

Problemas resueltos en clase de sistema de ficheros y entrada/salida

Problemas de sistema de ficheros

Problema 1

Determinar el tamaño máximo de un fichero UNIX, especificando los bloques direccionados con cada tipo de puntero. Los parámetros corresponden a los siguientes valores:

Punteros a zonas de datos de 32 bits.

Tamaño de bloque 1K (1 zona = 1 bloque).

Estructura de nodo-i:

- 10 punteros directos.
- 1 puntero indirecto.
- 1 puntero indirecto doble.
- 1 puntero indirecto triple.

Realizar el mismo cálculo para un fichero en MINIX con los siguientes parámetros:

Punteros a zonas de datos de 16 bits.

Tamaño de bloque 1K (1 zona = 1 bloque).

Estructura de nodo-i:

- 7 punteros directos.
- 1 puntero indirecto.
- 1 puntero indirecto doble.

Solución:

Para determinar el tamaño máximo de un archivo en un sistema UNIX debemos tener en cuenta dos cifras:

- o El número máximo de bloques que puede llegarse a mantener. Esto se determina a partir del tamaño de los punteros a zona, ya que no podremos tener una partición de disco mayor que este límite y un archivo no puede extenderse por más de una partición.
- o Cuántos punteros a zona pueden llegarse a guardar a partir de un nodo-i, teniendo en cuenta los niveles de indirección que pueden mantenerse.

El tamaño máximo del archivo será el menor de estos dos valores.

UNIX.

Según el límite debido al tamaño de puntero, tenemos que el tamaño máximo sería:

tam. partición = 2^{32} bloques = 4 Gbloques => **4 Tbytes**

Según el número de punteros tendremos:

En una misma zona podría llegar a haber $1024 / 4 = 256$ punteros a zona, ya que una zona tiene 1024 bytes y cada puntero ocupa 4.

Por tanto, el número de punteros a zona sería:

Tipo de puntero	Punteros directos
Directos en nodo-i.	10
Indirecto simple.	256
Indirecto doble.	$256^2 = 65.536$
Indirecto triple.	$256^3 = 16.777.216$
TOTAL:	16.843.018

Por tanto, esta cifra obtenida es menor que la anterior y el tamaño máximo del archivo sería de:

16.843.018 bloques = 16.842.018 Kb => **16'064 Gbytes**

MINIX.

Debido al tamaño de puntero, el tamaño máximo es en este caso:

tam. partición = 2^{16} bloques = 64 Kbloques => **64 Mbytes**

Según el número de punteros tendremos:

En una zona podría haber $1024 / 2 = 512$ punteros a zona, ya que una zona tiene 1024 bytes y cada puntero ocupa 2.

Tipo de puntero	Punteros directos
Directos en nodo-i.	7
Indirecto simple.	512
Indirecto doble.	$512^2 = 262.144$
TOTAL:	262.663

Esto hace un total de 262.663 bloques => **256'5 Mbytes.**

En este caso, el límite lo proporciona el tamaño de la partición. Con esto el tamaño máximo queda en **64 Mbytes.**

Problema 2

Determinar la estructura de un disco de 20 Mb en MINIX con los siguientes parámetros:

- Punteros a zonas de datos de 16 bits.
- Tamaño de bloque 1K (1 zona = 1 bloque).
- Tamaño de nodo-i de 32 bytes.
- Número máximo de nodos-i: 511.

Se pide:

a) Especifique claramente todas las estructuras de datos que forman el sistema de archivos y los bloques que ocupan.

b) En caso de resultar dañada la estructura del mapa de bits de zonas, piense la forma de reconstruirla con la información de la que se dispone en el resto de estructuras del sistema de archivos (se supone que estas se conservan inalteradas).

Solución:

a) Estructura del disco.

Un disco MINIX tiene las siguientes partes:

(a) Bloque de arranque: **1**.

(b) Superbloque: **1**.

(c) Mapa de bits para nodos-i: **1**.

Necesitamos un bit por archivo. Como el disco únicamente va a permitir 511 archivos, basta con 511 bits.

En este sistema un bloque tiene 1024 bytes (8192 bits), con lo que el mapa cabe en un solo bloque.

(d) Mapa de bits para zonas: **3**.

Un disco de 20 Mb tiene 20480 bloques de 1 Kb. En cada bloque caben $8 * 1024$ bits.

Por tanto, se tendría:

$$nb = \frac{20 * 1024}{8 * 1024} = \frac{20}{8} = 2.5 \rightarrow 3$$

(e) Bloques para nodos-i: **16**.

Como cada nodo-i ocupa 32 bytes, en un bloque cabrían $1024 / 32 = 32$ nodos-i.

Si vamos a tener 511 nodos-i, necesitaremos:

$$nb = \left\lceil \frac{511}{32} \right\rceil = 16$$

(f) Bloques para datos: **20458**.

Únicamente hay que descontar los bloques vistos en los puntos anteriores de la capacidad total del disco.

$$nb = 20480 - (1 + 1 + 1 + 3 + 16) = 20480 - 22 = 20458$$

b) Reconstrucción del mapa de bits de zonas.

Suponiendo que el resto del disco no ha resultado dañado, se podría realizar lo siguiente:

(a) Reinicializar todo el mapa de bits de zonas de manera que todos las zonas de disco aparezcan como libres.

(b) Siguiendo el mapa de bits para nodos-i, recorrer todos los nodos-i utilizados. Para cada uno de ellos habría que consultar sus punteros directos a zona y marcar tales zonas como ocupadas en el mapa de bits que se dañó.

Lo mismo para los punteros indirecto simple e indirecto doble, bajando hasta el nivel de zona de punteros directos. De esta manera todas las zonas utilizadas por todos los archivos son marcadas en el mapa de bits como “en uso”.

Una vez terminados estos dos pasos, el mapa de bits para zonas estará recuperado.

Problema 3

Se tiene un dispositivo de almacenamiento de 32 Kb, con bloques de 512 bytes (1 zona = 1 bloque). Este dispositivo va a ser inicializado para poder utilizar un sistema de archivo similar al de UNIX con entradas de directorio de 16 bytes, un máximo de 31 archivos y nodos-i de 16 bytes de los cuales 60 bits pueden dedicarse a punteros directos a zona. Se supone que el tamaño de los punteros a zona es el mínimo posible. Se pide:

- Decidir cómo quedaría estructurado el disco. Es decir, cuántos bloques se dedican a cada una de las estructuras que mantiene el sistema de archivo.
- ¿Cuál sería el tamaño máximo de un archivo?

Solución:

a) El disco tendrá $(32 \times 1024) / 512 = 64$ bloques

1 bloque de arranque

1 superbloque

1 bloque para el mapa de nodos-i

Hay 32 nodos-i, por tanto hacen falta 31 bits para mapearlos (1 bloque)

1 bloque para mapa de zonas de datos

Como máximo habrá 64bits (algunos no hacen falta por estar ya ocupados por bloques de arranque, superbloque...)

1 bloque para nodos-i

Hay 31 nodos-i con 16 bytes por cada uno. Luego 496 bytes

59 bloques para datos

De 64 bloques, hay 5 ya ocupados.

b) Tamaño máximo de archivo: Hay que saber cuántos bloques podemos referenciar con 60 bits que mantiene un nodo-i para punteros directos.

El tamaño de puntero a bloque será:

64 bloques $\rightarrow 2^6$ bloques \rightarrow punteros 6 bits

Por tanto: $60/6=10$ punteros directos y cada bloque de 512 bytes \rightarrow **5Kb**

Problema 4

Sea un sistema de ficheros tipo MINIX con las siguientes características:

- Tamaño de bloque de 1Kbyte
- 1 zona = 2 bloques

- El tamaño del nodo-i es de 64 bytes y contiene las siguientes referencias a zonas: 10 referencias directas, 1 referencia indirecta y una referencia doblemente indirecta
- El tamaño de las referencias a zonas en disco es de 32 bits

Calcular el tamaño máximo que puede tener un fichero en este sistema de ficheros

Solución:

Es el mínimo de las siguientes cantidades:

a) $2^{32} * 2Kb = 2^{43} = 8 \text{ Tbytes}$

b) $(10 + (2048/4) + (2048/4)^2) * 2Kb = 525.332 \text{ Kb} = 513 \text{ Mb (aprox)}$

Por tanto **513Mb**