

Metodología para el aprendizaje del diseño de bases de datos relacionales

Autores:

M. Sc. Amaury Pérez Torres

amauryp@hlg.rimed.cu

M. Sc. José Soler Silva

jsoler@hlg.rimed.cu

Resumen

En el presente trabajo se realiza un análisis crítico de los enfoques metodológicos de distintos autores acerca del diseño de bases de datos relacionales, considerando sus ventajas y desventajas para el tratamiento docente de estos contenidos. Por último, los autores de este artículo exponen sus experiencias pedagógicas en este sentido y proponen el enfoque general adecuado para el tratamiento didáctico del tema.

Palabras claves: Diseño de bases de datos relacionales, normalización, formas normales.

Summary

Presently work is carried out a critical analysis of the methodological focuses of different authors about the design of databases you relate them considering its advantages and disadvantages for the educational treatment of these contents. Lastly the authors of this article expose their pedagogic experiences in this sense and they propose the appropriate general focus for the didactic treatment of the topic.

Key words: Design of databases relational, normalization, normal forms.

El trabajo con bases de datos es, sin dudas, uno de los temas más importantes en el ámbito de la informática. A diario se encuentran sistemas que manipulan rápidamente gran cantidad de información: búsquedas en una biblioteca, información sobre los pacientes en un hospital, trabajo con cuentas bancarias, páginas WEB dinámicas. Un dato que da la magnitud de la importancia de este tema es que más del 80% de las aplicaciones informáticas acceden a datos.

En muchas ocasiones, las personas con pocos conocimientos informáticos y, aún los que poseen cierta preparación básica, tienden a asociar el desarrollo de bases de datos relacionales (BDR) directamente con la manipulación de un Sistema de Gestión de Bases de

Datos Relacional (SGBDR), por ejemplo Access, soslayando la etapa más importante: el DISEÑO DE LA BASE DE DATOS.

Si en la creación de páginas web dinámicas, las tablas no están correctamente diseñadas, pueden causar problemas cuando se tengan que realizar otras complicadas, llamadas SQL en el código de la aplicación para extraer los datos que se necesitan. La mayoría de esas situaciones indeseadas se evitan realizando el proceso de normalización de las tablas y estableciendo correctamente las relaciones entre ellas.

Por lo antes expuesto, se puede considerar el diseño de BDR como el proceso encaminado a obtener relaciones (tablas) normalizadas.

La normalización es una técnica que se utiliza para crear relaciones lógicas apropiadas entre tablas de una BDR; ayuda a prevenir errores lógicos en la manipulación de datos y facilita agregar nuevas columnas sin romper el esquema actual ni las relaciones.

El proceso de normalización consiste, básicamente, en ir estableciendo determinadas reglas, las cuales están encaminadas a eliminar redundancias e inconsistencias de dependencia en el diseño de las tablas, lo que ayuda a crear una BDR funcional y eficiente.

Existen varios niveles de normalización llamados formas normales (FN), sin embargo, las tres primeras proporcionan suficiente rigor, para cumplir con las necesidades de gestión de la mayoría de las BDR. En ciertas ocasiones normalizar demasiado puede conducir a un esquema muy complejo para trabajar.

- ♦ *Metodología de la normalización progresiva.*

Un primer enfoque metodológico para el diseño de BDR está encaminado a establecer, inicialmente, una relación con los datos que son de interés y luego ir desprendiendo otras relaciones a través del proceso de normalización. Un ejemplo de esta vertiente es el problema que expone Gigi (2004), el cual pretende crear una base de datos para almacenar la información de usuarios, y los datos a guardar son: nombre, empresa, dirección de la empresa y algún e-mail, o bien URL si las tienen.

Al principio se hace lo que el autor llama *Formulación Cero*, creando una tabla (tabla 1) con los datos de interés, aun sin tener en cuenta las reglas de normalización.

Tabla 1. Estructura de datos inicial (sin normalizar).

Usuarios				
Nombre	Empresa	Dirección_empresa	Url1	Url2
Joe	ABC	1 Work Lane	abc.com	xyz.com
Jill	XYZ	2 Job Street	abc.com	xyz.com

A partir de este momento se deben analizar los principales inconvenientes de la relación no normalizada.

El primer riesgo en esa estructura consiste en que si hay que añadir una tercera URL, es necesario adicionar otra columna y por ende reprogramar el código de la aplicación, lo cual está en contraposición con el objetivo de crear un sistema funcional que pueda crecer y adaptarse fácilmente a los nuevos requisitos. Este es el punto adecuado para explicar lo establecido en la primera forma normal y cómo esta viene a resolver el problema que presenta la primera relación establecida.

La primera forma normal (1FN) establece los siguientes requisitos:

- En una tabla no deben existir grupos de columnas repetidos.
- Crear una tabla separada por cada grupo de datos relacionados.
- Los valores en las columnas deben ser atómicos.
- Identificar cada grupo de datos relacionados con una clave primaria, la cual no puede repetirse. En la tabla 1 se observa que esta regla no se cumple, debido a que pueden existir dos usuarios con el mismo nombre.

La tabla 2 es la resultante después de aplicar las reglas de la 1FN y se observa que han quedado resueltos los problemas anteriormente mencionados: se pueden añadir nuevas direcciones URL sin necesidad de adicionar una nueva columna y los usuarios son identificados por una clave única.

Tabla 2. Relación en 1FN.

Usuarios				
UserId	Nombre	Empresa	Dirección_empresa	Url
1	Joe	ABC	1 Work Lane	abc.com
1	Joe	ABC	1 Work Lane	xyz.com
2	Jill	XYZ	1 Job Street	abc.com
2	Jill	XYZ	1 Job Street	xyz.com

En la tabla 2 aún subsisten varios problemas

Creación: La información sobre una nueva empresa no se puede insertar si no hay un usuario en esa empresa.

Supresión: Eliminar una línea de usuario que sea la única que esté en una empresa implica perder la información de la empresa.

Modificación: Cada línea de dirección URL en la que esté implicada determinada empresa repite la información sobre el usuario y la empresa. Si cambia algún atributo de esta, entonces es necesario hacer muchas actualizaciones. Cada vez que se introduce un nuevo registro en la tabla *usuarios*, es necesario duplicar el nombre de la empresa y del usuario; la base de datos crecerá muchísimo.

En este punto quedan demostrados todos los problemas que puede traer una base de datos, si no está en la segunda forma normal (2FN) y se pasa a tratar los requisitos para cumplir con ella.

Una relación R se dice que está en 2FN si está en 1FN y si, y sólo si los atributos no llaves (ni primarias, ni candidatas) de R, si los hubiese, son funcionales y completamente dependientes de la llave primaria de R.

Para obtener la segunda forma normal se hace creando una relación para todos los atributos que dependen funcional y completamente de la llave (y los atributos que no se analizan por ser atributos llaves, pertenecientes a llaves candidatas); y una relación para los atributos que dependan de parte de la llave.

Como se puede observar, se hace necesario darle tratamiento al siguiente sistema de conceptos: Atributo Llave, Atributo No Llave, Llave , Llave Primaria, Llave Candidata, Llave Exterior o Extranjera, Dependencia Funcional Completa, Dependencia Funcional, además del procedimiento para hallar la Llave Primaria.

Las relaciones resultantes después de aplicar las reglas para la 2FN corresponden a las tablas 3 y 4.

Tabla 3. Tabla usuarios en 2FN.

Usuarios			
UserId	Nombre	Empresa	dirección_empresa
1	Joe	ABC	1 Work Lane
2	Jill	XYZ	1 Job Street

Tabla 4. tabla para los datos de direcciones url.

Urls		
Urllid	RelUserId	Url
1	1	Abc.com
2	1	xyz.com
3	2	abc.com
4	2	xyz.com

Después de esto es necesario volver a insistir en que han quedado resueltos los problemas planteados en la 1FN:

Creación: Se puede insertar la información sobre una empresa.

Supresión: Se puede eliminar una línea de usuario y no se pierde la información sobre la empresa, aunque sea el único usuario de esa empresa.

Modificación: Si cambia un atributo de la empresa, sólo hay que cambiarlo en un lugar. Se elimina redundancia.

El proceso continúa hasta obtener todas las relaciones hasta la forma normal deseada. No es objetivo de este trabajo describir todo el procedimiento, sino el de tratar su esencia.

Consideraciones sobre este enfoque:

- El profesor demuestra con relativa facilidad los principales problemas que pueden ocurrir cuando una relación no está normalizada, y por ende, demostrar la importancia de esta.
- Se pueden ir tratando progresivamente las definiciones de los niveles de normalización y otros conceptos relacionados con esta, a medida que se va resolviendo el problema.
- No es un proceso sencillo, principalmente cuando existen múltiples dependencias funcionales.
- Los autores no lo consideran adecuado para resolver nuevos problemas de diseño de bases de datos; sin embargo, desde el punto de vista práctico puede ser de gran utilidad para el mejoramiento de la estructura de bases de datos que tengan problemas de normalización.
- ♦ *Metodología orientada a la construcción del diagrama entidad – relación*

Varios autores (Berzal, 2004; Estraviz, 2004; Aldarías, 2004) enfocan el diseño de bases de datos relacionales a través del modelo entidad – relación. Este es definido de la siguiente

manera: *“Técnica de análisis basada en la identificación de las entidades y de las relaciones que se dan entre ellas en la parte de la realidad que pretendemos modelar.”* (Berzal, 2004)

Este enfoque consiste, básicamente, en la construcción del diagrama entidad – relación, el cual es una representación gráfica de las entidades, con sus atributos y relaciones, que intervienen en la base de datos. Luego se aplican determinadas reglas, las cuales conducen a relaciones normalizadas.

Se utiliza parte de un ejemplo desarrollado por Berzal (2004), el cual consiste en el diseño conceptual de la estructura de una base de datos de una facultad.

La primera cuestión esencial es el esclarecimiento del problema. Para ello es necesario establecer los siguientes requerimientos (restricciones semánticas):

- Cada profesor pertenece a un solo departamento.
- Todo profesor pertenece a algún departamento.
- Todo departamento debe tener un director, que es un profesor.
- Un profesor puede impartir a varios grupos la misma o diferentes asignaturas.
- Un grupo de una asignatura ha de estar impartido por, al menos, un profesor.
- Las asignaturas se imparten en clases, en días, horas y aulas determinadas.
- Los alumnos se matriculan en varias asignaturas (al menos una).
- Una asignatura puede tener varios alumnos matriculados, en varios grupos.
- Por cada aula rotan los distintos grupos.

Luego se pasa a la identificación de entidades, atributos y relaciones:

Entidades	Asignatura	Alumno	Profesor	Departamento	Aula	Grupo
Atributos	<u>ID</u>	<u>DNI</u>	<u>NRP</u>	<u>ID</u>	<u>ID</u>	<u>ID</u>
	Nombre	Nombre	Nombre	Nombre	Capacidad	Tipo
	Créditos	Dirección	Categoría
	Carácter	Beca	Área			
	Curso			

Identificación de las relaciones:

Relación	Entidades participantes	Tipo	Atributos
Se matricula en	Alumno - Grupo	M:1	Calificación
Enseña	Profesor – Grupo	M: M	
Impartida en	Asignatura - Grupo	1: M	
Asignada a	Aula – Grupo	M: M	Día, hora
Pertenece a	Profesor – Departamento	M: 1	
Dirige	Profesor – Departamento	1: 1	

La figura 1 corresponde al diagrama entidad – relación de la base de datos analizada.

En el proceso de construcción del diagrama entidad–relación (DER) se pueden analizar las siguientes definiciones fundamentales:

Entidad: Objeto, real o abstracto, distinguible de otros objetos. Al grupo de entidades con cualidades similares, acerca de los cuales se almacena información, se le denomina TIPO (o, simplemente, conjunto de entidades). En el diagrama, las entidades se representan dentro de un rectángulo. Ejemplo: Profesor.

Atributo: Propiedad asociada a un conjunto de entidades (esto es, mediante los atributos se representan propiedades de los objetos). Para cada atributo hay un conjunto de valores permitidos, llamado DOMINIO. Ejemplo: categoría.

Clave: Conjunto de atributos que permite identificar unívocamente a una entidad dentro de un conjunto de entidades. Ejemplo: de un estudiante la combinación del grupo y su número.

Relación: Conexión semántica entre dos conjuntos de entidades. .Ejemplo: la relación entre los profesores y las asignaturas que imparten.

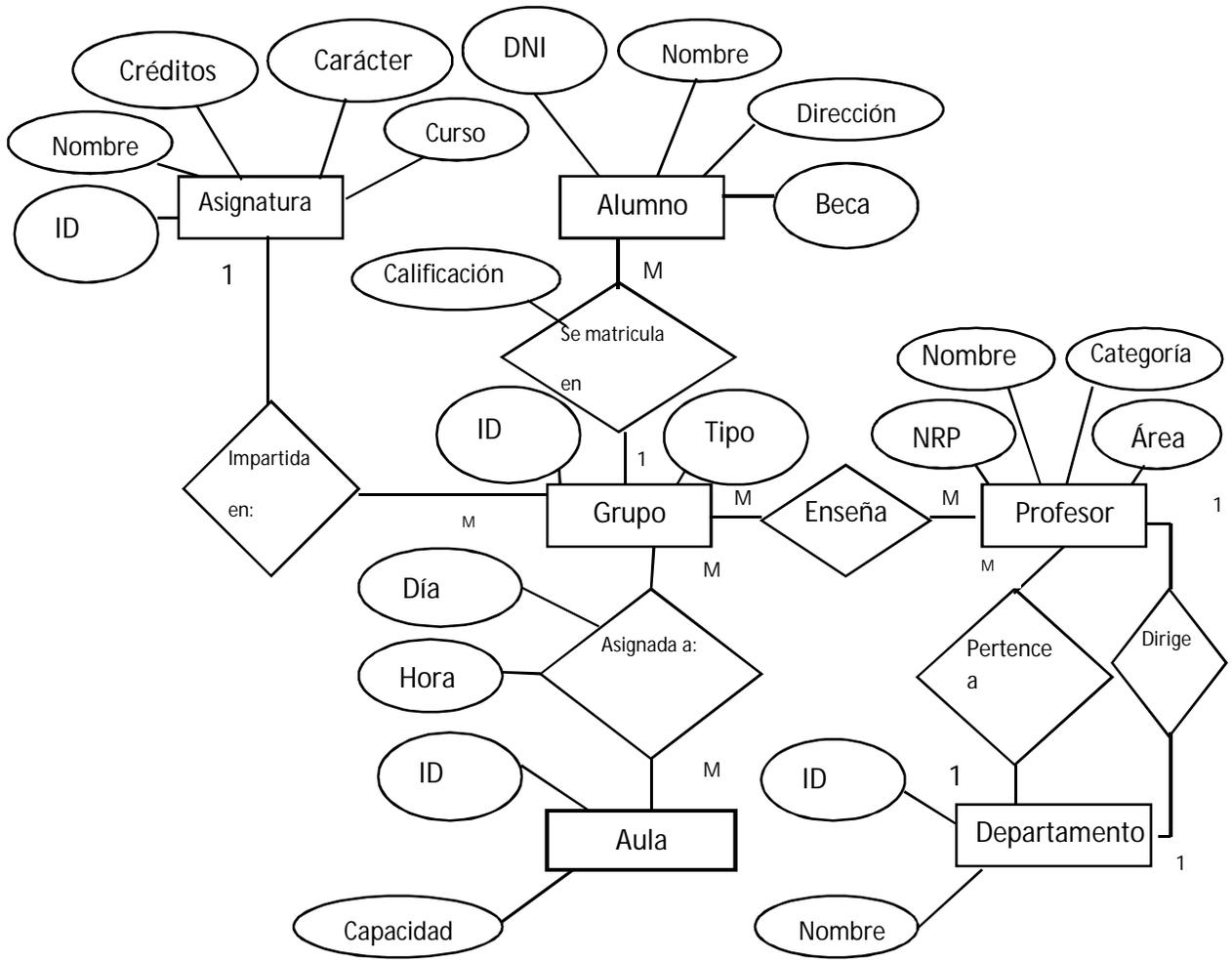


Fig. 1. Diagrama entidad – relación para la base de datos analizada.

Del modelo establecido para la BDR se extraen las relaciones normalizadas hasta la 3FN aplicando determinadas reglas.

- Regla 1: Cada entidad da como resultado una tabla.
Si se aplica esta regla al ejemplo estudiado se obtendrán seis tablas: Asignatura, Alumno, Grupo, Aula, Profesor, Departamento.
- Regla 2: Las interrelaciones *muchos a muchos* representan siempre una nueva tabla.
En este caso hay que agregar dos tablas más: profesor – grupo, aula – grupo.
- Regla 3: Los atributos multievaluados deben descomponerse a la hora de ser llevados a la tabla; por ejemplo, la dirección de un proveedor debe dividirse en varios campos: dirección, ciudad.

- Regla.4: En las interrelaciones *uno a muchos*, la parte *muchos* toma como llave extranjera o externa, la clave principal de la parte 1.

Atendiendo a esta regla en la tabla *Profesor* debe existir un campo con el ID del departamento al cual pertenece, la tabla *Grupo* debe contener un campo con el ID del aula en la cual reciben clases, etc.

Consideraciones: El enfoque orientado al modelo entidad – relación:

- Es sencillo y fácil de aprender.
- Permite obtener relaciones normalizadas con relativa rapidez.
- No se puede demostrar con facilidad los problemas que son inherentes a las relaciones no normalizadas.
- Los distintos niveles de normalización están implícitos en la solución del problema, esto hace que el profesor no los trabaje y tenga que dar un rodeo para poder explicarlos.

Al considerarse las ventajas e inconvenientes de ambos enfoques metodológicos y las experiencias de los autores en la enseñanza de este tema, se propone lo siguiente:

Desde el punto de vista didáctico se recomienda la utilización de ambas metodologías. Se explicarán a través de la solución de un problema práctico. La metodología de la normalización progresiva se recomienda trabajar primero, porque permite tratar explícitamente la importancia de la normalización de bases de datos. Luego se aplica la metodología orientada al modelo entidad – relación a través de un ejemplo. En ambos casos se irán tratando los sistemas de conceptos en la medida que sean necesarios. En las clases prácticas, así como en el estudio independiente, se orientará la solución de problemas aplicando ambos enfoques, aunque se hará énfasis en el segundo. Es provechoso que los estudiantes realicen una valoración de ambos procedimientos y lleguen a las conclusiones adecuadas. Deben ser generalizaciones teóricas, sobre la base de la solución del problema tratado, y la connotación de la solución propuesta.

BIBLIOGRAFÍA

ALDARÍAS, PACO. Diseño de Bases de Datos (DBD) [visitado 5 Abril de 2004; 1:00 p.m].

Disponible desde <http://inicia.es/de/pacodbdb/tema1.ppt>.

BERZAL, FERNANDO. Modelado de datos [visitado 5 Abril de 2004; 2:30 p.m]. Disponible

desde talento.ugr.es/nicmhome/docs/materialbdII/md.pdf.

ESTRAVIZ, ALEJANDRO. El modelado de las bases de datos [visitado 5 Abril de 2004; 10:30 a. m]._Disponible desde <http://www.elrinconcito.com/delphi/articulos/articulos.htm>

-----. El modelado de las bases de datos [visitado 13 Abril de 2004; 2:00 p.m]. Disponible desde <http://www.elrinconcito.com/delphi/articulos/ModeladoBD/Abajo.htm>

GIGI, GABRIEL. Normalización de Bases de Datos y Técnicas de diseño [visitado 5 Abril de 2004; 2:00 p.m].

Disponible desde [http:// www.trucostecnicos.com/trucos/ver.php?id_art=27](http://www.trucostecnicos.com/trucos/ver.php?id_art=27)