

WUOLAH



adrespi_

www.wuolah.com/student/adrespi_



652

FS - Tema 4_ Sistemas de archivos.pdf

Resumen Tema 4



1º Fundamentos del Software



Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada

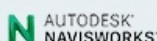
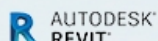
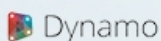


Escuela de **LÍDERES**

Master BIM Management



60 Créditos ECTS



Jose María Girela
Bim Manager.



TEMA 4

Sistemas de archivos. Introducción a las bases de datos.

4.1 CONCEPTO DE ARCHIVO Y DIRECTORIO

Archivo: conjunto de información tratada como una unidad de almacenamiento y organizada de forma estructurada para la búsqueda de un dato individual

Registros: estructuras o unidades que forman el archivo y que contiene la información correspondiente a cada elemento individual

Campo: es un dato que forma parte de un registro y representa una información unitaria o independiente

El SO permite aludir a un archivo mediante un nombre, independientemente de cómo se encuentre almacenado en el dispositivo, y proporciona operaciones como:

- Crear / copiar / borrar / renombrar un archivo
- Establecer / consultar atributos de un archivo
- Abrir / cerrar un archivo
- Leer / escribir un registro de un determinado archivo

Tipos de archivos según el tipo de registros:

- Longitud fija
- Longitud variable
 - Con delimitador: un carácter marca el fin de un registro
 - Con cabecera: cada registro contiene un campo inicial con el número de bytes del registro
- Indefinido : el archivo no tiene estructura alguna, por lo que es el programa de usuario el encargado de localizar el principio y final de cada registro

Directorio: estructura jerárquica como agrupación de archivos u otros directorios para facilitar la visión del conjunto de archivos al usuario. En dicha visión se utilizan los conceptos de:

- Directorio actual o de trabajo
- Directorio inicial o raíz
- Rutas absolutas y relativas
- Lista de búsqueda
- Enlace duro y simbólico

4.2 ORGANIZACIÓN DE ARCHIVOS

1. Organización secuencial

Los registros están almacenados físicamente contiguos.

Operaciones

- Añadir: solo es posible escribir al final del archivo. La información se graba escribiendo los registros en el orden en que se desea
- Consulta: se realiza en orden secuencial, es decir, para leer el registro en la posición n es necesario leer los $n-1$ elementos anteriores
- Inserción, modificación y borrado: no es fácil realizar estas operaciones, pudiendo ser necesario crear un nuevo archivo con dichas actualizaciones.

Adecuada cuando

- Los registros se recorren en la misma secuencia en que están almacenados
- Se procesarán la mayor parte de los registros del archivo

Ventajas

- Buen aprovechamiento del espacio
- Sencilla de utilizar

2. Organización secuencial encadenada

Junto a cada registro se almacena un puntero con la dirección del siguiente registro

Operaciones

- Consulta: la consulta es secuencial, siguiendo la secuencia lógica del archivo
- Inserción:
 - Localizar la posición donde se desea insertar, es decir, entre qué dos registros
 - Escribir el registro en una zona libre y asignar al puntero la dirección del siguiente registro
 - Modificar el registro anterior de forma que el puntero contenga la dirección del nuevo registro
- Borrado: basta con actualizar el puntero del registro anterior para que apunte al siguiente del que se quiere borrar
- Modificación: si el cambio no implica un aumento de longitud puede reescribirse en el mismo espacio. En otro caso se debe insertar el registro y borrar el antiguo

3. Organización secuencial indexada

El archivo está formado por dos estructuras o zonas:

- Zona de registros: se encuentran los registros ordenados por el valor de una llave, está dividida en diferentes tramos y permite direccionar los registros
- Zona de índices: hay una por cada uno de los tramos de la zona de registros y contiene el valor de la llave del último registro del tramo y la dirección del primer registro del tramo

Operaciones

- Consulta: permite realizar consultas por llave sin necesidad de leer los registros que le preceden. Pasos:
 - Leer en secuencia las llaves de la zona de índices hasta encontrar una mayor o igual
 - Obtener la dirección de comienzo del tramo donde está el registro
 - Leer secuencialmente la zona de registros a partir de la dirección obtenida hasta encontrar un registro con valor de la llave igual o mayor que el buscado (si es el caso, significa que no se encuentra en el archivo)
- Inserción: no es posible insertar un registro dado que ambas zonas son secuenciales
- Borrado: la única forma de eliminar un registro es marcándolo (borrado lógico)
- Modificación: sólo es posible si el registro no aumenta de longitud al ser modificado y no cambia el valor de su clave

4. Organización directa o aleatoria

Es un archivo escrito sobre un soporte de acceso directo para el cual existe una transformación conocida que genera la dirección de cada registro en el archivo a partir de un campo usado como llave. El principal problema es escoger adecuadamente la transformación o método de direccionamiento a utilizar.

Situaciones no deseadas

- Hay direcciones que no se corresponden con ninguna llave, quedando zonas de disco sin utilizar
- Hay direcciones que corresponden a más de una llave. En tal caso, se dice que las llaves son sinónimas. Para resolver este problema, se busca en el archivo una posición libre donde escribir el registro. Además, se reserva una zona de desbordamientos donde se escribirán los registros que no se pueden escribir en la posición que les corresponde.

Métodos de direccionamiento

- Direccionamiento directo: se usa como dirección la propia clave, siempre y cuando la llave sea numérica y su rango de valores no es mayor que el rango de direcciones.
- Direccionamiento asociado: se construye una tabla en la que se almacena para cada llave la dirección donde se encuentra el registro correspondiente
- Direccionamiento calculado (hashing): la dirección se obtiene realizando una transformación sobre la llave. Se usa cuando la llave no es numérica o cuando sí lo es pero toma valores en un rango inadecuado para usarse directamente como dirección.

Operaciones

- Consulta: se realiza por llave. Se le aplica a la llave el algoritmo de transformación y se lee el registro en la dirección resultante.
- Borrado: se realiza un borrado lógico, pudiéndose reutilizar el espacio del registro eliminado
- Modificación e inserción: siempre es posible, realizando la transformación de llave correspondiente

4.3 BASES DE DATOS

Problemas en una aplicación convencional

- Dificultad de mantenimiento: normalmente es necesario actualizar varios archivos con diferentes organizaciones. Si dicha actualización no se realiza correctamente resulta en información incoherente
- Redundancia: se da cuando un dato se puede deducir a partir de otros datos o el mismo dato se encuentra almacenado varias veces
- Rigidez de búsqueda: el archivo se concibe para acceder a los datos pero en la mayoría de los casos es necesario combinar acceso secuencial y directo por varias claves
- Dependencia con los programas: en un archivo no se refleja la relación entre los campos y registros, sino que es el programa el que trabaja con el archivo y determina dichas relaciones. Si se produce un cambio en la estructura del archivo habrá que modificar todos los programas que lo usen
- Seguridad: uno de los mayores problemas de cualquier sistema de información es mantener la seguridad necesaria sobre los datos que contiene. Si se está trabajando con archivos el control deberá realizarlo el propio programa

Base de datos

Surge como alternativa a los sistemas de archivos, intentando eliminar o al menos reducir sus inconvenientes. Se trata de un sistema formado por un conjunto de datos y un paquete software que gestiona dicho conjunto de datos de tal modo que:

- Se controla el almacenamiento de datos redundantes
- Los datos resultan independientes de los programas que los usan
- Se almacenan las relaciones entre los datos
- Se puede acceder a los datos de diversas formas

Requisitos para un buen sistema de base de datos

- **Acceso múltiple:** diversos usuarios pueden acceder a la base de datos sin provocar conflicto ni versiones incoherentes
- **Utilización múltiple:** cada usuario podrá tener una imagen o visión particular de la estructura de la base de datos
- **Flexibilidad:** se podrán usar distintos métodos de acceso sin grandes tiempos de respuesta

- **Seguridad:** se controlará el acceso a los datos
- **Protección contra fallos:** deben existir mecanismos de recuperación en caso de fallo de la computadora
- **Independencia física:** se puede cambiar el soporte físico sin que repercuta en la base de datos ni en los programas que la usan
- **Independencia lógica:** se pueden modificar los datos, las relaciones entre ellos o incluir nuevos datos sin afectar a los programas que la usan
- **Redundancia controlada:** los datos se almacenan una sola vez
- **Interrogación directa (query):** permite el acceso a los datos de forma interactiva o conversacional

Estructura de una base de datos

En una base de datos se almacena una serie de datos (**atributos**) sobre un conjunto de objetos o elementos (llamados **entidades**).

La información de cada entidad es almacenada en **registros**, siendo cada atributo un **campo** de dicho registro. Hay diferentes tipos de registros (uno por entidad), por lo que se reserva el nombre registro para especificar un tipo de registro y elemento, valor actual de registro u ocurrencia de registro para denominar a cada una de las apariciones de dicho registro en la base de datos.

Diremos que un conjunto de atributos de una entidad es un **identificador** o **clave primaria** de dicha entidad si el valor de dicho atributos determina de forma unívoca los elementos de dicha entidad y no existe ningún subconjunto del mismo que sea identificador de la entidad.

En una base de datos también se almacenan las relaciones entre las entidades que la forman. Se explicitan con punteros que son insertados automáticamente por el SGBD, es decir, de forma transparente al usuario.

Tipos de relaciones

Dadas dos entidades A y B, las relaciones entre éstas pueden ser:

- **1 a 1:** a cada elemento de A le corresponde un único elemento en B y viceversa
- **1 a muchos:** a cada elemento de A le corresponde uno o más elementos de B y a cada elemento de B le corresponde uno solo de A
- **Muchos a muchos:** a cada elemento de A le pueden corresponder varios elementos de B y viceversa

El esquema de una base de datos es la definición de la estructura lógica. Para que un programa concreto pueda acceder a una base de datos es necesario especificar la estructura lógica de la parte que éste va a usar.

La descripción de la estructura lógica de una parte de la base de datos a utilizar por uno o más programas recibe el nombre de vista o subesquema. Éste permite que varios usuarios utilicen distintas visiones de una misma base de datos, además de limitar el acceso a distintas partes de la base de datos

Tipos de bases de datos

- **Bases de datos jerárquicas:** se establece una relación jerárquica entre los datos en forma de árbol y no es posible definir relaciones muchos a muchos
- **Bases de datos en red:** modelo similar al jerárquico en donde pueden darse también relaciones muchos a muchos
- **Bases de datos relacionales:** formada por tablas, entendiendo tabla como una estructura bidimensional formada por una sucesión de registros del mismo tipo. Si se imponen condiciones a las tablas se pueden tratar como relaciones matemáticas.
 - Los registros son del mismo tipo
 - No aparecen campos repetidos
 - No existen registros duplicados
 - El orden de los registros en la tabla es indiferente
 - En cada tabla hay una llave formada por una o varios campos

4.4 SISTEMA DE GESTIÓN DE BASE DE DATOS

Se denomina SGBD al conjunto de software destinado a la creación, control y manipulación de la información de una base de datos. Actúa como intermediario entre los programas de aplicación y el sistema operativo, lo que permite que los programas sean independientes de la estructura física de los datos. Permite la realización de las siguientes tareas:

- Acceso a los datos desde un lenguaje de alto nivel
- Recuperación de la información directa en modo conversacional
- Definición del esquema de la base de datos y los subesquemas
- Organización física de la base de datos y recuperación tras fallos del sistema

Las tres primeras funciones se realizan mediante dos lenguajes específicos:

- **Lenguaje de manipulación de datos (DML):** se utiliza para el acceso a la base de datos desde lenguajes de alto nivel en modo conversacional
- **Lenguaje de descripción de datos (DDL):** se usa para la descripción del esquema y de los subesquemas