

1 Normalización

1.1 Introducción

La normalización es una teoría formal que ayuda a valorar la calidad de un diseño. La bondad de la calidad se ve en dos niveles: lógico (interpretación de los esquemas relacionales y el significado de sus atributos) y de manipulación (cómo se almacenan y actualizan las tuplas en las relaciones base).

La normalización es una técnica en el diseño de bases de datos mediante la cual se pueden agrupar los atributos en relaciones, logrando mínima redundancia; la agrupación es tal que no existe pérdida de información (lo cual significa que la relación original se vuelve a obtener aplicando junta natural a las relaciones normalizadas).

La normalización optimiza una base de datos en cuanto al almacenamiento de la información, no en cuanto al tiempo de procesamiento de la misma. A partir de un conjunto de relaciones originales se llega a un nuevo conjunto donde la redundancia es mínima y controlada.

Pautas informales de diseño

Son *medidas informales* de la calidad del diseño de los esquemas relacionales de una base de datos, las cuales se pueden aplicar “intuitivamente” durante este diseño.

Pauta 1 – Dar semántica a los atributos de las relaciones

Diseñar un esquema relacional de tal manera que sea fácil explicar su significado. Este significado especifica *cómo* interpretar los valores almacenados en una tupla de la relación (considerando que una relación puede interpretarse como un conjunto de *hechos*, esto es, de aseveraciones verdaderas).

No hay que combinar atributos de varios tipos de entidades y tipos de vínculos en una sola relación. Intuitivamente, si un esquema de relación corresponde a un tipo de entidad o a un tipo de vínculo, el significado tiende a ser claro. De otra manera, tiende a ser una mezcla de múltiples entidades y vínculos y, por lo tanto, semánticamente confuso.

Ejemplo:

Grupos(IdProf, NomProf, Categoría, ClaveM, Nombre, Creds, ClaveG, Salón)

Aunque este diseño es lógicamente correcto, es **pobre** porque *mezcla atributos de entidades distintas del mundo real*.

Pauta 2 – Reducir la redundancia en tuplas, con lo cual se eliminan anomalías de actualización

Una de las metas del diseño de los esquemas relacionales es minimizar el espacio de almacenamiento ocupado por las relaciones base. Esto se logra cuando se reduce al mínimo la *redundancia* que pueda existir en los datos de estas relaciones.

Por lo tanto, hay que diseñar los esquemas de las relaciones base de tal manera que se tenga el mínimo de redundancia y así no ocurran anomalías de inserción, eliminación o modificación. Si estas anomalías están presentes, hay que documentarlas claramente para que los programas que actualizan la base de datos operen correctamente.

Pauta 3 – Reducir valores nulos en tuplas

Tanto como sea posible, evitar colocar en una relación base atributos cuyos valores puedan ser nulos. Si los nulos son inevitables, asegurarse de que sólo se apliquen en casos excepcionales y que no se apliquen a la mayoría de las tuplas en la relación.

Pauta 4 – No generar tuplas falsas

Diseñar los esquemas relacionales tal que puedan ser juntados aplicando condiciones de igualdad sobre los atributos que son clave primaria o clave externa de una manera que garantice que no se generarán tuplas falsas.

1.2 Dependencias funcionales

Definición: un esquema relacional **universal** es un esquema que describe a todos los atributos de una base de datos.

Ejemplo (usando el esquema relacional de la base de datos del mini-sistema escolar):

SE(IdProf, NomProf, Categoría, ClaveM, Nombre, Creds, ClaveG, Salón, CU, NomAl, Carr, Prom, Folio, Calif, Fecha)

Definición: dado un esquema relacional $R(A_1, A_2, \dots, A_n)$, una **dependencia funcional**, denotada por $X \rightarrow Y$, entre dos conjuntos de atributos X y Y , subconjuntos de R , especifica una restricción sobre las posibles tuplas que pueden formar una relación r de R . La restricción establece que dadas dos tuplas cualesquiera de r , si sus valores en X coinciden, también deben coincidir sus valores en Y .

Se dice que X **determina funcionalmente** a Y , o bien que Y **depende funcionalmente** de X .

Definición: sea un esquema relacional $R(A_1, A_2, \dots, A_n)$; se dice que un subconjunto X de los atributos de R es **clave candidata** de R si se cumplen las dos propiedades siguientes:

- a) Unicidad: $X \rightarrow A_1, A_2, \dots, A_n$
- b) Minimalidad: ningún subconjunto $Y \subseteq X$ es tal que: $Y \rightarrow A_1, A_2, \dots, A_n$

1.3 Formas normales basadas en claves primarias

Forma normal: es un término que se utiliza para indicar que un esquema relacional cumple con un cierto conjunto de propiedades.

La **normalización de los datos** puede ser vista como un proceso durante el cual esquemas relacionales insatisfactorios pueden ser descompuestos separando sus atributos en esquemas relacionales más pequeños que posean ciertas propiedades aceptables.

No hay pérdida de información al normalizar; esto significa que si se aplica junta natural a las relaciones finales producto de la normalización, se obtendrán las relaciones originales que se tenían antes de aplicarla.

Existen cuatro formas normales (**1FN**, **2FN**, **3FN** y **FNBC**) que están basadas en las dependencias funcionales y en las restricciones de clave. La 4ª forma normal (**4FN**) está basada en el concepto de dependencia multivaluada y la 5ª (**5FN**) en el de dependencia de junta.

Si una relación está normalizada en n FN ($n \geq 2$), eso implica que ya se encuentra normalizada en $(n-1)$ FN.

Definiciones:

- Una **superclave** de un esquema de relación $R(A_1, A_2, \dots, A_n)$ es un conjunto de atributos $S \subseteq R$ con la propiedad de que para dos tuplas distintas t_1 y t_2 en cualquier relación legal r de R , $t_1[S]$ será distinto a $t_2[S]$.
- Una **clave** K es una superclave con la propiedad adicional de que al remover cualquier atributo de K , causará que K no sea más una superclave. Se dice que K es “mínima”, precisamente porque no se le pueden quitar atributos y seguir siendo una superclave.
- Una **clave candidata** es cualquier clave de un esquema de relación. Una de éstas es designada como la **clave primaria** y las demás se consideran como **claves secundarias** o **alternas**. Si hay varias claves candidatas, se escoge como clave primaria la que tenga menos atributos; si hay empate, se escoge entonces la que ocupe menos espacio.
- Un atributo de R es **principal** si es un miembro de cualquier clave de R . Un atributo es **no principal**, en caso contrario.

3.3.1 Primera forma normal (1FN)

Definición de 1FN: una relación está en 1FN si y sólo si los dominios de sus atributos sólo contienen *valores atómicos* (simples, indivisibles) y el valor de cualquier atributo en una tupla es un *valor individual* del dominio de dicho atributo.

La 1FN no permite tener un conjunto de valores, una tupla de valores, o una combinación de ambas, como un valor de atributo para una *tupla individual*; en otras palabras, la 1FN no permite “relaciones dentro de relaciones” o “relaciones como atributos de tuplas”.

Ejemplo: la siguiente relación no está en 1FN

CU	NomAl	Teléfono
10	Ana	5524-3211 5524-3325
20	José	5631-4415 5524-3611

Una posible normalización en 1FN podría ser:

CU	NomAl	Teléfono
10	Ana	5524-3211
10	Ana	5524-3325
20	José	5631-4415
20	José	5524-3611

¿Cuál sería la clave primaria?

Como se observa la 1FN no reduce la redundancia (al contrario, la incrementa), sin embargo fuerza a que las relaciones sólo tengan valores atómicos en los atributos de las tuplas, logrando con esto hacer posible el uso del álgebra o el cálculo relacional para su manipulación.

3.3.2 Segunda forma normal (2FN)

Definición de 2FN: una relación está en 2FN si y sólo si está en 1FN y cada atributo no principal es funcionalmente dependiente de la clave primaria completa.

3.3.3 Tercera forma normal (3FN)

Definición: dos atributos son mutuamente independientes si ninguno es funcionalmente dependiente del otro.

Definición de 3FN: una relación está en 3FN si y sólo si está en 2FN y todos sus atributos no principales son mutuamente independientes.

3.3.4 Forma normal de Boyce-Codd (FNBC)

Existen relaciones en 3FN que aún contienen redundancia susceptible de eliminarse; por ejemplo, la relación:

Grupos(CU, ClaveM, IdProf, Salón)

Donde:

- Cada materia la enseñan varios profesores.
- Cada profesor enseña sólo una materia.
- Cada materia a cada estudiante de la misma es impartida por un mismo profesor.
- Un estudiante cursa varias materias.

¿En qué parte está la redundancia innecesaria?

Definición: se le llama determinante (funcional) a un atributo (o conjunto de atributos) del cual depende funcionalmente en forma completa algún otro atributo (o conjunto de atributos).

Definición de FNBC: una relación está en Forma Normal de Boyce-Codd (FNBC) si y sólo si cada determinante es una clave candidata.

3.3.5 Cuarta forma normal (4FN)

Suponga que se da una relación no normalizada que tiene información sobre las materias, los profesores y los textos:

Cursos		
Materia	Profesor	Texto
Física	Rebeca Luis José Adriana	Mecánica básica Principios de óptica
Matemáticas	José Laura	Álgebra lineal Geometría analítica
Álgebra II	Laura Manuel	Álgebra lineal
	...	

Al llevar la relación anterior a 1FN tenemos:

Cursos

Materia	Profesor	Texto
Física	Rebeca	Mecánica básica
Física	Rebeca	Principios de óptica
Física	Luis	Mecánica básica
Física	Luis	Principios de óptica
Física	José	Mecánica básica
Física	José	Principios de óptica
Física	Adriana	Mecánica básica
Física	Adriana	Principios de óptica
Matemáticas	José	Álgebra lineal
Matemáticas	José	Geometría analítica
Matemáticas	Laura	Álgebra lineal
Matemáticas	Laura	Geometría analítica
Álgebra II	Laura	Álgebra lineal
Álgebra II	Manuel	Álgebra lineal
	...	

Aplicando las formas normales anteriores no se llega a reducción alguna; sin embargo, sigue existiendo redundancia innecesaria.

Definición de *dependencia multivaluada*: dada una relación, un atributo A de la misma *determina de manera multivaluada* a un atributo B, si un rango específico de valores de B está determinado por un valor particular de A, representándose como:

A \twoheadrightarrow B

Definición de 4FN: una relación está en cuarta forma normal (4FN) si y sólo si está en FNBC y no contiene dependencias multivaluadas.

1.4 Reglas de inferencia para las dependencias funcionales

Son reglas que pueden ser usadas para inferir nuevas dependencias a partir de un conjunto de dependencias existentes. Las más comunes son las siguientes:

IR1 (regla reflexiva)	Si $X \supseteq Y$, entonces $X \rightarrow Y$
IR2 (regla de aumento)	Si $X \rightarrow Y$, entonces $XZ \rightarrow YZ$
IR3 (regla transitiva)	Si $X \rightarrow Y$, $Y \rightarrow Z$, entonces $X \rightarrow Z$
IR4 (regla de descomposición)	Si $X \rightarrow YZ$, entonces $X \rightarrow Y$, $X \rightarrow Z$
IR5 (regla de unión)	Si $X \rightarrow Y$, $X \rightarrow Z$, entonces $X \rightarrow YZ$
IR6 (regla pseudo-transitiva)	Si $X \rightarrow Y$, $WY \rightarrow Z$, entonces $WX \rightarrow Z$

Nota: W, X, Y y Z, representan conjuntos de atributos de un esquema relacional.

1.5 Proceso de normalización

1. Construir el esquema relacional universal y su diagrama de dependencias funcionales y multivaluadas de acuerdo al enunciado del problema y a la semántica de los datos; determinar, además, una clave primaria.
2. Aplicar 1FN (para llevar a valores atómicos a los atributos multivaluados y a los compuestos).
3. Separar los atributos no principales para que todos ellos sean dependientes funcionalmente de una clave primaria completa; esto es, aplicar 2FN.
4. Separar los atributos no principales transitivamente dependientes; esto es, aplicar 3FN.
5. Aplicar FNBC, si corresponde.
6. Separar los atributos dependientes multivaluados; esto es, aplicar 4FN.
7. Obtener los esquemas relacionales de la base de datos ya normalizados.
8. Obtener el diagrama de entidad-vínculo correspondiente a los esquemas relacionales.
9. Revisar de nuevo el enunciado del problema para determinar los casos de condicionalidad y afinar el diagrama obtenido en el punto anterior.
10. Si aún no se ha creado la estructura de la base de datos, se puede hacer en este punto usando SQL y un DBMS concreto.