

## **TEMA 19**

### **SISTEMAS OPERATIVOS: GESTIÓN DE ARCHIVOS Y DISPOSITIVOS**

#### **ÍNDICE**

1. INTRODUCCIÓN
2. CONCEPTO DE FICHERO
3. ORGANIZACIONES DE FICHEROS
4. OPERACIONES SOBRE FICHEROS
5. ORGANIZACIÓN DE LOS DISCOS
  - 5.1. Discos duros
  - 5.2. Discos flexibles
6. PLANIFICACIÓN DE LOS DISCOS
  - 6.1. Formas de planificación
    - 6.1.1. Planificación FCFS
    - 6.1.2. Planificación SSFT
    - 6.1.3. Planificación SCAN
    - 6.1.4. Planificación C-SCAN
    - 6.1.5. Planificación LOOK y C-LOOK
7. GESTIÓN DEL ESPACIO LIBRE
8. ASIGNACIÓN DEL ALMACENAMIENTO
  - 8.1. Asignación contigua
  - 8.2. Asignación no contigua
    - 8.2.1. Asignación enlazada
    - 8.2.2. Asignación indexada
9. DIRECTORIOS
10. BIBLIOGRAFÍA

## 1. INTRODUCCIÓN

El sistema de archivos es la parte del sistema operativo encargada de administrar el almacenamiento secundario.

Es el responsable de confeccionar los medios para el almacenamiento de la información y para que ésta sea compartida por los usuarios de forma controlada.

Las *funciones* más importantes que debe realizar un sistema de gestión de archivos son las siguientes:

**Creación de archivos:** dar un nombre y reservar un espacio en disco.

**Borrar archivos:** liberar el espacio que ocupaba.

**Abrir archivos:** para ser utilizados con las características específicas (lectura, escritura, ejecución, etc.).

**Cerrar archivos:** para finalizar un proceso e impedir su utilización posterior.

**Consultar archivos:** en modo lectura, escritura, modificación.

Además de las operaciones básicas anteriores, el sistema de archivos debe proporcionar otro tipo de ayudas a la gestión., como pueden ser:

Posibilidad de *compartir archivos* entre usuarios.

*Transferencia de datos* entre archivos.

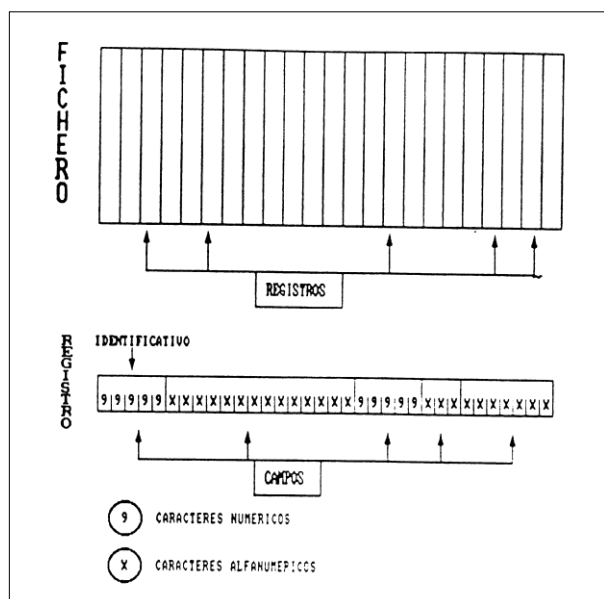
*Estructurar los archivos* de la forma más adecuada a cada aplicación concreta.

*Gestión de los soportes físicos y dispositivos* que los manejan.

*Seguridad y protección*, garantizando la integridad de los datos, los derechos de acceso y condiciones de utilización.

En general, los datos están jerarquizados según un modelo. El primer nivel de los mismos está formado por los **bits** o dígitos binarios (0,1). La agrupación de los mismos constituye los **bytes** o **caracteres**. A un nivel superior estarían los **campos**, que cuando se agrupan según una relación común forman un **registro**.

El archivo pues, sería una agrupación de registros relacionados entre sí. Por último el conjunto de archivos constituye una base de datos.



Los componentes de un sistema de archivos se organizan de forma jerárquica dependientes de sus niveles inferiores.

La *estructura básica* consta de los siguientes *niveles*:

Rutina del *sistema de archivos simbólico*, que utilizando el nombre del archivo, localiza la entrada de dicho archivo en el directorio.

Rutina de *archivos básicos* que utiliza un identificador que contiene las informaciones sobre el tamaño, número de registros, tipo de acceso y dirección de comienzo del archivo.

Rutina de *control de acceso*, que se encarga de verificar el control del acceso al archivo. Si la petición es válida se pasa al siguiente nivel y si no se produce un error.

Rutina del *sistema de archivos lógicos*, que convierte el registro lógico en un byte, obteniéndose una longitud y una dirección.

Rutina para el *sistema de archivos físicos*, que utiliza la dirección del byte mas una dirección del comienzo del archivo para determinar el bloque físico requerido.

Rutina de *asignación*, que se encarga de llevar el control de los bloques no utilizados de cada dispositivo.

Rutina de *estrategia para dispositivos*, convierte el número de bloque físico al formato requerido por el dispositivo donde se encuentra almacenado.

Rutinas de *planificación de E/S* y *manejador de dispositivos*, que son las encargadas de realizar la lectura del bloque solicitado.

## 2. CONCEPTO DE FICHERO

Un fichero o archivo es un conjunto de informaciones lógicas relacionadas entre sí. En general consiste en una serie de bits, bytes o registros cuyo significado lo proporciona el diseñador del fichero.

Para referirnos a un fichero, debemos proporcionarle un nombre y algunas características más como su longitud, su tipo, su forma de acceso, etc.

Cada fichero está subdividido en unidades menores que se conocen con el nombre de registros. Estos pueden ser en cuanto a la longitud fijos y variables.

Unas veces la estructura de los ficheros es controlada por el S.O. y otras no. En el primer caso el sistema operativo resulta de gran apoyo a la programación; en el caso contrario, se permite una gran flexibilidad a la hora de crear ficheros, pero como el apoyo del S.O. es mínimo, es necesario que los propios ficheros incluyan sus códigos para conocer la estructura de los mismos.

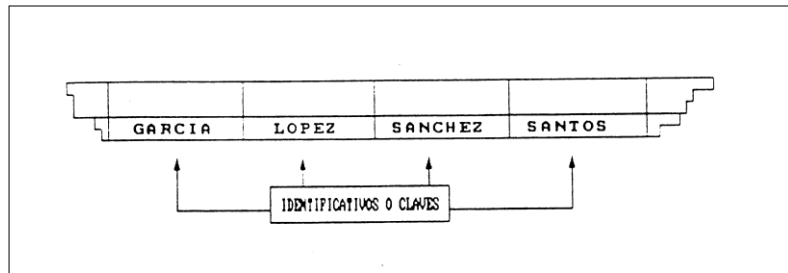
## 3. ORGANIZACIONES DE FICHEROS

Los *métodos de organización lógica* más frecuentes son:

**Secuencial.** En dicha organización los registros se colocan en orden físico y se pueden designar por números enteros consecutivos, aunque no nos sirven para referenciar su acceso.

Para acceder a un registro, hay que situarse al principio del fichero.

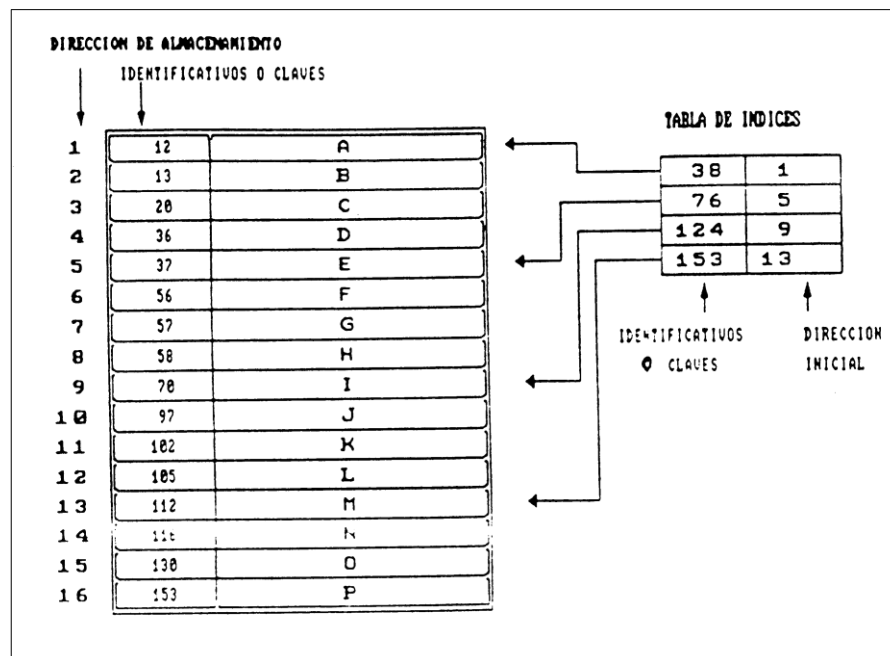
Esta organización se utiliza para archivos almacenados en cintas y a veces sobre discos magnéticos.



**Secuencial indexado.** Los registros están en secuencia lógica de una clave o índice contenida dentro de ellos.

Puede accederse a los registros de forma secuencial por orden de clave, o de forma directa a través del índice.

Este tipo de organización se utiliza sobre todo en discos.

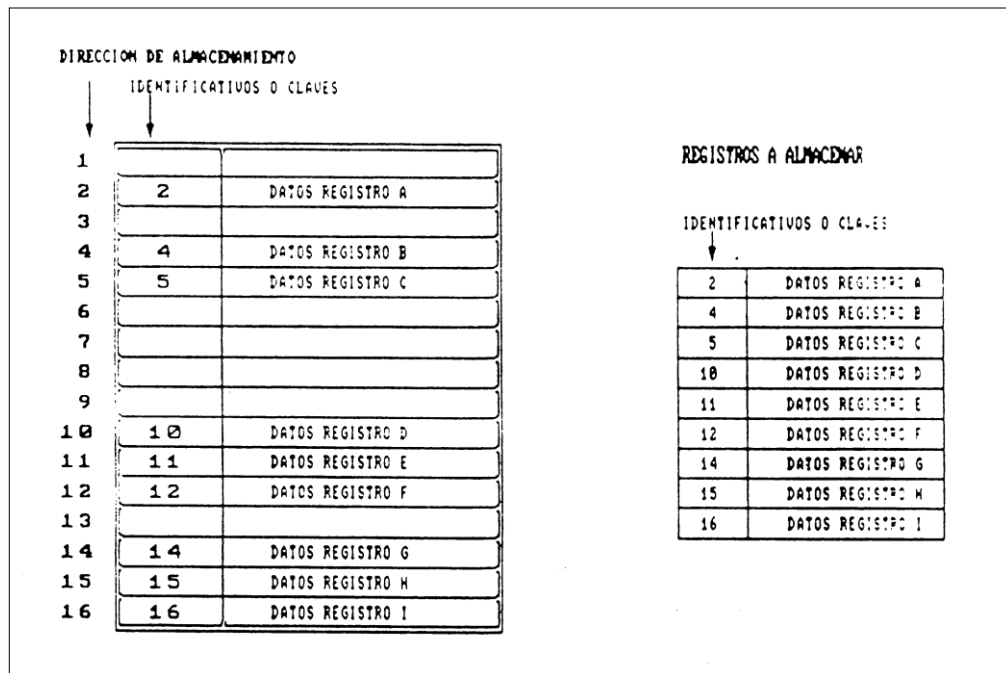


**Directo.** Las funciones de acceso se expresan en función de los atributos de los registros. Se llama *clave* a todo campo de un registro que puede servir para identificarlo.

La clave puede ser única o existir varias. En el caso de ser única se pueden utilizar dos métodos para buscar un registro.

El *direccionamiento disperso* consiste en la existencia de una función o algoritmo que transforma la clave en una dirección física donde se localiza el registro. A veces dos claves diferentes pueden dar la misma dirección, lo que se conoce con el nombre de *claves duplicadas*.

El otro método consiste en utilizar varias claves, siendo la primera de ellas la clave primaria. Para cada clave primaria se utiliza un índice que apunta a la cabecera de una lista que contiene todos los registros que posee dicha clave.



**De partición.** Consiste en un archivo grande cuyos subarchivos son archivos secuenciales llamados miembros.

#### 4. OPERACIONES SOBRE LOS FICHEROS

Las operaciones que deben ser posibles de realizar sobre un fichero, son aquellas que permiten una mejor utilización de la información contenida en dicho fichero.

Existen unas operaciones básicas y otras secundarias, pero en general, sobre un fichero se pueden realizar las siguientes:

**Creación.** Crear un fichero es definirlo mediante un nombre y unos atributos. Para ello hay que encontrar espacio libre en el sistema de ficheros y crear una nueva entrada en la lista de ficheros. Siempre es la primera operación a realizar si se quiere disponer de un fichero.

**Escritura.** Para escribir en un fichero hay que especificar su nombre y la información que queremos escribir en él. Mediante el nombre se busca en el directorio y cuál es su último registro. Con esto se busca nuevo espacio para incorporar la nueva información.

**Lectura.** Leer es tomar datos y para ello necesitamos el nombre del fichero, con objeto de localizarlo. El puntero que servía para localizar el último registro para escribir, sirve también para localizar el registro que acaba de leerse.

**Restauración.** Muchas veces, para buscar un registro determinado o para encontrar una entrada adecuada es necesario colocar la posición actual del fichero al principio del mismo; este proceso se conoce con el nombre de restauración.

**Borrado.** Cuando un fichero deja de realizar su función, podemos aprovechar el espacio que ocupa para otros almacenamientos. Para ello existe la operación de borrar, que consiste en dejar disponible el espacio que ocupa el fichero y anular la entrada correspondiente en el directorio.

**Apertura y cierre.** La operación de apertura tiene por objeto colocar un fichero en un estado donde el acceso sea posible según un modo especificado en la propia apertura.

Para no tener que analizar todo el directorio cada vez que se hace una petición de operación por parte de un fichero con la operación de apertura, se coloca el fichero en una tabla de ficheros abiertos en la que se busca con mayor rapidez.

El cierre permite mantener prohibidos los accesos al fichero sacándole de la tabla de ficheros abiertos y además hace definitivas las modificaciones hechas después de la apertura.

**Modificación.** Consiste en variar el contenido del fichero borrado, añadiendo o variando el contenido de uno o varios de sus registros.

**Edición.** Editar un fichero consiste en grabar dicho fichero o parte de él en memoria central, para tener acceso directo al mismo y poder así realizar operaciones sobre él.

**Copia.** Por motivos de seguridad y protección se realizan copias periódicas de los ficheros sobre algún soporte magnético o sobre dispositivos de E/S como impresoras.

**Renombrado.** También se puede proceder al cambio de nombre de un fichero.

## 5. ORGANIZACIÓN DE LOS DISCOS

La mayor parte del procesamiento de datos gira alrededor de los discos.

Un disco magnético es similar físicamente a un disco de vinilo. Sus dos superficies están recubiertas de una sustancia magnética que le permite el registro de los datos.

Una unidad de disco dispone de varios de estos discos, cada uno de los cuales se les suele conocer con el nombre de plato. En general el número de caras para la grabación en una unidad de disco, es el doble del número de platos.

Dependiendo de si los discos pueden ser extraídos o no de la unidad, se les llama removibles o fijos.

Según el material soporte, los discos pueden ser duros o flexibles.

### 5.1. Discos duros

Los discos están girando constantemente a una velocidad de más de 3.000 rpm. Para leer o grabar datos sobre una superficie van montadas un conjunto de *cabezas de lectura/escritura* de forma que entren en contacto con la porción del disco donde residen los datos requeridos.

Los datos están grabados en **pistas** circulares a lo largo de las caras del disco, A su vez las pistas están divididas en partes llamadas **sectores**.

El conjunto de todas las pistas que se encuentran a la misma distancia del eje de giro se conoce con el nombre de **cilindro**.

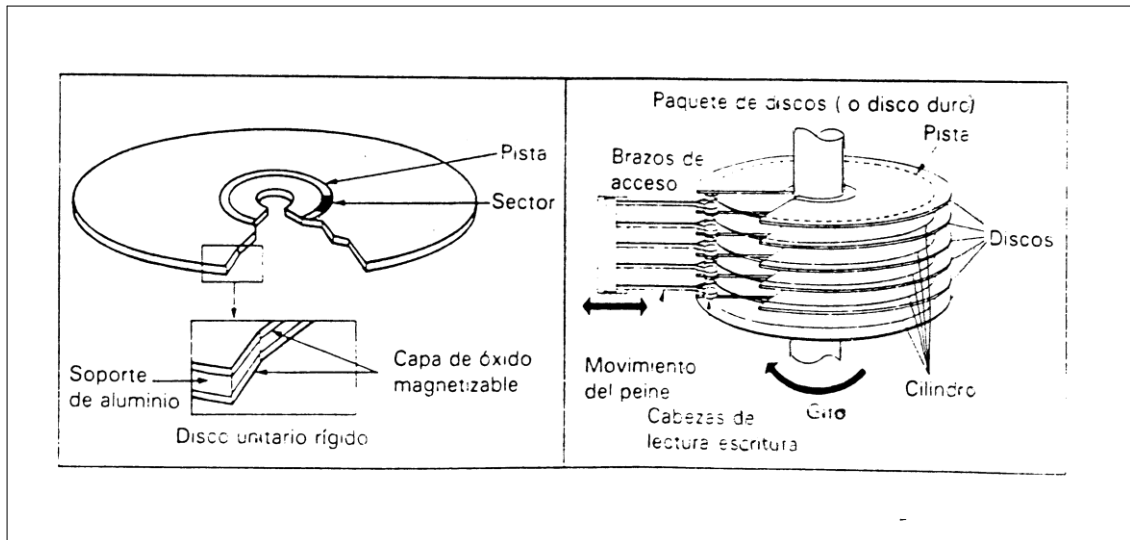
La grabación y lectura de los datos de la unidad de disco se hace a través de los cilindros.

El *hardware de un sistema de disco* puede dividirse en dos partes: la *unidad de disco* que es la parte mecánica e incluye el motor y las cabezas de lectura/escritura y el *controlador de disco*; éste toma las órdenes de la unidad central y las transforma en instrucciones básicas que puede interpretar la unidad de disco.

Atendiendo a los mecanismos de lectura/escritura, existen dos *tipos de dispositivos de discos*:

Disco de **cabezas fijas**. Contiene una cabeza de lectura/escritura por cada pista del disco. Aunque el sistema es muy rápido, requiere un hardware sofisticado, lo que hace muy caro el dispositivo.

\* Discos de **cabezas móviles**. La figura siguiente muestra un esquema del mismo. Todos los platos están unidos a un eje común que gira a gran velocidad. Las cabezas están unidas a una barra o peine, existiendo tantas cabezas como caras de disco tenga la unidad. El peine puede moverse acercándose o alejándose del eje de los discos.



El material base suele ser aluminio o cerámica. Debido a que por la gran velocidad de los discos éstos podrían ser rayados por las cabezas lectoras, dichas cabezas lectoras flotan sobre el disco gracias a un efecto aerodinámico.

A la hora de diseñar los discos se han de tener en cuenta los criterios de coste, tamaño y velocidad.

Para facilitar una mayor capacidad de almacenamiento con un tamaño adecuado, se aumenta la densidad de grabación o el número de pistas por pulgada.

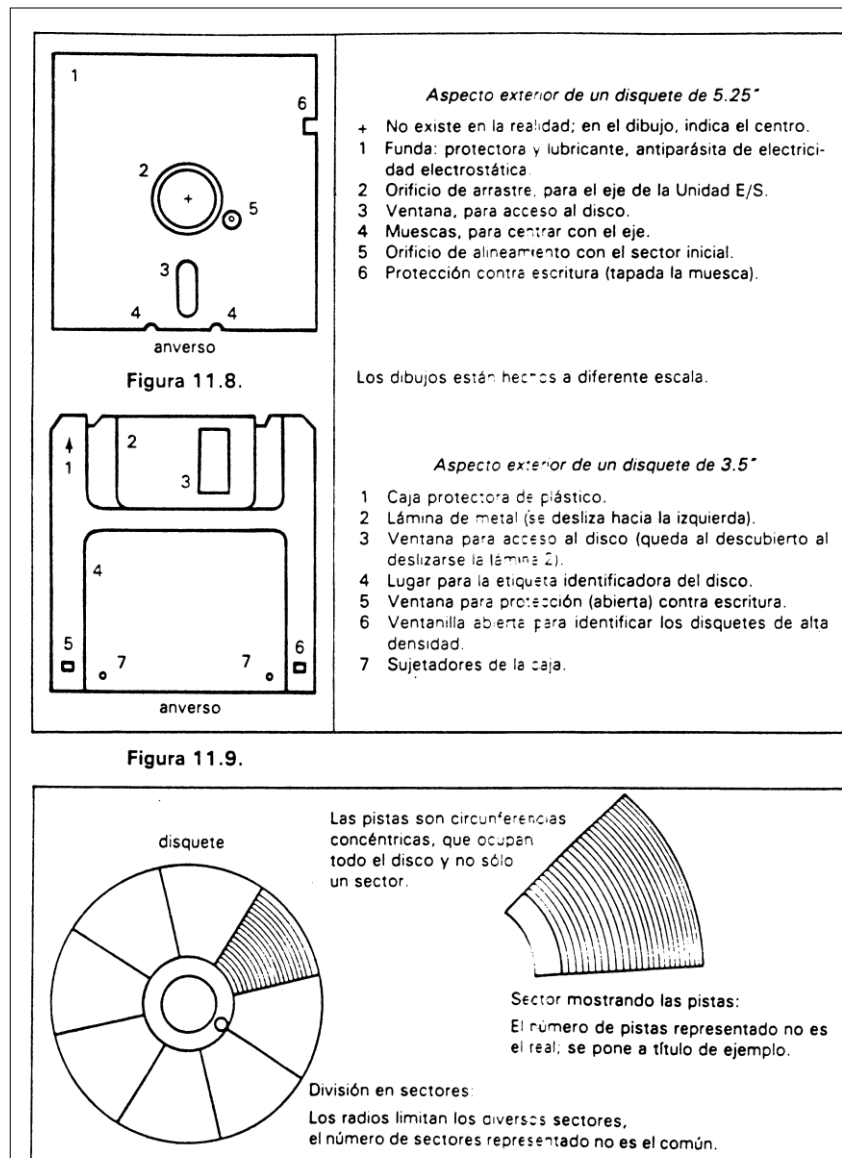
## 5.2. Discos flexibles

Son discos formados por material plástico recubierto de una sustancia magnética.

Los discos se hayan recubiertos por una sustancia resistente, de forma que las cabezas descansan directamente sobre el disco sin destruir los datos, aunque dicho recubrimiento se va desgastando con el paso del tiempo.

Los discos flexibles o floppy disks giran alrededor de 300 rpm. y son más pequeños que los discos fijos o duros.

Existen muchos tipos, aunque los más usuales son de 5 1/4 y 3 1/2 pulgadas, realizándose la grabación por ambas caras con simple, doble o alta densidad y capacidades de 720 Kbytes, 1.2 Mbytes, 1.44 Mbytes y hasta 2.88 Mbytes en la actualidad.



## 6. PLANIFICACIÓN DE LOS DISCOS

Los dispositivos de disco, como hemos visto, tienen varias partes mecánicas o electromecánicas que resultan lentas en comparación con el procesamiento de datos llevado a cabo por la CPU.

Por ello, en los sistemas basados en multiprogramación, se producen peticiones de leer o escribir, mucho más rápidas de como pueden ser atendidas por el sistema de discos. Así es frecuente encontrar colas de espera en los dispositivos de disco.

Es pues necesaria una planificación adecuada para gestionar las colas de la forma más rápida posible.

La mejor planificación será aquella que requiera menos movimientos mecánicos del sistema de discos.

### 6.1. Formas de planificación

Existen muchos *algoritmos de planificación*, unos tratando de reducir el tiempo de búsqueda y otros mejorando el tiempo de latencia (rotación de las cabezas).



Los que se describen a continuación son los más populares y tratan de optimizar la búsqueda con cabezas móviles.

#### **6.1.1. Planificación FCFS**

En dicha planificación la primera petición que llega es la primera que se atiende (FIFO); ninguna petición puede ser desplazada por la llegada de otra prioritaria.

Resulta aceptable cuando existen pocas peticiones, pero a medida que aumentan crecen los tiempos de respuesta.

Es la forma más sencilla de planificación, pero en promedio no es la que proporciona mejor servicio.

#### **6.1.2. Planificación SSFT**

En esta planificación se sirve la petición que se encuentra a la distancia más corta de la pista que se acaba de servir.

Suele proporcionar mejor resultado que la FCFS, pero tampoco es óptimo. Las pistas del centro reciben mayor atención que las externas.

Si mientras se atienden las pistas solicitadas llegan nuevas solicitudes, se ocasiona inanición de alguna de las solicitudes.

#### **6.1.3. Planificación SCAN**

Para solucionar el problema de la discriminación de pistas de la planificación SSFT, se diseñó la planificación SCAN que selecciona la petición que da una distancia de búsqueda más corta pero en una dirección preferida.

La cabeza de lectura/escritura empieza en un extremo del disco y se mueve hacia el otro sirviendo las solicitudes a medida que llega a cada pista. Cuando llega al extremo opuesto, se invierte la dirección del movimiento y continúa con el proceso.

SCAN es la base de la mayoría de las planificaciones actuales.

El algoritmo que utiliza se le denomina como *ascensor*, ya que va sirviendo según se encuentra en su camino.

#### **6.1.4. Planificación C-SCAN**

La planificación SCAN circular es una variante del anterior que proporciona un tiempo de espera más uniforme.

Se mueve la cabeza de un extremo al otro sirviendo las solicitudes según se van alcanzando. Cuando llega al otro extremo se vuelve inmediatamente al principio, sin servir ninguna solicitud para comenzar un nuevo recorrido.

#### **6.1.5. Planificaciones LOOK y C-LOOK**

Son planificaciones SCAN en las cuales la cabeza de lectura/escritura no llega hasta los extremos del disco sino sólo hasta que no quedan solicitudes en la dirección actual.

## 7. GESTIÓN DEL ESPACIO LIBRE

En cualquier sistema informático los ficheros se crean y se borran frecuentemente. Es necesario reutilizar el espacio que ocupaban los ficheros borrados y por tanto es adecuado tener un sistema adecuado de control de los espacios libres. Para llevar a cabo cualquier técnica de asignación, es necesario saber qué bloques del disco están disponibles. Por tanto, hace falta una **tabla de asignación de disco** además de una tabla de asignación de archivos. Tres técnicas son de uso común: las tablas de bits, las secciones libres encadenadas y la indexación.

### Tablas de bits

Se utiliza un vector que contiene un bit por cada bloque del disco. Cada entrada igual a 0 corresponde a un bloque libre y cada 1 corresponde a un bloque en uso.

Tienen la ventaja de que es relativamente fácil encontrar un bloque o un grupo contiguo de bloques libres. Puede ser tan pequeña como sea posible y puede mantenerse en memoria principal. Esto evita la necesidad de traer la tabla de asignación de disco a memoria cada vez que se realice una asignación.

### Secciones libres encadenadas

Las secciones libres pueden encadenarse juntas mediante un puntero y un valor de longitud en cada sección libre. Este método tiene un gasto insignificante porque no hay necesidad de tabla de asignación de disco, sino simplemente un puntero al comienzo de la cadena y la longitud de la primera sección.

### Indexación

Trata el espacio libre como si fuera un archivo y utiliza una tabla índice. Por razones de eficiencia, el índice debe trabajar con secciones de tamaño variable mejor que con bloques. De este modo, habrá una entrada en la tabla para cada sección libre del disco.

## 8. ASIGNACIÓN DEL ALMACENAMIENTO

La estrategia de asignación del almacenamiento secundario está relacionada con la eficacia de los accesos a los ficheros. La velocidad de realización de un proceso varía en función del tiempo que se requiere para acceder a los bloques que contienen la información necesaria para realizar dicho proceso; además, una utilización baja de los discos se corresponde con una mala asignación de los bloques disponibles.

Existen dos *estrategias básicas de asignación* que son homónimas a las de asignación de memoria: *asignación contigua* y *asignación no contigua*.

### 8.1. Asignación contigua

Dicho método establece que cada fichero ocupe un conjunto de bloques contiguos en el disco. Ésta es una estrategia de asignación previa que emplea secciones de tamaño variable.

La **tabla de asignación de archivos** necesita sólo una entrada por cada archivo, que muestre el bloque de comienzo y la longitud del archivo. Teniendo la dirección de comienzo y el tamaño del fichero se puede acceder a cualquier bloque del mismo.

Tanto el acceso secuencial como el directo son fáciles en dicha asignación; sin embargo, la asignación de espacio a un nuevo fichero puede ocasionar ciertos problemas como puede ser la fragmentación externa que produce tal asignación (se puede hacer difícil encontrar bloques contiguos de tamaño suficiente y de vez en cuando será necesario ejecutar un algoritmo de compactación para liberar espacio) y el cálculo del tamaño del fichero a la hora de crearlo.

## 8.2. Asignación no contigua

Debido a que los usuarios no conocen de antemano el tamaño de sus archivos, y a que estos pueden crecer o disminuir en tamaño, se utilizan dos métodos no contiguos:

### 8.2.1. Asignación enlazada o encadenada

Normalmente, la asignación se hace con bloques individuales. Cada bloque contendrá un puntero al siguiente bloque de la cadena. Cada fichero es una **lista enlazada** de bloques. Cada bloque puede encontrarse en cualquier parte del disco. Unos cuantos bytes de cada bloque se reservan para señalar el bloque siguiente. El primer bloque está en una dirección marcada por un puntero de encabezamiento grabado en el directorio.

La **tabla de asignación de archivos** necesita de nuevo una sola entrada por cada archivo que muestre el bloque de comienzo y la longitud del archivo. Aunque es posible la asignación previa, es más común simplemente asignar bloques a medida que se necesiten.

Como ventaja ofrece la facilidad para eliminar bloques defectuosos y la inserción de nuevos bloques. Por contra el acceso directo es lento y la pérdida de un puntero hace inaccesible muchos de sus bloques. Este tipo de organización física se adapta mejor a los archivos secuenciales que van a ser procesados secuencialmente.

### 8.2.2. Asignación Indexada

Es una mejora de la asignación enlazada. También mantiene punteros a los bloques asignados pero además reúne punteros a bloques de índices para facilitar el acceso directo. En este caso, la **tabla de asignación de archivos** contiene un índice separado de un nivel para cada archivo; el índice posee una entrada para cada sección asignada al archivo.

Normalmente, los índices no están almacenados físicamente como parte de la tabla de asignación de archivos. Más exactamente, el índice del archivo se guardará en un bloque aparte y la entrada del archivo en la tabla de asignación apuntará a dicho bloque. La asignación puede hacerse por bloques de tamaño fijo o en secciones de tamaño variable.

El directorio contiene la dirección del bloque de índices donde están contenidos los bloques del archivo.

Ahora el acceso directo es tan rápido como el secuencial pero el espacio de almacenamiento de bloques pudiera resultar muy grande (la tabla de asignación de archivos). Al soportar tanto el acceso secuencial como el acceso directo a los archivos se ha convertido en la forma más popular de asignación de archivos.

## 9. DIRECTORIOS

Los directorios son tablas simbólicas de los archivos del sistema.

Asociado con cualquier sistema de gestión de archivos o cualquier colección de archivos suele haber un **directorio** de archivos. El directorio contiene información sobre los archivos, incluyendo atributos, ubicación y propietario. Gran parte de esta información, especialmente la relativa al almacenamiento, la gestiona el sistema operativo. El directorio es propiamente un archivo, poseído por el sistema operativo y accesible a través de diversas rutinas de gestión de archivos. Aunque parte de la información de los directorios está disponible para los usuarios y aplicaciones, en general, la información se proporciona indirectamente, a través de rutinas del sistema.

En principio, cada entrada en un directorio define un archivo. Cada archivo se define con el nombre, sus atributos y el puntero de acceso o comienzo de los bloques de datos.

Los directorios se suelen jerarquizar en diversos *niveles*:

**Directorios de un solo nivel.** Es la forma más sencilla y en ella todos los ficheros están en un solo directorio, por lo que resulta muy fácil de gestionar.

**Directorios de dos niveles.** El directorio principal contiene la lista de los directorios subordinados y en cada uno de estos sus ficheros correspondientes.

**Estructura arborescente.** Cada usuario posee un subdirectorio o más del principal y a su vez se organiza sus archivos creando los subdirectorios que crea convenientes.

## 10. BIBLIOGRAFÍA

Stallings, W. *Sistemas operativos* Prentice Hall, 2ed., 1997

Tanenbaum, Andrew S.  
*Sistemas operativos: Diseño e implementación*  
Prentice Hall, 1990