

Evolución de los sistemas de almacenamiento de la información

- **Fichero manual**

Conjunto de carpetas etiquetadas cuyo contenido esta relacionado y se guardan en un armario o archivo. Es útil cuando el volumen de datos manejados no es muy grande y se puede extraer información con cierta facilidad, pero a medida que aumento y la información fue mas compleja fue necesario sustituirlo por otro informatizado

- **Sistemas de ficheros**

Los datos necesarios se guardaban en ficheros y las aplicaciones accedían a ellos para obtener los informes necesarios. Teníamos los ficheros necesarios para contener los datos con una estructura determinada y los programas de aplicación que accedían a estos datos. El lenguaje COBOL era el mas utilizado

- **Sistemas de bases de datos**

En ellos se almacenan todos los datos que necesita la empresa y los programas que los usen solo se tienen que preocupar por el almacenamiento físico de los mismos. Cualquier cambio no afectara a los programas que los usen.

Una base de datos es un conjunto de datos relacionados entre si, organizados y estructurados con información referente a algo. Son tratadas usando sistemas de gestión de bases de datos (SGBD) que proporcionan una serie de programas y utilidades que acceden y gestionan datos.

Ficheros

Para almacenar información de modo permanente se usan dispositivos de almacenamiento masivo, denominados memoria secundaria ya que los guardados en la memoria principal desaparecen al desconectar el ordenador. Esta información se estructura en ficheros, estructurados en:

- **Datos**. Hechos que necesitamos almacenar para obtener a partir de ellos una información. No están organizados de forma lógica y no tienen mucho significado.
- **Campo**. Carácter o conjunto de caracteres que tienen un significado específico utilizado para definir y guardar datos
- **Registro**. Conjunto de campos lógicamente relacionados que describen una persona o cosa.
- **Fichero**. Es el conjunto de registros.

| Campos: | FICHERO DE CLIENTES | | | | |
|-------------|---------------------|-------------|-----------------|-----------------------|-----------|
| | DNI | NOMBRE | APELLIDOS | DIRECCION | TELEFONO |
| Registro 1: | 73564765M | Javier | Barquín Arce | C/ Alta, 234 | 918342156 |
| Registro 2: | 56558765W | Luis | Gómez de Miguel | Avda. de Castilla, 2A | 956235567 |
| Registro 3: | 13874521M | Maria Belén | Márquez Ruiz | C/ Floranes, 2 | 568732212 |
| Registro 4: | 75675317R | Carmen | Rodríguez Mata | Paseo Pereda, 123 | 942665544 |

Los dispositivos de almacenamiento masivo se clasifican en:

- **Secuenciales**. La información se guarda en posiciones consecutivas y para acceder a un dato hay que recorrer los anteriores
- **Direccionales**. Permite el acceso directo a los datos.

El acceso a un registro es el procedimiento usado para seleccionarlo, condicionado por el tipo de soporte. Tipos:

- **Secuencial**. Los registros se leen uno tras otro desde el principio hasta localizarlo o el final. Se puede usar en dispositivos secuenciales y direccionales.

- **Directo.** Permite seleccionar un registro sin leer los anteriores, accediendo mediante su clave. Solo se puede usar en dispositivos direccionales.
- **Indexado.** Para seleccionar un registro consultamos la tabla que contiene la clave mas alta y la dirección de comienzo de cada bloque. Allí la lectura es secuencial hasta localizarlo. Solo se pueden usar dispositivos direccionales.
- **Dinámico.** Permite el acceso directo o por indice a un registro y a partir de ese a los demás de forma secuencial. Solo se puede usar en dispositivos direccionales.

Tipos de ficheros

- **Organización secuencial**

Los registros se localizan en posiciones físicamente contiguas sin dejar hueco y en el orden que hayan sido introducidos. Tienen correspondencia entre el orden lógico y el físico y se utilizan en dispositivos secuenciales y direccionales.

- **Ventajas**

- Rapidez en el acceso
- No es necesario realizar compactación de archivo
- No se desperdicia espacio
- Se pueden usar cualquier tipo de registros

- **Inconvenientes**

- Para acceder a un registro hay que recorrer los anteriores
- La adición de un registro se realiza al final
- No se pueden eliminar registros, se tienen que marcar
- Para mantenerlo ordenado y compactado hay que crear un nuevo fichero a partir del existente

Es recomendable para ficheros con indice de utilización muy elevado y que sean estables. Sin útiles cuando en cada operación se van a procesar la mayoría de los registros.

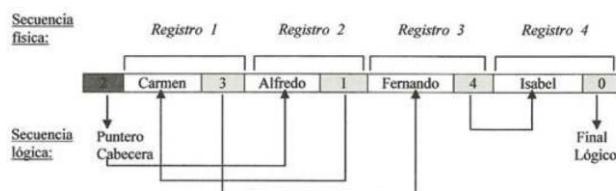
| Dirección de memoria | Marca de borrado | Nombre | Apellidos | Teléfono |
|----------------------|------------------|-------------|-------------|-----------|
| 1200 | | Alfredo | Bárcena | 768334472 |
| 1300 | * | Isabel | De los Ríos | 987335612 |
| 1400 | | Carmen | Sierra | 955347612 |
| 1500 | | Fernando | Ruiz | 674992455 |
| 1600 | | Juan Carlos | Abad | 573982277 |

- **Organización secuencial encadenada**

Ficheros de organización secuencial gestionados mediante punteros que permiten tenerlos ordenados y con un orden lógico diferente al físico.

- Su consulta es secuencial
- La adición se realiza al final
- La eliminación se realiza modificando el puntero anterior
- Para modificar un registro se reestrenó sobre la información anterior

Registros encadenados con punteros



- **Organización secuencial indexada**

Los registros con los datos se graban en un fichero manualmente, pero se recuperan con acceso directo gracias a la utilización de un fichero adicional, de indices. Tienen tres áreas:

- **Primaria.** Se escriben los registros cuando se crea el fichero. Contiene los registros ordenados ascendente por el valor de su clave. Esta dividida en segmentos y cada uno de ellos almacena registros consecutivos y almacenados en posiciones contiguas. El acceso se realiza con una doble operación:
 - Acceso directo al segmento
 - Acceso secuencial a los registros del segmento
- **Índices.** Es creada por el sistema, en ella los registros están formados por campos.
 - Calor del campo clave del último registro de un bloque
 - El segundo campo contiene la dirección del comienzo de los segmentos.
- **Overflow.** Destinada a contener registros almacenados posteriormente.

El acceso se realiza mediante consulta secuencial al área de índices y con el valor del segmento seleccionado se recorren los registros del segmento.

El borrado mediante marcas.

La inserción en el área de overflow.

Cuando los registros borrados son muchos o las cadenas de desbordamientos son largas su utilización deja de ser eficiente, por lo que es necesario reorganizar el archivo.

Es utilizada en procesos con pocos registros y que manejan el fichero completo.

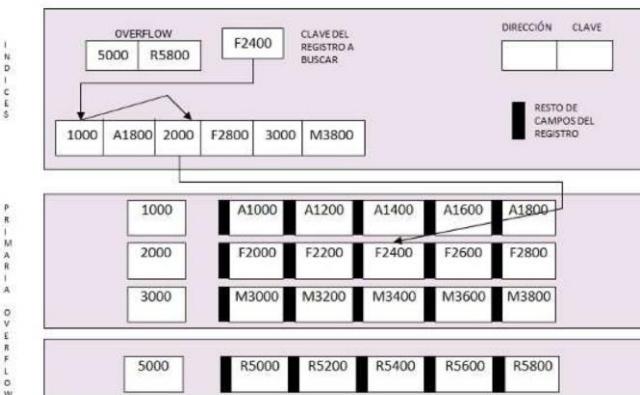
- **Organización indexada-encadenada**

Se utilizan punteros e índices de forma simultánea que proporciona gran rapidez en la búsqueda de registros.

Su estructura es igual que la indexada, añadiendo punteros en los registros de forma primaria, por lo que se consigue mejorar los tiempos de búsqueda y mantener la organización lógica.

- Para eliminar registros se marcan.
- Adiciones en la zona de overflow.
- Para acceder a un editor se busca en el área de índices la dirección de inicio del bloque de registros al que pertenece el buscado.
- Para consultar todo el fichero, el último registro de cada bloque en el área de overflow tiene un puntero al primer registro del área siguiente.
- Los nuevos registros se insertan y quedan enlazados mediante punteros conservando en orden lógico que marca la clave.

- De los indices y punteros se encarga el Sistema Operativo.



- Organización relativa**

Se basa en la independencia entre el orden y la posición de los registros. El almacenamiento físico se realiza mediante el empleo de una clave que relaciona la posición del registro y la posición de memoria.

Hay dos tipos:

- **Directa.** Se emplean claves numéricas, por lo que la secuencia lógica coincide con física. No se puede almacenar un registro por encima de los límites máximos del fichero.

- **Ventajas**

- Acceso directo a los registros
 - Permite leer y escribir simultáneamente
 - Acceso a datos de forma directa y secuencial

- **Inconvenientes**

- Para realizar un acceso secuencial hay que recorrer todas las direcciones
 - Deja gran cantidad de huecos
 - Se producen colisiones entre registros con igual clave

| Direcciones de memoria | Clave | Datos | Espacios o huecos libres |
|------------------------|-------|-----------|--------------------------|
| 1 | | | |
| 2 | 2 | Cliente A | |
| 3 | | | |
| 4 | 4 | Cliente B | |
| 5 | 5 | Cliente C | |
| 6 | | | |
| 7 | | | |
| 8 | 8 | Cliente D | |
| 9 | 9 | Cliente E | |
| 10 | | | |

- **Aleatoria**

Organización relativa con clave alfanumérica o numérica que se transforma obteniendo un valor numérico entero que facilite la correspondencia directa entre la clave y la dirección. El valor de la clave está en relación con la capacidad máxima del soporte físico. No se pueden almacenar registros con dirección mayor a los límites del fichero.

- **Ventajas:**

- Acceso inmediato a los registros mediante su clave

- No es necesario ordenar el fichero
- Se pueden realizar operaciones de escritura y lectura a la vez
- Son muy rápidos en el tratamiento individual de registros
- Se pueden realizar accesos secuenciales
- **Inconvenientes**
 - Consultas sobre todo el fichero son muy lentas
 - Fichero con gran cantidad de espacio
 - El algoritmo para la conversión de las claves y el necesario para el almacenamiento de sinónimos han de ser creados de modo que deben el menor número de huecos libres y menor número de sinónimos.

| Direcciones de memoria | Clave | Datos |
|------------------------|-------|-----------|
| 1 | | |
| 2 | AB85 | Cliente C |
| 3 | | |
| 4 | DG49 | Cliente A |
| 5 | EH23 | Cliente D |
| 6 | | |
| 7 | | |
| 8 | BS12 | Cliente F |
| 9 | KL92 | Cliente E |
| 10 | | |
| | FB43 | Cliente B |
| | | |
| | | |

Espacios o huecos libres

Zona de overflow o sinónimos

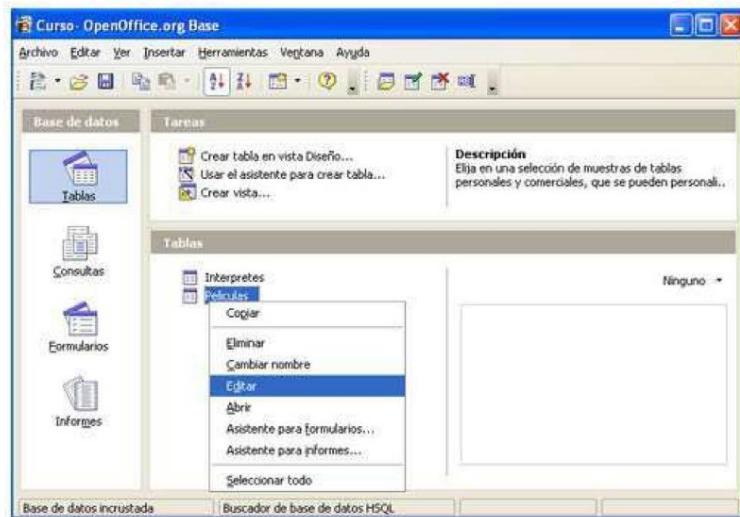
Bases de datos

- **Ventajas**
 - Control sobre la redundancia de datos
 - Consistencia de datos
 - Más información sobre la misma cantidad de datos
 - Compartición de dato
 - Se simplifica el esfuerzo de programación
 - Mantenimiento de estándares
- **Conceptos**
 - **Base de datos**
Conjunto de datos relacionados entre sí, organizados y estructurados con información referente a algo

| Id_genero | Nombre | Id_pelicula | Título | Director | Año | Formato | Visionada |
|-----------|-----------------|-------------|-----------------------------------------|---------------------|----------|---------|-----------|
| 1 | Ciencia-Ficción | 1 | Blade Runner | Ridley Scott | 01/01/83 | DVD | ✓ |
| 2 | Aventuras | 2 | La Guerra de las Galaxias | George Lucas | 01/01/77 | VHS | ✓ |
| 3 | Historico | 3 | Indiana Jones en Busca del Arca Perdida | Steven Spielberg | 01/01/81 | DVD | ✓ |
| 4 | Comedia | 4 | Millon Dolar Baby | Clint Eastwood | 01/01/04 | DVD | |
| 5 | Drama | 5 | Le Comunidad del Anillo | Peter Jackson | 01/01/01 | DVD | ✓ |
| 6 | Thriller | 6 | La Belleza Robada | Bernardo Bertolucci | 01/01/96 | VHS | ✓ |
| 7 | Suspense | 7 | Gladiator | Ridley Scott | 01/01/00 | DVD | ✓ |
| 8 | Terror | 8 | Rocky | John G. Avildsen | 01/01/76 | VHS | ✓ |
| 9 | Fantasia | 9 | Los Lunes al Sol | Fernando León | 01/01/02 | DVD | ✓ |
| | | 10 | Mar Adentro | Alejandro Amenábar | 01/01/04 | DVD | |

- Sistema gestor de bases de datos

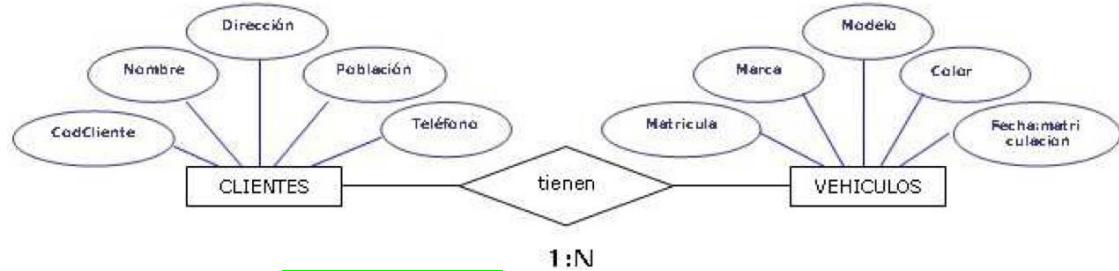
Aplicación que permite a los usuarios definir, crear y mantener la Base



de datos.

- **Para que sea efectivo debe cumplir:**
 - Los datos deben estar compartidos entre distintas personas
 - El uso de datos debe de estar controlado
 - Los datos se integran de forma lógica, eliminando las redundancias y manteniendo la consistencia.
- **Arquitectura de las bases de datos.**
 - **Nivel físico o interno.** Como se almacenan físicamente las estructuras de datos en el ordenador. Los usuarios se trabajan a este nivel son los administradores.
 - **Nivel conceptual.** Describe la organización de los datos en la base de datos y las relaciones existentes entre ellos. Implica hacer análisis de las necesidades información de los usuarios y definir las clases de datos para satisfacer esas necesidades. Intervienen los programadores.
 - Modelo entidad relación
 - Normalizar
 - Modelo racional
 - **Nivel externo.** Describe la base de datos como es percibida por los usuarios, cuyos conocimientos de informática no tienen por qué ser grandes y hay que ocultarles la complejidad interna de la base de datos. Con esta estructura hay independencia de datos:
 - Independencia lógica. Modificar un esquema conceptual sin modificar los esquemas externos
 - Independencia física. Posibilidad de modificar el esquema interno sin hacerlo en el conceptual ni los externos.
- **Modelos de bases de datos**
- **Modelar.** consiste en definir un mundo abstracto donde las conclusiones que se sacan de él coinciden con las manifestaciones del mundo real.

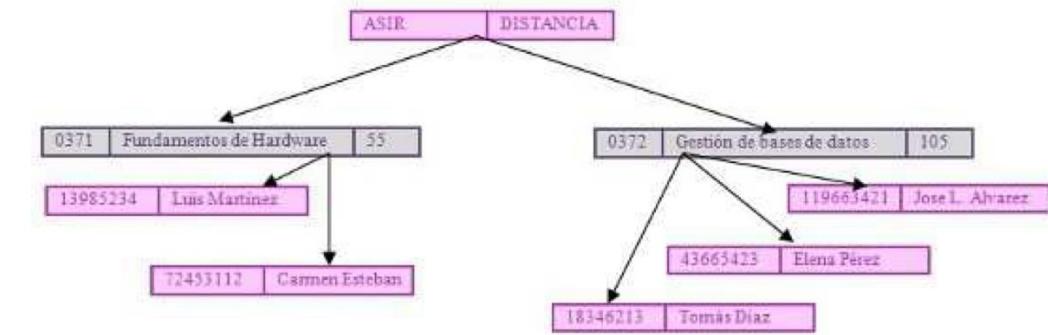
- **Modelo entidad relación**



- **Modelo jerárquico**

| MODULOS | CODIGO | NOMBRE | HORAS |
|----------------|---------------|---------------|--------------|
| | | | |

| MODULOS | CODIGO | NOMBRE | HORAS |
|----------------|---------------|---------------|--------------|
| | | | |



- **Modelo en red**

- **Modelo racional (tablas)**. Permite que usuario y diseñador operen en un entorno percibido como un conjunto de tablas y los detalles físicos los maneje el sistema.

- **Características**

- Representa todos los datos y relaciones
- Relaciona y representa una tabla que no es un conjunto de filas. Cada fila es un conjunto de campos y cada campo representa un valor que describe el mundo real.
- Tablas independientes pero relacionadas
- Proporciona redundancia y consistencia
- Independencia de datos de las aplicaciones y dispositivo de almacenamiento.

- **Ventajas**

- Independencia estructural
- Simplicidad
- Facilidad para diseñar, administrar y utilizar la base de datos
- Capacidad para hacer consultas rápidas y sencillas
- Un SGBD relacional incluye elementos de software que realizan mas tareas y mas complejas para los usuarios y diseñadores

- **Inconvenientes**

- A medida que la base de datos crece, si el sistema es inapropiado es mas lento y con mas anomalías.

| CLIENTES | | | | |
|------------|--------------------|--------------------|-------------|-----------|
| CodCliente | Nombre | Dirección | Población | Teléfono |
| 1 | Francisco Álvarez | C/ La Mata 9. | Alcántara | 925767788 |
| 2 | Raquel Marcos | C/ La Amapola 7. | Toledo | 925998811 |
| 3 | C. Los Revueltas | C/ La Arboleda 12. | Madrid | 919090771 |
| 4 | José María Cabello | C/ La plaza 72 | Madrid | 914409071 |
| 5 | Jorge Peña | C/ Fresneda, 23 | Guadalajara | 949788896 |
| 6 | Dolores Manzano | C/ Autonomía, 8 | Madrid | 916767560 |

| VEHICULOS | | | | | |
|-----------|---------|--------|------------|---------------------|------------|
| Matricula | Marca | Modelo | Color | Fecha Matriculación | CodCliente |
| 4534 FNG | Ford | Focus | Negro | 14/04/2007 | 2 |
| 1203 CLL | Citroën | C4 | Magenta | 23/08/2005 | 5 |
| 3367 GHB | Suzuki | Vitara | Cobalto | 16/05/2009 | 2 |
| 1004 JLG | Kia | Rio | Rojo | 02/07/2009 | 6 |
| 6709 BFG | Peugeot | 206 | Gris plata | 12/10/2006 | 3 |

```

classDiagram
    class CLIENTES {
        CodCliente
        Nombre
        Direccion
        Poblacion
        Telefono
    }
    class VEHICULOS {
        Matricula
        Marca
        Modelo
        Color
        FechaMatriculacion
        CodCliente
    }
    CLIENTES "1" -- "∞" VEHICULOS : Vehiculos
  
```

■ **Modelo orientado a objetos**

Sistemas gestores de bases de datos

- **Lenguajes de los SGBD**
 - **LDD(Lenguaje de definición de datos)**. Crear tablas e indices. Se usa para definir el esquema conceptual e interno
 - **LMD(Lenguaje de manipulación de datos)**. Se usa para consultar y actualizar los datos
- **Seguridad e integridad de los datos**
 - Garantizar el acceso solo a usuarios autorizados
 - Realizar copias de seguridad y restauración
- **Arquitectura cliente servidor**
 - **SGBD centralizados** (un solo ordenador)
 - **SGBD distribuidos** (varios ordenadores)