

Tema 4 - Sistema de archivos. Introducción a las bases de datos

Archivo \Rightarrow Conjunto de información sobre un mismo tema organizada de forma estructurada para la búsqueda de un dato individual. Se compone de registros.

Los registros se pueden dividir en campos.

Estructura dentro del archivo que contiene información correspondiente a un elemento individual.

Todo archivo tiene atributos como nombre, tamaño, permisos, propietario, etc. El SO proporciona servicios para operar sobre archivos: abrir, cerrar, crear, copiar, borrar, etc.

Clasificación

1. Longitud fija. Todos los registros presentan mismo tamaño.
2. Longitud variable. Con delimitador: El final de cada registro presenta un carácter especial que indica su final. Con cabecera: Existe un campo inicial que almacena el n.º de bytes del registro.
3. Longitud indefinida. El archivo carece de estructura interna y en cada lectura/escritura se transfiere una subcadena del archivo, siendo el programa de usuario el que indique el fin del registro al SO.

Directorio \Rightarrow Archivo especial que permite agrupar archivos según las necesidades de los usuarios.

Organización de archivos

1º) Secuencial

Los registros se almacenan físicamente de forma contigua. Es barata, sencilla de usar y permite aprovechar el espacio.

-) Añadir \Rightarrow Siempre se añaden registros al final, después del último.
-) Consulta o recuperación \Rightarrow Para leer el registro de la posición n , hay que leer los $n-1$ anteriores.
-) Inserción, modificación y eliminación \Rightarrow Solo se pueden modificar registros si no se aumenta su longitud. No se pueden borrar registros, por eso se hace borrado lógico.

2º) Secuencial encadenada

Cada registro se almacena un puntero con la dirección física del registro que le sigue. El último registro presenta una marca especial que indica que ya no hay más registros. Aquí es más fácil la inserción y el borrado pero limita las consultas de forma secuencial.

-) **Consulta** \Rightarrow Igual que en el caso anterior
-) **Inserción** \Rightarrow 1°) Se localiza entre qué 2 registros se inserta.
2°) Escribir el registro en una zona libre de usuarios. 3°) El puntero del nuevo registro apunta al siguiente y el puntero del anterior pasa a apuntar al nuevo.
-) **Borrado** \Rightarrow Borrado lógico (el puntero del anterior pasa a apuntar al siguiente del que se quiere borrar).
-) **Modificación** \Rightarrow Es posible si no hay aumento de longitud. Si no, se inserta el registro y se borra la versión anterior.

3°) Secuencial indexada

Organización con dos estructuras:

Zona de registros \Rightarrow Zona donde se direccionan los registros del archivo dividida en tramos en los que los registros están ordenados según una llave.

Zona de índices \Rightarrow Registros que contienen el mayor valor de la llave de un tramo y la dirección del primer registro del tramo.

Ejemplo

Ana	1
J.A.	3
Soledad	5

1 Aarón } 1er tramo
2 Ana }
3 llanolo } 2° tramo
4 J.A. }
5 Soledad } 3er tramo

-) **Consulta** \Rightarrow Se realiza por llave. 1°) Se busca la llave inmediatamente mayor que la del registro buscado.
2°) Se obtiene la 1ª dirección del tramo cuya llave cumple lo anterior. 3°) Se leen los registros del tramo hasta llegar al buscado o uno con llave mayor al buscado si este no existe.
-) **Inserción** \Rightarrow Como ambas zonas son secuenciales, no es posible insertar un registro.
-) **Borrado** \Rightarrow No se pueden borrar por la misma razón que antes, solo se puede hacer borrado lógico.
-) **Modificación** \Rightarrow Es posible si no se modifica la llave del registro ni se aumenta su tamaño.

4°) Directa o aleatoria

Existe una función de transformación que genera la dirección de cada registro en el archivo a partir de un campo que se usa como llave. Dicha función debe escogerse adecuadamente ya que se pueden dar situaciones no deseadas:

- Direcciones que no se corresponden con ninguna llave.
- Direcciones que se correspondan con más de una llave, dando lugar a colisiones.

Soluciones:

- [->] Al asociar una llave a una dirección ya ocupada por otro registro, se busca en el archivo una posición libre donde escribir el registro.
- [->] La existencia de una zona de desbordamiento donde estarán los registros que han dado problemas.

1) **Direccionamiento directo** => La llave se usa como dirección (solo si es numérica y su rango de valores no es mayor que el de las direcciones).

2) **Direccionamiento asociado** => Se usa una tabla que almacena la correspondencia registro-llave.

3) **Direccionamiento calculado (hashing)** => La dirección del registro se obtiene transformando la llave. Si es numérica y toma valores fuera del rango adecuado, actúa el hashing. Si no es numérica, se realiza una conversión (por ejemplo, de char a int usando la tabla ASCII).

•) **Consulta** => Se consulta por llave, de forma que la función de transformación devuelve la dirección del registro correspondiente.

•) **Borrado** => Siempre es lógico y se puede reutilizar el espacio

•) **Modificación e inserción** => Se puede hacer realizando la transformación de la llave correspondiente.

Los sistemas de archivos presentan problemas como la dificultad de mantenimiento, redundancia (duplicación de archivos), rigidez de búsqueda (siempre se accede de la misma forma), falta de seguridad, etc. De ahí la necesidad de las bases de datos.

Base de datos

Sistema formado por un conjunto de datos y un paquete software para la gestión de esos datos. Permite acceder a datos de formas diversas, los datos ya no dependen de los programas y se controlan las redundancias.

Requisitos de toda base de datos

1. Acceso múltiple
2. Utilización múltiple
3. Flexibilidad
4. Seguridad
5. Protección contra fallos
6. Independencia física
7. Independencia lógica
8. Redundancia controlada
9. Interfaz de alto nivel
10. Interrogación directa ("query")

Modelo entidad relación

Rectángulos \rightarrow Representan entidades.

Rowbos \rightarrow Representan relaciones

Elipses \rightarrow Representan atributos de las entidades o relaciones.

Representación de la cardinalidad \rightarrow La forma más segura es sin flechas, es decir, con una etiqueta asociada al rowbo y que ponga $1:1$, $1:N$, $N:1$ ó $N:M$.

Para pasar el diagrama ER a tablas relacionales hemos de seguir el siguiente esquema:

1º) Para cada entidad se configura una tabla con tantas columnas como atributos tenga.

2º) En cuanto a las relaciones:

* Si tiene atributos \Rightarrow Se crea una tabla para dicha relación de forma que tenga en sus columnas sus respectivos atributos y las claves de las entidades involucradas.

* Si no tiene atributos:

Tipo $1:1$ \Rightarrow En la tabla de la entidad que se considere más importante se añade la clave de la otra entidad.

Tipo $1:N$ ó $N:1$ \Rightarrow En la tabla de la entidad de "N" se añade la clave de la otra entidad.

Tipo $N:M$ \Rightarrow Se configura una tabla para la relación que tiene en sus columnas las claves de las entidades involucradas.