

TEMA 2: MODELOS DE DATOS

1- INTRODUCCIÓN A LOS MODELOS DE DATOS

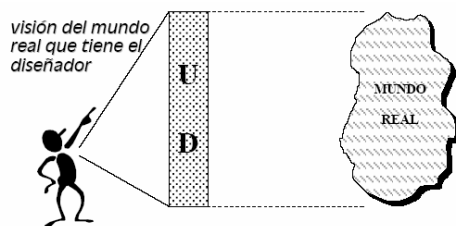
Para crear un producto software el ser humano se vale también de lenguajes (lenguajes de programación en la fase de construcción o programación, lenguajes gráficos o matemáticos en la fase de concepción: análisis y diseño del sistema).

Una parte estructural muy importante en un producto software complejo pueden ser las estructuras de datos. Se requerirán lenguajes que permitan especificar la estructura de estos datos en la fase de concepción: análisis y diseño y en la fase de implementación o construcción del sistema. Para crear un lenguaje es necesario convenir unas reglas que permitan transmitir el conocimiento.

¿Qué es un modelo...?... ¿Qué es modelar...?

MODELAR: (Flory 1982) “Consiste en definir un mundo abstracto y teórico tal que las conclusiones que se puedan sacar de él coinciden con las manifestaciones aparentes del mundo real”.

UNIVERSO DISCURSO: “Visión que del mundo real tiene el diseñador”.



Todo software ha de tener un alcance (limitado) funcional: está dirigido a una farmacia, a una universidad, a una petrolera, etc.

El diseñador debe establecer el contorno del problema; es decir “lo que forma y lo que no forma parte del problema”. Este contorno se denomina universo de discurso y es la parte del mundo que, para el fin del software a construir, interesa al diseñador. Lo que está dentro del contorno forma parte del problema y lo que está fuera no forma parte del problema. Es relevante definir claramente este contorno.

MODELO DE DATOS: “Un conjunto de conceptos, reglas y convenciones que nos permiten describir los datos del Universo Discurso que deseamos almacenar en la base de datos”.

Los modelos de datos permiten establecer las estructuras y las restricciones de los datos. Deben utilizarse para diseñar, son el elemento básico de diseño en la metodología de creación de una BBDD.

Un modelo de datos va a ser ese conjunto de conceptos y reglas que, dotado con una sintaxis, formalizan un lenguaje. Con los lenguajes (p.ej. SQL es un lenguaje basado en el modelo relacional de datos, en concreto, está basado en el cálculo relacional de tuplas) se podrán expresar diseños o concepciones de estructuras de datos: esquemas de esos datos.

Así existen conceptos asociados al de modelos de datos: esquemas, objetos, propiedades, asociaciones, operaciones, restricciones, etc.; se realiza una clasificación de los diversos tipos de modelos de datos y se estudian los principales mecanismos de abstracción utilizados en esta área.

TEMA 2: MODELOS DE DATOS

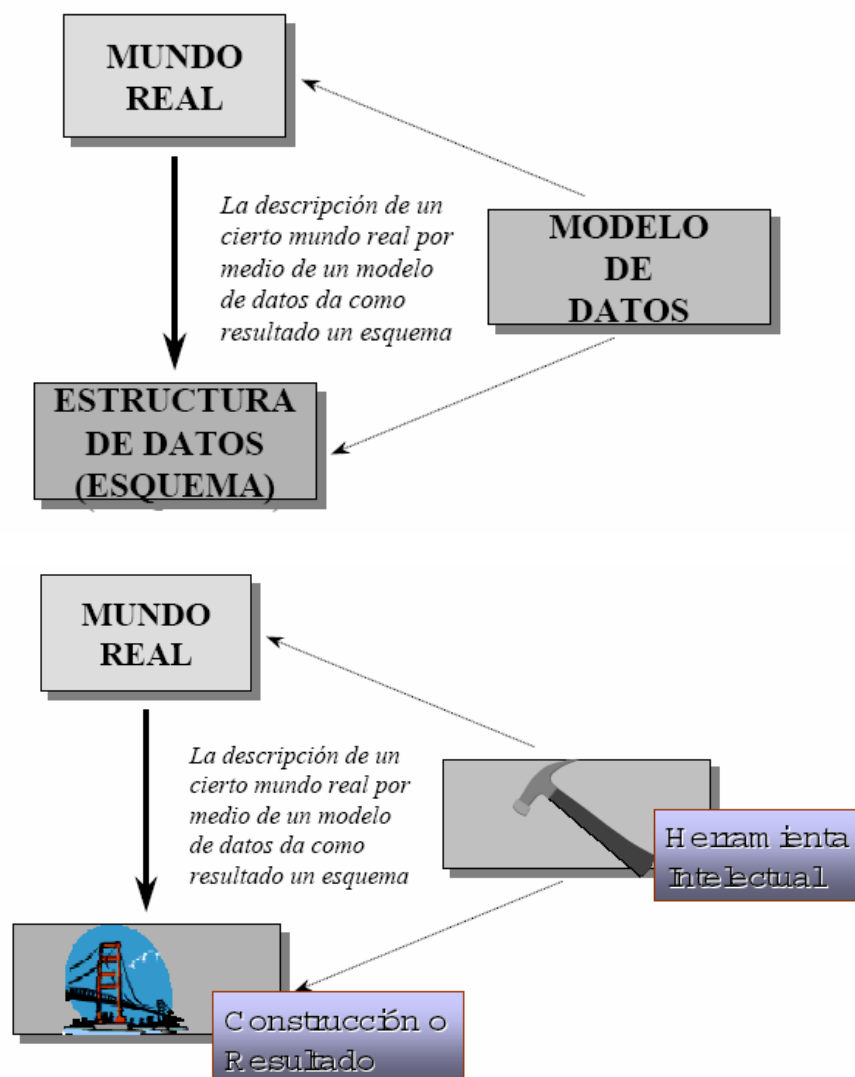
Es importante distinguir entre modelo de datos y esquema.

Un **ESQUEMA** puede definirse como:

- Dittrich (1994): "Descripción específica de un determinado mini-mundo en términos de un modelo de datos se denomina esquema (o esquema de datos) del mini-mundo. La colección de datos que representan la información acerca del mini-mundo constituye la base de datos".
- De Miguel, Piattini y Marcos (1999): "Representación de un determinado mundo real (universo del discurso) en términos de un modelo de datos".

ESQUEMA: Descripción de la estructura de la base de datos

OCURRENCIA DEL ESQUEMA: Hace referencia a los datos que se encuentran almacenados en la base de datos en un determinado momento



TEMA 2: MODELOS DE DATOS

2- CARACTERÍSTICAS DE UN MODELO DE DATOS - COMPONENTES (G,O)

Los componentes de todo modelo de datos se pueden estructurar en estática y dinámica del modelo. Un modelo de datos (MD) se define formalmente como el par $MD = \langle G, O \rangle$.

- **COMPONENTE ESTÁTICA (G):** conjunto de reglas de generación que permiten representar la componente estática (estructuras del modelo y sus restricciones). Estudia las propiedades invariables en el tiempo, aquellas que funcionan como contenedores de información, son las estructuras de información.
- **COMPONENTE DINÁMICA (O):** conjunto de operaciones autorizadas sobre la componente estática G y operaciones que permiten representar la componente dinámica. Estudia las propiedades variables en el tiempo, las relacionadas con el contenido de información, los datos.

La **estática** permite formalizar la especificación de estructuras admitidas (serán el soporte de los lenguajes de definición de datos: LDD o DDL (data definition languages). La representación ellas depende de cada modelo de datos, pudiendo hacerse en forma de grafos (Red CODASYL, ME/R de Chen, UML), relaciones o conjuntos (Relacional), árboles (jerárquico), etc. Es fundamental conocer el tipo de estructuras admisibles y la semántica que soportan.

La componente estática del modelo de datos engloba los siguientes elementos:

- **Objetos** (entidades, relaciones, registros, etc.)
- **Asociaciones** entre objetos (interrelaciones, ...)
- **Propiedades** o características de los objetos o asociaciones (atributos campos elementos de datos, etc.)
- **Dominios** (Conjuntos nominados de valores sobre los que se definen las propiedades)
- **Restricciones:** Limitaciones impuestas a la estructura del esquema o a los datos que invalidan ciertas ocurrencias de la base de datos. Expresan los elementos no permitidos.

La **dinámica** es un componente del modelo que facilitará la manipulación de instancias de las estructuras de la base de datos definidas con los componentes estáticos (serán el soporte de los lenguajes de manipulación de datos LMD o DML (data manipulation languages) que permitirán escribir transacciones). La dinámica ofrecerá operadores o expresiones para realizar acciones sobre los datos.

TEMA 2: MODELOS DE DATOS

CLASIFICACIÓN DE LAS COMPONENTES

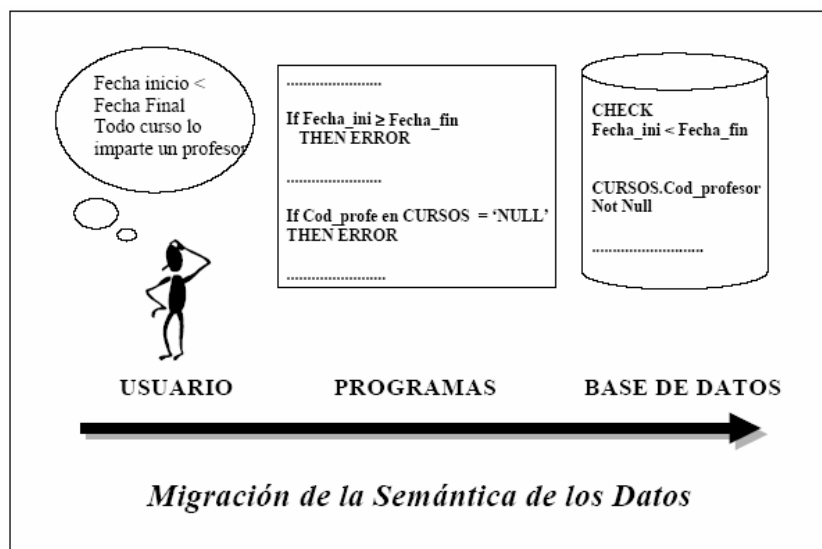
Ge: Conjunto de reglas que permiten definir las estructuras de datos y las **restricciones inherentes** (las propias del modelo de datos, no se pueden cambiar)

Gr: Conjunto de reglas que permiten definir al usuario las restricciones generales.

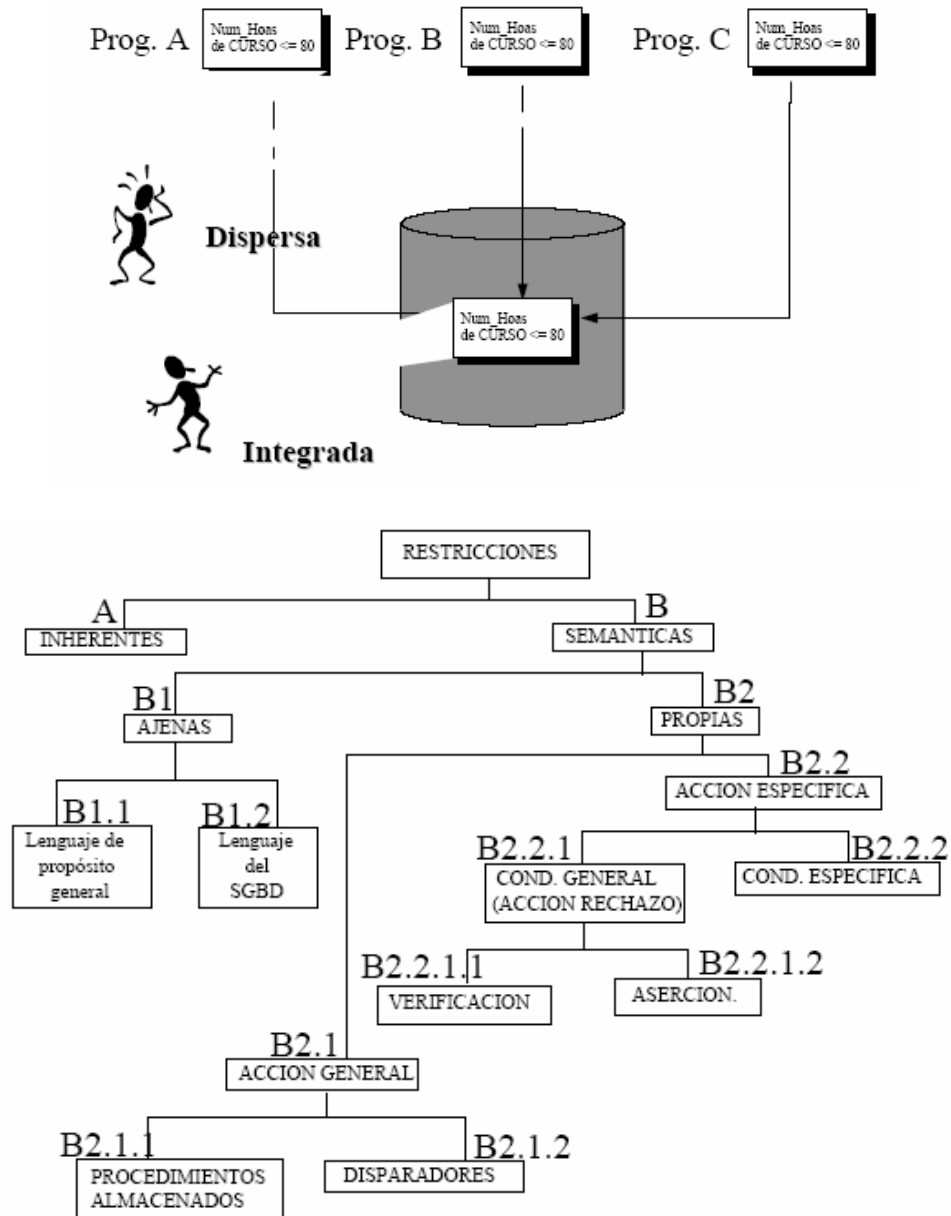
- **Restricciones propias u opcionales** (semántica integrada): su definición corresponde al diseñador, pero su gestión es responsabilidad del modelo de datos, el cual las reconoce y recoge en el esquema. La reusabilidad está garantizada al especificarse universalmente las reglas.
- **Restricciones ajenas, libres o de usuario** (semántica dispersa): son, por completo, responsabilidad del diseñador, el modelo no las reconoce, ni proporciona herramientas para manejarlas. El diseñador tiene que hacer código ajeno a la BD para incluirlas. Se dificulta la reusabilidad y se pueden generar colisiones o inconsistencias de reglas.

Os: Operaciones que permiten la selección de datos (Selección).

Oa: Operaciones relacionadas con la manipulación de datos (Acción).



TEMA 2: MODELOS DE DATOS



TEMA 2: MODELOS DE DATOS

3- TIPOS DE MODELOS DE DATOS

- **M.D. EXTERNOS:** Aquellos modelos de datos que permiten representar el nivel externo de los diferentes usuarios (orientados a cada usuario en particular).
- **M.D. CONCEPTUALES:** Aquellos modelos de datos que permiten representar el nivel conceptual de un sistema de información. Ej.: E/R, E/R Extendido. Orientados al conjunto de usuarios (contienen el modelo integrado para toda la organización).
- **M.D. INTERNOS:** Aquellos modelos de datos que permiten representar el nivel interno de los diferentes usuarios. Ej.: Jerárquico, en Red y Relacional. Consideran la implementación física (entorno tecnológico: sistema operativo, organización de archivos soportados, etc.).

Esta primera clasificación se puede agrupar en dos: **MODELOS LÓGICOS**, ambos describen aspectos lógicos de los datos (Externos y conceptuales) y **FÍSICOS** (Internos), que describen aspectos físicos.

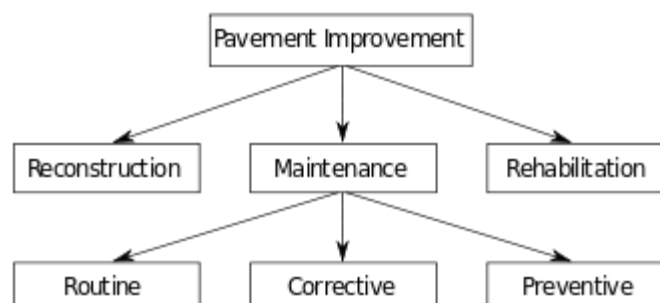
MODELO JERÁRQUICO

Consiste en que todas las interrelaciones de los datos se basan en **jerarquías**. Los archivos se conectan entre sí mediante punteros físicos (dirección física que indica donde puede encontrarse un registro sobre el disco) o campos añadidos a los registros individuales.

Almacena la información en una estructura jerárquica que enlaza los registros en forma de estructura de árbol (similar a un árbol visto al revés). Así, en una jerarquía un nodo padre (registro propietario) puede tener varios nodos hijo (registro subordinado), y así sucesivamente, pero un nodo hijo sólo puede tener un nodo padre. Por ejemplo si tenemos un fichero de facturas (FACTURA) y otro de líneas de factura (LINEA_F), el padre (registro propietario) sería los datos de la factura y los hijos (registros subordinados) las líneas de la factura.

Este modelo tenía algunas limitaciones ya que no todas las interrelaciones se pueden representar en una estructura jerárquica. Para intentar solucionar estas limitaciones se desarrollaron los sistemas de base de datos en red. Ejemplos de bases de datos jerárquicas son, IMS (Information Management System) de IBM y ADABAS (Adaptable Database System) de la empresa alemana Software AG.

Hierarchical Model



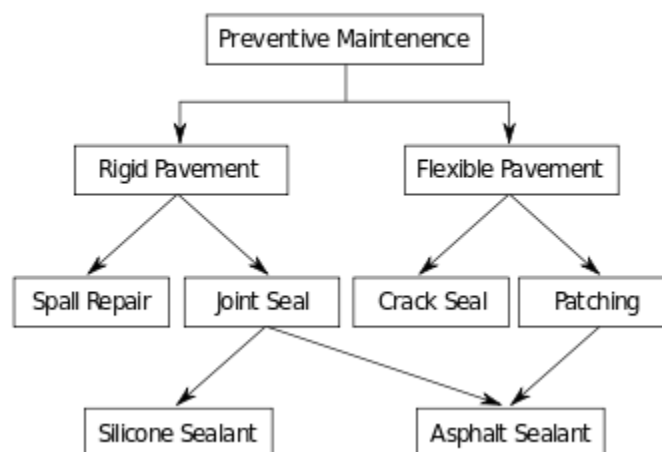
TEMA 2: MODELOS DE DATOS

MODELO EN RED

La estructura de árbol jerárquico se puede considerar como un caso especial de la estructura de un modelo en red. Los modelos en red al igual que los jerárquicos utilizan punteros físicos; pero la estructura de un modelo de datos en red, llamada algunas veces estructura de **plex**, abarca más que la estructura de árbol, en este caso, un padre puede tener muchos hijos y un hijo puede tener muchos padres, es decir, un registro puede estar subordinado a registros de más de un archivo. Pueden establecerse relaciones entre nodos hermanos, y en este caso, la estructura en forma de árbol se convierte en una estructura en forma de **grafo dirigido**.

Se desarrollaron y se comercializaron varios SGBD en red y este modelo de datos se normalizó como el **Modelo Codasyl**. Ejemplos de bases de datos en red son ADABAS, TOTAL, IMAGE,...

Network Model



MODELO RELACIONAL

El uso de punteros era simultáneamente una fortaleza y una debilidad de los sistemas de bases de datos jerárquicos y en red. Los punteros permitían una rápida recuperación de los datos, pero las interrelaciones tenían que definirse antes de que el sistema se pusiera en explotación. Era muy difícil recuperar datos basados en otras interrelaciones.

En 1970, E.F. Codd publica un artículo en el que argumenta que los datos deberían relacionarse mediante interrelaciones naturales, lógicas e inherentes a los datos, más que mediante punteros físicos. Codd propone así, un modelo simple de datos en el que todos ellos se representarían en tablas constituidas por filas y columnas. A estas tablas se les dio el nombre de **relaciones** y por eso se denominó al modelo relacional. Pensando en cada **relación** como si fuese una tabla que está compuesta por registros (cada fila de la tabla sería un registro o "**tupla**") y columnas (también llamadas "**campos**").

TEMA 2: MODELOS DE DATOS

Codd también propuso dos “lenguajes” para manipular los datos en las tablas:

- **Álgebra relacional:** conjunto de operaciones que describen paso a paso cómo computar una respuesta sobre las relaciones. Denominada de tipo procedimental.
- **Cálculo relacional:** de tipo declarativo. Lenguaje de consulta que describe la respuesta deseada sobre una Base de datos sin especificar como obtenerla.

Sean las relaciones:

Alumno			Maestro		
Exp	Nombre	Edad	Exp	Nombre	Edad
001	Maria	24	004	Ana	34
002	Alicia	25	006	Jose	39
050	Sara	28	057	Isabel	30
			072	Pablo	32

*para proyectar sólo algunos atributos de una relación se requiere utilizar el cuantificador existencial

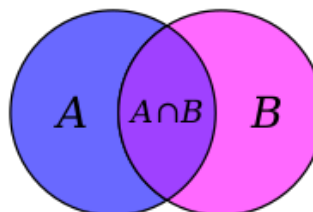
1. Obtener la información de todos los alumnos
 $\{t/ t \in \text{Alumno}\}$
2. Obtener la información de los alumnos que son mayores de 25 años
 $\{t/ t \in \text{Alumno} \wedge t[\text{edad}] > 25\}$
3. Obtener solamente los números de expedientes de los alumnos
 $\{t/ \exists s \in \text{Alumno} (t[\text{exp}] = s[\text{exp}])\}$
4. Obtener los números de expedientes de los alumnos mayores de 25 años
 $\{t/ \exists r \in \text{Alumno} (t[\text{exp}] = r[\text{exp}] \wedge r[\text{edad}] > 25)\}$

La manipulación lógica de los datos también hace factible la creación de lenguajes de interrogación más accesibles para un usuario no especialista en programación. Es un modelo de datos basado en la lógica de predicados y en la teoría de conjuntos.

Lógica de predicados.

Conexiones lógicas y Términos de enlace.
Palabras de enlace que unen proposiciones atómicas para formar proposiciones moleculares.

Término	Significado	Símbolo
AND	"Y"	&
OR	"O"	V
NOT	"No"	¬
IF	"Si.. entonces"	→



Actualmente los sistemas relacionales son un estándar en el mercado, especialmente en operaciones comerciales. Ejemplos de sistemas (motores) de bases de datos relacionales son: INFORMIX, SYBASE, DBASE, FOXPRO, INGRES, DB2, ORACLE, MySQL, MS SQL Server, Amazon Relational Database Service (RDS), Amazon Aurora, PostgreSQL, MariaDB, MS Access.

Motores de bases de datos de Amazon RDS



TEMA 2: MODELOS DE DATOS

4- LOS MODELOS DE DATOS EN EL PROCESO DE DISEÑO.

Se conoce como **proceso de diseño de una BD** al conjunto de tareas necesarias para representar un sistema objetivo (Universo de discurso) mediante un esquema. Los MD juegan un importante papel en el proceso de diseño de una BD al ofrecer el soporte de abstracción adecuado para obtener estos esquemas.

Los objetivos que persigue todo MD son de dos tipos:

- **Formalización.** El MD permite definir formalmente las estructuras y restricciones a través de lenguajes de datos.
- **Base del diseño.** El MD es un elemento fundamental en el desarrollo de una metodología de diseño de BD; entorno al esquema formalizado se construirán los otros artefactos.

