

## **Elementos fundamentales de los sistemas operativos**

***Sistemas de Procesamiento de Datos  
Tecnatura Superior en Programación.  
UTN-FRA***

**Autores:** *Prof. Martín Isusi Seff*

**Revisores:** *Prof. Marcos Pablo Russo*

*Versión : 1*



Esta obra está bajo una [Licencia Creative Commons Atribución-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-sa/4.0/).

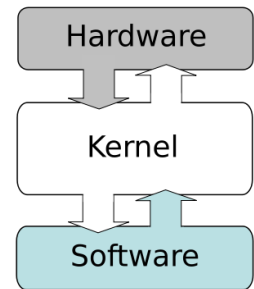
El sistema operativo, desde el punto de vista del software, es aquel que provee las funciones fundamentales a una computadora. Es aquel que permite interactuar con el hardware, define la interfaz con el usuario y da las herramientas básicas para que una computadora sea útil.

**¿Qué es el kernel?**

El kernel, o núcleo del sistema operativo es el componente responsable de realizar el manejo de las funciones de bajo nivel de la computadora, por ejemplo:

- Ubicar un programa en la memoria
- Asignar el tiempo de CPU de un programa
- Hacer de interfaz entre el software y los dispositivos
- Permitir la interacción entre diferentes programas

Cuando uno usa cualquier programa, este recae en el kernel y muchas de sus funciones básicas. Por ejemplo, un navegador de internet se comunica con el exterior utilizando las funciones de red del kernel. Además, será el kernel quién ubique al navegador en la memoria principal y asigne el tiempo de la CPU sin el cual el navegador no podría funcionar. Si el navegador necesita de algún software adicional para mostrar, por ejemplo, contenido multimedia, será el kernel quién administre la comunicación entre ambos programas. Podríamos decir que el kernel es el componente que mantiene unidas todas las partes de una computadora, sin él, las computadoras serían prácticamente inútiles.



*Figura 1. Esquema de interacción entre el hardware, el kernel y el software*

Cada sistema operativo tiene su propio kernel, es decir que, Mac OS X tiene un kernel distinto al que podemos encontrar en cualquier versión de Windows o las distribuciones de Linux. Cada uno está diseñado de una manera distinta, por lo que provee las funcionalidades de una manera diferente, aunque la funcionalidad general sea la misma.

### **Además del kernel, ¿Qué diferencia un SO de otro?**

El kernel es el componente central de cualquier sistema operativo, pero no es algo que el usuario manipule directamente. En su lugar, la mayoría de los usuarios interactúa con distintos componentes de software, muchos de los cuales están asociados con un sistema operativo en particular. Estos programas incluyen:

- **Intérprete de línea de comandos o command-line shells.** Hace algunos años, los usuarios interactuaban con la computadora exclusivamente escribiendo comandos en un programa conocido como shell o intérprete de línea de comandos. Tales comandos se utilizaban para renombrar archivos, ejecutar programas o realizar cualquiera de las funciones que permitía el sistema operativo. Aunque hoy en día la mayoría de los usuarios no utiliza (y en algunos casos ni conoce) la línea de comandos, esta interfaz sigue siendo imprescindible en muchos casos. Existen distintos tipos de shells, entre los cuales se encuentran sh (bourne shell), bash (bourne-again shell), csh (c shell), entre otros.
- **Interfaz gráfica.** La interfaz gráfica o GUI (Graphical User Interface) es una interfaz que proporciona un uso mucho más intuitivo de la computadora. En lugar de ejecutar comandos, una interfaz gráfica basa su manejo en íconos, menús, y punteros de mouse. Microsoft Windows y Mac OS tiene sus propias interfaces gráficas, mientras que Linux se basa en una interfaz gráfica conocida como X Windows System o simplemente X. X es una interfaz de usuario básica, por lo que las distintas distribuciones basadas en Linux utilizan


además paquetes de software conocidos como entornos de escritorio. Algunos ejemplos de entornos de escritorio son GNOME, KDE, XFCE, etc.

- **Programas utilitarios.** Hoy en día, la mayoría de los sistemas operativos se instalan con varios programas utilitarios como calculadoras, calendarios, aplicaciones para el mantenimiento de discos, etc. Estos programas varían entre los distintos sistemas operativos, así como el nombre, y la forma de usarlos de cada uno<sup>1</sup>.
- **Librerías.** Las librerías son archivos que contienen las funciones básicas que utilizan las distintas aplicaciones. Existen librerías específicas para cada aplicación o para cada funcionalidad del sistema operativo. Cada sistema operativo tiene su propio conjunto de librerías y, además, puede tener librerías para distintas arquitecturas de hardware, por ejemplo, las distribuciones basadas en GNU/Linux tienen todas las librerías ubicadas en un directorio llamado lib, pero, además, existe lib64 que almacena las mismas librerías, pero compiladas para la arquitectura de microprocesadores de 64 bits. Cuando un sistema contiene tanto los directorios lib y lib64 se dice que el sistema operativo es Multilib.
- **Programas varios.** Cada sistema operativo tiene su propio conjunto de aplicaciones de uso común como navegadores, software de ofimática, etc. Aunque generalmente estas aplicaciones no son un componente del sistema operativo, estrictamente hablando, pueden estar instaladas por defecto en ellos.

## Uso básico de las distintas interfaces de usuario

### Uso de la interfaz de línea de comandos

Hoy en día, la mayor parte de los usuarios utiliza una interfaz gráfica, independientemente del sistema operativo. Usamos una interfaz gráfica en la computadora, en un dispositivo móvil o en una tele. Esto no quiere decir que la interfaz de línea de comandos no sea útil, de hecho, es todo lo contrario. La interfaz gráfica es muy importante para el uso general de una computadora. Realizar tareas



```

GNU nano 2.2.4 File: /mnt/etc/fstab
/etc/fstab: static file system information

# file system      <dir>      <type>      <options>      <dump> <pass>
# /dev/pts         /dev/pts   devpts      defaults       0       0
# /dev/shm         /dev/shm   tmpfs       nodev,nosuid   0       0
# /dev/cdrom       /media/cd   auto        ro,user,noauto,unhide 0       0
# /dev/dvd         /media/dvd  auto        ro,user,noauto,unhide 0       0
# /dev/fd0         /media/f1   auto        user,noauto     0       0
UUID=6dad4fac-0f3c-4939-80d6-f2004d47c166 / ext3 defaults 0 1
UUID=74650592-b40e-448b-a2df-02867c756f04 swap swap defaults 0 0
UUID=c69098b4-a0d5-451c-9935-dc2ad4a29949 /home ext3 defaults 0 1
UUID=d38a7644-c989-4d46-95f9-b44990d9ffe7 /boot ext2 defaults 0 1

[ Read 15 lines ]
G Get Help  O WriteOut  R Read File  Y Prev Page  R Cut Text   C Cur Pos
X Exit      J Justify   W Where Is  U Next Page  U UnCut Text T To Spell

```

Figura 2. El editor de texto GNU nano, incluido en la mayoría de los sistemas operativos GNU/Linux, es un ejemplo de aplicación que se ejecuta en línea de comandos en lugar de tener una interfaz gráfica.

<sup>1</sup> Existen páginas en las que podemos encontrar las equivalencias entre aplicaciones para los distintos sistemas operativos. Algunos ejemplos son [www.linuxrsp.ru/win-lin-soft/table-eng](http://www.linuxrsp.ru/win-lin-soft/table-eng) o [www.linuxalt.com](http://www.linuxalt.com).  
UTN FRA – Tecnicatura Superior en Programación. <http://www.sistemas-utnfra.com.ar>

básicas es siempre más fácil con la ayuda de íconos, un cursor, o cualquiera de los elementos que proveen las interfaces gráficas. Pero algunas tareas que no son de uso general, no son tan simples o no es tan conveniente realizarlas con una interfaz gráfica y, por el contrario, es mejor usar una línea de comandos. Un ejemplo podría ser la creación de usuarios en un sistema: Si tengo que crear mil usuarios en un sistema operativo, con una interfaz gráfica puede ser una tarea tediosa, porque implica que deba moverme por los diferentes controles, ingresando la información, aceptando mensajes de alerta, etc. Automatizar una interfaz gráfica no es una tarea simple y siempre requiere de algún software externo. Por el contrario, si esta tarea fuera hecha a través de una interfaz de línea de comandos, probablemente la creación de los usuarios se realice con un solo comando lo cual simplificaría el asunto y, además, las interfaces de línea de comandos son muy simples de automatizar. Los servidores no utilizan interfaz gráfica y su administración es a través de línea de comandos.

### **Uso de la interfaz gráfica**

La mayoría de los usuarios prefiere utilizar interfaces gráficas (GUI por sus siglas en inglés. Graphical User Interface), dada la simplicidad y la manera intuitiva en la que presenta la información. Es por eso que la mayoría de los sistemas operativos orientados al usuario común se instalan con una interfaz de usuario.

A diferencia de Windows y OS X, GNU/Linux provee distintas interfaces gráficas, o, entornos de escritorio. Los entornos de escritorio dependen de la distribución de GNU/Linux que se esté utilizando y de las aplicaciones que se elijan al momento de la instalación. Las opciones más comunes en lo que respecta a entornos de escritorio en Linux son: GNOME, KDE, XFCE, y Unity (Basado en GNOME y que actualmente es el escritorio por defecto de Ubuntu).

### **¿Qué rol cumple GNU/Linux dentro de los SO?**

Si bien el apunte estará enfocado en el análisis de la arquitectura y los componentes que conforman un sistema operativo, la mayoría de los análisis prácticos estarán hechos puntualmente sobre sistemas operativos basados en GNU/Linux. Por lo tanto, será necesario conocer las diferencias entre los distintos tipos de sistemas operativos.

### **Comparando GNU/Linux con Unix**

Unix es un sistema operativo creado en 1969 en los laboratorios Bell de la compañía AT&T. Inicialmente, este SO estaba escrito en un lenguaje de ensamblado, lo que hacía que sólo pueda ser ejecutado en configuraciones de hardware muy específicas. Es por eso que, en el año 1972, el sistema operativo completo se reescribió en lenguaje C y de ahí en más, fue posible la compilación para distintos dispositivos. Esto hizo que sea posible la revisión y recompilación del código fuente por parte de distintos equipos de desarrollo, tal es el caso del equipo de investigación y desarrollo de la Universidad de Berkeley que en 1977 desarrolla BSD,

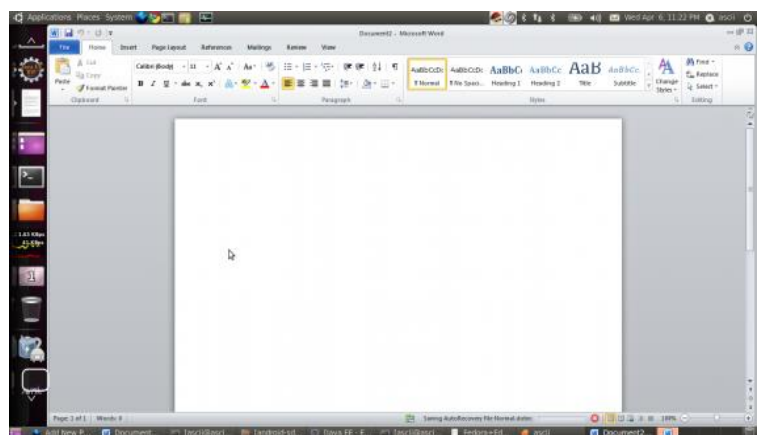
que era un sistema operativo completo basado en la sexta versión de Unix. Ante el avance de BSD, Bell realiza una demanda contra la Universidad de Berkeley, argumentando que BSD contenía código que era propiedad de la empresa. Luego de esta demanda, el desarrollo de sistemas operativos basados en Unix se vio limitado, es por eso que, en 1983, Richard Stallman decide crear GNU que comprenderá una copia de todas las aplicaciones que conformaban Unix, sin utilizar el código fuente del mismo. Para poder completar un sistema operativo completo, al proyecto GNU le faltaba un kernel. Esto se solucionó en el año 1991 con el desarrollo del kernel Linux por Linus Torvalds. El conjunto de GNU y Linux da como resultado la mayoría de los sistemas operativos actuales, que nosotros llamamos simplemente "Linux" aunque en realidad deberían ser llamados GNU/Linux.

Para concluir la comparación, podemos decir que GNU/Linux (muchas veces se dice Linux y esto en realidad está mal dicho, ya que Linux es el kernel y GNU las aplicaciones) puede ser considerado con un miembro de la familia de sistemas operativos Unix, aunque técnicamente no lo sea. Debido a su popularidad, GNU/Linux es compatible con una variedad más amplia de hardware que los sistemas operativos íntegramente basados en Unix.

### **Comparando GNU/Linux con Mac OS X**

Hoy en día, el sistema operativo más utilizado en dispositivos de escritorio o notebooks es Windows, pero, ¿Cuál es la diferencia entre Windows y GNU/Linux?

La mayor diferencia radica en la licencia. Los sistemas operativos GNU/Linux son de código abierto, mientras que Windows se distribuye en base a una licencia comercial de software propietario. Por otro lado,



*Figura 3. Microsoft Office Word, ejecutándose en Ubuntu, utilizando WineHQ*

muchas distribuciones de Linux son gratuitas (al menos para su instalación), esto hace que el costo sea otra de las grandes diferencias, ya que todas las versiones de Windows se comercializan.

Con respecto a la compatibilidad, generalmente los fabricantes de hardware desarrollan los drivers específicamente para ser instalados en las diferentes versiones de Windows, aunque muchas veces también proveen de los drivers necesarios en su versión para GNU/Linux. Muchas veces, ciertos componentes suelen funcionar mejor bajo Windows, ya que fueron fabricados específicamente para ese sistema operativo, tal es el caso de las placas de video, que generalmente suelen funcionar mejor en las distintas versiones de Windows. Pero este no

es siempre el caso, ya que Windows es un sistema operativo comercial, esto hace que cada vez que una nueva versión del sistema operativo sale al mercado, los desarrollos se orienten al mismo, dejando atrás los desarrollos para versiones anteriores. Esto aplica también a los drivers, por lo tanto, muchas veces no es posible utilizar ciertos dispositivos “viejos” en nuevas versiones de Windows. Linux, por su parte, recibe soporte de los fabricantes, pero también existen muchos drivers alternativos que son de código abierto. Generalmente estos drivers se mantienen a pesar del desarrollo de nuevas versiones del sistema operativo, por lo que es posible utilizar hardware “viejo” con nuevas versiones del SO.

Una de las principales comparaciones que muchas veces deja afuera la opción de un sistema operativo GNU/Linux es la disponibilidad de aplicaciones. Al ser Windows el sistema operativo más instalado en equipos de escritorio y notebooks, la mayoría de las aplicaciones suele desarrollarse para esta plataforma. Tal es el caso de Office, Photoshop, etc. De todas maneras, existen alternativas a cada una de estas aplicaciones, disponibles para las distribuciones de GNU/Linux. Tal es el caso que, por cada aplicación o suite de aplicaciones para Windows, existen una o más de una alternativa en GNU/Linux. Existe, además, la posibilidad de instalar librerías que hacen posible la ejecución de aplicaciones Windows en Linux. La librería más conocida corresponde al software WineHQ, que hace posible la ejecución de la mayoría de las aplicaciones de Windows bajo el entorno GNU/Linux.

En cuanto a la interfaz gráfica, al igual que OS X, Windows posee su propia interfaz de usuario. Desde las primeras versiones de Windows, lo que se intentó hacer fue crear un sistema operativo diseñado para el usuario común, por lo tanto, la interfaz gráfica siempre fue amigable e intuitiva. Esto hizo que hoy en día Windows sea el SO más popular. De todas maneras, los entornos de escritorio disponibles para GNU/Linux fueron evolucionando y hoy pueden competir sin dudas con la interfaz de Windows. Existen, de hecho, muchos entornos de escritorio que plantean una similitud con la interfaz de Windows en cuanto al diseño (XFCE, LXDE, etc).

Como último punto de comparación tenemos la configuración y seguridad. Con configuración, nos referimos a la posibilidad de modificar el SO y adaptarlo a cada necesidad particular. En este punto, GNU/Linux es superior a Windows. Evidencia de esto es la innumerable cantidad de sistemas operativos basados en GNU/Linux y orientadas a distintos usos, por ejemplo, Scientific Linux, orientado a grandes aplicaciones científicas (utilizado por la Organización Europea para la Investigación Nuclear (CERN) en el acelerador de partículas más grande del mundo), Ubuntu, orientado al usuario medio, Kali Linux, orientado a seguridad informática, Huemul, orientado a informática forense etc. Si hablamos de seguridad, la arquitectura bajo la cual se desarrolla Linux, provee un uso más seguro del sistema operativo, limitando a los usuarios de una manera más estricta en la que se maneja Windows.

Para resumir, por casi dos décadas, Windows fue el sistema operativo dominante en lo que respecta a computadoras de escritorio y notebooks. Tanto en uso particular como en uso de oficina, los usuarios fueron familiarizándose con la interfaz de ese sistema operativo y muchas de las aplicaciones de uso general como Microsoft Office, etc. Si bien GNU/Linux podría ser utilizado en esos casos, el hecho de que los usuarios estén acostumbrados a utilizar Windows hace que el cambio hacia GNU/Linux sea una ardua tarea. Unix y GNU/Linux particularmente, han sido dominantes en el mercado de los servidores, debido a su manejo de la seguridad y los permisos, el rendimiento de los mismos, y las posibilidades de configuración y escalabilidad. Si bien Windows comercializa una versión dedicada a ser utilizada como servidor, nunca logró captar la mirada del mercado.

### **¿Qué es una distribución?**

Hasta ahora, esta palabra se presentó varias veces en el texto, pero ¿Qué es una distribución específicamente? ¿A qué nos referimos con 'distribución'?

Nos referimos a distribución al conjunto del kernel de Linux, el conjunto de aplicaciones fundamentales que hacen al sistema operativo, y las configuraciones del mismo.

Desde el comienzo, numerosas distribuciones de GNU/Linux se fueron desarrollando, como se muestra en el siguiente gráfico:



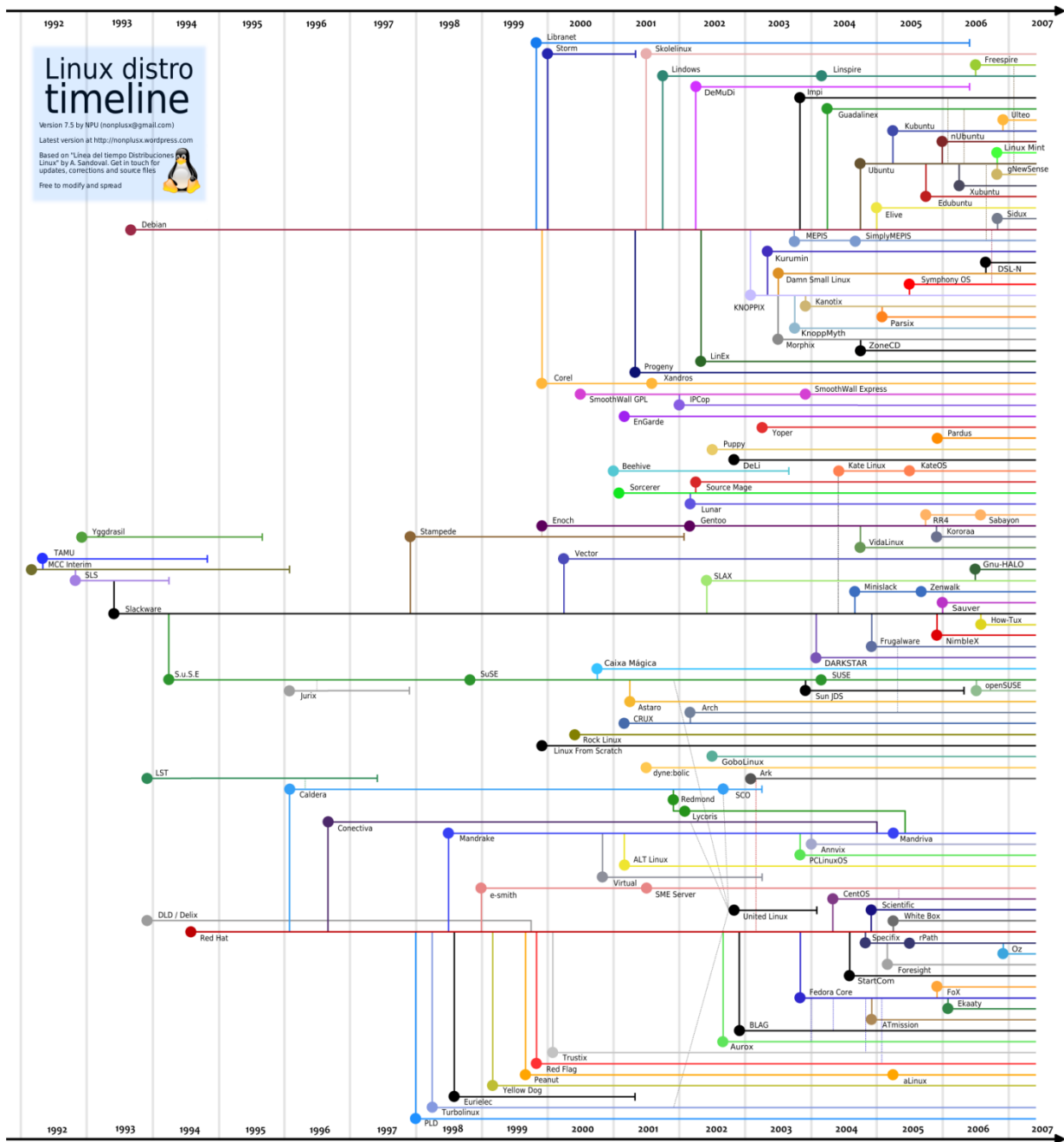


Fig. 1 Si bien el gráfico muestra las distribuciones, la intención del mismo es demostrar la enorme cantidad de distribuciones surgieron, y dar una idea de la cantidad de distribuciones que aún están por surgir.