)

  Example: Book (ISBN, title,...)

- Database : Collection of Related data

  Example: Library (Book, Member, Journal...)

- Database Management System:

  Collection of programs that enables users to create and maintain the database

  Example : Library Management system

What is ?

- Metadata

  One of the salient feature of DBMS is its self describing nature.

  → Data about data.

  → Description of the Db.

- Catalog

  Every DBMS software creates a description of the Db created in a file called catalog.

  → Contains Metadata.

Characteristics of DBMS

- Self Describing Nature
  - Metadata
- Data Abstraction
  - Program-data independence
  - Program-operation independence
- Multiple Views
  - Subset of DB
  - Authorized access
- Multiple Users(OLTP)

Advantages

- Controls redundancy
  - 17MSE students file in finance office, site office, coe office
- Restricts Unauthorized access
  - Login, password, roles
- Persistent Storage
  - Permanently stored like files
- Backup and Recovery
  - Facilities , log
- Multiple users and views
  - Concurrency control, different data different people
- Enforcing integrity constraints
  - Relationship between data and mandatory fields

Actors On the Scene

- DBA-
  - create , maintain DB, authorize access, monitor the use
- DB Designers-
  - identify the data to be stored and choose appropriate structures
- Software Engineers
  - System Analyst: develop specifications
  - Application Programmers: implement specifications

Users of DBMS

Casual→

occasionally query the Db
high level or middle level managers
use interactive query interface
Dean, Admission Director, team lead

Sophisticated →

have knowledge of Db
design the db and use it
scientists or engineers

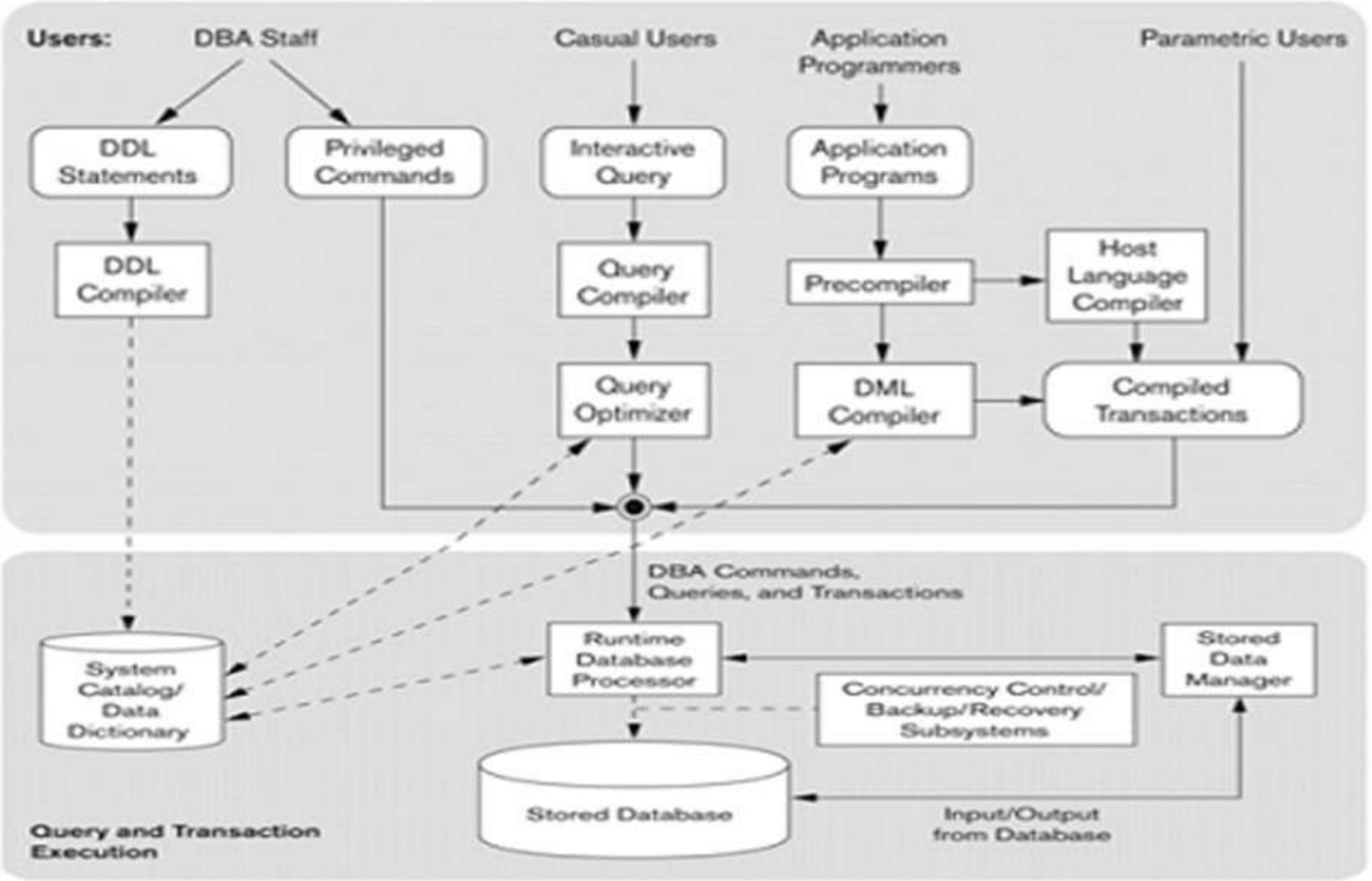
Naïve or parametric →

frequently query
Make transactions with exe file called canned transactions
bank clerk

Workers Behind the Scene

- DBMS designers and Implementers-
  - develop programs for full functionality of dbms
- Tool Developers-
  - develop tools for designing, monitoring or performance
- Operators and Maintenance Personnel-
  - Actual running of hardware and software.

Components(modules) of DBMS



Components cont...

- Query Compiler:
  - Checks and Coverts the query given by sophisticated user to set of operations.
- Query Optimizer:
  - Rewrites the query to properly use the access path
- Pre compiler:
  - Extracts the DML statements from host language and sends it to DML compiler
- DML Compiler:
  - Checks and converts DML statements to set of operations

Classification of DBMS

- Site :
  - Centralized ,Distributed, HDDBMS-Federated
- Data Model:
  - Network, Hierarchical , Relational, Object Relational, Object Oriented Relational, Big data system –column or document store
- File Organization:
  - Inverted file, BTree, Multilevel Indexing
- Purpose:
  - General, Specific
- Users:
  - Single, Multiple

Pros and Cons

- Easy to use when created
- Authentication, security can be provided
- Persistent Storage of data
- Enormous data structures involved
- Trained users are required
- Good Database designers are needed

Schemas and Instances

- Schema – Description of the database (Intension)
- Schema Construct – Description of an object
- Instance – State/snapshot of the database(Extension) [initial, current state(valid state)

COURSES				
Code	Title	Credits	type	School
PMCA503L	Database Systems	3	TH	SCORE
BITE302L	DBMS	3	TH	SCORE
BITE302P	DBMS Lab	1	LA	SCORE

Data Model

- What is a Data Model?
- Categories of Data Model
- Architecture and Independence

Data Model:

Collection of concepts that can be used to describe the structure of the database like data types, constraints and Relationships.

In addition, basic operations for retrieval and updates on the database can also be included.

Categories of Data Model

- High Level or Conceptual Model:

  Provides concepts such as Entities, Attributes and Relationships that are as users perceive data.
  Ex: ER Model, EER Model, UML Model

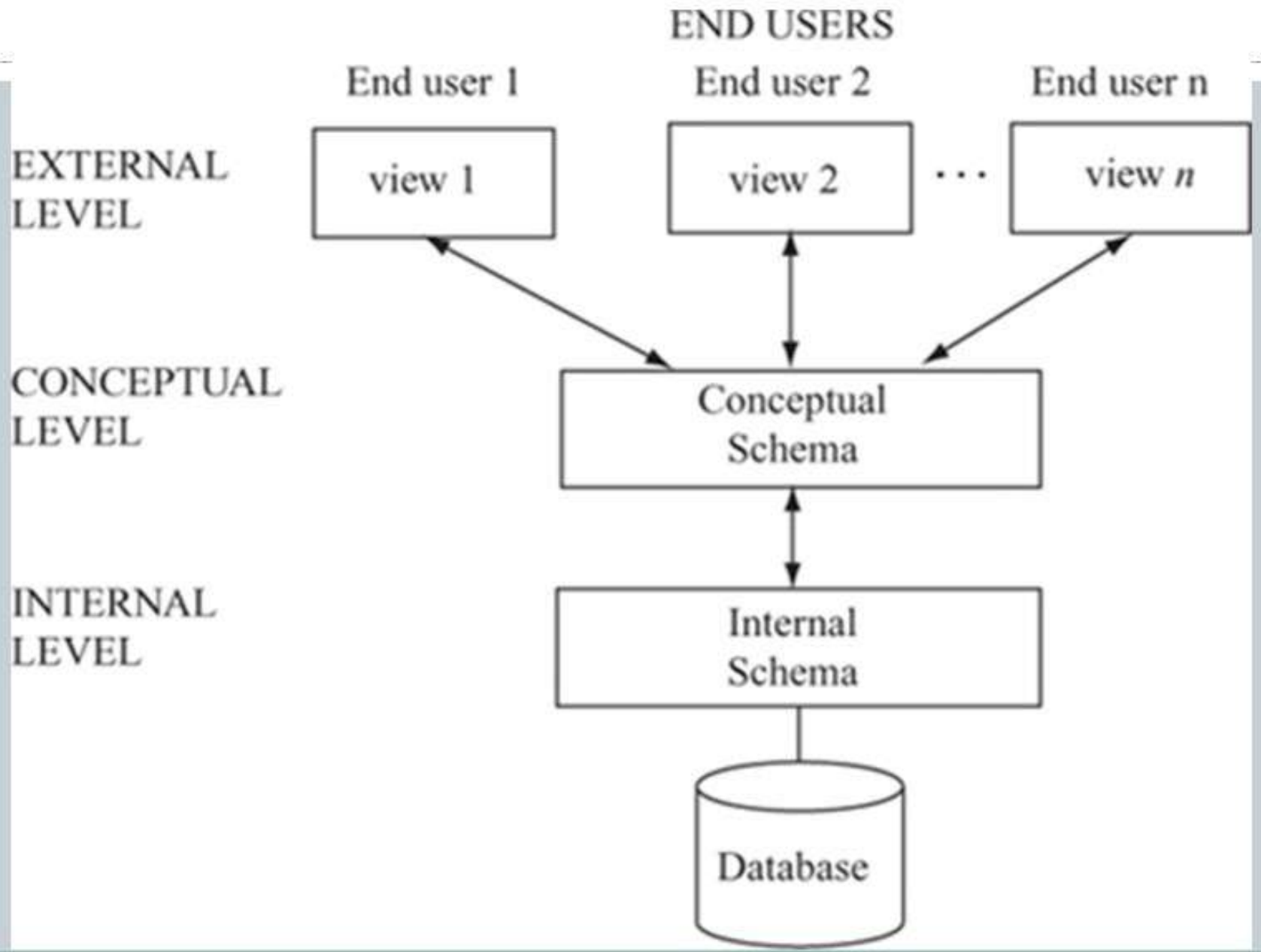
- Low Level or Physical Model:
  - Provides concepts that describes the details of data storage in computer.
  - Ex: Index- Btree, Hash[no standard Model]

- Representational or Implementation Model:
  - Provides concepts that are understood by end users and also the way data is organized in the system.
    Ex: Network, Hierarchical, Relational Models

DBMS Architecture

- The overall database description can be defined at three levels, namely, internal, conceptual, and external levels and thus, named **three-level DBMS architecture or 3 Schema architecture**.
- This architecture is proposed by ANSI/SPARC (American National Standards Institute/ Standards Planning and Requirements Committee) and hence, is also known as **ANSI/SPARC architecture**.

3 Schema Architecture



Architecture

- **Internal level**: deals with the physical representation of the database on the computer and thus, is also known as physical level. It describes how the data is physically stored and organized on the storage medium. Designed by physical model.

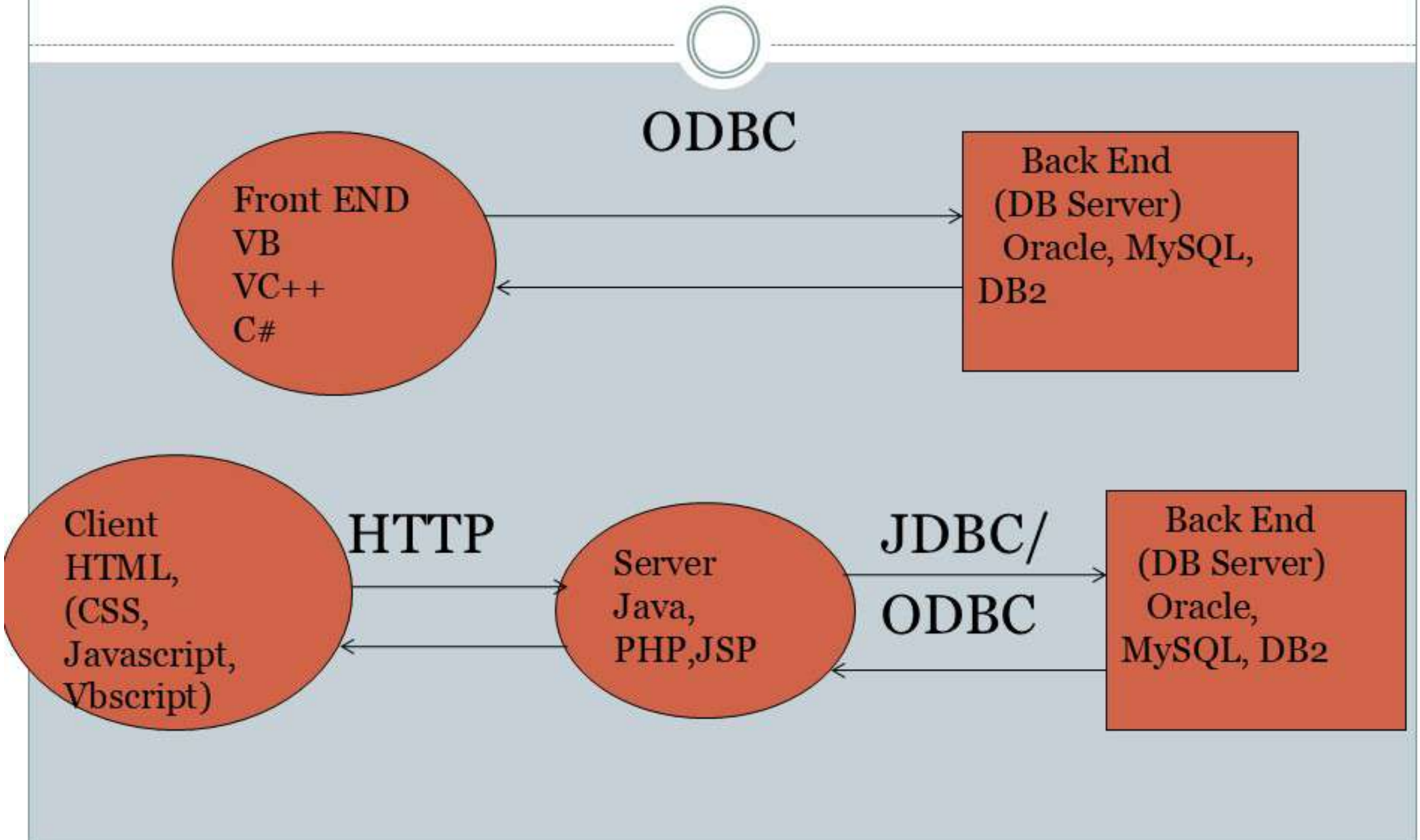
- **Conceptual level**: deals with the logical structure of the entire database and thus, is also known as logical level. It describes what data is stored in the database. The schema is designed by implementational model.

- **External level**: It is the highest level of abstraction that deals with the user's view of the database and thus, is also known as view level. The external level describes a part of the database for a particular group of users. The schema is designed by implementational model.

Data Independence

- Mapping- passing request/reply b/w levels.
- Logical Data Independence- Changing the conceptual schema without having to change the external schema
- Physical Data Independence- Changing the physical schema without having to change the conceptual

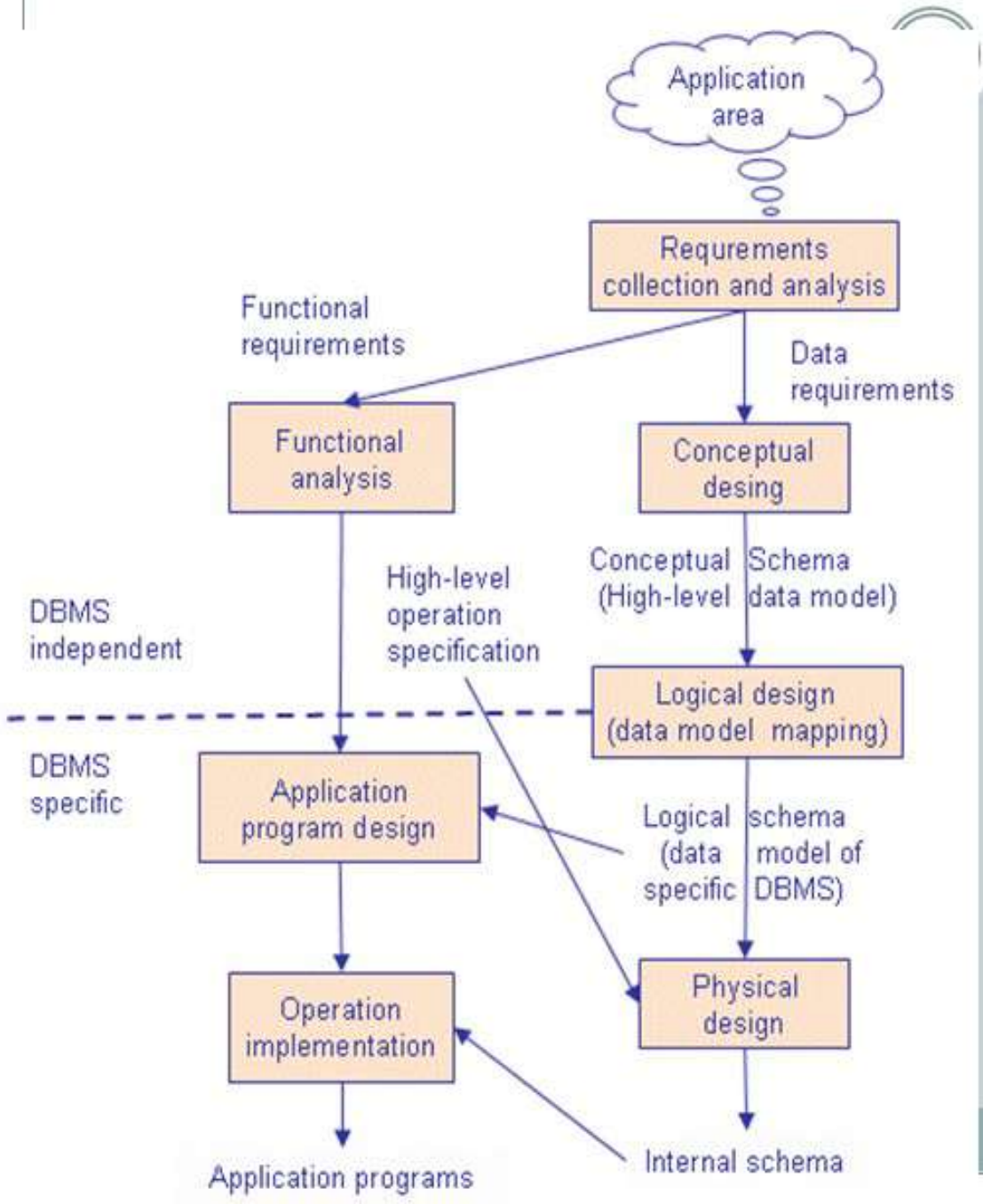
A DB Application



Conceptual Design

- ER
- EER

Design Phases



Collect and Analyze Data

Identify Entity and relationship ↓

Draw an ER Diagram

↓

Map to Relational using the rules ↓

Draw a Relational Schema

Identify the workload of the DB ↓

List the indexes required

Conceptual Design- ER Model
Entity Relationship Model

- Entity : 'Thing' in the real world or Object with a physical existence.

  Example: a Student, a Car
- Attributes: Properties that describe an entity.

  Example: Student (Register number, Name, Age)
- Entity Type: Set of entities with same attributes

  Example: Student
  - Entity types that have a key attribute are called strong entity types
  - Entity types that do not have a key attribute are called weak entity types
- Entity Set: Group of entities of similar type

  Example: Ten students

- Strong Entity

- Weak Entity

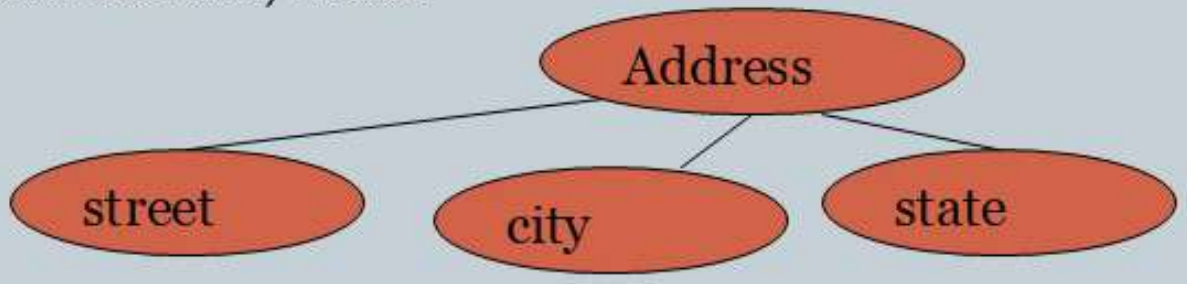
- Types of Attributes:

1. Single valued – Attribute has one value for an entity.
   Example: Register number

2. Multi valued - Attribute has more than one value for an entity.
   Example: Degree

3. Simple – Attributes that are not divisible
   Example: Age

4. Composite - Attributes that are divisible
   Example: Address, Name



Types of Attributes

5. Stored – Attribute Whose value must be available in the database.

Example: Date of birth

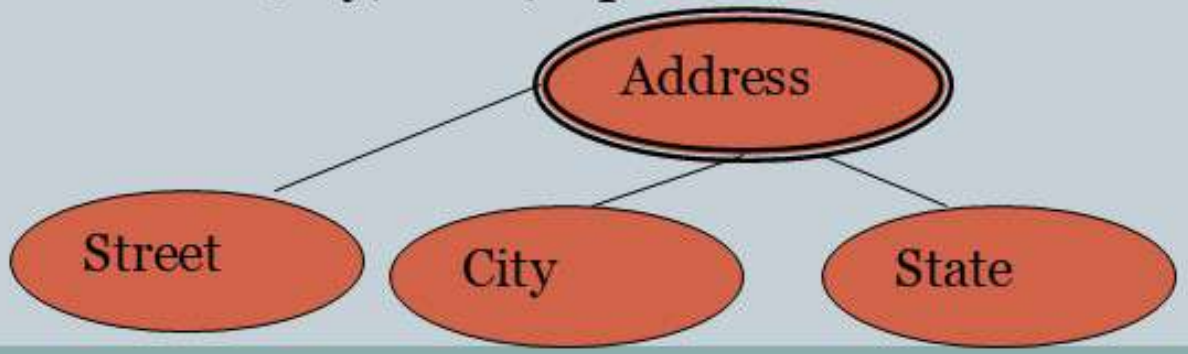
6. Derived – Attribute whose value can be calculated from stored attribute values.

Example: Age = Sysdate – Date of birth

7. Complex – Combination of Composite and Multi valued attribute

Example: Permanent and or temporary address

Address (city, state, zip code)



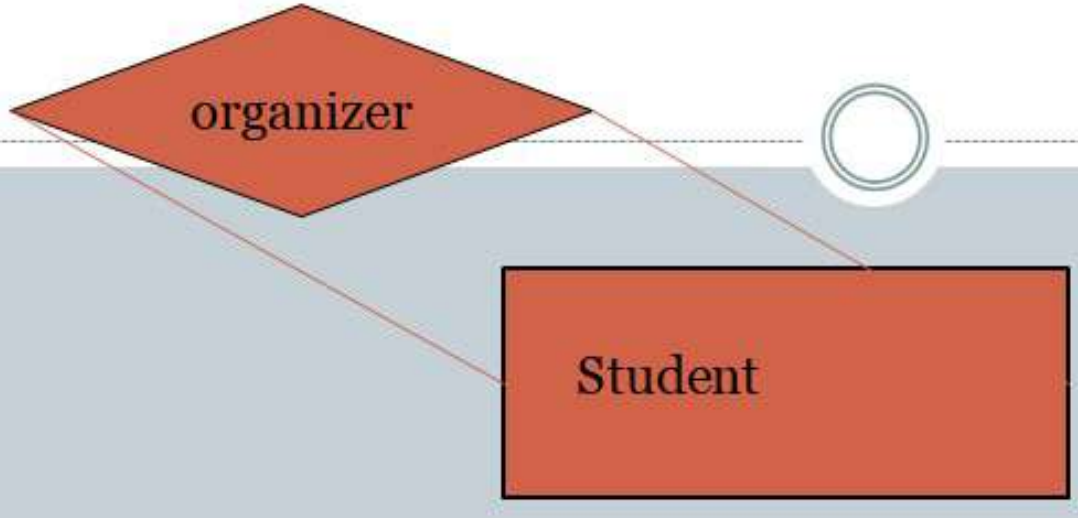
- Relationship type: Set of associations among different entity types

  Example: “belongs to” is a Relationship type between a Student and a Department

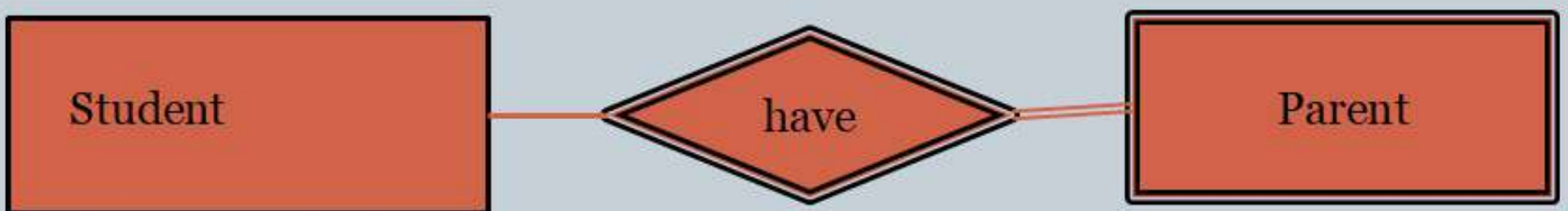
- Degree of a Relationship type: number of participating Entity types.

  Binary, Ternary

- Recursive Relationship: Same entity type participating more than once in a relationship type in different roles



Weak Entity: Depends on strong, donot have Key attribute, relation ship -identifying



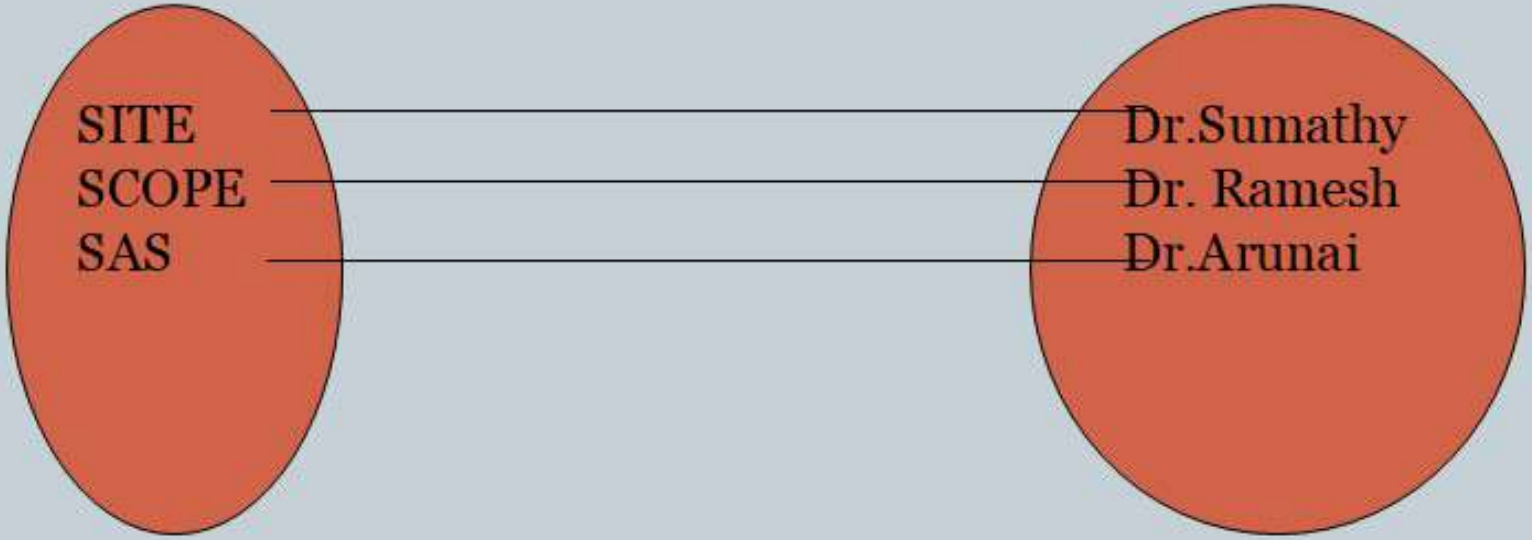
- Structural constraints :

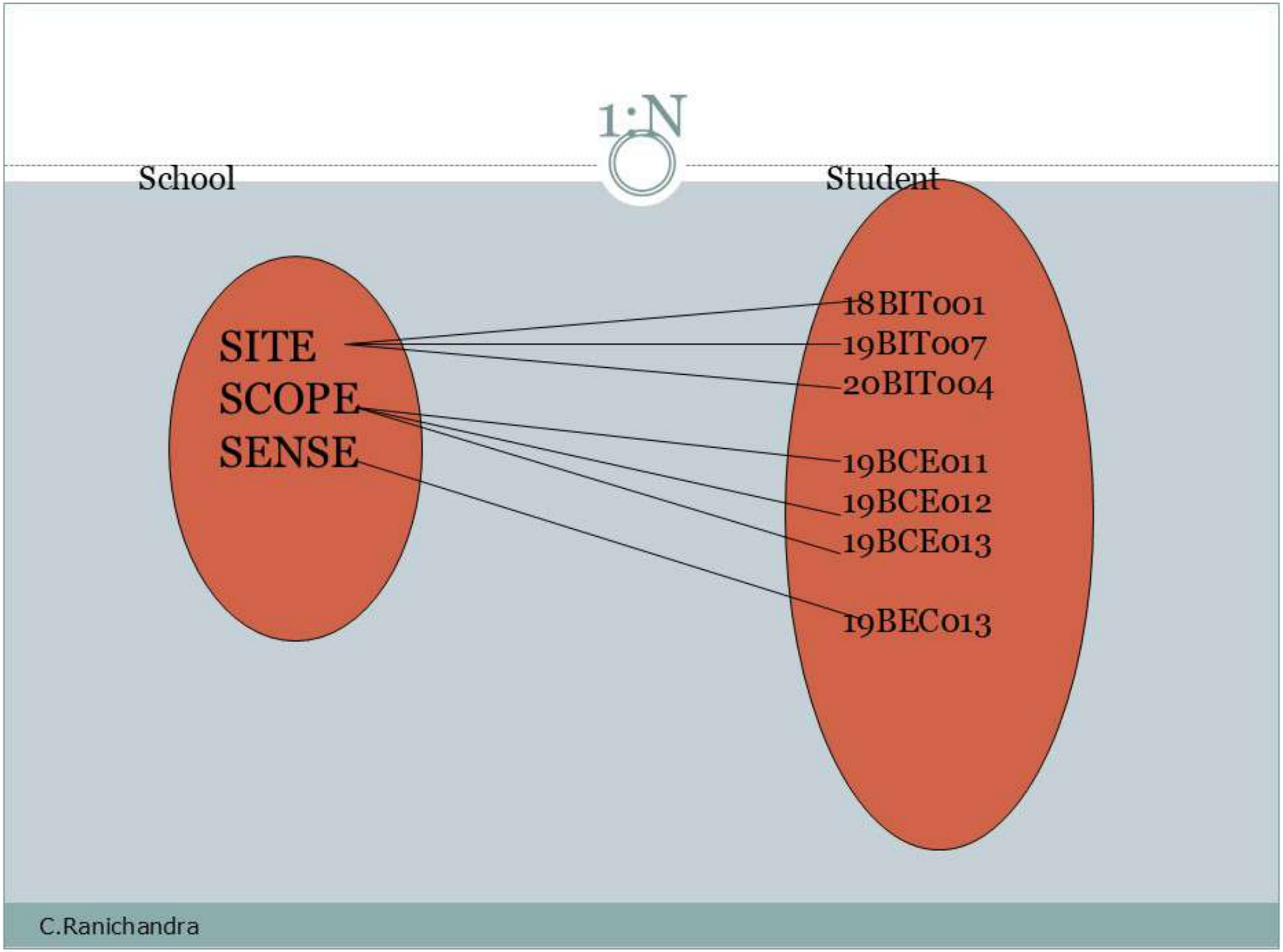
- Cardinality ratio: Number of relationship instances an entity can participate in a relationship.

Ratios for binary relationship types are 1:1, 1:N, M:N

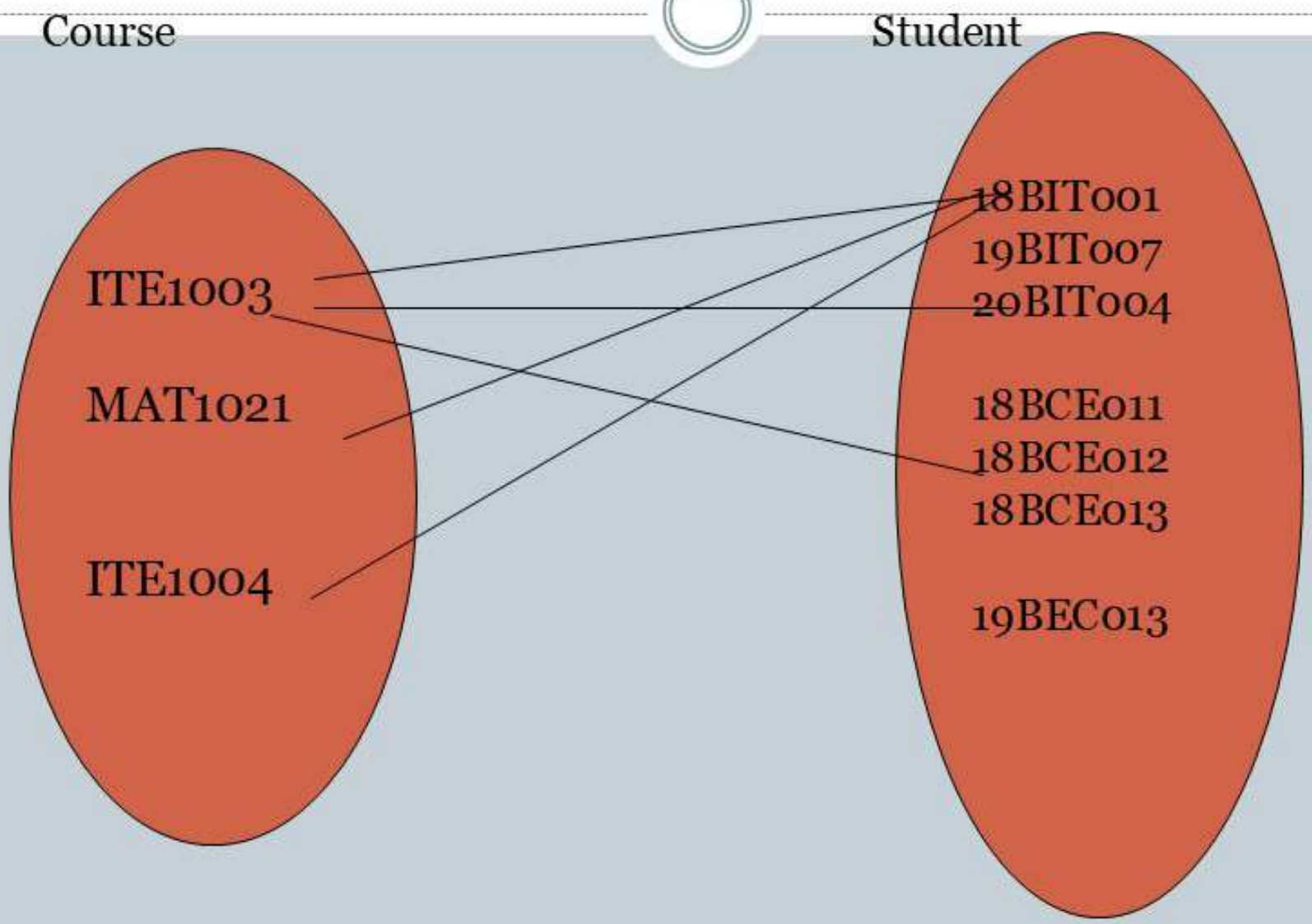
1:1

- School Dean




1:N
School
Student
SITE
SCOPE
SENSE
18BIT001
19BIT007
20BIT004
19BCE011
19BCE012
19BCE013
19BEC013

M:N


Course
Student
ITE1003
MAT1021
ITE1004
18BIT001
19BIT007
20BIT004
18BCE011
18BCE012
18BCE013
19BEC013

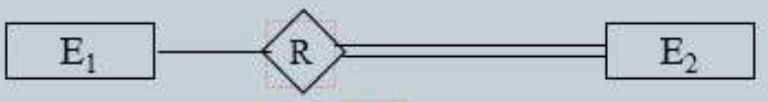
Participation constraints:

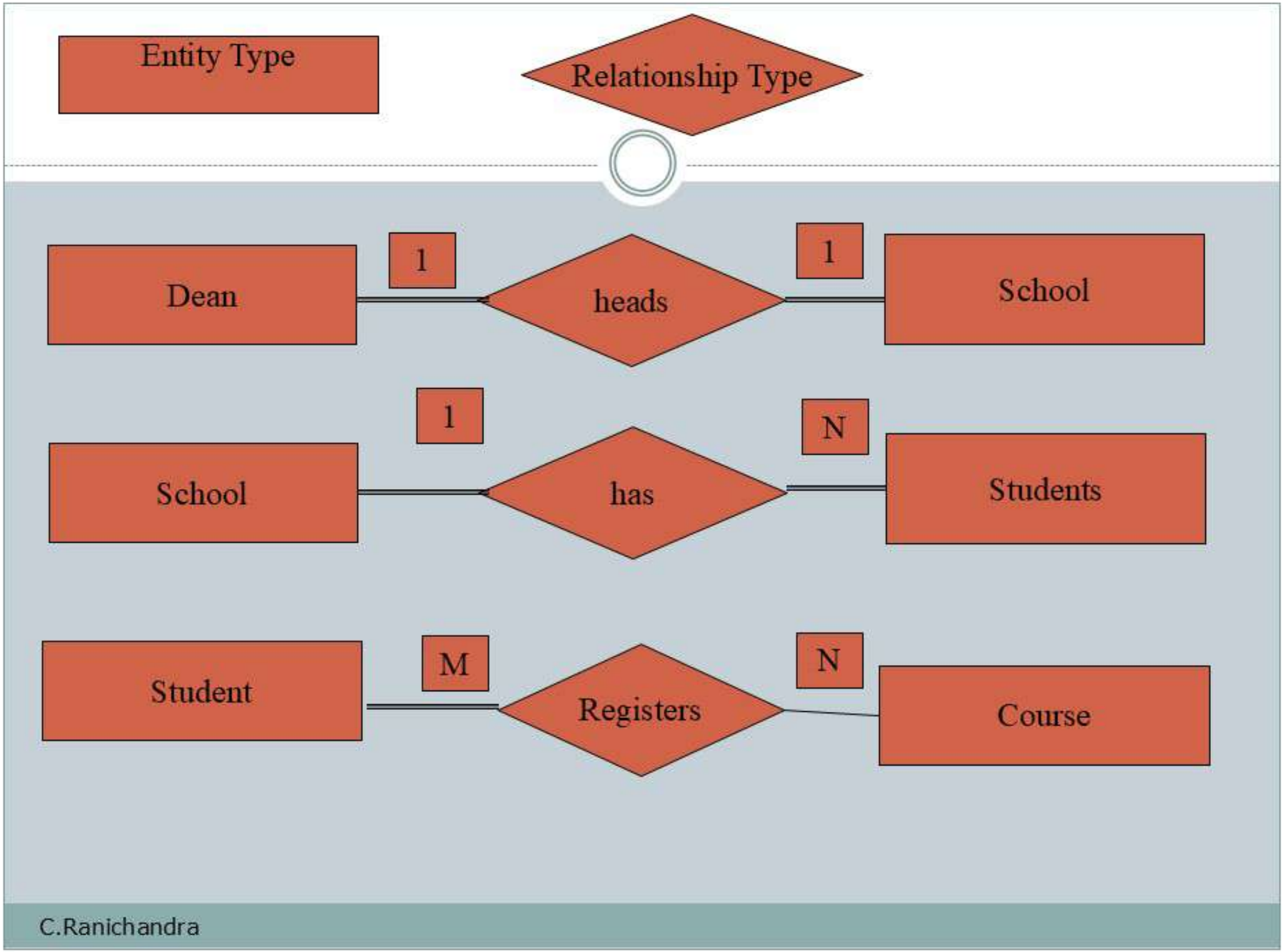
Participation constraints:

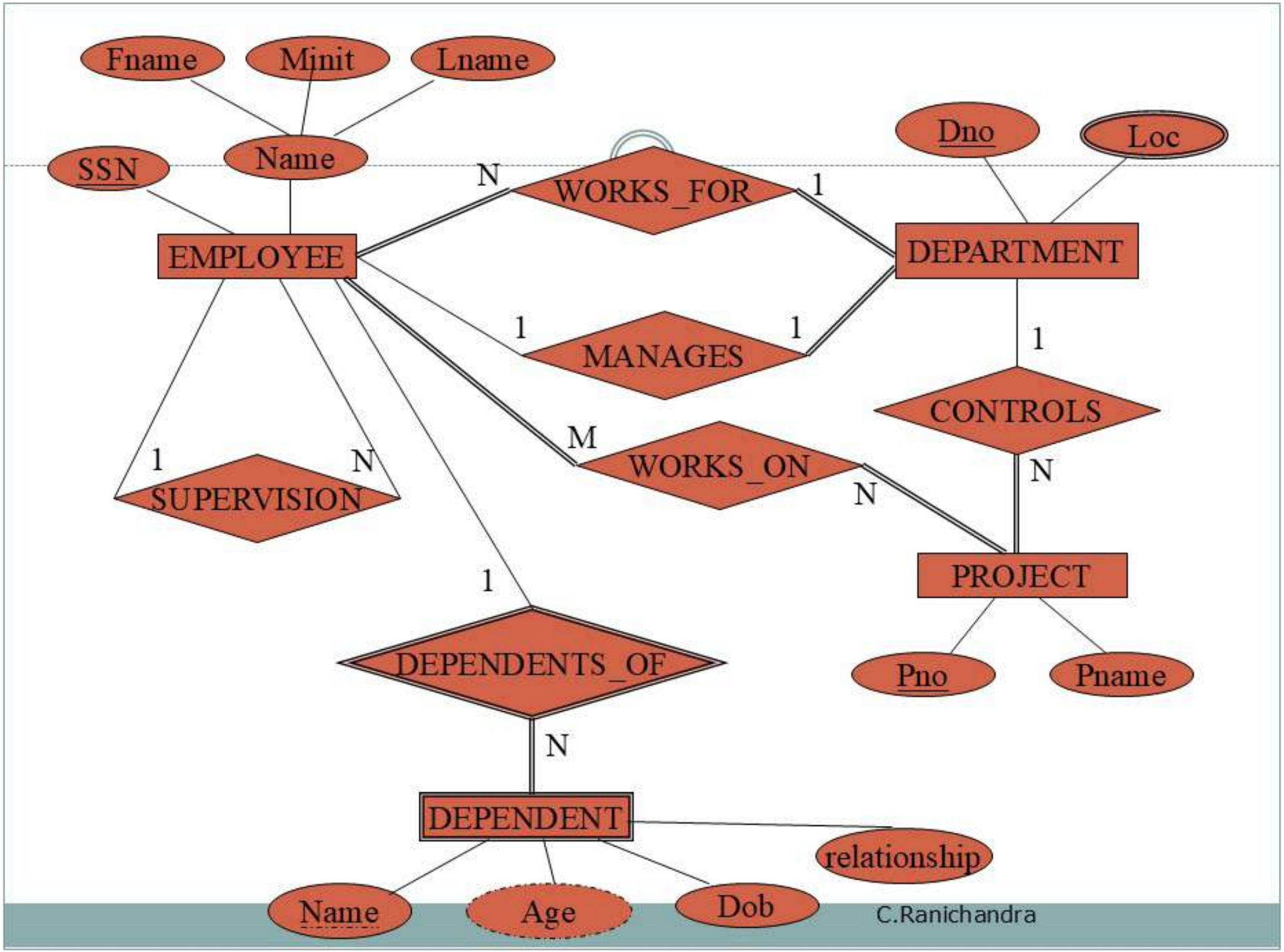
Total: Every entity of an entity type participates in the relationship type.

Partial: Only few entity of an entity type participates in the relationship type.

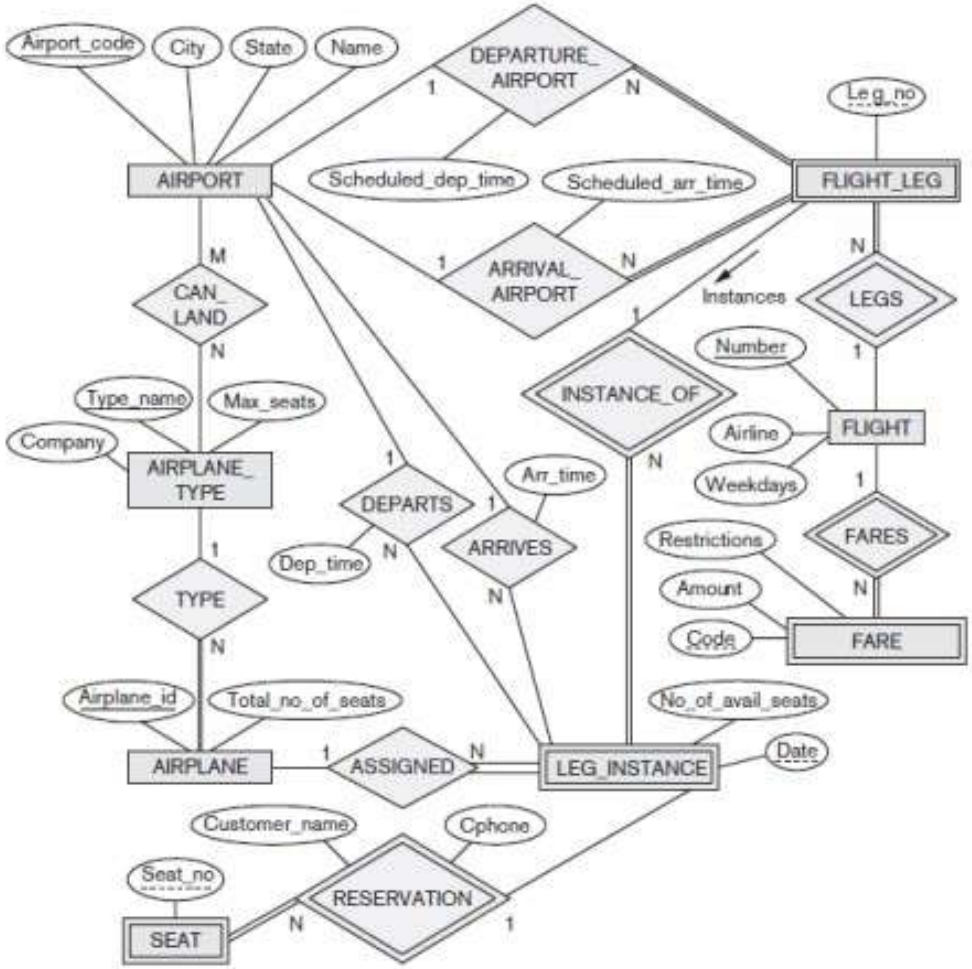
Designing an ER Model

Symbol	Meaning
	ENTITY TYPE
	WEAK ENTITY TYPE
	RELATIONSHIP TYPE
	IDENTIFYING RELATIONSHIP TYPE
	ATTRIBUTE
	KEY ATTRIBUTE
	MULTIVALUED ATTRIBUTE
	COMPOSITE ATTRIBUTE
	DERIVED ATTRIBUTE
	TOTAL PARTICIPATION OF E2 IN R
	CARDINALITY RATIO 1:N FOR E1:E2 IN R


Entity Type
Relationship Type
Dean
1
heads
1
School
School
1
has
N
Students
Student
M
Registers
N
Course


Fname
Minit
Lname
Name
SSN
EMPLOYEE
N
WORKS_FOR
1
Dno
Loc
DEPARTMENT
1
MANAGES
1
1
CONTROLS
1
N
M
WORKS_ON
N
1
SUPERVISION
N
PROJECT
1
DEPENDENTS_OF
N
Pno
Pname
DEPENDENT
relationship
Name
Age
Dob
C.Ranichandra

- Design an ER diagram for an airline reservation system.
- The database represents each Airport, keeping its unique Airport Code, the Airport Name, and the City and State in which the Airport is located.
- Each airline flight has a unique number, the Airline for the flight, and the Weekdays on which the flight is scheduled. The fare information of each flight differs and details like code, amount and restrictions must be recorded.
- A flight is composed of one or more flight legs. Each flight leg has a leg number, Departure airport and Scheduled Departure Time, and an arrival airport and Scheduled Arrival Time.
- A leg instance is an instance of a flight leg on a specific date. The Number of available seats and the airplane used in the leg instance are also kept.
- The customer reservations on each leg instance include the customer name, phone, and seat numbers for each reservation.
- Information on Airplanes and Airplane Types are also kept. For each Airplane type the Type Name, manufacturing company, and maximum number of seats are kept. The Airports in which planes of this type Can Land are kept in the database. For each Airplane, the Airplane Id, Total number of seats, and Type are kept.
-

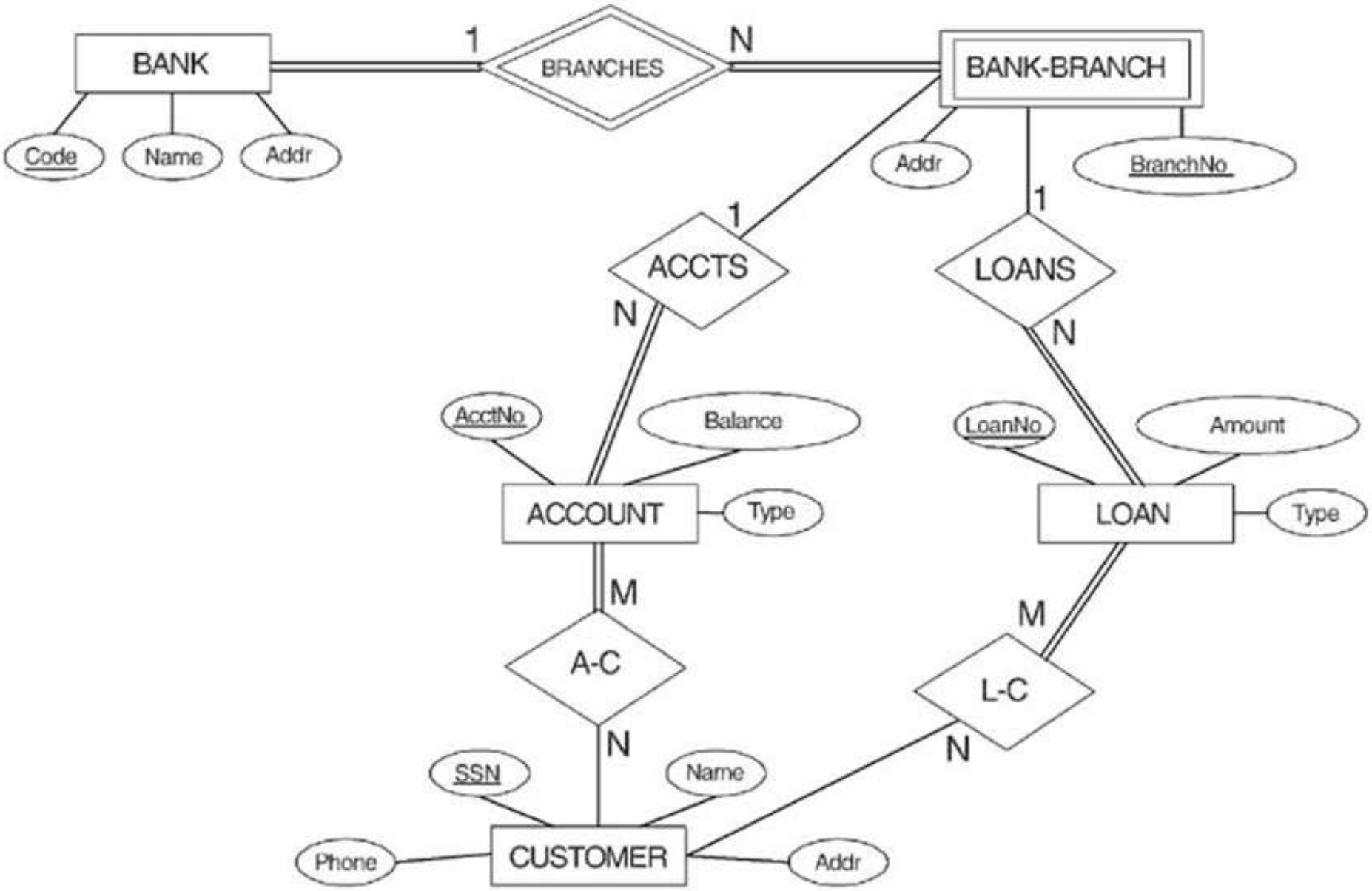

Airport_code
City
State
Name
DEPARTURE_ AIRPORT
1
N
Leg_no
AIRPORT
Scheduled_dep_time
Scheduled_arr_time
FLIGHT_LEG
1
ARRIVAL_ AIRPORT
N
M
CAN_ LAND
Instances
N
LEGS
1
N
Number
INSTANCE_OF
Type_name
Max_seats
Company
AIRPLANE_ TYPE
1
DEPARTS
Arr_time
1
Airline
FLIGHT
N
Weekdays
1
Dep_time
N
ARRIVES
Restrictions
FARES
1
N
TYPE
Amount
N
Code
FARE
Airplane_id
Total_no_of_seats
No_of_avail_seats
AIRPLANE
1
ASSIGNED
N
LEG_INSTANCE
Date
Customer_name
Cphone
Seat_no
RESERVATION
SEAT
N
1

ER- Exercise

Patty's Playschool is a child daycare center. A parent registers their child or children at the school using a registration form. A parent can submit more than one registration form. Each room in the daycare is assigned an age group. For example an infant is under 1 year of age and toddlers are from 1 to 3 years of age. A child is assigned to a room based on their age and availability of space. A room may be assigned one or more employees. An employee can only be assigned to one room. The minimum number of employees required for a room is determined by the number of children assigned to the room and the child:staff ratio identified by the government. For example one employee can care for 5 infants or 8 toddlers.

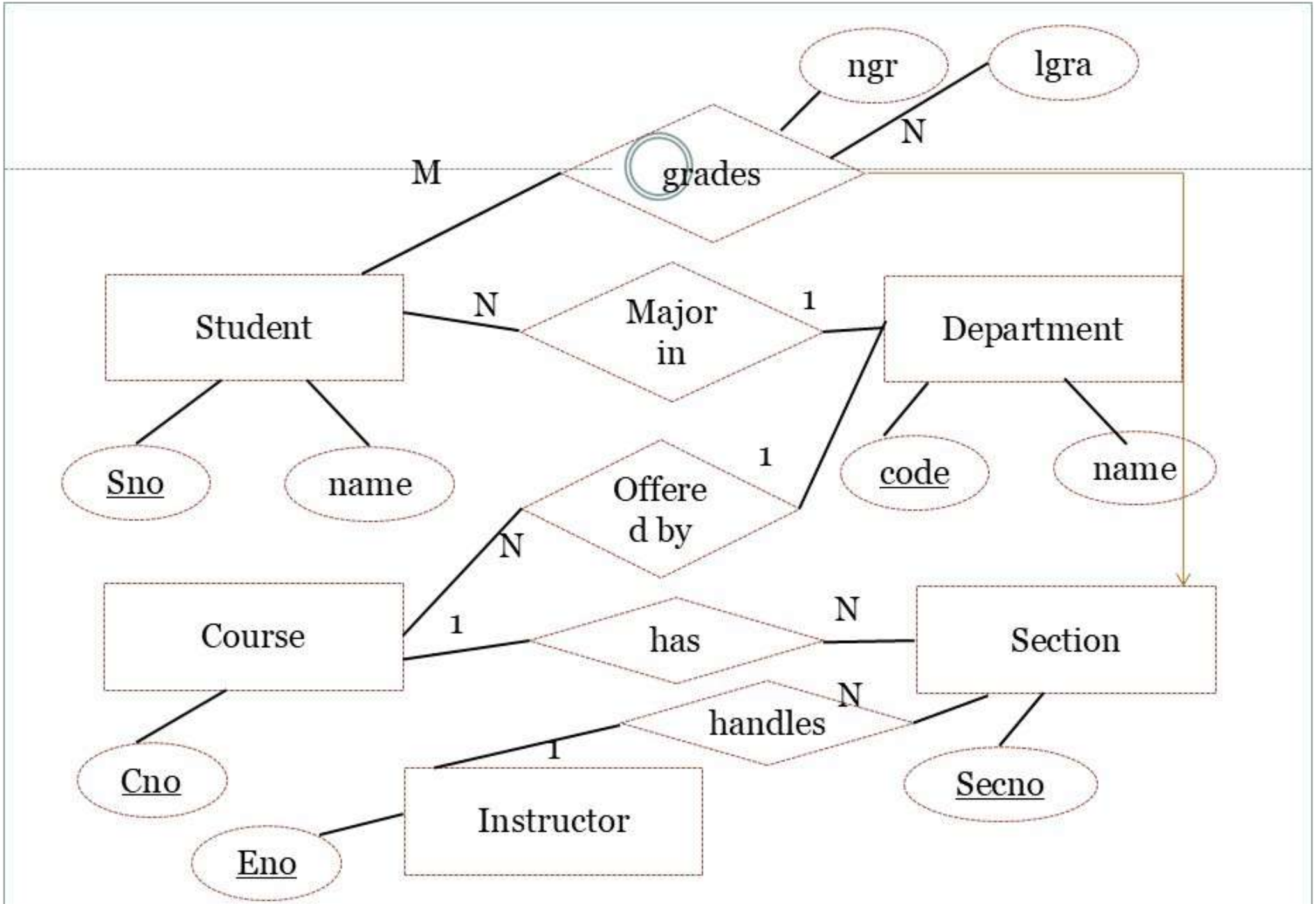
1. A bank has code, name and address (of main)
2. Each bank has many branches (branch no repeats) and the address has to be maintained
3. Each branch offers account and loan where no, type, balance are major attributes. A branch may offer either of or both.
4. A customer is a citizen of country identified by panno or Aadhar card no, his name, address and mobile
5. A customer can hold any number of account or loan.
6. Loan/Account can be joint loan/account i.e held by many customers

ER Diagram(schema) –Bank Database



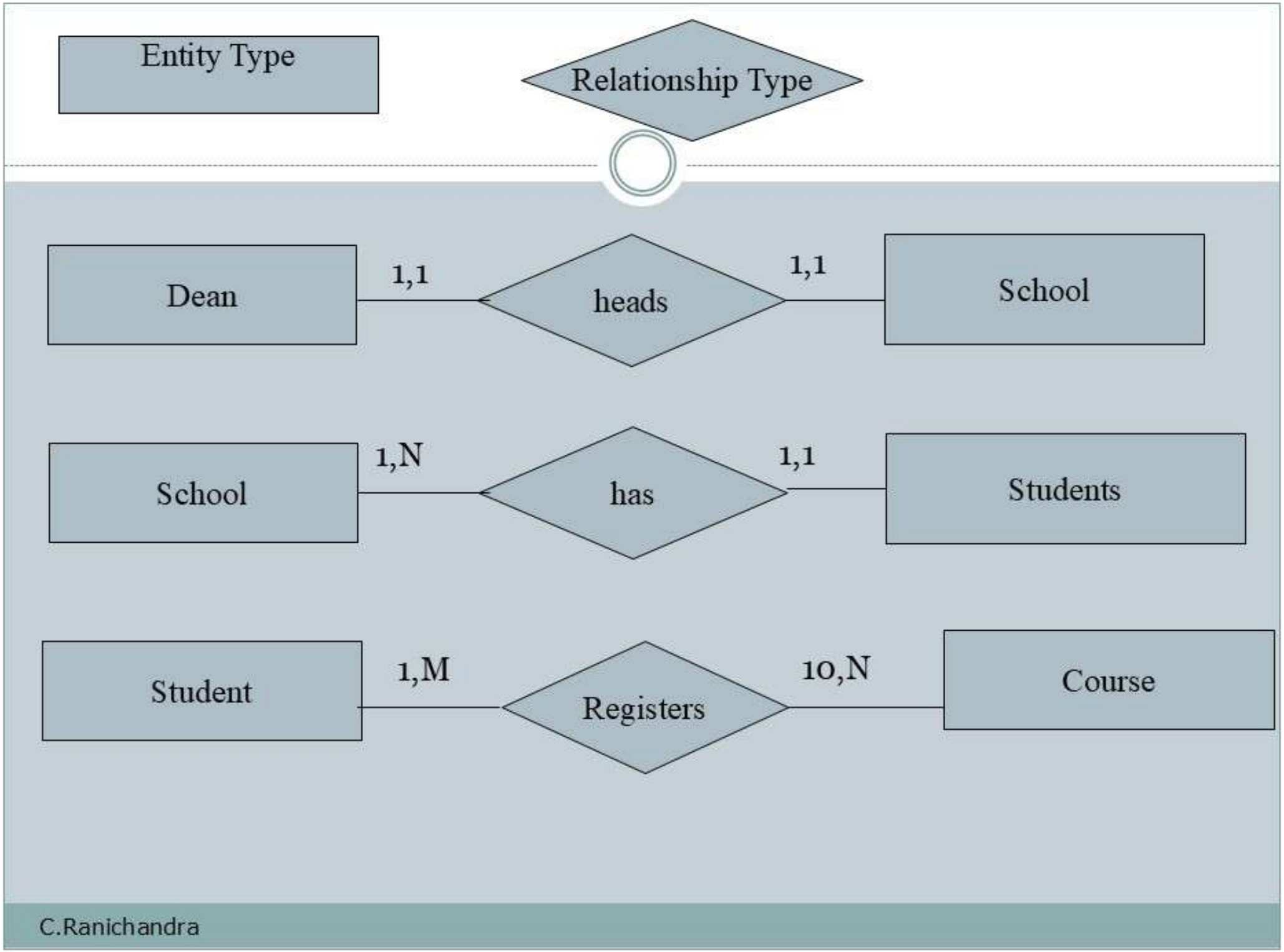
Consider the following set of requirements for a University database. Design an ER diagram for this application:

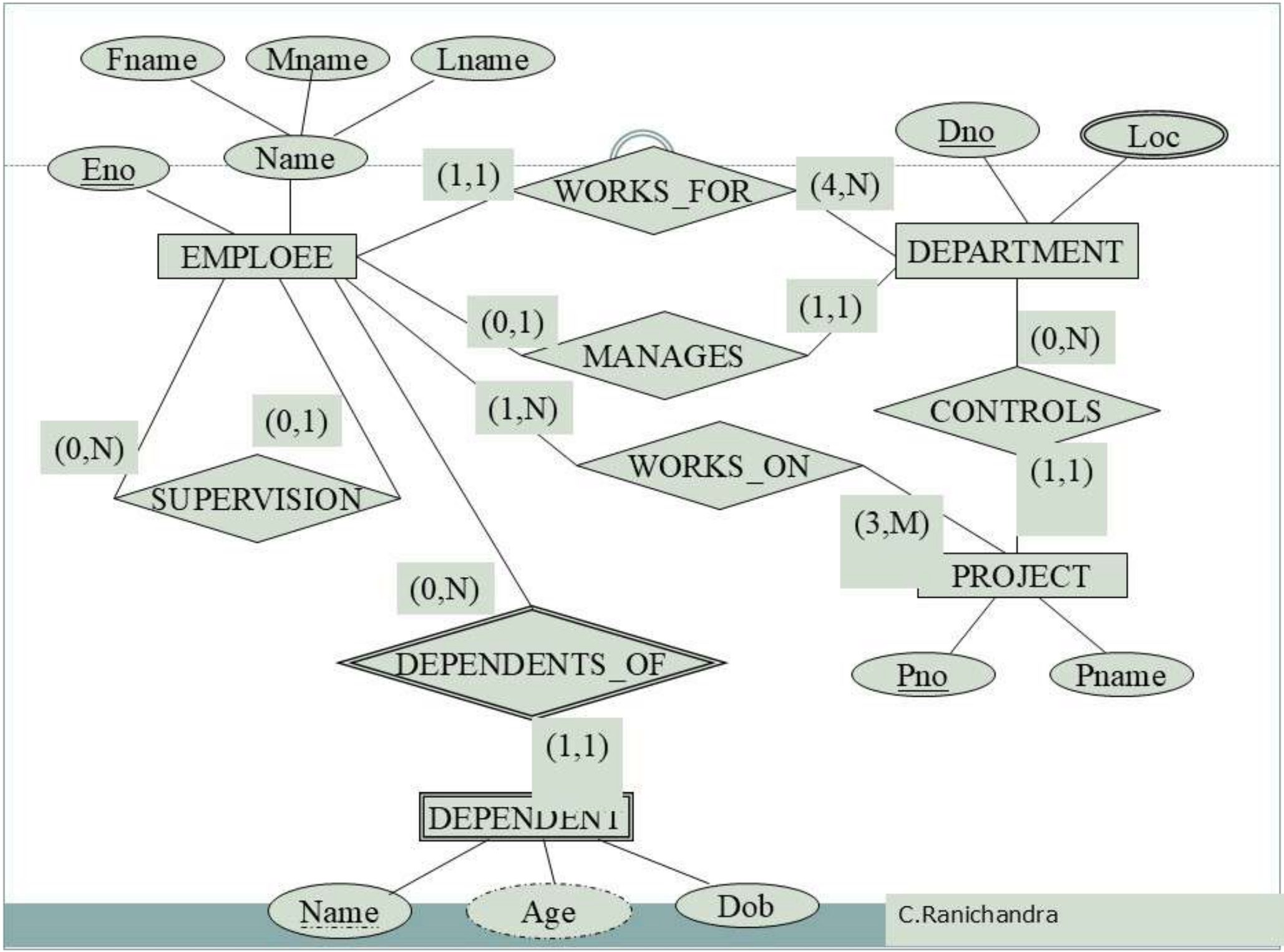
- The university keeps track of each student's name, student number, social security number, current address and phone number, permanent address and phone number, birthdate, sex, class (freshman, graduate), major department, minor department (if any), degree program (B.A., B.S., ... Ph.D.). Some user applications need to refer to the city, state, and zip code of the student's permanent address and to the student's last name. Both social security number and student number are unique for each student. All students will have at least a major department.
- Each department is described by a name, department code, office number, office phone, and college. Both the name and code have unique values for each department.
- Each course has a course name, description, course number, number of credits, level and offering department. The course number is unique for each course.
- Each section has an instructor, semester, year, course, and section number. The section number distinguishes sections of the same course that are taught during the same semester/year; its value is an integer (1, 2, 3, ... up to the number of sections taught during each semester).
- A grade report must be generated for each student that lists the section, letter grade, and numeric grade (0,1,2,3, or 4) for each student and calculates his or her average GPA.


ngr
lgra
N
grades
M
Student
N
Major
in
1
Department
Sno
name
1
code
name
Offere
d by
N
Course
1
has
N
Section
handles
N
1
Cno
Instructor
Secno
Eno

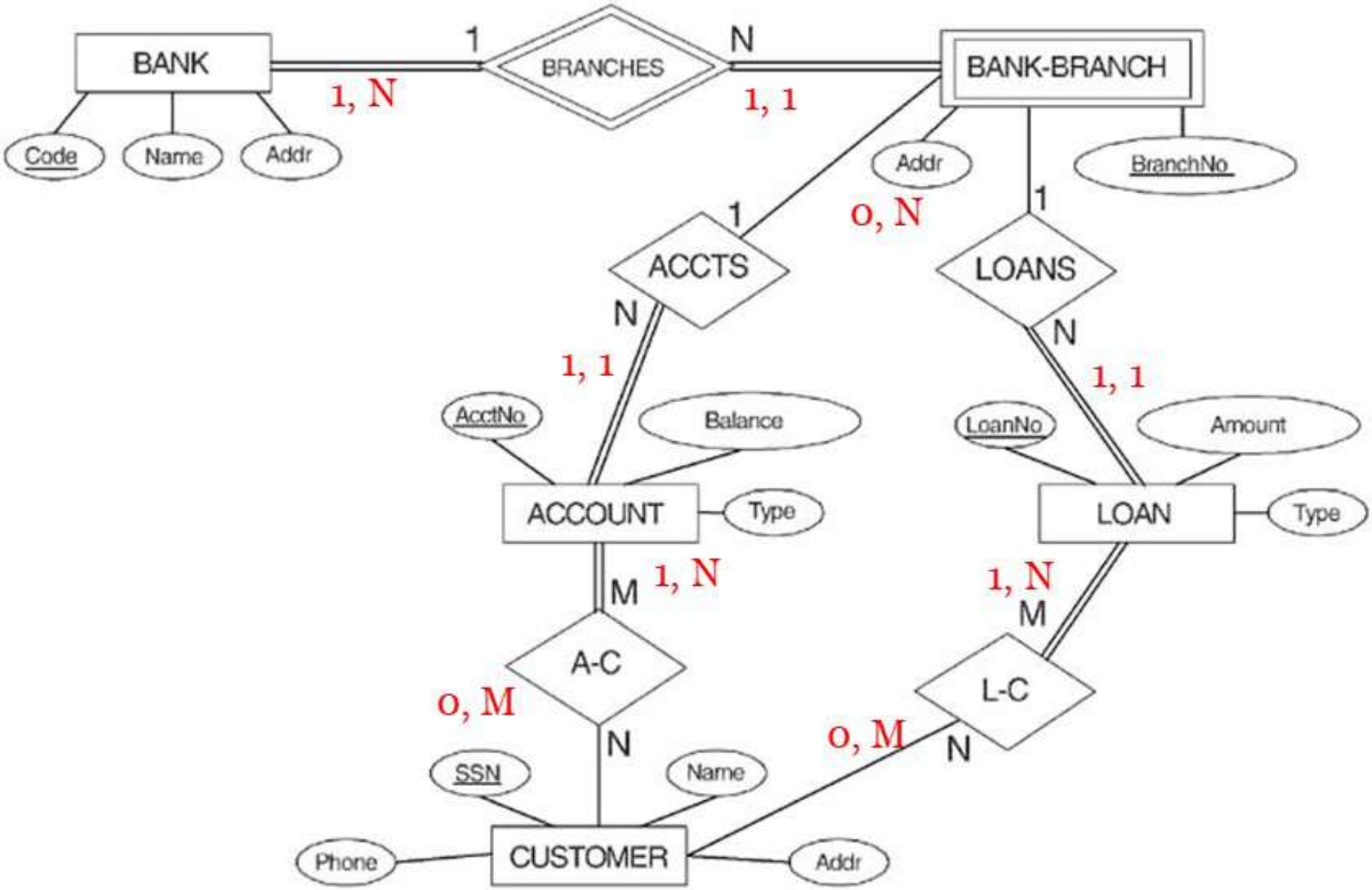
- Min Max Constraints

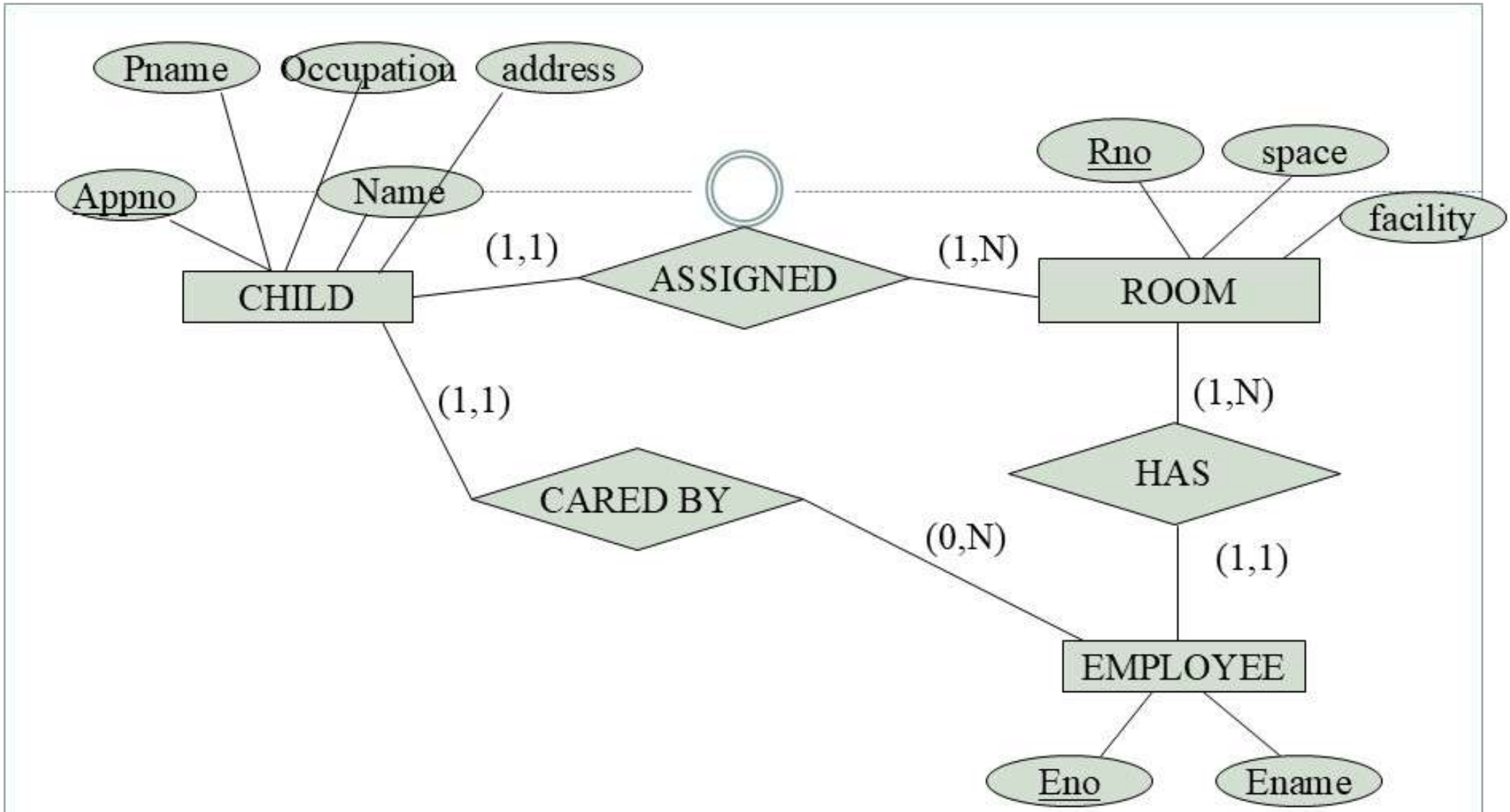
- (min,max)
- An entity is related to min and max how many entity in the relationship
- 1:N ratio
  - On 1 side (1,N)
  - On N side (1,1)
  - Max represents the ratio
  - Min represents the participation
  - If ‘0’ its partial participation
  - If 1 or more its total participation


Entity Type
Relationship Type
Dean
1,1
heads
1,1
School
School
1,N
has
1,1
Students
Student
1,M
Registers
10,N
Course


Fname
Mname
Lname
Name
Eno
EMPLOEE
(1,1)
WORKS_FOR
(4,N)
Dno
Loc
DEPARTMENT
(1,1)
(0,1)
MANAGES
(0,N)
CONTROLS
(1,1)
(1,N)
WORKS_ON
(3,M)
PROJECT
Pno
Pname
(0,N)
SUPERVISION
(0,1)
(0,N)
DEPENDENTS_OF
(1,1)
DEPENDENT
Name
Age
Dob

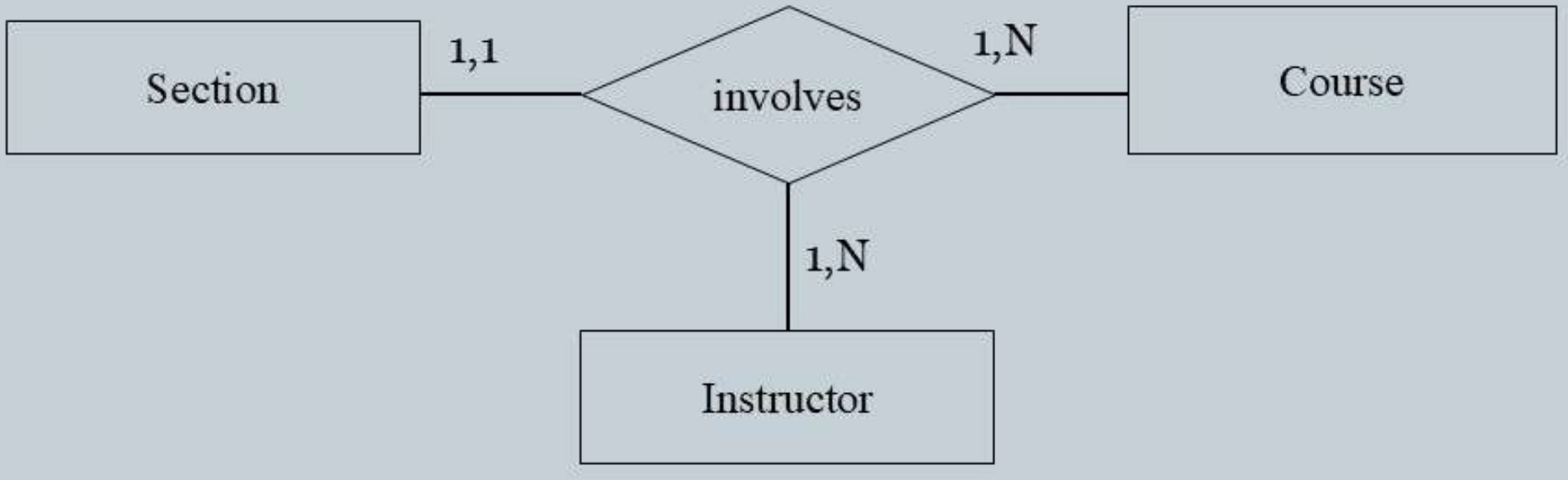
ER Diagram –Bank Database – Min Max

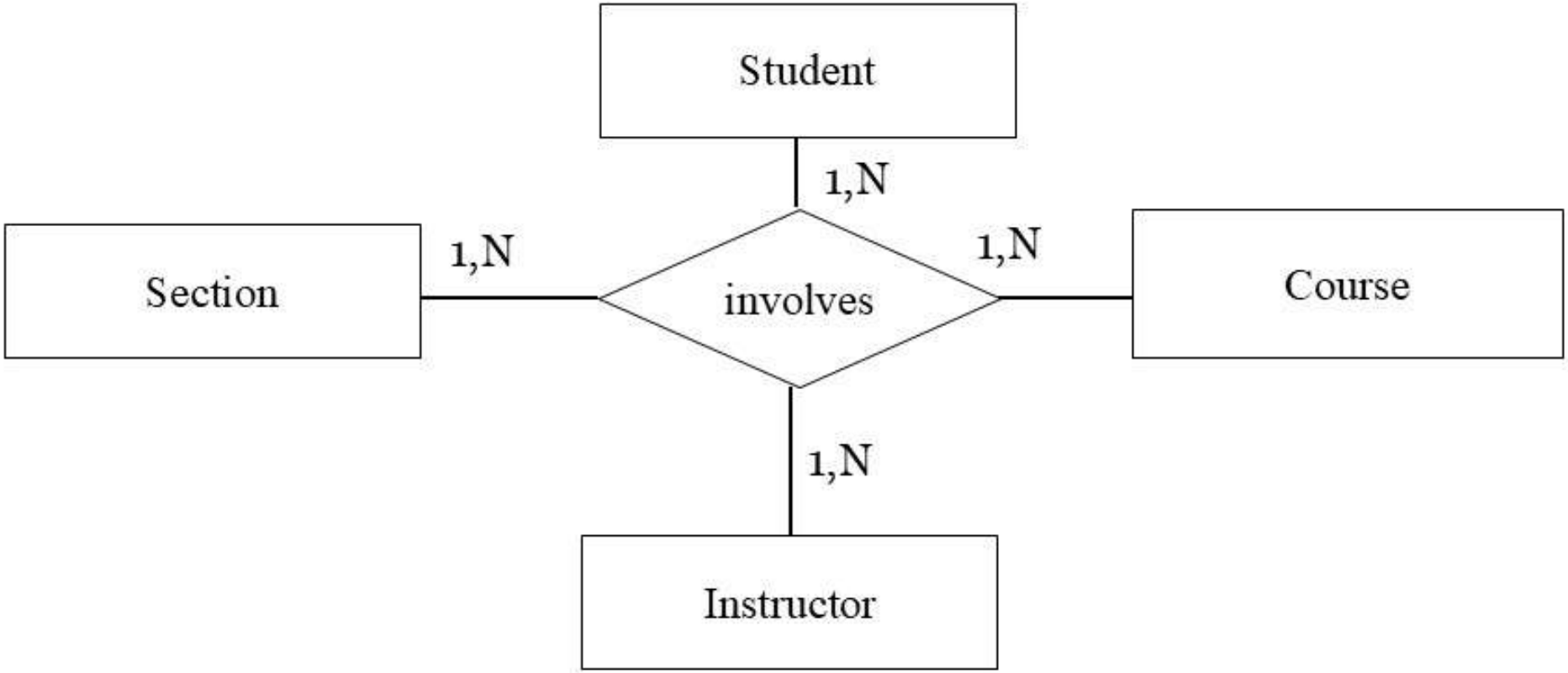



Pname
Occupation
address
Appno
Name
(1,1)
ASSIGNED
(1,N)
Rno
space
facility
CHILD
ROOM
(1,1)
(1,N)
HAS
CARED BY
(0,N)
(1,1)
EMPLOYEE
Eno
Ename

- Degree higher than two

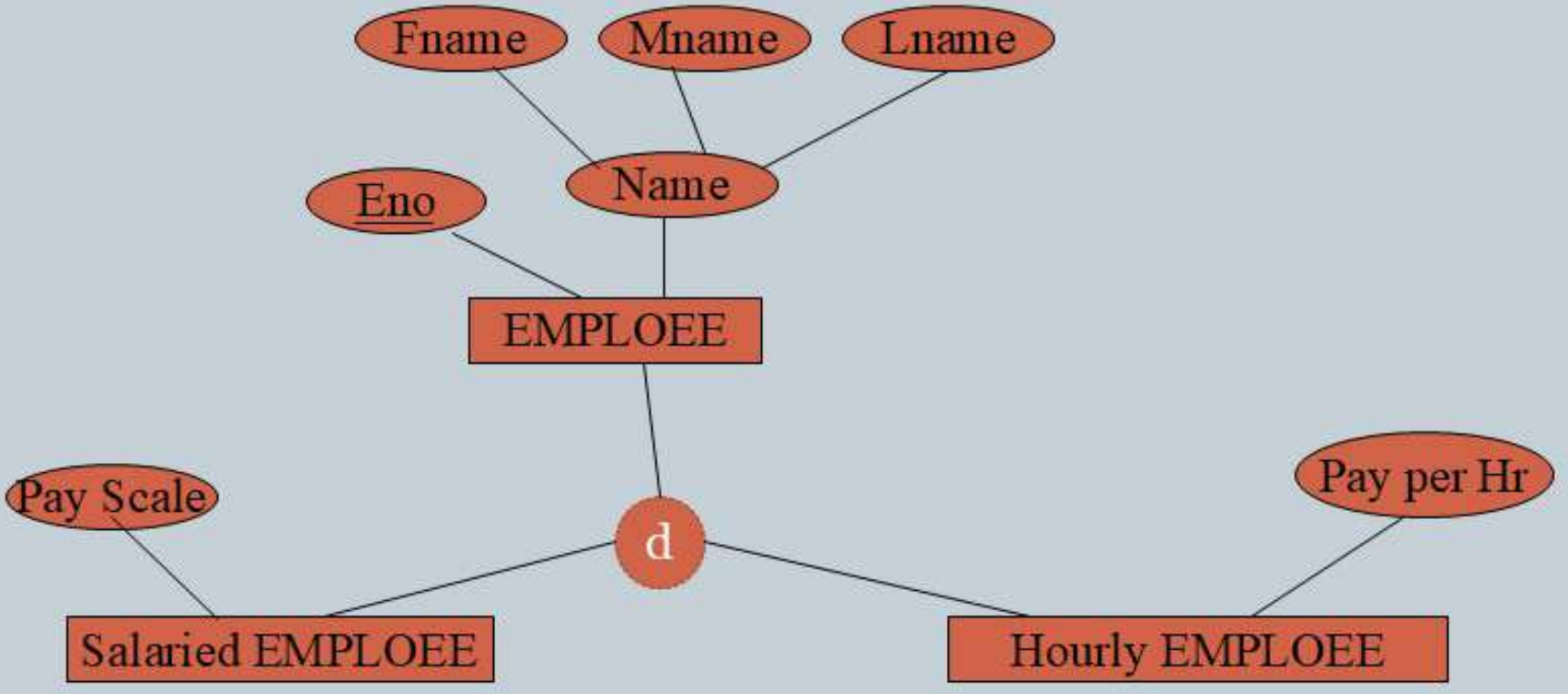
  - Ternary
  - N-ary


Section
1,1
involves
1,N
Course
1,N
Instructor


Student
1,N
Section
1,N
involves
1,N
Course
1,N
Instructor

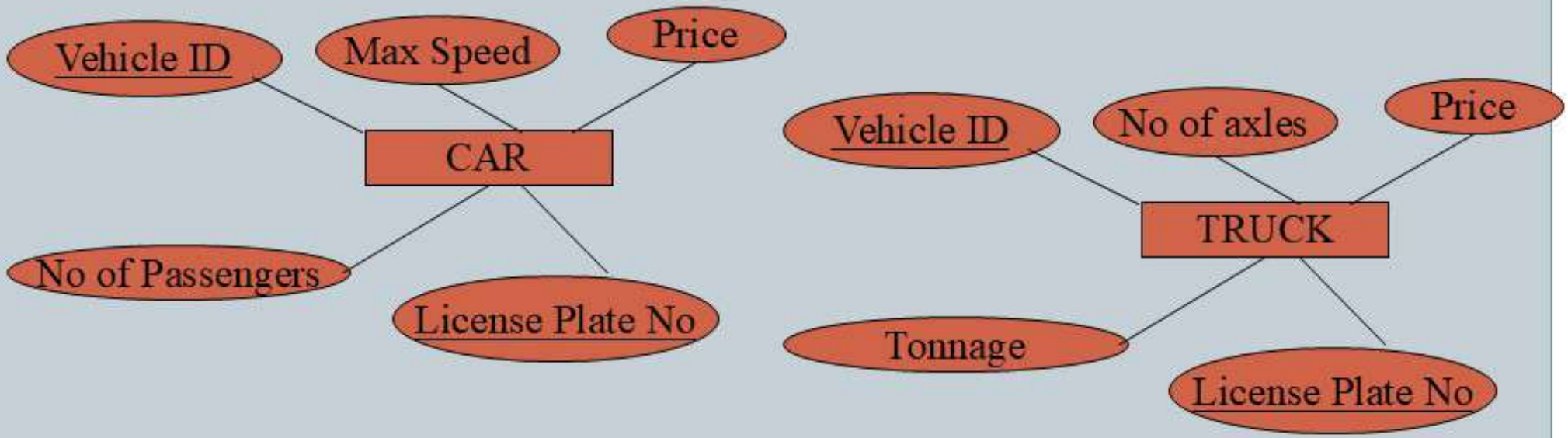
Conceptual Design- EER Model
Enhanced Entity Relationship Model

- Based on Inheritance –Superclass , subclass
- Specialization – process of defining a set of subclass of an entity type

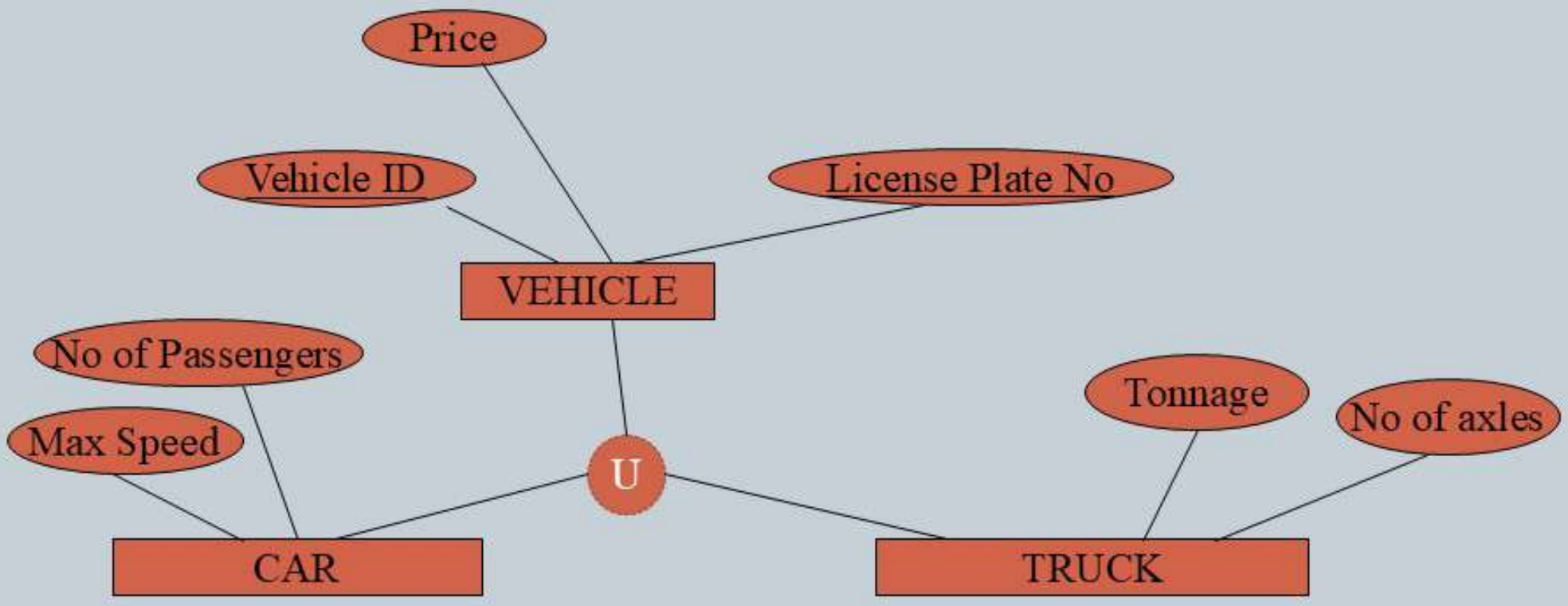


Conceptual Design- EER Model
Enhanced Entity Relationship Model

- Generalization – reverse process, combining the entity type with common attributes into single super class



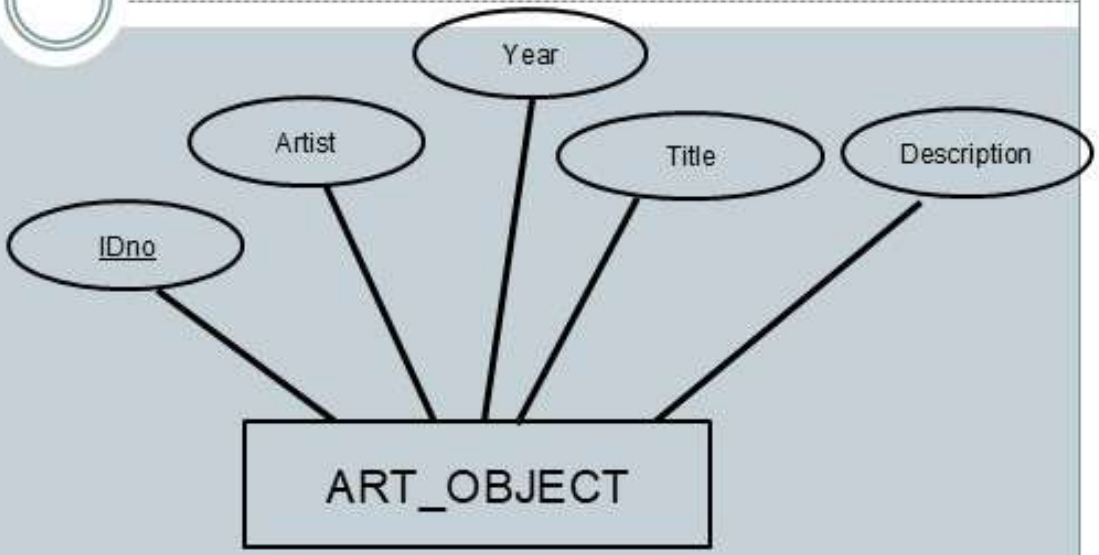
Conceptual Design- EER Model
Enhanced Entity Relationship Model



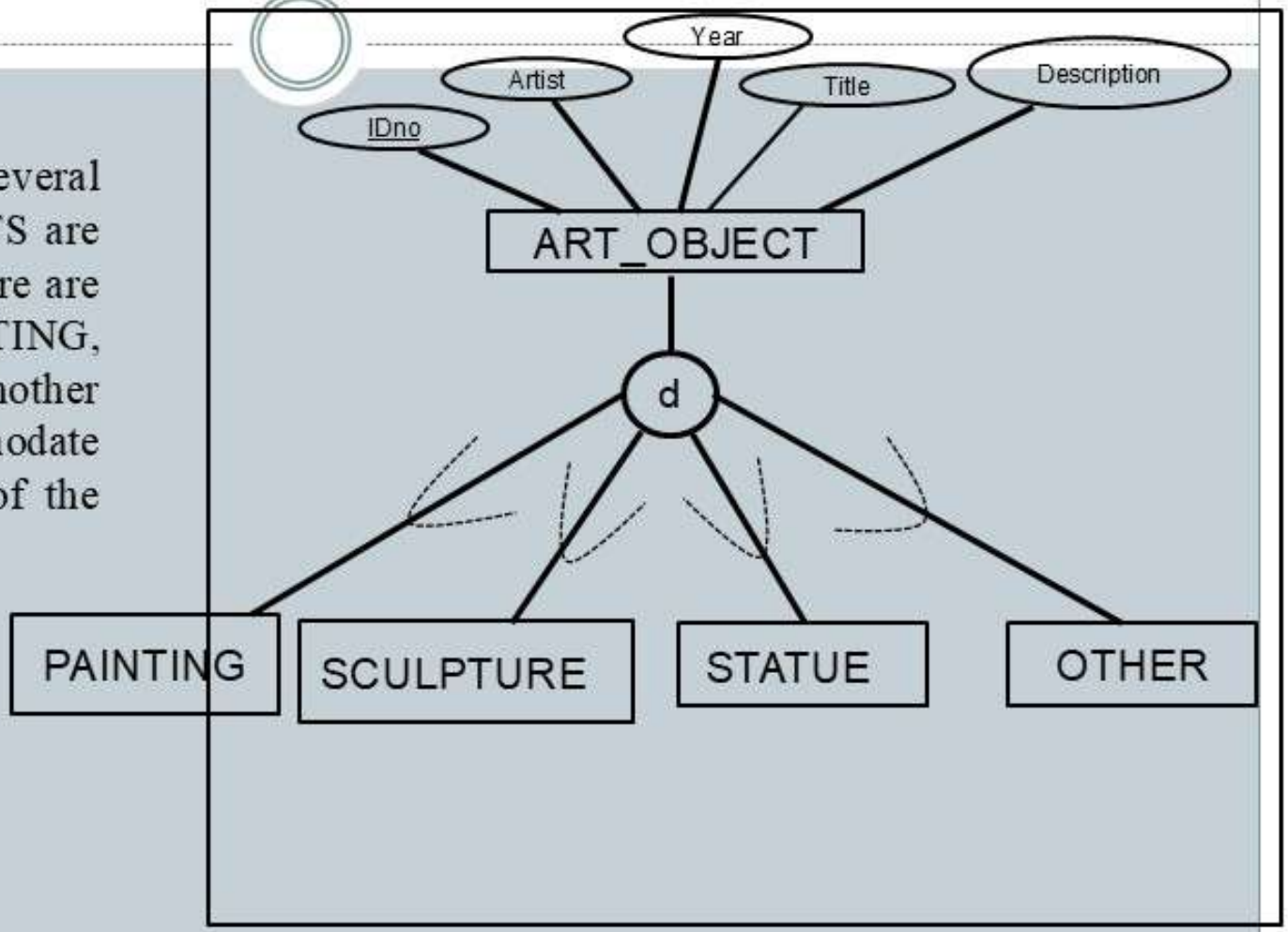
EER- example

- Design a database to keep track of information for an art museum. Assume that the following requirements were collected: ■ The museum has a collection of ART_OBJECTS. Each ART_OBJECT has a unique Id_no, an Artist (if known), a Year (when it was created, if known), a Title, and a Description. The art objects are categorized in several ways, as discussed below. ■ ART_OBJECTS are categorized based on their type. There are three main types: PAINTING, SCULPTURE, and STATUE, plus another type called OTHER to accommodate objects that do not fall into one of the three main types. ■ A PAINTING has a Paint_type (oil, watercolor, etc.), material on which it is Drawn_on (paper, canvas, wood, etc.), and Style (modern, abstract, etc.). ■ A SCULPTURE or a statue has a Material from which it was created (wood, stone, etc.), Height, Weight, and Style. ■ An art object in the OTHER category has a Type (print, photo, etc.) and Style. ■ ART_OBJECTs are categorized as either PERMANENT_COLLECTION (objects that are owned by the museum) and BORROWED. Information captured about objects in the PERMANENT_COLLECTION includes Date_acquired, Status (on display, on loan, or stored), and Cost. Information captured about BORROWED objects includes the Collection from which it was borrowed, Date_borrowed, and Date_returned. ■ Information describing the country or culture of Origin (Italian, Egyptian, American, Indian, and so forth) and Epoch (Renaissance, Modern, Ancient, and so forth) is captured for each ART_OBJECT. ■ The museum keeps track of ARTIST information, if known: Name, DateBorn (if known), Date_died (if not living), Country_of_origin, Epoch, Main_style, and Description. The Name is assumed to be unique. ■ Different EXHIBITIONS occur, each having a Name, Start_date, and End_date. EXHIBITIONS are related to all the art objects that were on display during the exhibition. ■ Information is kept on other COLLECTIONS with which the museum interacts, including Name (unique), Type (museum, personal, etc.), Description, Address, Phone, and current Contact_person.

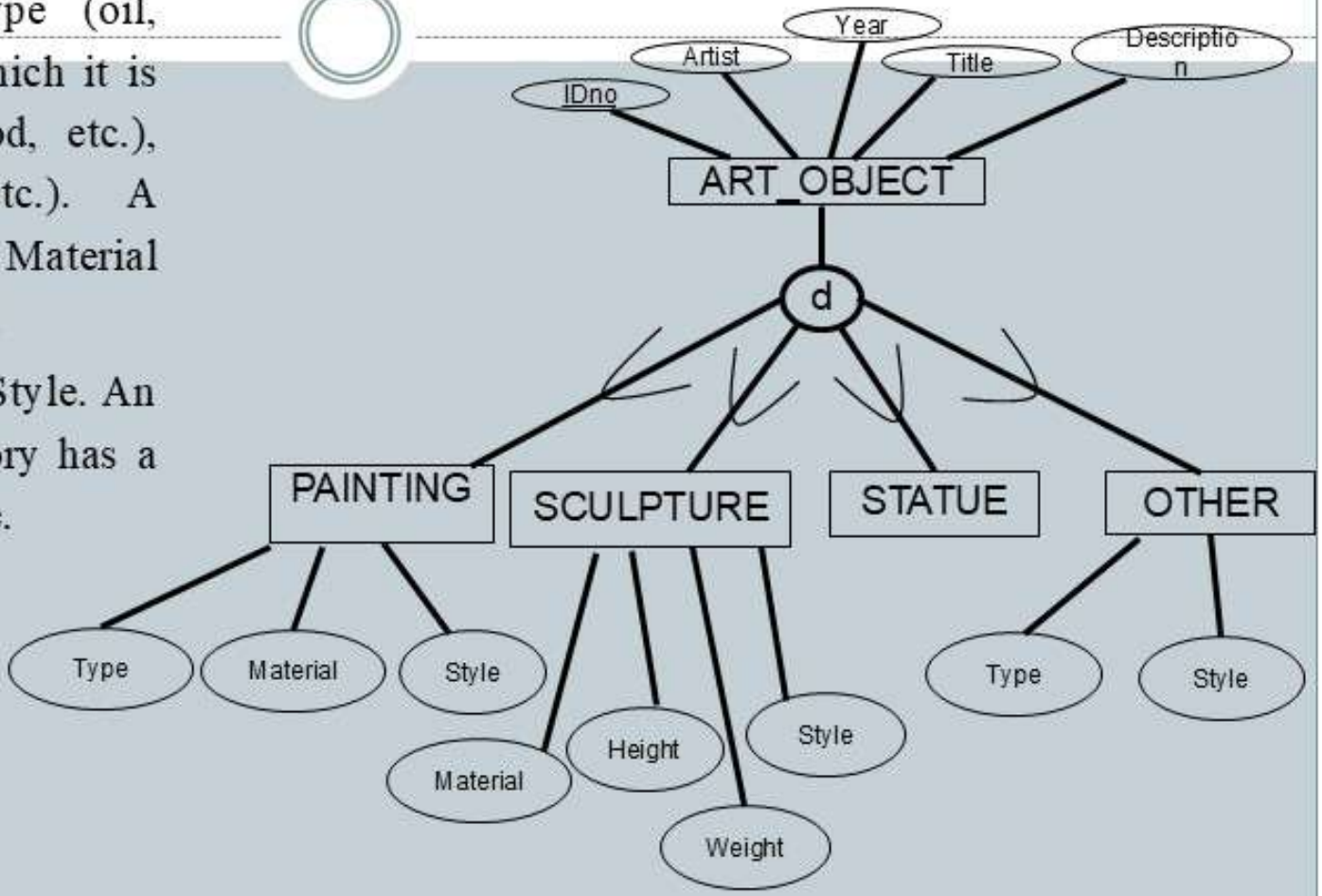
The museum has a collection of ART_OBJECTS. Each ART_OBJECT has a unique Id_no, an Artist (if known), a Year (when it was created, if known), a Title, and a Description.



The art objects are categorized in several ways, as discussed - ART_OBJECTS are categorized based on their type. There are three main types: PAINTING, SCULPTURE, and STATUE, plus another type called OTHER to accommodate objects that do not fall into one of the three main types.



A PAINTING has a Paint_type (oil, watercolor, etc.), material on which it is Drawn_on (paper, canvas, wood, etc.), and Style (modern, abstract, etc.). A SCULPTURE or a statue has a Material from which it was created (wood, stone, etc.), Height,Weight, and Style. An art object in the OTHER category has a Type (print, photo, etc.) and Style.



ART_OBJECTs are categorized as either PERMANENT_COLLECTION (objects that are owned by the museum) and BORROWED. Information captured about objects in the

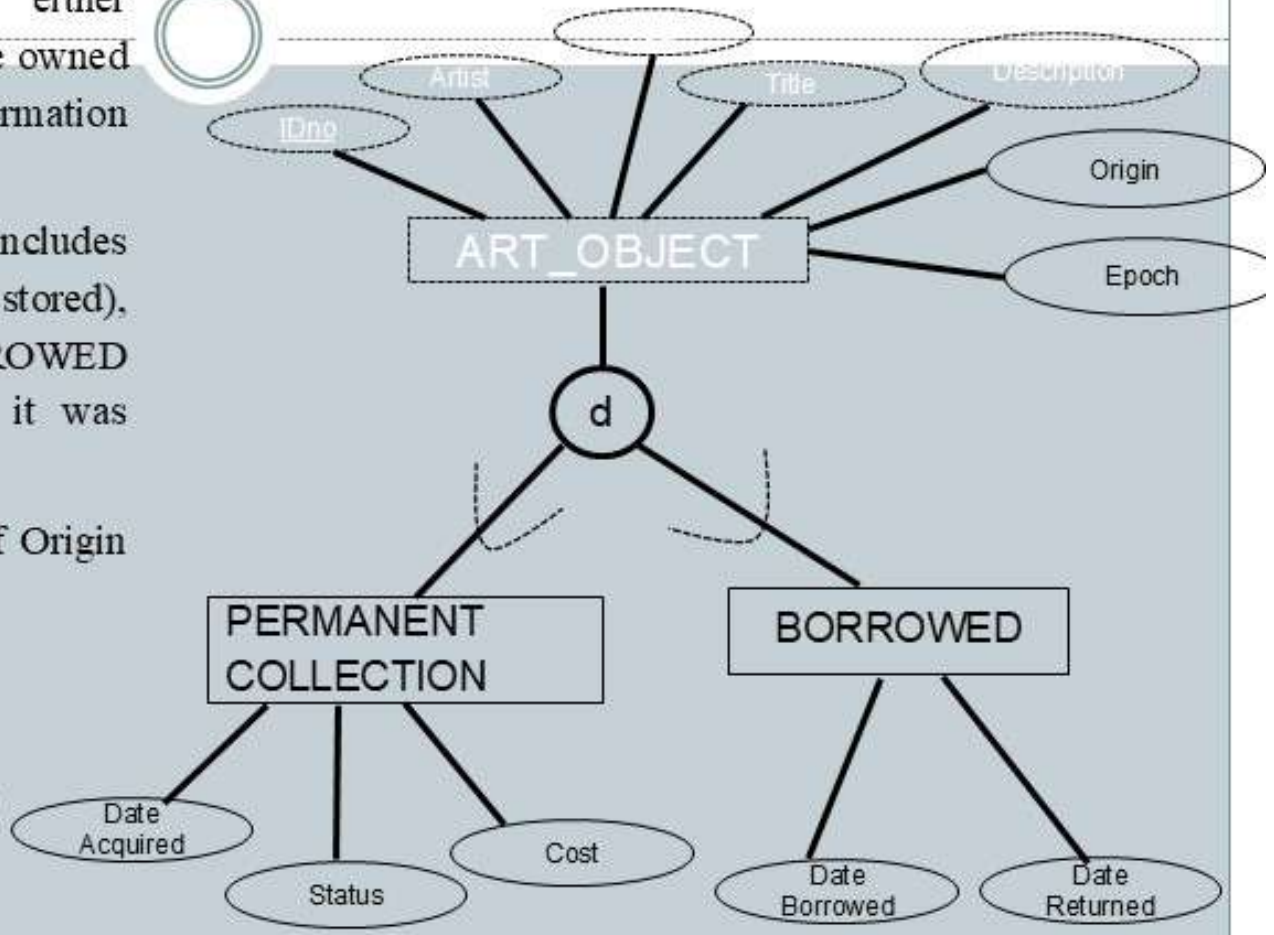
PERMANENT_COLLECTION includes Date_acquired, Status (on display, on loan, or stored), and Cost. Information captured about BORROWED objects includes the Collection from which it was borrowed, Date_borrowed, and Date_returned.

Information describing the country or culture of Origin (Italian, Egyptian,American, Indian,

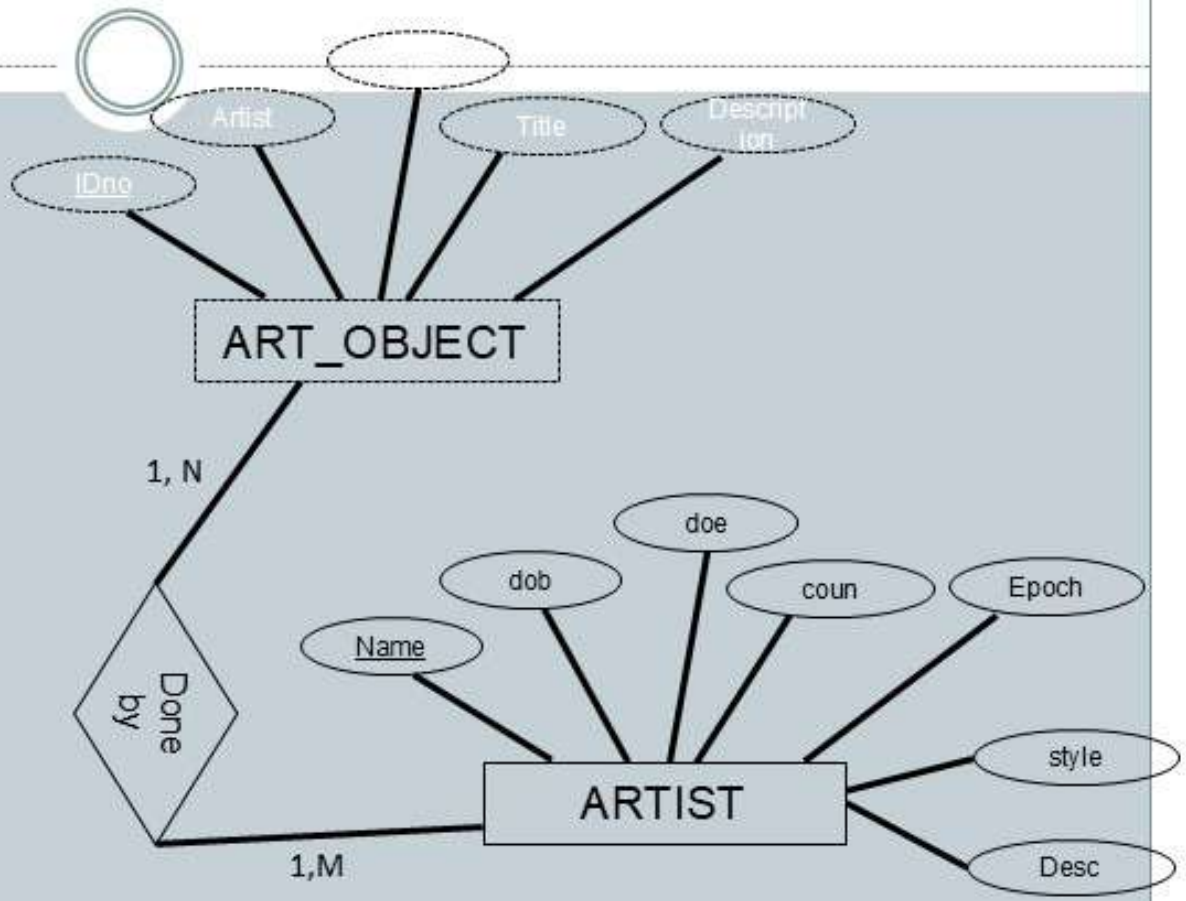
and so forth) and Epoch

(Renaissance, Modern,Ancient,

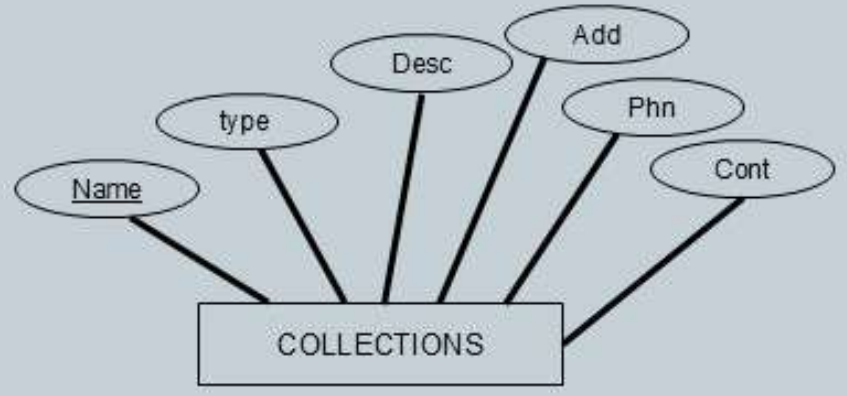
and so forth) is captured for each ART_OBJECT.

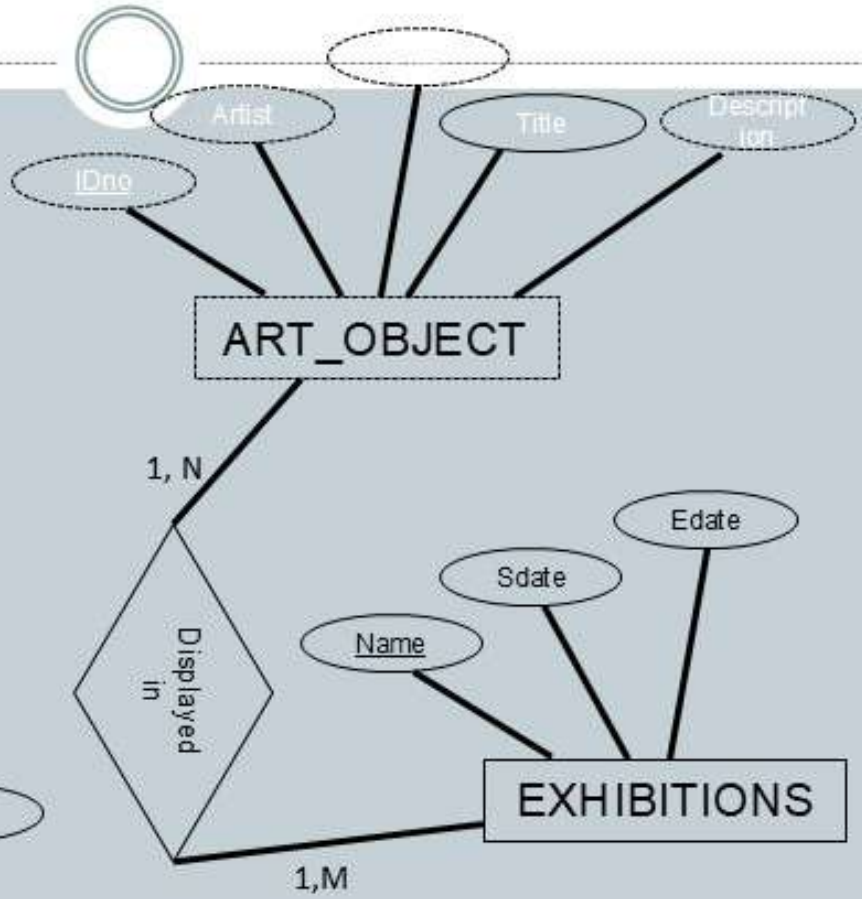


The museum keeps track of ARTIST information, if known: Name, DateBorn (if known), Date_died (if not living), Country_of_origin, Epoch,Main_style, and Description. The Name is assumed to be unique.

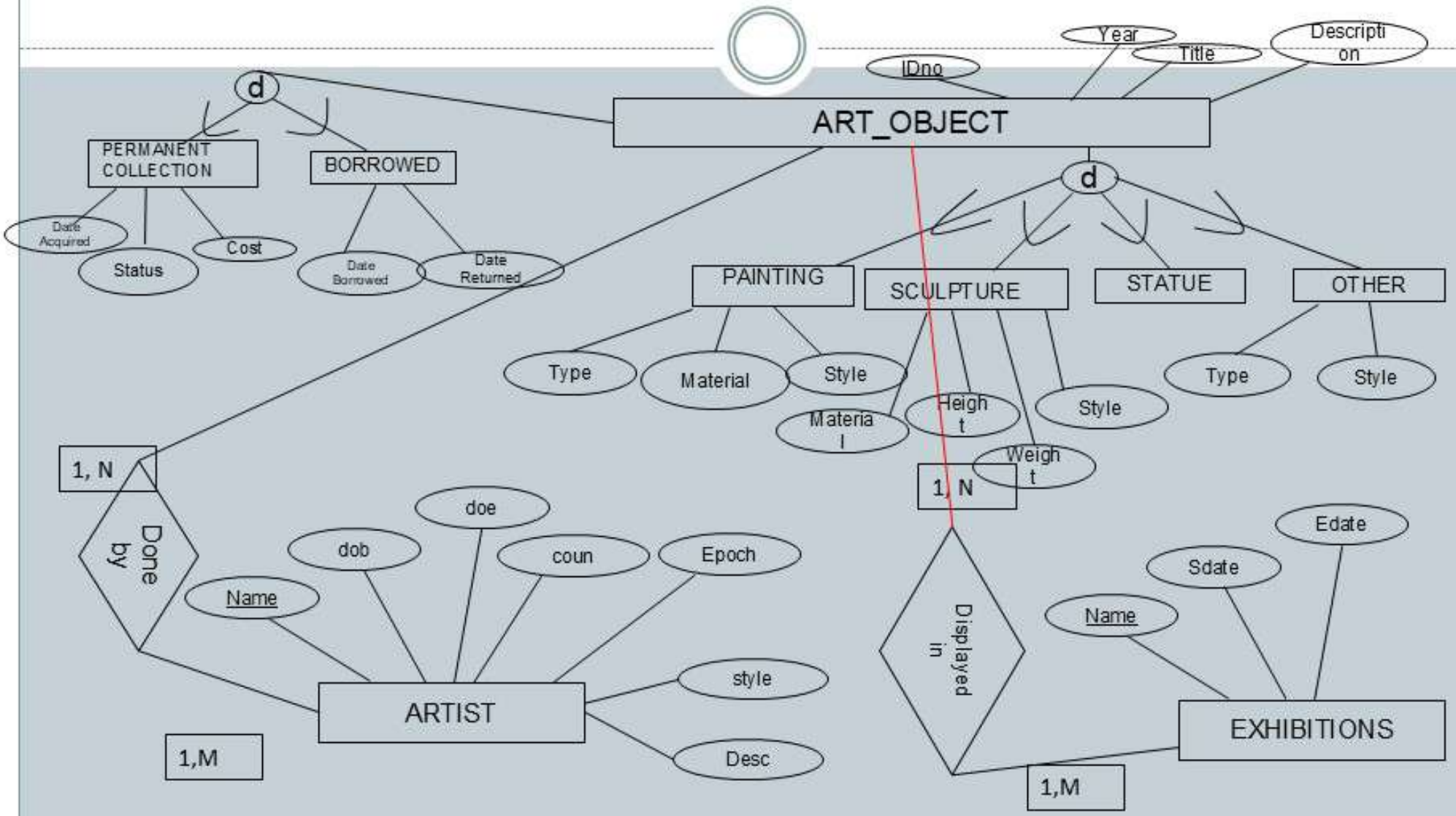


Different EXHIBITIONS occur, each having a Name, Start_date, and End_date. EXHIBITIONS are related to all the art objects that were on display during the exhibition. Information is kept on other COLLECTIONS with which the museum interacts, including Name (unique), Type (museum, personal, etc.), Description, Address, Phone, and current Contact_person.



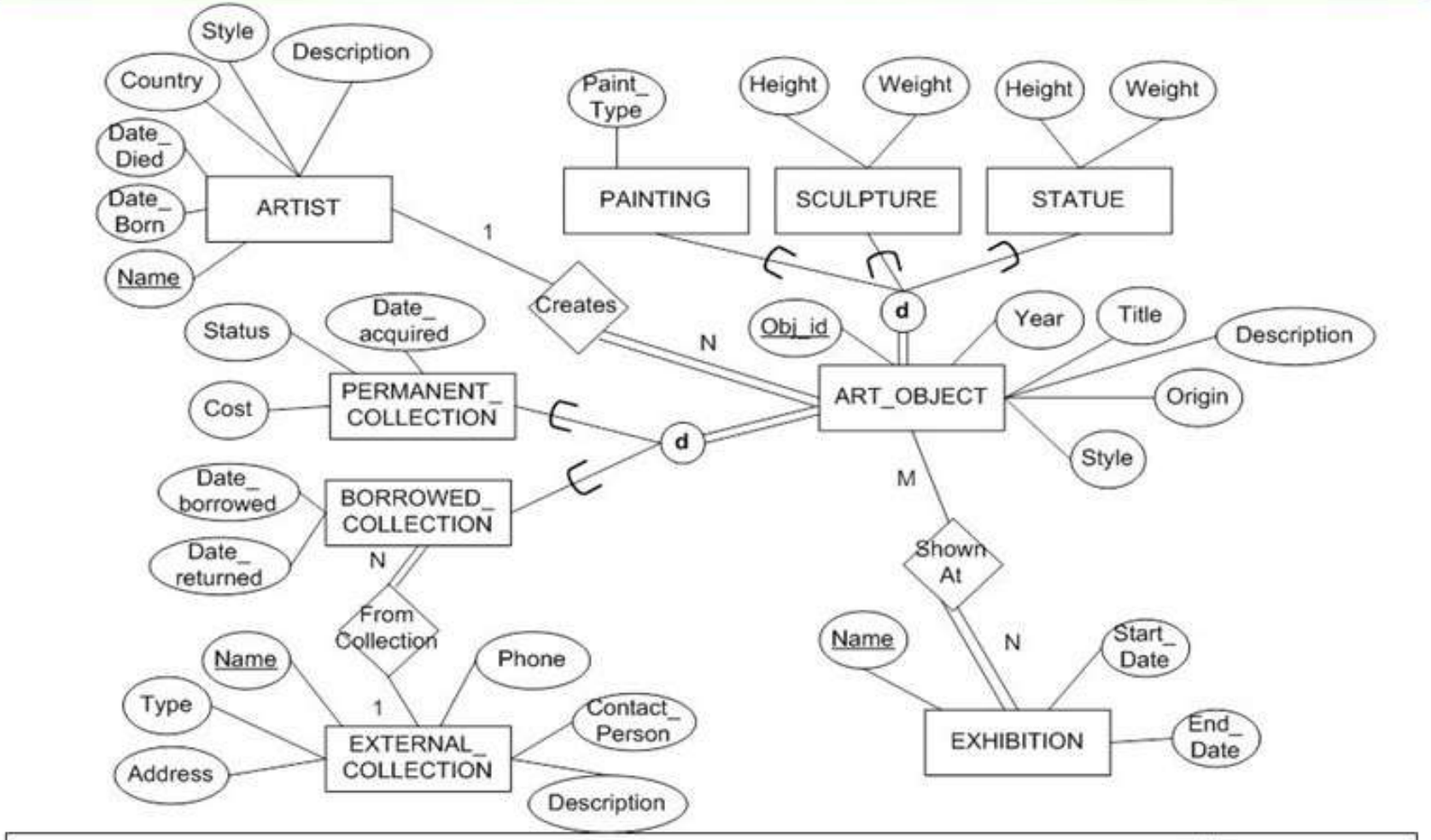


Museum DB : EER Schema



Solution

Exercise: Art Museum Database



Constraints

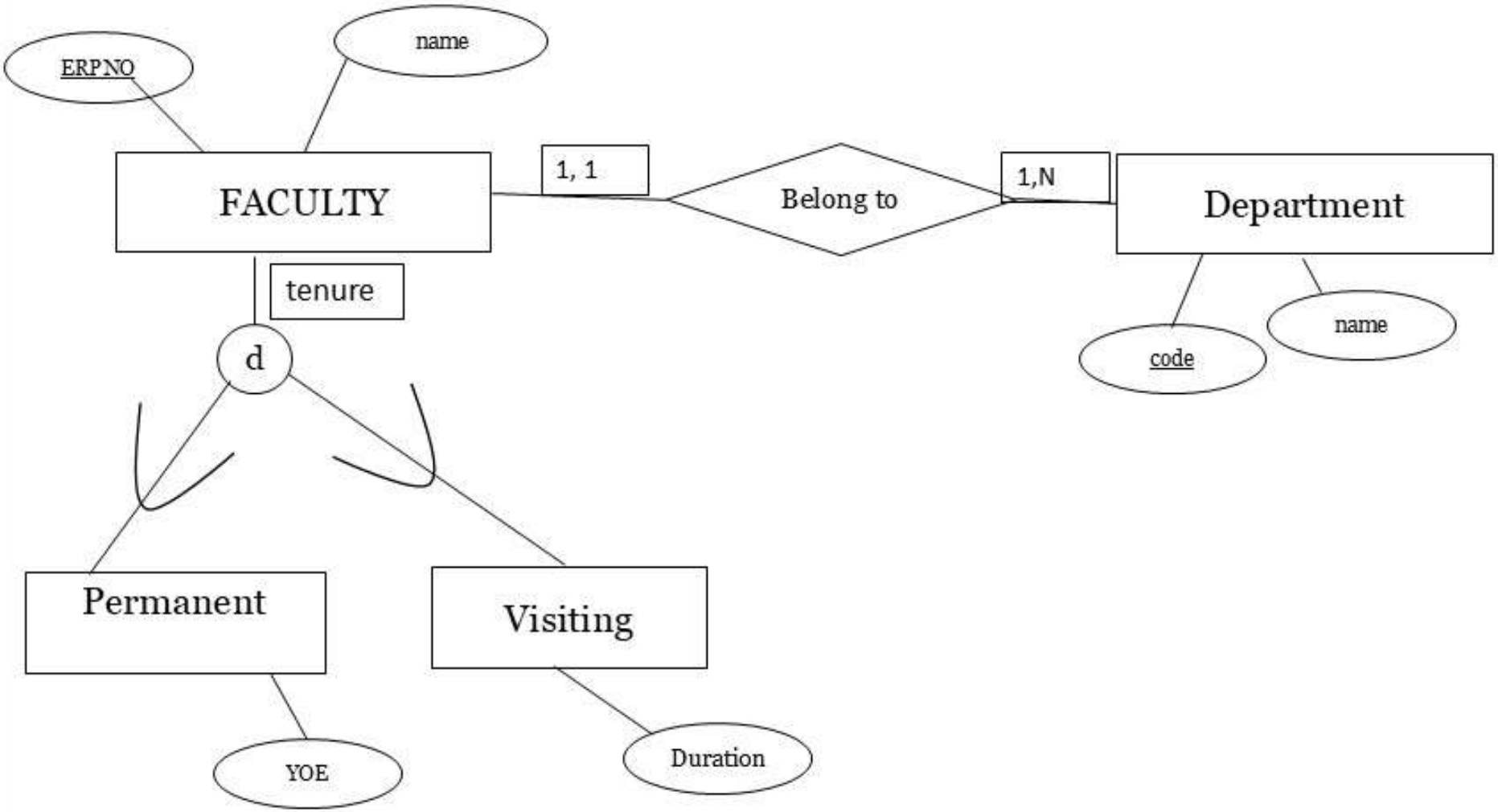
In EER

Constraints in Generalization/Specialization

If we can determine exactly those entities that will become members of each subclass by a condition, the subclasses are called predicate-defined (or condition-defined) subclasses

Condition is a constraint that determines subclass members

Display a predicate-defined subclass by writing the predicate condition next to the line attaching the subclass to its superclass


ERPNO
name
FACULTY
1, 1
Belong to
1,N
Department
tenure
d
code
name
Permanent
Visiting
YOE
Duration

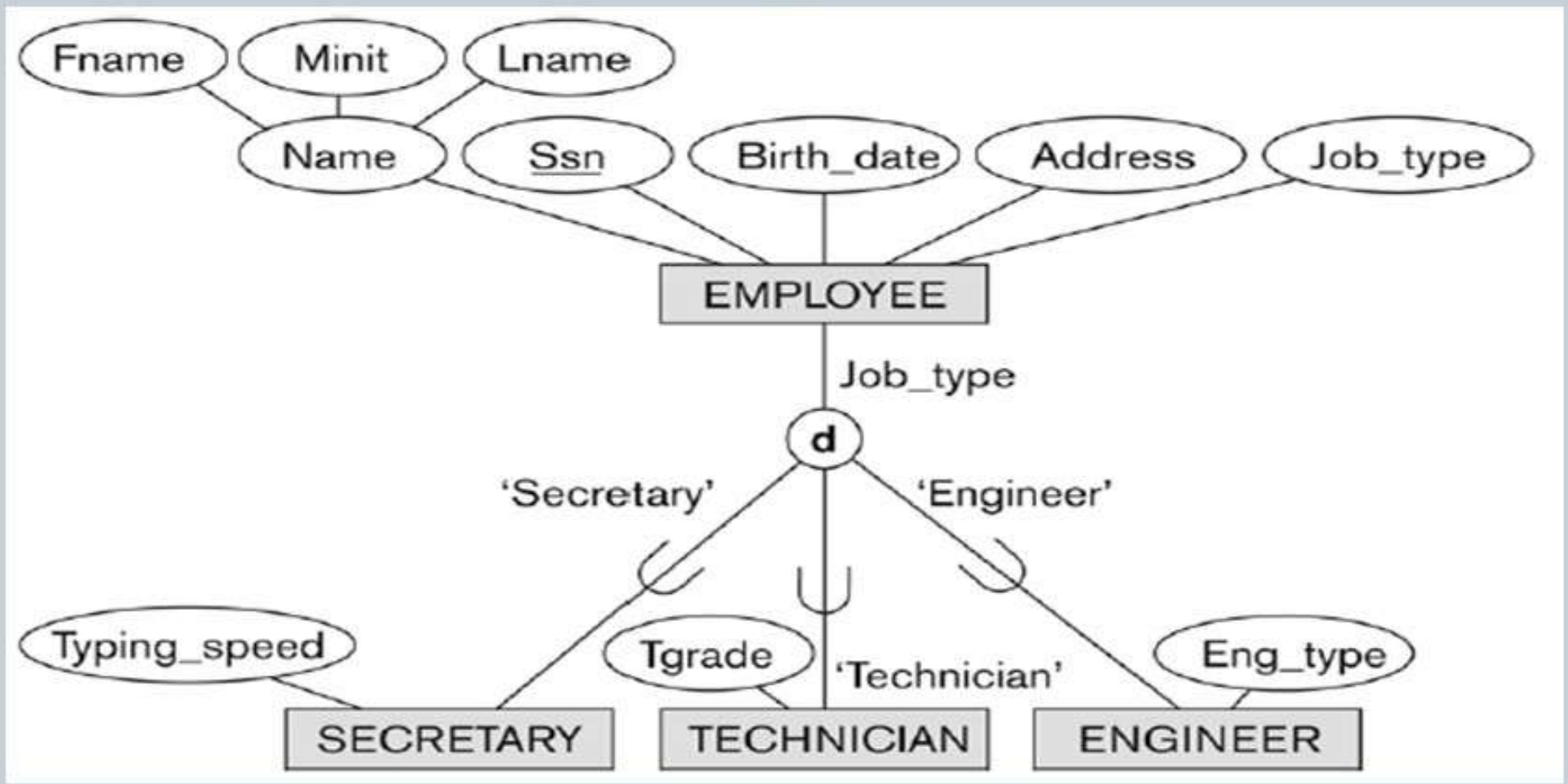
Constraints in Generalization/Specialization

If all subclasses in a specialization have membership condition on same attribute of the superclass, specialization is called an attribute-defined specialization

- Attribute is called the defining attribute of the specialization
- Example: Tenure is the defining attribute of the specialization {PERMANENT, VISITING} of FACULTY

If no condition determines membership, the subclass is called user-defined

- Membership in a subclass is determined by the database users by applying an operation to add an entity to the subclass
- Membership in the subclass is specified individually for each entity in the superclass by the user


Fname
Minit
Lname
Name
Ssn
Birth_date
Address
Job_type
EMPLOYEE
Job_type
d
'Secretary'
'Engineer'
'Technician'
Typing_speed
Tgrade
Eng_type
SECRETARY
TECHNICIAN
ENGINEER

Constraints in Generalization/Specialization

Two basic constraints can apply to a specialization/generalization:

- Disjointness Constraint
- Completeness Constraint

Disjointness Constraint:

- Specifies that the subclasses of the specialization must be *disjoint*:
  - an entity can be a member of at most one of the subclasses of the specialization
- Specified by ***d*** in EER diagram
- If not disjoint, specialization is *overlapping* that is the same entity may be a member of more than one subclass of the specialization
- Specified by ***o*** in EER diagram

Completeness Constraint:

Total specifies that every entity in the superclass must be a member of some subclass in the specialization/generalization

Shown in EER diagrams by a ***double line***

Partial allows an entity not to belong to any of the subclasses

Shown in EER diagrams by a single line

four types of specialization/generalization:

Disjoint, total

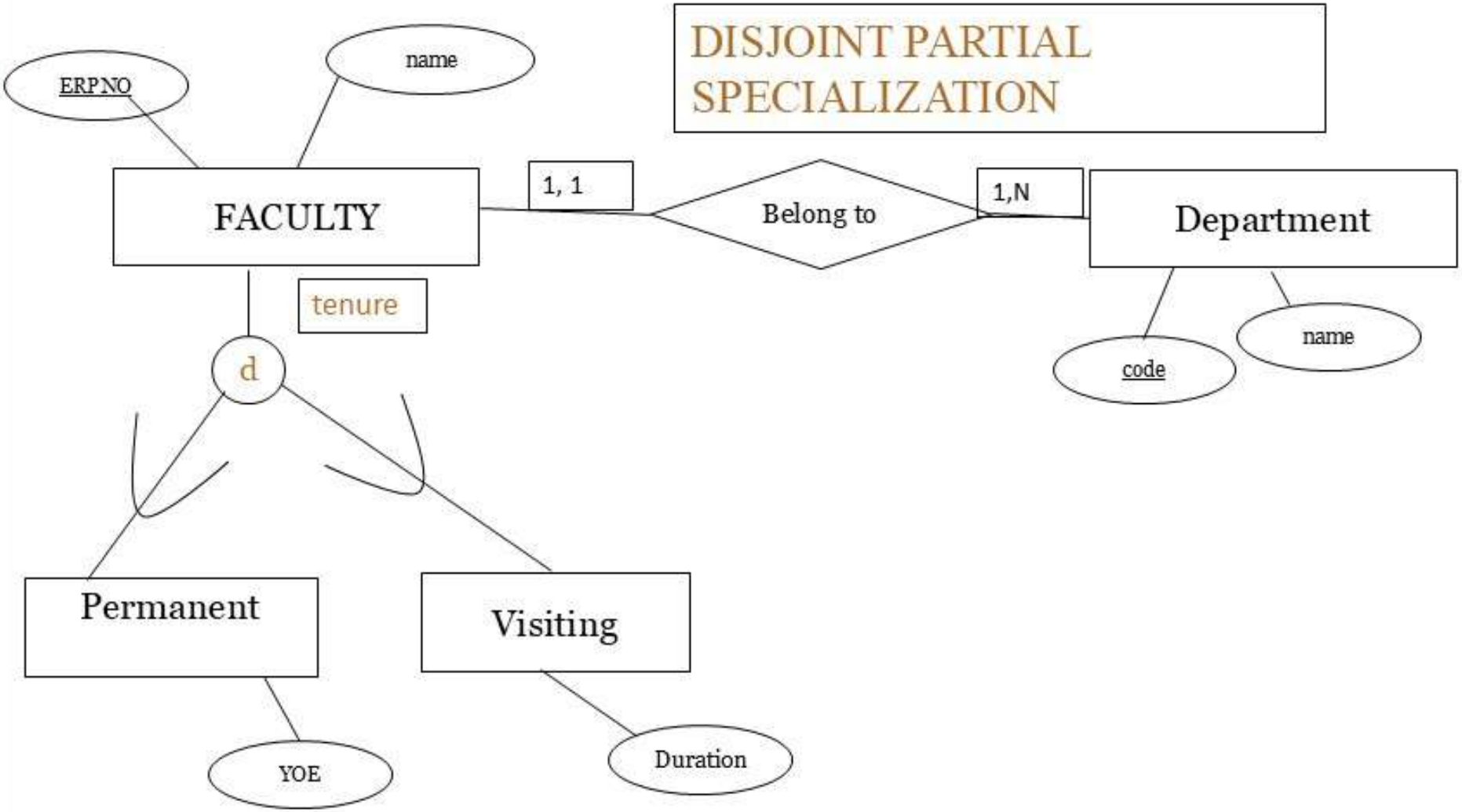
Disjoint, partial

Overlapping, total

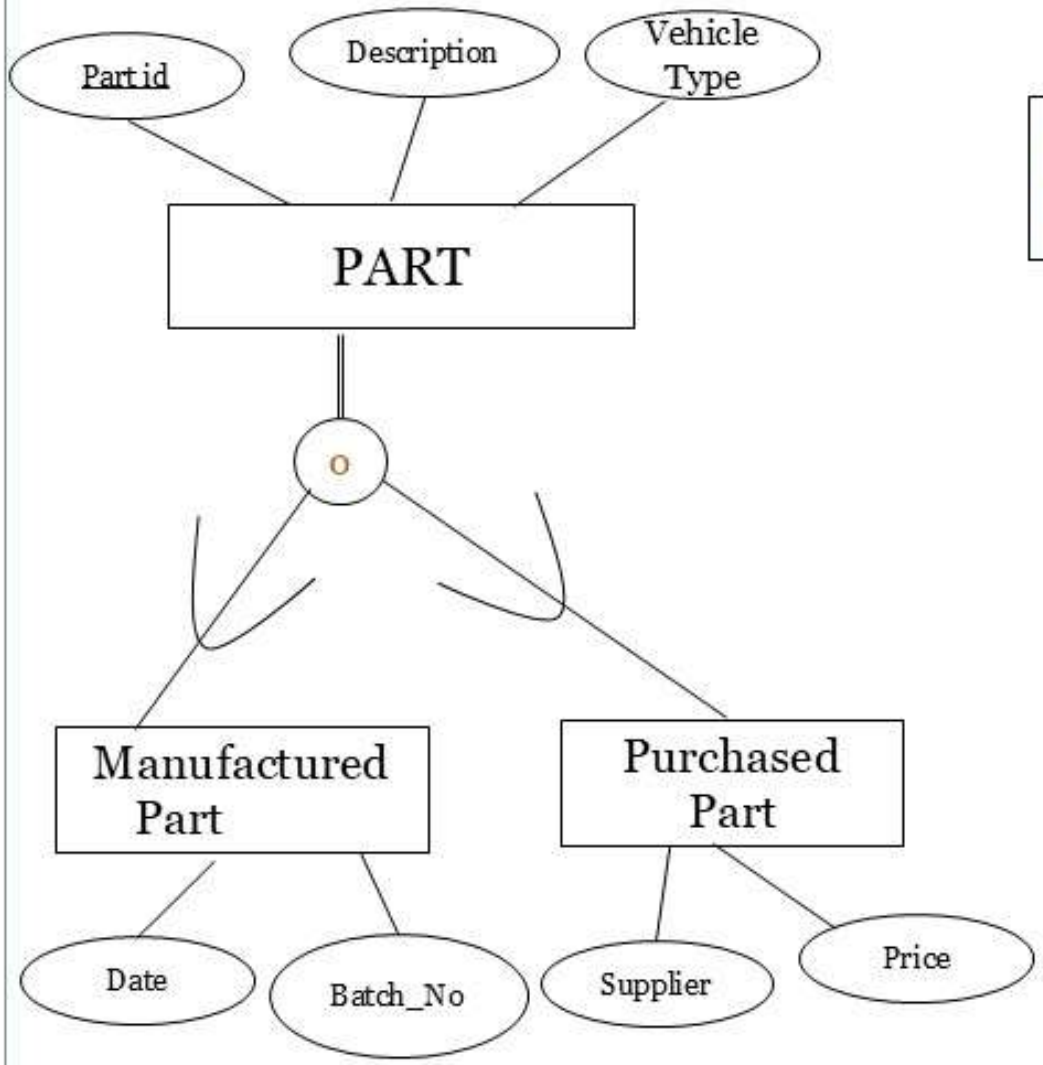
Overlapping, partial

Generalization usually is total because the superclass is derived from the subclasses.

DISJOINT PARTIAL SPECIALIZATION



OVERLAPPING TOTAL SPECIALIZATION



Union Types

All of the *superclass/subclass relationships* we have seen thus far have a single superclass.

A shared subclass is a subclass in which:

- *more than one* distinct superclass/subclass relationships
- each relationships has a single superclass
- shared subclass leads to multiple inheritance

In some cases, we need to model a *single superclass/subclass relationship* with *more than one* superclass

Super classes can represent different entity types

Such a subclass is called a category or UNION TYPE

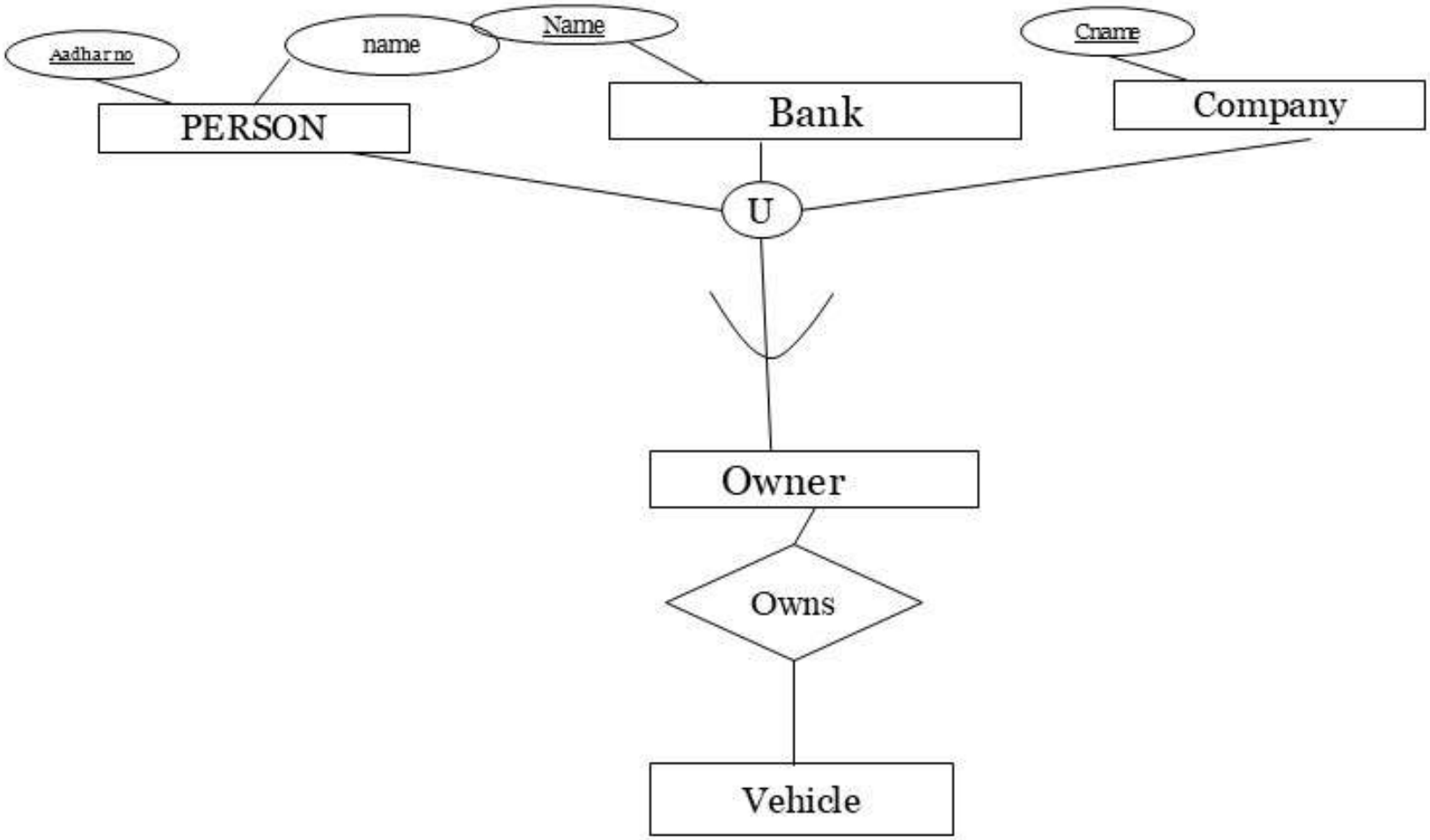
Union Types

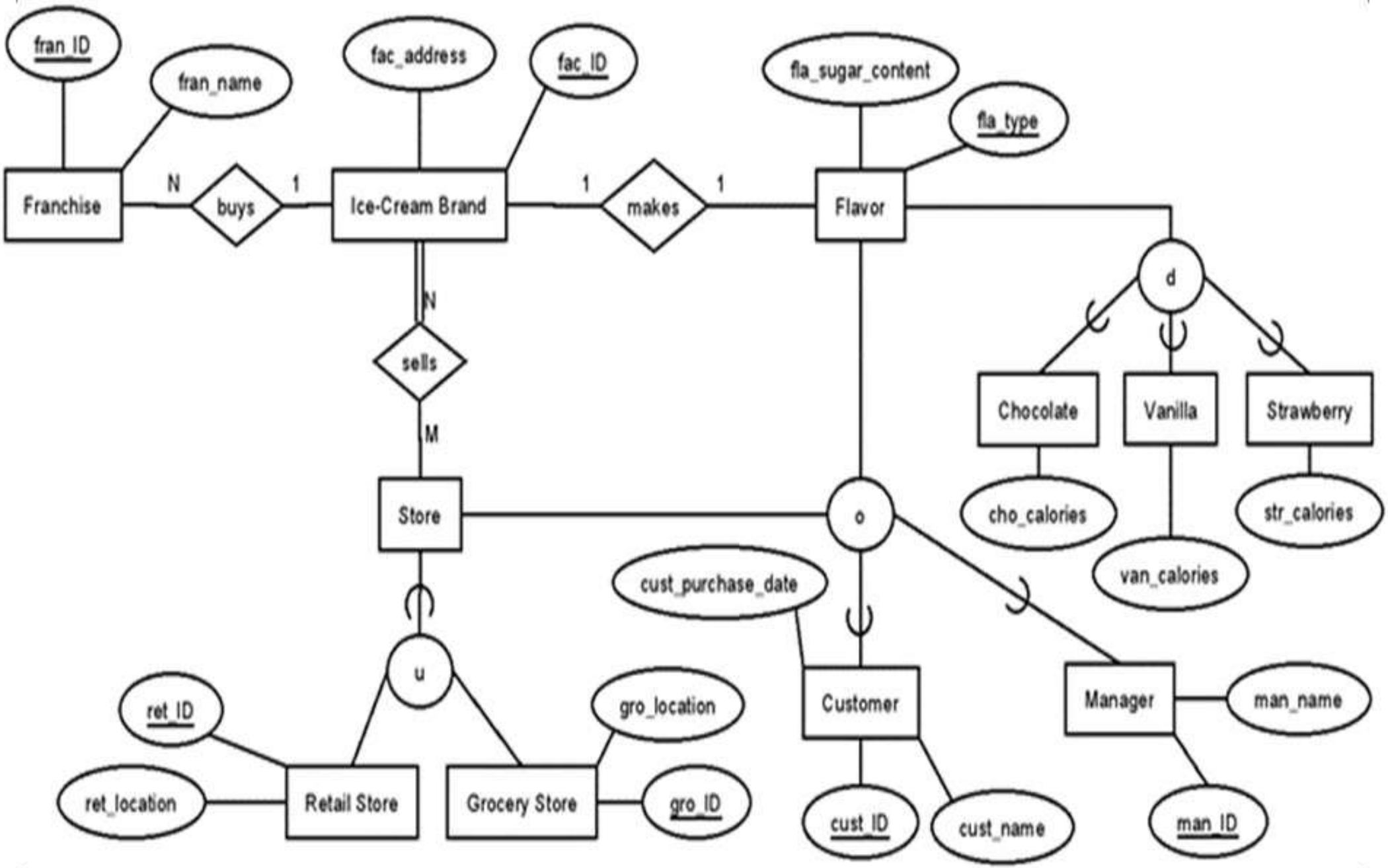
In a database for vehicle registration, a vehicle owner can be a PERSON, a BANK (holding a lien on a vehicle) or a COMPANY.

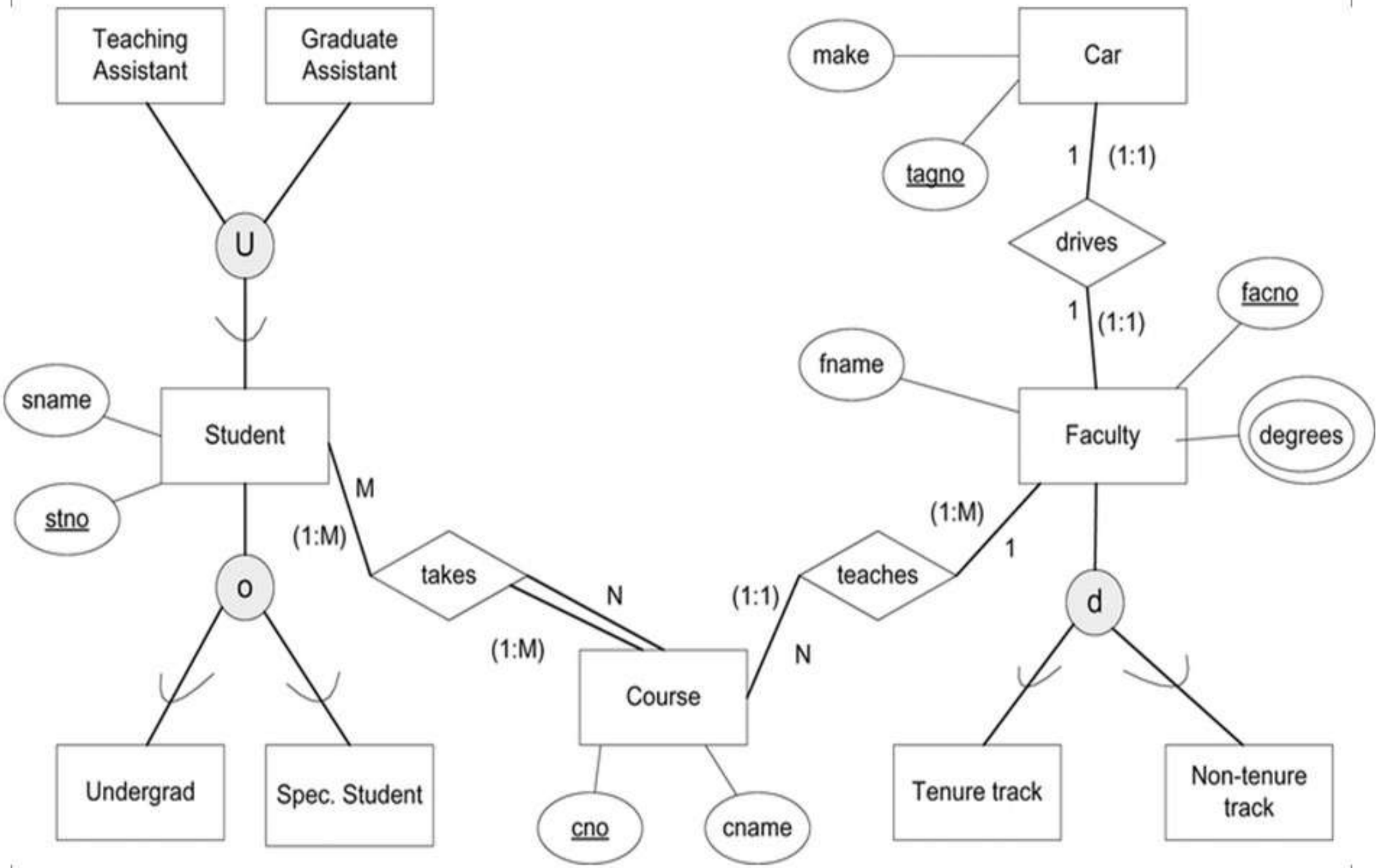
- A *category* (UNION type) called OWNER is created to represent a subset of the *union* of the three superclasses COMPANY, BANK, and PERSON
- A category member must exist in ***at least one*** of its superclasses

Difference from *shared subclass*, which is a:

- subset of the *intersection* of its superclasses
- shared subclass member must exist in ***all*** of its superclasses


Aadhar no
name
Name
Cname
PERSON
Bank
Company
U
Owner
Owns
Vehicle


fran_ID
fran_name
fac_address
fac_ID
fla_sugar_content
fla_type
Franchise
N
buys
1
Ice-Cream Brand
1
makes
1
Flavor
d
N
sells
M
Chocolate
Vanilla
Strawberry
cho_calories
van_calories
str_calories
Store
o
cust_purchase_date
u
ret_ID
gro_location
Customer
Manager
man_name
ret_location
Retail Store
Grocery Store
gro_ID
cust_ID
cust_name
man_ID


Teaching Assistant
Graduate Assistant
U
sname
Student
stno
o
Undergrad
Spec. Student
M
(1:M)
takes
N
(1:M)
Course
cno
cname
(1:1)
teaches
(1:M)
1
d
Tenure track
Non-tenure track
make
Car
tagno
1
(1:1)
drives
1
(1:1)
facno
fname
Faculty
degrees