

Informática II

Introducción al sistema operativo GNU/Linux

Gonzalo F. Pérez Paina



Universidad Tecnológica Nacional
Facultad Regional Córdoba
UTN-FRC

– 2018 –

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)
 - ▶ Administración: tareas (scheduler), memoria, red, seguridad, disco.

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)
 - ▶ Administración: tareas (scheduler), memoria, red, seguridad, disco.
- ▶ ¿Cuales conocen?

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)
 - ▶ Administración: tareas (scheduler), memoria, red, seguridad, disco.
- ▶ ¿Cuales conocen?
 - ▶ Windows

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)
 - ▶ Administración: tareas (scheduler), memoria, red, seguridad, disco.

- ▶ ¿Cuales conocen?
 - ▶ Windows
 - ▶ Android

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)
 - ▶ Administración: tareas (scheduler), memoria, red, seguridad, disco.

- ▶ ¿Cuales conocen?
 - ▶ Windows
 - ▶ Android
 - ▶ Unix

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)
 - ▶ Administración: tareas (scheduler), memoria, red, seguridad, disco.

- ▶ ¿Cuales conocen?
 - ▶ Windows
 - ▶ Android
 - ▶ Unix
 - ▶ Mac OS

¹Berkeley Software Distribution, desarrollado desde 1970 (Univ. de California)

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)
 - ▶ Administración: tareas (scheduler), memoria, red, seguridad, disco.

- ▶ ¿Cuales conocen?
 - ▶ Windows
 - ▶ Android
 - ▶ Unix
 - ▶ Mac OS
 - ▶ BSD¹

¹Berkeley Software Distribution, desarrollado desde 1970 (Univ. de California)

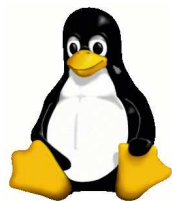
²GNU: "GNU's Not Unix (clon de Unix), bajo licencia GPL

Introducción

Sistemas operativos: historia y contexto

- ▶ ¿Qué es un sistema operativo?
 - ▶ Programa o conjunto de programas para administrar los recursos de hardware y dar servicios a los programas de aplicación (software)
 - ▶ Administración: tareas (scheduler), memoria, red, seguridad, disco.

- ▶ ¿Cuales conocen?
 - ▶ Windows
 - ▶ Android
 - ▶ Unix
 - ▶ Mac OS
 - ▶ BSD¹
 - ▶ GNU/Linux²

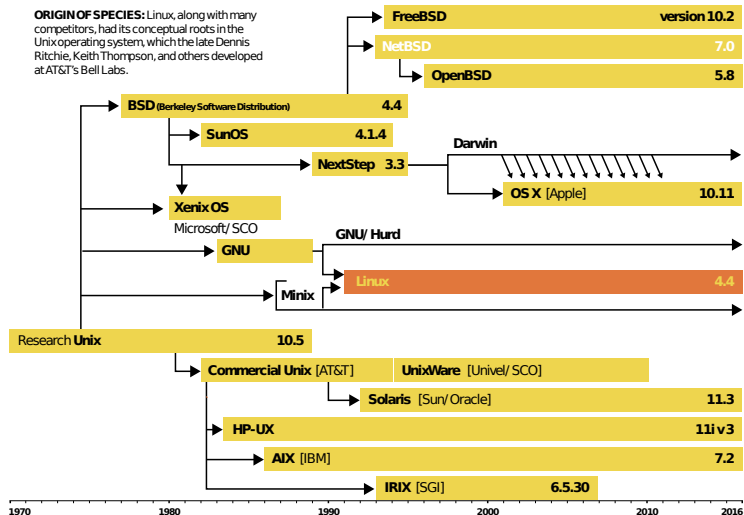


¹Berkeley Software Distribution, desarrollado desde 1970 (Univ. de California)

²GNU: "GNU's Not Unix (clon de Unix), bajo licencia GPL

Introducción

Sistemas operativos: historia y contexto



Introducción

GNU/Linux

Richard Stallman

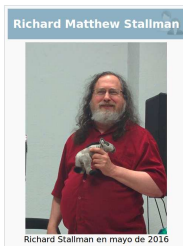
Richard Matthew Stallman (nacido en Manhattan, Nueva York, 16 de marzo de 1953), con frecuencia abreviado como «**rms**»,¹ es un programador estadounidense y fundador del movimiento por el software libre en el mundo.

Entre sus logros destacados como programador se incluye la realización del editor de texto GNU Emacs,² el compilador GCC,³ y el depurador GDB,⁴ bajo la rúbrica del Proyecto GNU. Sin embargo, es principalmente conocido por el establecimiento de un marco de referencia moral, político y legal para el movimiento del software libre, como una alternativa al desarrollo y distribución del software no libre o privativo. Es también inventor del concepto de copyleft (aunque no del término), un método para licenciar software de tal forma que su uso y modificación permanezcan siempre libres y queden en la comunidad de usuarios y desarrolladores.

Índice [ocultar]

1 Biografía

- 1.1 Primeros años
- 1.2 Laboratorio de Inteligencia artificial del MIT



Richard Stallman inició el proyecto GNU en enero de 1984

Linus Torvalds liberó el código del Kernel de Linux en agosto de 1991 (PC 386)

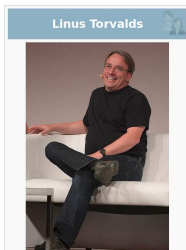
Linus Torvalds

Linus Benedict Torvalds (28 de diciembre de 1969, Helsinki, Finlandia) es un ingeniero de software finés americano,¹ conocido por iniciar y mantener el desarrollo del "kernel" (en español, núcleo) Linux, basándose en el sistema operativo libre Minix creado por Andrew S. Tanenbaum y en algunas herramientas, varias utilidades y los compiladores desarrollados por el proyecto GNU. Actualmente Torvalds es responsable de la coordinación del proyecto. Pertenecer a la comunidad sueco-parlante de Finlandia.

Sus padres tomaron su nombre de Linus Pauling (estadounidense, Premio Nobel de Química 1954). Comenzó sus andanzas informáticas a los 11 años cuando su abuelo, un matemático y estadístico de la Universidad, compró uno de los primeros microordenadores Commodore en 1980 y le pidió ayuda para usarlo.²

A finales de los años 80 tomó contacto con los ordenadores IBM, PC y en 1991 adquirió un ordenador con procesador modelo 80386 de Intel.

En 1988 fue admitido en la Universidad de Helsinki, donde estudio Ciencias de la Computación. Ese mismo año el profesor Andrew S. Tanenbaum saca a la luz el S.O. Minix con propósitos didácticos. Dos años después, en 1990, Torvalds



Introducción

GNU/Linux

- ▶ Linux 1.0 lanzado en 1994
- ▶ Desarrollado a partir de Minix (1987), usado por Linus (BSD no corría en PC, y GNU Hurd no estaba listo)
- ▶ Sigue el estándar POSIX (Portable Operating System Interface for Unix): estándar para sistemas operativo tipo Unix (IEEE-CS)
- ▶ Propiedades importantes de Linux: Estabilidad, Seguro, No necesita rebuteo frecuente, Portabilidad & Escalabilidad, etc.
- ▶ GNU (copyleft): Licenciado bajo GPL (General Public License)
- ▶ Distribuciones (Linux flavors)

Introducción

GNU/Linux

- ▶ Linux 1.0 lanzado en 1994
- ▶ Desarrollado a partir de Minix (1987), usado por Linus (BSD no corría en PC, y GNU Hurd no estaba listo)
- ▶ Sigue el estándar POSIX (Portable Operating System Interface for Unix): estándar para sistemas operativo tipo Unix (IEEE-CS)
- ▶ Propiedades importantes de Linux: Estabilidad, Seguro, No necesita rebuteo frecuente, Portabilidad & Escalabilidad, etc.
- ▶ GNU (copyleft): Licenciado bajo GPL (General Public License)
- ▶ Distribuciones (Linux flavors)

Memorable Linux Milestones

*From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: What would you like to see most in minix?
Summary: small poll for my new operating system
Message-ID: <1991Aug25.205708.9541@klaava.Helsinki.FI>
Date: 25 Aug 91 20:57:08 GMT
Organization: University of Helsinki*

Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them 😊

Linus (torvalds@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have 😞 .

Componentes

GNU/Linux

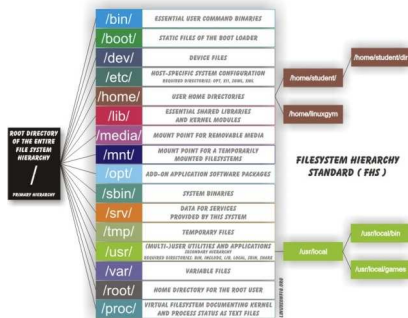
Componentes de un sistema GNU/Linux:

- ▶ El kernel/núcleo (Linux)
- ▶ El shell o intérprete de comandos (bash, ash, csh, etc.)
- ▶ Procesos y archivos/sistema de archivos (File System)
- ▶ Entorno de escritorio

Componentes GNU/Linux

Componentes de un sistema GNU/Linux:

- ▶ El kernel/núcleo (Linux)
- ▶ El shell o intérprete de comandos (bash, ash, csh, etc.)
- ▶ Procesos y archivos/sistema de archivos (File System)
- ▶ Entorno de escritorio



Software libre

GNU/Linux

- ▶ Software libre vs. privativo

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis
- ▶ RMS: en 1983 inicia oficialmente el proyecto GNU, en 1985 crea la Free Software Foundation, luego redacta la licencia GPL

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis
- ▶ RMS: en 1983 inicia oficialmente el proyecto GNU, en 1985 crea la Free Software Foundation, luego redacta la licencia GPL

Resumen GPL

- ▶ Cualquiera es libre de utilizar el Software Libre para cualquier propósito.

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis
- ▶ RMS: en 1983 inicia oficialmente el proyecto GNU, en 1985 crea la Free Software Foundation, luego redacta la licencia GPL

Resumen GPL

- ▶ Cualquiera es libre de utilizar el Software Libre para cualquier propósito.
- ▶ Cualquiera es libre de acceder a su código fuente y estudiarlo.

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis
- ▶ RMS: en 1983 inicia oficialmente el proyecto GNU, en 1985 crea la Free Software Foundation, luego redacta la licencia GPL

Resumen GPL

- ▶ Cualquiera es libre de utilizar el Software Libre para cualquier propósito.
- ▶ Cualquiera es libre de acceder a su código fuente y estudiarlo.
- ▶ Cualquiera es libre de distribuirlo.

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis
- ▶ RMS: en 1983 inicia oficialmente el proyecto GNU, en 1985 crea la Free Software Foundation, luego redacta la licencia GPL

Resumen GPL

- ▶ Cualquiera es libre de utilizar el Software Libre para cualquier propósito.
- ▶ Cualquiera es libre de acceder a su código fuente y estudiarlo.
- ▶ Cualquiera es libre de distribuirlo.
- ▶ Cualquiera es libre de mejorarlo o adaptarlo y de distribuir el programa modificado.

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis
- ▶ RMS: en 1983 inicia oficialmente el proyecto GNU, en 1985 crea la Free Software Foundation, luego redacta la licencia GPL

Resumen GPL

- ▶ Cualquiera es libre de utilizar el Software Libre para cualquier propósito.
- ▶ Cualquiera es libre de acceder a su código fuente y estudiarlo.
- ▶ Cualquiera es libre de distribuirlo.
- ▶ Cualquiera es libre de mejorarlo o adaptarlo y de distribuir el programa modificado.
- ▶ La única obligación es que si se distribuye, haya que hacerlo bajo la misma licencia GPL.

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis
- ▶ RMS: en 1983 inicia oficialmente el proyecto GNU, en 1985 crea la Free Software Foundation, luego redacta la licencia GPL

Resumen GPL

- ▶ Cualquiera es libre de utilizar el Software Libre para cualquier propósito.
 - ▶ Cualquiera es libre de acceder a su código fuente y estudiarlo.
 - ▶ Cualquiera es libre de distribuirlo.
 - ▶ Cualquiera es libre de mejorarlo o adaptarlo y de distribuir el programa modificado.
 - ▶ La única obligación es que si se distribuye, haya que hacerlo bajo la misma licencia GPL.
-
- ▶ **FOSS:** Free and Open Source Software

Software libre

GNU/Linux

- ▶ Software libre vs. privativo
- ▶ Software libre vs. gratis
- ▶ RMS: en 1983 inicia oficialmente el proyecto GNU, en 1985 crea la Free Software Foundation, luego redacta la licencia GPL

Resumen GPL

- ▶ Cualquiera es libre de utilizar el Software Libre para cualquier propósito.
 - ▶ Cualquiera es libre de acceder a su código fuente y estudiarlo.
 - ▶ Cualquiera es libre de distribuirlo.
 - ▶ Cualquiera es libre de mejorarlo o adaptarlo y de distribuir el programa modificado.
 - ▶ La única obligación es que si se distribuye, haya que hacerlo bajo la misma licencia GPL.
-
- ▶ **FOSS:** Free and Open Source Software
 - ▶ **OSHW:** Open Source Hardware

Software libre

Distribuciones – versiones/flavors



Software libre

Distribuciones – versiones/flavors



Linux Distribution Timeline

Software libre

Algunas aplicaciones

- ▶ Oficina: LibreOffice, L^AT_EX, LyX
- ▶ Edición imágenes, diagramas, etc.: Inkscape, GIMP, Blender
- ▶ Programación: GCC, GNU Compiler Collection
- ▶ IDEs de programación: Gedit, Geany, Code::Blocks, Eclipse CDT, Qt Creator, Vim, Emacs
- ▶ Diseño de circuitos (esquemático y PCB), simulación, etc.: KiCAD, QUCS
- ▶ Matemática: GNU Octave, Maxima (wxMaxima), Scilab, R, etc.
- ▶ Otros: Firefox, Iceweasel, Pidgin, Evince, Okular, VLC, Audacity, ...



Software libre

Entornos gráficos

- ▶ **Display Manager:** LightDM, GDM, KDM, LXDM, etc.
- ▶ **Window Manager:** Compiz, Metacity, Mutter, W9dk, fluxbox, etc.
- ▶ **Desktop Environment:** Gnome (GTK), KDE (Qt), LXDE (Lightweight X11 Desktop Environment), XFCE
- ▶ **Graphical User Interface (GUI)**

Software libre

Instalación y administración de software

- ▶ Partición de disco (gparted). Dual-boot
 - ▶ MBR¹: Master Boot Record (particiones primarias, extendidas y lógicas)
 - ▶ GPT²: GUID Partition Table (mas de 2TB y mayor # de particiones primarias)

¹Origen en IBM-PC 1980s, MS-DOS

²UEFI, Unified Extensible Firmware Interface

Software libre

Instalación y administración de software

- ▶ Partición de disco (gparted). Dual-boot
 - ▶ MBR¹: Master Boot Record (particiones primarias, extendidas y lógicas)
 - ▶ GPT²: GUID Partition Table (mas de 2TB y mayor # de particiones primarias)
- ▶ Instalación CD, USB (bootable), red, etc.

¹Origen en IBM-PC 1980s, MS-DOS

²UEFI, Unified Extensible Firmware Interface

Software libre

Instalación y administración de software

- ▶ Partición de disco (gparted). Dual-boot
 - ▶ MBR¹: Master Boot Record (particiones primarias, extendidas y lógicas)
 - ▶ GPT²: GUID Partition Table (mas de 2TB y mayor # de particiones primarias)
- ▶ Instalación CD, USB (bootable), red, etc.
- ▶ Sistema de gestión de paquetes (gestor de paquetes): colección de herramientas para automatizar el proceso de instalación, actualización, configuración y eliminación de paquetes de software.
 - ▶ Debian, **dpkg**: usa el formato `.deb`. Herramienta de resolución de dependencias **apt** (`apt-get update`, `apt-get install ...`)
 - ▶ Red Hat, **rpm**: RPM Package Manager
 - ▶ Suse, **YaST**: Yet another Setup Tool

¹Origen en IBM-PC 1980s, MS-DOS

²UEFI, Unified Extensible Firmware Interface

Software libre

Instalación y administración de software

- ▶ Partición de disco (gparted). Dual-boot
 - ▶ MBR¹: Master Boot Record (particiones primarias, extendidas y lógicas)
 - ▶ GPT²: GUID Partition Table (mas de 2TB y mayor # de particiones primarias)
- ▶ Instalación CD, USB (bootable), red, etc.
- ▶ Sistema de gestión de paquetes (gestor de paquetes): colección de herramientas para automatizar el proceso de instalación, actualización, configuración y eliminación de paquetes de software.
 - ▶ Debian, **dpkg**: usa el formato `.deb`. Herramienta de resolución de dependencias **apt** (`apt-get update`, `apt-get install ...`)
 - ▶ Red Hat, **rpm**: RPM Package Manager
 - ▶ Suse, **YaST**: Yet another Setup Tool
- ▶ Repositorio (mirrors)

¹Origen en IBM-PC 1980s, MS-DOS

²UEFI, Unified Extensible Firmware Interface