

TEMA 40

DISEÑO DE BASES DE DATOS RELACIONALES

ÍNDICE

1. INTRODUCCIÓN
2. FUNDAMENTOS DEL MÉTODO
3. MODELO DE DATOS
 - 3.1. Modelo Conceptual
 - 3.2. Estructura del modelo de datos
 - 3.3. Representación del modelo conceptual
4. MÉTODO DE DISEÑO
5. FASE DE SÍNTESIS: MODELO ENTIDAD/RELACIÓN
6. PASO DEL MODELO E/R AL ESQUEMA RELACIONAL
7. FASE DE ANÁLISIS
8. BIBLIOGRAFÍA

1. INTRODUCCIÓN

Los modelos de datos son instrumentos (objetos y reglas) que nos ayudan a representar la realidad, es decir, nuestro universo del discurso (UD). El proceso de diseño de una Base de Datos consiste en representar un determinado UD mediante los objetos que proporciona el modelo de datos que estamos utilizando, aplicando para ello las reglas de dicho modelo, como prohibición de un determinado tipo de asociaciones o posibilidad de incluir ciertas restricciones. Cuando se diseña una Base de Datos mediante el modelo relacional, al igual que ocurre con otros modelos de datos, tenemos distintas alternativas, es decir, podemos obtener diferentes esquemas relacionales, y no todos ellos son equivalentes, ya que unos van a representar la realidad mejor que otros.

El diseño es un proceso de representación, en el cual los sistemas del mundo real se simulan mediante objetos del ordenador. El proceso de diseño es creativo, y requiere una gran cantidad de imaginación y experiencia. La habilidad para el diseño de Bases de Datos sólo se adquiere con un largo entrenamiento y una práctica, y después de conseguir la capacidad de aprovechar experiencias anteriores, para sintetizar resultados en situaciones nuevas.

2. FUNDAMENTOS DEL MÉTODO

A la hora de determinar una Base de Datos debemos establecer un proceso partiendo del acotamiento de una parcela del mundo exterior, aquella que nos interesa representar en los datos. En este proceso se debe aprender, comprender y conceptualizar dicho mundo exterior transformándolo en un conjunto de ideas y definiciones que supongan una imagen fiel del comportamiento del mundo real. A esta imagen del mundo exterior la llamaremos **Modelo Conceptual**.

Una vez definido el modelo conceptual, éste se ha de transformar en una descripción de datos, atributos y relaciones, que denominaremos **Esquema Conceptual de Datos**. Por último, este esquema conceptual habrá que traducirlo a estructuras almacenadas en soportes físicos.

Resumiendo, mediante un proceso de abstracción, pasaremos del mundo real al mundo de las ideas estableciendo un modelo conceptual, y a partir de éste, a través de un proceso de organización, pasaremos del mundo de las ideas al de los datos, estableciendo así un modelo convencional de datos.

El proceso puede hacerse pasando directamente del mundo real al mundo de los datos. Ello depende de la complejidad del problema y de la capacidad y experiencia del sujeto que lo realiza.

Como primer paso en la construcción de nuestro sistema de información, es necesario delimitar la parcela del mundo real que se somete a estudio.

Identificaremos los objetos que la componen, así como el conjunto de propiedades de cada uno de ellos que de alguna forma resulten relevantes dentro del contexto en el que situamos el problema. Además, como es lógico, existirán una serie de relaciones entre los objetos identificados que resultarán de interés en la confección del sistema.

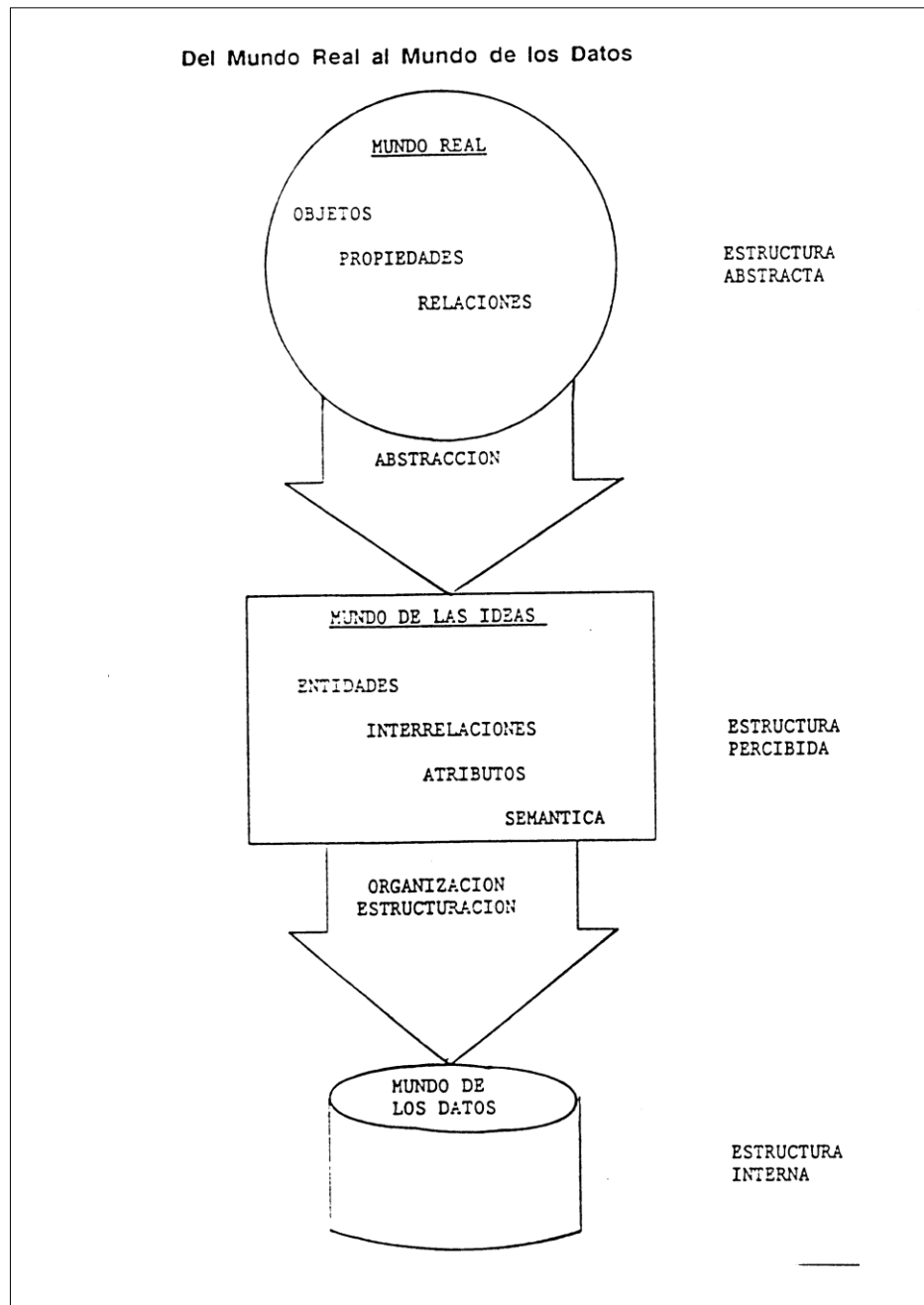
De esta forma, obtendremos, a partir del mundo real, lo que conocemos como el mundo de las ideas o estructura percibida.

Dicho mundo de las ideas, ha de tener una representación final en la máquina que sirve como soporte de nuestro sistema, y que sea posible su implementación.

Para ello, la estructura percibida se organiza y representa de forma que sea almacenada, con el máximo rendimiento posible, en los soportes de máquina disponibles.

El analista es el responsable de la concepción y el diseño del sistema, siendo el programador quien se encarga de la instrumentación del mismo.

Es fundamental, en todo diseño, realizar una estructuración de los datos, lo más óptima y cercana a la realidad posible, con el fin de obtener los mayores índices de rendimiento, flexibilidad y fiabilidad del sistema.



3. MODELO DE DATOS

Es un grupo de herramientas conceptuales para describir los datos, sus relaciones, su semántica y sus limitaciones; de tal forma, que facilita la interpretación de nuestro mundo real y su representación en forma de datos, en nuestro sistema informático. O de otra forma: Un modelo de datos es aquel conjunto de reglas y convenciones, que utilizamos para poder definir la información de aquella parte del mundo real que analizamos.

3.1. Modelo conceptual

Un esquema conceptual comprende la descripción única de los distintos contenidos de información que pueden coexistir en una Base de Datos.

Los **objetivos** del modelo conceptual son los siguientes:

Captar información del mundo real mediante una descripción formal.
 Aislar la representación de la información de las exigencias de cada usuario y de las características de la máquina.
 Independizar la definición de la información de los SGBD que en particular van a manejarla.
 Se entenderá en todo momento que la información se compone de los datos más la interpretación de los mismos, con el fin de tener en cuenta la semántica del mundo estudiado.

3.2. Estructura del modelo de datos

Las propiedades del mundo real pueden ser:

- a) *Estáticas*. Propiedades invariantes en el tiempo. En el modelo de datos son las *Estructuras* y las *Restricciones*.
- b) *Dinámicas*. Propiedades que varían con el tiempo. En el modelo de datos son las *operaciones*, y se definen mediante el *lenguaje de manipulación de datos*.

Estructuras y Restricciones (Estática)

Una **estructura** queda definida por los objetos del modelo y las **restricciones** inherentes, conformando un conjunto de reglas de definición de dichas estructuras.

Los objetos y las restricciones de la estructura dependen de cada modelo, pero en general son:

Entidades, atributos, relaciones, dominios, representación y restricciones inherentes (éstas vienen impuestas por la propia naturaleza del modelo introduciendo rigideces en la modelización).

Las restricciones opcionales o de usuario, restricciones propiamente dichas de la Empresa, son definidas por el usuario, pero el modelo de datos las reconoce y suministra herramientas para manejarlas.

Las restricciones libres de usuario, por último, son responsabilidad del usuario y el modelo de datos ni las reconoce ni las maneja.

Operaciones (Dinámica)

Las operaciones que se pueden realizar básicamente son las siguientes:

Selección: localización de los datos deseados.

Acción: realización de una operación sobre los datos seleccionados como:

- Recuperación (obtención de datos).
- Actualización (Modificación, Inserción o Borrado).

3.3. Representación del modelo conceptual

El modelo conceptual no es más que una simple abstracción del mundo real que pretendemos representar.

Una de las etapas del diseño será **representar el mundo de las ideas** o estructura percibida a la que hemos llegado a través de dicha abstracción.

Básicamente existen 5 **modelos de datos** que formalizan la representación de nuestro modelo conceptual:

- Modelo Entidad/Relación (E/R).
- Modelo Relacional.
- Modelo en Red.
- Modelo Jerárquico.
- Ficheros convencionales.

4. MÉTODO DE DISEÑO

Para llegar a una modelización de datos bajo un esquema relacional es necesario, en primera instancia, obtener un modelo conceptual que represente fielmente la realidad estudiada, para realizar, posteriormente, una “traducción” del mismo hacia el modelo relacional.

Dicha traducción puede realizarse directamente, el esquema de representación del **modelo entidad/relación**, es el de propósito más general por su sencillez y claridad.

Por esto *vamos a pasar del modelo conceptual al esquema relacional utilizando los formalismos del modelo entidad/relación como paso intermedio.*

Fases:

Síntesis: Se determinan, a partir del mundo real, las entidades, asociaciones y propiedades de unos y otros, sintetizando la adecuada estructura relacional a partir de las reglas.

Análisis: En esta fase se analiza una estructura relacional existente, descomponiéndola en nuevas estructuras relacionales más regulares que cumplen determinadas propiedades. Se analiza la relación y su grado de normalización con el fin de evitar anomalías en la actualización de datos.

5. FASE DE SÍNTESIS: MODELO ENTIDAD/RELACIÓN (E/R)

Conjunto de conceptos y reglas, que permiten formalizar el modelo en un grafo de representación.

Los *conceptos básicos* son:

Entidades: Objetos del mundo real (ej. un libro).

Relaciones: Asociaciones entre entidades (ej. un socio dispone de un libro).

Atributos: Propiedades de las entidades y relaciones (ej. título de un libro).

(1, N) **Cardinalidad.**

1:N **Correspondencias.**

Entidades (Concepto y Representación)

Una **entidad** es un objeto, una persona, un lugar, un concepto, etc., real o abstracto, de interés para la empresa u organización. Toda entidad tiene existencia propia.

Cada ocurrencia de la entidad debe poderse distinguir de las demás. Todas las ocurrencias (Profesor A, Profesor B, etc.) de un tipo de entidad deben tener los mismos atributos.

En el formalismo, *una entidad se representa mediante un rectángulo en cuyo interior se especifican los atributos (opcional).*

Relación (Concepto, Representación y Componentes)

Una **relación** es una asociación entre entidades. No tiene existencia propia. Una ocurrencia de relación no es distinguible por sí misma, sino a través de la designación de las ocurrencias de las entidades asociadas.

Todas las ocurrencias de una relación se caracterizan por tener la misma lista de propiedades. Por *ejemplo*, Ser Propietario, asocia Persona y Vivienda. Ocurrencias:

Juan es propietario de A.
Pepe es propietario de B.

En el formalismo, *la relación se representa mediante un rombo en cuyo interior se especifica el nombre de la relación.*

Si la relación posee atributos propios, éstos se representan en círculos enlazados con la relación, en cuyo interior se encuentra el nombre del atributo correspondiente.

Entre dos objetos (entidades) determinados pueden existir varias asociaciones (relaciones) distintas. Cuando la asociación se produce entre ocurrencias del mismo tipo de entidad se dice que es *recursiva*.

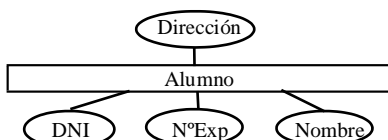
Los *componentes de una relación* son: *Colección* (lista de entidades involucradas en la asociación) y *Dimensión* (número de entidades involucradas en la asociación).

Atributos

Un **atributo** es una propiedad o característica de una entidad o de una relación. Distinguiremos entre *atributos principales* (identifican de forma unívoca cada ocurrencia de la entidad) y *atributos no principales* (propiedades no identificativas de la entidad).

Cada atributo se define por su nombre. Dentro de una misma entidad o relación dicho nombre debe ser único. El nombre de un atributo no principal puede repetirse en diferentes entidades o relaciones.

Los atributos *se representan mediante círculos enlazados con la entidad o relación, en cuyo interior se encuentra el nombre del atributo correspondiente.*



¿Entidad o Relación?

A menudo el diseñador encontrará que un objeto del mundo real es representable tanto por una entidad como por una relación.

En cada caso, la elección depende tanto del contexto donde nos encontramos como del criterio subjetivo del diseñador.

A la hora de discernir, será de gran importancia tener presente que la diferencia esencial entre entidad y relación, es que la primera tiene existencia propia, y la segunda sólo tiene razón de ser como asociación entre entidades.

Restricciones

A partir de este momento hay que determinar los atributos que definen los objetos del mundo real. Con esto, la semántica del problema estudiado no queda completamente, representada, pues hay

restricciones, bien declaradas en forma explícita, bien inherentes al mundo real; que han de tenerse en cuenta en la modelización. Además, el modelo debe completarse con la observación de determinados detalles referentes a la integridad de los datos.

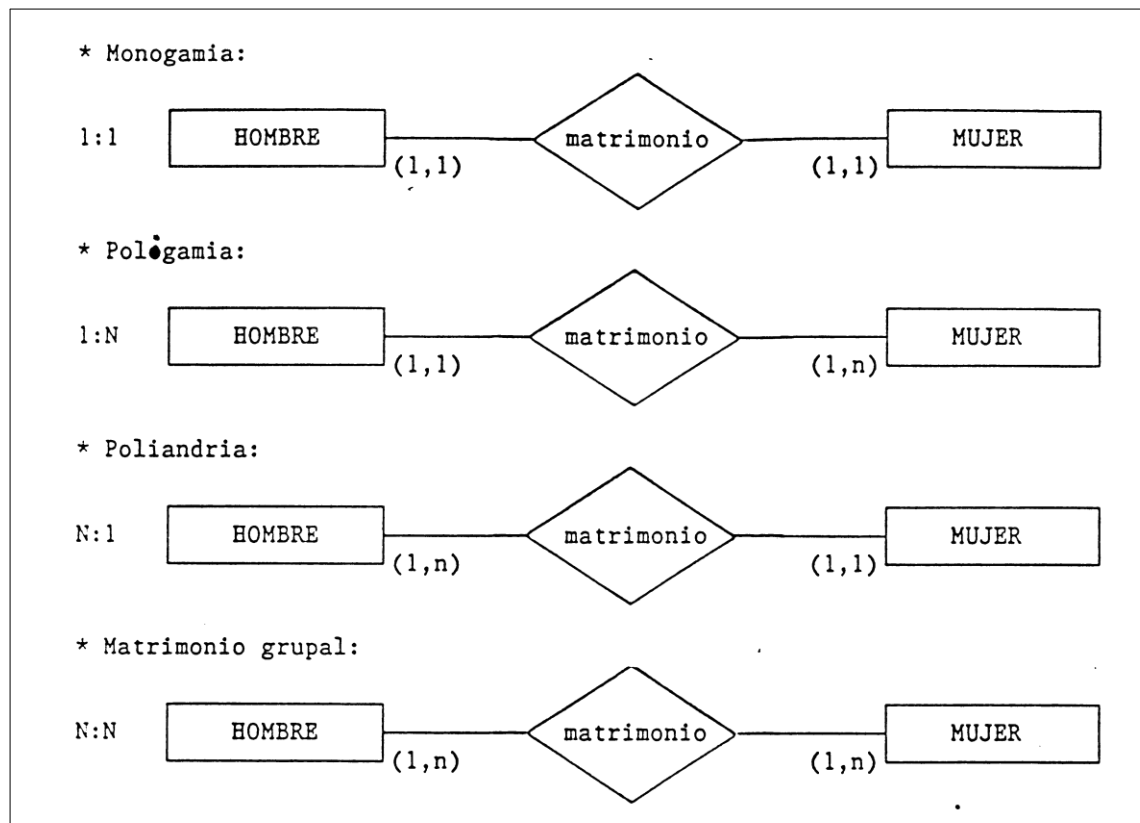
Limitaciones sobre las relaciones

Se llama **Cardinalidad** de una entidad involucrada en una relación al nº de veces mínimo y máximo que una misma ocurrencia de esa entidad puede aparecer en una ocurrencia de relación (n es el valor genérico más de 1; 0 ninguno y 1 el valor 1).

Para calcular la cardinalidad en relaciones binarias se fija una entidad para valorar la otra. En relaciones triples se fijan dos ocurrencias de cada una de las 2 entidades y así se valora la que queda libre. En relaciones con n entidades, se fijan $n - 1$ ocurrencias de cada una de las $n - 1$ entidades y se valora la entidad que queda libre.

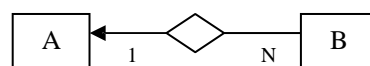
Tipos de correspondencia

Las relaciones quedan tipificadas bajo un tipo de correspondencia en función de las cardinalidades máximas de las entidades involucradas. *Ejemplo:*



Tendremos las siguientes *restricciones de cardinalidad*:

- De muchos a muchos. **N:N**
- De uno a muchos. **1:N**
- De uno a uno. **1:1**



(Al representarlas, la “punta de flecha” jugará el papel de “uno”)

Claves

En todos los modelos debe existir una forma de indentificar cada tupla de la relación. Para ello se utiliza un atributo o bien un conjunto de atributos.

- **Superclave:** Conjunto de atributos que permite identificar unívocamente una entidad dentro de un conjunto de entidades.
- **Clave candidata:** Toda superclave que deja de serlo al quitarle un atributo. Es algo así como una superclave mínima.
- **Clave primaria:** Es la clave elegida de entre las claves candidatas.
- **Claves externas:** Sirven para conectar las relaciones o tablas. Permiten representar las asociaciones que existen entre entidades. *Ejemplo*, dadas las relaciones:

EMP(NSS#, Nombre, Ciudad, Departamento#)

DEP(Departamento#, NombreDep, Otros)

Departamento# es clave externa de la relación EMP para la relación DEP. Dicho valor sirve para referenciar una tupla de DEP dentro de la relación EMP.

Reglas de Integridad

1ª. *Regla de integridad de la entidad:* Un valor de la clave primaria no puede ser nulo.

2ª. *Regla de integridad referencial:* Si sobre una clave externa aparece un valor distinto de nulo, tiene que existir necesariamente una tupla que de referencia con ese valor de clave primaria, es decir, todas las referencias tienen que ser íntegras y válidas.

Optimización

Hasta ahora hemos obtenido un diagrama que refleja fielmente la realidad estructurada. Podemos decir, que hemos elaborado lo que se conoce como *modelo bruto*. Esto es así, porque esta representación, a pesar de reflejar la realidad, no significa que sea la mejor posible, por esa razón buscaremos una optimización del diagrama. Dicha optimización se basa en que en un diagrama, la ruta de acceso a una determinada información ha de ser única, por lo que nos interesa analizar si alguna de las relaciones entre entidades, aporta una vía de acceso redundante.

A partir de esta optimización tendremos el llamado *diagrama refinado*.

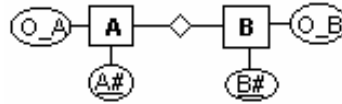
6. PASO DEL MODELO E/R AL ESQUEMA RELACIONAL

Una vez hemos formalizado nuestra concepción del mundo real a través del diagrama entidad/relación, es posible traducir éste hacia un modelo relacional. Se convierten entidades y relaciones en tablas siguiendo una serie de *normas*:

- a) Cada **entidad** del modelo E/R, se representa explícitamente mediante una tabla en el modelo relacional. Los atributos compondrán las columnas, cada fila representa una ocurrencia de la entidad y el conjunto de entidades será la tabla.
Si la entidad es *débil* (entidad a la que no se le encuentra ninguna superclave, existirá una dependencia por existencia y siempre estará relacionada con una entidad fuerte o una relación) se tendrán tantas columnas como atributos mas una columna con la llave primaria de la entidad fuerte o relación.
- b) Cada **relación** del tipo N:M del modelo E/R se representa explícitamente mediante una tabla cuya clave es la concatenación de las claves de las tablas representativas de las entidades

involucradas en la relación. Las columnas serán la claves primarias de las entidades relacionadas, el conjunto de relaciones es la tabla, y cada fila es una asociación.

Relación 1:1. En una de las dos relaciones se introduce como clave externa el atributo identificativo de la otra. *Ejemplo:*

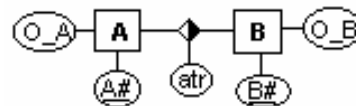


A (A#, O_A, B#)

B# es clave externa para la relación B

B (B#, O_B)

Relación 1:N. Se introduce en la relación “muchos” el atributo identificativo de la relación “uno” como clave externa. *Ejemplo:*

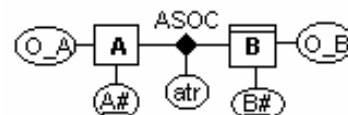


La relación que se representa en la entidad del lado “muchos” es la que incluye como clave externa el identificador de la entidad del lado “uno” y todos los atributos conectados sobre la asociación.

A (A#, O_A)

B (B#, O_B, A#, atr)

Relación N:N. Se crea una nueva relación con el nombre de la asociación que contenga los atributos identificadores de ambas relaciones además de los suyos propios en el caso de existir. (Recordemos que a las tablas se les suele llamar también relaciones). En el caso de que la clave primaria de la nueva relación no coincida con la concatenación de las claves primarias de las dos relaciones seguramente se ha realizado un diseño erróneo de la base de datos. *Ejemplo:*



Las relaciones muchos a muchos generan una nueva relación con el nombre de la asociación, la clave primaria de A, la clave primaria de B y los atributos de la asociación.

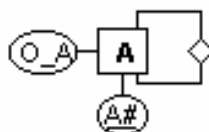
A (A#, O_A)

B (B#, O_B)

ASOC (A#, B#, atr)

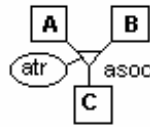
A#B# clave primaria, A# clav. ext., B# clav. ext.

Relación unaria: El funcionamiento es similar al de las relaciones binarias. *Ejemplo:*



A (A#, O_A, Ref_A#). No podemos poner A(A#, O_A, A#) pues habría dos atributos con el mismo nombre. Renombramos el último como Ref_A#.

Relación ternaria: Se convierte en una relación o tabla (con excepción de algunas relaciones 1:1:1 que no interesa convertirlas en relación y conviene llevar las claves externas a una de las tres relaciones junto con los atributos de la asociación). *Ejemplo:*



Las claves, serán:

- relación 1:1:1 => A#B# ó B#C# ó A#C# ó B#C# (cualquier par entre A#,B#,C#)
- relación N:1:1 => A#B# ó A#C# (la clave del muchos siempre participa)
- relación N:M:1 => A#B# (las dos del muchos)
- relación N:M:L => A#B#C# (las tres)

7. FASE DE ANÁLISIS

Analiza la estructura relacional existente, descomponiéndola en nuevas estructuras relacionales más regulares que cumplen determinadas propiedades. Se analiza la relación y su grado de normalización con el fin de evitar anomalías de actualización de datos.

Para que una Base de Datos Relacional este normalizada, sólo es necesario que los dominios simples subyacentes contengan sólo valores atómicos (que no tenga grupos repetitivos, intersección entre fila y columna).

El punto fundamental es, que una relación dada, aún cuando pudiera estar normalizada, podría poseer todavía ciertas propiedades indeseables. La **Teoría de la Normalización** nos permite reconocer esos casos y nos muestra cómo podemos convertir esas relaciones a una forma más deseable.

Los *objetivos* fundamentales de la normalización son:

- Eliminar ciertos tipos de redundancia.
- Evitar ciertas anomalías de actualización.
- Producir un diseño que sea una “buena” representación del mundo real: que sea fácil de entender intuitivamente y constituya una buena base para un crecimiento futuro.
- Simplificar la imposición de ciertas restricciones de integridad.

Las ideas de normalización son útiles pero no son una panacea:

- No eliminan todas las redundancias.
- Hay proyecciones de tablas (según Boyce-Codd) que están en conflicto.

Un tema fundamental dentro de la normalización son las dependencias.

Dependencias

Son restricciones de integridad que permiten conocer qué asociaciones existen entre los atributos en el mundo real.

Existe otro tipo de restricciones de integridad que se pueden expresar por medio de dependencias. Son

propiedades inherentes al contenido semántico de los datos. Además, se deben cumplir para cualquier posible extensión del esquema relacional.

No cambian en el tiempo mientras no cambie el mundo real. La existencia de una dependencia no se puede demostrar. Solamente podemos afirmar que existe la dependencia observando el mundo real que estamos modelizando.

Tipos: Funcionales, Multivaluadas, Jerárquicas, de Combinación, etc...

- **Dependencia funcional**

Dada la relación $R(t)$ que contiene los atributos X e Y se dice que Y depende funcionalmente de X si, y solo si, en todo momento cada valor de X tiene asociado un sólo valor de Y .

$X \rightarrow Y$ (X determina Y)

- **Dependencia funcional completa (o total)**

Si el atributo X es compuesto $X(X_1, X_2)$, Y tiene dependencia funcional completa respecto de X si:

$X \rightarrow Y$ con X_1 sin determinar a Y y X_2 sin determinar a Y . Se representa por:

$X \Rightarrow Y$

Por ejemplo, (PROVEEDOR, ARTÍCULO) \Rightarrow PRECIO

El artículo tendrá un precio u otro, dependiendo del proveedor que lo distribuya. Siempre se trabaja con dependencias funcionales completas.

- **Dependencia transitiva**

Dada la relación $R(X, Y, Z)$ en la que existen las siguientes dependencias funcionales:

$X \rightarrow Y$ y por otro lado $Z \rightarrow Y$; se dice que Y tiene una dependencia transitiva de X a través de Z . Es decir:

$X \rightarrow Y$

A partir de estas dependencias se realiza la Normalización del esquema relacional que consiste en la descomposición sin pérdida de información de la relación universal (o de una colección de relaciones equivalentes a la misma) en una colección de relaciones en las que las anomalías de actualización (inserción, borrado y modificación) no existan o sean mínimas.

- **Descomposición de una relación**

La descomposición es un proceso de refinamientos sucesivos que deberá conducir a aislar las entidades y asociaciones del mundo real.

Consiste en la sustitución de una relación $R(A_1, A_2, \dots, A_n)$ por una colección de relaciones R_1, R_2, \dots, R_p tales que la combinación de los atributos de R_1, R_2, \dots, R_p de en extensión, la relación R (igual a la combinación de los R_i).

- **Descomposición sin pérdida de información**

Es la llamada propiedad LLJ (Loss Less Join). La descomposición sin pérdida de información de una relación $R(A_1, A_2, \dots, A_n)$ es una descomposición en p relaciones R_1, R_2, \dots, R_p , tal que, para toda extensión de R se cumple:

$R = R_1 * R_2 * \dots * R_p$ (* es la unión)

Después de lo expuesto, se puede concluir que los pasos necesarios para diseñar una base de datos relacional partiendo del modelo conceptual y utilizando como paso intermedio el modelo E/R son los siguientes:

- 1º) Determinar entidades.
- 2º) Establecer las relaciones.
- 3º) Definir atributos.
- 4º) Calcular cardinalidades.
- 5º) Eliminar redundancias.
- 6º) Dependencias.
- 7º) Convertir el modelo E/R en estructura relacional.
- 8º) Normalizar.

Un *ejemplo* muy simple podría ser el siguiente:

Supongamos que se pretende diseñar un sistema de base de datos sobre personas, municipios y viviendas.

Los supuestos o restricciones semánticas son las siguientes:

- Cada persona puede habitar sólo una vivienda.
- Cada persona puede poseer más de una vivienda.

1º) Entidades: municipio; vivienda; persona.

2º) Relaciones: Habita, Posee, Está empadronado, Está en.

3º) Atributos:

Persona: Nombre, Municipio.

Municipio: Nombre.

Vivienda: N°

4º) El diagrama con las cardinalidades y correspondencias será el de la página siguiente.

5º) La relación Está empadronado: es redundante.

Suponiendo que los datos están normalizados las tablas necesarias serían:

PERSONA (# NOMBRE_P, NOMBRE_M, N°_VIVIENDA)

MUNICIPIO (# NOMBRE_MUNICIPIO)

VIVIENDA (# N°_VIVIENDA, NOMBRE_M)

POSEE (# NOMBRE_P, # NOMBRE_M)

8. BIBLIOGRAFÍA

Gary W. Hansen

Diseño y Administración de Bases de Datos

Prentice Hall, 1998

Alberto Prieto

Introducción a la Informática

Mc Graw-Hill, 2ª edición, 1997

Alfonso Ureña López

Fundamentos de Informática

Ra-ma, 1997

