

# Introducción a GNU/Linux

Guillermo Rubilar

[Confeccionado usando parte del código fuente disponible en <https://github.com/giomba/beamer-intro-linux> y <https://github.com/alexpacheco/linux>]

25 de marzo de 2024

## 1 Un poco de historia

## 2 Linux hoy

- Distribuciones
- Componentes de GNU/Linux
- Archivos y Procesos
- Bash

# Software libre y GNU



Richard Stallman

# Software libre y GNU



Richard Stallman

## Software libre

- 0 Libertad de usar el programa para cualquier propósito
- 1 Libertad de estudiar el funcionamiento del programa
- 2 Libertad de poder modificar el programa
- 3 Libertad de poder distribuir el programa modificado

# Software libre y GNU



Richard Stallman

## Software libre

- 0 Libertad de usar el programa para cualquier propósito
- 1 Libertad de estudiar el funcionamiento del programa
- 2 Libertad de poder modificar el programa
- 3 Libertad de poder distribuir el programa modificado

1984



Nace **GNU**, sistema operativo completamente libre, basado en Unix

# Linus Torvalds y Linux



Linus Torvalds

## El problema




En la universidad, es un apasionado de los sistemas Unix

# Linus Torvalds y Linux



Linus Torvalds

## El problema




- I ♥ Unix En la universidad, es un apasionado de los sistemas Unix
-  Compra un PC i386

# Linus Torvalds y Linux



Linus Torvalds

## El problema

-  En la universidad, es un apasionado de los sistemas Unix
-  Compra un PC i386
-  Instala **Minix**-Unix en su PC







# Linus Torvalds y Linux



Linus Torvalds

## El problema





-  En la universidad, es un apasionado de los sistemas Unix
-  Compra un PC i386
-  Instala **Minix-Unix** en su PC
-  Imposibilidad de modificar libremente Minix

# Linus Torvalds y Linux



Linus Torvalds

## El problema

-  En la universidad, es un apasionado de los sistemas Unix
-  Compra un PC i386
-  Instala **Minix-Unix** en su PC
-  Imposibilidad de modificar libremente Minix

1991



Nace el **kernel Linux**

# El desarrollo de GNU / Linux



- 1984 – Nace el sistema operativo GNU
- 1991 – Nace el kernel Linux
- 1992 – El kernel de Linux se libera bajo la licencia GPL
- 1993 – Nacen Slackware y Debian
- 1994 – Nacen Suse y RedHat
- 2004 – Nace Ubuntu
- 2006 – Nace Linux Mint

# Las razones del éxito



GNU/Linux



Gratuito

# Las razones del éxito



GNU/Linux



Gratuito



Soporte multiprocesadores y multiplataforma (Linux en Marte: R1, R2)

# Las razones del éxito



GNU/Linux



Gratuito



Soporte multiprocesadores y multiplataforma (Linux en Marte: R1, R2)







Servidores Web (Apache)

# Las razones del éxito



GNU/Linux

-  Gratuito
-  Soporte multiprocesadores y multiplataforma (Linux en Marte: R1, R2)
-  Servidores Web (Apache)
-  Productos comerciales con hardware certificado

# Flexibilidad



Minicomputadoras



Teléfonos inteligentes



Modem, Router



# Distribuciones

- Una **distribución** es un conjunto particular de software que permite instalar, configurar y utilizar el kernel linux, los programas GNU, y software adicional.



Instalador



Utilidades



Navegador



Programas de  
Oficina

# Las distribuciones más famosas

- Existen *miles de distribuciones distintas* (ver [www.distrowatch.com](http://www.distrowatch.com)), que difieren en la selección de programas que incluyen y su configuración.



Ubuntu



LinuxMint



Fedora



Manjaro

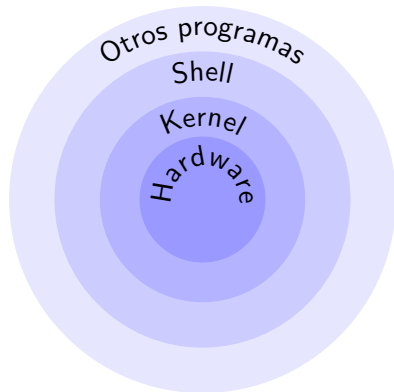


# Entornos Gráficos

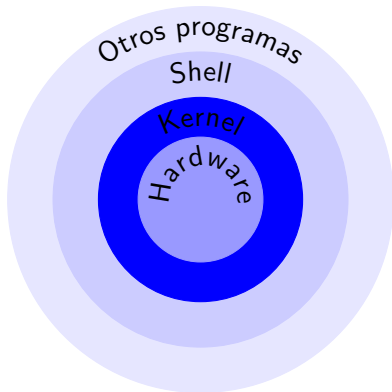
- KDE ([www.kde.org](http://www.kde.org))
- GNOME ([www.gnome.org](http://www.gnome.org))
- CINNAMON ([linuxmint.com](http://linuxmint.com))
- MATE ([www.mate-desktop.org](http://www.mate-desktop.org))
- XFCE ([www.xfce.org](http://www.xfce.org))
- FLUXBOX ([fluxbox.org](http://fluxbox.org))
- DEEPIN ([www.deepin.org/en/dde](http://www.deepin.org/en/dde))
- AWESOME ([awesomewm.org](http://awesomewm.org))
- Etc, etc, etc.

En general, cada entorno gráfico cuenta con sus propios programas para gestionar archivos, configurar el entorno, etc.

- Kernel
- Shell
- Aplicaciones/Programas



- Kernel
  - El kernel es el “núcleo” del sistema operativo.
  - Es el puente entre las aplicaciones y el procesamiento de datos realizado directamente en el hardware del sistema.
  - Una de las tareas principales del kernel es administrar los recursos del sistema.

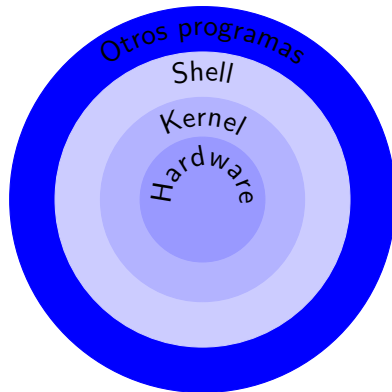


- Shell

- 
- Diagrama de las capas de un sistema operativo:
- Otros programas
  - Shell
  - Kernel
  - Hardware



- Aplicaciones/Programas
  - Linux suministra un gran número de aplicaciones útiles en el ámbito científico (y también para tareas cotidianas y recreativas).
  - La mayoría de los programas disponibles para Windows tienen su equivalente o similar en Linux.



# Archivos y Procesos

- En Linux/Unix *todo* es un archivo o bien un proceso.
- Un **archivo** es un conjunto de datos, creados por un usuario usando algún programa.
- Un **proceso** es un programa que se está ejecutando. Tiene asociado un código identificador único (PID).



# El sistema jerárquico de archivos



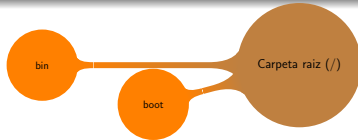
- Todos los archivos están ordenados en una estructura jerárquica.
- La parte más alta de la jerarquía es usualmente llamada *raíz* (root) (y simbolizada por un slash / )

# El sistema jerárquico de archivos



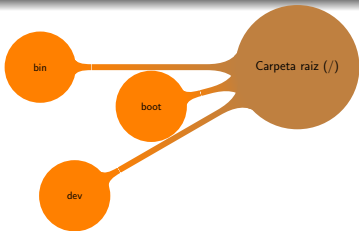
- **bin:** Contiene archivos esenciales para la operación del sistema, que pueden ser utilizados por todos los usuarios del sistema.

# El sistema jerárquico de archivos



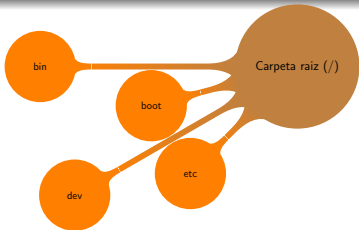
- **boot:** Contiene el kernel y los archivos necesarios para que el sistema pueda cargarlo al iniciar (bootloader).

# El sistema jerárquico de archivos



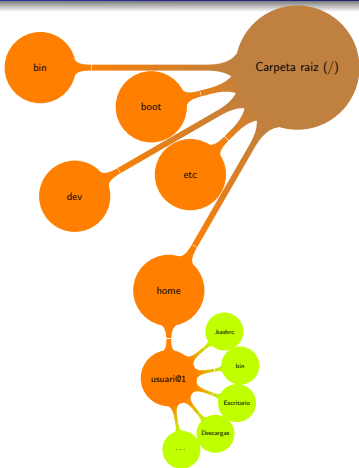
- **dev:** Contiene los distintos dispositivos conectados al sistema (discos duros, CD-ROMs, teclado, pantalla, etc.).

# El sistema jerárquico de archivos



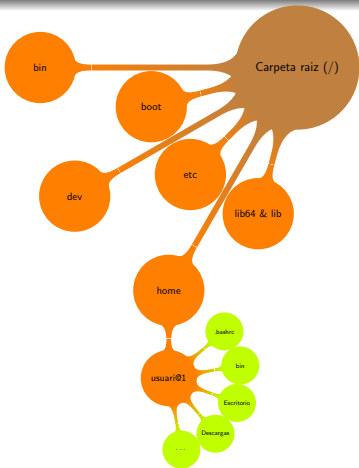
- **var:** Contiene archivos de configuración del sistema.

# El sistema jerárquico de archivos



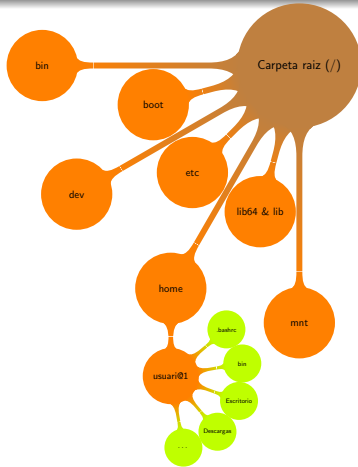
- **home:** Contiene las carpetas de cada usuari@. Es en esta carpeta donde cada usuari@ almacena sus archivos y desde donde se ejecutan inicialmente los comandos en la consola.

# El sistema jerárquico de archivos



- Contiene librerías esenciales para la operación del sistema, disponible para todos los usuarios.

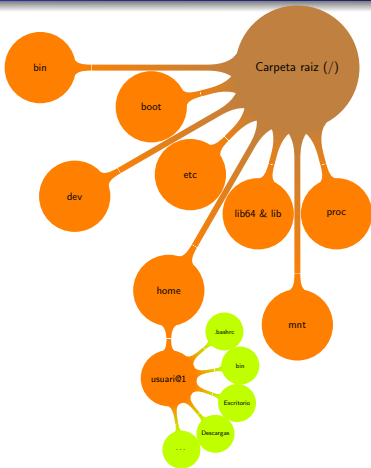
## El sistema jerárquico de archivos



- Carpetas donde son “montados” los distintos discos disponibles.

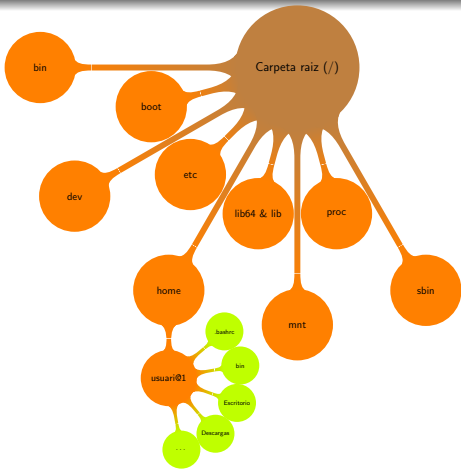


# El sistema jerárquico de archivos



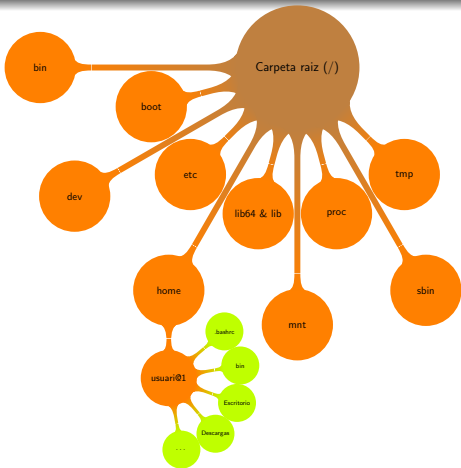
- Contiene pseudo-archivos que contiene información asociada a cada proceso en ejecución.
- Puede ser considerado como el centro de control e información para el kernel.

# El sistema jerárquico de archivos



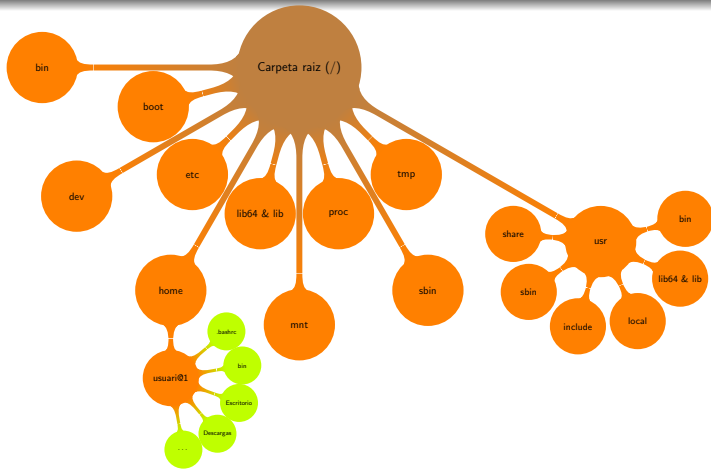
- Similar a **bin** pero sólo accesible por el superusuario **root**.

# El sistema jerárquico de archivos



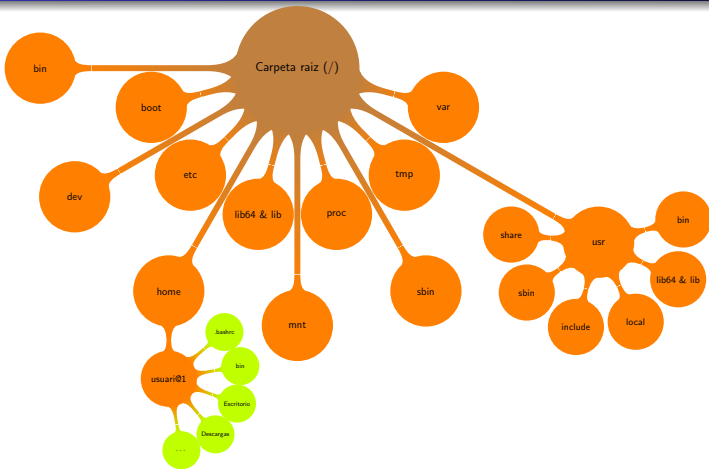
- Almacena archivos temporales.

# El sistema jerárquico de archivos



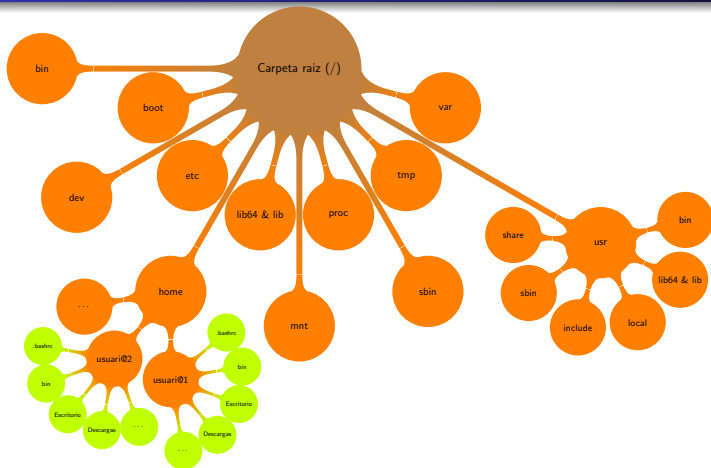
- Contiene documentación de los programas instalados, archivos binarios, librerías, etc.

# El sistema jerárquico de archivos



- Usado para almacenar archivos que cambian frecuentemente (a nivel de sistema, no de usuario).

# El sistema jerárquico de archivos



- Los sistemas tipo UNIX están diseñados para ser *multiusuarios*.
- Existe un usuario especial llamado **root**, el *administrador* del sistema. Puede acceder *todos* los archivos del sistema.

# Usando Linux

## Archivos y carpetas

- Las extensiones no son obligatorias (pero es útil usarlas).
- Se distingue los nombres de archivos entre mayúsculas y minúsculas.
- Los nombres de archivos pueden tener hasta de 256 caracteres.

# Bash y la consola de comandos

Mucho del poder y flexibilidad de Linux (Unix) radica en el uso de **comandos Bash**, ingresados en una **consola de comandos**.

Al ingresar comandos Bash en una consola (virtual) el sistema *interpreta y ejecuta* dichos comandos.

Algunos comandos Bash básicos son: `ls`, `cd`, `pwd`, `rm`, `mkdir`, `rmdir`, `cp`, `mv`, `rename`, `cat`, `more`, `man`, **etc.**



# Usando Linux

Estos comandos permiten el uso de caracteres **comodines** (\* y ?), y de caracteres de **redireccionamiento** de entrada/salida (>, < y |).

Para más detalles, ver sección 4 del [tutorial de GNU/Linux](#) de J. Smaldone.

## Más comandos útiles

tar, gzip, nano, ssh, history, !, locate, wc.