

LABOEXPRESS
Puppy Linux

EVENTOS
1er Encuentro Trimestral
"MiSoL Técnico".

EVENTOS
Software Freedom Day 2009
C.A.B.A

EVENTOS
Red Hat presentó
Red Hat Enterprise Linux 5.4
en su Summit 2009.

OPINIÓN
La incomprendida Libertad número 3

LABOEXPRESS
XvideoServiceThief (xVST)
Descarga vídeos de 70 sitios diferentes

DISTRIBUCIONES GNU/LINUX
Lin-X 1.1 y Linux Mint 7 'Gloria' XFCE

IT
Soluciones Service Manager - ITIL Compliance

DESARROLLO WEB
NVU, un excelente software
de diseño Simple "en extinción".

TUXINFO 21

PACK TUXINFO 21
¿cómo obtenerlo?

PROGRAMACIÓN
PHP + MySQL
Parte 2

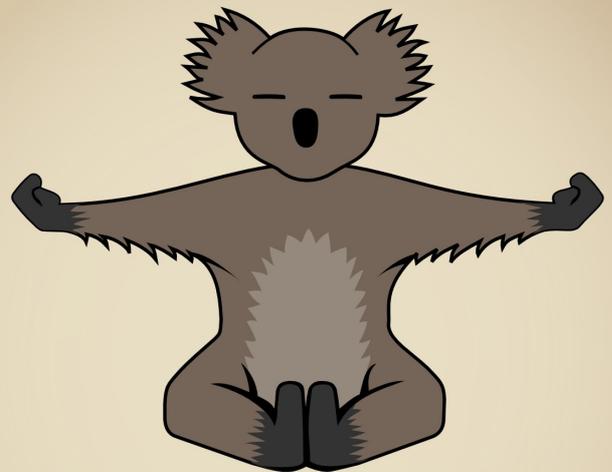
PROGRAMACIÓN
Curso de Python
Programación estructurada

coming soon

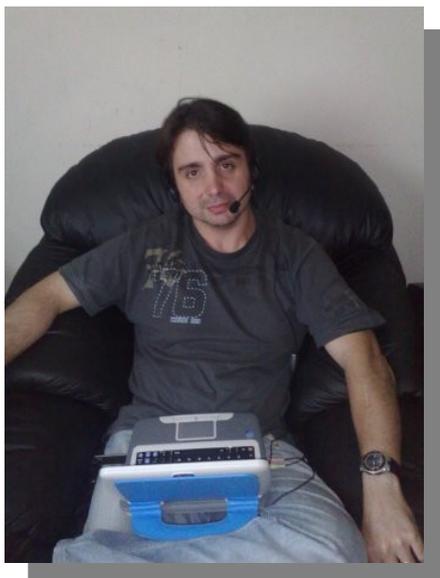
 **ubuntu 9.10**



**Probando
Ubuntu 9.10
Karmic Koala Beta
a fondo**



ubuntu has a good karma



EDITORIAL

Este mes como ya los tenemos acostumbrados vamos a tratar varios temas de la más preciada relevancia en el mundo del software libre. Como nota de tapa tenemos un excelente adelanto de lo que se viene este mismo mes con la versión 9.10 de Ubuntu Karmic. Luego tenemos la segunda clase del curso de programación basado en PHP+MySQL, también seguimos con el curso de Python cuyo tema es "Programación estructurada" y además tenemos un compendio de reviews de distribuciones GNU/Linux. Como así también los temas clásicos que mes a mes podrán encontrar en TuxInfo. Los invito a recorrer la revista y recomendarla ya que esa es nuestra mejor recompensa.

Esperamos sus comentarios, sus propuestas de los temas que desean que incluyamos en los próximos números a nuestra casilla de mail (info@tuxinfo.com.ar).

Saludos a todos y recuerden pasar la Voz!!!.

Ariel M. Corgatelli

colaboradores

Ariel Corgatelli (director, editor y coordinador)

Claudia A. Juri (marketing, edición y ventas)

Oscar Reckziegel (el corrector)

María Alicia Viana (Tapa TuxInfo 21 y asesoramiento gráfico)

Claudio de Brasi

David J. Casco

Emiliano Piscitelli

Ernesto Vázquez

Ezequiel Vera

Franco Rivero

Guillermo Movia

Gustavo A. Papasergio

Gustavo Tell

Hernán Claudio Saltiel

Juan Manuel Abrigo

Marcelo Guazzardo

Mario Colque

Matías Gutiérrez Reto

Nicolás Alejandro Guallan

Olemis Lang

Pablo Mileti

Pablo Terradillos

Reynier Pérez Mira

Rodney Rodríguez López

Samuel José Rocha Martos

Samuel Morales Cambrón

Sebastian Osterc

Victor Hugo Garcia

diseño

Ariel Corgatelli y María Alicia Viana
info@tuxinfo.com.ar

contáctenos

info@tuxinfo.com.ar

- Pág. 2 - Editorial.*
- Pág. 3 - Índice.*
- Pág. 4- Pack TuxInfo 21. ¿Cómo obtenerlo?.*
- Pág. 5 - Noticias del mes.*
- Pág. 9 - Eventos - 1er Encuentro Trimestral "MiSoL Técnico".*
- Pág. 11 - Eventos - Software Freedom Day 2009 – C.A.B.A.*
- Pág. 18 - Páginas recomendadas de Tuxinfo.*
- Pág. 21 - Desarrollo web - NVU, un excelente software de diseño Simple "en extinción".*
- Pág. 25 - Servidores - Bacula: Software para salvadas automáticas II Parte.*
- Pág. 30 - Nota de Tapa - Probando Ubuntu 9.10 Karmic Koala Beta a Fondo.*
- Pág. 40 - Opinión - La incomprendida Libertad número 3.*
- Pág. 42 - IT - Soluciones Service Manager – ITIL Compliance.*
- Pág. 45 - Programación - Curso de Python. Programación estructurada.*
- Pág. 50 - Rincón del Lector.*
- Pág. 52 - Eventos - Red Hat presentó Red Hat Enterprise Linux 5.4 en su Summit 2009.*
- Pág. 57 - Opensolaris - El día que los pingüinos se asolearon.*
- Pág. 65 - Laboexpress - PUPPY LINUX.*
- Pág. 68 - Programación - PHP + MySQL parte 2.*
- Pág. 74 - Distribuciones GNU/Linux - Lin-X 1.1 y Linux Mint 7 "Gloria" XFCE.*

TUX **OPEN**
WWW.TUXINFO.COM.AR

TuxInfo Número 20, recibió la colaboración económica desde Paypal/Dineromail de las siguientes personas

Ruben Perez; Alfonso González; Matías Cimas; Omar Sanchez; Horacio Lena; Fernando Herrera Arciga; Leonel Burgos; Luis Alberto D'Ardis; Carlos Correa.

Donaciones sin tarjeta de crédito

Donaciones con tarjeta de crédito

PACK TUXINFO 21 ¿CÓMO OBTENERLO?

El pack se envía por correo postal certificado bajo firma del receptor en donde se garantiza la recepción del mismo (sólo Argentina).

La forma de pago desde Argentina es mediante las oficinas de pago; Rapi-pago, Pago Fácil y Bapro con un costo final de \$ 38 (ARS). Para los residentes fuera de la República Argentina comenzamos a realizar los envíos, el cual tiene un costo (en dólares) de 28 USD con envío incluido (pago mediante Paypal).

IMPORTANTE: Formas de pago y Enlaces de Pago

Desde los siguientes enlaces encontrarán los URL directos para el pago del pack, sólo debe elegir la opción correcta, si reside dentro de la República Argentina, fuera de ella, si opta por una suscripción de 6 meses (ahorra el pago de un pack), o 12 meses (ahorra el pago de 2 packs). Así, se puede optar por una suscripción de 6 meses (con lo cual se abonaría 5 packs) o bien anual abonando el valor de 10 pack.



SOLICITAR Y ABONAR PACK 21 TUXINFO (desde Argentina mediante Pago Fácil, Rapipago, Bapro).

SOLICITAR Y ABONAR PACK 21 TUXINFO (fuera de Argentina, mediante Paypal)

PACK 6 MESES (desde Argentina mediante Pago Fácil, Rapipago, Bapro)

PACK 6 MESES (fuera de Argentina, mediante Paypal)

PACK 12 MESES (desde Argentina mediante Pago Fácil, Rapipago, Bapro)

PACK 12 MESES (fuera de Argentina, mediante Paypal)

CONTENIDO COMPLETO DEL PACK TUXINFO 21 UBUNTU - KUBUNTU - VERSIONES DE 32 BITS Y 64 BITS - DESKTOP, ALTERNATE, CD Y DVD

Más información, detalle de los tres DVD que compone el pack 21, costos y medios de pago.

Consultas info@tuxinfo.com.ar

Oracle y SUN: MySQL al parecer seguirá siendo libre

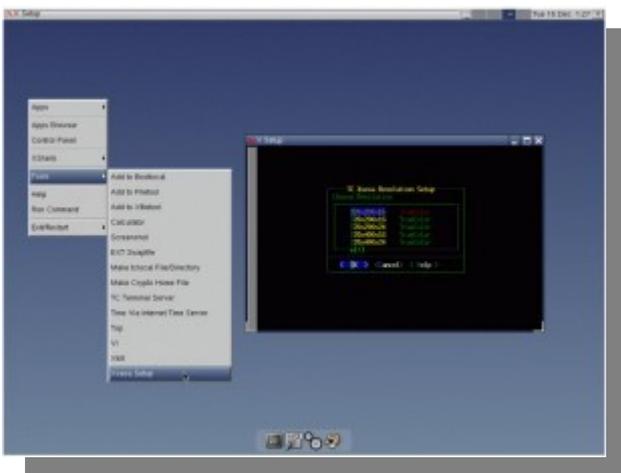
Scott McNealy co-fundador de SUN Microsystems y Tony Stark Larry Ellison, su par en Oracle, se subieron al escenario, para dar inicio a una de las semanas más movidas de San Francisco, la del Oracle Open World, que por primera vez tiene a Oracle y SUN juntos a vista de una fusión inminente.



McNealy, vino a calmar a su gente, y dejar en claro que SUN Microsystems seguirá con su estrategia y que sus proyectos más importantes no sólo seguirán intactos (incluido MySQL), sino que además se invertirá más dinero. En su presentación, enumeró las 9 innovaciones más grandes de SUN:

- NFS / PC-NFS
- Arquitectura SPARC (la primera estación de 7-10 MIPS bajo 10 mil dolares)
- Software de código abierto
- Java (Java card, Java SE/ME/EE y JavaFX)
- SUN E10K (Solaris corriendo sobre 64-way)
- ZFS/ Open Storage / Flash (Exadata)
- Proyect Blackbox
- Sun Ray
- Chips multi-hilos (coolthreads)

La segunda parte consistió en Ellison hablando de las sinergias entre el software Oracle y el hardware de SUN, afirmando en más de una ocasión, que la combinación SPARC más Oracle es la más rápida del planeta.



Tiny Core: la distribución linux se actualiza a la versión 2.4.1 y ocupa tan solo 11 MB

Esta nueva distribución de Linux llamada Tiny Core, que está en la versión 2.4.1, ha sido diseñada para ser una distro de escritorio liviana que nos permitirá instalar aplicaciones de su elección. Funciona como un sistema operativo en la nube después de que se cargue en la memoria RAM de nuestra PC, y solo pesa 11 MB.

Tiny Core 2.4.1 tiene hasta cuatro formas de funcionamiento, el primero ya lo hemos dicho, en la nube el cual es el que viene por defecto. En cuanto al segundo es PPR/TCE, con este tendremos que especificar la partición de almacenamiento y puede usar el repositorio de software TCE. Por otro lado tenemos al PPR/TCZ el cual es muy parecido al anterior, pero este utiliza extensión TCZ. Por último tenemos a PPI/TCE la cual instala extensiones en una partición Linux o en un archivo Loopback.

Descargas: [Tiny Core 2.4.1](#)

Ya se encuentra disponible para linux: Hulu Desktop

El sitio de streaming de programas de televisión más popular de los Estados Unidos acaba de lanzar su aplicación de escritorio para Linux, concretamente para Fedora y Ubuntu, pero se espera que en breve lo esté también para el resto de distribuciones Linux.

Hulu Desktop nos permite ver un streaming de video clips, programas de televisión, series y películas en la pantalla de nuestra Pc, y lo mejor de todo a "full screen", ya que no tendrás que estar viendo el sitio desde el navegador web, además si posees un control remoto infrarrojo es posible que Hulu trabaje con él.

Es extraño que Hulu lanzara un cliente para sistemas operativos libres, pues con Boxee es posible ver el contenido del sitio; aun así, es de agradecerse que por fin hayan lanzado esta aplicación. Hulu Desktop para Linux es totalmente gratis.

Para ejecutarlo necesitás tener instaladas las últimas versiones de Ubuntu (9.04) y Fedora (11), Adobe Flash y las siguientes bibliotecas:

-GTK+ 2.12 o superior.

-GLib 2.16 o superior.

-LIRC 0.8.2 o superior (requerido para funcionar con control remoto).

Descarga: Hulu Desktop



Paraguay se suma al mundo del software libre

Paraguay será otro de los países de América Latina que adoptará el modelo de software que con treinta soluciones libres, impulsó el gobierno brasileño en 2007 para administraciones, empresas y el sector educativo.

Esta iniciativa del gobierno paraguayo se integra con el proyecto Software Público Internacional coordinado con Naciones Unidas presentado en febrero y que ha sido difundido en Uruguay, Venezuela y Argentina.

La adopción de este tipo de software se motiva en la independencia tecnológica y seguridad que el proyecto proporcionará al gobierno de Paraguay, indicaron sus responsables. "Vamos a fortalecer la economía local y la capacidad de evolucionar las aplicaciones sin las restricciones de las licencias de software propietario", explicaron.

El lanzamiento del portal de software público de Paraguay se realizará el 22 de octubre en la quinta conferencia de Latinoware.

<http://www.softwarepublico.gov.br/news-item54>

Acelera Firefox con: SpeedyFox

Speedyfox es un programa portátil que te hará la vida más sencilla, por el momento sólo funciona en Windows (Vista, XP, 2000), pero pronto saldrá para Mac, y que con sólo hacer un click, optimizará y mejorará la velocidad de Firefox a niveles nunca antes imaginados, a su vez reorganizará y mejorará la distribución de las bases de datos, así que no esperen más, adquieran su SpeedyFox gratuitamente en el siguiente enlace:

Descargar: Speedyfox





Flickroom, ya disponible

Flickroom es una aplicación que te permite interactuar con el servicio de intercambio de fotos Flickr, todo esto sin la necesidad de abrir tu navegador web, así vas a poder tener control total de tu cuenta de Flickr.

Ingresando tu cuenta de Flickr podés acceder al programa, y así ver las fotos de tu galería o bien las de todos tus contactos. A su vez, esta herramienta posee un buscador con el cual poder buscar imágenes en toda la comunidad de Flickr, subir tus propias imágenes, guardar imágenes en Flickr, agregar comentarios, notas o simplemente compartir una imagen a través de Twitter.

Flickroom también cuenta con una sala de chat, en ella es posible comunicarse en tiempo real con otros usuarios de la aplicación.

Flickroom está basado en Adobe AIR, lo que significa que está disponible para Mac, Windows y Linux.

<http://www.flickroom.org/beta/>

Culebrón en el software libre: Richard Stallman acusa a Miguel de Icaza como traidor, Miguel le contesta y para sumar Linus habla de Linux

Esta semana podemos decir que estuvo completa la semana con problemas en el mundo del software libre. Primero Linus Torvalds habló sobre su creación el núcleo Linux.

Donde expreso que el núcleo “estaba inflado y era enorme”. Una situación lógica ante los millones de líneas agregadas al código fuente en los últimos años, siendo un ejercicio de autocrítica ya que Torvalds es responsable del kernel, algo que ha iniciado un amplio debate en Internet.

Seguido del culebrón entre Stallman e Icaza:

Donde Richard en el Software Freedom Day de este año en la ciudad de Boston (EE.UU.) respondió una muy interesante pregunta donde indudablemente se destaca su afirmación de que Miguel de Icaza “es básicamente un traidor a la comunidad del Software Libre”.

La respuesta se basa en el rumor de que Miguel estaría por ocupar el lugar de Sam Ramji con su puesto de director del Laboratorio Open Source de Microsoft.

La respuesta de Miguel de Icaza no se hizo esperar

“Quiero decir que Dios ama a todas las criaturas. Desde el formidable elefante hasta la más pequeña hormiga. Y eso incluye a Richard Stallman.

En cuanto a mí, creo que hay un mundo de posibilidades, y si Richard quiere discutir cómo podemos mejorar la mancomunidad del Open Source/Software Libre en el mundo, él tiene mi dirección de correo electrónico. Con amor, Miguel”.

Haga crecer su negocio con un *Sitio Web*

“ **Dattatec.com** me brinda, en un sólo producto, todo lo necesario para tener una presencia efectiva en internet. ”

Pablo Barrios (<http://www.e-veo.com>)



Registro de Dominio

Identifique su sitio web de manera que sus visitantes puedan recordarlo fácilmente a través de un dominio .com, protegiendo además su marca y otorgándole una imagen profesional.



Sitio web & E-mail

Construya un sitio web de calidad profesional Usted mismo y en sólo cinco pasos.

Elija entre más de 100 diseños profesionales y modifíquelo cada vez que lo necesite.



Su sitio web en Google

Con la contratación del servicio, Dattatec.com le regala un cupón por valor de u\$s 70 en Crédito de Google® AdWords® para que promocióne su sitio en internet y obtenga visitas calificadas y efectivas.



Sistema de Gestión de la Calidad
Certificado bajo Normas ISO 9001:2000
en todos los procesos de la compañía.
Certificado en los siguientes países: Argentina,
Brasil, Chile, España, México y Venezuela.

Comience hoy mismo..!

www.tengasusitio.com



Dattatec.com es la primer Entidad Registrante Acreditada por ICANN en Latinoamérica.

El Sábado 26 de septiembre se llevó a cabo el 1er Encuentro Trimestral "MiSoL Técnico" en la ciudad de Posadas Misiones.

Este encuentro fué organizado por una joven comunidad de usuarios de software libre "MiSoL - Misiones Software Libre", con la colaboración de Software Libre Dachary. El objeto de "MiSoL Técnico" fue presentar las ventajas técnicas que ofrece el software libre en diferentes ámbitos, así como servir de marco de encuentro entre profesionales, empresarios y hobbystas de la provincia de Misiones y provincias y países limítrofes. La convocatoria está abierta a los interesados en aprender, debatir, compartir conocimiento y experiencias, conocer otros usuarios y divertirse.



Las conferencias fueron de diferentes niveles desde carácter introductorio a avanzado, el público se mostró muy interesado y participativo gracias al buen nivel técnico que demostraron los disertantes, las conferencias iniciaron a las 8AM y finalizaron a las 8PM. El modelo de evento trimestral es muy interesante, la premisa es fomentar la participación de todos los interesados en las temáticas de software libre y una de las condiciones es no repetir ninguna de las temáticas expuestas en el marco de estos eventos.

Las conferencias fueron las siguientes:

"Migración de Windows a GNU/Linux" por Pedro José Pezzarini.

"Ecosistema para industrias Tics en Misiones" por Carlos Gonella.

"Supermercado migrado a GNU/Linux" por Marcos Henning.

"Informatización de la Biblioteca Pública De Las Misiones con Software Libre" por Diego Sanchez Cassol

"Python sockets para gurises!" por José María Guaimas.

"Load & Performance Testing de una aplicación web" por Sergio Vier.

"SFyCSOL (Sistema de Facturación y Control de Stock On Line)" por Carlos Benitez.

"Codelgniter Framework: Estructura modular y optimización de vistas" por Sebastián Paniagua.

"Django: El framework web para perfeccionistas con deadlines" por Leandro Tocalini.

El evento es libre y gratuito, la 2da edición se planea realizarla en Marzo 2010...puedes invitar a tus amigos.



Los esperamos!!! + info en www.misol.org.ar (Misiones Software Libre) o info@misol.org.ar



Los que piensan que un evento sale bien cuando hay mucho tiempo de planificación, y muchísimas manos trabajando al unísono para que se haga realidad, tienen que conocer las experiencias que hemos tenido, en el AOSUG, al organizar un evento como lo fue el Software Freedom Day 2009 de la Ciudad Autónoma de Buenos Aires.

Todo comenzó unas tres semanas antes del evento, cuando en una de las reuniones mensuales del AOSUG nos preguntamos si este año haríamos algo para esta fecha. Las caras de todos se iluminaron, noté el entusiasmo antes de escuchar la primera voz favorable. Por supuesto, todos estuvimos de acuerdo en que sí, que era necesario que esta vez estemos nosotros mismos en el ojo de la tormenta, que podamos ver plasmado en la participación de otras comunidades y de toda la gente en general el esfuerzo que venimos desarrollando desde hace ya varios años con la comunidad local.

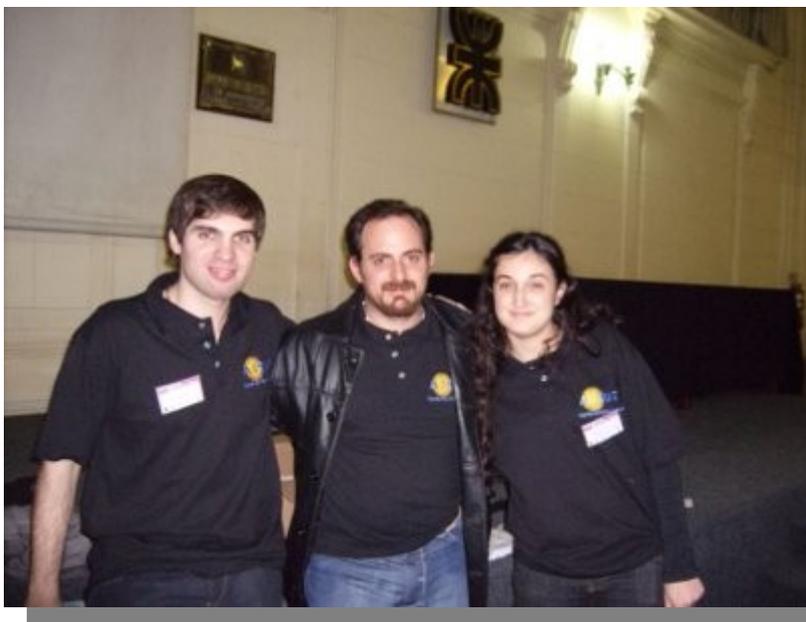


HeCSa y Ariel Corgatelli, de Tuxinfo. Ah, y un piano.

Lo primero que pensamos fue que tres semanas era realmente muy poco tiempo. Lo mejor sería mover la fecha, como lo estaban haciendo las demás comunidades del mundo, para una o dos semanas después del sábado 19 de setiembre. Y como siempre, lo sometimos a votación.

Y ahí empezó el desafío. Se decidió que el día sería el mismo del SFD, y que esa fecha no se movería por nada del mundo. Y que sería un sábado, lo que también resultaba un desafío considerando que siempre hemos tenido nuestras reuniones en días de semana, con la sola excepción de los eventos a los que otras comunidades amigas nos habían invitado.

Y comenzamos a hablar con los Campus Ambassadors para ver si se podía hacer en una u otra, comenzamos a visitar a los representantes de cada universidad, recorrimos salas de reunión, salones, y demás yerbas. Al final, encontramos en la UTN de Medrano un lugar que podía estar abierto hasta tarde un sábado, y que tenía el espacio suficiente para albergar a las posiblemente más de cien personas que esperábamos, concurrieran a este evento. Y eso de considerar esa cantidad de personas era sólo una cuestión de ojo, porque aún no teníamos ni idea de si llegaríamos a ese número.



Juan Daniel Pérez, HeCSa y Paola Gutiérrez, con sus chombas del AOSUG

Pero alguna luz nos alumbró, y solicitamos diferentes salones para diferentes charlas. Sin palabras de agradecimiento para la gente del Centro de Estudiantes de Ingeniería Tecnológica de la UTN (Marcelo Araujo superstar!), que se mostró no sólo interesada, sino muy solidaria para con el evento. Si hasta nos recibieron con credenciales plastificadas para cada uno de los que exponían. “Hola, lo llamamos de la aduana para avisarle que acá quedó retenido un paquete con cosas tuyas, cuando quiera pase”. Así comencé mi odisea por la máquina de impedir.

Que no te lo doy, que sí te lo doy, que si querés la mercadería vos fijate cómo hacés. Lo de siempre. Salí de la aduana a los gritos, porque aún no me resigno. “Señor, ¿se siente bien? Está muy colorado, eh!”. Claro, el taxista que me llevó de nuevo a mi casa no entendía mi desesperación por pensar que iba a llegar el día del SFD, y no iba a tener nada de material para entregar a los asistentes. Pero lo tuvimos, lo entregamos, y todos contentos.

“Uh, ¿quién se va a querer sacrificar y quedar en la puerta registrando a todo el mundo?”. Como pasa en nuestras comunidades, siempre aparecen personas con la mejor onda, y una voluntad de acero para bancarse eso. Paola, Pablo, y Luis Garrido Orozco. Cuando pasé por la puerta y lo vi a Luis que estaba famélico, me dí cuenta de lo que esto significa para él. Y el orgullo que tenía con lo que estaba haciendo. Así es, mis muchachos, cuando abran los libros de OpenSolaris que recibieron al registrarse, o cuando prueben el sistema operativo que ahora tienen, o cuando se pongan el pin, o la pulsera de goma, sepan que alguien estuvo detrás de todo eso, cuidando hasta el más mínimo detalle, sin pensar en él o ella, sino en los que asistieron. “Please, traeme un alfajor y un café, que me estoy muriendo de hambre. Creo que en cualquier momento me caigo redondo”. ¿Hace falta agregar algo más?



¿Y ahora? Ahora a ver quiénes serán los que expondrán. Tarea ardua, porque uno siempre piensa que seguimos siendo ese pequeño grupo de cuatro o cinco personas, y que no nos conoce nadie, y que nadie va a querer acercarse a este evento. Pero allí llegó la sorpresa.

No hizo más que un mail al CaFeLUG para que Rafa, siempre dispuesto a ayudar, nos pase los nombres de las personas que podrían asistir y exponer en el evento. Tipos así no se encuentran todos los días.

A la derecha, Luis Garrido Orozco, siempre colaborando con la comunidad

Y así llegó Diego Cañizares, con su eterna sonrisa y buena onda, y el excelente nivel que lo caracteriza cuando expone Python y sus cátedras magistrales de Software Libre en la Educación; y llegó Pedro Visintín, dando una charla magistral de Ruby on Rails que dejó a la gente más que feliz.

Pero no terminó allí. Dos especialistas en storage, como lo son Pablo Oddera y Gabriel Bensem pudieron iluminar los ojos de los asistentes cuando mostraron con increíble simplicidad algo que es complicado, como lo es el proyecto OpenStorage, punto de confluencia de varias tecnologías que anidadas entregan un producto de escala excelente. Cuando veía sus presentaciones antes de exponerlas, no pude hacer más que sorprenderme una vez más. Y los nervios. “¿Cuánta gente va a haber? Mirá que si son más de veinte, me va a dar como cosita...”. Fueron muchos, muchos más, y los comentarios que recibí fueron excelentes.



Diego Cañizares en plena exposición: Como siempre, un capo

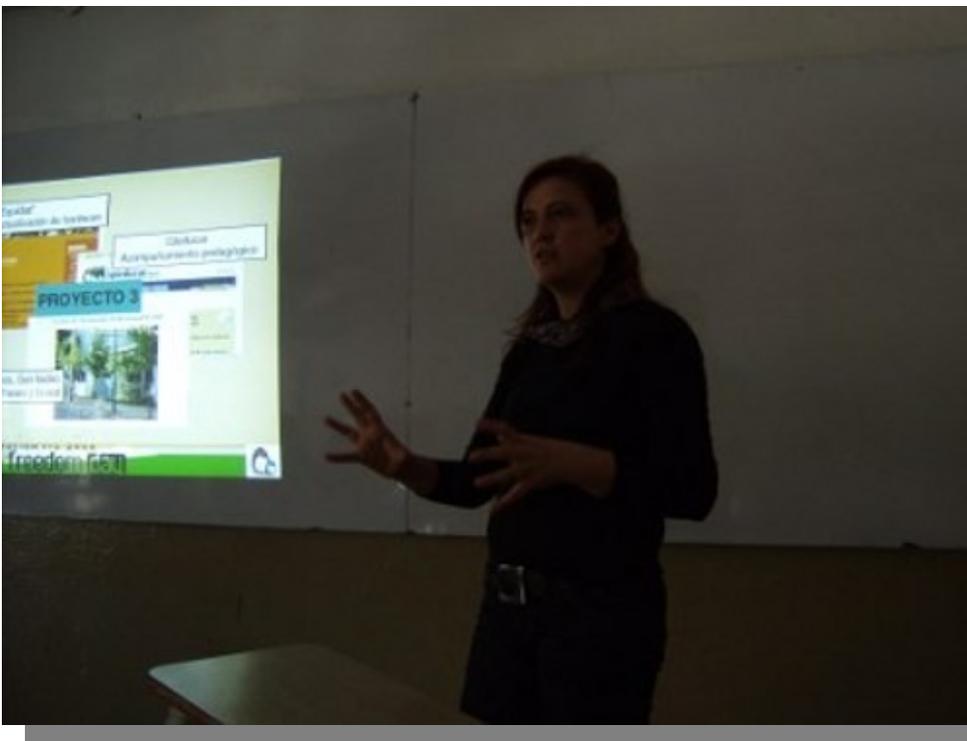


*Pablo Oddera y Gabriel Benselum:
almacenamos un lujo*



*Pedro Visintín: Calidad en una exposición
de Ruby on Rails*

Llegó el turno de María Eugenia Núñez y su exposición sobre el proyecto Gleducar. Como siempre, radiante frente a la presentación de un proyecto que deja a la gente pensando en que otros paradigmas son posibles, y que hay gente que realmente pone el hombro para que se hagan realidad. Y allí estaba ella, mostrándola con su sonrisa y la claridad de ideas y conceptos que sólo alguien que dedicó su vida a la docencia puede entregar.



No voy a comentar lo que significó que Paola Gutiérrez esté dando un taller de OpenSolaris sobre VirtualBox, porque sería injusto para con ella, aún cuando pusiera mis mejores razonamientos. Al igual que Juan Daniel Pérez, que se ocupó de entregar a los asistentes una excelente, y por mi culpa prolongada charla de OpenSolaris, sólo puedo decir que son dos profesionales en lo que se refiere a la didáctica, la claridad de conceptos,

María Eugenia Núñez: Una maestra, en el sentido amplio de la palabra

la excelencia en las exposiciones. Y ni que hablar cuando se necesita de manos para ayudar a organizar eventos de este estilo, y hasta para llevar bolsos con materiales para entregar a los que asistieron.



Paola Gutiérrez: Un taller sencillamente excelente

La gente que estuvo en sus charlas no pudo más que aplaudirlos de pié, como siempre que ellos entregan esa parte que los hace tan distintivos. Juan sacó la mayor parte de las fotos. Está del otro lado de esta lente. No paró en todo el día. El pibe es incansable, me encanta laburar con gente así. Cuando me llegó el aviso de que había más gente interesada en dar charlas, no me imaginé que se trataría ni más ni menos que de EDS, y su representante, Gabriel Bertone. "Ah, pero esto se pone cada vez mejor!" nos dijimos. Y así fue. Gabriel entregó una excelente presentación sobre los usos del software

libre en los ámbitos de las grandes empresas. Como en los demás casos, también los comentarios fueron excelentes.

Bueno, que hable de las presentaciones que yo dí, no sería ético. Pero acá vamos: siempre pienso que un par de bostezos no hubieran estado de más, y que si algún tomate me rebotaba en la nuca la culpa hubiera sido 100% mía. Pero la gente se quedó, y miró atentamente un tema también complicado como lo es Crossbow. Lo coronamos a la tarde haciendo un taller de esa tecnología, donde simulamos que un gerente nervioso nos llamaba para avisarnos que habían ganado un negocio nuevo, y que debíamos proveer equipos en diferentes redes virtualizadas en una misma máquina. Y la gente participó, no sólo en español.



Gabriel Bertone: Fusión de software libre y conceptos empresariales.



Hernán "HeCSa" Saltiel: Apertura y Crossbow



Y llegó el sorteo, los premios, las fotos, y el aplauso final.

Al terminar el día nos quedó el sabor dulce de la victoria. ¿Pero todo comenzó pensando que iba a ser así? Para nada. Si hasta las tres de la mañana de la noche anterior al evento, en estos casos, uno se queda caminando en círculos en el living de la casa, repasando cada palabra, cada gesto, cada idea y sentimiento que interesa que la gente se lleve.

Sólo esperamos desde el AOSUG haber podido encender en los que asistieron una inquietud lo suficientemente grande como para que el año que viene seamos más, y que muchos más se animen a evangelizar con ideas transparentes y poderosas en torno a lo que el software libre nos dio.

No es tan complicado. Aunque todo haya comenzado con una votación, y una sensación de vértigo frente a lo que se venía. A lo que se vino, y salió bien.

Muchas gracias a todos los que colaboraron, a los que pusieron el hombro, y sobre todo, a los que vinieron, y disfrutaron un día espléndido.

Hernán "HeCSa" Saltiel



Soluciones openSource con soporte real

Implementación de Soluciones Open Source

Capacitación y Soporte en Español

Integración y/o reemplazo de plataformas propietarias



SugarCRM - Primer Partner Oficial
en la República Argentina



Libertya - Software Libre de Gestión
Socio Corporativo - Training Center
autorizado

Oficina Comercial: Adolfo Alsina 425 - Piso 5 - Of. "A" - Ciudad Autónoma de Buenos Aires
Tel +54 11 4342-2110

www.disytel.com

Páginas recomendadas de Tuxinfo

Mes a mes vamos a ingresar nuestros sitios más destacados además de útiles para que nuestros lectores puedan recorrer tomando información relevante acerca de distribuciones, aplicaciones, utilidades, etc.

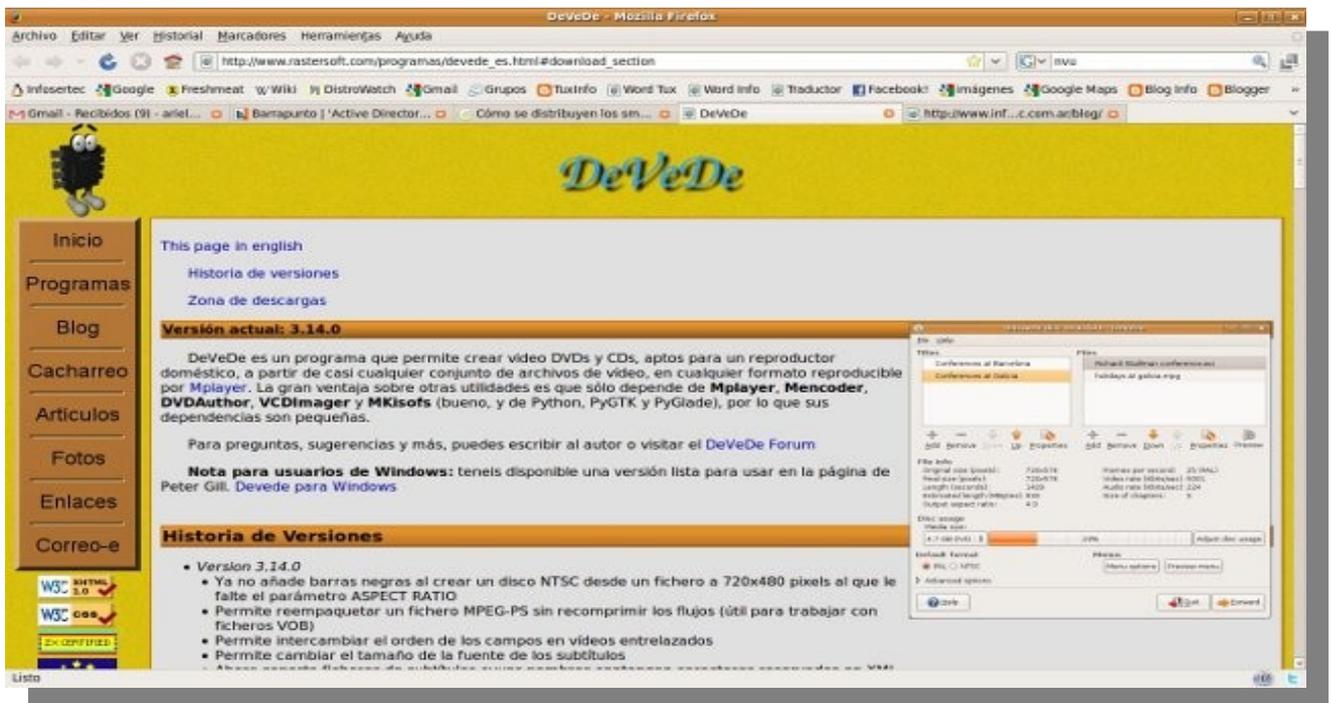
Por supuesto que no podemos ingresar todas las que quisiéramos por razones de espacio, pero poco a poco estarán reflejadas en esta sección. Además aceptamos las suyas para ser publicadas, como también poder brindar difusión a los sitios propios de cada uno de nuestros lectores (para ingresar sus sitios favoritos deben enviar un mail a info@tuxinfo.com.ar).

Ariel M. Corgatelli

DeVeDe

<http://www.rastersoft.com/index2.html>

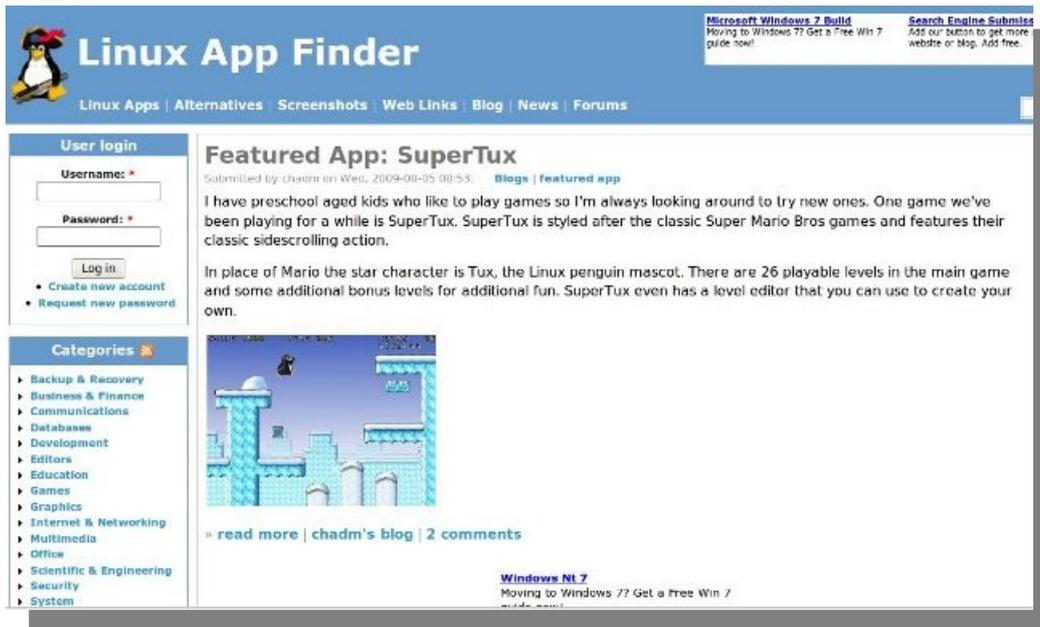
DeVeDe es un programa que permite crear video DVDs y CDs, aptos para un reproductor doméstico, a partir de casi cualquier conjunto de archivos de vídeo, en cualquier formato reproducible por Mplayer. La gran ventaja sobre otras utilidades es que sólo depende de Mplayer, Mencoder, DVDAuthor, VCDImager y MKisofs (bueno, y de Python, PyGTK y PyGlade), por lo que sus dependencias son pequeñas.



Linux App Finder, todas las aplicaciones para Linux

<http://linuxappfinder.com/>

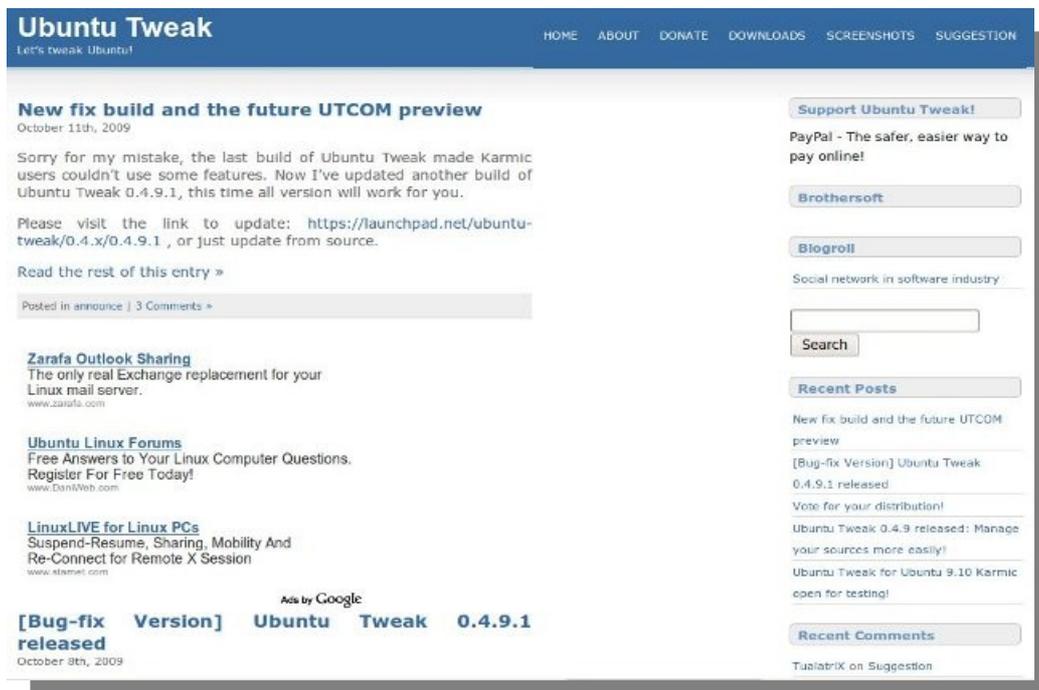
Linux App Finder es una web que reúne una importante colección de aplicaciones para Linux, listadas como posibles alternativas de código libre a programas para sistemas operativos Windows y Mac. De cada aplicación, podremos ver una descripción, un enlace a su web oficial, la interfaz gráfica para la que fue diseñada, plugins, programas asociados, así como una serie de enlaces de descarga para las diferentes distribuciones en las que se encuentra disponible (Debian, Ubuntu, Opensuse, etc).



Ubuntu Tweak 0.4.9

<http://ubuntu-tweak.com>

Esta herramienta nos permite simplificar la configuración de algunos parámetros de un sistema Ubuntu. Entre las novedades más importantes de esta última actualización destaca, el soporte para Karmic Koala, la opción de manejar las fuentes de terceros por separado a las fuentes comunes, nuevo sistema de notificaciones, corrección de algunos bugs y mejoras en seguridad.



Xirrus Wi-Fi Monitor

<http://www.xirrus.com/library/wifitools.php>

Xirrus Wi-Fi Monitor es una interesante herramienta con la que además de asegurarnos de no estar siendo víctima de usuarios externos que estén tratando de ingresar a Internet por medio de nuestra conexión, podremos obtener completa información de las redes inalámbricas bajo nuestra área de cobertura.



PODES SOLICITAR TU PACK TUXINFO

EL CUAL INCLUYE MUCHAS APLICACIONES LIBRES EN CD O DVD



Consulta como podes recibir el tuyo escribiendonos a: info@tuxinfo.com.ar

NVU, un excelente software de diseño Simple "en extinción"

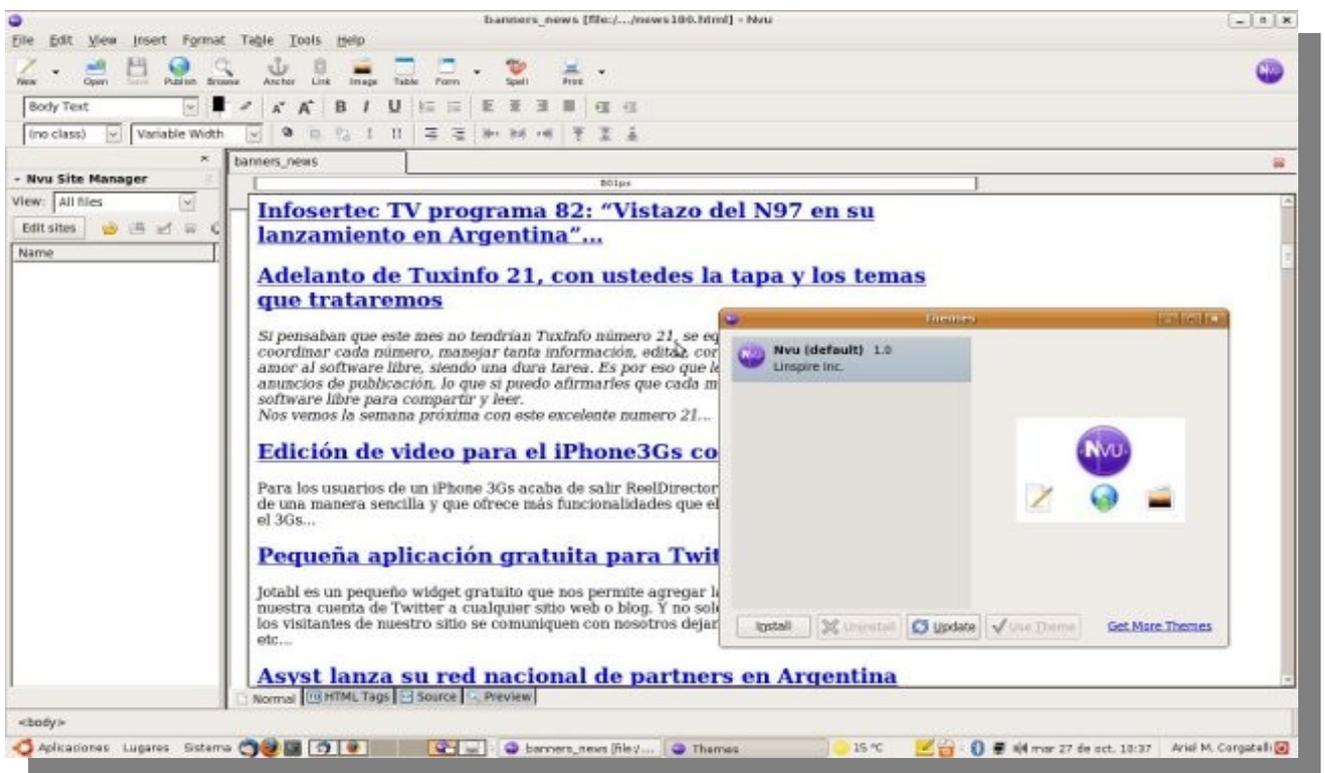
Como todos sabemos en GNU/Linux contamos con mucha variedad de soft para realizar las operaciones normales que se realizan con software propietario/privativo.

Puedo decir que en muchas oportunidades me han consultado por un software simple o sencillo para poder trabajar sobre diseño web en formato html. Y para ello puedo decirles que el más simple, efectivo y compatible con los estándares html se llama NVU y casi podríamos decir que es lo más parecido al famoso Dreamweaver de Macromedia.

En su última versión Beta 0.5 podemos encontrar una potente herramienta para el diseño en forma totalmente transparente.

NVU desde sus comienzos dedicó todo el esfuerzo a diseñar una aplicación libre con su código abierto para poder brindar una solución a los usuarios de Linux.

En su web oficial podemos encontrarnos como primera medida que el mismo dejó de existir con su nombre originario y los enlaces a sus descargas.



Nvu es un editor de páginas web WYSIWYG Multiplataforma basado en Mozilla Composer, pero de ejecución independiente. Añade características nuevas como soporte integrado de CSS y mejor gestión del soporte FTP para actualización de los archivos.

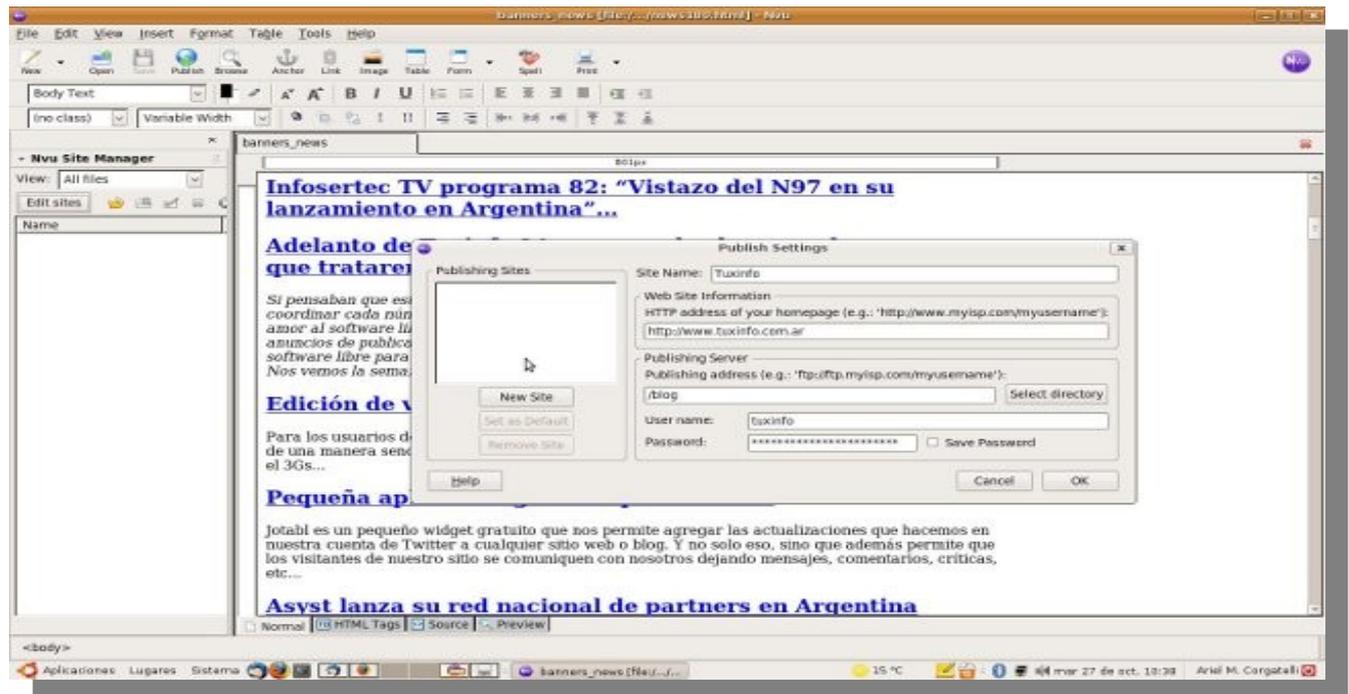
Gracias a su editor y a sus diferentes visualizaciones disponibles dentro de su interfaz (código fuente, ventana WYSIWYG, visión con tags de HTML realzados), entre los cuales es posible cambiar mediante un sistema de pestañas.

Además incluye características como gestión de trabajo mediante proyectos, cliente FTP integrado para subir la página directamente desde Nvu y soporte para todos los elementos típicos: marcos, formularios, tablas, plantillas de diseño, hojas de estilo CSS, etc.

Nvu está disponible para Linux, Mac OS X y Microsoft Windows, aunque puede compilarse para cualquier plataforma con el Netscape Portable Runtime. Mozilla Composer está todavía en desarrollo como parte de la suite Mozilla para otras plataformas.

Su desarrollo original comenzó en junio de 2005 en donde se realizaron cambios dentro del código de Mozilla Composer. Desde que Mozilla Suite se dejó de desarrollar (continuada por la comunidad como SeaMonkey), nadie ha introducido el código de NVU de vuelta a Composer.

Daniel Glazman anunció el 15 de septiembre de 2006 que dejaba oficialmente el desarrollo de Nvu. KompoZer es un derivado de NVU mantenido por la comunidad, mantiene el código base y corrige varios bugs hasta que se lance el sustituto de NVU. Glazman está desarrollando el sucesor de Nvu, llamado en un principio Composer, como el proyecto de Mozilla.org. Estará escrito desde cero y basado en Gecko 1.8 y XULRunner.



Sobre Nvu

Nvu (la N-vista pronunciada, para una "nueva vista") es un proyecto comenzado por Lindows, Lindows contribuye capital significativo, especialización, servidores, bandwidth, mercadeo, y otros recursos para garantizar la continuación y éxito del Nvu project. (Fragmento extraído del sitio NVU.ORG)

DESCARGA

Sólo tendremos que dirigirnos a su sitio oficial:

<http://nvu.com/index.html>

Donde encontraremos un enlace :

<http://net2.com/nvu/download.html>

Una vez ahí veremos todas las opciones de descarga para varios sistemas operativos, Linux, Microsoft, Solaris.

Nuestra opción es descargar el comprimido TAR.GZ y la opción RPM para las populares Distribuciones Linux Fedora y Mandriva.

Como novedad podemos contar que en los paquetes de NVU vienen incluidas las librerías GTK, de esta forma los mismos están resueltos en su integridad para una mejor y más fácil Instalación.

Instalación del TarGZ.

Este archivo una vez descargado tendremos que descomprimirlo, lo cual podremos hacerlo de dos formas, uno en forma gráfica y la otra en modo texto.

En forma gráfica sólo tendremos que llamar a la aplicación ARK y descomprimirlo en alguna carpeta. También en forma gráfica según la distribución que tenemos corriendo en nuestra PC haciendo un click en la descarga se lanza la aplicación en forma automática.

En forma textual tendremos que ejecutar:

```
# tar -zxvf nombre del archivo
```

Con lo cual tendremos así el soft listo para ejecutarlo desde la misma ubicación donde lo descomprimos.

Instalación de RPM.

Esta operación es muy simple y dependiendo de la Distribución utilizada se realiza en forma gráfica con un simple Clic en el archivo rpm.

Desde una consola simplemente con ingresar el siguiente comando:

```
# rpm -i xxx.rpm
```

ACLARACIÓN

No siempre el soft descargado funciona con sólo descomprimirlo, hay veces que también tendremos que compilarlo, no es así este caso. Estos son los paquetes que vienen provistos de las fuentes en su mayoría.

EJECUTARLO

Sólo tendremos que entrar en la carpeta que generamos en la descompresión y buscar el ejecutable NVU, que sólo haciendo un clic sobre este tendremos ya lanzado el soft como muestra en la figura 1.

En el caso de la instalación desde un archivo RPM sólo con ejecutar:

```
$ ./nvu
```

El uso es casi igual a cualquier editor web, se puede insertar imágenes, hipervínculos, texto y demás.

Podremos trabajar directamente del sitio FTP de nuestra web para lo cual tendremos que ingresar a él con los datos del FTP.

Sus menú son muy sencillos podemos encontrar los siguientes :

New. Es simplemente el encargado de darnos la opción para iniciar la creación de una página.
Open. Con este podemos llamar a una ya existente con la potente opción de buscar en nuestro disco.

Save. Nos permite la opción de guardar y guardar como. (viejo sistema Microsoft)

Publish. Opción útil para poder subir directamente el formulario en el cual estamos trabajando a nuestro sitio en forma directa y transparente.

Browse. Utilidad que nos permite previsualizar le pagina para luego poder subirla.

Anchor. Permite modificar las propiedades de una tipografía, imagen, etc.

Link. Permite la creación de Link a diferentes url.

Image. Posibilidad de insertar imágenes.

Table. Posibilidad de añadir tablas de contenido.

Spell. útil opción con la salvedad de ser una aplicación creada en el Idioma Inglés. (si alguien quisiera cambiarlo al Español mucha gente se lo agradecería)

Print. Posibilidad de Imprimir.

Error de Uso.

En un momento nos encontramos con error al ingresar el fondo en el formulario donde al haber previamente ingresado uno no podíamos cambiarlo y eso se debió simplemente a que como primera medida tendremos que eliminar el anterior y luego ingresar el nuevo.

La forma de agregar elementos a nuestra web es muy simple, sólo debemos navegar por la barra superior en donde tendremos todos los botones generados para llamar la acción requerida.

Diferencias con otros editores.

Podemos decir que sólo varían en su uso y nombrando podemos dar algunos ejemplos como la modificación del tamaño del texto el cual se genera haciendo doble clic como muestra en la figura cuatro.

Otro sería cuando tenemos que cambiar el formato del texto para lo cual tendremos dos opciones, o modificarlo directamente en la barra superior donde dice body text, ahí tenemos varios tipos de textos prearmados, o bien ingresamos en formato de texto como muestra en la figura cuatro y podemos definir la fuente, el color y la textura.

En la barra inferior podemos encontrar varias solapas para ver el código html, viewer de nuestra web y demás.

Y la principal diferencia es que todavía no está manejando las capas en forma óptima.

CONCLUSIÓN

El mundo de GNU/Linux es muy amplio y siempre nos permite tener varias opciones para realizar nuestros trabajos cotidianos y en este caso lo cumple en perfecto estado.

Libre y a la altura de un soft como lo es el de Macromedia, el cual cabe destacar que posee muchas más opciones de configuración, idiomas soportados, etc., pero a no desanimarse, Linux viene con todo y de a poco ya está llegando a cubrir hasta los lugares más difíciles.

Los escritorios.

Suerte y no podemos dejar de probarlo.

Ariel M. Corgatelli

Bacula: Software para salvas automáticas II Parte

En el artículo anterior comentamos la importancia de realizar copias de seguridad de la información y qué cosa es Bacula. En el presente artículo comentaremos cómo se debe configurar Bacula para realizar salvas.

Los componentes de Bacula

Como también se comentó en el pasado artículo, Bacula está compuesto por 5 componentes fundamentales:

- Director
- Storage
- Catalog
- File Daemon
- Consola

Por defecto, cuando compilamos desde los códigos fuentes o instalamos desde repositorios, Bacula está listo para funcionar de forma local pero nosotros no queremos quedarnos solamente en un entorno local sino que queremos instalar y configurar un entorno de Salvas y Recuperación de la Información para nuestra empresa, pymes u otros.

Iremos explicando paso a paso como se deben configurar los componentes de Bacula para su correcto funcionamiento.

Configuración del Director

El fichero de configuración del Director se encuentra, sino cambiamos las opciones a la hora de compilar, dentro del directorio `/etc/bacula` con el nombre de `bacula-dir.conf`. Por defecto contiene un gran número de configuraciones pero yo, en lo personal, prefiero separar las cosas para tener una mejor organización y en caso de problemas poder darle solución directamente en el fichero donde está el problema.

La primera sección que nos encontramos es "Director". Veamos un ejemplo (totalmente funcional ;)):

```
Director {
  Name = bacula-dir
  Description = "Bacula Director"
  DIRport = 9101
  DirAddress = aaa.bbb.ccc.ddd
  QueryFile = "/etc/bacula/query.sql"
  WorkingDirectory = "/var/bacula/working"
  PidDirectory = "/var/run"
  Maximum Concurrent Jobs = 20
  Password = "xxxx"
  Messages = Daemon
  FD Connect Timeout = 5 min
  SD Connect Timeout = 5 min
}
```

Explicaremos ahora qué cosa es cada una de las configuraciones establecidas anteriormente.

- Name:** es el nombre con el cual se identificará el servidor de Bacula en la red.
- Description:** es una breve descripción del servidor
- DirPort:** es el puerto donde escuchará el servidor y al cual los clientes u otros componentes realizarán peticiones
- DirAddress:** es la dirección IP de la PC donde está ejecutándose el servidor de Bacula
- QueryFile:** es el fichero usado para realizar consultas a la BD desde la consola
- WorkingDirectory:** es el directorio donde están ubicados los ficheros que Bacula usa para definir estados u otros
- PidDirectory:** es el directorio donde está ubicado el fichero de PID (Process ID)
- Maximum Concurrent Jobs:** es la cantidad máxima de trabajos que Bacula puede realizar de forma concurrente sin importar el tipo de trabajo (Restore, Backup, Verify)
- Password:** es la contraseña que permite la comunicación entre el servidor de Bacula y la consola de Bacula
- Messages:** especifica a quien(es) serán enviados los mensajes de Logs u otros
- FD Connect Timeout:** establece el tiempo que el servidor intentará establecer conexión con los clientes (FD)
- SD Connect Timeout:** establece el tiempo que el servidor intentará establecer conexión con los storage (SD)

Existen otras configuraciones las cuales puedes consultar en el manual del usuario de Bacula pero por no ser de gran importancia para mi entorno no las he tocado en este artículo al igual que no tocaré muchas en los componentes que restan.

```
JobDefs {
  Name = "DefinicionPorDefectoParaCopias"
  Type = Backup
  Level = Incremental
  Storage = FileSAN
  Messages = Standard
  Priority = 5 #Es la opcion por defecto, a mayor numero menos prioridad
  Reschedule on error = yes
  Reschedule interval = 30 minutes
  Reschedule times = 2
  Max Start Delay = 2 hours
}
```

JobDefs establece un nivel de configuraciones que puede ser usado en cualquier trabajo. Se pueden definir tantos JobDefs como se quiera en dependencia de las características de los clientes y la información a salvaguardar.

Name: es el nombre con el cual se identificará la definición del trabajo.

Type: es el tipo de trabajo que se realizará: Backup, Restore, Verify y otros que están en planes para futuras versiones.

Level: es el nivel de la salva. ¿Por qué incremental? Si Bacula no encuentra una copia completa previa de algún trabajo entonces actualiza la copia de Incremental a Full de forma tal que la primera salva de cualquier trabajo es Completa (Full). Se pone Incremental para que luego que se realice la copia Full los trabajos sigan realizando copias Incrementales.

Storage: es el nombre del Storage donde se salvará la información

Messages: especifica a quien(es) serán enviados los mensajes de Logs u otros

Priority: establece la prioridad del trabajo. Se usa fundamentalmente cuando poseemos un gran número de clientes y de trabajos de forma tal que algunos trabajos tengan prioridad en dependencia de las características.

Reschedule on error: pone el trabajo en cola nuevamente en caso de que haya ocurrido algún problema (no hay comunicación con el cliente, fallo en la conectividad, etc)

Reschedule interval: es el tiempo que demora el mismo trabajo en ser mandado a ejecutar nuevamente. En el caso de mi configuración es cada 30 minutos.

Reschedule times: establece la cantidad de veces que el trabajo será puesto en cola. Si ponemos un número muy grande podemos correr el riesgo de que los demás trabajos restantes no se ejecuten pues el mismo trabajo se estará ejecutando por cuantas veces se haya especificado.

Max start delay: es el tiempo que el trabajo intentará ejecutarse. Si por algún fallo en un período de 2 horas el trabajo no es ejecutado pasa a un estado de Fallido y da paso a que los demás trabajos se ejecuten.

```
Schedule {
  Name = "ScheduleFull"
  Run = Full 1st sun at 2:00
  Run = Differential 2nd-5th sun at 2:00
  Run = Incremental mon-sat at 2:00
}
```

Schedule establece los períodos en los cuales se realizarán las copias de seguridad.

Name: es el nombre con el cual se identificará el schedule o programación en la definición de los trabajos.

Run: establece los períodos en los cuales se realizará la salva. En mi caso personal se realiza una copia completa todos los primeros domingos de cada mes, a partir de esa copia completa se realizan copias incrementales todos los días y las copias diferenciales el 2do y 5to domingo de cada mes. Esta configuración depende en su mayoría de las políticas establecidas para salvar la información.

```
Storage {
  Name = FileSAN
  Address = aaa.bbb.ccc.ddd
  SDPort = 9103
  Password = "xxxxxxx"
  Device = FileStorage
  Media Type = File
}
```

Storage establece el lugar donde se almacenará la información: HDD, dispositivos USB, Cintas, DVD, etc.

Name: es el nombre con el cual se identificará el storage en la definición de los trabajos.

Address: es la dirección IP de la PC donde está el Storage. Es importante no usar "localhost" o nombres que no estén especificados en algún DNS.

SDPort: es el puerto por donde escuchará el demonio de Storage.

Storage: es el nombre del Storage donde se salvará la información

Password: es la contraseña que permite la comunicación del Director con el Storage Daemon.

Device: es el nombre que se le da al dispositivo en el fichero de configuración del SD (bacula-sd.conf)

Media Type: establece el tipo de almacenamiento (File, DVD, Tape, etc)

```
Catalog {
  Name = MyCatalog
  dbname = "bacula_db";
  dbuser = "bacula_user";
  dbpassword = "xxxxxx"
  DB Address = aaa.bbb.ccc.ddd
}
```

Catalog establece las configuraciones para que Bacula pueda conectarse a la BD y salvar la información relacionada con las salvas.

Name: es el nombre con el cual se identificará el Catálogo en la BD.

dbname: nombre de la BD

dbuser: usuario con privilegios en esa BD

dbpassword: contraseña del usuario con privilegios en esa BD

DB Address: es la dirección IP donde está instalado el servidor de BD

```
Messages {
  Name = Standard
  mailcommand = "/sbin/bsmtp -h servidor -f \"\"(Bacula Service\)
\<direccion_correo\>\" -s \"Bacula: %t %e of %c %l\" %r"
  operatorcommand = "/sbin/bsmtp -h servidor -f \"\"(Bacula Service\)
\<direccion_correo\>\" -s \"Bacula: Intervention needed for %j\" $
  mail = direccion_correo_receptor = all, !skipped
  operator = direccion_correo_receptor = mount
  console = all, !skipped, !saved
  append = "/var/bacula/working/log" = all, !skipped
}
```

Messages define qué se enviará y a qué personas.

Name: es el nombre con el cual se identificará en otras configuraciones.

mailcommand: es el comando a ejecutar con los parámetros “servidor” que es el servidor SMTP que se encargará de enviar el correo y `-f` que define el “Desde(from)” `-s` define el “Título del Mensaje(subject)” y los parámetros restantes `%t`, `%e`, `%c`, `%l`, `%r` definen informaciones específicas de Bacula (ver el Manual oficial de Bacula para mayor información al respecto)

mail: define las direcciones de correo de los clientes o cualquier otra persona, separadas por coma, a las cuales será enviado el correo

operator: define las direcciones de correo de los operadores (administradores de Bacula generalmente), separadas por coma, a las cuales será enviado el correo

console: define qué mensajes serán enviados a la consola

append: define el texto que formará el cuerpo del mensaje, en este caso, todo el contenido de `/var/bacula/working/log` referente a ese trabajo que falló

Esto es todo por el momento, en próximos números detallaremos un poco más la configuración de Bacula.

Reynier Pérez Mira

País: Cuba

Correo: rperezm@uci.cu, reynierpm@gmail.com

Profesión: Ingeniero en Ciencias Informáticas



Cámara Argentina
de Empresas

de Software Libre

Inscribí a tu empresa en

www.cadesol.org.ar

Probando Ubuntu 9.10 Karmic Koala Beta a Fondo

La nueva versión de Ubuntu está en el horno, y mientras tanto, la gente de Canonical, va largando pre releases para que la comunidad vaya testeando y dejando su feedback.

En esta ocasión probamos la última beta disponible en Ubuntu 9.10 (Karmic Koala) Daily Build la cual probé (sigo probando) a fondo.

Antes que nada voy a comentar un par de cosas que van a determinar el alcance de este laboratorio.

El objeto de prueba es mi netbook Toshiba NB105 ¿Por qué? Bueno, considero que si algo puede correr ágil, consistente y además potenciar al hardware, aunque este se modesto, estamos ante la presencia de un buen S.O.

Hace rato que vengo probando distribuciones GNU LINUX para alojar definitivamente en mi netbook (Publicidad subliminal: ver mi blog) y hasta ahora sólo dos han calificado, gOs Linux, la cual no saca lo mejor del hardware y Linux Mint 7 Gloria XFCE Edition (Ver en este mismo número) La cual es ¡Asombrosa! Pero no llega tener la facilidad de uso del Escritorio GNOME. ¿Cumplirá el Koala Kármico este objetivo? Veamos....

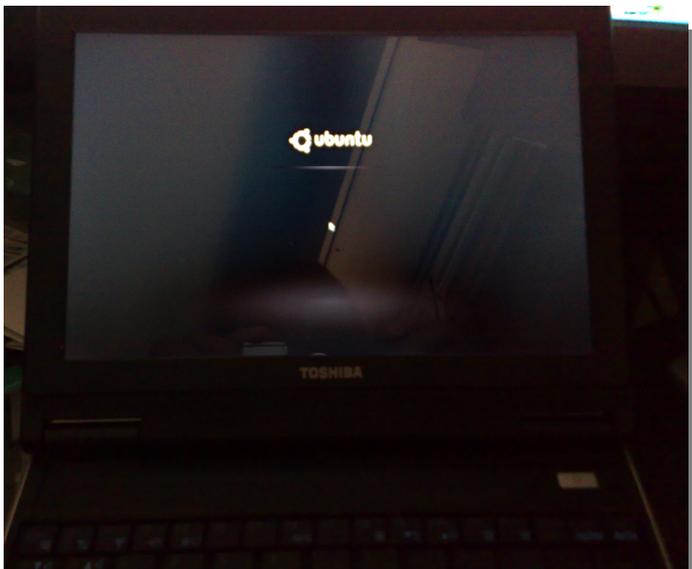
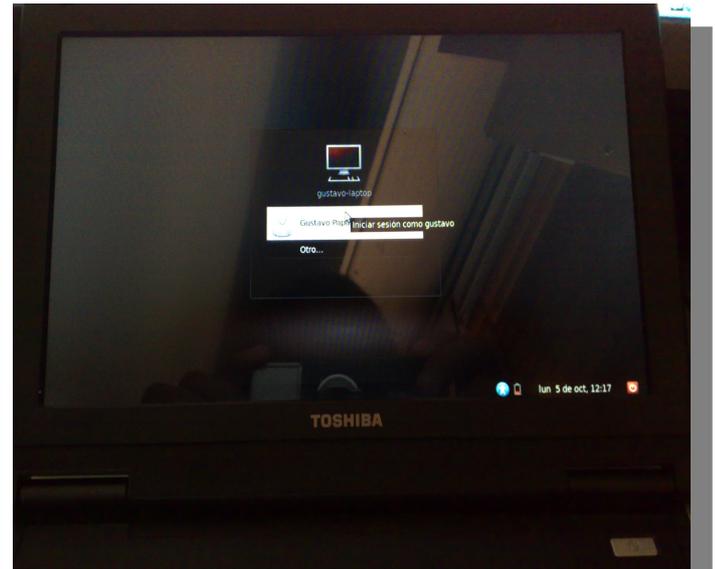
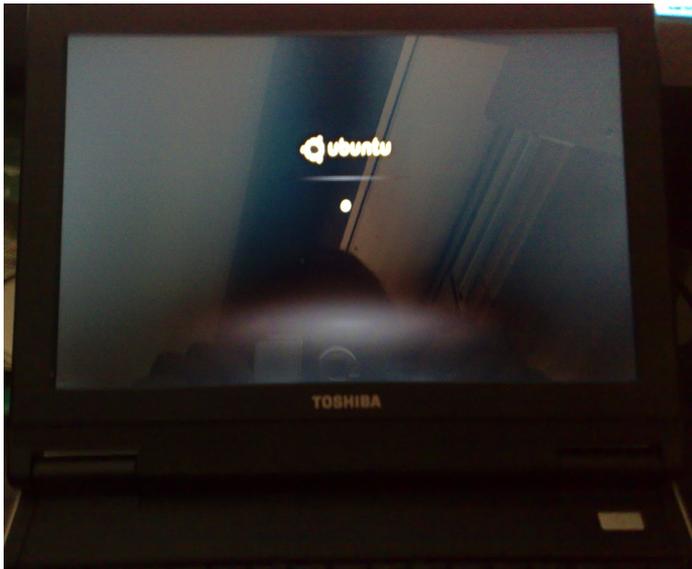
Otro tema. A esta altura, explicar la instalación de un sistema Ubuntu o basado en Ubuntu, puede ser desgastante (Dado que es la instalación más fácil de un S.O. que he visto en mi vida), pero si todavía no están seguros acerca de esto, bueno, lean el artículo sobre Lin-X 1.1 en este mismo número y todas sus dudas acerca de la instalación quedarán despejadas.

Comienza el test

Luego de la instalación Ubuntu típica, La cual usa por defecto el sistema de archivos Ext4 salvo que le indiquemos lo contrario, lo primero que nos va a llamar la atención es lo rápido que bootea el sistema, **8 segundos** desde que comienza a cargar grub, hasta que se llega a ventana de login y **dos segundos más** desde que ponemos la clave y le damos enter hasta la carga total del escritorio. ¿Bien no? Tengamos en cuenta que estamos hablando de un procesador Atom N270 con 1 gb de RAM del cual una parte se usa para la placa de video.

Durante este corto proceso de booteo, veremos la siguiente secuencia de pantallas.



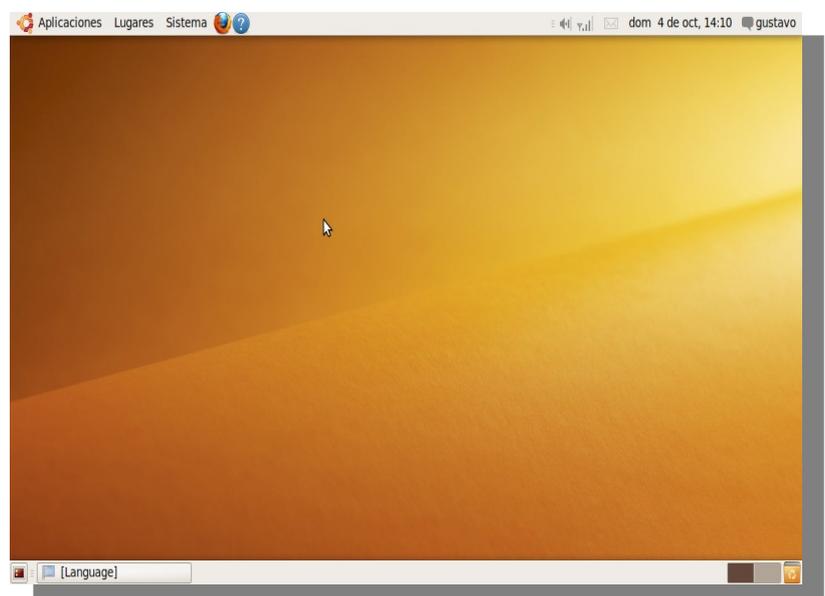


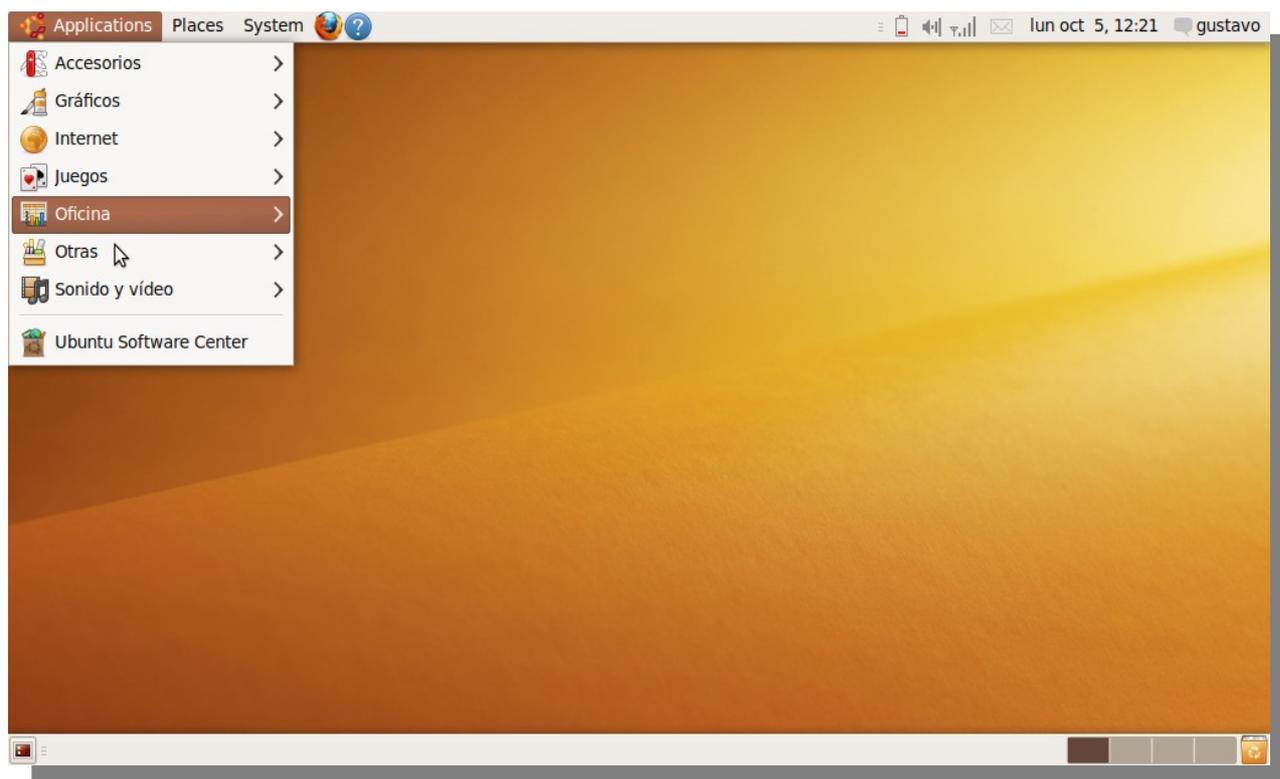
Como vemos, los cambios en la estética de la distribución, sutiles pero notorios, se notan desde el inicio.

Como habíamos dicho recién, los cambios en el diseño gráfico de la interfaz se notan, tanto como la mejora en la velocidad de carga del sistema, sin embargo, no deja de ser el mismo Ubuntu en la apariencia general.

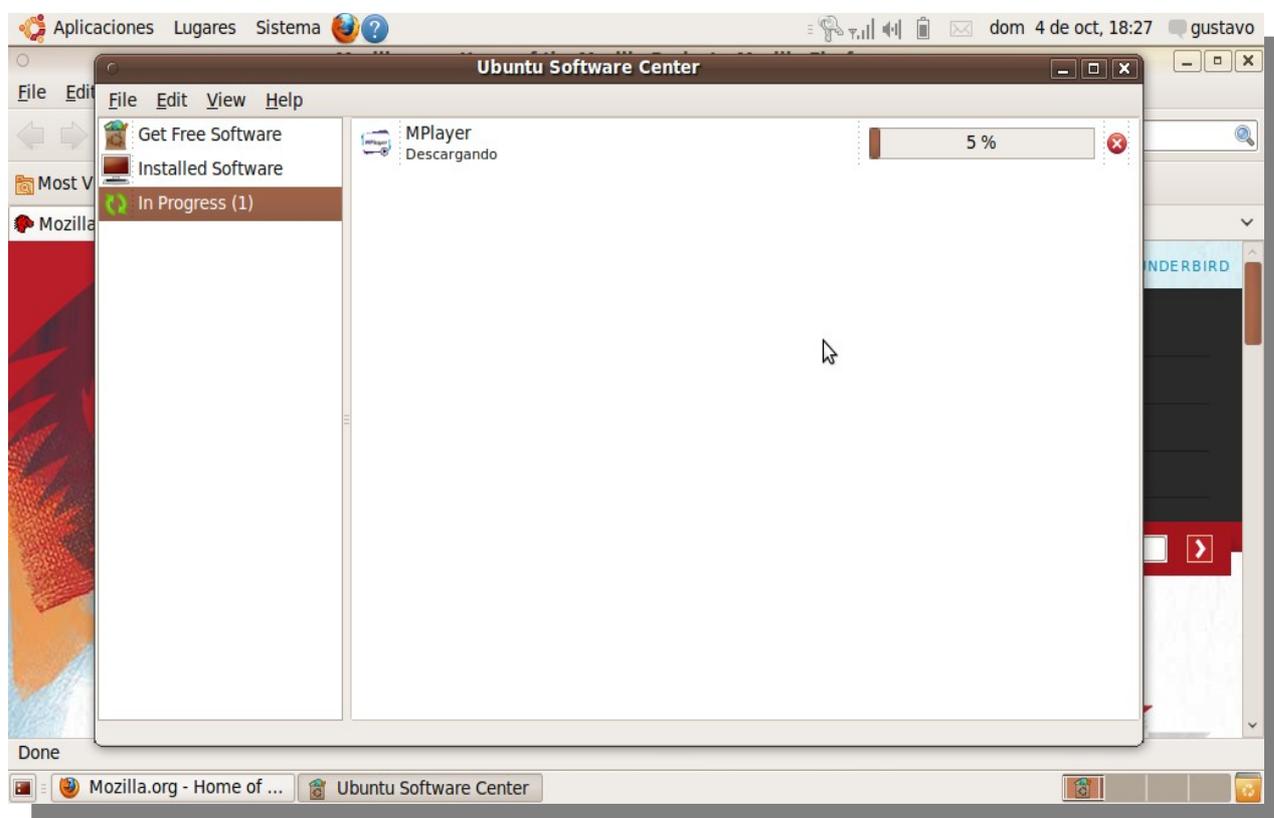
Esto es bueno por un lado, dado que lograr un cambio visible sin grandes cambios aparentes es todo un logro de diseño, pero malo por el otro, ya que la esperada (al menos por mí) renovación radical del escritorio, es algo que aún no ocurre y que seguramente no ocurrirá de una beta a una versión final (¡Sorpréndanme por favor!). Tal y como vemos en la captura del escritorio recién instalado.

¡Lo ven! Igual pero distinta, y eso es algo que se aprecia mejor al desplegar los menús y ver la gráfica de los mismos.

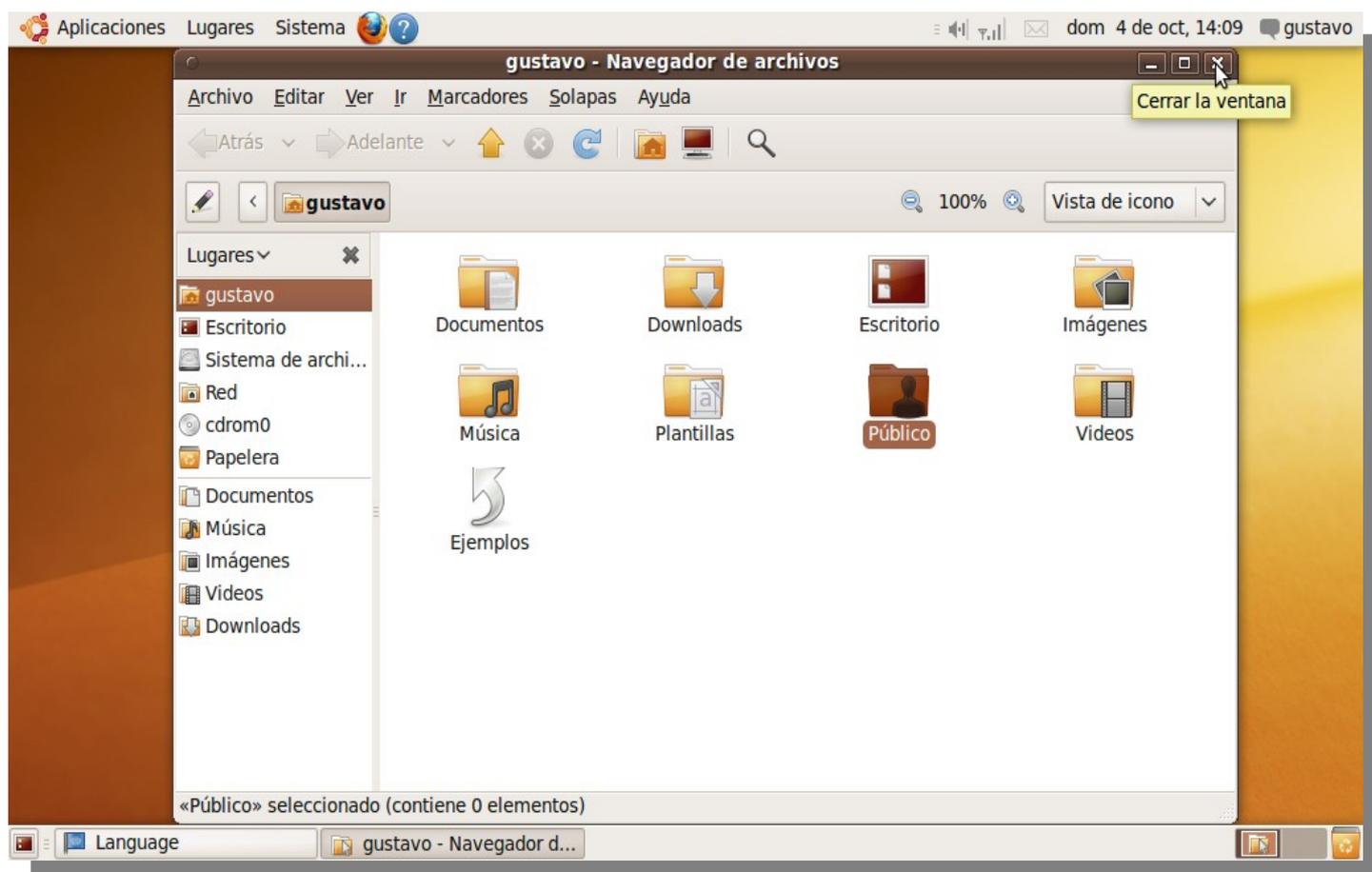




Notarán que Agregar nuevo software cambió por Ubuntu Software Center, el cual se parece mucho al MintInstall de Mint Linux, pero Marrón en vez de verde :-).



Otro detalle estético importante es el cambio en los iconos del sistema y el explorador de directorios.



Si observamos bien la captura anterior del escritorio recién instalado el fondo por defecto también es algo distinto pero nada espectacular, esperemos que sea un poco más impactante.

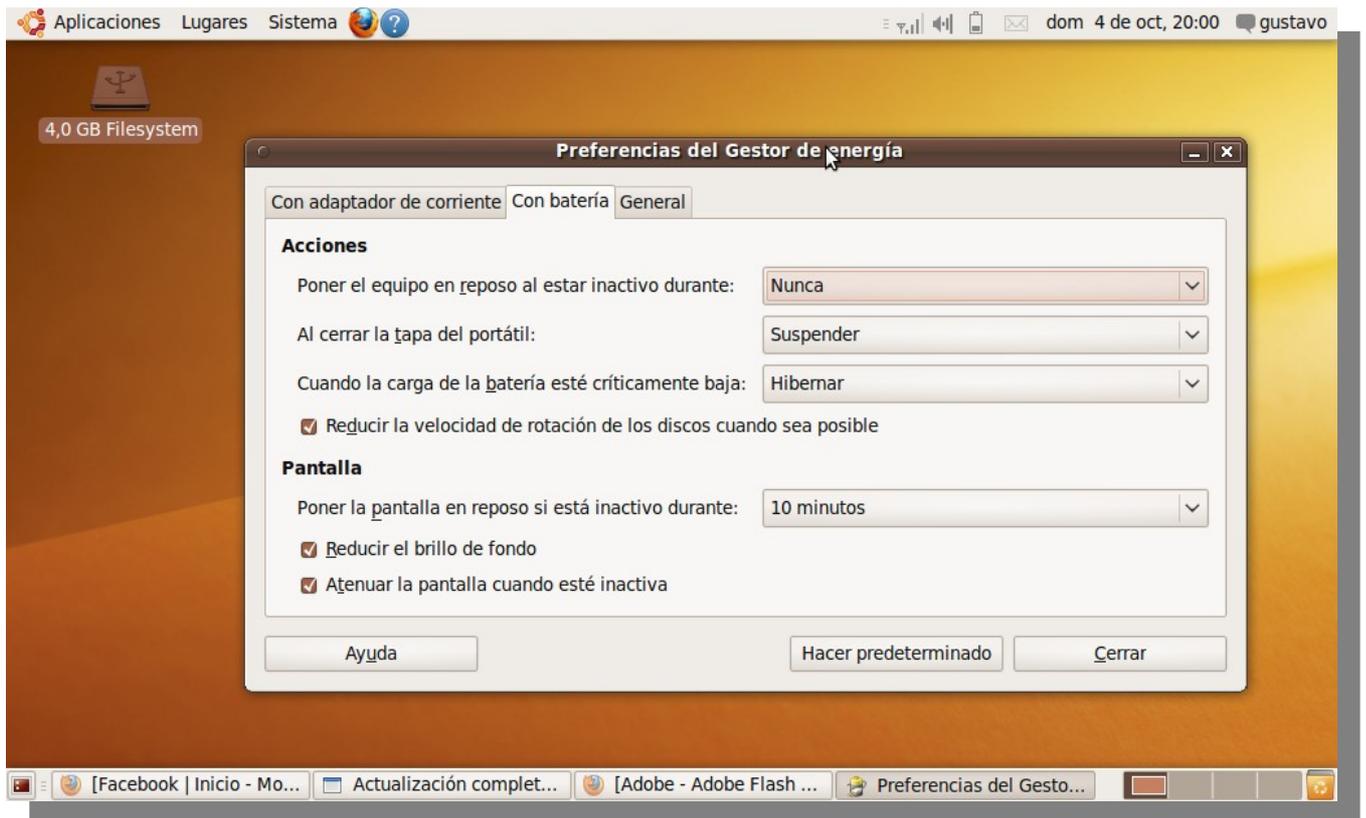
Cambios interesantes:

Vamos a uno de los cambios más relevantes, Si hay algo que me pasaba al instalar Jaunty 9.04 era que la netbook se tornaba bastante pesada de usar; al punto de que la tarea de desplazarse (scroll) en el navegador de internet parezca algo ciclópeo.

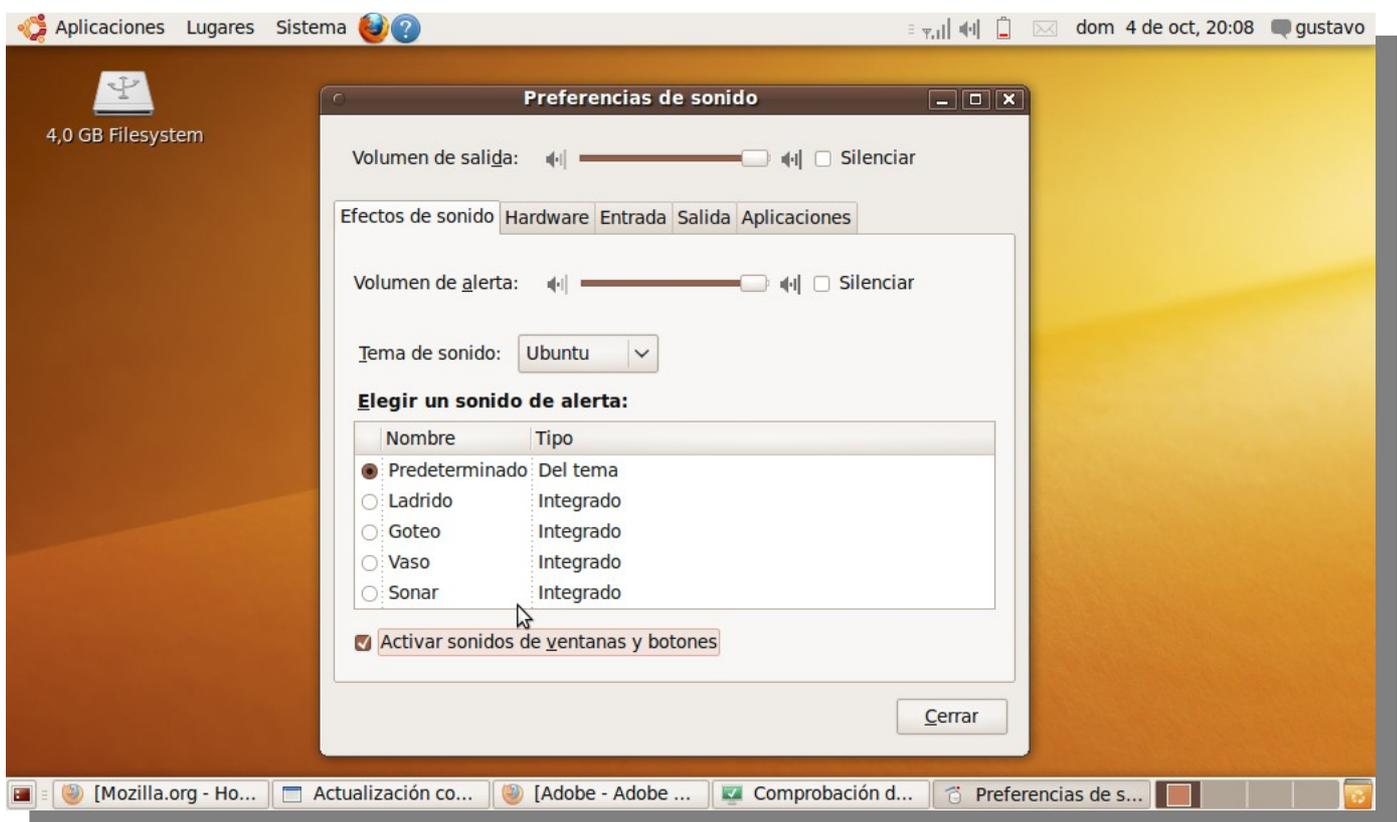
Afortunadamente el Koala Kármico (¡Qué buen nombre!), no sólo arranca más rápido, ¡funciona con la velocidad de una gacela! Quizás esto se deba al Kernel 2.6.31-11, el cual se sabe; está optimizado para rendir mejor en sistemas de bajos recursos.

Esperemos que esto no cambie en la versión final, y que en lo posible mejore. Ubuntu 9.10 funciona más ágil y estable que Mint 7 con XFCE (También en este número).

Otra mejora importante está dada en el administrador de energía, ya que no sólo suspender e hibernar vienen configurados por defecto (antes venía con apagar pantalla) en las opciones de alimentación con batería, sino que el administrador de brillo de pantalla se comporta de manera más coherente que en 9.4 (recuerden que esto sucede en mi Toshiba) lo cual no significa que sea igual en otra netbook/notebook.



El control de audio también está muy mejorado, dado que ya no es un caos de cuanto controlador multimedia hay en la PC, sino que trae lo justo y necesario para el control estándar de esta opción, ya que si queremos funciones multimedia más avanzadas, el mundo GNU suele proveernos de utilerías mucho más acordes que apilar controles en este panel. Bien por Ubuntu.

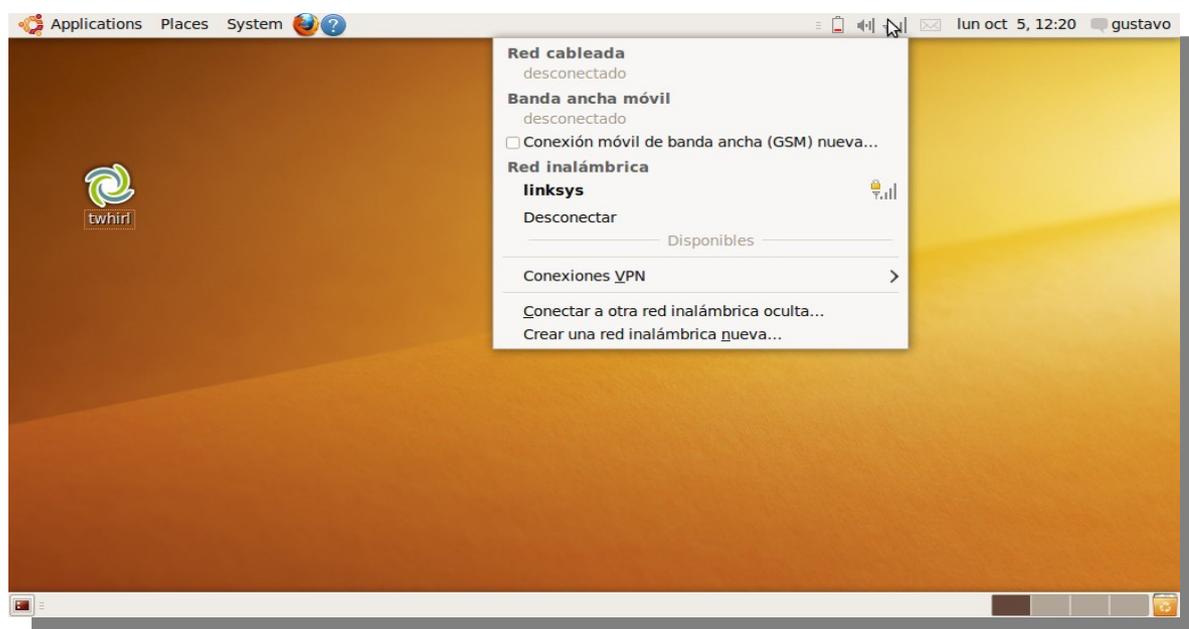


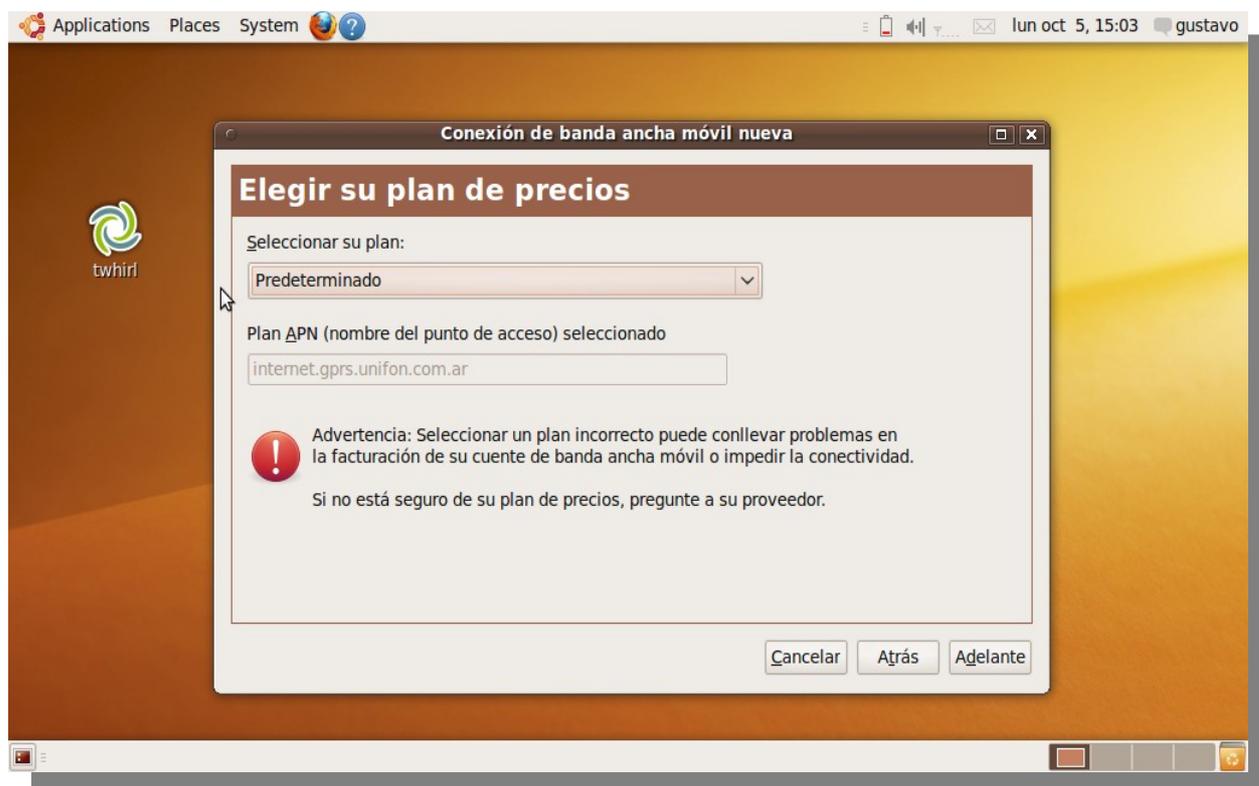
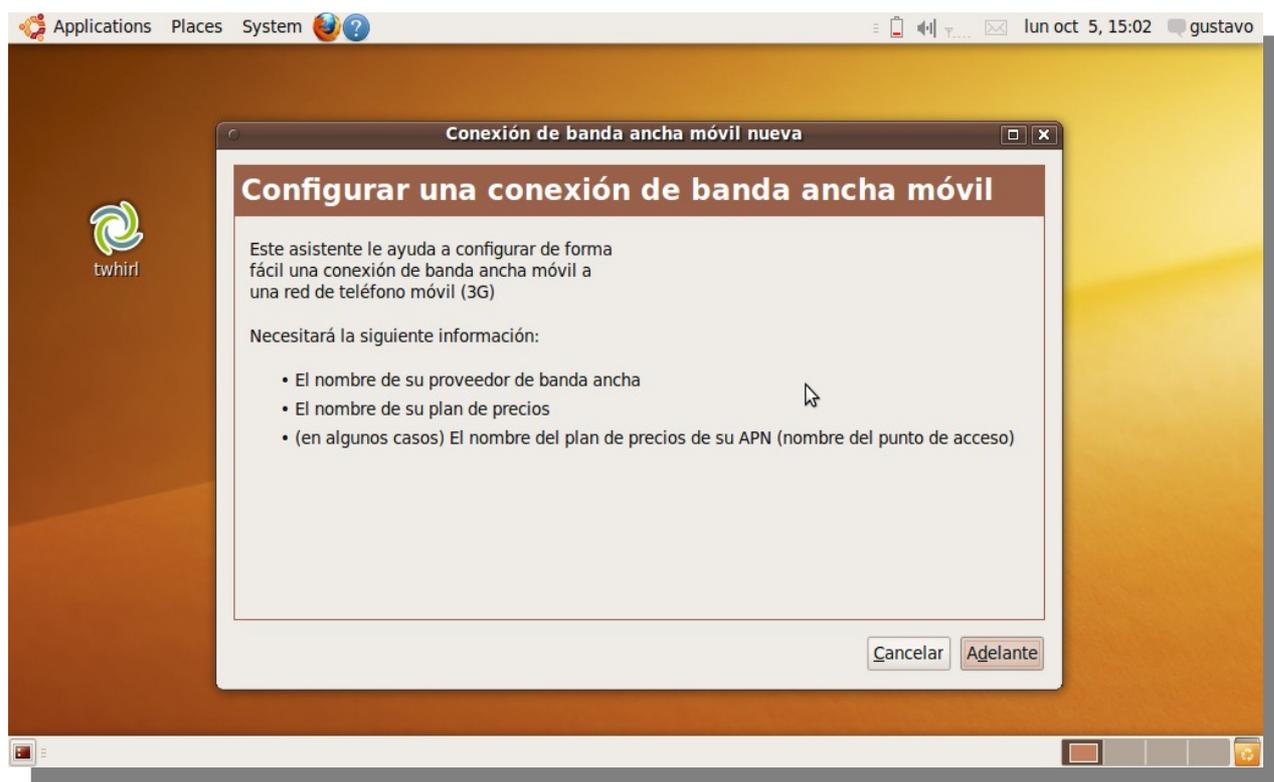
¡Firefox 3.5!

Si! Si! Si! Al fin! ¡Ya nunca más ir al menú de Firefox para abrir una pestaña, una combinación de teclas! Ya tenemos a Firefox en las mismas condiciones que el mundo Windows. Observen al glorioso “+” al lado de la pestaña abierta. Y ahora hablando en serio, la posibilidad de poder contar con todas las mejoras de la versión 3.5 comentadas en cientos de miles de sitios de internet.



Una cosa que me llamó la atención, fue que al conectar mi teléfono 3g, el sistema no me abrió el configurador de conexión 3g como habitualmente lo hacía en Jaunty, estaba a punto de decepcionarme, cuando vi que en el administrador de conexiones mi conexión estaba detectada y lista para ser configurada. Les dejo la secuencia en imágenes, de todas formas creo que esto seguramente será mejorado en la versión final.





Otra cosa a destacar, en las 48 horas que llevo usándolo intensivamente, no he tenido un solo cuelgue que me haya obligado a cerrar la sesión y abrir una nueva, esta beta es muy estable ¡y está muy acabada!. Sin embargo algunos detalles están por cerrarse. La localización al español todavía no está del todo refinada, hay algunas cosas que a pesar de haber bajado el paquete de idiomas, no cambiaron.

Conclusión

Karmic Koala 9.10 beta, pinta para ser la mejor release de Ubuntu hasta la fecha, y recordemos que a Canonical no siempre las cosas le salen bien, 8.04 fué desde mi punto de vista la peor en la historia de la distribución.

Todo lo que me queda por decir es que espero que de acá al lanzamiento, esperemos que lo que hagan sea para mejorar aún más. Dado que si es así Mark Shuttleworth esta vez sí cumplirá con la promesa de tener el mejor sistema operativo y el más fácil de usar.

Les dejo más capturas de esta beta. Espero les haya gustado la revisión. ¡Y a esperar la versión final!

Gustavo Papasergio Tecnotravel 2.0



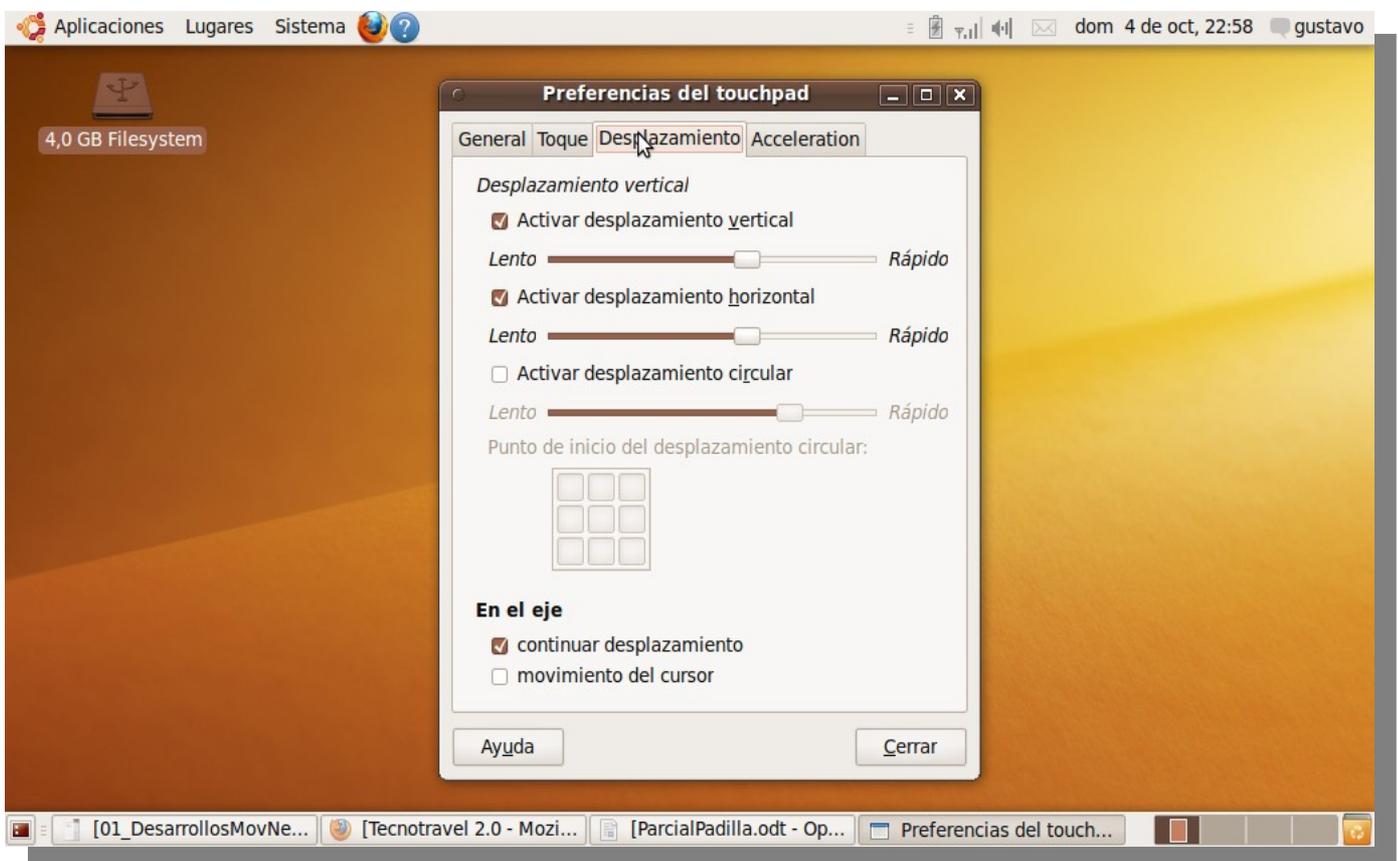
Aplicaciones adobe air

Testea tu sistema y envi a los resultados a launchpad.

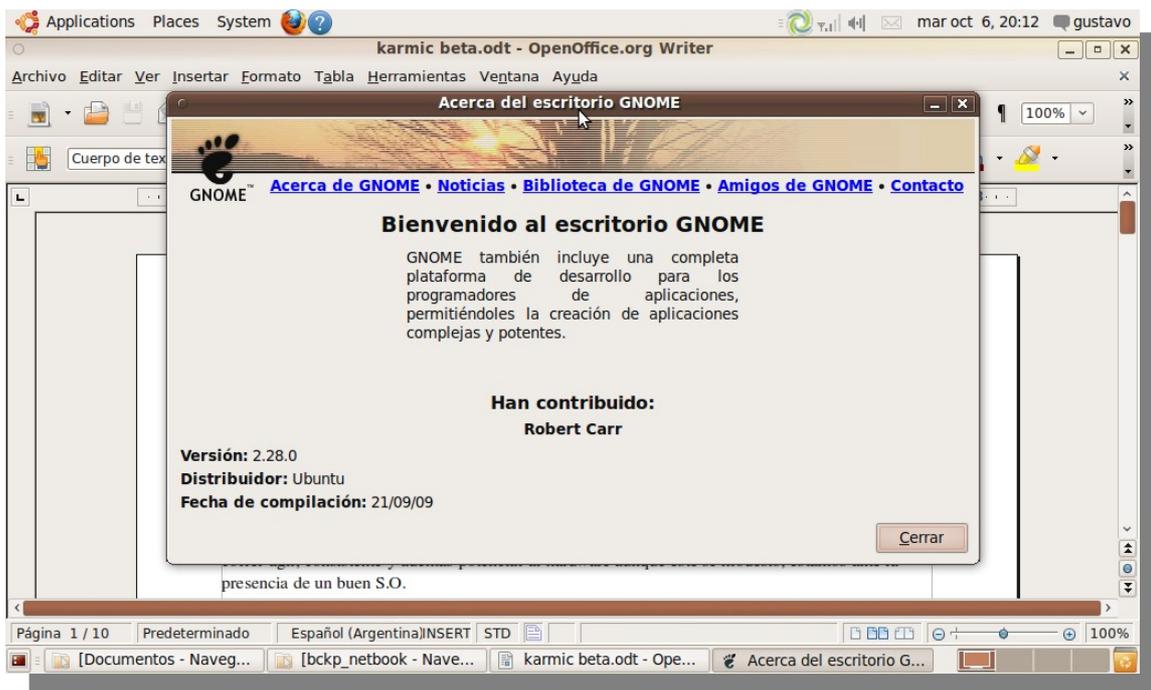
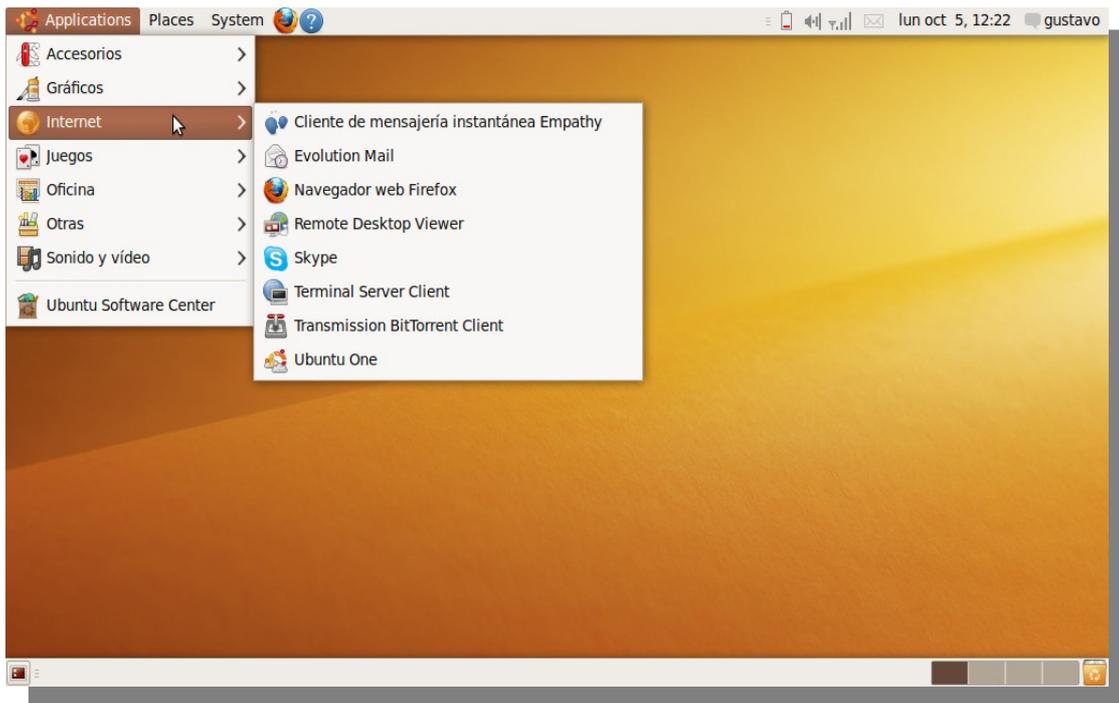




Abriendo OpenOffice



Retocando las opciones del touchpad.



La incomprendida Libertad número 3.

Una de las cosas que normalmente sabemos es que cuando se acepta un contrato hay derechos y obligaciones. Las licencias de software son un tipo de contrato entre el creador/desarrollador de un programa y el usuario/empresa que lo va a usar. Pero cuando una licencia es demasiado rígida (*que tiene más obligaciones que derechos para el usuario*), empieza a ser vulnerada o en un mejor caso el producto es reemplazado por algo más favorable al usuario cuando el mismo se cansa de la misma.

Las 4 libertades de la licencia GPL desde la versión 1.0 dice, (*a grandes rasgos*).

0 – *El usuario tiene la libertad de usar el programa.*

1 – *El usuario tiene la libertad de acceso al código fuente.*

2 – *El usuario tiene la libertad a modificar el código a sus necesidades.*

3 – *El usuario tiene la libertad de distribuir la versión modificada del software.*

Ahora bien, hay algunos usuarios que se encerraron más allá de la letra de la licencia, normalmente conocidos como el ala dura, que se oponen a muchas cosas que pueden llevar a un daño en la evolución del software libre.

Hace tiempo en mi blog subía las grabaciones de conferencias que asistía, con el simple hecho de permitir a aquellos que no pudieron asistir que también la puedan escuchar. Una persona muy sensibilizada me preguntó por e-mail por qué no las subía en formato OGG. Le respondí, (*de la misma forma*), que lamentablemente de los reproductores vendidos en el mercado en los últimos años, el 1% o menos podía reproducir OGG y el 90% reproducían MP3. (*El resto sólo reproducían CD Audio en ese momento*). La charla terminó allí y sin problemas. Hoy día la proporción es mayor en MP3 (casi el 100%), y apenas cambió en OGG, cosa que espero que en el futuro mejore con la adopción del HTML5.

Hace unos meses en un fin de artículo, (*que dejo en algunos casos para dejar pensando al lector*), alguien interpretó mal el mismo creyendo que me refería a otra distro en lugar de a Windows y me tildó de “racista”. Le aclaré amablemente el tema y no hubo más inconvenientes. Desde que decidí tirar a Windows después de determinar que sin el fuente nunca se va a poder solucionar los problemas de seguridad que tiene desde el diseño base. He usado Slackware, Red Hat, Suse, Mandrake, Conectiva, Lycoris, Ututo, Tuquito, Yoper, Debian, Mandriva, DSL, SAM y finalmente me quedé con Ubuntu, (*4.10 en su momento*). Pero siempre estoy viendo que más hay en el mercado y qué se puede mejorar o seguir usando.

Hace poco en una conferencia escuché a alguien decir al pasar -Ese es un Ubuntero Asqueroso-, por suerte no fue a mí, pero me sentí mal por la persona que lo dijo. ¿Por qué?. Porque esa persona no entiende la libertad número 3. Veamos.

La libertad número 2 es esencial para el progreso de los programas y sistemas. Sin ella los usuarios quedarían atados como las licencias OpenSource de Microsoft, (*Ver pero no modificar*). Lo que equivale a “la libertad con fijador”, como dice la Marcha de la Bronca.

La libertad número 3 asegura la pluralidad de implementaciones para satisfacer los requerimientos de los usuarios. Si una distro se basa en otra distro está indicando 2 cosas.

a) La distro es muy buena, b) la misma puede mejorarse. (*Y de allí el despegue de la misma*).

El tratar a un usuario de una distro descendiente en forma despectiva es una forma de no respetar a esa distro, va contra la idea original de la libertad número 3 para fomentar la pluralidad del desarrollo. Si a alguien no le gusta una de las libertades de la licencia GPL, tiene todo el derecho de usar otra licencia, pero si va a abrazar con su corazón la GPL, Nunca se debe discriminar a otra distro ya que de lo contrario se estará contradiciendo. Cada uno tiene derecho a usar la distro que más le agrade por el motivo que sea, Pero si tenemos la libertad de elegir también tenemos la **obligación** de respetar o como mínimo tratar con respeto cuando en algo no estemos de acuerdo.

Claudio De Brasi.
Doldraug@gmail.com
Http://UL64.blogspot.com
@doldraug

PD: Cualquier cosa rígida en exceso tiende a rajarse y luego a romperse. Yo aquí vi una fisura y siempre es mejor arreglarla a esperar que se rompa. ¿O me equivoco?
PD2: Si está enojado, medite un rato antes de contestar. (Ommmmm...Ommmmm...).



Soluciones Service Manager – ITIL Compliance

Se está implementando cada vez más en las empresas la mesa de ayuda o de soporte. Existen gran variedad de herramientas que cumplen desde las funciones básicas hasta las más completas.

Nos concentraremos en lo que pueden llamarse soluciones ITIL compliance, o sea que pueden tener los diferentes roles necesarios y que poseen escalabilidad.

Primero daremos una descripción básica de la función y gestiones que posee el módulo Service Support de ITIL

Service Desk: Extiende el rango de servicios y ofrece una aproximación global, permitiendo que los procesos de negocio se integren en una infraestructura de Service Management.

Gestión de Incidentes: Es una tarea reactiva, ej. reducir o eliminar los efectos de (potenciales) alteraciones en los servicios IT, asegurando que los usuarios puedan volver a trabajar lo más pronto posible. Por esta razón todos los incidentes se registran, clasifican y se asignan a los especialistas adecuados, luego se controlan y por último se resuelven y se cierran.

Gestión de Problemas: Es un proceso de planificación y control para minimizar el impacto de Incidentes y Problemas en los servicios del negocio, causados por *errores* en la infraestructura de IT y prevenir la recurrencia de Incidentes relacionados con estos *errores*.

Gestión de Cambios: Asegurar el uso de procedimientos y métodos estándar en toda el área de IT para el manejo eficiente de los cambios, de manera que se minimice el impacto de los servicios que soportan el negocio.

Gestión de Versiones: Es una visión integral de un cambio a un servicio de IT y deberá asegurar todos los aspectos de una Release, tanto técnicos, como no técnicos, considerados en conjunto. El foco es la protección del ambiente productivo y sus servicios mediante el uso de procedimientos formales y controles.

Gestión de Configuraciones: Facilitar la administración de los componentes, (hardware, software, documentación, etc.) relacionados con los servicios de IT que son críticos para el negocio. Con la intención de proveer a los clientes internos y externos la calidad comprometida de los servicios.

Bien, ahora que ya hemos descripto los requisitos de ITIL pasaremos a la comparativa de herramientas tanto licenciatarias como open source.

Por parte del esquema licenciado nos encontramos (por orden en el mercado):

BMC Remedy: Es una de las mejores soluciones que existen en el mercado tanto por funcionalidad como complementaridad, su costo de licencias es oneroso y soporte y renovación aún más.

HP Service Desk/Service Center/Service Manager: Debo poner todos los nombres porque fue como se desarrolló la evolución de esta herramienta, una muy buena solución pero muy cara para el promedio local.

CA Service Desk Manager: Otra muy buena opción a las anteriores dentro del esquema licenciado

IBM Tivoli Service Manager: No podía faltar, otra empresa histórica con una solución ITIL compliance.

Por parte del esquema open source.

OTRS ITSM: Esta herramienta, es la única ITIL compliance que conozco y hemos implementado en diversos clientes, inclusive compitiendo contra los grandes mencionados anteriormente.

Ahora vamos a pasar a la comparativa basándonos en los siguientes datos

Mesa de ayuda con un total 20 usuarios con diferentes roles

15 efectúan Help Desk, Gestión Incidentes

10 efectúan Gestión de Problemas

5 efectúan Gestión de Cambios

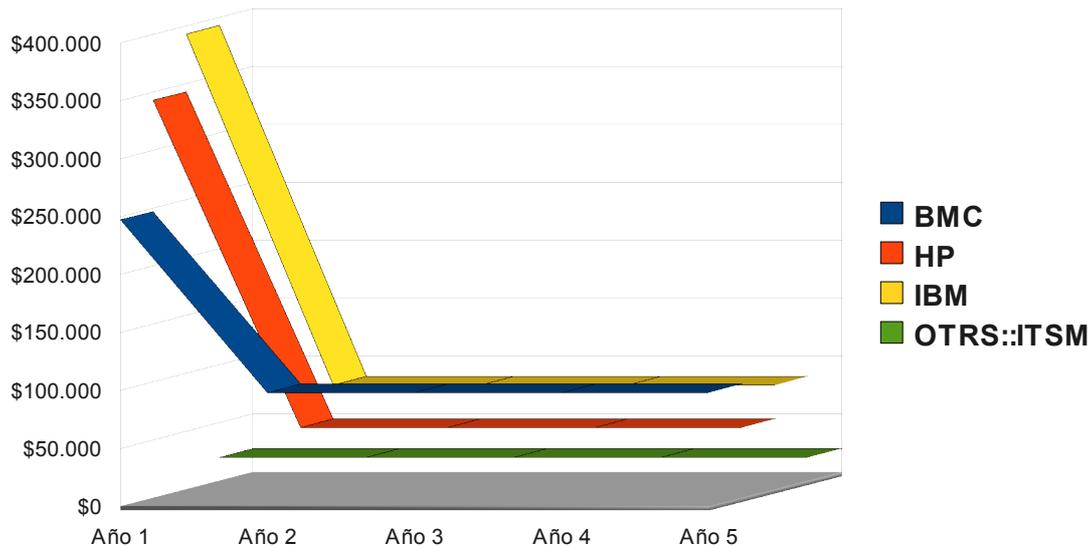
5 efectúan Gestión de Configuraciones

Parque de clientes internos 500 desktop's para la CMDDB (explicada en artículos anteriores)

Catálogo de Servicios, Base de conocimiento, Autoself

Acuerdo de Niveles de Servicio

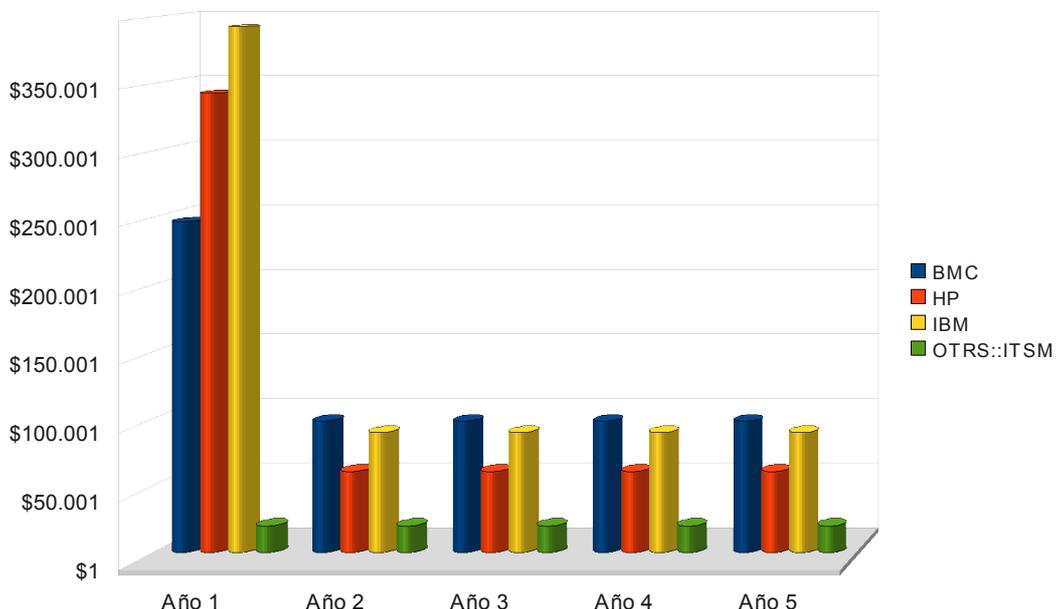
La resultante en forma gráfica nos muestra lo siguiente



Es más que notable la diferencia de costo de inversión inicial que representan los dos esquemas, y si bien con las renovaciones de soporte sigue habiendo un margen muy amplio.

Cabe destacar que OTRS no posee módulo de Gestión de Cambios, pero hay soluciones complementarias que lo efectúan.

El hecho es que para la administración de la mesa de ayuda y sin tener una vasta cantidad de personal, como lo detallado en los datos de base, emerge una brecha más que importante. Que sirve tanto para evaluar hacia qué dirección pretende ir una compañía, o inclusive si desea migrar desde un estado licenciado a uno abierto.



Analizando objetivamente las soluciones, notamos que la de mayor costo es la provista por IBM, le siguen la de HP, BMC y OTRS.

Pero si nos enfocamos a un mediano plazo el resultado nos muestra que dentro del licenciamiento, lo que en un principio pudo haber sido una decisión acertada, a mediano/largo plazo pudo haber sido más oneroso que cualquier otra, ya que el soporte y renovación de las licencias suman mucho más que la de las competencia (sin incluir a OTRS en estas discusión). Por eso es más que sugerible no optar por herramientas sin un completo análisis de impacto de necesidad a corto plazo, sino también económico/financiero a mediano/largo plazo.

Este tipo de software es prácticamente inservible si en una empresa no están definidos roles, sla, procesos, procedimientos, etc. El software no es una solución mágica a nuestros problemas, ni tampoco la culpa de nuestros males.

Fuente

BMC

HP

CA

IBM

OTRS

IT Monitoring

<http://www.itmonitoring.com.ar/>

<http://www.itmonitoring.com.ar/>

Sebastián Osterc

sosterc(arroba)gmail.com



InfosertecTV

www.infosertec.com.ar

Curso de Python. Programación estructurada.

La programación estructurada es un paradigma promovido por figuras como *Charles Anthony Richard Hoare*, *Donald E. Knuth* y *Edsger W. Dijkstra*; En este caso se organizan los programas en forma de sentencias jerárquicas de control de flujo (secuencial, condicional y repetitivo).



Figura 1: Figuras relevantes de la programación estructurada.

Los innumerables aportes y fricciones entre estos tres caballeros impulsaron las principales ramas de las ciencias de la computación. Por solo citar ejemplos de su vasta obra, el primero formalizó una teoría conocida como *CSP* que es útil para demostrar que un programa es correcto, siendo la base de lenguajes modernos como *Eiffel*. Mientras, *D. E. Knuth* es reconocido por su serie *The Art of Computer Programming* donde analiza muchísimos algoritmos muy útiles. Por su parte *E. W. Dijkstra* se destaca por participar en el diseño de *Algol 60* (i.e. el primer lenguaje estructurado); por su activa resistencia al uso de los saltos incondicionales (e.g. la instrucción **goto**); por el famoso problema de los *cinco filósofos* para introducir los bloqueos de los procesos (*deadlocks*) y por el algoritmo para encontrar el camino más corto en un grafo, tema muy popular en el enrutamiento de paquetes en *Internet* y otras ramas. Son sólo ejemplos puntuales. Realmente no alcanzarían las páginas de la revista para hacer una reseña de la vida y obra de estos científicos. Por tanto continúo.

Otro rasgo importante de la programación estructurada es que el código para tareas afines se organiza en grupos mayores. De esta manera se puede reutilizar rápidamente varias funcionalidades. Como resultado se facilita la creación de nuevos programas a partir de soluciones existentes, con el consecuente decrecimiento del tiempo de desarrollo y un mantenimiento más sencillo. *Python* no podía hacer caso omiso a estas ventajas. Por tanto ofrece la posibilidad de declarar módulos y funciones.

🔗 Módulos y paquetes.

Los módulos son el principal mecanismo de reutilización disponible en *Python*. Su rol es agrupar el código y los objetos que tienen un nexo lógico muy fuerte. Por tanto estos objetos son usualmente utilizados en conjunto para resolver problemas parecidos. Hay varios tipos de módulos. El más importante es el módulo *built-in* (`__builtins__`). En él se incluyen todos los objetos básicos del lenguaje, los que usa el intérprete, y funciones indispensables; es decir, el mínimo de funcionalidad necesaria para ejecutar un script de *Python*. Sin embargo, esto no es suficiente para construir aplicaciones más complejas. Es por eso que existen módulos adicionales que permiten disímiles tareas. Estos pueden ser de dos tipos: los módulos de extensión y los script de *Python*.

El código de *Python* se ejecuta en una máquina virtual (específica para cada contexto de ejecución). Los módulos de extensión no son más que aquellos que están codificados en otro lenguaje y se conectan «dinámicamente» para proveer cierta funcionalidad. El lenguaje utilizado para construir el intérprete determina las características del módulo de extensión. *CPython* ofrece la *API Python/C*. Ella permite que librerías escritas en *C* y *C++* sean reutilizadas por los programadores de *Python*. Existen otras herramientas que agilizan la integración con *C++* (*SCXX*, *CXX* y *Boost*), *Fortran* (*PyFort*, *F2PY*), *VisualBasic* y *COM* (*win32all*). Si se desea utilizar desde *Python* una librería ya compilada, entonces *SWIG* hace todo el trabajo a partir de sus ficheros *.h*. Este último ha sido el enfoque utilizado, por ejemplo, por *wxPython* para ofrecer acceso a las librerías del popular producto *wxWidgets*, y así construir interfaces de usuario. Otro módulo llamado *ctypes*, permite invocar desde el intérprete las funciones exportadas por *DLLs* (*Windows*) y ficheros de extensión *SO* (*Solaris*, *Linux*). Por otra parte *Python/C* también permite utilizar el intérprete de *Python* como un componente de una aplicación mayor (embedding) implementada en *C/C++*.



Figura 1: Autores de intérpretes y la arquitectura de módulos para Python.

Jython es otro intérprete que permite que todos los paquetes de *Java* y clases declarados en la variable *CLASSPATH* (librerías estándares, comerciales y extensiones *JNI*), estén automáticamente disponibles para los scripts de *Python*. Además, el compilador **jythonc** convierte el código fuente de *Python* en código de bytes de *Java* que puede ser posteriormente importado por nuevas clases de *Java*. En el caso del intérprete *IronPython* el módulo **clr** permite acceder a las clases del framework *.NET*. Además el namespace **IronPython** permite utilizar desde *.NET* los módulos escritos en *Python*. En todos los casos se puede utilizar la variable **sys.path** (una lista ;o) para añadir, en tiempo de ejecución, más rutas dónde existan módulos que puedan ser cargados dinámicamente por el intérprete.

Por ejemplo, en la Figura 3 se muestra todos los pasos que hay realizar para realizar desde *Jython* una simulación basada en eventos con ayuda de la librería de simulación *Stochastic Simulation in Java* (aka. *SSJ*). Las dos primeras líneas son útiles, muy frecuentes y por tanto recomendadas. La primera indica que este fichero es un script que debe ser ejecutado con el intérprete de *Python*. La segunda se utiliza para describir la codificación del texto, en este caso *UTF-8*, pero podría ser *latin-1* o cualquiera de las que se han mencionado en artículos previos. Preste atención a las diferentes variantes de la instrucción **import**. En la tercera línea se importa el módulo de *Python* llamado **sys**. Este módulo siempre está disponible y provee acceso a algunas variables y funciones vinculadas directamente con el intérprete y el sistema. La variante empleada implica que todos los elementos incluidos en este módulo están accesibles a través de la variable **sys**.

```
#!/usr/bin/env python
# -*- coding: UTF-8 -*-

import sys
sys.path.append("/path/to/ssj/lib/ssj.jar")
sys.path.append("/path/to/ssj/lib/colt.jar")

from umontreal.iro.lecuyer.simevents import Sim,Event,Accumulate
from umontreal.iro.lecuyer.rng import MRG32k3a
from umontreal.iro.lecuyer.probdist import ExponentialDist
from umontreal.iro.lecuyer.randvar import RandomVariateGen
from umontreal.iro.lecuyer.stat import Tally
```

Figura 3: Sentencias import requeridas para usar SSJ en Jython.

Especialmente `sys.path` es una lista que contiene las rutas donde el intérprete busca los módulos. La misma se inicializa con el contenido de la variable de entorno `PYTHONPATH`, pero en el caso de *Jython*, como ya se mencionó, se adicionan las rutas incluidas en la variable `CLASSPATH`. La segunda y tercera línea muestran cómo añadir los ficheros *jar* en los que se distribuye SSJ y la librería *Colt* del CERN (que es una dependencia) si no estuvieran ya allí. Por tanto las instrucciones siguientes motivan que las clases dentro del paquete `umontreal.iro.lecuyer` (*Java*) sean localizadas en estos ficheros y puestas a disposición del script de Python. ¡Qué fácil verdad! Como en este caso se utiliza la instrucción `from ... import` sólo estarán disponibles las clases incluidas después de la palabra clave `import`. Como se puede constatar, las mismas instrucciones se utilizan para importar módulos de *Python* y, en otros casos, paquetes de *Java* y namespaces de *.NET*. El intérprete es bastante listo como para determinar el tipo de módulo que se desea importar.

Después de tener a mano las funcionalidades básicas que ofrece el intérprete y de acceder al gran cúmulo de librerías y herramientas disponibles hoy... ¡a programar! El código *Python* se coloca normalmente en ficheros de texto con la extensión `.py`. En ellos los programadores implementan las aplicaciones finales, y también los artefactos reutilizables. Para utilizar un componente definido en un módulo, primeramente hay que localizarlo. Actualmente el intérprete de *Python* es capaz de cargar un módulo situado en el sistema local de archivos o dentro de varios tipos de archivos (e.g. **ZIP**, **EGG**, **JAR**, **WAR**, **EAR**), e incluso **en un servidor** en la red local o **en Internet** (módulo `urlimport`). En cualquier caso, como ya vimos se usa la sintaxis `import module as name`. Como consecuencia el intérprete crea un objeto del tipo `module` y le añade los campos `__name__` y `__file__` respectivamente con el nombre del módulo y la ruta del fichero de donde fue cargado. Considerar un módulo como un objeto es una característica que diferencia a *Python* de *Java*, *Object Pascal*, e incluso *Smalltalk*. La posterior ejecución del script va añadiendo al módulo los objetos, funciones, etcétera que el programador desea ofrecer. También es posible cargar directamente los elementos contenidos en un módulo, ya sea uno a la vez (`from module import field1 as name`) o al unísono (`from module import *`). En el último caso, si el módulo contiene en el atributo `__all__` una lista de cadenas de caracteres, sólo se cargan los elementos del módulo con estos nombres. Sin embargo, a diferencia de *Java*, las declaraciones hechas dentro de una clase no se pueden importar directamente.

Hay otros dos usos importantes para los objetos de tipo `module`. Al igual que en *Java*, el código de *Python* se puede organizar de manera jerárquica con ayuda de paquetes (carpetas en el sistema local de archivos). La diferencia con respecto a *Java* es que un paquete ya cargado en *Python* se representa también con un objeto `module`. La carpeta correspondiente tiene que incluir un archivo llamado `__init__.py`, que se ejecuta en el momento en que se importa el paquete

Funciones en Python.

Además de los módulos, el lenguaje permite la declaración de funciones que aceptan parámetros. La palabra clave **def** inicia la definición de una función. Luego se especifica el nombre de la función y, entre paréntesis, sus parámetros. Las instrucciones en el cuerpo de la función comienzan en la línea siguiente y separadas por una sangría adicional. En el cuerpo de la función se puede retornar un valor con la ayuda de la instrucción **return**. Si la ejecución llega al final de una función, se retorna el valor **None** (i.e. nada fue retornado ;o).

| | |
|--|--|
| <pre>>>> def fib(n=0): ... a, b = 1, 1 ... for _ in xrange(n - 1): ... a, b = b, a + b ... return b >>> fib <function fib at 0x00AF15F0> >>> fib.func_name 'fib' >>> fib(), fib(0), fib(1) (1, 1, 1) >>> fib(n=2), fib(n=3), fib(n=4) (2, 3, 5) >>> fib(5), fib(n=6), fib(7) (8, 13, 21) >>> fib(n=8) 34 >>> fib(9) 55</pre> | <pre>>>> def format_msg(msg, *args, **kwargs): ... if args != (): ... return msg % args ... elif kwargs != {}: ... return msg % kwargs ... else: ... return msg >>> format_msg(msg="Hola a todos !") Hola a todos ! >>> format_msg("Hola %s !", "Olemis") Hola Olemis ! >>> >>> attrs = {'n': 'Olemis', 'a': 'Lang'} >>> m = "Hola %(n)s %(a)s !" >>> format_msg(m, n='Olemis', a='Lang') Hola Olemis Lang ! >>> format_msg(m, **attrs) Hola Olemis Lang !</pre> |
| a) Números de Fibonacci | b) Argumentos posicionales y por nombre |

Figura 4: Funciones en Python.

En las llamadas comunes como `mi_funcion(1, 2.0, 3, 'b')` se dice que se pasan los argumentos *por posición*. Una característica muy útil es la posibilidad de declarar valores predeterminados. De esta forma la función puede ser llamada con menos argumentos. En un caso como ese las variables que faltan tomarían los valores predeterminados. La Figura 4 muestra también las llamadas a una función utilizando los nombres de los argumentos. En este caso se dice que se pasan *por nombre*. La función puede tener un parámetro que comience con un asterisco (e.g. `*args` en la Figura 4b). Si hubiera un número excesivo de argumentos por posición, los que sobren se le asignan a esta variable (e.g. `args`) empaquetados en una tupla. La función puede tener un parámetro que comience con dos asteriscos (e.g. `**kwargs` en la Figura 4b). Si hubiera un número excesivo de argumentos por nombre en la llamada a la función, los que sobren se le asignan a esta variable (e.g. `kwargs`) empaquetados en un diccionario. La última llamada de la Figura 4b ilustra lo que conoce como *desempaquetamiento* de argumentos. En este caso todos los argumentos se incluyen previamente ya sea en una secuencia o en un diccionario, y en la llamada se antepone uno o dos asteriscos al valor en cuestión.

Para aquellos que todavía piensan que la frase **todo es un objeto** es demasiado, téngase en cuenta que el compilador convierte las funciones en un objeto que contiene varios atributos interesantes. Esto es una ventaja sobre los punteros a funciones (*Object Pascal*, *C*) y otros mecanismos de meta-programación (*Reflection* en *Java*, *RTTI* en *Object Pascal*). Pero esto no es todo. Su código compilado y la pila de ejecución también son representados a través de otros objetos estándares. De todo esto iremos hablando a lo largo de este curso, por tanto: **¡Continúe con nosotros hasta el final!**

=|;o)x-

Conclusiones.

Al programar en *Python* se pueden utilizar ficheros de texto con extensión *.py* para escribir módulos. El código dentro de estos módulos se puede reutilizar haciendo uso de alguna de las variantes de la instrucción **import**. Los mecanismos de importación del lenguaje son tan sólidos y transparentes que se utilizan **sin variaciones** para cargar clases de *Java* (*Jython*) o *.NET* (*IronPython*), ubicando automáticamente el código directamente en el sistema de archivos o dentro de ficheros (e.g. **ZIP**, **EGG**, **JAR**, **WAR**, **EAR**) o incluso en Internet a través del protocolo *HTTP* (e.g. **urlimport**). Grupos de módulos relacionados se pueden agrupar (e.g. para distribuirlos con facilidad) en paquetes. Estos no son más que carpetas que incluyen un fichero *__init__.py*. Para el intérprete no existe diferencia alguna entre un paquete y un módulo. Por tanto ambos reciben un tratamiento uniforme.

Las funciones son otro mecanismo de reutilización de código. Ellas aceptan parámetros tanto por posición como por nombre. En ambos casos el lenguaje ofrece alternativas para manejar uniformemente tanto el déficit (e.g. valores predeterminados) como el exceso (e.g. empaquetamiento) de parámetros en una llamada. También se puede utilizar el *desempaquetamiento* de argumentos en caso que no se conozca de antemano el número de parámetros que recibirá la función. Las declaraciones de las funciones comienzan con la palabra clave **def**, mientras que se utiliza **return** para retornar valores.

El lenguaje posee mecanismos de meta-programación más eficientes que otros lenguajes muy populares, y de ahí se derivan ventajas que mencionaremos más adelante en esta serie. **¡Siga con nosotros!** El próximo artículo estará dedicado a las clases en *Python*, y para ello abordaremos un ejemplo simple pero interesante. **¡No se lo pierda!**

Olemis Lang

olemis@gmail.com

Blog ES : <http://simelo-es.blogspot.com>

Blog EN : <http://simelo-en.blogspot.com>



Partner Oficial

Libertya
Software Libre de Gestión

disytel
OpenSource for Management

www.disytel.com

Rincón del Lector

En esta sección ingresamos todos los mails que se reciben a nuestra cuenta de email info@tuxinfo.com.ar. Si alguno no es publicado, es sólo por razones de espacio, de cualquier manera todos reciben respuesta.

IMPORTANTE. Los emails recibidos a nuestra cuenta electrónica no son editados ni corregidos como tampoco censurados tanto en la revista (rincón del lector) como tampoco en el sitio web www.tuxinfo.com.ar bajo comentarios. Únicamente se reemplazarán por "..." las palabras que puedan ser ofensivas.

Ariel M. Corgatelli

Como has estado amigo!!!! Saludos desde Costa Rica, quiero decirte que ya se puede descargar TuxInfo 20 desde <http://www.anunciate-cr.com/>. Que estés bien!!!!

amigaso!!!!

queria hacerte una consulta, para una maquinita k5, con hd 1gb, disketera de 3/2, 128 de ram!!!! que Distro me podes recomendar, si es posible basada en Debian!!!

te digo que esta maquina le hace frente core i7!!!! jajaj!!!! agradecere tu respuesta!!!

muchas gracias y saludos!!!

Rodolfo!!!!

Rodolfo, justamente este mes en la sección laboexpress vas a tener la solución detallada. La misma es Puppy Linux.
Saludos, Ariel.

Errata en Tuxinfo número 20

No sólo la sección rincón del lector sirve para hacernos consultas, sino que también para corregirnos cuando nos equivocamos en algo. Muchas gracias Quique.

Hola Ariel:

Te escribo con una carta para El Rincón del Lector y unos artículos para que consideres su publicación (uno de ellos es una traducción pero cuento con el permiso escrito del autor).

Carta de lectores:

Estimado Ariel:

Te escribo para corregir una información que salió publicada en TuxInfo N° 20.

En la sección "Noticias del mes", bajo el título de "Linux cumple 18 años", en su segundo párrafo, dice: "Ese sistema operativo, fue en un principio sólo un kernel o núcleo".

Quería aclarar para los lectores noveles --y porque el desliz no se puede dejar pasar-- que en realidad sucedió exactamente al revés.

La redacción correcta sería así: "El sistema operativo que hoy conocemos con el nombre de "GNU/Linux", en el principio (1983) fue un conjunto de programas llevado adelante por Richard Stallman pero sin el núcleo/kernel. Gracias a Linus Torvalds, seis años más tarde (1991) se le incorporó lo que faltaba: el núcleo/kernel llamado "Linux". De ahí el verdadero nombre del sistema operativo: "GNU/Linux" o "GNU + Linux" o también "GNU con Linux"."
Por otra parte, al final de la misma nota hay otra afirmación que es incorrecta justamente por lo que acabo de explicar. Dice así: "Obviamente la (fecha) más importante es la que festejamos hoy, ¡Felices 18!", en relación a los 18 años que cumple "Linux" (el núcleo) en la fecha en que se redactó la nota: 26 de agosto.

Lo correcto es afirmar que la fecha más importante para el sistema operativo "GNU/Linux" es el 27-09-1983, cuando Richard Stallman anunció el proyecto GNU, y la posterior publicación del Manifiesto GNU en marzo de 1985.

También cabe aclarar que en un principio el núcleo "Linux" era libre (bajo licencia GPL) pero ya no es así en todos los casos. Por este motivo, entre otros, hoy en día, no todas las distribuciones de GNU/Linux son libres, sólo aquellas aprobadas por la Free Software Foundation, que aparecen en esta web: <http://www.gnu.org/distros/free-distros.html> .

Sobre las razones de este particular he escrito un breve artículo que los interesados (todos, creo yo) pueden consultar: ¿Usás una distro libre?

Sin más, me despido con un abrazo y las felicitaciones de siempre para vos y el equipo de TuxInfo.

Quique Marzo.

(Paraná, Entre Ríos, Argentina)

<http://www.inquietamente.com.ar>

Andres Gregori

Bueno, gracias de todas maneras por el excelente trabajo que hacen. hace muy poco que descubri Tux INFO , (gracias a taringa), asi que de a poquito estoy comenzando a leer sus articulos. muchas gracias nuevamente y un gran saludo desde bahia blanca Andres.

No hay por qué agradecer y desde ahora te invito a que descargues todas las revistas desde nuestro sitio oficial www.tuxinfo.com.ar en donde vas a encontrar los enlaces directos a cada revista.

Saludos, Ariel.



www.tengasudominio.com



Primera entidad
acreditada por ICANN
en Hispanoamérica

Red Hat presentó Red Hat Enterprise Linux 5.4 en su Summit 2009

La compañía anunció la disponibilidad de la última edición de su plataforma que incluye mejores capacidades de virtualización, una mayor velocidad de procesamiento y herramientas adicionales para desarrolladores.

Chicago, EE.UU. Red Hat, Inc. (NYSE: RHT), el proveedor líder mundial de soluciones de código abierto, anunció hoy la disponibilidad de la cuarta actualización de su plataforma Red Hat Enterprise Linux 5, que provee mejoras tecnológicas a los clientes a través del modelo de suscripción que ofrece la compañía. Red Hat Enterprise Linux 5.4 constituye el cimiento de la cartera de soluciones Red Hat Enterprise Virtualization. Este anuncio se realizó en el marco del Red Hat Summit 2009, evento que cada año convoca a cientos de desarrolladores, clientes y partners de la industria durante 3 días.



Red Hat Enterprise Linux 5.4 es una plataforma completa que incluye sistema operativo y virtualización. Esta versión incorpora Red Hat Enterprise Virtualization para la administración de virtualización en servidores y desktops, y la tecnología KVM (Kernel Virtual Machine) y VDI (Virtual Desktop Infrastructure).

“Red Hat Enterprise Linux juega un papel muy importante en la estrategia de virtualización de Red Hat. La disponibilidad de Red Hat Enterprise Linux 5.4 con la misma base tecnológica de virtualización que Red Hat Enterprise Virtualization constituye hoy un paso significativo en nuestra oferta al mercado”, comentó Scott Crenshaw, vicepresidente de la Unidad de Negocio Plataformas de Red Hat. “Este lanzamiento también incluye avances en rendimiento, seguridad y almacenamiento tanto en entornos virtuales como físicos. Con esta actualización, Red Hat Enterprise Linux apunta a elevar el nivel una vez más, ofreciendo tecnologías de software atractivas con una calidad y confiabilidad sorprendentes”.

Red Hat Enterprise Linux 5.4 ya se encuentra disponible a nivel mundial y se entrega automáticamente a los clientes que posean una suscripción activa.

Un evento que crece cada año

El Red Hat Summit 2009 presentó en la ciudad de Chicago las últimas novedades de la compañía en términos de open source, cloud computing y virtualización. Más de 1500 personas participaron en la apertura de las conferencias de Jim Whitehurst, CEO de Red Hat y Paul Cormier, vicepresidente ejecutivo y presidente de productos y tecnología.

Jim Whitehurst dio su visión de la industria tecnológica de hoy y comentó que: “El open source permite crear mejor software, pero nuestro negocio no se basa en los bits. Lo que hace exitoso nuestro modelo es lo que hacemos para que esos bits estén listos para aplicaciones de misión crítica, del ecosistema que hemos creado alrededor de ellas y de la visión abierta que hemos desarrollado con nuestros clientes”.

Por su parte, Paul Cormier dio algunos números de la posición de Red Hat en el mercado. “Con respecto a sistemas operativos, Red Hat posee entre el 60 y el 90% de market share en servidores pagos de Linux, dependiendo de la geografía y las aplicaciones. En cuanto a middleware, el 30% de sus usuarios en el mundo tienen desarrollos JBoss de Red Hat”.

Durante el evento se presentaron asimismo casos destacados de empresas como la Bolsa de Comercio de Chicago (CME Group), entidad que procesa millones de transacciones por día. CME Group migró de una plataforma Unix propietaria a Red Hat Enterprise Linux, reduciendo costos en un 50% y aumentando el desempeño, confiabilidad, escalabilidad y agilidad de estos sistemas que manejan operaciones de misión crítica. La Bolsa de Chicago se suma a la de Nueva York (NYSE Euronext) que desde este año también utiliza software open source de Red Hat.

También participó Derek Chan, Jefe de Operaciones Digitales de la compañía DreamWorks Animation, quien disertó sobre el uso de la computación en la nube. “El software de código abierto ha sido un posibilitador clave de toda la infraestructura de computación de la cual dependemos para crear nuestras películas. Observamos una convergencia natural de las plataformas basadas en el código abierto y la computación en nube que beneficiará enormemente a la computación de empresa”, comentó Chan.

Desde sus comienzos con “Antz”, “Shrek2” y “Madagascar” hasta los próximos largometrajes en 3D como “Cómo entrenar a tu dragón” y “Shrek para siempre”, las tecnologías de computación en nube son fundamentales en la provisión de los recursos informáticos necesarios para crear estas películas.

Otros anuncios

Durante el Summit, Red Hat presentó novedades en otros frentes:

JBoss Enterprise Application Platform 5.0

La última investigación que realizó IDC predice que el gasto en servicios de nube informática estará cerca de triplicarse, alcanzando los USD 42.000 millones en el año 2012, y captando un 25% del crecimiento del gasto de TI en 2012, y casi un tercio del crecimiento en el siguiente año. Asimismo IDC señala que los proveedores exitosos deberán enfrentar los mayores desafíos de los servicios en nube así como también los mayores problemas tradicionales del usuario informático.

En este contexto, Red Hat presentó JBoss Enterprise Application Platform 5.0. Este producto de middleware representa las plataformas de aplicaciones Java de próxima generación y formará parte del fundamento de la computación en nube de Red Hat.

“No sólo hemos cumplido con nuestra estrategia JBoss Open Choice al lanzar JBoss Enterprise Application Platform 5.0, sino que estamos contribuyendo al futuro de las plataformas de aplicaciones Java desde entornos tradicionales hacia entornos en nube”, comentó Craig Muzilla, vicepresidente de middleware en Red Hat. “En última instancia, apuntamos a brindar a los clientes la flexibilidad de elegir los modelos de programación e implantación adecuados para mantenerse a la vanguardia de la competencia”.

Red Hat Network Satellite 5.3

La virtualización constituye un requerimiento clave en muchos entornos informáticos de hoy en día. Con Red Hat Network Satellite, el administrador de sistemas sólo necesita una única consola para actualizar, proveer y controlar sus sistemas Red Hat Enterprise Linux tanto físicos como virtuales.

Este último lanzamiento de Red Hat Network Satellite ofrece mayores beneficios a los clientes empresariales, como oportunidades de mejorar la productividad, reducir los gastos operativos y acelerar la implementación en servidores tanto físicos como virtuales.

Red Hat Network Satellite 5.3 ya está disponible a nivel mundial y se entrega en forma automática a los clientes que posean una suscripción de Red Hat Network Satellite.

Acerca de Red Hat, Inc.
<http://www.latam.redhat.com>

Road Show 2009 Argentina, Conferencia de prensa

TuxInfo estuvo presente en la conferencia de Prensa que brindó la empresa Red Hat, la cual se llevó a cabo en el Hotel Meliá en Bs. As.



En ella estuvieron presentes Martín D'elia, (Gerente de Marketing para Latinoamérica) y Germán Soracco (Sales Manager para Sudamérica y Andino de Red Hat).



La primer parte de la charla fue realizada por Martín D'elia, quien nos contó que la Compañía se encuentra en un absoluto crecimiento. Se duplicaron la cantidad de empleados, en su totalidad hay 120 personas en todo América latina. De los cuales en Argentina existen 40, en Brasil 70, y luego se reparten entre Chile, México y Colombia. Hablando a nivel global los empleados son aproximadamente 2.200 y ascendieron a 3.300 personas en 1 año.

Unos de los centros de soporte se encuentra en Argentina con nuevas oficinas, la cual es considerada la 2da oficina mas importante de Latinoamérica.

"Nos encontramos frente a productos y clientes nuevos, así como competidores nuevos también. Y los antiguos de siempre, Microsoft, IBM y Oracle". Agrego Martín



Hablando específicamente de la compañía, esta tuvo un crecimiento en segmentos, uno de ellos es la llegada de sus productos a nivel gobierno, Nacional y Provincial, el ejemplo más importante lo vivió la Pcia. de Río Negro con una visión innovadora en cuanto al modelo, ya que están armando una universidad con base en el sistema operativo Red Hat Enterprise Linux, luego el Ministerio de Economía de la Ciudad Autónoma de Bs As migró todo hacia nuestra plataforma. También compañías de Telecomunicaciones y compañías de Retail. Agregaron también que nueve bolsas del mundo corren en Red Hat.

La idea es poder llevar a la Compañía a un crecimiento orgánico, que pueda tener la mayor cantidad de desarrolladores, estos trabajan desde sus casas, no existe un laboratorio especial para ellos, ya que trabajan de forma individual, y desde la comodidad de sus hogares, para luego llevar el proyecto a la empresa.

En la rueda de prensa se les hizo algunas preguntas y quizás la más destacada por la respuesta fue sobre la crisis mundial que tuvo tanta repercusión. La respuesta fue simple: La compañía no sufrió por la crisis mundial, no les afectó, muy por el contrario, la benefició.

Otra de las preguntas fue en lo referente al mercado retail.

Y la respuesta estuvo basada en la gran cantidad de equipamiento con el que se está dotando grandes tiendas de consumo masivo con servicio de Red Hat. Y por supuesto todo lo referido a soluciones GNU/Linux.

Y para cerrar una de las preguntas que le realizamos fue: ¿Les afectó de forma positiva o negativa la salida de software legal hacia la legalización de software privativo?.

Cabe destacar que la excelente respuesta que brindó Martín al respecto, en donde sin entrar en detalles, nos contó que muchos nuevos clientes que se acercaron a la empresa les preguntaban sobre la licencia de sus productos en donde ellos le decían: (DOS PUNTOS) "nosotros no vendemos licencias, brindamos servicios". Con este caso y la respuesta hacia ese cliente podemos afirmar que fue positivo la "persecución" que software legal en Argentina hizo a los usuarios de software privativo no en regla.

Redacción TuxInfo



El día que los pingüinos se asolearon

Ejecución de código nativo GNU/Linux sobre OpenSolaris



En este artículo veremos una suerte de continuación de los anteriores sobre virtualización, pero ahora analizando cómo es que el kernel de OpenSolaris puede ejecutar código originalmente escrito para GNU/Linux dentro de un container.

Introducción

En los artículos anteriores sobre virtualización con OpenSolaris analizamos algunas de las formas de crear containers. También vimos cómo manejar cada aspecto de los recursos utilizados por ellos.

Ahora, nos focalizaremos en una de las tecnologías que se pueden utilizar para ejecutar código nativo de GNU/Linux dentro de máquinas OpenSolaris.

Dado que la tecnología dedicada a la generación de zonas se llama, valga la redundancia, “zones”, y que las GNU/Linux son zonas de marca, se denominará a esta tecnología “**Branded Zones**”, o sencillamente, “**BrandZ**”.

Una de las ventajas que encontraremos en las BrandZ es que dentro de los posibles métodos de implementación tenemos el de ejecutar un sencillo tar en una máquina real con GNU/Linux instalado, y luego utilizar el archivo de backup resultante para generar una copia idéntica de la máquina física original.

Si lo que queremos es instalar una BrandZ desde un CD, DVD, o imagen de cualquiera de ellos, también podremos.

No todas las versiones de GNU/Linux se pueden implementar en forma directa en una BrandZ, dado que como en el caso de las zonas de OpenSolaris, es el mismo kernel el que estará ejecutando las instrucciones de cada binario GNU/Linux. En este artículo veremos cómo agregar distros a la lista de las que aparecen como soportadas desde el momento en el cual instalamos nuestra copia de OpenSolaris, que no son más que algunas cuantas de Red Hat, Centos, etc.

En el caso de las BrandZ los métodos que ya vimos de control de recursos se aplicarán en forma idéntica, por lo que no se abordarán los mismos en este artículo, recomendando que, en el caso de tener alguna duda, nos refiramos a los números anteriores (19 y 20) de esta publicación.

Ahora, manos a la obra.

Algo de arquitectura nunca viene mal

No es mi idea que alguien se estudie la arquitectura del kernel de OpenSolaris, pero veamos cómo es que funciona, a vuelo de pájaro, esta funcionalidad.

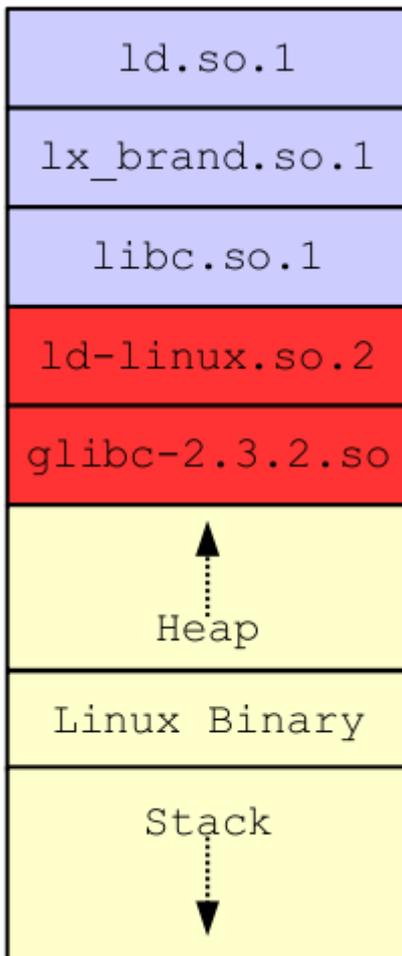
Cuando un proceso es lanzado desde la zona global, o desde una local, el kernel de OpenSolaris lo pasa por un sencillo "if":

```
if (p->p_brand)
    p->p_brand->br_syscall();
else
    rval= do_syscall();
return to userspace
```

Por lo tanto, lo que se intenta analizar en este punto es si hay que utilizar un determinado system call, o no.

Y en el caso de necesitarse ese salto, se hará uso de la denominada "LX emulation library", que posee un contenido similar al siguiente:

```
lx_open(args) {
    fd = open((lx_to_solaris(args))
    if (fd < 0)
        return (solaris_to_lx(errno))
    else
        return (fd);
}
```



Antes de pensar que cometimos un gran error al comenzar a leer este artículo, tengamos en mente que lo que hace el kernel de OpenSolaris para poder implementar las BrandZ es interpretar lo que los programas escritos para GNU/Linux deben hacer, pudiendo entonces ejecutar un ambiente completo mantenido dentro de los límites de un container, y generando sólo un 5% de overhead por sobre el valor normal del proceso. No consume lo que los programas de virtualización, porque, justamente, no se trata de uno de ellos.

Ejemplo de ejecución

de un proceso GNU/Linux

en OpenSolaris

Mi primer BrandZ

Como primer ejemplo de una BrandZ, utilizaremos para su instalación un archivo tar preexistente, que podremos descargar de http://dlc.sun.com/osol/brandz/downloads/centos_fs_image.tar.bz2, y como vemos por el nombre, se trata de una vieja CentOS.

Supondremos que el espacio donde ha quedado descargado este archivo se llama /export/home/usuario/Download/centos_fs_image.tar.bz2, que nuestra BrandZ se llamará "linux01", que la querremos depositar en el path /zones/linux01 de la zona global, que querremos limitar su espacio en disco a 20 GB, y su memoria a 512 MB, con un swap de 1 GB. Desde el punto de vista de la red, le entregaremos la dirección IP 10.100.100.47, con máscara 255.255.255.0, y la misma se montará sobre la interfaz "iprb0".

Primero, tendremos que tener generado el espacio /zones, tal como hicimos en los casos anteriores de zonas.

Es importante saber que en el sistema operativo encontramos varios templates para generar zonas, y uno de ellos es el dedicado a las BrandZ de GNU/Linux con kernel 2.4, como el que trabajaremos en un principio (luego veremos un template para kernel 2.6). Ese template se invoca por el nombre "SUNWlx" desde el prompt del zonecfg.

Ahora, configuraremos la BrandZ:

```
# zonecfg -z linux01
linux01: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:linux01> create -t SUNWlx
zonecfg:linux01> set zonepath=/zones/linux01
zonecfg:linux01> add capped-memory
zonecfg:linux01:capped-memory> set physical=512m
zonecfg:linux01:capped-memory> set swap=1024m
zonecfg:linux01:capped-memory> end
zonecfg:linux01> add net
zonecfg:linux01:net> set address=10.100.100.47/24
zonecfg:linux01:net> set physical=iprb0
zonecfg:linux01:net> end
zonecfg:linux01> add attr
zonecfg:linux01:attr> set name="audio"
zonecfg:linux01:attr> set type=boolean
zonecfg:linux01:attr> set value=true
zonecfg:linux01:attr> end
zonecfg:linux01> verify
zonecfg:linux01> commit
zonecfg:linux01> exit
```

Ahora, deberemos proceder a la instalación en sí misma de la BrandZ GNU/Linux "linux01":

```
# zoneadm -z linux01 install -d
/export/home/usuario/Download/centos_fs_image.tar.bz2
A ZFS file system has been created for this zone.
Installing zone 'linux01' at root directory '/zones/linux01'
from archive '/export/home/usuario/Download/centos_fs_image.tar.bz2'

This process may take several minutes.

Setting up the initial lx brand environment.
System configuration modifications complete.
Setting up the initial lx brand environment.
System configuration modifications complete.

Installation of zone 'linux01' completed successfully.

Details saved to log file:
    "/zones/linux01/root/var/log/linux01.install.1530.log"
```

Sencillo, ¿no? Ahora, sólo deberemos bootear la BrandZ, y listo

```
# zoneadm -z linux01 boot
# zlogin linux01
```

```
[Connected to zone 'linux01' pts/3]
Welcome to your shiny new Linux zone.

    - The root password is 'root'. Please change it immediately.

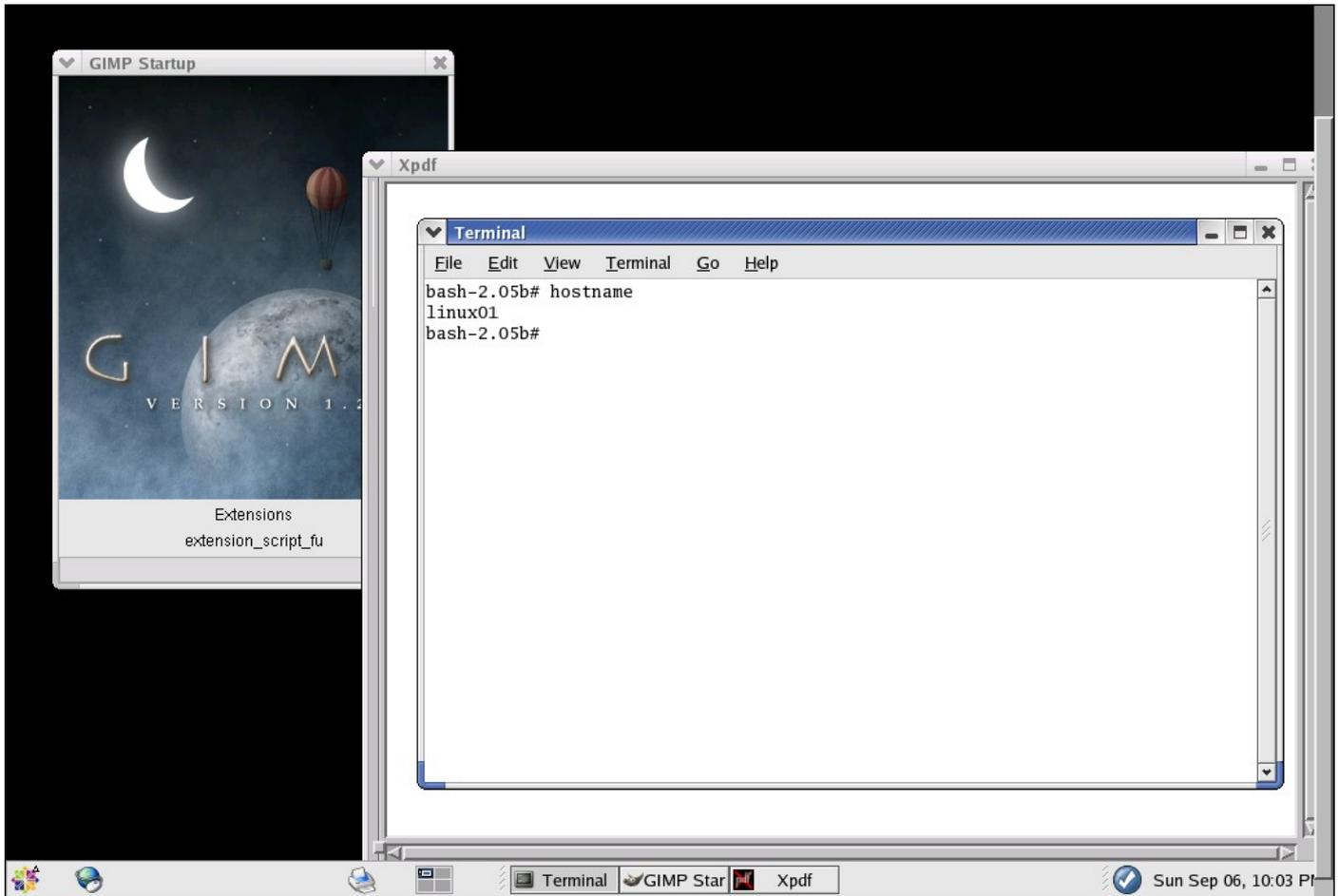
    - To enable networking goodness, see
/etc/sysconfig/network.example.

    - This message is in /etc/motd. Feel free to change it.

For anything more complicated, see:
    http://opensolaris.org/os/community/brandz/

You have mail.
-bash-2.05b# uname -a
Linux linux01 2.4.21 BrandZ fake linux i686 i686 i386 GNU/Linux
```

Ahora que la BrandZ está lista, y funcionando, veremos otras configuraciones más complicadas. Pero antes, y si no querés tener problemas, acordate de agregar una línea con tu dirección IP y tu hostname (linux01) en el archivo /etc/hosts, y de cambiar la clave de "root"!!!.



VNC de Centos corriendo sobre OpenSolaris

Pedile tu 2.6 al kiosquero de tu barrio

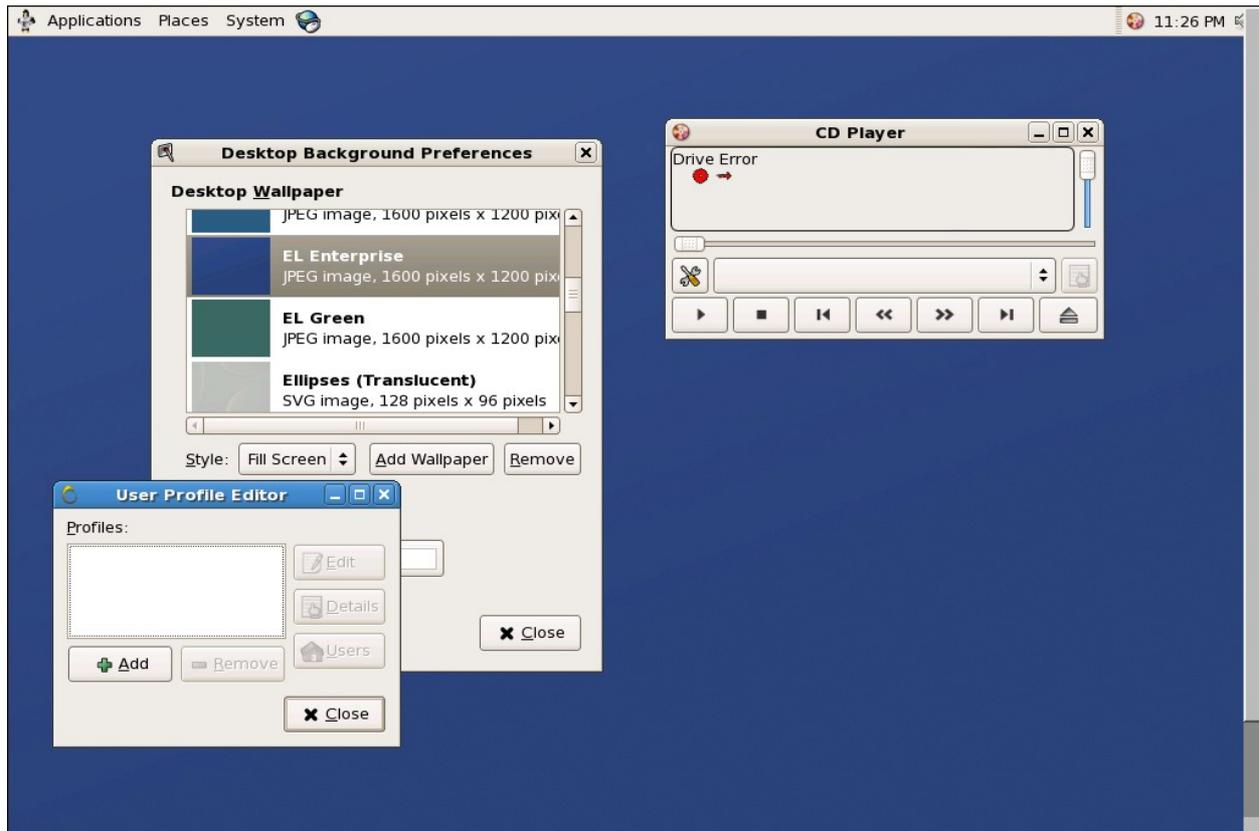
Bien, estamos de acuerdo en que en esta época implementar una BrandZ con un kernel 2.4 es una auténtica antigüedad, y que por lo menos, vamos a querer implementar un kernel 2.6.x. La cuestión es que cuando configuramos nuestro BrandZ, lo hacemos invocando al entorno SUNWlx, que no es más que el representado en un archivo XML, el /etc/zones/SUNWlx.xml. Entonces lo que debemos hacer es crear, o bajar, un nuevo archivo, que nos permita hacer uso de una distribución de GNU/Linux con kernel 2.6.x.

Ese archivo ya existe, se llama SUNWlx26.xml, y se puede bajar de <http://www.opensolaris.org/os/community/brandz/files/SUNWlx26.xml>. Hasta ahora, bastante sencillo.

Ahora no debemos hacer más que configurar una BrandZ de la misma forma que lo hicimos antes para un kernel 2.4.x.

La diferencia radica también en que en este caso deberemos utilizar, en lugar de un archivo .tar.bz2 correspondiente a una distro GNU/Linux con kernel 2.4.x, uno correspondiente a kernel 2.6.x.

He tenido el gusto de probar todas las versiones de GNU/Linux Red Hat, CentOS, Fedora y Oracle Unbreakable Linux como BrandZ, siempre con buenos resultados.



Oracle Unbreakable Linux 5.3 como una BrandZ más.

Ahora bien, ¿podemos decir que todas las distros están soportadas por omisión? No, sólo algunas, y tienen más que ver con las Red Hat y sus derivadas, que con las que generalmente los amantes de GNU/Linux con filosofía realmente OpenSource usan. Aún así, ya hay procedimientos para utilizar BrandZ con Ubuntu y Debian.

Instalando desde discos e iso's

Hasta ahora, hemos instalado nuestras BrandZ desde archivos tipo backup de un servidor que ya tenía implementada una determinada versión de GNU/Linux.

Pero qué pasa cuando no tenemos una máquina ya instalada, y lo que estamos tratando de hacer no es una migración, sino una instalación desde un disco de sistema operativo, o desde una iso que hemos bajado? En este caso, el procedimiento cambia un poco.

Si exploramos el directorio `/usr/lib/brand/lx/distros`, veremos varios archivos cuyo nombre y contenido nos resultará familiar.

Veremos que existen varios con extensión `.distro`, cada uno correspondiente a una distribución de GNU/Linux determinada. ¿Qué significa esto? Que cada una de esas distros está automáticamente soportada para ser instalada desde un disco, o desde un archivo `.iso`. Pero qué pasa cuando nuestra distro no se encuentra entre ellas? Tendremos que modificar un poco un nuevo archivo, cuya extensión será también `.distro`, y cuyo nombre será el correspondiente a la distribución que queremos instalar.

Por ejemplo, supongamos que tenemos versiones de CentOS hasta las 3.8 dentro del directorio mencionado. ¿Cómo noto eso? Porque tengo un archivo llamado `centos38.distro`.

Y supongamos que lo que quiero instalar es una distribución de CentOS 3.9. Ya lo sé, ya lo sé...es un poco vieja, pero vayamos por partes! Primero vamos a aprender cómo hacerlo, luego volaremos y ¡haremos locuras con otras distros!

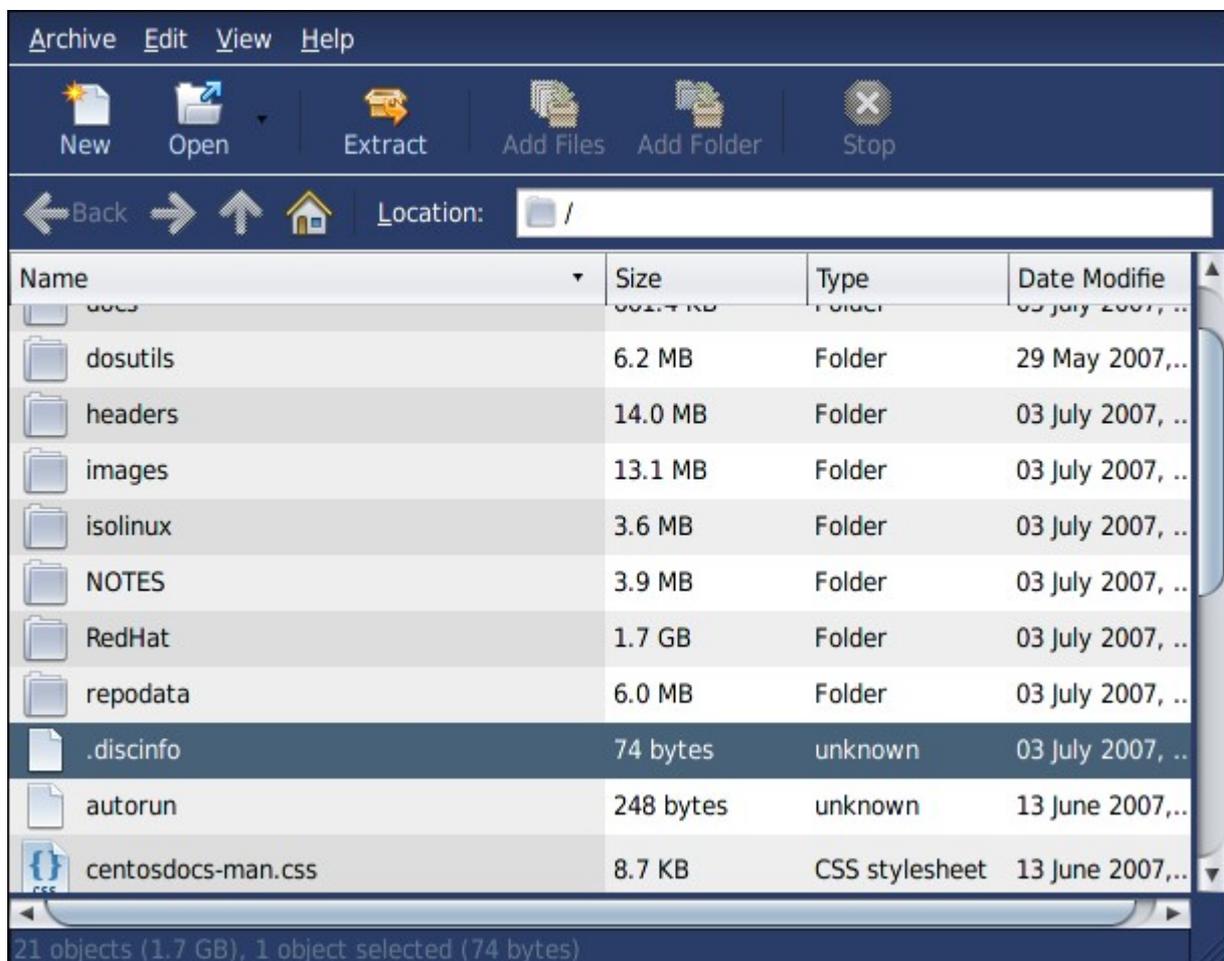
Generaré el archivo centos39.distro, como primera medida. Y ¿cómo es que lo genero? De la manera más sencilla que conozco, es decir, ni más ni menos que con un “cp centos38.distro centos39.distro”. Como siempre, el “copy & paste” viene a ayudarnos a ahorrar tiempo.

Ahora, editamos el archivo con un simple “vi centos39.distro”. ¿Qué es lo que encuentro de particular? Veamos:

```
distro_serial=1155307611.42
distro_version="3.8"
set -A distro_cdorder 1 2 3
distro_mb_required=500
. ${distro_dir}/rhel_centos_common
delta_miniroot_rpms=centos-release
delta_core_rpms="centos-yumconf centos-yumcache yum"
delta_server_rpms="$delta_core_rpms nss_db-compat sendmail-doc qt-
config"
delta_desktop_rpms="$delta_server_rpms \
    expectk \
    seamonkey \
    seamonkey-chat \
...

```

El campo “distro_serial” hace referencia al que encontraremos en el archivo “.discinfo”, en la raíz del DVD de sistema operativo que queremos implementar:



En dicho archivo encontraremos en la primer línea un número que será el que deberemos incorporar al campo “distro_serial” de nuestro nuevo archivo.

```
1183469235.986110
final
i386
1,2,3
RedHat/base
RedHat/RPMS
RedHat/pixmaps
```

Luego, tendremos que cambiar el valor del campo “distro_version” de “3.8” a “3.9”. Con lo anterior, nuestro archivo tendrá este aspecto:

```
distro_serial=1183469235.986110
distro_version="3.9"
set -A distro_cdorder 1 2 3
distro_mb_required=500
. ${distro_dir}/rhel_centos_common
delta_miniroot_rpms=centos-release
delta_core_rpms="centos-yumconf centos-yumcache yum"
delta_server_rpms="$delta_core_rpms nss_db-compat sendmail-doc qt-
config"
delta_desktop_rpms="$delta_server_rpms \
    expectk \
    seamonkey \
...

```

Ahora, podremos configurar nuestra BrandZ como de costumbre. Al finalizar, podremos instalarla agregando, al comando zoneadm el flag “-d” apuntando al directorio donde tengamos montado nuestro DVD, o bien al que posee la imagen .iso de nuestro sistema operativo.

Un tema importante es que en el caso de usar una imagen .iso, en el directorio al cual se apunte no debe haber otra cosa más que esa misma iso.

Suponiendo que le hemos dado a nuestra BrandZ el nombre “centos39”, el comando, en nuestro caso, quedaría más o menos así: “zoneadm -z centos39 install -d /path/a/la/imagen”.

Conclusión

Hemos visto cómo implementar un container con GNU/Linux tanto desde un archivo de backup como desde un disco de sistema operativo, o desde una imagen iso.

¿Qué es lo interesante de este esquema?

Que si queremos consolidar servidores sin demasiado esfuerzo, o si no contamos con el 100% de la documentación sobre todas las personalizaciones que se han realizado en un sistema operativo GNU/Linux ya instalado, o inclusive si ya no tenemos en nuestro poder los medios de instalación de un determinado servidor (aunque se arranquen las vestiduras les cuento que hay algunas personas, herejes, por supuesto, que no siguen al pié de la letra lo que nuestro amo, ITIL, nos ha propuesto, y por lo tanto no tienen una biblioteca de software como debiera...no sé cómo es que pasan esas cosas...), lo podremos hacer.

Sólo tomar un backup de un servidor existente y recrearlo como un container basado en la tecnología BrandZ nos permitirá tenerlo funcionando de nuevo.

Por otro lado, si queremos obtener de un ambiente de producción uno de desarrollo, o de test que sea 100% igual al original, podremos hacerlo siguiendo los procedimientos que antes vimos. Y finalmente, si queremos armar un esquema de aprovisionamiento on-demand de servidores (alguien por algún lado escuchó un murmullo? Hasta me pareció escuchar “Cloud Computing”, pero puedo estar equivocado...), no tiene más que armar una “Gold Copy” de un sistema operativo configurado, y hacer un deploy del mismo en cuestión de minutos.

Teniendo ya buenas nociones de virtualización sobre ambientes OpenSolaris, en las próximas entregas estaremos focalizándonos en un esquema de virtualización de redes en OpenSolaris, para finalmente meternos de lleno en Cloud Computing, sin quedar colgados de la nube.

¡Hasta la próxima!

Hernán “HeCSa” Saltiel
hsaltiel@gmail.com
<http://www.aosug.com.ar>

PUPPY LINUX

Cuántos de nosotros tenemos una vieja PC guardada en la bodega, la cual por obvias razones jubilamos, crees que ese equipo ya no tiene vida, prepárate para resucitarla y darle una segunda oportunidad con Puppy Linux

Muchas de las ventajas que tiene el sistema linux es la adaptación a cualquier tipo de hardware por muy obsoleto que sea, y digo obsoleto porque la distro que hoy presento puede correr en un olvidado 486, así es señores conozcamos al cachorro más veloz, bienvenidos.

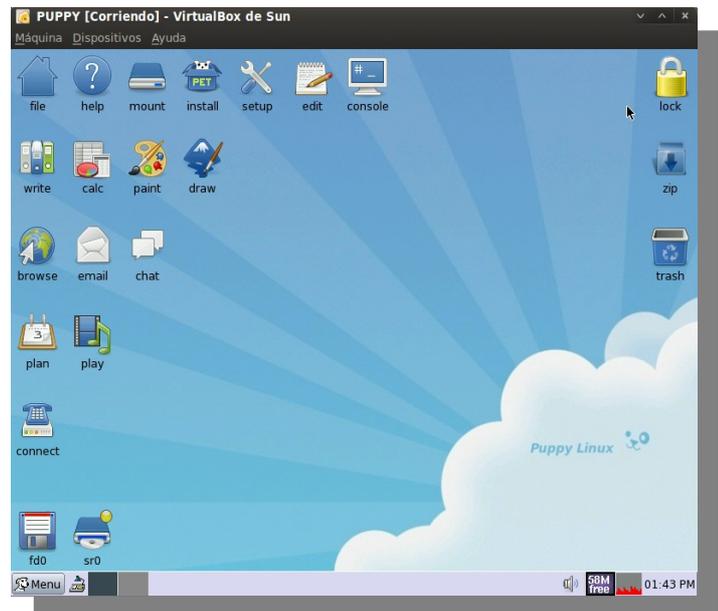
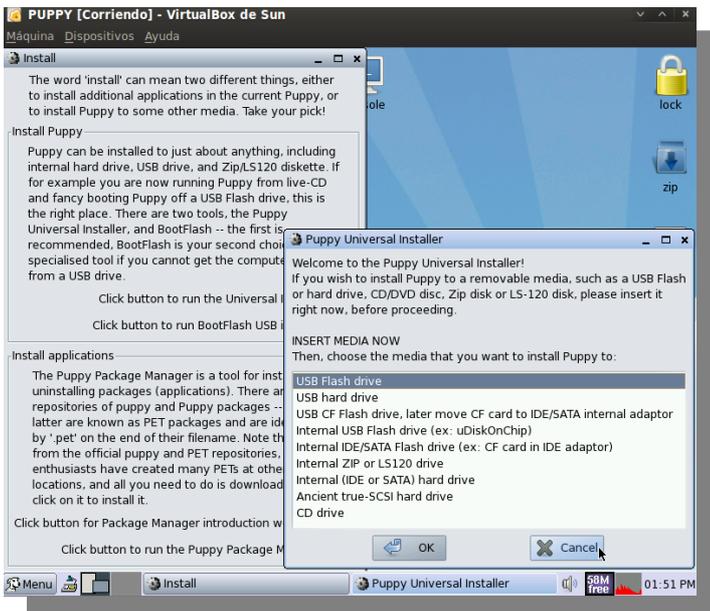
Es una distribución linux que se ejecuta desde un live cd , con la ventaja de instalarse en nuestro disco duro y con la opción de ejecutarse también desde un pendrive, con tan solo 105 megas en su versión más reciente, la 4.3, nos brinda toda una gama de aplicaciones para el uso diario, entre ellas encontramos el ya conocido procesador de textos Abiword, seamonkey para navegar por internet el cual se desempeña muy bien, en la parte multimedia tenemos un menú bastante amplio para reproducción de música y video, la parte gráfica no se queda atrás y contamos de igual manera con varias aplicaciones para la edición y visualización de fotos, el asistente para conexión de redes e internet viene con una interfaz muy sencilla e intuitiva, como dato debo decirles que la detección de redes inalámbricas no tuvo ningún inconveniente y funciona perfectamente, esto para los que se animen a probar la distro en una laptop.

Puppy linux funciona utilizando la memoria ram de la pc por lo cual es conveniente tener 128 MB, aunque funciona con 64 y se desempeña muy bien, una cosa interesante es que permite guardar la sesión en una parte del disco duro, en un cd o bien en un pendrive, si personalizas la apariencia o realizas alguna instalación adicional puede guardar cómodamente esos cambios.

* La clásica pantalla de bienvenida de Puppy, limpia y muy agradable.

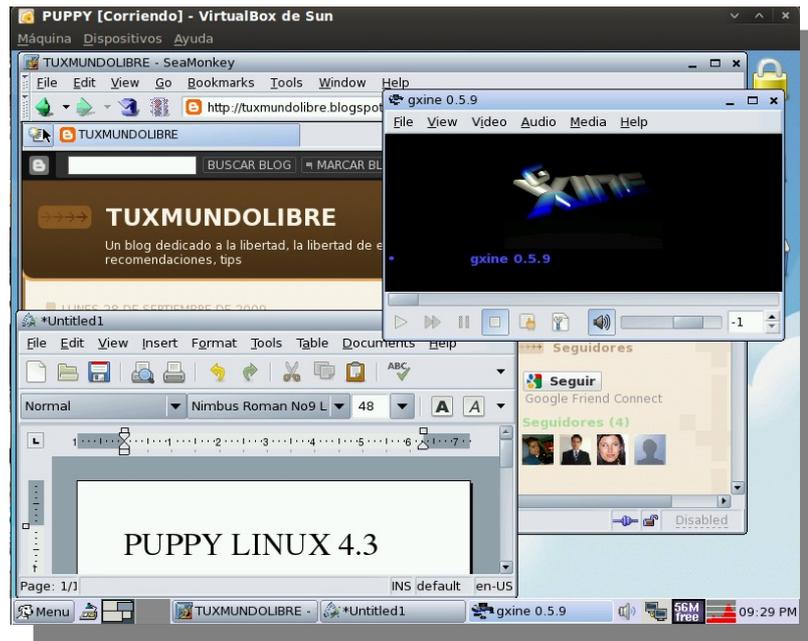


Esta es la pantalla principal de Puppy, como podemos ver viene con gran cantidad de accesos a las aplicaciones que más se utilizan, así como el instalador, visualmente puppy es muy agradable y por supuesto se puede personalizar.

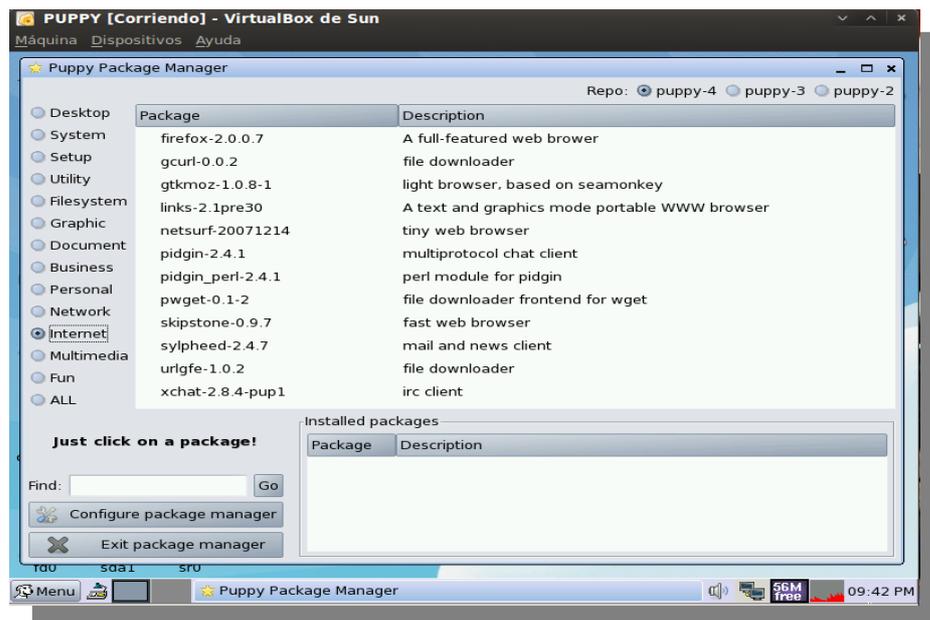


Como verán en esta pantalla tenemos las varias opciones sobre donde queremos instalar nuestro Puppy, desde un disco duro, pasando por un dispositivo usb, yo me animé por dejarle una partición de disco duro, la instalación fue muy sencilla y no duró más de 20 min, si, 20 min, increíble ¿no ?.

Aquí vemos corriendo tres aplicaciones a la vez, navegador web, reproductor multimedia y procesador de textos, todos se desempeñaron de una excelente forma, y pude abrir tanto páginas, como videos y documentos sin ningun inconveniente



Nos vamos a entretener bastante con las muchas aplicaciones que vienen instaladas en Puppy, otra de las maravillas que encontramos en esta distribución es que podemos instalar más aplicaciones si así lo requerimos, tan sencillo como ejecutar el puppy install manager, seleccionar una categoría y buscar el programa que deseamos instalar, si, así de fácil.



Como acostumbro a decir, prueben y exploten al máximo esta distribución que tiene muchas cosas que ofrecemos para el trabajo diario, vamos a darle la oportunidad que sin duda nos va a dejar muy convencidos de que un sistema operativo no tiene por qué ser robusto y devorador de recursos para funcionar, alguien se anima a probarlo ????

Espero sus propuestas, sugerencias y críticas, y nos vemos en el próximo número.

Samuel Morales Cambrón
ASESOR EN SISTEMAS

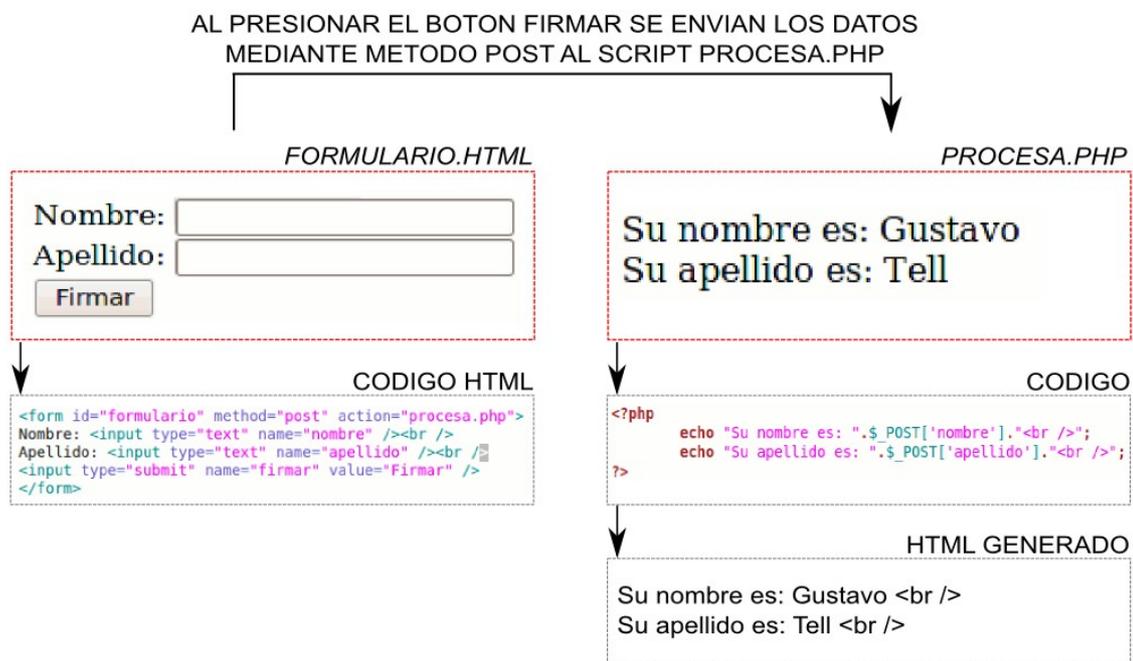
mcambbron76@gmail.com
<http://tuxmundolibre.blogspot.com/>
Twitter : smcambron

PHP + MySQL (2º PARTE)

Repaso: En el número anterior de TuxInfo conocimos los componentes y software con los que vamos a trabajar, una visión general del funcionamiento de los scripts PHP, definimos lo que era una variable, una constante y una función. En esta entrega continuaremos conociendo comandos esenciales que nos servirán como herramientas cotidianas en el desarrollo de sitios web dinámicos.

VARIABLES GET Y POST

Seguramente muchos de nosotros habremos utilizado un formulario de contacto en alguna página web visitada. En dicho formulario encontramos elementos tales como campos de texto y botones. Una vez rellenos los campos y ser presionado el botón “**ENVIAR**” estos datos son volcados a un script de destino que será el encargado de procesar estos datos.



En la columna izquierda del diagrama vemos que la página del formulario puede ser un simple HTML, en el cual creamos un formulario, con la propiedad **METHOD** definimos el método de envío que puede ser **GET** o **POST**, y luego definimos que sus resultados se deben volcar en PROCESA.PHP, (propiedad **ACTION**), a continuación creamos dos campos de texto (Inputs del tipo “text”) y definimos sus respectivos nombres (“nombre” y “apellido”). Por último un elemento del tipo “**SUBMIT**” (botón de envío) de nombre “firmar” y el texto que contendrá será “Firmar” (propiedad Value).

En la columna derecha del diagrama vemos a **PROCESA.PHP**, el archivo encargado de recibir los datos y mostrarlos o hacer de ellos lo que deseemos.

VARIABLES GET

Estas variables pueden ser pasadas por URL, es decir, por ejemplo, tipeando la siguiente dirección en el navegador.

http://127.0.0.1/tu_script.php?nombre=Gustavo&apellido=tell

Ya que tanto nombres de variable como su contenido son visibles no hay que utilizar este método para el envío de datos sensibles como pueden ser contraseñas. Este método está limitado a 100 caracteres por variable, y la dirección completa de la URL por cuestiones de compatibilidad con navegadores y servidores no debe exceder los 2.000 caracteres.

Este método puede resultar útil por ejemplo, para elegir un determinado producto en una web:

```
http://www.tu_empresa.com.ar/producto.php?id=24
```

Podemos acceder al valor de esta variable utilizando `$_GET['id'];`

VARIABLES POST

Este método envía los datos por STDIO, por lo que las variables y sus valores no resultan visibles en la URL, además, este método no presenta límites en la cantidad de caracteres que puede tener cada variable y permite también el envío de archivos. Es el método más utilizado en formularios.

Podemos acceder al valor de las variables POST mediante `$_POST['nombre_de_campo'];`

VARIABLES SESSION Y COOKIES

Estas variables son las utilizadas al momento de, por ejemplo, loguearse en alguna web.

VARIABLES SESSION

El valor de estas variables permanecen en el servidor, y a cada navegador visitando nuestra web se le asigna un identificador único y de esta forma se crea lo que llamamos 'sesión'. Esta 'sesión' tiene una cierta duración de algunos minutos de inactividad y pasada esta la sesión caduca, al igual que lo haría si cerrásemos el navegador.

Para empezar a utilizar las variables **session** es necesario crear la sesión:

```
session_start();
```

Si la sesión no existe, se creará, en caso contrario la sesión existente será retomada y ya se podrá acceder a sus variables.

Podemos acceder a alguna variable de sesión mediante `$_SESSION['nombre'];`

Podemos destruir una sesión utilizando el comando `session_destroy();`

VARIABLES COOKIES

Al contrario de las variables session, el valor de las variables cookies se almacena en el disco del cliente (por eso hay que tener cuidado cómo tratamos esta información ya que pudo haber sido manipulada) y no se pierden al cerrar el navegador, aunque también tienen fecha de caducidad.

Podemos guardar una cookie de la siguiente forma:

```
// Esta Cookie caducará en una hora (hora actual más 3600 segundos)
setcookie("nombre", "valor", time()+3600);
```

Leerla utilizando `$_COOKIE['nombre'];`

Y destruirla definiendo una fecha de caducidad que ya ha pasado:
`setcookie("nombre", "", time()-3600);`

OPERADORES DE INCREMENTO / DECREMENTO

Estos operadores permiten aumentar o disminuir en una unidad el valor de una variable, por ejemplo:

```
<?php
    $a=2;
    $b=5;

    echo $a++; //SE IMPRIME 2
    echo $b--; //SE IMPRIME 5

    // HASTA AQUI $a=3 y $b=4

    echo ++$a; //SE IMPRIME 4
    echo --$b; //SE IMPRIME 3

    // HASTA AQUI $a=4 y $b=3
?>
```

En el primer grupo utilizamos los operadores (++) y (--) a la derecha del nombre de variable, esto hace que se lea el valor de la variable, con el echo se imprime, y luego se le incrementa/decrementa el valor en una unidad.

Cuando utilizamos los operadores a la izquierda del nombre de variable, primero se incrementa/decrementa el valor en una unidad, luego se lee el valor de la variable y se imprime.

CONDICIONALES

Una sentencia condicional nos permite ejecutar o no cierto código dependiendo de una expresión a evaluar. Se podría pensar como un camino en el cual nos encontramos una bifurcación, entonces el programa debe decidir si seguir por un camino u otro.

SENTENCIA IF

```
if(expresión){
    bloque de código si la expresión es verdadera.
} else {
    bloque de código si la expresión es falsa.
};
```

Si la expresión evaluada es verdadera, se ejecuta el primer bloque de código, de lo contrario, se ejecutara el código colocado luego del **else**.

```
if($items>3){
    echo "hay más de 3 items";
} else {
    echo "hay 3 o menos items";
};
```

Dentro de la expresión podemos utilizar los operadores:

```

== (igual)
!= (distinto)
>(mayor)
< (menor)
<= (menor o igual)
>= (mayor o igual)

```

Tanto el comando **ELSE** como el código en caso que la expresión sea falsa puede ser omitido de no ser necesario.

SENTENCIA SWITCH

```

switch($variable){
case "valor 1":
Código que se ejecuta si el contenido de la variable es "valor 1".
break; //Evito que se ejecute el código del valor 2.
case "valor 2":
Código que se ejecuta si el contenido de la variable es "valor 2".
break; //Evito que se ejecute el código que continúa.
default:
Código que se ejecuta si no hay coincidencias.
};

```

Este método se utiliza para encontrar coincidencias en el valor de una variable. Nótese que sin el comando **Break** se ejecutarían todos los códigos de cada caso empezando desde el caso en el cual se encontró la coincidencia.

```

switch($color){
case "rojo":
echo "el color es rojo";
break;
case "azul":
echo "el color es rojo";
break;
case default:
echo "el color no es rojo ni azul";
};

```

BUCLES

Un bucle es un bloque de código que se repite mientras una condición sea verdadera.

BUCLE WHILE Y DO-WHILE

El funcionamiento del bucle while es sencillo, se repetirá un código y podremos hacer que se analice la veracidad de la condición antes de ejecutar el código (**While**) o al finalizar cada repetición del bloque (**Do-While**, este último permite ejecutar aunque sea una vez el bloque por más que la condición resulte falsa). Una vez que se evalúe la condición y esta resulte falsa, el bucle while se dará por finalizado, continuando con el código restante del programa. Vemos el funcionamiento de este bucle con ejemplos:

<?php

```
// ASIGNAMOS VALORES A LAS VARIABLES '$PRIMERO' y '$SEGUNDO'
```

```
$primero = 1;
$segundo = 20;
```

```
//EJEMPLO DE WHILE
```

```
while($primero>$segundo){
    $primero++; //SUMO UNO AL VALOR DE $PRIMERO
    $segundo--; //RESTO UNO AL VALOR DE $SEGUNDO
};
echo $primero."<br />";
echo $segundo."<br />";
echo "<br />";
```

```
// ASIGNAMOS VALORES A LAS VARIABLES '$PRIMERO' y '$SEGUNDO'
```

```
$primero = 1;
$segundo = 20;
```

```
//EJEMPLOS DE DO-WHILE
```

```
do{
    $primero++; //SUMO UNO AL VALOR DE $PRIMERO
    $segundo--; //RESTO UNO AL VALOR DE $SEGUNDO
}while($primero>$segundo);
echo $primero."<br />";
echo $segundo."<br />"; //EL <br /> PRODUCE UN SALTO DE LINEA EN HTML
echo "<br />";
?>
```

En la pantalla del navegador veremos:

```
1
20
```

```
2
19
```

El primer grupo de números corresponde al WHILE, que al analizar la condición al principio y encontrarla falsa directamente no entra en el bloque de código, Mostrándonos los valores originales al salir del bucle.

El segundo grupo de números corresponde al DO-WHILE, que al analizar la condición al final del código permitió que se ejecute una vez el bloque de código sin controlar nada, luego, evalúa la condición, y como en nuestro caso la encontró falsa no volvió a repetir el código.

BUCLE FOR

En este tipo de bucle le asignamos un valor inicial a una variable y lo vamos modificando en cada ciclo, hasta que su valor afecte la veracidad de una condición que estará evaluándose al inicio de cada ciclo.

```
for([Inicialización de la variable];[Condición];[Incremento o decremento de la variable])
{
  Bloque de Código que se ejecutará de manera cíclica.
};
```

Ejemplo:

```
<?php
for($i=0;$i<5;$i++){
echo 'El valor de $i en este ciclo es:'. $i. '<br />';
};
?>
```

En el navegador veremos:

```
El valor de $i en este ciclo es:0
El valor de $i en este ciclo es:1
El valor de $i en este ciclo es:2
El valor de $i en este ciclo es:3
El valor de $i en este ciclo es:4
```

Aquí inicializamos el ciclo **FOR** con una variable **\$i** con valor **0**, al comienzo de cada ciclo evaluamos una condición, en nuestro caso **\$i<5**, si **\$i** llegase a ser igual o mayor a **5** el ciclo se cortaría continuando con el código que este inmediatamente debajo, esto se logra incrementando **\$i** en uno cada ciclo , con **\$i++**.

Hasta aquí vimos como intercambiar datos entre páginas, crear cookies y sesiones, incrementar/decrementar valores de variable y hacer que nuestro programa tome decisiones o realice ciclos en base a estos valores. Hasta la próxima!

Gustavo Tell

TUX OPEN
WWW.TUXINFO.COM.AR

Lin-X 1.1 y Linux Mint 7 'Gloria' XFCE

Originalmente esta iba a ser una entrada completa sobre Lin-x 1.1, pero resultó ser tan igual a Ubuntu 8.10 en cuanto a estructura y software; que no dió para una nota larga. De paso la usaremos como repaso de una instalación basada en Ubuntu.

Por lo cual, se coló Linux Mint 7 Gloria XFCE Edition. Así que, vayamos por partes,

Lin-X 1.1

¿Qué es Linx 1.1?

Es un sistema operativo basado hasta la fecha en Ubuntu 8.10, el cual está personalizado como el Apple OSX. Implementando algunas de sus funciones (Ej. el finder).

Al estar basado en Ubuntu, comparte los mismos repositorios y el mismo update manager. Entre muchas de sus ventajas está la de tener compiz-fusion (El manejador de ventanas 3D) completamente configurado, personalizado y listo para usar. Además del famosísimo dock awn y todo el look and feel de Mac.

Lin-x fue creada por Christopher Carmichael de Panama City, Florida en Marzo de 2009.

Lo primero que haremos será dirigirnos a <http://lin-x.info/download.html> y bajar de alguno de los sitios o en su defecto el torrent de la imagen iso.

Una vez descargada la quemamos en un cd y arrancamos la PC haciéndola bootear desde el CD/DVD, donde nos encontraremos con la siguiente pantalla:

```
ISOLINUX 3.63 Debian-2008-07-15 Copyright (C) 1994-2008 H. Peter Anvin  
Lin-X 1.1
```

```
For the default live system, press ENTER or enter 'live'.
```

```
To start in safe graphics mode, enter 'xforcevesa'.
```

```
To start the installer directly, enter 'install'.
```

```
To verify the CD for errors, enter 'check'.
```

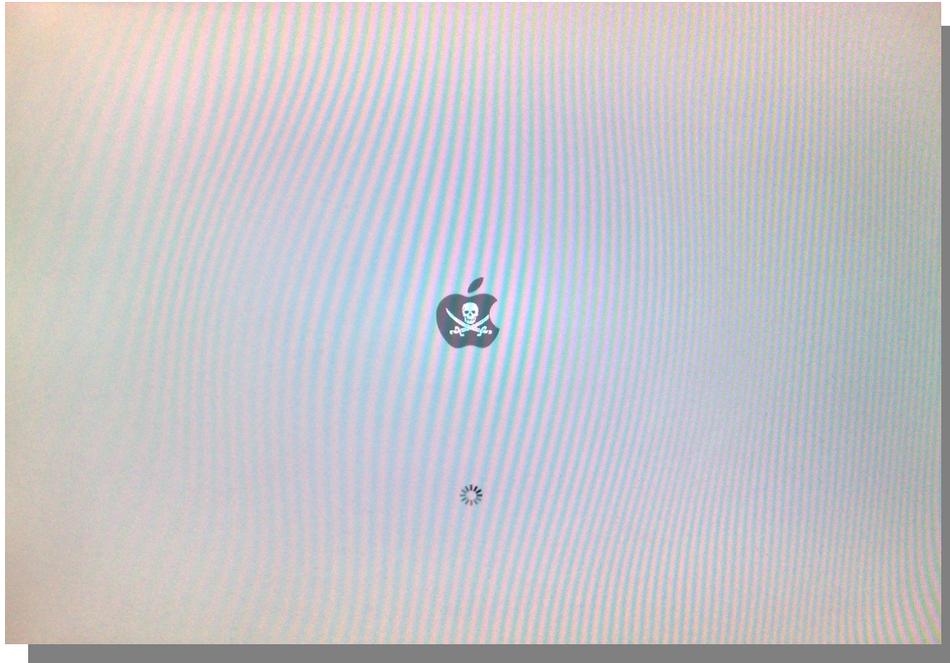
```
To run memtest86+, enter 'memtest'
```

```
To boot from the first hard disk, enter 'hd'
```

```
boot:  
Loading /casper/vmlinuz.....
```

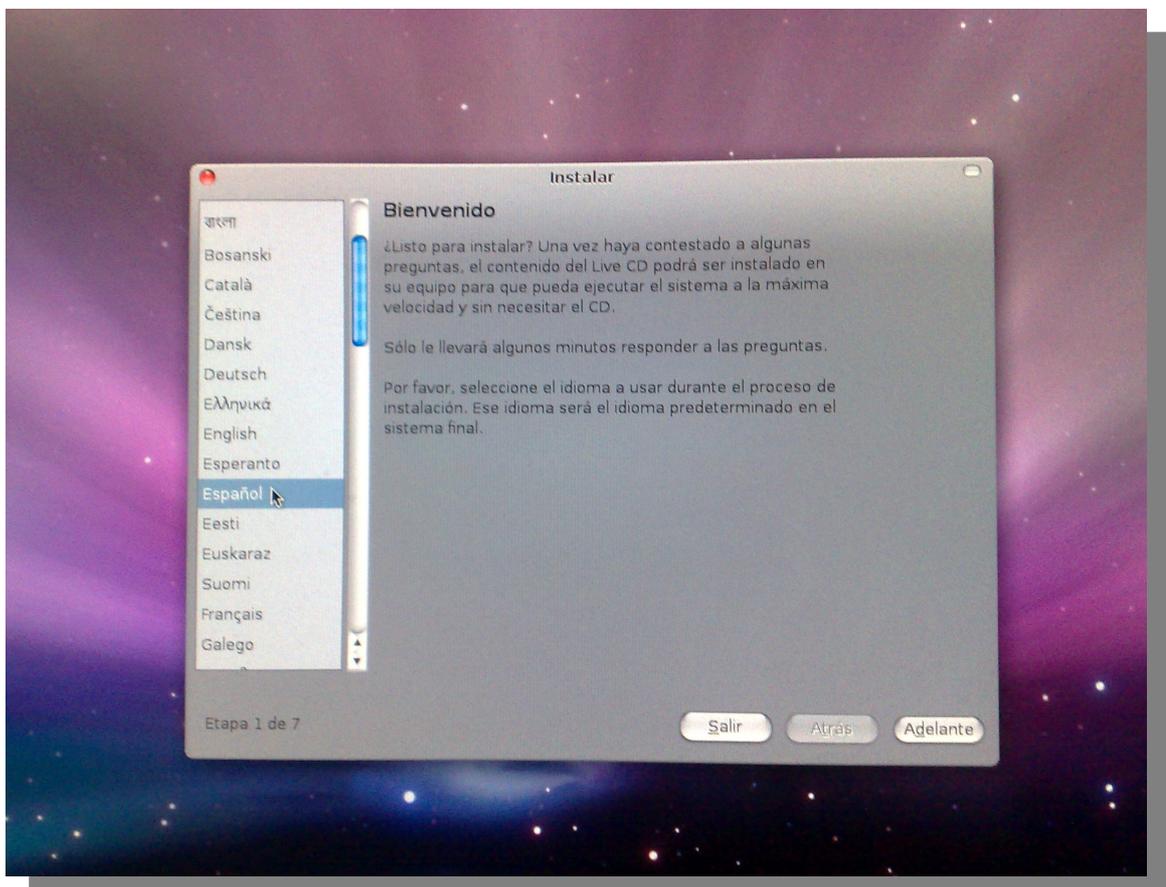
Aquí podemos elegir 'live' si sólo queremos probar Lin-x sin alterar el equipo o install si queremos instalarlo. En este caso elegimos 'install', a fines de poder escribir esta nota.

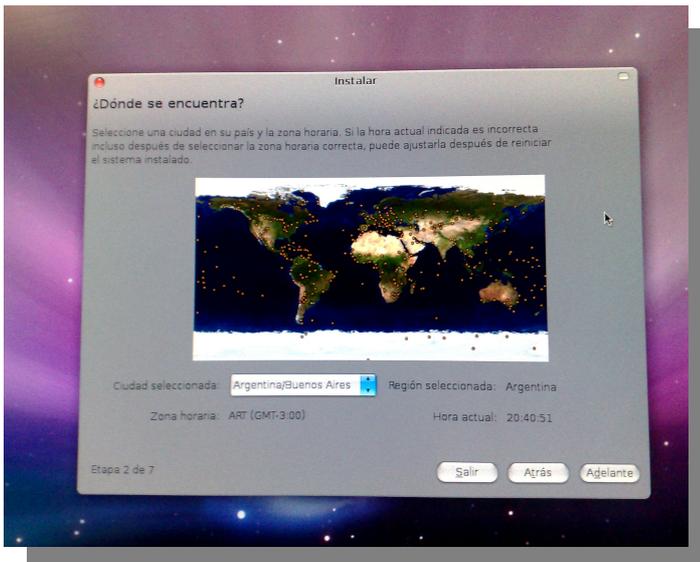
Lo siguiente que veremos es la “sugerente” pantalla de carga del sistema:



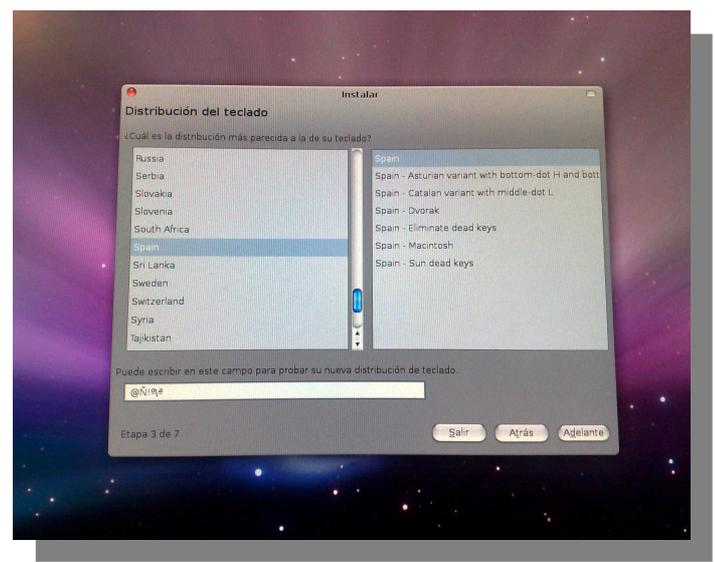
Después de la carga veremos el primer paso de la instalación, la que como no es muy difícil de imaginar es exactamente igual a la de cualquier Ubuntu.

Idioma.



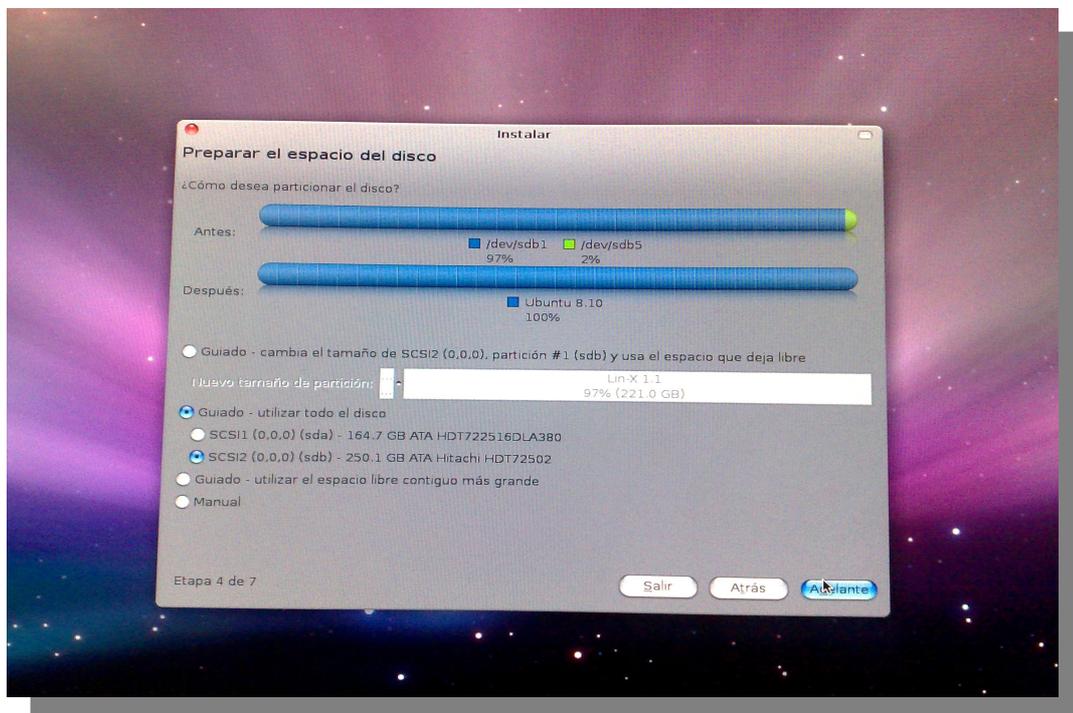


Localización.

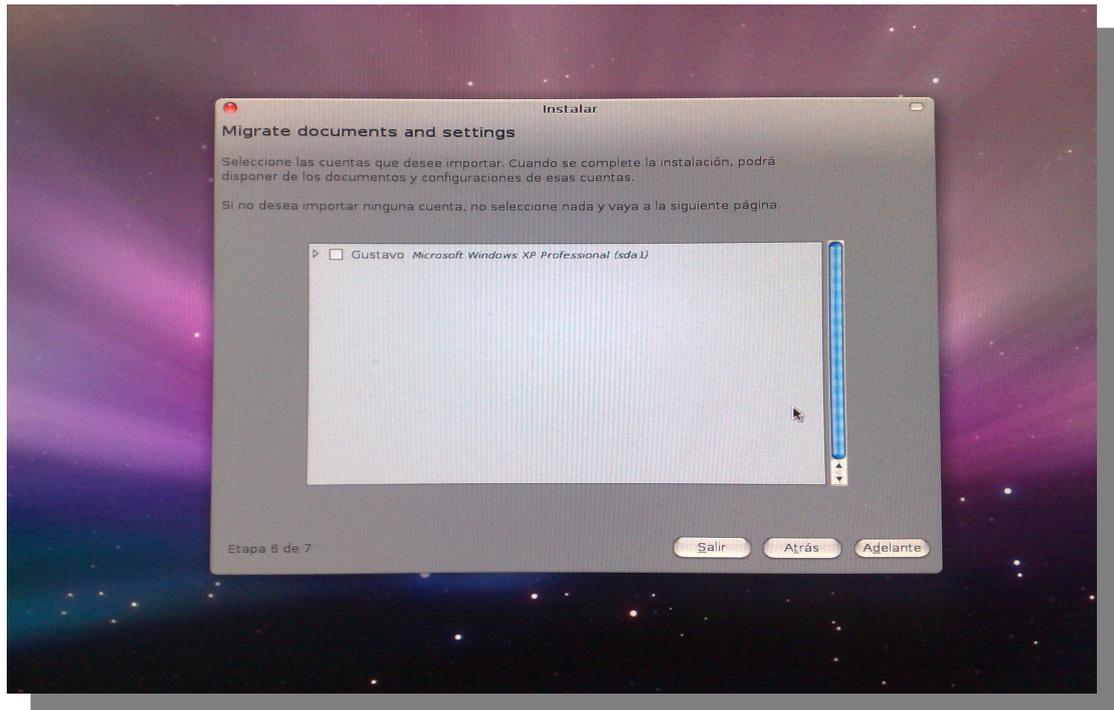


Distribución del teclado.

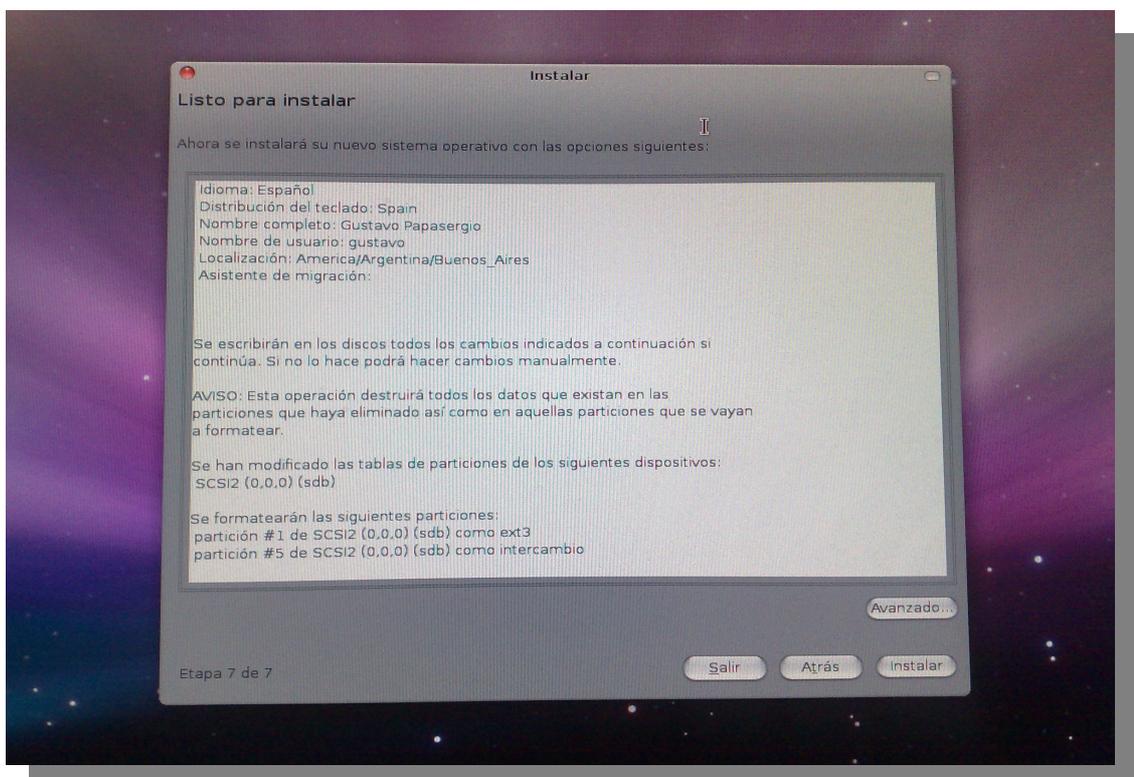
El paso siguiente es indicarle al sistema el lugar de instalación, en mi caso al tener dos discos, verán que decidí usar el de 250gb completo, lo cual hace que particionar no represente ningún misterio, pero podrían haber elegido usar espacio libre de un disco ocupado por windows, y lo hubieran podido hacer aquí también sin ningún problema (haciendo un backup previo de los datos, obviamente).



Luego de esta sencilla manera de hacerle lugar a Lin-X, El instalador nos ofrece importar las configuraciones y Documentos de la Partición XP detectada.



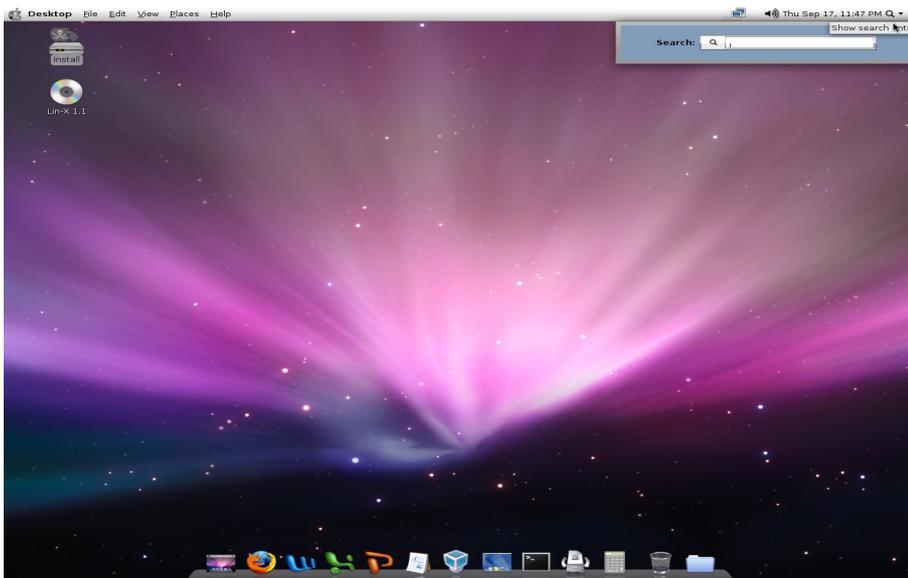
En el paso siguiente, el sistema de instalación nos da un resumen de lo ocurrido, el cual debemos aceptar para comenzar realmente la instalación.



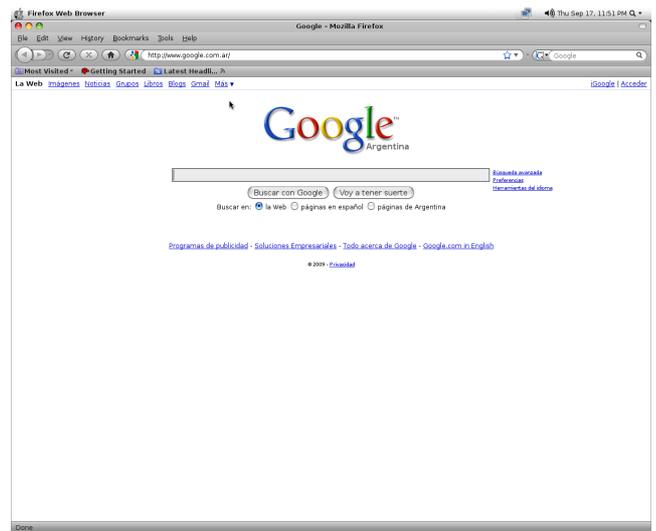
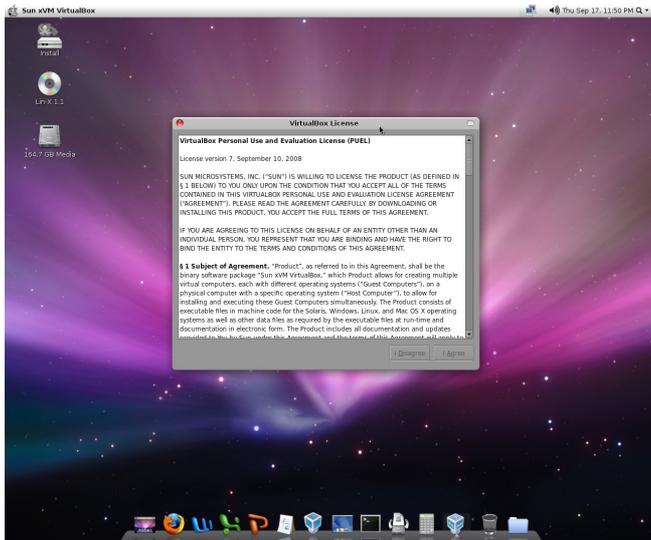
Luego de unos minutos, que dependen del tipo de sistema que tengan, en mi caso fueron 15, la instalación pide reiniciar la PC, expulsa el disco de la lectora y ¡Voilà! En el siguiente Booteo podremos elegir entre Lin-X y Windows, y pasaremos a ver el escritorio.



¿Muy Mac Verdad? Realmente estuve revisando el sistema para encontrar diferencias sustanciales con Ubuntu 8.10, en la cual como habíamos comentado está basado. Y la verdad es que salvo el Finder,

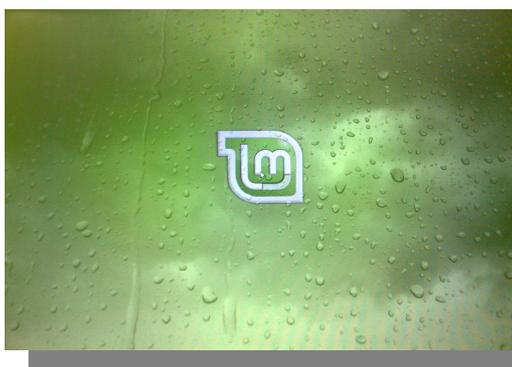


Los íconos de Awn y VirtualBox por defecto



No hay mayores diferencias. Pueden Observar también el detalle del diseño Mac hasta en las ventanas de las aplicaciones, por ejemplo en esta de Firefox.

Y no hay mucho más que esto, una distribución con toda la estabilidad de Ubuntu 8.10 LTS pero con toda la onda OSX para los fans de esta estética.



Por lo cual, pasemos a nuestra Segunda Opción.

Linux Mint 7 'Gloria' XFCE

Linux Mint siempre tuvo dos premisas, elegancia y facilidad de uso, basada en Ubuntu, desde la pantalla de inicio se hace hincapié en el primer atributo en el splash screen cuando dice "From freedom came elegance".

Sin embargo, como cualquier otro Ubuntu estándar, los requerimientos del sistema son algo elevados, sobre todo si son usuarios de netbooks como yo. Por eso cuando vía

twitter me enteré que estaba en línea la versión Gloria con XFCE4, un escritorio superliviano para GNU-Linux; no lo dudé y acá estamos.

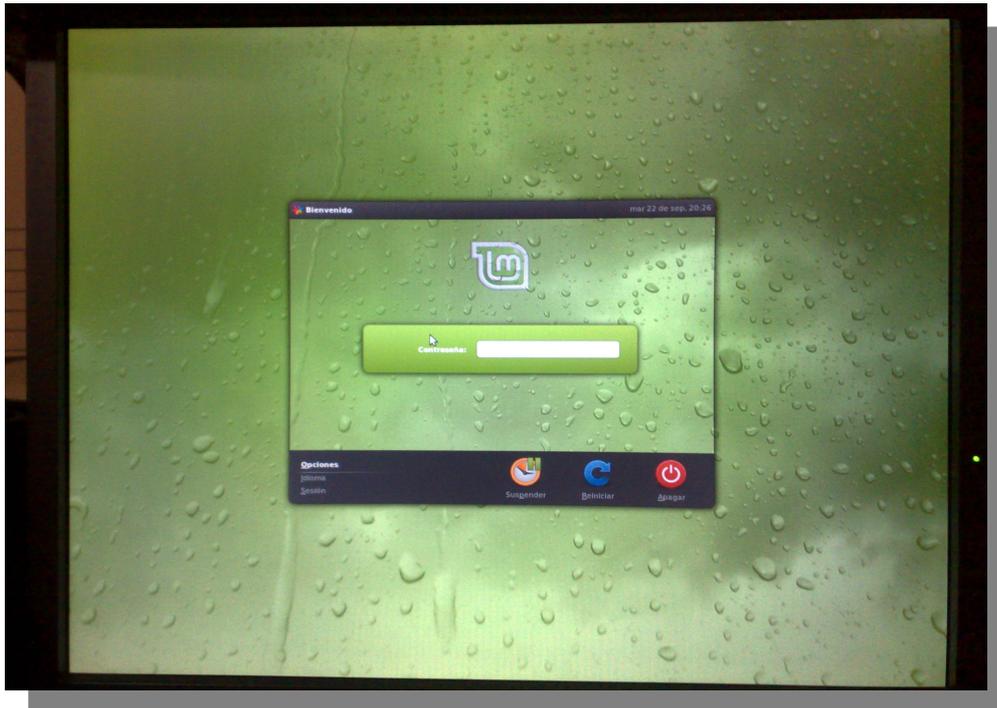
Lo primero que veremos al Iniciar el Live Cd que podemos descargar de estos enlaces ISO o torrent,

<http://www.linuxmint.com/torrent/LinuxMint-7-XFCE.iso.torrent>
<http://www.linuxmint.com/torrent/LinuxMint-7-XFCE.iso.torrent>

A partir de acá, podría comentar la instalación de Mint Gloria XFCE, pero si están viendo la de Arriba (Linux 1.1) Ya vieron esta, dado que no cambia ni en el más mínimo detalle salvo la pantalla de inicio que a diferencia de Linx en vez de ser en modo texto es en modo gráfico. Las distribuciones son basadas en Ubuntu, solo que Mint XFCE en Ubuntu 9.04.

Así que entonces pasamos a la primer pantalla post instalación.

Y a la elegantísima pantalla de Login:



¿Es rápido Gloria 7 con XFCE4?

Sí, ¡vuela! Literalmente, para prueba les cuento que antes de instalarlo en la desktop, lo probé en modo live en la netbook, y a pesar de tener que leer todo el tiempo del dvd externo, el rendimiento del escritorio, era superior incluso al de Xubuntu 9.04 instalado en disco. Demás esta decirles que en el Dual Core con 2gb de ram de la desktop, no se ven pasar las pantallas.

¿Qué trae Esta versión?

Más bien, lo que hay que preguntarse acá es ¿Qué no trae esta versión? Dado que tiene de todo.

Tanto que sería imposible comentarlo en esta nota, realmente hay aplicaciones para todos los gustos.

Para destacar

Gprs Easy Connect: que hace que conectarse a una red celular para hacer uso de Internet Móvil sea una pavada.

EnvyNg Terminal: Terminal para instalar drivers Nvidia de forma muy simple

MintDisk: Montador de particiones con Soporte para ntfs

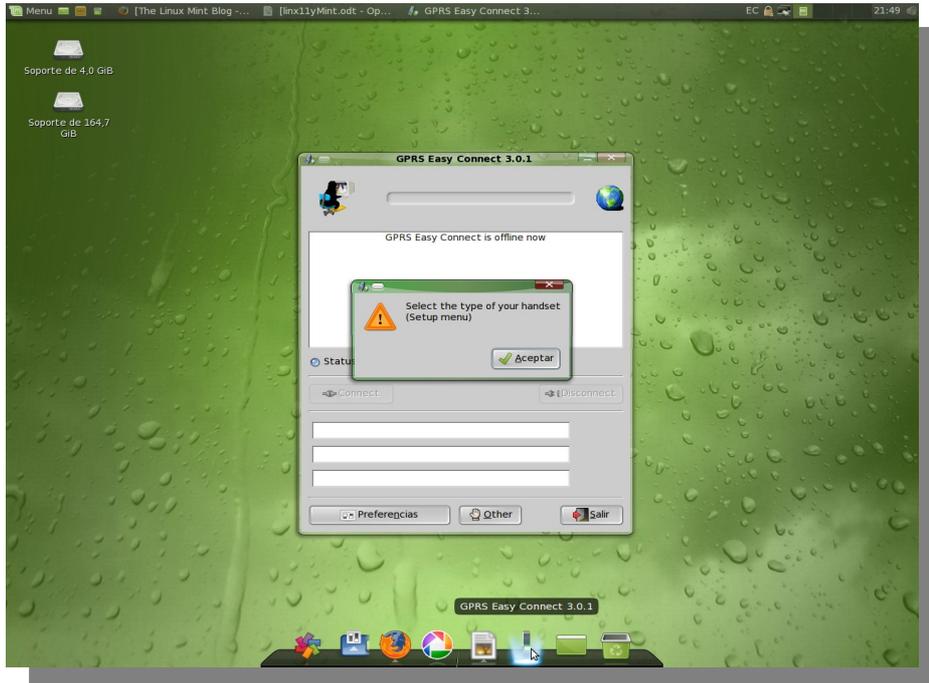
MintNanny: Para proteger a nuestros niños de los sitios que les gusta ver a los padres

Mplayer: instalado por defecto (¡Excelente!)

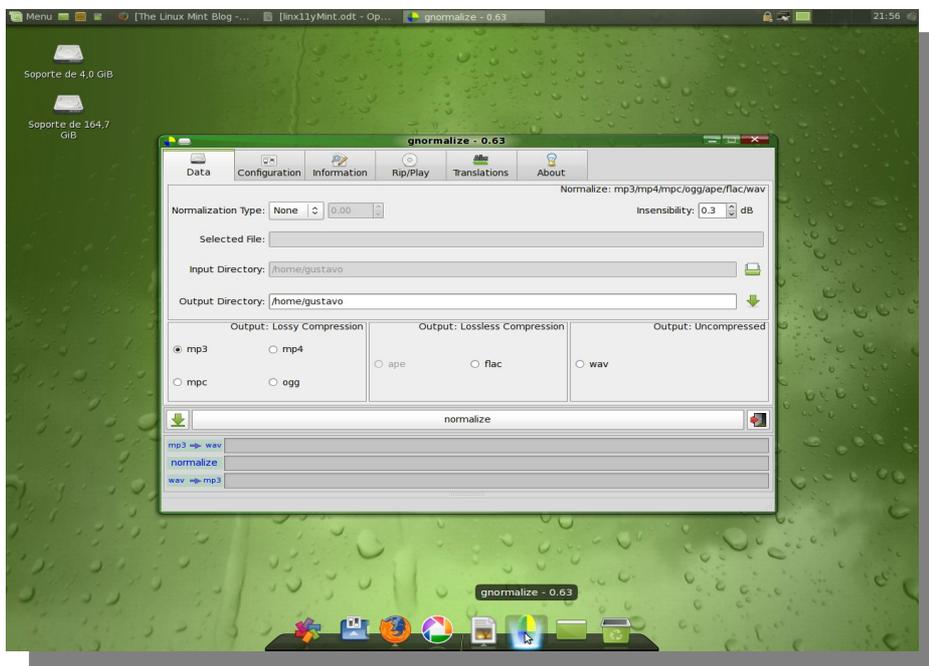
Gnormalize: Ripeador, Normalizador de Archivos de Música multiformato (si tienen una gran colección de Música van a amar este programa)

Open base: Instalado por defecto, normalmente en Ubuntu hay que instalarlo después.

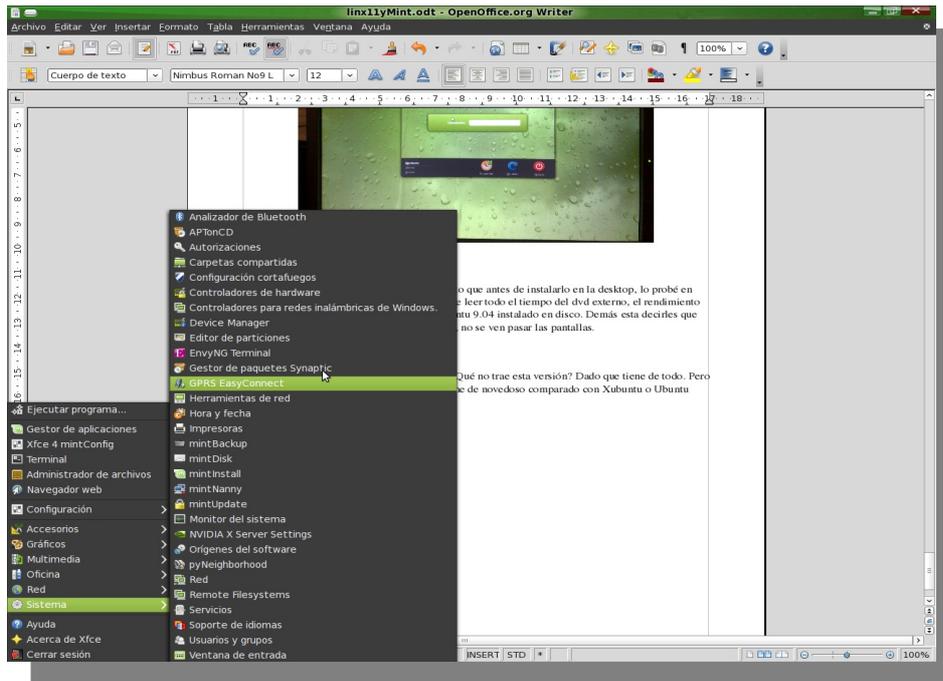
Increíblemente, no trae ningún juego instalado por defecto, la buena noticia es que se pueden instalar por MintInstall, y que tiene la mayoría de los mejores juegos 3D para GNU/Linux.



A saber, Glest, TuxRacer, Open Arena, Varios FPS muy buenos, Torcs, etc.



La otra gran ventaja de Mint 7 XFCE es que MintInstall ya tiene en los repositorios por defecto, la posibilidad de instalar skype, picasa, awn y otros programas en los cuales en Ubuntu hay que hacer más de un paso de complejidad mediana.



Como muestra de todo lo dicho arriba, va esta captura con los numerosos ítems, de por ejemplo la categoría sistema del menú principal.

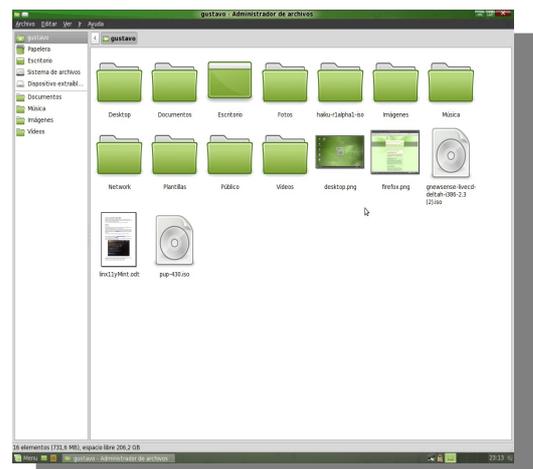
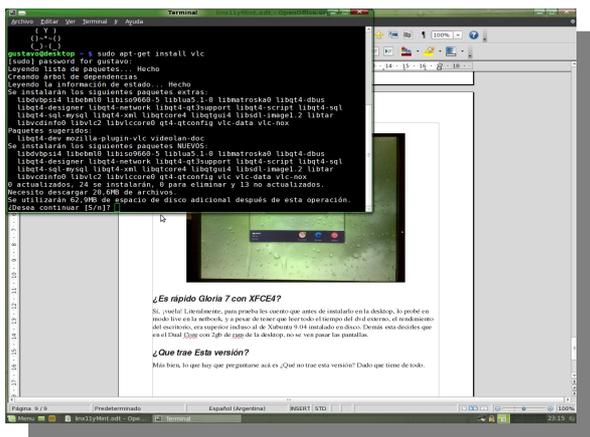
¿Y la usabilidad que tal?

Quizás para los viejos usuarios de Linux, XFCE pueda sonar a escritorio rápido pero poco flexible, sin embargo, esta versión 4 personalizada por la Gente de Mint, hace que sea posible realizar en el escritorio, cualquier operación que podríamos llegar a realizar con Gnome o el mismísimo Windows XP. Lo cual combinado a la gran cantidad de opciones de Software, hacen que Mint 7 XFCE, esté mucho más allá de la expectativa que puede llegar a tener un usuario estándar de hoy en día, pero con una elegancia de interfaz y un rendimiento digno de una PC "tuneada". Y si no me creen miren qué hermosa estética tiene el administrador de archivos (ver última imagen).

Conclusión:

Linux Mint 7 Gloria XFCE, es una distribución elegante, rapidísima y llena de software a la que encontrarle una contra es realmente difícil. Apta para instalar desde una netbook hasta una potente estación de trabajo en la que no queramos desperdiciar recursos en el escritorio, por ejemplo para aprovecharlos en la edición multimedia o juegos; es una de las mejores distribuciones que he probado en los últimos 6 meses. A punto de salir la versión 8 de su versión estándar, es una que no pueden dejar de probar aunque sea en modo live.

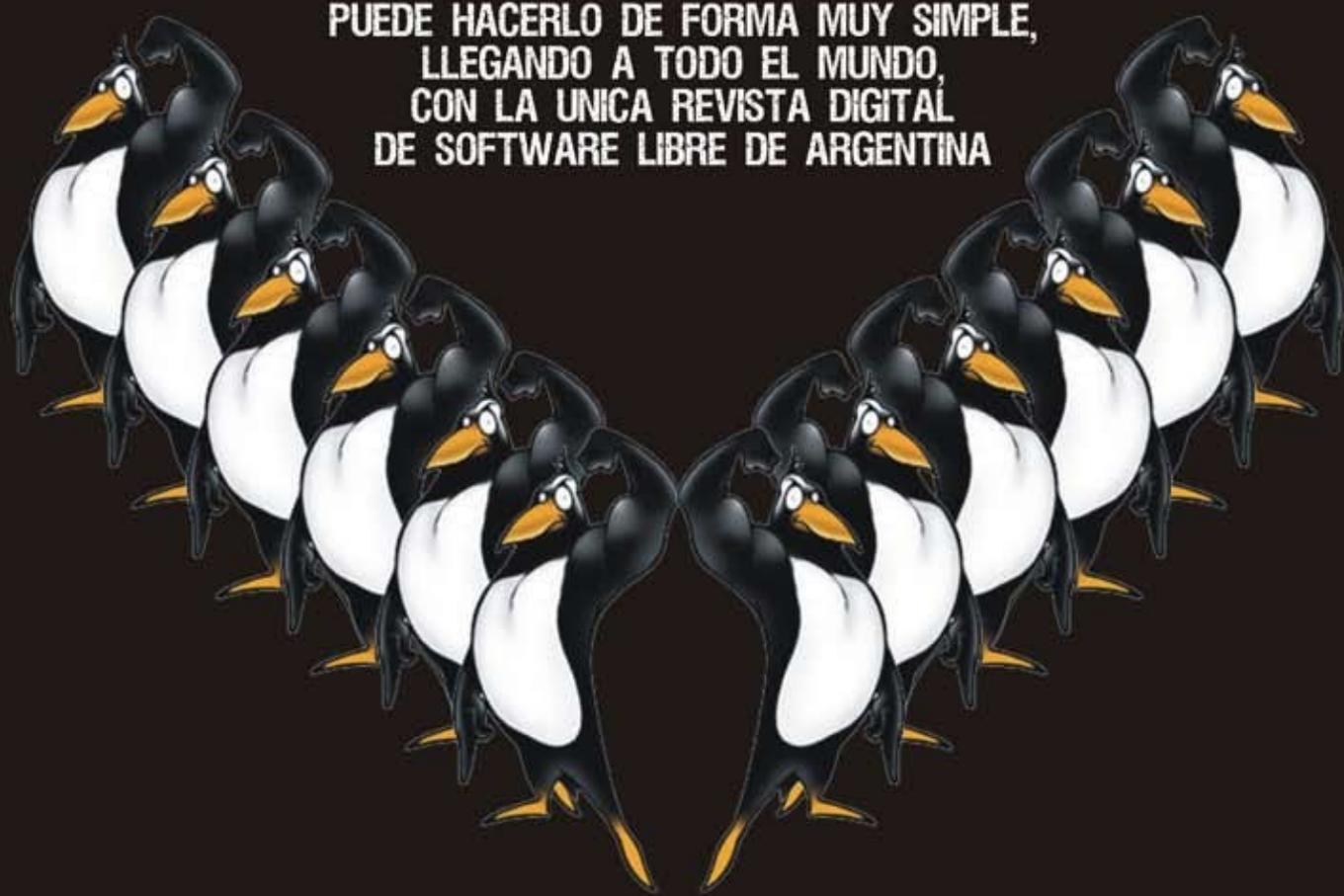
Gustavo Papasergio
Tecnotravel 2.0



SI QUIERE PUBLICITAR EN

TUXINFO
WWW.TUXINFO.COM.AR

PUEDA HACERLO DE FORMA MUY SIMPLE,
LLEGANDO A TODO EL MUNDO,
CON LA UNICA REVISTA DIGITAL
DE SOFTWARE LIBRE DE ARGENTINA



CON TUXINFO MULTIPLICARA SUS CLIENTES

Para mayor información comunicarse vía email a.:
info@tuxinfo.com.ar por skype usuario.: Infosertec

TUX **INFO**
WWW.TUXINFO.COM/AR