

Contenidos

- Organización del almacenamiento
 - Sectores, bloques y direccionamiento LBA
 - Clúster o unidad de asignación
- Particiones
 - Tipos de particionado
 - Sistemas de archivos
- Gestores de particiones en Windows
 - Administración de discos
 - Diskpart
 - Formatear particiones fuera de los gestores de particiones
- Gestores de particiones en Linux
 - fdisk
 - Formatear particiones con mkfs
 - Montaje manual de particiones
 - cfdisk
 - parted y gparted
- Montaje de particiones en el fichero fstab
- Particionado en la instalación del sistema operativo

Organización del almacenamiento

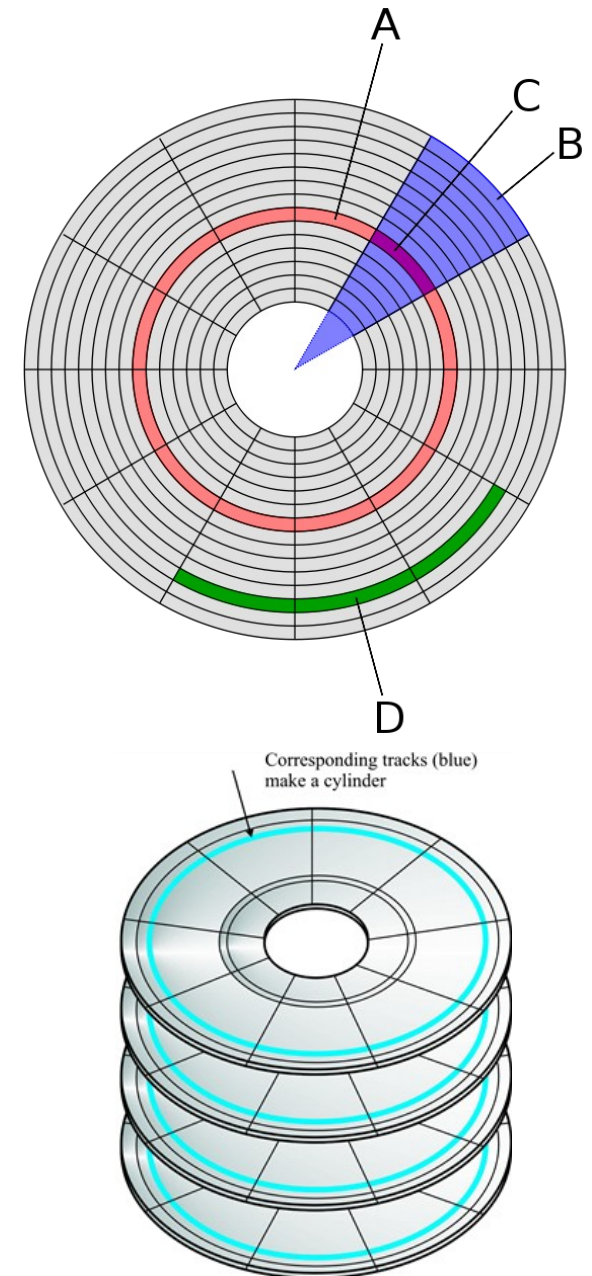
Cuando hablamos de los discos duros mecánicos habíamos visto que este se dividía en varios discos en los que se escribía en dos caras.

Sobre estos se establece la siguiente organización:

- A) **Pista:** Círculos concéntricos con respecto al eje del disco
- B) **Sector geométrico:** Agrupación de sectores del centro al borde
- C) **Sector:**
- D) **Clúster:** Agrupación de sectores contiguos

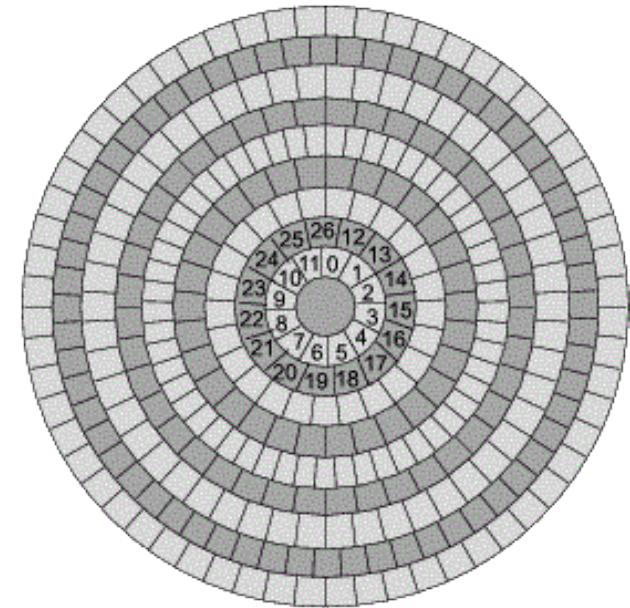
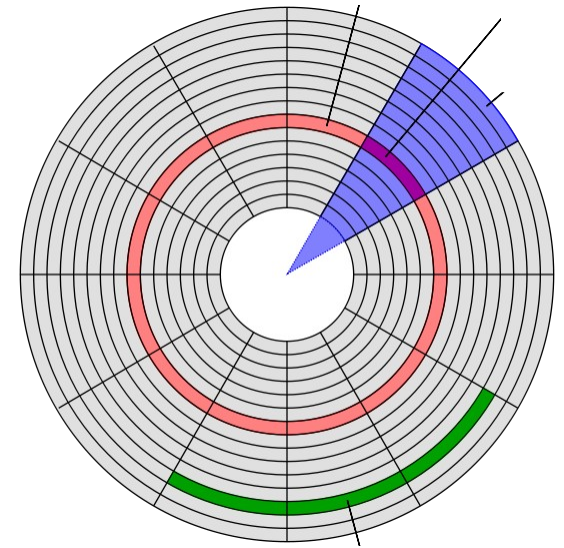
Además teníamos que las pistas que están a la misma altura entre varios discos forman un **cilindro**.

Esta organización sólo tiene sentido en discos duros mecánicos que se componen de varios discos divididos en pistas, **en el resto de dispositivos de almacenamiento esta distribución se hace únicamente por sectores y clústeres.**



Sector, clúster y direccionamiento LBA

- **Sector:** Es por tanto la unidad más pequeña de almacenamiento y estos sectores para un mismo dispositivo de almacenamiento se configura con un tamaño fijo (típicamente de 512B o 4KB).
- **Direccionamiento LBA (Logical Block Address):** Es el sistema por el cual se asigna una dirección a cada bloque del disco
 - A cada bloque se le asigna un número único, empezando en 0 y acabando en N-1, siendo N el número de sectores del disco.
 - Este modo de direccionamiento se aplica a todos los dispositivos de almacenamiento y **el número de bloques del disco variará dependiendo del tamaño de bloque que definamos.**
- **Clúster o unidad de asignación:**
 - Como vimos antes un clúster es una agrupación de sectores contiguos.
 - A nivel del sistema operativo acceder a cada sector de forma individual supondría una gran sobrecarga por el elevado volumen de estos y **esta agrupación facilita la gestión del disco y mejora el rendimiento**
 - Un clúster es la unidad mínima en la que un sistema operativo organiza el almacenamiento de los ficheros



LBA Addressing

Uso de clústeres al almacenar ficheros

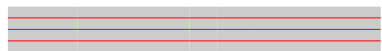
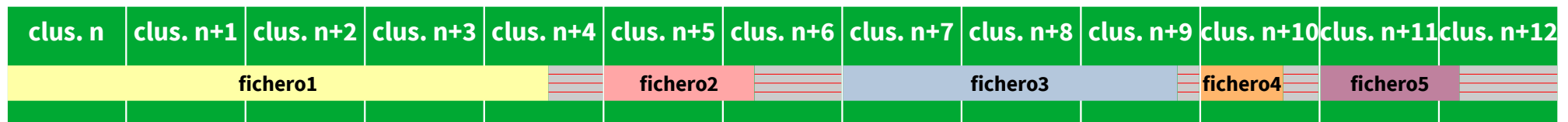
Un archivo almacenado en disco requerirá uno o varios clústeres dependiendo de sus tamaños, y **un clúster es exclusivo de un fichero por lo que no puede almacenar datos de varios ficheros al mismo tiempo.**

Esto implica que al guardar un fichero, cuando este no tenga un tamaño múltiplo del tamaño del clúster quedará espacio en su último clúster que no podrá ser utilizado por otro fichero.

Además un fichero de tamaño 0 ocupará un clúster entero, aunque en realidad casi no escriba información en el disco.

Vamos a ilustrarlo con el siguiente gráfico:

- Los recuadros verdes representan los clústeres de un disco numerados (clus. n, clus. n+1, ...)
- Tenemos 5 ficheros de distintos tamaños cada uno representado con un rectángulo de un color. Cada uno de ellos ocupa uno o varios clústeres dejando distintos espacios en el último clúster sin utilizar. Este espacio se representa con los rectángulos rayados en rojo
- Si juntamos todos estos espacios sin utilizar vemos como se queda sin utilizar el espacio equivalente a dos clústeres y medio y que sólo podrá ser utilizado si el fichero crece en tamaño.



Ocupación de clústeres según tamaño de fichero

Vamos a ver para distintos tamaños de clúster y ficheros cuantos clústeres ocuparía el fichero y cuando espacio libre quedaría inutilizado en el último clúster donde observaremos que:

- **Cuanto mayor sea el tamaño de clúster más espacio inutilizado quedará en el último clúster**
- Para minimizar el espacio desperdiciado podemos optar por poner **un clúster pequeño**, pero esto **implica una mayor cantidad de clústeres asociados a cada fichero**. Esto implica otro problema que es el de una **mayor sobrecarga a la hora de leer el fichero** al tener que acceder a más elementos. Además supone un **incremento en la fragmentación**, que trataremos más adelante.

Tamaño de clúster (KB)	Tamaño fichero (KB)	Número de clústers	Espacio libre inutilizado
4	32	8	0
4	66	17	2
4	123.809	30.953	3
16	32	2	0
16	66	5	14
16	123.809	7.739	15
128	32	1	96
128	66	1	62
128	123.809	968	95

Distintos tamaños de clúster para distintos usos

Teniendo en cuenta que un fichero puede ocupar varios clústeres y que el espacio del último clúster no se aprovecha hasta el final, podemos ajustar el tamaño del clúster según el uso:

- **Un tamaño de clúster grande es idóneo si trabajamos con ficheros grandes:**
 - La ventaja principal es que hay una menor sobrecarga en la lectura porque **hay que leer una menor cantidad de clústeres**.
 - Es verdad que **desperdiciamos más espacio** en ese último clúster de los ficheros, pero hay que tener en cuenta que **este desperdicio es reducido al haber menos ficheros** al ser estos más grandes.
- **Un tamaño de clúster pequeño es idóneo si vamos a tener muchos ficheros pequeños:**
 - En este caso **será menor la cantidad de espacio perdido por clústeres incompletos**.
 - El número de clústeres asociado a un fichero no será muy elevado por el reducido tamaño del mismo.

Por lo general **dejaremos la configuración por defecto y ajustaremos el tamaño si se nos da claramente uno de los casos anteriores**.

Particiones

Partición de un disco duro

- Una partición de un disco duro es una división **lógica** sobre una unidad de almacenamiento, como un disco duro o una memoria flash como un pendrive USB
- Podemos optar por crear una o varias particiones, pero el espacio asignado a una partición en un disco debe ser **contiguo**
- El espacio del disco que no pertenezca a ninguna partición será **espacio sin asignar**
- Desde el punto de vista de usuario cada partición aparece como una unidad de disco distinta con la que trabaja de forma independiente, aunque sean particiones de un mismo disco físico
- En la creación de una partición debemos indicar:
 - El **tamaño**
 - El **sistema de archivos** de dicha partición
 - **Flags**, entre los que podemos indicar cual es la **partición activa**, que será la partición en la que en el momento del arranque el ordenador buscará un gestor de arranque que permita iniciar un sistema operativo
- Una vez creada deberemos **formatear la partición para poder leer y escribir datos**
- La distribución de particiones se almacena en una estructura de datos llamada **tabla de particiones** almacenada en el propio disco

Sistema de archivos

- El sistema de archivos es un **componente que permite al Sistema operativo** controlar cómo se lee y escribe la información en los dispositivos de almacenamiento (discos duros, pendrive usb, tarjetas sd, ...).
- Hay múltiples sistemas de archivos, cada uno con sus propias características que condicionan aspectos como los valores máximos que es capaz de soportar:
 - Tamaño máximo de partición
 - Tamaño máximo de archivo
 - Número máximo de archivos
- Cada sistema operativo trabaja preferiblemente con unos sistemas de archivos determinados
 - Windows → FAT32, exFAT, NTFS y ReFS
 - Linux → ext2, ext3, ext4 y swap

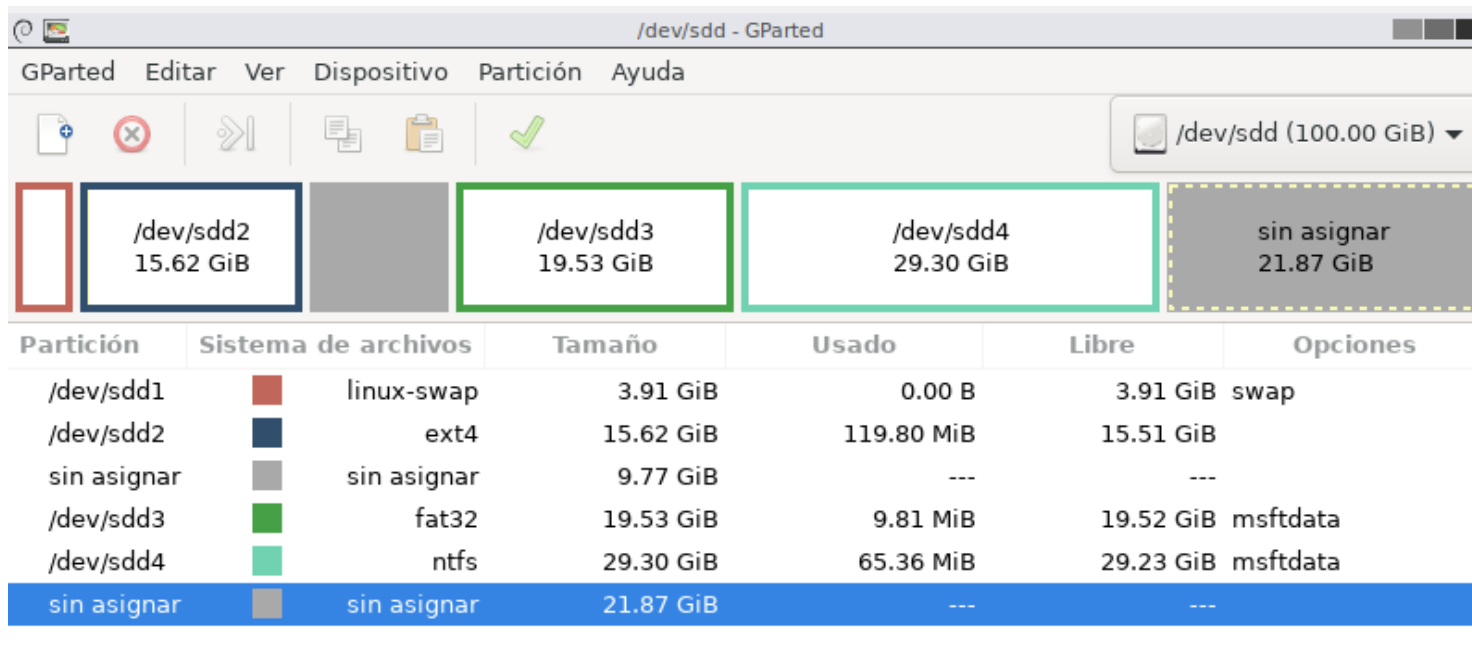
Ejemplo de particionado

En la siguiente captura podemos ver desde una utilidad de particionado un disco particionado con múltiples sistemas de archivos (Filesystems): ntfs, linux-swap y ext3

Vemos que todas las particiones tienen espacio contiguo y que incluso tenemos un hueco al inicio y en el medio sin asignar (unallocated)

Tenemos una partición marcada como de arranque, la que tiene en la columna flag el valor **boot**

La nomenclatura `/dev/sdd1 ... /dev/sdd4` es propia de los sistemas Linux para la identificación de las particiones (`/dev/sdd` identifica el disco duro y el número la partición dentro de un mismo disco)



Partición	Sistema de archivos	Tamaño	Usado	Libre	Opciones
/dev/sdd1	linux-swap	3.91 GiB	0.00 B	3.91 GiB	swap
/dev/sdd2	ext4	15.62 GiB	119.80 MiB	15.51 GiB	
sin asignar	sin asignar	9.77 GiB	---	---	
/dev/sdd3	fat32	19.53 GiB	9.81 MiB	19.52 GiB	msftdata
/dev/sdd4	ntfs	29.30 GiB	65.36 MiB	29.23 GiB	msftdata
sin asignar	sin asignar	21.87 GiB	---	---	

Flags de una partición

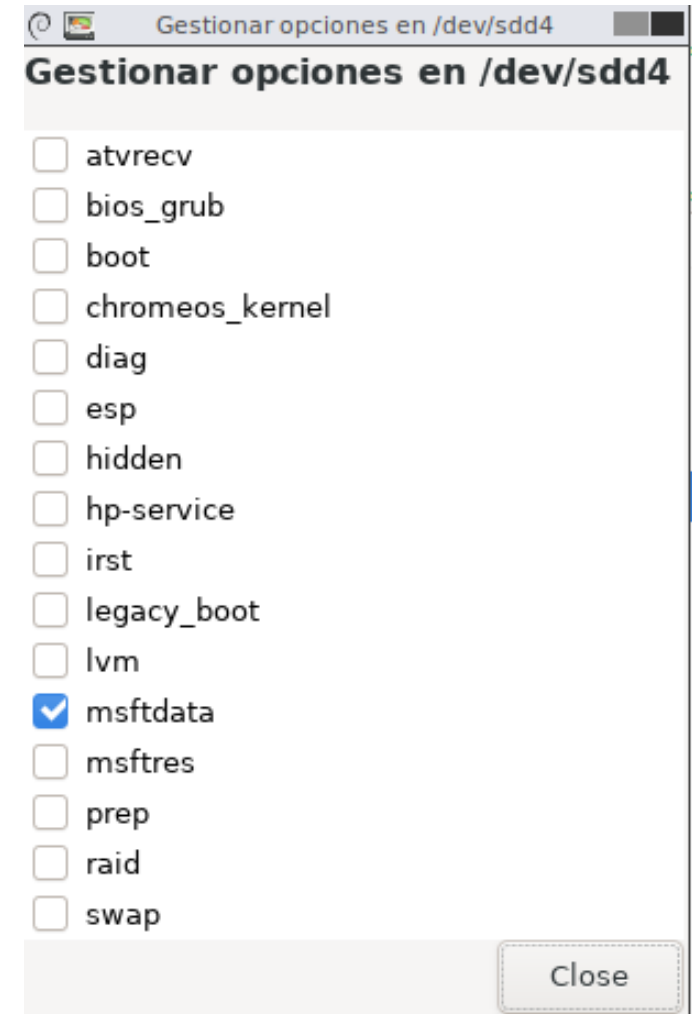
Cada partición además puede tener una serie de “flags” (se configuran como activado o desactivado) con los que se ajustan atributos adicionales con los que se indican usos específicos de las particiones.

Son varios los flags que podemos encontrar, pero los más típicos son:

- **boot**: Flag para indicar que se trata de una partición de arranque porque tiene un sistema operativo
- **root**: La utilizan los sistemas operativos Linux para saber la partición en la que se monta el raíz del sistema de archivos (/)
- **swap**: La utilizan los sistemas operativos Linux para saber la partición que se utiliza como swap
- **raid**: Flag para indicar que esta partición forma parte de una configuración de almacenamiento RAID
- **hidden**: Flag de partición oculta para particionado MBR
- **msftdata y msftres**: Son flags utilizados para particiones específicas de los sistemas Windows. **msftdata** para particiones que no tienen el flag boot y **msftres** para particiones “reservadas”, pero sólo en particionado GPT

Puedes consultar una descripción más detallada de todos los flags en el siguiente enlace:

<http://dcjtech.info/topic/gparted-partition-flags>



Flags que podemos configurar con el gestor de particiones Gparted, en este denominadas opciones, para discos con particionado GPT.

Tipos de particionado

Tipos de particionado

Antes de poder particionar un disco duro tenemos que elegir el tipo de particionado que va a utilizar. Hay múltiples tipos, de los que los más comunes son:

MBR (Master Boot Record) o MSDOS

- El tipo de particionado tradicional, y el que tendremos que utilizar cuando estamos trabajando con ordenadores con BIOS antiguas que no sean UEFI
- Entre otras cosas ofrecen limitaciones en cuanto al número de particiones que podremos crear y el tamaño de estas
- Lo encontraremos con el nombre MBR o MSDOS dependiendo de la utilidad de administración de particiones que utilicemos

GPT (GUID Partition Table)

- Es el tipo de particionado más reciente sólo soportado por BIOS más modernas del tipo UEFI
- Dan solución a las limitaciones que había con MBR, permitiendo un mayor número de particiones y de mayor tamaño

Particionado MBR (Master Boot Record) o DOS

- Es el tipo de particionado clásico y el único soportado por las BIOS clásicas (No UEFI)
- Tiene como límites máximos 4 TB en disco y 2 TB en partición
- Este tipo de particionado distingue entre los siguientes tipos de particiones:
 - **Primaria** (Hasta un máximo de 4)
 - **Extendida** (Sólo puede haber una)
 - **Lógica** (contenidas dentro de la extendida hasta un máximo de 32)
- Un disco particionado según MBR puede tener como máximo:
 - 4 particiones primarias
 - 3 primarias y una extendida con hasta un máximo de 32 lógicas
- La **tabla de particiones** se almacena en una zona del disco llamada MBR (Master Boot Record) o **pista cero** y esta incluye toda la configuración de particiones del disco. Está siempre **en el primer sector del disco**. Su tamaño dependerá del tamaño del sector que se configure (512 bytes o 4KB) y que habitualmente es **512 bytes**.



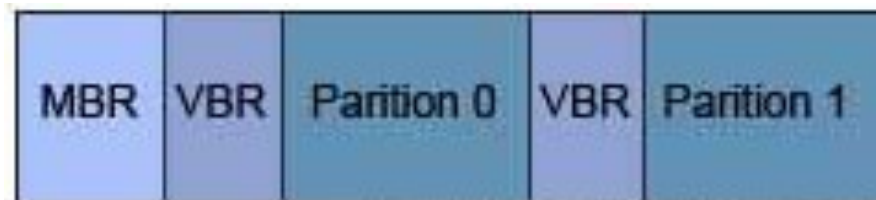
MBR (Master Boot Record) y el gestor de arranque

El MBR es por tanto el primer sector del disco (dirección LBA 0), que como dijimos almacena la información de las particiones del disco en la tabla de particiones, pero reserva espacio para almacenar las instrucciones del **gestor de arranque (boot manager)**

El gestor de arranque es un programa que se ejecuta justo después de las instrucciones de arranque de la BIOS, y este básicamente se encarga de **indicar qué partición tiene las instrucciones para arrancar un sistema operativo, redireccionando el arranque a esta.**

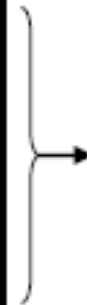
En el caso de que tengamos múltiples sistemas operativos instalados en un mismo ordenador un gestor de arranque puede configurarse para mostrar un menú en el que podremos seleccionar el que queremos arrancar.

No siempre tendremos un gestor de arranque en el MBR. En estos casos **se redirige a la partición marcada como activa**, donde tratará de arrancar un sistema operativo buscando en esta otro gestor de arranque almacenado que estará en el denominado **VBR (Volume Boot Record)** almacenado en los primeros sectores de la partición.



Distribución de datos en el MBR I

DOS partition table format	
Bytes	Purpose
0-445	Boot code
446-461	Partition Table Entry #1
462-477	Partition Table Entry #2
478-493	Partition Table Entry #3
494-509	Partition Table Entry #4
510-511	Signature value (0xAA55)



DOS Partition Table Entry format	
Bytes	Purpose
0	Bootable flag (0x80=active; else 0x00)
1-3	Starting CHS address
4	Partition type (e.g., 0x00=empty, 0x01=FAT12, 0x07=NTFS, 0x0b=FAT32 (CHS), 0x83=Linux, 0xa5=FreeBSD, 0xa8=MacOS X)*
5-7	Ending CHS address
8-11	Starting LBA address
12-15	Size (in sectors)

Distribución de datos en el MBR II



MASTER BOOT RECORD

≥ INVOKE-IR

BY: JARED ATKINSON
 TEMPLATE BY: ANGE ALBERTINI

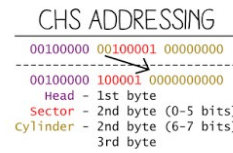


```

000: 33 C0 8E D0 BC 00 7C 8E C0 8E D8 BE 00 7C BF 00
010: 06 B9 00 02 FC F3 A4 50 68 1C 06 CB FB B9 04 00
020: BD BE 07 80 7E 00 00 7C 0B 0F 85 0E 01 83 C5 10
030: E2 F1 CD 18 88 56 00 55 C6 46 11 05 C6 46 10 00
040: B4 41 BB AA 55 CD 13 5D 72 0F 81 FB 55 AA 75 09
050: F7 C1 01 00 74 03 FE 46 10 66 60 80 7E 10 00 74
060: 26 66 68 00 00 00 00 66 FF 76 08 68 00 00 68 00
070: 7C 68 01 00 68 10 00 B4 42 8A 56 00 8B F4 CD 13
080: 9F 83 C4 10 9E EB 14 B8 01 02 BB 00 7C 8A 56 00
090: 8A 76 01 8A 4E 02 8A 6E 03 CD 13 66 61 73 1C FE
0A0: 4E 11 75 0C 80 7E 00 80 0F 84 8A 00 B2 80 EB 84
0B0: 55 32 E4 8A 56 00 CD 13 5D EB 9E 81 3E FE 7D 55
0C0: AA 75 6E FF 76 00 E8 8D 00 75 17 FA B0 D1 E6 64
0D0: E8 83 00 80 DF E6 60 E8 7C 00 80 FF E6 64 E8 75
0E0: 00 FB 88 00 BB CD 1A 66 23 C0 75 3B 66 81 FB 54
0F0: 43 50 41 75 32 81 F9 02 01 72 2C 66 68 07 BB 00
100: 00 66 68 00 02 00 00 66 68 08 00 00 66 53 66
110: 53 66 55 66 68 00 00 00 66 68 00 7C 00 00 66
120: 61 68 00 00 07 CD 1A 5A 32 F6 EA 00 7C 00 00 CD
130: 18 A0 B7 07 EB 08 A0 86 07 EB 03 A0 85 07 32 E4
140: 05 00 07 88 F0 AC 3C 00 74 09 8B 07 00 B4 0E CD
150: 10 EB F2 F4 EB FD 2B C9 E4 64 EB 00 24 02 E0 F8
160: 24 02 C3 49 6E 76 61 6C 69 64 20 70 61 72 74 69
170: 74 69 6F 6E 20 74 61 62 6C 65 00 45 72 72 6F 72
180: 20 6C 6F 61 64 69 6E 67 20 6F 70 65 72 61 74 69
190: 6E 67 20 73 79 73 74 65 6D 00 4D 69 73 73 69 6E
1A0: 67 20 6F 70 65 72 61 74 69 6E 67 20 73 79 73 74
1B0: 65 6D 00 00 00 63 7B 9A 82 D4 BA 7D 00 00 00 20
1C0: 21 00 07 FE FF FF 00 08 00 00 90 36 06 80 FE
1D0: FF FF 07 FE FF FF 00 A0 36 06 00 60 09 00 00 00
1E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1F0: 00 00 00 00 00 00 00 00 00 00 00 00 55 AA
    
```

BOOT CODE

FIELDS	VALUES
jump to boot program	
disk parameters	
boot program code	
disk signature	82D4BA7D



PARTITION TABLE

status	0x00 - Non-Bootable
starting head	0x20
starting sector	0x21
starting cylinder	0x00
partition type	0x07 - NTFS
ending head	0xFE
ending sector	0x3F
ending cylinder	0x3FF
relative start sector	0x800
total sectors	0x6369000
status	0x80 - Bootable
starting head	0xFE
starting sector	0x3F
starting cylinder	0x3FF
partition type	0x07 - NTFS
ending head	0xFE
ending sector	0x3F
ending cylinder	0x3FF
relative start sector	0x636A000
total sectors	0x96000

PARTITION TYPES

0x00 - EMPTY	0x83 - LINUX
0x01 - FAT12	0x84 - HIBERNATION
0x04 - FAT16	0x85 - LINUX_EXTENDED
0x05 - MS_EXTENDED	0x86 - NTFS_VOLUME_SET
0x06 - FAT16	0x87 - NTFS_VOLUME_SET_1
0x07 - NTFS	0xa0 - HIBERNATION_1
0x0b - FAT32	0xa1 - HIBERNATION_2
0x0c - FAT32	0xa5 - FREEBSD
0x0e - FAT16	0xa6 - OPENBSD
0x0f - MS_EXTENDED	0xa8 - MACOSX
0x11 - HIDDEN_FAT12	0xa9 - NETBSD
0x14 - HIDDEN_FAT16	0xab - MAC_OSX_BOOT
0x16 - HIDDEN_FAT16	0xb7 - BSDI
0x1b - HIDDEN_FAT32	0xb8 - BSDI_SWAP
0x1c - HIDDEN_FAT32	0xee - EFI_GPT_DISK
0x1e - HIDDEN_FAT16	0xef - EFI_SYSTEM_PARTITION
0x42 - MS_MBR_DYNAMIC	0xfb - VMWARE_FILE_SYSTEM
0x82 - SOLARIS_X86	0xfc - VMWARE_SWAP
0x82 - LINUX_SWAP	

END OF MBR

partition type	0x00 - EMPTY
partition type	0x00 - EMPTY
marker	0x55AA

<https://nixhacker.com/explaining-the-magic-of-mbr-and-its>

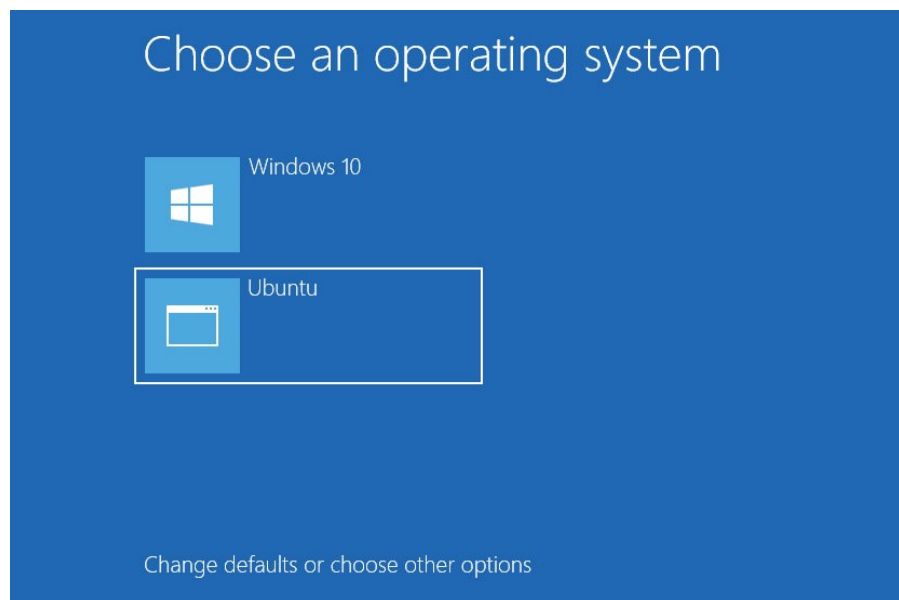
Ejemplos de gestores de arranque

Dos gestores de arranque distintos para seleccionar entre arrancar un Windows 10 y una Ubuntu instalados en distintas particiones:

- El **Windows Boot Manager**, de los sistemas Windows
- El **grub**, gestor de arranque típico de las distribuciones Linux

Cada uno de estos gestores de arranque se instalan por defecto con sus sistemas operativos y los podemos configurar posteriormente de forma manual o semiautomática para añadir nuevas entradas a los menús con con sistemas operativos en otras particiones.

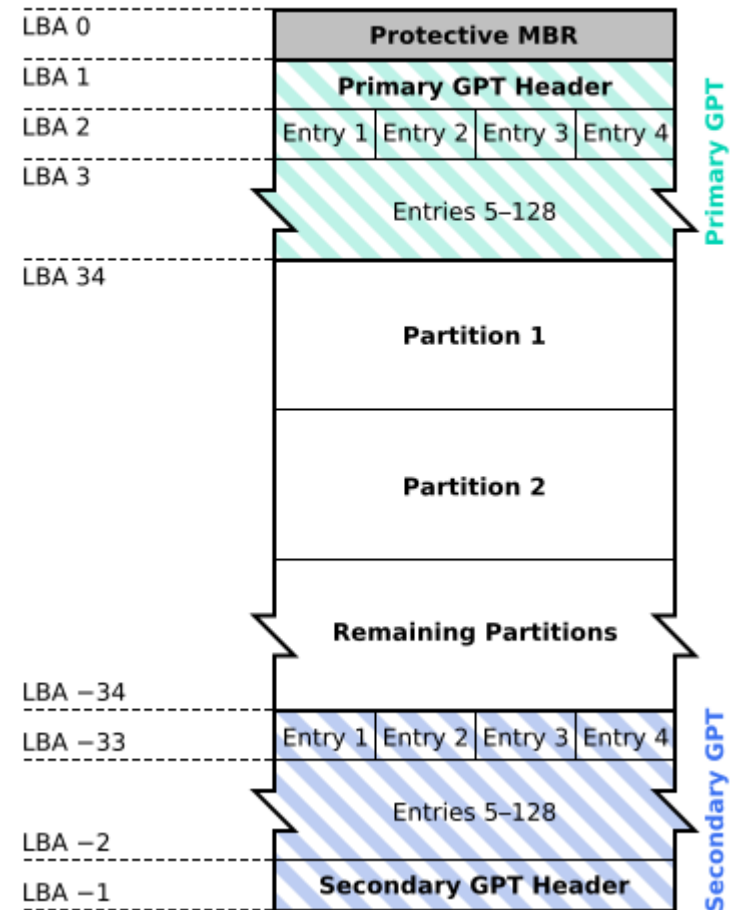
Veremos más adelante utilidades de configuración de gestores de arranque.



Particionado GPT (GUID Partition Table)

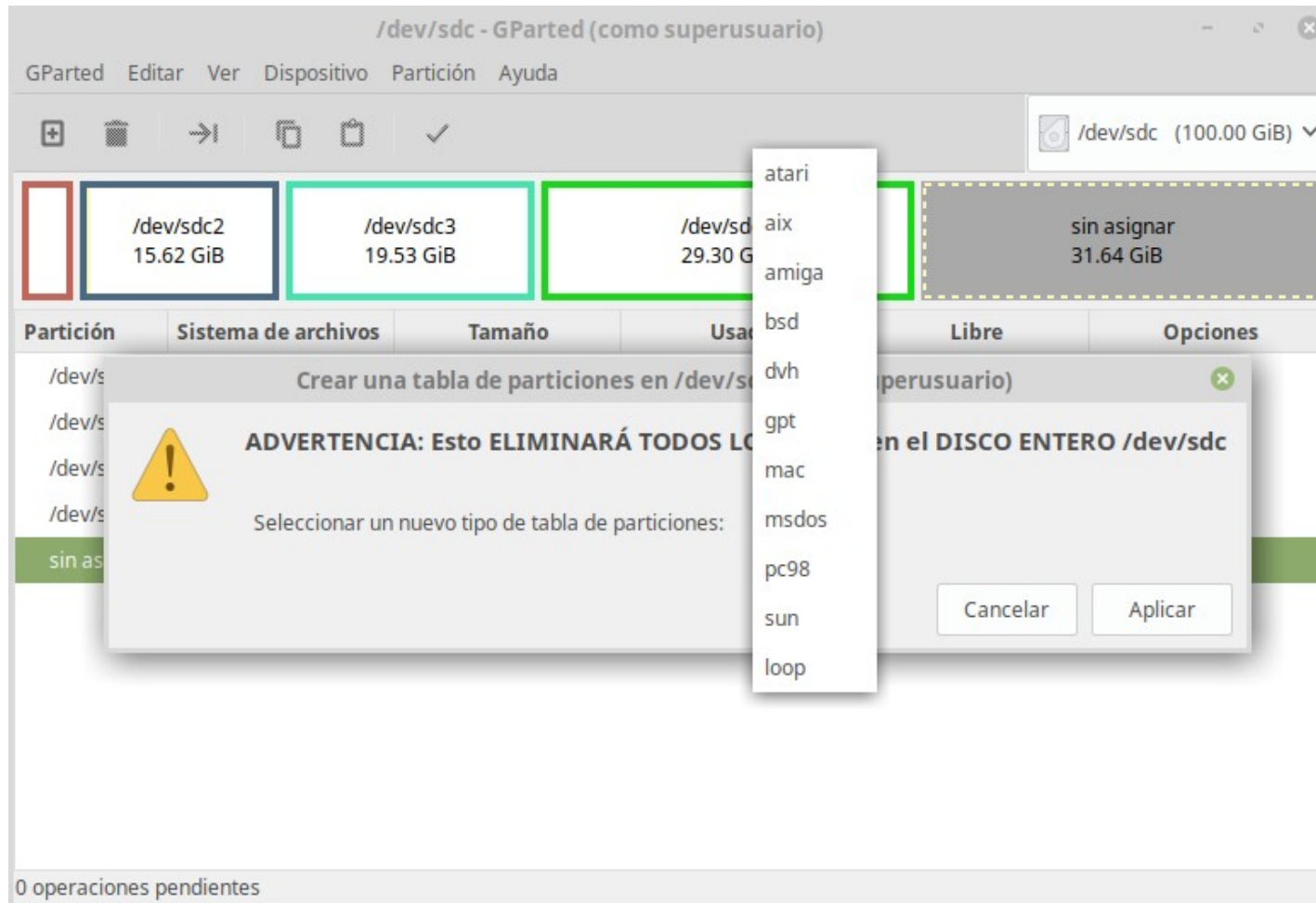
- Vino a solucionar el problema de las limitaciones de 2 TiB por partición y de 4 TiB por disco del MBR
- No hay distinción entre particiones, todas se consideran primarias, y se pueden crear hasta **un máximo de 128**
- Reserva también un espacio al principio del disco donde almacena la tabla de particiones y el gestor de arranque. **Este espacio es mayor que en MBR** ya que al poder particionar en más trozos necesita más espacio para guarda los datos de todas las particiones
- Dentro de este espacio reserva el primer sector (LBA 0) para el también llamado MBR por compatibilidad con equipos con BIOS antiguas
- Después del MBR utiliza 33 sectores (LBA de la 1 a la 33) para guardar información referente a las particiones del disco.
- Además reserva a modo de backup los últimos 33 sectores para replicar los sectores del encabezado GPT al principio del disco para casos de borrados accidentales

GUID Partition Table Scheme



Otros tipos particionado

GPT y MBR son los tipos de particionado más comunes algunos específicos para otros sistemas operativos. Vemos todos los que podemos configurar en un gestor de particiones como Gparted:



Diferencias de flags entre tipos de particionado

Otras diferencias entre los tipos de particionado es el tipo de flags que soportan. Vemos algunas diferencias notables:

- **boot**: Lo hay para los dos tipos de particionado para indicar aquellas particiones que tienen un sistema operativo con capacidad de arranque.
 - En MBR sólo puede haber una partición con el flag **boot**. Esta partición también se denomina **partición activa**.
 - En GPT puede varias particiones con el flag **boot** al mismo tiempo.
- **hidden**: Es específico del particionado MBR. En GPT se utiliza al flag **msftres** (Microsoft Reserved Partition) para tener algo parecido
- **msftdata y msftres**: Son específicas de particionado GPT para las particiones de los sistemas Windows (NTFS, FAT32 y exFAT)
- **raid**: Es específica del particionado MBR

En este enlace además de describir cada tipo de flag se indica para que tipo de particionado están disponibles:

<https://www.linux.org/threads/gparted-partition-and-filesystem-flags.11640>

Sistemas de archivos

Sistemas de archivos en Windows (FAT)

- **FAT (File Allocation Table)** es un sistema de archivos que ha tenido varias revisiones:
 - 8-bit FAT
 - FAT12
 - FAT16
 - FAT32
 - exFAT
- De las revisiones de FAT actualmente sólo están en uso **FAT32 y exFAT**
- FAT32 limita el tamaño máximo de los archivos a 4GiB, que en el caso de exFAT sube hasta los 128PiB
- Por esa limitación de FAT32 el uso de exFAT está recomendado para formatear dispositivos de almacenamiento externos (Pendrive, tarjetas sd, ...) que requieran almacenar archivos grandes

Sistemas de archivos en Windows (NTFS)

- **NTFS (New Technology File System)** es un sistema de archivos desarrollado por Microsoft
- Fue el sistema de archivos por defecto en sus sistemas Windows NT (precursores de los Server actuales) y empezó a ser soportado en las versiones de escritorio a partir de Windows XP
- Actualmente es el sistema de archivos por defecto en todos los sistemas de archivos Windows
- Su principal diferencia con respecto al FAT32 y exFAT es que NTFS tiene **soporte para ACL** (Access Control List)
- Las ACLs dan la capacidad de asignar permisos a los ficheros y carpetas para poder **limitar quien puede ver o escribir** determinados contenidos del disco
- Otras diferencias con respecto a FAT32 son los límites en cuanto a capacidades:
 - Particiones de un máximo de 256 TB
 - El tamaño máximo de fichero es de 16 EiB, pero este cambia según la versión de Windows
 - 16 TB en Windows 7, Windows Server 2008 R2 y versiones previas
 - 256 TB en Windows 8 y Server 2012
 - Un número máximo de ficheros de 4.294.967.295

Sistemas de archivos en Windows (ReFS)

- **ReFS (Resilient File System)** es un sistema de archivos desarrollado por Microsoft e introducido con Windows Server 2012 con intención de ser la evolución de NTFS
- Fue diseñado para solucionar limitaciones de NTFS derivadas de la evolución de los sistemas de almacenamiento y su uso:
 - Capacidad de almacenamiento cada vez mayor
 - Necesidad de disponer de un almacenamiento más fiable
- Fue añadido en Windows Server 2012 con la intención de ser introducido gradualmente en los sistemas Windows de escritorio a los que llegó en Windows 8.1
- Aun así la capacidad de crear particiones ReFS fue eliminada de todas las ediciones de Windows 10 “2017 Fall Creators Update”, a excepción de las ediciones Enterprise y Pro for Workstations
- Actualmente la creación de volúmenes ReFS sólo se mantiene en Windows 10 a través de la utilidad Espacios de almacenamiento
- ReFS mejora a NTFS en algunos aspectos, pero en otros le falta estabilidad y herramientas compatibles, por lo que de momento no va a ser adoptado como sustituto de NTFS en el futuro próximo

Sistemas de archivos en Linux

En Linux tenemos dos tipos principales de sistemas de archivos:

ext (extended file system)

- Fue el sistema de archivos creado en 1992 específicamente para el kernel de Linux. Desde su primera versión ext ha tenido tres revisiones adicionales: ext2, ext3 y ext4
- ext4, su última versión, soporta **particiones con un máximo de 1 EiB y ficheros hasta un máximo de 16 TiB**
- Es un sistema de archivos con **soporte de ACLs (Access Control List)**, por lo que se pueden definir permisos de acceso a los ficheros y carpetas a usuarios y grupos específicos del sistema de archivos

swap

- Es el sistema de archivos de Linux para la partición de intercambio, utilizada por el sistema operativo como zona auxiliar para liberar espacio en la memoria principal
- La partición de intercambio es utilizada únicamente por el sistema operativo y los usuarios no

Resumen sistemas de archivos Windows y Linux

Nombre	SO	Tamaño máximo archivo	Tamaño máximo partición	Número máximo de archivos	ACLs
FAT32	Windows	4 GiB	10 TiB (teóricos, en la práctica Windows solo dejar formatear hasta 32GB)	263.173.300	No
exFAT	Windows	128 PiB (16 EiB-1 teóricos)	128 PiB (recomendados 512 TiB)	2,796,202 por directorio	No
NTFS	Windows	256 TB	256 TiB	4.294.967.295	Sí
ReFS	Windows	35 PiB	35 PiB		Sí
ext4	Linux	16 TiB	1 EiB	4.000.000.000	Sí
Swap	Linux	N/A	N/A	N/A	N/A

Sistemas de archivos en sistema Apple

Apple en sus sistemas operativos Mac OS emplea sistemas de archivos propios:

HFS (Hierarchical File System) y HFS+ (HFS Plus)

HFS fue creado en 1985 para sus ordenadores macintosh y tuvo una evolución con HFS+ en 1998 con la salida de Mac OS 8.1

APFS (Apple File System)

Introducido en el año 2017 en el macOS High Sierra (10.13) y posteriormente en el iOS 10.3 es una evolución del HFS con optimizaciones para dispositivos de almacenamiento flash como los discos SSD.

Soporte de sistemas de archivos entre SOs

De forma nativa los sistemas operativos sólo soportan sus propios sistemas de ficheros a excepción de Linux que es capaz de montar y acceder a particiones con sistemas de archivos de sistemas Windows y Apple

Para acceder desde Windows o Mac a particiones de otros tipos hay que emplear utilidades de terceros.

Particionado del disco

- El particionado de un disco se realiza con unas **utilidades de disco** llamadas **gestores de particiones**
- Particionar un disco con un gestor de particiones implica que:
 - Hay que seleccionar el tipo de particionado (MBR o GPT)
 - Hay que crear las particiones asignándoles un tamaño y sistema de archivos
 - Hay que formatearlas, que es un proceso necesario para que el sistema operativo pueda trabajar con ellas. En este paso opcionalmente podremos ajustar un tamaño de clúster
- En Windows (de servidor o escritorio) disponemos de dos gestores de particiones:
 - **Administración de discos**: Gestor de particiones en modo gráfico
 - **Diskpart**: Gestor de particiones desde consola
- En Linux disponemos de múltiples herramientas:
 - **fdisk**: Un gestor de particiones de consola presente en casi todas las distribuciones Linux
 - **cdisk**: Un gestor de particiones de consola que incorpora las funcionalidades de fdisk pero con una interfaz que facilita
 - **parted y Gparted**: Un gestor de particiones de consola que también dispone de un gestor gráfico y que viene preinstalado en múltiples distribuciones

Gestores de particiones en Windows

Administración de discos

- La herramienta de administración de discos es la opción más sencilla en los sistemas Windows para administrar los discos a la hora de crear particiones y volúmenes
- Es una herramienta disponible tanto en los Windows de escritorio (Windows 7, 8.1 y 10) como en los de servidor (2012 R2, 2016 y 2019)
- Con ella podemos ver y administrar gráficamente los discos existentes y sus particiones y volúmenes realizando entre otras tareas como:
 - Cambiar el tipo de particionado (MBR o GPT)
 - Crear, borrar y formatear volúmenes
 - Expandir el tamaño de los volúmenes
 - Marcar la partición activa
 - Asignar letras de unidad o cambiar etiqueta a un volumen
 - Creación de volúmenes aplicando mecanismos de almacenamiento tipo RAID
- Nosotros de momento la utilizaremos en la creación de particiones clásicas, que en en nomenclatura Windows se llama **volumen simple**

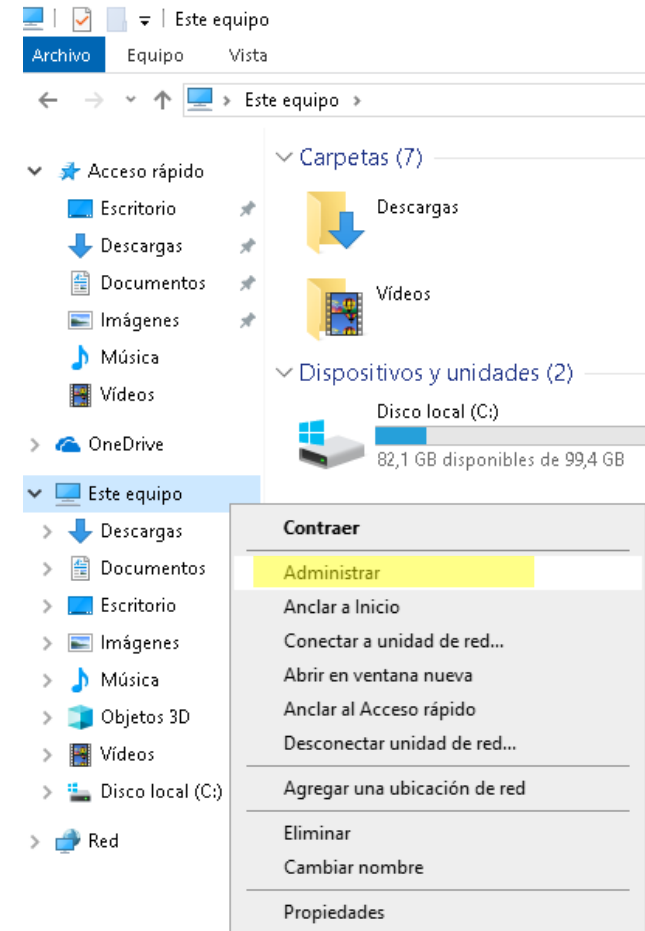
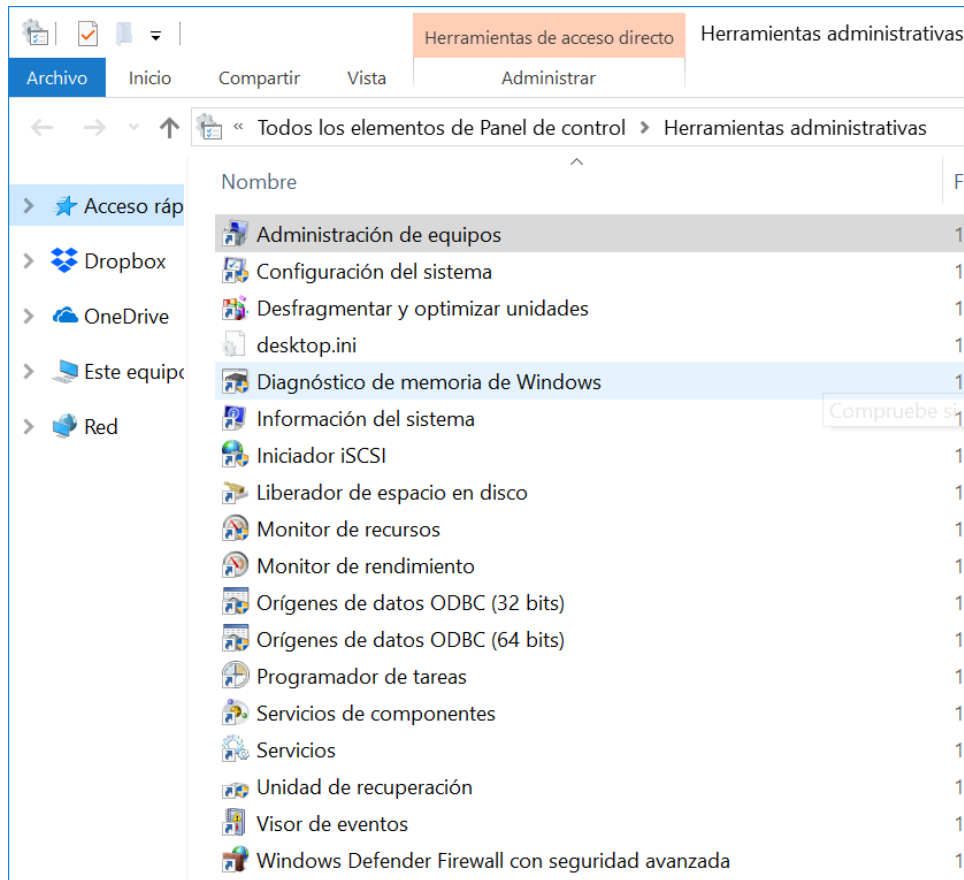
Particiones y volúmenes

- En la herramienta veremos que a la hora de dividir el disco sólo habla de volúmenes y que no hace referencia a particiones en ningún momento
- Veremos que hay varios tipos de volúmenes, cada uno con unas características de almacenamiento distintas más o menos complejas
- Entre estos tipos el más sencillo es el **Volumen simple** que es el que más se parece a crear una partición en el sentido más tradicional (Espacio contiguo en un mismo disco sin ningún tipo de almacenamiento RAID)
- En esta herramienta a mayores de los modos de particionado MBR y GPT podremos configurar los discos en dos modos adicionales específicos de esta herramienta:
 - **Básico**: En este modo el disco trabaja con particiones tradicionales que no se pueden redimensionar. Tendremos que utilizar este modo en discos que alojan sistemas operativos
 - **Dinámico**: En este modo los volúmenes simples se pueden redimensionar y permite la creación de volúmenes de almacenamiento tipo RAID que veremos en el siguiente tema (distribuido, reflejado, seccionado y RAID-5). En este modo no podemos crear particiones que alojarán sistemas operativos

Administración de discos en Windows 10

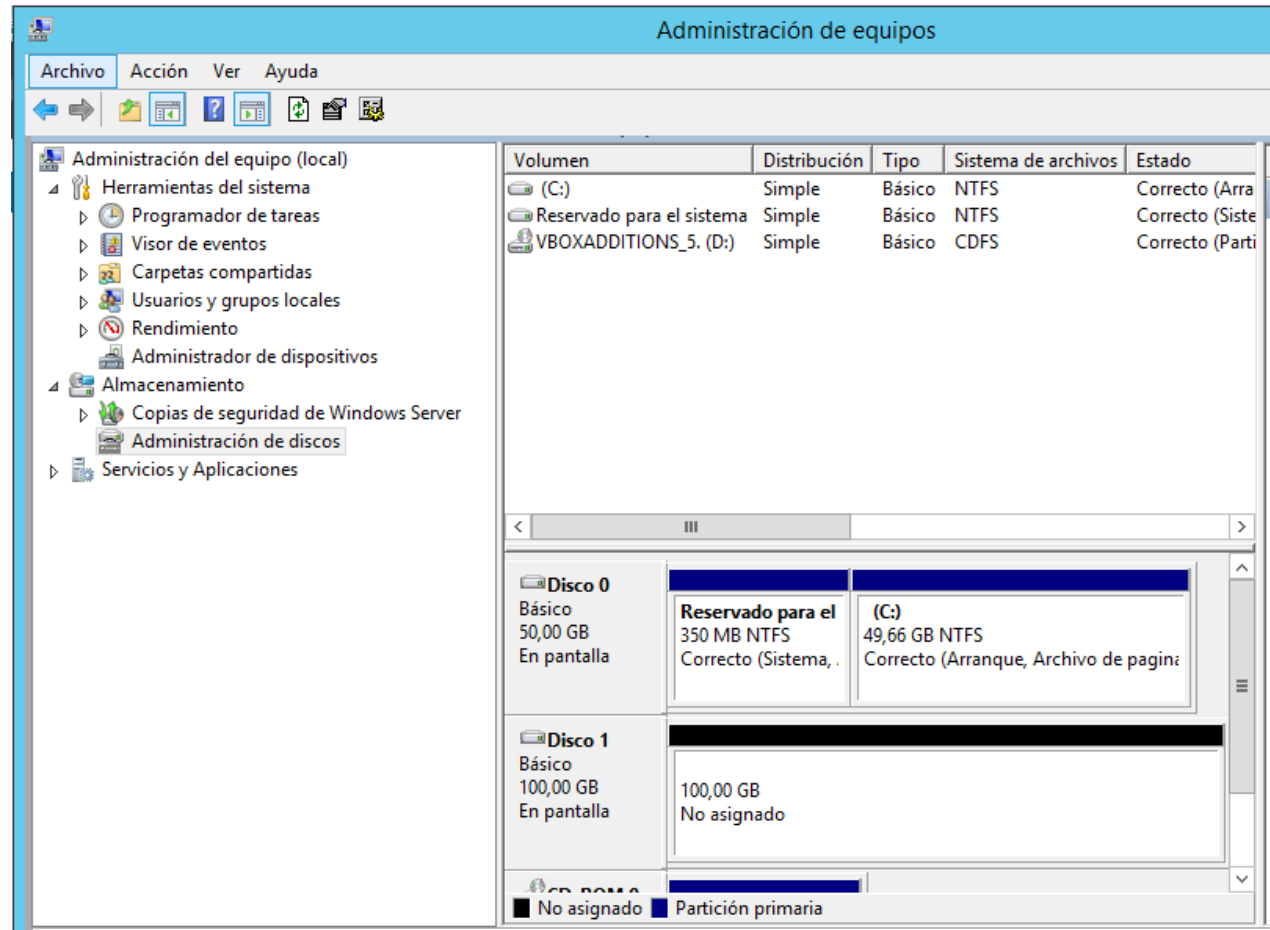
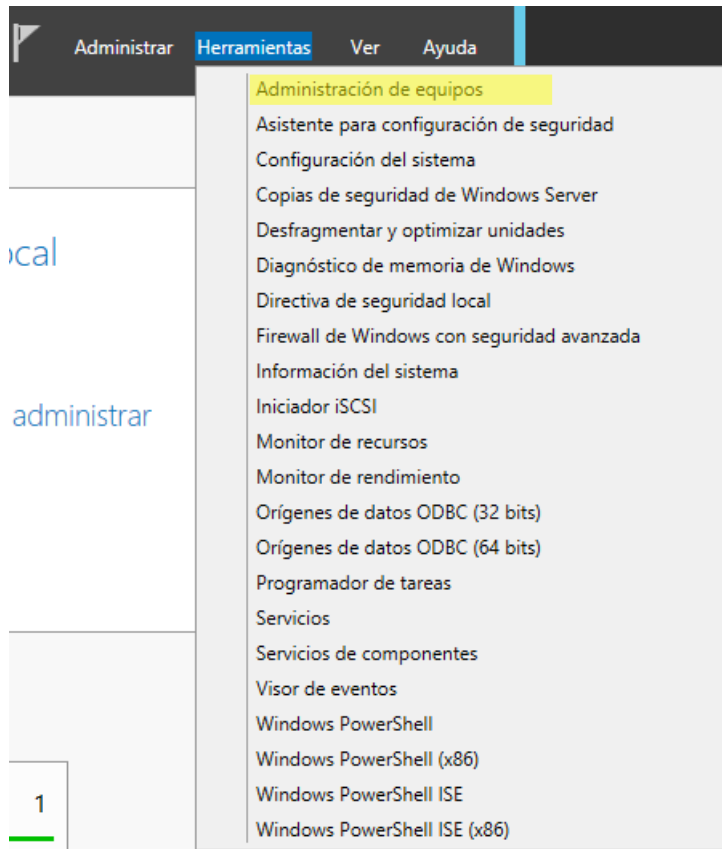
Podemos acceder a la herramienta de administración de discos por dos vías:

- Panel de control → Herramientas administrativas → Administración de equipos
- Seleccionando la opción “Administrar” en el menú contextual de “Este equipo”:



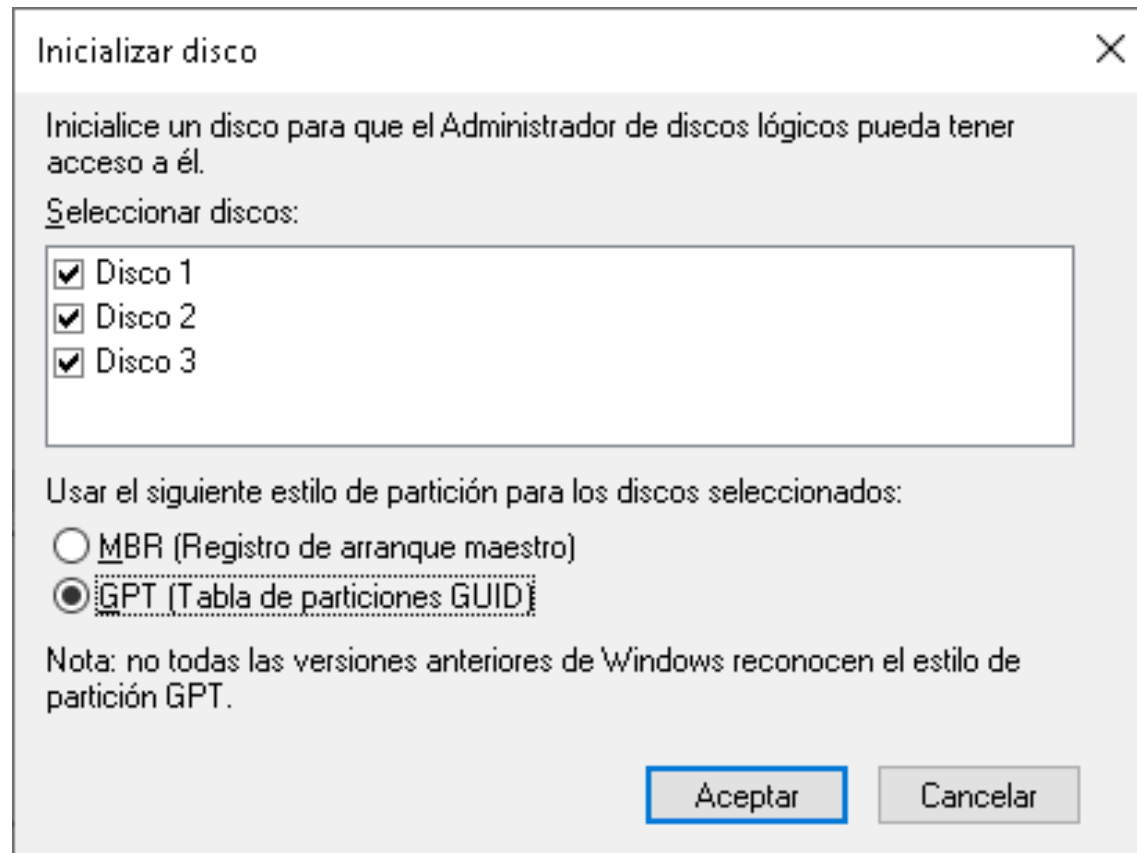
Administración de discos en Windows Server

En los Windows Server también podemos acceder vía Administración de equipos, que podemos abrir utilizando el enlace en el Administrador de servidor:



Inicializar disco

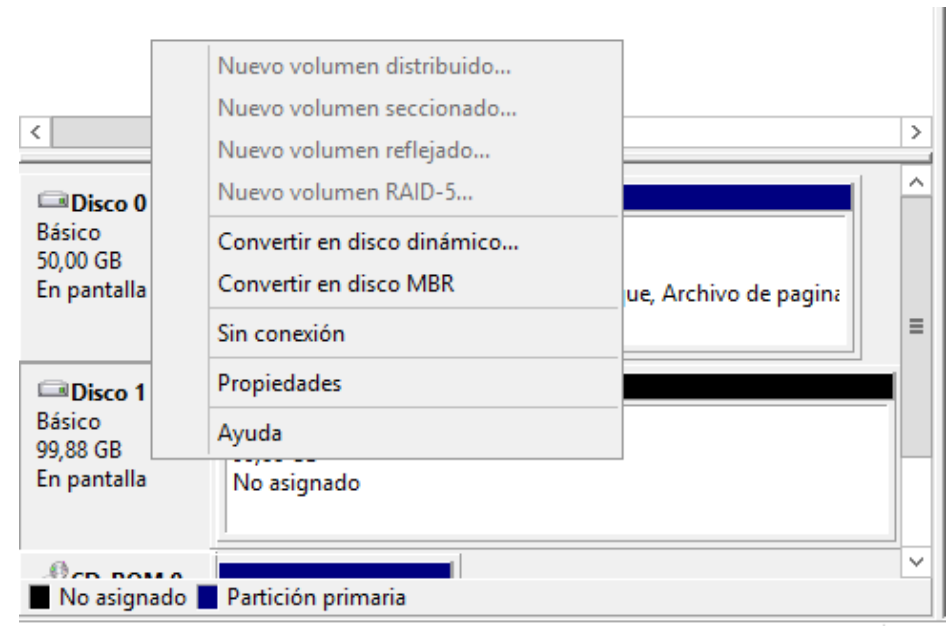
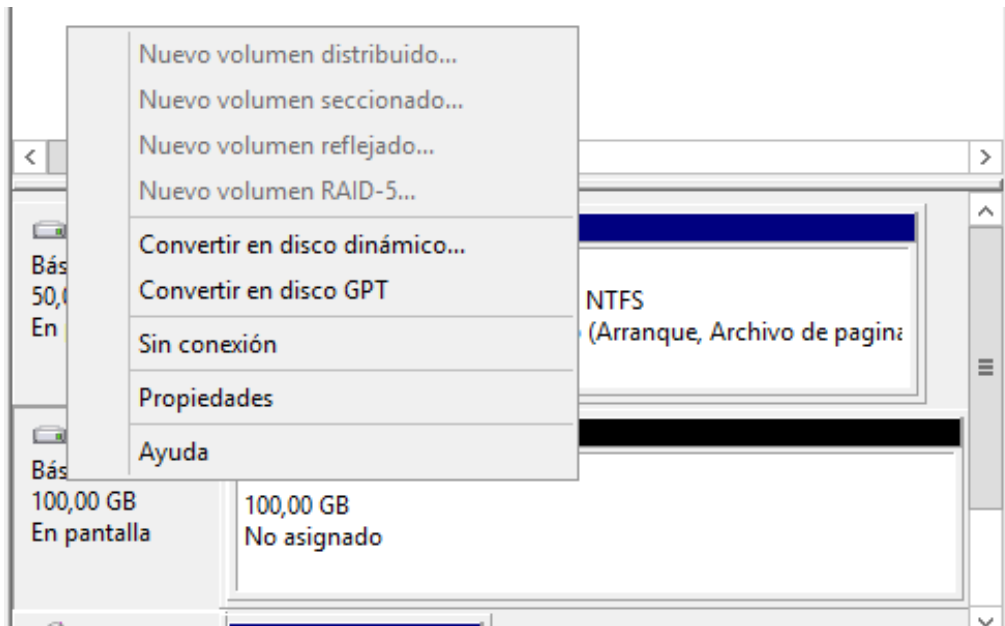
Los discos nuevos para el Administrador de discos requieren una inicialización en la que podremos seleccionar el tipo de particionado que le queremos aplicar (GPT o MBR):



Selección de modo básico o dinámico y GPT o MBR

Por defecto el disco quedará como disco básico, aunque podremos convertirlo a dinámico en cualquier momento

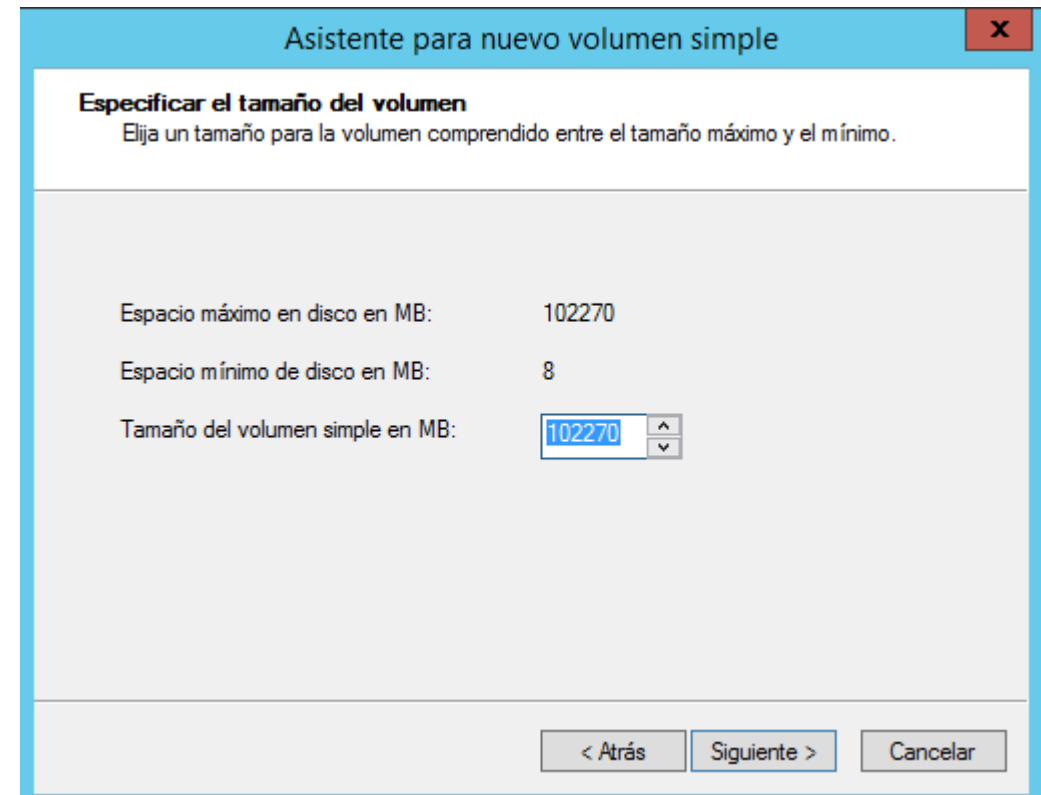
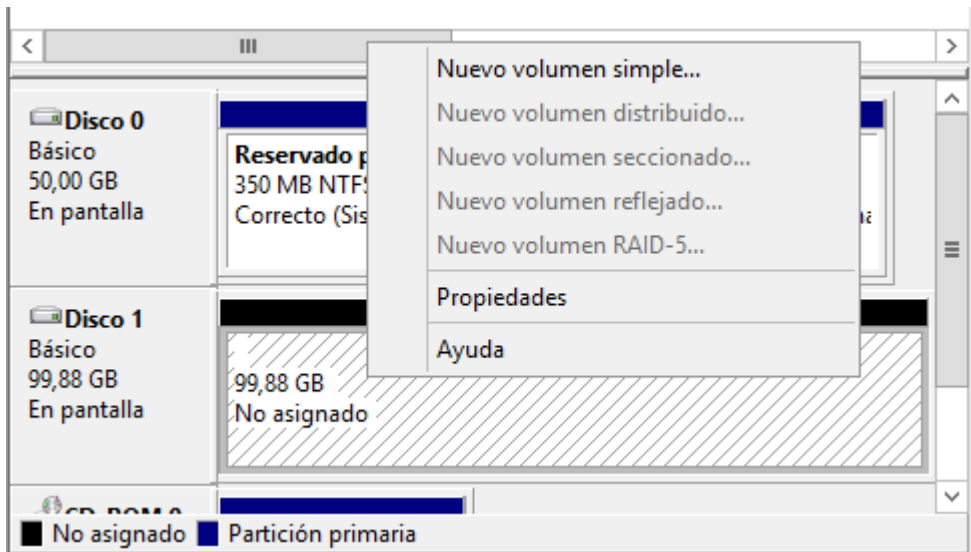
Podremos cambiar el tipo de particionado (de MBR a GPT y viceversa) más tarde si el disco está vacío y no tiene particiones. La conversión de disco simple a dinámico podemos hacerlo en cualquier momento



Creación de un volumen simple I

Para crear un volumen simple, una vez configurado el modo del disco y el tipo de particionado, utilizaremos el menú contextual en el diagrama que representa la distribución de particiones del disco.

En el primer paso indicamos el tamaño del volumen en MB:



Creación de un volumen simple II

En los siguientes pasos indicamos:

- La letra de unidad, que se utiliza en los sistemas windows para identificar cada partición (C:, D:, E:, ...)
- El sistema de archivos
- La etiqueta del volumen, un texto libre que nos ayudará para identificarla
- Opcionalmente marcar el check de formato rápido

Asistente para nuevo volumen simple

Asignar letra de unidad o ruta de acceso
Para facilitar el acceso, puede asignar una letra de unidad o ruta de acceso de unidad a su partición.

Asignar la letra de unidad siguiente:

Montar en la siguiente carpeta NTFS vacía:

No asignar una letra o ruta de acceso de unidad

< Atrás Siguiete > Cancelar

Asistente para nuevo volumen simple

Formatear la partición
Debe formatear esta partición antes de poder almacenar datos en ella.

Elija si desea formatear este volumen y, de ser así, la configuración que desea usar.

No formatear este volumen

Formatear este volumen con la configuración siguiente:

Sistema de archivos:

Tamaño de la unidad de asignación:

Etiqueta del volumen:

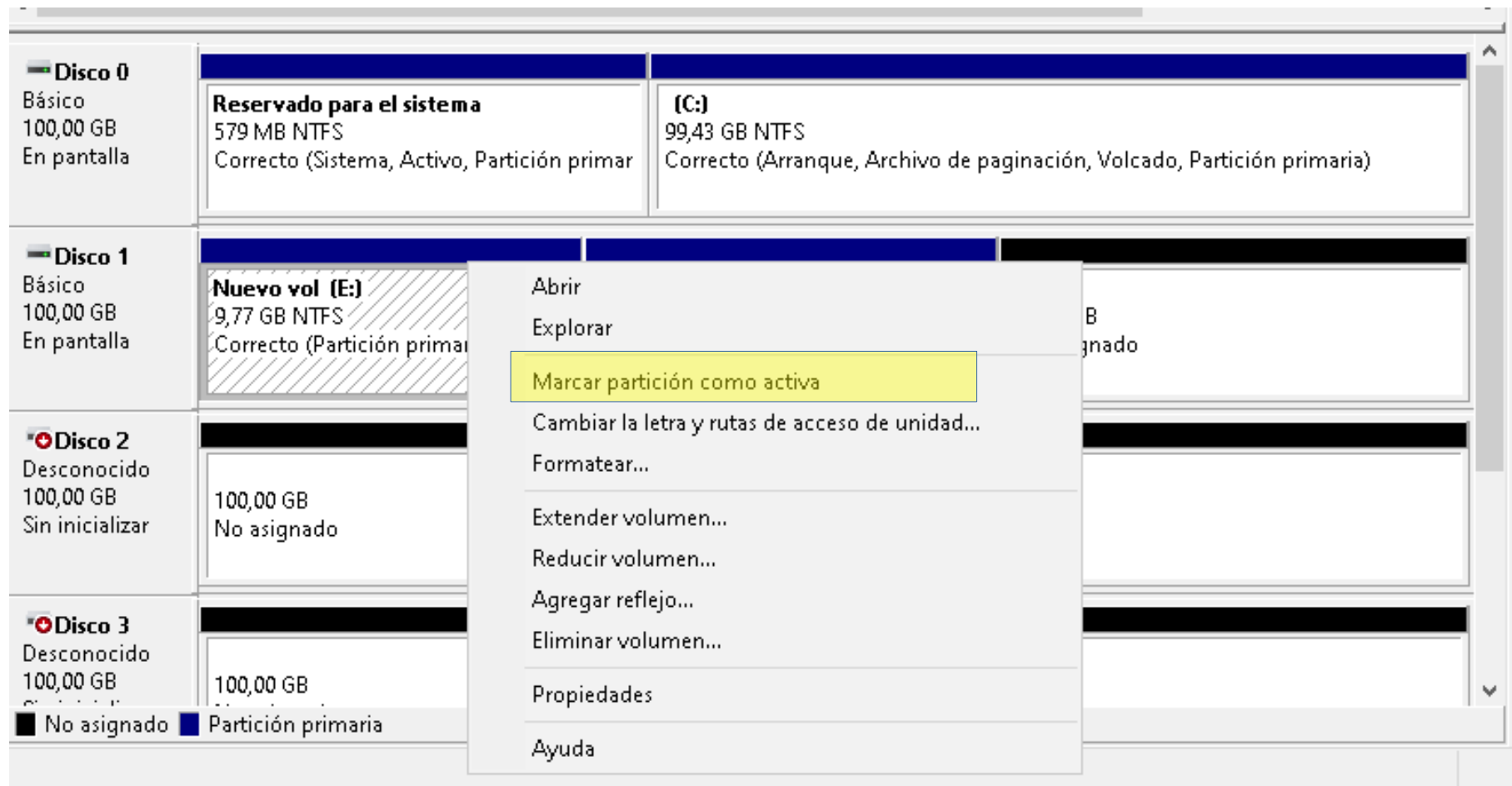
Dar formato rápido

Habilitar compresión de archivos y carpetas

< Atrás Siguiete > Cancelar

Marcar partición activa

Desde el administrador de archivos también podremos marcar la partición activa, pero sólo si tenemos un tipo de particionado MBR en el que sólo hay una única partición activa.



Diskpart

Diskpart

- Diskpart es un **gestor de particiones por línea de comandos** que forma parte de los sistemas Windows de Servidor y Escritorio
- Permite realizar las mismas operaciones que hacemos desde el Administrador de disco
- Una característica interesante del Diskpart es que es **posible crear scripts**. Esto permite automatizar y acelerar el particionado en situaciones en las que por ejemplo tengamos muchos ordenadores a particionar con el mismo disco.
- Antes de poder hacer algún tipo de operación sobre un disco o partición hay que hacer un **paso previo de selección del disco con el que trabajaremos**
- Los comandos que lancemos si son de disco o partición, aplicarán **sólo al seleccionado**. Es por eso que es muy importante asegurarse de cual es el seleccionado en todo momento

Iniciar y listar discos y particiones

- Abrir línea de comandos **cmd.exe**
- Ejecutar comando **diskpart**
- Listar discos existentes → **list disk**
- Seleccionar disco → **select disk [n]**
- Listar particiones → **list partition**
- Seleccionar partición → **select partition [n]**

```
C:\Windows\system32>diskpart
Microsoft DiskPart versión 6.3.9600
Copyright (C) 1999-2013 Microsoft Corporation.
En el equipo: WIN-SR08QLJEPRO

DISKPART> list disk

  Núm Disco  Estado      Tamaño  Disp    Din  Gpt
-----
Disco 0     En línea    50 GB   0 B
Disco 1     En línea    100 GB  60 GB  *   *
Disco 2     En línea    100 GB  0 B
Disco 3     En línea    100 GB  60 GB  *   *
Disco M0    Falta       0 B     0 B   *
```

```
DISKPART> select disk 1
El disco 1 es ahora el disco seleccionado.
```

```
DISKPART> list partition

  Núm Partición  Tipo                Tamaño  Desplazamiento
-----
Partición 1     Dinámico reserva   1024 KB  17 KB
Partición 2     Reservado           127 MB  1041 KB
Partición 3     Datos dinámicos    99 GB   128 MB

DISKPART> select partition 1
La partición 1 es ahora la partición seleccionada.
```

Selección de disco y partición

Podemos seleccionar con el comando **select**, indicando número de disco o partición en el que queremos realizar alguna operación.

Para ver cual es el seleccionado/a en un momento dado utilizamos el comando **list disk** o **list partition**. El seleccionado es que tiene una marca con una *****.

```
DISKPART> list disk

Núm Disco  Estado      Tamaño  Disp    Din    Gpt
-----
Disco 0    En línea    50 GB   0 B
Disco 1    En línea    100 GB  51 GB   *
Disco 2    En línea    100 GB  0 B
Disco 3    En línea    100 GB  60 GB   *   *
Disco M0   Falta       0 B     0 B     *
Disco M1   Falta       0 B     0 B     *
```

```
DISKPART> select disk 1

El disco 1 es ahora el disco seleccionado.
```

```
DISKPART> list disk

Núm Disco  Estado      Tamaño  Disp    Din    Gpt
-----
* Disco 0    En línea    50 GB   0 B
* Disco 1    En línea    100 GB  51 GB   *
Disco 2    En línea    100 GB  0 B
Disco 3    En línea    100 GB  60 GB   *   *
Disco M0   Falta       0 B     0 B     *
Disco M1   Falta       0 B     0 B     *
```

Limpiar particiones del disco y crear una partición

- Borrar tabla de particiones del disco seleccionado → **clean**
- Crear una partición (Tamaño en MB):
create partition primary size=10000
- Asignamos una letra a la partición
 - **assign letter = F**
 - Si utilizamos sólo **assign** asigna la primera disponible
- Formateamos la partición → **format fs=ntfs quick**

```
DISKPART> clean
DiskPart ha limpiado el disco satisfactoriamente.
DISKPART> create partition primary size=10000
DiskPart ha creado satisfactoriamente la partición especificada.
DISKPART> list partition
  Núm Partición  Tipo                Tamaño      Desplazamiento
-----
* Partición 1    Principal              9 GB      1024 KB
DISKPART> assign letter = F
DiskPart asignó correctamente una letra de unidad o punto de montaje.
DISKPART> format fs=ntfs quick
 100 por ciento completado
DiskPart formateó el volumen correctamente.
```

El comando clean y el comando clean all

- El comando **clean** borra la tabla de particiones del disco, pero **no borra los datos en la superficie del disco**
- Con el borrado de la tabla de particiones se dificulta el acceso a archivos desde el sistema operativo, pero la información **en el disco duro es recuperable**
- Para recuperar esta información habría que utilizar **software de recuperación de datos**, que escanean el disco para identificar los archivos existentes
- Hay una forma de hacer un borrado dificultando la recuperación de datos y haciéndola casi imposible con software de recuperación, que es utilizando el comando clean con la opción all (**clean all**)
- El comando **clean all**
 - Hace un **borrado seguro o borrado con ceros**
 - Este borrado sobrescribe todos los sectores del disco con 0's, de forma que ya queda información útil en la superficie del disco
 - Este tipo de borrado **requiere mucho más tiempo para ejecutarse** que el comando clean, variable según el tamaño del disco, por la acción de escritura de 0's en toda su superficie

Crear nueva partición en FAT32

Vemos un ejemplo de crear una partición primaria y formatearla en el sistema de archivos FAT32.

```
DISKPART> create partition primary size=20000
DiskPart ha creado satisfactoriamente la partición especificada.
DISKPART> list partition
```

Núm Partición	Tipo	Tamaño	Desplazamiento
Partición 1	Principal	9 GB	1024 KB
* Partición 2	Principal	19 GB	9 GB

```
DISKPART> format fs=fat32 quick
100 por ciento completado
DiskPart formateó el volumen correctamente.
DISKPART> assign letter=G
DiskPart asignó correctamente una letra de unidad o punto de montaje.
DISKPART> list partition
```

Núm Partición	Tipo	Tamaño	Desplazamiento
Partición 1	Principal	9 GB	1024 KB
* Partición 2	Principal	19 GB	9 GB

Crear una partición extendida con varias lógicas

```
DISKPART> list partition

  Núm Partición  Tipo                Tamaño      Desplazamiento
  -----
  Partición 1    Principal                9 GB      1024 KB
* Partición 2    Principal               19 GB         9 GB

DISKPART> create partition extended

DiskPart ha creado satisfactoriamente la partición especificada.

DISKPART> list partition

  Núm Partición  Tipo                Tamaño      Desplazamiento
  -----
  Partición 1    Principal                9 GB      1024 KB
  Partición 2    Principal               19 GB         9 GB
* Partición 0    Extendido              70 GB        29 GB

DISKPART> create partition logical size=10000

DiskPart ha creado satisfactoriamente la partición especificada.

DISKPART> create partition logical size=30000

DiskPart ha creado satisfactoriamente la partición especificada.

DISKPART> create partition logical

DiskPart ha creado satisfactoriamente la partición especificada.

DISKPART> list partition

  Núm Partición  Tipo                Tamaño      Desplazamiento
  -----
  Partición 1    Principal                9 GB      1024 KB
  Partición 2    Principal               19 GB         9 GB
  Partición 0    Extendido              70 GB        29 GB
  Partición 3    Lógico                  9 GB        29 GB
  Partición 4    Lógico                 29 GB        39 GB
* Partición 5    Lógico                 31 GB        68 GB
```

Uso de la ayuda para borrar una partición

- **help** (Nos lista todos los comandos)
- **help delete** (Nos lista las opciones del comando delete)

```
DISKPART> delete partition 5

Los argumentos especificados para este comando no son válidos.
Para obtener más información acerca del comando, escriba: HELP DELETE PARTITION

DISKPART> help delete

Microsoft DiskPart versión 6.3.9600

DISK          - Eliminar un disco que falta de la lista del disco.
PARTITION    - Eliminar la partición seleccionada.
VOLUME       - Eliminar el volumen seleccionado.

DISKPART> list partition

  Núm Partición  Tipo                Tamaño      Desplazamiento
-----
Partición 1     Principal             9 GB      1024 KB
Partición 2     Principal            19 GB           9 GB
Partición 0     Extendido            70 GB          29 GB
Partición 3     Lógico               9 GB          29 GB
Partición 4     Lógico              29 GB          39 GB
* Partición 5   Lógico              31 GB          68 GB

DISKPART> delete partition

DiskPart eliminó correctamente la partición seleccionada.

DISKPART> list partition

  Núm Partición  Tipo                Tamaño      Desplazamiento
-----
Partición 1     Principal             9 GB      1024 KB
Partición 2     Principal            19 GB           9 GB
Partición 0     Extendido            70 GB          29 GB
Partición 3     Lógico               9 GB          29 GB
Partición 4     Lógico              29 GB          39 GB
```

Uso de la ayuda para convertir un disco a GPT

- help convert
- clean
- convert GPT
- Particionamos
 - create partition ...
 - ...

```
DISKPART> clean
DiskPart ha limpiado el disco satisfactoriamente.
DISKPART> convert GPT
DiskPart convirtió correctamente el disco seleccionado en el formato GPT.
DISKPART> create partition primary SIZE=100000
DiskPart ha creado satisfactoriamente la partición especificada.
DISKPART> create partition primary SIZE=100000
DiskPart ha creado satisfactoriamente la partición especificada.
DISKPART> create partition primary SIZE=100000
DiskPart ha creado satisfactoriamente la partición especificada.
DISKPART> create partition primary SIZE=100000
DiskPart ha creado satisfactoriamente la partición especificada.
DISKPART> create partition primary SIZE=100000
DiskPart ha creado satisfactoriamente la partición especificada.
DISKPART> list partition
```

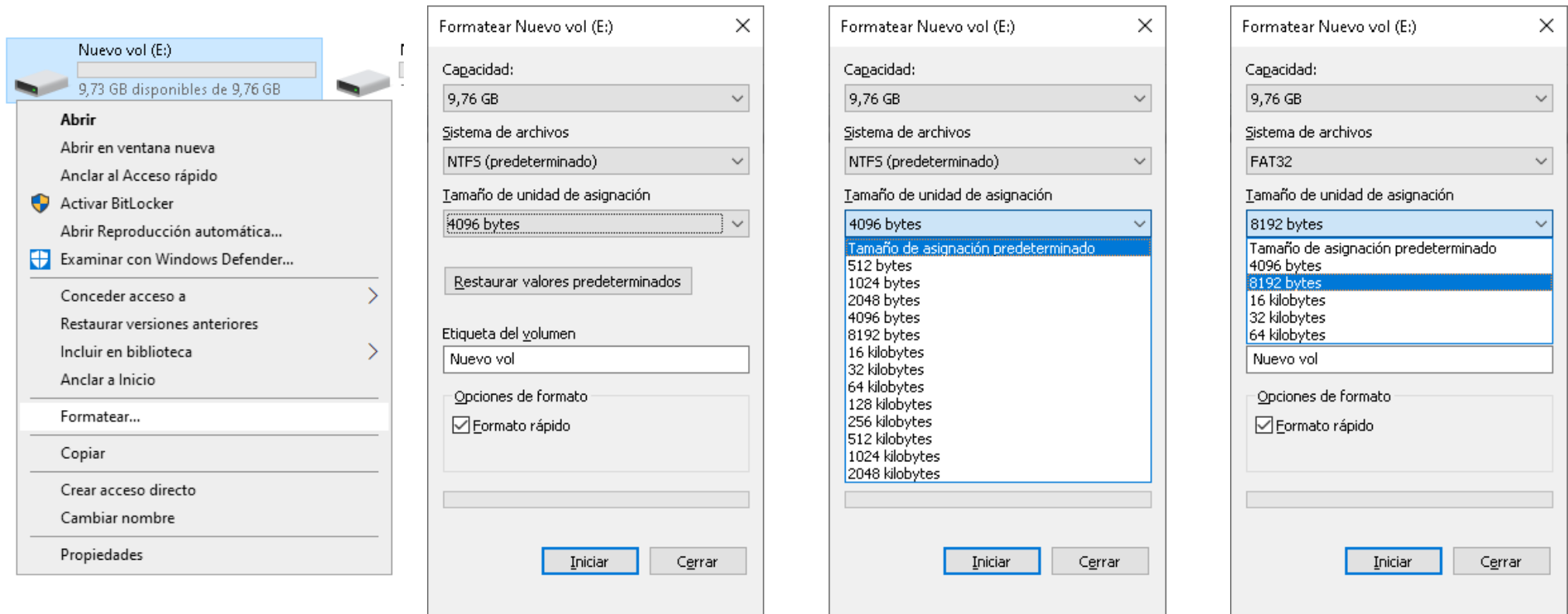
Núm Partición	Tipo	Tamaño	Desplazamiento
Partición 1	Reservado	128 MB	17 KB
Partición 2	Principal	9 GB	129 MB
Partición 3	Principal	9 GB	9 GB
Partición 4	Principal	9 GB	19 GB
Partición 5	Principal	9 GB	29 GB
* Partición 6	Principal	9 GB	39 GB

Formatear particiones fuera de los gestores de particiones

Ambos gestores de particiones Windows dan opciones para dejar las particiones formateadas y listas para usar, con una letra única que las identifica (C:, D:, E:, ...).

En este formateo aplicará el tamaño de clúster por defecto pero opcionalmente podremos especificar el tamaño del clúster u optar por el tamaño de clúster por defecto de dicho sistema de archivos formateándolas nosotros posteriormente en el menú contextual sobre la unidad de disco.

Un formateo sobre una partición con datos provocará un borrado de los datos que sólo podremos recuperar empleando herramientas especiales de recuperación de datos.



Diálogos de formateo de una partición en Windows en NTFS y FAT32.

El tamaño por defecto y las opciones de tamaño varían de un sistema de archivos a otro

Gestores de particiones en Linux

Nomenclatura de discos y particiones

- En linux los discos vienen especificados en nomenclaturas /dev/sda, /dev/sdb, /dev/sdc, ... en función de la cantidad de discos que dispongamos
 - dev → es la abreviatura de device
 - sd → Es la abreviatura de SCSI mass-storage driver
 - a, b, c, ...-→ Es la parte que nos distingue cada disco (a es el disco 1, b el disco 2, c el disco 3, ...)
- Las particiones se identifican para cada disco añadiendo el número de partición al final:
 - /dev/sda → /dev/sda1, /dev/sda2, ...
 - /dev/sdb → /dev/sdb1, /dev/sdb2, ...
- Cada disco y partición además tienen el **identificador único UUID** que podemos consultar con el comando **blkid**

```
root@debian:/media# blkid
/dev/sda1: UUID="4b5b566e-28a9-42a3-b05e-594f907c8aab" TYPE="ext4" PARTUUID="b528cbd0-01"
/dev/sda5: UUID="84af4c13-1e2c-4ad0-bdf5-fc9dd734e6e1" TYPE="swap" PARTUUID="b528cbd0-05"
/dev/sr0: UUID="2019-10-10-18-52-14-12" LABEL="VBox_GAs_6.0.14" TYPE="iso9660"
/dev/sdb1: UUID="af665adf-dc90-417b-983d-be235158338a" TYPE="ext4" PARTUUID="83c58727-73f3-a944-8ea1-5a9cac743157"
/dev/sdb2: UUID="417491b9-292f-45a4-b61e-65dc446ce7a6" TYPE="ext4" PARTUUID="03d8b847-7853-3442-95b0-447512a4ed50"
/dev/sdb3: UUID="BB27-3CFD" TYPE="vfat" PARTUUID="f557b7c1-9795-3045-910f-79916d233e92"
```

fdisk

- **fdisk** es la herramienta de administración de discos y particiones tradicional del shell de Linux
- **fdisk** ofrece un tipo de particionado tradicional, esto es, crear particiones de tamaño fijo en espacio contiguo lo que implica que es una solución poco flexible a la hora de afrontar situaciones en las que nos quedemos cortos de espacio o queramos configuraciones de almacenamiento con redundancia que nos permite RAID
- Vemos algunos ejemplos de comandos de **fdisk**:
 - **fdisk -l** → Lista todos los discos del sistema y sus tablas de particiones
 - **fdisk -l /dev/sda** → Lista la tabla de particiones del disco /dev/sda
 - **fdisk /dev/sdb** → Pasamos al modo de edición del disco /dev/sdb

Tipos de particionado en fdisk

fdisk permite el particionado GPT y DOS (que es la nomenclatura que utiliza para MBR) pero permite configurar otros dos adicionales, el SGI (IRIX/SGI-type) y SUN (BSD/Sun-type) que son otras alternativas menos comunes con características distintas.

En nomenclatura fdisk y cfdisk esto se denomina etiqueta de disco y en caso de que para un disco no esté definida le podremos ajustar la que queramos. Aún así fdisk y cfdisk dan opciones al inicio cuando detectan que no tiene una etiqueta asignada:

- fdisk le asigna por defecto DOS, esto es, particionado MBR
- cfdisk deja seleccionar entre los 4 tipos de etiquetas

```
usuario@usuario-VirtualBox: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
usuario@usuario-VirtualBox:~$ sudo cfdisk /dev/sdb  
usuario@usuario-VirtualBox:~$ sudo fdisk /dev/sdb  
  
Bienvenido a fdisk (util-linux 2.31.1).  
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.  
Tenga cuidado antes de utilizar la orden de escritura.  
  
El dispositivo no contiene una tabla de particiones reconocida.  
Se ha creado una nueva etiqueta de disco DOS con el identificador de disco 0x977  
be849.  
  
Orden (m para obtener ayuda):
```

```
usuario@usuario-VirtualBox: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
  
Selecciona el tipo de etiqueta  
gpt  
dos  
sgi  
sun  
  
Seleccione un tipo para crear una etiqueta o pulse 'L' para cargar el fichero de script.
```

Administración con fdisk

Ya en fdisk podemos listar todas las opciones típicas de administración de la tabla de particiones del disco seleccionado.

Todas las operaciones no se aplicarán hasta que confirmemos la escritura en la tabla de particiones con la opción `w`

```
Orden (m para obtener ayuda): p
Disco /dev/sda: 100 GiB, 107374182400 bytes, 209715200 sectores
Modelo de disco: VBOX HARDDISK
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0xb528cbd0

Disposit.  Inicio  Comienzo      Final  Sectores  Tamaño  Id Tipo
/dev/sda1  *              2048 207620095 207618048    99G 83 Linux
/dev/sda2              207622142 209713151   2091010   1021M  5 Extendida
/dev/sda5              207622144 209713151   2091008   1021M 82 Linux swap / Solaris

Orden (m para obtener ayuda): █
```

Ayuda:

DOS (MBR)

- a conmuta el indicador de iniciable
- b modifica la etiqueta de disco BSD anidada
- c conmuta el indicador de compatibilidad con DOS

General

- d borra una partición
- F lista el espacio libre no particionado
- l lista los tipos de particiones conocidos
- n añade una nueva partición
- p muestra la tabla de particiones
- t cambia el tipo de una partición
- v verifica la tabla de particiones
- i imprime información sobre una partición

Miscelánea

- m muestra este menú
- u cambia las unidades de visualización/entrada
- x funciones adicionales (sólo para usuarios avanzados)

Script

- I carga la estructura del disco de un fichero de script sfdisk
- O vuelca la estructura del disco a un fichero de script sfdisk

Guardar y Salir

- w escribe la tabla en el disco y sale
- q sale sin guardar los cambios

Crea una nueva etiqueta

- g crea una nueva tabla de particiones GPT vacía
- G crea una nueva tabla de particiones SGI (IRIX) vacía
- o crea una nueva tabla de particiones DOS vacía
- s crea una nueva tabla de particiones Sun vacía

Crear una partición primaria con fdisk

Para crear una partición utilizamos la **opción n**, tras lo cual nos pedirá:

- **Donde empieza la partición**, especificado como el número de sector. Por defecto cogerá el primer sector del espacio no particionado que encuentre
- Si estamos con etiqueta DOS (MBR), tendremos que indicar el **tipo de partición**, primaria o extendida
- El tamaño de la partición, que podremos especificar de dos formas:
 - Con el número del último sector de la partición (Si no especificamos nada coge automáticamente el último sector del espacio sin particionar, y es lo que haremos cuando queramos que la partición ocupe todo el espacio disponible)
 - Con el tamaño en la forma **+tamaño{K,M,G,T,P}**. Cada letra representa una unidad de almacenamiento KiB, MiB, GiB, TiB y PiB

Vemos un ejemplo para crear una partición primaria de 10 GiB en un particionado MBR:

```
Orden (m para obtener ayuda): n
Tipo de partición
  p  primaria (0 primaria(s), 0 extendida(s), 4 libre(s))
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): p
Número de partición (1-4, valor predeterminado 1):
Primer sector (2048-209715199, valor predeterminado 2048):
Último sector, +sectores o +tamaño{K,M,G,T,P} (2048-209715199, valor predeterminado 209715199): +10G

Crea una nueva partición 1 de tipo 'Linux' y de tamaño 10 GiB.
```

Crear una partición extendida con fdisk

Vamos ahora a crear una partición extendida para lo cual nos pide la misma información sólo que ahora tendremos que indicar que es una extendida

Vemos un ejemplo para crear una extendida que ocupa todo el espacio disponible. Dejando todos los valores por defecto se crea justo después de la primaria creada antes y coge todo el espacio disponible hasta el final:

```
Orden (m para obtener ayuda): n
Tipo de partición
  p  primaria (1 primaria(s), 0 extendida(s), 3 libre(s))
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): e
Número de partición (2-4, valor predeterminado 2):
Primer sector (20973568-209715199, valor predeterminado 20973568):
Último sector, +sectores o +tamaño{K,M,G,T,P} (20973568-209715199, valor predeterminado 209715199):

Crea una nueva partición 2 de tipo 'Extended' y de tamaño 90 GiB.
```

Crear particiones lógicas con fdisk

Con el particionado anterior (una primaria y una extendida que ocupan todo el disco) al crear ahora nuevas particiones sólo nos deja crear particiones lógicas que solicitan los mismos datos.

Vemos como crear tres lógicas de 35 GiB, 25 GiB y la última sin especificar ningún tamaño, por lo que cogerá el disponible, que en ejemplo son 30 GiB:

```
Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 6
Primer sector (94377984-209715199, valor predeterminado 94377984):
Último sector, +sectores o +tamaño{K,M,G,T,P} (94377984-209715199, valor predeterminado 209715199): +25G

Crea una nueva partición 6 de tipo 'Linux' y de tamaño 25 GiB.

Orden (m para obtener ayuda): n
Se está utilizando todo el espacio para particiones primarias.
Se añade la partición lógica 7
Primer sector (146808832-209715199, valor predeterminado 146808832):
Último sector, +sectores o +tamaño{K,M,G,T,P} (146808832-209715199, valor predeterminado 209715199):

Crea una nueva partición 7 de tipo 'Linux' y de tamaño 30 GiB.
```

Imprimir la tabla de particiones

Con la **opción p** podremos ver la configuración de particiones realizada:

```
Orden (m para obtener ayuda): p
Disco /dev/sdb: 100 GiB, 107374182400 bytes, 209715200 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0xf7e79887

Dispositivo Inicio Comienzo Final Sectores Tamaño Id Tipo
/dev/sdb1          2048 20973567 20971520 10G 83 Linux
/dev/sdb2          20973568 209715199 188741632 90G 5 Extendida
/dev/sdb5          20975616 94375935 73400320 35G 83 Linux
/dev/sdb6          94377984 146806783 52428800 25G 83 Linux
/dev/sdb7         146808832 209715199 62906368 30G 83 Linux
```

Antes de poder utilizar ninguna de estas particiones tenemos que:

- Escribir la tabla de particiones a disco con la **opción w**
- Formatear la partición con el comando **mkfs**

```
Orden (m para obtener ayuda): w
Se ha modificado la tabla de particiones.
Llamando a ioctl() para volver a leer la tabla de particiones.
Se están sincronizando los discos.

usuario@usuario-VirtualBox:~$ sudo mkfs -t ext4 /dev/sdb1
mke2fs 1.44.1 (24-Mar-2018)
Se está creando un sistema de ficheros con 2621440 bloques de 4k y 655360 nodos-i
UUID del sistema de ficheros: 885efbaf-c5cf-4fe9-9c55-831ce1edbd3b
Respaldo del superbloque guardado en los bloques:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Reservando las tablas de grupo: hecho
Escribiendo las tablas de nodos-i: hecho
Creando el fichero de transacciones (16384 bloques): hecho
Escribiendo superbloques y la información contable del sistema de archivos: hecho
```

Borrado de la tabla de particiones en fdisk

Si quisiéramos borrar una tabla de particiones en fdisk nos bastaría con cambiar la etiqueta, por ejemplo de DOS (MBR) a GPT con la **opción g**, o a la inversa con la **opción o**. Es una opción equivalente al clean de diskpart

Vemos un ejemplo pasando de la tabla de particiones DOS anterior a GPT:

```
usuario@usuario-VirtualBox:~$ sudo fdisk /dev/sdb
```

```
Bienvenido a fdisk (util-linux 2.31.1).
```

```
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.  
Tenga cuidado antes de utilizar la orden de escritura.
```

```
Orden (m para obtener ayuda): p
```

```
Disco /dev/sdb: 100 GiB, 107374182400 bytes, 209715200 sectores
```

```
Unidades: sectores de 1 * 512 = 512 bytes
```

```
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
```

```
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
```

```
Tipo de etiqueta de disco: dos
```

```
Identificador del disco: 0xf059adba
```

Dispositivo	Inicio	Comienzo	Final	Sectores	Tamaño	Id	Tipo
/dev/sdb1		2048	20973567	20971520	10G	83	Linux
/dev/sdb2		20973568	209715199	188741632	90G	5	Extendida
/dev/sdb5		20975616	94375935	73400320	35G	83	Linux
/dev/sdb6		94377984	146806783	52428800	25G	83	Linux
/dev/sdb7		146808832	209715199	62906368	30G	83	Linux

```
Orden (m para obtener ayuda): g
```

```
Se ha creado una nueva etiqueta de disco GPT (GUID: 2F584C2B-4DE7-FD45-BE0C-6163899F6A1B).  
La firma antigua dos se borrará mediante una orden de escritura.
```

```
Orden (m para obtener ayuda): p
```

```
Disco /dev/sdb: 100 GiB, 107374182400 bytes, 209715200 sectores
```

```
Unidades: sectores de 1 * 512 = 512 bytes
```

```
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
```

```
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
```

```
Tipo de etiqueta de disco: gpt
```

```
Identificador del disco: 2F584C2B-4DE7-FD45-BE0C-6163899F6A1B
```

```
Orden (m para obtener ayuda): w
```

```
Se ha modificado la tabla de particiones.
```

```
Llamando a ioctl() para volver a leer la tabla de particiones.
```

```
Se están sincronizando los discos.
```

Cambiar el tipo de partición en fdisk (GPT)

fdisk por defecto crea las particiones como “Linux filesystem” que se corresponde con el sistema de ficheros ext4. Podremos cambiar el sistema de ficheros asociado a esta partición con la opción t, y a modo de ayuda listar todos los tipos de particiones con la opción L.

Esta lista no es muy concreta en los tipos, las NTFS y las FAT32 son ambas del tipo 11 Microsoft basic data y todos los tipos de ext (ext2, ext3 y ext4) son Linux filesystem.

La concreción del sistema de ficheros se lo daremos después cuando formateemos la partición con el comando mkfs.

```
Orden (m para obtener ayuda): t
Número de partición (1-4, valor predeterminado 4): 4
Tipo de partición (teclea L para ver todos los tipos): L
 1 EFI System                C12A7328-F81F-11D2-BA4B-00A0C93EC93B
 2 MBR partition scheme      024DEE41-33E7-11D3-9069-0008C781F39F
 3 Intel Fast Flash          D3BFE2DE-3DAF-11DF-BA40-E3A556D89593
 4 BIOS boot                  21686148-6449-6E6F-744E-656564454649
 5 Sony boot partition        F4019732-066E-4E12-8273-346C5641494F
 6 Lenovo boot partition      BFBFAFE7-A34F-448A-9A5B-6213EB736C22
 7 PowerPC PreP boot          9E1A2D38-C612-4316-AA26-8B49521E5A8B
 8 ONIE boot                   7412F7D5-A156-4B13-81DC-867174929325
 9 ONIE config                 D4E6E2CD-4469-46F3-B5CB-1BFF57AFC149
10 Microsoft reserved         E3C9E316-0B5C-4DB8-817D-F92DF00215AE
11 Microsoft basic data       EBD0A0A2-B9E5-4433-87C0-68B6B72699C7
12 Microsoft LDM metadata     5808C8AA-7E8F-42E0-85D2-E1E90434CFB3
13 Microsoft LDM data         AF9B60A0-1431-4F62-BC68-3311714A69AD
14 Windows recovery environment DE94BBA4-06D1-4D40-A16A-BFD50179D6AC
15 IBM General Parallel Fs     37AFFC90-EF7D-4E96-91C3-2D7AE055B174
16 Microsoft Storage Spaces   E75CAF8F-F680-4CEE-AFA3-B001E56EFC2D
17 HP-UX data                  75894C1E-3AEB-11D3-B7C1-7B03A0000000
18 HP-UX service               E2A1E728-32E3-11D6-A682-7B03A0000000
19 Linux swap                   0657FD6D-A4AB-43C4-84E5-0933C84B4F4F
20 Linux filesystem             0FC63DAF-8483-4772-8E79-3D69D8477DE4
21 Linux server data           3B8F8425-20E0-4F3B-907F-1A25A76F98E8
22 Linux root (x86)            44479540-F297-41B2-9AF7-D131D5F0458A
23 Linux root (ARM)            69DAD710-2CE4-4E3C-B16C-21A1D49ABED3
24 Linux root (x86-64)         4F68BCE3-E8CD-4DB1-96E7-FBCAF984B709
25 Linux root (ARM-64)         B921B045-1DF0-41C3-AF44-4C6F280D3FAE
26 Linux root (IA-64)         993D8D3D-F80E-4225-855A-9DAF8ED7EA97
27 Linux reserved              8DA63339-0007-60C0-C436-083AC8230908
28 Linux home                   933AC7E1-2EB4-4F13-B844-0E14E2AEF915
29 Linux RAID                   A19D880F-05FC-4D3B-A006-743F0F84911E
30 Linux extended boot         BC13C2FF-59E6-4262-A352-B275FD6F7172
31 Linux LVM                     E6D6D379-F507-44C2-A23C-238F2A3DF928
32 FreeBSD data                 516E7CB4-6ECF-11D6-8FF8-00022D09712B
33 FreeBSD boot                 83BD6B9D-7F41-11DC-BE0B-001560B84F0F
34 FreeBSD swap                 516E7CB5-6ECF-11D6-8FF8-00022D09712B
35 FreeBSD UFS                   516E7CB6-6ECF-11D6-8FF8-00022D09712B
```

Dispositivo	Comienzo	Final	Sectores	Tamaño	Tipo
/dev/sdc1	2048	10242047	10240000	4,9G	Linux swap
/dev/sdc2	10242048	30722047	20480000	9,8G	Sistema de ficheros de Linux
/dev/sdc3	30722048	71682047	40960000	19,5G	Datos básicos de Microsoft
/dev/sdc4	71682048	133122047	61440000	29,3G	Datos básicos de Microsoft

Ejemplo de tabla de particiones en GPT con una swap, una ext4, una NTFS y una FAT32

Cambiar el tipo de partición en fdisk (MBR)

Los tipos de particiones en modo MBR son distintos y los códigos asociados también, que incluso aparecen codificados en hexadecimal.

También veremos que MBR es más concreta con los tipos de archivos Windows, pero igualmente será cuando les demos formatos con mkfs que concretaremos el sistema de archivos.

```
Orden (m para obtener ayuda): t
Número de partición (1-6, valor predeterminado 6):
Código hexadecimal (escriba L para ver todos los códigos): L

0  Vacía                24  DOS de NEC            81  Minix / Linux a bf  Solaris
1  FAT12                27  NTFS de WinRE o      82  Linux swap / So c1  DRDOS/sec (FAT-
2  XENIX root           39  Plan 9               83  Linux              c4  DRDOS/sec (FAT-
3  XENIX usr            3c  PartitionMagic      84  OS/2 oculto o h c6  DRDOS/sec (FAT-
4  FAT16 <32M          40  Venix 80286         85  Linux extendida c7  Syrix
5  Extendida           41  PPC PReP Boot       86  Conjunto de vol da  Datos sin SF
6  FAT16               42  SFS                  87  Conjunto de vol db  CP/M / CTOS / .
7  HPFS/NTFS/exFAT     4d  QNX4.x              88  Linux plaintext de  Utilidad Dell
8  AIX                 4e  QNX4.x segunda      8e  Linux LVM           df  BootIt
9  AIX arrancable     4f  QNX4.x tercera      93  Amoeba              e1  DOS access
a  Gestor de arran 50  OnTrack DM          94  Amoeba BBT          e3  DOS R/O
b  W95 FAT32           51  OnTrack DM6 Aux    9f  BSD/OS              e4  SpeedStor
c  W95 FAT32 (LBA)    52  CP/M                 a0  Hibernación de     ea  alineamiento Ru
e  W95 FAT16 (LBA)    53  OnTrack DM6 Aux    a5  FreeBSD             eb  BeOS fs
f  W95 Ext'd (LBA)    54  OnTrackDM6          a6  OpenBSD             ee  GPT
10 OPUS                55  EZ-Drive            a7  NeXTSTEP            ef  EFI (FAT-12/16/
11 FAT12 oculta       56  Golden Bow          a8  UFS de Darwin       f0  inicio Linux/PA
12 Compaq diagnost 5c  Priam Edisk          a9  NetBSD              f1  SpeedStor
14 FAT16 oculta <3   61  SpeedStor           ab  arranque de Dar f4  SpeedStor
16 FAT16 oculta       63  GNU HURD o SysV    af  HFS / HFS+          f2  DOS secondary
17 HPFS/NTFS ocult 64  Novell Netware      b7  BSDI fs             fb  VMFS de VMware
18 SmartSleep de A  65  Novell Netware      b8  BSDI swap           b8  VMKCORE de VMwa
1b FAT32 de W95 oc 70  DiskSecure Mult    bb  Boot Wizard hid    fd  Linux raid auto
1c FAT32 de W95 (L 75  PC/IX               bc  Acronis FAT32 L fe  LANstep
1e FAT16 de W95 (L 80  Minix antiguo       be  arranque de Sol ff  BBT
Código hexadecimal (escriba L para ver todos los códigos):
```

Dispositivo	Inicio	Comienzo	Final	Sectores	Tamaño	Id	Tipo
/dev/sdb1		2048	10242047	10240000	4,9G	82	Linux swap / Solaris
/dev/sdb2		10242048	30722047	20480000	9,8G	83	Linux
/dev/sdb3		30722048	71682047	40960000	19,5G	7	HPFS/NTFS/exFAT
/dev/sdb4		71682048	209715199	138033152	65,8G	5	Extendida
/dev/sdb5		71684096	133124095	61440000	29,3G	83	Linux
/dev/sdb6		133126144	209715199	76589056	36,5G	b	W95 FAT32

Ejemplo de tabla de particiones en MBR con una swap, una ext4, una NTFS y una extendida con una ext4 y una FAT32

Formatear particiones con mkfs

Antes de poder escribir datos en una partición particionada con fdisk tendremos que aplicarle un formato, para lo cual utilizaremos el **comando mkfs** (make filesystem).

Vemos algunos ejemplos de uso:

- **mkfs.ext4 /dev/sdb1** → Formatea la partición 1 del disco /dev/sdb con formato ext4
- **mkfs -t ext4 /dev/sdb2** → Igual al anterior
- **mkfs.vfat /dev/sdb3** → Formatea la partición 3 en FAT32
- **mkfs -t vfat /dev/sdb3** → Igual que el anterior
- **mkfs.ntfs /dev/sdb4** → Formatea la partición 4 del disco /dev/sdb con formato NTFS
- **mkfs -t ntfs /dev/sdb4** → Igual que el anterior

Para el caso de una partición swap utilizaremos el comando mkswap:

- **mkswap /dev/sdb1** → Formatea la partición 1 del disco /dev/sdb como Linux swap

Un formateo sobre una partición con datos provocará un borrado de los datos que sólo podremos recuperar empleando herramientas especiales de recuperación de datos.

Montaje manual de particiones

- El paso final para poder leer y escribir de particiones es montarlas en una ruta del sistema de archivos para lo cual utilizaremos el comando **mount**:
 - **mount /dev/sdb1 /media/datos1** → Monta la partición 1 del disco /dev/sdb en la ruta /media/datos1
 - **umount /dev/sdb1** → Desmonta la partición 1 del disco /dev/sdb de donde esté montada
- El montaje de una partición lo tendremos que repetir tras cada arranque, pero podemos hacerlo automáticamente en el arranque del sistema editando el fichero **/etc/fstab**
- El fichero /etc/fstab nos permita automatizar el montaje de particiones, bien configurando o no su montaje automático en el arranque como configurar otras opciones que queremos que se apliquen por defecto cuando montamos la partición

cfdisk

cfdisk

cfdisk permite hacer lo mismo que fdisk pero con un menú más interactivo que tenemos en la parte inferior con las opciones más comunes. En este menú nos movemos con los cursores izquierdo y derecho y seleccionamos los comandos con Enter.

cfdisk, tampoco formatea las particiones, por lo que tendremos que seguir utilizando el comando mkfs después del particionado y después montarlo en una ruta del sistema de archivos.

Para abrir cfdisk lanzamos el comando indicando el disco que queremos particionar en la forma **cfdisk /dev/sda** o **cfdisk /dev/sdb**

```
usuario@usuario-VirtualBox: ~
Archivo  Editar  Ver     Buscar  Terminal  Ayuda

Disco: /dev/sdb
Tamaño: 100 GiB, 107374182400 bytes, 209715200 sectores
Etiqueta: gpt, identificador: 2F584C2B-4DE7-FD45-BE0C-6163899F6A1B

Dispositivo      Comienzo      Final          Sectores      Tamaño Tipo
>> Espacio libre  2048          209715166     209713119     100G

[ Nueva ] [ Salir ] [ Ayuda ] [ Escribir ] [ Vuelca ]

Crea una nueva partición a partir del espacio libre
```

Crear una partición con fdisk

Para crear una partición seleccionamos la opción Nueva e indicamos el tamaño de la misma, en una unidad de almacenamiento (G para GiB, M para MiB, ...) o especificando un número de sectores:

```
usuario@usuario-VirtualBox: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
Disco: /dev/sdb  
Tamaño: 100 GiB, 107374182400 bytes, 209715200 sectores  
Etiqueta: gpt, identificador: 2F584C2B-4DE7-FD45-BE0C-6163899F6A1B  
Dispositivo Comienzo Final Sectores Tamaño Tipo  
>> Espacio libre 2048 209715166 209713119 100G  
Tamaño de la partición: 10G  
Puede estar seguido de M (MiB), G (GiB), T (TiB) o S (sectores).
```

```
usuario@usuario-VirtualBox: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
Disco: /dev/sdb  
Tamaño: 100 GiB, 107374182400 bytes, 209715200 sectores  
Etiqueta: gpt, identificador: 2F584C2B-4DE7-FD45-BE0C-6163899F6A1B  
Dispositivo Comienzo Final Sectores Tamaño Tipo  
>> /dev/sdb1 2048 20973567 20971520 10G Sistema de ficheros de Lin  
Espacio libre 20973568 209715166 188741599 90G  
UUID de la partición: 82631516-260C-644A-8EBE-8B6516F601A5  
Tipo de partición: Sistema de ficheros de Linux (0FC63DAF-8483-4772-8  
[ Suprimir ] [ Resize ] [ Salir ] [ Tipo ] [ Ayuda ] [ Escribir ]  
[ Vuelca ]  
Sale del programa sin escribir los cambios
```

Vista de la tabla de particiones y redimensionado

Además de una interfaz más amigable cfdisk tiene un par de ventajas adicionales:

- Siempre tenemos a la vista en la parte superior la configuración de particionado que estamos haciendo. Esta no se aplicará definitivamente en el disco hasta que seleccionemos la **opción Escribir**. Utilizaremos los cursores arriba y abajo para seleccionaremos la partición o el espacio sin particionando sobre el queramos realizar alguna de las opciones del menú inferior.
- Nos deja redimensionar el tamaño de las particiones, aumentando o reduciendo (**Opción resize**). Para aumentar será necesario tener espacio sin particionar a continuación.

```
usuario@usuario-VirtualBox: ~
Archivo Editar Ver Buscar Terminal Ayuda

Disco: /dev/sdb
Tamaño: 100 GiB, 107374182400 bytes, 209715200 sectores
Etiqueta: gpt, identificador: 2F584C2B-4DE7-FD45-BE0C-6163899F6A1B

Dispositivo Comienzo Final Sectores Tamaño Tipo
/dev/sdb1 2048 20973567 20971520 10G Sistema de ficheros de Lin
>> /dev/sdb2 20973568 83888127 62914560 30G Sistema de ficheros de Lin
Espacio libre 83888128 209715166 125827039 60G

UUID de la partición: 78A71AB2-5CF8-C74F-B948-B1DDD1A987B6
Tipo de partición: Sistema de ficheros de Linux (0FC63DAF-8483-4772-8E79-

[ Suprimir ] [ Resize ] [ Salir ] [ Tipo ] [ Ayuda ] [ Escribir ]
[ Vuelca ]

Partition 2 resized.
```

```
usuario@usuario-VirtualBox: ~
Archivo Editar Ver Buscar Terminal Ayuda

Disco: /dev/sdb
Tamaño: 100 GiB, 107374182400 bytes, 209715200 sectores
Etiqueta: gpt, identificador: 2F584C2B-4DE7-FD45-BE0C-6163899F6A1B

Dispositivo Comienzo Final Sectores Tamaño Tipo
/dev/sdb1 2048 20973567 20971520 10G Sistema de ficheros de Lin
>> /dev/sdb2 20973568 83888127 62914560 30G Sistema de ficheros de Lin
Espacio libre 83888128 209715166 125827039 60G

UUID de la partición: 78A71AB2-5CF8-C74F-B948-B1DDD1A987B6
Tipo de partición: Sistema de ficheros de Linux (0FC63DAF-848

New size: 40G

Puede estar seguido de M (MiB), G (GiB), T (TiB) o S (sectores).
```

Comandos fdisk en cfdisk

Aunque echemos de menos algunas opciones de fdisk podemos ver en la ayuda una lista de comandos que nos dan más opciones que las que nos presenta cfdisk en su menú y que estas comparten la nomenclatura de comando con fdisk. Entre estas tenemos la opción de ajustar la partición activa (indicador de iniciable)

```
cfdisk es un programa de particiones de disco basado en curses.
Permite crear, suprimir y modificar particiones en dispositivo de bloques.

Orden      Significado
-----
b          Conmuta el indicador de iniciable de la partición actual
d          Suprime la partición actual
h          Imprime esta pantalla
n          Crea una nueva partición a partir del espacio libre
q          Sale del programa sin escribir la tabla de particiones
s          Corrige el orden de las particiones (solo cuando hay desajuste)
t          Cambia el tipo de partición
u          Vuelca la estructura del disco a un fichero de script compatible con sfdisk
W          Escribe la tabla de particiones en el disco (W debe teclearse en mayúsculas).
           esta operación de escritura puede causar la destrucción de
           datos del disco, por lo que debe confirmarla o rechazarla
e x       Muestra/oculta información adicional sobre una partición
Flecha arriba   Desplaza el cursor a la partición anterior
Flecha abajo    Desplaza el cursor a la partición siguiente
Flecha izquierda Desplaza el cursor a la opción de menú anterior
Flecha derecha  Desplaza el cursor a la opción de menú siguiente

Nota: todas las órdenes pueden escribirse en mayúsculas o minúsculas
(salvo W para operaciones de escritura).

Utilice lsblk(8) o partx(8) para ver más detalles del dispositivo.

Copyright (C) 2014-2017 Karel Zak <kzak@redhat.com>
```

Parted y GParted

Parted

Como alternativa a fdisk tenemos **Parted** ([Parted en la wiki de ArchLinux](#)) es otro gestor de particiones que es posible que no tengamos instalado por defecto y que tengamos que instalar con **apt install parted**. Este admite dos modos de particionado:

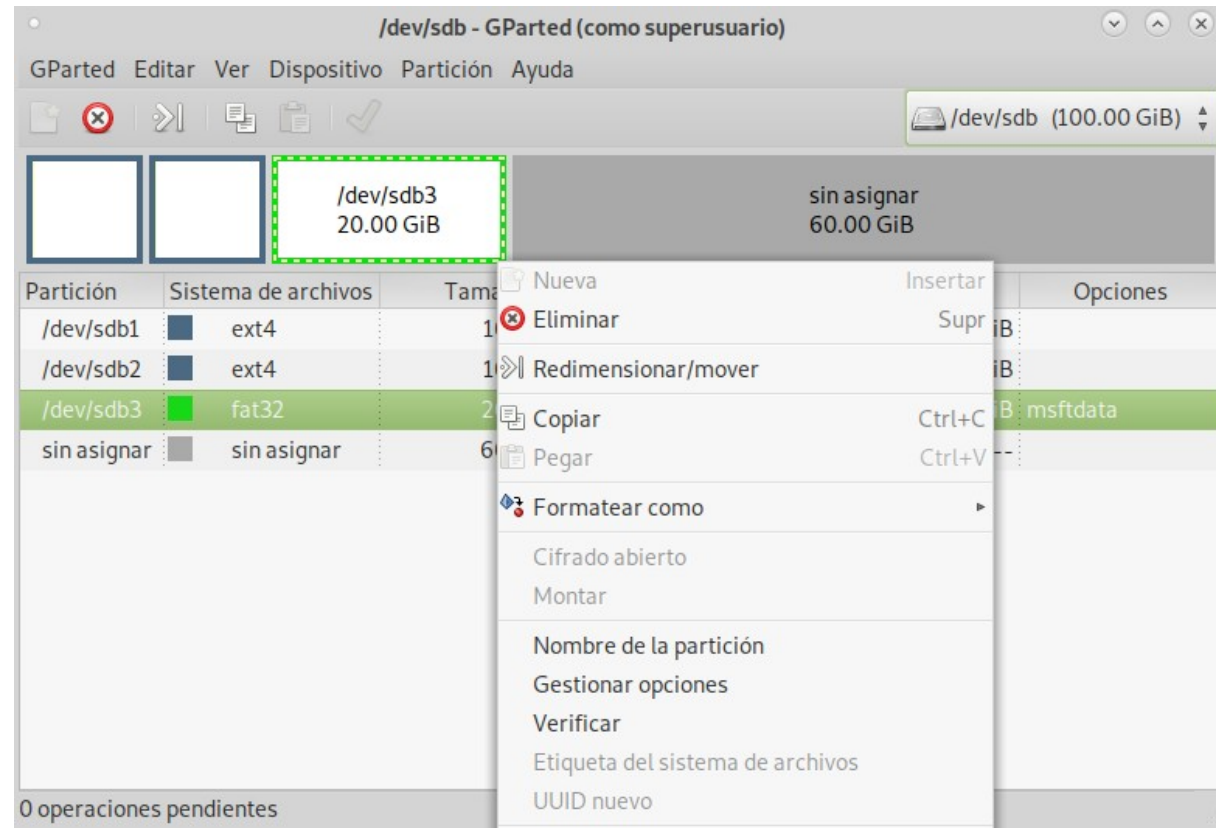
- **Modo interactivo**, con una consola en la que ejecutamos comandos para particionar, de una forma similar al diskpart de los sistemas Microsoft
- **Modo consola**, escribiendo directamente el comando completo para crear la partición

```
(parted) help
align-check TYPE N          check partition N for TYPE(min|opt) alignment
help [COMMAND]             print general help, or help on COMMAND
mklabel,mktable LABEL-TYPE create a new disklabel (partition table)
mkpart PART-TYPE [FS-TYPE] START END make a partition
name NUMBER NAME           name partition NUMBER as NAME
print [devices|free|list,all|NUMBER] display the partition table, available devices,
quit                       exit program
rescue START END          rescue a lost partition near START and END
resizepart NUMBER END     resize partition NUMBER
rm NUMBER                 delete partition NUMBER
select DEVICE             choose the device to edit
disk_set FLAG STATE       change the FLAG on selected device
disk_toggle [FLAG]       toggle the state of FLAG on selected device
set NUMBER FLAG STATE    change the FLAG on partition NUMBER
toggle [NUMBER [FLAG]]   toggle the state of FLAG on partition NUMBER
unit UNIT                set the default unit to UNIT
version                  display the version number and copyright inform
```

Gparted (Gnome parted)

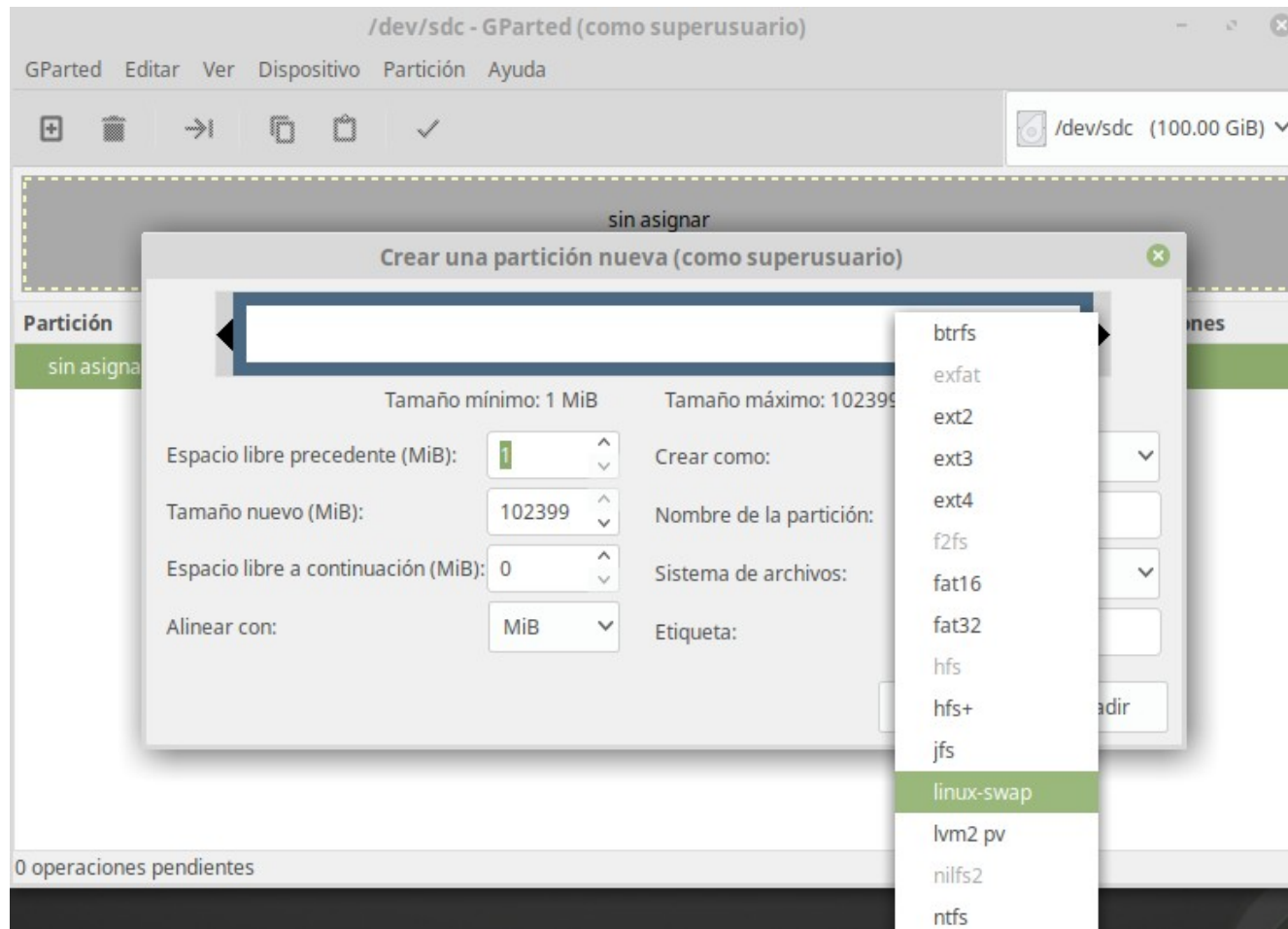
Además tenemos el gestor de particiones gráfico **Gparted** de GNOME, que internamente utiliza comandos de parted y que viene preinstalado en muchos entornos gráficos Linux, entre ellos los de Debian, o está disponible en los repositorios de las distribuciones para instalar con apt o yum

En Gparted vamos aplicando todas nuestras configuraciones, pero estos no se escribirán en el disco hasta que seleccionemos



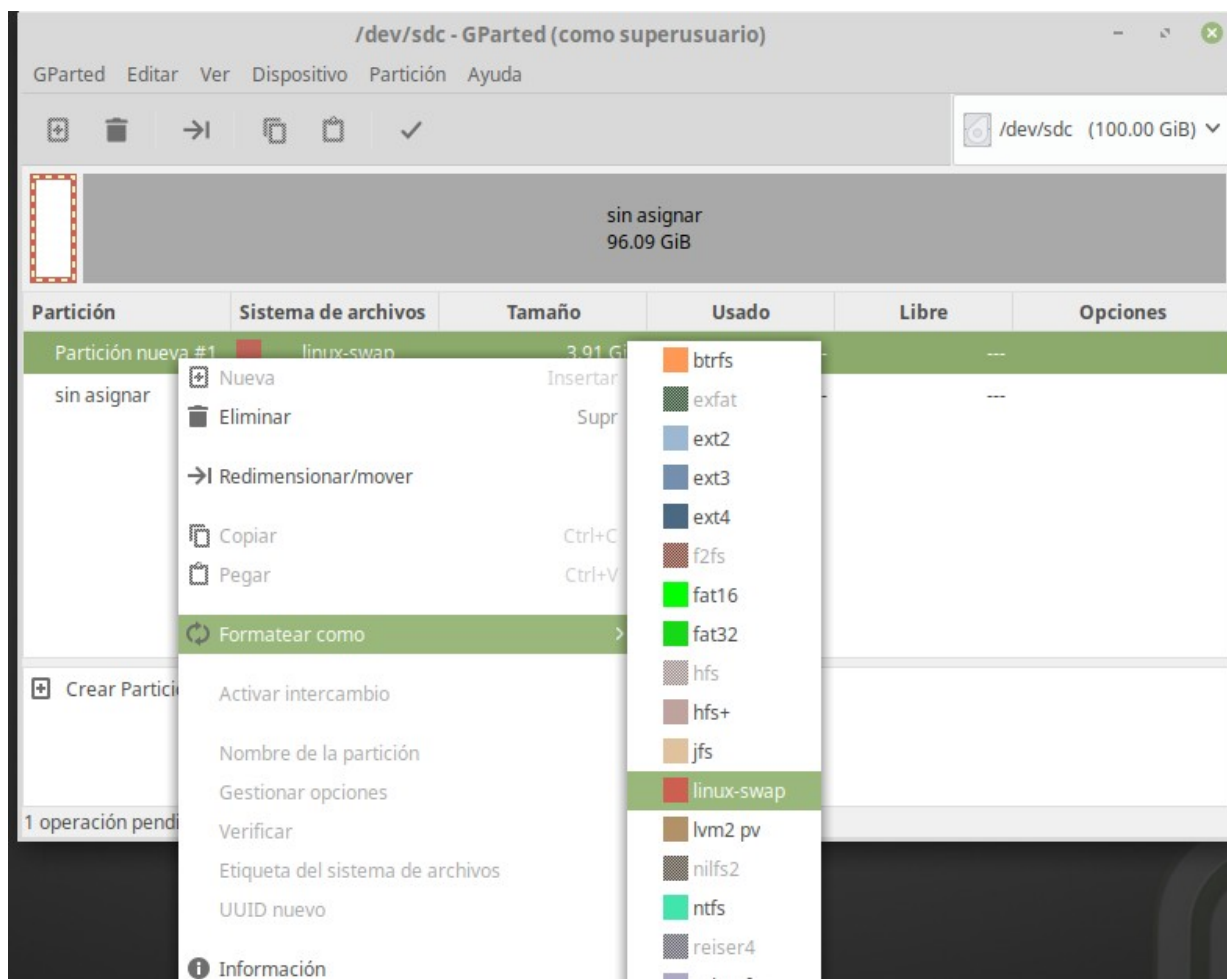
Crear una partición con Gparted

Con Gparted veremos que al crear una partición nos deja seleccionar entre una gran variedad de sistemas de archivos, a diferencia de Windows en el que sólo podíamos crear partición de sus sistemas de archivos propios:



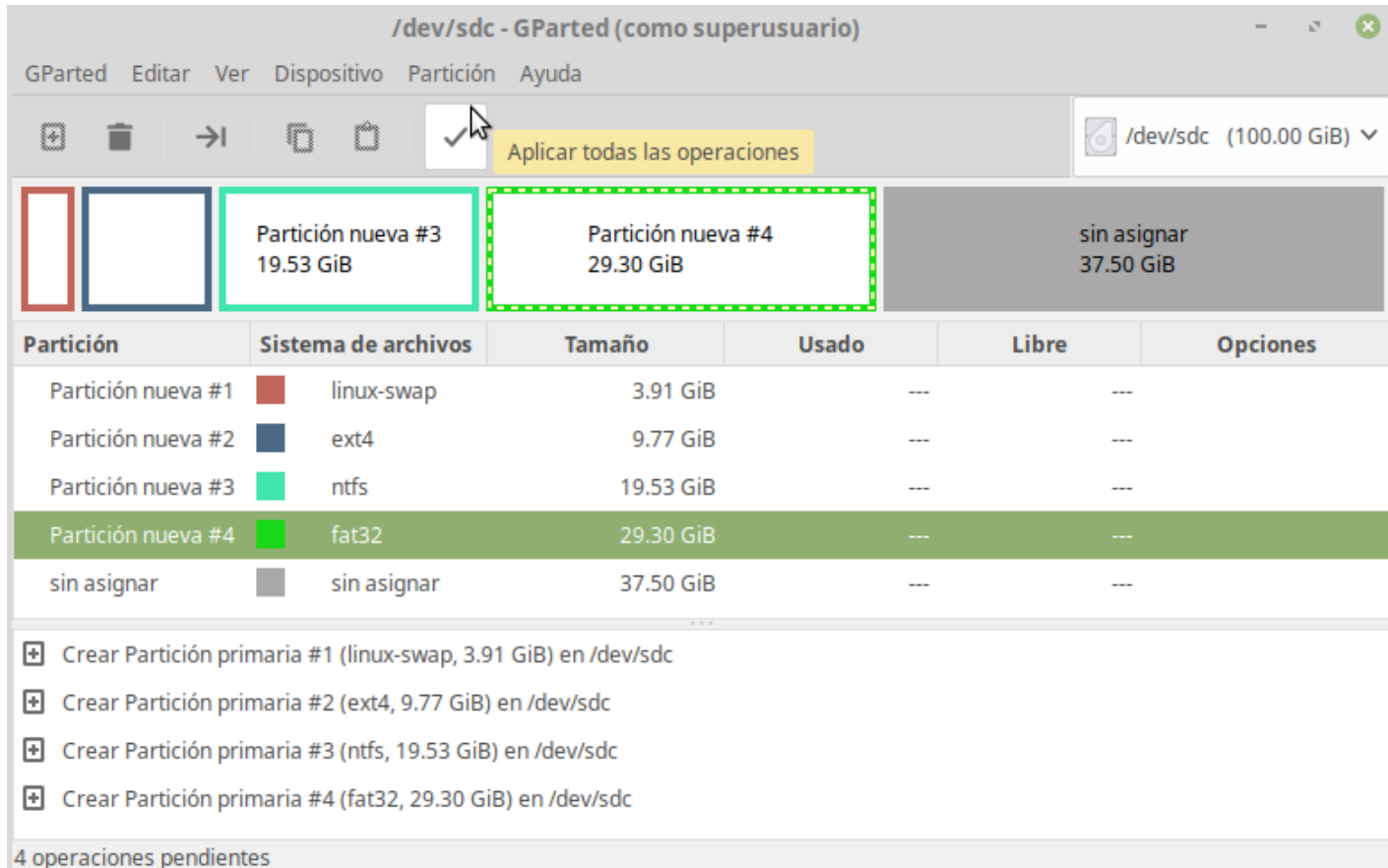
Formatear en GParted

Opcionalmente en Gparted podemos elegir formatear la partición, con lo que así podemos optar por dejar la partición lista para escribir en ella tras el particionado:



Cola de operaciones y trasladar los cambios al disco

Todas las operaciones que hagamos de particionado se van poniendo en un cola que podemos consultar en la parte de abajo. Estos cambios no se trasladarán al disco hasta que seleccionemos la opción “Aplicar todas las operaciones” que tenemos en la barra de herramientas:



The screenshot shows the GParted interface for the device /dev/sdc (100.00 GiB). The top toolbar includes a menu (GParted, Editar, Ver, Dispositivo, Partición, Ayuda) and a set of icons. A yellow button labeled "Aplicar todas las operaciones" is highlighted, with a mouse cursor hovering over it. Below the toolbar, a visual representation of the disk layout shows four partitions: "Partición nueva #1" (red), "Partición nueva #2" (blue), "Partición nueva #3" (green), and "Partición nueva #4" (dashed green). A grey area represents "sin asignar" (37.50 GiB). Below this is a table of partitions:

Partición	Sistema de archivos	Tamaño	Usado	Libre	Opciones
Partición nueva #1	linux-swap	3.91 GiB	---	---	
Partición nueva #2	ext4	9.77 GiB	---	---	
Partición nueva #3	ntfs	19.53 GiB	---	---	
Partición nueva #4	fat32	29.30 GiB	---	---	
sin asignar	sin asignar	37.50 GiB	---	---	

At the bottom, a list of operations is shown:

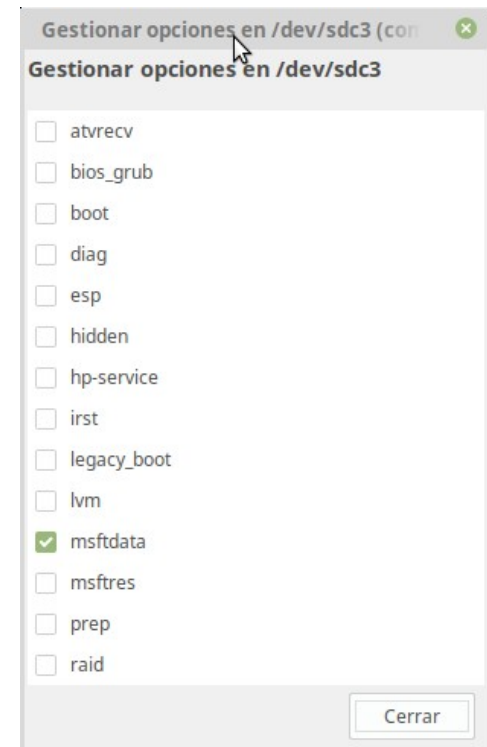
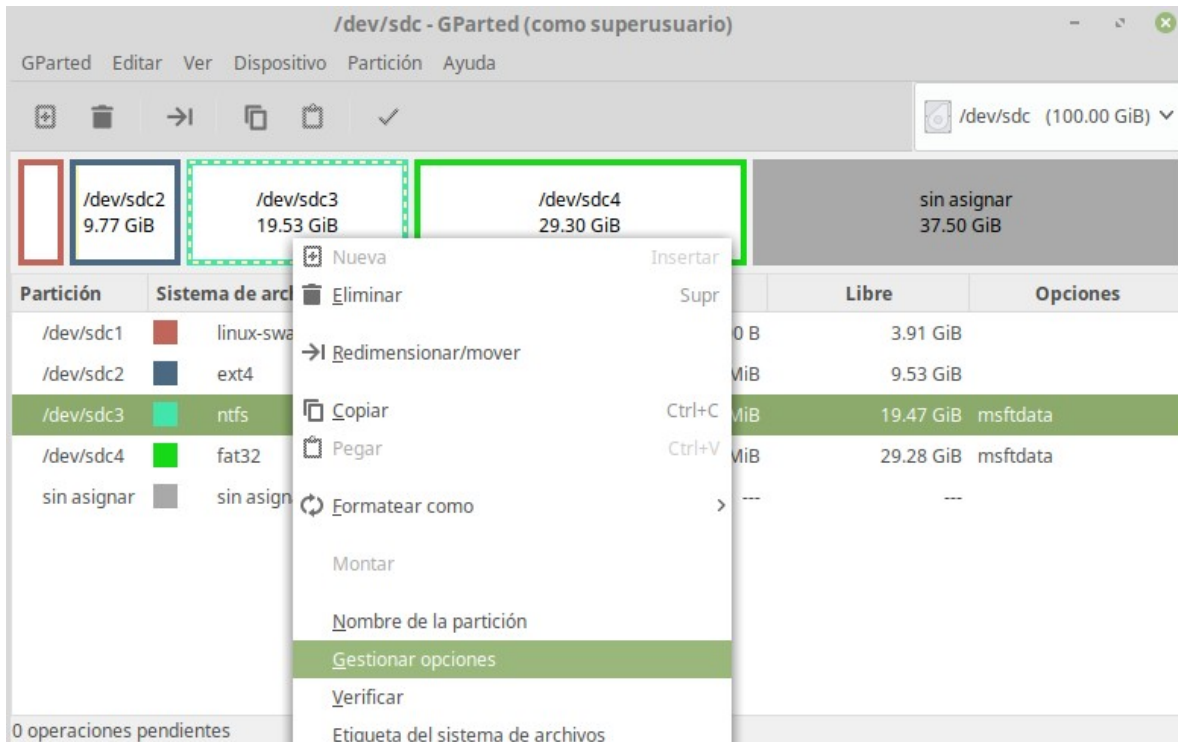
- + Crear Partición primaria #1 (linux-swap, 3.91 GiB) en /dev/sdc
- + Crear Partición primaria #2 (ext4, 9.77 GiB) en /dev/sdc
- + Crear Partición primaria #3 (ntfs, 19.53 GiB) en /dev/sdc
- + Crear Partición primaria #4 (fat32, 29.30 GiB) en /dev/sdc

4 operaciones pendientes

Activar y desactivar flags

En Gparted además es muy fácil activar y desactivar flags asociados a una partición en el menú contextual sobre la partición seleccionando “Gestionar opciones”.

Si ves esta opción deshabilitada es porque la partición ya debé estar en la tabla de particiones del disco, no puede estar en operaciones pendientes:



Montaje de particiones en el fichero fstab

Configurando el fichero fstab tenemos que especificar:

- El **UUID** o nombre de partición, el **punto de montaje** y el **tipo de sistema de archivos**
- **Opciones de montaje.** Si dejamos **defaults** se aplicarán opciones por defecto según el sistema de archivos, pero hay múltiples opciones que nos permite configurar: si queremos o no montaje automático, modo de sólo lectura o lectura/escritura, permitir o bloquear el uso de los bits suid y sgid, limitar los usuarios que pueden montar la partición, ...
- **Opción dump:** Número de veces que se aplicará un backup al sistema de ficheros por el programa dump (0 indica que no se aplica)
- **Opción pass:** Orden en el que se comprobará el sistema de archivos en el arranque. El 1 se reserva para el sistema raíz (/). Si ponemos un 0 no se comprueba en el arranque

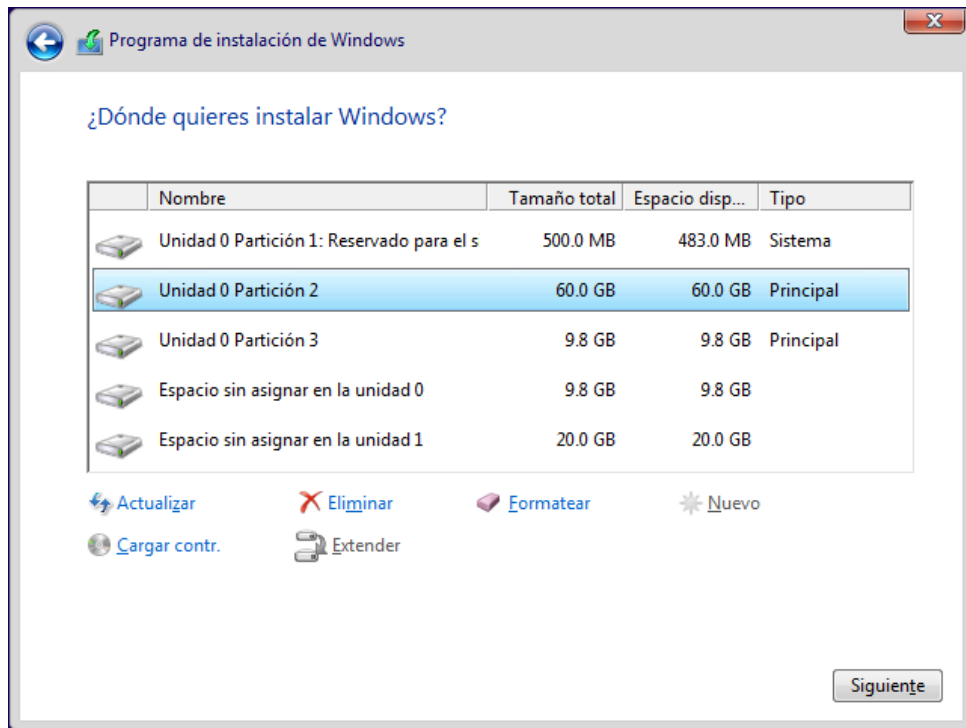
Podemos ver más detalle de las opciones en [https://wiki.archlinux.org/index.php/Fstab_\(Español\)](https://wiki.archlinux.org/index.php/Fstab_(Español))

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
UUID=f6faa53b-e8c3-406c-827e-9c938ad9819d / ext4 errors=remount-ro 0 1
# swap was on /dev/sda5 during installation
UUID=f73bcfeb-7dae-43f6-b9a3-da623df578c0 none swap sw 0 0
/dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0

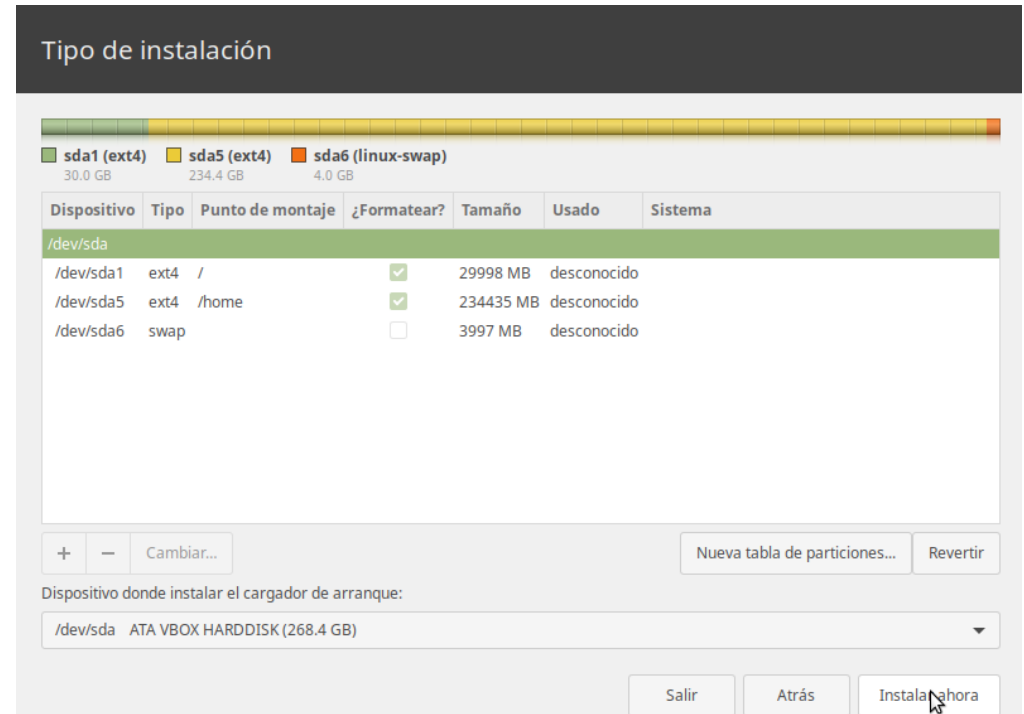
# Particion con montaje en /mnt/ datos1 para montaje con opciones por defecto y que se compruebe en el arranque
UUID=b5ed36c9-5a85-491d-adb3-273b1aebb759 /mnt/datos1 ext4 defaults 0 2
# Particion con montaje en /mnt/datos2 para que NO se monte automáticamente en el arranque (noauto)
/dev/sdb2 /mnt/datos2 ext4 defaults,noauto 0 0
# Particion con montaje en /mnt/datos3 de solo lectura (ro) y que no lo haga automáticamente (noauto)
UUID=BEBA-D173 /mnt/datos3 vfat defaults,ro,noauto 0 0
```

Particionado en la instalación del sistema operativo

Los programas de instalación de Windows y Linux incorporan un paso donde podemos seleccionar el disco y la partición en los que realizar la instalación y opcionalmente ajustar el particionado para lo cual proporcionan una herramienta de particionado integrada:



Herramienta de particionado del instalador de Windows 10



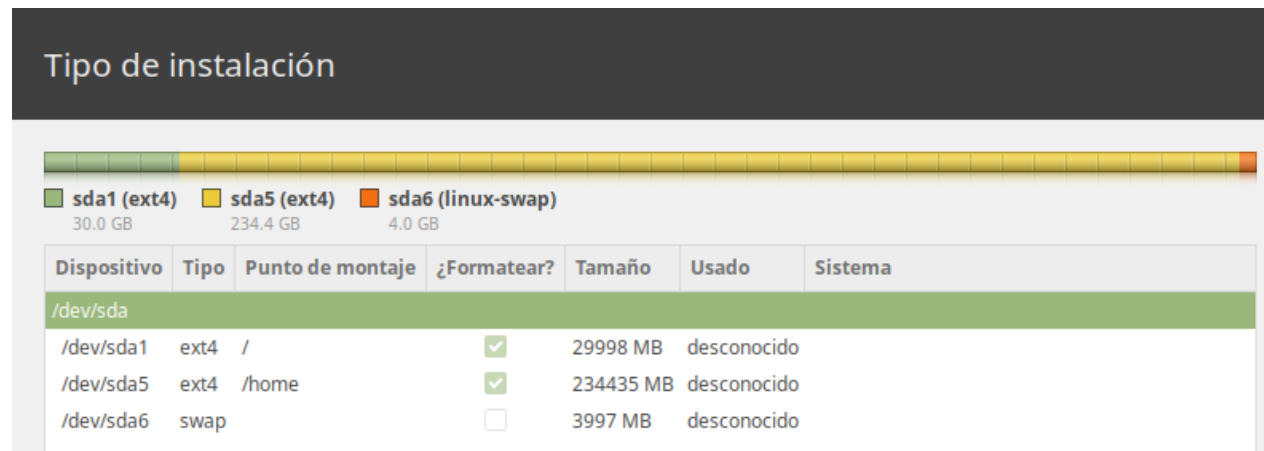
Herramienta de particionado del instalador de Linux Mint

Linux y puntos de montaje

En la instalación de las distribuciones Linux por lo general podemos optar por:

- Un particionado automático, que cogerá todo el disco haciendo la distribución de particiones que considere oportuna. Esta opción no nos interesa si tenemos particiones en el disco que queremos conservar
- Un particionado manual, en el que podremos crear la distribución de particiones que queramos. En este modo tendremos que indicar para cada partición el **punto de montaje**, que será el directorio en el que accederemos a la partición asociada.

La particiones y sus puntos de montaje configuradas en el programa de instalación se trasladarán al fichero `fstab` para que se monten automáticamente durante el arranque del sistema operativo.



Dispositivo	Tipo	Punto de montaje	¿Formatear?	Tamaño	Usado	Sistema
/dev/sda						
/dev/sda1	ext4	/	<input checked="" type="checkbox"/>	29998 MB	desconocido	
/dev/sda5	ext4	/home	<input checked="" type="checkbox"/>	234435 MB	desconocido	
/dev/sda6	swap		<input type="checkbox"/>	3997 MB	desconocido	

En este caso tenemos tres particiones:

- Una partición de 30 GB que se montá en la ruta raíz del sistema de ficheros /. Siempre estaremos obligados a montar una partición en el raíz.
- Una partición de 234,4 GB que se montará en la ruta /home. No es obligatorio montar una partición en /home, pero es recomendable por ser la ruta en la que se guardan los datos de los usuarios
- Una partición swap de 4 GB