



# **INTRODUCCIÓN A LAS BASES DE DATOS RELACIONALES**

**Jesús Reyes Carvajal**

## BREVE HISTORIA

Durante los años ochenta apareció una gran cantidad de SGBD basados en el modelo relacional propuesto en 1969 por E.F. Codd, de IBM, y prácticamente todos utilizaban como lenguaje nativo el SQL.

El modelo relacional se basa en el concepto matemático de *relación*, que aquí podemos considerar de momento equivalente al término *tabla* (formada por filas y columnas).

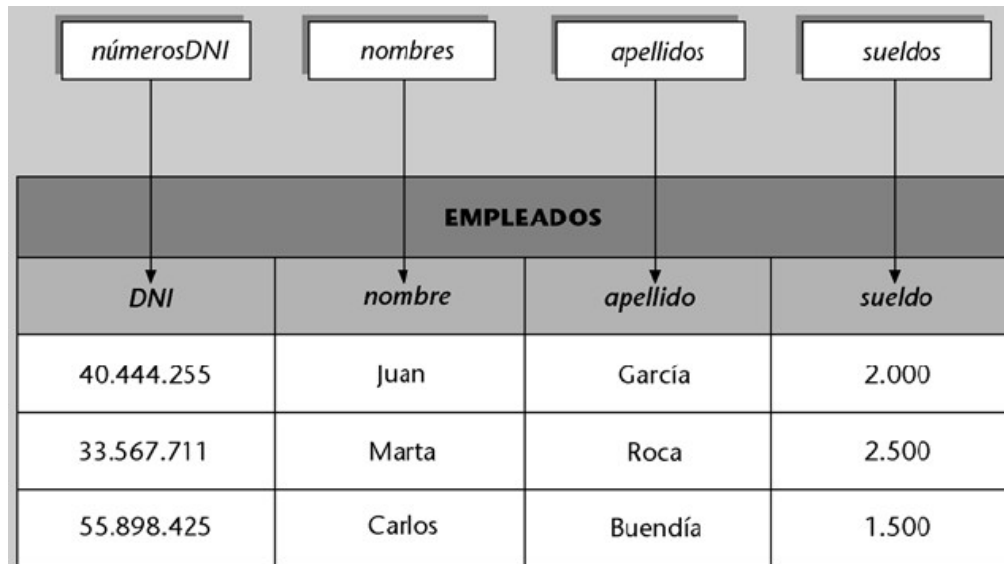
La mayor parte de los Sistemas de información que actualmente están en funcionamiento utilizan SGBD relacionales, pero algunos siguen utilizando los jerárquicos o en red.

## ESTRUCTURA DE LOS DATOS

### Visión informal

El modelo relacional proporciona una estructura de los datos que consiste en un conjunto de relaciones con el objeto de representar la información que nos interesa del mundo real. (El modelo relacional se basa, pues, en el concepto de *relación*)

Se puede obtener una buena idea intuitiva de lo que es una relación si la visualizamos como una tabla o un fichero. (Relación Empleados)



The diagram illustrates the mapping from domain names to table columns for the EMPLEADOS relation. At the top, four boxes represent the domains: *númerosDNI*, *nombres*, *apellidos*, and *suelos*. Arrows point from each of these boxes to a corresponding column in a table below. The table has a header row with the column names *DNI*, *nombre*, *apellido*, and *sueldo*. Below the header are three rows of data representing employees.

EMPLEADOS			
<i>DNI</i>	<i>nombre</i>	<i>apellido</i>	<i>sueldo</i>
40.444.255	Juan	García	2.000
33.567.711	Marta	Roca	2.500
55.898.425	Carlos	Buendía	1.500

Cada columna contiene valores de un cierto dominio; por ejemplo, la columna *DNI* contiene valores del dominio *númerosDNI*.

## Visión conceptual

Conceptos que están vinculados a ellas, como por ejemplo *dominio*, *esquema de relación*, etc.

**Un dominio  $D$**  es un conjunto de valores atómicos. Por lo que respecta al modelo relacional, *atómico* significa *indivisible*; es decir, que por muy complejo o largo que sea un valor atómico, no tiene una estructuración interna para un SGBD relacional.

## Tipos de dominios

1) Dominios predefinidos, que corresponde a los tipos de datos que normalmente proporcionan los lenguajes de bases de datos, como por ejemplo los enteros, las cadenas de caracteres, los reales, etc.

2) Dominios definidos por el usuario, que pueden ser más específicos. Toda definición de un dominio debe constar, como mínimo, del nombre del dominio y de la descripción de los valores que forman parte de éste.

## Un relación se compone del esquema y de la extensión

Empleados				► Esquema
<i>DNI</i>	<i>nombre</i>	<i>apellido</i>	<i>sueldo</i>	
40.444.255	Juan	García	2.000	► Extensión
33.567.711	Marta	Roca	2.500	
55.898.425	Carlos	Buendía	1.500	

El esquema de la relación consiste en un nombre de relación  $R$  y un conjunto de atributos  $\{A1, A2, \dots, An\}$ .

### Nombre y conjunto de atributos de la relación **EMPLEADOS**

El nombre de la relación es *EMPLEADOS* y el conjunto de atributos es  $\{DNI, nombre, apellido, sueldo\}$ .

El esquema de la relación sería:  $R(A1, A2, \dots, An)$ , donde  $R$  es el nombre la relación y  $A1, A2, \dots, An$  es una ordenación cualquiera de los atributos que pertenecen al conjunto  $\{A1, A2, \dots, An\}$ .

## Dominio del atributo DNI

El atributo *DNI* corresponde al papel que ejerce el dominio *númerosDNI* en el esquema de la relación *EMPLEADOS* y, entonces,  $\text{dominio}(\text{DNI}) = \text{númerosDNI}$ .

La extensión de la relación  $R(A_1, A_2, \dots, A_n)$  es un conjunto de tuplas  $t_i$  ( $i = 1, 2, \dots, m$ ), donde cada tupla  $t_i$  es, a su vez un valor.

Así que para cada par  $\langle A_j: v_{ij} \rangle$ , se cumple que  $v_{ij}$  es un valor de dominio( $A_j$ ), o bien un valor especial que denominaremos nulo.

A1	A2	A3	A4
	A2:vi2	A3:vi3	
			A4:vi4
A1:vi1		A3:vi3	

El esquema de la relación EMPLEADOS(*DNI, nombre, apellido, sueldo*), el conjunto de tuplas de su extensión sería la siguiente:



Si en una tupla  $ti = \langle vi1, vi2, \dots, vin \rangle$ , el valor  $vij$  es un valor nulo, entonces el valor del atributo  $Aj$  es desconocido para la tupla  $ti$  de la relación, o bien no es aplicable a esta tupla.

El grado de una relación es el número de atributos que pertenecen a su esquema, el grado de la relación de esquema EMPLEADOS(*DNI, nombre, apellido, sueldo*), es 4.

La cardinalidad de una relación es el número de tuplas que pertenecen a su extensión, La cardinalidad de la relación EMPLEADOS es 3.

## **Propiedades de una relación(tabla)**

Cada tabla debe tener un nombre distinto.

Cada atributo de la tabla solo puede tener un valor en cada tupla.

Cada atributo tiene un nombre distinto en cada tabla.

Cada tupla es única (no hay tuplas duplicadas)

El orden de los atributos no importa.

El orden de las tuplas no importa.



## Clave candidata, clave primaria y clave alternativa de las relaciones

Toda la información que contiene una base de datos debe poderse identificar de alguna forma. En el caso particular de las bases de datos que siguen el modelo relacional, para identificar los datos que la base de datos contiene, se pueden utilizar las claves candidatas de las relaciones.

Una superclave de una relación de esquema  $R(A_1, A_2, \dots, A_n)$  es *un* subconjunto de los atributos del esquema tal que no puede haber dos tuplas en la extensión de la relación que tengan la misma combinación de valores para los atributos del subconjunto.

Una superclave, por lo tanto, nos permite identificar todas las tuplas que contiene la relación.

### Claves candidatas de *EMPLEADOS*

En la relación de esquema *EMPLEADOS*(*DNI*, *NSS*, *nombre*, *apellido*, *teléfono*), sólo hay dos claves candidatas:  $\{DNI\}$  y  $\{NSS\}$ .

Habitualmente, una de las claves candidatas de una relación se designa clave primaria de la relación. La clave primaria es la clave candidata cuyos valores se utilizarán para identificar las tuplas de la relación.

Las claves candidatas no elegidas como primaria se denominan claves alternativas.

En la relación de esquema EMPLEADOS(*DNI, NSS, nombre, apellido, teléfono*), hay dos claves candidatas, {*DNI*} y {*NSS*}, se puede elegir como clave primaria {*DNI*}.

Lo indicaremos subrayando el atributo *DNI* en el esquema de la relación EMPLEADOS(*DNI*, *NSS, nombre, apellido, teléfono*).

En este caso, la clave {*NSS*} será una clave alternativa de EMPLEADOS.

Es posible que una clave candidata o una clave primaria conste de más de un atributo.

## Claves foráneas de las relaciones

El mecanismo que proporcionan las bases de datos relacionales para conectar tuplas de una relación con otra relación, son las claves foráneas de las relaciones.

### Equipos

Equipo	Nº Equipo
Real Madrid	1
F.C. Barcelona	2
Athletic Bilbao	3

### Jugadores

Nº Jugador	Jugador	Nº Equipo
1	Muniain	3
2	Messi	2
3	Cristiano Ronaldo	1
4	Bale	1

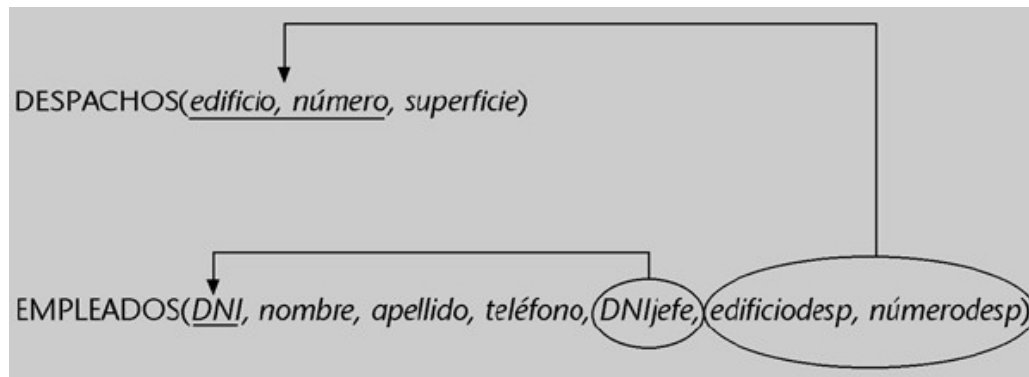
El atributo **Nº Equipo** sirve para relacionar el Jugador con el equipo al que pertenece.

El campo **Nº Equipo** en la tabla de jugadores es una clave foránea

El campo **Nº Equipo** en la tabla de equipos es una clave primaria

## Veamos otro ejemplo con una llave foránea compuesta por dos campos

En la relación EMPLEADOS(*DNI*, *nombre*, *apellido*, *teléfono*, *DNIjefe*, *edificiodesp*, *númerodesp*), tiene una clave foránea formada por los atributos *edificiodesp* y *númerodesp* que se refiere a la clave primaria de la relación DESPACHOS(*edificio*, *número*, *superficie*).



Las claves foráneas tienen por objetivo establecer una conexión con la clave primaria que referencian. Por lo tanto, los valores de una clave foránea deben estar presentes en la clave primaria correspondiente, o bien deben ser valores nulos. En caso contrario, la clave foránea representaría una referencia o conexión incorrecta.

En la relación de esquema EMPLEADOS(DNI, nombre, apellido, DNIjefe, edificiodesp, númerodesp), la clave foránea {edificiodesp, númerodesp} referencia la relación DESPACHOS(edificio, número, superficie).

Se cumple que todos los valores que no son nulos de los atributos edificiodesp y númerodesp son valores que existen para los atributos edificio y número de DESPACHOS

- Relación DESPACHOS:

DESPACHOS		
<u>edificio</u>	<u>número</u>	superficie
Marina	120	10
Marina	122	15
Marina	230	20
Diagonal	120	10

- Relación EMPLEADOS

EMPLEADOS					
<u>DNI</u>	nombre	apellido	DNIjefe	edificiodesp	númerodesp
40.444.255	Juan	García	NULO	Marina	120
33.567.711	Marta	Roca	40.444.255	Marina	120
55.898.425	Carlos	Buendía	40.444.255	Diagonal	120
77.232.144	Elena	Pla	40.444.255	NULO	NULO

## OPERACIONES DEL MODELO RELACIONAL

Las operaciones del modelo relacional deben permitir manipular datos almacenados en una base de datos relacional y por lo tanto están estructurados en forma de relaciones. La manipulación de datos incluye básicamente dos aspectos: la actualización y la consulta.

La actualización de los datos consiste en hacer que los cambios que se producen en la realidad queden reflejados en las relaciones de la base de datos.

Existen tres operaciones básicas de actualización:

- a) Inserción, que sirve para añadir una o más tuplas a una relación.
- b) Borrado, que sirve para eliminar una o más tuplas de una relación.
- c) Modificación, que sirve para alterar los valores que tienen una o más tuplas de una relación para uno o más de sus atributos.

La consulta de los datos consiste en la obtención de datos deducibles a partir de las relaciones que contiene la base de datos.

La obtención de los datos que responden a una consulta puede requerir el análisis y la extracción de datos de una o más de las relaciones que mantiene la base de datos.

**Según la forma como se especifican las consultas, podemos clasificar los lenguajes relacionales en dos tipos:**

**Lenguajes basados en el álgebra relacional.**

El álgebra relacional se fundamenta en la teoría de conjuntos, consiste de un conjunto de operadores de alto nivel que operan sobre las relaciones.

**Codd** definió un conjunto específico de 8 operadores el cual dividió en dos grupos :

Operaciones tradicionales (unión, intersección, diferencia y producto cartesiano).

Operaciones especiales (Selección, proyección, reunión y división)

$$A \cap B$$

$$\Pi \text{ nomb,apel (A)}$$

$$\sigma_{\text{edad} > 18} (A)$$

$$A \bowtie_{x3=x3} B$$

Los lenguajes basados en el álgebra relacional son lenguajes procedimentales, ya que los pasos que forman la consulta describen un procedimiento.

**Lenguajes basados en el cálculo relacional.** El cálculo relacional tiene su fundamento teórico en el cálculo de predicados de la lógica matemática.

#### **Cálculo relacional de tuplas (CRT)**

El cálculo relacional de tuplas se utiliza para seleccionar tuplas de una relación.

El resultado de la relación puede tener una o más tuplas.

#### **Ejemplo:**

**{ T.name | Author(T) AND T.article = 'biblioteca' }**

La consulta selecciona tuplas de la relación Author y devuelve el nombre de los autores que ha escrito un artículo de la base de datos 'biblioteca'.

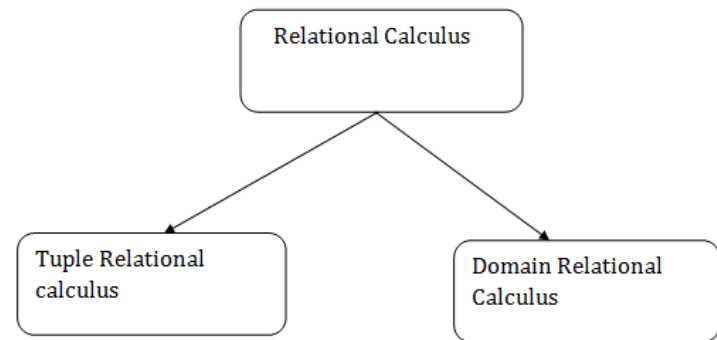
#### **Calculo relacional de dominio (CRD)**

la variable de filtrado utiliza el dominio de atributos.

#### **Ejemplo:**

**{< article, page, topic > | ∈ document ∧ subject = 'biblioteca'}**

Esta consulta obtendrá el artículo, página y tema de la relación document, de la base de datos 'biblioteca'.



El álgebra relacional y el cálculo relacional son dos notaciones alternativas que Codd definió como bases formales en la parte de manipulación de datos.

La diferencia entre ambas notaciones radica en que el álgebra relacional ofrece una serie de operadores para obtener nuevas relaciones a partir de relaciones dadas y el cálculo relacional brinda los medios para definir mediante fórmulas la relación deseada, también a partir de relaciones dadas.



Una sentencia de consulta en **SQL(lenguaje de consulta estructurado)**, combina construcciones del álgebra relacional y del cálculo relacional con un predominio de las construcciones del cálculo. Este predominio determina que SQL sea un lenguaje declarativo.

