

TEMA 2. EL MODELO RELACIONAL

1 Características generales del modelo

El modelo relacional fue propuesto por Codd en 1970 a través de un artículo publicado en un boletín informático. Codd propone un modelo de datos basado en la teoría matemática de las relaciones en el que se concede gran importancia al hecho de que la estructura lógica de los datos sea independiente de su almacenamiento interno, cosa que los modelos de la época no recogían.

Como todos los modelos de datos, el relacional tiene una parte estática constituida por los objetos propios del modelo y las restricciones, y una parte dinámica o conjunto de operaciones definidas sobre la estructura, que permite la modificación de los datos. La parte dinámica del modelo está constituida por el **álgebra relacional** y el **cálculo relacional**.

2 Estructura del modelo

El elemento fundamental del modelo relacional es la **tabla**, que consta de **filas** o tuplas y **columnas**, atributos o campos.

El número de filas de una tabla constituye su **cardinalidad** y el número de columnas es el **grado**.

TABLA CLIENTES

DNI	Nombre	Apellido1	Apellido2	Localidad	
1234567	Luis	Pérez	Rodríguez	Valencia	-> Fila 1
8901234	Pedro	González	Gutiérrez	Alicante	-> Fila 2
5678901	Jorge	Rodríguez	García	Castellón	-> Fila 3
2345678	Antonio	López	Abad	Murcia	-> Fila 4

Columna1 Columna2 Columna3 Columna4 Columna5

Cardinalidad de la tabla “Clientes”: 4

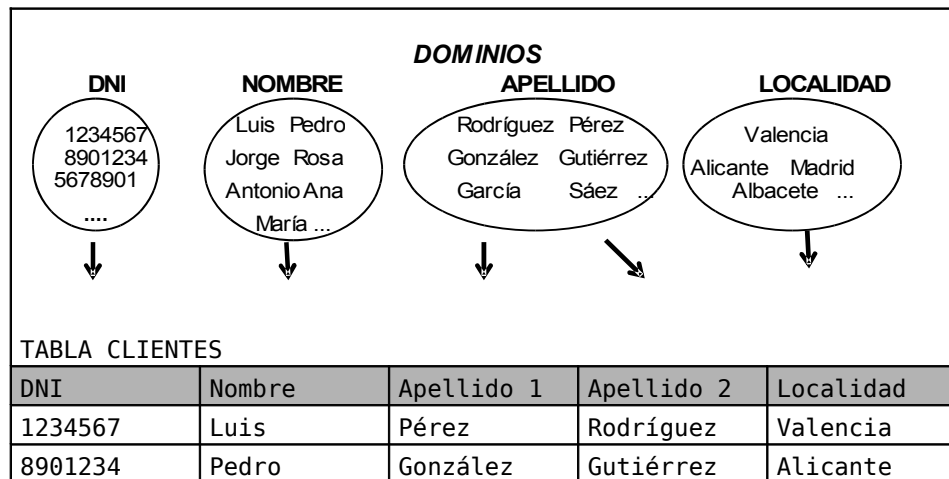
Grado de la tabla “Clientes”: 5

El valor que adopta en cada fila, un atributo o columna determinada es tomado de un conjunto de valores conocido como “dominio” asociado a la columna. Varias columnas pueden tomar sus valores de un mismo dominio, pero cada columna toma sus valores de un dominio único.

2.1. Dominios

Un **dominio** es un conjunto de elementos o valores, (V1, V2,..., Vn) que se asocian a cada atributo. Por ejemplo, el dominio de sexo tiene dos valores: *Varón*, *Mujer*.

Todo dominio ha de tener un *nombre* por el cual nos podamos referir a él y un *tipo de datos*; así el tipo de datos del dominio sexo es una tira de caracteres de longitud 5.



Los dominios participan en las tablas a través de las columnas o atributos: Toda columna o atributo de una tabla debe tomar sus valores de un único dominio que se denomina **dominio subyacente** a la columna.

Pese a que el concepto de dominio es importante en el modelo relacional, la mayoría de los sistemas comerciales **no** lo incorporan.

- Otra forma de ver un dominio es considerándolo como un **tipo de datos** de un lenguaje de programación. De hecho, este tipo de dominios es el que usan los SGBD relacionales actuales, ya que las columnas de las tablas se definen como INTEGER, CHAR, etc.

2.2. Tablas

Se puede describir una tabla del siguiente modo: Una tabla T se compone de dos partes, una cabecera y un cuerpo.

- La **cabecera** (o esquema de la tabla) es un conjunto de n pares "atributo-dominio subyacente", siendo n el grado de la tabla.
- El **cuerpo** (u ocurrencias de la tabla) es un conjunto de m filas donde cada fila es un conjunto de n pares "atributo-valor". El valor asociado a cada atributo es tomado de su dominio subyacente. El número de tuplas, m, es la cardinalidad de la tabla.

DNI	Nombre	Apellido1	Apellido2	Localidad	-> Cabecera
1234567	Luis	Pérez	Rodríguez	Valencia	} ->Cuerpo
8901234	Pedro	González	Gutiérrez	Alicante	

Las tablas del modelo relacional tienen las siguientes limitaciones:

- Una tabla nunca puede tener filas duplicadas (filas con el mismo valor en todas las columnas).
- Las filas de una tabla no tienen orden, es decir, el orden de las filas de una tabla no es significativo.
- Lo mismo ocurre con los atributos o columnas: su orden no es significativo.
- Cada intersección entre una fila y una columna (es decir, cada celda) sólo puede tener un valor.

3. Restricciones al modelo

El modelo relacional, como todo modelo de datos, tiene unas restricciones, es decir, una serie de estructuras u ocurrencias no permitidas. Estas restricciones pueden ser inherentes al modelo o restricciones de usuario.

- Las **restricciones inherentes** al modelo son las cuatro limitaciones que acabamos de ver en el punto anterior, junto con la regla de integridad de entidad, que veremos más adelante.
- Una **restricción de usuario** es una limitación definida sobre un conjunto de dominios, atributos o filas, que debe ser cumplido por todas las ocurrencias del esquema. Entre las restricciones de usuario se incluye la regla de integridad referencial.

Los datos almacenados en la base de datos han de adaptarse tanto a las restricciones inherentes al modelo como a las de usuario. Antes de ver las restricciones más importantes, vamos a introducir el concepto de clave.

3.1. Claves

Algunas de las restricciones están definidas en torno a los conceptos de clave candidata, clave primaria y clave ajena. Veamos estos conceptos:

- **Clave candidata** de una tabla es un subconjunto K no vacío de atributos de esta tabla, que en cualquier instante de tiempo cumple las dos propiedades siguientes:
 - ♦ Unicidad: Nunca existen dos filas de la tabla con el mismo valor para K .
 - ♦ Minimalidad: Ninguno de sus atributos puede ser extraído de la clave sin que se pierda su función de unicidad.

Toda tabla tiene, por lo menos, una clave candidata.

Código	DNI	Nombre	Apellido 1	Apellido 2
1	1234567	Luis	Pérez	Rodríguez
2	8901234	Pedro	González	Gutiérrez

Esta tabla tendría dos claves candidatas: CODIGO y DNI.

La concatenación de NOMBRE, APELLIDO 1 y APELLIDO 2 podría ser clave candidata si aceptamos que no es posible tener dos clientes con el mismo nombre y apellidos.

Una clave candidata puede estar formada por más de un campo. En este caso, se dice que la clave candidata es **compuesta**.

- **Clave primaria** es una de las claves candidatas que se elige como primaria según criterios ajenos al propio modelo. En el modelo *relacional es obligatorio elegir una clave primaria* para cada tabla.

En el ejemplo anterior, deberíamos elegir una clave primaria de entre las 3 candidatas que tenemos: La combinación de nombre, apellido 1 y apellido 2 es la más larga y compleja de las 3, por lo que la podemos descartar. De entre el DNI y el código, podríamos elegir cualquiera de las dos.

- Una vez elegida la clave primaria, el resto de claves candidatas reciben el nombre de **claves alternativas**.

- **Clave ajena** de una tabla T_1 es un conjunto no vacío de atributos cuyos valores han de coincidir con los valores de la clave primaria de otra tabla T_2 . T_2 es la tabla referenciada y T_1 es la tabla que referencia.

La clave ajena y la clave primaria, por tanto, están definidas sobre el mismo dominio.

CLIENTES

Código	DNI	Nombre	Apellido 1	Apellido 2
1	1234567	Luis	Pérez	Rodríguez
2	8901234	Pedro	González	Gutiérrez

FACTURAS

num_factura	Cliente	Fecha	...
008765	1	24/08/2001	
008766	2	25/08/2001	
008767	2	26/08/2001	

En este ejemplo tenemos dos tablas que pertenecen a la misma Base de Datos: CLIENTES (ya conocida) y cuya clave primaria es “Código”, y FACTURAS, cuya clave primaria podría ser la columna “num_factura”. La tabla de FACTURAS tiene una columna llamada “Cliente” cuyo contenido hace referencia al cliente al que pertenece la factura. Observar que los valores de este campo no son ni el nombre del cliente ni el DNI, sino los códigos de cliente de la tabla CLIENTES. Esta columna, pues, está relacionada con la columna CODIGO de la tabla de clientes:

- La columna “Cliente” de la tabla FACTURAS es una clave ajena, ya que sus valores coinciden con los valores tomados por la clave primaria de otra tabla de la misma base de datos (tabla CLIENTES).
- La tabla CLIENTES es la tabla referenciada por la clave ajena, y la tabla FACTURAS es la que hace referencia.
- Ambas columnas están definidas sobre el mismo dominio subyacente (que se podría llamar, por ejemplo “Códigos de Cliente”).

3.2. Regla de integridad de la entidad

Es una de las restricciones inherentes del modelo. Dice lo siguiente:

“Ningún componente de la clave primaria de una tabla puede ser nulo”.

El concepto de nulo, importante en el modelo relacional, indica que una columna determinada en un momento dado, no tiene valor, lo que puede ser debido a distintas causas, como por ejemplo, que no conocemos su valor, o a que en esta fila concreta, el atributo no es aplicable.

CLIENTES

Cód	Nombre	Apellido 1	Apellido 2	Calle	Nº	Pis o	Pta	Ciudad
1	Luis	Pérez	Rodríguez	Mayor	14	6	30	Valencia
2	Pedro	González	Gutiérrez	Gran Vía	59			Madrid
3	Jorge	Rodríguez	García					

- En la fila 2, las columnas Piso y Puerta tienen los dos el valor NULO, debido a que no son aplicables pues esta persona vive en un chalet.

- En la fila 3, calle, Nº, Piso, Puerta y Ciudad tienen todos el valor NULO. En este caso es debido a que no se conoce la dirección en la que vive esta persona.

Cuando insertemos valores en una tabla de Base de Datos y estemos en alguno de los dos casos anteriores (valor no conocido o no aplicable), no debemos insertar en esas columnas un blanco, sino el valor nulo. En el tema dedicado a SQL veremos como se inserta el valor nulo en una tabla.

La regla de integridad de la entidad dice que una columna que sea clave primaria de una tabla nunca debe tener asignado el valor nulo. La clave primaria es la columna (o conjunto de columnas) que identifican de forma única una fila de la tabla (ya nunca pueden existir dos filas con el mismo valor de clave primaria). Si dicha clave primaria tiene el valor nulo en alguna fila, significa que no conocemos el valor de la clave primaria de esa fila, con lo que esa fila no será identificable.

3.3. Regla de integridad referencial

Es la restricción de usuario más importante del modelo y afecta a las claves ajenas:

“Una Base de Datos no debe contener valores de clave ajena sin concordancia”. Es decir, el valor de una clave ajena SIEMPRE debe coincidir con un valor de clave primaria de la tabla a la que apunta, o tener valor nulo.

Volvamos al ejemplo expuesto cuando se habló de las claves ajenas:

CLIENTES

Código	DNI	Nombre	Apellido 1	Apellido 2
1	1234567	Luis	Pérez	Rodríguez
2	8901234	Pedro	González	Gutiérrez

FACTURAS

num_factura	Cliente	Fecha	...	
008765	1	24/08/2001		OK
008766	2	25/08/2001		OK
008767	3	28/08/2001		Incorrecta
008768		01/09/2001		OK

- Las filas 1ª y 2ª de la tabla de facturas cumplen la regla de integridad referencial ya que los valores que toma su clave ajena (valores 1 y 2) existen en la clave primaria de la tabla CLIENTES.
- La fila 3ª **no** cumple la regla de integridad referencial ya que el valor que toma su clave ajena (3) no existe como código en la tabla CLIENTES. ESTE CASO ES INCORRECTO. **Una base de datos relacional bien diseñada nunca permitirá que un caso como éste se produzca.**
- La fila 4ª **si** cumple la regla de integridad referencial. Su clave ajena toma el valor NULO, es decir, no se conoce el código del cliente al que pertenece la factura, pero la regla de integridad referencial permite que una clave ajena tome como valor el nulo.

3.4. Reglas para claves ajenas

La regla de integridad referencial conlleva un problema importante que afecta a los borrados y modificaciones de datos en las tablas. Vamos a verlo con un ejemplo:

Suponer las tablas de CLIENTES y FACTURAS vistas en el apartado anterior. La tabla de facturas contendrá las filas 1ª, 2ª y 4ª, pero no la 3ª ya que, como hemos visto, esa fila era incorrecta.

Ahora suponer que el cliente 1 (Luis Pérez) decide darse de baja como cliente de la empresa. Lo normal es borrar la fila 1 de la tabla de Clientes, para que esta persona no conste como cliente de la empresa a partir de ese momento. Las tablas quedarán en el siguiente estado:

CLIENTES

Código	DNI	Nombre	Apellido 1	Apellido 2
2	8901234	Pedro	González	Gutiérrez

FACTURAS

num_factura	Cliente	Fecha	...	
008765	1	24/08/2001		Incorrecta
008766	2	25/08/2001		OK
008768		01/09/2001		OK

Al eliminar al cliente 1 de la tabla de clientes, la tabla de facturas queda inconsistente, ya que existe una fila (la correspondiente a la factura 008765) que incumple la regla de integridad referencial, pues tiene un valor de clave ajena (1) que no existe como clave primaria en la tabla de clientes.

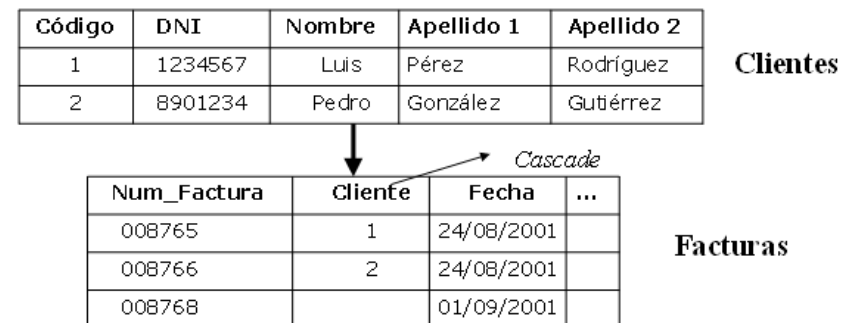
El mismo problema ocurriría si, en vez de borrar un cliente, modificáramos el valor de su clave primaria (por ejemplo, modificar el código del cliente 1 y asignarle el valor 4).

Existen distintas formas de evitar estos problemas. Una de ellas sería rechazar las operaciones de borrado o modificación de la clave, es decir, no permitir borrar o modificar la clave de filas apuntadas por alguna clave ajena. Otra sería permitir la operación, pero eliminar o modificar la clave de las filas apuntadas, etc.

Por tanto, al diseñar una base de datos, *además de definir las claves ajenas, es preciso especificar qué debe hacer el sistema en caso de que se presenten operaciones de borrado o modificación que atenten contra la integridad referencial*. Se pueden distinguir las siguientes posibilidades:

- **Operación restringida (RESTRICT):** No se permite borrar filas ni modificar el valor de su clave primaria si existen claves ajenas con valores concordantes. Si adoptamos esta opción, al intentar borrar el cliente 1 de la tabla CLIENTES, el SGBD lanzaría un mensaje de error indicando que no se puede eliminar el cliente porque tiene facturas asociadas.

- **Operación en cascada (CASCADE):** El borrado de filas o la modificación de la clave primaria lleva consigo el borrado o modificación de las filas con valores concordantes de clave ajena. En este caso, al borrar el cliente 1 de la tabla CLIENTES, el SGBD automáticamente borraría de la tabla de FACTURAS todas las filas asociadas al cliente 1. Si en vez de borrar lo que hacemos es cambiar el valor de la clave del cliente (por un 4, por ejemplo), el SGBD automáticamente escribiría un 4 en el campo “Cliente” de la tabla FACTURAS en todas las líneas asociadas al cliente 1.



- **Operación con puesta a nulos (SET NULL):** El borrado o modificación de las filas de la tabla que contiene la clave primaria lleva consigo la puesta a nulos de las claves ajenas con valores concordantes. Es decir, al borrar el cliente 1, no se borrarían las facturas asociadas al cliente pero se escribiría un valor NULO en la clave ajena de estas filas.

- **Operación con puesta a valor por defecto (SET DEFAULT):** El borrado o modificación de las filas de la tabla que contiene la clave primaria lleva consigo la puesta a un valor por defecto de las claves ajenas con valores concordantes. Es decir, al borrar el cliente 1, no se borrarían las facturas asociadas al cliente pero se escribiría un valor por defecto en la clave ajena de estas filas. El valor por defecto concreto debe ser especificado al definir la clave ajena.

- **Operación que dispara un programa:** El borrado o modificación de las filas de la tabla referenciada hace que se ejecute un programa previamente especificado.

Cuando se especifica la opción de CASCADE para una operación de borrado o modificación, si la cascada falla en algún nivel, la operación en conjunto falla y no se realiza ninguna modificación o borrado en ninguna tabla.

En este ejemplo tenemos dos claves ajenas: La columna “Cliente” de la tabla FACTURAS, definida como CASCADE, y la columna “Num_Factura” de la tabla LINEAS_FACTURAS, que se relaciona con la columna del mismo nombre de la tabla FACTURAS y definida como RESTRICT. Al borrar el cliente 1 se intentarán borrar las facturas asociadas a él (en este caso sólo 008765), puesto que en este caso la clave ajena es de tipo CASCADE. Pero la factura 008765 tiene 3 líneas asociadas. Estas líneas están definidas como RESTRICT lo que implica que no se podrá borrar la factura. Como resultado de la operación conjunta, el SGBD no dejará borrar el cliente, por lo que la B.D. quedará en el estado inicial.

Código	DNI	Nombre	Apellido 1	Apellido 2
1	1234567	Luis	Pérez	Rodríguez
2	8901234	Pedro	González	Gutiérrez

Clientes

Num_Factura	Cliente	Fecha	...
008765	1	24/08/2001	
008766	2	24/08/2001	
008768		01/09/2001	

Facturas

Num_Factura	Línea	Producto	...
008765	1	Tornillo	
008765	2	Tuerca	
008765	3	Perno	
008766	1	Perno	
008766	2	Clavo	
008768	1	Tirador	

Lineas_Facturas

3.5. Otras restricciones de usuario

Una restricción de usuario es una regla que se aplica a una columna o conjunto de columnas en el momento de la definición de la tabla y que debe ser verificado por todas las filas.

Por ejemplo:

- El campo “Edad” debe ser mayor que 0 y menor que 120.
- El campo “Nº de hijos” debe ser mayor que 0 y menor que 20.
- El campo “Sueldo” debe estar comprendido entre 1.500 y 5.000 €
- Si el campo “Estado Civil” contiene el valor “Soltero”, el campo “DNI cónyuge” será nulo.
- etc.

Las restricciones de usuario se declaran en la definición de la tabla, por lo que el lenguaje DDL de definición de tablas contiene sentencias apropiadas para definirlas.

Cuando se intenta insertar en la tabla valores que incumplen alguna restricción de usuario, el SGBD lo detecta y emite un mensaje de error.

Tema 2. Preguntas de repaso

1. Define cardinalidad y grado en una tabla.
2. ¿Qué es un dominio? Pon un ejemplo.
3. Enumera las restricciones que deben cumplir las tablas en el modelo relacional.
4. Diferencia entre clave candidata, clave primaria, claves alternativas y clave ajena.
5. Integridad de la entidad. ¿Por qué no puede tener una clave primaria un valor nulo en alguna de sus ocurrencias?
6. Integridad referencial. ¿Una clave ajena puede tener valor nulo en su tabla?
- 7, ¿Cuántas tuplas hay en un ejemplar de una relación con cardinalidad 22?

8. En las siguientes tablas:

Código	CIF	Empresa	Representante
1	45678932N	Válvulas del Mediterráneo, S. L.	Ramón Giménez Marchante
2	45324567T	Carburos Metálicos, S. A.	Eduardo Trujillo Sanz
3	20678123Q	Rodamientos Elásticos, C. B.	Arturo Díaz González

Tabla de los datos de las empresas proveedoras con sus representantes.

Núm_factura	Proveedor	Fecha	...
008765	2	24/03/08	
008766	3	05/02/08	
008767	2	22/02/08	

Tabla de los datos de las compras que se realizan a los proveedores.

En este caso el campo Proveedor es una clave ajena. Si desaparece el proveedor 3, ¿qué ocurre? ¿Cómo se puede solucionar? Comenta las diferentes alternativas que se ofrecen ante un caso similar.

9, ¿Qué es la integridad referencial?