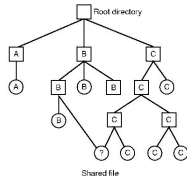


# SISTEMAS OPERATIVOS

# Sistema de Archivos



Mg. Samuel Oporto Díaz

## Gestión de archivos y directorios

- **Objetivos:**
  - Que el alumno entienda los conceptos de fichero y directorio
  - Mostrar los métodos de acceso y los mecanismos de protección
  - Estudiar las semánticas de compartición
  - Que el alumno comprenda la estructura del sistema de ficheros
  - Presentar las llamadas al sistema y ejemplos de programación en LINUX y Windows
  - Presentar distintas técnicas de gestión de archivos y directorios a nivel de diseño.

1 / 57

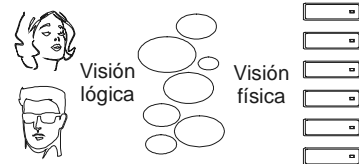
## Contenido

- Visión del usuario del sistema de archivos
- Archivos
- Directorios
- Servicios de archivos
- Servicios de directorios
- Sistemas de archivos
- El servidor de archivos
- Puntos a recordar

2 / 57

## Visión del usuario

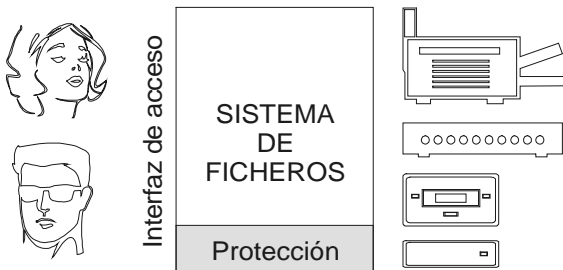
- Visión lógica:
  - Archivos
  - Directorios
  - Sistemas de archivos y particiones
- Visión física:
  - Bloques o bytes ubicados en dispositivos



3 / 57

## Función principal

- El SF establece una correspondencia entre los archivos y los dispositivos lógicos.



4 / 57

## Características para el usuario

- Almacenamiento permanentes de información. No desaparecen aunque se apague el computador.
- Conjunto de información estructurada de forma lógica según criterios de aplicación.
- Nombres lógicos y estructurados.
- No están ligados al ciclo de vida de una aplicación particular.
- Abstraen los dispositivos de almacenamiento físico.
- Se acceden a través de llamadas al sistema operativo o de bibliotecas de utilidades.

5 / 57

## Sistema de archivos

- El acceso a los dispositivos es:
  - Incómodo
    - Detalles físicos de los dispositivos
    - Dependiente de las direcciones físicas
  - No seguro
    - Si el usuario accede a nivel físico no tiene restricciones
- El sistema de archivos es la capa de software entre dispositivos y usuarios.
- Objetivos:
  - Suministrar una visión lógica de los dispositivos
  - Ofrecer primitivas de acceso cómodas e independientes de los detalles físicos
  - Mecanismos de protección

6 / 57

## Contenido

- Visión del usuario del sistema de archivos
- Archivos
- Directorios
- Servicios de archivos
- Servicios de directorios
- Sistemas de archivos
- El servidor de archivos
- Puntos a recordar

7 / 57

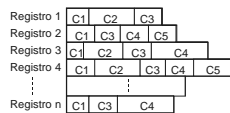
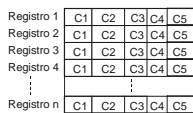
## Archivos: visión lógica

- Conjunto de información relacionada que ha sido definida por su creador
- Estructura de un archivo:
  - Secuencia o tira de bytes (UNIX, POSIX)



Posición

- Registros (de tamaño fijo o variable)



8 / 57

## Contenido

- Visión del usuario del sistema de archivos
- Archivos
- Directorios
- Servicios de archivos
- Servicios de directorios
- Sistemas de archivos
- El servidor de archivos
- Puntos a recordar

9 / 57

## Concepto de archivo

- Un espacio lógico de direcciones contiguas usado para almacenar datos
- Tipos de archivos:
  - Datos:
    - numéricos
    - carácter
    - binarios
  - Programas:
    - código fuente
    - archivos objetos (imagen de carga)
  - Documentos

10 / 57

## Atributos del archivo

- **Nombre:** la única información en formato legible por una persona.
- **Identificación única del archivo y del usuario:** descriptor interno del archivo, dueño y grupo del archivo
- **Tipo de archivo:** necesario en sistemas que proporciona distintos formatos de archivos.
- **Tamaño del archivo:** número de bytes en el archivo, máximo tamaño posible, etc.
- **Protección:** control de accesos y de las operaciones sobre archivos
- **Información temporal:** de creación, de acceso, de modificación, etc.
- **Información de control:** archivo oculto, de sistema, normal o directorio, etc.

11 / 57

## Representación del archivo

- La información relacionada con el archivo se mantiene en el descriptor del archivo, al que se apunta desde los directorios.
- Es distinto en cada sistema operativo: nodo-i, registro Windows, etc.
- Tipos de archivos:
  - Archivos normales: ASCII y binarios.
  - Archivos especiales: de bloques y de caracteres
- Atributos de un archivo:
  - Nombre
  - Tipo
  - Dueño y grupo
  - Información de protección
  - Tamaño real
  - Hora y fecha de creación
  - Hora y fecha del último acceso
  - Hora y fecha de la última modificación
  - Número de enlaces

12 /57

## Ejemplos de representación

Tipo de archivo y Protección	
Número de Nombres	
Propietario	
Grupo del Propietario	
Tamaño	
Instante de creación	
Instante del último acceso	
Instante de la última modificación	
Puntero a bloque de datos 0	
Puntero a bloque de datos 1	
Puntero a bloque de datos 9	
Puntero indirecto simple	
Puntero indirecto doble	
Puntero indirecto triple	

Nombre
Atrib.
Size KB
Agrup. FAT

Entrada de directorio de MS-DOS

cabecera	
Atributos	
Tamaño	
Nombre	
Seguridad	
Datos	
Vclusters	

Registro MFT de WINDOWS-NT

Nodo-i de UNIX

13 /57

## Nombres de fichero y extensiones I

- Muy importante para los usuarios. Es característico de cada sistema de archivos.
- Problema: usar nombre lógicos basados en tiras de caracteres.
- Motivo: los usuarios no recuerdan el nombre 001223407654
- Tipo y longitud cambian de un sistema a otro
  - Longitud: fija en MS-DOS o variable en UNIX
  - Extensión: obligatoria o no, más de una o no, fija para cada tipo de archivos, etc.
- Sensibles a tipografía. Ejemplo: CATALINA y catalina son el mismo archivo en Windows pero distintos en LINUX.
- El sistema de ficheros trabaja con descriptores internos, sólo distingue algunos formatos (ejecutables, texto, ...). Ejemplo: número mágico UNIX.

14 /57

## Nombres de fichero y extensiones II

- Los directorios relacionan nombres lógicos y descriptores internos de ficheros
- Las extensiones son significativas para las aplicaciones (html, c, cpp, etc.)

Nombre	Size	Type	Modified
My Pictures		File Folder	07/09/2000 11:36
My Webs		File Folder	06/09/2000 11:57
post-inf_files		File Folder	14/09/2000 16:21
awmrc	13 KB	PRIVATE File	06/05/1999 18:00
cartacas.tex	1 KB	COREL Texture	06/05/1999 17:55
catag99.ps	103 KB	PS File	06/05/1999 17:55
control.lab	16 KB	SBF File	06/05/1999 17:55
Faring.log	4 KB	Text Document	06/05/1999 17:55
fig-1.tif	734 KB	COREL PHOTO-PAINT...	22/08/2000 11:59
fig-7.cdr	27 KB	CDR File	03/05/2000 18:27
post-inf.doc	53 KB	Microsoft Word Doc...	14/09/2000 16:21
post-inf.htm	1 KB	Microsoft HTML Doc...	14/09/2000 9:51
remain.zip	0 KB	WinZip File	30/05/2000 12:34
Sample.spl	10 KB	COREL PHOTO-PAINT...	05/09/2000 17:08
winamp265.exe	2.112 KB	Application	07/09/2000 13:10
cinutec.cpp	3 KB	CPP File	11/07/2000 15:30
acobject.c	2 KB	C File	14/07/2000 12:52
adsmaspkg.adb	13 KB	ADB File	24/02/2000 9:49
Demo.ppt	345 KB	Microsoft PowerPoint...	24/07/1998 8:15
collec4.html	10 KB	Microsoft HTML Doc...	22/12/1999 11:28
monocolor.pdf	4.110 KB	Adobe Acrobat Doc...	07/04/1999 11:11

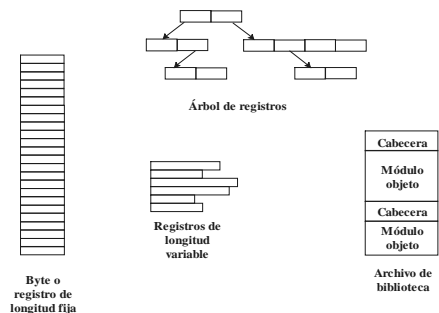
15 /57

## Estructura del fichero

- Ninguna - secuencia de palabras o bytes (UNIX)
- Estructura sencilla de registros
  - Líneas
  - Longitud fija
  - Longitud variable
- Estructuras complejas
  - Documentos con formato (HTML, postscript)
  - Fichero de carga reubicable (módulo de carga)
- Se puede simular estructuras de registro y complejas con una estructura plana y secuencias de control
- ¿Quién decide la estructura?
  - Interna: El sistema operativo
  - Externa: Las aplicaciones

16 /57

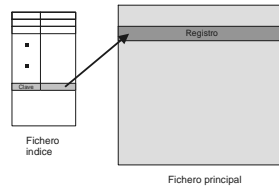
## Distintas estructuras lógicas



17 /57

## Archivos: visión lógica

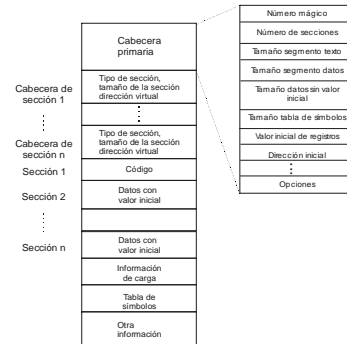
- Estructura de un archivo:
  - Archivos de estructura compleja
    - Archivos indexados
    - Archivos directos o de dispersión
  - Ejemplo de archivo indexado:



- Los archivos estructurados en registros y los archivos con estructuras complejas se pueden construir como una capa sobre la visión de tira de bytes.

18 / 57

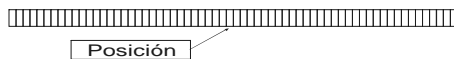
## Estructura de archivo ejecutable LINUX



19 / 57

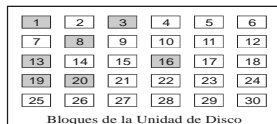
## Archivos: visión lógica y física

- Usuario: Visión lógica.
- Sistema operativo: visión física ligada a dispositivos. Conjunto de bloques.



Visión lógica

Archivo A  
Bloques: 13  
20  
1  
8  
3  
16  
19



Bloques de la Unidad de Disco

Visión física

20 / 57

## Archivos: visión física

- Bloque
  - Unidad de transferencia
  - 2<sup>n</sup> sectores
  - Parámetro fijo por sistema de archivos
- Agrupación
  - Unidad de asignación
  - 2<sup>p</sup> bloques
  - Aumenta la secuencialidad del archivo
- Descripción de la estructura física:
  - Bloques utilizados

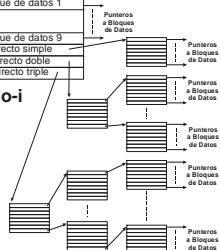
21 / 57

## Descripción física en UNIX (nodo-i)

Tipo de Fichero y Protección
Número de Nombres
Propietario
Grupo del Propietario
Tamaño
Instante de creación
Instante de la última modificación
Puntero a bloque de datos 0
Puntero a bloque de datos 1
Puntero a bloque de datos 2
Puntero indirecto simple
Puntero indirecto doble
Puntero indirecto triple

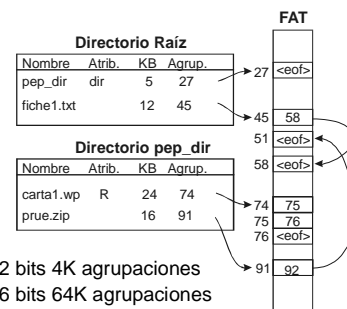
nodo-i

Tamaño máximo del archivo:  
 $10Sb + (Sb/4)Sb + (Sb/4)^2Sb + (Sb/4)^3Sb$   
 Sb el tamaño del bloque y direcciones de bloques de 4 bytes.



22 / 57

## Descripción física en MS-DOS (FAT)



- FAT de 12 bits 4K agrupaciones
- FAT de 16 bits 64K agrupaciones

23 / 57

## Métodos de Acceso

- Acceso secuencial: lectura de los bytes del archivo en orden ascendente, empezando por el principio.
  - read next, write next, reset, no read after last write, ...
  - rewind: ir al principio para buscar hacia delante
  - Lectura -> posición = posición + datos leídos
  - Dispositivos de cinta
  - ISAM: método de acceso secuencial indexado
- Acceso Directo:
  - read n, write n, goto n, rewrite n, read next, write next, ...
  - n = número de bloque relativo al origen
  - Dispositivos: discos magnéticos

24 /57

## Archivos: semántica de coutilización

- Cualquier forma de acceso tiene problemas cuando varios usuarios trabajan con el archivo simultáneamente.
- Semántica de coutilización: especifica el efecto de varios procesos accediendo de forma simultánea al mismo archivo y cuando se hacen efectivas las modificaciones.
- Tipos de semánticas:
  - Semántica UNIX (POSIX)
    - Las escrituras son inmediatamente visibles para todos los procesos con el archivo abierto.
    - Los procesos pueden compartir archivos. Si existe relación de parentesco pueden compartir el puntero. La coutilización afecta también a los metadatos.

25 /57

## Archivos: semántica de coutilización (II)

- Semántica de sesión
  - Las escrituras que hace un proceso no son inmediatamente visibles para los demás procesos con el archivo abierto.
  - Cuando se cierra el archivo los cambios se hacen visibles para las futuras sesiones.
  - Un archivo puede asociarse temporalmente a varias imágenes.
- Semántica de versiones
  - Las actualizaciones se hacen sobre copias con nº versión.
  - Sólo son visibles cuando se consolidan versiones.
  - Sincronización explícita si se requiere actualización inmediata
- Semántica de archivos inmutables
  - Una vez creado el archivo sólo puede ser compartido para lectura y no cambia nunca

26 /57

## Contenido

- Visión del usuario del sistema de archivos
- Archivos
- Directorios
  - Servicios de archivos
  - Servicios de directorios
  - Sistemas de archivos
  - El servidor de archivos
- Puntos a recordar

27 /57

## Concepto de directorio

- Objeto que relaciona de forma unívoca un nombre de usuario de archivo con su descriptor interno
- Organizan y proporcionan información sobre la estructuración de los sistemas de archivos
- Una colección de nodos que contienen información acerca de los archivos

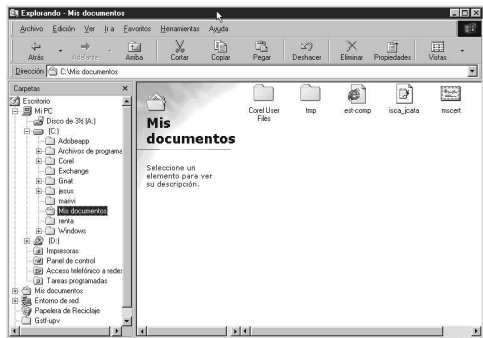
28 /57

## Directorios: visión lógica

- Esquema jerárquico.
- Cuando se abre un archivo el SO busca el nombre en la estructura de directorios.
- Operaciones sobre un directorio:
  - Crear (insertar) y borrar (eliminar) directorios.
  - Abrir y cerrar directorios.
  - Renombrar directorios.
  - Leer entradas de un directorio.
  - Montar (combinar)
- La organización jerárquica de un directorio
  - Simplifica el nombrado de archivos (nombres únicos)
  - Proporciona una gestión de la distribución => agrupar archivos de forma lógica (mismo usuario, misma aplicación)

29 /57

## Ejemplo: explorador de Windows



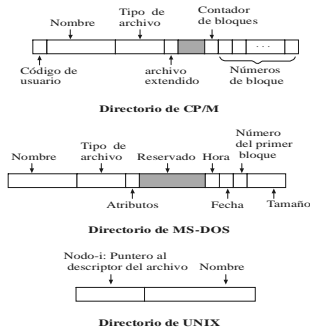
30 / 57

## Estructura de los directorios

- Tanto la estructura del directorio como los archivos residen en discos
- Los directorios se suelen implementar como archivos
- Copias de respaldo en cintas, por seguridad
- Información en un directorio: nombre, tipo, dirección, longitud máxima y actual, tiempos de acceso y modificación, dueño, etc.
- Hay estructuras de directorio muy distintas. La información depende de esa estructura.
- Dos alternativas principales:
  - Almacenar atributos de archivo en entrada directorio
  - Almacenar <nombre, identificador>, con datos archivo en una estructura distinta. Esta es mejor.

31 / 57

## Ejemplo de entradas de directorio



32 / 57

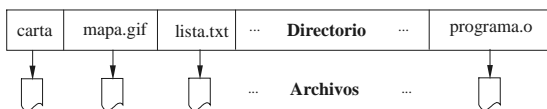
## Organización del directorio

- Eficiencia: localizar un archivo rápidamente
- Nombrado: conveniente y sencillo para los usuarios
  - Dos usuarios pueden tener el mismo nombre para archivos distintos
  - Los mismos archivos pueden tener nombres distintos
  - Nombres de longitud variable
- Agrupación: agrupación lógica de los archivos según sus propiedades (por ejemplo: programas Pascal, juegos, etc.)
- Estructurado: operaciones claramente definidas y ocultación
- Sencillo: la entrada de directorio debe ser lo más sencilla posible.

33 / 57

## Directorio de un nivel

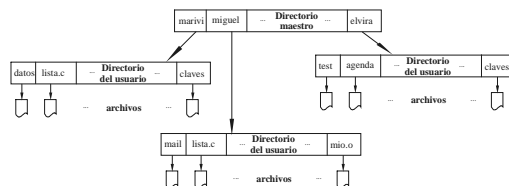
- Un único directorio para todos los usuarios
- Problemas de nombrado y agrupación



34 / 57

## Directorio de dos niveles

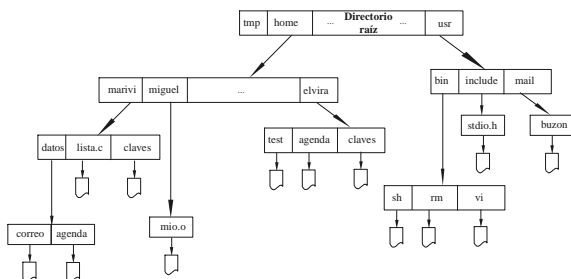
- Un directorio por cada usuario
- Camino de acceso automático o manual
- El mismo nombre de archivo para varios usuarios
- Búsqueda eficiente, pero problemas de agrupación



35 / 57

## Directorio con estructura de árbol

- Búsqueda eficiente y agrupación
- Nombres relativos y absolutos -> directorio de trabajo



36 / 57

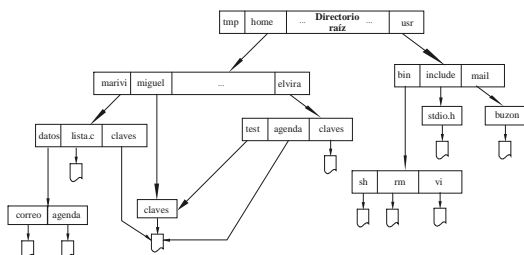
## Directorio con estructura de árbol

- Los nombres absolutos contienen todo el camino
- Los nombres relativos parten del directorio de trabajo o actual
- Cambio de directorio:
  - `cd /spell/mail/prog`
  - `cd prog`
- Borrar un archivo: `rm <nombre-archivo>`
- Crear un subdirectorio: `mkdir <nombre_dir>`
- Ejemplo:
  - `cd /spell/mail`
  - `mkdir count`
  - `ls /spell/mail/count`
- Borrar un subdirectorio: `rm -r mail`

37 / 57

## Directorio de grafo acíclico I

- Tienen archivos y subdirectorios compartidos
- Este concepto no existe en Windows



38 / 57

## Directorio de grafo acíclico II

- **link:** Un archivo con varios nombres -> control de enlaces
  - un único archivo con contador enlaces en descriptor (e. Físicos)
  - archivos nuevos con el nombre destino dentro (e. simbólicos)
- Borrado de enlaces:
  - a) decrementar contador; si 0 borrar archivo
  - b) recorrer los enlaces y borrar todos
  - c) borrar únicamente el enlace y dejar los demás
- Problema grave: existencia de bucles en el árbol. Soluciones:
  - Permitir sólo enlaces a archivos, no subdirectorios
  - Algoritmo de búsqueda de bucle cuando se hace un enlace
- Limitación de implementación en UNIX: sólo enlaces físicos dentro del mismo sistema de archivos.

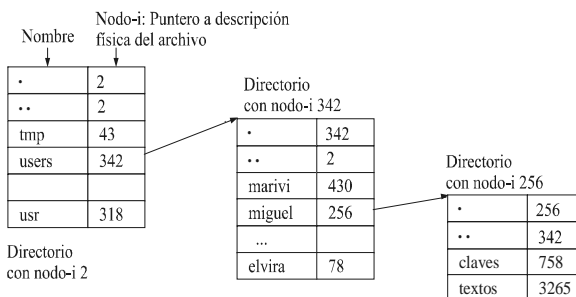
39 / 57

## Nombres jerárquicos

- Nombre absoluto: especificación del nombre respecto a la raíz (/ en LINUX, \ en Windows).
- Nombre relativo: especificación del nombre respecto a un directorio distinto del raíz
  - Ejemplo: (Estamos en /users/) miguel/claves
  - Relativos al dir. de trabajo o actual: aquel en el se está al indicar el nombre relativo. En Linux se obtiene con `pwd`
- Directorios especiales:
  - . Directorio de trabajo. Ejemplo: `cp / users/miguel/claves .`
  - .. Directorio *padre*. Ejemplo: `ls ..`
  - Directorio HOME: el directorio base del usuario

40 / 57

## Interpretación de nombres en LINUX. I



41 / 57

## Interpretación de nombres en LINUX. II

- Interpretar `/users/miguel/claves`
  - Traer a memoria entradas archivo con nodo-i 2
  - Se busca dentro `users` y se obtiene el nodo-i 342
  - Traer a memoria entradas archivo con nodo-i 342
  - Se busca dentro `miguel` y se obtiene el nodo-i 256
  - Traer a memoria entradas archivo con nodo-i 256
  - Se busca dentro `claves` y se obtiene el nodo-i 758
  - Se lee el nodo-i 758 y ya se tienen los datos del archivo
- ¿Cuándo parar?
  - Se ha encontrado el nodo-i del archivo
  - No se ha encontrado y no hay más subdirectorios
  - Estamos en un directorio y no contiene la siguiente componente del nombre (por ejemplo, `miguel`).

42 /57

## Jerarquía de directorios

- ¿Árbol único de directorios?
  - Por dispositivo lógico en Windows (`c:\users\miguel\claves`, `j:\pepe\tmp`, ...)
  - Para todo el sistema en UNIX (`/users/miguel/claves`, `/pepe/tmp`, ...).
- Hacen falta servicios para construir la jerarquía: `mount` y `umount`.
  - `mount /dev/hda /users`
  - `umount /users`
- Ventajas: imagen única del sistema y ocultan el tipo de dispositivo
- Desventajas: complican la traducción de nombres, problemas para enlaces físicos entre archivos

43 /57

## Sistemas de archivos y particiones

- Volumen: conjunto coherente de metainformación y datos.
- Ejemplos de Sistemas de archivos:

### MS-DOS

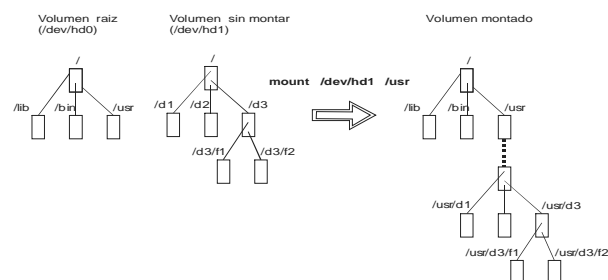


### UNIX



44 /57

## Montado de Sistemas de archivos o particiones



45 /57

## Contenido

- Visión del usuario del sistema de archivos
- Archivos
- Directorios
- Servicios de archivos
- Servicios de directorios
- Sistemas de archivos
  - El servidor de archivos
  - Puntos a recordar

46 /57

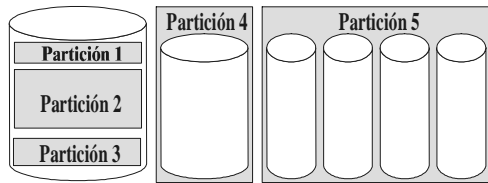
## Sistemas de archivos y particiones

- El sistema de archivos permite organizar la información dentro de los dispositivos de almacenamiento secundario en un formato inteligible para el sistema operativo.
- Previamente a la instalación del sistema de archivos es necesario dividir físicamente, o lógicamente, los discos en **particiones** o **volúmenes** [Pinkert 1989].
- Una **partición** es una *porción de un disco a la que se le dota de una identidad propia y que puede ser manipulada por el sistema operativo como una entidad lógica independiente*.
- Una vez creadas las particiones, el sistema operativo debe crear las estructuras de los sistemas de archivos dentro de esas particiones. Para ello se proporcionan mandatos como `format` o `mkfs` al usuario.
  - `#mkswap -c /dev/hda2 20800`
  - `#mkfs -c /dev/hda3 -b 8196 123100`

47 /57



## Tipos de particiones



48 / 57

## Sistemas de archivos y particiones (II)

- Sistema de archivos: conjunto coherente de metainformación y datos.
- Ejemplos de Sistemas de archivos:

### MS-DOS



### UNIX



49 / 57

## Sistemas de archivos y particiones (III)

- Descripción de sistemas de archivos:
  - El sector de arranque en MS-DOS
  - El superbloque en UNIX
- Relación sistema de archivos-dispositivo:
  - Típico: 1 dispositivo N sistemas de archivos (particiones)
  - Grandes archivos: N dispositivos 1 sistema de archivos
- Típicamente cada dispositivo se divide en una o más particiones (en cada partición sistema de archivos)
- La tabla de particiones indica el principio, el tamaño y el tipo de cada partición.

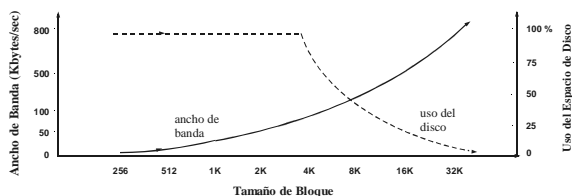
50 / 57

## Bloques y agrupaciones

- **Bloque:** agrupación lógica de sectores de disco y es la unidad de transferencia mínima que usa el sistema de archivos.
  - Optimizar la eficiencia de la entrada/salida de los dispositivos secundarios de almacenamiento.
  - Todos los sistemas operativos proporcionan un tamaño de bloque por defecto.
  - Los usuarios pueden definir el tamaño de bloque a usar dentro de un sistema de archivos mediante el mandato `mkfs`.
- **Agrupación:** conjunto de bloques que se gestionan como una unidad lógica de gestión del almacenamiento.
  - El problema que introducen las agrupaciones, y los bloques grandes, es la existencia de fragmentación interna.

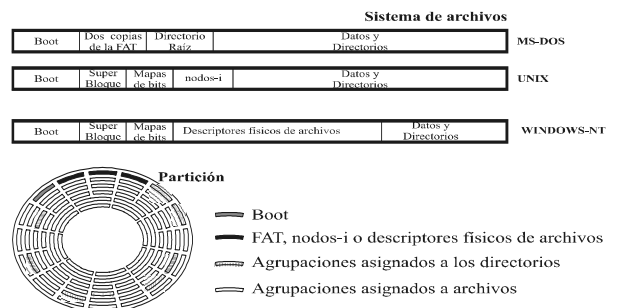
51 / 57

## Tamaño bloque, ancho banda y uso disco



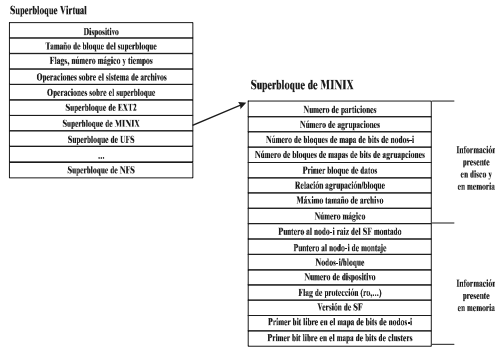
52 / 57

## Estructuras de sistemas de archivos



53 / 57

## Superbloque de sistemas de archivos en LINUX



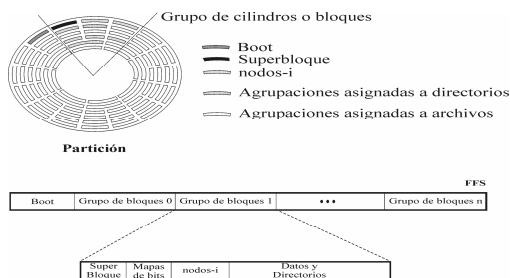
54 / 57

## Otros sistemas de archivos

- Fast File System
- EXT2 (*extended file system*)
- Archivos con bandas
- LFS (*log structured file system*)
- Sistemas de archivos paralelos
- ....

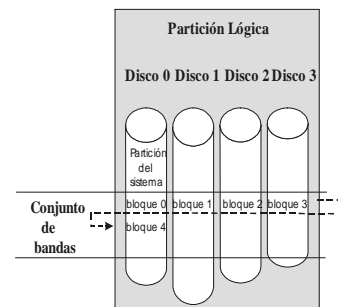
55 / 57

## Estructura del FFS



56 / 57

## Estructura de un sistema de archivos con bandas



57 / 57