

# Sistemas de Archivos. Fundamentos teóricos

## Sumario

- 1 Sistemas de Archivos
  - ◆ 1.1 Definición
  - ◆ 1.2 Archivo
  - ◆ 1.3 Directorio
  - ◆ 1.4 Metadatos o Atributo
  - ◆ 1.5 Path o Ruta
- 2 Objetivos
- 3 Características de archivos y directorios
  - ◆ 3.1 Introducción
  - ◆ 3.2 Archivos
  - ◆ 3.3 Gestión del espacio libre en el dispositivo
  - ◆ 3.4 Directorios
  - ◆ 3.5 Gestión de la asignación de espacio de almacenamiento
    - ◇ 3.5.1 Mediante listas enlazadas. Tablas FAT
    - ◇ 3.5.2 Mediante índices. I-nodos
    - ◇ 3.5.3 Estructura del sistema de archivos en una partición de disco
    - ◇ 3.5.4 FAT y NTFS
  - ◆ 3.6 Características avanzadas de Sistemas de Archivos
  - ◆ 3.7 Sistemas de Archivos en Sistemas Operativos actuales
    - ◇ 3.7.1 FAT
    - ◇ 3.7.2 NTFS
    - ◇ 3.7.3 ext2, ext3 y ext4
    - ◇ 3.7.4 Otros: reiserfs, xfs, jfs, hfs+
  - ◆ 3.8 Sistemas de Archivos en Red
    - ◇ 3.8.1 NFS
    - ◇ 3.8.2 FTP
    - ◇ 3.8.3 Samba
  - ◆ 3.9 Herramientas
    - ◇ 3.9.1 Recopilación de información del Sistema de Archivos
    - ◇ 3.9.2 Creación de Sistemas de Archivos
    - ◇ 3.9.3 Herramientas de monitorización, chequeo y reparación del Sistema de Archivos

## Sistemas de Archivos

### Definición

Como elemento estructural clave de los Sistemas Operativo, el concepto de Sistema de Archivos está relacionado con una abstracción accesible y manejable de cara al usuario. Por encima de las complejidades inherentes a los dispositivos y al sistema de Entrada y Salida, el usuario final dispone de una visión de alto nivel y uniforme de los recursos de almacenamiento secundario.

Independientemente de la base de almacenamiento subyacente, disco duro, disco óptico, dispositivo de estado sólido, pendrive, cinta, etc., la visión que el usuario tiene de los recursos de almacenamiento es la misma; basada en una estructura jerárquica de directorios y archivos.

Las estructuras y conceptos claves, relacionados con los Sistemas de Archivos son los siguientes:

### Archivo

Un archivo es un elemento de información abstracto. Los archivos son elementos identificables de forma única en el Sistema de Archivos. La manera de identificar ese elemento depende del contexto:

- Para un **usuario** un archivo o directorio se identifica mediante su path o ruta absoluta, es decir, un identificador textual único que contiene la secuencia de directorios en el sistema de archivos a los que es necesario acceder para llegar al elemento en cuestión, más el nombre del propio elemento.

- Para el propio **sistema operativo** todo archivo o directorio se identifica mediante un identificador numérico, cuando éste está siendo utilizado por algún proceso. Además todo archivo o directorio tiene asociados un conjunto de **metadatos**, o atributos, que son utilizados por el sistema de archivos para la gestión del elemento.

Asociados a los archivos encontramos los programas que permiten procesar la información contenida en ellos. Por ejemplo, algunas asociaciones podrían ser:

- archivo de texto **odt** -> LibreOffice Writer (o cualquier procesador de textos que interprete el formato .odt)
- archivo **pdf** -> Visor de Documentos (o cualquier otro programa que interprete el formato .pdf)
- archivo **html** -> Navegador Web Firefox (o cualquier otro navegador)

La extensión de un archivo es por tanto un elemento importante a la hora de asociar el tipo de formato de un archivo con el programa utilizado para su procesamiento. Debemos tener en cuenta lo siguiente:

- Puede haber un programa por defecto asociado a un tipo de archivo en el sistema. Esto es un aspecto de configuración del sistema operativo y es posible que se pueda modificar para que sea otro programa el encargado, por defecto, de abrir el tipo de archivo en cuestión.
- La extensión de un archivo es un identificador, a la derecha del último carácter "." (punto) en el nombre del archivo. Por ejemplo: carta. **odt**, index. **html**, nominas. **pdf**

## Directorio

Un directorio es un tipo especial de archivo cuyo fin es dotar de estructura organizativa al sistema de archivos. El paradigma de implementación en los sistemas de archivos actuales es el de **estructura de árbol invertido** de anidamiento arbitrario.

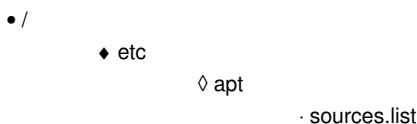
## Metadatos o Atributo

Información relacionada con los archivos y directorios que el sistema de archivos necesita para su gestión. En el caso de GNU/Linux se utilizan estructuras de metadatos denominadas nodos-i, o i-nodos, de las cuales se hablará más adelante.

## Path o Ruta

Identificador textual único de cada elemento del sistema de archivos, archivo o directorio. Desde el punto de vista del usuario es el elemento principal para la localización e identificación de éstos. Un path o ruta es un identificador único del camino a seguir para poder localizar un elemento del sistema de archivos.

Por ejemplo, la ruta **/etc/apt/sources.list**, identifica el archivo **sources.list**, dentro del directorio apt, que a su vez reside dentro del directorio etc, el cual se encuentra en la raíz del sistema de archivos



## Objetivos

El sistema operativo utiliza los Sistemas de Archivos como elemento estructural de organización y despliegue de la información en dispositivos de almacenamiento secundario. Estos dispositivos son variados y están contruidos sobre distintas tecnologías, como discos duros, unidades ópticas, unidades flash, etc.

Teniendo en cuenta la naturaleza de los recursos de almacenamiento gestionados por los Sistemas de Archivos, los objetivos principales que éstos deben proporcionar son:

- **Implementación de los conceptos de Archivo y Directorio**
- **Uniformidad de nombres y rutas**
- **Gestión del espacio libre en el dispositivo**
- **Gestión del espacio usado por archivos en el dispositivo**
- **Eficiencia en la gestión del espacio**
- **Consistencia de Archivos y Directorios**
- **Seguridad de acceso para Archivos y Directorios**

- Rendimiento y eficiencia en uso de los dispositivos
- Sencillez y ausencia de errores
- Interfaz amigable

## Características de archivos y directorios

### Introducción

Para poder gestionar de un modo eficiente y general los dispositivos de almacenamiento secundario se utilizan los Sistemas de Archivos. Existen 2 conceptos estructurales centrales que ayudan a implementar los Sistemas de Archivos, los archivos y directorios. Los archivos almacenan la información, tanto programas como datos utilizados por los mismos, de un modo homogéneo, transparente y extensible. Los directorios ayudan a mantener una estructura de acceso y visualización del Sistema de Archivos que lo haga fácilmente comprensible y accesible. La información en un Sistema de Archivos se organiza mediante una **estructura jerárquica**, con forma de árbol invertido, cuya raíz está en el punto más elevado de la jerarquía. Llamaremos a esta estructura **Árbol de Directorios**, la cual se despliega en anchura y profundidad de un modo arbitrario y recursivo. Un directorio puede contener archivos o directorios anidados en él. De este modo la estructura puede crecer en niveles de un modo ilimitado.

Por tanto y según lo anterior, para hacer referencia a un archivo o directorio, es necesario conocer su ubicación dentro del Árbol de Directorios. Para este fin se utilizan las **rutas o paths**, las cuales son cadenas de texto, separadas por caracteres tipográficos especiales (/ en Linux, \ en Windows) que indican la posición exacta de un archivo o directorio en el árbol. Existen 2 tipos de rutas o paths:

- **Absoluto**: Indica toda la secuencia en orden, desde la raíz, que es necesario recorrer para llegar a un archivo o directorio concreto
- **Relativo**: Indica la secuencia en orden, desde un directorio predefinido, directorio actual o directorio de trabajo, que es necesario recorrer para llegar a un archivo o directorio concreto

### Archivos

Un archivo es una abstracción utilizada por el sistema operativo, utilizada para representar un elemento de información persistente que podrá ser accedido y modificado de un modo, a priori, indeterminado. Es habitual, que los contenidos y estructuras de un archivo estén relacionados con algún programa que accederá a los contenidos del mismo. Sin embargo, para el sistema operativo, un archivo no es más que una secuencia de bytes, sin orden ni estructura predefinidas. Esta simplificación permite, que los archivos sean gestionados por el sistema operativo de un modo flexible, eficiente y homogéneo. Según esto, para el sistema operativo los contenidos y estructura del archivo son irrelevantes, simplemente proporcionará a los programas de aplicación un modo de acceso y una serie de funciones genéricas de manejo del archivo. Funciones como:

- Apertura
- Lectura
- Escritura
- Renombrado
- Cierre
- Borrado

serán accesibles a los programas de aplicación a través de APIs de programación, que efectuarán las correspondientes llamadas al sistema.

Además de los propios contenidos del archivo, es decir, la información semánticamente relevante almacenada en los mismos, se dispone de los denominados **metadatos o atributos** asociados al archivo. Éstos son elementos informativos utilizados por el sistema operativo para gestionar el archivo de un modo coherente y seguro. Por ejemplo, metadatos podrían ser:

- Fechas de creación o modificación
- Usuario propietario del archivo
- Modo de acceso (lectura, escritura, lectura/escritura)
- Enlaces que apuntan al archivo
- Máscaras de permisos

y un largo etc., que dependerá en gran medida del tipo del Sistema de Archivos considerado

### Gestión del espacio libre en el dispositivo

La base de almacenamiento del Sistema de Archivos (SA) es un dispositivo de de almacenamiento secundario (disco, unidad óptica, flash drive, etc.). Un aspecto fundamental gestionado por el SA es el espacio de almacenamiento libre del dispositivo. Al ser estos dispositivos de tipo bloque la gestión de espacio libre estará relacionada con la identificación de qué bloques de dispositivo están libres, es decir, disponibles para ser asignados a nuevos

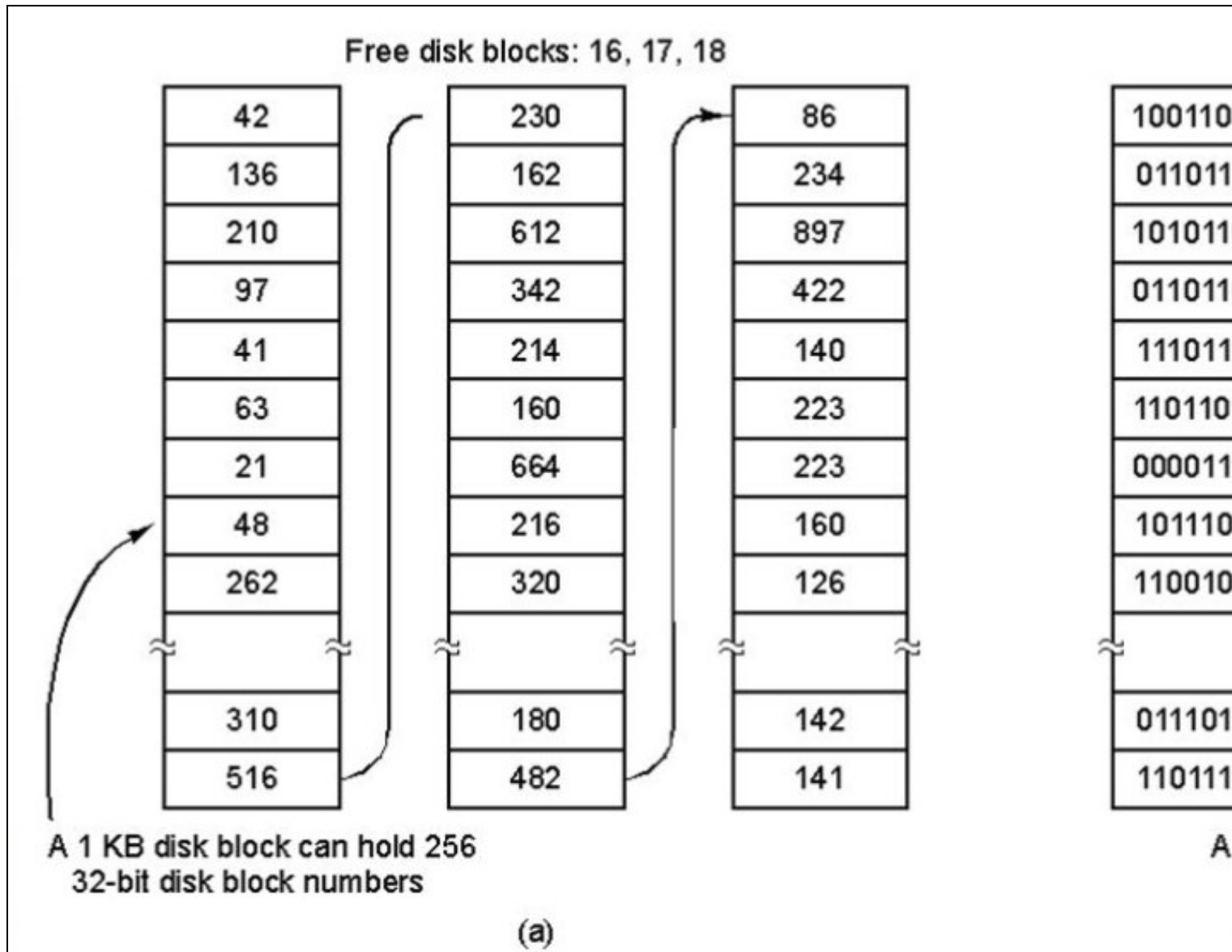
archivos y directorios del SA.

Un sistema de gestión muy utilizado del espacio libre es el conocido como **mapa de bits**. Un mapa de bits es una estructura en la que se asigna un bit a cada bloque de almacenamiento del dispositivo. Cada uno de estos bits puede estar a 0, indicando que el bloque está disponible, es decir libre, o estar a 1, lo cual indicaría que ese bloque forma parte de la asignación de espacio de almacenamiento asociada a algún archivo dentro del SA.

Otra técnica de gestión de espacio libre es la de **lista enlazada de números de bloques libres**. En esta aproximación se almacena una lista de bloques de disco que almacenan números de bloques libres.

Las estructuras de gestión de espacio libre son mantenidas por el SA en zonas reservadas del dispositivo que se crean durante el proceso de formateo y creación del SA.

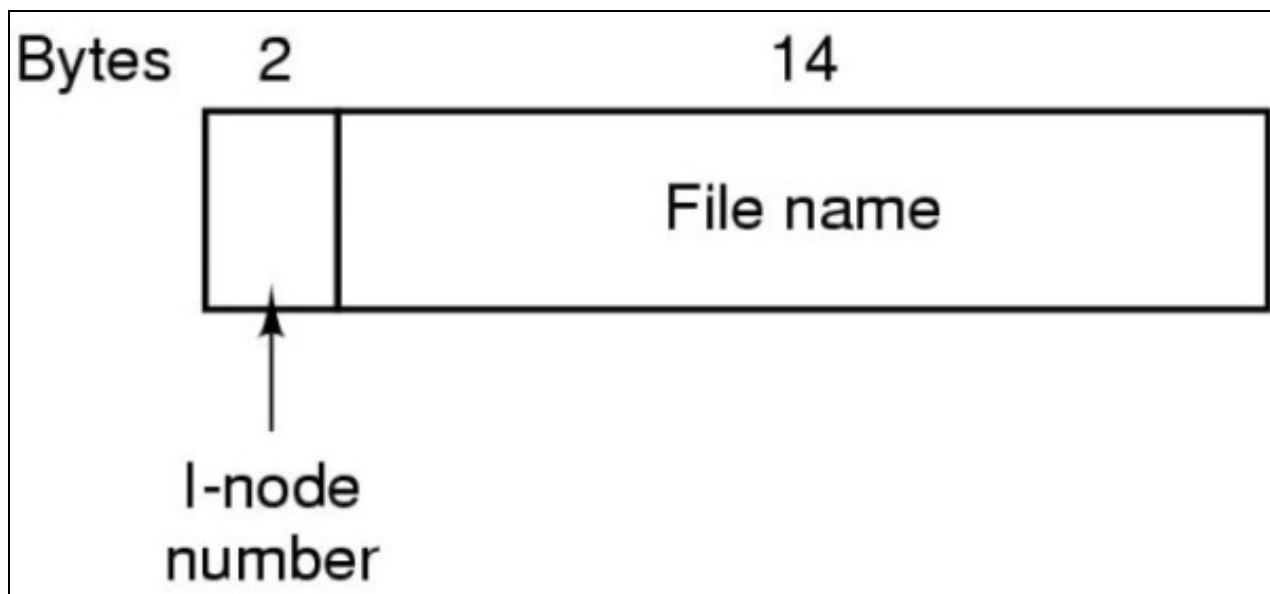
En la siguiente figura podemos ver la gestión de bloques libres utilizando **lista enlazada (figura a)** y **mapa de bits (figura b)**



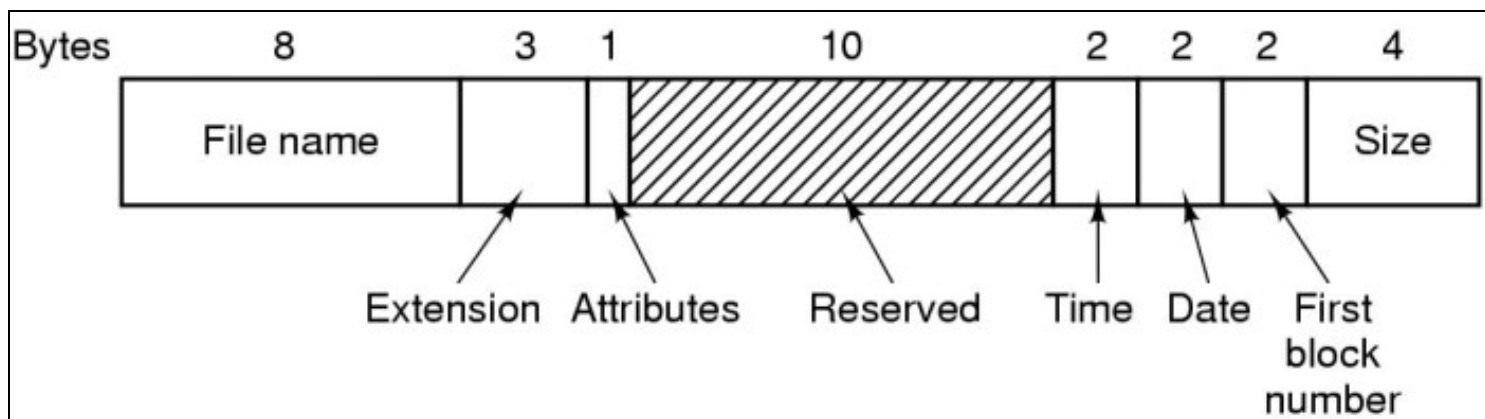
## Directorios

Los directorios **son un tipo especial de archivos que se utilizan para almacenar listas de archivos y otros directorios**. Se utilizan para implementar la estructura jerárquica de directorios del Sistema de Archivos. Un directorio, por tanto, no es más que una lista que asocia nombres de archivos y directorios con elementos que permiten al Sistema de Archivos localizar los contenidos del mismo en el dispositivo de almacenamiento.

En el caso de Linux, una entrada de directorio asocia un nombre de archivo o directorio con un número de **i-nodo**, una estructura del Sistema de Archivos que permite acceder a los metadatos del archivo y directorio, a través del cual se puede acceder al mismo en el soporte de almacenamiento correspondiente.



Otros sistemas como MS-DOS y Windows utilizan la propia entrada de directorio para almacenar la metainformación (metadatos) del archivo o directorio



Veremos un poco más adelante la estructura de un i-nodo (o nodo-i) utilizado en los Sistemas de Archivos de UNIX y Linux

Existen, como en el caso de los archivos, una serie de funciones que pueden ser invocadas, vía API, por los programas de aplicación para acceder a los directorios del Sistema de Archivos. Algunas son:

- Apertura
- Lectura
- Escritura
- Cierre
- Borrado

## Gestión de la asignación de espacio de almacenamiento

Un aspecto fundamental a gestionar por parte de los Sistemas de Archivos (SA) es la asignación de espacio de almacenamiento asociado a archivos y directorios. Como hemos visto, un archivo es un conjunto de información persistente. En consecuencia, habrá un soporte de almacenamiento de datos subyacente: el dispositivo de almacenamiento secundario sobre el que se despliega el SA.

Cada archivo, que debe ser almacenado en el soporte de almacenamiento secundario gestionado por el sistema de archivos, constará de un conjunto de bloques, los cuales son elementos fundamentales de información en el dispositivo. Para poder localizar los datos asociados a un archivo o directorio en el dispositivo, a través de la identificación de los bloques asociados al mismo, el sistema de archivos debe de gestionar datos al respecto, es decir, metadatos, de estos elementos.

A la hora de determinar el modo en que se gestionan la asignación de bloques de dispositivo para almacenamiento de datos de un archivo o directorio, existen varias alternativas, siendo las más utilizadas las descritas a continuación

### Mediante listas enlazadas. Tablas FAT

En esta modalidad se parte de la **entrada del archivo o directorio** del directorio en el que se encuentra el elemento (archivo o directorio). Esa entrada debe contener la información del **número del primer bloque** de datos del archivo o directorio en el dispositivo. Partiendo de ese primer bloque se consulta una tabla en memoria, denominada tabla **FAT (File Allocation Table)**, a partir de la cual determinamos la cadena, o lista enlazada, de bloques consecutivos en el dispositivo en el que se encuentran los datos del archivo o directorio.

La tabla FAT contiene una entrada para cada bloque lógico asociado al dispositivo, en esa entrada se encuentra el **número del siguiente bloque lógico** de almacenamiento asociado al archivo o directorio procesado. Siguiendo esa lista de bloques consecutivamente llegaremos al último bloque de datos, el cual estará marcado con un número de bloque no asignable, normalmente indicado con el número -1, indicador de que se ha alcanzado el fin de archivo **EOF** (End Of File) y que la cadena de bloques ha finalizado.

Un ejemplo lo podemos ver más abajo cuando se habla del Sistema de Archivos FAT de Windows

Physical  
block

0		
1		
2	10	
3	11	
4	7	← File A starts here
5		
6	3	← File B starts here
7	2	
8		
9		
10	12	
11	14	
12	-1	
13		
14	-1	
15		← Unused block

## Mediante índices. I-nodos

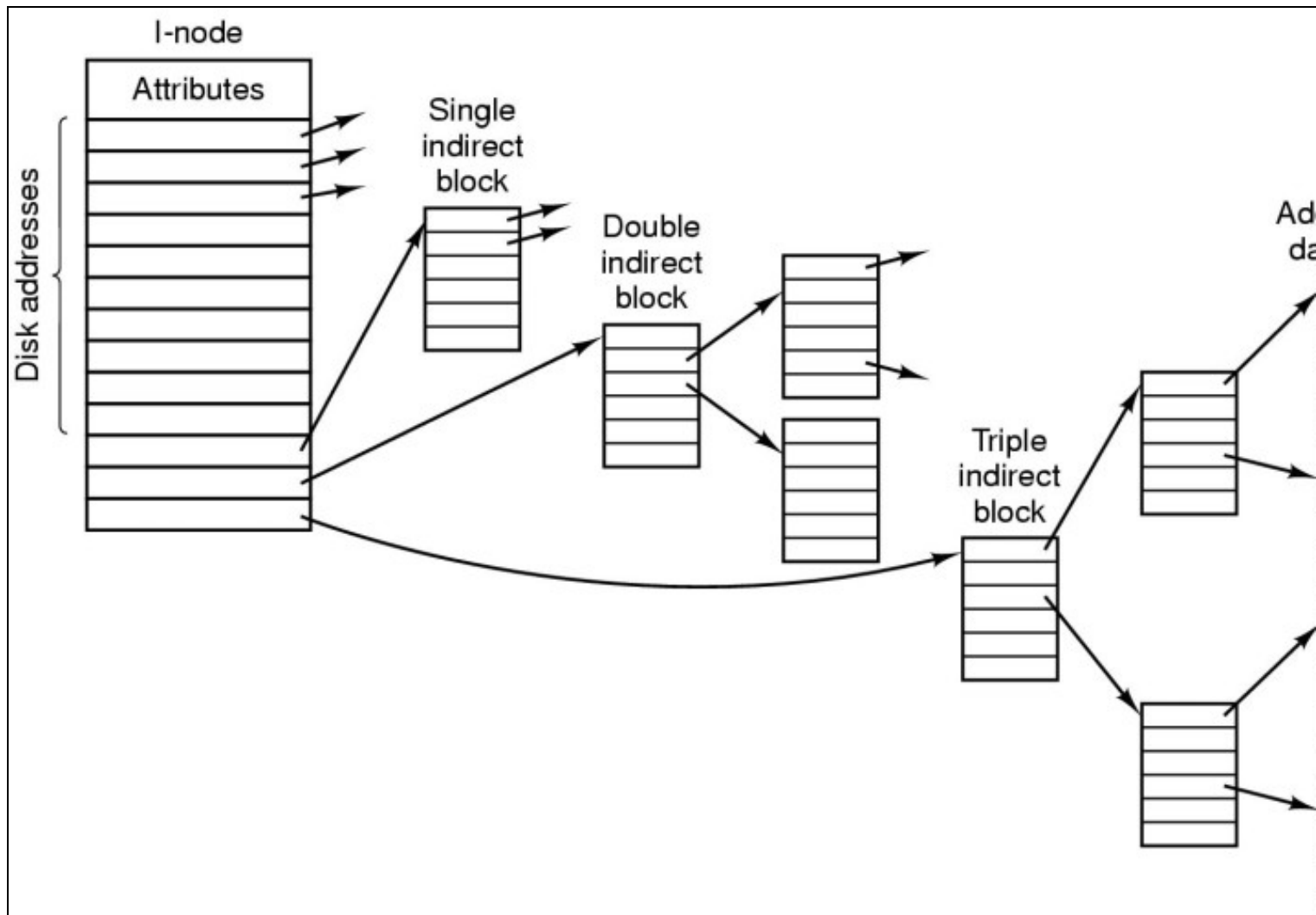
Otro sistema de gestión de la asignación de bloques lógicos de almacenamiento a archivos y directorios es mediante estructuras indexadas.

Un claro ejemplo de esta modalidad son los Sistemas de Archivos utilizados por **UNIX y GNU/Linux** que se basan en el concepto de **i-nodo** (nodo índice). Un i-nodo es un elemento de metadatos asociado a un archivo o directorio a través de la entrada del directorio descriptiva del elemento en cuestión. Ese elemento almacena toda la información del archivo o directorio.

**Todo archivo o directorio en un sistema de archivos en Linux tiene asociado un i-nodo.** A través de ese elemento el sistema de archivos puede conocer los detalles necesarios para gestionar los recursos asignados al archivo o directorio. El i-nodo consta de una serie de elementos informativos almacenados en secuencia.

- En los primeros elementos se almacenan los metadatos del archivo, como **fechas, propietario, grupo del propietario, modo, permisos**, etc.
- A continuación se almacenan las **direcciones de bloques en disco** que contienen datos del archivo. Habitualmente se almacenan sobre unas **10 de esas direcciones en el propio i-nodo**, de modo que, si el tamaño de bloque son, por ejemplo, 4KB, podríamos direccionar de este modo archivos de  $10 \times 4 = 40\text{KB}$ , pero... ¿qué pasa con archivos más grandes?. En este caso necesitamos utilizar los 3 últimos elementos del i-nodo, **los apuntadores indirectos simple, doble y triple**
- **Apuntador indirecto simple:** Almacena la dirección de un bloque de disco que contiene direcciones de bloques de disco del archivo
- **Apuntador indirecto doble:** Almacena la dirección de un bloque de disco que contiene apuntadores a direcciones de bloque de disco que contienen direcciones de bloque de disco del archivo
- **Apuntador indirecto triple:** Almacena la dirección de un bloque de disco que contiene apuntadores a direcciones de bloque de disco que contienen bloques con direcciones que apuntan a direcciones de bloque de disco del archivo

De este modo pueden ubicarse bloques de archivos de hasta TB de tamaño





Los i-nodos se almacenan en la partición del dispositivo sobre la que se crea el sistema de archivos, en una zona reservada para ello. Durante el proceso de formateo del volumen físico se reserva una zona para albergar estas estructuras definitorias de los archivos y directorios.

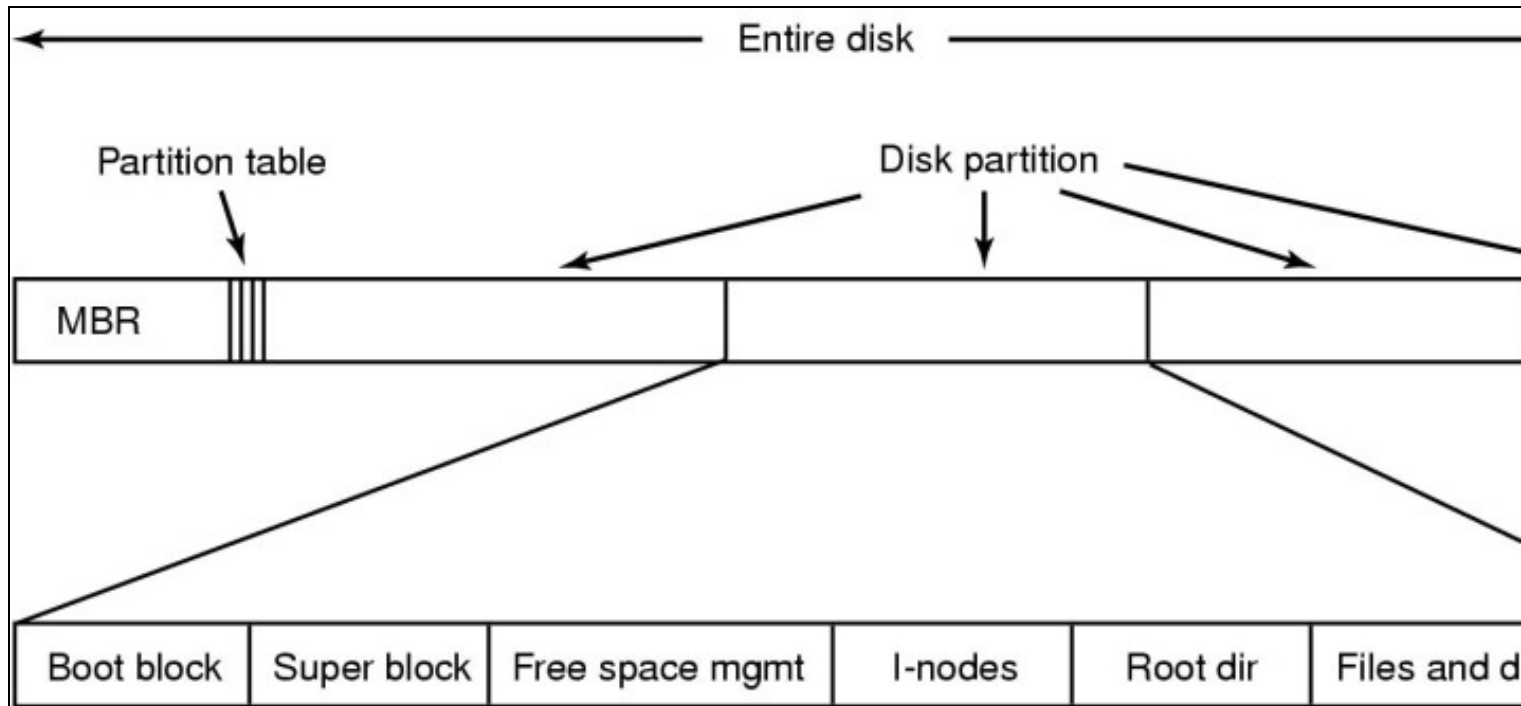
Cada i-nodo tiene asociado un identificador numérico único, de este modo cada archivo o directorio es gestionado por un único i-nodo. Este es el motivo por el cual las entradas de directorio en sistemas de archivos que utilizan i-nodos solo contienen el nombre del archivo y el número de su i-nodo asociado. Una vez localizado el i-nodo toda la información del archivo o directorio puede ser localizada a partir de éste.

## Estructura del sistema de archivos en una partición de disco

Un Sistema de archivos no es más que una **estructura de almacenamiento normalizado de archivos y directorios en un volumen de almacenamiento de datos asociado a un dispositivo de bloques** (disco duro, pendrive, unidad óptica, dispositivo SSD, etc.).

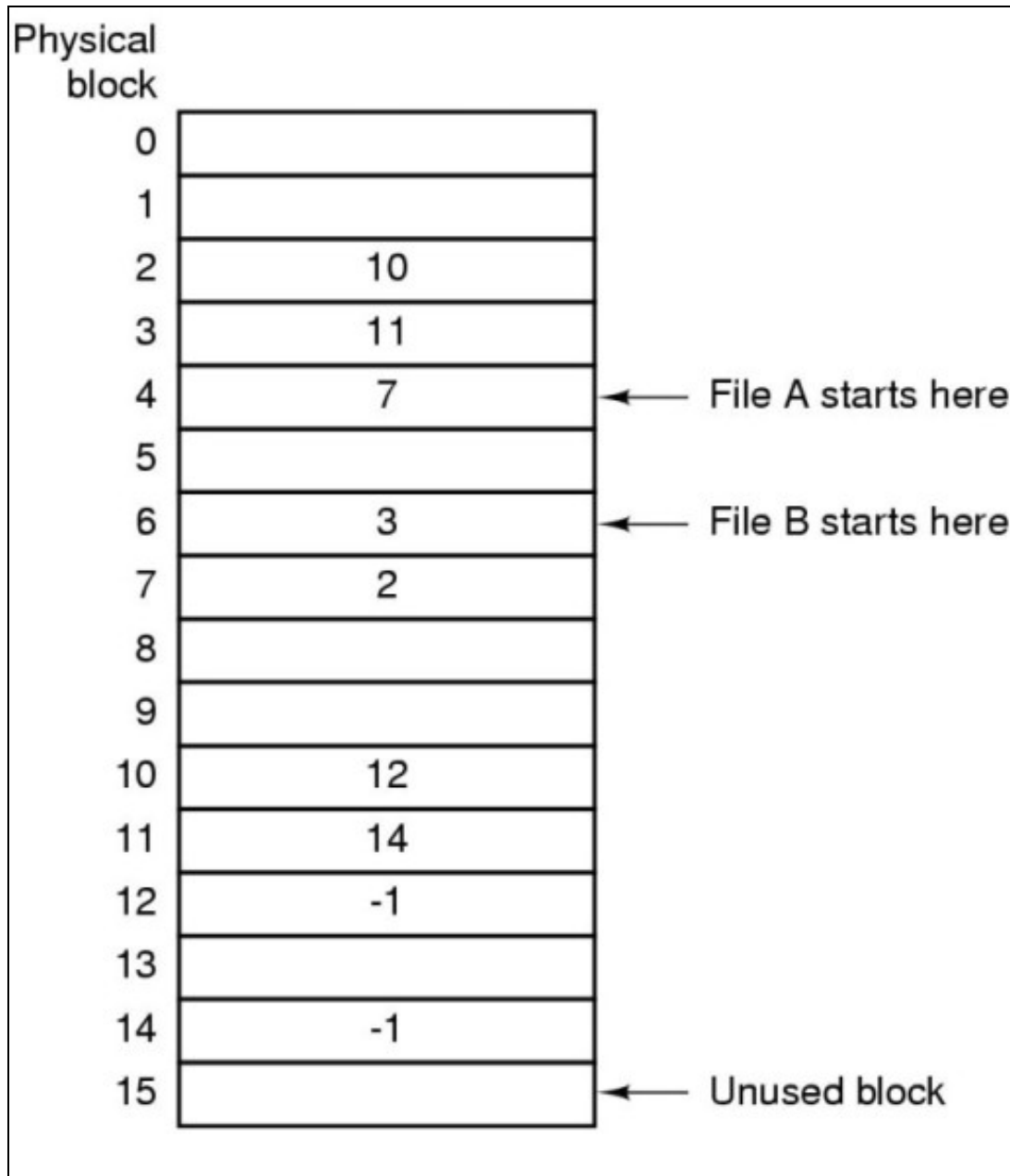
Para poder desplegar un sistema de archivos necesitamos una partición dentro de un dispositivo de bloques. Dentro de la partición, durante el proceso de formateo del sistema de archivos, se crearán todas las estructuras de datos necesarias para dar soporte al almacenamiento de archivos, como por ejemplo:

- **Superbloque:** Zona en la que se indica el tipo y datos del propio sistema de archivos
- **Zona de gestión del espacio libre:** en el que se almacena por ejemplo el mapa de bits
- **Bloque de arranque,** utilizado para arrancar el sistema operativo, en caso de que sea un sistema de archivos asociado a un sistema operativo
- **Zona para almacenamiento de i-nodos**
- **Zona de almacenamiento de archivos**



## FAT y NTFS

En el caso de Windows, que utiliza actualmente **NTFS** como único Sistema de Archivos soportado, la orientación es ligeramente diferente. En este caso los directorios contienen parte de la información de metadatos del archivo. En el caso de **FAT** (File Allocation Table), ver entrada de directorio de figura anterior, se almacenan, además del nombre y extensión, otros atributos, como: fechas, tamaño... y un apuntador al primer bloque del archivo en el disco. A partir de este número de bloque, mediante una tabla ubicada en RAM, la FAT, se pueden encontrar el resto de bloques que constituyen el archivo, simplemente siguiendo una cadena de enlaces definidos en la tabla FAT.



NTFS no utiliza una tabla FAT, ya que éstas tienen el inconveniente de que ocupan en memoria un espacio proporcional al número de bloques lógicos del disco. Si el disco es muy grande, o el tamaño de bloque muy pequeño, la tabla FAT puede utilizar cantidades ingentes de memoria en RAM. Otro inconveniente es que las búsquedas de bloques tienen que seguir una cadena secuencial de localización en la FAT, por lo que es un sistema lento en general. Por estos motivos las FAT se consideran obsoletas y poco fiables.

NTFS utiliza otra estructura para gestionar la ubicación de archivos y directorios en disco y sus metadatos. Esa estructura es la **MFT** (Master File Table), la cual es compleja, aunque ofrece características de gestión del Sistema de Archivos muy potentes:

- **Transaccionalidad (journaling)**
- **Soporte para ACL (Listas de Control de Acceso)**
- **Cifrado del Sistema de Archivos**
- **Cuotas**

## Características avanzadas de Sistemas de Archivos

- **Transaccionalidad (journaling):** Esta característica, cuando está soportada, permite utilizar las operaciones del Sistema de Archivos, como creación, borrado y actualización de los metadatos de archivos y directorios, como si se tratara de operaciones transaccionales. Esto quiere decir que una operación de esta naturaleza no se confirma hasta que esté completamente terminada y, en caso de que ocurra algún error durante su ejecución, el Sistema de Archivos revierte su estado al momento inmediatamente anterior al inicio de la operación. En este aspecto son similares a los procesos de control transaccional de las bases de datos.

Para implementar esta característica se utiliza un archivo de **journal (bitácora)**, en el que se van anotando las operaciones a realizar, eliminándose del mismo cuando esta se complete. Si ocurre un problema y el sistema, por ejemplo, se reinicia, habrá anotaciones en la bitácora que indicarán aquellas operaciones que han quedado sin terminar en el momento de la falla. De este modo el sistema puede revertir, deshacer, esas operaciones, dejando al Sistema de Archivos en un estado consistente.

- **Cifrado:** Consistente en la posibilidad de cifrar el contenido del Sistema de Archivos para que solamente el usuario propietario de los archivos pueda acceder a su contenido al disponer de la clave de cifrado/descifrado correspondiente.
- **Compresión:** Capacidad de almacenar los archivos y directorios comprimidos en el dispositivo de almacenamiento. Esta característica, aunque es transparente para el usuario, suele empeorar el rendimiento del Sistema de Archivos en las operaciones de lectura y escritura.

## Sistemas de Archivos en Sistemas Operativos actuales

### FAT

Sistema de Archivos de MS-DOS y versiones de Windows no basadas en tecnología NT. Utiliza una tabla FAT para acceder a los bloques de archivo, lo que lo hace lento e ineficiente. Además no posee control transaccional del Sistema de Archivos y está limitado en cada versión por el número de bits de su arquitectura (FAT16, FAT32...). Sin embargo, debido a su sencillez y amplia difusión, sobrevive como Sistema de Archivos compatible con todas las arquitecturas de sistema operativo y por ser utilizado también en dispositivos de almacenamiento externo y multimedia.

### NTFS

Sistema de Archivos de las versiones de Windows NT. Ha evolucionado a lo largo del tiempo y de sus versiones. Es un sistema de gestión de archivos y directorios muy avanzado y con amplio soporte de características

- Transaccionalidad
- Cifrado
- Compresión
- Gestión avanzada de permisos
- Cuotas

Para discos muy pequeños (por debajo de unos GB de capacidad) no es recomendable, debido a que las estructuras de control de NTFS (principalmente la MFT) ocupan bastante espacio en el disco

### ext2, ext3 y ext4

Diferentes versiones del Sistema de Archivos ext (extended Filesystem)

- **ext2**: rápida, ligera y eficiente. No tiene soporte transaccional, lo que puede suponer un problema para la consistencia del Sistema de Archivos. Se considera obsoleta en general
- **ext3**: derivada de ext2, con soporte para características avanzadas, como la transaccionalidad (journaling) del Sistema de Archivos
- **ext4**: última versión liberada. Deriva de ext3, a la que añade nuevas características y extensión de los límites en el número de archivos y tamaño de los mismos soportado por el Sistema de Archivos

**Otros: reiserfs, xfs, jfs, hfs+**

## Sistemas de Archivos en Red

### NFS

Network File System. Permite compartir directorios en red de modo que sean públicamente accesibles y de un modo independiente al cliente (Linux, Windows, Mac...). Permite compartir en LAN, o bien en WAN (a través de Internet), por tanto la seguridad debe de ser un aspecto importante a considerar. Utiliza arquitectura cliente/servidor, es decir necesitaremos los módulos correspondientes para acceder y/o publicar recursos NFS. Existen versiones libres y gratuitas de clientes y servidores NFS para Linux, Windows y Mac. Es importante considerar que versión de NFS se está utilizando. En la actualidad conviven NFSv3 y NFSv4, siendo incompatibles entre sí.

### FTP

File Transfer Protocol. Protocolo de transferencia de archivos en redes TCP/IP. Fue uno de los primeros protocolos de aplicación implementados en redes TCP/IP. Por este motivo, al ser tan antiguo, presenta características de seguridad obsoletas, que pueden ser solventadas con medidas de seguridad adicionales. Es simple, rápido y existe una amplia base de servidores y clientes disponibles para todas las plataformas.

### Samba

Implementación libre de los protocolos de red SMB/CIFS de Microsoft. Desarrollado para hacer compatibles redes con clientes Windows y Linux. Un servidor Samba puede ser utilizado para compartir directorios e impresoras en redes con clientes Windows y Linux. También puede utilizarse como Controlador de Dominio de red, al estilo de los Controladores de Dominio de Microsoft (NT, o Active Directory). Su administración es compleja y requiere profundos conocimientos de redes y sistemas. Actualmente conviven las versiones Samba 3 y Samba 4, siendo esta última compatible con características de los dominios Active Directory, que hasta el momento no estaban soportadas.

## Herramientas

### Recopilación de información del Sistema de Archivos

En Windows disponemos de herramientas integradas para acceder a características y estadísticas de uso del Sistema de Archivos. Bien desde el Panel de Control, o bien, desde el propio explorador de archivos y las Herramientas Administrativas del Sistema.

En Linux existen también herramientas en modo gráfico y modo consola. Algunos comandos relacionados:

- **df**: Obtención del espacio libre de un sistema de archivos
- **du**: Obtención del espacio utilizado por archivos y directorios del sistema de archivos
- **fsck**: File System Check, chequeo del sistema de archivos

### Creación de Sistemas de Archivos

De nuevo existen herramientas de este tipo integradas en todos los sistemas. Desde el **Administrador de Discos** de Windows, el cual permite gestionar las particiones de los discos y creación de Sistemas de Archivos NTFS y FAT. En Linux tenemos, en modo gráfico y consola, tanto en distribuciones instaladas como **LiveCD**, herramientas de creación de Sistemas de Archivos. Algunos comandos relacionados:

- **fdisk**: Creación y gestión de particiones
- **cgdisk**: Creación y gestión de particiones, permite gestionar particiones en formato GPT, algo que fdisk no permite
- **mkfs**: Creación de sistemas de archivos

### Herramientas de monitorización, chequeo y reparación del Sistema de Archivos

Pueden instalarse en cualquier sistema herramientas **de terceros**, o bien utilizar herramientas **integradas**, para obtener información sobre el estado de uso y salud del Sistema de Archivos y los dispositivos que lo soportan. Es importante controlar el estado de uso de recursos del Sistema de Archivos y

su salud, es decir, si el soporte de almacenamiento, como por ejemplo un disco duro, está en buen estado y no se han registrado errores de funcionamiento, que podrían indicar un próximo punto de falla.

Existen compilaciones de software, como por ejemplo **HirensBoot**, o una LiveCD específica, así como herramientas de fabricantes de dispositivos que permiten hacer comprobaciones exhaustivas del estado y prestaciones del mismo.

Ilustraciones de A.S. Tanenbaum distribuídas con licencia [Creative Commons](#)

[Volver](#)

JavierFP 16:31 03 dec 2018 (GET)