

FENT LINUX

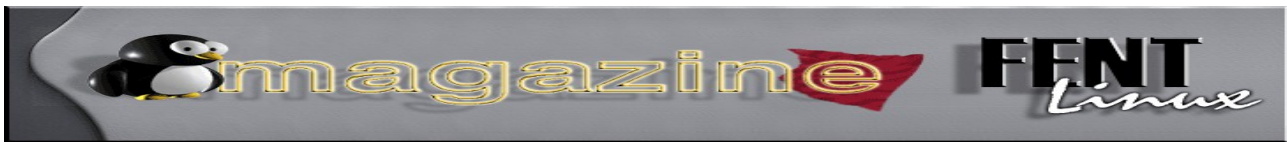


magazine

*Una despedida:
punto y aparte...*

**Si hay que particionar,
mejor hazlo con
(gnu)Linux**

qué es, tipos, clases, qparted, gparted



Índice de contenido

1. Geometría de un disco duro

2. ¿Qué es, en este mundo, una partición?

División lógica

Formato lógico de un sistema operativo

Sistemas de archivos

Dar formato a un volumen o partición

Swap

3. Tipos de particiones

4. Nombres de las particiones

5. Puntos de montaje de las particiones

6. Decisiones a tomar sobre particiones para una instalación de GNU/Linux

7. Programas para particionar

En consola

Fdisk

Cfdisk



Particionadores gráficos

Qtparted

Gparted

Casos prácticos: Qtparted y Gparted

Antes de particionar

8. Algunas sugerencias para distintos esquemas de particionado

9. Comandos relacionados con las particiones y su formateo

mke2fs y mkswap

mkdir

dumpe2fs y fsck

e2fsck

tune2fs

10. El archivo /etc/fstab

11. Formateado de una memoria flash desde Linux

12. Concluyendo

13. Agradecimientos y créditos. Punto y final.

"Todo empezó un abril del 2005 para conmemorar nuestro primero aniversario, hoy el equipo que forma [FL](#) lanza su última entrega, 100% dedicado al fascinante mundo de las particiones desde [gnu/linux](#)."



PARTICIONES: palabra que amedrenta, barrunta desastres y propicia taquicardias. Aunque existen algunas maneras de usar una distribución Linux sin tener que reparticionar el disco, siendo las más populares los live-cds, si queremos una instalación de GNU/Linux, digamos, "en serio", no habrá más remedio que enfrentarse a ellas, de modo que vamos a realizar el intento de compartir nuestros modestos conocimientos, explicando lo mejor que sepamos lo que hemos aprendido.

Trataremos el asunto desde nuestras propias experiencias, así que quede bien entendido que allá cada cual con lo que decida hacer, a nosotros no nos pidáis cuentas si se os tuesta el disco duro o se os enfría la pizza.

1. Geometría de un disco duro.



Un disco duro no es en verdad un único disco, sino varios metidos en una caja hermética. Dispone de varios brazos que se encargan de moverse hacia la zona donde están los datos requeridos, los cuales son leídos por unas cabezas lectoras que están al final de cada brazo. Suele haber un brazo con cabeza lectora por cada cara de cada disco, por lo tanto 2 por disco.

Vamos a ver las diferentes partes de cada disco:

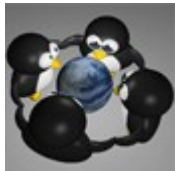
Pista: Cada disco tiene unos surcos llamados pistas, las cuales son concéntricas, no espirales como las de un disco de vinilo. Van numeradas desde el cero (0) la más externa hasta el centro.

Cilindro: Es el conjunto de pistas de todos los discos. Por ejemplo, si hay 6 discos y elegimos la pista 20, el conjunto de todas las pistas 20 de los 6 discos nos da el cilindro. Esto se hace para ahorrar movimientos al brazo

lector.

Sector: Cada pista está dividida en porciones y resulta la unidad básica de almacenamiento de datos, normalmente de 512 bytes. Es la controladora del disco duro la que fija el tamaño al formatear, aunque algunos discos permiten especificar otra cifra.

Cluster: Es un grupo de sectores, es decir, la porción que decía antes.



2. ¿Qué es, en este mundo, una partición?

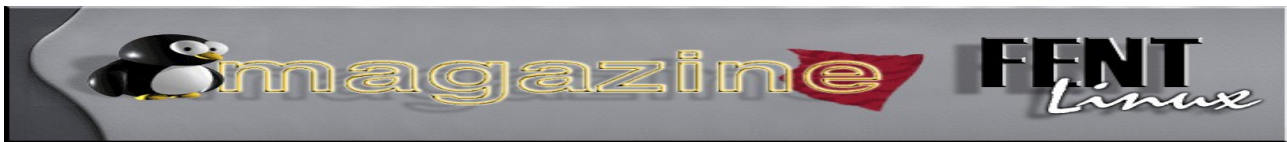
Nos vamos a la [Wikipedia](#), de la cual extraemos lo siguiente:

“En el mundo de la ingeniería de la computación, el particionamiento de discos duros es la creación de divisiones lógicas en un disco duro que permite aplicar el formato lógico de un sistema operativo específico”.

“A un volumen se le da formato mediante un sistema de archivos, como [FAT](#) o [ReiserFS](#), y se le asigna una letra de unidad (en Windows) o un nombre como `hda1` en Linux. En los sistemas Windows puede ver el contenido de un volumen haciendo clic en su icono en el Explorador de Windows o en Mi PC. En Linux y [Unix-like](#) se usa el comando `'df'` para ver la lista de particiones. Un único disco duro puede tener varios volúmenes, que también pueden incluir varios discos.”

“Es usual que en los sistemas Linux se creen hasta 3 particiones: la principal representado por el símbolo `/`, una segunda para el directorio home que contiene las configuraciones de los usuarios y una tercera llamada swap para la memoria virtual temporal.”

Ahora a ver si acertamos a explicar las dudas que puedan haber quedado tras la lectura anterior.



División lógica (o partición).

Es una porción del disco que no está dividida físicamente. Vamos, que no se trata de desmontar el disco duro y trocearlo con el martillo grande, sino que el programa que usemos para el particionado ya se encarga de poner las cosas en su sitio.

Formato lógico de un sistema operativo.

Esto podemos entenderlo como la forma que tiene el SO de guardar y ordenar las cosas para que podamos utilizarlas. “Las cosas” son, evidentemente, todo tipo de archivos y asimilables que puedan ser usados con el software instalado en un ordenador.

También existe el formato físico, o “de bajo nivel”, pero lo normal es que venga hecho de fábrica y que nunca nos haga falta formatear un disco a bajo nivel. Lo mencionamos únicamente de paso y más bien por si alguien se confunde, porque nada tiene que ver con el propósito que nos ocupa, el cual se basa siempre en el formato lógico de divisiones lógicas o particiones.

Sistemas de archivos.

El modo que un sistema operativo tiene de asignar los sectores físicos del disco a los archivos y de relacionar el nombre de los archivos con la situación en el disco de su contenido, más o menos explicado, y ampliando un poco lo dicho anteriormente sobre el formato lógico. La expresión visual de un sistema de archivos la proporciona el software gestor de archivos, que es el encargado de listar el contenido de los directorios y de realizar las operaciones comunes de crear, copiar, mover, eliminar... etc, archivos.

Aparte de los comandos en consola, se pueden utilizar en GNU/Linux, entre otros, los gestores de archivos Nautilus y Konqueror, pertenecientes a GNOME y KDE respectivamente.

Dar formato a un volumen o partición (o, simplemente, formatear).

Implantar un sistema de archivos en la partición, normalmente despidiéndonos de lo que hubiese allí antes, por lo que ya podemos adelantar

una primera buena práctica: Si se va a particionar y/o formatear algo, hay que estar seguros, muy seguros, completamente seguros, de que no vamos a necesitar en la vida la información que allí esté; o de que hemos hecho un par de copias de seguridad y las hemos guardado en lugares diferentes y a prueba de desastres, si es que tenemos pensado recuperar datos de la unidad a particionar y/o de la partición a formatear.

Swap.

Esto, que suena a tortazo bien dado, no es más que un espacio en el disco duro (una partición, aunque también puede ser un archivo) que actúa como si fuera memoria RAM, pero es bastante más lenta, claro está. También se le denomina memoria virtual, y Linux no es el único sistema operativo o, mejor dicho, núcleo, que hace uso de esta técnica. No vamos a entrar en detalles sobre su funcionamiento, pero podemos decir, a modo de ayuda para formarse una idea, que cuando el sistema necesita más memoria libre de la que tiene disponible, guarda unos cuantos datos en el espacio swap del disco y utiliza el que ocupaban en la RAM, volviendo a recuperar los datos guardados cuando los necesite, aún a costa de sustituirlos por otros. Si tuviésemos realmente muy poca memoria RAM la lentitud del sistema puede llegar a ser exasperante, o incluso algo más.

3. Tipos de particiones.



Hay dos tipos de particiones, denominados primarias y extendidas. Las particiones extendidas se inventaron debido a la limitación de cuatro por disco que presentan las particiones primarias. Sólo es posible crear una partición extendida por disco, pero a partir de ella o “dentro”, para entendernos, se pueden crear, dicen, que hasta infinitas particiones lógicas (sí, suena redundante), también llamadas unidades lógicas. Cualquiera puede deducir, incluso sin usar la calculadora, que el máximo de particiones posibles en un disco es de cuatro primarias, o de tres primarias más una extendida, la cual contendrá el resto de particiones que nos hagan falta.



4. Nombres de las particiones.

En un sistema Linux, las distintas particiones posibles tienen nombres de archivos, y se encuentran en el directorio /dev, aunque, según parece, puede que esto cambie en un futuro no demasiado lejano. Pero, de momento, se llaman así:

hdXY

Donde:

hd: Disco.

X: Una letra, empezando por la a, que denomina al disco así: hda = primer disco.

hdb = segundo... etc.

Y: Un número, que será el de la partición en ese disco.

Así:

/dev/hda1 = primer disco, primera partición.

/dev/hdb3 = segundo disco, tercera partición.

/dev/hdc y/o /dev/hdd = Suelen ser las unidades de CD o DVD

Cuando tenemos una partición extendida, su nombre se omite. Por ejemplo, supongamos un primer disco con una partición primaria, otra extendida, y dentro de la extendida dos más, veríamos algo así al utilizar el comando df:

/dev/hda1 = primera partición primaria.

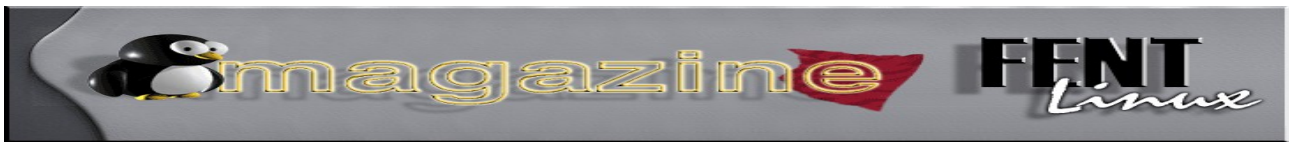
/dev/hda3 = primera unidad lógica dentro de la partición extendida.

/dev/hda4 = segunda unidad lógica dentro de la partición extendida.

/dev/hda2, que no aparece, corresponde a la partición extendida.

En un disco SCSI, o quizás también SATA, las particiones se llamarían **sdXY**, siendo válido el resto de lo expuesto para los discos llamados hd.

El cargador de arranque GRUB llama a los discos y particiones de forma



distinta, y no está de más que veamos cómo, ya que su uso se ha extendido prácticamente a todas las distribuciones. Según GRUB, una partición se denomina con el siguiente esquema:

(hdX,Y)

El nombre completo va entre paréntesis, donde:

hd: Disco.

X: Un número, empezando por el 0, que denomina al disco así: hd0 = primer disco, hd1 = segundo... etc.

,: Una coma que separa el disco de la partición.

Y: Un número, empezando por el 0, que será el de la partición de ese disco.

Así:

(hd0,0) = primer disco, primera partición.

(hd1,2) = segundo disco, tercera partición.

A partir de aquí trataremos a todas las particiones por igual, independientemente de que sean primarias o unidades lógicas dentro de la extendida, por lo que hacemos hincapié en recordar el máximo número de particiones posibles: 4 primarias, ó 3 primarias y 1 extendida.

Y como una imagen vale más que mil palabras ahí tenéis una captura:

Number	Partition	Type	Status	Size	Used space	Start	End
01	/dev/hda1	ntfs	Active	4.89GB	0.00MB	0.03MB	4.89GB
02	/dev/hda2	ext3		4.89GB	0.00MB	4.89GB	9.77GB
03	/dev/hda3	linux-swap		517.72MB	0.00MB	9.77GB	10.28GB
04	/dev/hda4	extended		27.88GB	N/A	10.28GB	38.16GB
05	/dev/hda5	ext3		4.89GB	0.00MB	10.28GB	15.17GB
06	/dev/hda6	fat32		5.75GB	N/A	15.17GB	20.91GB
07	/dev/hda7	fat32		5.75GB	4.63GB	20.91GB	26.66GB
08	/dev/hda8	fat32		5.75GB	1.86GB	26.66GB	32.40GB
09	/dev/hda9	fat32		5.76GB	5.65GB	32.40GB	38.16GB

En esta captura se pueden ver 3 particiones primarias que son hda1 en ntfs para windows, hda2 bajo ext3 para Linux y hda3 para swap (en el centro a ser posible como dijimos anteriormente).

Le sigue otra partición que es extendida (fíjate en el cuadro azul que la rodea para distinguirla de las primarias). Esta partición contendrá las particiones lógicas y que como hemos comentado antes no tiene nombre, aunque a efectos prácticos todos sabemos que se llama hda4. En la captura se puede apreciar una, llamada hda5, bajo ext3 para linux y el resto son formatos fat32 para windows.



5. Puntos de montaje de las particiones.

Son los directorios cuyos datos están contenidos en cada partición. “Montar una partición” significa que el sistema hace accesibles los datos que en ella se encuentran, normalmente a través de la jerarquía de directorios.

Por ejemplo, creamos 4 particiones que distribuimos así: hda1 para raíz /, hda2 para /usr, hda3 para swap y hda4 para /home.

Aunque todo el sistema de ficheros cuelgue del raíz /, los datos que haya en /usr, en /home y, por supuesto, en swap, estarán contenidos en diferentes porciones del primer disco (hda). Podríamos, por tanto, formatear hda1 sin perder los datos que haya en /usr y en /home, y lo mismo para los demás directorios montados en distintas particiones.

Otra recomendación importante es que nunca se debe reparticionar o formatear la partición activa, es decir, aquella desde donde hemos cargado el sistema que tengamos en ejecución.

Lo que sigue ha sido extraído del manual de instalación de *Debian*:

“La partición raíz / siempre debe contener físicamente las particiones /etc, /bin, /sbin, /lib y /dev, si no el sistema no podrá arrancar.”

Lo que, según nuestra libre interpretación, significa en la práctica que los

directorios /etc, /bin, /sbin, /lib y /dev deben estar en la partición montada en el directorio raíz /, luego no podrán montarse en particiones separadas. Todos los demás directorios pueden ser punto de montaje de una partición.

Es posible montar una partición en un directorio que ya contiene datos. En este caso, los datos que quedarán accesibles serán los de la partición montada, y los que había en el directorio quedarán ocultos, o inaccesibles, hasta que se desmonte la partición.



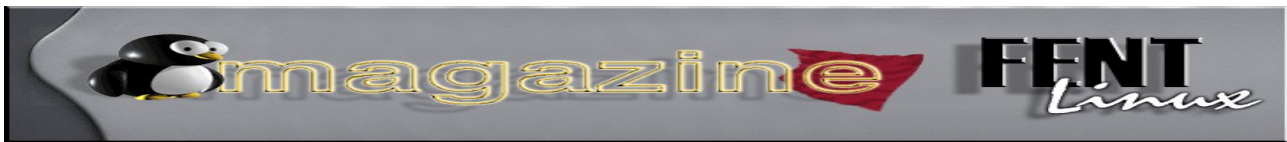
6. Decisiones a tomar sobre particiones para una instalación de GNU/Linux.

Deberemos conocer los siguientes datos y tener una idea lo más clara posible del uso que le vamos a dar a cada sistema operativo que pensemos instalar.

- Espacio del disco que vamos a particionar.
- Cantidad de memoria RAM de que disponemos.
- Espacio a dedicar a la partición de intercambio.
- Número de sistemas operativos a instalar.

Cuando las máquinas tenían menos de 256 Mb de RAM, se recomendaba que el tamaño de la partición de intercambio (swap) fuese el doble de la RAM disponible, pero a partir de ahí, una partición swap de 256 Mb puede cumplir perfectamente su cometido, aunque para utilizar más de 2 gigas de swap sería preferible crear varias particiones de intercambio.

Una práctica que algunos seguimos es aplicar la regla del doble de la RAM hasta un máximo de 512 Mb swap y, a partir de ahí, no variar el tamaño de la swap de 512 Mb por mucha RAM que se tenga. Insistimos en que lo del doble de swap que de RAM es una regla que viene de muy atrás, cuando los ordenadores poseían poca RAM y era usual ver equipos 386 con 8 Mb de RAM. Actualmente no hay razón para que un ordenador con 1 Gb de RAM deba llevar 2 Gb de SWAP, de echo, experiencias y tests personales arrojan que para un



equipo con 1 Gb de RAM es suficiente con 256 Mb o 512 Mb, no usando más de un 2% de la SWAP.

El rendimiento de la partición swap será mejor si está situada en la zona central del disco. Es posible, pero no recomendable, instalar Linux sin partición swap.

También habrá que decidir entre cuántas particiones vamos a repartir la distribución de Linux que hayamos elegido, y cuanto espacio vamos a darle a cada partición, pero más adelante daremos algunas sugerencias sobre estas cuestiones.



7. Programas para particionar

En consola:

Los programas para manejar particiones son varios y van desde los conocidos programas gráficos como el de la anterior imagen, como son `qtparted` o `gparted`, y los ya más imponentes programas de consola, como `fdisk` o `cfdisk`.

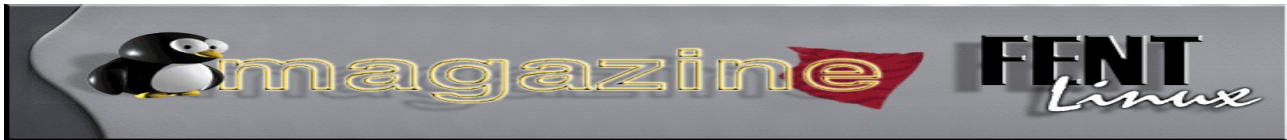
Vamos a tratar de repasar un poco su funcionamiento, empezando por la base: ¡¡¡la consola!!!

Fdisk

`Fdisk` suele ser el programa más utilizado a la hora de tratar particiones, tal vez porque existe una versión en MS-DOS y OS/2, o porque es más antiguo que otros programas.

Antes de empezar a trabajar con particiones debemos tener claros los conceptos vistos anteriormente para que no haya ningún tipo de confusión y, por supuesto, hacerlo como root o no podremos. ;-)

Desde un live-CD (o CD directo, traducción que desde aquí nos atrevemos a proponer), o bien desde un CD instalador, cargamos Linux y abrimos una consola, nos logueamos como root, y ejecutamos:



fdisk /dev/hda (dev/sda si tenemos discos sata) y aparece la información del disco y una línea a la espera de que le pasemos opciones.

En la mayoría de los casos nos aparecerá un mensaje de aviso como en el ejemplo. No hay problema, ya que ese aviso es para los casos indicados, normalmente para discos más antiguos, así que continuaremos con las explicaciones.

El número de cilindros para este disco está establecido en 77557. No hay nada malo en ello, pero es mayor que 1024, y en algunos casos podría causar problemas con:

- 1) software que funciona en el inicio (p.ej. versiones antiguas de LILO)
- 2) software de arranque o particionamiento de otros sistemas operativos (p.ej. FDISK de DOS, FDISK de OS/2)

Orden (m para obtener ayuda):

Si ahora tecleamos m nos saldrán las opciones, de las cuales vamos a explicar las más comunes:

a: Conmuta el indicador de iniciable: Esto sirve para establecer la partición activa, que será la de arranque y que normalmente es la primera, donde suele estar instalado Windows cuando tenemos más de un Sistema Operativo.

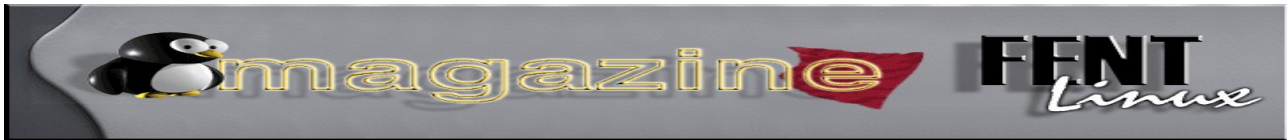
d: Suprime una partición: Nos borra la partición perdiendo los datos que hay en ella.

e: Lista las particiones extendidas: Como vimos antes, no suele salir la partición extendida por defecto, así que esta opción nos la muestra.

f: Corrige el orden de las particiones: A veces sucede que las particiones, aun estando en orden correlativo, no tienen la nomenclatura adecuada, cosa que solventaríamos con esta opción.

n: Añade una nueva partición: Poco que añadir sobre esta opción ;-)

p: Imprime la tabla de particiones: Esta es la opción que se suele usar primero para ver como tenemos mapeado el disco.



q: Sale sin guardar los cambios: Si nos equivocamos en algo podemos salir con esta opción.

r: Vuelve al menú principal desde el menú avanzado: Para los más aventurados que se atrevan con el menú avanzado, podrán volver al menú básico con esta opción.

t: Cambia el identificador de sistema de una partición: Esto es para cambiar el sistema de archivos.

v: Verifica la tabla de particiones: Una vez echas las particiones podremos comprobarlas con esta opción y ver si hay algún tipo de error.

w: Escribe la tabla en el disco y sale: Cuando estemos seguros de los cambios realizados usaremos esta opción.

x: Menú avanzado: Este menú no lo tocaremos, ya que es para gente que sabe y tu aún estás aprendiendo.

¡Manos a la obra! Vamos a particionar un disco duro vacío.

NOTA: Los valores que se dan son los de mi disco. Cada disco tiene una geometría diferente por lo que los valores no serán iguales en otros casos.

Ejecutamos `fdisk` como `root` y usamos la orden `p` para obtener información de la geometría del disco:

Orden (m para obtener ayuda): p

Disco /dev/hdb: 40.0 GB, 40027029504 bytes

16 cabezas, 63 sectores/pista, 77557 cilindros

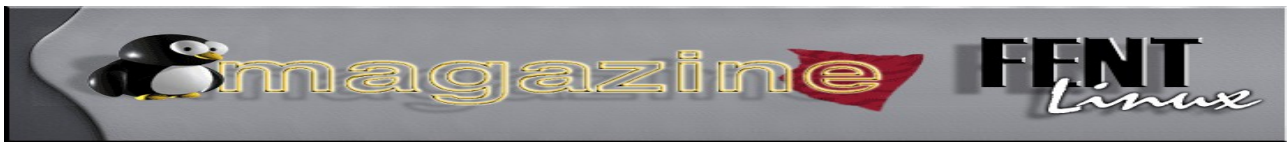
*Unidades = cilindros de 1008 * 512 = 516096 bytes*

Disposit. Inicio Comienzo Fin Bloques Id Sistema

Orden (m para obtener ayuda):

Bueno, pues la información nos dice que tenemos un disco de 40 Gb o lo que es lo mismo 40027029504 bytes. Con la cuenta de la vieja nos sale algo más ajustado: $16 \cdot 63 \cdot 512 \cdot 77557 = 40\ 026\ 857\ 472$.

Vamos a crear la primera partición, por lo que a la pregunta pondremos



n:

Orden (m para obtener ayuda): n

Acción de la orden

e Partición extendida

p Partición primaria (1-4)

p

Número de partición (1-4): 1

Primer cilindro (1-77557, valor predeterminado 1): <INTRO>

Se está utilizando el valor predeterminado 1

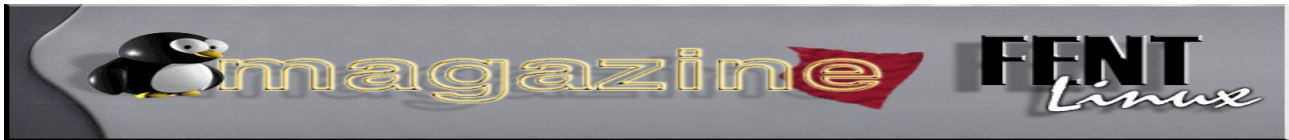
Último cilindro o +tamaño o +tamañoM o +tamañoK (1-77557, valor predeterminado 77557): 48473

Explicamos esto un poco. Al elegir una nueva partición con n nos pregunta si queremos una partición primaria o extendida. Nos interesa hacer una primaria así que elegimos p. Como cada partición se le asigna un número como vimos al principio, y no tenemos ninguna todavía, por lógica queremos la hda1, por lo que a la pregunta de Número de Partición le daremos 1 y pulsamos la tecla INTRO para empezar desde el valor del primer cilindro que nos da fdisk por defecto.

Atención ahora, ya que nos pregunta por el tamaño que tendrá la partición. El tamaño total en este ejemplo es de 77557, así que puedo usar todo el tamaño, pero no me interesa usarlo todo porque quiero hacer más particiones, así que reservo espacio y elijo +48473 que equivale a unos 25 Gb (se puede calcular mediante una regla de tres: 25 Gb por 77557 cilindros entre 40 Gb totales).

Si lo queremos poner en Megas deberíamos elegir +25600M (otra regla de tres), que indica que el valor va en Megas. Elegiría K si lo quisiera elegir en Ks, más recomendable porque se afina más el tamaño.

Bueeeeeeeeno, pues ahora voy a hacer una partición para swap, de la



que solo quiero 128 Mb:

Orden (m para obtener ayuda): n

Acción de la orden

e Partición extendida

p Partición primaria (1-4)

p

Número de partición (1-4): 2

Primer cilindro (29084-77557, valor predeterminado 29084): <INTRO>

Se está utilizando el valor predeterminado 29084

Último cilindro o +tamaño o +tamañoM o +tamañoK (1-77557, valor predeterminado 77557): +128M

Hala, ya está. Como se puede ver, **fdisk** recalcula el espacio que queda libre y como usamos en la anterior partición 29083 cilindros la nueva partición empezará donde acabó la otra, que será el 29084.

Ahora quiero el resto de espacio del disco para otra partición que la voy a hacer extendida para llenarla de particiones lógicas:

Orden (m para obtener ayuda): n

Acción de la orden

e Partición extendida

p Partición primaria (1-4)

e

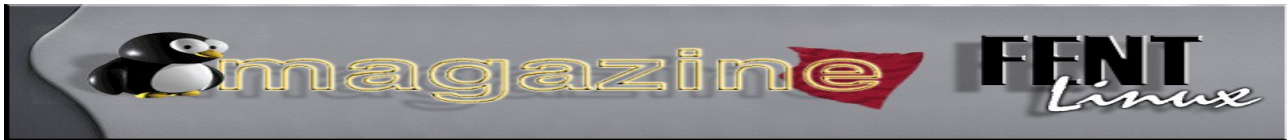
Número de partición (1-4): 3

Primer cilindro (29326-77557, valor predeterminado 29326):<INTRO>

Se está utilizando el valor predeterminado 29326

Último cilindro o +tamaño o +tamañoM o +tamañoK (1-77557, valor predeterminado 77557): 77557

Ahora, sobre esa partición extendida hacemos una lógica con el total de



lo que nos queda de disco:

Orden (m para obtener ayuda): n

Primer cilindro (29326-77557, valor predeterminado 77557): <INTRO>

Se está utilizando el valor predeterminado 77557

Último cilindro o +tamaño o +tamañoM o +tamañoK (1-77557, valor predeterminado 77557): <INTRO>

Ya tenemos el disco particionado. Ahora a la famosa pregunta "Orden (m para obtener ayuda):" ponemos p y nos muestra como ha quedado el disco:

<i>Disposit.</i>	<i>Inicio</i>	<i>Comienzo</i>	<i>Fin Bloques</i>	<i>Id</i>	<i>Sistema</i>
<i>/dev/hda1</i>	<i>1</i>	<i>29083</i>	<i>26214400</i>	<i>83</i>	<i>Linux</i>
<i>/dev/hda2</i>	<i>29084</i>	<i>29325</i>	<i>131072</i>	<i>83</i>	<i>Linux</i>
<i>/dev/hda3</i>	<i>29326</i>	<i>77557</i>	<i>15597568</i>	<i>5</i>	<i>Extended</i>
<i>/dev/hda4</i>	<i>29326</i>	<i>77557</i>	<i>15597568</i>	<i>83</i>	<i>Linux</i>

Bueno, como vemos aún le falta el toque de gracia, que es asignar partición activa y cambiarle el sistema de archivos a las particiones (por defecto todas en ext).

Orden (m para obtener ayuda): a

Número de partición (1-4): 1

Ya tenemos la partición primaria como activa. Vamos a cambiar el sistema de la partición swap, que es la /dev/hda2

fdisk /dev/hda2

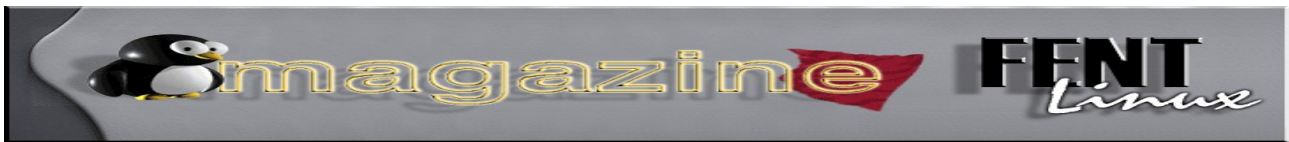
Orden (m para obtener ayuda): t

Número de partición (1-6): 2

Código hexadecimal (escriba L para ver los códigos): 82

Cambiado el tipo de sistema de la partición 2 a 82 (Linux swap)

Como no sabemos qué código hay que usar, pulsaremos L como nos indica el enunciado y nos saldrá una lista de sistemas de archivos. Si volvemos a ejecutar fdisk con la tabla de particiones ahora si que vemos que está como queremos:



Command (m for help): p

Disposit. Inicio Comienzo Fin Bloques Id Sistema

/dev/hda1 * 1 29083 26214400 83 Linux

/dev/hda2 29084 29325 131072 82 Linux swap

/dev/hda3 29326 77557 15597568 5 Extended

/dev/hda3 29326 77557 15597568 83 Linux

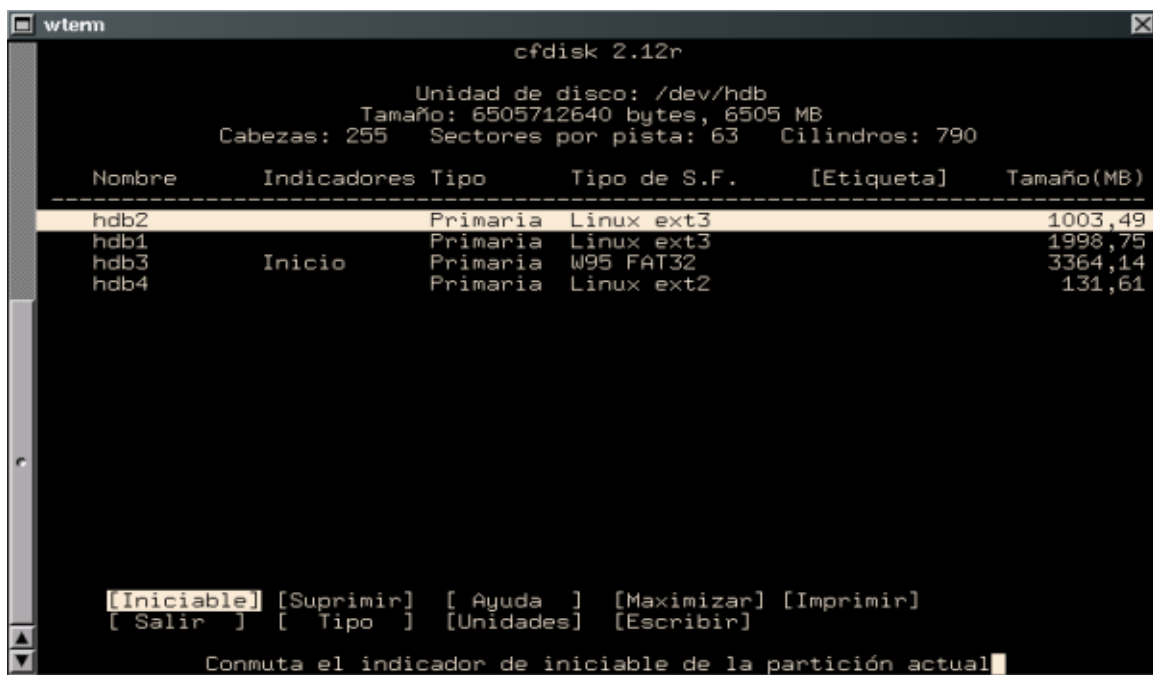
Ahora solo debemos fijar estos valores mediante la orden w y ya tendremos el disco apto para la instalación, no sin antes formatearlas como es debido.

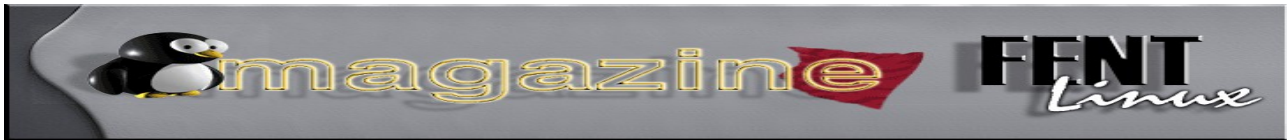
Cfdisk

Cfdisk es también un particionador en modo texto, y funciona a través de menús. Al escribir en un terminal de texto, conectado como root:

```
~# cfdisk /dev/hdb
```

accedo al menú de la siguiente captura de pantalla. Si hubiese tecleado únicamente '**cfdisk**', habría accedido a los datos del disco del sistema activo (/dev/hda), cosa que no es aconsejable como ya dijimos.





Vemos que ese disco está dividido en 4 particiones primarias, que es el máximo permitido, y que tiene 3 sistemas de archivos: ext3 en hbd1 y hdb2, FAT32 (el del Windows anterior al NTFS) en hdb3, y una pequeña con ext2 en hdb4. Y sí, hay un win'98 instalado en la tercera partición del segundo disco, y funciona sin problemas gracias a las facilidades del cargador de arranque GRUB.

En la parte superior aparece la información relativa al tamaño del disco, número de sectores, cabezas y cilindros, en la parte central está la información de las particiones que ya están hechas y, si hubiese espacio libre, aparecería indicando cuanto y en qué orden respecto a las demás particiones.

Este disco no tiene partición swap porque en él no está instalado el sistema Linux, que se encuentra en /dev/hda, lo utilizo para guardar algunas cosas y para que mi hijo juegue con el win. En la parte inferior tenemos las opciones disponibles para modificar las particiones. Utilizando las teclas de movimiento del cursor (las de las flechitas), arriba y abajo nos desplazamos por las particiones de la zona central, e izquierda y derecha por las opciones de la parte inferior.

La partición señalada como "inicio" indica que es arrancable, y sólo puede haber una. Cuando tengo Win junto con Linux, se lo asigno a la del Win, y cuando no, a la partición que contenga el directorio /boot.

Si nos posicionamos en la opción [Ayuda] y pulsamos Enter, accederemos a 2 pantallas en las que se explica de forma rápida, pero clara, el funcionamiento de cfdisk con comandos de tecla en lugar de usando las flechitas. En la siguiente captura vemos la primera de las dos pantallas que aparecen cuando elegimos la opción [Tipo] para cambiar el sistema de archivos de una partición:



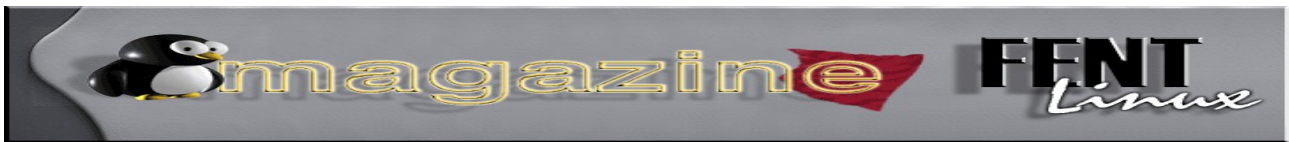
```
wtem
01 FAT12
02 XENIX root
03 XENIX usr
04 FAT16 <32M
05 Extended
06 FAT16
07 HPFS/NTFS
08 AIX
09 AIX bootable
0A OS/2 Boot Manager
0B W95 FAT32
0C W95 FAT32 (LBA)
0E W95 FAT16 (LBA)
0F W95 Ext'd (LBA)
10 OPUS
11 Hidden FAT12
12 Compaq diagnostics
14 Hidden FAT16 <32M
16 Hidden FAT16
17 Hidden HPFS/NTFS
18 AST SmartSleep
1B Hidden W95 FAT32
1C Hidden W95 FAT32 (LB
1E Hidden W95 FAT16 (LB
4F QNX4.x 3rd part
50 OnTrack DM
51 OnTrack DM6 Aux1
52 CP/M
53 OnTrack DM6 Aux3
54 OnTrackDM6
55 EZ-Drive
56 Golden Bow
5C Priam Edisk
61 SpeedStor
63 GNU HURD or SysV
64 Novell Netware 286
65 Novell Netware 386
70 DiskSecure Multi-Boo
75 PC/IX
80 Old Minix
81 Minix / old Linux
82 Linux swap / Solaris
83 Linux
84 OS/2 hidden C: drive
85 Linux extended
86 NTFS volume set
87 NTFS volume set
88 Linux plaintext
A8 Darwin UFS
A9 NetBSD
AB Darwin boot
B7 BSDI fs
B8 BSDI swap
BB Boot Wizard hidden
BE Solaris boot
BF Solaris
C1 DRDOS/sec (FAT-12)
C4 DRDOS/sec (FAT-16 <
C6 DRDOS/sec (FAT-16)
C7 Syrinx
DA Non-FS data
DB CP/M / CTOS / ...
DE Dell Utility
DF BootIt
E1 DOS access
E3 DOS R/O
E4 SpeedStor
EB BeOS fs
EE EFI GPT
EF EFI (FAT-12/16/32)
F0 Linux/PA-RISC boot
F1 SpeedStor
Pulse una tecla para continuar
```

Para elegir el tipo de partición hay que teclear en la segunda pantalla el número hexadecimal que precede a su nombre.

Se han marcado con un rectángulo amarillo las opciones que nos parecen más comunes a la hora de elegir el sistema de archivos para una partición.

Quede bien entendido que elegir un sistema de archivos para una partición no implica su disponibilidad inmediata para el uso, antes de poder usarla es necesario darle formato, o formatearla, que es otra forma de decir lo mismo: “El reconocimiento de una partición no implica su formateo”. Con un sistema GNU/Linux podemos acceder a casi cualquier tipo de sistemas de archivos, pero quede claro que sólo podremos formatear algunas: las propias de Linux, la FAT32 en algunos casos (por ejemplo, para una memoria flash), pero nada de pensar en darle formato a particiones para NTFS; en estos casos podemos crear la partición desde nuestro Linux, pero luego habrá que formatearla con las herramientas propias del SO que vayamos a instalar en ella.

Cuando lo que queremos es crear una nueva partición en un espacio libre, lo resaltamos, posicionándonos sobre él en la parte central de la pantalla,



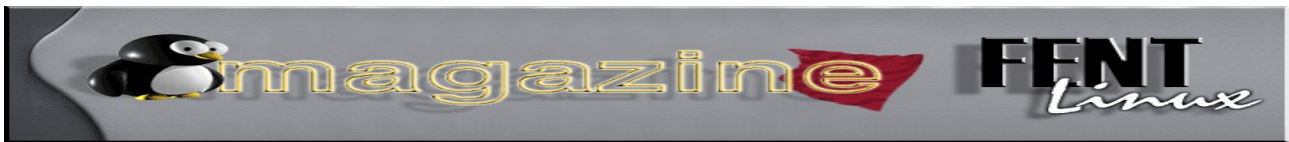
y luego nos situamos en la opción [Nueva] que aparece en la parte inferior. Pulsamos Enter y nos pedirá el tamaño, el cual podremos expresar cómodamente en Megas o Gigas, por ejemplo: “20000M” o “20G”, que viene a ser lo mismo, y también nos puede preguntar si queremos situar la partición a partir del principio del disco o del final.

Cuando hayamos terminado de definir las particiones, seleccionamos [Escribir], pulsamos Enter, confirmamos, y adiós a todo lo que hubiese en las particiones modificadas y hola a nuestra nueva tabla de particiones. Por último, una captura de pantalla de un disco de unos 160 Gigas con un particionado algo más extenso de lo habitual, y en el que podemos observar como hay una única partición primaria y otra extendida, comenzando la primera partición lógica en hda5, la siguiente a hda4 que sería la propia partición extendida, quedando hda2 y hda3 reservadas para particiones primarias.

```
wtem
cfdisk 2.12r
Unidad de disco: /dev/hda
Tamaño: 163928604672 bytes, 163.9 GB
Cabezas: 255   Sectores por pista: 63   Cilindros: 19929
```

Nombre	Indicadores	Tipo	Tipo de S.F.	[Etiqueta]	Tamaño(MB)
hda1	Inicio	Primaria	W95 FAT32 (LBA)		15998,17
hda5		Lógica	Linux ext3	[/]	1480,56
hda6		Lógica	Linux ext3	[/boot]	197,41
hda7		Lógica	Linux ext3	[/home]	7312,28
hda8		Lógica	Linux ext3	[/tmp]	16672,65
hda9		Lógica	Linux ext3	[/usr]	6111,39
hda10		Lógica	Linux ext3	[/usr/local]	3487,52
hda11		Lógica	Linux ext3	[/var]	15776,09
hda12		Lógica	Linux ext3	[/var/log]	1998,75
hda13		Lógica	Linux swap / Solaris		518,20
hda14		Lógica	Linux ext3	[/opt]	39999,54
hda15		Lógica	W95 FAT32 (LBA)		6399,27
hda16		Lógica	Linux ext3		47969,84

```
[Iniciable] [Suprimir] [ Ayuda ] [Maximizar] [Imprimir]
[ Salir ] [ Tipo ] [Unidades] [Escribir]
Commuta el indicador de iniciable de la partición actual
```



En este caso he etiquetado la mayoría de las particiones con el nombre del directorio en el que cada una se monta. La última la tenía “en reserva”, y por eso no la etiqueté al principio, aunque actualmente le estoy dando uso. Podría etiquetar las que faltan con el siguiente comando (siendo root):

```
~# e2label /dev/hda1 "winbugs"  
~# e2label /dev/hda15 "datos_Fat32"  
~# e2label /dev/hda16 "/cage" (por ejemplo).
```

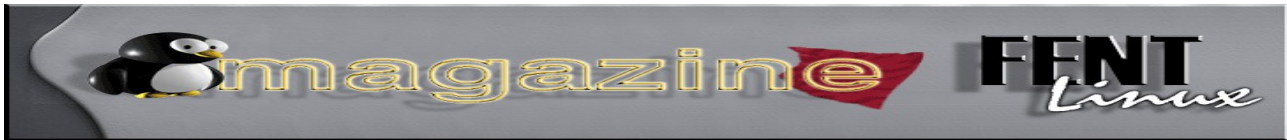
Si ahora quisiera reparticionar el disco completo, me iría posicionando en cada una de las particiones y pulsando Enter sobre la opción [suprimir] hasta que sólo quede una línea en la parte central indicando la capacidad total del disco como espacio libre.

Particionadores gráficos.

Llega el turno de adentrarse en el conocimiento de los particionadores gráficos, que no todo iba a ser consola y consola... con ellos gestionar disco duros y particiones puede realizarse con pequeños golpes de ratón, rápida y fácilmente. Todo lo explicado hasta aquí puede realizarse visualmente sin mayores complicaciones

Los más conocidos son gparted y qtparted, habituales en la práctica totalidad de distribuciones, sobre todo en las de formato live cd/dvd, (o CD/DVD directo) incluso muchos instaladores de las diferentes distribuciones linux las incorporan en el proceso de instalación, sin ir más lejos ubuntu utiliza gparted durante el proceso de particionamiento, para facilitar al máximo todo el proceso y que el usuario tenga el control visual de todo lo que está haciendo. Si no instalarlo es cosa sencilla, ya que contamos con versiones empaquetadas que facilitan el proceso al máximo.

Con ellos es posible: montar y desmontar particiones, formatearlas, borrarlas, redimensionarlas, incluso podemos trastear sin miedo a cargarnos algo; ojo: siempre y cuando no guardemos los cambios, que entonces la vuelta atrás ya está más complicada.



Características similares de ambos:

A simple vista podemos observar una zona más o menos central donde se nos muestra, tanto gráficamente como en forma de tabla, el contenido del disco (particiones existentes, o estructura del disco duro) que hemos seleccionado. Nos proporciona información de cada partición: Sistema de archivos que contiene, tamaño, espacio ocupado por datos y libre.

Contamos con menús superiores o unos bonitos iconos para realizar todas las operaciones necesarias. Gracias a ellos podemos:

- * Crear: basta con hacer click con el botón derecho del ratón sobre el espacio libre del disco.

- * Formatear: Para dejar la partición lista y preparada para ser usada posteriormente por el sistema.

- * Redimensionar: para cambiar el tamaño de una partición. Para muchos es una operación arriesgada que puede darnos algún que otro disgusto en el sistema instalado, si es windows, mejor si realizamos antes una desfragmentación del disco duro que vayamos a redimensionar.

- * Mover: Mueve una partición a otro sitio del disco.

- * Borrar: Elimina una partición y la convierte en espacio libre.

- * Guardar o ejecutar los cambios, se escriben todos los cambios que le hayamos dado. Aquí ya no hay vuelta atrás: ¿estáis seguros?

- * Podemos elegir el sistema de archivos: NTFS; ReiserFS; JFS; Ext2; Ext3 (se suele aconsejar este, pero siempre depende un poco de la distribución que vayamos a utilizar) XFS

- * Seleccionar si deseamos o no etiquetar los discos.

- * Si las particiones son primarias, lógicas extendidas...

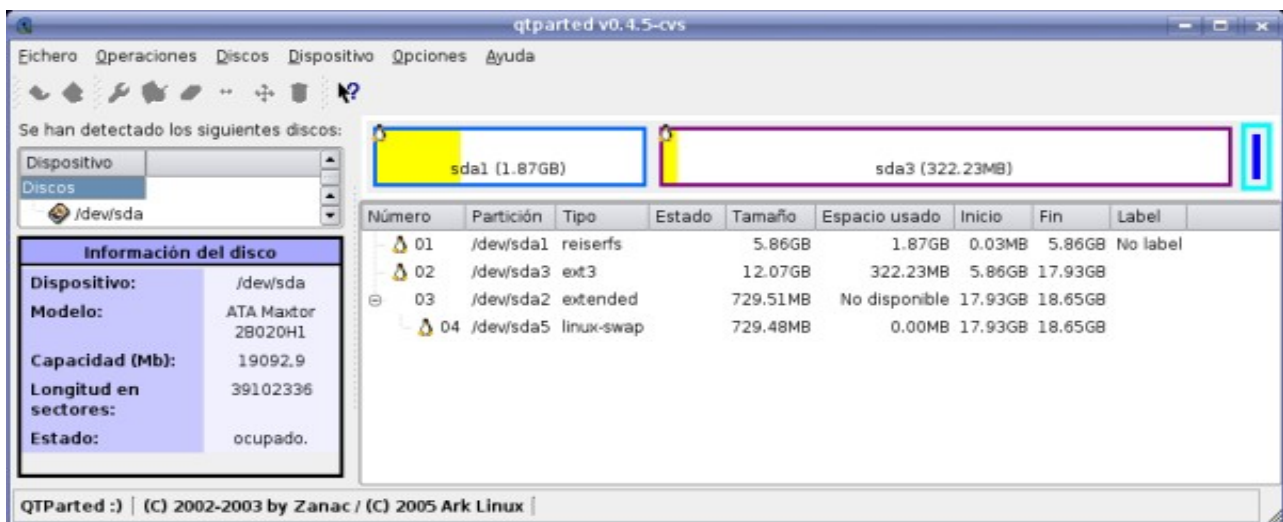
Qtparted

Suele ser el más habitual de *kde*. Contamos con paquetes *rpm* y *deb*, y el código fuente desde <http://qtparted.sourceforge.net/>

Su instalación puede ser tan sencilla como ejecutar el siguiente comando en distribuciones derivadas de *debian*:

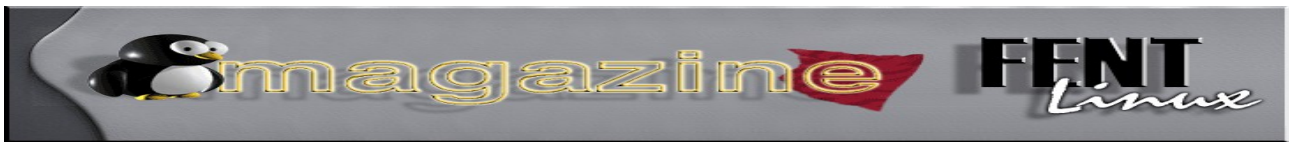
```
# apt-get install qtparted
```

Una imagen del mismo. En él observamos los menús superiores e iconos (ensombrecidos) a los que antes hicimos referencia y dos zonas, la derecha y la izquierda principales de la aplicación, donde en nuestra derecha aparece reflejada la estructura actual del disco duro y la izquierda aparecen reflejados los discos duros existentes y donde se nos facilita información sobre los mismos.



Gparted

GParted es una interfaz gráfica que utiliza *libparted* para detectar dispositivos y particiones, además de utilizar distintas herramientas de sistemas de archivos (*e2fsprogs*, *dosfstools*, *reiserfsprogs*, *ntfsprogs*, etc.) para completar su funcionamiento.



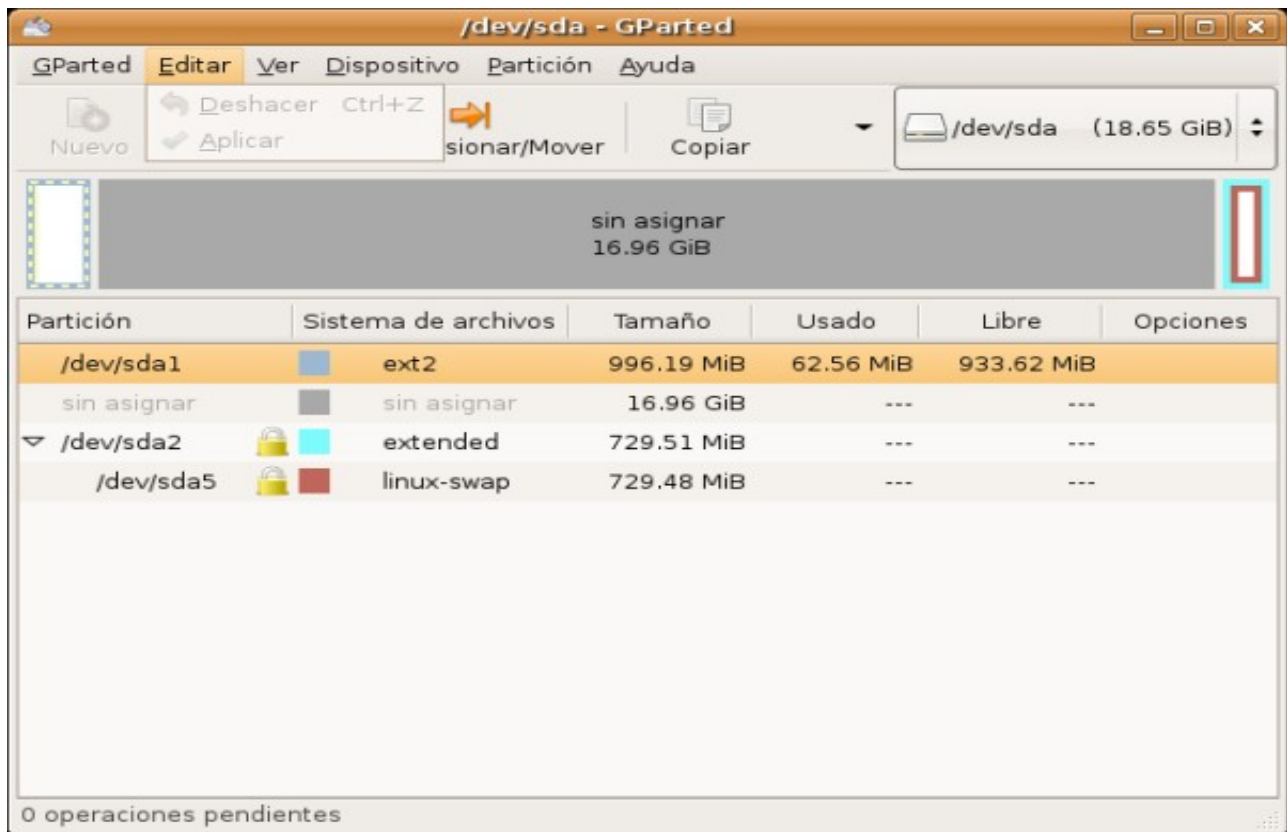
Web del mismo: <http://gparted.sourceforge.net/>

Más detalles: <http://sourceforge.net/projects/gparted/>

Descarga del mismo: http://sourceforge.net/project/showfiles.php?group_id=115843

Contamos con versión independiente, instalable en cualquier linux o con versiones específicas para live cd o live usb.

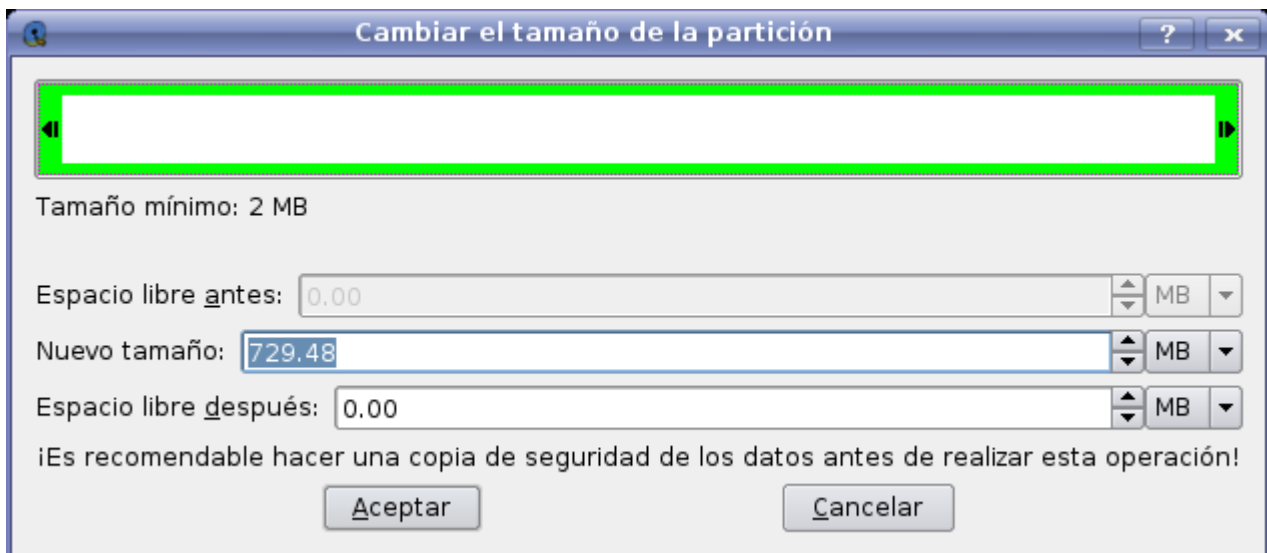
Una imagen del mismo, donde podemos observar los diferentes menús superiores e iconos, así como información sobre el disco duro y que nos posibilita realizar todas las acciones mencionadas con anterioridad. Y la ventana principal donde se refleja la estructura del disco duro antes de proceder a realizar las funciones de particionamiento.



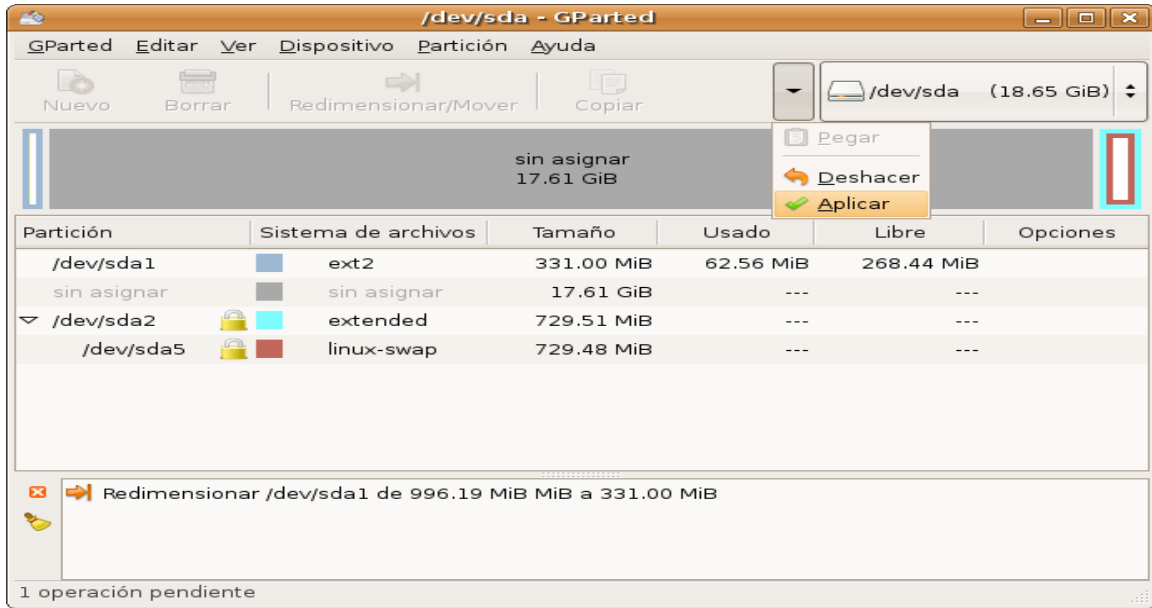
Casos Prácticos: Qtparted y Gparted

Estas herramientas se usan principalmente para modificar discos secundarios o particiones del principal que no formen parte de la instalación de ningún sistema, o también para reparticionar antes de una reinstalación completa en un disco diferente del que tengamos en uso, pero el particionado inicial se suele hacer con el instalador de la distro.

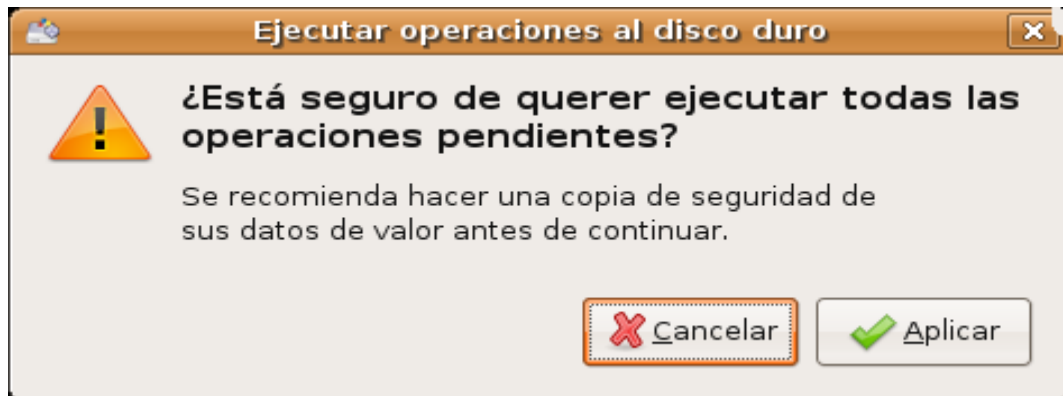
* Redimensionar: Basta con seleccionar el espacio y en la ventana que se nos abre modificar el tamaño ya sea a golpe de ratón con la imagen superior o indicándole a mano (con el teclado) el tamaño de la nueva partición. Ejemplos con **qtparted** para cambiar el tamaño o redimensionar una partición. Basta con arrastrar con el ratón la zona verde e indicarle su nueva ubicación, como se puede observar en la imagen posterior.



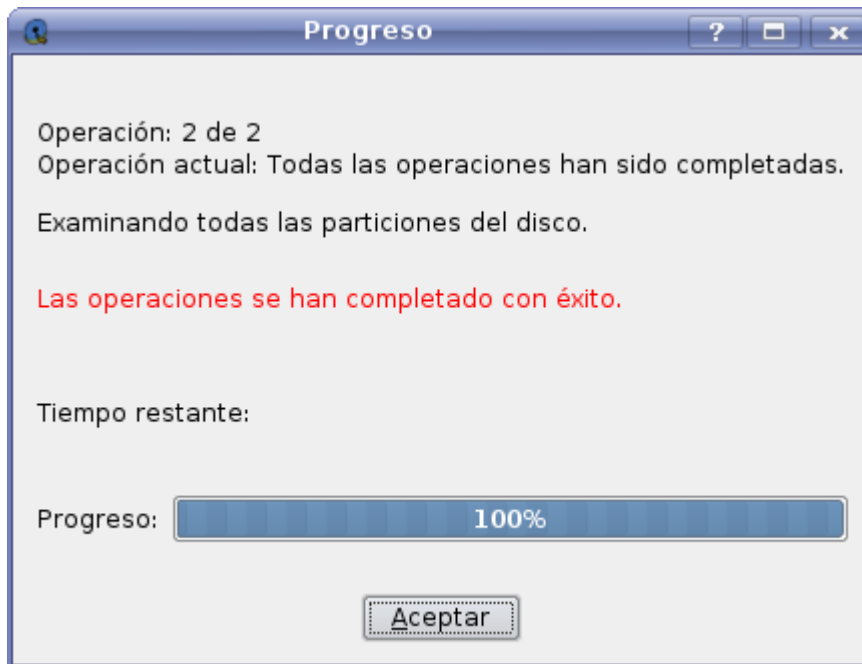
* Es posible volver atrás, siempre y cuando no apliquemos los cambios:



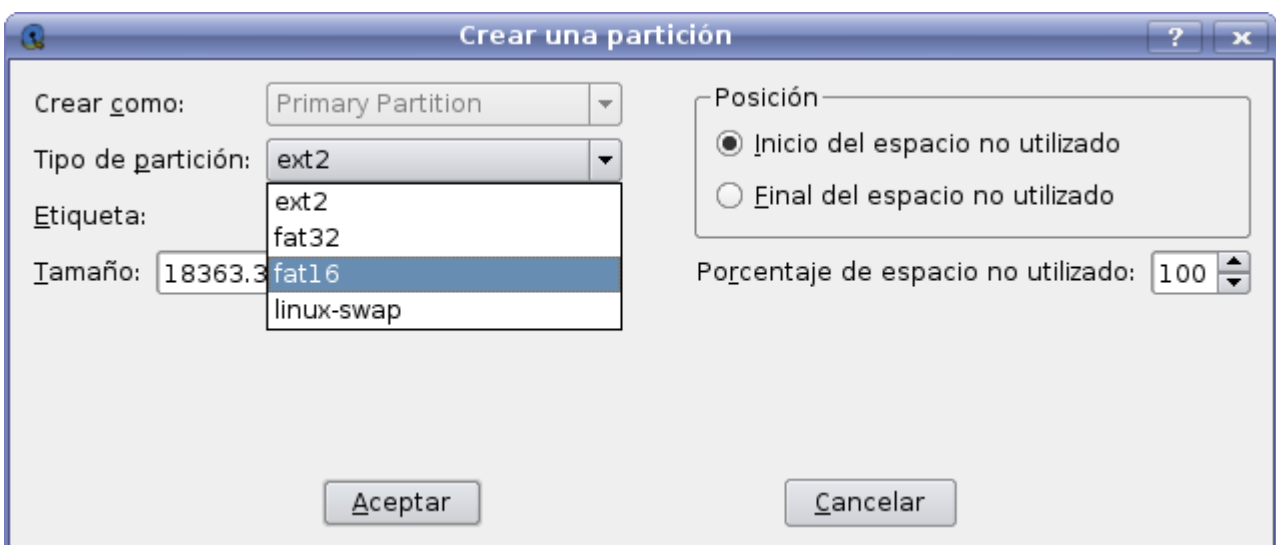
Pero cuando aplicamos los cambios, observamos que nos avisa de que una vez realizado ya no habrá vuelta atrás.



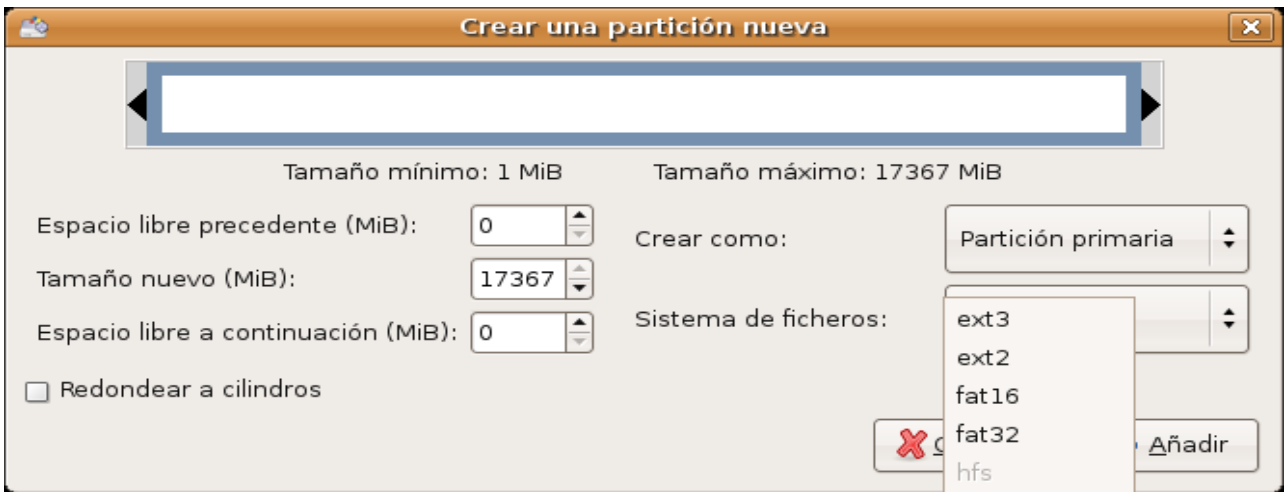
Y si aceptamos en principio todas las operaciones solicitadas acaban con éxito.



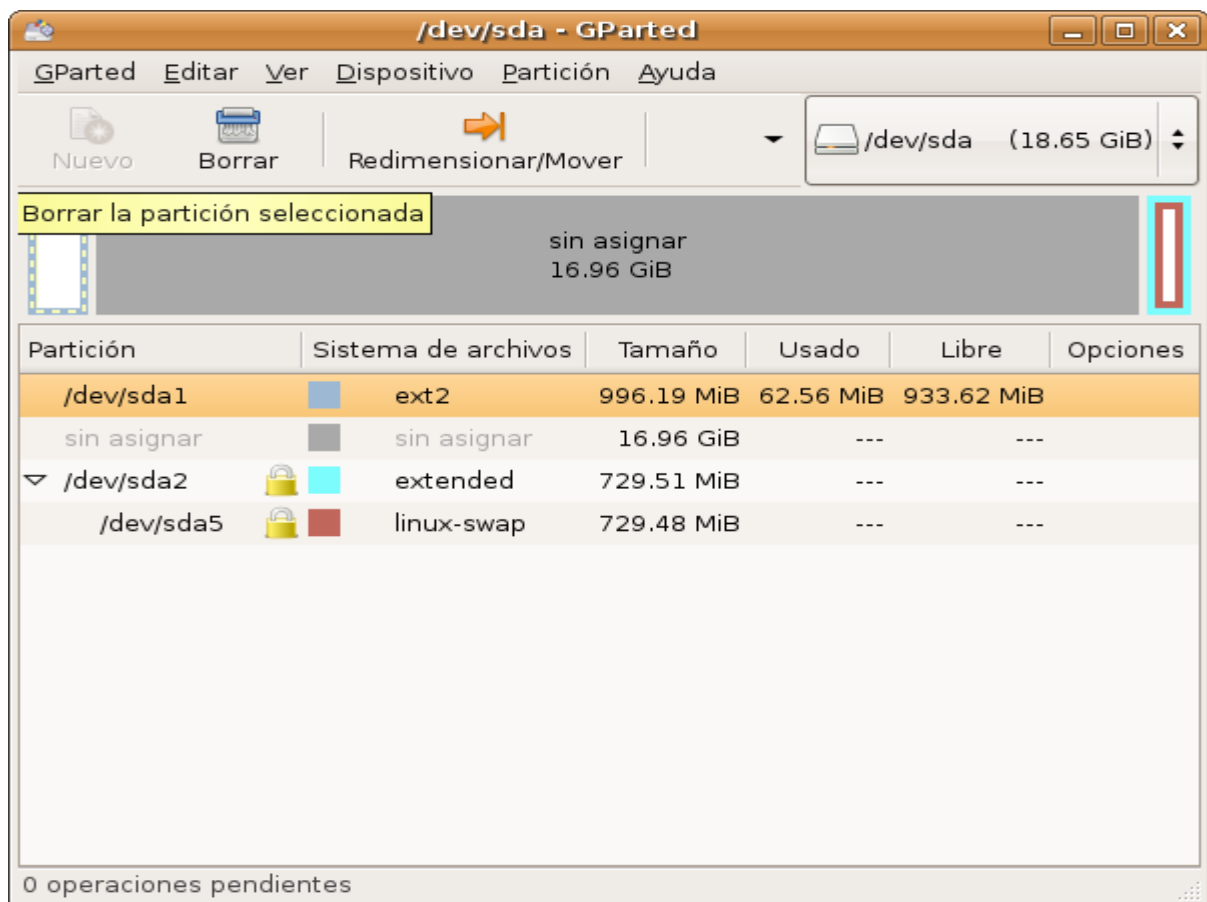
* Crear particiones: Y si lo que pretendemos es crear particiones con **qtparted**, un ejemplo donde podemos indicarle el tipo de partición, tamaño, tipo de archivos, su ubicación...

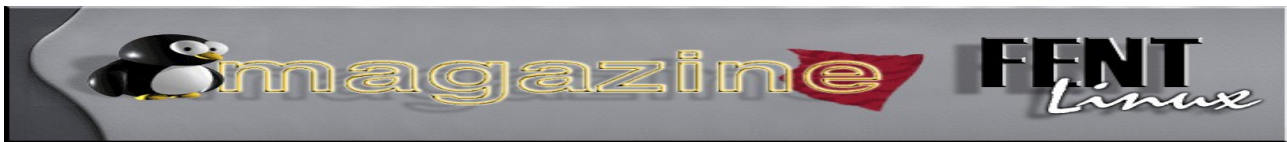


Con Gparted



* También podemos borrar las particiones, para ello basta con seleccionarla y darle al icono de borrar.





Y alguna que otra opción más. Llegados a este punto, recomendamos la presente revisión al siguiente [manual sobre particiones con qtparted](#).

Antes de particionar:

* Hay que tener claro que es lo que pretendemos realizar y por qué. Y si no pregunten, que para algo estamos y no cobramos por ello.

* Tener las copias de seguridad o imágenes de los o del sistema instalado, a punto y actualizadas. Ya saben: mejor prevenir que llorar después.

* Recordar que los cambios son reversible si salimos sin darle la orden de guardar los cambios que la hayamos proporcionado.

En resumen

Conviene tener claro que todo lo aquí explicado no son más que un cúmulo de datos genéricos, que posteriormente el usuario conforme a su realidad debe llevarlos a la práctica, y que no todos deben sufrir las consecuencias del particionamiento, que no es lo mismo un pc con sólo un sistema operativo que con varios, la cosa cambia y mucho, o un disco duro entero para linux, o dos, o tres...

Señalar que los menús habituales de particionamiento durante los procesos de instalación de un sistema, proporcionan tres opciones diferentes, y que estas, como siempre, depende de las características previas del usuario y del equipo donde se va a realizar la instalación, siendo las opciones más habituales:

La de Borrar y usar el disco completo: la más sencilla de realizar, el instalador se encarga de borrar y particionar automáticamente. El usuario sólo mira.

La de utilizar espacio libre más grande: el instalador se encarga de configurar automáticamente las particiones, pero no usa todo el disco, sólo aquel espacio que no este particionado y además sea el de mayor tamaño, en el caso de que hay más de uno.

Y claro, la de Editar todas las particiones o disco duro: la fase más compleja y para “expertos” (entre comillas). En el caso de que escojamos esta opción, todo se realiza a voluntad del usuario, pudiendo escoger o crear, entre otras cosas diversas, particiones para albergar los diferentes archivos y donde seleccionamos el tamaño de la partición y/o particiones así como el sistema de ficheros que deseemos para nuestro sistema.



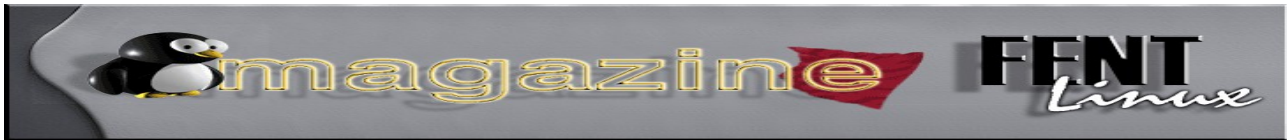
8. Algunas sugerencias para distintos esquemas de particionado.

Dedicando todo el disco para Linux y con una sola distribución a instalar, con al menos 256 Mb RAM y 40 Gigas de disco. Suficiente con 3 particiones para: / (raíz), /home y swap.

Dado que /usr se lleva aproximadamente el 80% del espacio disponible para el / (raíz), podemos aventurar un cálculo del espacio necesario para cada partición basándonos en nuestra experiencia en la instalación de sistemas GNU/Linux con distintos esquemas de particionado.

Previendo que queramos incluir los archivos fuentes, con 15-20 Gigas nos debería sobrar para el /, aunque si sólo vamos a instalar binarios, unos 8-12 Gigas estarían bien; 512 Megas (o 256 si se quiere ser ahorrativo) para swap, y el resto para /home. El orden de particionamiento podría ser: / al principio del disco, luego swap, para que al menos no quede en un extremo, y por último /home.

A la partición dedicada a /home se le deja el mayor espacio posible porque es ahí donde deberán ir a parar todos nuestros archivos, sean del tipo



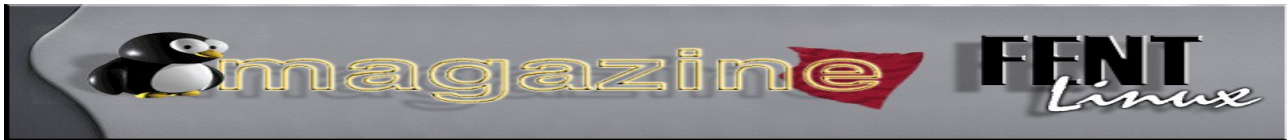
que sean, ya que normalmente no estaremos conectados como root, y ese será el único sitio, aparte de /tmp, donde tendremos permiso de escritura. Si alguna vez tuviésemos que reinstalar el sistema, sólo sería necesario tocar la partición /, conservando todos nuestros datos y configuraciones en /home. Esto es algo que no se suele apreciar hasta que uno lo ve con sus propios ojos: realizar una instalación desde cero, incluso formateando, y que todas nuestras configuraciones, marcadores del navegador... etc, continúen estando como si nada hubiese ocurrido.

En lugar de continuar escribiendo una serie de opciones “prefabricadas” según nuestros gustos y preferencias, tal vez venga mejor dar una idea del espacio que puede requerir cada directorio para albergar sus datos y comentar algunos detalles a tener en cuenta, y que cada cual se organice a su mejor conveniencia.

Si vamos a situar /boot, /home/, /usr, /tmp, /var y quizá también /opt en particiones separadas, con que le demos 200 megas a la partición raíz / ya irá sobrada, e incluso puede que desperdiciemos espacio. En ese disco de 160 Gigas que utilizo en la captura del cfdisk, la partición raíz tiene ocupados exactamente 79 Megas después de dos años de uso ininterrumpido. Si le asigné 1,2 gigas al principio fue pensando en dejar 1 Gb para poner cosas en /root, pero no ha sido necesario, ni creo que lo sea en el futuro.

La partición que montemos en /usr será la que contenga la inmensa mayoría de los programas del sistema. Una Debian al completo, con los fuentes incluidos, no debería llegar a ocupar 20 gigas ni con mucho, y eso tirando bastante por arriba. Para una instalación normal, y siendo generosos, con 10-12 Gigas seguro que sobraré espacio. En el disco de 160 Gb que nos está sirviendo de ejemplo, /usr ocupa 3,1 Gigas, y tiene Gnome, Kde, Window Maker, Xfce4, y no sé si alguno más, aunque están instalados solamente los binarios y no contiene juegos ni florituras varias.

Habrá que tener en cuenta el uso que vayamos a darle a /usr/local, si es que pensamos instalar muchas cosas desde los fuentes o, como es mi caso, utilizar ese directorio para los fuentes del kernel, archivos de programas,



scripts, escritos, y demás “cosas mías”, para lo que he reservado unos 3 Gb y lo he situado en una partición separada. Luego, dándole los permisos adecuados y con algunos enlaces simbólicos en mi \$HOME, accedo a los datos más cómodamente. Si lo vamos a dejar en la misma partición que /usr habrá que prever el espacio que vayamos a necesitar, en su caso, y sumarlo.

También se puede dejar una partición separada para /boot. En mi caso, acostumbro a tener más de un kernel disponible, normalmente el que me instala la distro y uno o varios compilados por mí, depende de si actualizo versiones. Con una partición pequeña para /boot, de unos 200 Mb o incluso algo menos, me cubro si quiero formatear el raíz y /usr para una nueva instalación conservando mis kernels compilados, cuyos fuentes están en /usr/local/src/kernel/linux-[version], siendo /usr/src/linux un enlace simbólico a los fuentes de la versión del kernel compilado. Esto es también útil para compilar el kernel como usuario normal, dejando el acceso a root para casos realmente necesarios.

El directorio /tmp tiene su pequeña particularidad en caso de que deseemos situarlo en una partición independiente. Sería aconsejable dejarle al menos tanto espacio como la mayor cantidad de datos que podamos grabar en un sólo disco, ya sea CD o DVD. Digo esto porque hay que crear el archivo .iso con todo el contenido que queremos grabar, y el lugar idóneo para situar ese archivo es el directorio /tmp. Lógicamente, también habrá que tener esto en cuenta si decidimos dejar /tmp en la partición raíz /.

Sobre el espacio necesario para /var no tengo aún experiencia directa. Normalmente ocupará poca cosa salvo, según parece, el caso de que tengamos un uso intensivo de bases de datos. Actualmente me ocupa 360 Mb, pero hay previsto para un hipotético futuro un “uso intenso” de bases de datos.

Otro directorio candidato a una partición independiente es /var/log. Ahí se guardan los registros de los logs del sistema y, aunque normalmente su ocupación puede rondar los 25-50 Mb, si tenemos instalados servidores web, de correo, de ftp y cosillas de esas nos vendrían bien un par de Gb para esa partición que, estando separada del resto, contribuye a cubrirnos las espaldas

de un posible ataque de esos que inundan los servidores de logs, ya que se colapsaría únicamente esa partición, dejando el resto del sistema intacto.

En el caso de que Linux vaya a convivir con otro sistema operativo, es bueno saber que se dejará instalar en cualquier sitio, ya sea partición primaria, extendida, del primer disco o de cualquier otro. Si el “otro” SO es Windows® es preferible dejarle la primera partición primaria del primer disco para procurar que dé los menos problemas posibles en su instalación.



9. Comandos relacionados con las particiones y su formateo.

Hay varios comandos para acabar de perfilar el particionado de un disco, como la creación de los puntos de montaje, verificación de la propia partición, modificación de la revisión automática en el arranque... Vamos viendo los más usados:

mke2fs y mkswap.

Este comando se usa para la creación de un sistema de ficheros Linux con o sin Journaling (aunque se recomienda encarecidamente Journaling). Cuando tenemos una partición recién hecha deberemos usar este comando para crear el sistema de archivos:

```
# mke2fs -j /dev/hda1
```

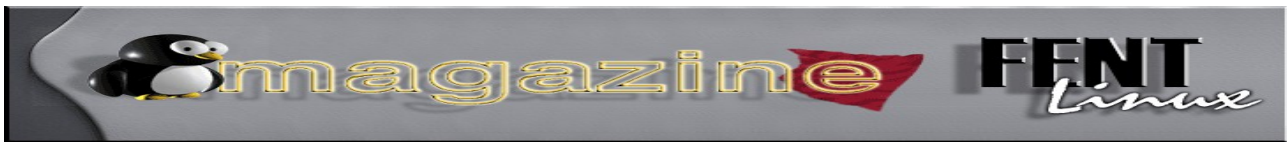
```
# mkfs.ext3 /dev/hda1
```

Esto crea un sistema de archivos en ext3, es decir, un ext2 con journaling (veáse el parámetro -j de la primera opción) y lo haremos en la partición primera del primer disco.

Para crear el sistema de archivos en la partición destinada para SWAP haremos:

```
# mkswap /dev/hda2
```

Que nos hará el sistema de archivos SWAP en la partición 2 del disco primero. Para activar esta partición deberemos teclear: `swapon /dev/hda2`



mkdir

Este comando se usa para crear directorios y lo usaremos para crear los puntos de montaje. Si tenemos un Windows instalado en el sistema y queremos que se monte en el inicio, deberemos decirle al sistema dónde hacerlo. Esto lo indicaremos en el archivo `/etc/fstab` que veremos más tarde, pero decir que se necesita un directorio para ello, por lo que lo crearemos a nuestro gusto.

Existen unas carpetas ya creadas llamadas `/mnt` ó `/media` en las que si entramos podremos crear mediante `mkdir` el directorio donde queremos montar nuestro Windows:

```
cd /mnt
```

```
mkdir mi-windows
```

Ya tenemos la carpeta para windows creada. Si tuviéramos una partición de almacenamiento (como es mi caso) tb deberíamos crear una carpeta para ella:

```
cd /mnt
```

```
mkdir almacenaje
```

Ala, ya la tenemos. Podemos darle el nombre que queramos a las carpetas.

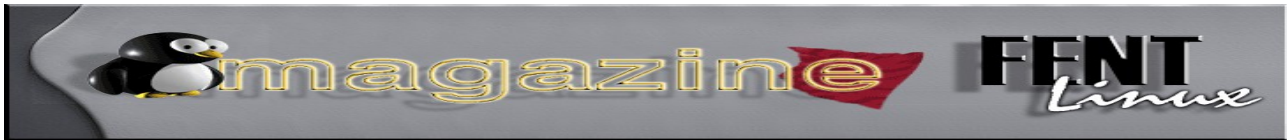
dumpe2fs y fsck.

Antes de entrar al trapo explicaremos algo sobre el "Superbloque". Es una sección de disco donde se guarda la información del sistema de archivos. Se divide en grupos de bloque, donde cada grupo es de 8192 bloques. Existen copias en los bloques 8193, 16385 ($8192 \times 2 + 1$), 14577...

Para comprobar esto se usa el comando `dumpe2fs`:

```
# dumpe2fs dispositivo | more
```

Para revisar la integridad de nuestro sistema de ficheros usaremos `fsck`, pero hay que hacerlo con la partición DESMONTADA (por ejemplo desde un



Live-CD) ya que si no se pueden producir daños en la partición. Lo haríamos del siguiente modo:

```
# fsck -t ext3 dispositivo
```

NOTA: Cambiar ext3 por el sistema de archivos que se tenga, sea ext2, Reiser, xfs...

```
e2fsck
```

Otro comando importante es e2fsck útil para diagnosticar y reparar problemas en las particiones. Si fallase el sistema de arranque del sistema de ficheros podríamos restaurar una copia del Superbloque mediante este comando. Teclearíamos la sintaxis siguiente:

```
# e2fsck -f -b copia dispositivo
```

Donde copia pertenece a la copia del Superbloque que nos da dumpe2fs y dispositivo sería nuestro disco/partición. Si no queremos complicarnos pondríamos:

```
# e2fsck -p /dev/hda3
```

para reparar automáticamente la partición /dev/hda3, aunque a veces esto no funciona porque las copias del Superbloque están corruptas en la ruta por defecto.

```
tune2fs.
```

Cuando arrancamos, Linux verifica las particiones y al cabo de cierto número de arranques (por defecto suele ser 30 arranques), automáticamente las revisa en mitad del arranque. Esto es algo heredado del sistema sin Journaling ext2, pero con el sistema ext3 esto apenas sirve de nada y a veces resulta molesto, por lo que podemos variar el número de arranques para que salte el chequeo o desactivarlo mediante tune2fs.

Primero activaremos el "indexado de directorios" para obtener un mayor rendimiento del sistema:

```
# tune2fs -O dir_index /dev/hda1
```

Si queremos desactivar los chequeos del inicio aplicaremos la siguiente sintaxis:

```
# tune2fs -c 0 -i 0 /dev/hda1
```

Esto hace que pongamos a cero la cuenta de montajes y también a cero el intervalo entre chequeos.

¿Y si quiero hacer los chequeos pero por mi cuenta?, pues usamos el comando anteriormente visto `e2fsck`:

```
# e2fsck -Dfy /dev/hda1
```

Y algo muy importante. Si nos equivocamos al elegir sistema de archivos y pusimos `ext2` en vez de `ext3`, `tune2fs` puede crear un `Journaling`. La forma de hacerlo es:

```
# tune2fs -O has_journal -o journal_data /dev/hda1
```



10. El archivo `/etc/fstab`

El archivo `/etc/fstab` contiene líneas donde se indica qué dispositivo debe montar y donde hacerlo, así como el sistema de archivos y otras opciones. Si intentamos montar un dispositivo que no se encuentra registrado en `/etc/fstab` no nos funcionará, a menos que dominemos el manejo del comando `mount` y sus numerosas opciones.

Aquí entra en juego todo lo que hemos visto anteriormente sobre las nomenclaturas que usa Linux para discos y particiones.

Un ejemplo de `/etc/fstab` es el siguiente:

```
# Discos SATA
/dev/sda1 / ext3 defaults 0 1
/dev/sda2 /mnt/winxp ntfs defaults,uid=1000,user,auto,umask=27 0 0
/dev/sda5 /mnt/store vfat defaults,uid=1000,user,auto,umask=27 0 0
/dev/sda6 none swap sw 0 0
```

```
# Disco IDE
/dev/hdb /mnt/home ext3 defaults 0 0

# Discos Compactos
/dev/hdd /mnt/cdrom iso9660 defaults,nosuid,nodev,noauto,user 0 0
/dev/hdc /mnt/dvd iso9660 defaults,nosuid,nodev,noauto,user 0 0

# Otros sistemas
none /proc proc defaults 0 0
none /dev/shm tmpfs defaults 0 0
usbdevfs /proc/bus/usb usbdevfs defaults 0 0
```

Como se puede ver, en este caso, que es mi fstab, tengo un disco SATA. Los cuales son reconocidos como sda (si tuviera otro disco seria sdb) y al lado aparece el punto de montaje, que debe existir (si no está se crea mediante el comando mkdir y como root).

Aparecen opciones como auto y noauto. Estas son para que el sistema monte o no monte los dispositivos en el arranque. Además aparece user, que es para que cualquier usuario pueda montar el dispositivo.

Los ceros y unos que aparecen es para que haga un chequeo en el arranque o no. El resto de opciones son más enrevesadas y se recomienda la lectura de algún documento que trate explícitamente sobre fstab.

11. Formateado de una memoria flash desde Linux.



Si queremos formatear una memoria flash desde Linux en vfat, que es el formato que se les suele aplicar, lo podemos hacer instalando el paquete dosfstools y ejecutando como root la siguiente orden sin que la unidad esté montada:

```
# mkdosfs -l -v /dev/sdX
```

Donde sdX es el dispositivo que el sistema reconoce al insertarlo en el conector USB. Si lo insertamos cuando estamos frente a un terminal no gráfico, de los que se consiguen tecleando Control-Alt-F2, por ejemplo (hasta F6), veremos aparecer información que nos indicará cuál es ese dispositivo.

12. Concluyendo.

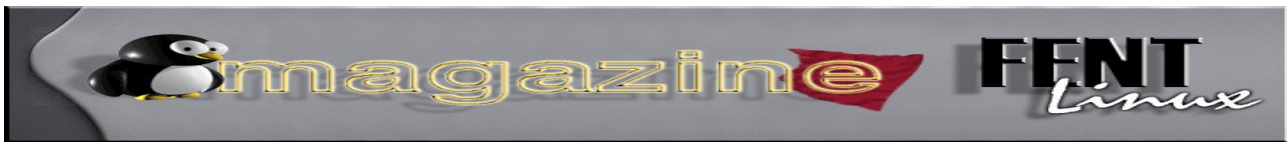


Como habéis podido ver en el documento, no es tan difícil el particionado cuando se conoce la geometría del disco y palabrejas como sectores, cluster... dejan de sonarnos a chino, y con la gran cantidad de herramientas que tenemos disponibles aun se nos facilita más la faena del particionado.

Esperamos que después de todas estas líneas haya quedado un poco más claro todo el lío este de las particiones, y si no ya saben por donde andamos, pregunten todo lo que necesiten saber y tengan claro siempre qué tareas van a realizar y por qué, guarden sus copias en buen lugar, nunca se sabe cuando pueden ser necesarias, y no tengan miedo a estropear algo, el método de ensayo y error, para muchos es el mejor sistema de aprendizaje.

Particionar no es misión imposible ni es especialmente complicado, que todos tarde o temprano nos hemos enfrentado a ello y si nosotros hemos sido capaces, ustedes también pueden lograrlo. Incluso con el tamaño actual de los discos, es hasta recomendable hacerlo.

Aún así, no me duelen prendas al reconocer que particionar puede ser el proceso más difícil y espinoso de todos los relacionados con linux, pero si tienen las ideas claras de qué es lo que quieren, una vez realizado veréis que es mucho más simple de lo que parece a priori, y que cuando lo hayáis hecho un par de veces hasta os reiréis de su dificultad. Así pues, a particionar se ha dicho.



13. Agradecimientos y créditos. Punto y final.



Desde [FentLinux](#) esperamos que este **magazine** os haya sido de utilidad, con él cerramos un ciclo, [el de los magazines](#), y a la par abrimos o mantenemos la apuesta por los **boletines**.

Para posibles interesados en colaborar en su mantenimiento y difusión:

[Contacto](#)

Recordar que el presente magazine, como todas las imágenes y texto de [FL](#) se distribuyen bajo la [licencia de contenidos](#) que rige en [FL](#)

Para el recuerdo: Enlaces a los magazines de fentlinux.

- * [1er Magazine FentLinux](#)
- * [2º Magazine FenLinux](#)
- * [3er Magazine FentLinux](#)
- * [4º Magazine FentLinux](#)
- * [5º Magazine FentLinux](#)

Nuestro más sincero agradecimiento a todos los que lo habéis hecho posible, mil gracias a todos aquellos que habéis participado en ellos con vuestros textos, imágenes, ayudas, sugerencias, reseñas y demás. Hasta pronto...