

Block 3

RELATIONAL DESIGN

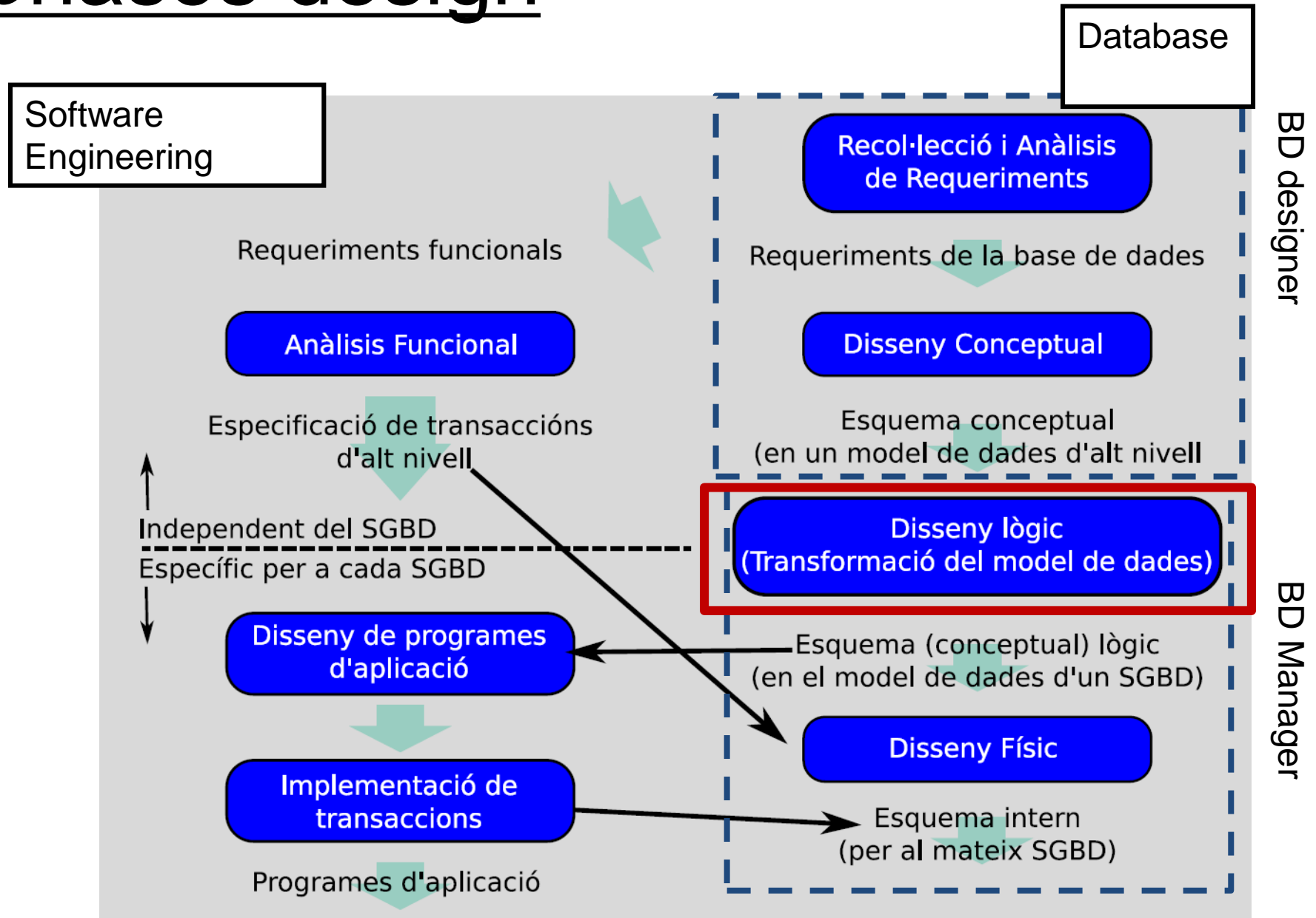
Debora Gil, Oriol Ramos, Carles Sanchez

Contents

1. Introduction
2. ER design to Relational Model

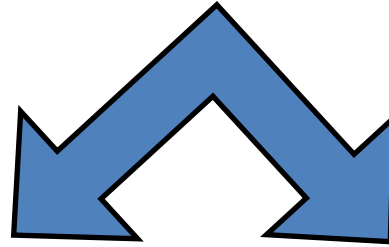
1. Introduction

phases design



Logical design

Conceptual Design (ER
Model)



Databases

Queries



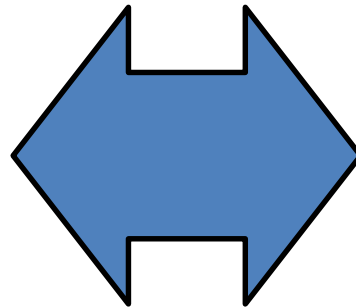
Tables diagram

Software Engineering

Functionalities



Classes UML



1. ER Design Relational Model

1.1 Relational Model

1.2 Tables Diagram from an ER Design

1.3 Examples

1.1 Relational Model

Definition

- Data high level description that guarantees its integrity.
- Is the unique theoretical model implementable with basic rules
- First defined in June 1970 by Edgar Codd, of IBM's San Jose Research Laboratory.
- Has become the predominant type of database, although another models exist.

Components

Data structure. Description by Relations / Tables.

Integrity rules. Rules to ensure the queries consistency

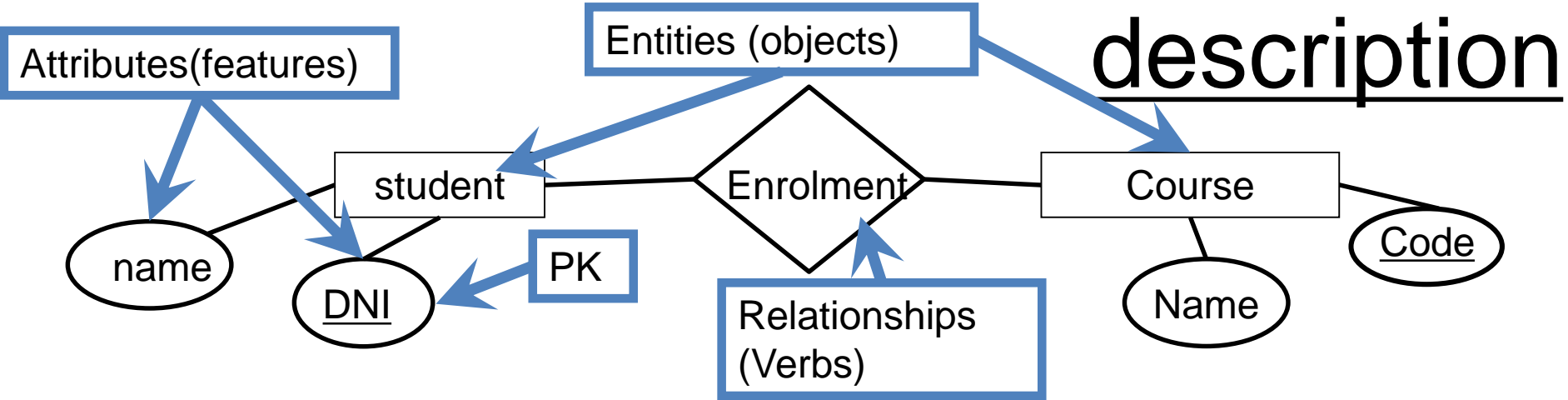
Entity (Table, PK)

Referential (relations between tables, FK)

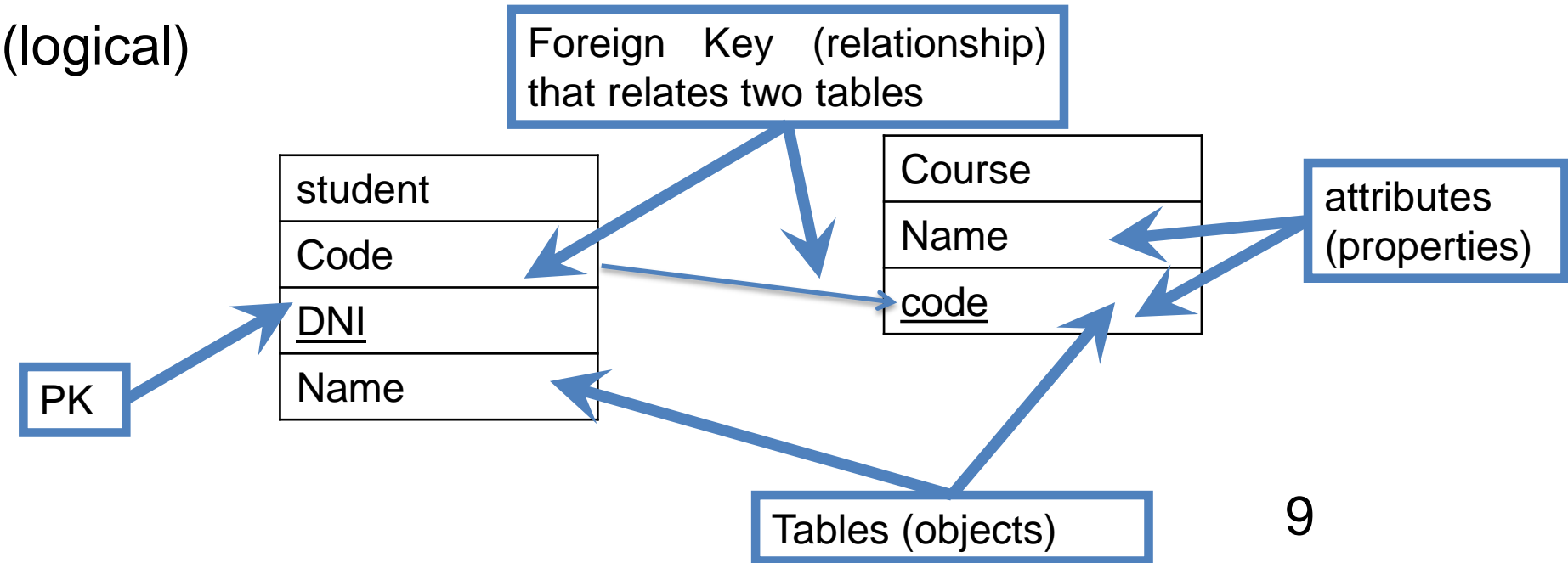
Operators to access data by content: Relational algebra

Relational calculus

ER Diagram (abstract)



Relational model (logical)

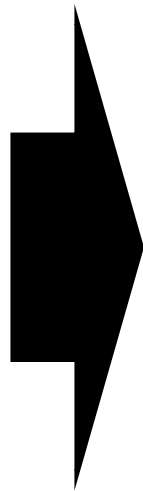


Relational components

Objects

Features

Related information



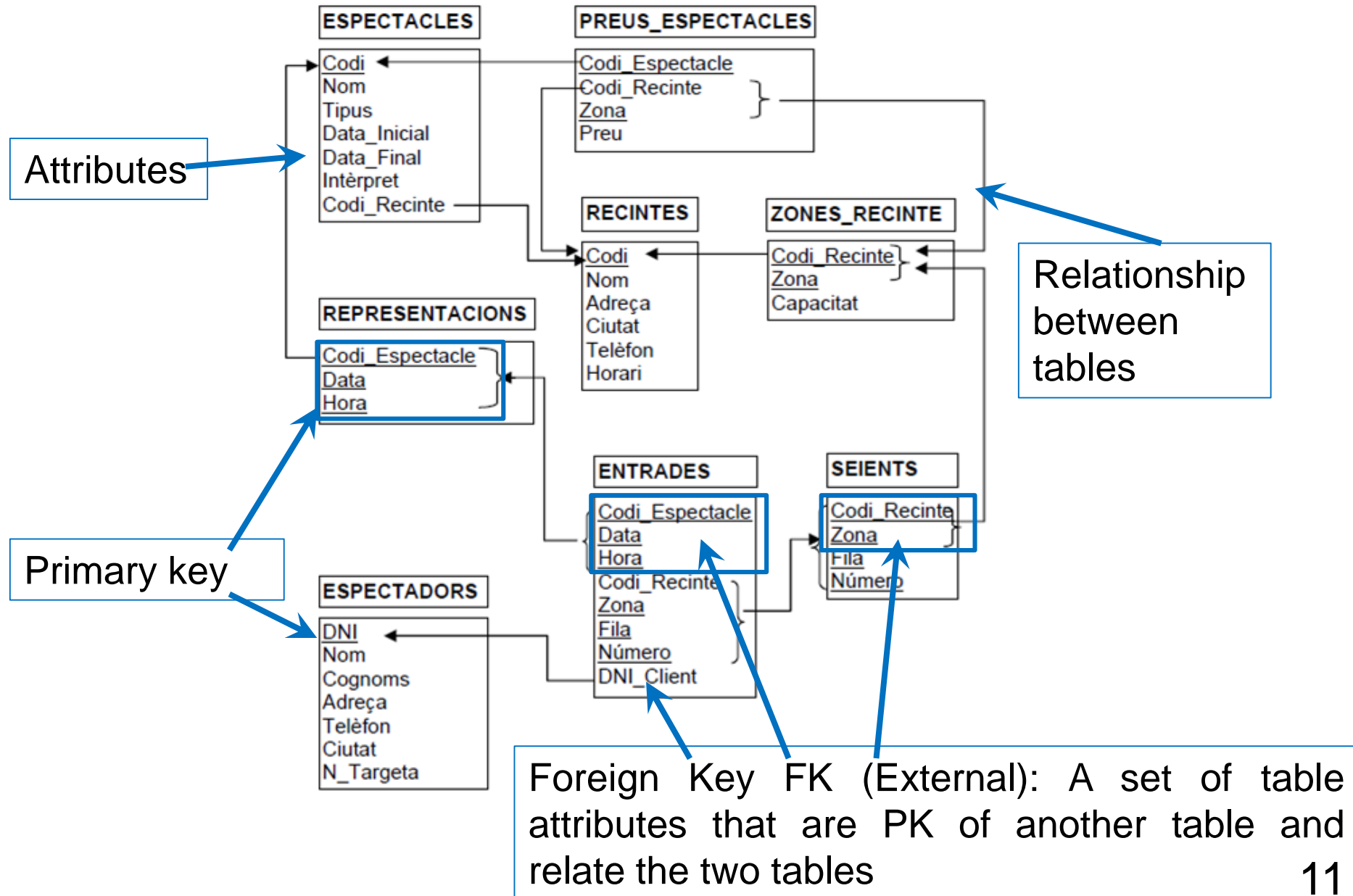
Relation (Table)

Domain (allowed values of attributes)

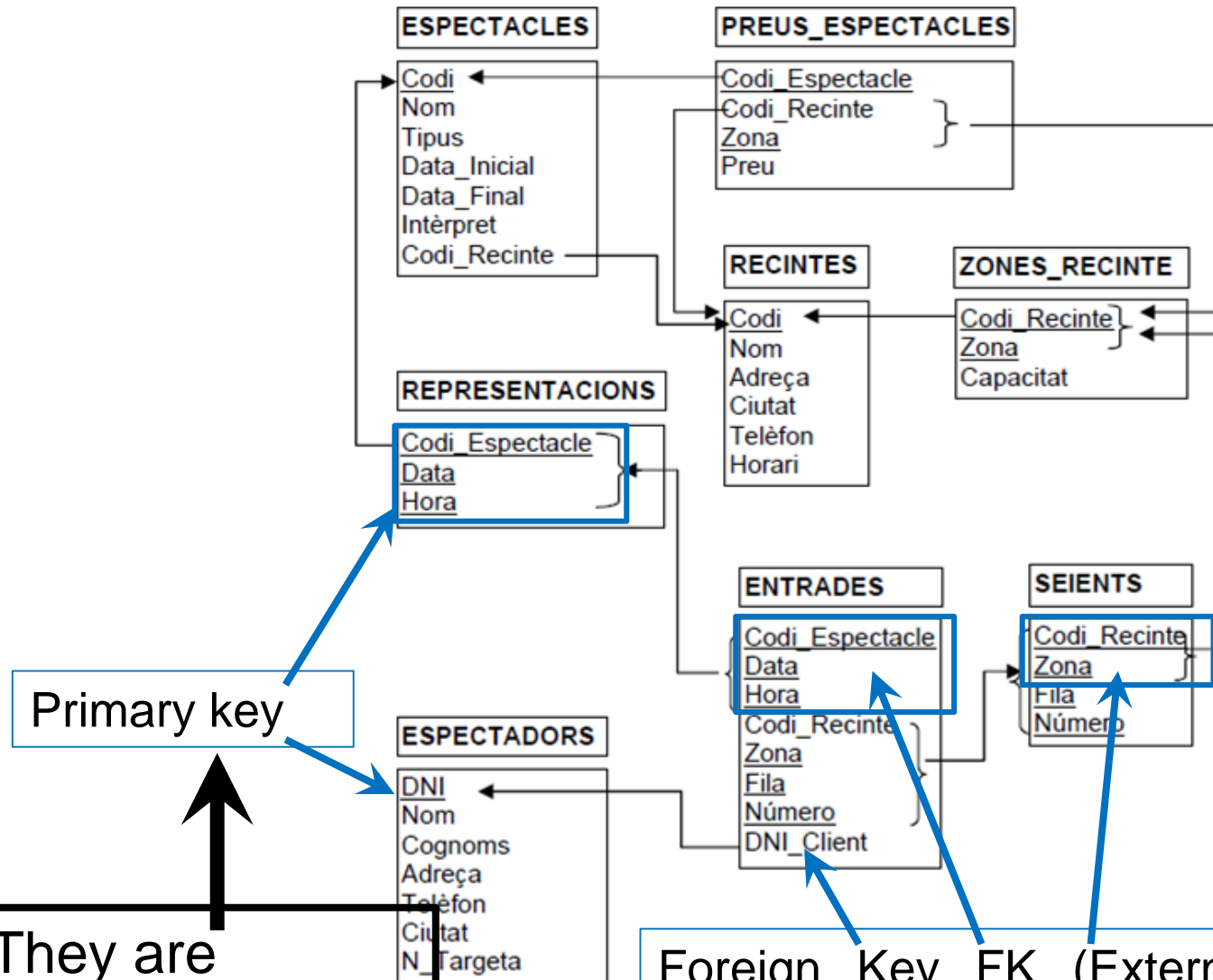
Relational database (set of referenced tables)

Database

Collection of related tables
(referenced, linked) together



Database

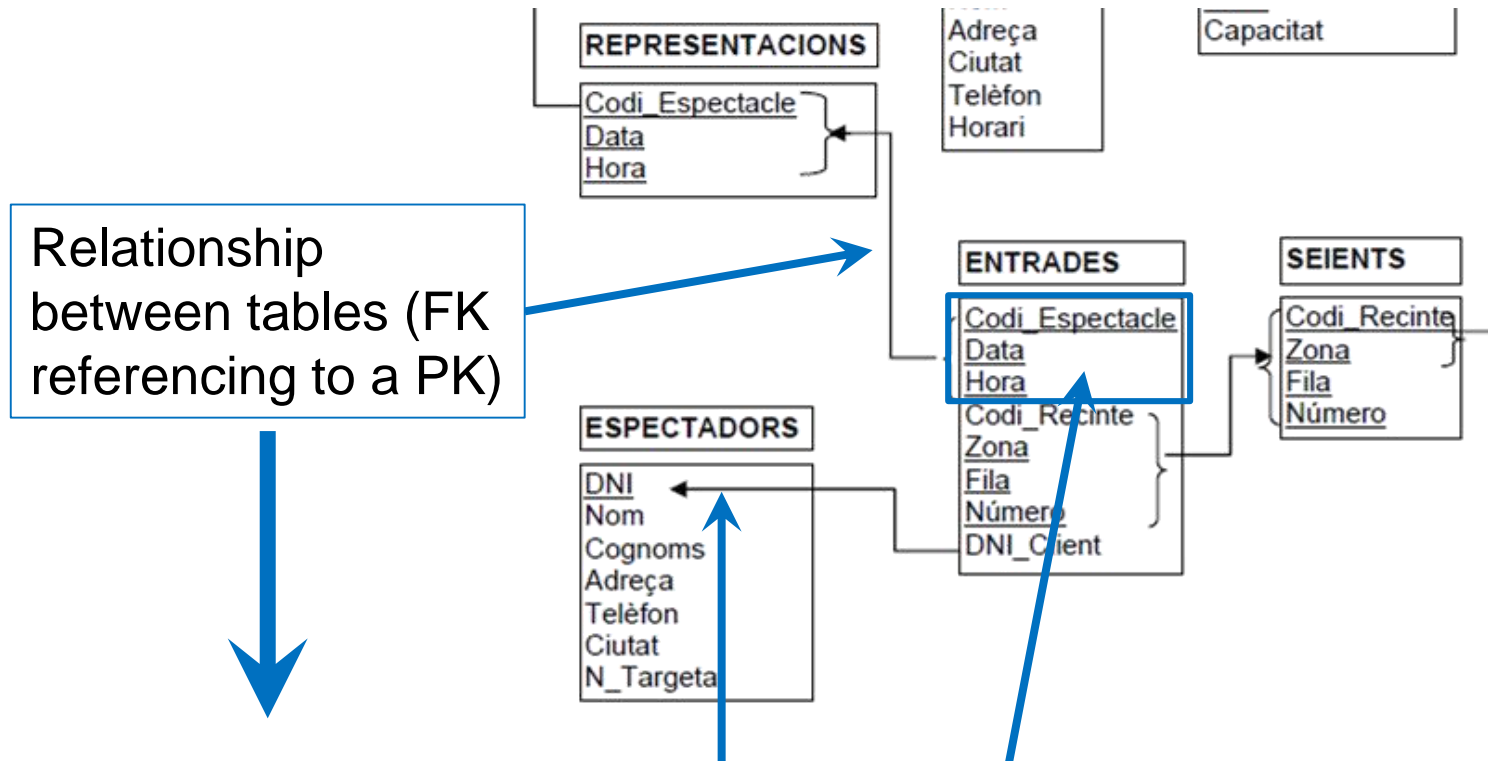


Primary key

They are essential for integrity

Foreign Key FK (External): A set of table attributes that are PK of another table and relate the two tables

Database Integrity

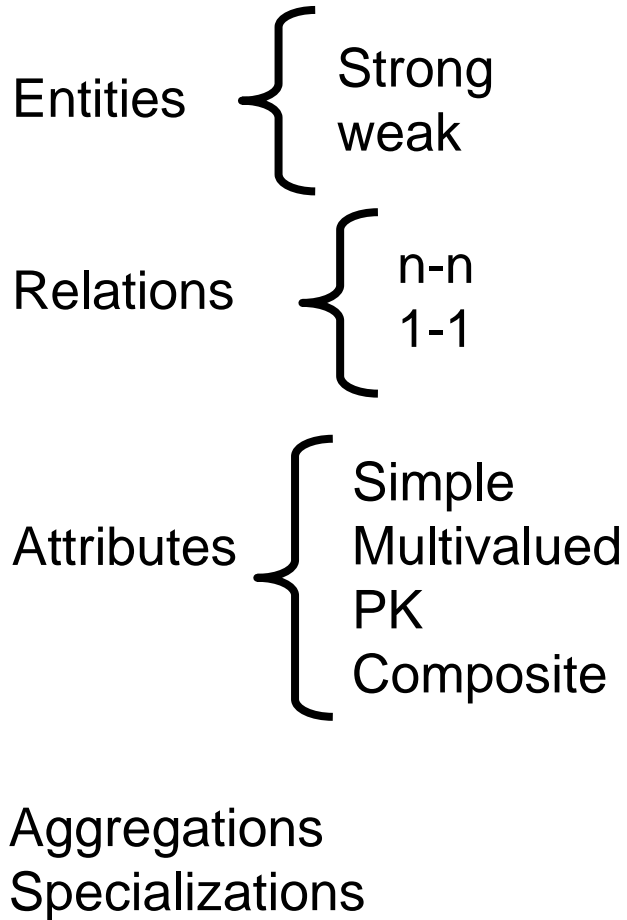


FK contains all the PK attributes of the PK referred (pointed to)
The arrow must always point to PK to ensure propagation changes

1.2 Tables diagram from an ER Design

Structures

ER design



Tables diagram

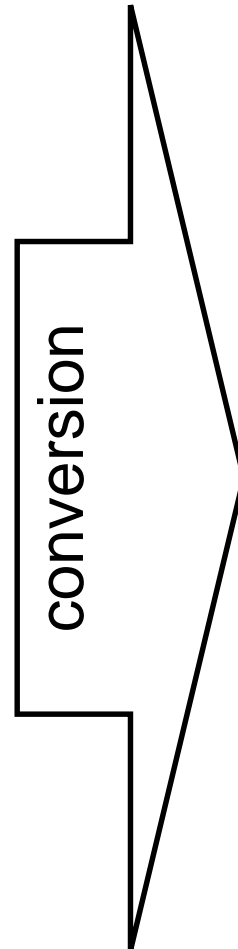


Table (header with the attributes, PK)

Foreign Key FK (references)

Attributes (Simple)

Conversions

- Entities → Table

Strong: Entity PK **Weak:** Weak entity PK + FK with value strong PK

- Relationships

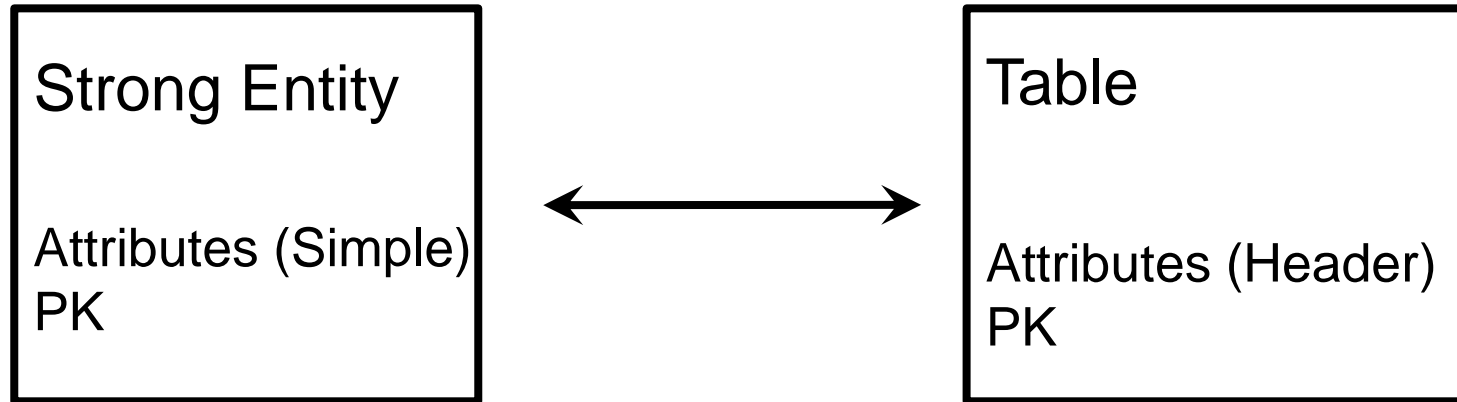
- **Binary 1-n:** FK on n-side entity with PK value on 1-side entity
Ex: Table A - PK A Table B - PK B + FK A
- **Binary 1-1:** FK on 1-side entity with PK value on 1-side entity
- + FK uniqueness constraint
Ex: Table A - PK A + FK B Table B - PK B + FK A
- **Binary n-n:** Table with PK union participating entities PK's +FK for each participating entity
Ex: Table A - PK A Table B - PK B Table C – PK(FK A + FK B)
- **Ternari:** Table with PK union n-side participating entities PK's +FK for each participating entity

- Attributes

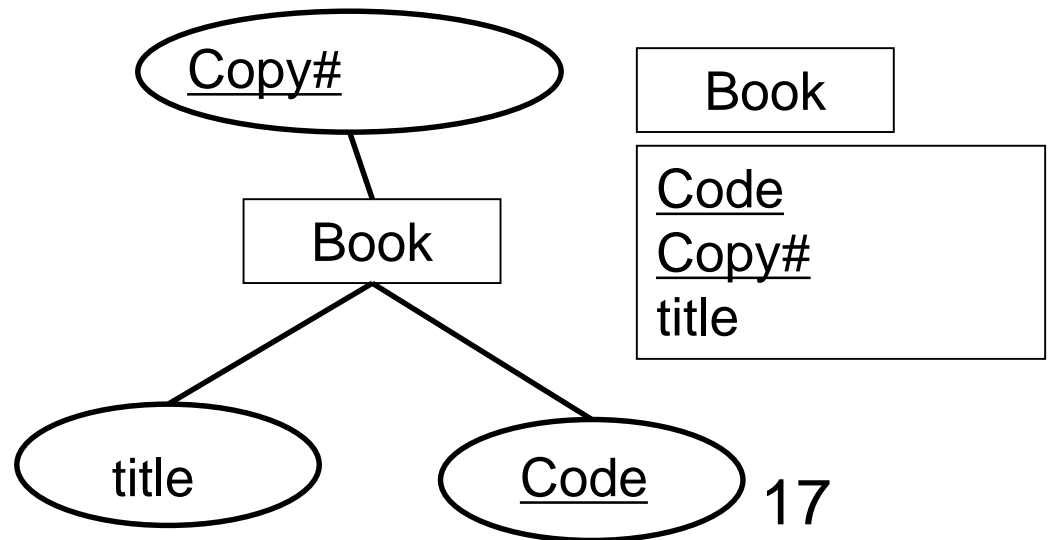
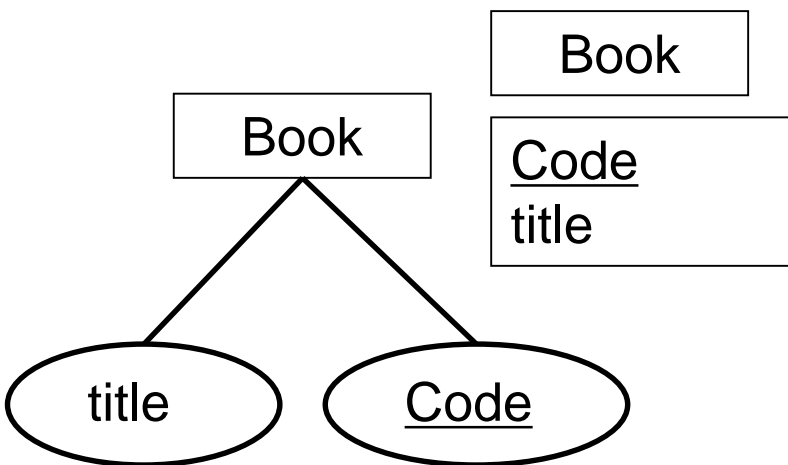
Simple, composite → attributes

Multivalued: Table with attribute PK and entity PK + FK with value entity PK

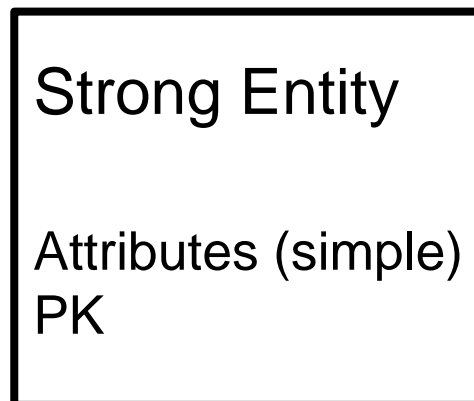
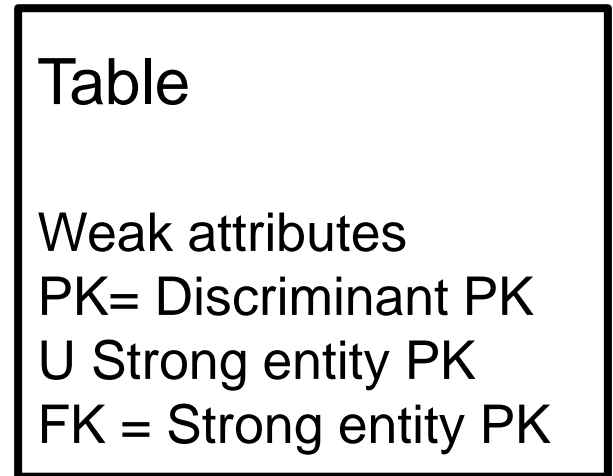
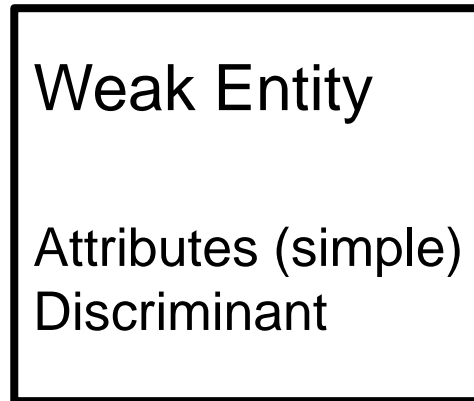
Strong Entities



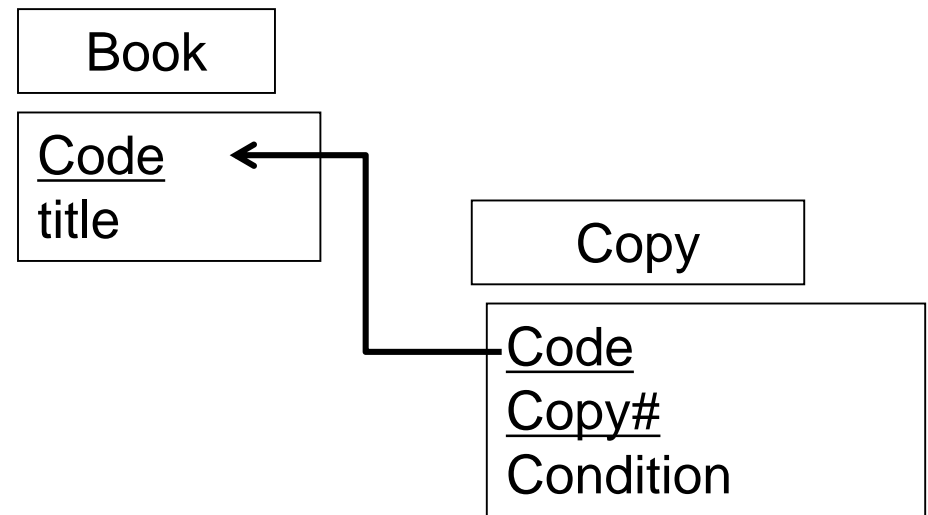
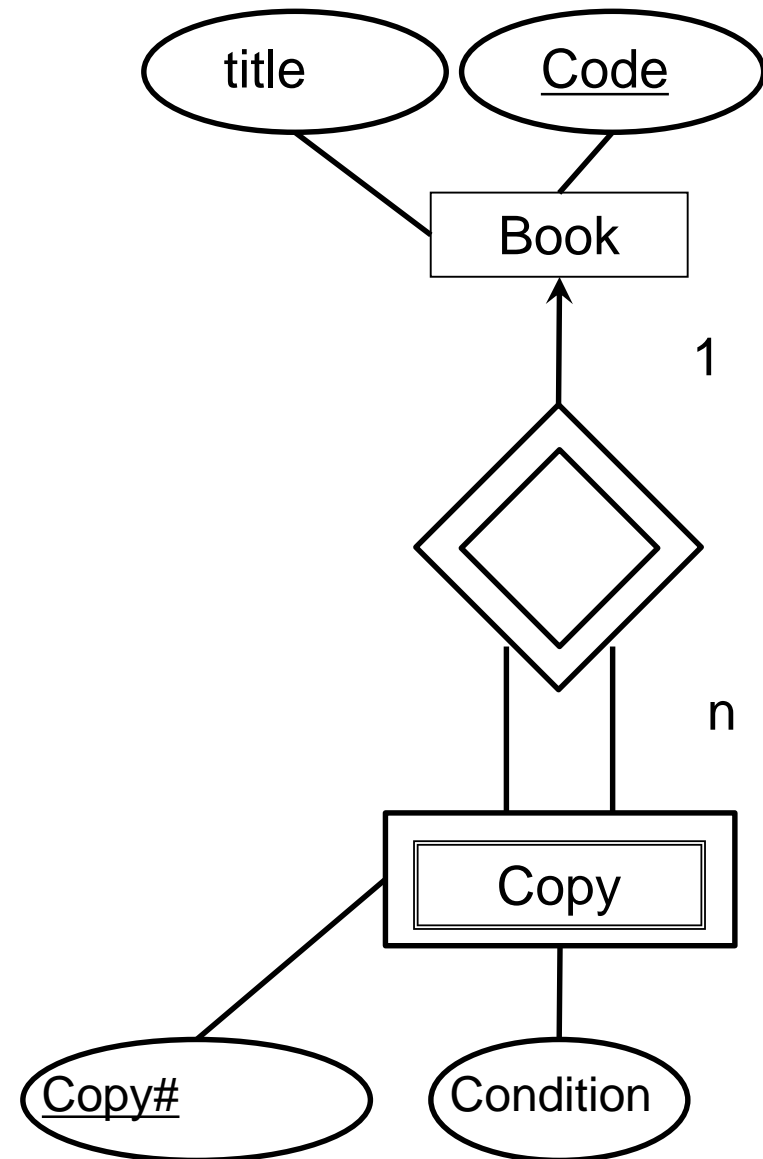
Example:



Weak Entities



Example



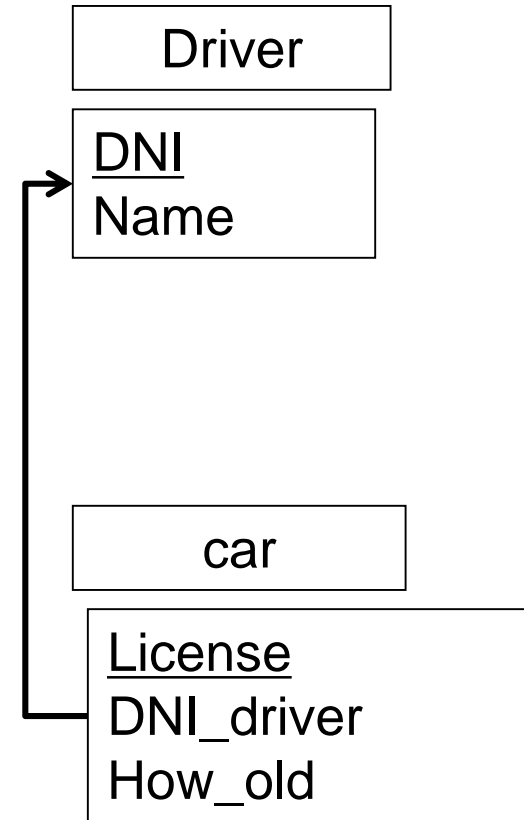
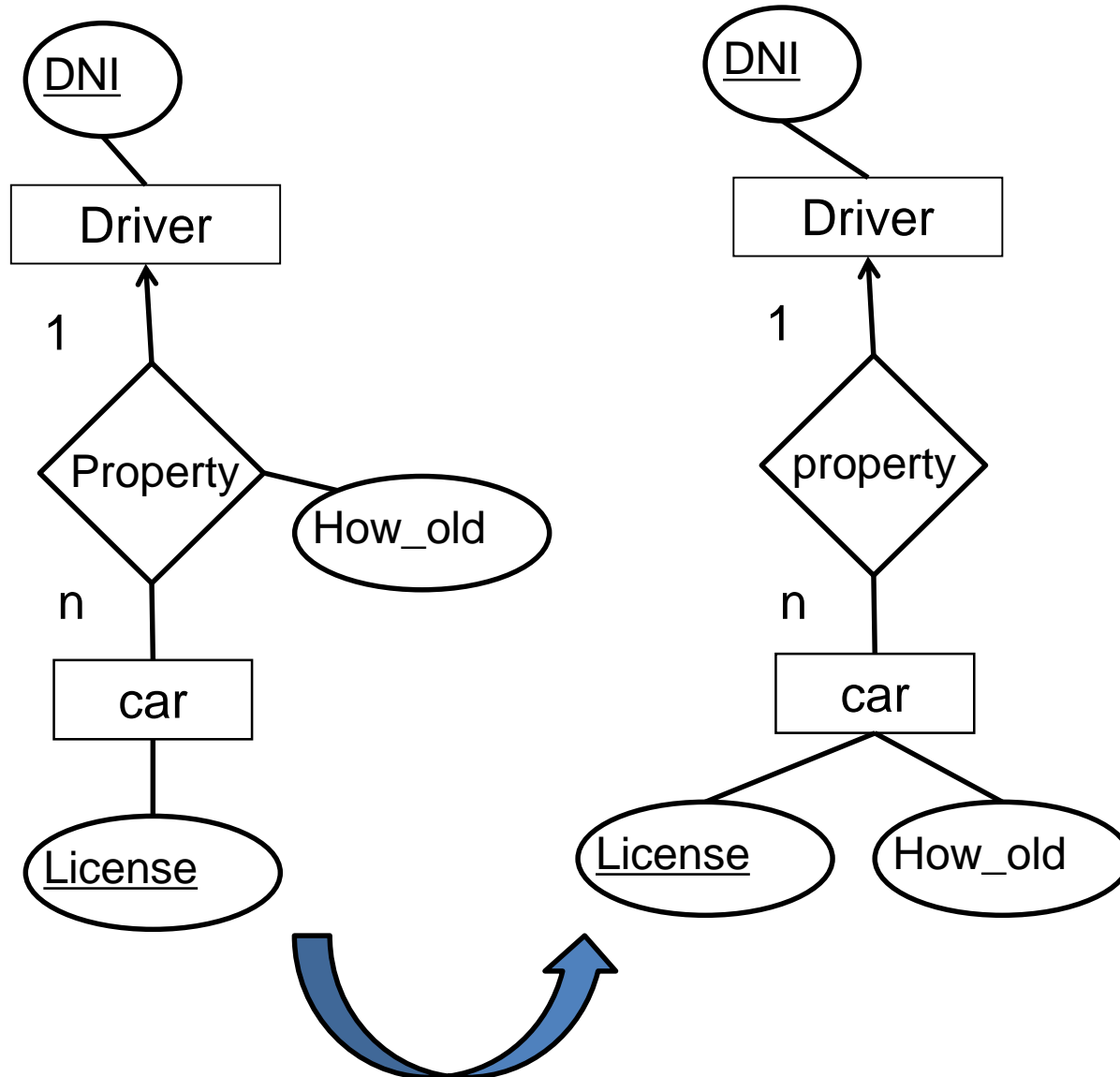
Relationships

They become a table or a relationship between interrelated entities depending on the cardinality:

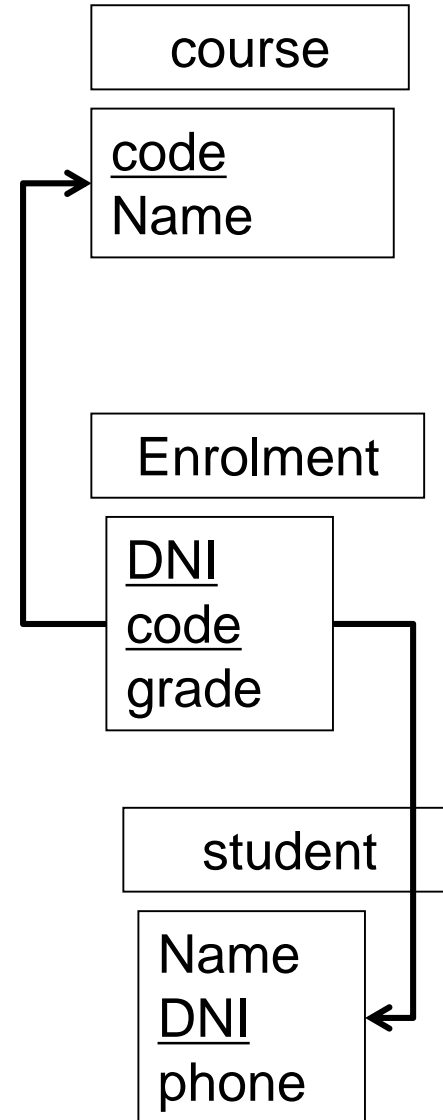
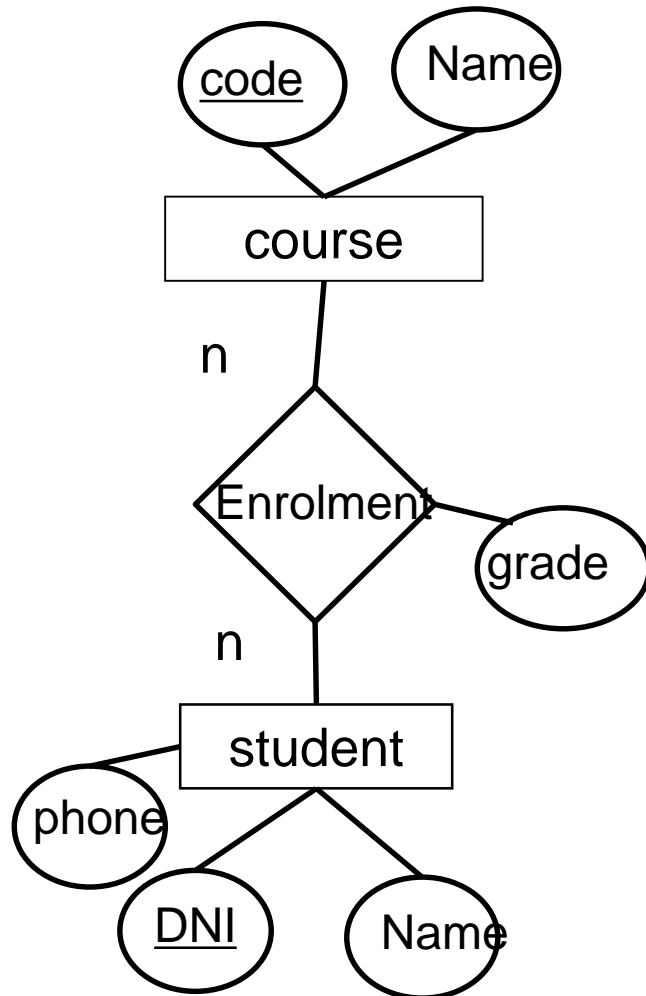
Cardinality 1-n: Attributes are passing to n-side entity and the interrelationship is converted to relationship between tables (FK to the n-side entity)

Cardinality n-n: Relationship is converted to a new table with FK linking two related tables. The new table PK is the unión of the related tables PK

Example. cardinality 1-n



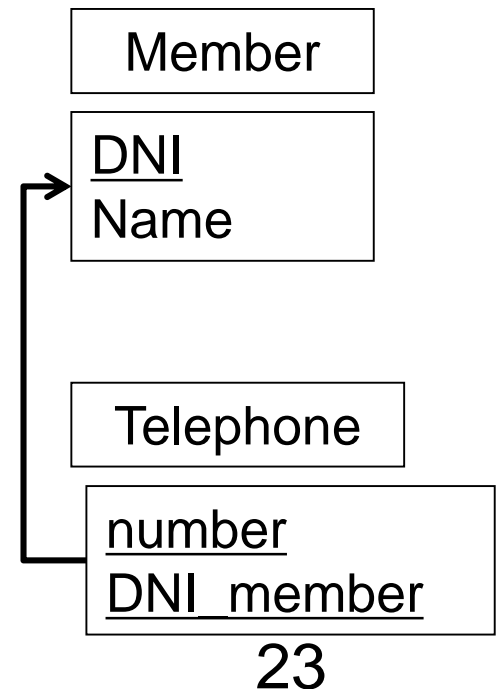
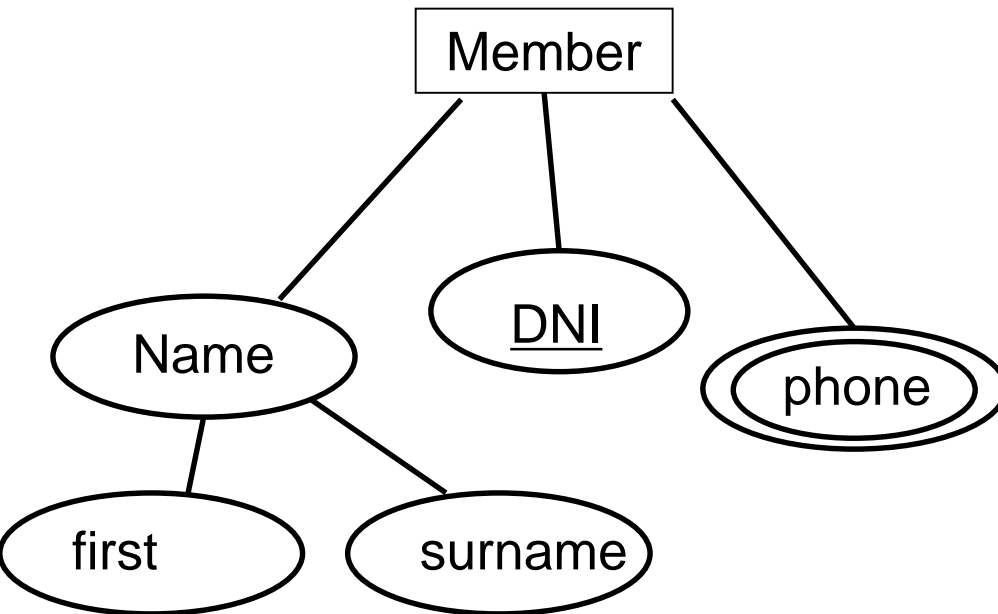
Example. cardinality n-n



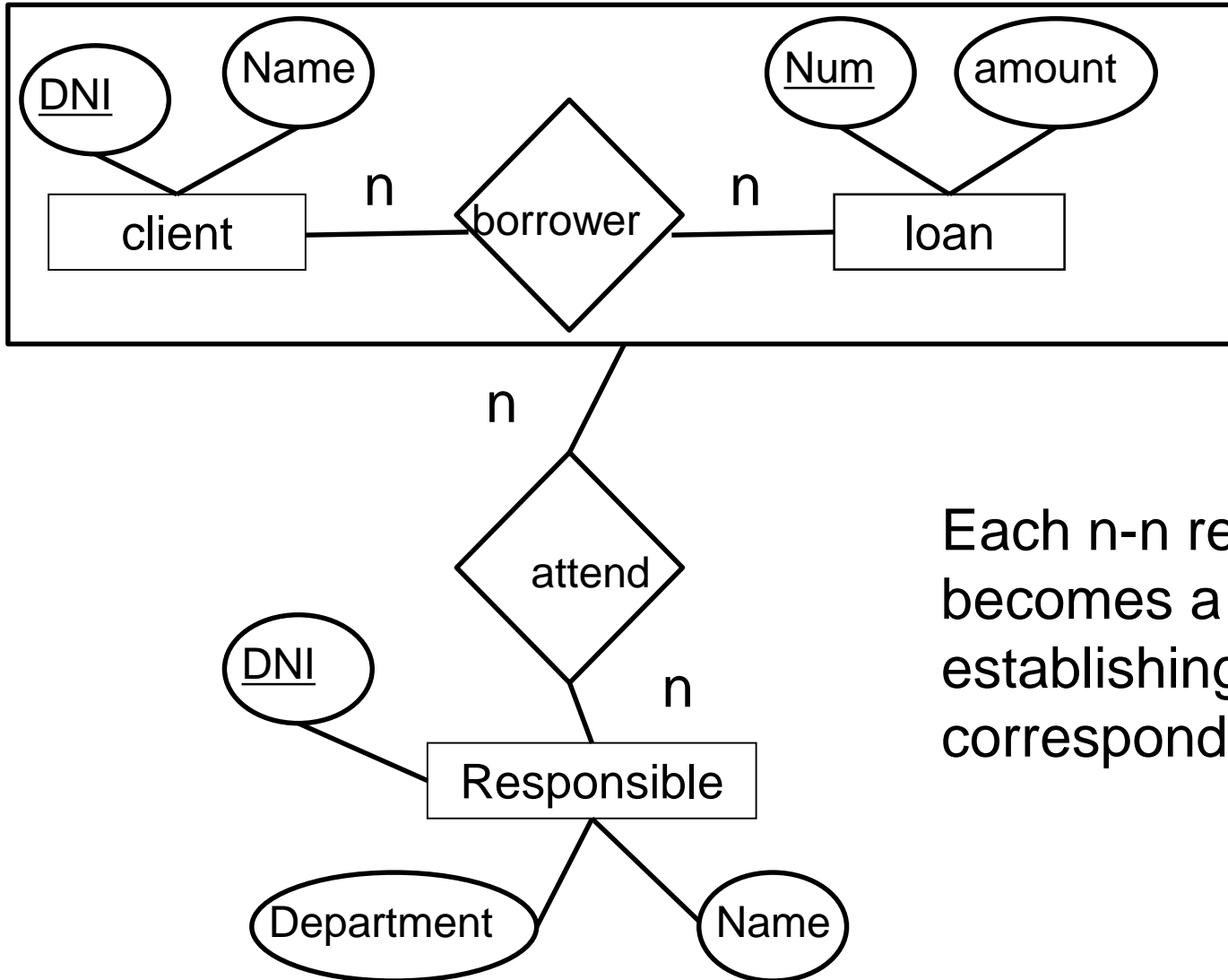
Attributes

Simple and composite attributes have direct translation into simple attributes with a certain domain.

Multivalued generate a table with FK:

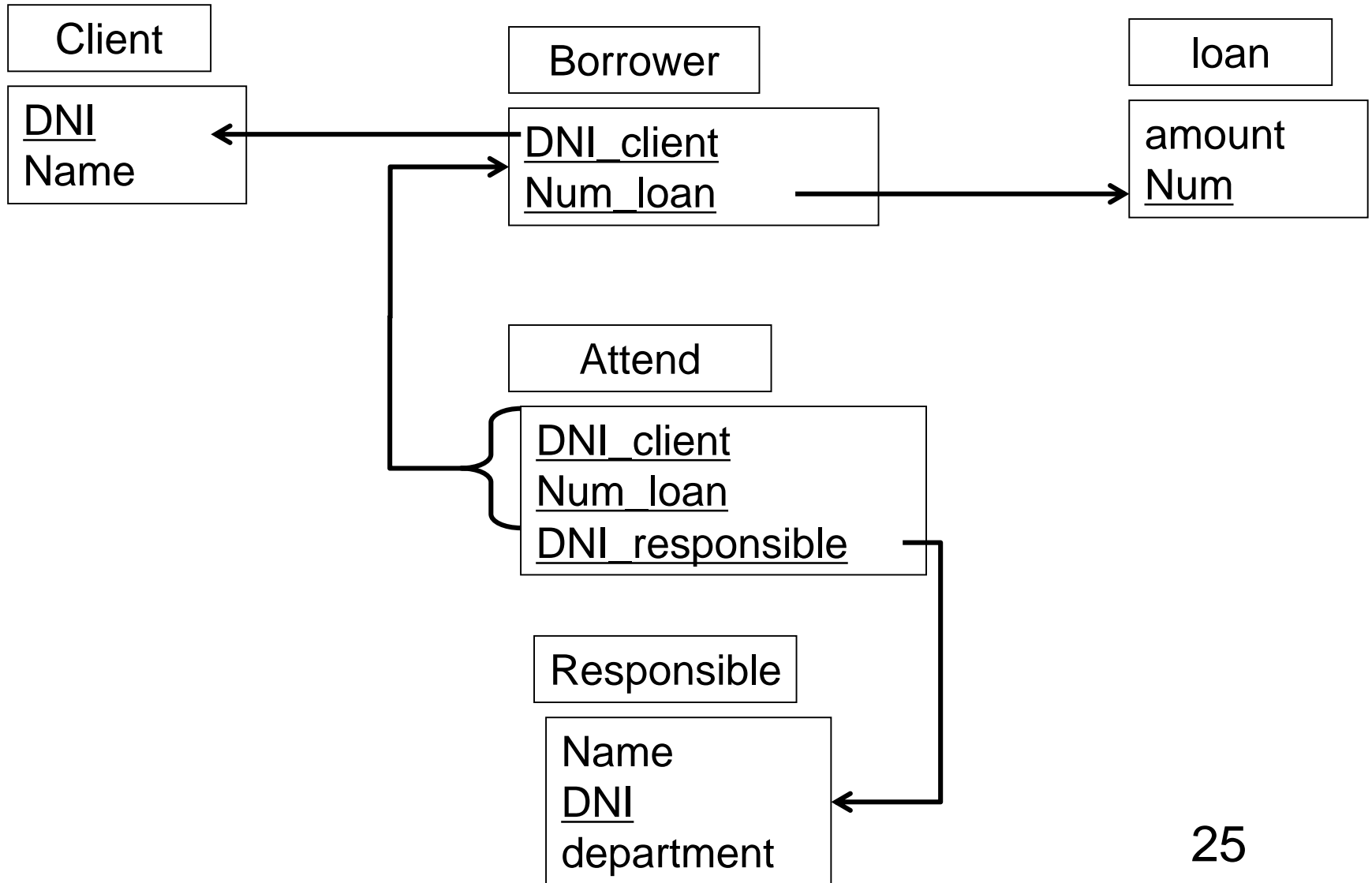


Aggregations

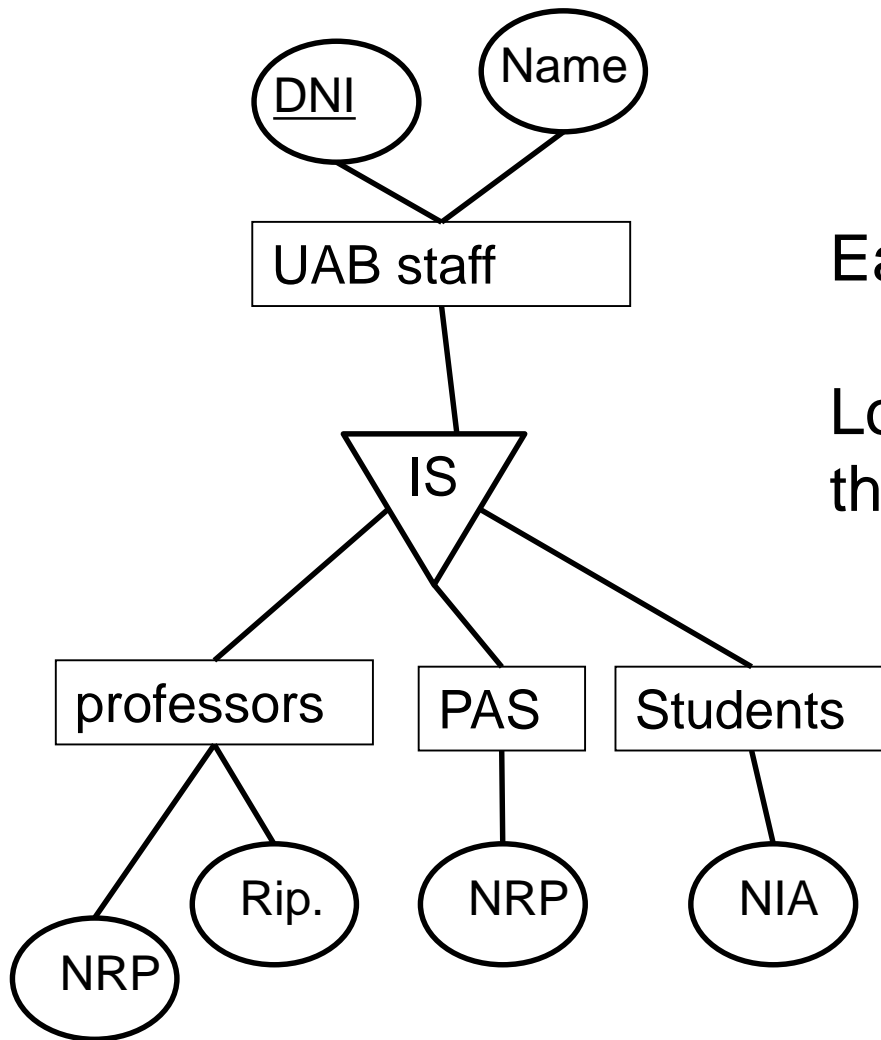


Each n-n relationship becomes a table with FK establishing the corresponding links

Example



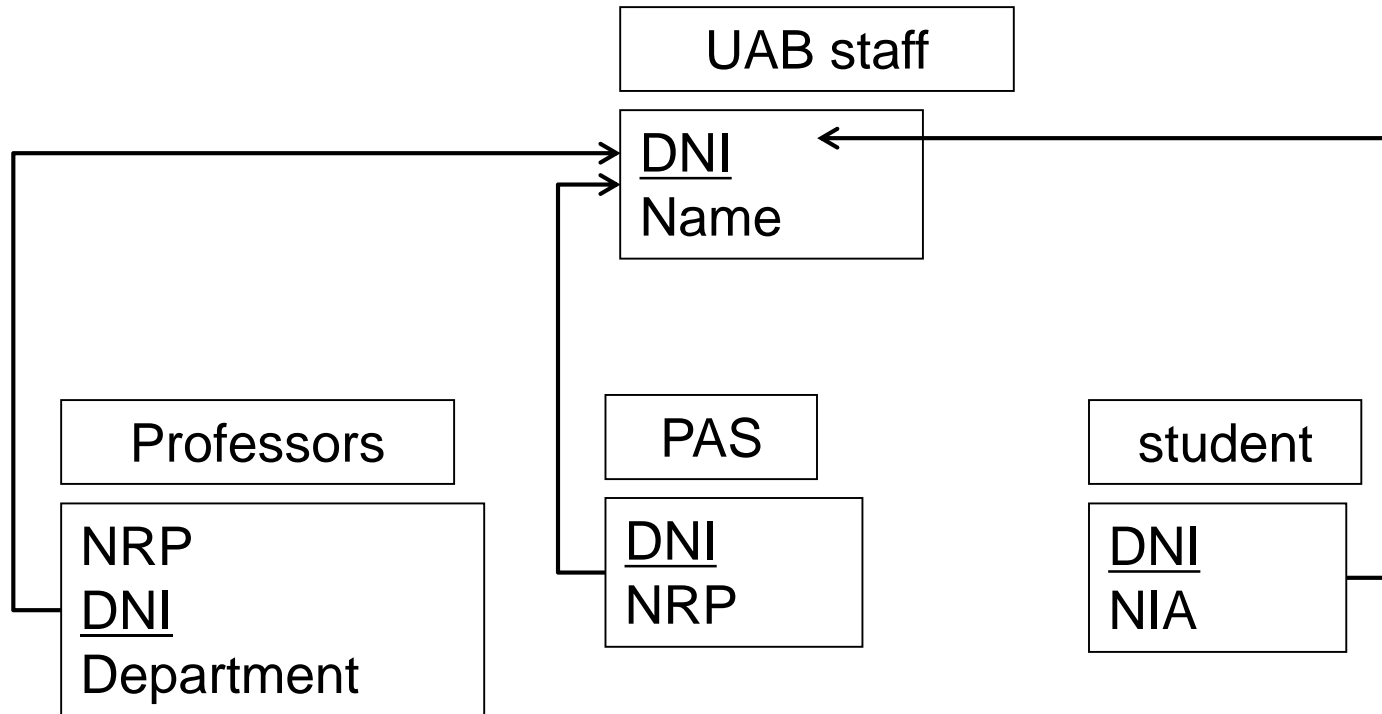
Specializations



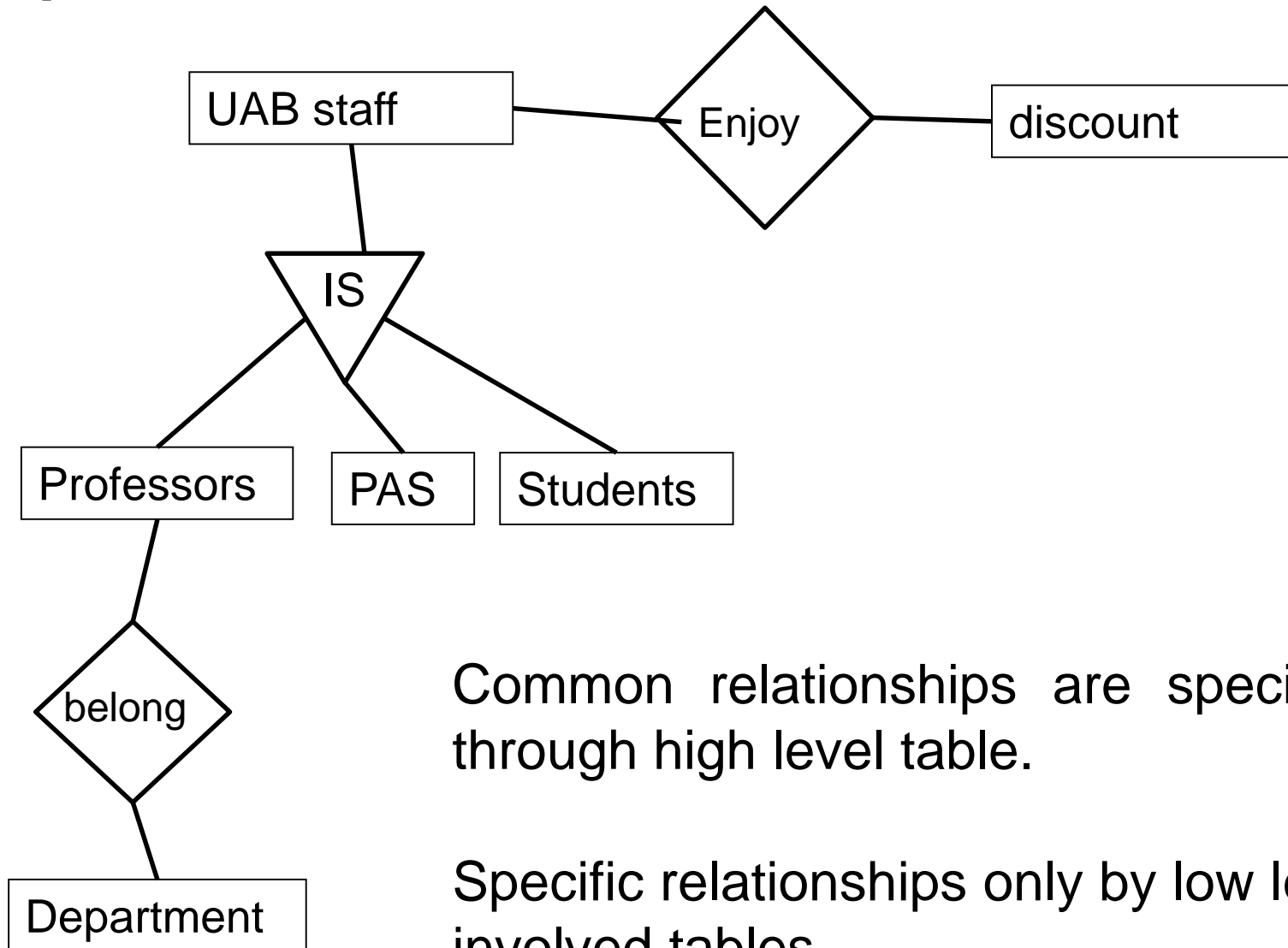
Each entity becomes a table.

Low level entities have FK linking them to high level entity

Example



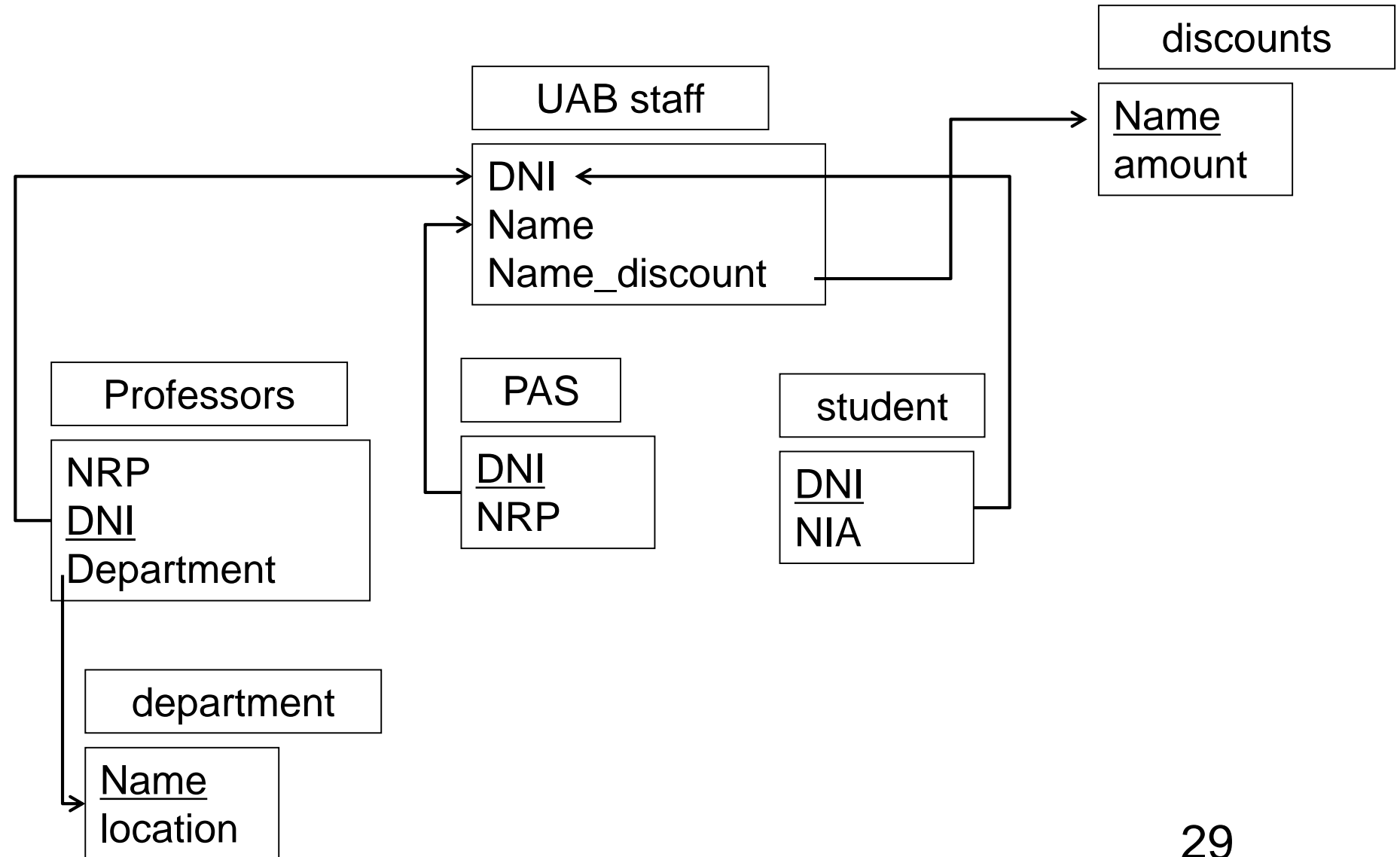
Specializations



Common relationships are specified through high level table.

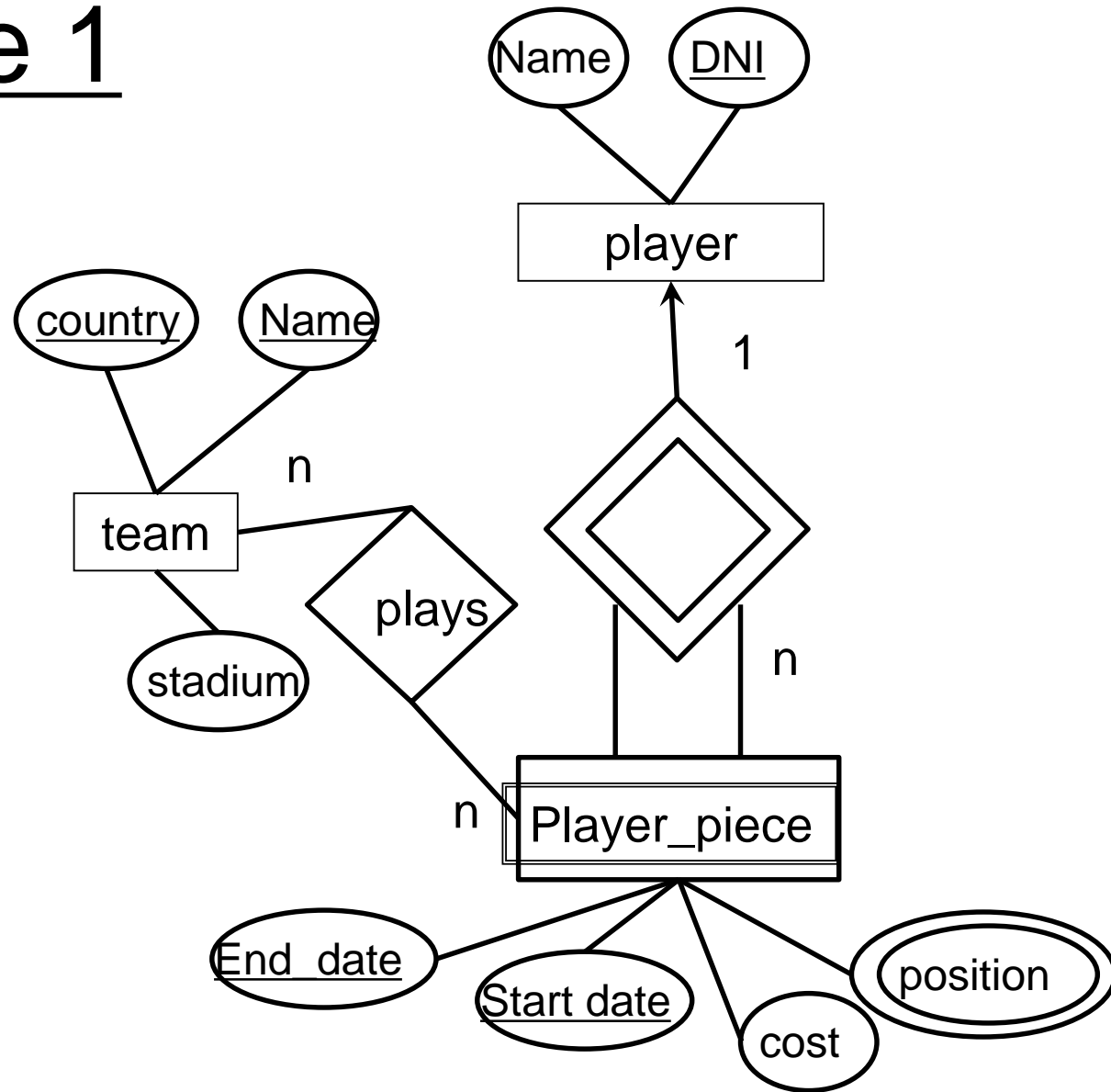
Specific relationships only by low level involved tables

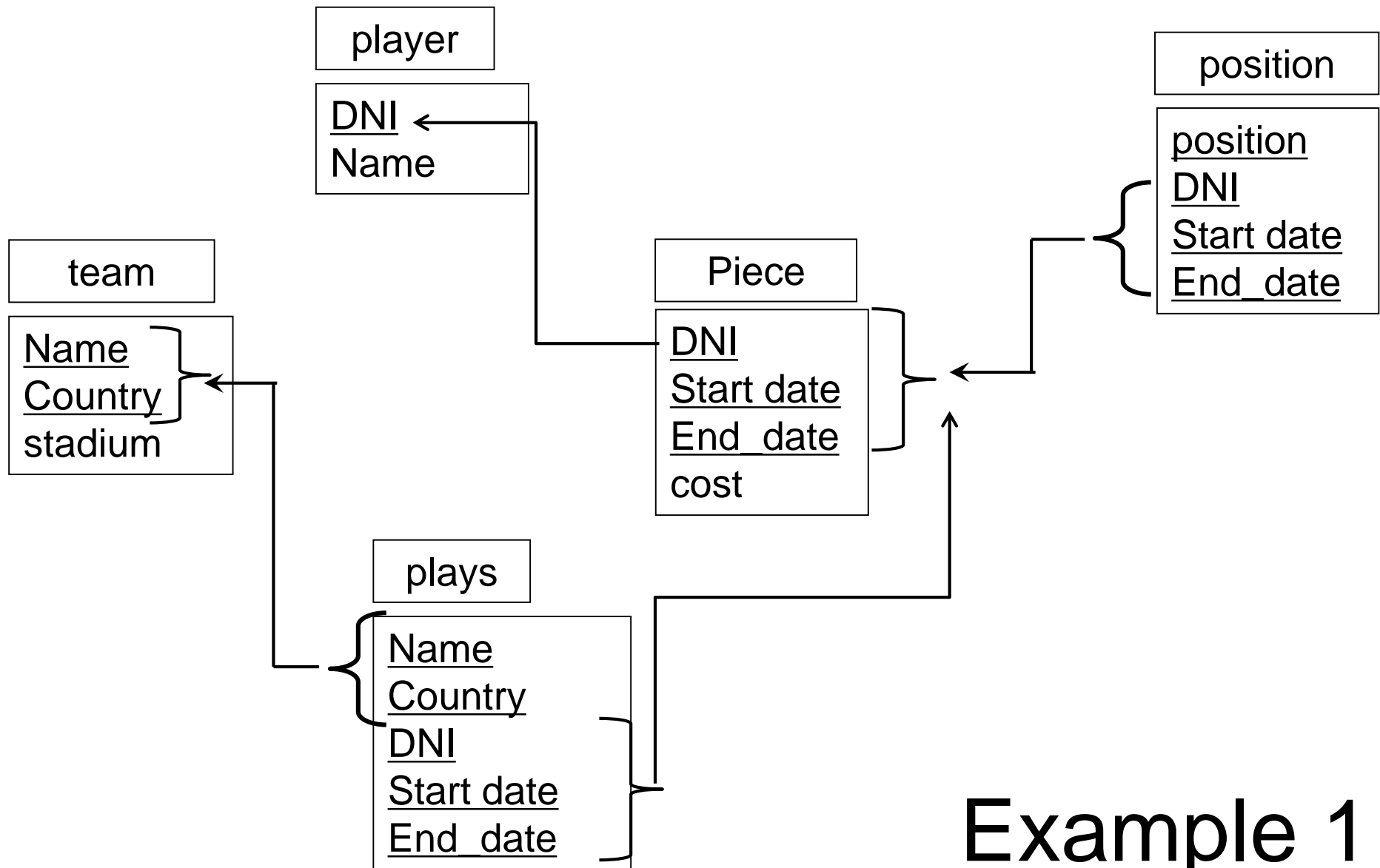
Example



1.3 Examples

Example 1





Example 2

