

# Capítulo II Modelos de los datos.

## II.1 Abstracción de datos.

Un sistema de bases de datos es una colección de archivos interrelacionados y un conjunto de programas que permiten a los usuarios tener acceso y modificar estos archivos. Uno de los propósitos principales de un sistema de bases de datos es proporcionar al usuario una visión *abstracta* de los datos. Es decir, el sistema esconde ciertos detalles de cómo se almacenan y mantienen los datos.

Para que el sistema sea útil debe recuperar los datos eficientemente, es decir, rápidamente. Con la finalidad de lograr esto, los diseñadores construyen estructuras de datos que a veces son muy complejas. Los desarrolladores esconden la complejidad a los usuarios a través de varios *niveles de abstracción*, esto simplifica la interacción de los usuarios con los sistemas. Los tres niveles de abstracción que se utilizan son los siguientes:

- *Nivel físico.*- Es el nivel más bajo de abstracción y describe como se almacenan realmente los datos, en este nivel se describen con detalle las estructuras complejas de los datos.
- *Nivel lógico.*- Es el siguiente nivel de abstracción, describe que datos se almacenan en la base de datos y que relaciones existen entre esos datos. En este nivel, la base de datos completa se describe en términos de un número pequeño de estructuras relativamente simples. Los que trabajan con este nivel son los administradores de bases de datos, ellos son las personas que deciden que información se mantiene en la base de datos. La implementación de estructuras simples a nivel lógico puede involucrar estructuras complejas a nivel físico.
- *Nivel externo.*- Es el nivel más alto de abstracción, describe solo parte de la base de datos completa, la que el usuario necesita. El sistema puede proporcionar diferentes vistas de una misma base de datos.

Usando una analogía para comprender mejor lo que son los diferentes niveles de abstracción, supongamos que necesitamos un registro que capture los siguientes datos para cada alumno:

Su nombre completo  
Su matrícula  
Su promedio.

Esto es lo que el usuario ve y con lo que trabaja y sería el equivalente al *nivel externo*. Para implantar esto en un lenguaje de programación, por ejemplo C o C++, podemos declarar una estructura que agrupe toda esta información:

```
struct s_alumno{  
    char    nombre[40];  
    long int matricula;  
    float    promedio;  
};
```

Con esta estructura trabajarían los diseñadores del programa, y sería el equivalente al *nivel lógico* de abstracción. Finalmente, el *nivel físico*, estaría determinado por el orden y la cantidad de bytes que ocupan los tres campos de la estructura s\_alumno.

### II.1.1 Ejemplares y esquemas.

Las bases de datos van cambiando a lo largo del tiempo conforme la información se inserta y se borra. La colección de información almacenada en la base de datos en un momento particular se denomina un *ejemplar* de la base de datos. El diseño completo de la base de datos se llama el *esquema* de la base de datos. Los esquemas raramente se modifican. Utilizando una vez más la programación para hacer una analogía, un *esquema* de base de datos corresponde a las declaraciones de las variables. Cada variable tiene un valor particular en un instante de tiempo. Los valores de las variables en un programa en un instante de tiempo corresponde a un *ejemplar* de un esquema de bases de datos.

## II.2 Definición del modelo de datos.

“Un modelo de datos es una colección de herramientas conceptuales para describir los datos, las relaciones, la semántica y las restricciones de consistencia” [Silberschatz et al., 2003].

“Un modelo de datos es un conjunto de conceptos y reglas que permiten estructurar los datos resultantes de la observación de la realidad, de forma que queden representadas todas sus propiedades, tanto estáticas como dinámicas” [Celma et al., 2003].

“Un modelo de datos es una notación para describir información acerca de los datos” [García-Molina et al., 2009].

Un modelo de datos es una representación, usualmente gráfica, de estructuras de datos. Los modelos de datos de alto nivel, o conceptuales, utilizan conceptos muy cercanos a la forma en que los usuarios perciben los datos, mientras que los modelos de bajo nivel o físicos describen detalles de cómo se almacenan los datos en la computadora.

### II.3 Clasificación de los modelos de datos.

Los modelos de datos se clasifican en tres grupos:

- Modelos lógicos basados en objetos.
- Modelos lógicos basados en registros.
- Modelos físicos de datos.

Los *modelos lógicos basados en objetos* se usan para describir datos en los niveles conceptuales. Se caracterizan por que proporcionan capacidad de estructuración flexible y permiten especificar restricciones en los datos explícitamente.

Los modelos lógicos basados en objetos más conocidos son:

- Modelo Entidad-Relación.
- Modelo orientado a objetos.
- Modelo binario.
- Modelo semántico de datos. Basado en modelos de redes semánticas con sus raíces en la inteligencia artificial.
- Modelo funcional de datos.

Los *modelos lógicos basados en registros* se utilizan para describir datos en los niveles conceptual y físico, permiten especificar la estructura lógica global de la base de datos y proporcionan una descripción a un nivel más alto en la implantación. Estos modelos se llaman así porque la base de datos está estructurada en registros de formato fijo de varios tipos. Cada registro tiene un número fijo de campos, que su vez son de tamaño fijo. El uso de registros de tamaño fijo facilita la implantación del nivel físico de la base de datos.

Los modelos lógicos basados en registros más conocidos son:

- Modelo relacional.
- Modelo de red.
- Modelo jerárquico.

En este curso estudiaremos el modelo relacional. El modelo de red y el modelo jerárquico son un antecedente del modelo relacional, actualmente se usan muy poco, ya que complican

la tarea del modelado de datos, sin embargo aún están implantados en el código de bases de datos antiguas.

Los *modelos físicos de datos* se usan para describir datos en el nivel más bajo. Son de interés principalmente para los fabricantes de SGBD. Los más conocidos son:

- Modelo unificador.
- Modelo de elementos.

## II.4 Tipos de bases de datos.

Existen cuatro tipos diferentes de bases de datos:

- Bases de datos jerárquicas.
- Bases de datos de red.
- Bases de datos relacionales.
- Bases de datos orientadas a objetos.

Las *bases de datos jerárquicas* constituyen el primer modelo lógico de bases de datos que surgió. Es un modelo rígido soportado sobre una estructura de árbol con relaciones exclusivas de padre/hijo, las bases de datos jerárquicas pretenden modelar relaciones jerárquicas del mundo real. Con este tipo de base de datos se obtiene unos excelentes resultados en casos en los que en los modelos donde prevalece el tipo de relación 1:N.

En las bases de datos jerárquicas la representación gráfica se apoya sobre un conjunto de árboles cuyos nodos representan entidades de información y los segmentos de unión representan relaciones 1:N.

El modelo jerárquico utiliza dos conceptos de estructuración: *registros* y *vínculo padre-hijo*. Un *registro* es una colección de valores de campo que proporcionan información sobre una entidad. Un *tipo de vínculo padre-hijo* es un vínculo 1:N entre dos tipos de registros. El tipo de registros del lado 1 se denomina *tipo de registros padre*, y el del lado N se denomina *tipo de registros hijo*.

Un esquema de base de datos jerárquica consiste en varios esquemas jerárquicos. Cada esquema jerárquico consta de varios tipos de registros y varios tipos de vínculo-padre-hijo. En la *figura 2.1* se muestra un ejemplo de esquema jerárquico.

El problema principal del modelo jerárquico es que el mundo real no se adapta fácilmente a este tipo de organización [Martens, 1999].

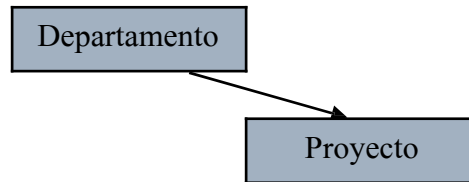


Figura 2.1: Esquema de base de datos de tipo jerárquico (Material del curso del Dr. Ovidio Peña Rodríguez).

Las *bases de datos de red* se basan en dos estructuras básicas: *registros* y *conjuntos*. Cada registro consiste en un grupo de valores de datos relacionados entre sí. Hay diferentes *tipos de registros*, cada uno de los cuáles tiene un nombre. Las relaciones entre los datos se representan mediante enlaces, los cuáles pueden verse como apuntadores. Los registros se organizan como colecciones de grafos arbitrarios.

Un *tipo de conjunto* es un vínculo 1:N entre dos tipos de registros.

Cada definición de tipo de conjuntos consta de 3 elementos:

- Un nombre para el tipo de conjuntos.
- Un tipo de registros propietario.
- Un tipo de registros miembro.

Cada ejemplar del conjunto relaciona un registro de tipo propietario, con el conjunto de registros miembros.

En la *figura 2.2* se muestra un ejemplo de esquema de red.

Figura 2.2: Esquema de base de datos de tipo red (Material del curso del Dr. Ovidio Peña Rodríguez).

El modelo en red se desechó debido a dos razones fundamentales, la primera es que para obtener cualquier información era indispensable tener una idea muy clara de cómo estaban

organizados los datos y la segunda razón es que no existían herramientas sencillas que permitieran realizar consultas arbitrarias en una base de datos [Martens,1999].

Las *bases de datos relacionales* fueron definidas por el matemático Codd en los años 70. La teoría relacional ha ido evolucionando a lo largo del tiempo, incluyendo cada vez nuevas características. Este modelo representa los datos y las relaciones entre los datos mediante una colección de tablas, cada una de las cuáles tiene un número de columnas con nombres únicos. En el capítulo III se estudia con más detenimiento las bases de datos relacionales, ya que es un modelo muy utilizado en la actualidad.

El *modelo de datos orientado a objetos* es otro modelo de datos que está recibiendo una atención creciente, éste se puede observar como una extensión del modelo entidad-relación con los conceptos adicionales de encapsulación, métodos(funciones) e identidad de objetos, que son parte fundamental del diseño orientado a objetos.

Las *bases de datos orientadas a objetos* se propusieron con la idea de satisfacer las necesidades de aplicaciones complejas, como por ejemplo estructuras complejas de datos, transacciones de mayor duración que las tradicionales y accesos a múltiples bases de datos.

Las bases de datos orientadas a objetos permiten al diseñador especificar tanto la estructura de objetos complejos como las operaciones que se pueden aplicar entre los mismos. Una base de datos orientada a objetos provee una *identidad única* a cada objeto independiente almacenado en la base de datos y se parte de la base de que los objetos complejos pueden construirse a partir de otros más simples.

A diferencia de las entidades en el modelo relacional, cada objeto tiene su propia identidad única independiente de los valores que contiene. Así, dos objetos que contienen los mismos valores son, sin embargo, distintos.