

### **FASES DEL DISEÑO DE BD**

El diseño de una base de datos consiste en extraer todos los datos relevantes de un problema.

Para extraer estos datos, se debe realizar un análisis en profundidad del problema, para averiguar qué datos son esenciales para la base de datos y descartar los que no sean necesarios.

Una vez extraídos los datos esenciales comenzamos a construir los modelos adecuados. Un esquema que exprese con total exactitud todos los datos que el problema requiere almacenar.

### **FASE DE ANÁLISIS: ESPECIFICACIÓN DE REQUISITOS SOFTWARE (E.R.S.)**

Antes de pasar a diseñar una BD hay que tener claro que es lo que queremos hacer. Para ello, típicamente los informáticos se reúnen con los futuros usuarios del sistema para recopilar la información que necesitan para saber que desean dichos usuarios.

Normalmente se hace una reunión inicial a y partir de ella se elabora una batería de preguntas para entrevistar a los usuarios finales en una segunda reunión y obtener de ella una información detallada de lo que se espera de nuestra BD.

De estas entrevistas, se extrae el documento más importante del análisis, el documento de Especificación de Requisitos Software o E.R.S. A partir de dicha E.R.S. Se extrae toda la información necesaria para la modelización de datos.

### **FASE DE DISEÑO CONCEPTUAL: MODELO ENTIDAD/RELACIÓN (E/R)**

Habitualmente quien realiza la modelización es un analista informático que no tiene porqué ser un experto en el problema que pretende resolver (Contabilidad, Gestión de Reservas hoteleras, medicina, economía, etc.).

Es por esto que es imprescindible contar con la experiencia de un futuro usuario de la BD que conozca a fondo todos los entresijos del negocio, y que, a su vez, no tienen porqué tener ningún conocimiento de informática.

El objetivo de esta fase del diseño consiste en representar la información obtenida del usuario final y concretada en el E.R.S. mediante estándares para que el resto de la comunidad informática pueda entender y comprender el modelo realizado.

El modelo que se utiliza en esta primera fase del diseño tiene un gran poder expresivo para poder comunicarse con el usuario que no es experto en informática y se denomina Modelo Conceptual. El modelo conceptual que utilizaremos es el Modelo Entidad/Relación.

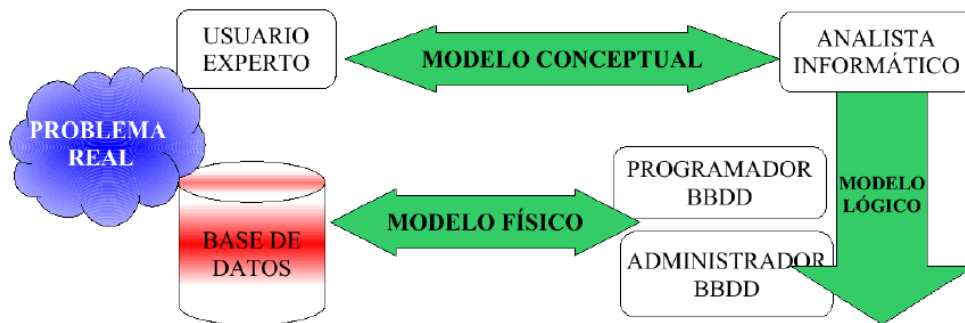
### **FASE 2 DEL DISEÑO. DISEÑO LÓGICO: MODELO RELACIONAL**

Este modelo es más técnico que el anterior porque está orientado al personal informático y generalmente tiene traducción directa al modelo físico que entiende el SGBD. Se obtienen a partir del modelo conceptual y dependerá de la implementación de la BD.

Así, no es lo mismo implementar una base de datos jerárquica u orientada a objetos que una BD relacional. El modelo que se usará en este módulo es el Modelo Relacional.

### FASE 3 DEL DISEÑO. DISEÑO FÍSICO: MODELO FÍSICO

Es el resultado de aplicar el modelo lógico a un SGBD concreto. Generalmente está expresado en un lenguaje de programación de BBDD tipo SQL.



### El proceso de diseño de bases de datos

#### ■ Fase 1:

##### **Análisis de requisitos**

Recabar información sobre el uso que se piensa dar a la base de datos (elicitación de requisitos del sistema).

#### ■ Fase 2:

##### **Diseño conceptual (modelo E/R)**

Creación de un esquema conceptual de la base de datos independiente del DBMS que se vaya a utilizar. 🌊

## El proceso de diseño de bases de datos

### ■ Fase 3:

#### **Elección del sistema gestor de bases de datos**

Elección del modelo de datos (tipo de DBMS) y del DBMS concreto (p.ej. relacional, multidimensional...).

### ■ Fase 4:

#### **Diseño lógico**

Creación del esquema conceptual para el modelo de datos del DBMS elegido (p.ej. paso del modelo E/R a un conjunto de tablas).

### ■ Fase 5:

#### **Diseño físico**

Creación de la base de datos utilizando el DDL (lenguaje de definición de datos del DBMS).

### ■ Fase 6:

#### **Uso y mantenimiento**

Gestión de los datos utilizando el DML (lenguaje de manipulación de datos del DBMS).

## **Del modelo E/R al modelo relacional: Entidades**

Cada tipo de entidad da lugar a una tabla en la base de datos.



- **Atributos:**  
Los atributos del tipo de entidad.
- **Clave primaria:**  
Una de las claves candidatas del conjunto de entidades.

### **1. El modelo relacional**

El modelo de datos relacional fue desarrollado por E.F. Codd para IBM, a finales de los años sesenta. Propone un modelo basado en la teoría matemática de las relaciones, con el objetivo de mantener la independencia de la estructura lógica respecto al modo de almacenamiento y otras características de tipo físico.

El modelo de Codd persigue, al igual que la mayoría de los modelos de datos, los siguientes objetivos:


- **Independencia física de los datos.** El modo de almacenamiento de los datos no debe influir en su manipulación lógica.
- **Independencia lógica de los datos.** Los cambios que se realicen en los objetos de la base de datos no deben repercutir en los programas y usuarios que acceden a la misma.
- **Flexibilidad.** Para presentar a los usuarios los datos de la forma mas adecuada a la aplicación que utilicen.

- **Uniformidad** en la presentación de las estructuras lógicas de los datos, que son tablas, lo que facilita la concepción y manipulación de la base de datos por parte de los usuarios.
- **Sencillez.** Pues las características anteriores, así como unos lenguajes de usuario sencillos, hacen que este modelo sea fácil de comprender y utilizar por el usuario.

Para conseguir estos objetivos, Codd introduce el concepto de *relación* (tabla) como estructura básica del modelo. Todos los datos de una base de datos se representan en forma de relaciones cuyo contenido varía en el tiempo. El modelo relacional se basa en dos ramas de las matemáticas: la *teoría de conjuntos* y la *lógica de predicados*. Esto hace que sea un modelo seguro y robusto.

El modelo de datos relacional organiza y representa los datos en forma de tablas o relaciones:


**Una base de datos relacional es una colección de relaciones [tablas].**

<b>Representación lógica</b>	<b>Representación física</b>	<b>Modelo relacional</b>
 Tabla	Archivo secuencial	Relación
 Fila	Registro	Tupla
 Columna	Campo	Atributo

**El concepto de relación:**  
**Tuplas, atributos y dominios**

id_trabajador	nombre	tarifa_hr	tipo_de_oficio	id_supv
1235	F. Aguilera	12,50	Electricista	1311
1412	A. Calvo	13,75	Fontanero	1540
2920	N. Marín	10,00	Carpintero	null
3231	O. Pons	17,40	Albañil	null
1540	J.M. Medina	11,75	Fontanero	null
1311	J.C. Cubero	15,50	Electricista	null
3001	D. Sánchez	8,20	Albañil	3231

**El concepto de relación:**  
**Tuplas, atributos y dominios**

- **Atributo ( $A_i$ ):** Elemento susceptible de tomar valores (cada una de las columnas de la tabla).
- **Dominio ( $D_i$ ):** Conjunto de valores que puede tomar un atributo (se considera finito).
- **Tupla:** Cada uno de los elementos que contiene una instancia de la relación (filas). 

## El concepto de relación

### **Relación $R(A_1..A_n)$**

Subconjunto del producto cartesiano  $D_1 \times \dots \times D_n$   
(esto es, una tabla).

En una relación hay que distinguir dos aspectos:

- **Esquema de la relación:** Los atributos  $A_1..A_n$   
p.ej. Trabajadores (id\_trabajador, nombre, tarifa\_hr, tipo\_de\_oficio, id\_supv)
- **Instancia de la relación:** El conjunto de tuplas  $\{(x_1, x_2, \dots, x_n)\} \subseteq D_1 \times D_2 \times \dots \times D_n$  que la componen en cada momento.

## El concepto de relación

### **Relación $R(A_1..A_n)$**

Subconjunto del producto cartesiano  $D_1 \times \dots \times D_n$   
(esto es, una tabla).

Consecuencias de la definición de relación  
como conjunto de tuplas:

- No existen tuplas duplicadas  
(concepto de clave primaria).
- No existe orden en las tuplas  
(ni en los atributos).



## Instancia de la base de datos

- **Instancia (o estado) de la base de datos:**

Colección de instancias de relaciones que verifican las restricciones de integridad.

- **Base de datos relacional:**

Instancia de la base de datos junto con su esquema.



## Restricciones de integridad:

### Asociadas a las tuplas de una relación

p.ej.  $0 \leq \text{edad} \leq 120$   
 $\text{impuestos} \leq \text{suelo}$

- En ocasiones, no se conoce el valor de un atributo para una determinada tupla. En esos casos, a ese atributo de esa tupla se le asigna un **valor nulo (null)**, que indica que el valor de ese atributo es desconocido o, simplemente, que ese atributo no es aplicable a esa tupla.



## **Restricciones de integridad:**

### **Asociadas a las relaciones de la base de datos**

#### ■ **Clave primaria:**

Conjunto de atributos seleccionados para identificar unívocamente a las tuplas de una relación.

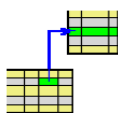
#### ■ **Integridad de entidad:**

Los atributos de la clave primaria no pueden tomar valores nulos, ya que la clave primaria debe permitirnos identificar unívocamente cada tupla de la relación.

## **Restricciones de integridad:**

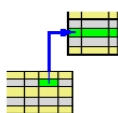
### **Asociadas a las relaciones de la base de datos**

La integridad referencial mantiene las conexiones en las bases de datos relacionales:



$\text{imparte.NRP} \in \text{profesor.NRP}$

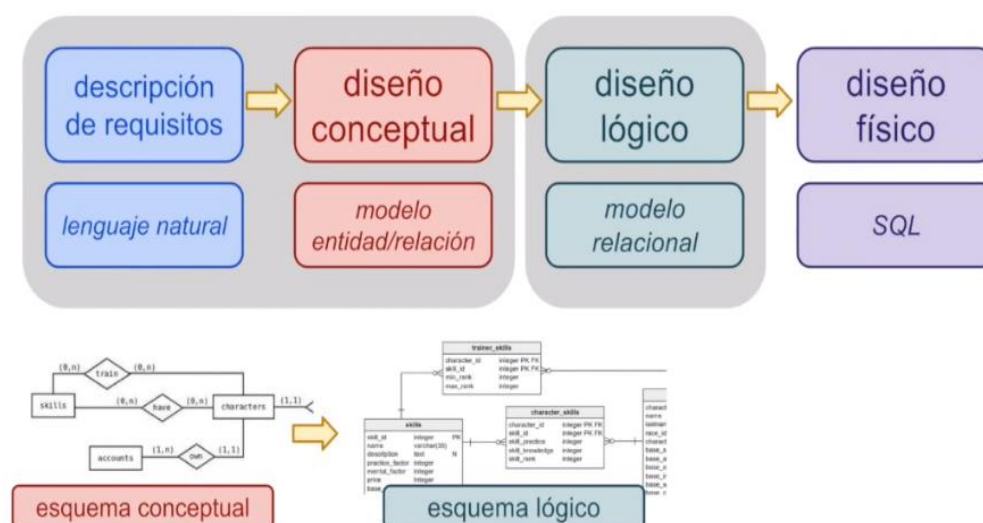
El profesor que imparte una asignatura debe existir en la tabla de profesores.



$\text{cuenta.sucursal} \in \text{sucursal.número}$

Una cuenta tiene que pertenecer a una sucursal existente.

## 2. Estructura del modelo relacional



La estructura fundamental del modelo relacional es precisamente esa, la «relación», es decir una tabla bidimensional constituida por filas (registros o tuplas) y columnas (atributos o campos).

Tabla 1: PERSONAS

D.N.I.	Nombre	Apellido	Nacimiento	Sexo	Estado civil
52.768.987	Juan	Loza	15/06/1976	H	Soltero
06.876.983	Isabel	Gálvez	23/12/1969	M	Casada
34.678.987	Micaela	Ruiz	02/10/1985	M	Soltera

PERSONAS	
♦ <b>DNI</b>	<b>PK</b>
◊ Nombre	
◊ Apellido	
◊ Nacimiento	
◊ Sexo	
◊ Estado civil	

Las relaciones o tablas representan las entidades del modelo E/R, mientras que los atributos de la relación representarán las propiedades o atributos de dichas entidades. Por ejemplo, si en la base de datos se tienen que representar la entidad PERSONA, está pasará a ser una relación o tabla llamada

«PERSONAS», cuyos atributos describen las características de las personas (tabla siguiente).

Cada tupla o registro de la relación «PERSONAS» representará una persona concreta.

La **relación** es el elemento básico del modelo relacional y se representa como una tabla, en la que se puede distinguir el nombre de la Tabla, el conjunto de columnas que representan las propiedades de la tabla y que se les llama *atributos*, y el conjunto de filas llamadas *tuplas*, que contienen los valores que toma cada uno de los atributos para cada elemento de la relación.

Una relación tiene una serie de elementos característicos que la distinguen de una tabla:

- No admiten filas duplicadas.
- Las filas y columnas no están ordenadas.
- La tabla es plana. En el cruce de una fila y una columna solo puede haber un valor, no se admiten atributos multivaluados.

En la figura que se muestra a continuación se representa una relación llamada ALUMNO en forma de tabla.

<b>Término relacional formal</b>	<b>Equivalente informal</b>
Relación	Tabla
Tupla	Fila o registro
Cardinalidad	Número de filas o registros
Atributo	Columna o campo
Grado	Número de columnas o campos
Clave primaria	Identificador único
Dominio	Fondos de valores legales

Para distinguir un registro de otro, se usa la «clave primaria o clave principal».

En una relación puede haber más combinaciones de atributos que permitan identificar unívocamente una fila (estos se llamarán «llaves o claves candidatas»), pero entre éstas se elegirá una sola para utilizar como llave primaria. Los atributos de la llave primaria no pueden asumir el valor nulo.

### **3. Elementos y propiedades del modelo relacional**

Relación (tabla): Representan las entidades de las que se quiere almacenar información en la BD. Esta formada por:

- Filas (Registros o Tuplas): Corresponden a cada ocurrencia de la entidad.
- Columnas (Atributos o campos): Corresponden a las propiedades de la entidad. Siendo rigurosos una relación está constituida sólo por los atributos, sin las tuplas.

Las relaciones tienen las siguientes propiedades:

- Cada relación tiene un nombre y éste es distinto del nombre de todas las demás relaciones de la misma BD.
- No hay dos atributos que se llamen igual en la misma relación.
- El orden de los atributos no importa: los atributos no están ordenados.
- Cada tupla es distinta de las demás: no hay tuplas duplicadas. (Como mínimo se diferenciarán en la clave principal)
- El orden de las tuplas no importa: las tuplas no están ordenadas.

Clave candidata: atributo que identifica unívocamente una tupla. Cualquiera de las claves candidatas se podría elegir como clave principal.

Clave Principal: Clave candidata que elegimos como identificador de la tuplas.

Clave Alternativa: Toda clave candidata que no es clave primaria (las que no hayamos elegido como clave principal)

Una clave principal no puede asumir el valor nulo (Integridad de la entidad).

Dominio de un atributo: Conjunto de valores que pueden ser asumidos por dicho atributo.

Clave Externa o foránea o ajena: el atributo o conjunto de atributos que forman la clave principal de otra relación. Que un atributo sea clave ajena en una tabla significa que para introducir datos en ese atributo, previamente han debido introducirse en la tabla de origen. Es decir, los valores presentes en la clave externa tienen que corresponder a valores presentes en la clave principal correspondiente (Integridad Referencial).

RELACIÓN ALUMNOS				
NUM_MAT	NOMBRE	APELLIDOS	CURSO	← ATRIBUTOS
5467	JUAN	CABELLO	1BACH-A	← TUPLAS
3421	DOLORES	GARCÍA	1BACH-C	
7622	JESÚS	SÁNCHEZ	2BACH-C	

### CASO PRÁCTICO

**Disponemos de las tablas TDEPART y TEMPLE. Las columnas de la tabla TDEPART son:**

No. de departamento (NUMDEPT), Nombre de departamento (NOMDEPT), Presupuesto (PRESUPUESTO).

Las claves candidatas son:

No. de departamento y Nombre de departamento, pues son únicos y no se van a repetir. Podemos escoger como primaria el No. de departamento. Normalmente se elegirán como claves primarias los campos códigos o Número de empleado, departamento o artículo, que suelen ser numéricos.

Las columnas de la tabla TEMPLE son:

Nº. de empleado (NUMEMP), Apellido (APELLIDO), Nº. de departamento (NUMDEP), Salario (SALARIO).

La clave candidata es el Nº. de empleado. El apellido se puede repetir, así que no se la considera candidata. Como sólo hay una se escoge primaria el Nº. de empleado.

El atributo Nº. de departamento es clave ajena; relaciona las tablas TEMPLE y TDEPART.

La Figura 2.3 muestra el contenido de las tablas TEMPLE y TDEPART con sus claves primarias y ajenas.

TDEPART

CLAVE PRIMARIA			
NUMDEPT	NOMDEPT	PRESUPUESTO	
D1	Marketing	1.000	
D2	Desarrollo	1.200	
D3	Investigación	5.000	

TEMPLE

CLAVE PRIMARIA		CLAVE AJENA		
NUMEMP	APELLIDO	NUMDEP	SALARIO	
E1	López	D1	1.500	
E2	Fernández	D2	1.600	
E3	Martínez	D3	1.800	
E4	Sánchez	D2	2.000	

Figura 2.3. Tablas TEMPLE y TDEPART con las claves primarias y ajenas.

### Esquema de la base de datos

Una **base de datos relacional** es un conjunto de relaciones normalizadas. Para representar un esquema de una base de datos se debe dar el nombre de sus relaciones, los atributos de estas, los dominios sobre los que se definen estos atributos, las claves primarias y las claves ajenas.

En el esquema los nombres de las relaciones aparecen seguidos de los nombres de los atributos encerrados entre paréntesis.

Las claves primarias son los atributos subrayados, y las claves ajenas se representan mediante diagramas referenciales.

El esquema de la base de datos de empleados y departamentos es el siguiente:

TDEPART (NUMDEPT, NOMDEPT, PRESUPUESTO)  
TEMPLE (NUMEMP, APELLIDO, NUMDEP, SALARIO)

TEMPLE → NUMDEP TEDEPART: Departamento al que pertenece el empleado

#### **4. Restricciones del modelo relacional**

En todos los modelos de datos existen restricciones que a la hora de diseñar una base de datos se tienen que tener en cuenta.

Los datos almacenados en la base de datos han de adaptarse a las estructuras impuestas por el modelo y deben cumplir una serie de reglas para garantizar que son correctos.

El modelo relacional impone dos tipos de restricciones. Algunas de ellas ya las hemos citado en las propiedades de las relaciones y las claves.

Los tipos de restricciones son:

– **Restricciones inherentes al modelo**, indican las características propias de una relación que han de cumplirse obligatoriamente y que diferencian una relación de una tabla. No hay dos tuplas iguales, el orden de las tuplas y los atributos no es relevante.

Cada atributo solo puede tomar un único valor del dominio al que pertenece. Ningún atributo que forme parte de la clave primaria de una relación puede tomar un valor nulo.

– **Restricciones semánticas o de usuario**, que representan la semántica del mundo real.

Estas hacen que las ocurrencias de los esquemas de la base de datos sean validos. Los mecanismos que proporciona el modelo para este tipo de restricciones son los siguientes:

b) La restricción de unicidad (UNIQUE) permite definir claves alternativas. Los valores de los atributos no pueden repetirse.



c) La restricción de obligatoriedad (NOT NULL) permite declarar si uno o varios atributos no pueden tomar valores nulos.

d) Integridad referencial o restricción de clave ajena (FOREIGN KEY). Se utiliza para enlazar relaciones, mediante claves ajenas, de una base de datos.

La integridad referencial indica que los valores de la clave ajena en la relación hijo se corresponden con los de la clave primaria en la relación padre.

Además de definir las claves ajenas hay que tener en cuenta las operaciones de borrado y actualización que se realizan sobre las tuplas de la relación referenciada.

Las posibilidades son las siguientes:

- **Borrado y/o modificación en cascada (CASCADE).** El borrado o modificación de una tulpas en la relación padre (relación con la clave primaria) ocasiona un borrado o modificación de las tuplas relacionadas en la relación hija (relación que contiene la clave ajena).

En el caso de empleados y departamentos, si se borra un departamento de la tabla TDEPART se borrarán los empleados que pertenecen a ese departamento. Igualmente ocurrirá si se modifica el NUMDEPT de la tabla TDEPART esa modificación se arrastra a los empleados que pertenezcan a ese departamento.

- **Borrado y/o modificación restringido (RESTRICT).** En este caso no es posible realizar el borrado o la modificación de las tuplas de la relación padre si existen tuplas relacionadas en la relación hija. Es decir, no podría borrar un departamento que tiene empleados.