

LAS FORMAS NORMALES

Rafael Fernández Ortiz

Las **formas normales (NF)** proporcionan los criterios para determinar el grado de *vulnerabilidad* de una tabla frente a inconsistencias y anomalías lógicas, es decir, **cuanto más alta sea la forma normal aplicable a una tabla, mas difícil será que se desestructure frente a los distintos cambios que se puedan hacer en una tabla.**

Las formas normales son aplicadas a las tablas de una base de datos. Decir que una base de datos está en la forma normal N es decir que todas sus tablas están en la forma normal N.

Así, la *normalización* es el proceso de descomposición en el diseño de las relaciones/tablas en otros de menor tamaño de forma que cumplan las *restricciones* necesarias que hacen que se definan como formas normales.

Las bases de datos relacionales se normalizan para: evitar la redundancia de datos, disminuir problemas de actualización de los datos en las tablas y proteger la integridad de los datos, esto es, proteger la correctitud y completitud de la información en dicha base.

Ejemplo: cuando los contenidos se modifican con sentencias *insert*, *delete* o *update*, la integridad de los datos almacenados puede perderse de muchas maneras, pueden añadirse datos no validos o descuadrarse.

Cada tabla tiene una **forma normal más alta (HNF)**, esto es, por definición:

Una tabla siempre satisface los requisitos de su HNF y de todas las formas normales más bajas que su HNF. También, por definición, una tabla no puede satisfacer los requisitos de ninguna forma normal más arriba que su HNF.

En general, las primeras tres formas normales son suficientes para cubrir las necesidades de la mayoría de las bases de datos. El creador de estas 3 primeras formas normales (o reglas) fue [Edgar F. Codd](#)

Para tratar de explicar las distintas formas normales, se ha de conocer el concepto de **Dependencia Funcional**.

Una *dependencia funcional* es una conexión entre uno o mas atributos. El concepto de dependencia funcional aparece en varios contextos de la matemática y la lógica. Véase:

Sean X e Y atributos (o conjunto de atributos) de una [relación](#) R . Se dice que Y depende funcionalmente de X (se denota por $X \rightarrow Y$) si cada valor de X tiene asociado un solo valor de Y .

En esta relación, a X se le denomina [determinante](#). Esto es, se define como determinante el atributo del cual depende funcionalmente algún otro atributo.

Propiedades de la dependencia funcional:

- **Reflexiva:** Si Y esta contenido en X entonces Y depende funcionalmente de X ($X \rightarrow Y$)
- **Argumentativa:** Si $X \rightarrow Y$, entonces $ZX \rightarrow ZY$, es decir, que si se le añade un argumento de más a X entonces Y también lo tendrá.

Ej: Si $DNI \rightarrow Nombre$, entonces $DNI, Dirección \rightarrow Nombre, Dirección$

- **Transitiva:** Si $X \rightarrow Y$, y $Y \rightarrow Z$, entonces $X \rightarrow Z$

De dichas propiedades se sacan:

- **Unión:** Si $X \rightarrow Y$, $X \rightarrow Z$, entonces $X \rightarrow YZ$
- **Pseudo-Transitiva:** $X \rightarrow Y$ y $WY \rightarrow Z$, entonces $WX \rightarrow Z$
- **Descomposición:** $X \rightarrow Y$ y Z esta incluido en Y , entonces $X \rightarrow Z$

LA PRIMERA FORMA NORMAL – 1FN

Una tabla está en 1FN si:

- Todos los atributos son **atómicos**. Un atributo es atómico si los elementos del dominio son simples e indivisibles.

Ejemplo: El atributo direccion no es atómico pues puede dividirse en Domicilio, Ciudad, Estado y Codigo Postal. En nuestro caso, el atributo nombre del producto es atómico.

- La tabla contiene una clave primaria única..
- La clave primaria no contiene atributos nulos.
- No debe existir variación en el número de columnas.
- Los campos no clave deben identificarse por la clave (Dependencia Funcional)

Ejemplo: Si tienes el atributo DNI como clave, los demas atributos asociados en la misma tabla como pueden ser *Nombre*, *Apellido1*, *Dirección*, etc. Tienen una dependencian funcional al DNI. En nuestro caso, los atributos **NombreProducto** y **CosteActual** en la tabla Productos se identifican por la clave **IdProducto**, pues existe una dependencia funcional entre ellos y la clave.

- Debe existir una independencia del orden tanto de las filas como de las columnas, es decir, si los datos cambian de orden no deben cambiar sus significados.

Esta forma normal elimina los valores repetidos dentro de una Base de Datos.

LA SEGUNDA FORMA NORMAL – 2FN

Para hablar de la segunda forma normal, debemos retomar el tema de la dependencia funcional.

Se dice que una dependencia es **completamente funcional** o que los atributos **depende completamente**, si al eliminar los atributos A de X se tiene que la dependencia no es mantenida. Esto es, que si A pertenece a X y $X \rightarrow Y$, entonces Y no depende de $X \setminus A$.

Ejemplo: Dados $X = \{DNI, IdProyecto\}$, $Y = \text{Horas de trabajo}$, se tiene que $X \rightarrow Y$, sin embargo si solo tenemos *DNI* o *IdProyecto*, es decir $X \setminus A$ con A alguno de dichos atributos, la dependencia no se mantiene pues con un solo atributo no puedes saber las horas de trabajo. Sin embargo si se tiene $X = \{DNI, IdProyecto\}$ $Y = \text{Nombre empleado}$, si tan solo tenemos *DNI* o *IdProyecto*, se puede sacar *Nombre empleado*, a esto se le llama **dependencia parcial**.

Entonces, una tabla está en 2FN si está en 1FN y si los atributos que no forman parte de ninguna clave dependen de forma completa de la clave principal.

LA TERCERA FORMA NORMAL – 3FN

La tabla se encuentra en 3FN si es 2FN y si no existe ninguna dependencia funcional transitiva en los atributos que no son clave.

Ejemplo: $Zona \rightarrow País$ y $País \rightarrow Ciudad$ entonces $Zona \rightarrow Ciudad$

Una condición suficiente para que una tabla este en la **tercera forma normal** es que si, para toda dependencia funcional $X \rightarrow Y$, se cumple al menos una de las dos:

1. X es superllave o clave.
2. Si Y es miembro de alguna clave en la tabla

FORMA NORMAL DE BOYCE-CODD - FNBC

La tabla se encuentra en FNBC si está en 3FN y **los únicos determinantes son claves candidatas**. Es una versión ligeramente más fuerte de la [Tercera forma normal](#) (3FN).

Ejemplo: Consideremos una empresa donde un trabajador puede trabajar en varios departamentos. En cada departamento hay varios responsables, pero cada trabajador sólo tiene asignado uno.

Tendríamos una tabla con las columnas:

<i>IdTrabajador</i>	<i>IdDepartamento</i>	<i>IdResponsable</i>
---------------------	-----------------------	----------------------

La única clave candidata es *IdTrabajador* (que será por tanto la clave primaria).

Si añadimos la limitación de que el responsable sólo puede serlo de un departamento, este detalle produce una dependencia funcional ya que:

$Responsable \rightarrow Departamento$

Por lo tanto hemos encontrado un determinante (*IdResponsable*) que sin embargo no es clave candidata.

Recordamos: definimos determinante al conjunto del cual salen las dependencias, es decir, si $X \rightarrow Y$, el determinante es X.

Rara vez una tabla en 3FN no cumple las condiciones de FNBC.

OTRAS FORMAS NORMALES

Existen mas formas normales, que son la 4FN, 5FN y la forma normal de dominio/clave, sin embargo estas formas son a nivel teórico, en la practica o son imposibles o redundantes.

En principio se entendería que para llegar a la forma normal 3, por ejemplo, habría que pasar por cada una de las formas normales, pero realmente esto no es necesario, se puede establecer desde el origen tablas con la forma normal 3 y, lo ideal, ya que no supone apenas coste, es conseguir formas normal mas alta por encima del correspondiente, es decir, si tienes una tabla en forma normal 3, creada tal cual desde el principio, lo normal seria tener HNF4, pero puedes tener 5HNF.

EJEMPLOS

Veamos un ejemplo de una tabla que no esté en ninguna forma normal

ID	NOMBRE	APELLIDO	DIRECCIÓN	CP	LOCALIDAD	PAÍS
1	Rafael	Fernández	C/ Palito 1	29100	Málaga	España
2	Vincent Willem	van Gogh	C/ Oreja 1 C/ Pinturas 900	32567	Zundert	Países Bajos

Esta tabla no está en 1NF pues para el ID 2, tiene dos direcciones.

Para pasarlo a 1NF, se hace el siguiente cambio:

ID	NOMBRE	APELLIDO	DIRECCIÓN	CP	LOCALIDAD	PAÍS
1	Rafael	Fernández	C/ Palito 1	29100	Málaga	España
2	Vincent Willem	van Gogh	C/ Oreja 1	32567	Zundert	Países Bajos
2	Vincent Willem	van Gogh	C/ Pinturas 900	32567	Zundert	Países Bajos

Ahora bien, si deseamos pasarlo a la 2NF, tan solo debemos crear una tabla que contenga los atributos *Nombre y Apellido*, y retirarlo de la tabla que teníamos, quedando el siguiente esquema:

ID	NOMBRE	APELLIDO
1	Rafael	Fernández
2	Vincent Willem	van Gogh

ID	DIRECCIÓN	CP	LOCALIDAD	PAÍS
1	C/ Palito 1	29100	Málaga	España
2	C/ Oreja 1	32567	Zundert	Países Bajos
2	C/ Pinturas 900	32567	Zundert	Países Bajos

Siguiendo con el ejemplo, si deseamos pasarlo a la 3NF, debemos ver donde están las dependencias funcionales transitivas y seguido se crea una nueva tabla con los atributos que tienen dicha dependencia y eliminándose de la tabla original, es decir:

ID	NOMBRE	APELLIDO
1	Rafael	Fernández
2	Vincent Willem	van Gogh

ID	PAÍS
1	España
2	Países Bajos

PAÍS	DIRECCIÓN	CP	LOCALIDAD
España	C/ Palito 1	29100	Málaga
Países Bajos	C/ Oreja 1	32567	Zundert
Países Bajos	C/ Pinturas 900	32567	Zundert