

RESUMEN DEPENDENCIAS FUNCIONALES ENTRE ATRIBUTOS

Los conceptos sobre los que se basa el análisis de dependencias entre atributos, que se lleva a cabo en el proceso de normalización antes indicado, son los siguientes:

- **Dependencia Funcional (definición general):** dados los atributos X e Y, se dice que Y depende funcionalmente de X, sí, y solo sí, para cada valor de X sólo puede existir un valor de Y. La dependencia funcional siempre se establece entre atributos de una misma tabla o relación R. Esto es lo mismo que decir que si dos tuplas de R tienen el mismo valor para su atributo X forzosamente han de tener el mismo valor para el atributo Y.

El atributo X se denomina determinante, ya que X determina el valor de Y. Para representar esta dependencia funcional utilizamos la siguiente notación: $X \rightarrow Y$ (se lee "X determina a Y").

Hay que indicar que X e Y podrían ser un solo atributo o un conjunto de ellos.

Dentro de las dependencias funcionales, existen los siguientes tipos:

- **Dependencia funcional completa:** dados los atributos X_1, X_2, \dots, X_k e Y, se dice que Y depende funcionalmente de forma completa de X_1, X_2, \dots, X_k , si y solo si Y depende funcionalmente del conjunto de atributos X_1, X_2, \dots, X_k , pero no de ninguno de sus posibles subconjuntos.
 - **Dependencia funcional elemental:** una dependencia funcional es elemental si Y es solo un atributo, y no es un conjunto de ellos.
 - **Dependencia funcional trivial:** ocurre cuando X es un conjunto de atributos, e Y es un subconjunto de X.
- **Dependencia transitiva:** dados tres atributos X, Y y Z de la relación R, se dice que existe una dependencia transitiva entre X y Z: $Y \rightarrow Z$, si:

Y depende funcionalmente de X: $X \rightarrow Y$ y,

Z depende funcionalmente de Y: $Y \rightarrow Z$.

(X, Y y Z podrían ser un solo atributo o un conjunto de ellos).

- **Dependencia multivaluada:** en una relación con los atributos X, Y y Z existe una dependencia multivaluada de Y con respecto a X si los posibles valores de Y para un par de valores de X y Z dependen únicamente del valor de X y son independientes de Z (*definición según Ponniah*)

Se representa de la siguiente manera: $X \twoheadrightarrow Y$, y se lee "X multidetermina Y"

- **Dependencia en unión (reunión):**

Definimos los conceptos

- **Proyección:** operación de crear una nueva tabla a partir de otra cuyos elementos forman un subconjunto de la original. Se incluyen todas las filas y algunas columnas.
- **Unión (reunión):** formar a partir de dos tablas una nueva con todos los campos de una de ellas y los registros de ambos, excepto los repetidos. Ambas tablas han de tener el mismo grado y las mismas columnas.

Se dice que una relación satisface la dependencia de reunión ($X, Y, \dots Z$) si y solo si la relación es igual a la reunión de sus proyecciones según X, Y, \dots, Z . Donde X, Y, \dots, Z son subconjuntos de atributos de la relación.

RESUMEN NORMALIZACIÓN

El proceso de normalización se basa en el análisis de las dependencias entre atributos. Para lo cual se tienen en cuenta los conceptos sobre dependencias vistos anteriormente.

Se trata de llevar a cabo una serie de etapas consecutivas en las que se aplican las propiedades de cada una de las formas normales definidas por Codd.

A continuación se exponen los requisitos a cumplir por las tablas de nuestra base de datos según la forma normal que apliquemos:

- **1ª Forma Normal (1FN):** una tabla está en Primera Forma Normal si, y solo si, todos los atributos de la tabla contienen valores atómicos, es decir, no hay grupos repetitivos (un atributo de una tupla no puede tomar más de un valor). Es decir, si los atributos no clave dependen funcionalmente de la clave.

Realmente toda tabla relacional cumple dicho requisito.

¿Cómo se normaliza a 1FN?

- a) A partir de la tabla inicial se crea una nueva tabla cuyos atributos son los que presentan dependencia funcional de la clave primaria. La clave de esta tabla será la misma clave primaria de la tabla inicial. Esta nueva tabla ya estará en 1FN.
- b) Con los atributos restantes se crea otra tabla y se elige entre ellos uno que será la clave primaria de dicha tabla. Se comprobará que esta segunda tabla esté en 1FN. Si es así, el proceso termina. Si esta segunda tabla no está en 1FN, se tomará como tabla inicial y se repetirá el proceso.

- **2ª Forma Normal (2FN):** una tabla está en Segunda Forma Normal si, y solo si, está en 1FN, y además todos los atributos que no pertenecen a la clave dependen funcionalmente de forma completa de ella. Es obvio que una tabla que esté en 1FN, y cuya clave primaria esté compuesta por un único atributo estará en 2FN. Pero puede pasar que se tengan dependencias parciales,

es decir, dependencias basadas en una parte de la clave primaria que producirán redundancias al insertar nuevas filas en la tabla.

¿Cómo se normaliza a 2FN?

- a) A partir de la tabla inicial se crea una nueva tabla con los atributos que dependen funcionalmente de forma completa de la clave. La clave de esta tabla será la misma clave primaria de la tabla inicial. Esta nueva tabla ya estará en 2FN.
- b) Con los atributos restantes (los que dependen solo de parte de la clave) se crea otra tabla que tendrá por clave ese subconjunto de atributos de la clave inicial de los que dependen de forma completa. Se comprueba si esta tabla está en 2FN. Si es así, esta segunda tabla ya está normalizada y el proceso termina. Si no está en 2FN, se toma esta tabla como inicial y se repite el proceso.

- **3ª Forma Normal (3FN):** una tabla está en Tercera Forma Normal si, y solo si, está en 2FN, y además cada atributo que no está en la clave primaria no depende transitivamente de la clave primaria. Es decir, que no existen atributos no clave que puedan ser conocidos mediante otro atributo no clave.

¿Cómo se normaliza a 3FN?

- a) A partir de la tabla inicial se crea una nueva tabla con los atributos que no poseen dependencias transitivas de la clave primaria. Esta tabla ya estará en 3FN.
- b) Con los atributos restantes se crea otra tabla con los dos atributos no clave que intervienen en la dependencia transitiva, y se elige uno de ellos como clave primaria si cumple los requisitos para ello. Se comprueba si esta nueva tabla está en 3FN. Si es así, la tabla ya está normalizada y el proceso termina. Si no está en 3FN, se toma como tabla inicial y se repite el proceso.

- **Forma Normal de Boyce-Codd (FNBC):** una tabla está en Forma Normal de Boyce-Codd si, y solo si, está en 3FN, y todo determinante es una clave candidata. Un determinante será todo atributo simple o compuesto del que depende funcionalmente de forma completa algún atributo de la tabla. Aquellas tablas en las que todos sus atributos forman parte de la clave primaria estarán en FNBC. Por tanto, si encontramos un determinante que no es clave candidata, la tabla no estará en FNBC. Esta redundancia suele ocurrir por una mala elección de la clave.

¿Cómo se normaliza a FNBC?

Para normalizar a FNBC se tendrá que descomponer la tabla inicial en dos, evitando la pérdida de información en dicha descomposición.

- **4ª Forma Normal (4FN):** una tabla está en Cuarta Forma Normal si, y solo si, está en FNBC, y las únicas dependencias funcionales multivaluadas que existen son las dependencias funcionales de la clave con los atributos no clave.

¿Cómo se normaliza a 4FN?

En caso de tener dos o más atributos multivaluados independientes en la misma tabla, se descompondrá la tabla en otras dos, manteniéndose en cada una de ellas una dependencia múltiple, siendo la clave de las nuevas tablas la combinación de todos los campos.

- **5ª Forma Normal (5FN):** una tabla está en Quinta Forma Normal si, y solo si, está en 4FN, y las únicas dependencias que existen son las dependencias de unión de una tabla con sus proyecciones relacionándose entre sí mediante la clave primaria o cualquier clave candidata.

¿Cómo se normaliza a 5FN?

Se divide la tabla original que está en 4FN en tantas como sea necesario. Dichas tablas tendrán en común los campos que forman la clave primaria en la tabla original.

En caso de que la tabla original que está en 4FN tenga gran cantidad de atributos o bien tenga pocos atributos pero muchos registros, se divide la tabla en tantas como sea necesario. Dichas tablas tendrán en común con una o más de las otras tablas la existencia de una o más claves ajenas.