



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

“BASE DE DATOS RELACIONALES”

AUTORÍA RAQUEL ZAMBRANO RAMÍREZ
TEMÁTICA INFORMÁTICA
ETAPA CICLO FORMATIVO GRADO MEDIO

Resumen

Estudio de las bases de datos relacionales. Comenzamos analizando los distintos conceptos que componen el modelo entidad-relación y el modelo relacional para, más tarde, estudiar la transformación del modelo entidad-relación al modelo relacional.

Palabras clave

bases de datos relacionales

modelo entidad-relación

modelo relacional

1. EL MODELO ENTIDAD RELACIÓN

El modelo entidad relación fue creado por Peter Chan en la década de los 70.

Es un modelo de datos gráfico, que trabaja en el nivel conceptual de los SGBD, y que se estructura en torno a dos conceptos: entidades y relaciones.

2. ENTIDADES

Una entidad es algo que existe en la realidad y de la que, además, podemos almacenar información. En un supuesto práctico, las entidades suelen ser los sustantivos de la frase.

Ejemplos de entidades en una base de datos creada para gestionar un instituto pueden ser alumnos o profesores; en una base de datos para llevar la contabilidad de una empresa pueden ser clientes, pedidos, artículos o proveedores.

En el modelo entidad relación, las entidades están representadas por un rectángulo.



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

2.1. Componentes de las entidades

En una entidad podemos distinguir los siguientes componentes:

- **Atributos:** es la información que podemos almacenar de una entidad determinada. Por ejemplo, en la entidad alumnos, posibles atributos podrían ser el nombre, los apellidos, el número de expediente, la dirección, el teléfono, etc. Los atributos se representan, en el modelo entidad relación como elipses.
- **Clave primaria:** es el atributo o conjunto de atributos que identifican de forma única a una entidad. Por ejemplo, en la entidad alumnos, la clave primaria podría ser el número de expediente del alumno. También podríamos tener una clave multiatributo en dicha entidad formada por el nombre y los apellidos de nuestros alumnos. Ésta elección de clave primaria sería peor que la anterior por dos motivos. Primero porque es posible, aunque no frecuente, que existan dos personas con el mismo nombre y los dos apellidos. Y, sobretodo, porque es conveniente elegir claves numéricas. Sólo existe una forma de escribir el número de expediente del alumno. Sin embargo, podemos escribir de muchas formas el nombre y los apellidos de los alumnos, podemos utilizar mayúsculas y minúsculas, podemos dejar más o menos espacios en blanco o incluso podemos escribir hasta abreviaturas. Esta arbitrariedad al escribir los atributos textuales, los hace peores candidatos a la hora de ser elegidos como clave primaria. Normalmente, elegimos como clave primaria atributos numéricos, el DNI, el número de expediente, un código, etc. Para distinguir, en el modelo entidad relación, la clave primaria del resto de atributos, subrayamos su nombre en la representación gráfica.
- **Clave candidata:** Tal y como hemos visto, en una entidad puede existir más de un atributo o más de un conjunto de atributos que podrían ser elegidos como clave primaria. Cada atributo o cada conjunto de atributos que podría ser elegido como clave primaria recibe el nombre de clave candidata. En el ejemplo anterior, tanto el número de expediente como el conjunto formado por los atributos nombre y apellidos serían claves candidatas. Del conjunto de claves candidatas elijo una de ellas que va a ser clave primaria, en este caso el número de expediente del alumno. El resto siguen siendo claves candidatas en mi modelo entidad relación.

2. 2. Tipos de entidades

Existen dos tipos de entidades:

- **Entidades fuertes:** son aquellas que no dependen de ninguna otra para existir. Las entidades que hemos estudiado hasta ahora son entidades fuertes: alumnos, profesores, clientes, etc.
- **Entidades débiles:** son aquellas que necesitan de otras para existir. Por ejemplo, en una base de datos creada para llevar la contabilidad de una empresa tenemos una entidad direcciones en la que almacenamos las direcciones de nuestros clientes. Esta entidad en este sistema sería una entidad débil, ya que depende de la entidad clientes para existir. Si sólo observamos dicha entidad, no sabemos si las direcciones que contiene hacen referencia a clientes, a proveedores o a trabajadores. Es necesario especificar la entidad de la que depende para poder comprender la



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

información que contiene. Sin embargo, podríamos haber almacenado las direcciones de nuestros clientes como un atributo más de nuestra entidad, ¿qué es lo que hace a la información direcciones ser candidata para almacenarlo como entidad débil y no como atributo? Por un lado, que direcciones está compuesto, a su vez, de varias unidades de información: la calle, el número, el bloque, el piso, etc. Y, por otro, y más importante, que queramos que nuestro sistema contemple la posibilidad de que nuestros clientes puedan tener más de una dirección. Las entidades débiles se representan en el modelo entidad relación como un rectángulo con un borde doble.

3. RELACIONES

Decimos que existe una relación entre dos entidades cuando existe una asociación entre ambas. En un supuesto práctico, las relaciones suelen ser el verbo que relaciona dos entidades.

Veamos más claramente, cuando existe una relación entre entidades. Imaginemos que tenemos una base de datos de una empresa en la que tenemos como entidades, clientes, pedidos, artículos y proveedores. La entidad clientes va a estar relacionada con la entidad pedidos, puesto que van a ser los clientes de mi empresa los que realicen los pedidos. A su vez, los pedidos van a estar relacionados con los artículos pues, cada pedido que realiza cada cliente va a estar compuesto por una serie de artículos. También los artículos van a estar relacionados con los proveedores, ya que son los proveedores los que facilitan dichos artículos a mi empresa. Sin embargo, la entidad clientes no va a estar relacionada directamente ni con artículos, ni con proveedores. Va a estar relacionada de forma indirecta a través de la entidad pedidos. Tampoco pedidos está relacionada directamente con proveedores, sino que se relacionan entre sí a través de la entidad artículos.

Las relaciones se representan en el modelo entidad relación como un rombo.

3. 1. Tipos de relaciones

Podemos hacer dos clasificaciones diferentes de las relaciones. Una atendiendo al grado de las relaciones y otra atendiendo a la cardinalidad de la relación.

Se define el grado de la relación como el número de entidades que participan en la relación. Distinguiamos tres tipos de relaciones en función del grado:

- Reflexivas: son relaciones de una entidad consigo misma. Por ejemplo, si tenemos una entidad alumnos en la que almacenamos todos los alumnos de un curso, incluido su delegado, aparece la relación “delegado” de la entidad alumnos consigo misma. Igual sucede en la entidad empleados si almacenamos los datos de todos los empleados de la empresa, incluidos los jefes. En este caso aparecería la relación “jefe” de la entidad empleados consigo misma.
- Binarias: son relaciones entre dos entidades. Por ejemplo, la relación que aparece entre clientes y pedidos o la relación que aparece entre alumnos y profesores. Son las relaciones que nos encontramos con más frecuencia.
- Terciarias: son relaciones entre tres entidades. Este tipo de relaciones aparece, por ejemplo, entre las entidades cuenta, sucursal y cliente para la base de datos de un banco. Cada cliente va a



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 Nº 14 – ENERO DE 2008

disponer de una o varias cuentas en una sucursal determinada, por tanto la relación afecta a las tres entidades y no sólo a dos de ellas.

Se define la cardinalidad de la relación cómo el número de instancias de una entidad que pueden asociarse a otra entidad diferente, mediante una relación. Distinguimos tres tipos de relaciones en función de la cardinalidad:

- 1:1: En este tipo de relaciones, una instancia de una entidad se relaciona con una, y sólo una, instancia de otra entidad. Por ejemplo, si tenemos una base de datos para gestionar un conjunto de empresas y tenemos una entidad en la que almacenamos las empresas y otra en la que almacenamos los directores, la cardinalidad de la relación entre empresas y directores va a ser 1:1 ya que, cada empresa va a tener uno, y sólo un director y, a su vez, cada director va a dirigir una, y sólo una empresa.
- 1:M: En este tipo de relaciones, una instancia de una entidad se relaciona con muchas instancias de otra entidad. Sería el caso de la relación entre clientes y pedidos ya que un cliente va a realizar muchos pedidos a la empresa pero un pedido sólo puede haber sido realizado por uno, y sólo un cliente.
- N:M: En este tipo de relaciones, muchas instancias de una entidad se relacionan con muchas instancias de otra entidad. Es el caso de la relación entre alumnos y profesores. Un alumno tiene muchos profesores que le imparten clase y, a su vez, un profesor, tiene muchos alumnos a los que impartir clase.

La cardinalidad se representa en el modelo entidad relación cómo unos paréntesis que situamos justo al lado de la relación, de la siguiente forma: (1,1), (1, M) o (N:M).

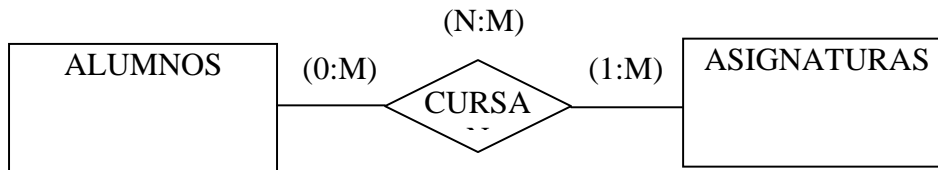
A veces, las relaciones también pueden llevar atributos, al igual que las entidades. Casi siempre, las relaciones que llevan atributos son relaciones del tipo N:M y, dichos atributos, suelen hacer referencia a fechas. A veces, también las relaciones del tipo 1:M llevan atributos, pero es menos frecuente. Imaginemos una base de datos para gestionar una empresa de autobuses. Entre otras entidades, en el sistema nos encontramos una entidad para almacenar los conductores y, otra entidad, para almacenar los autobuses. Un conductor no siempre va a conducir el mismo autobús y, a su vez, un autobús va a ser conducido por varios conductores, luego estaríamos ante una relación del tipo 1:M. Aunque, según hemos dicho, un conductor puede conducir muchos autobuses, para un día concreto, para un viaje en concreto, va a conducir sólo un autobús, el que sea. Es por eso que tenemos que encontrar la manera de determinar qué autobús es conducido por qué conductor, en cada momento. La forma de hacer esto es añadir los atributos fecha y días de viaje a la relación existente entre las entidades autobuses y conductores. Cómo veremos más adelante, al ver el paso del modelo entidad relación al relacional, esta relación de tipo N:M con atributos se va a convertir, al pasarlo al modelo relacional, en una nueva tabla. Y los atributos fecha y días de viaje van a ser atributos de esta nueva tabla.

Además de la cardinalidad de las relaciones es necesario representar, en el modelo entidad relación, la obligatoriedad de las relaciones. En una relación, la obligatoriedad hace referencia a los valores mínimo y máximo en los que aparece una entidad en una relación. Veamos esto con un



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

ejemplo. En la relación entre las entidades alumnos y asignaturas, un alumno tiene que tener como mínimo una asignatura y como máximo, puede tener muchas. Sin embargo una asignatura no tiene porque ser cursada por un alumno. Imaginad, en un centro educativo, que existe una determinada oferta de optativas. Puede ser que algún año, por alguna circunstancia, una asignatura no se imparta porque ningún alumno la haya elegido. Sin embargo, ese no es motivo para eliminar definitivamente la asignatura de nuestra base de datos ya que puede que en próximos cursos, dicha asignatura si que tenga alumnos que la cursen. Así, el valor mínimo para asignaturas va a ser cero y el valor máximo va a ser muchos, ya que una asignatura puede ser cursada por muchos alumnos. Observar que la cardinalidad de la relación la obtenemos con los valores máximos con los que interviene cada entidad en la relación. La representación de esta relación en el modelo entidad relación sería:



4. GENERALIZACIÓN Y JERARQUÍAS DE GENERALIZACIÓN

Mediante la generalización podemos especializar una entidad, que se llamará supertipo, en varias entidades, que se denominarán subtipos.

Nos encontramos con una generalización cuando tenemos atributos comunes para un conjunto de entidades y, a su vez, tenemos atributos específicos para cada una de ellas.

Imaginemos la base de datos de una empresa en la que tenemos una entidad empleados, que va a ser la entidad supertipo, y varias entidades subtipos, que son una especialización de la primera: directivo, en la que almacenaremos los datos de los directivos de la empresa, técnicos, en la que almacenaremos los datos de los técnicos de la empresa, administrativo, en la que vamos a guardar los datos de los administrativos de la empresa, y limpieza, en la que vamos a almacenar los datos del personal de limpieza de nuestra empresa. Para todos ellos vamos a tener los siguientes atributos comunes: número de empleado, nombre, dirección, fecha de nacimiento, salario y puesto. Para los directivos vamos a tener, además, los atributos comisiones y número de empleados a su cargo. Para los técnicos vamos a tener el atributo zona asociada, que especificará la zona en la que trabaja el técnico. Para los administrativos tendremos la oficina en la que trabaja como atributo y para el personal de limpieza tendremos como atributo las horas semanales que trabaja en nuestras oficinas.

En el modelo entidad relación asociaremos los atributos comunes a todas las entidades al supertipo. Una característica importante de la generalización es la herencia, es decir, los subtipos van a heredar todos los atributos del supertipo aunque, en la representación gráfica del modelo entidad relación, no se asocian estos atributos a las entidades subtipo. Es necesario también representar, en el modelo entidad relación, los atributos específicos de cada subtipo.

Cuando el supertipo participa en una relación, también los subtipos participan en la misma.



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

Decimos que una generalización es total si no hay ocurrencias en el supertipo que no pertenezcan a algún subtipo. Por ejemplo, la generalización anterior de directivos, técnicos, administrativos y personal de limpieza en empleados, es una generalización total, puesto que cualquier empleado de la empresa tiene que estar clasificado en una de esas categorías. No existe ninguna otra categoría de empleados en nuestra empresa y, por tanto, en nuestra base de datos. Si existen ocurrencias en el supertipo que no pertenezcan a algún subtipo, decimos que la generalización es parcial. Este sería el caso de nuestra base de datos si, por ejemplo, en la plantilla de la empresa también hubiera personal de seguridad. Como el personal de seguridad no tiene ningún atributo específico, no es necesario que creamos un subtipo con estos empleados. Sin embargo, el personal de seguridad de la empresa si está recogido en la tabla empleados. Así, existen empleados de la empresa que no están incluidos en ninguno de los subtipos en los que se divide el supertipo: directivos, técnicos, administrativo y limpieza pero sí están incluidos en el supertipo.

Decimos que una generalización es exclusiva si, cualquier ocurrencia del supertipo, sólo puede estar incluido en uno de los subtipos. En caso contrario, decimos que la generalización es solapada o superpuesta. Por ejemplo, si, en la base de datos anterior, un empleado sólo puede ser directivo o técnico o administrativo o personal de limpieza, es decir, sólo puede tener a la vez una categoría, la base de datos sería exclusiva. Si, por ejemplo, en nuestra empresa, un directivo puede ser, a la vez, técnico, entonces nuestra base de datos sería solapada.

A veces, es necesario en las generalizaciones añadir un atributo que identifique el tipo de subtipo. Por ejemplo, en el caso de la base de datos con la entidad empleados, el atributo que identificara el subtipo podría ser categoría. Este atributo haría referencia a la categoría del empleado dentro de la empresa.

5. EL MODELO RELACIONAL

El modelo relacional fue descrito en la década de los 70 por E. F. Codd y surge como mejora de los modelos jerárquico y en red.

Está basado en una estructura de datos de dos dimensiones que se conoce con el nombre de tabla o relación. Es importante no confundir el término relación en el modelo relacional que hace referencia a una tabla, con el término relación del modelo entidad relación que hace referencia a la asociación entre ocurrencias de entidades. Por ello, evitaremos en todo momento utilizar el término relación en el modelo relacional e intentaremos utilizar en su lugar el término tabla para evitar posibles confusiones. Observad que en el modelo relacional no hablamos nunca de entidades. Esto es, tal y como estudiaremos con más detalle más adelante, porque no todas las entidades del modelo entidad relación se van a corresponder con una tabla en el modelo relacional.

6. TABLAS

Una tabla, tal y como dijimos antes, es una estructura de datos bidimensional, que está formada por filas y columnas.



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

En una tabla almacenamos, por ejemplo, los datos correspondientes a los alumnos de un centro. En cada fila o tupla, almacenaríamos los datos de un alumno y en cada columna, almacenaríamos los datos del alumno correspondiente: su nombre, sus apellidos, su número de expediente, su dirección, etc.

El grado de una tabla es el número de columnas que posee dicha tabla. La cardinalidad de una tabla corresponde al número de filas que tiene una tabla.

7. CLAVES

En el modelo entidad relación hemos estudiado las claves candidatas y las claves primarias. En el modelo relacional aparece un nuevo tipo de clave, la clave ajena. Además, se siguen manteniendo los conceptos de clave candidata y clave ajena.

La clave ajena aparece en el modelo relacional como consecuencia de la necesidad de repetir información en una tabla. Por ejemplo, imaginemos que creamos una base de datos en el modelo relacional para gestionar la contabilidad de una empresa. Tenemos una tabla en la que almacenamos los datos de los clientes y otra tabla en la que almacenamos los datos de los pedidos que realizan esos clientes. Vimos en el modelo entidad relación que, al ser los clientes los que realizan los pedidos, existía una relación entre las entidades clientes y pedidos. En el modelo relacional ya no existen las entidades y las relaciones, ahora tenemos únicamente tablas. Para mantener en nuestro modelo de datos la asociación existente entre los clientes y los pedidos, lo que hacemos en el modelo relacional es repetir un campo de la tabla clientes en la tabla pedidos. Es decir, a la tabla pedidos le añadimos un campo que va a ser el código del cliente. De forma que, si en cualquier momento, quisieramos conocer los datos personales de un cliente que ha realizado un pedido, simplemente tendríamos que recuperar su código de cliente, y consultar en la tabla clientes cuáles son los datos del cliente con ese código de cliente. Cómo resulta que el código del cliente va a ser la clave primaria en la tabla clientes, decimos que el código del cliente va a ser, además, clave ajena en la tabla pedidos.

En resumen, una clave ajena de una tabla es un conjunto de uno o más atributos que son clave primaria en otra tabla diferente.

8. REPRESENTACIÓN EN EL MODELO RELACIONAL

En el modelo relacional, a la hora de representar una tabla, escribimos su nombre y, a continuación, sus atributos entre paréntesis. La clave primaria va precedida del símbolo #. Las claves ajenas se representan subrayadas y con el nombre de la tabla en la que dicha clave es clave primaria debajo. Veamos el ejemplo de clientes y pedidos:

```
clientes (#cod_cli, nombre, apellidos, direccion)
pedidos (#cod_pedido, cod_cliente, importe, fecha)
          clientes
```



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

9. REGLAS DE INTEGRIDAD

Mantener la integridad en una base de datos significa asegurarnos de que los datos que tenemos almacenados en ella son, en todo momento, correctos y de que no existen repeticiones de datos innecesarias.

Las reglas de integridad pueden venir especificadas por el propio modelo de datos o, bien, pueden ser definidas por el usuario de la base de datos.

9. 1. Reglas de integridad del modelo relacional

El modelo de datos relacional impone dos reglas de integridad, una relacionada con la integridad de las claves primarias y otra relacionada con la integridad de las claves ajenas:

- Cómo hemos visto, la clave primaria sirve para identificar, de forma única, una ocurrencia de una entidad en el modelo entidad relación, o una fila de una tabla en el modelo relacional. Por tanto, las reglas de integridad asociadas a la clave primaria van a ser, por un lado, que no puede haber claves primarias repetidas y, por otro, que no puede haber claves primarias con valores vacíos. Por ejemplo, en la tabla alumnos de una base de datos de un centro educativo dónde la clave primaria va a ser el número de expediente del alumno es necesario que no haya números de expediente repetidos y que, además, ningún número de expediente de ningún alumno esté vacío en nuestra base de datos.
- La regla de integridad de las claves ajenas establece que no puede haber un valor para la clave ajena que no exista previamente en la clave primaria de la tabla de la que procede. Por ejemplo, en la tabla pedidos, en la que aparece un código de cliente que es clave ajena de la tabla clientes, no puede haber ningún valor de código de cliente que no esté previamente registrado cómo tal en la tabla clientes. Hay que tener cuidado con esta regla de integridad a la hora de realizar las operaciones de mantenimiento de la base de datos. Así:
 - Al introducir una nueva fila en una tabla hay que verificar que los valores que introducimos en las claves ajenas ya están dados de alta en la tabla en la que son claves primarias. En el ejemplo anterior, hay que verificar a la hora de insertar un pedido en nuestra base de datos que el cliente que realiza ese pedido, ya está registrado en nuestro sistema cómo cliente.
 - Al borrar una fila de una tabla o al modificar los valores de su clave primaria, hay que verificar si existen claves ajenas derivadas de la clave primaria de dicha tabla. Por ejemplo, al borrar un cliente de una base de datos, o al cambiar su código de cliente, hay que verificar que ese cliente no tenga realizado ningún pedido o, si lo tiene, realizar las modificaciones oportunas para que no se generen datos erróneos en nuestra base de datos. Los SGBD incorporan mecanismos automáticos de controlar esta regla de integridad. Así, al borrar o modificar una clave primaria, en caso de que existan claves ajenas, el SGBD puede actuar de dos maneras diferentes:



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

- No permitir el borrado o la actualización de aquellas filas de una tabla que estén referenciadas en otra tabla cómo clave ajena. O sea, no permitir el borrado o la modificación de aquellos clientes que ya tienen realizados pedidos.
- Borrar o modificar en cascada todas las filas de todas las tablas que estén referenciadas por las filas borradas o modificadas en la tabla original. Es decir, borrar o modificar el código de cliente en la tabla pedidos de aquellos clientes borrados o modificados en la tabla clientes.

9. 2. Reglas de integridad definidas por los usuarios

Además de las reglas de integridad impuestas por el propio modelo de datos, es posible establecer reglas de integridad definidas por los usuarios. Se trata, normalmente, de condiciones no informáticas relacionadas con el uso que se va a hacer de la información.

Un ejemplo de regla de integridad definida por el usuario puede ser que no podamos matricular alumnos en un centro educativo de secundaria, menores de 12 años. En principio, para el modelo de datos sería indiferente que en nuestro sistema tuvieramos alumnos matriculados menores de 12 años, pero no para los encargados de gestionar el centro educativo, para los que sería inaceptable que alumnos menores de esa edad estuvieran cursando estudios en el centro.

Ni el modelo entidad relación ni el modelo relacional incorporan mecanismos para registrar las restricciones de los usuarios en sus representaciones gráficas. Sin embargo, casi todos los SGBD ofrecen la posibilidad de definir estas restricciones a la vez que vamos diseñando la base de datos. Por ello, sería conveniente registrar estas restricciones en las representaciones gráficas tanto del modelo relacional, cómo del modelo entidad relación. Para registrarlas podemos, simplemente, hacer una anotación al margen en la representación o, incluso, crear una ficha en la que expliquemos todos los campos o atributos de las tablas o entidades y sus correspondientes restricciones de usuario.

10. TRANSFORMACIÓN DEL MODELO ENTIDAD RELACIÓN AL MODELO RELACIONAL

Para realizar la transformación del modelo entidad relación al modelo relación procederemos de la siguiente manera:

- Las entidades se transforman directamente en tablas. Los atributos se transforman en columnas de las tablas. La clave primaria del modelo entidad relación pasa a ser clave primaria en el modelo relacional.
- Las relaciones en las que dos o más entidades tienen cardinalidad M, por ejemplo, relaciones ternarias 1:N:M o relaciones binarias N:M, se transforman en una nueva tabla, que recibirá el nombre de la relación y que tendrá cómo clave primaria el conjunto de todos los atributos que forman las claves primarias de las entidades asociadas a través de la relación. Por ejemplo, imaginemos las entidades profesores y asignaturas en el modelo entidad relación, asociadas mediante la relación imparte. La clave primaria de profesores va a ser el código de profesor y la



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

clave primaria de asignaturas va a ser el código de asignatura. En la tabla imparte, procedente de la relación imparte en el modelo entidad relación, la clave primaria va a ser el conjunto de los atributos código de profesor y código de asignatura. Además, habría que añadir, en caso de que los hubiera, los atributos de la relación como columnas de la tabla.

- Para transformar las relaciones en las que sólo una entidad tiene cardinalidad M, por ejemplo, las relaciones ternarias 1:1:M o las binarias 1:M, propagamos la clave primaria de la entidad o las entidades que participan con cardinalidad 1 como clave ajena de la entidad que participa con cardinalidad M. Por ejemplo, imaginemos las entidades grupo y alumnos de una base de datos de un centro escolar, cuya relación, pertenece, es el tipo 1:M. Al pasar al modelo relacional, únicamente crearemos dos tablas, la tabla grupo que procede de la entidad grupo y la tabla alumnos que procede de la entidad alumnos. Para reflejar en el modelo relacional la relación pertenece del modelo entidad relación, lo que hacemos es propagar la clave primaria de la entidad que participa con cardinalidad 1 como clave ajena en la entidad que participa con cardinalidad M. Es decir, a la tabla alumnos, le añadimos un nuevo campo, código grupo, que va a ser clave ajena dentro de la tabla alumnos y que va a contener, necesariamente, los valores del campo código de grupo, que es clave primaria en la tabla grupos.
- En algunos casos concretos, es posible transformar la relación 1:M del modelo entidad relación a una nueva tabla en el modelo relacional. Sólo procederemos de esta forma en aquellos casos en los que la relación tenga atributos o cuando las entidades participan opcionalmente en una relación. Es decir, cuando los valores mínimos de ocurrencia de ambas entidades en una relación son cero. En caso que tuvieramos que crear una nueva tabla en el modelo relacional, derivada de una relación 1:M en el modelo entidad relación, la clave primaria de esta nueva tabla sería la clave primaria de la entidad que participa en la relación con cardinalidad M.
- Las relaciones entre entidades fuerte y débil van a ser, generalmente, del tipo 1:M. Así, al pasar al modelo relacional una relación fuerte-débil del modelo entidad relación, procedemos exactamente como en el caso de una relación 1:M con una pequeña diferencia. Ahora, la clave primaria de la entidad que participa en la relación con cardinalidad 1, se propaga a la entidad que participa en la relación con cardinalidad M y, además, pasa a formar parte de su clave primaria. En el caso de las relaciones 1:M normales, la clave propagada únicamente pasaba a ser clave ajena en la entidad que participa con cardinalidad M. Ahora, además, pasa a formar parte de la clave primaria.
- Para transformar convenientemente las relaciones 1:1 del modelo entidad relación al modelo relacional es necesario tener en cuenta lo siguiente:
 - transformaremos dicha relación en una tabla si la obligatoriedad de las dos entidades que participan en la relación son (0,1).
 - en el caso de que una entidad posea obligatoriedad (0,1) y la otra (1,1), propagaremos la clave primaria de la entidad que participa con obligatoriedad (1,1) como clave ajena a la entidad que participa con obligatoriedad (0,1).
 - en el caso de que ambas entidades posean obligatoriedad (1,1), propagaríamos la clave primaria de cualquiera de ellas como clave ajena de la otra.



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

- Para transformar las relaciones reflexivas del modelo entidad relación al modelo relacionar, debemos fijarnos, al igual que hacíamos en el caso de las relaciones binarias y ternarias, en la cardinalidad y en la obligatoriedad. Así:
 - Si la relación es 1:1, tendremos una única tabla resultante pero con el atributo que compone la clave primaria en el modelo entidad relación repetido en dos atributos, uno que va a ser clave primaria y otro que va a ser clave ajena del primero. Por ejemplo, imaginar una entidad hombres en la que estén almacenados padres e hijos de una ciudad. Esta entidad va a tener una relación, que vamos a denominar padre, reflexiva consigo misma. La clave primaria de esta entidad va a ser el DNI. Al pasar al modelo relacional, añadimos a la tabla resultante, un nuevo campo DNI_padre, que va a contener el DNI del padre de cada hombre que aparezca en dicha tabla. Este nuevo campo va a ser clave ajena de la clave primaria DNI de esta misma tabla.
 - Si la relación es 1:M distinguiremos dos casos:
 - Si la entidad que participa con cardinalidad muchos es siempre obligatoria, se procede cómo en el caso de una relación reflexiva 1:1.
 - Si la entidad que participa con cardinalidad muchos no es siempre obligatoria, se crea una nueva tabla cuya clave primaria será la clave de la entidad que participa con cardinalidad muchos. Además, y al igual que hacíamos en el caso de las relaciones reflexivas 1:1, se repite este campo en la nueva tabla, y va a ser, a su vez, clave ajena. Por ejemplo, en la entidad empleados que tiene una relación reflexiva (1,M), que denominaremos jefe, y cuya clave primaria es el código de empleado, aparecería una nueva tabla, llamada jefe, que tendría dos campos, por un lado, el código de empleado, que va a ser clave primaria de la nueva tabla y, a la vez, clave ajena de la tabla empleados. Por otro lado, aparecería un nuevo campo, que podríamos denominar código jefe, que va a contener el código de empleado del jefe de cada uno de los empleados. Por supuesto, este código de empleado va a ser también clave ajena en la tabla empleados, ya que los jefes siguen siendo empleados de la empresa.
 - Si la relación reflexiva es del tipo N:M procedemos igual que en las relaciones binarias. La tabla resultante de la relación contendrá una clave primaria formada por la clave primaria de la entidad de la que procede repetida dos veces. Además, habrá que añadir a la tabla los posibles atributos de la relación reflexiva.
- En el modelo relacional no es posible representar las generalizaciones de entidades. Así, debemos eliminar este tipo de relaciones al pasar al modelo relacional. Para eliminarlas es necesario proceder de una de las siguientes formas:
 - Integrar todas las entidades en una única entidad eliminando los subtipos. Esta nueva entidad resultante contendrá cómo atributos todos los atributos del supertipo, todos los atributos de los subtipos y un atributo discriminativo que me va a permitir identificar a qué subtipo pertenece cada atributo. Esta nueva entidad recibe todas las relaciones que, en la representación anterior, tenían el supertipo o cualquiera de los subtipos. Esta forma de



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 14 – ENERO DE 2008

eliminar las jerarquías es muy simple pero tiene el inconveniente de que se generan muchos valores vacíos en la tabla ya que cualquier subtipo va a tener vacíos los atributos que, en el modelo original, eran específicos de un subtipo diferente.

- Eliminar el supertipo pasando sus relaciones, su clave primaria y los atributos comunes a cada uno de los subtipos. Este método sólo se puede aplicar a las generalizaciones totales y exclusivas y, además, presenta los siguientes inconvenientes:
 - se introduce redundancia en la información ya que los valores de los atributos que, en el modelo anterior, pertenecían al supertipo van a aparecer repetidos en todos los subtipos.
 - aumenta considerablemente el número de relaciones en el modelo entidad relación ya que las relaciones del supertipo se repiten ahora en cada uno de los subtipos.
- Insertar una relación del tipo 1:1 entre el supertipo y cada uno de los subtipos. El atributo que especifica la generalización, es decir, el atributo que me permite identificar si una ocurrencia del supertipo pertenece a uno u otro subtipo, pasa a ser atributo del supertipo. Para pasar de este modelo resultante al modelo relacional procedemos cómo en el caso de una relación binaria 1:1 que ya hemos visto.

11. BIBLIOGRAFÍA

- Ramos A. y Ramos M. J. (2007). *Operaciones con bases de datos ofimáticas y corporativas*. Madrid: Paraninfo.
- Oltra F., Albert J. y Vericat A. (2006). *Operaciones con bases de datos ofimáticas y corporativas*. Madrid: Mc Graw Hill.

Autoría

Nombre y Apellidos: Raquel Zambrano Ramírez

Centro, localidad, provincia: IES Florencio Pintado, Peñarroya-Pueblonuevo, Córdoba

E-mail: raquelzambrano1509@hotmail.com