

# Software libre y GNU/Linux

Día a día y paso a paso el software libre se va haciendo hueco en nuestra vida. El ICAI y la Universidad Pontificia Comillas no se han quedado atrás y por fin el pingüino (Tux), buque insignia de Linux, accede a nuestra escuela. Pero, ¿qué es realmente el software libre? ¿en qué se diferencia del software gratuito? ¿qué es realmente GNU/Linux? Éstas serán algunas de las preguntas a las que buscamos respuesta, además de definir la arquitectura de un sistema GNU/Linux o explicar los pasos básicos necesarios para llevar a cabo la instalación del mismo.

## ¿Qué es el software libre?

Al principio de los años ochenta, un investigador del Instituto Tecnológico de Massachusetts (M.I.T.), Richard M. Stallman, se encontraba bastante descontento porque una de las impresoras de su departamento no funcionaba como él quería. Años atrás, Stallman habría pedido el software de la impresora a la empresa fabricante de la misma y él, aprovechando su amplios conocimientos de informática, habría retocado este software para adaptarlo a sus necesidades. Sin embargo, por aquél entonces las empresas fabricantes de hardware ya empezaban a ser reticentes a suministrar el código fuente de la aplicación que lo controlaba y esto fue la gota que colmó el vaso de la paciencia de Richard, quién recordó su época de estudiante en la cual trabajaba junto con sus compañeros, compartiendo y completando conocimientos. Fue entonces cuando decidió empezar desde cero y construir un sistema operativo en el que no ocurriera nada parecido a su incidente con la impresora. Por los tiempos que corrían a principios de los años ochenta, el mejor candidato en el que basarse era UNIX, y de éste modo

nació el proyecto GNU, acrónimo recursivo de GNU's Not UNIX (GNU no es UNIX), que se amparaba en la Licencia General Pública (GPL), compañera de cuna del GNU y en la que se definían las libertades que debían ir implícitas en lo que se conocería como software libre. Estas libertades son cuatro:

- **Libertad 0:** Libertad para usar el programa con cualquier propósito.
- **Libertad 1:** Libertad para estudiar cómo funciona el programa y adaptarlo a tus necesidades.
- **Libertad 2:** Libertad para distribuir copias sin restricciones de número.
- **Libertad 3:** Libertad para mejorar el programa y hacer públicas las mejoras, de forma que toda la comunidad se beneficie.

Estas cuatro libertades son los pilares del proyecto GNU y del software libre, además de los cuatro puntos que han revolucionado el modelo de desarrollo de software actual. Estas libertades dotan al usuario y al desarrollador de unas maravillosas ventajas completamente contrarias a lo que se conoce como *copyright*, por lo que todos aquellos programas amparados



**Joaquín Reyes González**

Estudiante de 5º curso de Ingeniería Industrial, en el ICAI. Escuela Técnica de Ingeniería Superior de la Universidad Pontificia Comillas.  
Presidente de la Asociación "GNU/Linux Entre Comillas".



**José Daniel Muñoz Frías**

Doctor Ingeniero del ICAI por la Universidad Pontificia Comillas. Profesor del Departamento de Electrónica y Automática de la Escuela Técnica. Sus áreas de conocimientos son sistemas digitales y control de accionamiento electrónicos.

por la GPL se les conoce como programas bajo licencia *copyleft*.

Para poder ejercer las libertades 1 y 3 es necesario disponer del código fuente del programa. Además, el hecho de hacer disponible el código fuente no sólo es beneficioso para el usuario final, sino para el autor del programa original y para la comunidad en general, ya que al existir un mayor número de personas mirando el código, éste se depura mucho más rápido y se adapta mejor a las diversas plataformas hardware existentes en el mercado. Esto hace que el modelo de desarrollo del software libre sea mucho más potente que el modelo del software propietario, en el que sólo tienen acceso al código sus programadores, normalmente un grupo reducido dentro de una empresa<sup>1</sup>. No obstante, entre las empresas siempre ha existido una enorme resistencia al término software libre, sobre todo en el mundo anglosajón, en donde *free* significa libre y gratis. Este hecho, unido a que la mayoría del software libre se distribuye sin cargo, hace que aunque la Free Software Foundation no cesa de insistir que *free* ha de entenderse como *free as in speech* (libre de libertad de expresión) no *free as in beer* (cerveza gratis) la mayoría de la gente piensa que el software libre ha de ser además gratuito, incluso fuera del mundo angloparlante. Para solucionar este problema e intentar embarcar al mundo empresarial nació el movimiento de código abierto (*open source*), en el cual se hace más énfasis en la necesidad de hacer accesible el código fuente por motivos prácticos de facilidad de uso y de desarrollo que por motivos filosóficos de acceso a la información. Además este movimiento aplica unos criterios un poco más laxos a las 4 libertades fundamentales para catalogar a un programa como de código abierto<sup>2</sup>.

### Ventajas del software libre

Pasados unos años desde la adquisición del software en propiedad, éste deja de ser mantenido por sus creadores (porque es antiguo, porque no se vende, porque no es rentable, etc.), de forma que si surge algún problema, no es posible acudir a nadie para solucionarlo. Como consecuencia de esto, acabaremos comprando una nueva versión (en el caso de que exista), que a su vez es

muy posible que no funcione con los ordenadores que teníamos, por lo que será necesario actualizarlos. Con el software libre, en cambio, dado que disponemos del código fuente podemos arreglarlo nosotros mismos o encontrar a alguien que nos ayude a hacerlo.

Antes de que un software en propiedad salga al mercado, se ve afectado por multitud de presiones comerciales, que generalmente recaen en los programadores (que está obligados a cumplir unos plazos), y que conllevan una disminución de la calidad del software: fallos de diseño, fallos imprevistos o falta de eficiencia. Sin embargo, como el software libre no está sujeto a su rentabilidad comercial, gran parte de estos problemas dejan de existir. Además, el hecho de poder copiar el software y distribuir las mejoras efectuadas sobre el mismo fomenta la cooperación, creando una sociedad más solidaria.

Actualmente, el gasto en software a la hora de comprar un ordenador supone, aproximadamente, un 30% o más del coste del mismo. Por ello, si utilizamos software libre siempre dispondremos de las cuatro libertades, pudiendo acceder al código fuente y distribuir tantas copias como deseemos, dejando así de tener gastos en la adquisición del software o, por lo menos, gastos considerables, ya que con una sola adquisición podremos instalarlo en tantos equipos como queramos, pudiendo incluso pedirlo prestado y copiarlo. Además del gasto inicial de licenciar el software, una buena parte del presupuesto de los departamentos de Informática lo comprende el mantenimiento. En el caso del software propietario, el mantenimiento puede realizarlo exclusivamente el fabricante del software, por lo que se crea una dependencia total de éste con los problemas que existen cuando no hay competencia, dada la dificultad de migrar los datos. Con un programa libre estos dos inconvenientes desaparecen: por un lado, pueden existir varias empresas que ofrezcan mantenimiento de un programa, además de los creadores, con lo que la competencia creada mejorará el servicio y los precios. Por otro lado, el formato de los datos deja de ser secreto, con lo que es posible migrar a otra plataforma sin un gasto excesivo al migrar los datos.

<sup>(1)</sup> Para que esto ocurra ha de crearse una comunidad de usuarios del programa con los conocimientos y la motivación necesaria para mejorarlo.

<sup>(2)</sup> Para obtener mayor información sobre el tema puede consultar las páginas de la Free Software Foundation ([www.fsf.org](http://www.fsf.org)) y de Open Source Initiative ([www.opensource.org](http://www.opensource.org)). También son ilustrativos los libros [1] y [2].

## ¿Qué es Linux?

Es un sistema operativo libre, multiusuario, multitarea y multiplataforma. A la hora de nombrar a Linux existen dos posibilidades: Linux o GNU/Linux, siendo éste último el nombre que algunas fuentes dan al conjunto del núcleo del sistema operativo (Linux) junto con las aplicaciones libres disponibles, desarrolladas en el proyecto GNU (como, por ejemplo, compiladores `-gcc-`, editores de texto `-emacs-` y un largo etcétera) en su mayoría, para reconocer la importancia del proyecto GNU. Sin embargo, otras fuentes no hacen esta distinción, sino que denominan del mismo modo, Linux, tanto al núcleo del sistema operativo como al resto de aplicaciones básicas que se distribuyen con él.

En cuanto a los requisitos para implantar Linux, cabe mencionar que las exigencias de hardware son mucho menores que las de otros sistemas operativos, por lo que se puede utilizar en máquinas con menos prestaciones. Esto es precisamente una consecuencia de ser un sistema libre, ya que cada usuario puede incluir los componentes que necesite, al contrario que un sistema cerrado en que se suministra el programa compilado con todas las opciones para que funcione en todo tipo de máquinas. Sin embargo, los inconvenientes llegan a la hora de evaluar la compatibilidad de Linux con el hardware más actual, ya que normalmente los fabricantes no suelen suministrar los drivers de forma libre, aunque cada día se plantean más alternativas y posibilidades, permitiendo a Linux funcionar cada vez en más plataformas.

Además, no hay que olvidar que todo lo que es Linux depende de la comunidad de usuarios Linux, ya que ellos son los que han fijado el pasado, fijan el presente y fijarán el futuro de Linux, ya que posee una gran cantidad de usuarios en todo el mundo y, aunque inicialmente sus usuarios eran de perfil técnico, actualmente cualquiera con un mínimo de curiosidad puede utilizarlo gracias a la mayor facilidad de instalación de las distribuciones actuales.

## ¿Qué es una distribución?

Linux es el nombre del sistema operativo, pero cuando se instala una distribución Linux no sólo se instala el sistema operativo, sino un conjunto de aplicaciones libres. Por lo tanto, una distribución es un conjunto de paquetes (aplicaciones organizadas por una empresa o asociación) que engloba el sistema operativo y un conjunto de aplicaciones

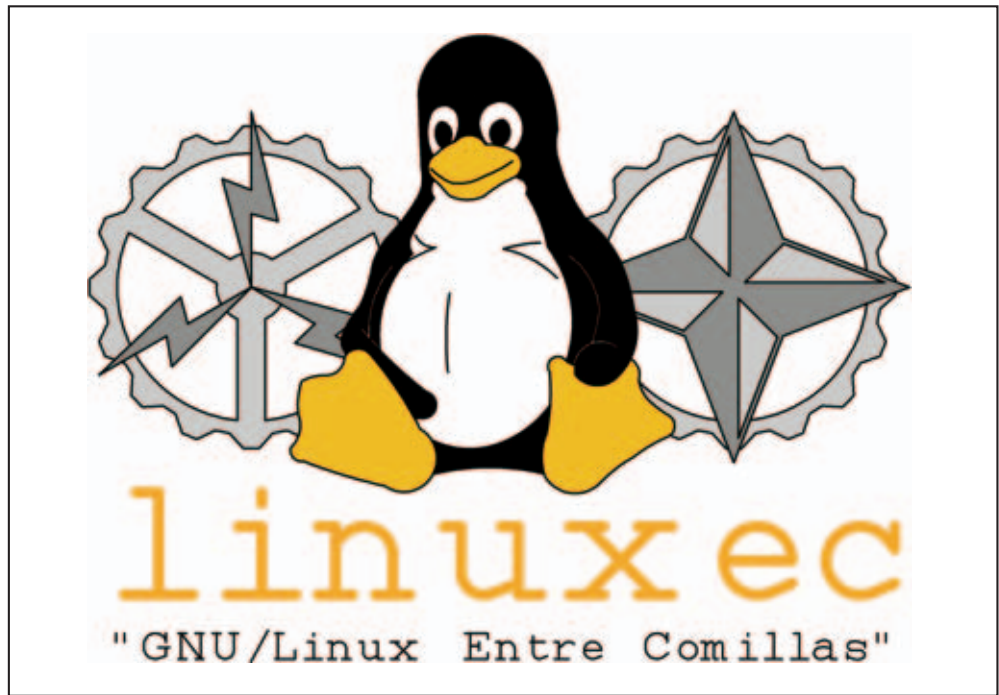
(editores y procesadores de texto, hojas de cálculo, programas para gráficos, multimedia, compiladores y herramientas de programación, Internet, juegos, etc.). Las distribuciones de GNU/Linux están accesibles por medio de una serie de CDs, DVDs o por Internet (tanto la instalación como las actualizaciones) y pueden ser instaladas de forma sencilla.

Casi la totalidad del software contenido en una distribución es software libre recopilado, y dado que las aplicaciones son recogidas de sus autores, existe una gran compatibilidad entre distribuciones (los paquetes no son compatibles, pero sí las aplicaciones). Para asegurar la correcta instalación de los programas dentro de las distribuciones, estos se distribuyen en un formato determinado (`rpm` y `deb`, principalmente) que no sólo almacena el propio programa como tal, sino también una descripción del mismo y los programas de los que depende, de forma que las herramientas de instalación de paquetes, cuando instalan un programa, instalan también los programas y librerías de los que depende, facilitándose enormemente la instalación.

## Arquitectura de un sistema Linux

Lo primero que vemos al acceder a un sistema Linux es algo relativamente similar a lo que encontraríamos frente a un sistema Unix, pero considerablemente diferente a lo que podemos encontrar en otros sistemas propietarios. En un principio se podría decir que todo sistema Linux posee un núcleo, *kernel*, que es lo primero que se carga al arrancar la máquina y permite al usuario interactuar con ella a un nivel muy básico. Una vez que se ha cargado el núcleo del sistema, llega el momento de ir activando los servicios necesarios (red, hora del sistema, acceso a dispositivos, etc.). En este momento, ya tendríamos un sistema Linux funcionando pero sin un entorno gráfico, por lo que el siguiente y último paso, aunque no siempre necesario, sería arrancar un servidor y un cliente gráfico, con los cuales tengamos un aspecto visual más atractivo y un manejo más cómodo.

En el caso de un ordenador personal, la primera diferencia apreciable es que cualquier dispositivo es identificado por un archivo. De esta forma, por ejemplo, la unidad de lectura óptica de nuestro ordenador sería identificada por el archivo `hdb`, siempre y cuando se encuentre conectada como esclava del primer disco duro. Esto conlleva algunas



Logo de la Asociación "GNU/Linux Entre Comillas"

modificaciones como la necesidad de tener que montar y desmontar los dispositivos de lectura y/o escritura. La segunda diferencia sería la organización de los archivos dentro del ordenador, ya que todos los archivos cuelgan del directorio raíz, *root* *-/-*. Por lo tanto, siguiendo con el ejemplo anterior, nuestra unidad de lectura óptica se correspondería con el archivo *hdb*, que se encontraría dentro del directorio correspondiente a los dispositivos (*dev*), es decir, nuestra unidad de lectura óptica, en un sistema Linux, la encontraríamos ubicada en */dev/hdb*. La tercera y última diferencia es el formato que posee el sistema de archivos, ya que cada sistema operativo suele poseer el suyo, por lo que al igual que estamos acostumbrados a oír siglas como NTFS o FAT32, frente a un sistema Linux tendríamos que decir sus análogas, que son: EXT3 o ReiserFS. Otra característica de los sistemas UNIX es la necesidad de contar con una partición específica para ser usada como memoria virtual, que se denomina área de intercambio (*swap* en inglés). La ventaja de esta solución es independizar este espacio del resto del sistema, para hacerlo más robusto, aunque complica un poco la instalación.

### Organización de ficheros

A la hora de decidir dónde se coloca cada archivo que compone un sistema GNU/Linux es necesario establecer un cierto estándar,

para que sea posible alcanzar una uniformidad y poder así facilitar el desarrollo en grupo, ya que de nada sirve que dispongamos del código fuente si luego cada desarrollador coloca su código donde él cree que es mejor. Fue por este motivo por el que en 1994 se creó un documento, llamado FSSTND (abreviatura poco afortunada de *File System STaNDard*) que posteriormente fue revisado en 1999, para alcanzar una mayor semejanza con Unix, dando lugar al definitivo documento conocido como FHS (*Filesystem Hierarchy Standard* o Jerarquía Estándar del Sistema de Ficheros). En este documento quedan establecidos los directorios mínimos que deben encontrarse dentro del directorio raíz (recordemos que hacíamos alusión a él con la barra de dividir), como el uso que se le debe dar a cada uno de ellos. En resumen, se podría decir que los directorios básicos y su uso deben ser los siguientes:

- **/bin:** Aloja los comandos esenciales utilizados por todos los usuarios.
- **/boot:** Aquí se encuentran ficheros necesarios para el arranque, como por ejemplo el núcleo del sistema operativo.
- **/dev:** Éste directorio ya nos resulta familiar, ya que en él se encuentran todos los archivos correspondientes a los dispositivos de la máquina.
- **/etc:** Archivos de configuración específica del sistema.

- **/home:** A este directorio van a parar los archivos personales de los usuarios de la máquina, los cuales, a su vez, suelen tener su propio subdirectorio.
- **/lib:** Librerías y bibliotecas esenciales, las cuales son usadas por los programas para realizar tareas sencillas como, por ejemplo, adquirir datos.
- **/mnt:** Directorio en el que se suelen montar de forma temporal los diferentes dispositivos de lectura y/o escritura.
- **/opt:** Aquí se suelen instalar los paquetes de software opcionales, que no vienen incluidos en la distribución GNU/Linux.
- **/sbin:** Comandos esenciales del sistema, que debido al riesgo que puede conllevar su uso se reservan aparte para el administrador de la máquina.
- **/tmp:** Directorio para los archivos temporales.
- **/usr:** Destino de los paquetes de software que se adjuntan a cada distribución.
- **/var:** Los datos variables que hacen que cada aplicación funcione tal y como el usuario desea van a parar a este directorio.

Merece la pena destacar cómo sistemas operativos basados en Unix más recientes, como puede ser el caso de Mac OS X, se han decantado por modificar levemente esta estructura de directorios, dando lugar a cierta confusión en algunos casos.

## Instalación de un sistema Linux

Llegados a este punto, resulta esencial describir cómo es el proceso de instalación de un sistema GNU/Linux, de forma que podamos poner en práctica todo lo descrito hasta aquí, siendo conscientes de que cada distribución posee su propio manual de instalación, correctamente explicado y particularizado. Además es interesante destacar que, en el caso de un ordenador personal, es posible tener varios sistemas operativos instalados, así como varias distribuciones del mismo sistema operativo.

El primer paso sería tener en cuenta dónde se desea instalar el sistema, esto es, tener claro qué discos duros o qué particiones se desean utilizar, partiendo de la base que necesitamos al menos un disco duro con un mínimo de dos particiones o, en su defecto, con dos discos duros y una partición en cada uno de ellos, siendo esta la opción menos recomendada. De las dos particiones necesarias, una de ellas será destinada a ser usada como área de intercambio (para la memoria virtual) y la otra

albergará el sistema de ficheros, con la estructura ya conocida.

El segundo paso necesario sería recopilar la mayor información posible del hardware de la máquina en la que queremos llevar a cabo la instalación, para que podamos ajustar y personalizar nuestro sistema al máximo, consiguiendo así el mejor rendimiento posible.

El tercer y último paso es llevar a cabo la instalación propiamente dicha, copiando los ficheros necesarios al sistema de ficheros y creando los usuarios necesarios, los cuales suelen ser al menos dos: un usuario para administrar la máquina (normalmente conocido como *root*) y un usuario para el uso cotidiano de forma que no exista el riesgo de realizar alguna acción contraria al correcto funcionamiento del sistema de forma accidental. Si se trata de una instalación con más sistemas operativos ya instalados, será necesario terminar el proceso con la instalación de un gestor de arranque, que permita elegir el sistema operativo al arrancar la máquina.

Actualmente, el proceso de instalación de una distribución GNU/Linux se encuentra cada vez más automatizado y mejor asistido, por lo que se convierte en algo relativamente sencillo si se tienen claro los tres pasos descritos. Además, ya es posible probar una determinada distribución sin necesidad de instalarla gracias a la existencia de las versiones denominadas Live, que se pueden cargar directamente desde un CD o un DVD y que permiten probar cómo se comportan en una máquina determinada. ■

### Links de interés

<http://www.gnu.org/gnu/thegnuproject.es.html>  
<http://www.gnu.org/philosophy/free-sw.es.html>  
<http://www.gnu.org/philosophy/why-free.es.html>  
<http://www.gnu.org/philosophy/right-to-read.es.html>  
<http://www.sindominio.net/biblioweb/telematica/catedral.html>  
<http://www.linuxiso.org>  
<http://www.linuxec.tk>

### Bibliografía

- [1] Eric S. Raimond. "The cathedral and the bazaar. Musing on Linux and open source by an accidental revolutionary". O'Reilly 2001. Disponible on-line en <http://www.catb.org/~esr/writings/cathedral-bazaar/>
- [2] Richard M. Stallman. "Free Software, Free Society: Selected Essays of Richard M. Stallman". GNU Press 2002.