

1) Las 3 formas principales para identificar el tipo de un archivo son:

-Extensión: El nombre de cada archivo se divide en dos partes, nombre y extensión, separadas por un punto. Para saber como abrirlo, el SO mantiene una lista de extensiones conocidas. En este caso la información del tipo forma parte de los metadatos.

-Número mágico: El SO mantiene una lista de las huellas digitales de los formatos que debe manejar. Reconoce el contenido del archivo basado en sus primeros bytes. En este caso la información del tipo forma parte de los datos.

-Metadatos externos: La información de un archivo se separa en dos: el data fork y el resource fork (que es una estructura independiente). Para saber cómo abrir el archivo utiliza un campo creador en el resource fork que indica qué programa abrió el archivo para lanzarlo. En este caso la información del tipo forma parte de los metadatos.

2) Los sistemas basados en Windows también soportan una estructura de directorios basada en grafos pero, a diferencia de Unix, no son frecuentemente utilizados los enlaces duros y débiles (pero si soportados). En su lugar, en Windows es más común utilizar accesos directos (identificados por la extensión .lnk), creados para apuntar a los archivos de forma directa, pero si se solicita la edición del acceso directo se editará la referencia y no el archivo al que apunta.

3) Los atributos presentes en un archivo en fat son:

-Oculto: De estar encendido, el archivo no se le muestra al usuario. En Unix su equivalente es comenzar el nombre de un archivo con el caracter '.'

-Solo lectura: Indica que el archivo no debe ser modificado. En Unix se traduce en otorgarle al usuario permisos de lectura pero no de escritura

-Sistema: Indica que el SO no debe mover ni fragmentar el archivo. No tiene equivalente en Unix

-Archivo: Indica si el siguiente respaldo de los archivos debe incluir este archivo o no. No tiene equivalente en Unix

4) A partir de los datos obtenidos podemos decir que el mapa de bits tendrá encendido el bit 4, no podemos asegurar nada del resto de los bits. Pero si asumimos que los fragmentos del archivo son contiguos, entonces los bits 5 y 6 también estarán encendidos.

5) Para decidir cuál es el valor correcto para el cluster 10, podemos utilizar el siguiente algoritmo. Buscamos en la lista de clusters aquel que contiene el valor 10. Suponiendo que este fue el cluster n, volvemos a repetir la búsqueda pero esta vez tratando de hallar el cluster que contiene el valor n. Eventualmente no encontraremos referencia a nuestro cluster actual pero sí habrá una cabecera que apunte a este. Habiendo almacenado la cantidad de clusters recorridos hasta aquí, podemos multiplicarla por el tamaño de un cluster y compararla con el tamaño almacenado en la cabecera. Si el primero es menor al segundo, luego el valor correcto para el cluster 10 es 5. De lo contrario, el valor correcto es -1.