

# Clase 1 - ER - BDD

## I. Introducción a las Bases de Datos

### 1. Definiciones y Conceptos Básicos

- **Base de Datos:**

Es un conjunto de datos relacionados que representan algún aspecto del mundo real. No se trata simplemente de acumular datos, sino de mantener información coherente y con un significado definido para un grupo de usuarios.

- *Dato vs. Información:* Un dato es un hecho que se puede almacenar (por ejemplo, "100 gramos de chipá tienen 350 calorías"), mientras que la información es el conjunto organizado y contextualizado de esos datos.

- **Proposiciones en Bases Tradicionales:**

En las bases tradicionales se almacenan proposiciones (enunciados verdaderos) que describen hechos, como "Juan Martín Del Potro ganó el US Open en 2009".

- **Bases de Datos No Tradicionales:**

Además de datos textuales o numéricos, hoy existen bases que manejan imágenes, audio, video, datos geoespaciales o vectores de alta dimensión, aunque en el curso se hace énfasis en las tradicionales.

### 2. Sistema Gestor de Bases de Datos (SGBD)

- **Definición:**

Conjunto de programas que permiten a los usuarios crear, mantener y manipular bases de datos (en inglés, DBMS).

- **Funciones Principales:**

- **Almacenamiento y Consulta:** Definición de tipos de datos, estructuras y restricciones; ofrece un lenguaje de consulta eficiente.
- **Integridad:** Controla y garantiza que se cumplan las restricciones impuestas sobre los datos.

- **Seguridad:** Previene accesos no autorizados.
- **Concurrencia:** Permite que múltiples usuarios accedan y modifiquen la base simultáneamente.
- **Recuperación y Soporte Transaccional:** Maneja fallos y asegura que las transacciones se completen de manera confiable.

### 3. Breve Historia de los SGBD y Arquitectura de 3 Capas

- **Evolución Histórica:**
  - **Años 50 y 60:** Transición de ficheros manuales a archivos en cintas magnéticas y discos magnéticos; aparición de los primeros SGBD como IMS (modelo jerárquico) e IDS (modelo de red).
  - **1970 en adelante:** Propuesta del modelo relacional por Codd, estandarización de SQL (SQL-92) y aparición de SGBD comerciales (DB2, Oracle).
  - **Avances Recientes:** Incorporación de tecnologías como Data Warehouses, OLAP, ERPs y, más recientemente, bases distribuidas y NoSQL.
- **Arquitectura de 3 Capas (ANSI-SPARC):**
  - **Nivel Interno:** Se encarga del almacenamiento físico y la forma de acceso a los datos.
  - **Nivel Conceptual:** Define la semántica y estructura lógica de los datos (tipos, operaciones y restricciones), independiente del almacenamiento físico.
  - **Nivel Externo (o de Vistas):** Representa esquemas adaptados a grupos de usuarios, mostrando cómo perciben los datos.

---

## II. Diseño Conceptual de Bases de Datos: Modelo Entidad-Interrelación (ER)

### 1. Diseño Conceptual

- **Objetivo:**  
Especificar de manera abstracta la estructura de la base de datos antes de

su implementación física. Se enfoca en describir qué datos se almacenan, cómo se relacionan y cuáles son las restricciones semánticas.

## 2. Modelo Entidad-Interrelación (ER)

- **Elementos Básicos:**

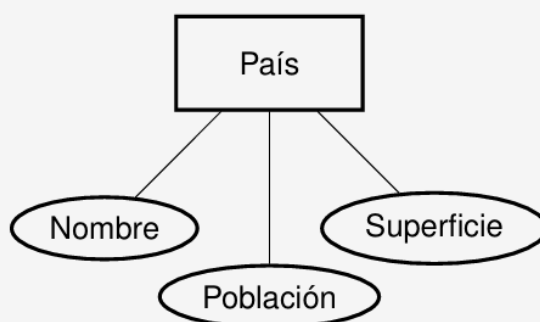
- **Entidades:**

Representan objetos o conceptos del mundo real (por ejemplo, "País", "Futbolista", "Libro"). Cada entidad es una instancia de un tipo de entidad y posee atributos.

- **Atributos:**

Son propiedades que describen a las entidades (ejemplo: en "País" los atributos pueden ser nombre, población y superficie).

El diagrama Entidad-Interrelación representa los tipos de entidades y sus atributos como:

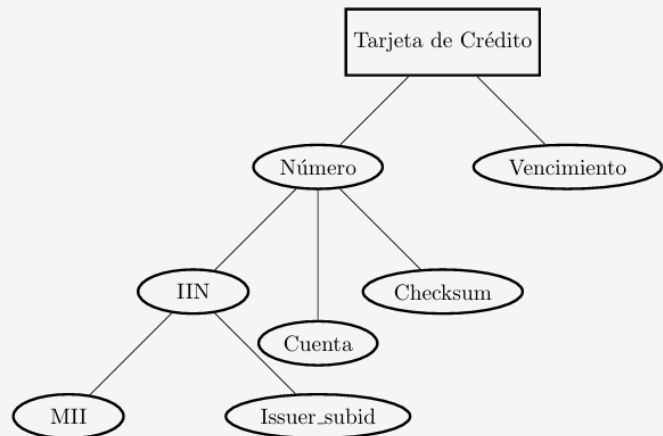


- **Tipos de Atributos:**

- *Simples vs. Compuestos:* Un atributo simple no se descompone, mientras que uno compuesto (como una tarjeta de crédito con número, fecha, etc.) se puede dividir en partes.

## Atributos compuestos vs. simples

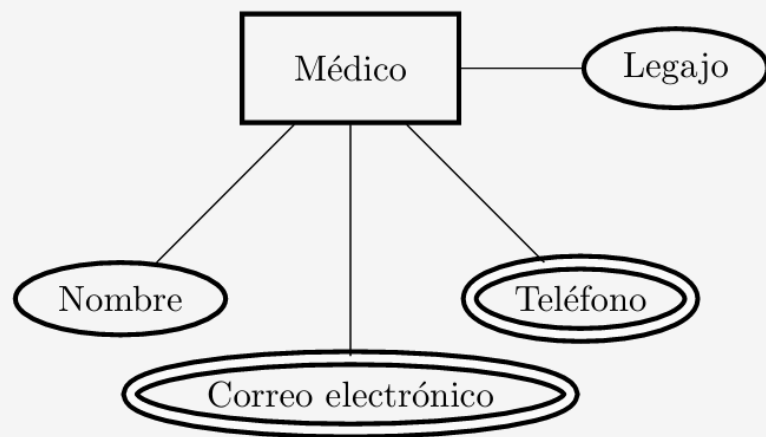
Ejemplo: Tarjeta de crédito



- *Monovaluados vs. Multivaluados:* Un atributo monovaluado tiene un solo valor por entidad; el multivaluado puede tener varios (por ejemplo, varios teléfonos de contacto).

## Atributos multivaluados vs. monovaluados

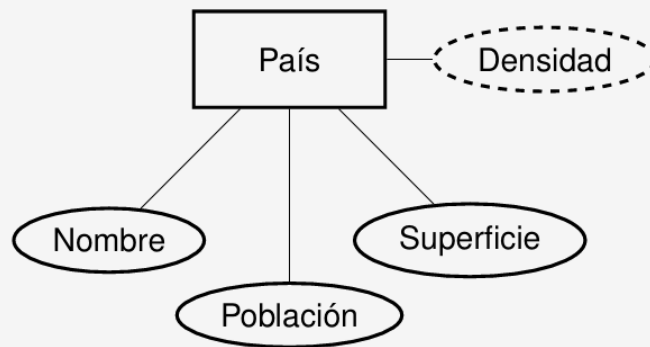
Ejemplo: teléfono y mail de contacto



- *Almacenados vs. Derivados:* Algunos se almacenan directamente y otros se calculan (como la densidad de población a partir de población y superficie).

### Atributos almacenados vs. derivados

Ejemplo: densidad de población

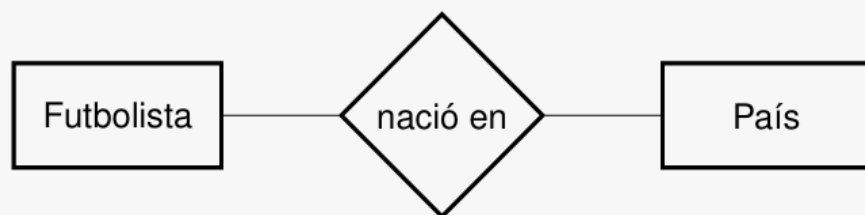


$$Densidad = \frac{Población}{Superficie}$$

#### ◦ **Interrelaciones:**

Definen asociaciones entre dos o más tipos de entidad (por ejemplo, "Futbolista nació en País").

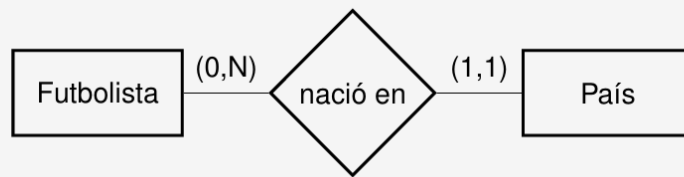
Y los tipos de interrelaciones con:



#### ■ **Aspectos Clave de las Interrelaciones:**

- *Aridad o Grado:* Número de entidades que participan (comúnmente se analizan las binarias, pero existen interrelaciones ternarias y n-arias).
- *Cardinalidad:* Indica el número máximo de instancias que pueden relacionarse (por ejemplo, 1:N, N:M).
- *Participación:* Define la mínima cantidad de instancias que deben participar en la relación (total o parcial).

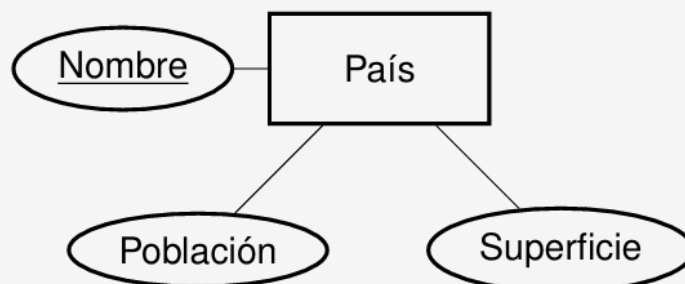
■ Ejemplo:



■ Restricciones de cardinalidad + Restricciones de participación =  
**Restricciones estructurales**

- *Atributos en la Relación:* A veces la interrelación posee atributos propios (por ejemplo, la fecha en que un alumno aprobó una asignatura).
- **Restricción de Unicidad y Atributos Clave:**
  - Cada entidad (y, en ocasiones, cada relación) debe tener un conjunto de atributos que identifiquen unívocamente sus instancias. Si no se identifica un atributo único, se crea uno (como un id).

En el diagrama Entidad-Interrelación los representamos subrayados:



### 3. Extensiones y Modelos Avanzados en ER

- **Entidades Débiles:**

Aquellas que no pueden identificarse de forma independiente y dependen de una entidad fuerte (por ejemplo, "Habitación" de un "Hotel"). La clave de una entidad débil incluye la clave de su entidad identificadora junto con discriminantes propios.

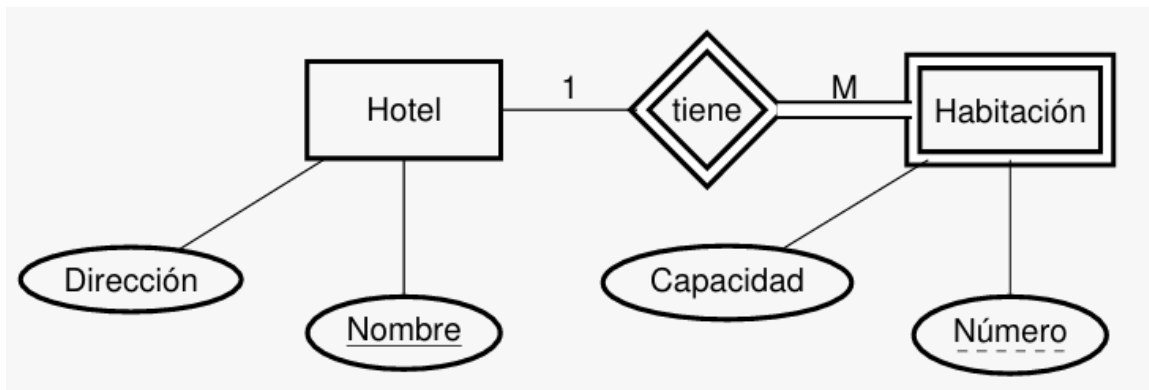
- **Clave compuesta:**

- La clave primaria de una entidad débil se forma combinando la **clave primaria de la entidad fuerte** con uno o más atributos propios de la entidad débil, llamados **discriminantes**.

- Por ejemplo, si "Hotel" tiene una clave primaria **Nombre**, y "Habitación" tiene un atributo **Número**, la clave primaria de "Habitación" sería **(Nombre, Número)**.

#### ○ Notación en el diagrama ER:

- En un diagrama ER, una entidad débil se representa con un **rectángulo de doble línea**.
- La relación que la conecta con la entidad fuerte se representa con un **rombo de doble línea** (si la relación es identificadora).
- La clave primaria de la entidad débil se indica con una **línea punteada** que conecta los atributos discriminantes.



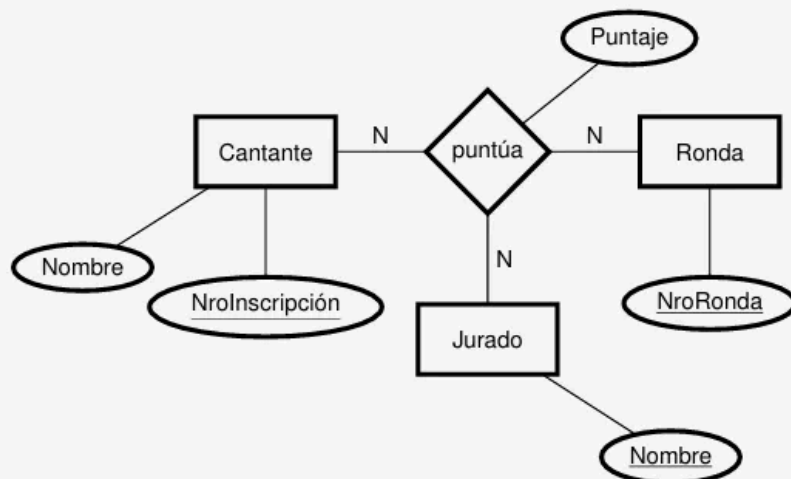
#### • Interrelaciones n-arias:

Relaciones que involucran tres o más entidades. Se debe prestar especial atención a la cardinalidad y a la forma de definir claves.

#### ○ Preguntas clave para determinar la cardinalidad:

- Por ejemplo, fijamos **Cantante** y **Ronda**, y nos preguntamos: ¿Cuántos **Jurados** pueden puntuar a ese cantante en esa ronda?
- O fijamos **Jurado** y **Ronda**, y nos preguntamos: ¿A cuántos **Cantantes** puede puntuar ese jurado en esa ronda?

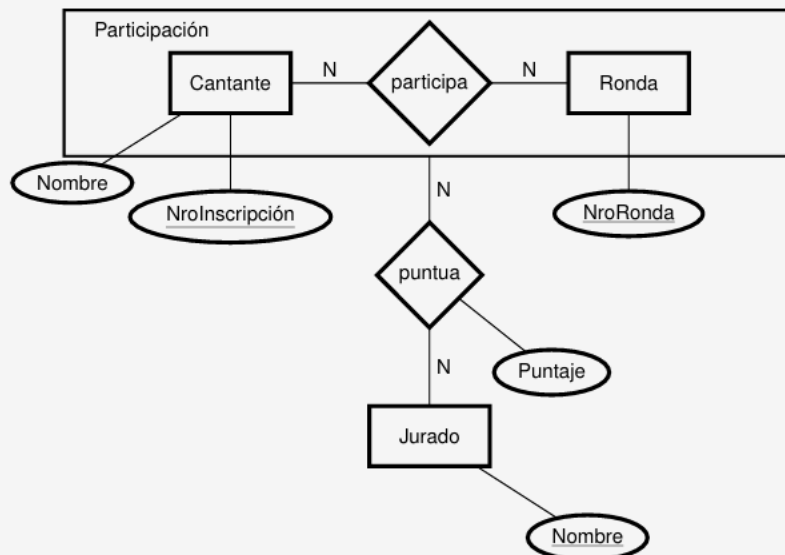
- Las cardinalidades correspondientes son:



- **Agregación:**

Permite tratar una interrelación compleja (por ejemplo, la participación de un "Cantante" en una "Ronda") como una entidad a secas, facilitando el modelado.

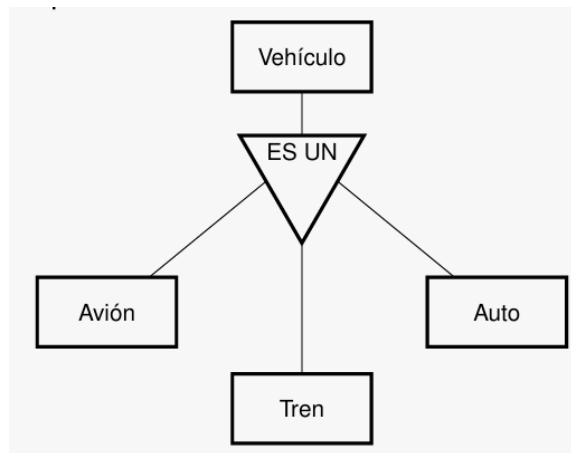
- La desventaja del diseño anterior es que no nos permite registrar cantantes en rondas si no fueron evaluados por ningún jurado. Esto puede resolverse definiendo a la **agregación** de un Cantante y una Ronda como una entidad en sí misma:



- **Generalización/Especialización y Unión:**

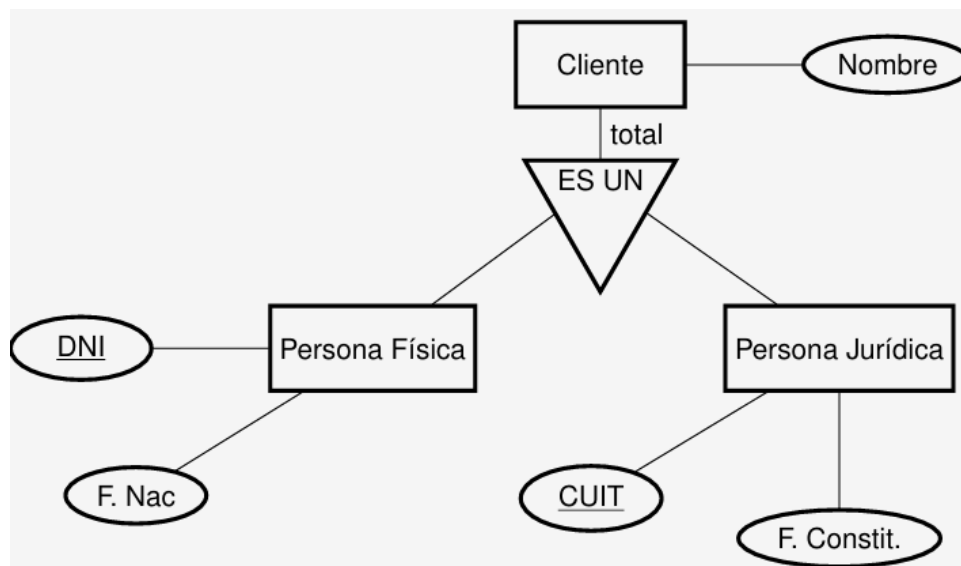
- *Generalización/Especialización:* Permite derivar subtipos de una entidad padre, heredando atributos comunes pero permitiendo que cada subtipo tenga atributos específicos.





◦ *Propiedades:*

- **Superposición:** Los subtipos pueden ser disjuntos o superpuestos (una instancia puede pertenecer a más de un subtipo).
  - **Compleitud:** Determina si todos los casos del tipo padre deben pertenecer a algún subtipo (total) o no (parcial).
- *Unión:* Se utiliza cuando el tipo padre es, en cierto sentido, una unión de los subtipos, implicando que la identificación reside en los subtipos.



## 4. Ejemplos y Aplicaciones Prácticas

- Se presentan casos prácticos (como la librería “Jennifer”, el RENAPER y la biblioteca pública) que muestran la aplicación paso a paso del proceso de diseño conceptual:
  1. Identificación de tipos de entidad.

2. Determinación de atributos.
3. Definición de interrelaciones.
4. Establecimiento de atributos clave.
5. Aplicación de restricciones estructurales (cardinalidad y participación).

## 5. Herramientas de Software para Diseño Conceptual

- **Software CASE y No-CASE:**

Herramientas como ERWin, MS Visio, Visual Paradigm o MySQL Workbench que facilitan el diseño y la representación gráfica de modelos ER, permitiendo crear diagramas de forma organizada y estandarizada.

---

## Conclusión

El material abarca desde la introducción a las bases de datos y su evolución histórica hasta el diseño conceptual utilizando el modelo Entidad-Interrelación. Se destacan:

- La importancia de definir de forma clara y estructurada qué datos se almacenan y cómo se relacionan.
- Las funciones esenciales de un SGBD y la relevancia de la arquitectura en tres niveles para mantener la independencia de los datos.
- El uso del modelo ER para capturar la semántica del dominio, sus elementos (entidades, atributos, interrelaciones) y extensiones para modelar casos complejos (entidades débiles, relaciones n-arias, agregación, generalización/especialización).