

# Actividad X04.

## Montaje de dispositivos en Linux



*"¿Qué diferencias observas en cuanto a cómo se especifica el nombre con la ruta completa de un mismo fichero almacenado en el dispositivo USB en Windows y en Linux?"*

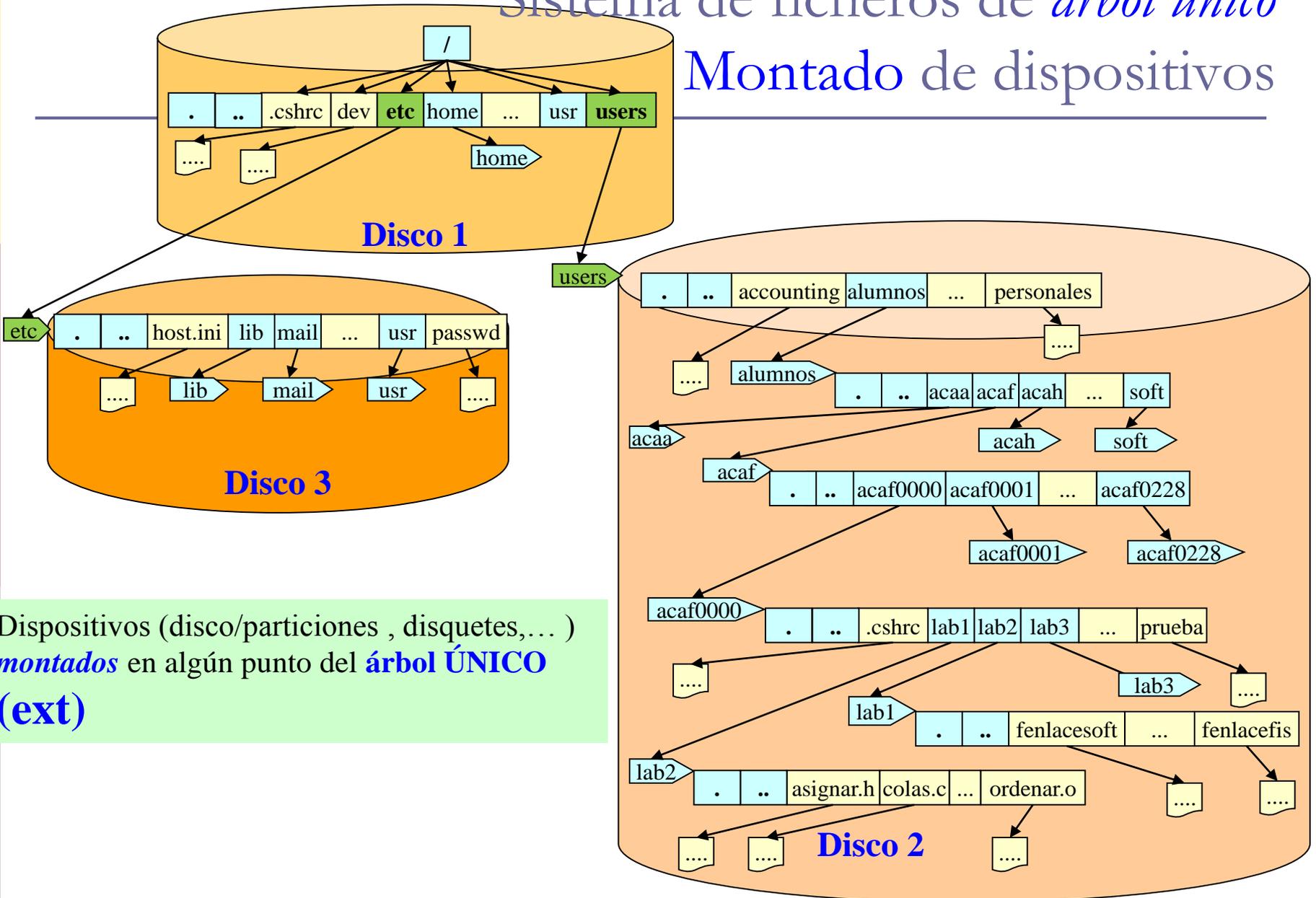
# Los discos y el sistema de ficheros

---

- ❑ La jerarquía de un sistema de ficheros puede componerse de diferentes dispositivos de almacenamiento (Varios discos, CDs, DVDs, ...)
- ❑ Algunos directorios pueden estar situados en un equipo remoto.
- ❑ Existen dos tendencias en la organización de los directorios:
  - Sistemas de **árbol único** (UNIX).
  - Sistemas de **múltiples árboles** (Windows).

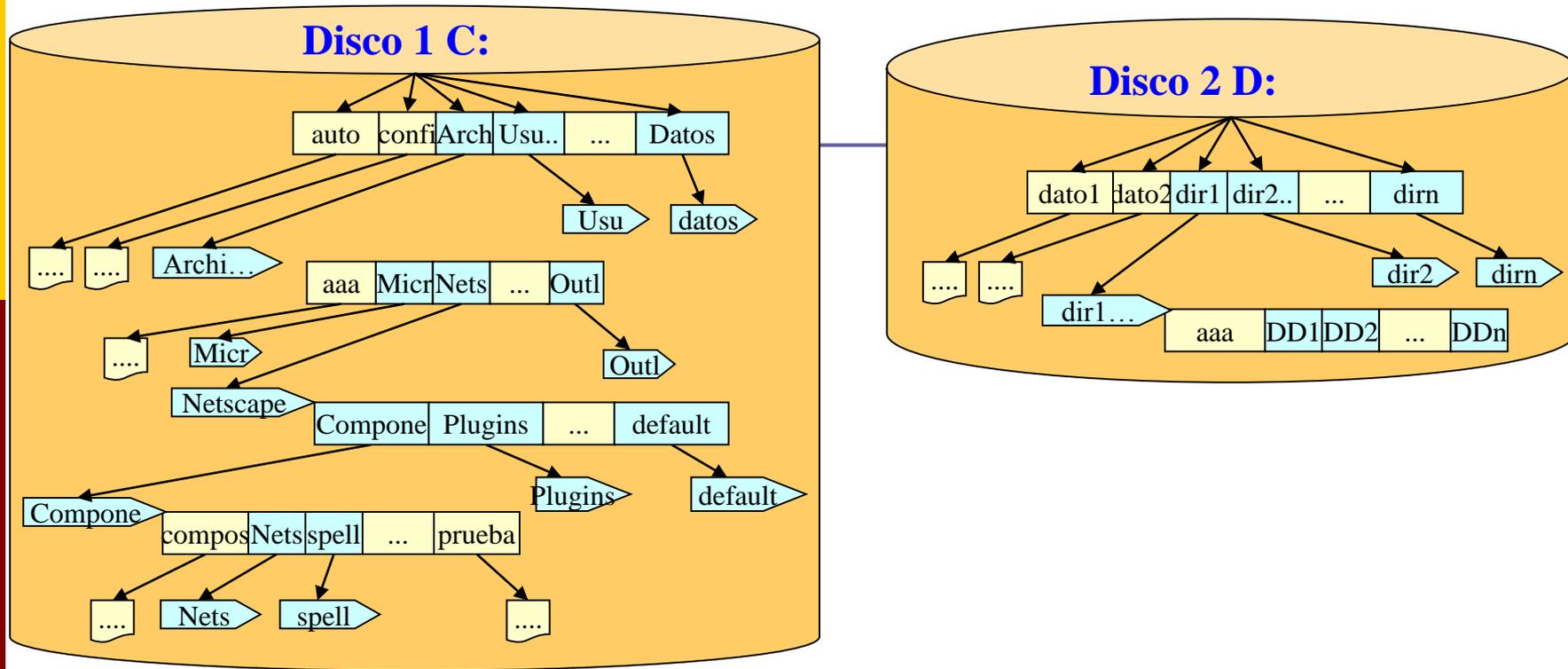
# Sistema de ficheros de *árbol único*

## Montado de dispositivos



Dispositivos (disco/particiones , disquetes,... )  
*montados* en algún punto del **árbol ÚNICO**  
**(ext)**

# Sistema de ficheros de múltiples árboles



Cada dispositivo (disco/particiones , disquetes,... ) tiene su propio árbol

# Visión del Sistema de Ficheros (UNIX)



*Bloques del Disco*

*Inode*

Código encargado de buscar el SO y cargarlo en memoria para inicializarlo

Estado del sistema de ficheros:  
Tamaño, Lista de bloque libres, Tamaño de la lista de Inodes, Lista de Inodes libres,...

Una entrada por cada fichero, con información de control del mismo.

*Bloques de datos del Disco*

Tipo

Fechas  
Nºenlaces  
Tamaño

Índices de  
Bloques Datos

# Sistema de Ficheros Virtual de Linux

---

- Diseñado para soportar diferentes sistemas de gestión de ficheros
- Sistema de Ficheros Virtual (VFS)
  - Representa una única interfaz a los procesos de usuario
  - Define un modelo de ficheros común
  - Supone que los ficheros son objetos con propiedades básicas, independientemente del sistema de ficheros real sobre el que están implementados

# Sistema de Ficheros Virtual de Linux

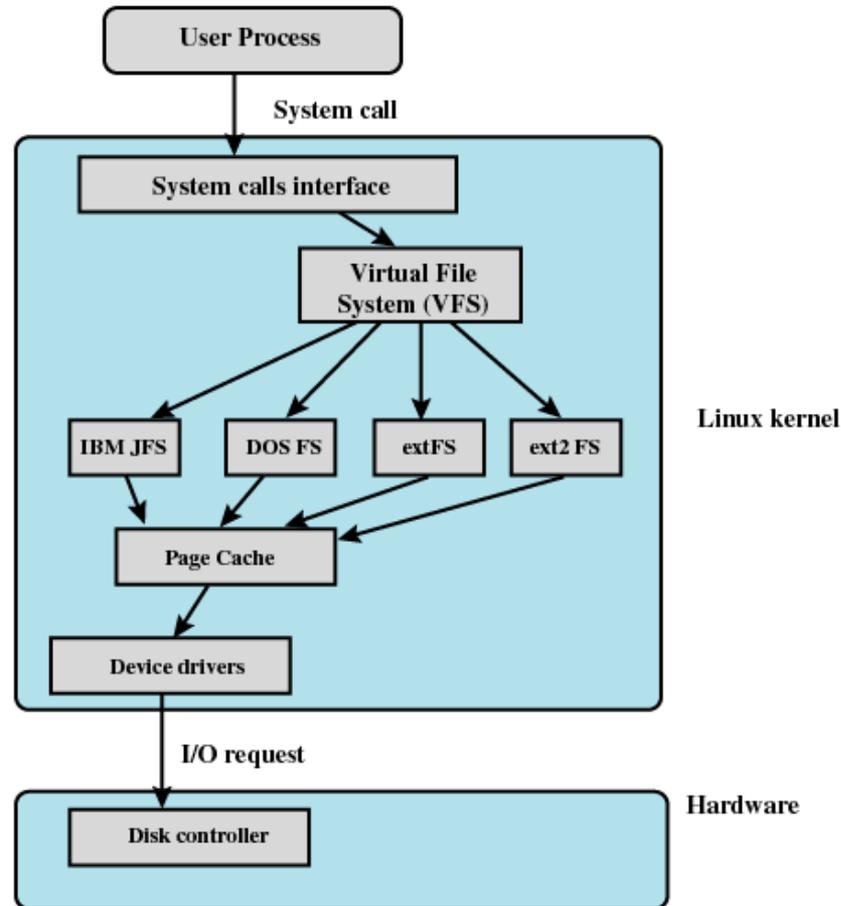
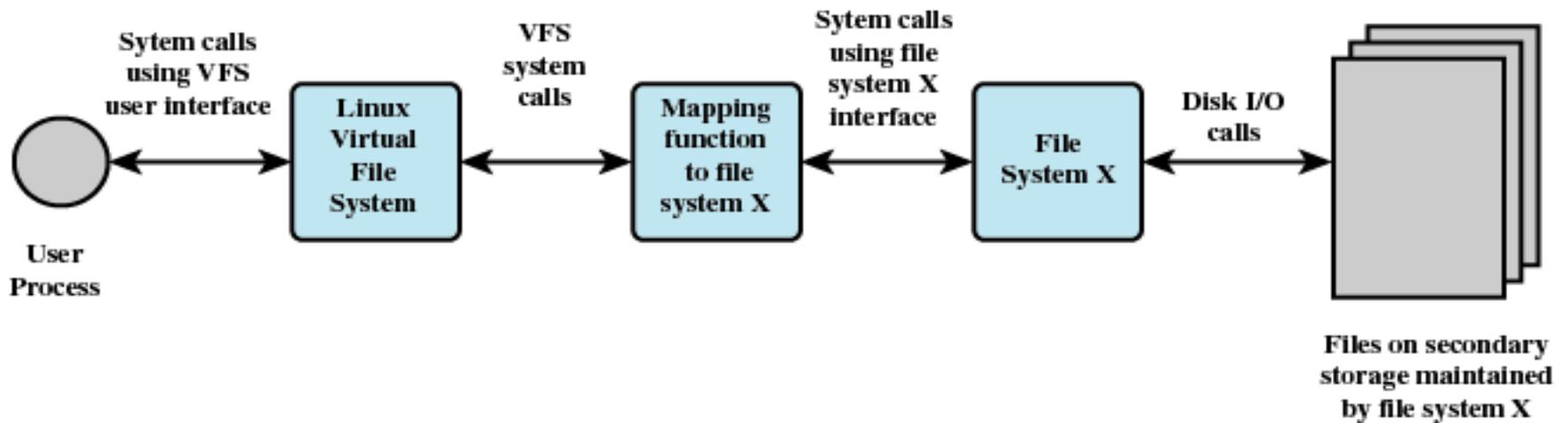


Figure 12.15 Linux Virtual File System Context

Fuente: W. Stallings: Sistemas Operativos. 5º Ed. Pearson Prentice-Hall, 2005

# Sistema de Ficheros Virtual de Linux



**Figure 12.16 Linux Virtual File System Concept**

Fuente: W. Stallings: Sistemas Operativos. 5º Ed. Pearson Prentice-Hall, 2005

# Objetos principales de VFS

---

- ❑ Superbloque (Superblock object)
  - Representa un sistema de ficheros concreto que esta montado
- ❑ Inodo (Inode object)
  - Representa a un determinado fichero
- ❑ Entrada de Directorio (Dentry object)
  - Representa a una entrada de directorio concreta
- ❑ Fichero (File object)
  - Representa a un fichero abierto asociado a un proceso

Fuente: W. Stallings: Sistemas Operativos. 5º Ed. Pearson Prentice-Hall, 2005

# Proyecto P07. Múltiples nombres de los ficheros y su borrado



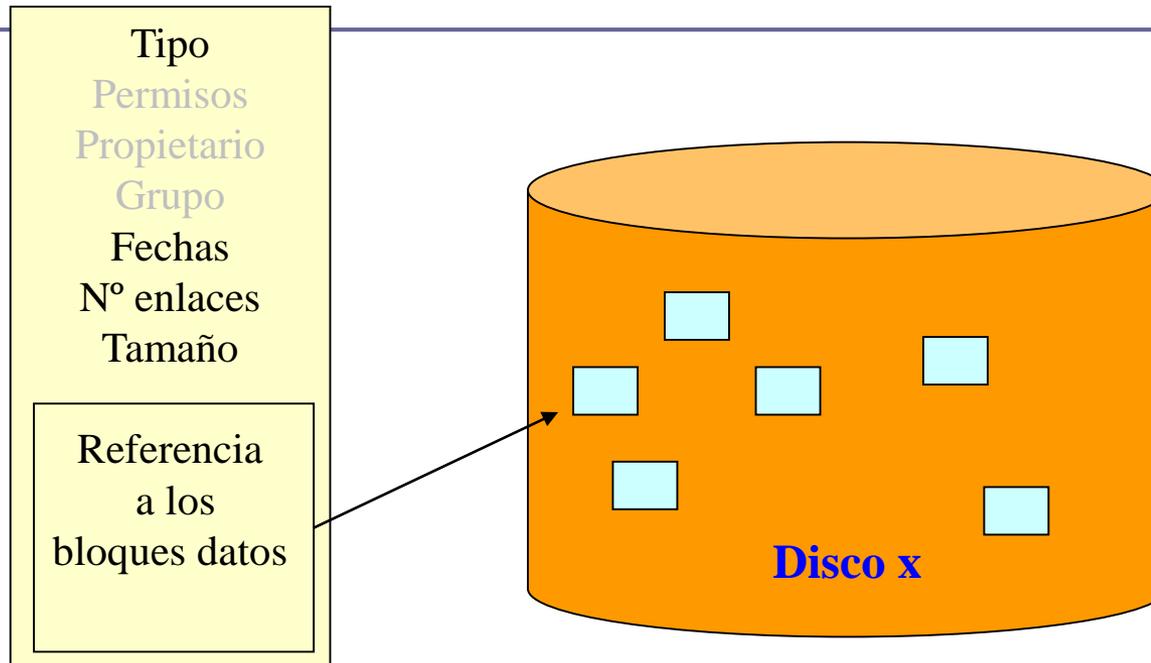
En otros sistemas operativos conocemos una forma de crear un "acceso directo" a un fichero, de esta manera podemos hacer referencia al fichero bien desde su nombre original o desde su "acceso directo".

¿Proporciona Linux algún mecanismo similar?

¿Qué implicaciones tiene en el borrado de los ficheros?

# Información de control de un fichero

## *Inode*



Indica de alguna forma los bloques del dispositivo de almacenamiento donde se encuentran los datos del fichero

# Información de un control de un fichero en Linux

## □ Información de control

- En el **i-node**  
(EXT, ...)

### *Inode*

Tipo
Permisos
Propietario
Grupo
Fechas
<i>Nº enlaces</i>
Tamaño
Bloques Datos

### *Directorio (fichero especial)*

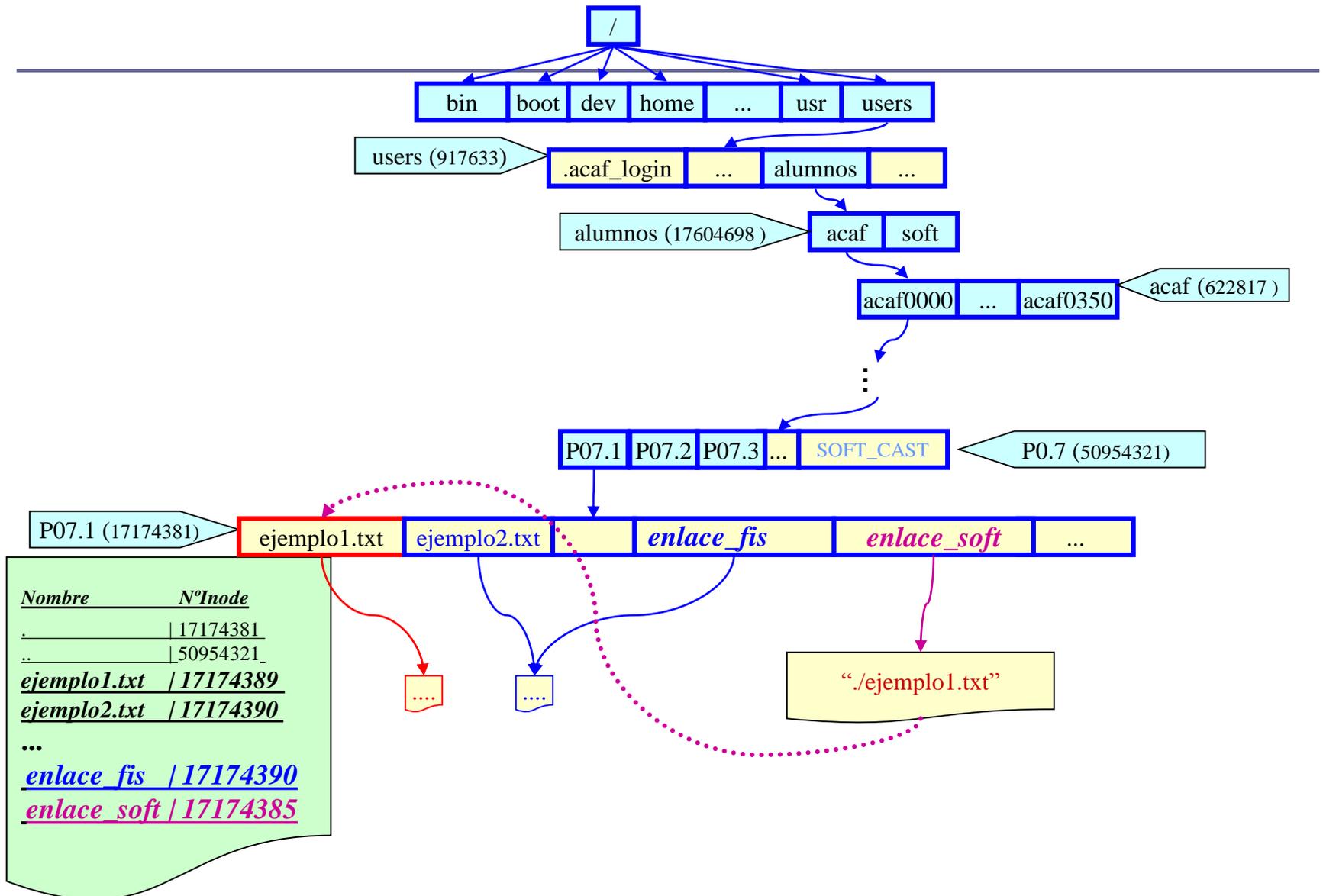
<i>Nombre</i>	<i>/ N°Inode</i>
.	8733108
..	5243238
asignar.h	8733916
colas.c	8733917
colas.h	8733918
...	

¿Proporciona Linux algún mecanismo similar ?  
(a acceso directo a un fichero)

---

- Enlaces **por nombre** (o enlace simbólico)
  - *Equivalente a "acceso directo" de Windows*
  - *Se crea **OTRO FICHERO** especial de tipo (Symbolic Link) que se enlaza (por nombre) al fichero original.*
  
- Enlaces **por nº de i-node** (o enlace físico/hardware)
  - *Se trata de crear "otro nombre" para el **MISMO FICHERO (inode)***
  - *Se crea una nueva entrada en el directorio correspondiente que se relaciona **CON EL MISMO INODE***

# Enlaces: *por nombre* – *por n° de i-node*



# Llamadas al Sistema (UNIX)

## Para gestionar Enlaces

---

### ■ Control de ficheros-directorios

- `int chdir (char *path);`
- `int mkdir (char *path, mode_t mode);`
- `int rmdir (char *path);`

*Crear un enlace físico/hardware/por inode (crear un nuevo nombre para un fichero)*

- `int link (char *path_existente,  
char *path_nuevo);`

*Crear un enlace simbólico/por nombre*

- `int symlink (char *path_existente,  
char *path_nuevo);`

*Borrar un nombre (y a veces sus datos)*

- `int unlink (char *path);`

# Comando(UNIX)

## Para gestionar Enlaces

---

### □ **Comando ln (crear enlaces)**

**Crea un nuevo enlace hard** (*enlace\_hard*) para **el inode** del fichero *path\_existente*

```
ln path_existente enlace_hard
```

**Crear un enlace simbólico/por nombre** (**para el nombre**)

```
ln -s path_existente enlace_soft
```

**Borrar un nombre (y a veces sus datos)**

```
rm fichero
```

# Organización de directorios/Windows

---

- C:\Archivos de programa      Aplicaciones instaladas
- C:\Temp      ficheros temporales
- C:\Windows      ficheros del sistema, boot y ficheros del sistema
- C:\ Windows \system32      ficheros del sistema, bibliotecas compartidas, ficheros de programas, ficheros de configuración
- C:\ Windows \system32\drivers      más drivers
- C:\ Windows \system32\drivers\etc      más ficheros de configuración
- C:\ Usuarios      Directorios de los usuarios

# Organización de directorios/Unix

---

- En los sistemas UNIX, la organización de directorios habitual es la siguiente:
  - ***/bin*** para los programas y utilidades del usuario (*binary*)
  - ***/lib*** para las librerías
  - ***/usr*** para los programas y ficheros del usuario
  - ***/var*** para contenidos que el sistema actualiza durante su operación
  - ***/tmp*** para ficheros temporales
  - ***/dev*** para los ficheros de los dispositivos
  - ***/mnt*** para disquete, CD-ROM, y en general para todo dispositivo en el que se pueda montar de *forma temporal* un sistema de ficheros ( ***/media*** )
  - ***/etc*** para ficheros de configuración y del sistema
  - ***/sbin*** para utilidades y programas de administración (*system binary files*)
  - ***/home*** para las cuentas del usuario (***suele cambiar- administrador***)
  - ***/root*** para la cuenta del usuario ***administrador***
  - ***/proc*** para la gestión de los recursos del sistema
  - ...