

Unidad 3:

Gestión de Archivos

Tema 5, Sistema de Archivos:

- 5.1 Introducción: Requisitos para el almacenamiento de información.
- 5.2 Archivo: Concepto y operaciones con archivos.
- 5.3 Métodos de Acceso: Acceso secuencial y acceso directo.
- 5.4 Estructura de Directorios: Directorios de varios niveles, árbol y grafo.

5.1 Requisitos para el almacenamiento de información.

- Introducción (Sistema de Archivos):
 - Cada proceso necesita almacenar gran cantidad de información en su espacio de direcciones virtuales:
 - Problemas:
 - Tamaño escaso para muchas aplicaciones.
 - Información condicionada por el proceso.
 - Información no compartible entre procesos.
 - Soluciones: Información independiente de los procesos que la manipulen.

5.1 Requisitos para el almacenamiento de información.

- Introducción (Requisitos para almacenar información):
 - Capacidad de almacenamiento de gran cantidad de información a largo plazo.
 - Información superviviente al proceso que la utiliza (no volatilidad).
 - Varios procesos debe poder acceder concurrentemente a la información.

- Variedad de medios donde almacenar información: discos magnéticos, cintas magnéticas, discos ópticos, etc...
 - Ventaja: Permite elegir el medio más adecuado en función de las necesidades particulares: cantidad de información a almacenar, velocidad de acceso, fiabilidad, etc...
 - Desventaja: Requiere conocer las particularidades de cada medio.

5.2 Archivo: Concepto y operaciones con archivos.

- Concepto de archivo:
 - Necesidad del uso de archivos:
 - SO abstrae las propiedades físicas de sus dispositivos de almacen.
 - ¿Cómo?
 - Definiendo una unidad de almacenamiento lógica, el ARCHIVO.
 - **Archivo:** *Unidad de almacenamiento lógica (se graba en almacenamiento secundario) que para el usuario no constituye más que un conjunto de datos relacionados con un nombre asociado.*
 - Es una **abstracción del almacenamiento secundario** (tipo abstracto de datos):
 - Oculta la realidad física del almacenamiento (cintas, discos, etc.).
 - Organiza la información.

5.2 Archivo: Concepto y operaciones con archivos.

- Atributos de un archivo:
 - **Nombre:** Simbólico, relacionado con su contenido, debe ser comprensible e identificable por el usuario.
 - **Tipo de archivo:** Necesario en SO que reconocen distintos tipos.
 - **Ubicación:** Posición en el dispositivo.
 - **Tamaño:** El del archivo.
 - **Protección:** Información del control de acceso.
 - **Hora, Fecha e Identificación del usuario:** Información relativa a la creación, última modificación, último uso, ...

5.2 Archivo: Concepto y operaciones con archivos.

- Operaciones con archivos:

- **6 Operaciones básicas:**

- Creación de un archivo.
 - Eliminar un archivo.
 - Escritura de un archivo.
 - Lectura de un archivo.
 - Posicionarse dentro del archivo (por posición o por clave).
 - Truncar un archivo..

5.2 Archivo: Concepto y operaciones con archivos.

- Operaciones con archivos:
 - **Crear un archivo:**
 - 1º Encontrar espacio para él en el sistema de archivos.
 - 2º Insertar una entrada para el nuevo archivo en el directorio (esta entrada registra el nombre del archivo y su ubicación en el sistema de archivos).
 - **Eliminar un archivo:**
 - Libera todo el espacio que el archivo ocupa.
 - Se borra la entrada del directorio.
 - **Escritura de un archivo:**
 - Especificar el nombre del archivo y la información que se escribirá en él.
 - Debemos mantener un puntero de escritura (que se irá actualizando cada vez que se escriba en el archivo) que indique en qué posición del archivo se efectuará la siguiente escritura.

5.2 Archivo: Concepto y operaciones con archivos.

- Operaciones con archivos:
 - **Leer un archivo:**
 - Especificamos nombre y lugar (de la memoria) donde colocar el siguiente bloque del archivo.
 - Necesitamos un puntero para la lectura a la posición del archivo donde se efectuará la siguiente lectura.
 - Nota: En general, un archivo o bien se lee o bien se escribe con lo que las dos operaciones pueden usar un mismo puntero, ahorrando espacio y aportando sencillez.
 - **Reubicación dentro de un archivo:**
 - Búsqueda en un archivo.
 - **Truncar un archivo:**
 - Borrar el contenido de un archivo pero haciendo que sus atributos no cambien (salvo el tamaño del archivo).

5.2 Archivo: Concepto y operaciones con archivos.

- Operaciones con archivos:
 - **Otras Operaciones Secundarias:**
 - **Appending:** Añadir información nueva al final de un archivo ya existente.
 - **Rename:** Renombrado de un archivo.
 - Nota: La mayor parte de las operaciones implican buscar la entrada en el directorio asociada al archivo ⇒ Para evitar estas búsquedas constantes:
 - El SO mantiene una tabla de archivos abiertos.
 - Puntero al archivo, Contador de aperturas del archivo, ubicación del archivo en disco.

5.2 Archivo: Concepto y operaciones con archivos.

- Operaciones con archivos:
 - **¿Qué implica una operación de apertura de archivo?:**
 - Se toma un nombre de archivo.
 - Se busca en el directorio.
 - Se comprueba que las protecciones del archivo permiten tal acceso.
 - Se copia la entrada correspondiente en la tabla de archivos abiertos.
 - Se devuelve un puntero a la entrada de la tabla de archivos abiertos que se usará en el resto de las operaciones de E/S.

5.2 Archivo: Concepto y operaciones con archivos.

- Operaciones con archivos:
 - ¿Qué ocurre en un entorno multiusuario?.
 - ¿Qué ocurre si varios usuarios abren un mismo archivo?.

- **Dos niveles de tablas internas:**
 - Una tabla por proceso.
 - Una tabla para todo el sistema.

- **¿Qué información almacena cada tabla?:**
 - Procesos:
 - Puntero al archivo.
 - Sistema:
 - Ubicación del archivo en disco, fechas de acceso, tamaño del archivo, contador de aperturas del archivo.

5.2 Archivo: Concepto y operaciones con archivos.

- Tipos de archivos:
 - La información guardada puede ser de muchos tipos:
 - Programas ejecutables.
 - Ficheros de texto.
 - Bases de datos.
 - Imágenes, sonidos, vídeos, ...
 - Para implementar los tipos de archivos se incluye el tipo como parte del nombre del archivo (extensión).

5.2 Archivo: Concepto y operaciones con archivos.

- Estructura de un archivo:
 - Según el tipo de archivo \Rightarrow estructura interna.
 - ¿Debe el SO reconocer y manejar la estructura interna de diferentes tipos de archivos que pueden existir en el Sistema?
 - UNIX y NT suponen que un archivo es simplemente una cadena de bytes.
 - Algunos SO reconocen un n^o mínimo de estructuras de archivo.
 - Todos los SO deben reconocer, al menos, una estructura: La de un ARCHIVO EJECUTABLE.

- **Ventajas:**
 - Aumenta la fiabilidad del sistema (se evita la utilización indebida de algunos archivos, por ejemplo, imprimir un fichero binario).
- **Desventajas:**
 - Aumenta el tamaño del SO.
 - Disminuye la flexibilidad.
 - Disminuye la eficiencia (menos memoria para los programas de usuario y menor rapidez de ejecución).

5.3 Métodos de Acceso: Accesos secuencial y directo.

- Métodos de Acceso:
 - ¿De qué manera se accede a la información almacenada en los archivos?:
 - La información que guardan los archivos es preciso leerla y colocarla en la memoria de la CPU.
 - Hay varias formas de acceso a la información de un archivo.
 - Acceso secuencial.
 - Acceso directo.

5.3 Métodos de Acceso: Accesos secuencial y directo.

■ Acceso Secuencial:

- Método de acceso más común.
- La información del archivo se procesa en orden, un registro tras otro.
- El archivo debe tener la capacidad de adelantarse secuencialmente y también de restablecerse al principio.
- El puntero del fichero apunta a la posición que acabamos de leer y escribir.

■ **Desventaja:**

- Problema de inserción de un bloque.

■ **Ventaja:**

- Aprovechamiento del espacio cuando no se necesita hacer modificaciones.

■ **Nota:** El sistema es bueno para hacer backups.

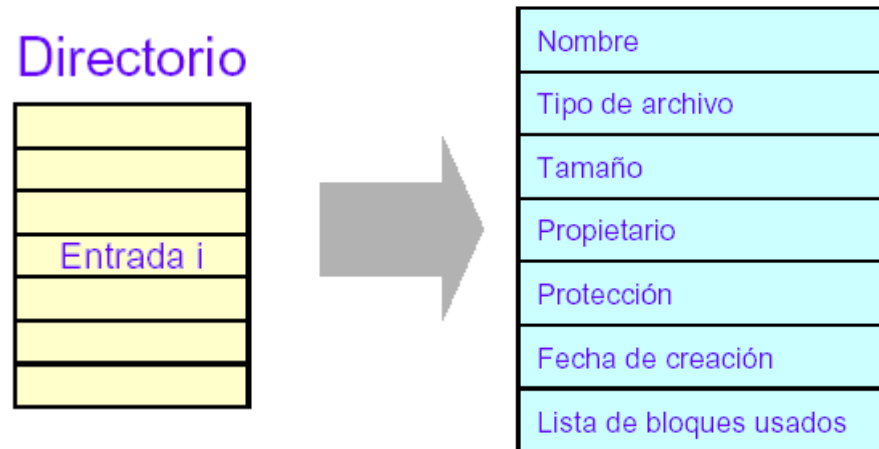
■ **Primitivas:** write(datos), read(datos), rewind.

5.3 Métodos de Acceso: Accesos secuencial y directo.

- Acceso Directo:
 - **Archivo:** Secuencia numerada de registros lógicos de longitud fija.
 - Permite que podamos leer y escribir registros rápidamente sin ningún orden específico.
 - Útiles para acceder inmediatamente a grandes cantidades de información.
 - Se puede especificar la posición del dispositivo en la que se desea leer o escribir.
 - **Primitivas:** write(posición, datos), read(posición, datos), seek(posición).
 - **Ejemplo:** Discos magnéticos.

5.4 Estructura de directorios.

- Introducción:
 - **Directorios:** Estruct. que permite organizar y acceder a los archivos.
 - Los atributos de los archivos se guardan en una estructura: **Directorio o tabla de contenidos.**
 - Igual que los archivos los directorios deben de ser no volátiles \Rightarrow se almacenan en disco.
 - Deben traerse a memoria cuando se necesitan.



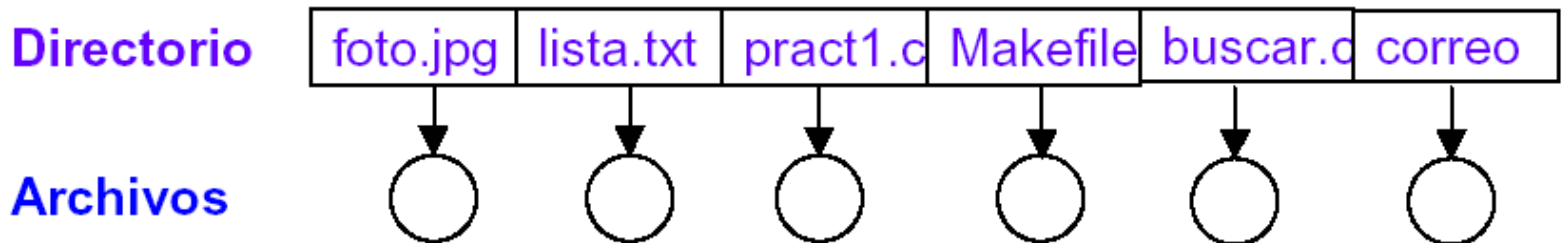
5.4 Estructura de directorios.

- **Operaciones sobre directorios:**
 - **Buscar un archivo por nombre:** Entrada en el directorio.
 - **Crear un archivo:** Añadir una entrada.
 - **Borrar un archivo:** Eliminar una entrada.
 - **Listar un directorio** (total o parcialmente).
 - **Renombrar un archivo.**
 - **Actualizar un directorio:** Aconsejables, copias de seguridad.

- **Tipos de organización de directorios:**
 - Directorio **de un nivel.**
 - Directorio **de dos niveles.**
 - Estructura **jerárquica en árbol:**
 - De grafo acíclico.
 - De grafo general.

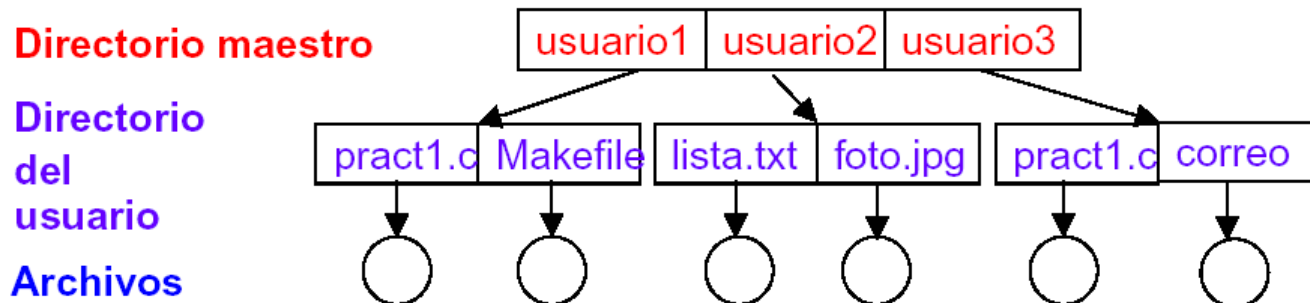
5.4 Estructura de directorios.

- Directorio de un solo nivel:
 - Todos los archivos se guardan en un solo directorio.
 - Estructura más simple y más fácil de soportar.
- Características:
 - No permite clasificar la información de ninguna manera.
 - Nombres distintos para cada archivo.
 - Problema:
 - En sistemas multiusuario hay problemas para nombrar los archivos.



5.4 Estructura de directorios.

- Directorio de dos niveles:
 - Cada usuario tiene su propio directorio de archivos de usuario.
 - Características:
 - Se asigna un directorio a cada usuario o tipo de archivo.
 - Existe un directorio maestro o padre, por encima.
 - Las operaciones de un usuario sobre los arch. están restringidas a su direc.
 - Existen operaciones de actualización del directorio maestro.
 - Existen problemas para “cooperar” entre usuarios.
 - Conceptos: “ruta de acceso” (pathname) y “camino de búsqueda” (path).
 - Fácil gestión en cuanto a la protección.



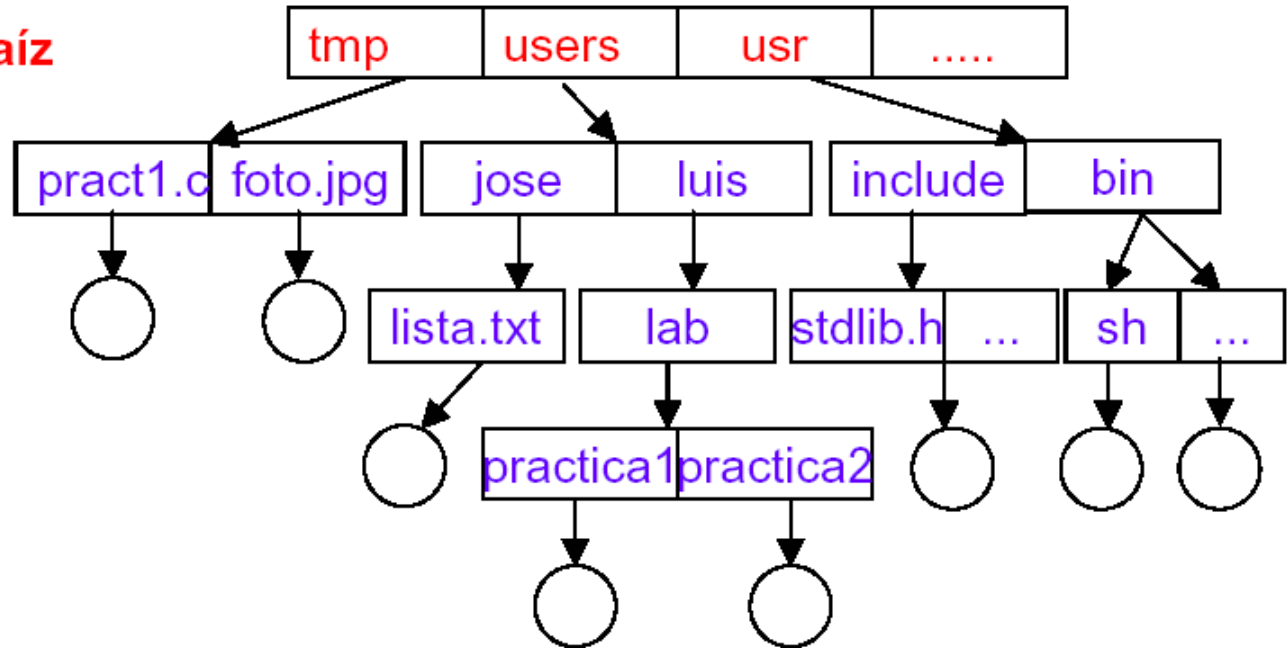
5.4 Estructura de directorios.

- Estructura jerárquica en árbol:
 - Generalización del directorio de dos niveles \Rightarrow estructura de directorios, subdirectorios, archivos, ...
 - Características:
 - Directorio raíz del que parten todos los directorios (y archivos).
 - Simplicidad en la gestión agrupando en un mismo directorio todos los archivos relacionados.
 - Conceptos: directorio "home", directorio de trabajo, ...
 - Los nombres de los archivos se pueden especificar mediante nombres de camino relativos y absolutos.
 - El identificador de archivos distingue entre archivos ordinarios y directorios.

5.4 Estructura de directorios.

- Estructura jerárquica en árbol:

Directorio raíz



5.4 Estructura de directorios.

- De grafo acíclico y de grafo general:
 - Estructura de árbol: prohíbe compartir archivos o directorios.
 - Se puede generalizar el esquema de directorio con estructura de árbol permitiendo a los directorios tener subdirectorios y archivos compartidos.

 - **De grafo acíclico:**
 - Se permite que un directorio o un fichero cuelgue de varios directorios, para ello añadimos los enlaces o los links necesarios.
 - Problema: Posibilidad de caer en ciclos de longitud infinita.

 - **De grafo general:**
 - Acceso rápido entre niveles.
 - Estructuras cerradas en ciclos.