

Problema

Información redundante: está almacenada en varios sitios; se almacena cuando no hace falta, porque puede calcularse. Efectos colaterales: borrados de una cosa, afectan a otras, porque están unidos en un solo registro.

Solución: formas normales, de progresiva exigencia. En teoría, cuanto mayor sea la normalización, más eficiente es el sistema. Conduce a dividir la información en más tablas.

Primera forma normal

Todos los atributos tienen siempre valores atómicos.

Esto es tan básico, que ya viene de serie. No se pueden hacer bases de datos relacionales sin este requisito. Pero podemos cometer el error de poner como atributos atómicos cosas que en realidad no lo son.

Ejemplos

Banco: conjunto de cuentas de un cliente, conjunto de clientes con acceso a una cuenta. Si hay varios ítems que almacenar, posiblemente necesites otra tabla que sale de la relación uno a muchos. No valen varias columnas ítem 1, ítem 2, etc. porque no sabes la cantidad exacta.

Dependencias

Una dependencia es una implicación, de forma que conocido el valor de una o varias variables o atributos, otro u otros vienen dados. Algunas salen de las asociaciones o relaciones y otras del significado o requisitos de los atributos.

Por ejemplo, una relación uno a uno conlleva dos dependencias: del identificador del primero al segundo y viceversa. Una relación uno a muchos quiere decir que el identificador del lado muchos implica el del lado uno.

Segunda forma normal

Añade que cualquier atributo de una tabla sea dependiente del identificador.

Esto se puede conseguir poniendo en varias tablas información que estaba originalmente en una sola. Sacando de la tabla las variables que no vienen obligadas por el identificador.

Ejemplo

Identificador	Fecha de nacimiento	Edad
---------------	---------------------	------

Edad no depende del identificador, sino de la fecha de nacimiento, luego no estaría en segunda forma normal

Nº de pedido (identificador)	Nº de producto (identificador)	Nombre de producto
------------------------------	--------------------------------	--------------------

Aquí hay un identificador compuesto y el nombre del producto depende sólo del número de producto, luego no está en segunda forma normal. El nombre de producto debería estar en la tabla Productos.

Tercera forma normal

Añade que en cualquier dependencia no trivial uno de los dos términos es identificador, o en el caso del dependiente, al menos parte de un identificador. Si hay dependencias que no sean del identificador, se sacan a otra tabla.

Hasta aquí es habitual pedirlo.

Ejemplo

Nº de pedido (identificador)	Precio	IVA
------------------------------	--------	-----

Aquí IVA depende de precio, que no es identificador, luego no está en tercera forma normal.

Forma normal de Boyce-Codd

No es habitual, pero puede darse el caso.

Añade que en cualquier dependencia el independiente sea identificador.

Ejemplo

Estudiantes que se pueden apuntar a varios cursos. Un curso puede tener varios profesores. Cada profesor sólo da en un curso.

Supongamos que hacemos una tabla que tenga las columnas: idestudiante, curso, profesor y que la clave primaria es curso-identudiante

Existe la dependencia profesor=>curso

Esta tabla está en tercera forma normal, pero no en forma Boyce-Codd porque profesor no es clave.

La solución es poner dos tablas:

- idestudiante, idprofesor
- idprofesor, profesor, curso

Denormalización

A veces se prefieren tablas no normalizadas por rendimiento. Por ejemplo para sacar más datos con menos consultas o con consultas más simples.

- Almacenar valores calculables: si se pide mucho, las variables implicadas en sí no interesan, cambian poco y están en varias tablas. Debería calcularse al modificar.
- Juntar tablas: provoca duplicación de información, pero acelera consultas. Ejemplo: si guardamos archivos con referencia al id de la carpeta, podemos meter los datos de la carpeta

en la tabla de archivos, y quitar la tabla de carpetas. Ídem si son variables que no cambian y sólo unos pocos valores posibles.

- Poner columnas para elementos de conjunto (sí, lo que dijimos que no hicieras cuando vimos la primera forma normal). Sobre todo si la cantidad es fija. También puede meterse sólo una de ellas (la más importante, la más reciente,...)
- Saltos en las implicaciones: si $A \rightarrow B$ y $B \rightarrow C$, en lugar de tener en A sólo el id de B, puedo tener también el de C

Otros ejemplos

- Almacenar la fecha de expiración de algo que puede deducirse de la inicial del siguiente
- En una jerarquía almacenar el escalón, en vez de deducirlo de las conexiones