

Software Libre

Ing. Ernesto Pérez Estévez

<http://www.ecualinux.com>

Situación anterior al código abierto

- No se daba importancia al software
- Éste se distribuía junto con el hardware, su principal ingreso
- Las universidades tenían permisos para estudiar el código fuente

Surgimiento de Unix

- AT&T creó el Unix:
- Buena gestión de recursos
- Estabilidad
- Compatibilidad con HW de diferentes fabricantes
- Este último factor le proporcionó mucha popularidad

Comienzan los problemas

- A inicio de los 70 el software comenzó a ser visto como un factor de ingresos
- Las empresas comienzan a vender el software como valor agregado
- Las universidades dejan de recibir el código fuente que estudiaban
- Richard Stallman crea el concepto de GNU (**GNU** is **Not** **U**nix)

¿Qué es el software propietario?

- El autor cobra un precio por el uso del software (licencias)
- No se puede ceder ni alquilar el software en cuestión
- El único autorizado a realizar modificaciones al software es el desarrollador de éste. Siempre y cuando sea rentable. Cambio de plataforma, mejoras, customizaciones
- No se podrá realizar ingeniería inversa para estudiar su comportamiento.

¿Qué es el software libre?

- Libertad de los usuarios de copiar, rentar, ejecutar, distribuir, estudiar, cambiar y mejorar el software. Esto es:
- Libertad de usar el programa con cualquier propósito
- Libertad de estudiar cómo funciona y adaptarlo a sus necesidades
- Libertad de distribuir copias de éste
- Libertad de mejorarlo y hacerlo público

Ventajas del GNU

- Internacionalización, traducción
- Reutilización del código
- Reutilización de paquetes, aspell
- Rapidez del desarrollo
- Robustez: Pruebas extensivas y ciclo rápido
- Extensibilidad
- Soporte técnico: comunidad, pymes, grandes empresas, google
- Respaldado por la licencia GPL

Modelos de desarrollo

Tradicionalmente estudiamos un modelo orientado a la economía, costos, tiempos de desarrollo, modelos, horas hombre.

Sin embargo, GNU ha demostrado, mediante la práctica que el modelo de desarrollo que sigue es caótico y sin embargo funciona. Veamos:

Modelo tipo Catedral

Es el modelo tradicional de desarrollo de software

- Paso a paso, pequeños avances
- Se sigue el diseño de un arquitecto magistral
- Gran secreto
- Grandes recursos
- Los feligreses entran una vez terminada la obra

Modelo tipo Bazar

Es el modo más habitual en el software libre. A menudo considerado inviable por los expertos en análisis y diseño de sistemas. Pero funciona

- Gran número de desarrolladores
- Diferente lugar geográfico
- Voluntarios
- Diferente idioma
- No hay diseños escritos, sino problemas a resolver

Inicios de Linux

- 1987: Tannenbaum idea el Minix y publica su código. Sólo él acepta y realiza modificaciones.
- Fines docentes, bien documentado, fácil comprensión, modular, pedagógico
- 1991: Linus Torvalds decide crear Linux. Un unix para PC que todos pudieran usar.
- En sus inicios se identificó como un SO de hackers por su difícil comprensión y carencia de drivers y aplicaciones. No confiable, de juguete

¿Qué tiene Linux de interesante?

- Primer kernel libre (GNU)
- Apertura por parte del autor a cambios
- Amplia comunidad mundial que programa y contribuye a los cambios
- Estable, buen desempeño
- Multiplataforma (ARM, i386, Sun, Digital, motorola)
- Ayuda a bajar costos de operación

¿Qué es una distribución?

- Colecciones de paquetes GNU (mayormente)
- Fácil instalación y autodetección
- Fácil actualización
- Realizadas mayormente por empresas comerciales.
- Forma propia de nombrar paquetes
- Al final todas las distribuciones son basadas en el kernel linux y usan paquetes similares

Ejemplos de distribuciones

- Debian (No comercial, difícil de comprender)
- RedHat, Fedora, WhiteBoxLinux
- SUSE
- Mandrake
- Slackware
- www.ibiblio.org - Linux Distributions

Productos comerciales con Linux

- Server Appliances
 - Kypus Multifunction Security Appliance
 - Sun Cobalt RaQ 550
 - ServGate
 - PowerElf
 - NetWinder

Productos comerciales con Linux

- Thin Client
 - Kypus Thin Client
 - HP Thin Client Model T5515
 - Wyse Thin Client

¿Qué opina la competencia?

- No tiene soporte. ¿A quién pedirás ayuda? Principal fuente de ingresos de distros, diferentes esquemas de soporte y precios. Soporte local, google.
- Si algo se echa a perder no tienes a quién reclamar Google, millones de programadores en el mundo, cambios en casa
- Sigue un método desordenado de crecimiento Aunque casi cierto, ha demostrado validez
- Es difícil al usuario. Casi cierto, recomiendo para servidores.

Más opiniones

- Al estar disponible el código, se generarán muchas variantes. Precisamente se logra lo contrario, todos aportan a un mismo proyecto. No tiene sentido repetir lo mismo en otro lugar
- Si no conozco suficiente de programación, de qué me sirve el código? Otros sí conocen y lo cambiarán para su bien y el tuyo.
- Al estar disponible el código, los intrusos pueden estudiar sus fallas y explotarlas. También los whitehat ayudarán a detectarlas.

Algunas instituciones con Linux

- Andinanet, ETAPA, OnNet, Easynet
- Ejército
- MEF
- Espoch, UNITA, UTA, Espe, EPN

Ejemplos de paquetes GNU

- OpenOffice: www.openoffice.org - win32, linux
- Gaim: gaim.sourceforge.net – win32, linux
- Apache: httpd.apache.org – win32, linux, sun, alpha, MacOS, etc
- Named - www.isc.org - win32, linux
- PHP - www.php.net - win32, linux
- Snort - www.snort.org - win32, unix

Enlaces de interés

- <http://www.fsf.org/philosophy/philosophy.es.html>
- Manifiesto de Richard Stallman, creador del movimiento GNU:
<http://www.fsf.org/gnu/manifiesto.es.html>
- Cursos, conferencias, artículos sobre software libre en español: <http://es.tldp.org/>