

Gestión de bases de datos

Diseño lógico de bases
de datos

¿Qué es la normalización?

Es el proceso por el que se obliga a los atributos de un diseño el cumplimiento de ciertas formas normales.

¿Por qué se normaliza?

Cuando se pasa del modelo entidad-relación al modelo relacional hay que revisar la calidad de acuerdo a ciertos criterios hasta que se alcance el diseño óptimo.

Una vez correctamente normalizado se pasará al modelo físico.

Objetivos de las formas normales

- ❑ Evitar la redundancia de datos. Para ello se deberá guardar la información de cada hecho una única vez. Así se reduce el espacio de almacenamiento.
- ❑ Guardar los hechos distintos en sitios distintos y así evitar problemas al operar con los datos.

Formas normales

Cada forma normal superior incluye a todas las anteriores.

Dependencia funcional

Un atributo A depende funcionalmente de un atributo B ($B \rightarrow A$) si cada valor de B tiene asociado en todo momento un único valor de A (B implica A siendo B el implicante).

Dependencia funcional

Ejemplo:

SOCIOS(CódigoSocio, Nombre)

CódigoSocio->Nombre

Un código de socio solo puede tener asociado un único nombre. A través del código de socio se localiza a un único nombre.

Dependencia funcional completa

Teniendo una combinación de atributos $A(A_1, A_2, \dots)$ se dice que B tiene dependencia funcional completa de A ($A \Rightarrow B$) si depende funcionalmente de A pero no depende de ningún subconjunto del mismo.

Dependencia funcional completa

Ejemplo:

CAPÍTULOS(Temporada, Orden, Título,
Fecha1aEmision)

Temporada, Orden => Fecha1aEmision

La fecha 1ª emisión es única para la combinación de Temporada y Orden y sin embargo en una misma temporada puede emitirse diferentes capítulos en fechas distintas luego

Temporada -/> Fecha1aEmision

Dependencia funcional transitiva

Si tenemos una tabla con varios atributos A, B y C, donde se cumpla que $A \rightarrow B$, $B \rightarrow C$ y $B \not\rightarrow A$, entonces se dice que A depende transitivamente de C, o $X \twoheadrightarrow C$

Dependencia funcional transitiva

Ejemplo:

PRODUCTOS(Código, Nombre,
Fabricante, País)

Código -> Fabricante

Fabricante -> País

Código - -> País

Forma normal 1 (FN 1)

No se permite que en una tabla haya atributos que puedan tomar más de un valor.

Título	Editorial	Autor
Libro A	Editorial	Autor A Autor B



Título	Editorial	Autor
Libro A	Editorial	Autor A
Libro A	Editorial	Autor B

Forma normal 2 (FN 2)

Los atributos que no sean clave deben ser hechos en la totalidad de la clave.

Relación:

(pieza, almacén, cantidad, direcciónAlmacen)

La dirección del almacen no es un hecho en la conjunción pieza,almacén, solo en el almacén. Sin embargo cantidad sí lo es de pieza, almacén.

Solución:

(pieza, almacén, cantidad)

(almacén, direcciónAlmacen)

Forma normal 3 (FN 3)

No hay atributos no clave que dependan de forma transitiva de la clave.

Ejemplo:

PRODUCTOS(código, nombre, fabricante, país)

Código -> fabricante

Fabricante -> país

Código - -> país (fabricante -/> código)

País es un atributo no clave que depende transitivamente de la clave código por lo que **no está en FN3**.

Forma normal de Boyce-Codd (FNBC)

Modelo en FN3 y que todo implicante de la tabla sea una clave candidata.

Ejemplo:

NOTAS(DNI-P, NOMBRE-P, DNI-A, CALIF)

Está en FN1 porque no hay atributos que puedan tomar más de un valor. Está en FN2 porque hay dependencia completa de CALIF y NOMBRE-P es atributo principal. Está en FN3 porque no hay dependencias transitivas.

No está en FNBC porque NOMBRE-P y DNI-P son determinantes (se pueden usar para obtener el valor en otros atributos) pero no son claves candidatas.



CEU

*Centro de Estudios
Profesionales*

Fundación San Pablo Andalucía