

# Conociendo el potencial de la comunidad

8/11/2024



# Whoami

- ◆ Apasionado del software libre
- ◆ Estudiante de la Universidad de Castilla-La Mancha
- ◆ Nix Advocate
- ◆ Fediverse: @amerinor01@mastodon.social
- ◆ Matrix: @amerino:matrix.org

# Historia del Open Source

En los años 70 y 80...



Todo por una impresora

# El crecimiento y la expansión



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:  
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 portable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-).

Judging from the post, 0.01 wasn't actually out yet, but it's close. I'd guess the first version went out in the middle of September -91. I got some responses to this (most by mail, which I haven't saved), and I even got a few mails asking to be beta-testers for linux. After that just a few general answers to questions on the net:



# ¿Qué es el Software Libre?

- **Definición:** Software que respeta la libertad de los usuarios.
- Permite:
  - **Usar** el software para cualquier propósito.
  - **Estudiar** cómo funciona y modificarlo.
  - **Distribuir copias** para ayudar a otros.
  - **Mejorar el software** y compartirlo.

# Software Libre vs. Software Privado

- **Software libre:**
  - Transparente, colaborativo y adaptable.
  - Menos costos de licencia.
  - Mayor seguridad por ser auditable.
- **Software privado:**
  - Restricciones de uso y modificaciones.
  - Dependencia del proveedor.
  - Menos control sobre tus datos.



# Casos de Software Libre vs. Privado

# GNU/Linux vs. MacOS/Windows

- **GNU/Linux:**
  - Flexible y personalizable.
  - Gran variedad de distribuciones.
- **MacOS/Windows:**
  - Sistemas populares, pero cerrados.
  - Menor control para el usuario.

# GIMP vs. Photoshop

- **GIMP:**
  - Alternativa libre para edición de imágenes.
  - Sin costos de licencia.
- **Photoshop:**
  - Software propietario con mayor popularidad.
  - Licencias de pago.

## FreeCAD/KiCad vs. AutoCAD/Altium

- **FreeCAD/KiCad:**
  - Diseño 3D y de PCB.
  - Acceso gratuito.
- **AutoCAD/Altium:**
  - Potentes, pero con licencias costosas.

# Herramientas de Desarrollo de Software Libre

## Editores de Código

- **Vim/Neovim:** Configurables y ligeros.
- **Emacs:** Plataforma extensible y flexible.
- **VSCode:** Popular y extensible (Code - OSS es la versión libre).

# Dónde Formarse en Software Libre



## Páginas Web

- [GNU.org](#): Filosofía del software libre.
- [Linux Foundation](#): Cursos sobre Linux.
- [Arch Wiki](#): BTW.
- [Github](#).

## Blogs

- [Opensource.com](#): Proyectos y noticias.
- [Planet Debian](#): Blog de desarrolladores de Debian.
- [Linux Adictos](#): Noticias Open Source en Español.

# Podcasts y Canales de YouTube

- **Podcasts:**

- *Linux Unplugged*: Noticias de Linux.
- *AtareaoConLinux*: Podcast en español.

- **YouTube:**

- *Michael Horn*: Tutoriales de software libre y análisis.
- *The Linux Experiment*: Comparativas y análisis de distros.

# Cómo Introducirse como Desarrollador de Software Libre

## Proceso Personal

1. **Escoge** una tecnología que te interese o uses en tu carrera.
2. **Explora repositorios** en GitHub o GitLab.
3. **Busca issues** con etiquetas como "Good First Issue".

# Oportunidades

- **Summer of Nix y Google Summer of Code:**
  - Programas de prácticas remuneradas.
  - Trabajas directamente con mentores y desarrolladores expertos.

# Conociendo el potencial de la comunidad

8/11/2024

