escola d'enginyeria

UAB

Block 2
Basic ER-Design (PART1)

Debora Gil, Oriol Ramos, Carles Sánchez, Alejandro Párraga

Basic ER Design Contents

1. E-R Model Introduction

2. Basic Structures

    2.1 Entities

    2.2 Attributes

    2.3 Relationships

3. Relationships Features

    3.1 Cardinality
    3.2 Degree
    3.3 Participation

Basic ER Design Contents

1. E-R Model Introduction

Definition

Description / Schematic representation of a real situation (which our application must manage) that requires saving information from different objects (entities) that are related to each other.

Example:

University Enrolment Management: In an enrolment management system I want to know which students have enrolled in engineering subjects.

Definition

Description / Schematic representation of a real situation (which our application must manage) that requires saving information from different objects (entities) that are related to each other.

Exemple:

Enrolment Management: In an enrolment management system I want to know which students have enrolled in engineering subjects.

For each student (and subject) I want to keep certain information (name, contact details, ...). In particular, I will have to save the necessary data so as not to confuse 2 students

Components

They let you describe WHO DOES WHAT:

Entities. Object (event) from real world→ "Subjects, complements"

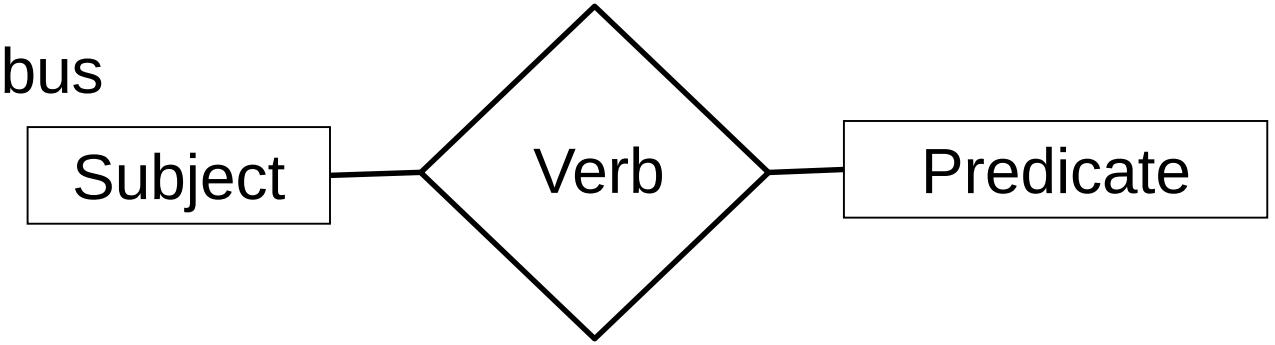
Relationships (Relations, Interrelations). Association relating several entities → "Verbs"

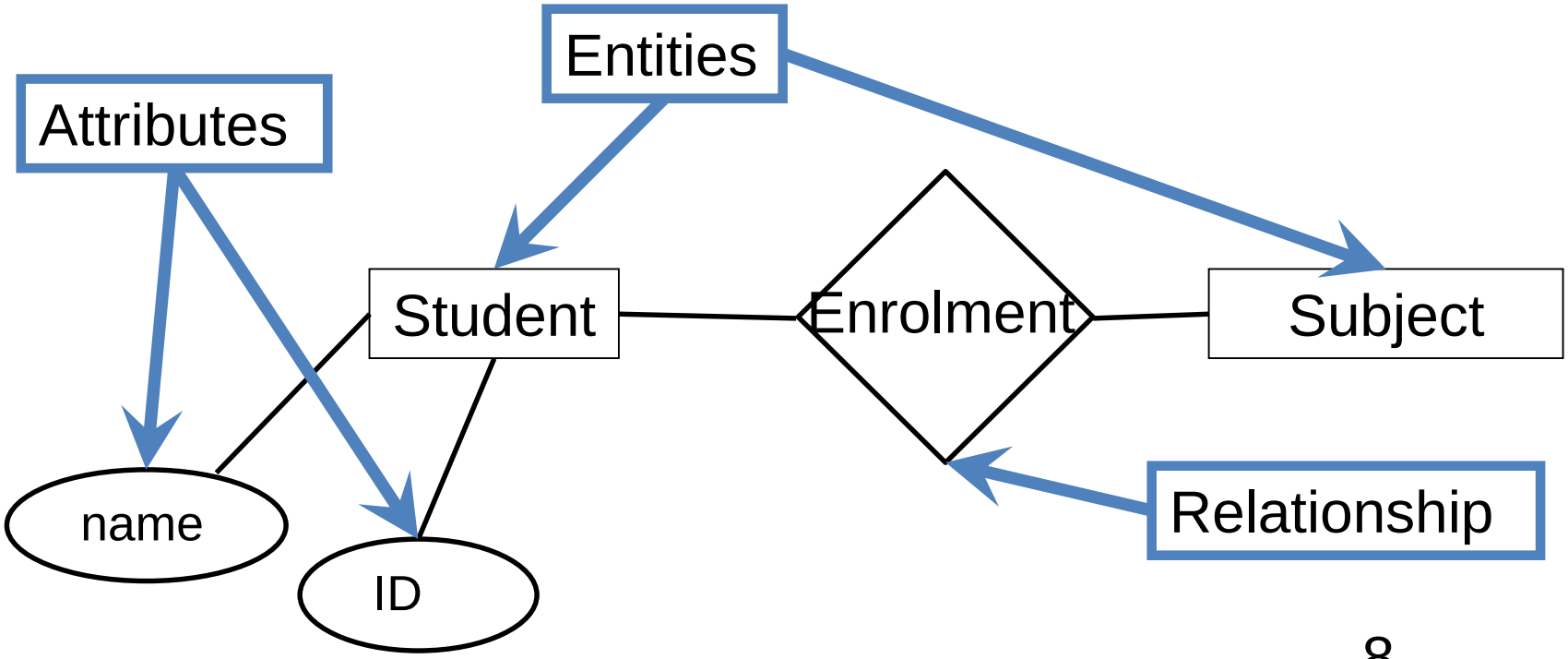
Attributes. Entities and relationships Information or Features

E-R Diagram is a graphical representation of all entities and relationships with their attributes

E-R Diagram

Relationships→ rhombus
Attributes→ ellipses
Entities → rectangles





2. Basic Structures

2.1 Entities

2.2 Attributes

2.1 Entities

2.1.1 Definition
2.1.2 Candidate Keys
2.1.3 Primary Key

Definition

Real-world object or action distinguishable from the rest of which we are interested in keeping some properties. Semantic description of an object. They are represented by squares.

An entity can be Concrete (corresponds to a physical object) or Abstract (corresponds to an action or concept).

Entities have a set of characteristics (attributes) with values that uniquely identify each instance (case, tuple) of the entity

Examples

- A university **student** is a concrete entity.
- Each student must have a characteristic that uniquely identifies them. For example, NIU 1007899 could uniquely identify a particular student.
- Similarly, book loans can be considered as abstract entities, and the UAB library's loan code L-15AJY9 uniquely identifies "loan" instances.

Candidate Keys

Every entity must have a subset of its attributes with values that uniquely identify each instance (case, tuple) of the entity:

Candidate Keys (CK):

Minimum set of attributes that uniquely identify each instance. A set of attributes that uniquely identify an instance and do not contain other keys inside or redundant attributes

Candidate Keys

Identify: CK attribute values are not repeated. Different instances have some of the CK attributes different

Minimality: If we drop any of CK attributes we could have instances with the same values of CK and therefore they would no longer identify

Examples

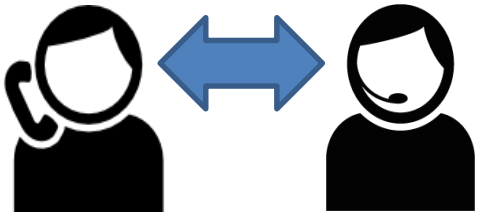
What are the necessary attributes to uniquely identify…

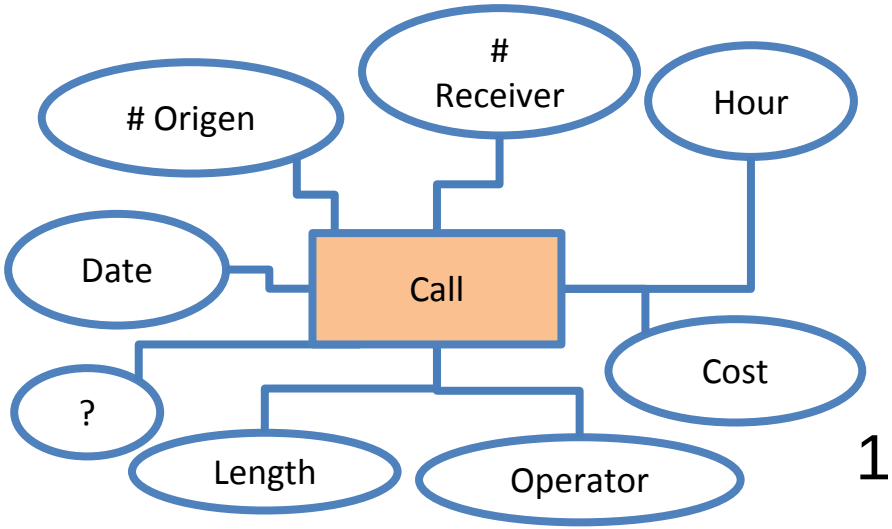
- A member?

- Name
- Surname
- Date of birth
- DNI

- Marital status
- Telephone
- Member code
- Address

- a phone call?





Examples

- Member

1. DNI
2. Member Code

DNI			Date	Member Code	Name
3676373L			1-1-22	123	Juan
4748474P			1-1-22	333	Juan
6727271Q			1-1-23	667	Pere

Equivalents as a minimal identifiers

- Phone call

#Origin	Cost	Hour	Date	#Receiver	Min
66234560	0.8	10:00	1-9-23	93581444	1
93581444	0.4	10:20	1-9-23	66234501	5
66234501	0.8	10:00	2-9-23	93581444	1

1. (date, hour, #Origen)
2. (date, hour, #Receiver)

All 3 attributes are required to identify. If we drop anyone, there are repetitions

Primary Key

Primary Key (PK): CK selected by the database designer
An entity without PK is NOT WELL-DEFINED

- Member

  Member code

DNI			Date	Member Code	Name
3676373L			1-1-22	123	Juan
4748474P			1-1-22	333	Juan
6727271Q			1-1-23	667	Pere

#Origin	Cost	Hour	Date	#Receiver	Min
66234560	0.8	10:00	1-9-23	93581444	1
93581444	0.4	10:20	1-9-23	66234501	5
66234501	0.8	10:00	2-9-23	93581444	1

- Phone call

(date,hour,#Origen)

2.2 Attributes

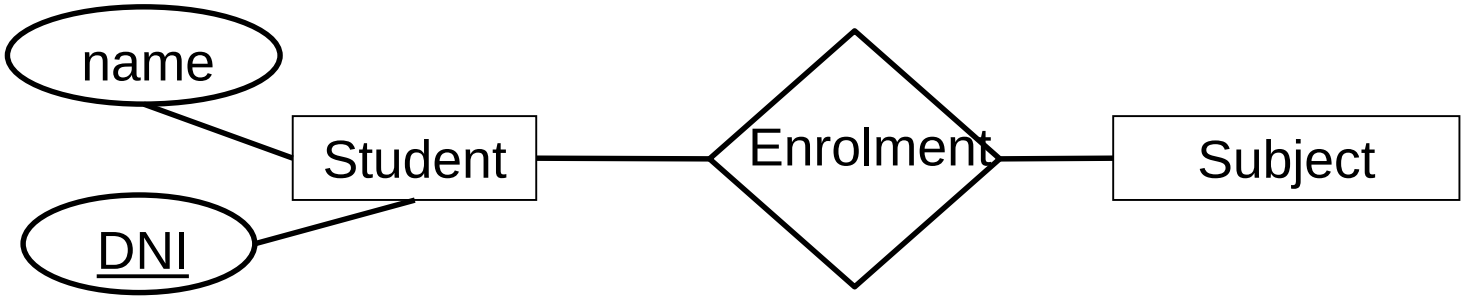
2.1.1 Definition
2.1.2 Domain
2.1.3 Attributes types

Definition

Characteristics that describe each entity / relationship. They are represented by ellipses

Their values define the instances of the entities that is part of the information that is stored in the DB.

OBS: The attribute values can change over time



DNI	Name
3676373L	Pere
4748474P	Juan
6727271Q	Pere

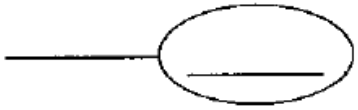
Student instances

Domain

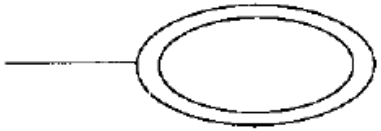
Each attribute has associated a domain or values set (type and range) permitted

Attributes	Domain
Name	Set of strings with length less than 50
Date of Birth	Set of strings with the format “DD/MM/YYY”
Phone number	Set of all 9-digit integers
Penalties	Set of {1, 0}

Attribute types

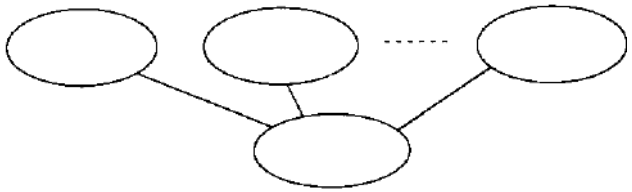


PK: set of attributes that uniquely identify entity's instances



Multivalued: the attribute has more than one value for each instance (they are "vectors" of variable length for each instance)

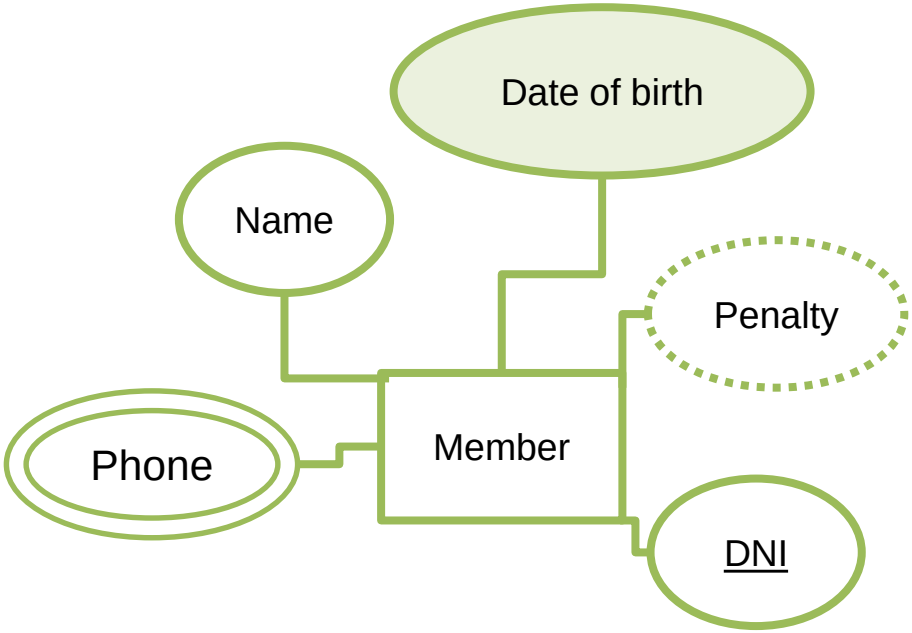
Derived: Can be computed from the values of the other attributes



Composite: Can be splitted into simplier attributes

Examples

Multivalued:

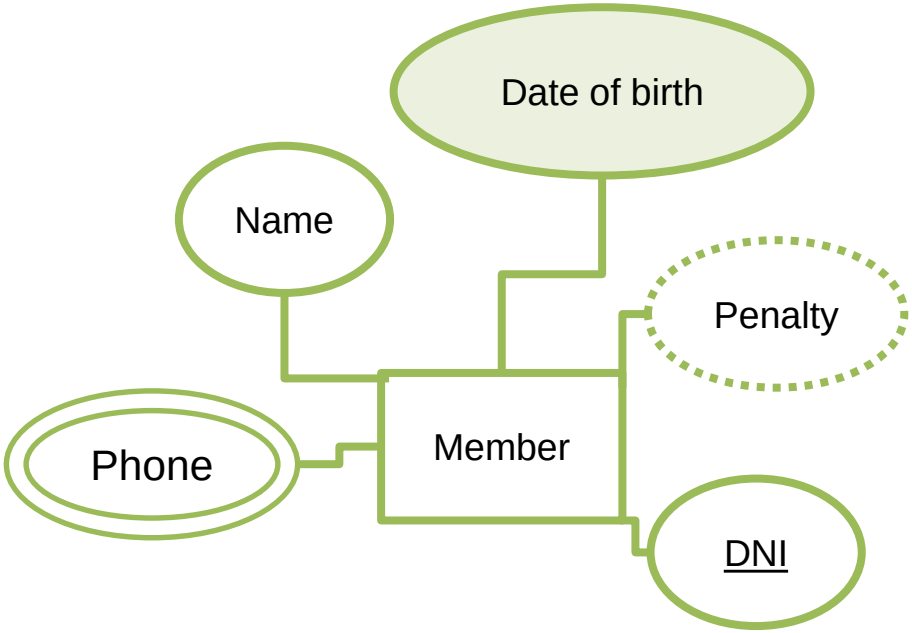


A member can have more than one phone number. Dani Alves has 3 phone numbers and Peio has only one

DNI	Name	Surname	Phone
3676373L	Peio	Artola	938373893
4748474P	Dani	Alves	617232066, 623344112, 617734333

Examples

Composite:

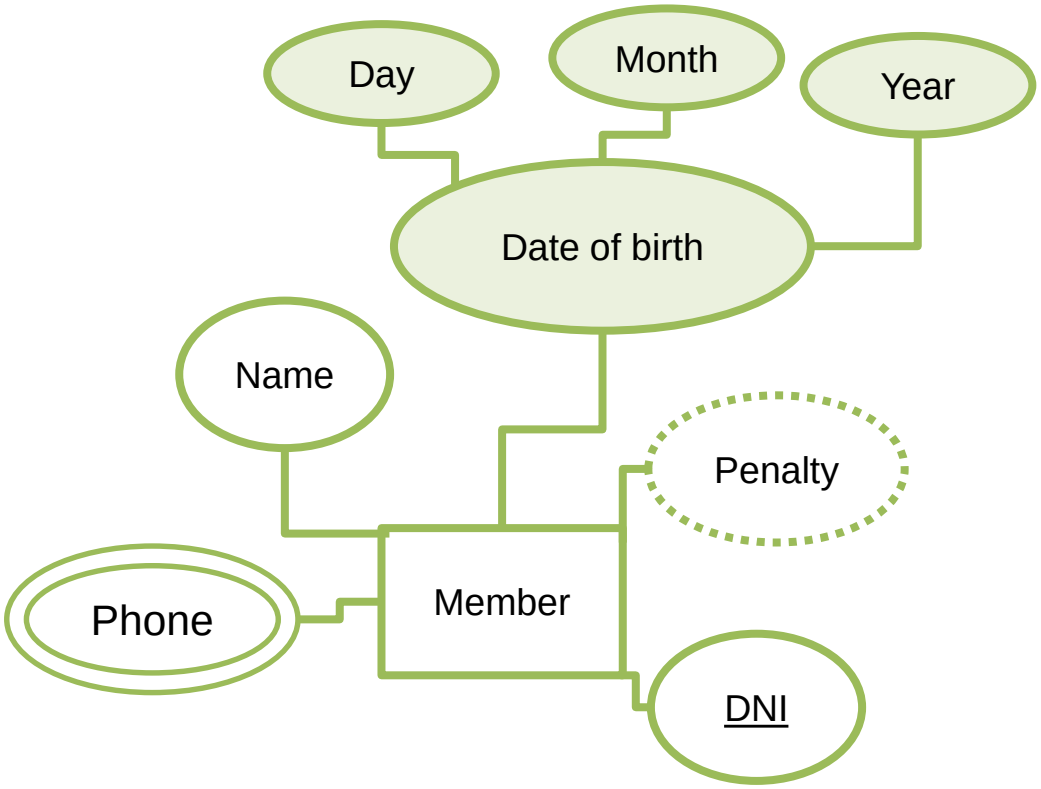


Date of birth can be splitted into day, month and year

DNI	Name	Surname	Phone	Date of birth
3676373L	Peio	Artola	938373893	23-Jan-1965
4748474P	Dani	Alves	617232066, 623344112, 617734333	12-Jun-2001

Examples

Composite:



DNI	Name	Surname	Phone	Birth day	Birth month	Birth year
3676373L	Peio	Artola	938373893	23	1	1965
4748474P	Dani	Alves	617232066, 623344112, 617734333	12	6	2001

Examples. Composite vs Entity

In the composite attributes the pieces are always of the same type. A composite attribute with different domains, surely should be an entity

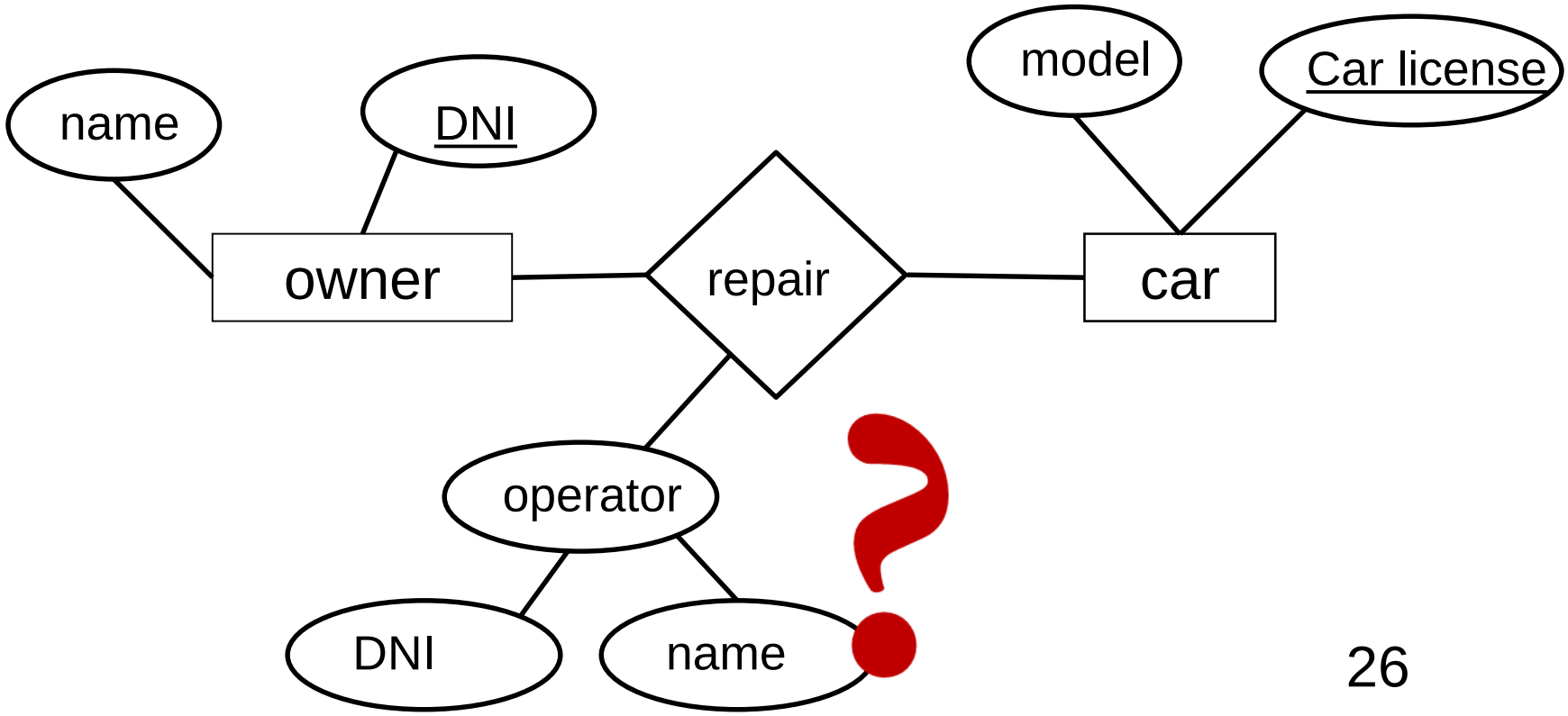
Car Repair Shop:

A car repair shop wants to keep information about its customers, and repairs. We want to keep the ID and name of the clients, the attendant and the car being repaired. We want to save the name, ID and category of the operator. Of the car, the license plate, model, who repairs it and to whom it belongs.

Example. Composite vs Entities

Car Repair Shop:

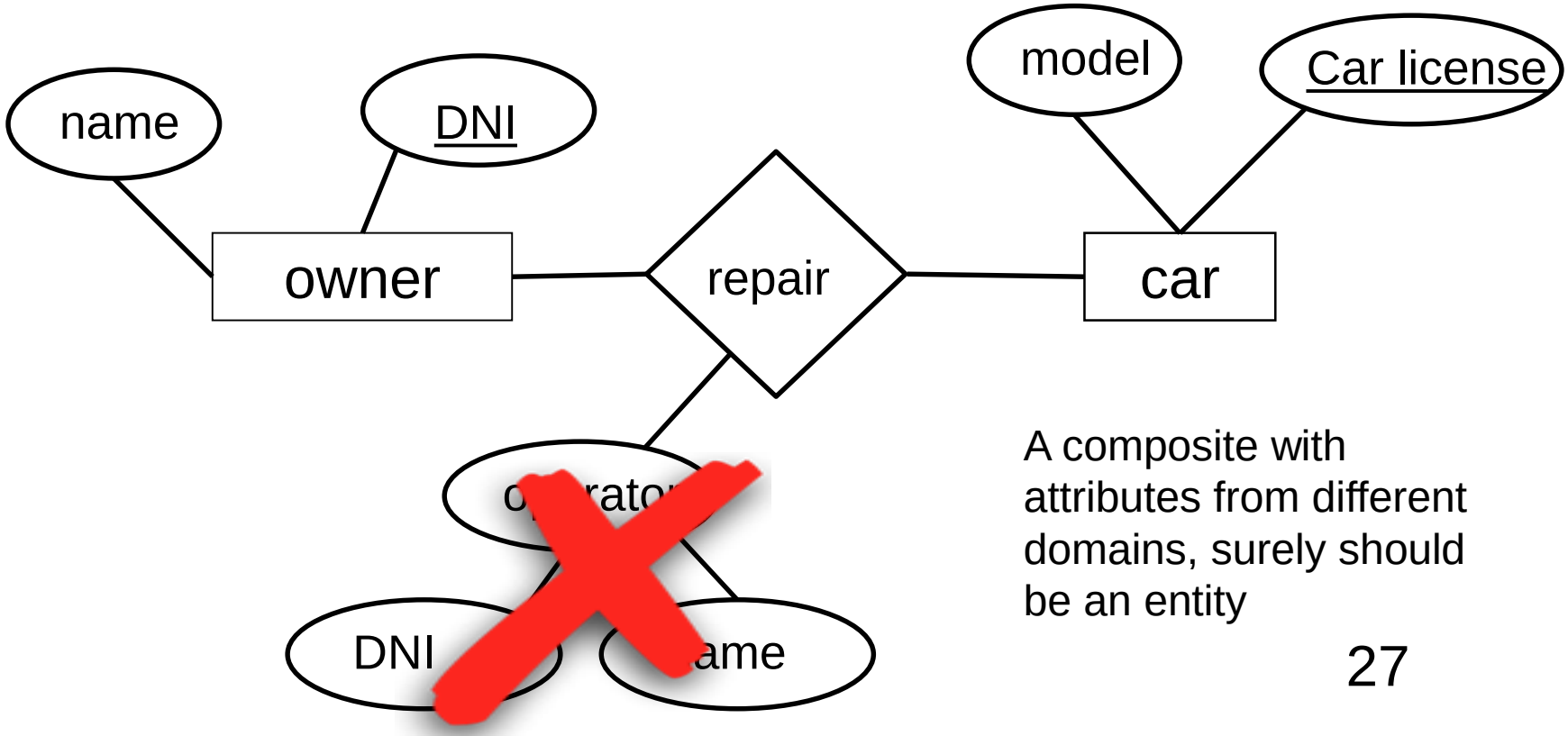
A car repair shop wants to keep information about its customers, and repairs. We want to keep the ID and name of the clients, the attendant and the car being repaired. We want to save the name, ID and category of the operator. Of the car, the license plate, model, who repairs it and to whom it belongs.



Example. Composite vs Entities

Car Repair Shop:

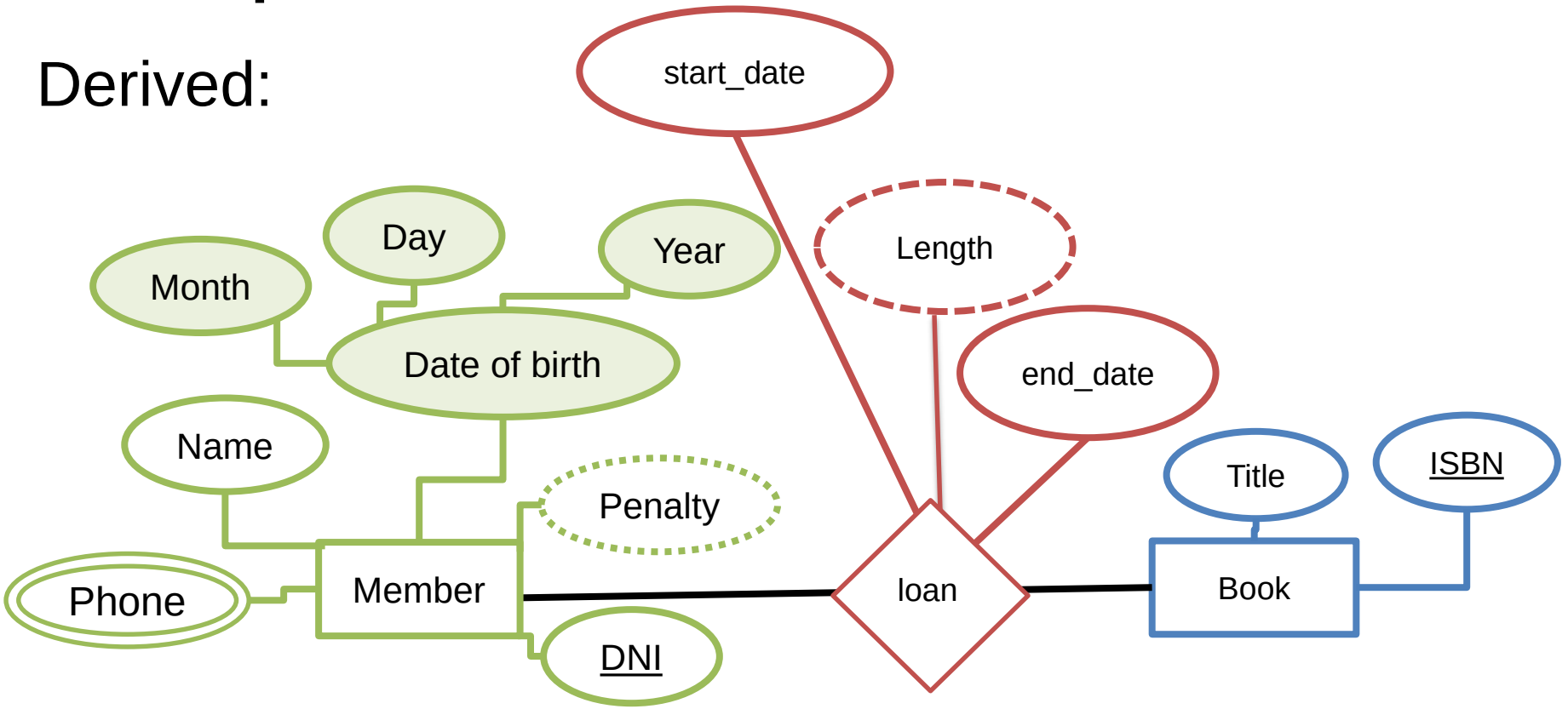
A car repair shop wants to keep information about its customers, and repairs. We want to keep the ID and name of the clients, the attendant and the car being repaired. We want to save the name, ID and category of the operator. Of the car, the license plate, model, who repairs it and to whom it belongs.



A composite with attributes from different domains, surely should be an entity

Examples

Derived:



The penalty can be obtained from the list of overdue books represented by the relation

Length is the difference between start date and end date