

Tema 1 Base de datos

Sistema de ficheros

Teníamos por una parte los ficheros necesarios para contener los **datos** con una estructura determinada y por otro los **programas de aplicación** que accedían a estos datos para producir información.

Hasta no hace mucho tiempo el lenguaje de programación COBOL era el más extendido

Sistemas de bases de datos

A partir de innovaciones tecnológicas tuvieron que:

Implementar la posibilidad de hacer consultas y actualizaciones de los mismos datos

.Interrelacionar archivos

Eliminar la redundancia, la repetición innecesaria de dato

A finales de los años 60 surgen **las bases de datos**: Como *conjuntos de archivos relacionados entre sí, compartidos por diferentes procesos y usuarios simultáneamente, y con estructuras complejas*.

En una **base de datos** se almacenan datos y los programas que utilicen esos datos no se tienen que preocupar del almacenamiento físico de los mismos.

Podremos utilizar una base de datos para cosas tan sencillas como mantener un registro de todos los **vídeos de nuestras películas**, nuestra **biblioteca personal** o la **música** que tenemos almacenada o tan complicadas como llevar toda la gestión de nuestra empresa.

2. Ficheros

1. Introducción: Concepto Sobre los Ficheros

El elemento que permitió llevar a cabo el almacenamiento de datos de forma permanente en dispositivos de memoria masiva fue **el fichero o archivo**.

La representación informática más frecuente en el ámbito de las BD es la representación tabular (*en forma de tabla*), la que se implementa habitualmente en archivos que se estructuran en **registros** y **campos**:

Los ficheros están formados por **registros** que contienen datos relativos a un mismo elemento u objeto (**entidad**).

En el ejemplo, un **registro** lo formarían todos los datos relativos a un cliente concreto: su nombre

Campo o **atributo**: equivale a una columna de la representación tabular. Los **campos** o **atributos** son las propiedades

Por definición, un **Fichero** es un conjunto de registros relacionados.

2. Tipos de Ficheros

Los tipos de acceso a un registro son:

1. Fichero Secuencial.

posiciones físicamente contiguas, sin dejar huecos entre ellos, en el mismo orden en que hayan sido introducidos.

Las ventajas de los ficheros con organización secuencial son:

- Rapidez en el acceso a un bloque de registros contiguos
- No se desperdicia espacio
- Se pueden utilizar cualquier tipo de registros: de longitud fija, variable o indefinida.

Los inconvenientes de este tipo de organización son:

Para acceder al registro n hay que recorrer los n-1 registros anteriores

La adición de registros se realiza a continuación del último registro ya existente. No se pueden insertar nuevos registros.

No se pueden eliminar registros

Para mantener ordenado y compactado el fichero, hay que crear un fichero nuevo a partir del existente.

2. Acceso Directo.

Accede a traves de una clave

Campo clave: campo que permite identificar y localizar un registro de manera ágil y organizada.

Cada uno de los registros se guarda en una posición física, que dependerá del espacio disponible en memoria masiva.

A través de una transformación específica aplicada a la clave, se obtendrá la dirección física en la que se encuentra el registro.

Varios tipos de acceso:

Directo

Indexado

Calculado

Clave --> Direccion Clave -- Clave-Direccion -- > Direccion Clave---Funcion matema-- > Direcc

Acceso directo (la clave coincide con la direccion)

-Mas rapida

-Se borra con un 0

-Permite algortimos de compactación de huecos

-Los archivos se crean con un tamaño definido

3. Ficheros Indexados

Se graban secuencialmente pero se recuperan con acceso directo gracias a un fichero adicional

En este tipo de ficheros se distinguen tres áreas:

Área Primaria: En esta área se escriben los registros ordenados ascendentemente por el valor de su clave. Esta área del fichero está dividida en segmentos. Cada segmento almacena “n” registros consecutivos y almacenados en posiciones contiguas.

- **Acceso directo** al segmento donde se haya ubicado el registro buscado
- **Acceso secuencial** posteriormente, a los registros del segmento, hasta localizar el registro buscado o alcanzar el final del segmento, en caso de que no se halle.

Área de Índices

El valor del campo **clave del último registro** de un bloque o segmento

El segundo campo contiene la **dirección de comienzo**

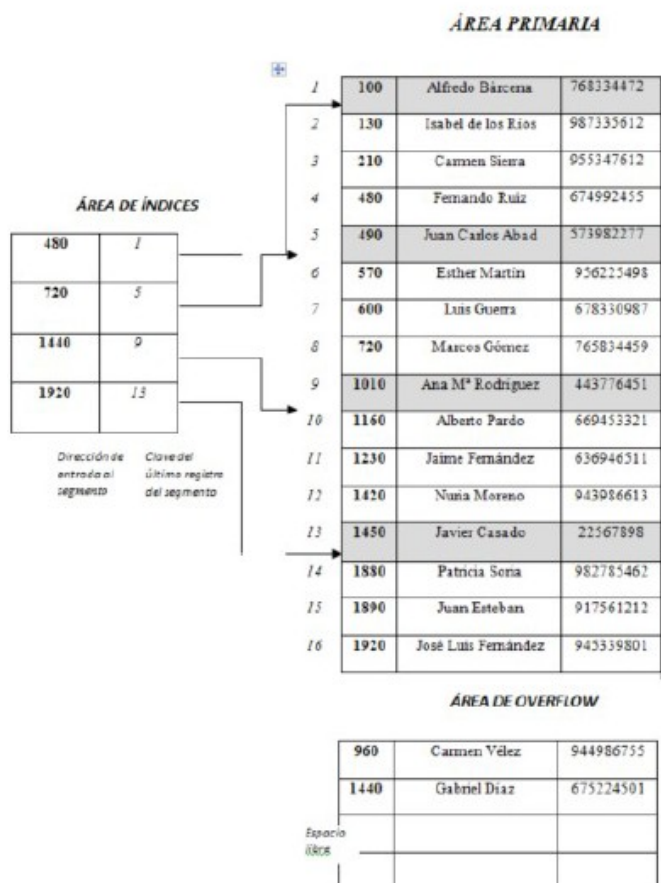
Área de Overflow: Es la zona destinada a contener los registros

almacenados posteriormente a la creación del fichero, por lo que no han sido incluidos en el área primaria.

El acceso a los registros se realiza mediante una consulta secuencial al área de índices

El borrado: la eliminación de los registros debe realizarse mediante marcas. Se generan huecos que realmente son posiciones de memoria ocupadas por registros marcados pero que no han sido eliminados

Esquema de un archivo con organización secuencial indexada



3. Bases de Datos

1. Introducción

Una base de datos se define como una estructura de datos interrelacionados a los que pueden acceder varios usuarios simultáneamente.

Las ventajas que aportan las bases de datos sobre los sistemas de ficheros son:

-Control sobre la redundancia de datos

-Consistencia de datos. Eliminando o controlando las redundancias

-Acceso múltiple. Diversos usuarios o aplicaciones podrán acceder a la base de datos, sin que existan problemas en el acceso o los datos.

-Utilización múltiple. *Un usuario solo accede a la parte que le corresponde*

-Confidencialidad y seguridad

2. Independencia de los datos

La descripción del nivel físico se realiza mediante un esquema interno, que es un conjunto de definiciones y reglas que permite definir las tablas y cómo se relacionan entre sí.

La descripción de este nivel se realiza mediante un esquema conceptual, que permite definir las entidades, los atributos y sus propiedades, las relaciones, operaciones de los usuarios y las restricciones y reglas de validación

Nivel Externo

O de usuario. Describe la base de datos como es percibida por los usuarios. *Cada usuario verá una base de datos (esquema externo) distinto según sea el nivel de acceso que se le haya concedido, ya que tendrá acceso a aquellos datos que necesite*

Con la arquitectura a tres niveles se introduce el concepto de **independencia de datos**. Se definen dos tipos de independencia:

Logica: Posibilidad de modificar un esquema conceptual de la base de datos sin tener que modificar esquemas externos ni programas. Por ejemplo: si se borra una entidad las vistas que no se refieran a ella no se alteran.

Independencia física: se refiere a la posibilidad de modificar el esquema interno sin tener que modificar ni el esquema conceptual ni los esquemas externos. Por ejemplo: se pueden reorganizar los archivos físicos o añadir nuevos archivos de datos para mejorar el rendimiento.

3. Modelos de Bases de Datos

En su **evolución** las bases de datos se han basado en 3 *modelos de datos*.

- Jerárquico.
- En red.
- Relacional.

En los 60 apareció el jerárquico sustituido por el modelo en red a finales de los 60 porque podía tener uno a muchos

El **modelo relacional** reemplazó al **modelo de red** porque permite representar la realidad de una forma más simple

1. Modelo Entidad-Relación

Objetos llamados entidades y relaciones entre estos

2. Modelo Jerárquico

Modo árbol

- Cada padre puede tener muchos hijos
- Cada hijo sólo tiene un padre

3. Modelo en red

permitía que los nodos tuvieran más de un solo padre.

En este modelo se pueden representar perfectamente cualquier tipo de relación entre los datos, pero hace muy complicado su manejo y es por ello que no ha tenido tanta fortuna como su predecesor.

4. Modelo Relacional

Su principal ventaja es que permite que el usuario y el diseñador operen en un entorno que se percibe como un **conjunto de tablas** y los detalles físicos complejos los maneja el sistema.

Características:

usa tablas

cada fila es un conjunto de campos y cada campo es un valor real

tablas independientes pero con vínculos

Ventajas:

Independencia estructural:

Simple

Fácil para administrar, diseñar ...

Consultas rápidas

Desventajas

Gasta muchos software y hardware

5. Modelo Orientado a Objeto

Se denomina así porque su estructura básica es un objeto, que recoge tanto datos como sus relaciones

El modelo de datos orientado a objetos está basado en los siguientes componentes

La estructura básica es un **objeto**: equivale a una entidad individual del modelo Entidad-Relación

Una **clase** es la definición del objeto. Esta definición explica los **métodos** y **atributos** que tiene. Las clases se organizan en una **jerarquía de clase**

Un método representa una acción del mundo real. Por ejemplo: localizar el nombre de un cliente

La herencia es la capacidad de un objeto de heredar los atributos y los métodos de los objetos

Ejemplo de representación de una factura

emitida a nuestros clientes.

FACTURA se representa incluyendo dentro del cuadro todos los atributos y relaciones con otros objetos.

En cada factura un cliente puede comprar varios artículos

4. Sistemas Gestores de Bases de Datos

El Sistema gestor de la base de datos (SGBD) es una aplicación que permite a los usuarios definir, crear y mantener la Base de datos

Los datos pueden estar compartidos entre distintas personas, entre distintas localidades geográficas **uso de los datos debe estar controlado Los datos se integran de forma lógica, eliminando las** redundancias y manteniendo la consistencia

1. Tipos

Modelo lógico en que se basan

Numero de usuarios al que da servicios al sistemas

.Monousuario

-Multiusuario

El tercer criterio se basa en el número de sitios en los que está distribuida la base de datos:

Cenralizados

Distribuidos

El cuarto es el coste

El quinto según el proposito

General

Especifico

Los SGBD son paquetes de software complejos que deben proporcionar una serie de servicios que

permiten almacenar y explotar los datos de forma eficiente.

Los componentes y funciones principales son:

1. **Lenguajes de los SGBD:** todos los SGBD ofrecen lenguajes apropiados para realizar la comunicación entre los SGBD y los usuarios. Estos lenguajes se clasifican en:

DDL (Lenguaje de definición de datos): se utiliza para definir el esquema de la base de datos: los objetos de la base de datos, las estructuras

DML (Lenguaje de manipulación de datos): se utiliza para consultar y actualizar los datos de la base de datos

El diccionario de datos: es una descripción de la base de datos con todos los objetos que la forman: las definiciones de los datos y sus relaciones

Seguridad, redundancia e integridad de los datos: son una serie de mecanismos que proporciona el SGBD para garantizar un acceso correcto, seguro y eficiente a los datos. Se hace mediante un componente software que se encarga de

Los usuarios del SGBD: existen distintos tipos de usuarios que acceden al sistema

Programadores Expertos Ocasionales Administradores

Arquitecturas de BD

La arquitectura de todo sistema de BD está muy condicionada también por las características del sistema informático sobre el que se ejecuta, y en especial por los siguientes aspectos:

La conexión en red de diferentes computadoras.

- El procesamiento paralelo de consultas dentro de una misma computadora.
- La distribución de los datos en diferentes computadoras, incluso alejadas entre sí.

1. Arquitectura Cliente/Servidor

El **servidor** permite llevar a cabo las funciones propias del SGBD, se puede decir que el servidor es en sí, el SGBD.

2. Un **cliente** es cada consumidor de recursos de la base de datos

2. BD Distribuidas

*Cuando una aplicación cliente accede a datos de distintos servidores, se denomina **Bases de Datos Distribuidas**.* En estos sistemas la base de datos está distribuida en más de un servidor. Los usuarios no tienen por qué conocer la ubicación física de los datos con los que trabajan y han de acceder simultáneamente a varios servidores.

Para el usuario (cliente) es indistinto si los orígenes de datos son únicos o múltiples, pero para el administrador es un trabajo extra considerable,

V: Mas rapido, menos costes, mas tolerante a fallos, independencia localidades

DV: Menos seguro, mas dificil de mantener, mas dificil controlar la recurrencia