


Semana 1: Primeros pasos con Linux para sistemas embebidos

SISTEMAS EMBEBIDOS 
(EMBEDED SYSTEMS)



Deusto

Facultad de Ingeniería
Ingeniaritza Fakultatea

Grado Dual en Industria Digital

Campus Vitoria

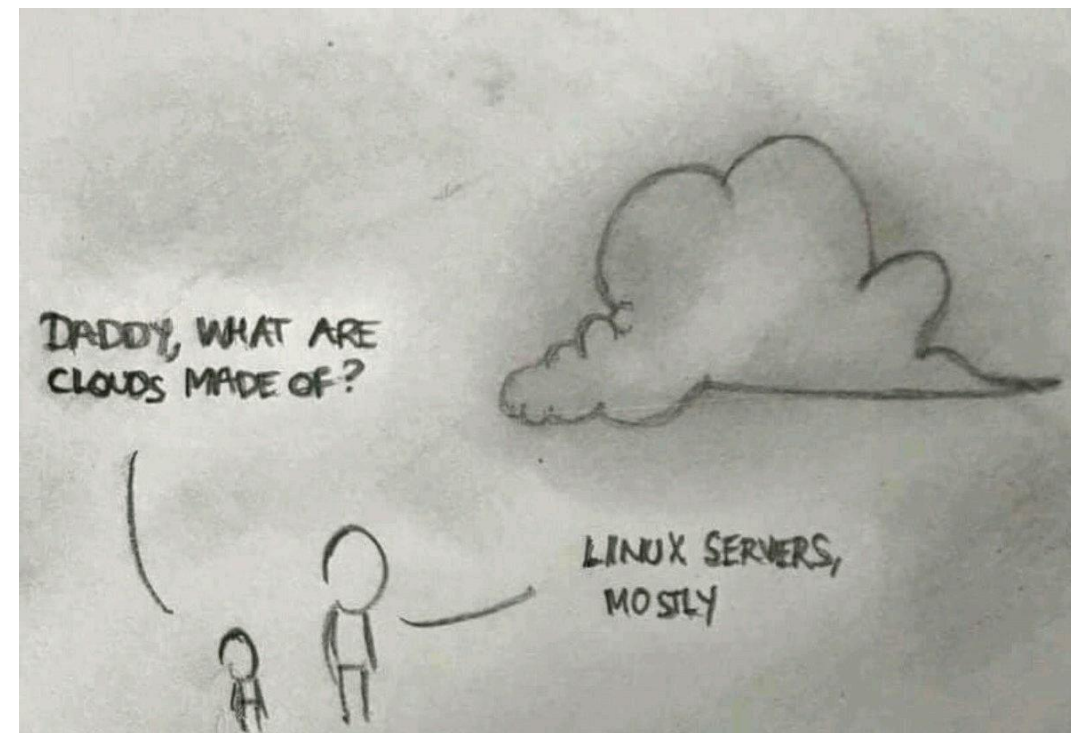
Curso 2020-2021

“Linux Embebido”?

- A pesar del termino Linux Embebido, no hay tal cosa como Linux Embebido!
- No hay una version especial del kernel de Linux para sistemas embebidos, sino el kernel principal de Linux ejecutandose en un Sistema embebido.
- El termino Linux Embebido tiene un uso muy Amplio y comun; por tanto, se usa en vez de “Linux en un Sistema embebido”.

Pros

- ✓ Linux es un Sistema operativo escalable y eficiente, que puede encontrarse desde dispositivos de bajo coste orientados al consumidor, hasta servidores a gran escala. Ha evolucionado mucho a lo largo de los años, y ha mantenido muchas de sus eficiencias.
- ✓ Una gran cantidad de programas y herramientas *open source* han sido desarrolladas de forma que pueden usarse directamente en una aplicación embebida. Si necesitas servidor web para tu aplicación embebida, puedes instalar la misma que usarías en un servidor linux.
- ✓ Es de código libre y no requiere un coste alguno su uso
- ✓ El Kernel y el código de aplicación se usa a nivel mundial en tantos dispositivos que los bugs son muy poco frecuentes y detectados rápidamente.



- Diapositivas adaptadas de Diego Casado Mansilla



Powered by
LINUX

GNU/Linux

```
Kernel
+
| + eloha 3.1
+
Weather
+
| + Now: Sunny
Today: 13-21C
Tomorrow: 14-21C
+
Time
+
| + 09:05 20
+
System
+
| + CPU: 10%
| + RAM: 21%
| + Temp: 44C
| + /: 16%
| + /var: 15%
| + ~/: 37%
+
Net
+
| + Up: 08h
| + Down: 21h
+
Processes
+
| + NAME
| + pulseaudio
| + pavucontrol
| + plugin-container
| + mpd
| + firefox
+
Gmail
+
| Unread: 4
+
Music
+
| Annaes Okei
| Aventura
| The Curse
| 4/15/5:53
| Playing
```



- **UNIX:**
 - Sistema de comienzos de los 70.
 - Bastante revolucionario para la época:
 - Portable.
 - Multiusuario.
 - Multitarea.
 - Basado en dos ideas:
 - Todo es un fichero.
 - **KISS:** Keep It Simple, Stupid.



- Años 70:
 - Pocos ordenadores.
 - Muy caros, el negocio está en el hardware, software es un añadido.
 - Informática artesanal, se posee el código fuente de todo y se suele modificar.



- AT&T es la propietaria de ese código, pero lo distribuye a universidades, gobiernos y empresas, nacen:
 - Unix BSD: Berkeley Software Distribution
 - HP-UX: HP
 - AIX: IBM
 - IRIX: SGI
 - SunOS/Solaris: Sun
 - ...

- Años 80:
 - Nace el PC: ordenadores baratos.
 - El negocio pasa a la venta de software.
 - Licencias privadas.
 - Cláusulas de no divulgación.



- Años 80:
 - Richard Stallman (MIT) no está de acuerdo con esta pérdida de libertad y crea la Fundación del Software Libre (FSF).
 - Proyecto GNU:
 - Crear un Sistema Operativo libre.



- "Libertad 0"
 - Libertad para ejecutar el programa con cualquier propósito (privado, educativo, público, comercial, etc.).



- "Libertad 1"
 - Libertad para estudiar y modificar el programa (para lo cual es necesario poder acceder al código fuente).



- "Libertad 2"
 - Libertad para copiar el programa de manera que se pueda ayudar al vecino o a cualquiera.



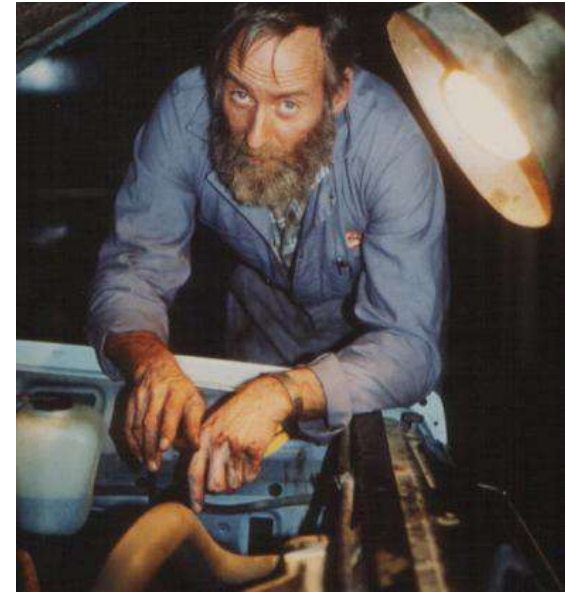
- "Libertad 3"
 - Libertad para mejorar el programa, y hacer públicas las mejoras, de forma que se beneficie toda la comunidad.



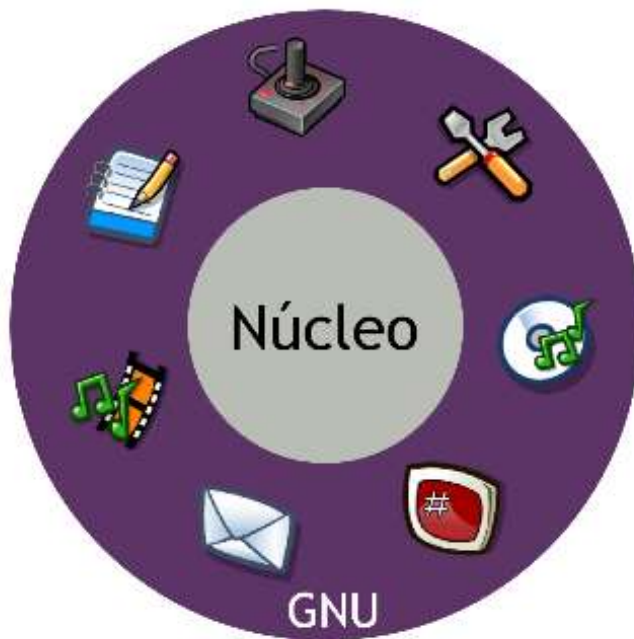
- Para la FSF el Software Libre es aquel que cumple 4 libertades:
 - Libertad 0: libertad de **uso** del programa, para cualquier propósito.
 - Libertad 1: libertad de estudiar **cómo funciona** un programa y poder adaptarlo.
 - Libertad 2: libertad de **distribuir y copiar libremente** un programa.
 - Libertad 3: libertad de mejorar el programa y hacer **públicas las mejoras**.



- Años 90:
 - El proyecto GNU está casi terminado. Le falta la pieza más importante, el núcleo.
 - Linus Torvalds desarrolla un núcleo, Linux, solamente para divertirse (*“just for fun”*).



- Años 90:
Proyecto GNU

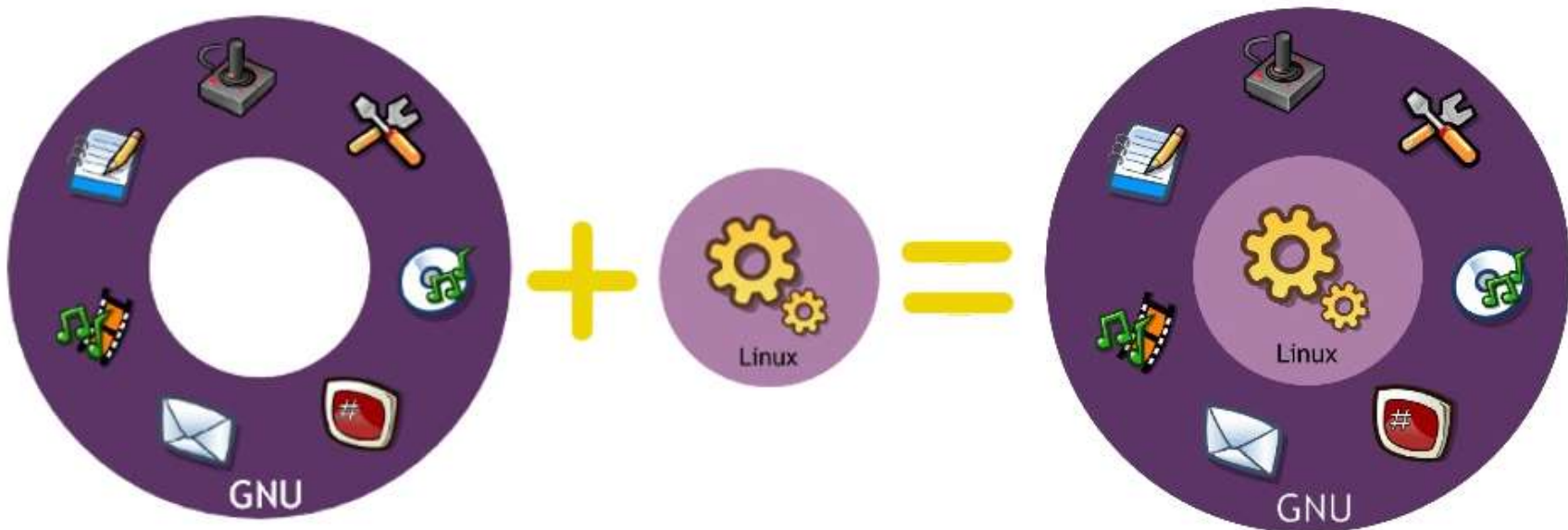


Proyecto Linux



- Años 90:

GNU/Linux

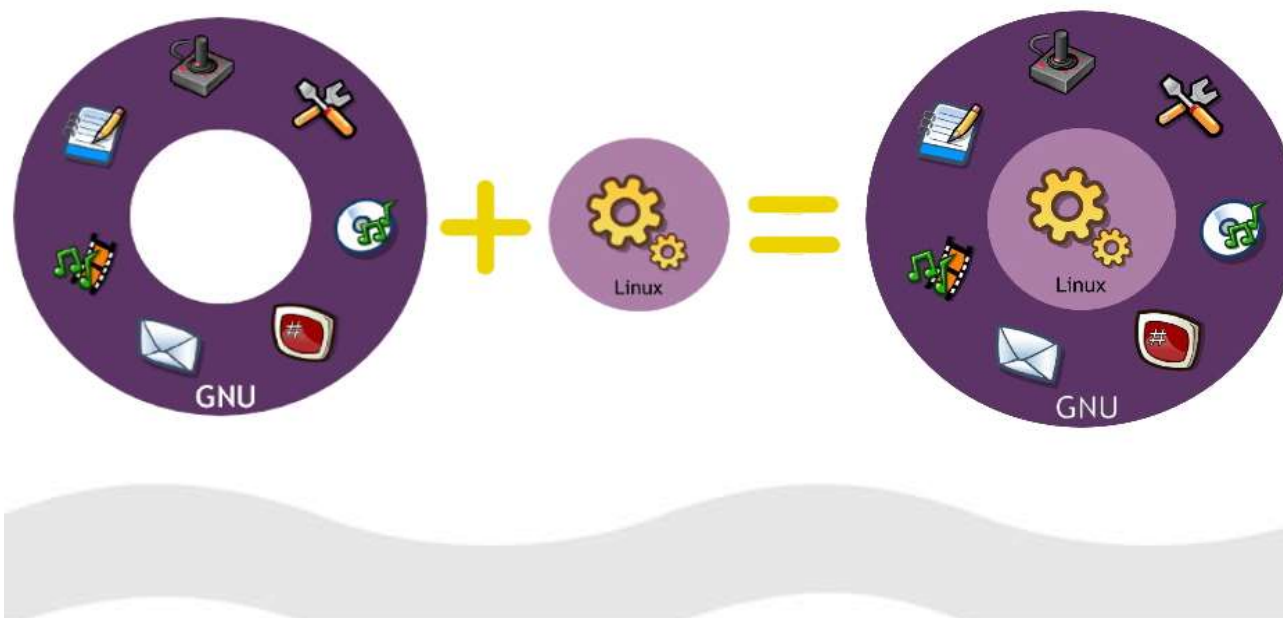


- Linux, el kernel de GNU/Linux:
 - Es un núcleo monolítico con soporte para módulos cargables (drivers).
 - Escrito en C y ensamblador.
 - Soporta:
 - Varios TB de RAM.
 - Varios cientos de procesadores.
 - Diferentes arquitecturas (x86, x64, **ARM**, etc.).

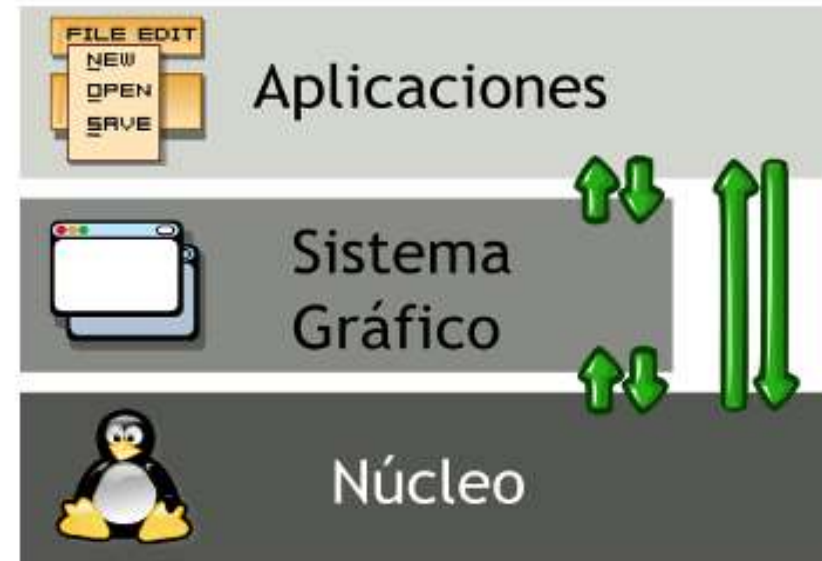


¿Qué es GNU/Linux?

- Es...
 - un Sistema Operativo.
 - la suma del Sistema GNU y el núcleo Linux.
 - compatible con sistemas UNIX.



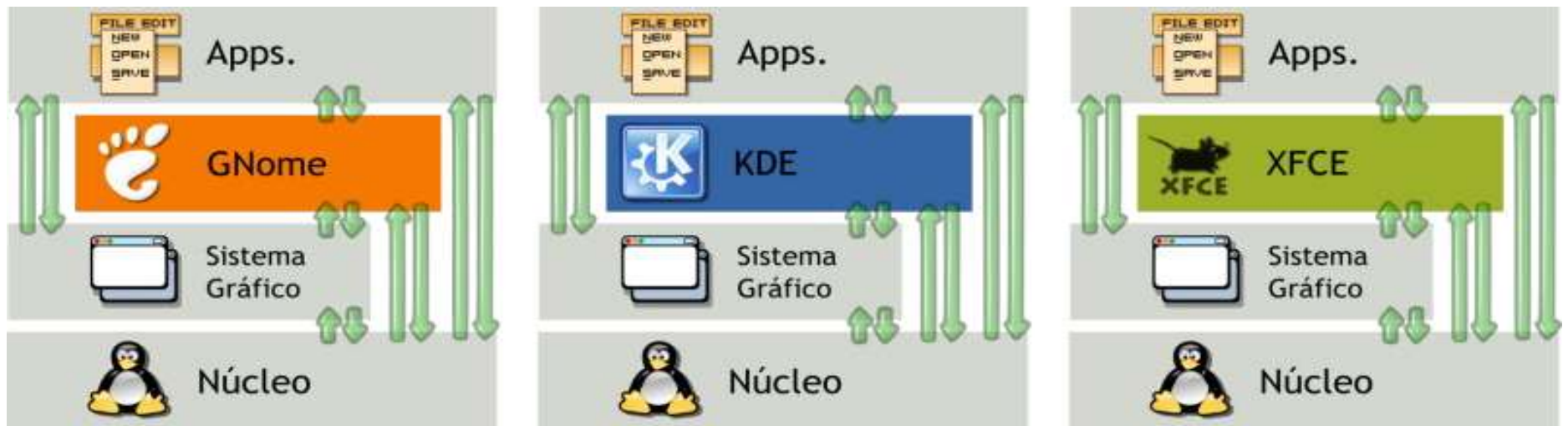
- Entorno gráfico en GNU/Linux



- Entorno gráfico en GNU/Linux

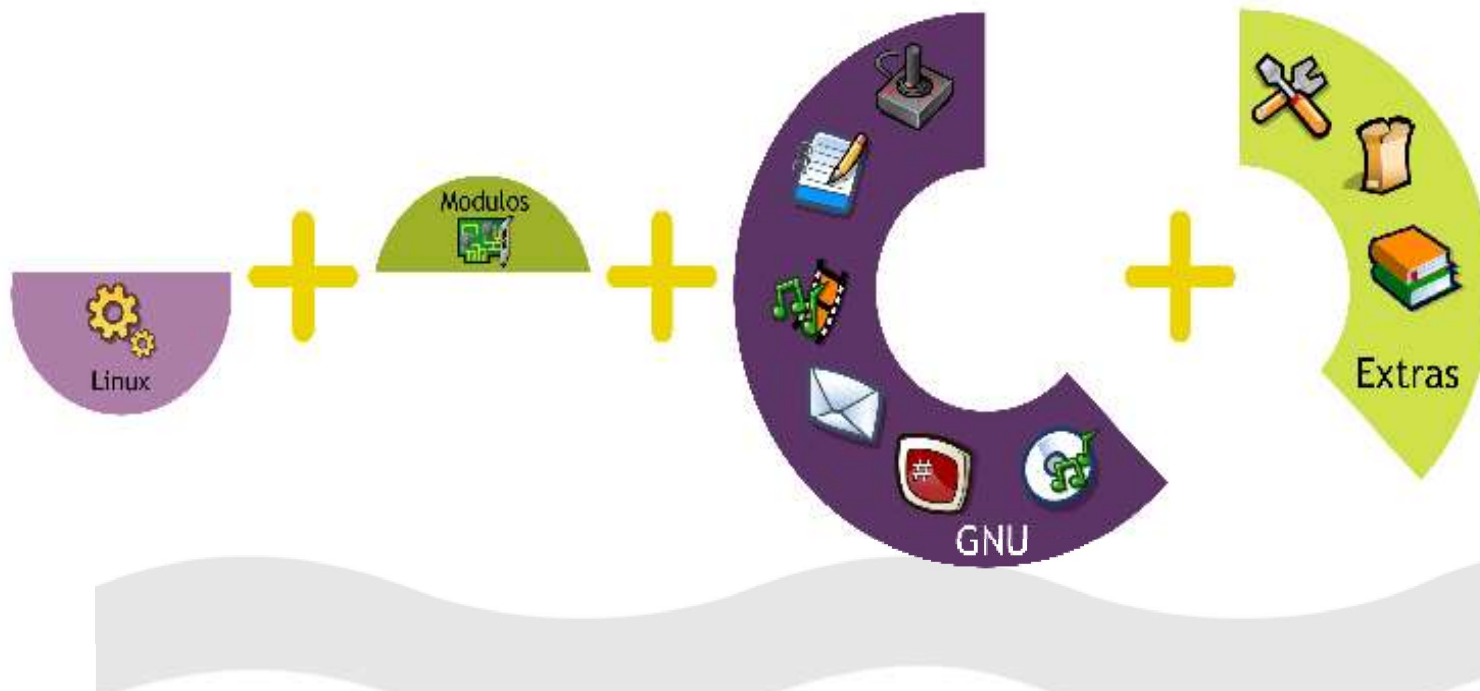


- Entorno gráfico en GNU/Linux

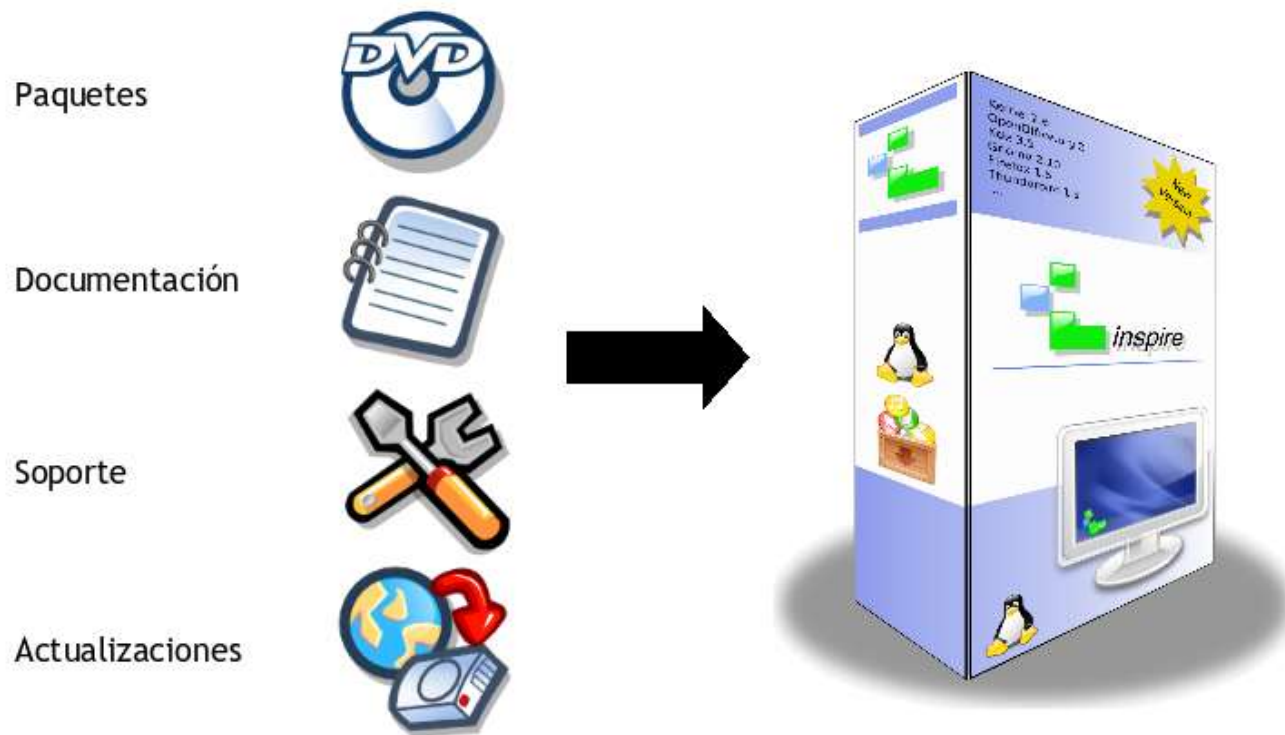


- Mitos del software libre:
 - “*Es inseguro*”.
 - “*Es poco profesional*”.
 - “*No tiene garantía*”.
 - “*Es feo*”.
 - “*No hay programas*”.
 - “*No hay drivers para mi hardware*”.
 - “*Usa formatos extraños*”.

- ¿Qué es una distribución de GNU/Linux?
 - Una “colección” de software libre:
 - Núcleo + Drivers + Utilidades + Programas + Extras + ¿Soporte? + ¿Discos? + ¿Documentación?



- ¿Qué es una distribución de GNU/Linux?



- Distribuciones más populares:



Debian



Mandriva



Ubuntu



Novell/SuSE



Linspire



Red Hat



Fedora Core

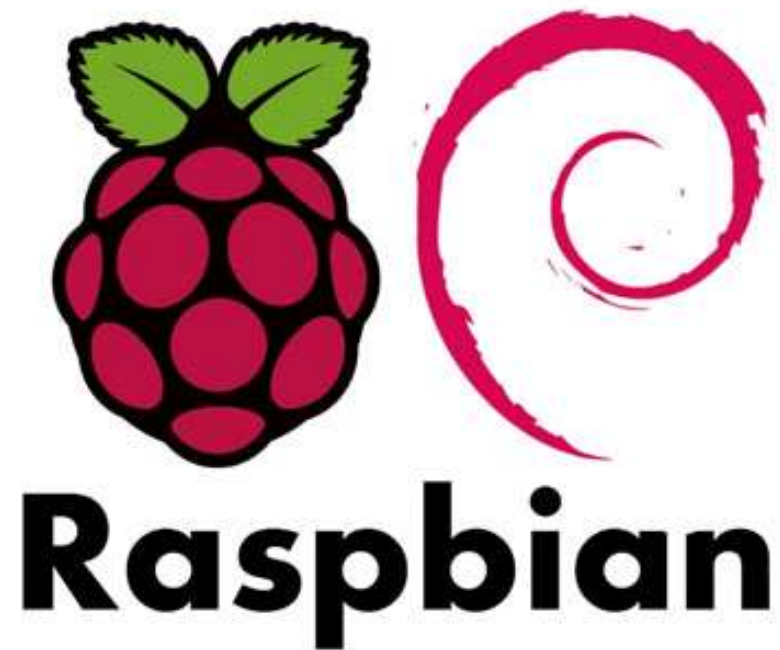


Knoppix



Gentoo

- Distribución que usaremos:





Powered by
LINUX

```
Kernel
+
| + eloha 3.1
+
Weather
+
| + Now: Sunny
Today: 13-21C
Tomorrow: 14-21C
+
Time
+
| + 09:05 20
+
System
+
| + CPU: 10%
| + RAM: 21%
| + Temp: 44C
| + /: 16%
| + /var: 15%
| + ~/: 37%
+
Net
+
| + Up: 08:00
| + Down: 21:00
+
Processes
+
| + NAME
| + pulseaudio
| + pavucontrol
| + plugin-container
| + mpd
| + firefox
+
Gmail
+
| Unread: 4
+
Music
+
| Annaes Okei
| Aventura
| The Curse
| 4/15/5:53
| Playing
```

Primeros pasos con Linux



- El árbol de directorios:
 - /bin: comandos básicos.
 - /boot: archivos de arranque y kernel.
 - /dev: ficheros especiales para dispositivos.
 - /etc: ficheros de configuración del sistema.
 - /usr: programas instalados.
 - /var: ficheros variables (logs, backups, etc.).
 - /home: directorios y ficheros de cada usuario.
 - /root: home de root.

- Sistema de Ficheros:
 - Algunos de estos directorios pueden estar contenidos en otro dispositivo (otro disco, un dispositivo USB, un DVD, etc.).
 - Para que eso sea posible es necesario “montar” ese dispositivo en una rama del árbol de directorios.
 - Montar un dispositivo es como podar una rama del árbol de directorios e injertar los directorios del dispositivo nuevo a partir de ese punto.



touch: crea fichero.

rm: borra fichero.

mkdir: crea directorio.

rmdir: borra directorio.

cp: copia ficheros.

mv: mueve ficheros.

ln: crea enlaces.

head: primeras líneas.

tail: últimas líneas.

wc: cuenta líneas.

ls: listar.

man: manual.

pwd: directorio actual.

cd: cambia de directorio.

echo: escribe texto.

cat: muestra fichero.

more: paginador.

file: muestra tipo de fichero.

Comandos de búsqueda:

- find: busca ficheros.
- grep: busca patrones en ficheros.
- sed: modifica patrones.
- awk: busca y modifica campos en ficheros.

Todos usan patrones, expresiones regulares.



find: búsquedas de ficheros.

Sintaxis:

find [path] opciones

Opciones típicas:

-name "nombre"

-type (d|f|l|s|...)

-maxdepth n

-mindepth n

-mtime n

-exec comando \; ({} indica el fichero)

grep: busca una cadena o patrón en ficheros y muestra las líneas que la contienen.

Sintaxis:

grep [opciones] patron fichero

Opciones:

- c: escribe sólo el número de ocurrencias encontradas.
- i: ignora mayúsculas y minúsculas.
- v: escribe líneas que NO contienen el patrón.
- 3: muestra 3 líneas antes y 3 después del patrón.

Patrones (regexp):

. (un carácter cualquiera)

* (cero o más ocurrencias del carácter anterior)

^ (principio de línea)

\$ (fin de línea)

[a-f] (cualquier carácter del rango)

[-AD] (caracteres '-', 'A', 'D')

[^56] (todos los caracteres excepto '5' y '6')

[5^6] (caracteres '5', '^', '6')

[[:alnum:]] (todos los caracteres alfanuméricos)



sed: editor de *streams* (flujos de datos).

Sintaxis:

sed -e “s/patron/reemplazo/g” fichero

Se emplea normalmente para sustituir palabras y frases, o para eliminarlas.

awk:

Awk es un lenguaje de programación utilizado para manipular texto.

Los datos se manipulan como palabras (campos) dentro de una línea (registro).

Un comando awk consiste en un patrón y una acción que comprende una o más sentencias.

Sintaxis:

```
awk 'patron { accion }' fichero
```

Propietarios y permisos:

chmod: modifica el modo de acceso de un fichero.

chown: modifica el propietario de un fichero.

chgrp: modifica el grupo propietario de un fichero.

who: muestra usuarios conectados al sistema.

whoami: muestra el nombre de usuario del usuario actual.

id: muestra las propiedades del usuario y grupo actuales.

su: cambio de usuario actual.



Gestión de procesos:

top: muestra información de procesador, procesos y memoria.

ps: muestra la lista de procesos que se están ejecutando.

pstree: ps en forma de árbol.

pgrep: ps + grep.

pidof: muestra el PID del proceso que solicitemos.

kill: envía una señal a un proceso (PID).

killall: envía una señal a un proceso (nombre).

