

PROTOTIPO PARA EL EMPAQUETADO DE OBJETOS DE APRENDIZAJE
MEDIANTE LINUX LIVE CD.

JUAN CARLOS GARCÍA MOYA

UNIVERSIDAD EAN
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
BOGOTÁ D.C.
NOVIEMBRE DE 2012

PROTOTIPO PARA EL EMPAQUETADO DE OBJETOS DE APRENDIZAJE
MEDIANTE LINUX LIVE CD.

Presentado por:

JUAN CARLOS GARCÍA MOYA

Asesorado Por:

Ing. RUBEN DARIO DORADO
Línea PhEANix - Teklar

UNIVERSIDAD EAN
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
BOGOTA D.C.
NOVIEMBRE DE 2012

TABLA DE CONTENIDO

TABLA DE CONTENIDO	2
TABLA DE IMÁGENES.....	5
RESUMEN.....	7
CAPITULO 1	8
Introducción, Problema, Objetivos y Estructura.	8
INTRODUCCIÓN	8
DESCRIPCIÓN DEL PROBLEMA	9
OBJETIVOS.....	11
ESTRUCTURA	12
CAPITULO 2	14
2. MARCO CONCEPTUAL.....	14
2.1. ESTADO DEL ARTE DE LOS LIVECD.....	14
2.1.1. Concepto de LiveCD.....	14
2.1.2. Historia de los LiveCD.....	14
2.1.3. Uso de un LiveCD	17
2.1.4. La selección del Live CD Base.	17
2.2. ESTADO DEL ARTE OBJETOS DE CONOCIMIENTO	24
2.2.1. Definición de Conocimiento	24
2.2.2. Concepto de Objetos de Conocimiento.....	25
2.2.3. Tipos de Objetos de Conocimiento que se podrán empaquetar en el LiveCD. 27	
2.2.4. ¿Por qué son relevantes los objetos de conocimiento para el proceso educativo?	30
2.2.5. El proceso de educación continuada, la autoformación y el uso de LiveCD 31	
2.2.6. Problemas planteados al modelo de empaquetar objetos de conocimiento en un LiveCD:.....	33
CAPITULO 3	34
3. ARQUITECTURA DEL SOFTWARE	34
3.1. Arquitectura General.....	34
3.2. Módulos de software desarrollados.	35
3.2.1. Captura de los Objetos de Conocimiento a Empaquetar:	35
3.2.2. Captura de los Parámetros de Personalización del LiveCD:.....	36
3.2.3. Lanzador del Proceso de Personalización:	36
3.2.4. Prueba del LiveCD Personalizado:	36
3.2.5. Generar LiveCD en medio físico.	36
3.3. Módulos de software integrados.	36
3.3.1. Descompresión de la ISO Plantilla:.....	37
3.3.2. Creación del ambiente de Trabajo CHROOT:.....	37
3.3.3. Empaquetado de Objetos de Conocimiento y sus Accesos:.....	37
3.3.4. Adición Modificación de Paquetes de Software:	37

3.3.5.	Reconstrucción del nuevo LiveCD:	38
3.4.	Parámetros de entrada.	38
3.4.1.	Lista de Documentos (Objetos de Conocimiento):.....	38
3.4.2.	Idioma del LiveCD.....	39
3.4.3.	Ambiente del Desktop (KDE, Gnome, Otros):.....	39
3.4.4.	Imagen ISO Base para usar en la Masterización:.....	39
3.4.5.	Nombre Etiqueta del LiveCD (32 caracteres Máximo):	39
3.4.6.	Personalizar Manualmente o Automáticamente el LiveCD.:	39
3.4.7.	Borrar Archivos del LiveCD visibles en Ambiente Windows:.....	39
3.4.8.	Confirmar Opciones e Iniciar el proceso.	40
3.5.	Archivos de Entrada.....	40
3.5.1.	Imagen ISO de un LiveCD Basado en Ubuntu.....	40
3.5.2.	Documentos (Objetos de Conocimiento).	40
3.6.	Archivos de Salida.	40
3.6.1.	Imagen ISO de un LiveCD Basado en Ubuntu:.....	40
3.6.2.	Documentos (Objetos de Conocimiento):	41
3.7.	Flujo Funcional de la Aplicación.....	41
3.7.1.	Captura de información:.....	41
3.7.2.	Inicio de la Personalización:.....	41
	• Preparar el ambiente de trabajo.....	41
	• Copia la fuente del sistema al directorio de trabajo.....	41
	• Generar el ambiente independiente con chroot en el nuevo sistema.....	41
	• Efectuar las modificaciones.....	41
	• Prepare el árbol de directorios del CD.	41
	• Construir de nuevo el CD / DVD	41
CAPITULO 4		43
4.	PROCESO DE DISEÑO DE SOFTWARE	43
4.1.	Requerimientos Funcionales.....	43
4.2.	Requerimientos de Integración del Software	44
4.3.	Requerimientos de Hardware y Software.....	44
4.4.	Casos de Uso para los cuales aplica.	45
4.5.	Diseño de la Interfaz	47
CAPITULO 5		50
5.	DISEÑO Y CONSTRUCCIÓN DE LOS OBJETOS DE CONOCIMIENTO.....	50
5.1.	Relación del estilo de aprendizaje del ser humano y el diseño de los Objetos de Conocimiento.	50
5.2.	Diseño del Modelo para los Objetos de Conocimiento.	54
5.2.1.	Diseño pedagógico del Objeto de Conocimiento.	54
5.2.2.	Diseño tecnológico del Objeto de Conocimiento.....	55
5.3.	Construcción de un Objeto de Conocimiento.....	56
5.4.	La construcción del primer Objeto	57
CAPITULO 6		59
6.	CONCLUSIONES Y EVOLUCIÓN.....	59
	Evolución de este proyecto.	61

7. BIBLIOGRAFÍA.....	62
ANEXOS.....	¡Error! Marcador no definido.
Anexo 1. Navegación y uso de la Interfaz gráfica del proyecto PhEANix.	¡Error! Marcador no definido.
Anexo 2. El Proceso de Adquisición e Instalación de la Distro Base.....	¡Error! Marcador no definido.
Anexo 3. El Proceso de Masterización.	¡Error! Marcador no definido.
Anexo 4. Ampliación conceptual de Conocimiento y Objetos de Conocimiento	¡Error! Marcador no definido.
Anexo 5. Formato y Contenido de los objetos de conocimiento que se pueden empaquetar en PhEANix	¡Error! Marcador no definido.

TABLA DE IMÁGENES

Ilustración 1: Gráfico Arquitectura de Niveles de Agregación de ODE según LOM-ES v.1.0	27
Ilustración 2: Arquitectura del Proyecto PhEANix	34
Ilustración 3: Pantalla Principal del Programa	48
Ilustración 4: Pantalla de Captura parámetros Personalización.....	48
Ilustración 5: Estilos de Aprendizaje de David Kold	51
Ilustración 6: Ventana Principal - Panel Material Didactico	¡Error! Marcador no definido.
Ilustración 7: Cuadro de Dialogo para Selección de Documentos	¡Error! Marcador no definido.
Ilustración 8: Selección de un Documento tipo PDF	¡Error! Marcador no definido.
Ilustración 9: Árbol de Documentos Seleccionados .	¡Error! Marcador no definido.
Ilustración 10: Copia de Archivos al Directorio de trabajo	¡Error! Marcador no definido.
Ilustración 11: Pestaña de Parámetros Básicos.....	¡Error! Marcador no definido.
Ilustración 12: Cuadro de dialogo selección de ISO	¡Error! Marcador no definido.
Ilustración 13: Selección demás parámetros de Personalización LiveCD e inicio del Proceso.....	¡Error! Marcador no definido.
Ilustración 14: Pantalla de Confirmación de Requerimientos	¡Error! Marcador no definido.
Ilustración 15: Pantalla de Confirmación de parámetros	¡Error! Marcador no definido.
Ilustración 16: Ventana Principal de scripting	¡Error! Marcador no definido.
Ilustración 17: Primeros pasos del proceso	¡Error! Marcador no definido.
Ilustración 18: Selección de Herramientas para Personalización Avanzada ..	¡Error! Marcador no definido.
Ilustración 19: Pantalla de Bienvenida del Paquete synaptic	¡Error! Marcador no definido.
Ilustración 20: Ventana principal de synaptic.	¡Error! Marcador no definido.
Ilustración 21: Selección de la consola chroot	¡Error! Marcador no definido.
Ilustración 22: Consola de personalización CHROOT	¡Error! Marcador no definido.
Ilustración 23: Se continúa con el proceso.	¡Error! Marcador no definido.
Ilustración 24: re compresión del filesystem principal en formato squashfs....	¡Error! Marcador no definido.
Ilustración 25: avance de la re-compresión.	¡Error! Marcador no definido.
Ilustración 26: Creación del nuevo ISO.....	¡Error! Marcador no definido.
Ilustración 27: Terminación del Proceso de Creación del LiveCD.	¡Error! Marcador no definido.
Ilustración 28: Ventana de terminación Exitosa.	¡Error! Marcador no definido.

Ilustración 29: Inicio de una sesión de qemu, con el nuevo liveCD.¡Error!
Marcador no definido.

Ilustración 30: Pantalla de inicio del LiveCD. ¡Error! **Marcador no definido.**

Ilustración 31: Menú de Inicio del LiveCD. ¡Error! **Marcador no definido.**

Ilustración 32: Escritorio del nuevo liveCD. ¡Error! **Marcador no definido.**

Ilustración 33: Ejecución del Link al Contenido Empaquetado.¡Error! **Marcador no definido.**

Ilustración 34: Visualización del Índice del Contenido;¡Error! **Marcador no definido.**

Ilustración 35: Apagado del LiveCD en qemu ¡Error! **Marcador no definido.**

Ilustración 36: Pantalla del LiveCD al apagar la sesión qemu.¡Error! **Marcador no definido.**

Ilustración 37: Pantalla Principal de Ubuntu 7.04 inmediatamente después de iniciarse el LiveCD ó de su instalación..... ¡Error! **Marcador no definido.**

Ilustración 38: Pantalla de la selección de Zona Horaria y Fijación de Hora en Ubuntu 6.06 ¡Error! **Marcador no definido.**

Ilustración 39: Asistente de Migración de Configuraciones Windows.¡Error!
Marcador no definido.

Ilustración 40: Pantalla de Confirmación de configuración previa instalación. ¡Error!
Marcador no definido.

Ilustración 41: Configuración de Usuarios. ¡Error! **Marcador no definido.**

Ilustración 42: Creación de la cuenta de usuario. ¡Error! **Marcador no definido.**

Ilustración 43: Privilegios default de Ubuntu. ¡Error! **Marcador no definido.**

Ilustración 44: Datos de Home, Shell y Grupo. ¡Error! **Marcador no definido.**

Ilustración 45: Configurando el Grupo. ¡Error! **Marcador no definido.**

Ilustración 46: Asignando el ID al Grupo..... ¡Error! **Marcador no definido.**

Ilustración 47: Cambio de Propiedades del Grupo... ¡Error! **Marcador no definido.**

Ilustración 48: Cambio Fondo del Escritorio. ¡Error! **Marcador no definido.**

Ilustración 49: Preferencias del Fondo del Escritorio.¡Error! **Marcador no definido.**

Ilustración 50: Configuración de KDesktop. ¡Error! **Marcador no definido.**

Ilustración 51: Imagen del Boot Splash Ubuntu 7.0.4 Original.¡Error! **Marcador no definido.**

Ilustración 52: Imagen del Boot Splash PhEANix Original.¡Error! **Marcador no definido.**

Ilustración 53: Menú de Administración. ¡Error! **Marcador no definido.**

Ilustración 54: Preferencias de la pantalla de Logon.¡Error! **Marcador no definido.**

Ilustración 55: Selección imagen de logon..... ¡Error! **Marcador no definido.**

Ilustración 56: Ajuste de Idioma de los Escritorios KDE y Gnome.¡Error! **Marcador no definido.**

Ilustración 57: Arranque del LiveDVD PhEANix..... ¡Error! **Marcador no definido.**

Ilustración 58: Boot Splash de PhEANix. ¡Error! **Marcador no definido.**

Ilustración 59: Inicio de Sesión KDE PhEANix..... ¡Error! **Marcador no definido.**

Ilustración 60: Cierre de Sesión KDE PhEANix. ¡Error! **Marcador no definido.**

Ilustración 61: Cambio de Entorno y Usuario..... ¡Error! Marcador no definido.
Ilustración 62: Inicio de Sesión GNOME PhEANix... ¡Error! Marcador no definido.
Ilustración 63: Correlación Conocimiento - Información y Aprendizaje.....¡Error!
Marcador no definido.
Ilustración 64: Representación de Modelo de Objetos LOM.¡Error! Marcador no
definido.

RESUMEN

El presente trabajo de grado aborda la construcción de un prototipo de interfaz grafica de usuario intuitiva y de fácil manejo que le permita a un docente o estudiante, ó cualquier miembro de la comunidad educativa en general empaquetar objetos de conocimiento dentro de un LiveCD (El rincon de Linux para Hispano Parlantes, 1998) tipo Linux, con el fin de brindarles una alternativa adicional en el manejo de herramientas informáticas en el ámbito académico.

El prototipo objeto de este trabajo se desarrollo a partir de la necesidad de ofrecer a la comunidad educativa una alternativa que le permita al docente entregar a sus alumnos material didáctico digital y programas de software, que puedan ser portados, visualizados y ejecutados en cualquier computador de la universidad o fuera de ella sin la limitante de tener que instalar dicho software.

En el desarrollo del prototipo se utilizaron como herramientas de desarrollo, eclipse java (Eclipse Foundation, 2012), Shell script de Linux (Lutus, 2009) y varios utilitarios de sistema operativo.

CAPITULO 1

Introducción, Problema, Objetivos y Estructura.

INTRODUCCIÓN

La creciente presión de la sociedad en el ámbito laboral para que los profesionales estén mejor preparados en las diferentes áreas del conocimiento, demanda de las entidades de formación tanto formal como informal, a proveer las estrategias y herramientas necesarias que les permitan a sus estudiantes acceder al conocimiento de forma flexible garantizando la calidad de los contenidos.

En este proceso de flexibilización y masificación del acceso a los centros de conocimiento, han participado tanto la industria, como la academia, generando necesidades y soluciones de forma mancomunada.

La ingeniería de sistemas no ha sido ajena a la sinergia generada por este proceso y está proporcionando soluciones prácticas, creativas e innovadoras, como las tecnologías emergentes de e-learning (e-ABC soluciones de e-learning, 2011). En este trabajo se propone una herramienta alternativa al tradicional e-learning en el cual los objetos de conocimiento puedan ser portados sin necesidad de acceso a los recursos de conectividad de la universidad.

DESCRIPCIÓN DEL PROBLEMA

Actualmente, no existen en los laboratorios de informática de la universidad objetos de aprendizaje empaquetados dentro de un LiveCD (El rincón de Linux para Hispano Parlantes, 1998), que permitan al estudiante realizar una interacción con herramientas de software libre (Free Software Foundation, Inc. , 2012) sobre sistemas operativos tipo Linux, para aplicarlos dentro y fuera del aula de clases, sin la limitación de tenerlos instalados.

Por lo anterior, se hace complejo contar con cierto tipo de material didáctico (Documentos, Videos, Imágenes, etc.), que este en formatos que requieran tener software específico para su utilización para dictar algunas clases, sin tener que acceder al permiso de un administrador y ciertas autorizaciones para su uso dentro del aula de clases.

Otro inconveniente se presenta a la hora de hacer que dicho material didáctico y software sean portables fuera de las aulas de clases, para que los estudiantes puedan desarrollar laboratorios informáticos sin necesidad de tener que instalar dicho software en sus computadores.

Por ejemplo preguntémosnos, ¿Que sucede cuando se tiene citado a un personal para el curso de sistemas operativos o desarrollo, que requiere tener acceso a ciertas herramientas que no están instaladas de forma estándar en las aulas de informática?.

Teniendo en cuenta los antecedentes aquí expuestos, el problema que buscamos resolver dentro del ámbito de la universidad EAN con este trabajo, es la construcción de un software que permita la construcción de LiveCD personalizados para que los docentes puedan empaquetar objetos de conocimiento utilizables dentro y fuera del aula de clases por la comunidad

estudiantil, sin tener la restricción de instalar software adicional en los computadores de las aulas de clase, o en los suyos propios.

OBJETIVOS

OBJETIVO GENERAL

Construir un software utilitario basado en herramientas Linux, con interfaz grafica en java que permita empaquetar objetos de conocimiento y software dentro de un medio portable como LiveCD ó LiveUSB mediante una interfaz grafica de fácil uso por parte de los docentes, mediante estrategia LiveCD de la línea PhEANix.

OBJETIVOS ESPECÍFICOS

- ❖ Generar un estado del arte de software libre que pueda apoyar el aprendizaje en el aula de clase.
- ❖ Crear una interfaz grafica basada en java, que permita integrar los diferentes scripts del proceso manual de personalización de un LiveCD.
- ❖ Realizar la capturar de los parámetros básicos para construir un LiveCD con contenido empaquetado a partir de un LiveCD estándar.
- ❖ Realizar la captura y empaquetado de los objetos de conocimiento que va a contener el LiveCD (Videos, Imágenes, Audios, Documentos, etc.).
- ❖ Generar un medio de acceso desde el entorno del LiveCD al contenido de objetos de conocimiento empaquetado dentro de este.
- ❖ Crear un modelo de objetos de conocimiento empaquetados en el LiveCD, el cual permita la interacción entre el software a utilizar y el estudiante.
- ❖ Crear la documentación básica de las herramientas de software empaquetadas en el LiveCD, y que sean el tema de los objetos de conocimiento.

ESTRUCTURA

Este trabajo de grado sigue la línea de las investigaciones experimentales, como se ilustra en la Tabla 1.1 que muestra la correspondencia entre cada capítulo y las etapas de del proceso de investigación. En este primer (1er) capítulo explicaremos cual es la razón de ser de este proyecto, plantearemos sus objetivos y alcances.

Tabla 1.1 Estructura del trabajo de Grado vs la estructura investigativa.

Etapas de la Investigación	Capítulos del trabajo de Grado
Planteamiento del Problema	Capitulo 1: Introducción, Problema, Objetivos y Estructura.
Antecedentes y Estado del Arte	Capitulo 2: Marco Conceptual 2.1 Estado del Arte de los LiveCD. 2.2 Estado del Arte de los Objetos de Conocimiento.
Modelamiento Teórico y Diseño Experimental	Capitulo 3: Arquitectura del Software. Capitulo 4: Proceso de Diseño Software.
Aplicación de Resultados	Capitulo 5: Diseño y Construcción de los Objetos de Conocimiento.
Conclusiones	Capitulo 6: Conclusiones y Evolución

En el segundo (2do) capítulo, se tratan los principales aspectos relacionados con el desarrollo del producto final “Software para empaquetar objetos de conocimiento en LiveCD PhEANix”, iniciando por los aspectos conceptuales de los Live CD, su historia, pasando por los proyectos más significativos que se han propuesto. Adicionalmente se, se tratan conceptualmente los objetos de conocimiento, se definen algunos de los tipos y como se pueden construir para el proyecto PhEANix; así como su interrelación con las estrategias y herramientas de la formación a distancia y virtual. Se tratan algunos modelos que ha adoptado la industria para e-learning (e-ABC soluciones de e-learning, 2011) y se plantea la necesidad y posibles beneficios de producto final.

En el tercer (3er) capítulo, se aborda la arquitectura con la que se desarrollo este trabajo. Se muestran los diferentes componentes, módulos y su interrelación.

En el capítulo cuatro (4to), se aborda el diseño que se adoptó para construir este software y cumplir los requerimientos expresados en los objetivos.

En el quinto (5to) capítulo, abordamos el proceso creativo para el diseño y construcción de los objetos de conocimiento y su empaquetado dentro del LiveCD, así como las posibles herramientas que se pueden empaquetar en el LiveCD.

Para finalizar en el capítulo sexto (6to), se presentan las principales conclusiones que dejan los resultados de este trabajo y se plantean algunos lineamientos para futuras investigaciones en esta rama del conocimiento.

CAPITULO 2

2. MARCO CONCEPTUAL

2.1. ESTADO DEL ARTE DE LOS LIVECD.

2.1.1. Concepto de LiveCD.

Un LiveCD o LiveDVD (El rincón de Linux para Hispano Parlantes, 1998), más genéricamente conocido como Live Distro, es un sistema operativo acompañado de un conjunto de aplicaciones empaquetado en un medio extraíble, tradicionalmente un CD o un DVD; de ahí su nombre, este Sistema Operativo puede ejecutarse desde el CD o el DVD sin necesidad de instalarlo en el disco duro del computador.

Para esto utiliza la memoria RAM como disco duro virtual y el propio medio como sistema de archivos; algunos LiveCD incluyen una herramienta que permite instalarlos en el disco duro del computador, aunque generalmente no se efectúan cambios en el equipo donde es utilizado, es posible almacenar las preferencias de la sesión de trabajo, si así se desea.

2.1.2. Historia de los LiveCD

La principal característica de los LiveCD es que la mayoría utiliza un sistema operativo basado en el kernel Linux (El rincón de Linux para hispanoparlantes, 2012), sin embargo también existen LiveCD basados en sistemas operativos como BeOS, FreeBSD, Minix y Solaris, entre otros.

El primer LiveCD Linux fue Yggdrasil Linux, creado en 1995, aunque tuvo poco éxito promovió la idea, permitiendo la aparición de DemoLinux en el año 2000, durante la feria “Expo Linux”, celebrada en la ciudad de París.

Aproximadamente en el año 2003, el movimiento de las Distribuciones de Linux en formato Live cobró más relevancia entre los usuarios y permitió la aparición de la que se considera una de las más importantes Distros Live, la distribución alemana de Knoppix (Knopper.net, 2009), basada, a su vez, en la distribución de Linux Debian (Debian org, 2009).

Con la aparición de Knoppix, una de las mejoras que se hizo evidente en esta Distro, fue la compresión mediante cloop (Knopper.net, 2009), un nuevo driver de compresión que superaba ampliamente al antiguo loop (Knopper.net, 2009), que logró empaquetar información en una relación de 3:1; esto permitió que se pudiera colocar mayor cantidad de información y sobrepasar de esta forma la restricción física de 700MB en los CD, logrando empaquetar dentro de un CD de 650-700MB hasta 2GB, con lo cual el número de utilitarios y la calidad de los ambientes gráficos de un LiveCD mejoró considerablemente.

Desde entonces se han desarrollado múltiples proyectos a partir de esta distribución como Morphix (morphix.org, 2007), que introdujo un concepto bastante interesante en la construcción de los LiveCD, llamado modularidad. Este concepto de modularidad en principio consiste en que el sistema operativo base y los programas utilitarios que se van a empaquetar dentro del LiveCD estén en módulos independientes, de tal forma que para un mismo modulo base se puedan crear diferentes módulos dependiendo de las necesidades de quien construye (Remasteriza) la versión del LiveCD.

En Colombia se desarrollaron proyectos alrededor de esta Distro, con propósitos educativos, como lo es el caso de SciLix (Oficina de Software Libre Universidad de Cadiz, 2009); que está basada a su vez en Morphix, LiveCD y a su vez basada en Knoppix, etc.

Sin embargo no todos los proyectos estaban basados en Knoppix, o derivaban de él; aproximadamente, por el año 2004, una asociación española, Hispalinux venia impulsando su propio sistema para la creación de LiveCD, denominado MetaDistros (wikipedia, 2007) , este se convirtió en el pilar de las distribuciones Linex y Guadalinex, también de origen español y que pretenden impulsar el uso de LiveCD como herramientas para las sociedades del conocimiento; el principal aporte de esta distro fue la introducción del sistema de compresión squashfs, que permitió incrementar un 33% el factor de compresión de sistema cloop. Obviamente, este nuevo sistema de compresión más potente redundó en que los LiveCD basados en el él pudieran empaquetar más información.

A finales del mismo año, el 20 de octubre de 2004, nace oficialmente Ubuntu, la distribución LiveCD de origen Africano basada en Debian, y que es hoy en día una de las más difundidas a nivel mundial en su versión de lanzamiento salió como 4.10., que Canonical llamó “The Warty Warthog Release”, que traducido al español es “La versión del Jabalí Verrugoso”.

El proyecto Debian ponía los nombres a sus diferentes versiones basándose en los personajes de la película Toy Story, y Ubuntu montó algo original alrededor de los nombres. Todas las versiones se llaman como un animal y un adjetivo que empiezan por la misma letra, y desde la cuarta versión se mantiene un orden alfabético:

4.10 – Warty Warthog (Jabalí verrugoso), 20 de octubre de 2004.

5.04 – Hoary Hedgehog (Erizo canoso), 8 de abril de 2005.

5.10 – Breezy Badger (Tejón despreocupado), 13 de octubre de 2005.

6.06 (LTS) – Dapper Drake (Pato elegante), 1 de junio de 2006.

6.10 – Edgy Eft (Tritón impaciente), 26 de octubre de 2006.

7.04 – Feisty Fawn (Cervatillo luchador), 19 de abril de 2007.

7.10 – Gutsy Gibbon (Gibón valiente), 18 de octubre de 2007.

- 8.04 (LTS) – Hardy Heron (Garza Resistente), 24 de abril de 2008.
- 8.10 Intrepid Ibex (Cabra Salvaje Intrépida) Octubre de 2008.
- 9.04 Jaunty Jackalope (Jackalope Alegre) 23 abril de 2009.
- 9.10 Karmic Koala 29 octubre de 2009.
- 10.04 (LTS) Lucid Lynx (Lince Lúcido), 29 abril de 2010.
- 10.10 Maverick Meerkat (Suricato Inconformista) 10 octubre de 2010.
- 11.04 Natty Narwhal (Narval Elegante) 28 abril de 2011.
- 11.10 Oneiric Ocelot (Ocelote Onírico) 13 octubre de 2011.
- 12.04 LTS Precise Pangolin (Pangolin Meticuloso), 26 de abril de 2012.

2.1.3. Uso de un LiveCD

Para utilizar un LiveCD es necesario adquirirlo, la mayoría de las veces mediante una descarga vía Web de un archivo ISO (wikipedia, 2010), el cual viene listo para grabarse en CD ó DVD; luego se debe configurar el computador para que el arranque ó boot (wikipedia, 2011) sea desde la unidad lectora de CD ó DVD. Hecho esto se debe, reiniciar el computador, asegurándose que el CD o DVD esta dentro de la unidad lectora, con lo cual el LiveCD se iniciará automáticamente.

2.1.4. La selección del Live CD Base.

Como se mencionó brevemente, antes de lograr un producto terminado; el LiveCD PhEANix, se experimentó con varias distribuciones y métodos para la masterización y producción de un LiveCD completamente funcional. A continuación explicare un poco la filosofía de algunos de ellos, teniendo en cuenta los siguientes aspectos en cada uno: Características, facilidad de instalación, facilidad de actualización, facilidad de masterización, soporte en español y disponibilidad de paquetes.

- a. Knoppix: Es una distribución de GNU/Linux basada en Debian y que utiliza KDE como entorno grafico, fue desarrollada por el

consultor de GNU/Linux Klaus Knopper y se libero por primera vez en enero de 2003.

Como principales características se pueden enumerar la utilización del modulo cloop para la compresión de su filesystem, utiliza un kernel monolítico, (wikipedia, 2012) la última versión estable disponible es la 5.1.1 liberada en enero de 2007. Posee un script de instalación bastante sencillo. Por tratarse de una distribución derivada de Debian, ha heredado el manejo de paquetes de este sistema, sin embargo sus desarrolladores, han implementado mejoras construyendo sus propios repositorios.

En cuanto a la masterización tengo que decir que en la época en la que desarrolle la mayor parte de la pruebas con esta distro, no existía una comunidad hispanoparlantes muy grande y esto dificultaba las cosas, porque la mayor cantidad de información estaba en alemán o en traducciones de este al inglés. En la actualidad la comunidad hispana de usuarios de Knoppix ha crecido considerablemente y se encuentra mucha información en nuestro idioma.

b. Debian: Este proyecto de software libre orientado a la creación de una distribución Linux completamente libre fue fundado en el año 1993 por Ian Murdock, después de haber estudiado en la Universidad de Purdue. El señor Murdock escribió el manifiesto de Debian que más tarde utilizó como base para la creación de la distribución Linux que lleva el mismo nombre. Dentro de este texto el punto que más se destaca es tener una distribución de manera abierta, coherente al espíritu del núcleo de Linux y de GNU.

Como principales características se pueden enumerar la utilización del modulo cloop para la compresión de su filesystem, utiliza un kernel monolítico, la última versión estable disponible es la 4.0 Etch liberada en abril de 2007. Aunque no se trata de una Distro Live, si es la base de muchas de las distros de este tipo que

existen hoy en día. La forma de instalación es relativamente sencilla, aún cuando se deben revisar previamente los documentos relacionados con el particionamiento de los discos, especialmente si se quiere instalar compartiendo un disco con otro sistema como Windows. Una vez instalada el manejo de paquetes es bueno, sin embargo el manejo de las dependencias para la instalación y/o actualización de paquetes es complejo en las versiones 3.0 Woody y 3.1 Sarge, con las que se desarrollaron las pruebas; porque existen demasiados paquetes disponibles y en ocasiones no todos son compatibles entre sí.

Otro punto en desventaja es la gran cantidad de CD's para la instalación, entre 5 y 11, que desalientan al usuario para su descarga.

En cuanto a la masterización de estas distribuciones para convertirlas en una Distro Live el mayor inconveniente es la instalación (compilación) de cloop y squashfs (sourceforge.net, 2011) como módulos del kernel, así como el intento de actualizaciones del mismo. El interés de tener una Distro con kernel 2.6.X fue uno de los mayores inconvenientes para su actualización desde la versión 2.4.X.

En el tema del soporte existe mucha información en foros de la comunidad, tanto en inglés, como en español, sin embargo un punto no tan positivo es la dificultad de obtener soporte eficaz de la comunidad Debian por lo mismo grande. Casi que se deben agotar todos los recursos de prueba y error, antes de elevar una pregunta que alguien conteste.

c. Scilix: Nació como un proyecto del departamento de matemáticas de la Facultad de ciencias de la universidad Javeriana, orientado a dar apoyo en la educación superior. Está basada en Morphix 4.0.5, incluye aplicaciones útiles para el ambiente universitario. Su interfaz grafica es XFCE.

Una de sus principales características es la herencia de la modularidad Morphix y fomentar la masterización de LiveCD ajustados a tópicos específicos dentro del ámbito universitario. Su Creador Ofray Luna inicio el proyecto PhEANix en la universidad EAN.

En cuanto a la masterización de esta distribución me pareció un poco compleja por la poca documentación disponible en línea, que existía en el momento, sin embargo por tratarse de un derivado de Morphix, toda la funcionalidad de éste, la pude aplicar; en el escenario de probar la Distro y su potencial observe que es fácilmente instalable, pero difícilmente actualizable.

d. Morphix: Es una distribución de GNU/Linux, basada en Knoppix, y aun que usa las características de LiveCD de Knoppix, es más modular, esto significa que diferentes módulos pueden ser combinados en un CD para varios propósitos, haciendo a Morphix una especie de LEGO de construcción de LiveCD.

Por ejemplo, un LiveCD Morphix puede contener un sistema base normal, pero usar GNOME como el ambiente grafico en lugar de KDE, o se pueden tener módulos adicionales para otros propósitos más específicos, como un Firewall, un disco de rescate o una suite de oficina. La última versión disponible de Morphix es 0.5-pre5.

Las pruebas se realizaron con Morphix 0.4-1 KDE y Morphix 0.5-pre4 Gnome, ambas son muy rápidas en su ejecución, A la hora de la masterización se deben distinguir dos tipos, una la relacionada con los módulos base y otra con la construcción o modificación de mini módulos para adicionarle nuevas funcionalidades al LiveCD. En este sentido la masterización del modulo base es más sencilla que la masterización de los mini módulos.

e. Fedora: Es una distribución de Linux basada en Red Hat, (RedHat, 2012) que se utiliza para propósitos generales; esta soportada por una comunidad internacional de ingenieros, diseñadores gráficos y usuarios y cuenta con el respaldo y la promoción de Red Hat.

A diferencia de Debian, Fedora (Fedora Project, 2012) no busca incluir únicamente software libre y de código abierto, sino ser el líder en el ámbito tecnológico. Algo para resaltar es el hecho de que los cambios en Fedora se aplican sobre las fuentes de Linux, en lugar de aplicar los parches específicos en su distribución, de forma que se asegura que las actualizaciones estén disponibles para todas las variantes de Linux. En sus primeras 6 versiones se llamó Fedora Core, por lo que solo incluía los paquetes más importantes del sistema operativo. La última versión es Fedora 8, la cual fue liberada el 8 de noviembre de 2007.

Para muchos Fedora en sus diferentes versiones, se trata de una distribución muy flexible y estable, tiene un número considerable de paquetes disponibles en sus repositorios, actualmente están soportando la instalación de paquetes de terceros y han implementado diversos paquetes de construcción basados en el proyecto Mono, que permiten a los usuarios crear personalizaciones de la distro a partir del Fedora original. Esta distro también cuenta con su proyecto LiveCD que actualmente va por la versión 12.

f. MetaDistros: Es un proyecto que tiene como objetivo la creación de una infraestructura para la creación de distribuciones de Linux fácilmente.

Una metadistro consta de dos partes, el calzador y una copia del sistema de archivos GNU/Linux comprimido mediante una Imagen Squashfs. El calzador es un conjunto de scripts, módulos del kernel y aplicaciones, como por ejemplo busybox, que preparan el ambiente, montan el sistema de archivos en RAM y auto configuran el sistema para que el usuario final solo tenga que poner el cdrom en su computador para que cargue en el arranque.

Cuando inicio el proyecto MetaDistros, el calzador estaba basado en la versión de kernel 2.4.27 con varios parches, y estaba configurado de manera muy personalizada que lo hacían bastante complejo a la hora de intentar una actualización del kernel a versiones más modernas (2.6.X) ya que se necesitaba configurar de manera especial el kernel para que por ejemplo el sistema de archivos, soporte SCSI, y soporte para chips de tarjetas estuviera dentro del kernel y no como módulos independientes.

Sin embargo hoy en día con la nueva estructura del calzador que está implementada en distros como Guadalinex y Ubuntu, el calzador se ha integrado en un archivo de arranque.

Lamentablemente, la documentación original del proyecto promovido por Hispalinux, ya no se encuentra disponible en internet, sin embargo varias distros entre ellas Ubuntu han retomado los métodos de este proyecto para sus creaciones. El sistema actual del calzador metadistros se basa en initramfs-tools, el cual genera una imagen comprimida de un sistema linux mínimo, que es el encargado del arranque de la distro Live o la distro instalada, actualmente este sistema esta implementado junto con yaird para los kernels desde la versión 2.6.13 en adelante.

- g. Ubuntu: Es una distribución Linux que ofrece un sistema operativo enfocado a computadoras de escritorio aunque también

proporciona soporte para servidores, está basada en Debian GNU/Linux y concentra su objetivo en la facilidad de uso y la libertad de uso y la facilidad en la instalación.

El proyecto Ubuntu (canonical, 2012) es de origen africano y es patrocinado por la compañía Canonical, una empresa privada fundada y financiada por el empresario sudafricano Mark Shuttleworth; el nombre de la distribución proviene del concepto zulú y xhosa de Ubuntu, que significa "humanidad hacia otros" o "yo soy porque nosotros somos". Su eslogan de, Ubuntu – “Linux para seres humanos” (en inglés "Linux for Human Beings") –, resume una de sus metas principales del proyecto que es hacer de Linux un sistema operativo más accesible y fácil de usar para todos.

Con Ubuntu se han probado casi todas las versiones desde la 5.04 Hoary Hedgehog ó Erizo vetusto hasta la 11.10 Oneiric Ocelot. La primera versión estable de este proyecto se baso en la versión 7.04 Feisty Fawn ó Cervatillo luchador.

En esta versión, se reúnen todas o casi todas de las características para el desarrollo de un proyecto como PhEANix. Tiene facilidades para su instalación, para su actualización y personalización, así con innumerable información tanto oficial como proveniente de las experiencias de otros usuarios de Ubuntu, Debian y otras distros que están interesados en el fenómeno Ubuntu, y que le aportan a estos proyectos de manera continua en los foros Internet.

2.2. ESTADO DEL ARTE OBJETOS DE CONOCIMIENTO

2.2.1. Definición de Conocimiento

Existen diferentes formas de abordar el tema; para efectos de este trabajo hablaremos del conocimiento desde el punto de vista filosófico y el conocimiento desde el punto de vista científico.

El termino conocimiento, en el sentido más general, es una interpretación que hace un individuo o un grupo de individuos a partir de la posesión de diversos datos interrelacionados entre sí, que de forma independiente tiene un menor valor o ponderación cualitativa. Esto en últimas no es más que la formación de un modelo mental de la realidad sobre cualquier cosa, actividad o ser, etc., que se forman los individuos que adquieren el conocimiento.

Generalmente podemos afirmar que el conocimiento comienza desde los sentidos, luego pasa de estos al entendimiento y de allí termina en la razón.

En el caso del entendimiento, se dice que existe un uso meramente formal del mismo, lo que significa un uso lógico; en el caso de la razón, esta hace abstracción de todo un contenido, pero también hay un uso real.

Finalmente el saber es el conjunto de conocimientos acumulativos en el tiempo, y que generan un pensamiento continuo, basado en los recuerdos de los conocimientos adquiridos previamente.

El conocimiento entonces no es más que una relación entre sujeto (el que observa), y objeto (el que es observado). Bien, para entenderlo mejor, pensémoslo de la siguiente manera:

A un ser se le considera como un objeto por la relación al objeto que está asociado ó que representa; mientras que a otro se le considera como un sujeto, por la relación al sujeto que representa.

El conocimiento es un fenómeno complejo que implica permanentemente cuatro elementos, Sujeto, Objeto, Operación y Representación mental.

La representación interna es el proceso Cognoscitivo del individuo que construye conocimiento, y no es otra cosa que la explicación a su propio criterio.

Ver Anexo 4.

2.2.2. Concepto de Objetos de Conocimiento

Una vez definido que es conocimiento veamos que son los objetos de conocimiento propiamente dichos. Conceptualmente, no son nada novedosos, son la extrapolación de la Programación Orientada a Objetos al campo de la educación y las implementaciones digitales. En Inglés se les conoce como LO's ó "Learning Objets", con esta sigla se hace referencia a las unidades mínimas de representación de conocimiento o aprendizaje (páginas, texto, imagen, vídeo, etc.) por cuya agrupación se logran crear proyectos formativos complejos y completos.

Estos objetos encajan perfectamente dentro de las teorías constructivistas del aprendizaje, que tanto impacto tienen en la actualidad. A continuación presentaré algunas de las propiedades y consideraciones que debe cumplir un objeto de conocimiento para que se pueda considerar como tal:

- Estar auto-contenido: Significa que se pueden comprender por sí mismos, aún en su mínima unidad.
- Ser reutilizable: Significa que se les pueda emplear con diferentes finalidades en diferentes medios.
- Es actualizable: Significa que por estar empaquetados digitalmente debe ser más sencilla su actualización de forma ó contenido.
- Que pueda comunicarse con sistemas LMS: Los LMS “Learning Management System”, que son plataformas de enseñanza en un sistema E-Learning, con el fin de impartir formación en línea ó en red.
- Que sean Inter-operables: Es decir, que se pueda portar a diferentes plataformas y siga operando correctamente.
- Es agregable: Significa que se pueden agrupar varios de ellos formando una entidad de orden mayor, como por ejemplo, un curso tradicional.

Los actuales estándares de E-Learning se orientan hacia este tipo de tecnologías. Uno de estos es el estándar SCORM (Sharable Content Object Reference Model), (moodle.org, 2009) tal vez el de mayor aceptación actualmente, que se basa en la agrupación de unidades mínimas de conocimiento llamadas “Shareable Content Objects” (SCOs).

La finalidad de estos Objetos de Conocimiento es poder servir como piezas de un gran LEGO digital, que por su propiedad de agrupación son capaces de generar verdaderos contenidos educativos. Por tratarse de piezas que se integran en un todo, a los OLs les podemos determinar su “granularidad”, ó el nivel de agregación de un determinado contenido. Para entenderlo mejor diremos que el nivel máximo nivel de agregación o granularidad lo representa todo un curso, mientras que el mínimo nivel de granularidad está representado por una imagen, un sonido, o una ilustración individual, ó lo que llamaremos mínimo nivel de granularidad, como se observa en la siguiente ilustración.

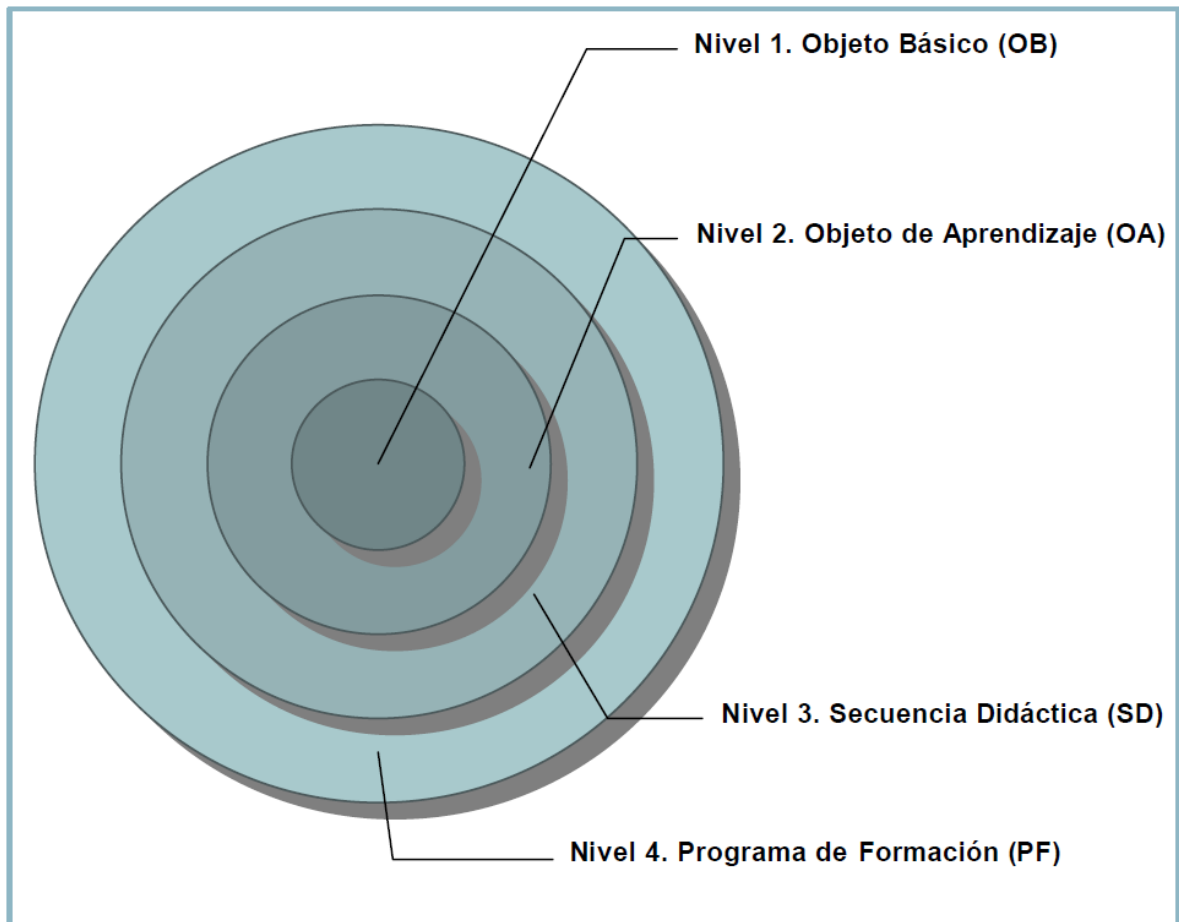


Ilustración 1: Gráfico Arquitectura de Niveles de Agregación de ODE según LOM-ES v.1.0

2.2.3. Tipos de Objetos de Conocimiento que se podrán empaquetar en el LiveCD.

Dado que uno de los objetivos de este proyecto, es proporcionar a la comunidad educativa de la EAN una herramienta de software que le permita empaquetar objetos de conocimiento en formato LiveCD Linux de diferente índole; a continuación enunciaremos y describiremos algunos tipos de objetos de conocimiento que se pueden empaquetar mediante el uso del prototipo de interfaz desarrollado.

- h. Documentos How To (Tradicionales ¿Cómo Hacer?): Son documentos de carácter informal, generalmente cortos, que describen paso a paso cómo cumplir con una cierta tarea. Son documentos propios de la jerga informática o electrónica, que son creados generalmente para ayudar a las personas noveles en un problema, suelen dejar de lado los detalles del mismo, solo para los expertos. En general son documentos que resumen el tema tratado.

- i. Videos de pantalla (Screen Cast): Son documentos digitales basados en videos capturados de la salida de pantalla de un computador, se pueden combinar con una narración de las acciones que se están ejecutando en el momento de realizar la captura. El termino screencast fue acuñado en el año 2004 por el columnista Jon Udell, que invitó a los lectores de su blog a proponer nombres para esta tecnología emergente. Udell seleccionó el término screencast, el cual fue propuesto por Joseph McDonald y Deeje Cooley.

Los Screencasts son útiles para demostrar las características de algún software específico, o tener un material de formación. Un video sustituye las explicaciones escritas potencialmente ambiguas, o simplemente facilita la tarea de mostrarles a terceras personas como realizar una tarea en un entorno de software específico.

Los Screencasts son herramientas excelentes para difundir el uso de software de una forma rápida, eficaz y a un bajo costo por lo que esta tecnología se está masificando entre los centros de formación formal e informal, como facultades y centros de formación empresarial, etc.

j. Tutoriales: Son sistemas instructivos de auto aprendizaje que mediante el uso de herramientas de software generalmente simulan al instructor y permiten al usuario el desarrollo de algún procedimiento o los pasos para realizar determinada actividad. Generalmente un sistema tipo tutorial contiene cuatro grandes fases: La fase introductoria, que genera motivación y se centra en la atención de quien lo está tomando; La fase de orientación inicial, en la que se ofrece la codificación, almacenaje y retención de lo aprendido; La fase de aplicación, en la que se presentan métodos para retener y aplicar lo aprendido; y la fase de retroalimentación en la que la persona que recibe el conocimiento demuestra lo aprendido y se le ofrece retroinformación y refuerzo.

Un tutorial normalmente consiste en una secuencia lógica que va aumentando el grado de dificultad y entendimiento requerido por el usuario y que se hace evidente en la presentación del mismo, a fin que este entienda todos los componentes. El término tutorial, tan de moda en los círculos informáticos, es un neologismo de origen inglés

k. Preguntas y Respuestas Frecuentes (FAQ, acrónimo del inglés Frequently Asked Questions): Se refiere a una lista de preguntas y respuestas que surgen frecuentemente dentro de un determinado contexto y para un tema en particular.

En el contexto de los Sistemas, las FAQ se originaron de la lista de correo de la NASA a comienzos de la década de los ochenta. Las primeras FAQ fueron desarrolladas comenzando en 1982, cuando el almacenamiento de información tenía un elevado costo. Esta costumbre de publicar FAQ se extendió a otras listas de correo. La primera persona en publicar un FAQ semanal fue Jef Poskanzer en USENET

net.graphics/comp.graphics. Hoy en día, en los USENET, preguntar por asuntos cubiertos por los FAQ se considera una falta de “**netiquette**”, dado que se supone que el usuario no se preocupó de revisar esta lista antes de hacer sus preguntas.

Para efectos prácticos de este proyecto se espera que la comunidad de estudiantes de la EAN en cabeza de la facultad desarrollen una lista FAQ alrededor del proyecto PhEANix, a fin de facilitar el acceso, uso y masificación del Live CD en el entorno de clases.

2.2.4. ¿Por qué son relevantes los objetos de conocimiento para el proceso educativo?

Para nadie es un secreto la creciente relevancia que los procesos de formación superior están tomando en nuestro país como consecuencia de las necesidades de especialización y la demanda de mayores exigencias por parte del mundo laboral. Sin lugar a dudas la formación es uno de los factores más importantes en el desarrollo de Colombia dentro del ámbito regional y global, teniendo como norte de este proceso la mejora de las competencias de los ciudadanos y por ende del Estado en general.

Hasta la década pasada un título universitario nos lanzaba al mercado laboral de forma sistemática. Hoy por hoy, nos encontramos con que el contexto en el que salen los profesionales al mercado laboral ha cambiado sustancialmente por distintos motivos. Entre otros están la exigencia del mercado de mayores y mejores competencias de formación, y una creciente saturación del mismo en diversas áreas. Esto ha conducido un proceso de acomodación por parte de las universidades, que conscientes de la necesidad de una formación que haga más competitivos a sus estudiantes, busca nuevas herramientas que permitan cubrir más y mejores contenidos.

Los diferentes programas de post-grado como diplomados, licenciaturas, etc., sitúan al egresado en el punto de partida laboralmente hablando, situación que anteriormente era considerada como suficiente, y en ocasiones, el punto final. Existen dos reflexiones de todo esto, la primera es que el profesional debe continuar permanentemente su proceso de formación académica; la segunda que las instituciones de formación superior deben adoptar tecnologías y herramientas que les permitan brindar este tipo de formación continuada a sus egresados.

2.2.5. El proceso de educación continuada, la autoformación y el uso de LiveCD

El concepto de educación a distancia y autoformación ha evolucionado mucho, especialmente en lo relacionado con las herramientas que apalancan dicho concepto. Desde sus inicios en 1728, cuando la Gaceta de Boston publicó material auto-instructivo para sus lectores preparado por un instructor, hasta el presente donde el internet y los medios multimedia ofrecen la calidad e inmediatez, para que cualquier persona en cualquier parte adquiera o comparta el conocimiento que se está generando.

Sin embargo en ciertas circunstancias se hace necesario que la persona quien recibe el conocimiento pueda tener un escenario de aprendizaje experiencial, que no necesariamente está cubierto por el internet y la multimedia. Es el caso del aprendizaje del sistema operativo Linux, de no poder llevarse al plano práctico, el aprendizaje se vuelve muy tedioso e incluso complicado más allá de lo que en realidad es.

Para solventar esta situación, PhEANix propone la utilización de la tecnología LiveCD como medio de empaquetamiento para los objetos de conocimiento, en

beneficio del modelo de formación a distancia y auto-aprendizaje. Esta tecnología reúne todo, teoría y práctica sobre una materia específica, en una sola herramienta, que es portable, cuenta con la ventaja de no depender de conexiones en línea y colocándolo a un bajo costo de producción en manos de la comunidad educativa.

Es importante entender que esta, no es más que una herramienta adicional a las ya implementadas y que eventualmente va a permitir potencia aun más su aprovechamiento en el entorno académico. Con su adopción, será posible realizar programas educativos especializados desde cualquier ubicación geográfica, sin tener que asistir presencialmente a laboratorios en un contexto físico. Se podrá maximizar la capacidad operativa no solo de los educadores sino de los educandos.

En combinación con el uso de internet y de las tecnologías de e-learning, el LiveCD con contenidos de auto-aprendizaje empaquetados, se convierte en un poderoso aliado en el aula, o fuera de ella. Este tipo de herramientas acercaran más las relaciones y la comunicación entre el alumnado que recibía educación a distancia tradicional y la institución. Como todo paso adelante en el uso de nuevas tecnologías, los docentes deberán ponerse a tono con la situación y ser ellos los primeros en adoptar el uso del LiveCD, y hacer los ajustes metodológicos que esto requiera.

2.2.6. Problemas planteados al modelo de empaquetar objetos de conocimiento en un LiveCD:

En la construcción del LiveCD PhEANix se ha considerado que el usuario tenga una buena experiencia tanto de uso del Linux, como a nivel formativo, por lo que se presta especial atención en temas como el idioma de localización de los programas instalados, entre otros.

Por otra parte es muy importante que los objetos de conocimiento sean creados desde la óptica de la docencia, por equipos multidisciplinarios para que se puedan cumplir con las premisas que supone trabajar con material didáctico, que será utilizado con propósitos de auto aprendizaje.

Es necesario prestar especial atención en el tema pedagógico para lograr que estos sean verdaderamente interesantes y logren enganchar al receptor y transmitir un mensaje claro.

Otro punto en el que se ha prestado especial atención es en la cohesión de los contenidos, tanto en su interior como en su apariencia externa. Para evitar precisamente que los diferentes fragmentos de conocimiento tengan poca o ninguna relación formal entre sí, y con el contenido educativo total. Estos serán entonces temas que deberán ser revisados detenidamente por futuros proyectos que aborden el tema bien para darle continuidad a los contenidos empaquetados, ampliándolos, corrigiéndolos, o para abrir nuevas líneas de conocimiento que deseen hacer uso de esta plataforma LiveCD.

Este proceso es imparable, y la comunidad universitaria lo sabe y entiende que supone la reutilización de contenidos como motor de empuje para la creación de repositorios de Objetos de Conocimiento.

3. ARQUITECTURA DEL SOFTWARE

En este capítulo se presenta la arquitectura a nivel de software con la que se desarrolló esta tesis, se muestran cuales son los componentes, herramientas y módulos que la componen, y su interrelación.

3.1. Arquitectura General

Tal como en la construcción de edificaciones, la arquitectura del software, nos presenta la hoja de ruta general para el desarrollo de aplicaciones. Nos permite plasmar los requerimientos y necesidades que se desean solucionar y visualizar los diferentes elementos que se van a integrar, así como el flujo de información que deseamos procesar y el resultado que podemos obtener.

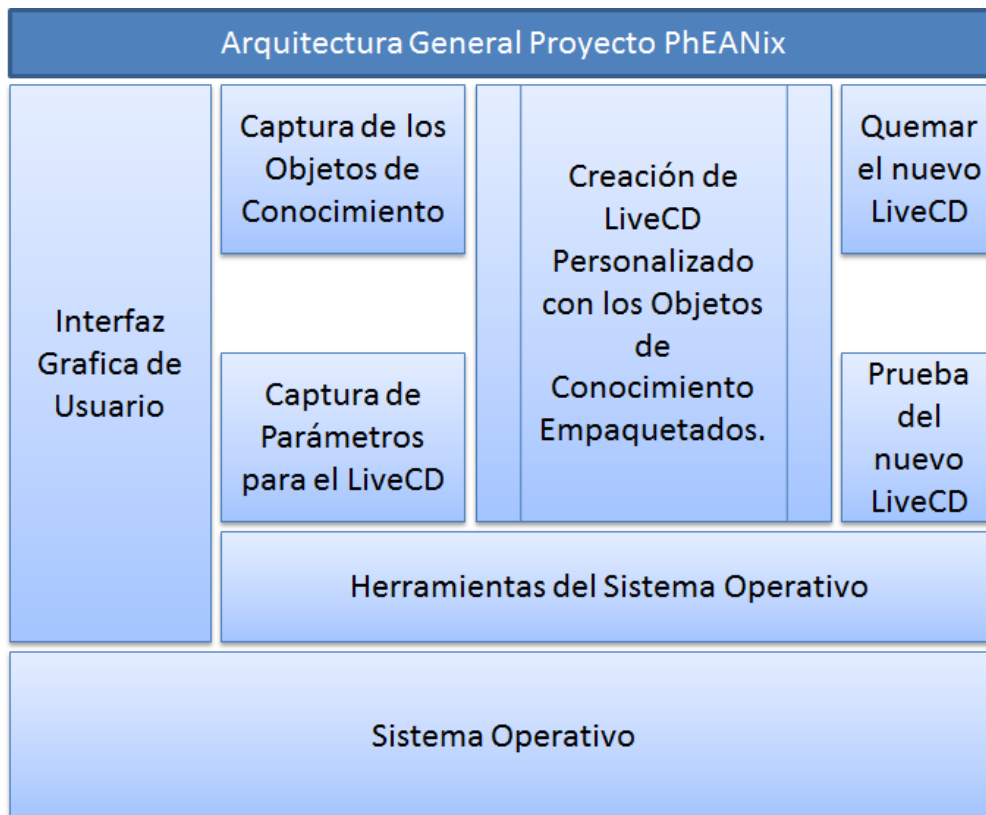


Ilustración 2: Arquitectura del Proyecto PhEANix

En la arquitectura que hemos establecido para el proyecto PhEANix podemos observar cada uno de los componentes que en conjunto permiten cumplir con el objetivo de empaquetar objetos de conocimiento dentro de un LiveCD.

Algunos de los componentes no son parte del desarrollo, sino que son componentes externos que se ha integrado desde el sistema operativo y otros desde utilitarios adicionales. Entre los componentes externos que se pueden citar están: squashfs, bash scripting, qemu, entre otros.

Los componentes desarrollados a medida están relacionados directamente con la interfaz de usuario que permita la captura de parámetros para la creación del LiveCD y algunos bash scripts al mismo son externos.

3.2. Módulos de software desarrollados.

Estos son los módulos que se desarrollaron específicamente para tener una interfaz gráfica que permitiera el control de la aplicación de forma sencilla e intuitiva.

3.2.1. Captura de los Objetos de Conocimiento a Empaquetar:

Permite capturar la lista de Objetos de Conocimiento a Empaquetar dentro del LiveCD, entre los que se encuentran imágenes, videos, sonidos, documentos tipo Office.

3.2.2. Captura de los Parámetros de Personalización del LiveCD:

Permite la captura de los parámetros para personalizar el LiveCD, tales como nombre del LiveCD, usuario, personalización manual ó automática, idioma del LiveCD, tipo de Escritorio.

3.2.3. Lanzador del Proceso de Personalización:

Verifica los parámetros capturados e inicia el proceso de personalización iniciando los respectivos scripts bash pasándole los parámetros capturados.

3.2.4. Prueba del LiveCD Personalizado:

Permite iniciar el nuevo LiveCD en un ambiente Virtualizado, antes de decidir si el producto terminado está acorde con lo que el usuario desea. Esto evita tener que generar un LiveCD cada vez para probar.

3.2.5. Generar LiveCD en medio físico.

Una vez ya se ha determinado que el LiveCD generado está acorde con lo que se desea permite iniciar el nuevo LiveCD en un ambiente Virtualizado, con el fin de probarlo y realizar las verificaciones de uso necesarias, luego de las cuales se puede generar la copia de la imagen ISO en un medio físico. Esto evita tener que generar un LiveCD en físico cada vez que se desea probar.

3.3. Módulos de software integrados.

Estos módulos corresponden a los scripts que hacen uso de las herramientas del sistema operativo y sus utilitarios que permiten la creación de un nuevo LiveCD con los objetos de conocimiento empaquetados dentro de este. Los módulos integrados, no tienen interfaz grafica, y su interacción con el usuario depende de los anteriores.

3.3.1. Descompresión de la ISO Plantilla:

Al iniciar el proceso de generación del nuevo LiveCD, se parte de una imagen ISO de una distribución Ubuntu, la cual se monta en un directorio de trabajo y se descomprime dentro de un subdirectorio para extraer tanto los elementos que hacen booteable un LiveCD como el sistema de Archivos comprimido con squashfs.

Este ultimo (Sistema de Archivos), contiene el sistema Linux que personalizaremos.

3.3.2. Creación del ambiente de Trabajo CHROOT:

Una vez realizada la descompresión de la imagen ISO Base, es necesario crear un ambiente independiente al del sistema Linux desde donde se hace la personalización; para esta labor se hace uso del utilitario CHROOT de Linux.

El ambiente generado es una jaula que permite hacer modificaciones al nuevo LiveCD como si se tratara de una instalación en una maquina nueva.

3.3.3. Empaquetado de Objetos de Conocimiento y sus Accesos:

Una vez se tiene descomprimidos los elementos que componen el LiveCD, se procede con la copia de la carpeta que contiene los objetos de conocimiento que hemos capturado previamente; a la ubicación predeterminada dentro de los scripts. Esto permite que al re-comprimir el sistema Linux en formato de LiveCD los Objetos de Conocimiento hagan parte de este.

3.3.4. Adición Modificación de Paquetes de Software:

Al igual que el modulo anterior, este modulo permite realizar la instalación de nuevos programas de software para el sistema Linux que se está personalizando. Por ejemplo se puede instalar un nuevo navegador, ó tal vez un nuevo plugin para reproducir multimedia.

Vale la pena aclarar que se utiliza como gestor de paquetes *synaptic*.

3.3.5. Reconstrucción del nuevo LiveCD:

Al finalizar el proceso de personalización del nuevo LiveCD (instalar Software e insertar los Objetos de Conocimiento), el ambiente de trabajo está listo para ser re-comprimido, para lo cual se utilizan nuevamente utilitarios del sistema operativo como *squashfs* y *mkisofs*.

De esta forma se comprime el contenido del sistema de archivos Linux y se genera nuevamente un archivo ISO con el nuevo LiveCD.

3.4. Parámetros de entrada.

3.4.1. Lista de Documentos (Objetos de Conocimiento):

Desde la interfaz gráfica el usuario cuenta con un control *Jtree* (Árbol), que contiene nodos principales que representan los tipos de objetos de conocimiento que se pueden adicionar en el LiveCD. Mediante un control *JButton*, se abre un cuadro de Archivos, que permite seccionar los Archivos, acorde con el filtro de si son videos, imágenes, sonidos, documentos, otros.

En la captura se genera un parámetro que contiene la ruta y el nombre del archivo a incluir. `/ruta_absoluta/archivo.extension`

Al iniciar el proceso de generación del nuevo LiveCD, se parte de una imagen ISO de una distribución Ubuntu, la cual se monta en un directorio de trabajo y se descomprime dentro de un subdirectorio para extraer tanto los elementos que hacen booteable un LiveCD como el sistema de Archivos comprimido con *squashfs*.

Este último sistema de Archivos contiene el sistema Linux que personalizaremos.

3.4.2. Idioma del LiveCD.

Este parámetro permite seleccionar mediante un control desplegable el idioma que tendrá el LiveCD por defecto tanto para el Boot como en la sesión de trabajo.

3.4.3. Ambiente del Desktop (KDE, Gnome, Otros):

Este parámetro permite la selección del ambiente gráfico que se va a instalar en el LiveCD, este parámetro determina si se instalará gnome, KDE, XFCE, otros.

3.4.4. Imagen ISO Base para usar en la Masterización:

Este parámetro permite la selección del archivo IOS Base que se va a utilizar para realizar la personalización, (Los ISO que están probados son de la distribución Ubuntu.).

3.4.5. Nombre Etiqueta del LiveCD (32 caracteres Máximo):

Este parámetro permite la selección de la etiqueta que tendrá el LiveCD y que identificará al CD ó DVD, (El máximo de caracteres permitidos es de 32).

3.4.6. Personalizar Manualmente o Automáticamente el LiveCD.:

Este parámetro permite determinar si el proceso de personalización cargue el Gestor de Paquetes synaptic y/o una consola bash carácter, con la finalidad de realizar personalizaciones avanzadas.

3.4.7. Borrar Archivos del LiveCD visibles en Ambiente Windows:

Este parámetro determina si se conservan o borran los archivos y directorios del LiveCD que son Visibles desde un entorno Microsoft Windows.

3.4.8. Confirmar Opciones e Iniciar el proceso.

Este parámetro verifica que todos los parámetros requeridos estén correctos e inicia el proceso de generación del LiveCD invocando los scripts bash y pasándoles los parámetros capturados.

3.5. Archivos de Entrada.

3.5.1. Imagen ISO de un LiveCD Basado en Ubuntu.

Los archivos ISO de entrada al proceso de personalización son imágenes de LiveCD Ubuntu, de arquitectura X86.

3.5.2. Documentos (Objetos de Conocimiento).

Los documentos que se van a empaquetar para conformar los objetos de conocimiento dentro del LiveCD son archivos de tipo imágenes (jpg; png; gif; bmp; etc.), videos (avi, mp4, flv, etc), sonidos (AU. RA. Mp3, etc.), documentos (DOC, XLS, PPT, ODF, etc.) y finalmente otros que pueda considerar el usuario como (html, xml, etc.).

3.6. Archivos de Salida.

3.6.1. Imagen ISO de un LiveCD Basado en Ubuntu:

Después de haber terminado el proceso de creación del LiveCD y empaquetado los Objetos de Conocimiento el resultado será otro archivo ISO que herede las características de la imagen base, adicionando nuevas características de software y/o suprimiendo algunas que el usuario considere.

3.6.2. Documentos (Objetos de Conocimiento):

En la salida los documentos seleccionados estarán organizados en una carpeta que hará parte del LiveCD y estarán referenciados mediante un índice en formato HTML que será el que el usuario pueda usar para accederlos.

3.7. Flujo Funcional de la Aplicación.

En resumen de todo lo expuesto en este capítulo, el aplicativo funciona de la siguiente forma:

3.7.1. Captura de información:

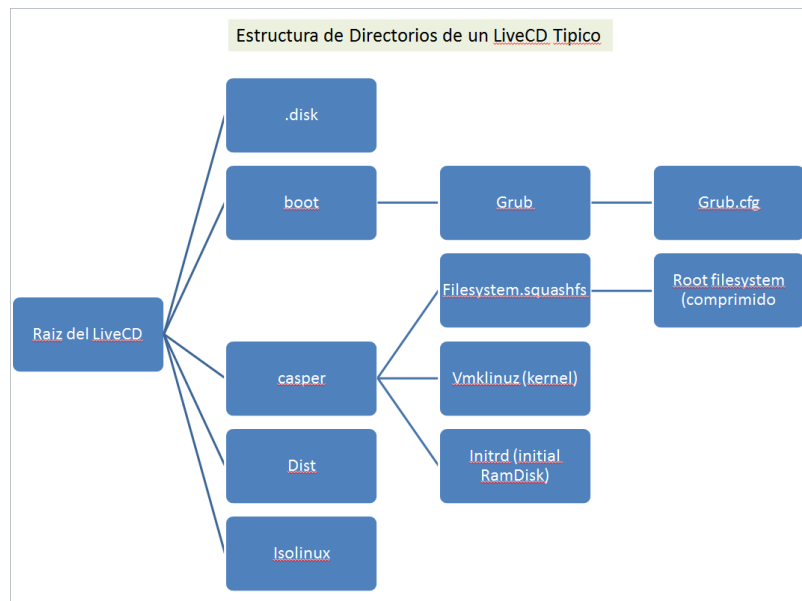
Se inicia la interfaz grafica, dentro de la cual se captura la información de los Objetos de Conocimiento a Incluir, así como los parámetros de personalización del LiveCD.

3.7.2. Inicio de la Personalización:

La interfaz grafica invoca los scripts bash pasándole los parámetros capturados. Este macro proceso está compuesto de los siguientes pasos:

- Preparar el ambiente de trabajo.
- Copia la fuente del sistema al directorio de trabajo.
- Generar el ambiente independiente con chroot en el nuevo sistema.
- Efectuar las modificaciones.
- Prepare el árbol de directorios del CD.
- Construir de nuevo el CD / DVD

La siguiente ilustración muestra como está construido un LiveCD en términos generales.



Los trabajos de personalización que permite realizar el software que se está creando están enfocados a permitir la adición de material didáctico en forma de documentos dentro del LiveCD y del software necesario para realizar su visualización adecuada, así como programas que el docente pueda llegar a requerir para impartir sus cátedras.

CAPITULO 4

4. PROCESO DE DISEÑO DE SOFTWARE

El presente capítulo explica los pasos que se ha desarrollado para poder implementar el software en las condiciones en las que se encuentra.

Presenta cuales han sido los requerimientos planteados a ser resueltos específicamente y el porqué de algunas consideraciones respecto de las herramientas utilizadas.

4.1. Requerimientos Funcionales

Teniendo en cuenta que los requerimientos funcionales expresan la interacción del sistema con los usuarios y con otros sistemas que se encuentren en su entorno, a continuación se describen los requerimientos funcionales para la construcción del software que va a crear los LiveCD con Objetos empaquetados basados en Linux.

1. El sistema debe contar con una interfaz grafica amigable.
2. El sistema debe permitir la captura de los parámetros básicos para la masterización de los LiveCD Basados en Ubuntu.
3. El sistema debe permitir la captura de los Objetos de Conocimiento que se van a empaquetar dentro del LiveCD.
4. El sistema debe permitir la modificación, adición y/o eliminación de software de los LiveCD base.
5. La interfaz debe permitir la visualización del avance del proceso.
6. La interfaz debe permitir contar con una salida de LOG para el seguimiento detallado del proceso.

4.2. Requerimientos de Integración del Software

Los requerimientos de integración son aquellos que le van a permitir a los diferentes componentes de la aplicación inter operar adecuadamente, teniendo en cuenta que el proceso de construcción de un LiveCD se puede realizar de diversas formas y que en él intervienen múltiples utilitarios del sistema operativo Linux.

1. La interfaz debe permitir la integración con herramientas del sistema operativo Linux que se invocan desde Shell script.
2. La herramienta debe pasar mediante parámetros la información que se captura en la interfaz grafica a los scripts de Shell.
3. La interfaz grafica será desarrollada en Java (Eclipse).

4.3. Requerimientos de Hardware y Software

Los requerimientos de hardware y software corresponden a las características mínimas que se debe cumplir para poder ejecutar el aplicativo, y que permiten su correcto funcionamiento, una vez se despliegue.

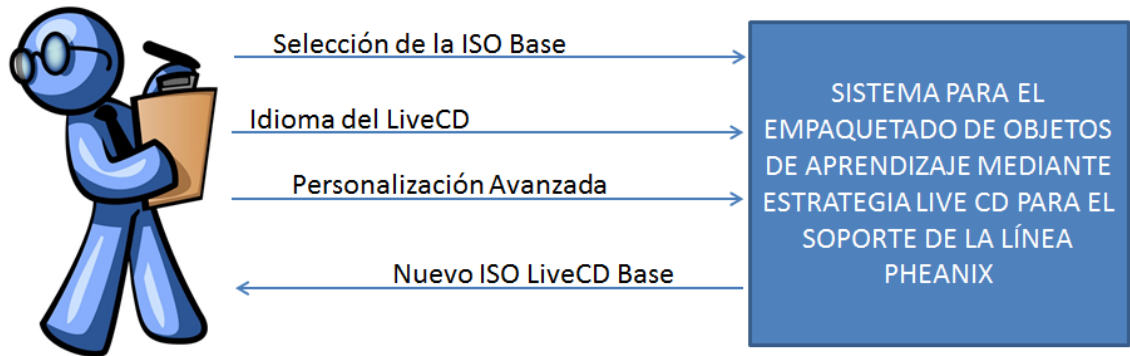
1. Contar con un PC con las siguientes características:
 - a. 1 Procesador DUAL Core 64Bits o superior
 - b. 4 GB de RAM o superior.
 - c. Disco Duro de 100GB Libres o superior
 - d. Sistema Operativo Linux Ubuntu 12.04 64 Bits.
 - e. 1 Tarjeta de RED, con conexión internet.
 - f. 1 Unidad de DVD-RW.
2. Contar con un ambiente de ejecución Java 6 o superior instalado.
3. Contar con las siguientes utilidades Linux instaladas en las versiones acordes a la versión de Ubuntu instalada.

- a. Bash
- b. Sudo
- c. Gfxboot gfxboot-dev yaboot
- d. Dialog xdialog zenity kdebase-bin
- e. Genisoimage
- f. Squashfs-tools
- g. Dpkg-dev
- h. Fakeroot
- i. Build-essential
- j. File
- k. Mktmp
- l. Xauth
- m. Gnnupg
- n. Wget
- o. Lzma
- p. Brasero kb3 burn
- q. Syslinux
- r. Qemu
- s. Xterm

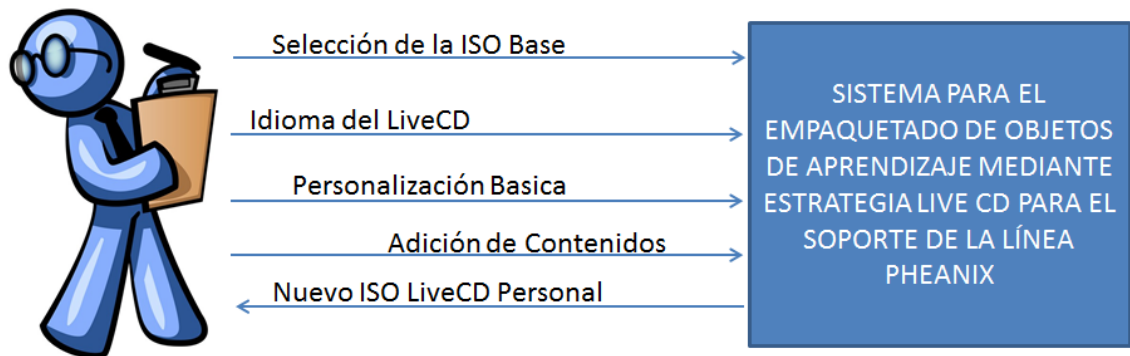
4.4. Casos de Uso para los cuales aplica.

A continuación se presentan y explican los casos de uso para los cuales aplica este trabajo de tesis.

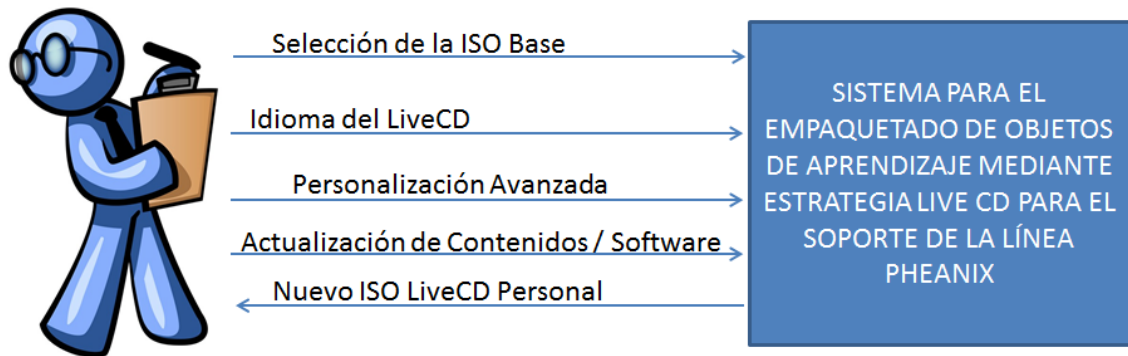
El primer caso de uso es la construcción de un LiveCD plantilla que se utilice en adelante para ser la base de las siguientes personalizaciones. En la siguiente grafica se ilustra Dicho proceso:



El segundo caso es la construcción de un LiveCD personalizado para una clase específica, que involucre la inclusión de material didáctico y la instalación de software.



Existe un tercer caso, que es una derivación del segundo y es la construcción de un LiveCD personalizado para una clase específica, que involucre la actualización del material didáctico y/o del software instalado.



4.5. Diseño de la Interfaz

Dados los requerimientos expuestos para la construcción del software, la interfaz se diseño teniendo en cuenta que en una misma ventana se pudiera capturar la mayor cantidad de información.

Se organizo la información en pestañas con nombre intuitivos para el usuario, permitiendo diferenciar fácilmente la información de Objetos de Conocimiento y la información de personalización del LiveCD Base.

Es así como se opta por que el desarrollo se realice en Java Eclipse y que se utilicen diversos clases contenedoras.

La principal es un JFrame que a su vez contiene un contenPane con 3 paneles cada uno de los cuales contiene los objetos necesarios acorde con la función que se designó.

Como se puede observar en la ilustración 3, cada uno de los tres paneles se destinó para contener información relacionada con el material didáctico, los

parámetros de personalización básicos y los parámetros de personalización avanzados.

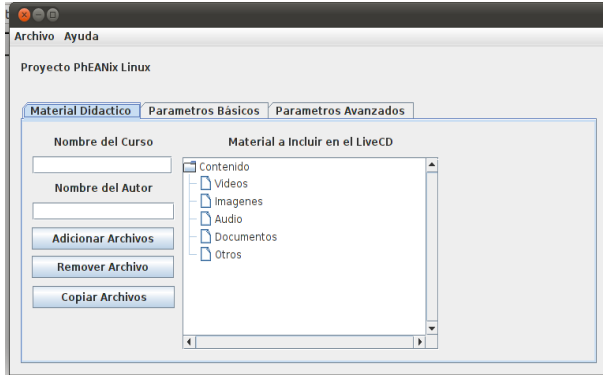


Ilustración 3: Pantalla Principal del Programa

Dentro del Panel para la captura del material didáctico existen objetos tipo Jtree, JLabel, JButton, JText y JScrollPane que le permiten a la interfaz captura la lista de documentos a incluir dentro del LiveCD.

Dentro del Panel, para la captura de la información de personalización Básica, se cuenta con objetos de tipo JComboBox, JLabel, JButton, JText y JRadioButton que le permiten capturar los parámetros para la personalización y lanzar el inicio de la ejecución de los scripts bash.

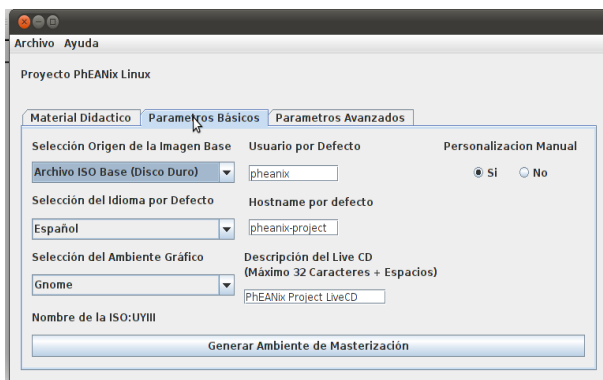


Ilustración 4: Pantalla de Captura parámetros Personalización

La interfaz adicionalmente cuenta con objetos de tipo Jmenu que le permiten realizar algunas tareas como salir, desplegar la Ayuda y mostrar los créditos y la versión.

En términos generales la interfaz grafica está diseñada para contener toda la información necesaria en el proceso de creación y/o personalización de un LiveCD y el empaquetado de contenido dentro de este.

En el anexo 1, se presenta la forma de navegación de la aplicación por su interfaz gráfica, hasta lograr el producto final (LiveCD Personalizado).

CAPITULO 5

5. DISEÑO Y CONSTRUCCIÓN DE LOS OBJETOS DE CONOCIMIENTO

En este capítulo, se proponen algunas consideraciones que deberán ser tenidas en cuenta por los usuarios del proyecto, que aborden la creación de sus propios objetos de conocimiento, con el fin de que el material que finalmente sea empaquetado dentro de los LiveCD que se produzcan a partir de este, sea coherente y aporte valor desde el punto de vista pedagógico.

5.1. Relación del estilo de aprendizaje del ser humano y el diseño de los Objetos de Conocimiento.

Consideramos que antes de iniciar con el proceso creativo de cualquier objeto de conocimiento que se desee empaquetar dentro de un LiveCD, es importante comprender los diferentes estilos de aprendizaje que tenemos los seres humanos, para direccionar el diseño de los objetos de conocimiento en función de estos estilos.

Los estilos de aprendizaje del ser humano son objeto de estudio permanente, lo que ha generado cambios interesantes en el proceso de enseñanza-aprendizaje desde hace tiempo. Muchos autores que abordan el tema concuerdan que los diferentes estilos de aprendizaje corresponden a características específicas de quienes aprenden.

Para propósitos de este trabajo hablaremos de los estilos de conocimiento propuestos por David Kolb, (CCA Centro Comunitario de Aprendizaje, 2012) quien señala que existen cuatro formas de adquirir conocimiento:

- por experiencia concreta,

- por observación reflexiva
- por conceptualización abstracta
- por experimentación activa

A partir de las respectivas combinaciones de estas cuatro formas de aprendizaje, se crean a su vez los cuatro estilos de aprendizaje en los que basaremos la propuesta para que los usuarios de este proyecto diseñen sus objetos de conocimiento:

- divergente
- asimilador
- convergente
- acomodador.

La ilustración 5 muestra las formas de aprendizaje y los estilos propuestos por David Kolb, que se utilizarán como base para el diseño de los objetos de aprendizaje.

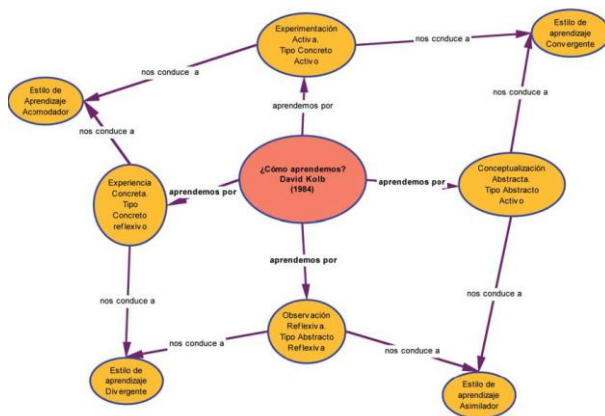


Ilustración 5: Estilos de Aprendizaje de David Kold

Las siguientes son algunas recomendaciones que consideramos importantes para el diseño de los objetos e aprendizaje. Están basadas en las definiciones de cada estilo de aprendizaje enunciado anteriormente. (Jimeno Zavala & Rodríguez Feliciano, 2012)

Estilo de Aprendizaje	Característica del Alumno	Recomendación de Diseño O A
Estilo Divergente	Describe a personas imaginativas, que prefieren aprender basándose en sentimientos y observaciones. Exige saber “POR QUÉ”.	Se recomienda que los Objetos de aprendizaje contengan actividades que combinen la experiencia concreta con la observación reflexiva.
Estilo Acomodador	Describe personas dinámicas que prefieren basarse en sentimientos y acción cuando aprenden. Exige saber si está correcto y adaptarlo a sus propias necesidades.	Se recomienda diseñar actividades donde se pueda modificar a partir de lo aprendido y hacer adaptaciones. El análisis de la importancia del uso práctico es importante.
Estilo Asimilador	Describe personas analíticas que se basan principalmente en la razón y en la observación. Desea saber “QUÉ”.	Se recomienda realizar actividades de conceptualización y abstracción con la observación reflexiva.

Estilo Convergente	Describe personas que aprenden utilizando el sentido común y se basan en la razón y las acciones. Desea saber “CÓMO”.	Se recomiendan actividades que apliquen lo aprendido, se debe mostrar la utilidad y desarrollar la habilidad de implementar lo aprendido.
--------------------	--	---

Con base en las recomendaciones pedagógicas condensadas en el cuadro anterior, de cómo orientar el contenido del material didáctico y las actividades que de este se deriven, consideramos importante que siempre se defina cual es la funcionalidad esperada.

Para ello siempre deberán resolverse las siguientes preguntas:

- ¿Quién es el público objetivo?
- ¿Cuál va a ser el enfoque pedagógico?
- ¿Qué tipo de aprendizaje se quiere promover?
- ¿Cuáles son los objetivos de aprendizaje que se va a evaluar?
- ¿Cuáles son las estrategias de aprendizaje?
- ¿Cuál es el modelo de evaluación?
- ¿Cuáles son las actividades de aprendizaje, individual?
- ¿Cuáles son las actividades de aprendizaje grupal?

5.2. Diseño del Modelo para los Objetos de Conocimiento.

El diseño de los objetos de conocimiento debe pensarse desde dos puntos de vista, uno es el pedagógico, del cual ya hemos enunciado que debe basarse en los estilos de aprendizaje de las persona; el otro punto de vista es el tecnológico.

5.2.1. Diseño pedagógico del Objeto de Conocimiento.

A nivel pedagógico, consideramos que es importante que los objetos de conocimiento cumplan con por lo menos los siguientes elementos:

Elementos teóricos: Es material escrito, que detalla los contenidos, a aprender.

Actividades de experimentación práctica: Es material multimedia que permite la interacción del alumno con los contenidos.

Actividades de trabajo grupal: Es material escrito y/o multimedia, que propone actividades grupales donde cada alumno exponga sus niveles de entendimiento y los enriquezca.

Evaluación individual de lo aprendido: Es material multimedia y/o escrito que debe trabajarse de forma individual.

La siguiente ilustración muestra la interacción de los elementos enunciados con el objeto de conocimiento.

Elementos Pedagógicos que debe tener un Objeto de Conocimiento.



5.2.2. Diseño tecnológico del Objeto de Conocimiento.

A nivel tecnológico, consideramos que es importante que los objetos de conocimiento estén enmarcados en diferentes escenarios de presentación, que deberían contener algunos de los siguientes elementos:

- Interfaz en Pantalla.
- Guión escrito.
- Guión Auditivo.
- Evento de usuario Disparador de Actividades.
- Acciones asociadas a los eventos Disparadores.

La siguiente tabla ilustra un ejemplo de los elementos descritos arriba.

Escenario	Guion Escrito	Guion Auditivo	Evento Disparador	Acciones del Objeto de Conocimiento.
Índice Principal de Contenidos	Introducción al material Didáctico.	Música de fondo.	Click en los Links del Índice Principal.	Navegación hacia el contenido seleccionado en el índice.
Navegación en el sub-contenido seleccionado.	Material escrito relativo al tema que se está tratando.	Música de fondo.	Click en un video alusivo al tema tratado.	Visualización del video en un reproductor multimedia embebido.

5.3. Construcción de un Objeto de Conocimiento

La construcción de los objetos de conocimiento que serán empaquetado utilizando el proyecto PhEANix, es un reto en si mismo dado que cada docente o usuario del software deberá construir sus propios objetos y el software será la herramienta que le permita colocarlos dentro del LiveCD, que será utilizado como elemento para su visualización y uso.

Dado que los medios físicos CD, DVD, presentan limitaciones de tamaño es importante aclarar que la suma del contenido más la imagen del sistema Linux embebido en el LiveCD tendrán un tamaño máximo de 4GB, en el caso de utilizar DVD, o 700MB en caso de utilizar un CD. Se recomienda tener estas restricciones

presentes a la hora de la creación de los LiveCD para no exceder los límites que impidan su portabilidad.

Atendiendo a estas necesidades particulares se ha planteado utilizar HTML como formato para los textos, PNG para las imágenes y MP4 para los screencast.

Se sugiere a los usuarios la utilización de herramientas libres para la construcción y edición de los contenidos (objetos de conocimiento), que se utilizaran las siguientes son algunas de las herramientas:

Nvu (N-view),

Quanta,

CamStudio,

Wink,

Camtasia

5.4. La construcción del primer Objeto

Ya hemos tratado conceptualmente el tema de los Objetos de conocimiento, ¿qué son?, ¿Cómo se pueden construir?, y ¿Qué aspectos se deben tener en cuenta a nivel pedagógico y tecnológico?.

Para quienes tienen una primera mirada al tema de construir material didáctico, mediante la utilización de herramientas tecnológicas, vamos a ejemplificar dicho proceso mediante la creación de un documento tipo HowTo.

Para tal efecto escogimos el siguiente ejemplo:

[HOWTO] Autenticación de un usuario en Linux.

Para crear documentos HOWTO, ó como lo hago, recomendamos crear al docente crear una lista de las preguntas más frecuentes, relacionadas con respuestas que implican realizar algún tipo de procedimiento.

Una lista de documentos HowTo debe contar con algunos campos necesarios para facilitar las búsquedas dentro de la lista, Estos campos son el título, la descripción la fecha de publicación, la fecha de Actualización, el autor y un campo de calificación de los usuarios. Ver Anexo 5.

Así pues como se esbozó en el contenido anterior serán los documentos que se empaqueten dentro del LiveCD en formato HTML, permitiendo la fácil navegación dentro de los mismos.

CAPITULO 6

6. CONCLUSIONES Y EVOLUCIÓN.

La incursión de los LiveCD como tecnología en la academia ha permitido, masificar el uso del software libre dentro del ámbito académico.

Sin embargo los LiveCD no se han explotado tan ampliamente para colocar conocimiento empaquetado dentro de ellos, aprovechando su carácter portable, en parte porque su estructura y mecanismos de modificación no son tan ampliamente conocidos.

Este proyecto presenta una alternativa que pone en manos de un usuario final, la posibilidad de empaquetar sus propios contenidos (Objetos de conocimiento), dentro de un LiveCD, para compartirlos con la comunidad académica y el público en general.

Con relación a las herramientas utilizadas para la masterización de los LiveCD empaquetando objetos de conocimiento, podemos concluir que la integración de Java como Front End y Scripting como Back End, es una combinación de alto desempeño, porque permite tener lo mejor de ambos mundos.

Por un lado la versatilidad y facilidad de Java para construir interfaces graficas, sencillas e intuitivas y por el otro la potencia de las herramientas modulares del software libre que se invocan mediante líneas de comandos y que permiten hacer la tarea de construcción del LiveCD.

Este proyecto no hubiera sido posible sin la integración de estas, dado que el proceso manual de creación de LiveCD reviste diversas complejidades que lo hacía poco atractivo para un usuario final sin una interfaz grafica.

Eclipse como plataforma de desarrollo como es sabido es muy versátil y potente, y se ve maximizada con la integración de los modeladores de interfaces graficas proporcionados por terceros.

Ubuntu se está posicionando como una de las plataformas Linux más amigable a nivel de sus versiones de escritorio, y está proporcionando a los usuarios la posibilidad de contar con un sistema muy estable, flexible y adaptable a las necesidades particulares de cada entorno, en especial en la academia.

El bash Shell de Linux, en conjunto con utilitarios del sistema es otra potente combinación, que permite resolver de forma rápida los retos que presentaba la construcción del LiveCD.

Los objetos de conocimiento deben ser construidos por grupos de trabajo interdisciplinario de profesores, alumnos, a fin de que los contenidos pedagógicos verdaderamente aporten valor al proceso de aprendizaje.

Una importante conclusión de este trabajo es la necesidad de disponibilidad de este tipo de herramientas para los docentes, y capacitarlos en su uso adecuado que permita su correcta adopción. Lo anterior potencializa aun más las posibilidades de impartir, compartir y construir conocimiento.

Evolución de este proyecto.

Este planteamiento es casi tan novedoso como el tema mismo de los LiveCD, pues si bien existen hoy en día distribuciones LiveCD que contienen software para aprendizaje, no existe una distribución que empaquete objetos de conocimiento como tal y cada una a su manera hace aproximaciones similares a las necesidades educativas de las comunidades o individuos, resolviendo temas muy particulares de cada una.

PhEANix LiveCD puede evolucionar para que no solamente la facultad de ingeniería de la EAN pueda empaquetar contenidos didácticos, sino para otras facultades puedan también hacerlo.

Otra evolución prevista es la posibilidad de integrar las herramientas de construcción de los objetos de conocimiento dentro de la misma interfaz gráfica, a fin de facilitar aun más el proceso para el docente.

Si bien es cierto que el proyecto inicialmente corre dentro de un entorno Linux Ubuntu, se puede evolucionar para que algunas de las funcionalidades de la plataforma puedan ser ejecutadas en entornos Windows, que en la actualidad son entornos más difundidos entre los usuarios convencionales.

Dado que a nivel mundial existen numerosas iniciativas de software para administración de objetos de conocimiento como MERLOT (California), Careo (Canadá), ARIADNE and SchoolNet (Europe), LRC (Universitas 21), es conveniente trabajar en una posible integración de estas tecnologías con la estrategia de LiveCD a fin de maximizar los beneficios que se puedan obtener de su utilización.

7. BIBLIOGRAFÍA

- canonical. (11 de 2012). *Ubuntu Project*. Obtenido de <http://www.ubuntu.com/project>
- CCA Centro Comunitario de Aprendizaje. (10 de 2012). *Modelo de David Kolb, aprendizaje basado en experiencias*. Obtenido de http://www.cca.org.mx/profesores/cursos/cep21-tec/modulo_2/modelo_kolb.htm
- Debian org. (09 de 2009). *Debian org*. Obtenido de <http://www.debian.org/index.en.html>
- e-ABC soluciones de e-learning. (09 de 2011). *e-ABC*. Obtenido de <http://www.e-abclearning.com/definicion-e-learning>
- Eclipse Foundation. (13 de 09 de 2012). *Eclipse Newcomers FAQ*. Obtenido de <http://www.eclipse.org/home/newcomers.php>
- El rincón de Linux para Hispano Parlantes*. (1998). Recuperado el 10 de 09 de 2012, de <http://www.linux-es.org/livecd>
- El rincón de Linux para hispanoparlantes. (09 de 2012). *Que es el kernel/núcleo?*. Obtenido de <http://www.linux-es.org/kernel>
- Fedora Project. (10 de 2012). *Proyecto Fedora*. Obtenido de <http://fedoraproject.org/es/>
- Free Software Foundation, Inc. . (09 de 2012). *The Free Software Definition*. Obtenido de <http://www.gnu.org/philosophy/free-sw.en.html>
- IMS Global Learning Consortium, Inc. . (11 de 2012). *IMS Interoperability Standards*. Obtenido de <http://www.imsglobal.org/>
- Jimeno Zavala, M. A., & Rodríguez Feliciano, M. Á. (11 de 2012). *APRENDIZAJE BASADO EN EXPERIENCIAS, EL MODELO DE DAVID KOLB*. Obtenido de La aplicación a un grupo de tutorados en la Facultad de Ciencias Químicas de la UnACh.: <http://www.sistemas.unach.mx/tutoria/ENCUENTROS/UNACH%202/Ponent es/17-Aprendizaje.pdf>
- Knopper.net. (09 de 2009). *CLOOP*. Obtenido de <http://knoppix.net/wiki/Cloop>
- Knopper.net. (09 de 2009). *What is KNOPPIX®?* Obtenido de <http://www.knopper.net/knoppix/index-en.html>
- Lutus, P. (15 de 09 de 2009). *Bash Shell Programming in Linux*. Obtenido de http://arachnoid.com/linux/shell_programming.html
- moodle.org. (06 de 2009). *SCORM*. Obtenido de <http://docs.moodle.org/all/es/SCORM>
- morphix.org. (09 de 2007). *Morphix Distro*. Obtenido de <http://www.morphix.org/>
- Oficina de Software Libre Universidad de Cadiz. (10 de 2009). *Proyecto SciLix*. Obtenido de <http://osl.uca.es/node/155>

RedHat. (10 de 2012). *Red Hat Enterprise Linux Desktop*. Obtenido de <http://www.redhat.com/products/enterprise-linux/desktop/>

sourceforge.net. (03 de 2011). *squashfs*. Obtenido de <http://squashfs.sourceforge.net/>

The Dublin Core Metadata Initiative (DCMI). (11 de 2012). *DCMI Specifications*. Obtenido de <http://dublincore.org/>

wikipedia. (06 de 2007). *Metadistros (herramienta GNU/Linux)*. Obtenido de [http://es.wikipedia.org/wiki/Metadistros_\(herramienta_GNU/Linux\)](http://es.wikipedia.org/wiki/Metadistros_(herramienta_GNU/Linux))

wikipedia. (10 de 2010). *ISO File*. Obtenido de http://en.wikipedia.org/wiki/ISO_image

wikipedia. (09 de 2011). *Booting Process*. Obtenido de http://en.wikipedia.org/wiki/Boot_process

wikipedia. (09 de 2012). *Monolithic kernel*. Obtenido de http://en.wikipedia.org/wiki/Monolithic_kernel

Anexo 1. Navegación y uso de la Interfaz gráfica del proyecto PhEANix.

Lo primero que observamos es la pestaña de Material Didáctico, en ella podemos colocar el nombre del curso como queremos que se llame.

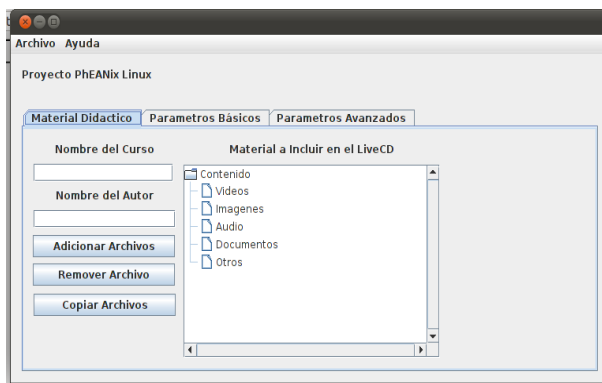


Ilustración 1: Ventana Principal - Panel Material Didactico

Como se observa en la ilustración 6: desde aquí se seleccionan los documentos, imágenes, etc., que se van a incluir en el material didáctico, mediante el botón Adicionar Archivos,

el cual abre un cuadro de dialogo para la sección de los archivos.

En las ilustraciones 7 y 8, se puede apreciar el cuadro de dialogo y los respectivos filtros para la selección de los archivos según su tipo.

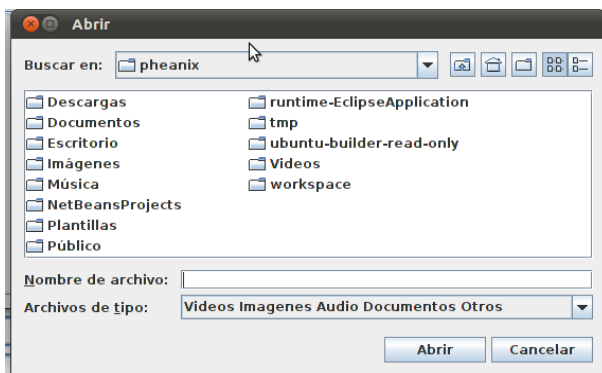


Ilustración 2: Cuadro de Dialogo para Selección de Documentos

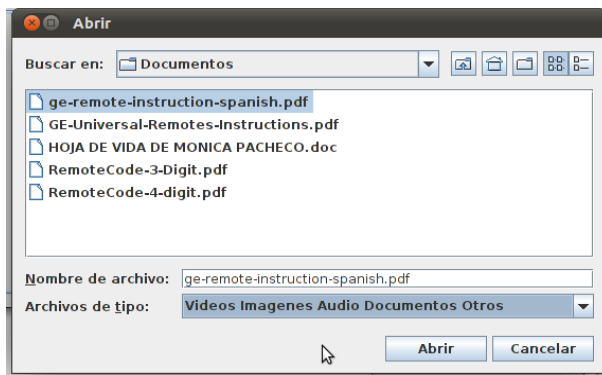


Ilustración 3: Selección de un Documento tipo PDF

Tal como se observa en la ilustración 9, una vez seleccionados los documentos que se van a adicionar al LiveCD, estos se pueden visualizar dentro del Árbol del material a incluir, uno de los objetivos de esto es permitirle al usuario decidir si elimina o no alguno de los documentos antes de incluirlos en el LiveCD.

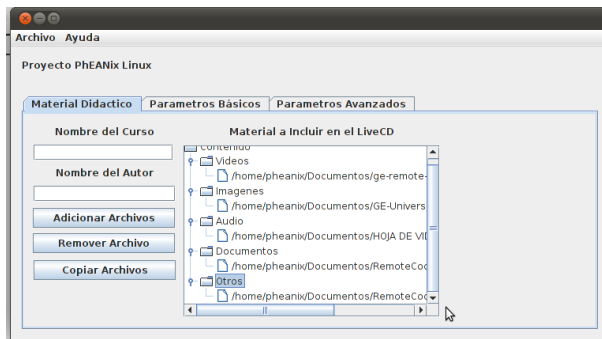


Ilustración 4: Árbol de Documentos Seleccionados

Una vez que el usuario ha establecido que todos los documentos que se encuentran incluidos dentro del árbol de selección son los que desea tener dentro del material didáctico que va a empaquetar, debe proceder con el proceso de copia a la ubicación temporal. Esto se realiza pulsando el botón Copiar Archivos ubicado a la parte inferior izquierda de la pantalla, como se observa en la ilustración 10.

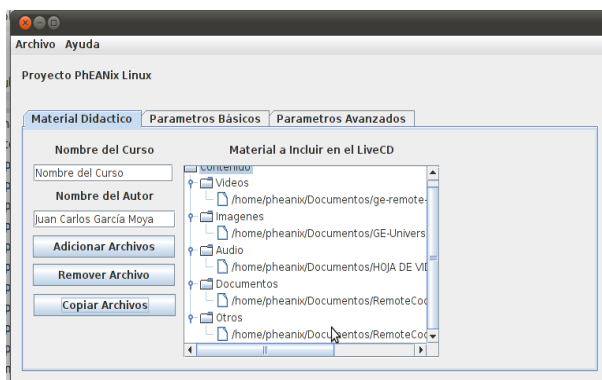


Ilustración 5: Copia de Archivos al Directorio de trabajo

Al terminar la copia del material didáctico al directorio de trabajo, las actividades del usuario en la pestaña Material Didáctico, esta completas y se puede continuar a la siguiente

pestaña, para la personalización del LiveCD base.

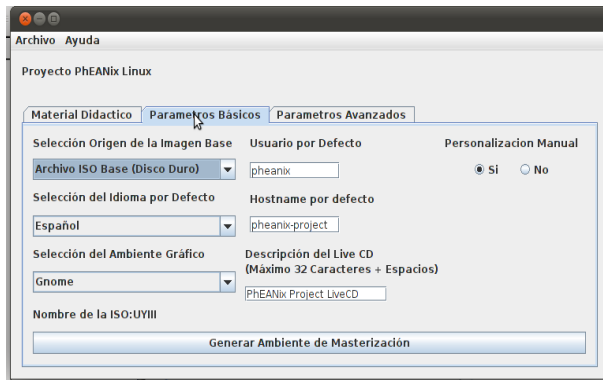


Ilustración 6: Pestaña de Parámetros Básicos

En esta pestaña se captura la información necesaria para personalizar el LiveCD a partir de una imagen ISO base.

La información que se captura es: La ruta y nombre del archivo ISO a partir del que se va a personalizar, el idioma por defecto que va a tener el LiveCD, el ambiente grafico, el nombre de usuario por defecto, el hostname del nuevo sistema, la etiqueta del nuevo ISO, la posibilidad de realizar personalizaciones manualmente, o no.

Las siguientes pantallas ilustran los pasos de personalización básica del LiveCD.

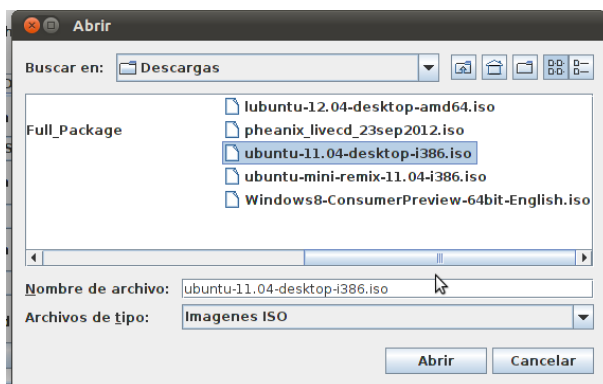


Ilustración 7: Cuadro de dialogo selección de ISO

Esto es básicamente la selección de la opción Archivo ISO desde el Control desplegable de la parte superior izquierda.

Esto carga un cuadro de dialogo de selección de archivos desde el que se selecciona el archivos ISO que se va a tomar como base para el proceso (importante que el archivos ISO corresponda a una distribución de Linux 10.10 o superior, preferiblemente de la misma versión del Sistema donde se realiza el proceso).

Otros parámetros que se seleccionan de la misma forma son el idioma por defecto y el tipo de ambiente grafico.

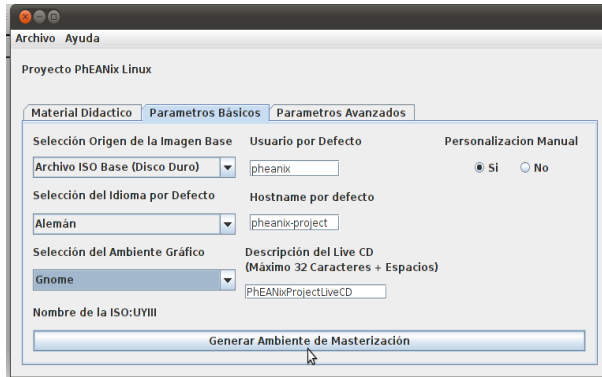


Ilustración 8: Selección demás parámetros de Personalización LiveCD e inicio del Proceso

Como se observa en la ilustración anterior, una vez que se han establecido los parámetros de personalización se presiona el botón

Generar Ambiente de Masterización.

Este botón se encarga de realizar la validación de los parámetros y de llamar al script principal pasándole todos y cada uno de los parámetros capturados en la interfaz gráfica.

Las siguientes pantallas tienen por objeto informarles a los usuarios algunos aspectos generales del proceso, como el requerimiento mínimo de espacio en disco durante el proceso y la ruta de trabajo donde se estará consumiendo dicho espacio, que es de al menos 10GB.

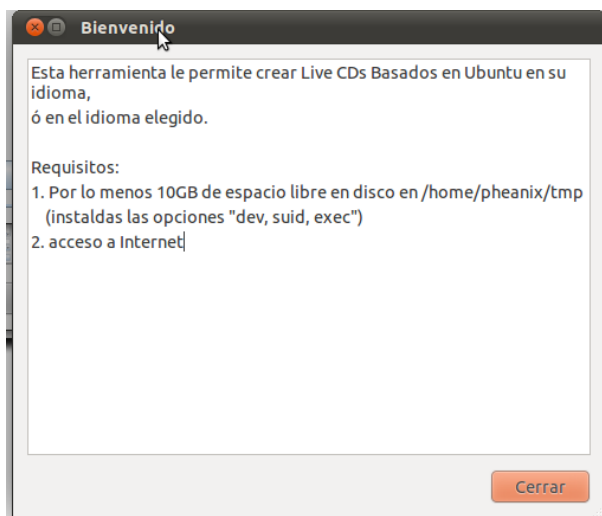


Ilustración 9: Pantalla de Confirmación de Requerimientos

La siguiente pantalla le brinda al usuario información indicándole que el proceso está a punto de iniciar, el nombre del archivo ISO de salida, que contendrá la nueva imagen ISO con las personalizaciones.

Adicionalmente le advierte que para el desarrollo de las tareas de creación del nuevo LiveCD se requieren las credenciales del usuario para autorizar el inicio de las mismas.

Importante este proceso se ejecuta en el sistema operativo con un usuario diferente a root y la escalación de privilegios administrativos hace uso del comando sudo.

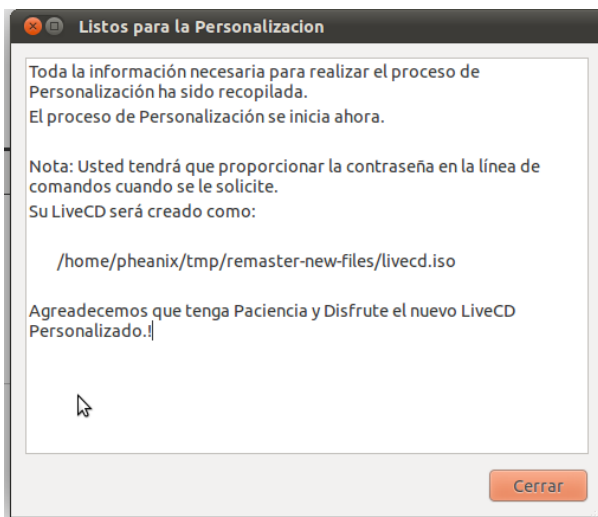


Ilustración 10: Pantalla de Confirmación de parámetros

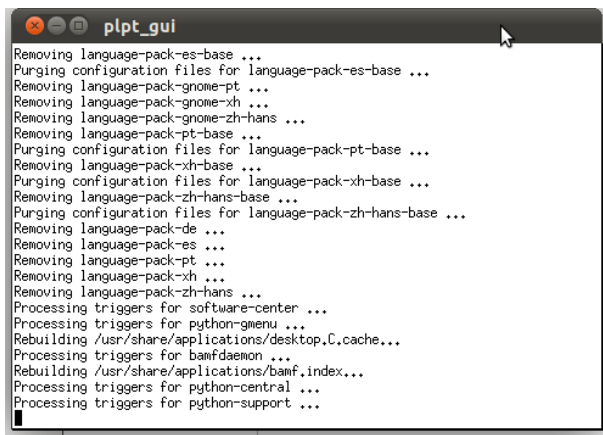
Una vez leídos y aceptados estos términos se abre una ventana de xterminal desde donde correrá todo el proceso del scripting.

Lo primero que se hace es la captura de las credenciales de usuario, para obtener la escalación de los privilegios administrativos tal como se observa en la ilustración 16.



```
plpt_gui
+ echo PHEANIXProjectLiveCD
PHEANIXProjectLiveCD
+ echo yes
yes
+ echo no
no
+ echo yes
yes
+ echo CARPETA CONTENIDO /home/pheanix/workspace/pheanix/contenido
CARPETA CONTENIDO /home/pheanix/workspace/pheanix/contenido
+ echo SCRIPTS_DIR /usr/bin/plpt_remaster
SCRIPTS_DIR /usr/bin/plpt_remaster
+ echo USE_MOUNT
USE_MOUNT
+ echo ISO_IMAGE /home/pheanix/Descargas/ubuntu-11.04-desktop-i386.iso
ISO_IMAGE /home/pheanix/Descargas/ubuntu-11.04-desktop-i386.iso
+ echo BUILD_DIR /home/pheanix/tmp/customization-scripts
BUILD_DIR /home/pheanix/tmp/customization-scripts
+ echo REMASTER_HOME /home/pheanix/tmp
REMASTER_HOME /home/pheanix/tmp
+ sudo bash /usr/bin/plpt_remaster /home/pheanix/Descargas/ubuntu-11.04-desktop-i386.iso /home/pheanix/tmp/customization-scripts /home/pheanix/tmp /home/pheanix/workspace/pheanix/contenido /home/pheanix/workspace/pheanix/Desktop
[sudo] password for pheanix: █
```

Ilustración 11: Ventana Principal de scripting



```
plpt_gui
Removing language-pack-es-base ...
Purging configuration files for language-pack-es-base ...
Removing language-pack-gnome-pt ...
Removing language-pack-gnome-xh ...
Removing language-pack-gnome-zh-hans ...
Removing language-pack-pt-base ...
Purging configuration files for language-pack-pt-base ...
Removing language-pack-xh-base ...
Purging configuration files for language-pack-xh-base ...
Removing language-pack-zh-hans-base ...
Purging configuration files for language-pack-zh-hans-base ...
Removing language-pack-de ...
Removing language-pack-es ...
Removing language-pack-pt ...
Removing language-pack-xh ...
Removing language-pack-zh-hans ...
Processing triggers for software-center ...
Processing triggers for python-gmenu ...
Rebuilding /usr/share/applications/desktop.C.cache...
Processing triggers for bamfdaemon ...
Rebuilding /usr/share/applications/bamf.index...
Processing triggers for python-central ...
Processing triggers for python-support ...
```

Ilustración 12: Primeros pasos del proceso

Tal y como se puede apreciar en la ilustración 17, el proceso va dejando un log de salida tanto en pantalla como a un archivo en disco que generalmente está en la ruta `/home/usuario/tmp/plpt_pheanix.log` en caso de que se desee verificar posteriormente, o en caso de falla, para poder establecer la causa.

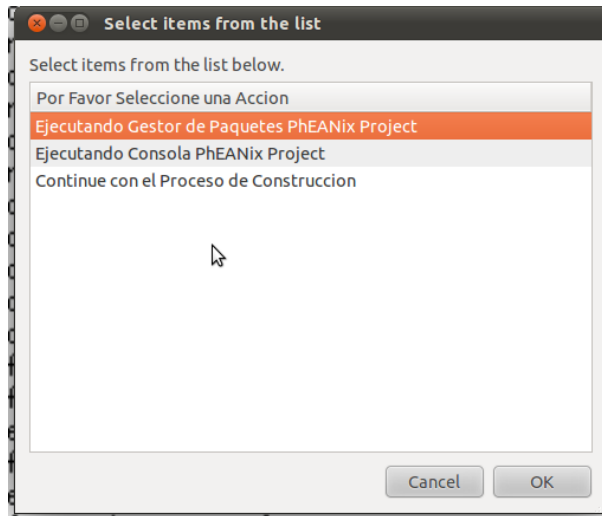


Ilustración 13: Selección de Herramientas para Personalización Avanzada

Tal y como se menciona anteriormente es posible realizar procesos de personalización del LiveCD de forma Avanzada, mediante la utilización del gestor de paquetes synaptic y/o mediante la utilización de comandos carácter en un ambiente

de consola chroot.

En las siguientes pantallas se ilustra el proceso de gestión de paquetes con Synaptic, acorde con la selección realizada.

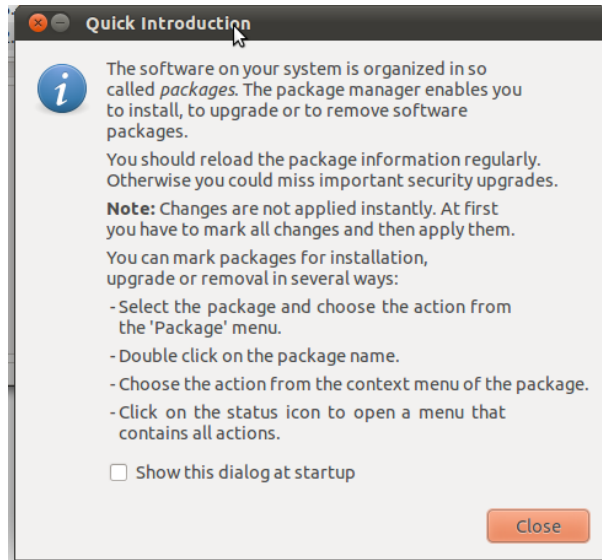


Ilustración 14: Pantalla de Bienvenida del Paquete synaptic

Synaptic, es un programa que permite realizar la gestión de software instalado en un sistema Ubuntu, mediante una interfaz de usuario grafica, muy fácil de manejar.

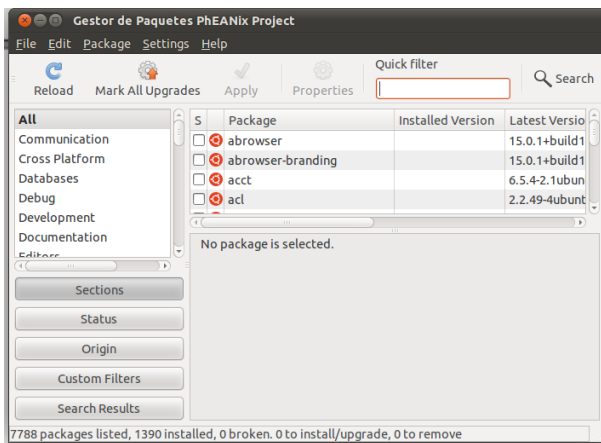


Ilustración 15: Ventana principal de synaptic.

Cabe anotar que el programa synaptic, se está ejecutando contenido dentro de una jaula chroot, para que los cambios solo aplique al ambiente del LiveCD que se está preparando.

Cuando se terminen de realizar las modificaciones desde synaptic, este se cierra y le retorna el control a los scripts de personalización, por lo cual, se regresa a la ventana de selección.

Para efectos de este documento se realizó luego la selección de la opción de la consola root como se muestra en la ilustración 21.

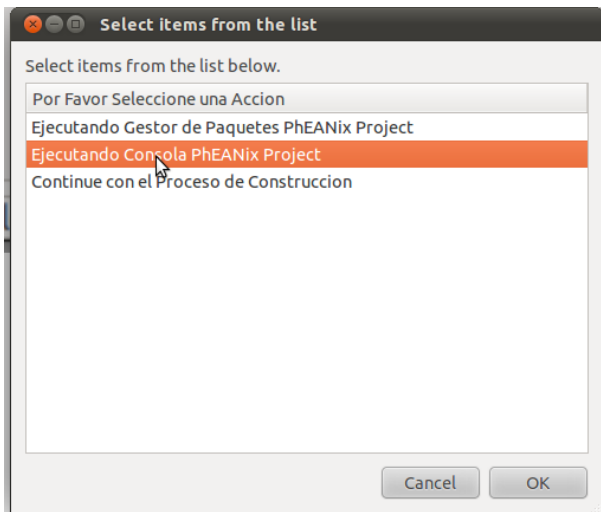


Ilustración 16: Selección de la consola chroot

Trabajar desde una consola carácter con usuario root, supone un nivel avanzado de conocimientos en el sistema operativo Linux y por demás en las herramientas de personalización de LiveCD.

Por lo anterior, solo vamos a mostrar que en el proceso se puede contar con esta opción, sin entrar en detalles.

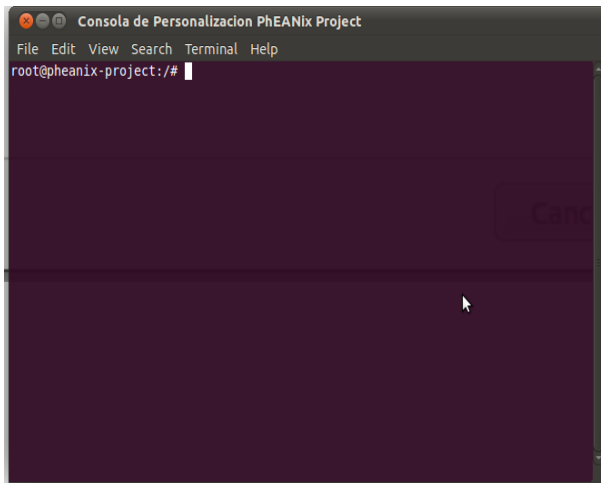


Ilustración 17: Consola de personalización CHROOT

Suponiendo que el usuario que realiza la personalización ha terminado de realizar las actividades en esta consola, se proceda a cerrarla y nuevamente el control retorna a los scripts, por lo que se le debe indicar en este punto que la opción deseada

es Continuar con el proceso de construcción del LiveCD.

Al hacer esto, los scripts le dan paso a la re compresión del sistema de archivos en formato squashfs, a la inclusión de los documentos (Objetos de Conocimiento) y a la reconstrucción del nuevo archivo ISO, como se ilustra en las ilustraciones 23, 24 y 25.

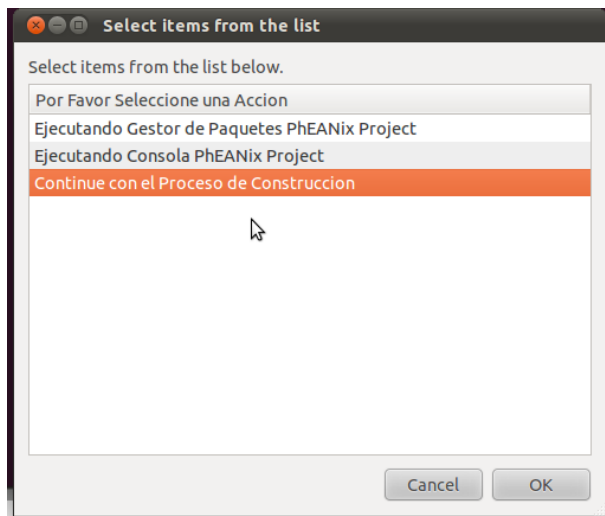


Ilustración 18: Se continúa con el proceso.

```

synaptics xserver-xorg-input-vmouse xserver-xorg-input-wacom xserver-xorg-video
-all xserver-xorg-video-apm xserver-xorg-video-ark xserver-xorg-video-ati xserve
r-xorg-video-chips xserver-xorg-video-cirrus xserver-xorg-video-fbdev xserver-xo
rg-video-geode xserver-xorg-video-i128 xserver-xorg-video-i740 xserver-xorg-vid
o-intel xserver-xorg-video-mach64 xserver-xorg-video-mga xserver-xorg-video-neo
agic xserver-xorg-video-nouveau xserver-xorg-video-openchrome xserver-xorg-vid
eo-qxl xserver-xorg-video-r128 xserver-xorg-video-radeon xserver-xorg-video-rendit
ion xserver-xorg-video-s3 xserver-xorg-video-s3virge xserver-xorg-video-savage x
server-xorg-video-siliconmotion xserver-xorg-video-sis xserver-xorg-video-sisub
xserver-xorg-video-tdfx xserver-xorg-video-trident xserver-xorg-video-tseng xse
rver-xorg-video-vesa xserver-xorg-video-vmware xserver-xorg-video-vooodoo xsltpro
c xterm xul-ext-ubufax xz-utils yelp yelp-xsl zeitgeist zeitgeist-core zeitgeist
-datahub zeitgeist-extension-fts zenity zip zlibg
+ rm /home/phenix/tmp/remaster-iso/filesystem,manifest,tmp /home/phenix/tmp/re
master-iso/filesystem,manifest-desktop,tmp
+ echo 'Empaquetando Imagen SquashFS...'
Empaquetando Imagen SquashFS...
+ '[' -e /home/phenix/tmp/remaster-iso/casper/filesystem,squashfs ']'
+ rm -f /home/phenix/tmp/remaster-iso/casper/filesystem,squashfs
+ EXTRA_OPTS=
+ '[' -e /rootfs.sort ']'
+ mk squashfs /home/phenix/tmp/remaster-root /home/phenix/tmp/remaster-iso/casp
er/filesystem,squashfs

```

Ilustración 19: re compresión del filesystem principal en formato squashfs

```

rg-video-geode xserver-xorg-video-i128 xserver-xorg-video-i740 xserver-xorg-vid
o-intel xserver-xorg-video-mach64 xserver-xorg-video-mga xserver-xorg-video-neo
agic xserver-xorg-video-nouveau xserver-xorg-video-openchrome xserver-xorg-vid
eo-qxl xserver-xorg-video-r128 xserver-xorg-video-radeon xserver-xorg-video-rendit
ion xserver-xorg-video-s3 xserver-xorg-video-s3virge xserver-xorg-video-savage x
server-xorg-video-siliconmotion xserver-xorg-video-sis xserver-xorg-video-sisub
xserver-xorg-video-tdfx xserver-xorg-video-trident xserver-xorg-video-tseng xse
rver-xorg-video-vesa xserver-xorg-video-vmware xserver-xorg-video-vooodoo xsltpro
c xterm xul-ext-ubufax xz-utils yelp yelp-xsl zeitgeist zeitgeist-core zeitgeist
-datahub zeitgeist-extension-fts zenity zip zlibg
+ rm /home/phenix/tmp/remaster-iso/filesystem,manifest,tmp /home/phenix/tmp/re
master-iso/filesystem,manifest-desktop,tmp
+ echo 'Empaquetando Imagen SquashFS...'
Empaquetando Imagen SquashFS...
+ '[' -e /home/phenix/tmp/remaster-iso/casper/filesystem,squashfs ']'
+ rm -f /home/phenix/tmp/remaster-iso/casper/filesystem,squashfs
+ EXTRA_OPTS=
+ '[' -e /rootfs.sort ']'
+ mk squashfs /home/phenix/tmp/remaster-root /home/phenix/tmp/remaster-iso/casp
er/filesystem,squashfs
Parallel mk squashfs: Using 2 processors
Creating 4,0 filesystem on /home/phenix/tmp/remaster-iso/casper/filesystem.squa
shfs, block size 131072.
[=====] 42300/94365 44%

```

Ilustración 20: avance de la re-compresión.

```

boot-info-table -V PhEaRnixProjectLiveCD -cache-nodes -r -J -l -x /home/phenix/
tmp/remaster-iso/casper/manifest.diff -joliect-long /home/phenix/tmp/remaster-is
o
Size of boot image is 4 sectors -> No emulation
1,50% done, estimate finish Thu Oct 4 00:26:11 2012
2,99% done, estimate finish Thu Oct 4 00:26:11 2012
4,48% done, estimate finish Thu Oct 4 00:26:11 2012
5,97% done, estimate finish Thu Oct 4 00:26:11 2012
7,47% done, estimate finish Thu Oct 4 00:26:11 2012
8,96% done, estimate finish Thu Oct 4 00:26:11 2012
10,46% done, estimate finish Thu Oct 4 00:26:11 2012
11,95% done, estimate finish Thu Oct 4 00:26:11 2012
13,44% done, estimate finish Thu Oct 4 00:26:11 2012
14,93% done, estimate finish Thu Oct 4 00:26:11 2012
16,43% done, estimate finish Thu Oct 4 00:26:11 2012
17,92% done, estimate finish Thu Oct 4 00:26:11 2012
19,42% done, estimate finish Thu Oct 4 00:26:11 2012
20,91% done, estimate finish Thu Oct 4 00:26:11 2012
22,40% done, estimate finish Thu Oct 4 00:26:11 2012
23,89% done, estimate finish Thu Oct 4 00:26:11 2012
25,39% done, estimate finish Thu Oct 4 00:26:14 2012
26,88% done, estimate finish Thu Oct 4 00:26:14 2012
28,37% done, estimate finish Thu Oct 4 00:26:21 2012

```

Ilustración 21: Creación del nuevo ISO

En las ilustraciones 26 y 27 se observan fases del proceso final de creación del nuevo LiveCD, consistentes en la generación de la nueva imagen ISO.

Una vez finalizado este proceso, en la ruta de trabajo estará disponible el archivo ISO, para realizar pruebas, antes de proceder a quemarlo en medio físico.

```
plpt_gui
/home/pheanix/tmp/remaster-new-files/livecd.iso

Si tiene instalado qemu, podra probar si nuevo ISO haciendo:

Para 32Bits (qemu -cdrom /home/pheanix/tmp/remaster-new-files/livecd.iso -boot d -m 256)
Para 64Bits (qemu-system-x86_64 -cdrom /home/pheanix/tmp/remaster-new-files/livecd.iso -boot d -m 1024)'
+ shift
+ case $_dialog_type in
+ zenity --title 'La personalizacion Termino Exitosamente' --text-info --width=500 --height=400
+ echo -n 'La Personalización Termino Exitosamente!!!
Usted puede encontrar la ISO del nuevo Live CD en:

/home/pheanix/tmp/remaster-new-files/livecd.iso

Si tiene instalado qemu, podra probar si nuevo ISO haciendo:

Para 32Bits (qemu -cdrom /home/pheanix/tmp/remaster-new-files/livecd.iso -boot d -m 256)
Para 64Bits (qemu-system-x86_64 -cdrom /home/pheanix/tmp/remaster-new-files/livecd.iso -boot d -m 1024)'
```

Ilustración 22: Terminación del Proceso de Creación del LiveCD.

La siguiente ventana muestra la salida que le mostrará al usuario el proceso en caso de terminación exitosa.

En dicha ventana se le indicaran los pasos a seguir para poder probar el nuevo ISO, en un ambiente virtualizado.

Es importante anotar que el proceso de prueba del LiveCD es opcional. Las tecnologías de Virtualización son muy diversas, para efectos de este trabajo se está utilizando qemu.

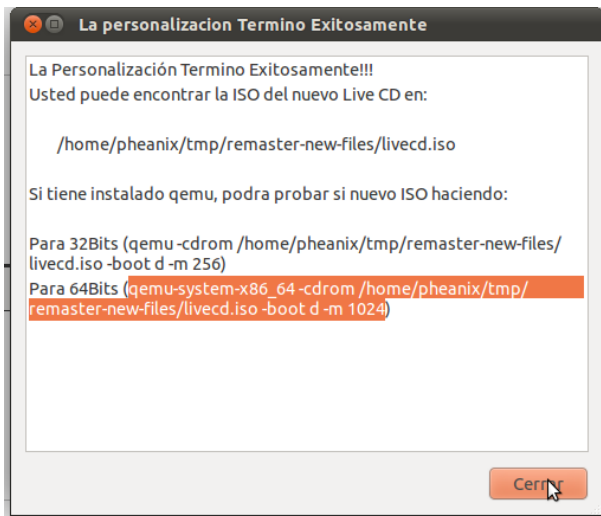


Ilustración 23: Ventana de terminación Exitosa.

Procedemos a iniciar el ambiente de virtualización con la línea de comandos resaltada en la ventana anterior, dependiendo del tipo de plataforma sobre la que este corriendo el proceso.

En las ilustraciones 29 y 30 se aprecia el proceso de arranque del LiveCD con el virtualizador qemu.

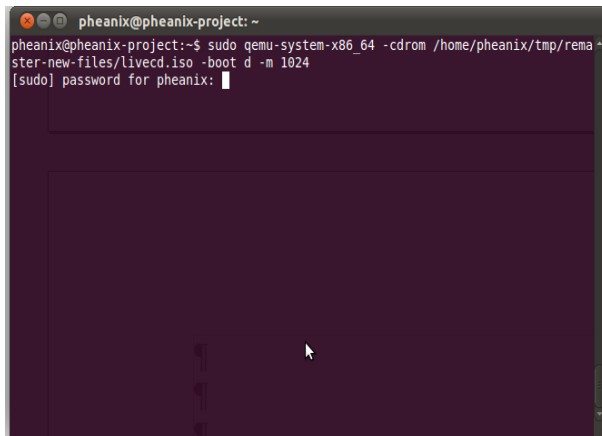


Ilustración 24: Inicio de una sesión de qemu, con el nuevo liveCD.

El proceso se debe lanzar con privilegios administrativos, por lo que se utiliza el comando sudo.

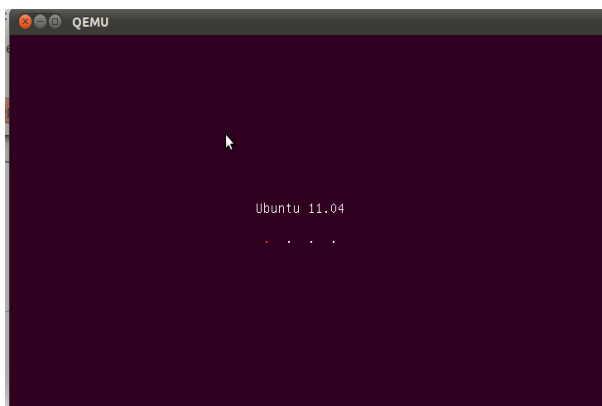


Ilustración 25: Pantalla de inicio del LiveCD.

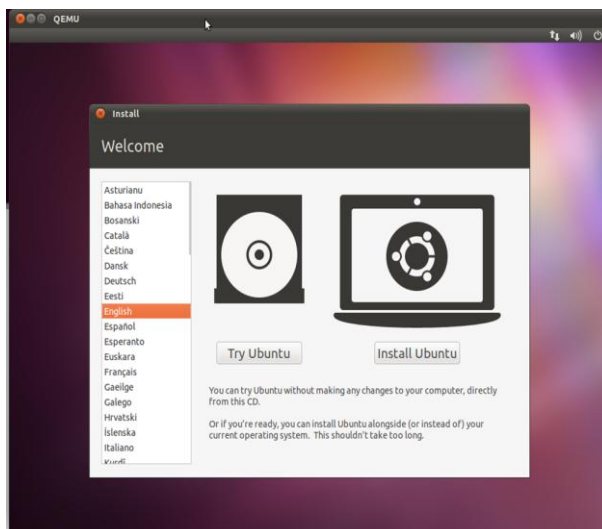


Ilustración 26: Menú de Inicio del LiveCD.

El menú de inicio del Live CD generalmente evalúa la posibilidad de realizar un proceso de inicio sin instalar, ó un proceso de instalación.

El usuario debe cerciorarse de cual

opción es la que va a seleccionar. Se recomienda que siempre se use la opción de iniciar sin instalar.

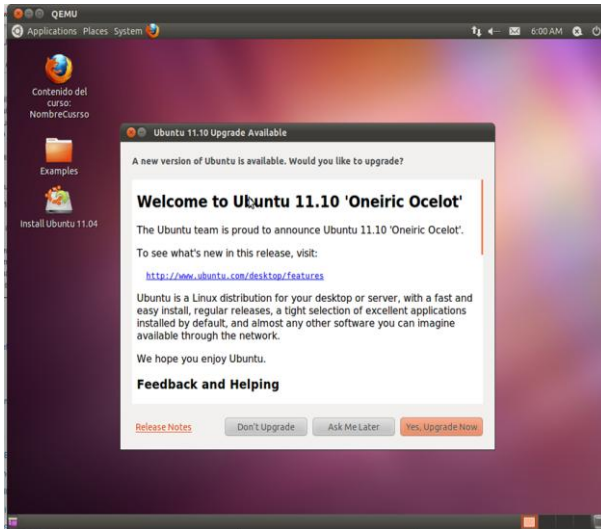


Ilustración 27: Escritorio del nuevo liveCD

Obsérvese que en el escritorio se encuentra un Link que dice Contenido del Curso “Contenido”

Este link es la ruta de acceso desde el navegador al material empaquetado dentro del nuevo LiveCD.

Cuando este link se abre se carga el navegador Firefox con el archivo de índice del materia empaquetado, tal y como se muestra en las siguientes pantallas.

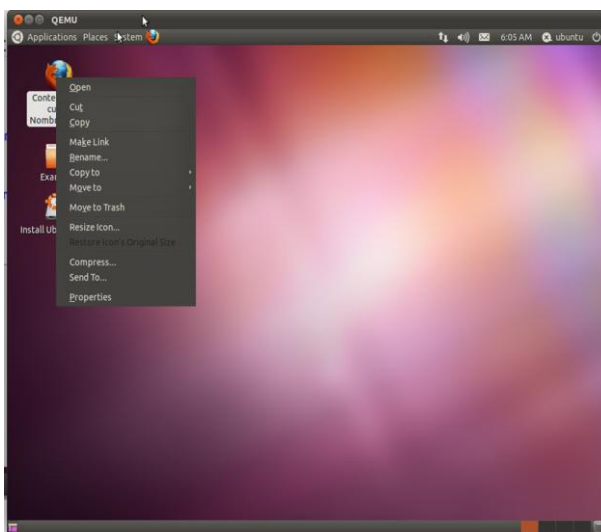


Ilustración 28: Ejecución del Link al Contenido Empaquetado.

Este acceso directo al contenido empaquetado dentro del LiveCD, está asociado al navegador por defecto FireFox, dado que se diseño para ser un índice del contenido.

El objetivo de manejar el acceso al contenido mediante formato HTML, es básicamente proveer facilidad en la navegación del estudiante por el material.

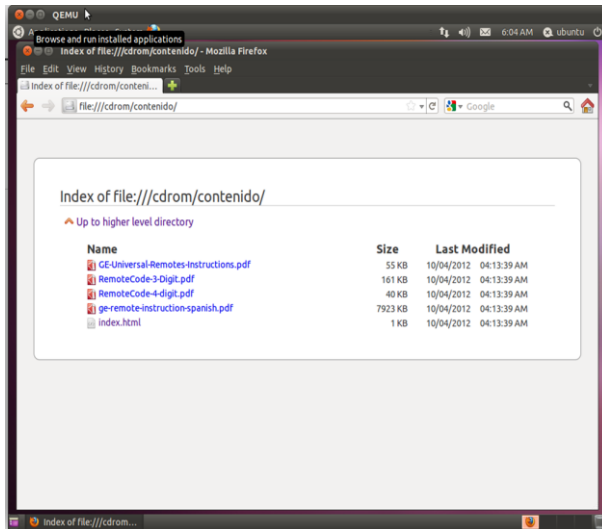


Ilustración 29: Visualización del Índice del Contenido

En la ilustración 34 se puede apreciar cómo se visualiza el material empaquetado dentro del LiveCD.

Por último en las ilustraciones 35 y 36 se observa el proceso de apagado del Ambiente virtual donde se cargó y probó el nuevo LiveCD.

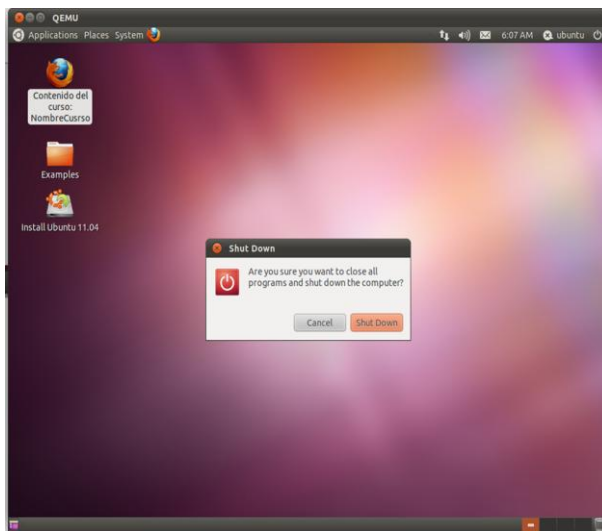


Ilustración 30: Apagado del LiveCD en qemu

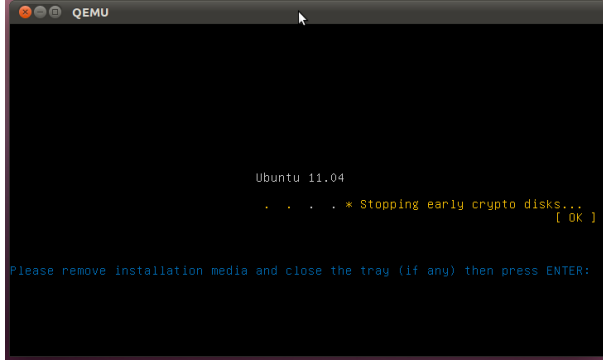


Ilustración 31: Pantalla del LiveCD al apagar la sesión qemu.

Anexo 2. El Proceso de Adquisición e Instalación de la Distro Base.

Con el panorama un poco más claro sobre la historia, las bondades y restricciones de las diferentes distros con las que se ha experimentado antes de decidir el uso de Ubuntu como la Distro Base para desarrollar el proyecto PhEANix; es momento abordar el tema de la adquisición e instalación.

Esta distro se puede solicitar registrándose en el sitio del proyecto <http://www.ubuntu.com/> , de forma que Canonical Ltda. Te envía a vuelta de correo un juego de medios (CD), live y de instaladores, para x86 y para Amd64.

El proyecto LiveCD PhEANix se basa en la última versión de Ubuntu disponible, sin embargo el proceso de adquisición de los medios, se realizó con la versión 7.04.

Como con otras distros, el proceso inicia con la descarga del archivo ISO del sitio oficial del proyecto. Hasta aquí no existe nada innovador en este proceso, ya que hoy por hoy con la masificación del acceso a Internet y el incremento en las velocidades de navegación, esto es lo común; sin embargo lo interesante del asunto radica en la utilización de software libre sobre ambiente Windows para realizar el proceso. El asunto es tratar de incentivar a los usuarios del mundo Microsoft a utilizar herramientas libres, como Free Download Manager, md5sum, sha1sum e InfraRecorder, que describiremos posteriormente.

Estamos listos, recuerden que el objetivo, es tener una distro de Ubuntu 7.04 instalada en nuestro computador, para luego desarrollar el trabajo de masterización o personalización hasta obtener una nueva distro, PhEANix.

En todo proceso de instalación de un nuevo sistema operativo es necesario planear bien la instalación, para lo cual es importante tener claro la configuración del hardware del computador, especialmente la configuración de RAM y la distribución de los discos duros.

Para efectos de este documento veremos una instalación típica que contempla tener un PC previamente con dos discos duros y una unidad de DVD, instalado con un sistema Windows, para dual boot.

1 procesador Intel dual core
4 GB de RAM.
2 Discos Duros IDE de 40Gb cada uno
1 unidad de CD/RW Lector DVD.

En toda distribución Linux que se va a ejecutar desde disco duro es conveniente realizar la distribución del espacio en disco para el sistema, los programas, los usuarios, y los procesos de paginación esencialmente; por esto para este caso la distribución de los discos quedo de la siguiente forma:

IDE 0:0 1 Partición NTFS de 40 GB para Windows.
IDE 1:1 1 Partición Primaria EXT3 de 39 GB para / (Ubuntu 7.04)
IDE 1:2 1 Partición Primaria SWAP de 1 GB para paginación (Ubuntu 7.04)
IDE 2:1 1 Partición Primaria EXT3 de 39 GB para /usr (Ubuntu 7.04)
IDE 2:2 1 Partición Primaria EXT3 de 1 GB para paginación (Ubuntu 7.04)

Con esto estará listo, para realizar la instalación de Ubuntu, los procesos de particionamiento de discos en Linux podrán ser consultados en fuentes externas y no serán parte de los temas en este documento. Recomendamos a los novatos en Linux tener un disco duro diferente.

La siguiente es una guía grafica con pantallas capturadas de una instalación genérica de Ubuntu que se han recopilado. Esta guía es una visión general de cómo funciona el proceso de instalación de Ubuntu 7.04 (Feisty Fawn), en esta versión el proceso de instalación se ha simplificado aun más que en las versiones anteriores y a diferencia de versión 6.06 (Dapper Drake), incorpora un paso adicional para la migración de las configuraciones Windows si así se desea.



Ilustración 32: Pantalla Principal de Ubuntu 7.04 inmediatamente después de iniciarse el LiveCD ó de su instalación.

Los siguientes son los siete pasos básicos para la instalación de Ubuntu 7.04:

- Elección del idioma de instalación
- Configuración del Conjunto Ubicación
- Elección del modelo de teclado
- Configuración del Disco duro y particiones
- Configuración de la Información sobre la Migración de Windows (Opcional)
- Configuración de las cuentas de acceso
- Ejecutar la Instalación, Haciendo clic en el botón "Instalar"

A diferencia de la instalación de Ubuntu 6.06, Feisty Fwan, no permite establecer el tiempo durante el proceso de instalación, en cambio, se debe fijar la ubicación durante la instalación y el tiempo después.

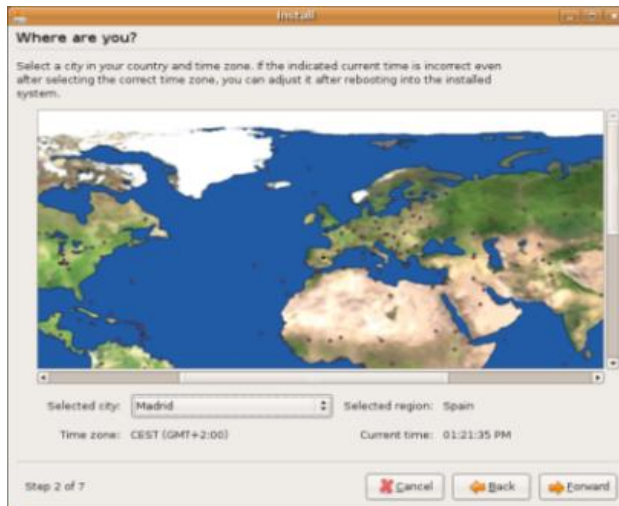


Ilustración 33: Pantalla de la selección de Zona Horaria y Fijación de Hora en Ubuntu 6.06

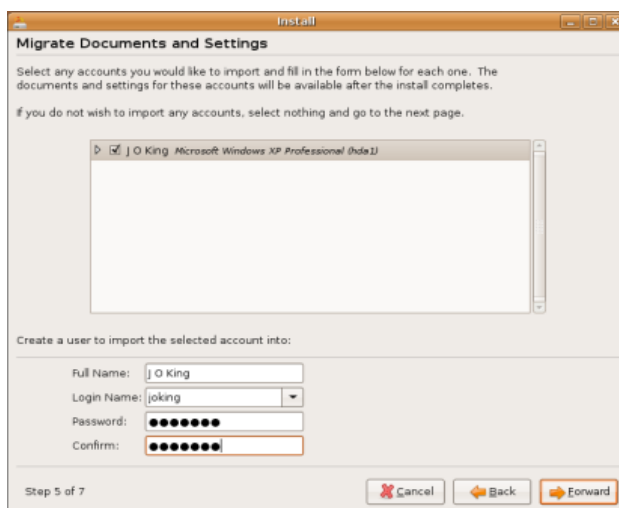


Ilustración 34: Asistente de Migración de Configuraciones Windows.

Ciertamente, la mayor novedad en la instalación de Ubuntu 7.04 es el asistente de migración. El asistente de instalación de Feisty pedirá su autorización, para importar la configuración de su sistema Windows a su nueva instalación de

Ubuntu, en cuyo caso si es afirmativa la selección, serán importados todos las configuraciones de su escritorio, navegador Web, favoritos, y sus carpetas personales (por ejemplo, Mis documentos, etc.). Este proceso se repetirá para todas las cuentas que se autoricen. Para cada cuenta, usted simplemente deberá introducir la información de inicio de sesión para ese usuario.

En el último paso de la instalación es una ventana similar a Ubuntu 6.06.

Solo es necesario hacer clic en "Instalar" y dejar que Ubuntu realice el trabajo sucio.

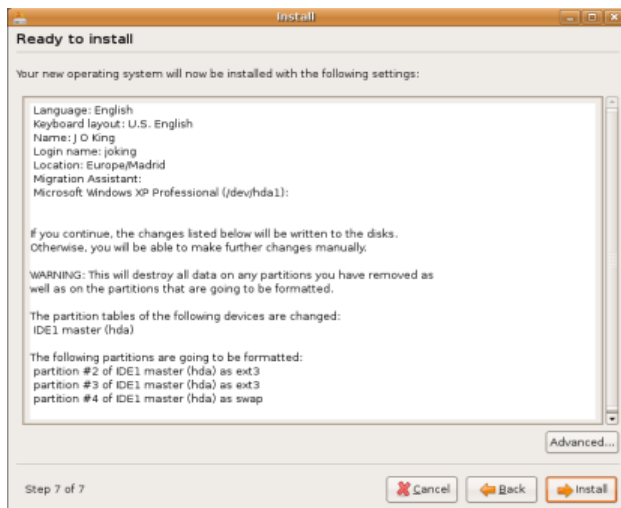


Ilustración 35: Pantalla de Confirmación de configuración previa instalación.

Anexo 3. El Proceso de Masterización.

Hasta ahora tenemos instalado Ubuntu básico en nuestro equipo, ya sea en un disco independiente (opción que recomiendo para los novatos) ó en un mismo disco en una partición adicional compartiendo espacio con disco.

El proceso de masterización consiste básicamente en personalizar la instalación de Ubuntu, instalando o desinstalando aplicativos, entornos gráficos, cambiando la presentación de estos entornos, las pantallas de inicio y cierre del sistema, la creación de un usuario para el inicio por defecto de la sesión de trabajo, etc.

A continuación describiré los principales pasos que seguí para llevar a cabo la masterización de Ubuntu a PhEANix; el orden en el que están presentados no implica necesariamente que se deben realizar de esta forma, es únicamente el orden en el que yo los ejecute.

- La creación del Usuario y el Grupo
- El cambio de los fondos de pantalla
- El cambio del splash de arranque
- El cambio del Spash de Login
- La Localización de la Distro
- La Primera Prueba.

a. La creación del Usuario y el Grupo

En Ubuntu como en todas las distribuciones de Linux existen dos formas de crear usuarios, mediante la interfaz gráfica ó mediante la línea de comandos, aquí tratare de presentar ambas formas.

El objetivo de realizar este proceso es generar nuestra propia cuenta para el inicio automático del LiveCD.

La herramienta de configuración de usuarios se inicia seleccionando la opción Users and Groups del menú System y dentro de este la opción Administración, entonces aparecerá en pantalla una ventana como esta.



Ilustración 36: Configuración de Usuarios.

Aquí se selecciona el botón Add User con lo cual aparecerá un cuadro de dialogo para diligenciar los datos de la nueva cuenta de usuario que se va a crear (Username, Nombre real, Perfil, información de Contacto y contraseña), como el siguiente.



Ilustración 37: Creación de la cuenta de usuario.

A continuación debemos definir los privilegios de la nueva cuenta (se sugiere que se tomen los datos de la cuenta por defecto de Ubuntu, previamente, utilizando la misma utilidad).

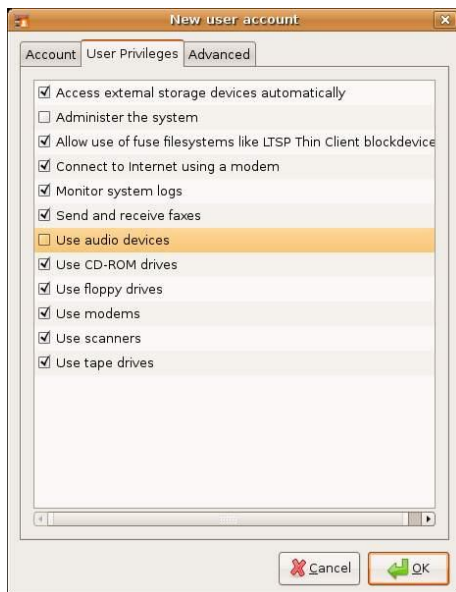


Ilustración 38: Privilegios default de Ubuntu.

Por último en la pestaña de Avanzados se configura la información relacionada con el directorio de trabajo por defecto de la cuenta, el grupo al que pertenece, el ID único del usuario y tipo de shell que ejecuta.



Ilustración 39: Datos de Home, Shell y Grupo.

Con esta información completa, se puede verificar nuevamente y la cuenta debe aparecer creada, ahora es posible hacer el primer logon. El siguiente paso es la creación del grupo al que va a pertenecer la cuenta, personalmente sugiero que se cree un grupo con el mismo nombre de la cuenta.

En los sistemas Linux toda cuenta de usuario pertenece a uno más grupos, sin embargo siempre tienen un grupo principal que es el que vamos a crear, el resto de grupos se sugiere que se tomen de la información previa de la cuenta original de Ubuntu.

La herramienta de configuración de grupos se inicia seleccionando la opción Users and Groups del menú System y dentro de este la opción Configuration Group, entonces aparecerá en pantalla una ventana como esta.



Ilustración 40: Configurando el Grupo.

Se debe entonces hacer clic en el botón Add Group y aparecerá una ventana como la siguiente donde se debe proporcionar la información del grupo (Nombre del grupo, ID del grupo y las cuentas miembros del mismo).



Ilustración 41: Asignando el ID al Grupo.

Una vez creado el nuevo grupo, es posible modificar las propiedades básicas como el ID, el nombre y los miembros utilizando el botón Properties, como se muestra a continuación.



Ilustración 42: Cambio de Propiedades del Grupo.

Ahora bien esta es solo la forma grafica para la creación de la cuenta y del grupo por la interfaz gráfica, veamos como es el proceso mediante línea de comandos utilizando una Terminal.

La creación del grupo

```
sudo addgroup pheanix
```

La creación de la cuenta

```
sudo -home /home/pheanix pheanix
```

Adicionar la cuenta al grupo

```
sudo adduser pheanix pheanix
```

Como siempre para cualquier referencia más exacta sobre la utilización de los comandos aquí expuestos es conveniente revisar la ayuda en línea (*man comando*), que proporciona cada distribución de Linux, en este caso Ubuntu.

b. El cambio de los fondos de pantalla

El procedimiento es realmente sencillo, sin embargo debemos tener presente que el LiveCD de PhEANix, es una masterización a partir de Ubuntu 7.04, el cual viene con el ambiente de escritorio Gnome 2.18 por defecto.

En PhEANix, adicionamos el ambiente de escritorio KDE 3.XX, por lo cual no solo veremos cómo se cambian los fondos de pantalla en Gnome, sino también en KDE.

Procedimiento para ambientes de escritorio GNOME 2.20 y posteriores en Ubuntu, 7.04 y posteriores.

Abrimos la ventana «Preferencias de la apariencia». Podemos hacerlo de varias maneras:



Ilustración 43: Cambio Fondo del Escritorio.

Yendo al menú Sistema -> Preferencias -> Apariencia.

Desplegando el menú contextual (botón secundario del ratón) sobre el escritorio y seleccionando Cambiar el fondo del escritorio.

Allí seleccionamos la pestaña Fondo. Nos aparecerá una lista, titulada «Tapiz», con miniaturas de los fondos de escritorio por defecto. Si queremos agregar un fondo, presionamos el botón «Añadir ...» y seleccionamos la imagen que queremos del cuadro de diálogo Abrir. Si, en cambio, queremos quitar uno de la lista, lo seleccionamos y presionamos el botón «Quitar».

Seleccionamos el fondo que queremos de la lista haciéndole clic, en el caso de PhEANix, y por tratarse de una masterización, primero hemos creado las imágenes que utilizaremos como fondo de la pantalla.



Ilustración 44: Preferencias del Fondo del Escritorio.

Podemos configurar aspectos como el estilo del fondo que indica cómo debe mostrarse la imagen (mosaico, centrado, expandido, etc.) y los colores del escritorio en caso de haber quitado el fondo de escritorio y haberlo dejado "sin tapiz"; en esta última opción podemos hacer degradados, tanto horizontales como verticales, que en el caso de PhEANix es la técnica que utilizamos.

Presionamos el botón «Cerrar» para cerrar la ventana.

Ahora exploraremos el Procedimiento para ambientes de escritorio KDE 3.XX en Ubuntu, 7.04 y posteriores.

Abrimos el módulo de configuración «Fondo», incluido dentro de la configuración de KDesktop. Podemos hacerlo de varias maneras:

Yendo al menú K -> Preferencias del sistema, y en la sección Apariencia seleccionando Fondo.

Desplegando el menú contextual (botón secundario del ratón) sobre el escritorio, y seleccionando Configurar Escritorio.

Nos aparecerán varias opciones para configurar el fondo:

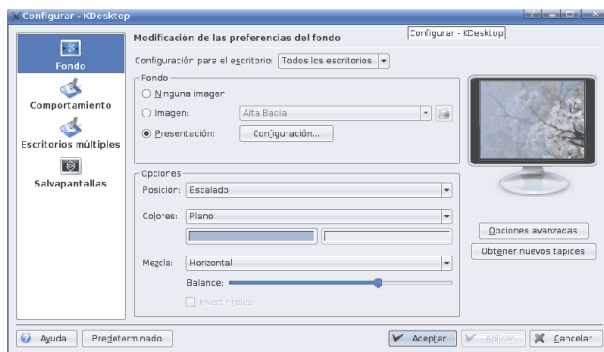


Ilustración 45: Configuración de KDesktop.

Configuración para el escritorio: Nos permite seleccionar un escritorio específico para configurar, o bien configurarlos todos. Esto es útil para escoger un fondo distinto para cada escritorio.

Fondo: Aquí especificamos las imágenes que usaremos de fondo, pudiendo optar por las siguientes opciones.

Ninguna imagen.

Imagen: una única imagen de fondo; podemos seleccionar los fondos por defecto de la lista, o bien pulsar el botón con el icono de la carpeta para buscar uno específico.

Presentación: nos permite agregar varios fondos a la vez, de manera que estos vayan cambiando cada cierto tiempo. Pulsando el botón Configuración podemos especificar los fondos (ficheros de imagen o carpetas completas) y el intervalo de tiempo para cambiar el fondo, además podemos hacer que los fondos se cambien aleatoriamente.

Opciones: Incluye configuraciones adicionales y algo más avanzadas, para los fondos.

Posición: permite especificar la posición del fondo (mosaicos, centrado, escalado, etc.), es como el estilo de GNOME.

Colores: al igual que en GNOME, podemos especificar los colores en caso de no tener una imagen de fondo; también podemos hacer degradados, tanto verticales como horizontales.

Una vez que hayamos cambiado y configurado el fondo, presionamos el botón «Aceptar» para guardar los cambios y cerrar el módulo de configuración.

c. El cambio del splash de arranque

Básicamente el objetivo de ejecutar los pasos que vamos a describir a continuación, es el de reemplazar la pantalla inicial que utiliza Ubuntu mientras se carga el sistema.

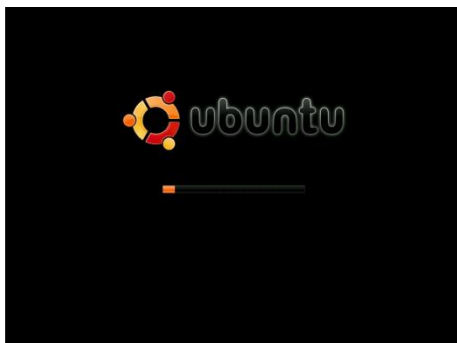


Ilustración 46: Imagen del Boot Splash Ubuntu 7.0.4 Original.

Por esta otra que hemos diseñado previamente.

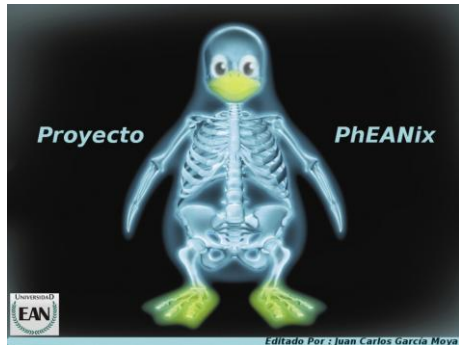


Ilustración 47: Imagen del Boot Splash PhEANix Original.

Antes de entrar en el proceso de reemplazo de la imagen enumeramos algunos tips importantes en la construcción de la imagen que se va a emplear.

La imagen se debe crear en resolución de 640x480, en modo indexado de 16 colores, por lo que no se puede utilizar una excelente foto. Se debe tener presente que algunos de los colores se utilizan para propósitos especiales, como el de señalar a los mensajes de texto que aparecen en pantalla, como se inician los servicios, por lo que la paleta utilizable es aún más restringida.

Estamos listos con la imagen, ahora para personalizar la pantalla de inicio, debemos instalar GCC y la el paquete de desarrollo BOGL framebuffer library development, para lo que utilizaremos el siguiente comando desde una Terminal.

```
$ sudo apt-get install gcc libbogl-dev
```

Finalizada la instalación de este utilitario procedemos a crear un directorio de trabajo donde copiaremos nuestra nueva imagen y procederemos a compilarla para generar un archivo objeto de Linux en formato Hexadecimal; recordemos que esta imagen, no es una simple imagen, es una imagen que será utilizada por el kernel del sistema en el proceso de inicio desde el boot, hasta tener la sesión de usuario; para enmascarar los procesos y las salidas en pantalla de estos.

Sugerimos utilizar estos pasos en el orden descrito. Teniendo en mente que el nombre de la imagen, así como las rutas dentro del sistema pueden ser modificadas de acuerdo con las preferencias de cada usuario.

```
$ mkdir usplash
$ cp myimage.png usplash/usplash-mine.png
$ pngtobogl usplash-mine.png > usplash-mine.c
$ gcc -Os -g -I/usr/include/bogl -fPIC -c usplash-mine.c \\  
    -o usplash-mine.o
$ gcc -shared -Wl,-soname,usplash-mine.so usplash-mine.o \\  
    -o usplash-mine.so
$ sudo cp usplash-mine.so /usr/lib/usplash/usplash-mine.so
$ sudo ln -sf /usr/lib/usplash/usplash-mine.so \\  
    /usr/lib/usplash/usplash-artwork.so
$ sudo dpkg-reconfigure linux-image-\Quname -r\Q
$ sudo ln -sf /usr/lib/usplash/usplash-default.so \\  
    /usr/lib/usplash/usplash-artwork.so
$ sudo dpkg-reconfigure linux-image-\Quname -r\Q
```

Para el caso, de ejemplo que estoy planteando, el archivo objeto resultante se llama usplash-mine.so, pero puede darle un nombre diferente, como lo mencione; debe tener presente que no podrá llamarlo usplash-artwork.so, por que este es el nombre que utiliza la imagen por defecto de Ubuntu actualmente para encontrar el bootsplash obra de arte. Ahora ya tenemos una imagen de inicio valida, sin embargo debemos integrarla al proceso de arranque. Para lo cual es necesario crear un link al archivo .so almacenado en el initramfs (RAM inicial de ficheros).

Como el initramfs está instalado de modo que fuerza al núcleo de paquetes instalados actualmente a ser reconfigurado; una vez que se han creado e instalado en /usr /lib /usplash, puede cambiar entre diferentes pantallas splash, incluida la original Ubuntu splash; para lo cual simplemente se debe cambiar ó actualizar el enlace simbólico del link y la regeneración de la initramfs.

d. El cambio del Spash de Login

PhEANix en modo LiveCD inicia automáticamente dentro de una sesión de usuario KDE, sin embargo pensando en que los usuarios deseen realizar la instalación en disco duro más adelante, y teniendo en cuenta que la distro incorpora ambos ambientes gráficos, es decir (KDE y Gnome), he decidido personalizar también la pantalla de logon, en la que además de autenticarse podrá seleccionar con cual de los dos escritorios trabaja.

En primer lugar, vamos al menú Sistema, Administración, seleccionamos ventana de inicio de sesión; aparecerá un diálogo, allí seleccionamos la pestaña de seguridad, como se muestra en las siguientes pantallas.

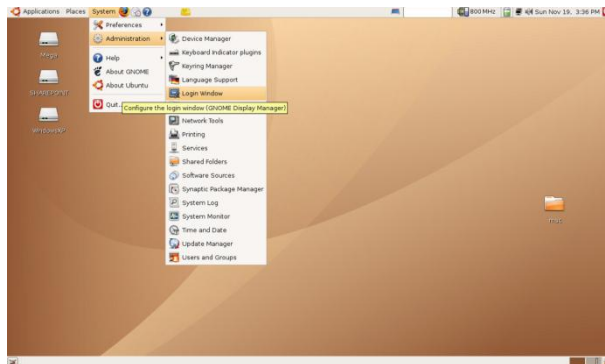


Ilustración 48: Menú de Administración.

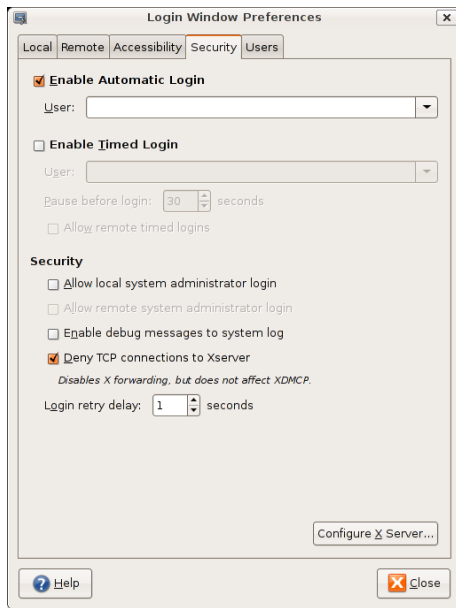


Ilustración 49: Preferencias de la pantalla de Logon.

En esta pantalla seleccionamos la opción habilitar inicio de sesión automática; a continuación, seleccionamos el nombre del usuario por defecto, que en este caso es pheanix. El siguiente paso es seleccionar la pestaña de local, en la que definiremos cual es la imagen que veremos en el logon.



Ilustración 50: Selección imagen de logon.

En esta pestaña también se pueden personalizar los mensajes que se reciben en pantalla, sugiero que los cambien a español. Al igual que en todos los demás procesos, que involucren imágenes, estas deben ser construidas previamente.

e. La Localización de la Distro

Ya hemos realizado bastantes cambios, pero aún nuestra distro instalada esta en idioma ingles, que es la instalación por defecto. Esto no es malo, pero por ser hispano parlante, es mejor instalar los paquetes de idiomas para español.

En este punto hay dos formas de hacerlo, mediante los administradores de paquetes gráficos o mediante el modo de línea de comandos.

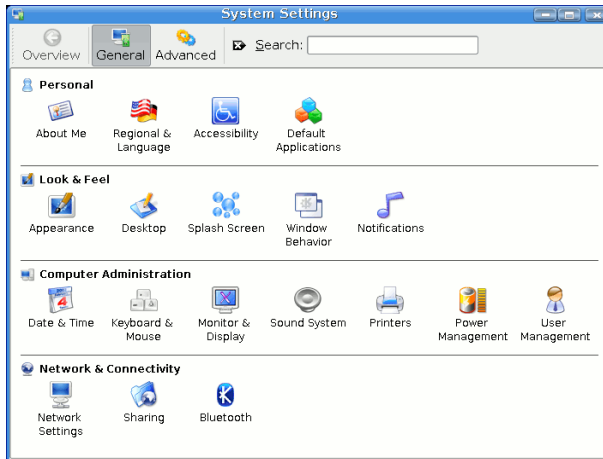
Veamos inicialmente la línea de comandos, que creo es la más sencilla.

```
$ sudo apt-get install <nombre del paquete>  
$ sudo apt-get install language-pack-kde-es
```

Así sucesivamente para cada uno de los paquetes que enumero a continuación:

- language-support-es
- language-pack-kde-es
- language-pack-kde-es-base
- language-pack-es
- language-pack-es-base
- language-pack-gnome-es
- language-pack-gnome-es-base
- language-pack-is
- language-pack-af
- language-pack-am
- language-pack-an
- language-pack-ar
- language-pack-as

Una vez instalados los paquetes de idioma necesarios, es conveniente ajustar la configuración predeterminada de los escritorios, para lo cual ingresaremos al panel de control y modificaremos la configuración regional y de idiomas como lo muestro en las siguientes pantallas.



Ahora nuestro escritorio y el sistema estarán en español.

Ilustración 51: Ajuste de Idioma de los Escritorios KDE y Gnome.

f. La Primera Prueba.

Bien en este punto ya tenemos lista la primera versión de la distro para iniciarla, veamos algunas pantallas que he capturado con la ayuda de Vmware Player.

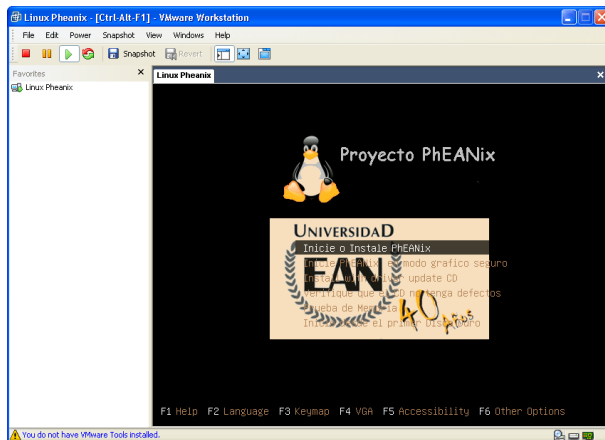


Ilustración 52: Arranque del LiveDVD PhEANix.

Al iniciar el LiveCD esta es la primera pantalla que aparece, es el menú de las opciones de boot; en este menú se puede seleccionar entre otros el tipo de teclado, el idioma, etc.

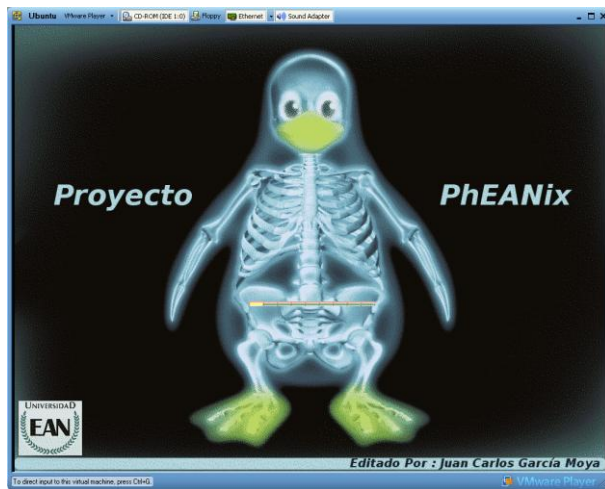


Ilustración 53: Boot Splash de PhEANix.

Este es el boot splash, que personalizamos en uno de los pasos anteriores y el cual enmascara todo el proceso de boot del LiveCD hasta que se inicia la sesión de usuario; en este punto podemos ver una barra de progreso que indique porcentaje de avance en el proceso de boot.

Las siguientes pantallas corresponden al inicio de sesión en el entorno KDE, que es el entorno gráfico predeterminado para el LiveCD de PhEANix.

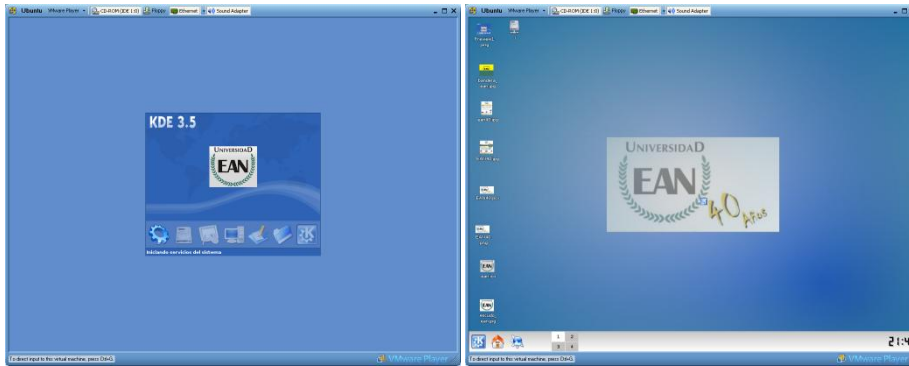


Ilustración 54: Inicio de Sesión KDE PhEANix.

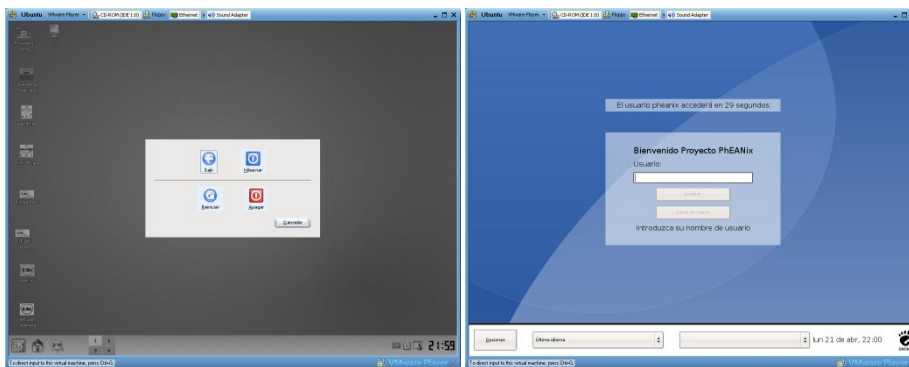


Ilustración 55: Cierre de Sesión KDE PhEANix.

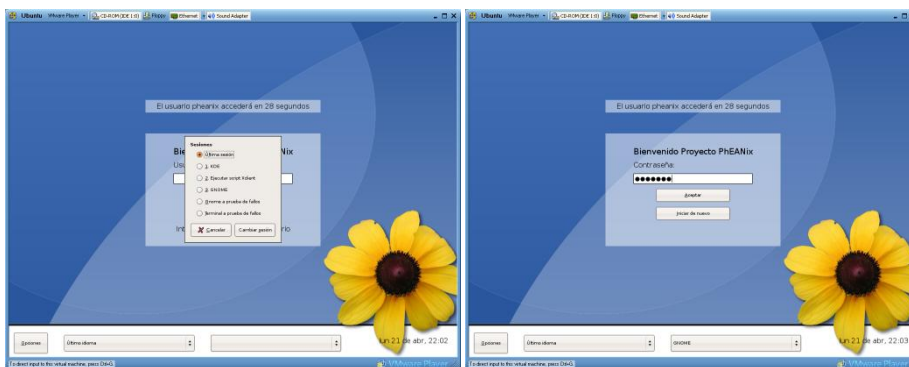


Ilustración 56: Cambio de Entorno y Usuario.

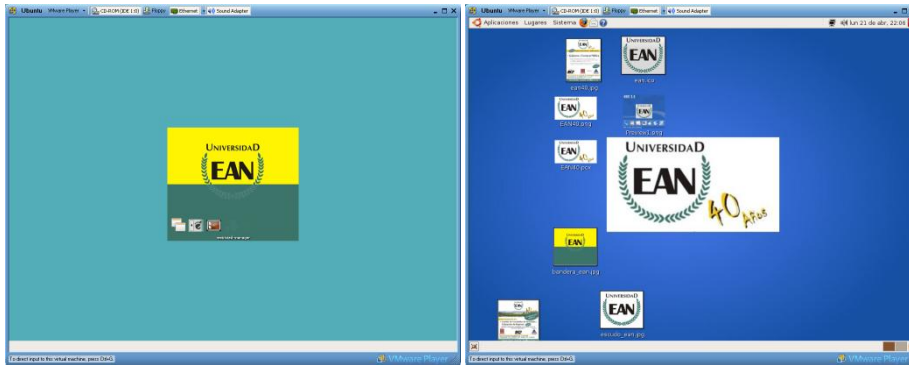


Ilustración 57: Inicio de Sesión GNOME PhEANix.

Anexo 4. Ampliación conceptual de Conocimiento y Objetos de Conocimiento

A. Visión filosófica clásica del conocimiento:

Según Platón, el conocimiento se caracteriza por ser necesariamente verdadero (episteme); de manera que la mera creencia y/ó opinión “ignorante de la realidad de las cosas”, quedan meramente en el plano de lo probable y lo aparente.

Podemos afirmar desde esta perspectiva, que una certeza de hoy, que el día de mañana se probará falsa, en verdad nunca habría sido conocimiento.

B. Visión científico técnica del conocimiento:

En ciencias de la información, se acostumbra a definir un esquema continuo, progresivo y complejo, como el de la siguiente figura, el cual está conformado por los datos, la información, el conocimiento y la sabiduría.

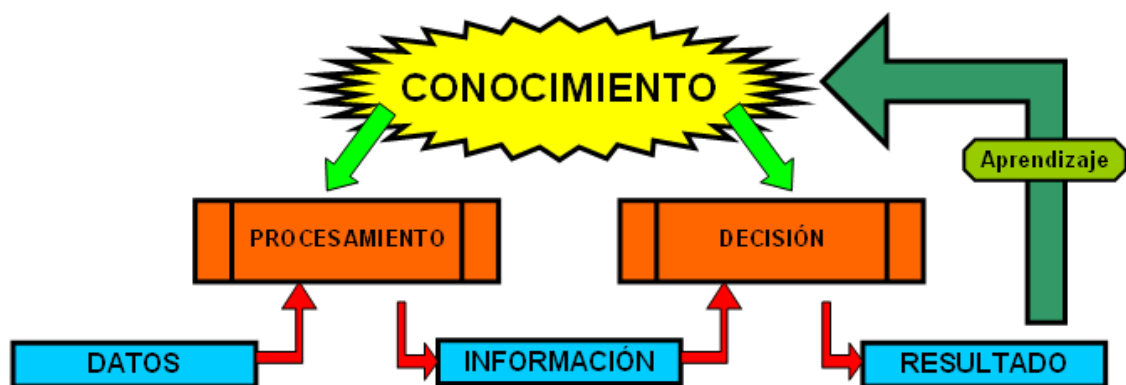


Ilustración 58: Correlación Conocimiento - Información y Aprendizaje.

Entonces, se define al conocimiento como un conjunto organizado de datos e información cuyo único fin es permitir la resolución un determinado problema.

Los objetos de conocimiento son objetos que nos permiten alcanzar el conocimiento y por ende la sabiduría; para lo cual se aplica un método, existiendo así múltiples vías de llegar obtener el conocimiento:

Método empírico

Método histórico

Método lógico

Analogía, etc.

En general, para que una creencia constituya conocimiento científico no basta con que sea válida y consistente lógicamente, pues ello no implica su verdad. Así por ejemplo, si tomamos un sistema lógico deductivo consistente y válido; y negamos la totalidad de las premisas del sistema, se obtendrá un sistema igualmente consistente y válido, sólo que contradictorio al sistema previo.

De tal manera, validez no garantiza verdad. Para que una teoría deba ser considerada como verdadera, deben existir, desde el punto de vista de la ciencia, pruebas que la apoyen. Es decir, debe poder demostrarse su verosimilitud empleando el método científico, también conocido como método experimental.

Esto sin embargo se ve seriamente complicado si se introducen interrogantes relativas a la suficiencia de dicho método, como por ejemplo, la transparencia de los hechos ¿existen los hechos puros o más bien interpretaciones?, la factibilidad de la pretensión de objetividad y neutralidad valórica ¿es posible la comprensión de la realidad desde un punto de vista neutro, tal como fuera el de un dios, o estamos condenados a perspectivas?, etc.

El siguiente punto que debemos considerar, es como encontrar el ó los objetos de conocimiento que nos interesan, en un mundo de infinitos objetos de

conocimiento. Para ello nos vemos obligados a catalogar cada uno de los objetos de conocimiento empleando metadatos; esto supone un problema adicional que consiste en la selección del estándar a utilizar, dentro de los varios que existen en la actualidad:

- Dublin Core: Estándar propuesto por la DCMI (Data Model Content Example), (The Dublin Core Metadata Initiative (DCMI), 2012) un grupo de trabajo internacional especializado en tecnología educativa. Los metadatos de Dublin Core son un elemento estándar para la transferencia de dominio de recursos de información para la descripción, de la misma.

Proporcionan una forma simple y un conjunto normalizado de convenciones para describir las cosas en línea de manera que sea más fácil encontrarlos. Dublin Core se utiliza ampliamente para describir materiales digitales como vídeo, sonido, imagen, texto y compuesto los medios de comunicación como páginas web. Algunas Implementaciones de Dublin Core suelen hacer uso de XML y son Resource Description Framework basado. Este estándar fue definido por la norma ISO 15836 en el 2003 y la norma NISO Z39.85-2007.

- IMS Global Learning Consortium: (generalmente conocido como IMS), (IMS Global Learning Consortium, Inc. , 2012) es un consorcio de instituciones y empresas que dispone de un estándar propio, orientados al establecimiento de la interoperabilidad de los sistemas de aprendizaje y los contenidos de aprendizaje y de la empresa de integración de estas capacidades. Su misión es "apoyar la adopción y el uso de la tecnología de la enseñanza en todo el mundo". Su principal actividad es el desarrollo de las especificaciones, algunos de los cuales como QTI y Contenido de embalaje son ampliamente utilizados.

El alcance de las especificaciones IMS cubren la mayoría de los elementos de datos utilizados en el "aprendizaje distribuido", incluyendo configuraciones tanto en línea como fuera de línea, que tienen lugar sincrónicamente (en tiempo real) o asincrónicamente.

Las especificaciones de IMS promueven la reutilización de E-learning y el contenido de Internet de entornos específicos (como los cursos de un sistema de gestión basados en la web), así como situaciones de aprendizaje que involucran fuera de línea, recursos electrónicos, como por ejemplo un estudiante al acceder los recursos en un CD-ROM.

Los alumnos pueden estar en un entorno de enseñanza tradicional, es decir, una escuela en un aula universitaria, en una empresa o establecimiento de formación del gobierno, o en el hogar.

- LOM: Learning Object Metadata. Es una iniciativa de la IEEE "Institute of Electrical and Electronic Engineers", quizás la que más éxito ha tenido hasta el momento. Iniciado en 1997 y reconocido en junio del 2002.

Learning Object Metadata es un modelo de datos, por lo general codificados en XML, utilizado para describir un objeto de aprendizaje y similares recursos digitales utilizados para apoyar el aprendizaje. El objetivo de incluir metadatos en la utilización de objetos de aprendizaje es prestar apoyo a la reutilización de estos objetos de aprendizaje, la ayuda para el descubrimiento, y para facilitar su interoperabilidad, por lo general en el contexto de los sistemas de gestión de aprendizaje en línea (LMS).

El estándar IEEE 1484.12.1 del 2002 para Learning Object Metadata es un estándar abierto, reconocido internacionalmente; fue publicado por el Instituto de Ingenieros Eléctricos y Electrónicos de Normas, de Nueva York para la descripción

de "objetos de aprendizaje". Pertinentes de los atributos de objetos de aprendizaje que se describen son: tipo de objeto; autor; propietario; términos de la distribución, el formato y los atributos pedagógicos, como la enseñanza o la interacción estilo. La siguiente es una representación esquemática de la jerarquía de los elementos del modelo de datos LOM

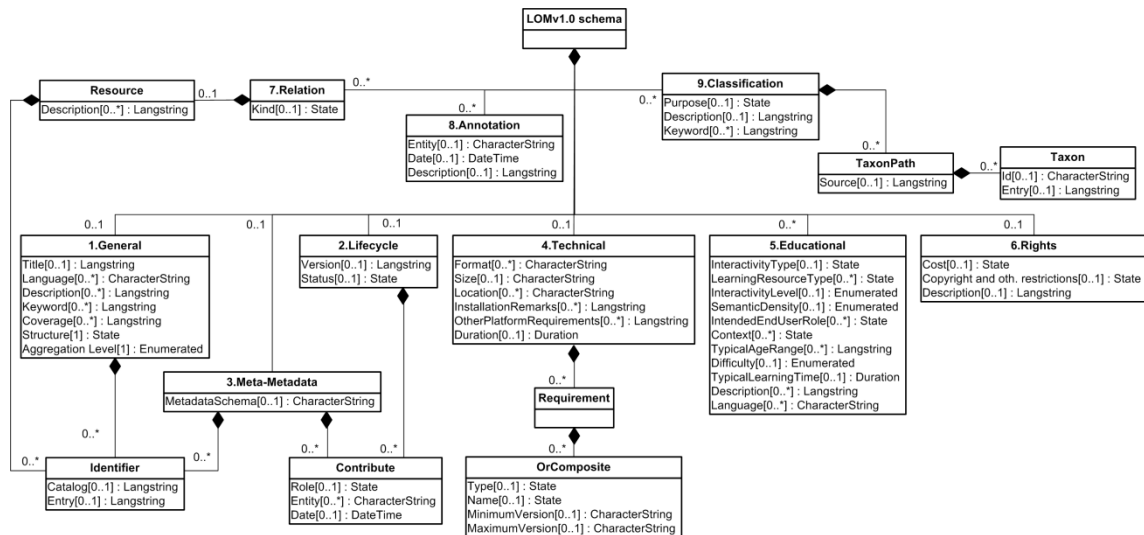


Ilustración 59: Representación de Modelo de Objetos LOM.

- CanCore: Grupo canadiense con un sistema compatible con el IMS, en realidad se trata de un subconjunto del mismo. Es una iniciativa para la definición de metadatos de recursos de enseñanza-aprendizaje (CanCore Learning Resource Metadata Initiative) trata de mejorar la capacidad de educadores, investigadores y estudiantes en Canadá y en todo el mundo para buscar y localizar material para establecer repositorios de recursos educativos online.

CanCore se basa y es totalmente compatible con el estándar IEEE Learning Object Metadata (LOM) y la especificación IMS Learning Resource Meta-data.

CanCore proporciona guías detalladas para interpretar e implementar cada uno de los elementos del estándar LOM del IEEE. Estas guías han sido desarrolladas a lo

largo de varios años, después de consultar a expertos de Canadá y del resto del mundo y han sido puestas a disposición de todo el mundo desde su sitio en Internet.

Todos estos criterios de catalogación no son más que descripciones del “ser” de cada Objeto. Al ser adaptados al ámbito educativo, no sólo recogen datos bibliotecarios como: el autor, precio, copyright, etc.; sino que abarcan otros tipos de datos curriculares, del ciclo de vida, el nivel educativo, etc.

En concreto los metadatos LOM operan en base a 9 categorías:

- General. Agrupa todas las propiedades independientes del contexto del Identificador, Título, Descripción, etc.
- Ciclo de Vida. Versión, Fecha, revisión, etc.
- META-Metadatos. Propiedades de los metadatos usado en la descripción del recurso.
- Técnicos. Formato, tamaño, Sistema Operativo, etc.
- Educativos. Tipo, clasificación, aproximación, granularidad, etc.
- Derechos de Autor. Rol, condiciones Uso, precio, etc.
- Relación. Tipo, recurso con el que se relaciona
- Anotación. Comentarios relacionados con el uso educativo del recurso. Autor, fecha, comentario.
- Clasificación. Describen el recurso en función de algún sistema de clasificación.

Este proceso de catalogación de Objetos hace más lenta y encarece la generación de contenidos educativos, pero sin este, nos enfrentaríamos al inmanejable inconveniente, que supone la incapacidad de localizar los objetos de nuestro interés ante la gran cantidad de Objetos existentes; sería tanto como si un buen

día desaparecieran todos los buscadores de la Internet; las páginas de nuestro interés seguirían estando ahí, pero no podríamos encontrarlas, o perderíamos mucho tiempo para poder llegar finalmente a ellas. El conocimiento basado en comunidades colaborativas se vería seriamente afectado.

Anexo 5. Formato y Contenido de los objetos de conocimiento que se pueden empaquetar en PhEANix

Antes de ir a cada herramienta y comenzar el trabajo de creación de los diferentes objetos de conocimiento, debemos establecer tanto la forma como el contenido que van a tener. Por lo que la metodología que sugerimos que se siga es la de definir estos dos puntos antes de utilizar cualquier herramienta.

Creemos que realizar un Guion es la manera más sencilla y eficaz de organizar el detalle del temario a seguir para cada uno.

a) Documentos HowTo (Tradicionales ¿Cómo Hacer?):

Como lo mencionamos previamente son documentos orientados a ilustrar mediante una serie de pasos un proceso de uso frecuente por parte de los usuarios.

El siguiente es un ejemplo de documento HowTo que se pueden empaquetar mediante el uso de este proyecto; lo presentamos con el fin de sentar las bases para futuras actualizaciones y/o publicaciones.

Estos documentos siguen algunas "normas" o consejos tomados de la misma comunidad Linux, que se deben seguir para escribirlos.

¿Cómo iniciar una sesión de LiveCD?

Antes de publicar un documento HowTo / Como, es conveniente seguir los siguientes pasos:

1. Asegúrese que no exista un tutorial de la misma temática ya publicado (Esto aplicara para cuando el material se suba a un repositorio único de la EAN).
2. Si encuentras un documento publicado que coincide, puede tomar alguna de las siguientes opciones:
 - Postear el documento publicado para actualizarlo (en caso de que haya puntos mejorables o actualizables).

- Contactarse con el moderador del sitio o del tema de publicado, o con el responsable de la edición del mismo para realizar las mejoras.
- Utilice un **título descriptivo**, precedido de la palabra **[Como]** o **[Howto]** delante del mismo.
- **Exponga sus fuentes de información**, o indique, que el documento describe su propia experiencia.
- Si está colocando referencias textuales de otro documento publicado al respecto (copiando / pegando), **cite SIEMPRE al autor**.
- Especifique siempre las **versiones de Sistema operativo (Kernel) y aplicaciones o utilitarios** que está utilizando y la arquitectura de hardware sobre la que está corriendo, (Intel, 32 bits, AMD 64, etc.).
- Explique lo más detalladamente posible **para qué sirve el documento Howto** que está publicando.
- Cuando sean necesario la utilización de aplicaciones o librerías extra, expóngalo claramente.
- Intente explicar la manera de deshacer lo que se está haciendo.
- Para que un documento howto sea fácil de leer, emplee formatos de texto que diferencien las explicaciones, de las líneas de comando. Por ejemplo, emplee la cursiva para las líneas de código.
- Advierta al lector sobre las consecuencias que puede tener al seguir las indicaciones del documento HowTo; A veces son simples trucos, pero en otras ocasiones, el lector puede correr el riesgo de desestabilizar su sistema.
- Poner solo un link a otro documento publicado fuera del sitio web no se considera un documento HowTo, por tanto será desechado.
- Los documentos HowTo que contengan errores garrafales de forma y/o concepto, y que puedan representar potencial riesgo para los lectores en su ejecución serán desechados.

[HOWTO] Autenticación de un usuario en Linux.

Descripción: Este documento explica las diferentes formas que tiene un usuario para tener acceso a un sistema Linux.

Publicado el: 10/Nov/2009

Actualizado el: 10/Nov/2009

Calificación: * * * * *

Una vez ubicado el documento WhoTo dentro de la lista, debemos ingresar a él para ver su contenido, generalmente veremos un índice que nos mostrara los temas allí contenidos y nos permitirá acceder a la información.

[HOWTO] Autenticación de un usuario en Linux.

Juan Carlos García
(v0.1).

2009-11-10

Historial de Revisiones

Revisión 0.1 2009-11-10 Revisado por: Juan Carlos García

Este documento explica las diferentes formas que tiene un usuario para tener acceso a un sistema Linux.

Tabla de Contenido

Tabla de contenidos

1. Introducción
 - 1.1. ¿Por qué se genero este documento?
 - 1.2. Sobre las actualizaciones
 - 1.3. Reacción
 - 1.4. Derechos de autor y marcas registradas
 - 1.5. Reconocimientos y agradecimientos
 - 1.6. Supuestos sobre el lector
2. ¿Qué información de usuario es almacenada en el sistema?
 - 2.1. /etc/passwd
 - 2.2. La contraseña oculta
 - 2.3. /etc/group y /etc/gshadow
 - 2.4. MD5 contraseñas encriptadas
- 3.1. ¿Por qué?
- 3.2. ¿Qué Hace PAM?
- 3.3. Las distribuciones que usan PAM
4. Conclusión

1. Introducción

1.1. ¿Por qué se genero este documento?

Este documento se genero a partir de la recurrente consulta de los usuarios de la red que atendemos, por cuanto siempre se presentaban diferentes soluciones al mismo inconveniente "Problemas de Autenticación", hasta que finalmente decidí investigar cómo funciona la autenticación de usuarios en Linux, y escribir un procedimiento que lo explicara, así como que permitiera que otros administradores interesados pudieran apoyar a sus usuarios.

1.2. Sobre las actualizaciones

Por ahora este documento se limitara a estar contenido en el LiveCD PhEANix y eventualmente se podrá actualizar si es requerido en futuras versiones

1.3. Comentarios y/o Correcciones

Los comentarios, correcciones y/o sugerencias, relacionadas con este documento pueden ser enviados por correo electrónico garmoya@yahoo.com.

1.4. Derechos de autor y marcas registradas

Este documento se acoge a las leyes del copyright de las licencias GNU y GPL, por lo que se permite su reproducción total o parcial del mismo.

1.5. Reconocimientos y agradecimientos

Gracias a mi familia por soportar tantas ausencias dedicadas a la creación de estos documentos. Gracias a la gente de que postea documentos en todas las comunidades Linux alrededor del mundo, a los proyectos Debian, Ubuntu, Fedora, etc.

1.6. Supuestos sobre el lector:

Para los efectos de este documento, se asume que el usuario es novato en el mundo Linux, pero que de algún modo ha tenido interacción con los sistemas de computadores en su entorno de la casa, la universidad y/o el trabajo.

2. ¿Qué información de usuario es almacenada en el sistema?

2.1. `/etc/passwd` : En la mayoría de las distribuciones Linux, tanto libres como Comerciales, la información del usuario se almacena en el archivo de texto `/etc/passwd`, en este se guardan la conexión del usuario, su contraseña encriptada, el ID único, en formato numérico, llamado UID, un ID numérico del grupo, llamado GID, un comentario opcional, que generalmente se usa para colocar los datos reales del usuario y su información de contacto, etc., su directorio principal dentro del sistema, y su shell predefinida (interface de comandos).

Una entrada típica dentro del archivo `/etc/passwd` sería como la siguiente:

```
pheanix: XXXXXXXXXXX: 1000:1000: Proyecto Pheanix,, ,Universidad EAN: /home/pheanix: /bin/bash
```

Es una línea muy simple que contiene los 6 campos referidos anteriormente, veamos que el separador de campo es el signo dos puntos (:)

2.2. La contraseña oculta

Si mira en su sistema es probable que el archivo `/etc/passwd` se parezca más a esto

```
pheanix: x: 1000:1000: Proyecto Pheanix,, ,Universidad EAN: /home/pheanix: /bin/bash
```

El campo de la contraseña ya no nos muestra sino una x, que indica que existe una contraseña, pero no nos deja ver el valor cifrado.

¿A dónde quedo la contraseña cifrada?

Como se han dado cuenta, el archivo `/etc/passwd`, es susceptible a ser visualizado por todos los usuarios dentro del sistema, y dada la importancia del contenido que este tiene, por contener incluso las contraseñas cifradas hace vulnerable un sistema sin asegurar. Para prevenir esta potencial amenaza a la seguridad, se desarrollaron las contraseñas ocultas o, shadow.

Cuando se activan las contraseñas ocultas o shadow dentro del sistema, el campo de la contraseña en el archivo `/etc/passwd` se sustituye por una "x" y la contraseña cifrada del usuario real se almacena en otro archivo `/etc/shadow`.

Este archivo `/etc/shadow` es legible únicamente por el super usuario (root), de tal suerte que los usuarios malintencionados no se puede crackear contraseñas a los usuarios de sus compañeros. Las entradas en el archivo `/etc/shadow` contienen la conexión del usuario, su contraseña encriptado, y una serie de ámbitos relacionados con la caducidad de la contraseña. Un archivo típico es:

```
pheanix: / 3GJllg1o4152: 11009:0:99999:7:::
```

2.3. `/etc/group` y `/etc/gshadow`: Hacen relación a la información del grupo al que pertenecen los usuarios y el formato de estos archivos es similar al de los anteriores.

Las entradas que contiene campos para el nombre del grupo, la contraseña, ID numérico (GID), y una lista separada por comas de los miembros del grupo. Una entrada en `/etc/group` tiene un aspecto similar al siguiente:

```
Ingenieria: x: 103:, administradores, estudiantes, docentes
```

Como se aprecia, la "x" en el campo de contraseña, se entiende que las contraseñas de grupo se pueden ocultadas también. Aunque los grupos casi nunca tienen sus propias contraseñas, vale la pena señalar que la información oculta de las contraseñas de grupos, se almacena en `/etc/gshadow`.

2.4. MD5 contraseñas encriptadas: Tradicionalmente, las contraseñas Unix se cifra con un la función estándar `crypt()`, lo que eventualmente con el crecimiento de los desarrollos tanto del hardware como del software se hizo más fácil romper dichas contraseñas.

Con la expansión de internet, nuevas herramientas de craqueo de contraseñas se difundieron rápidamente, obligando a mejorar la seguridad mediante algoritmos de encriptación más fuertes. Ahora la mayoría de las "nuevas" distribuciones vienen con la opción de encriptar las contraseñas con algoritmos Hash MD5.

(Para más información sobre el algoritmo de hash MD5, consulte RFC 1321.)

Lo anterior no elimina la posibilidad de craqueo de contraseñas, pero dificultaran más la tarea de quien lo intente.

2.5. Escudriñar en el lío: Hemos visto que existen una serie de formas para realizar la encriptación de los datos que autentican a un usuario y las diferentes formas en que esta información se almacena.

Ahora la pregunta es, ¿Qué sucede con los programas que usan sus contraseñas, cuando usted cambia la forma de almacenarlas, y de encriptarlas?, Como saben donde verificar y como validarlas?

PAM es la respuesta. (Pluggable Authentication Modules)

3. PAM (Pluggable Authentication Modules)

Los Pluggable Authentication Modules se encuentran en el núcleo de la autenticación de usuario en cualquier distribución de Linux moderna.

3.1. Por qué?: Si miramos cómo funcionaba la autenticación Linux en sus inicios, si un programa, como su, passwd, login, o xlock, necesitaba autenticar a un usuario, simplemente leía la información necesaria en el archivo /etc/passwd.

Esto presentaba numerosos problemas para los administradores de sistemas y desarrolladores de aplicaciones. Con la adición de MD5 y la popularización de los métodos para ocultar las contraseñas, cada programa que requería autenticación de usuario tenía que saber cómo obtener la información adecuada; cuando se trataba de sistemas diferentes. Si se quería cambiar el esquema de autenticación de usuario, todos estos programas que lo requerían, prácticamente debían ser cambiados. PAM elimina este lío al permitir que los programas pudieran autenticar a los usuarios de forma transparente, independientemente de cómo se almacena la información del usuario.

3.2. Qué Hace PAM?: "Es el propósito del proyecto Linux-PAM para separar el desarrollo de software de la concesión de privilegios por el desarrollo de esquemas de autenticación segura y apropiada. Esto se logra proporcionando una biblioteca de funciones que una la aplicación puede utilizar para solicitar que se autentica un usuario " Tomado de **Linux-PAM - Guía del administrador del sistema**

Con PAM, no importa si su contraseña se guarda en / etc/passwd o en otro servidor. Cuando un programa necesita para autenticar a un usuario, PAM proporciona una biblioteca que contiene las funciones adecuadas para el esquema de autenticación. Es importante indicar que esta biblioteca se carga de forma dinámica, cambiando los esquemas de autenticación que se puede hacer simplemente editando un archivo de configuración.

La flexibilidad es una de las mayores fortalezas de PAM, este se puede configurar para negar el derecho de ciertos programas para autenticar a los usuarios, para permitir sólo a determinados usuarios sea autenticado, para advertir cuando algunos programas intentan autenticarse, o incluso para privar a todos los usuarios de los privilegios de acceso. El diseño modular de PAM le da un control completo sobre cómo se autentican los usuarios.

3.3. Las distribuciones que usan PAM.

Como todo en el mundo, cuando algo es bueno tiende a convertirse en un estándar, por eso casi todas las distribuciones populares han adoptado PAM en algún momento

RedHat, a partir de la versión 5.0, Mandrake desde la versión 5.2, Debian desde la versión 2.1, Caldera desde la versión 1.3, Turbolinux desde la versión 3.6, SuSE desde la versión 6.2, Ubuntu desde la versión 5.10, etc

LICENCIA DE USO – AUTORIZACIÓN DE LOS AUTORES

Actuando en nombre propio identificado (s) de la siguiente forma:

Nombre Completo Juan Carlos García Maya

Tipo de documento de identidad: C.C. T.I. C.E. Número: 79870823 Bta

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

Nombre Completo _____

Tipo de documento de identidad: C.C. T.I. C.E. Número: _____

El (Los) suscrito(s) en calidad de autor (es) del trabajo de tesis, monografía o trabajo de grado, documento de investigación, denominado:

Prototipo para el empaquetado de objetos de aprendizaje mediante Linux Live CD.

Dejo (dejamos) constancia que la obra contiene información confidencial, secreta o similar: SI NO
(Si marqué (marcamos) SI, en un documento adjunto explicaremos tal condición, para que la Universidad EAN mantenga restricción de acceso sobre la obra).

Por medio del presente escrito autorizo (autorizamos) a la Universidad EAN, a los usuarios de la Biblioteca de la Universidad EAN y a los usuarios de bases de datos y sitios webs con los cuales la Institución tenga convenio, a ejercer las siguientes atribuciones sobre la obra anteriormente mencionada:

- A. Conservación de los ejemplares en la Biblioteca de la Universidad EAN.
- B. Comunicación pública de la obra por cualquier medio, incluyendo Internet
- C. Reproducción bajo cualquier formato que se conozca actualmente o que se conozca en el futuro
- D. Que los ejemplares sean consultados en medio electrónico
- E. Inclusión en bases de datos o redes o sitios web con los cuales la Universidad EAN tenga convenio con las mismas facultades y limitaciones que se expresan en este documento
- F. Distribución y consulta de la obra a las entidades con las cuales la Universidad EAN tenga convenio

Con el debido respeto de los derechos patrimoniales y morales de la obra, la presente licencia se otorga a título gratuito, de conformidad con la normatividad vigente en la materia y teniendo en cuenta que la Universidad EAN busca difundir y promover la formación académica, la enseñanza y el espíritu investigativo y emprendedor.

Manifiesto (manifestamos) que la obra objeto de la presente autorización es original, el (los) suscritos es (son) el (los) autor (es) exclusivo (s), fue producto de mi (nuestro) ingenio y esfuerzo personal y la realizo (zamos) sin violar o usurpar derechos de autor de terceros, por lo tanto la obra es de exclusiva autoría y tengo (tenemos) la titularidad sobre la misma. En vista de lo expuesto, asumo (asumimos) la total responsabilidad sobre la elaboración, presentación y contenidos de la obra, eximiendo de cualquier responsabilidad a la Universidad EAN por estos aspectos.

En constancia suscribimos el presente documento en la ciudad de Bogotá D.C.,

NOMBRE COMPLETO: <u>Juan Carlos García Mayan</u>	NOMBRE COMPLETO: _____
FIRMA: <u>Juan Carlos García Mayan</u>	FIRMA: _____
DOCUMENTO DE IDENTIDAD: <u>78870923 Bte</u>	DOCUMENTO DE IDENTIDAD: _____
FACULTAD: <u>de Ingeniería</u>	FACULTAD: _____
PROGRAMA ACADÉMICO: <u>Ingeniería de Sistemas</u>	PROGRAMA ACADÉMICO: _____

NOMBRE COMPLETO: _____	NOMBRE COMPLETO: _____
FIRMA: _____	FIRMA: _____
DOCUMENTO DE IDENTIDAD: _____	DOCUMENTO DE IDENTIDAD: _____
FACULTAD: _____	FACULTAD: _____
PROGRAMA ACADÉMICO: _____	PROGRAMA ACADÉMICO: _____

Fecha de firma: 10/Dic/2012.