



# Tema 3

Instalación y administración local del sistema

# 3.1 Arranque

- **Sistema básico de E/S (BIOS)**
  - Lo primero que hace la CPU cuando se inicia el sistema es buscar el firmware de la placa (**RAM CMOS**) y ejecutar la **BIOS**.
  - El primer paso es cargar un manejador primario del adaptador gráfico. Las actuales tarjetas gráficas no integradas incluyen su propio firmware.
  - El siguiente paso es realizar el **Autotest de encendido (POST)** para verificar el funcionamiento de los componentes de E/S de la placa base.
  - Después inicializa los controladores de los dispositivos para un funcionamiento general y primario.
  - También se detecta e inicializa nuevos dispositivos **PnP**.
  - La **BIOS** constituye el conjunto de rutinas, implementadas con instrucciones de **16 bits**, que manejan los dispositivos de forma primaria.
  - Durante este proceso la **BIOS** le ofrece la oportunidad de modificar la configuración de varias características accediendo a la **RAM CMOS**.
  - Por último la **BIOS** busca el **código de arranque**.
  - El sistema operativo cargará sus propios manejadores de ámbito más específico sustituyendo los de la **BIOS**.

# 3.1 Arranque



# 3.1 Arranque

PhoenixBIOS Setup - Copyright 1985-2004 Phoenix Technologies Ltd.					
Main	Advanced	Security	TPM State	Physical Presence operations	Boot
<p>Hard Disk Drive: Optical Disk Drive: Optical Disk Drive: Legacy Floppy Drive: [Disabled]</p> <p>▶ Serial Port ▶ Parallel Port ▶ PS/2 Mouse LCD Screen Expansion: [No]</p> <p>Network Boot: [Disabled] External Drive Boot: [Disabled] ▶ Workaround Control Sub-Menu ▶ Debug Intel Menu</p>				<p>Item Specific Help</p> <p>This Sub-Menu contains Setup Items which control the workarounds for the entire platform.</p>	
<p>F1 Help    ↑↓ Select Item    -/+ Change Values    F9 Setup Defaults Esc Exit    ↔ Select Menu    Enter Select ▶ Sub-Menu    F10 Previous Values</p>					

# 3.1 Arranque

- **Interfaz Extensible del Firmware (EFI)**
  - Es un sistema básico, desarrollado por Intel, para sustituir eventualmente a la **BIOS**.
  - Su funcionamiento es similar a la **BIOS**, ofreciendo un marco de emulación para sistemas operativos antiguos.
  - Las **EFI** tienen capacidades más avanzadas de acuerdo con el hardware actual, destaca la capacidad de gestionar unidades de almacenamiento grandes (La **BIOS** sólo puede gestionar particiones de hasta **2 TiB**, mientras que la **EFI** hasta **9 ZiB**)
  - Su implementación es más compleja, con estructura modular e instrucciones de **64 bits**.
  - Su herramienta de configuración presenta un aspecto mucho mas ameno, facilitando su utilización con la posibilidad de utilizar un ratón.
  - La selección de **código de arranque** del sistema operativo a ejecutar se efectúa desde la **EFI**, ya no se precisa de **gestores de arranque**.
  - El firmware de la **EFI** por lo general estará contenida en la memoria **RAM CMOS**, aunque también puede hallarse en pequeñas particiones de discos duros o unidades de memoria solida.

# 3.1 Arranque





# 3.1 Arranque

- **Código de arranque**
  - Es la secuencia de instrucciones que busca el sistema operativo en la unidad de almacenamiento y lo ejecuta (**bootstrap**).
  - Alternativamente puede ejecutar cualquier programa independiente.
  - La parte de la **BIOS/EFI** que busca este código en el sistema se denomina **cargador de arranque** o **bootloader**.
  - El **bootloader** cargará el primer **código de arranque** que encuentre de acuerdo con su orden de prioridad almacenado en la **RAM CMOS**.
  - El **código de arranque** se puede hallar en:
    - El primer sector de dispositivos de almacenamiento no particionados.
    - El primer sector de las unidades lógicas, de los dispositivos de almacenamiento particionados.
  - Los sistemas operativos basados en Unix no tienen código propio y deben de ser cargados mediante un **gestor de arranque**.

# 3.1 Arranque

- **Arranque de red**

- El **firmware de red** de las arquitecturas modernas ofrecen la posibilidad de arrancar el sistema desde la red y descargar una imagen de un sistema operativo.
- Este firmware esta integrado en los adaptadores de red modernos (antiguamente se insertaban en un zócalo destinado a tal efecto)
- En caso de redes LAN, este firmware especializado busca en la red mediante un protocolo de descubrimiento (**DHCP**) el **software inicial** del sistema operativo y lo descarga mediante protocolo **TFTP** (parecido al **FTP** pero más simple).
- Una vez descargado el software inicial en la memoria principal, este toma el control y descarga la imagen del sistema operativo en memoria principal.
- Según con que caso, el arranque de red posibilita prescindir del uso de discos duros locales.



# 3.1 Arranque

```
Network boot from Intel E1000
Copyright (C) 2003-2008 VMware, Inc.
Copyright (C) 1997-2000 Intel Corporation

CLIENT MAC ADDR: 00 0C 29 F3 8B 72  GUID: 564D0877-4ADE-70FE-42EB-2F6118F38B72
CLIENT IP: 192.168.1.3  MASK: 255.255.255.0  DHCP IP: 192.168.1.220
GATEWAY IP: 192.168.1.220

Downloaded WDSNBP...

Architecture: x86
Contacting Server: 192.168.1.220.
TFTP Download: boot\x86\pxeboot.com

Press F12 for network service boot
—
```

## 3.2 Particionado

- **Registro de arranque maestro (MBR)**

- Se encuentra en el **primer sector físico** (512 B) de las unidades de almacenamiento que han sido particionadas.



**MBR Sector físico 0** (1 sector = 512 B)

- **Área de Código (446 B):** Estará vacía si en el dispositivo no hay ningún sistema operativo. Puede contener varios posibles códigos:
  - El **código de arranque** del sistema operativo (antiguos Windows)
  - Un **gestor de arranque simple** que ubique el **código de arranque** de un sólo sistema operativo.
  - La primera fase de un **gestor de arranque múltiple** que permita seleccionar entre varios **códigos de arranque** de diferentes sistemas operativos.
- **Tabla de Particiones (64 B):** Describe la estructura de las particiones del dispositivo. Sólo tiene 4 entradas (4 particiones de 2 TiB máximo)
- **Firma de Arranque (2 B):** Indica si el dispositivo es la unidad de arranque o no (antiguos Windows).

## 3.2 Particionado

- **Gestor de Arranque (*Second-stage bootloader*)**
  - Es el programa que prepara al sistema para cargar el núcleo de un sistema operativo.
    - Para los sistemas **Windows**, el gestor de arranque va a ejecutar el código de arranque en primer sector de la partición (**registro de arranque de la partición PBR**) donde se encuentra los archivos del sistema operativo.
    - Para los **sistemas basados en Unix**, el gestor de arranque contiene en si mismo el código de arranque y se va a encargar el mismo de cargar el kernel del sistema.
  - Los gestores de arranque actuales se cargan en diferentes fases, estando la primera fase en el **MBR** y las siguientes en otra ubicación del dispositivo.
    - En sistemas **Windows** la segunda fase está en el directorio raíz de la partición del sistema.
    - En **sistemas basados en Unix**, va a estar en áreas estáticas del dispositivo.
  - **Principales gestores de arranque:**
    - **NTLDR** (Windows NT/2000/XP/Server 2003)
    - **BOOTMGR** (Windows Vista/7/8/Server 2008/2012)
    - **LILLO** (Linux compatible con otros sistemas)
    - **Grub** (Linux compatible con otros sistemas)
    - **Syslinux** (Linux compatible con otros sistemas)

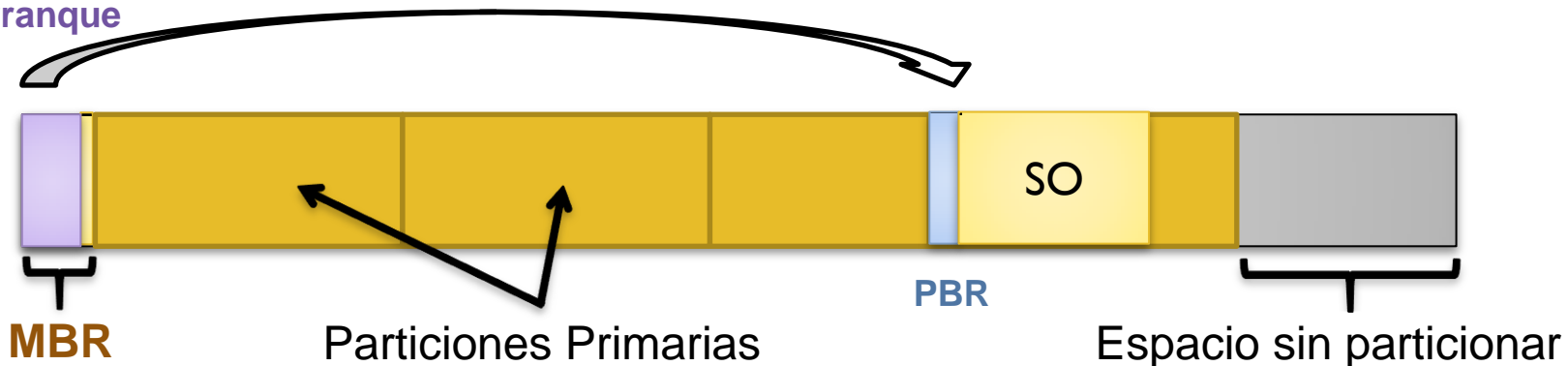
## 3.2 Particionado

- **Tipos de Particiones MBR**

- **Particiones Primarias:**

- En sistemas **BIOS** sólo se pueden tener 4 particiones primarias, de **2 TiB**, sus entradas que las mapean están en la **tabla de particiones** del **MBR**.
    - Las unidades de almacenamiento particionables cuentan como mínimo de una partición, que normalmente ocupará todo el espacio de almacenamiento.
    - El primer sector de la partición esta reservado para el **PBR**.

Gestor de arranque



## 3.2 Particionado

- **Tipos de Particiones MBR**

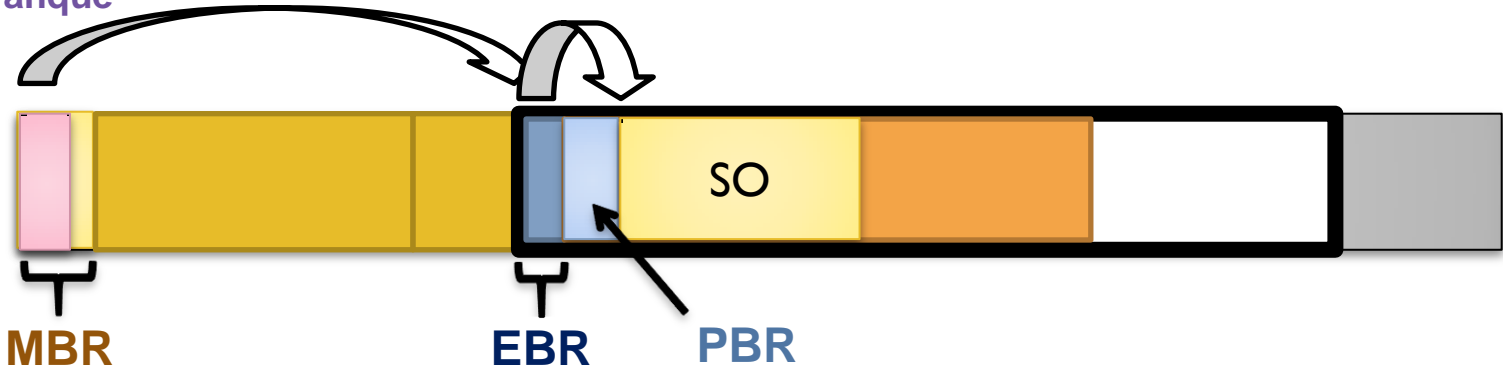
- **Particiones Extendidas:**

- Se puede sustituir una sola partición primaria por una partición extendida.
    - Es una partición que no puede contener datos, no obstante si puede contener particiones lógicas (hasta 23 particiones más).
    - Contiene su propia tabla de particiones en un **registro de arranque extendido (EBR)**.

- **Particiones Lógicas:**

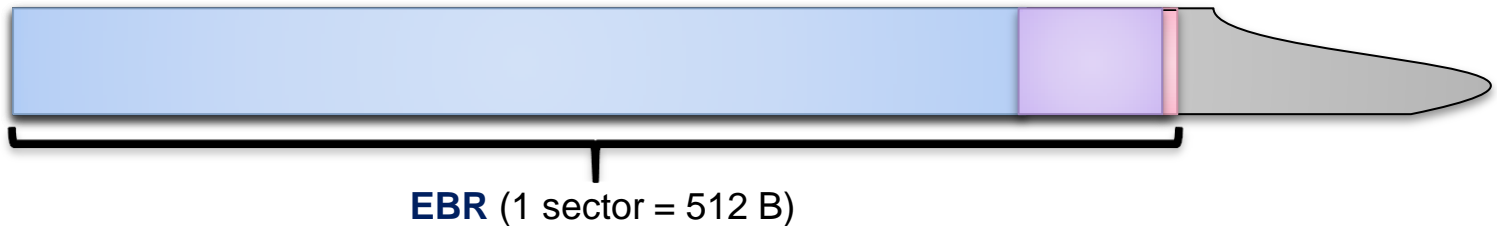
- Particiones destinadas a ocupar una porción de una partición extendida o toda ella.
    - Los sistemas operativos trabajarán con ellas sin distinguirlas de las particiones primarias.

Gestor de  
arranque

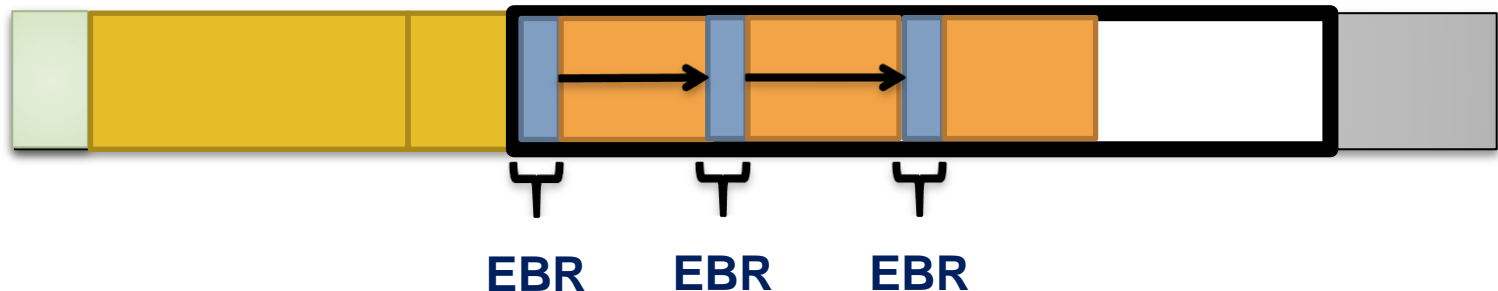


## 3.2 Particionado

- **Registro de arranque extendido (EBR)**
  - Se encuentra en el **primer sector físico** (512 B) de las **particiones extendidas**.



- Su estructura es muy parecida al **MBR**.
- En principio su tabla de particiones extendida solo puede gestionar **una partición lógica**.
- Si existe, la segunda entrada de la tabla de particiones no apunta a una partición sino a un nuevo **EBR** que formando una cadena se comportará de la misma manera.



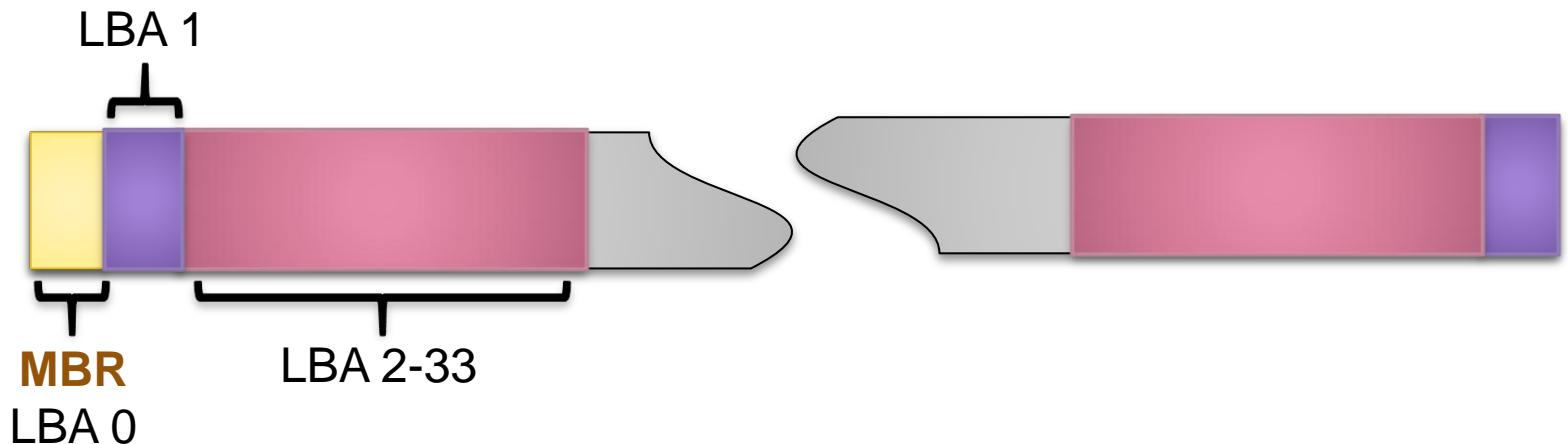


## 3.2 Particionado

- **Tipos de particiones MBR**
  - **00 Partición vacía**
  - 01 Partición **FAT12**
  - 04 Partición **FAT16** (hasta 32 MiB)
  - 06 Partición **FAT16** (más de 32 MiB)
  - **07 Partición NTFS**
  - 11 Partición **FAT32** CHS
  - **12 Partición FAT32** LBA
  - 13 Partición **FAT16** LBA
  - **15 Partición extendida**
  - 17 Partición **FAT12 Oculta**
  - 20 Partición **FAT16 Oculta** (hasta 32 MiB)
  - 22 Partición **FAT16 Oculta** (más de 32 MiB)
  - 27 Partición **FAT32** CHS **Oculta**
  - **28 Partición FAT32** LBA **Oculta**
  - **130 Partición Linux swap**
  - **131 Partición ext2, ext3 o ext4**
  - **175 Partición HFS o HFS+**

## 3.2 Particionado

- **Tabla de Partición GUID (GPT)**
  - Es el estándar implementado para las **EFI** (La **BIOS** utiliza **MBR**)
  - Mantiene compatibilidad con las actuales **MBR**
  - La **GPT** esta duplicada por seguridad al final del disco.
  - Utiliza el **direccionamiento lógico de sectores (LBA)**.
    - **LBA 0: MBR** (por compatibilidad).
    - **LBA 1:** Cabecera **GPT**, define los bloques utilizables por el usuario y el número de particiones (Windows limita a 128 particiones).
    - **LBA 2 a la 33:** Contienen las entradas de partición (8 ZiB máximo).
  - Las particiones son ahora **primarias** y no existe el concepto de particiones extendidas o lógicas.



## 3.2 Particionado

- **Discos dinámicos**

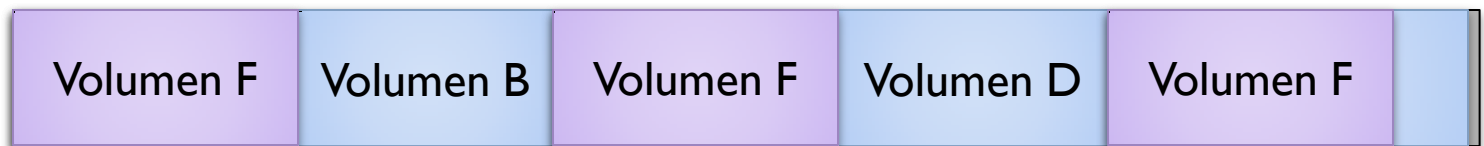
- Es un dispositivo (un disco duro o una SSD) que permite organizar el espacio de almacenamiento de una manera más flexible que con la organización normal.
- El concepto de partición desaparece, el espacio se divide en **volúmenes**. El PBR pasa a llamarse **registro de arranque del volumen (VBR)**
- Los dispositivos que utilizan un particionamiento tradicional (ya sea basado en **MBR** o **GPT**) se les llama disco básico.
- Los **discos dinámicos** empezaron a utilizarse con las versiones de 64 bits de Windows XP y Server 2003. Actualmente existen herramientas en otros sistemas operativos para crear **discos dinámicos**.
- En **discos dinámicos**, los volúmenes pueden ser:
  - Simples
  - Distribuidos
  - RAID
    - RAID0 (Seccionados)
    - RAID1 (Reflejados)
    - RAID5

## 3.2 Particionado

- **Discos dinámicos**

- **Volumen simple**

- Es el volumen más sencillo.
    - Se crea reuniendo en un solo volumen el espacio no particionado de una unidad física. Es espacio empleado no tiene por que ser contiguo.
    - En Windows se le asignará una única letra de unidad y será utilizado como una unidad lógica normal y corriente.
    - El volumen se puede expandir a posteriori, añadiéndole más espacio sin particionar.

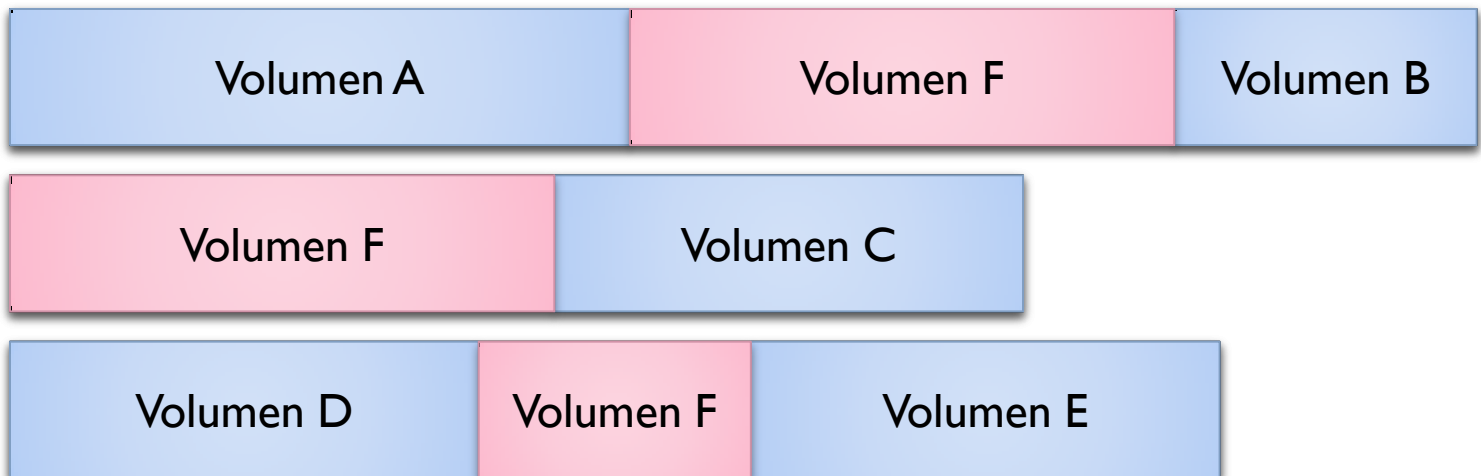


## 3.2 Particionado

- **Discos dinámicos**

- **Volumen distribuido**

- Es el volumen muy parecido al simple, sólo que se puede extender por varios discos.
    - Permite, al igual que el simple, gestionar distintas áreas de almacenaje con una única letra de unidad.
    - Éste volumen también se puede expandir posteriormente.



## 3.2 Particionado

- **Discos dinámicos**

- **Sistemas RAID**

- Un sistema RAID es un tipo de organización que utiliza técnicas de acceso simétrico y redundante en varios dispositivos de almacenamiento.
    - El acceso a un volumen RAID se realiza en paralelo entre todas las unidades físicas (aumenta la velocidad transferencia de E/S aunque no la velocidad de acceso).
    - Estos volúmenes presentan redundancia de datos lo que hace aumentar su fiabilidad.
    - Son útiles en servidores, no suelen presentar ventajas efectivas en sistemas de escritorio.
    - Se puede implementar de dos formas:
      - **Mediante Hardware:**
        - Los volúmenes RAID son gestionados por hardware en las interfaces SCSI, SAS o SATA.
        - Se necesita un adaptador especializado.
        - Se utiliza dispositivos enteros, es importante que sean del mismo tipo y velocidad de transmisión.
      - **Mediante Software:**
        - Por otro lado, en Windows y en sistemas Linux se puede crear volúmenes RAID gestionados por software.
        - En este caso se pueden utilizar volúmenes en vez de dispositivos enteros. Los sistemas por software son más flexibles pero menos eficaces.
        - La gestión por software de un volumen RAID es incompatible entre sistemas operativos.



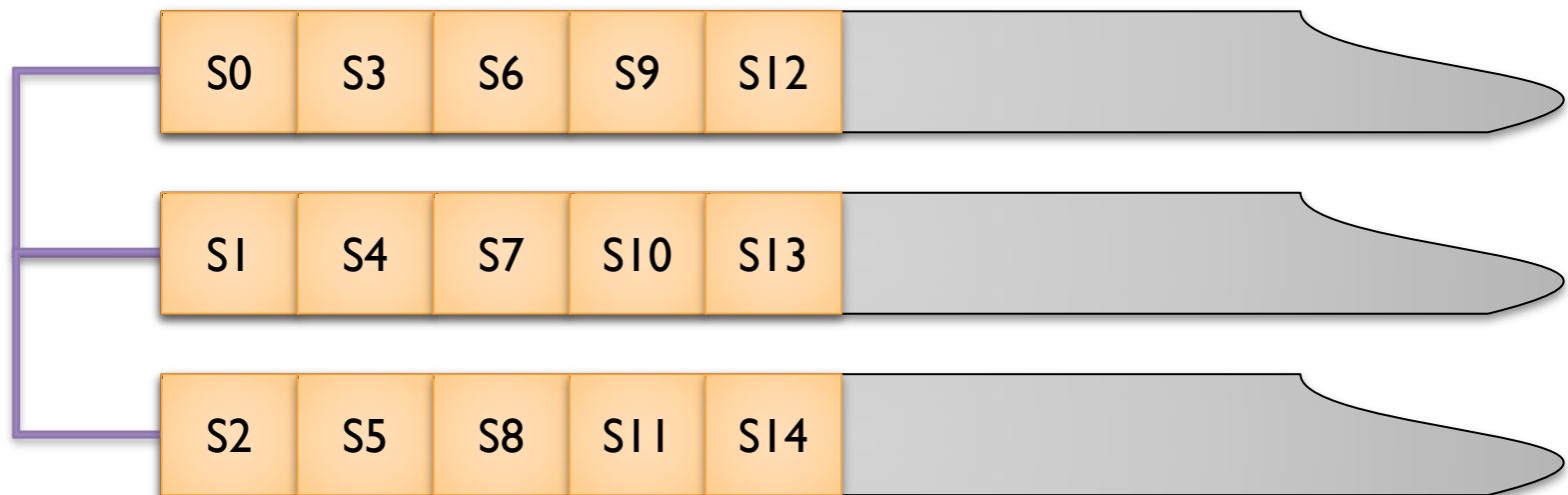
## 3.2 Particionado

- **Discos dinámicos**

- **Sistemas RAID**

- **RAID0: Volumen seccionado**

- Es un sistema RAID que no presenta redundancia de datos.
- Se puede utilizar cualquier número de unidades o volúmenes.
- Aumenta el rendimiento en conforme aumentamos el número de dispositivos.
- Por otro lado, al aumentar el número de dispositivos se pierde fiabilidad de acceso.



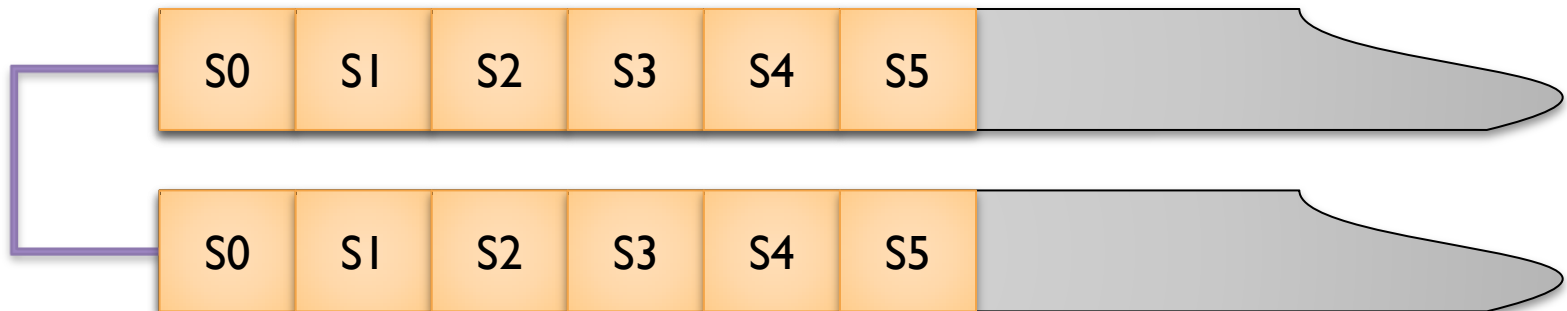
## 3.2 Particionado

- **Discos dinámicos**

- **Sistemas RAID**

- **RAID 1: Volumen reflejado**

- Es un sistema RAID cuyos discos físicos contienen la misma información sector por sector.
      - Se puede utilizar cualquier número de dispositivos o volúmenes, pero lo normal es dos.
      - Aumenta el rendimiento en la lectura continuada conforme aumentamos el número de discos físicos como sucedía en los volumen RAID 0.
      - Las escrituras se realizan en todos los sectores homólogos al mismo tiempo. No se gana velocidad en la escritura.
      - Se sacrifica espacio en el disco a cambio de una mayor fiabilidad.



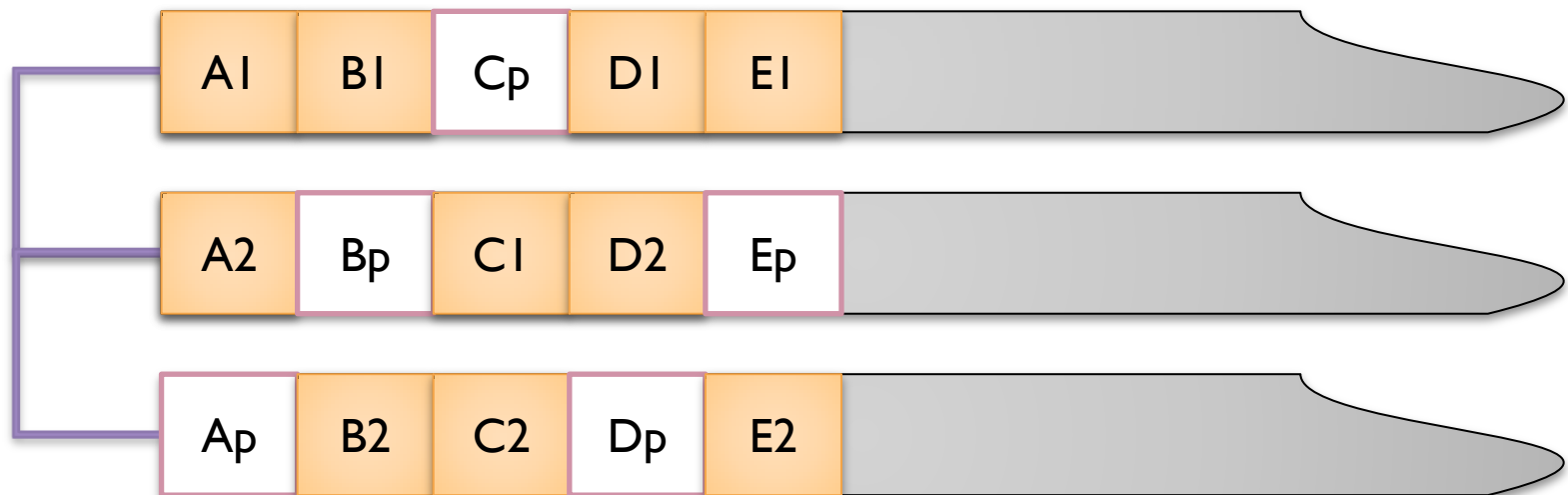
## 3.2 Particionado

- **Discos dinámicos**

- **Sistemas RAID**

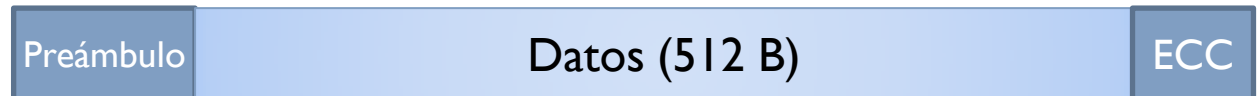
- **RAID 5: Volumen de paridad distribuida**

- Es un sistema RAID que no presenta redundancia de datos.
- Se necesitan como mínimo tres discos.
- Es tan rápido en lectura y escritura como un RAID 0.
- Los sectores se comparan con un sector de paridad. Eso lo hace muy robusto ante catástrofes.
- Se emplea menos memoria que en un RAID 1.



## 3.3 Formato de dispositivos

- **Proceso de Formateado:** Preparar una unidad de almacenamiento para su utilización.
  - **Formateado a bajo nivel (Físico)**
    - Este formateo se hace en fabrica y normalmente no es necesario volver a hacerlo.
    - Se realizan sobre todo el dispositivo.
    - Se crean las estructuras lógicas que contendrán la información sobre la superficie del disco.
    - Se crean los sectores indicando en un **preámbulo** el número de cilindro y sector.



- **Formateado a alto nivel (Lógico)**
  - Se realiza sobre una partición.
  - Se crean las estructuras de datos (sistema de archivos) que contendrán los ficheros y soportará el sistema.
  - Se crea la estructura del **PBR/VBR**.
  - Se crea el sistema de archivos elegido.
  - Se crea el directorio raíz.
  - Opcionalmente se puede “borrar” el área de datos direccionable realizando un formateado completo.

# 3.4 Instalación y actualizaciones

- **Actualización**

- Siempre es conveniente tener el sistema operativo debidamente actualizado:
  - Se añaden nuevas funcionalidades.
  - Se mejoran antiguas funcionalidades.
  - Se corrigen errores de programación (*bugs*).
  - Se solventan problemas de vulnerabilidad.
- Para garantizar la integridad del sistema operativo, las actualizaciones deben ser oficiales.
- Las actualizaciones, salvo en muy raras excepciones, necesitan acceso a Internet.
- Las actualizaciones suelen entrar en conflicto con modificaciones ajenas.
- Tipos:
  - **Criticas:** Corrigen problemas graves.
  - **Recomendables:** Optimizan funcionalidades concretas ya existentes.
  - **Opcionales:** Añaden nuevas funcionalidades.

# 3.4 Instalación y actualizaciones

- **Paquetes de distribución**

- Normalmente, la actualización e instalación de nuevas aplicaciones en el sistema se realiza a través de **paquetes de distribución**.
  - Un paquete de distribución es un archivo que contiene toda la información necesaria para instalar la aplicación en el sistema.
  - En algunos casos, todos los archivos estarán en el mismo paquete, en formato comprimido.
  - En otros serán archivos externos en la misma carpeta que el paquete.
  - En algunas ocasiones el paquete puede requerir la descarga de archivos a través de la red.
- Por otro lado debe de existir en el sistema una **herramienta de instalación** que lea el paquete y realice todas las operaciones necesarias.
- Algunas aplicaciones muy simples no requieren instalación y se pueden ejecutar directamente invocando su ejecutable.



# 3.4 Instalación y actualizaciones

- **Paquetes de distribución**

- **En sistemas Windows:**

- La Herramienta de Instalación se llama *Windows Installer*.
- Los archivos de instalación pueden estar en el paquete o ser externos.
- Tipos de paquetes compatibles con Windows:
  - **Paquetes .msi:** Información y opcionalmente archivos para la instalación de aplicaciones.
  - **Paquetes .mst:** Información y opcionalmente archivos para la modificación de una instalación ya realizada.
  - **Paquetes .msp:** Información y opcionalmente archivos para la actualización de aplicaciones.
  - **Paquetes .msu:** paquetes asociados a Windows Update.
  - **Paquetes .cab:** Son paquetes comprimidos, usuales en las carpetas de instalación.
  - **Paquetes .zap:** Contienen sólo información, en formato de texto plano, de la ubicación de archivos a través de la red.
- Muchas veces estos paquetes van a estar embebidos en archivos comprimidos ejecutables .exe (autoinstalables).
- El paquete de instalación normalmente incluye claves de configuración que se insertarán en el registro de Windows.

## 3.4 Instalación y actualizaciones

- **Paquetes de distribución**

- **En sistemas basados en *debian*:**

- La Herramienta de Instalación se llama ***dpkg***. Es una herramienta de bajo nivel y normalmente no se utiliza directamente.
- Para utilizar ***dpkg*** se utiliza la biblioteca ***Advanced Packaging Tool (APT)*** cuyas primitivas sirven para implementar aplicaciones más complejas:
  - ***apt-get*** en la línea de comandos.
  - ***aptitude*** en la línea de comandos, más completo que ***apt-get***.
  - ***Synaptic*** con un entorno gráfico.
- Los archivos de instalación van a estar dentro de paquetes ***DEB (repositorios)***, archivos únicos que contienen todo lo necesario para instalar el programa.
- Los repositorios ***DEB*** se van a poder descargar directamente de bases de datos en Internet.

## 3.4 Instalación y actualizaciones

- Descargar e instalar paquetes:  
`$ sudo aptitude install <paquetes>`
- Descargar paquetes sin instalarlos:  
`$ aptitude download <paquete>`
- Desinstalar paquetes:  
`$ sudo aptitude remove <paquetes>`
- Desinstalar paquetes (incluyendo archivos de configuración):  
`$ sudo aptitude purge <paquetes>`
- Actualizar la lista de paquetes disponibles:  
`$ sudo aptitude update`
- Actualizar el sistema con las actualizaciones de paquetes disponibles:  
`$ sudo aptitude upgrade`
- Obtener una lista de opciones del comando:  
`$ aptitude help`
- Instalar un paquete en el sistema de archivos local:  
`$ sudo dpkg -i ruta_paquete`

# 3.4 Instalación y actualizaciones

- **Paquetes de distribución**

- **En sistemas android:**

- La herramienta de alto nivel (nativa) de adquisición e instalación de paquetes es la aplicación **Vending.apk** (**Google Play (Google)**).
- Existen plataformas alternativas que compiten con el Market de Google.
- Las aplicaciones se van a encontrar contenidas en paquetes en formato *APK* que fundamentalmente contienen los scripts que ejecutará en la maquina virtual *Dalvik*.
- El formato *APK* es una variante del paquete *JAR* que a su vez una variante del formato comprimido *ZIP*. Se compone de las siguientes partes:
  - AndroidManifest.xml (nombre, versión, permisos, bibliotecas...)
  - classes.dex (clases)
  - resources.arsc (recursos precompilados en binario)
  - res (carpeta con recursos no compilados)
  - **META-INF** (carpeta con certificados e información varia)
  - lib (carpeta con código compilado que sirve de interfaz con la CPU)
    - armeabi (procesadores ARM)
    - armeabi-v7a (procesadores ARMv7)
    - x86 (procesadores tipo x86)
    - mips (procesadores basados en Mips)

# 3.5 Administración de Windows

- **Características principales de Windows**
  - Propietario. Se distribuye en diferentes ediciones:
    - Ediciones para dispositivos móviles
    - Ediciones Home
    - Ediciones para profesionales
    - Ediciones para servidores (Server)
  - Utilización de grupos de trabajo y de hogar.
  - Seguridad (Windows Defender y Firewall)
  - Compatibilidad total con procesadores x86, x86-64 y IA-32 (sólo versiones server).
  - La configuración básica se realiza mediante el **Panel de control**.
  - La administración avanzada se realiza mediante la **Microsoft Management Console (MMC)**.
    - Administrador de equipos
    - Configuración del sistema (arranque)
    - Directiva de seguridad local
    - Programador de tareas

## 3.5 Administración de Windows

Edición de Windows 8	Licencias	Precio (2013)
Windows RT (Dispositivos Moviles)	1 Dispositivo	Sólo OEM
Windows 8	1 Equipo	120 €
Windows 8 Pro	1 Equipo	280 €
Windows 8 Enterprise	Variable	Contrato
Windows Server 2012 Foundation	1 Servidor 15 Clientes	OEM
Windows Server 2012 Essential	1 Servidor 25 Clientes	385 €
Windows Server 2012 Standard	1 Servidor Clientes variables	680 € + CAL
Windows Server 2012 Datacenter	1 Servidor Clientes variables	3 706 € + CAL
CAL (Client Access License)	1 Clientes 5 Clientes	30 € 130 €



# 3.5 Administración de Windows

- **Gestión de Usuarios**

- Los sistemas operativos actuales, ya trabajen en modo multiusuario o en modo monousuario, soportan múltiples cuentas de usuario.
- Por cada persona que habitualmente utilice el sistema debería de haber una cuenta de usuario asociada
- Para que un usuario se pueda conectar a un dominio de una red local necesita una cuenta de usuario.
- En los sistemas móviles, sólo tiene sentido que exista una cuenta de usuario activa.
- Una cuenta de usuario estará asociada a uno o más grupos de usuarios.
- Tanto los usuarios individuales como los grupos contarán con una serie de privilegios sobre los objetos del sistema (archivos, recursos, operaciones, etc)
- En equipos en red, es importante que las cuentas estén protegidas por contraseñas fuertes. Sobre todo las de más altos privilegios.

# 3.5 Administración de Windows

- **Gestión de Usuarios en Windows**

- En los sistemas Windows actuales van a existir las siguientes cuentas fundamentales:
  - Cuenta de **Administrador**.
    - Tiene máximos privilegios, es recomendable no utilizarla si se puede evitar.
    - Puede estar oculta o estar desactivada. Pero siempre existe.
  - Cuentas de administradores.
    - Tienen altos privilegios, está destinada a usuarios avanzados.
    - Se suele crear al menos una en la Instalación del sistema operativo.
  - Cuentas limitadas o de usuario.
    - Cuentas de bajos privilegios que no permiten la administración del equipo.
  - Cuenta de **Invitado**.
    - Cuenta genérica de mínimos privilegios destinada a usuarios esporádicos.
  - Cuentas de sistema.
    - Realizan operaciones del sistema operativo que requieren sesiones especiales. No son accesibles por los usuarios.

# 3.5 Administración de Windows

- **Seguridad Local en Windows**

- Los **privilegios** o **permisos** otorgan o prohíben la posibilidad de realizar ciertas acciones en el equipo.
- Los privilegios se establecen sobre los objetos respecto a grupos de usuarios o a cuentas concretas.
- Siempre que exista conflicto entre privilegios, prevalecerá los restrictivos frente a los permisivos.
- Algunos privilegios están englobados en otros.
  - Control total
  - Modificar (Sin borrar ni cambiar permisos)
  - Lectura y ejecución
  - Mostrar contenido (Sólo directorios)
  - Lectura
  - Escritura
- Se pueden establecer permisos más específicos (permisos especiales)

# 3.5 Administración de Windows

- **Seguridad Local en Windows**

- Cada objeto dispone de amplio número de permisos asignables.
- Pueden permitir o denegar, pero pueden también no existir. Si no hay permiso especificado no se permite la acción referida.
- Un usuario puede pertenecer a varios grupos al mismo tiempo percibiendo permisos solapados.
- Los usuarios de grupos de altos privilegios suelen pertenecer también a los grupos de bajos privilegios.
- Es recomendable para evitar conflictos otorgar permisos a los grupos y denegar permisos solo a los usuarios concretos. Para facilitar esto ultimo se pueden crear grupos auxiliares con prohibiciones concretas.
- El propietario del objeto siempre tiene la última palabra respecto él, (aunque un usuario de altos privilegios puede apropiarse del archivo).

# 3.5 Administración de Windows

- **Seguridad Local en Windows**

- **Herencia:** Cuando se crea un archivo dentro de una carpeta, el archivo *hereda* por defecto los permisos de ésta (Aunque el usuario que lo creo sigue siendo el propietario)
- **Dependencia:** Cuando cambiamos los permisos de un directorio, se brinda la posibilidad de que la nueva configuración se transmita a todos los elementos que están dentro de ella.
- Aunque un usuario posea permisos plenos sobre un archivo, si el acceso al directorio esta denegado, el usuario en cuestión no podrá acceder a él. (Aunque sea el propietario)

## 3.6 Administración de sistemas Linux

- **Características principales de los sistemas Linux**
  - Gratuito, bajo licencia libre y mantenido mayoritariamente por la comunidad. Dependiendo de la distribución podemos obtener diferentes versiones:
    - Distribuciones para dispositivos móviles
    - Distribuciones de escritorio
    - Distribuciones profesionales
    - Distribuciones para servidores
  - Gran repertorio disponible (algunas preinstaladas) de herramientas y aplicaciones bajo licencia libre.
  - Multitud de entornos de escritorio disponibles.
  - Algunas distribuciones tienen capacidad de almacenar datos en nubes de computación.
  - La configuración básica se realiza mediante una herramienta de configuración del sistema, parecido al panel de control.
  - La administración avanzada se realiza mediante la edición de archivos de texto plano en el directorio /etc.

## 3.6 Administración de sistemas Linux

- **Comandos útiles**

Comando sustitución de usuario **sudo** y **su**

Comando de ayuda **man**

Comando matar **kill**

Comandos de información de procesos **ps**, **pstree**, **top** y **jobs**

Comando de información de memoria **vmstat**

Comando de información de dispositivos **dmesg**

Comandos de manipulación de archivos **ls**, **cd**, **pwd**, **touch**, **cp**, **mv**, **rm**, **mkdir**, **ln** y **cat**

Comandos de apagado **halt** y **reboot**

Comando de información de usuarios **users**, **who**, **whoami** y **id**

Comandos de creación de usuarios **adduser** y **useradd**

Comandos de creación de grupos **addgroup** y **groupadd**

Comandos de borrado de usuarios **deluser** y **userdel**

Comando de borrado de grupo **delgroup**

Comando de gestión de grupos **usermod**

Comandos de gestión de propiedad de archivos **chown** y **chgrp**

Comando de gestión de permisos **chmod**

## 3.6 Administración de sistemas Linux

- **Gestión de Usuarios en Linux**

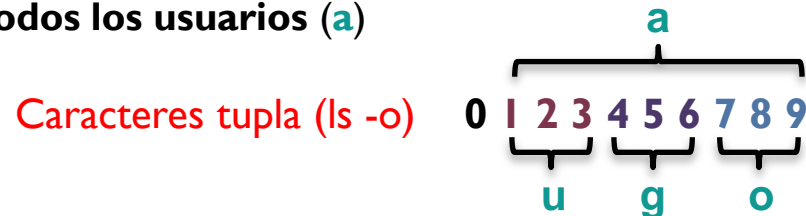
- La gestión de usuarios es parecida a VWindows, vamos a contar con cuentas de usuarios y grupos de usuario.
- Cada cuenta de usuario tiene un nombre y un identificador único UID asociado, pertenecerá a un grupo principal y puede que a varios secundarios.
- Cada grupo de usuarios tiene un nombre y un identificador único GID.
- En los sistemas Linux tradicionales existe un único administrador **root** y los demás usuarios disfrutan a priori del mismo estatus.
- En los sistemas Linux modernos, la cuenta **root** está deshabilitada, los demás usuarios siguen una organización parecida a VWindows.



## 3.6 Administración de sistemas Linux

- **Seguridad local en Linux**

- En el terminal la gestión de permisos se realiza con notación octal o notación simbólica.
- En la notación simbólica se establecen cuatro niveles para establecer los permisos:
  - **Usuario propietario (u)**: El creador del archivo.
  - **Grupo propietario (g)**: El grupo del creador del archivo.
  - **Otros usuarios (o)**: El resto de los usuarios.
  - **Todos los usuarios (a)**



- El primer carácter corresponde con el tipo de archivo:
  - **Archivo regular (-)**
  - **Directorio (d)**
  - **Enlace blando (l)**
  - **Archivo especial E/S de caracteres (c)**
  - **Archivo especial E/S de bloques (b)**

## 3.6 Administración de sistemas Linux

- **Seguridad local en Linux**

- Los permisos en sistemas Linux solo tienen dos estados: concedido y denegado.
- Se establecen tres tipos de permisos básicos:

Carácter	Permisos de los archivos	Permisos de los directorios	Concedido	Denegado
<b>l 4 7</b>	Lectura	Listar el contenido	<b>r</b>	-
<b>2 5 8</b>	Escritura	Crear o borrar contenido	<b>w</b>	-
<b>3 6 9</b>	Ejecución	Abrirlo o utilizarlo en rutas	<b>x</b>	-

- Existen dos tipos de permisos especiales:
  - **SUID/ SGID (s)** (carácter 3 y 6)
    - Permite que cualquier usuario **ejecute** un **archivo** como si fuera el propietario (**u±s**) o el grupo propietario (**g±s**).
    - Fuerza que los archivos del **directorio** donde se aplica pertenezcan al propietario (**u±s**) o el grupo propietario (**g±s**) de este.
  - **Sticky (t)** (carácter 9)
    - Sobre un **archivo** no tiene relevancia.
    - En los archivos del **directorio** donde se aplica restringe el borrado y renombrado al usuario propietario, grupo propietario y al **root** (**±t**).