

TEMA 3: DISEÑO DE DATOS: **MLD NORMALIZADO.**

1. INTRODUCCIÓN.
2. EL MODELO RELACIONAL (MR).
3. MÉTODO FORMAL DE DISEÑO RELACIONAL.
4. OBTENCIÓN DEL MR A PARTIR DEL MER.
5. NORMALIZACIÓN.
 - 5.1. TEORÍA DE DEPENDENCIAS.
 - 5.2. FORMAS NORMALES.
6. DINÁMICA DEL MODELO (ÁLGEBRA RELACIONAL).

1. INTRODUCCIÓN.

Vamos a estudiar el Modelo de Datos Relacional propuesto por Codd en 1970. Este modelo constituye un intento de simplificar las estructuras de las bases de datos.

El documento de Codd propone un modelo de datos basado en la teoría de las relaciones. En dicha teoría los datos se estructuran lógicamente en forma de tablas, siendo el objetivo fundamental del modelo mantener la independencia de esta estructura lógica respecto al modo de almacenamiento y a otras características de tipo físico.

Los objetivos del modelo se pueden resumir en:

- Independencia física: Es decir, que el modo en que se almacenan los datos no influya en su manipulación lógica, y por tanto, los usuarios que acceden a esos datos no tengan que modificar sus programas por cambios en el almacenamiento físico.
- Independencia lógica: Esto es, que al añadir, eliminar ó modificar objetos de la base de datos, no repercuta en los programas y/o usuarios que están accediendo a subconjuntos parciales de los mismos.
- Flexibilidad: En el sentido de poder representar a cada usuario los datos de la forma en que éste prefiera.
- Uniformidad: Las estructuras lógicas de los datos presentan un aspecto uniforme, facilitando la concepción y manipulación de la base de datos por parte de los usuarios.
- Sencillez: Las características anteriores, así como unos lenguajes de usuario muy sencillos, producen como resultado que el modelo de datos relacional sea fácil de comprender y de utilizar por parte del usuario final.

2. EL MODELO RELACIONAL (MR).

En el modelo de CODD los datos se estructuran lógicamente en forma de tabla o relaciones. (Se entiende por relación un conjunto de filas y columnas).

Tabla Alumnos

<u>DNI</u>	Nombre	Apellidos	Edad
11111111-A	Pedro	González Mar	24
22222222-W	María	Fernández Marín	21
....

A las columnas de las tablas se les denomina **atributos** y representan las propiedades de la tabla. A las filas de las tablas se les denomina **tuplas** y representan las ocurrencias de la tabla; el número de atributos se denomina **grado** y el número de tuplas se denomina **cardinalidad**.

También podemos representar las tablas sin que aparezcan las tuplas.

Alumnos(DNI, Nombre, Apellidos, Edad)

La relación o tabla del modelo relacional tiene unas ciertas características:

- No puede haber tuplas duplicadas.
- El orden de las tuplas es irrelevante.
- El orden de los atributos es irrelevante.
- En el cruce de un atributo y una tupla solo puede haber un valor (tabla plana)

El conjunto sobre el que toman valores los atributos se denomina **dominio** del atributo.

El conjunto no vacío de atributos que identifica unívoca y mínimamente cada tupla se denomina **clave candidata**; el usuario elige una de las claves candidatas para identificar las tuplas, pasando a ser ésta la **clave primaria o principal (Primary Key)**.

Se dice que una clave primaria es compuesta cuando está formada por más de un atributo, el atributo o atributos que la forman se denominan atributos principales o primos mientras que el resto son los atributos no primos.

Conceptos básicos del Modelo

Integridad de entidad (1ª Regla de Integridad):

Ningún atributo primo puede tomar un valor nulo. Se llama valor nulo a un símbolo especial que representa un valor desconocido.

Descriptor:

Es un conjunto no vacío de atributos de una relación.

Clave ajena o extranjera (Foreign Key):

La clave ajena de una tabla es un descriptor que aparece en otra tabla (ó en la misma) como clave primaria; ambas claves, primaria y ajena, toman valores del mismo dominio.

Nos permite relacionar tablas.

Integridad Referencial (2ª Regla de Integridad):

Los valores de una clave ajena deben concordar con los de clave primaria o ser nulos.

3. MÉTODO FORMAL DE DISEÑO RELACIONAL.

En el Modelo Relacional, el diseño de una base de datos se puede abordar de dos formas distintas:

- Obteniendo el esquema relacional (tablas, atributos,...) directamente a partir de la observación de nuestro Universo del Discurso. Esta aproximación no será muy viable, salvo si se trata de un sistema muy simple ó se conocen muy bien las necesidades de los usuarios.
- Realizando el proceso de diseño en dos fases:
 - Obtener el Modelo Conceptual (MER)
 - Transformar el MER en Modelo Relacional, siguiendo determinadas reglas de transformación.

Finalmente, tener en cuenta que cuando se diseña una BD mediante el MR, al igual que ocurre con otros modelos, podemos obtener distintos esquemas relacionales, no todos ellos equivalentes, ya que unos van a representar la realidad mejor que otros.

4. OBTENCIÓN DEL MR A PARTIR DEL MER.

Transformación de entidades:

Cada entidad se convierte en una tabla que tiene como columnas los atributos de la entidad. El identificador de la entidad se convierte en la clave primaria de la tabla.

Transformación de interrelaciones:

- **Interrelaciones con correspondencia 1: N**

En este caso propagamos la clave. Se propaga el atributo principal de la entidad que tiene de cardinalidad máxima 1 a la que tiene de cardinalidad máxima N, desapareciendo el nombre de la relación. El atributo que hemos propagado sería una clave ajena de esa tabla.

- **Interrelaciones con correspondencia N:M**

Toda relación N:M se transforma en una tabla que tendrá como clave primaria la concatenación de los atributos clave de las entidades que asocia.

- **Interrelaciones con correspondencia 1:1**

En este caso también hemos de tener en cuenta las cardinalidades mínimas de las entidades que participan.

- Si una de las entidades posee cardinalidad (0,1) y la otra (1,1) conviene propagar la clave de la entidad con cardinalidad (1,1) a la tabla resultante de la entidad de cardinalidad (0,1).
- Si las dos entidades poseen cardinalidades (1,1) ó (0,1) se puede propagar la clave de cualquiera de ellas a la tabla resultante de la otra.

5. NORMALIZACIÓN

Dentro del contexto del Modelo Relacional el objetivo de la Normalización es conseguir un conjunto de relaciones óptimo que, una vez implementado físicamente, forme una estructura flexible y fácil de mantener y evite anomalías en operaciones de manipulación de datos.

Ejemplo: Tenemos una relación denominada ESCRIBE que almacena datos sobre autores de libros (Autor y Nacionalidad) y sobre los propios libros (Códigolibro, Título, Editorial y Año). La clave de la relación esta formada por Autor y Códigolibro.

ESCRIBE(Autor, Nacionalidad, Códigolibro, Título, Editorial, Año)

<u>Autor</u>	Nacionalidad	<u>Códigolibro</u>	Título	Editorial	Año
Codd, E.F.	Australiana	1	A Relational Model	Comm ACM	1970
Codd, E.F.	Australiana	2	The Relational Model	Addison Wesley	1990
De Miguel, A.	Española	3	Concepción y Diseño de BD	Ra-ma	1993
Piattini, M	Española	3	Concepción y Diseño de BD	Ra-ma	1993
Fernández, B.	Española	4	El modelo relacional de datos	Díaz de Santos	1987

Problemas que se plantean:

- Gran cantidad de redundancia: ya que la nacionalidad del autor se repite por cada ocurrencia del mismo, y algo análogo sucede, cuando un libro tiene más de un autor, con la editorial, el título y el año de publicación.
- Anomalías de modificación: Inadvertidamente podemos cambiar el nombre de la editorial en una tupla, sin modificarlo en el resto de las que corresponden al mismo libro, dando lugar a incoherencias.
- Anomalías de inserción: Si se quisiera incluir información sobre algún autor del que no hubiera ningún libro en la BD, no sería posible, al formar el atributo códigolibro parte de la clave primaria. Tampoco podríamos introducir obras_anónimas. (Regla de Integridad de Entidad). Además, la inserción de un libro que tuviese dos autores obligaría a incluir dos tuplas en la BD.
- Anomalías de borrado: Si queremos dar de baja un libro, también se perderían datos sobre sus autores (si éstos no habían escrito nada más que un libro), y viceversa, si borramos un autor, desaparecerían de nuestra BD los libros escritos por él (a no ser que el libro tuviera más de un autor).

Mediante la Teoría de la Normalización llegaríamos al siguiente esquema relacional:

LIBRO (Códigolibro, Título, Editorial, Año).

AUTOR (Nombre, Nacionalidad).

ESCRIBE (Códigolibro, Nombre).

Donde vemos que cada relación recoge un hecho bien determinado y concreto del mundo real.

LIBRO

<u>Códigolibro</u>	Título	Editorial	Año
1	A Relational Model	Comm ACM	1970
2	The Relational Model	Addison Wesley	1990
3	Concepción y Diseño de BD	Ra-ma	1993
4	El modelo relacional de datos	Díaz de Santos	1987

AUTOR

<u>Nombre</u>	Nacionalidad
Codd, E.F.	Australiana
De Miguel, A.	Española
Piattini, M.	Española
Fernández, B.	Española

ESCRIBE

<u>Códigolibro</u>	<u>Nombre</u>
1	Codd, E.F.
2	Codd, E.F.
3	De Miguel, A.
3	Piattini, M.
4	Fernández, D.

La Teoría de la Normalización consiste en obtener esquemas relacionales que cumplan unas determinadas condiciones. A estas condiciones se las conoce como Formas Normales. Actualmente, en el Modelo Relacional se contemplan hasta seis formas normales diferentes: 1FN, 2FN, 3FN, FNBC, 4FN y 5FN. En este curso sólo vamos a estudiar las tres primeras.

5.1. TEORÍA DE DEPENDENCIAS

Las Formas Normales se basan en ciertas restricciones definidas sobre los atributos de una relación. Estas restricciones son conocidas como **Dependencias**.

Dependencia Funcional:

Un descriptor Y se dice que depende funcionalmente de otro X si y sólo si a cada valor de X le corresponde un único valor de Y, lo que se expresa de la siguiente forma: **X → Y**

También se dice que X determina ó implica a Y. X se denomina implicante ó determinante e Y es el implicado.

Tanto X como Y pueden estar formados por varios atributos ó ser un atributo único.

Dependencia Funcional Completa:

Un atributo Y tiene dependencia funcional completa respecto de un descriptor X, si depende funcionalmente de él en su totalidad, es decir, no depende de ninguno de los posibles atributos que formen parte de X.

Sea X un descriptor compuesto por x1 y x2. Y tiene dependencia funcional completa de X si se cumple:

$$X \rightarrow Y$$

$$x1 \rightarrow Y$$

$$x2 \rightarrow Y$$

lo que se expresa: $X \Rightarrow Y$

Todo implicante formado por un único atributo supone una dependencia funcional completa.

Dependencia Funcional Transitiva:

Un descriptor depende transitivamente de otro si y sólo si depende de él a través de otro descriptor. Así, Z depende transitivamente de X si:

$$X \rightarrow Y$$

$$Y \rightarrow Z$$

$$X \rightarrow Z$$

Ya que X implica Z, pero a través de Y.

5.2. FORMAS NORMALES

Primera Forma Normal (1FN): Es una restricción inherente al Modelo Relacional, por lo que su cumplimiento es obligatorio. Consiste en la prohibición de que en una relación existan grupos repetitivos, esto es, de que un atributo pueda tomar más de un valor del dominio subyacente.

Segunda Forma Normal (2FN): Una relación está en 2FN si está en 1FN y todos los atributos no primos tienen dependencia funcional completa respecto de la clave, es decir dependen totalmente de la clave y no de parte de ella.

Nota: Toda relación cuya clave está formada por un único atributo está en 2FN.

Tercera Forma Normal (3FN): Una relación está en 3FN si está en 2FN y todos los atributos no primos dependen directamente de la clave, es decir no hay dependencias transitivas de los atributos no primos respecto de la clave.

Nota: Toda relación con un único atributo no primo está en 3FN.

6. DINÁMICA DEL MODELO (ÁLGEBRA RELACIONAL).

Para manejar las tablas se utilizan algunas operaciones clásicas de la Teoría de Conjuntos y otras específicas del Modelo Relacional. Entendemos por Álgebra Relacional el conjunto de operadores que aplicados a las relaciones dan como resultado nuevas relaciones.

Operadores Primitivos: Pueden ser unarios ó binarios.

- Unarios: Tienen como operando una única relación.
 - Selección ó Restricción (σ):** La selección de una relación da como resultado otra relación formada por un subconjunto de tuplas que satisfacen una determinada condición. La relación resultante es un subconjunto horizontal de la primera.
 - Proyección (Π):** La proyección de una relación da como resultado otra relación donde se considera un subconjunto de atributos de la relación original. La relación resultante es un subconjunto vertical de la primera.
- Binarios: Se aplican a dos relaciones.
 - Unión (U):** La unión de dos relaciones da como resultado otra relación formada por las tuplas de las dos eliminando las duplicadas. (Sólo se pueden aplicar entre relaciones compatibles es decir que tengan el mismo número y tipo de atributos)
 - Diferencia (-):** La diferencia de dos relaciones es otra relación formada por las tuplas que pertenecen a la primera pero no a la segunda. (Sólo se pueden aplicar entre relaciones compatibles es decir que tengan el mismo número y tipo de atributos)
 - Producto Cartesiano (\times):** Tiene como atributos la unión de los atributos de las dos relaciones y sus tuplas se forman concatenando cada tupla de la primera relación con cada tupla de la segunda.

Operadores Derivados:

- Join, Combinación ó Concatenación (*):** Es el resultado de aplicar un producto cartesiano seguido de una selección por igualdad y de una proyección.
- Intersección (\cap):** La intersección de dos relaciones es otra relación cuyas tuplas pertenecen a ambas. (Sólo se pueden aplicar entre relaciones compatibles).
- División ($:$):** La división de dos relaciones R y S es otra relación formada por las tuplas de R que están combinadas con todas las de S. Se ha de cumplir que los atributos de S se encuentren también en R. Los atributos de la relación resultado son los atributos de R que no estén en S.