

Sistemas informáticos

Asignación de espacio

Semana 24

Abril 2023

Cruz García, Iago



[Introducción](#)

[Sistemas de ficheros](#)

[Bloques libres](#)

[Asignación de ficheros](#)

[Asignación contigua](#)

[Asignación enlazada](#)

[Asignación indexada](#)

[Directorio](#)

[Windows](#)

[Linux](#)

[Esquema sistema de ficheros](#)

[NTFS](#)

[EXT4](#)

Introducción

Cuándo hablamos de las funciones del Sistema Operativo, una de ellas era la **gestión de ficheros** o datos. Vimos por encima estas funciones, como la asignación, lo que era un directorio, un fichero, etc. Pero nos quedamos en la parte superficial, más abstracta.

Esta semana vamos a trabajar esto de forma más práctica, con descripciones más físicas y tangibles para luego realizar una pequeña práctica con para trabajar las particiones manualmente en Linux.

Sistemas de ficheros

Antes de comenzar, debemos entender que un **fichero** no es más que información lógica almacenada en una posición de memoria **y** los metadatos que indican a quién pertenece el fichero, los permisos, fecha de creación, modificación, peso... Pero, ¿en una sola posición de memoria?

Los ficheros se componen a su vez de **registros**, información sesgada de cada fichero, de tamaño fijo o variable (dependiendo del tipo de asignación y de formato), que a su vez se divide en **campos y subcampos**.

Habría que tener en cuenta además que estos registros tienen una posición **física** en el disco. En el caso de los discos duros o **HDD**, estarán físicamente dispuestos en un disco, sector y cilindro concreto, mientras que en los discos de estado sólido o **SSD** hablamos de almacenar la información en los elementos semiconductores.

Palabra es un término común también a la hora de referirse a los ficheros aunque también al trabajar con memoria en general en un computador. Este término indica el tamaño mínimo de información que el procesador puede redirigir y almacenar en nuestro Sistema Operativo.

Menos conocidos son los términos **factor de blocaje**, número de registros físicos que entran en uno lógico, o **clúster**, que es más utilizado para HDD, siendo el número de registros físicos que se leen o escriben a la vez.

Bloques libres

Pero, ¿cómo sabe el S.O donde almacenar los ficheros? Una nota importante es que el Sistema Operativo no sabe cuántos espacio está ocupado de buenas a primeras. Pero si sabe los bloques que **no están ocupados**, los **bloques libres**. Este mantiene una lista de los bloques que todavía no se han ocupado por ningún fichero, por lo que cuando se quiere almacenar uno nuevo, se recurre a dicha lista.

Hay una forma más de controlar los bloques libres, pero es costosa: los **mapas de bits**, que mantiene un conjunto de información donde el 0 es bloque libre y 1 bloque ocupado. El problema es el espacio y el procesamiento necesario para manejarlos.

Asignación de ficheros

Ya vimos cómo el Sistema Operativo organiza los ficheros, de forma más abstracta o lógica. Ahora vamos a recordar brevemente cómo era cada una de las asignaciones brevemente:

Asignación contigua

Los bloques de datos se asocian en un bloque de archivos contiguos entre sí, sin espacios libres intermedios. La asignación inicial es muy sencilla, pues cada bloque es consecutivo. Empiezan a surgir problemas cuando hay errores en el disco o borrado de ficheros, donde el espacio puede no ser suficiente para el nuevo bloque de datos y habrá que comprimir o

fragmentar ese fichero, provocando otra serie de problemas, generalmente conocidos como [fragmentación de disco](#).

Asignación enlazada

Los bloques de datos se dividen y el primer bloque “apunta” o señala al siguiente. Ya no es necesario almacenar los datos de forma consecutiva y alivia en gran medida la fragmentación, pero ocasiona otro problema: la corrupción de los datos. Si se borra algún bloque intermedio, se pierde la información completa.

Se puede solucionar mediante listas doblemente enlazadas, pues el primer bloque de datos apunta al segundo y al último bloque, y así sucesivamente (el segundo apunta al primero y al tercero, etc). Surge otro problema, la lista de bloque de datos ocupará el doble de espacio.

También puede ocurrir la **asignación enlazada por índices**, que funciona exactamente igual, solo que en vez de apuntar a los datos, apunta al índice de memoria.

Esta asignación es la que se realiza en los discos de formato FAT (File Allocation Table) en los sistemas de Microsoft

Asignación indexada

También conocida como **asignación de inodos**. Se almacenan una lista de información de índices que contiene información de los atributos y la dirección de cada nodo o bloque.

Algo a tener en cuenta es que si el inodo tiene un tamaño superior al del bloque del formato, se divide en sub inodos, haciendo listas que apuntan a otros índices y así sucesivamente. Por ello, si hay demasiados ficheros, el tamaño del fichero de información de índices aumentará rápidamente.

Esta es la forma de asignar los ficheros de los sistemas UNIX con el formato **ext**.

Directorio

Un directorio no es más que un fichero que apunta o “almacena” múltiples ficheros, e incluso directorios. Vamos a ver como en Windows y Linux estos ficheros especiales se conforman de múltiples formas.

Windows

Cada directorio ocupa 32 bytes, de la siguiente forma:

B	8	3	1	10	2	2	2	4
n	Nombre	Extensión	Atrib.	Reserv.	Hora	Fecha	Primer bloque	Tamaño

Linux

En Linux, cada directorio ocupa 16 bytes, ya que simplemente se indica el i-nodo y el nombre. La tabla de índices inodo se encargará del resto.

B	2	14
n	inodo	Nombre

Esquema sistema de ficheros

El cómo se dividen los archivos y se asignan en un disco es una tarea ya de por sí importante, pero todavía no hemos visto los sistemas de ficheros y cómo almacenan la información. Vamos a ver los dos sistemas más comunes a día de hoy: NTFS y EXT4

NTFS

La estructuración del sistema de ficheros es la siguiente:

Registro de partición de arranque	MFT (Master Format Table)	Archivos de información del S.O	Archivos
--	--------------------------------------	--	-----------------

- **Registro de partición de arranque:** Contiene la información del arranque del sistema.

- **MFT (Master Format Table):** Información sobre el formato de la tabla de particiones, como se almacenan los ficheros y la información de los mismos.
- **Archivos de información del S.O:** Si está el Sistema Operativo instalado, contendrá la información del mismo. Si no, tendrá la información de que partición o disco se trata.
- **Archivos:** Los archivos de usuario, el espacio útil.

Podemos encontrar más información en la página oficial de [NTFS](#)

EXT4

Ext es un tipo de formato de disco basado en bloques. Se divide de la siguiente forma:

Sector de arranque	Grupo de bloques 0	Grupo de bloques 1	...	Grupo de bloques n
--------------------	--------------------	--------------------	-----	--------------------

- **Sector de arranque:** Contiene información sobre el arranque del sistema.
- **Grupo de bloques n:** es donde se almacena la información de la partición, de la siguiente forma:

Super bloque	GDT	GDT reservado	Mapa de bits de bloques	Mapa de bits de inodos	Tabla de inodos	Bloques de datos
--------------	-----	---------------	-------------------------	------------------------	-----------------	------------------

- **Superbloque:** Información total del sistema de ficheros: número de inodos, número de bloques, bloques libres, inodos libres...
- **GDT (Group Descriptors):** La información de donde se almacena el mapa de bits de bloques, de inodos, la tabla, los datos, etc.
- **GDT reservado:** Espacio reservado para futuras expansiones.
- **Mapa de bits de bloques:** Indica que bloques de datos están en uso.
- **Mapa de bits de inodos:** Indica que entradas están en uso en la tabla de inodos.
- **Tabla de inodos:** Información de cada fichero y directorio
- **Bloque de datos:** La información o datos como tal.

Podemos encontrar más información en la página de [ext4](#), la wiki de Linux o en este [enlace donde viene un poco más visual](#).