

1	CONCEPTOS PREVIOS E INSTALACIÓN DE LINUX.	1-2
1.1	Breve historia de Linux.	1-2
	Debian	1-6
	Ubuntu	1-7
	Gentoo	1-9
	Red Hat.....	1-10
	Suse	1-10
	Guadalinex.....	1-11
1.2	Software libre. Licencias GPL y BSD.	1-12
1.3	Instalación de Linux.	1-14
	Puntos de montaje.....	1-14
	Partición Swap.....	1-16
1.4	Carga de Linux.	1-16
	Configuración del arranque dual.	1-17
1.5	Sistema de Ficheros en Linux.	1-17

1 CONCEPTOS PREVIOS E INSTALACIÓN DE LINUX.

1.1 Breve historia de Linux.

En los años setentas, dos investigadores de los Laboratorios Telefónicos Bell (Bell Telephone Labs o **BTL**) llamados Dennis Ritchie y Ken Thompson desarrollaron un sistema operativo muy elegante al que llamaron **Unix**. Eligieron el nombre Unix como una burla al proyecto en el que habían trabajado anteriormente: Multics. Al completar el desarrollo de Unix, Ritchie y Thompson expusieron su diseño en una conferencia internacional donde varios de los participantes les pidieron una copia de este sistema. En esa época, BTL había perdido un juicio antitrust (antimonopolio), y el Juez había prohibido a BTL incorporarse a cualquier negocio que no fuera el de las telecomunicaciones. Debido a esto, a BTL le era imposible entrar en el negocio de los sistemas operativos. La presión de los investigadores en obtener una copia de Unix motivó a los ejecutivos de BTL a licenciar su uso como una herramienta de investigación. La licencia de Unix era muy barata para las universidades.



Una de las universidades que adquirió una licencia de Unix fue la Universidad de California en Berkeley. La motivación principal era poder experimentar con el primer sistema operativo que incluía código fuente. Al poco tiempo, la gente de Berkeley había escrito varios programas utilitarios para Unix que otros investigadores podrían encontrar útiles. La Universidad decidió entonces distribuir este código a la comunidad y le llamó a sus distribuciones **BSD** (Berkeley Software Distribution). A pesar que al principio las distribuciones de Berkeley consistían principalmente en herramientas para los usuarios, muy pronto comenzaron a cambiar la forma en que el propio sistema operativo funcionaba. Implementaron el manejo de memoria virtual y programaron el soporte para los protocolos del Arpanet que luego se convertiría en el conocido Internet. Todos estos cambios eran distribuidos como BSD a quienes tenían una licencia de Unix de la división de BTL encargada de administrar este sistema (AT&T).



A mediados de los años ochentas, **Richard Stallman**, entonces en el Instituto Tecnológico de Massachussets (**MIT**) decidió dedicarse a la construcción de lo que denominó **software libre**. El razonamiento de Stallman era que los mayores progresos en la industria del software surgen cuando se coopera entre programadores. Según Stallman, las industrias de la época estaban atentando contra la libertad de los usuarios y programadores de compartir el software, así que decidió programar un sistema parecido a Unix y regalarlo. A este sistema le llamó **GNU**, un acrónimo recursivo que significa Gnu's Not Unix (GNU no es Unix).

Para este entonces, varias compañías estaban ya en el negocio de Unix. Microsoft distribuía Xenix, una versión de Unix para el procesador 80386. Sun Microsystems utilizaba BSD como base para su SunOS. Digital Equipment Corporation incluía otro sistema operativo basado en BSD con sus workstations, Ultrix.

A principios de los años noventa, la Universidad de California en Berkeley había cambiado tanto Unix que decidieron distribuirlo incluso a personas que no tuvieran una licencia de AT&T. Esto motivó que algunas personas crearán una empresa para distribuir BSD (como ahora le llamaban al Unix de Berkeley) comercialmente. A esta empresa le llamaron Berkeley Software Design, Inc. (BSDI).

Para este entonces, y como resultado de la finalización del juicio antimonopolio, **BTL** ya había sido desmembrado en varias compañías, una de ellas AT&T que de acuerdo a los nuevos arreglos legales ya podía comercializar Unix. En efecto, AT&T había creado una división llamada Unix System Laboratories (USL) con el propósito de vender Unix a la industria. USL había incorporado el código de BSD en el Unix de AT&T además de haber desarrollado algunos otros subsistemas. Cuando AT&T se enteró que BSDI estaba compitiendo en este mercado lanzó una demanda legal contra BSDI.

BSDI compareció ante el Juez e indicó que su producto no estaba basado en el Unix de AT&T sino en el Unix de Berkeley, que al final de cuentas la Universidad regalaba sin necesidad de una licencia.

Al darse cuenta AT&T que no tenía oportunidad de ganar una demanda contra BSDI decidió demandar a la Universidad de California en Berkeley por distribuir código basado en el Unix de AT&T a personas que no poseían una licencia de AT&T. La Universidad de California a su vez, demandó a AT&T por incorporar código desarrollado en la Universidad sin dar el crédito debido a los autores.

Además de la iniciativa comercial de BSDI, habían surgido ya las primeras distribuciones gratuitas del Unix de Berkeley. Bill Jolitz había traducido BSD al procesador Intel 80386 y le había llamado 386BSD. El grupo NetBSD había sido fundado para continuar traduciendo BSD a nuevas plataformas. **FreeBSD** había tomado el código de 386BSD que Bill Jolitz no tenía tiempo de mantener y estaba ahora distribuyéndolo gratis para el Intel 80386. Lamentablemente, a raíz de la demanda de AT&T a Berkeley, todas estas distribuciones fueron detenidas hasta que se arreglaran los aspectos legales de si BSD era en efecto un sistema operativo que no necesitaba licencia de AT&T.



Para las personas deseosas de correr Unix en las ahora populares PCs, quedaba únicamente una alternativa legal, **Minix**. Minix era un sistema operativo parecido a Unix desarrollado por el Profesor Andrew Tanenbaum para enseñar a sus alumnos el diseño de un sistema operativo. Sin embargo, debido al enfoque puramente educacional de Minix, Tanenbaum no permitía que este fuera modificado y el sistema se encontraba muy limitado en muchísimos aspectos.

Un estudiante de Finlandia, **Linus Torvalds**, al ver que no era posible extender Minix, decidió escribir su propio sistema operativo **compatible con Unix**. Miles de personas que querían correr Unix en sus PCs vieron aquí su única alternativa debido a que a Minix le faltaban demasiadas cosas y BSD, a pesar de tener toda la funcionalidad esperada, tenía problemas legales. El proyecto GNU que Stallman había iniciado hacía ya casi diez años había producido para este entonces un sistema casi completo a excepción del kernel, que es el programa que controla el hardware de la máquina. Torvalds decidió utilizar el casi completo sistema GNU y completarlo él mismo con su propio kernel, al resultado le llamó **Linux**.

Extracto del grupo de noticias comp.os.minix de Usenet.

Mensaje enviado por **Linus Torvalds**

Fecha: 25 Aug 91 20:57:08 GMT

Hola a todos allá afuera usando Minix -

Estoy haciendo un sistema operativo libre (es solo un hobby, no será grande ni profesional como GNU) pero funciona en 386(486) AT clones, lo he estado cocinando desde abril y está quedando listo. Quisiera alguna retroalimentación de las cosas que a ustedes les gustan y no les gustan de MINIX, ya que mi sistema operativo se parece (la misma disposición física (debido a razones prácticas) entre otras cosas).

Ya he portado bash (1.08) y gcc (1.40) y parece que las cosas funcionan. Esto implica que tendré algo usable en unos meses, y quiero saber que es lo que a la gente le gustaría que

Richard Stallman insiste aún que el sistema debiera ser llamado **GNU/Linux**, ya que incluye más código del proyecto GNU que del proyecto Linux. En la actualidad, el proyecto GNU esta desarrollando su propio Kernel (El Hurd) para poder distribuir un GNU sin tener que utilizar el kernel de Linux.

A mediados de los años noventa, AT&T vendió Unix a Novell, quién tomó como prioridad número uno resolver las demandas. El acuerdo fue que la Universidad de California eliminaría todo el código residual de AT&T y lanzaría una última distribución de BSD totalmente libre de problemas de licenciamiento. Esta distribución fue el 4.4-BSD Lite2. Quien quisiera distribuir BSD debería basar su distribución en 4.4-BSD Lite2 para no tener problemas legales. Inmediatamente los distribuidores de BSD reiniciaron sus labores de distribución migrando lentamente sus sistemas al 4.4-BSD Lite2.

Hoy en día, existen varias distribuciones del BSD así como existen varias distribuciones de Linux. Algunos grupos que distribuyen BSD son:

- **FreeBSD**: el énfasis de este sistema operativo está en la facilidad de uso del sistema. Entre otras metas están la eficiencia del uso de recursos y el rendimiento del sistema. Únicamente soportan las plataformas i386 y Alpha aunque ya han iniciado traducciones a la arquitectura SPARC. Recientemente se han anunciado planes de integrar el código fuente de FreeBSD con el de BSD/OS distribuido por BSDI. Esto proporcionaría una ventaja tecnológica aún más grande al grupo de FreeBSD.



Gracias a su modo de emulación de Linux, puede correr una variedad de software escrito para este último.

- ▶ **NetBSD:** el énfasis de este grupo es la portabilidad del sistema operativo. Actualmente existen sistemas NetBSD para casi cualquier plataforma, desde las antiguas VAX hasta las modernas iMac.
- ▶ **OpenBSD:** el énfasis de este grupo es en la seguridad, han hecho una auditoria de todo el código fuente buscando errores y fallas de seguridad. Incorporan sistemas criptográficos en su sistema operativo.
- ▶ **TrustedBSD:** este es un muy reciente sistema operativo, basado en FreeBSD y con la meta de alcanzar la certificación B1 del libro naranja del Departamento de Defensa de Estados Unidos.

Existen numerosas distribuciones Linux (también conocidas como "distros"), ensambladas por individuos, empresas y otros organismos. Cada distribución puede incluir cualquier número de software adicional, incluyendo software que facilite la instalación del sistema. La base del software incluido con cada distribución incluye el núcleo Linux, al que suelen adicionarse también varios paquetes de software.

Las herramientas que suelen incluirse en las distribuciones de este sistema operativo se obtienen de diversas fuentes, incluyendo de manera importante proyectos de código abierto o libre, como el GNU y el BSD. Debido a que las herramientas que en primera instancia volvieron funcional al núcleo de Linux provienen de un proyecto anterior a Linux, Richard Stallman (fundador del proyecto GNU) pide a los usuarios que se refieran a dicho sistema como GNU/Linux. A pesar de esto, la mayoría de los usuarios continúan llamando al sistema simplemente "Linux" y las razones expuestas por Richard Stallman son eterno motivo de discusión. La mayoría de los sistemas Linux incluyen también herramientas procedentes de BSD.

Usualmente se utiliza la plataforma XFree86 o la Xorg para sostener interfaces gráficas (esta última es un fork de Xfree86, surgido a raíz del cambio de licencia que este proyecto sufrió en la versión 4.4 y que lo hacía incompatible con la GPL).

Los sistemas Linux funcionan sobre más de 20 diferentes plataformas de hardware, entre ellas las más comunes son las de los sistemas compatibles con PC, computadoras Macintosh, procesadores PowerPC, Sparc y MIPS.

Algunos ejemplos de distribuciones de Linux son:

Debian

- **Debian** o más concretamente Debian GNU/Linux es una distribución Linux, basa sus principios y fin en el software libre. Creado por Debian Project el año 1993, la organización responsable de la creación y mantenimiento de la misma distribución, centrado en GNU/Linux y utilidades GNU. También mantienen y desarrollan otros sistemas operativos GNU basados en los núcleos the Hurd, llamado Debian GNU/Hurd, todavía en estado embrionario y NetBSD, llamado Debian GNU/NetBSD.

Debian nace como una apuesta por separar en sus versiones de software el software libre del software propietario. El modelo de desarrollo es independiente a empresas, creado por los propios usuarios, sin depender de ninguna manera de necesidades comerciales. Debian no vende directamente su software, lo pone a disposición de cualquiera en Internet, aunque sí permite a personas o empresas distribuir comercialmente este software mientras se respeta su licencia.

El origen de la palabra Debian viene determinado por el nombre del creador, Ian Murdock, y Deborah, su esposa.

Algunas de sus características principales son:

- ▶ Disponibilidad en varias plataformas hardware. La versión 3.0 es compatible con 11 plataformas.
- ▶ Una amplia colección de software disponible. La versión 3.0 viene con más de 8710 paquetes.
- ▶ Un grupo de herramientas para facilitar el proceso de instalación y actualización del software
- ▶ Su compromiso con los principios y valores involucrados en el movimiento del Software Libre.
- ▶ No tiene marcado ningún entorno gráfico en especial ya sea GNOME, KDE...

La última versión lanzada de Debian se denomina (actualmente) Sarge. Debian es una distribución sobre la que se han generado una gran cantidad de distribuciones propias. Algunas de ellas son:

- ▶ Augustux: live CD, KDE
- ▶ Catux-USB: USB
- ▶ Gnoppix: live CD, GNOME
- ▶ Guadalinex: live CD, GNOME
- ▶ Knoppix: live CD, KDE
- ▶ Kanotix: live CD, KDE
- ▶ Linex: live CD, GNOME
- ▶ Linspire: comercial, KDE
- ▶ MEPIS Linux: live CD, KDE
- ▶ Progeny Debian: comercial
- ▶ SkoleLinux: live CD, KDE
- ▶ Ubuntu: GNOME, KDE

- ▶ UserLinux: live CD, GNOME
- ▶ Xandros: comercial, KDE

Ubuntu

Ubuntu es una distribución de Linux de tipo escritorio, basada en Debian y patrocinada por Canonical Ltda, empresa cuyo dueño es el sudafricano Mark Shuttleworth.

Ubuntu debe su nombre al movimiento homónimo encabezado por el obispo Desmond Tutu, quien ganó el Premio Nobel de la Paz en 1984 por su lucha en contra del Apartheid en Sudáfrica. Mark Shuttleworth, el mecenas del proyecto, es sudafricano y por lo tanto se encontraba muy familiarizado con la corriente.

Tras ver similitudes entre los ideales de los proyectos GNU, Debian y en general con el movimiento del software libre, decidió aprovechar la ocasión para difundir los ideales de Ubuntu.

El 8 de julio de 2004, Mark Shuttleworth y la empresa Canonical Ltda anunciaron la creación de la distribución Ubuntu. Ésta tuvo un financiamiento inicial de 10 millones de dólares (US\$). El proyecto nació por iniciativa de algunos programadores de los proyectos Debian, Gnome y Arch que se encontraban decepcionados con la manera de operar del proyecto Debian, la distribución Linux sin ánimo de lucro más popular del mundo.

De acuerdo con sus fundadores, Debian era un proyecto demasiado burocrático donde no existían responsabilidades definidas y donde cualquier propuesta interesante se ahogaba en un mar de discusiones. Asimismo, Debian no ponía énfasis en estabilizar el desarrollo de sus versiones de prueba y sólo proporcionaba auditorías de seguridad a su versión estable, la cual era utilizada sólo por una minoría debido a la poca o nula vigencia que poseía en términos de la tecnología Linux actual.

Tras formar un grupo multidisciplinario, los programadores decidieron buscar el apoyo económico de Mark Shuttleworth, un emprendedor sudafricano que tras fundar la compañía Thawte en la cochera de su domicilio, logró venderla cuatro años después a la empresa VeriSign por 575 millones de dólares estadounidenses.

Shuttleworth vio con simpatía el proyecto y decidió convertirlo en una iniciativa autosostenible, combinando su experiencia en la creación de nuevas empresas con el talento y la experiencia de los programadores de la plataforma Linux. De esta forma nació la empresa Canonical, la cual se encarga de sostener económicamente el proyecto mediante la comercialización de servicios y soporte técnico a otras empresas. Mientras los programadores armaban el sistema, Shuttleworth aprovechó la ocasión para aplicar una pequeña campaña de mercadotecnia para despertar interés en "la distribución-sin-nombre" (en inglés: the no-name-distro).

Tras varios meses de trabajo y un breve período de pruebas, la primera versión de Ubuntu (Warty Warthog) fue lanzada el 20 de octubre de 2004.

Las principales características de Ubuntu son:

- ▶ Basada en la distribución Debian.



- ▶ Disponible en 3 arquitecturas: Intel x86, AMD64, PowerPC.
- ▶ Los desarrolladores de Ubuntu se basan en gran medida en el trabajo de las comunidades de Debian y GNOME.
- ▶ Las versiones estables se liberan cada 6 meses y se mantienen actualizadas en materia de seguridad hasta 18 meses después de su lanzamiento.
- ▶ El escritorio oficial es Gnome y se sincronizan con sus liberaciones. (Existe otro proyecto conocido como Kubuntu que usa el escritorio KDE).
- ▶ Para centrarse en solucionar raudamente los bugs, conflictos de paquetes, etc. se decidió eliminar ciertos paquetes del componente main, ya que no son populares o simplemente se escogieron de forma arbitraria por gusto o sus bases de apoyo al software libre. Por tales motivos inicialmente KDE no se encontraba con más soporte de lo que entregaban los mantenedores de Debian en sus repositorios, razón por la que se sumó la comunidad de KDE distribuyendo una distribución llamada Kubuntu.
- ▶ El navegador web oficial es Mozilla Firefox.
- ▶ El sistema incluye funciones avanzadas de seguridad y entre sus políticas se encuentra el no activar procesos latentes por omisión al momento de instalarse. Por eso mismo, no hay un firewall predeterminado, ya que no existen servicios que puedan atentar a la seguridad del sistema.
- ▶ Para labores/tareas administrativas incluye una herramienta llamada sudo (similar al Mac OS X), con la que se evita el uso del usuario root.
- ▶ Mejorar la accesibilidad y la internacionalización, de modo que el software esté disponible para tanta gente como sea posible. En la versión 5.04, el UTF-8 es la codificación de caracteres por defecto. (Sin embargo, existe un bug en las versiones actuales que dificulta el uso de tildes en los terminales de texto).
- ▶ No sólo tiene como lazo a Debian el uso del mismo formato de paquetes deb, Ubuntu tiene uniones muy fuertes con esa comunidad, contribuyendo cualquier cambio directamente e inmediatamente, más que anunciándolos. Esto sucede en los tiempos de lanzamiento. Muchos de los desarrolladores de Ubuntu son también responsables de los paquetes importantes dentro de la distribución de Debian.
- ▶ Todos los lanzamientos de Ubuntu se proporcionan sin costo alguno. Los CDs de la distribución se envían de forma gratuita a cualquier persona que los solicite. También es posible descargar las imágenes ISO de los discos por transferencia directa o bajo la tecnología Bittorrent.
- ▶ Ubuntu no cobra honorarios por la suscripción de las mejoras de la "Edición Enterprise".

Ubuntu divide todo el software en cuatro secciones, llamadas los "componentes", para reflejar diferencias en licencias y la prioridad con la que se atienden los problemas que informen los usuarios. Por defecto, se instala una selección de paquetes que cubre las necesidades básicas de la mayoría de los usuarios de computadoras.

- ▶ Main. El componente main contiene solamente los paquetes que cumplen los requisitos de la licencia de Ubuntu, y para los que hay soporte disponible por parte de su equipo. Éste está pensado para que incluya todo lo necesario para la mayoría de los sistemas Linux de uso general.

Los paquetes de este componente poseen ayuda técnica garantizada y mejoras de seguridad oportunas.

- ▶ **Restricted.** El componente restricted contiene el software que está soportado por los desarrolladores de Ubuntu debido a su importancia, pero que no está disponible bajo ningún tipo de licencia libre para incluir en main. En este lugar se incluyen los paquetes tales como los controladores propietarios de algunas tarjetas gráficas, como por ejemplo, los de nVIDIA. El nivel de la ayuda es más limitado que para main, puesto que los desarrolladores pueden no tener acceso al código fuente.
- ▶ **Universe.** El componente universe contiene una amplia gama del software, que puede o no tener una licencia restrictiva, pero que no recibe apoyo por parte del equipo de Ubuntu. Esto permite que los usuarios instalen toda clase de programas en el sistema, pero los guarda en un lugar a parte de los paquetes soportados: main y restricted.
- ▶ **Multiverse.** Finalmente, se encuentra el componente multiverse, que contiene los paquetes sin soporte debido a que no cumplen los requisitos del Software Libre.

Los paquetes de Ubuntu generalmente se basan en los paquetes de la rama inestable (Sid) de Debian.

Una nueva versión de Ubuntu es lanzada cada seis meses, y cada lanzamiento tiene un "nombre" y un número de versión. El número de versión está basado en la fecha de lanzamiento, por lo tanto, por ejemplo, la versión 5.04 fue lanzada en abril (04) de 2005 (5). Debajo hay una lista de las versiones previas y lanzamientos planeados.

Versión	Fecha de lanzamiento	Nombre
4.10	20 de Octubre, 2004	The Warty Warthog
5.04	8 de Abril, 2005	The Hoary Hedgehog
5.10	13 de Octubre, 2005	The Breezy Badger
6.04	(Abril de 2006, version planeada)	The Dapper Drake

En el futuro, habrá un lanzamiento llamado "Grumpy Groundhog" que quedará permanentemente en versión "inestable" solo para propósitos de pruebas.

Gentoo

Gentoo Linux es una distribución GNU/Linux orientada a usuarios con cierta experiencia en este sistema operativo. El nombre Gentoo proviene del nombre en inglés del pingüino de pico rojo, la mascota de Linux es un pingüino.

Una ventaja de Gentoo es que las versiones de software se actualizan de forma continua, a diferencia de otras distribuciones donde los paquetes pasan meses en comprobación. Ello permite tener un sistema con las últimas versiones de todo el software, ideal para tareas de escritorio. Por contra, el uso a veces de versiones del software insuficientemente comprobadas da como resultado bugs que pueden suponer un riesgo para servidores de producción.



Otra desventaja de este sistema es que poner en marcha un sistema completo, o actualizar un sistema que ha estado desatendido durante una temporada puede requerir una respetable cantidad de tiempo (horas o incluso días), mientras se descargan y compilan todos los paquetes nuevos. Aun así, Gentoo permite por regla general una actualización sin problemas, a diferencia de otras distribuciones donde puede llegar a resultar complicado o casi imposible. Esta actualización también es posible partir de binarios precompilados en la instalación, lo que requiere menos tiempo.

Red Hat



Red Hat Linux es una distribución Linux creada por Red Hat, la cual fue una de las más populares en los entornos de usuarios hogareños. Es una de las distribuciones Linux de "mediana edad". La versión 1.0 fue presentada el 3 de noviembre de 1994. No es tan antigua como la distribución Slackware, pero ciertamente es más antigua que muchas otras. Fue la primera distribución que usó RPM como su formato de paquete, y en un cierto plazo ha servido como el punto de partida para varias otras distribuciones, tales como la orientada hacia PCs de escritorio Mandrake Linux (originalmente Red Hat Linux con KDE), Yellow Dog Linux la cual se inició desde Red Hat Linux con soporte para PowerPC y ASPLinux (Red Hat Linux con mejor soporte para caracteres no-Latinos).

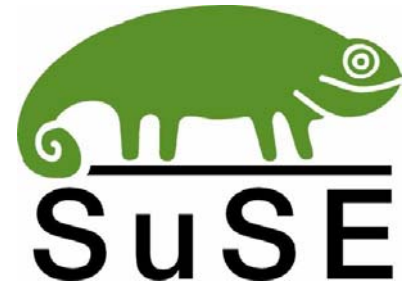
Desde el 2003, Red Hat ha desplazado su enfoque hacia el mercado de los negocios con la distribución Red Hat Enterprise Linux. Red Hat Linux 9, la versión final, llegó oficialmente al final de su vida útil el pasado 30 de abril de 2004, aunque el proyecto Fedora Legacy continua publicando actualizaciones.

Suse

SuSE es una de las principales distribuciones GNU/Linux existentes a nivel mundial, y su centro de producción está ubicado en Alemania.

Entre las principales virtudes de esta distribución se encuentra el que sea una de las más sencillas de instalar y administrar, ya que cuenta con varios asistentes gráficos para completar diversas tareas.

Su nombre "SuSE" es el acrónimo del alemán "Software- und Systementwicklung", el cual formaba parte del nombre original de la compañía y que se podría traducir como "desarrollo de software y sistemas". El nombre actual de la compañía es SUSE LINUX, habiendo perdido el primer término su significado (al menos oficialmente).



SuSE incluye un programa único de instalación y administración llamado YaST2 que permite realizar actualizaciones, configurar la red y el cortafuegos, administrar a los usuarios, y muchas más opciones todas ellas integradas en una sola interfaz. Además incluye varios escritorios, entre ellos los más conocidos que son GNOME y KDE, siendo este último el escritorio por omisión. La distribución incorpora las herramientas necesarias para redistribuir el espacio de tu disco duro permitiendo así la coexistencia con otros sistemas operativos existentes en el mismo.

Usa sistemas de paquetes RPM (RedHat package manager) aunque no guarda relación con esta distribución.

El 4 de noviembre de 2003, la compañía multinacional estadounidense Novell anunció que iba a comprar SUSE LINUX. La adquisición se llevó a cabo en enero de 2004.

Guadalinex

Guadalinex es una distribución Linux promovida por la Junta de Andalucía para fomentar el uso del software libre dentro de Andalucía. Está inspirada en un proyecto similar de la Junta de Extremadura: linex

Existen varias versiones de Guadalinex:

- ▶ La versión 1.0 usa como base Debian SID
- ▶ La versión anterior, la 2004 se basaba en Debian SARGE
- ▶ La versión 3.0 -disponible desde febrero del 2006- se basa en la distribución Ubuntu.
- ▶ Se basa en el esquema de Metadistros de Hispalinux.
- ▶ El entorno de escritorio principal es GNOME aunque incorpora también Fluxbox.



Justo en el momento de escribir estos apuntes (Febrero 2006) ha aparecido la versión 3.0 de Guadalinex. Se encuentra disponible la versión estable de Guadalinex en sus dos presentaciones: "live", que se inicia desde CD y plenamente funcional sin instalar nada en el PC y "no live" destinada a los equipos que presentan dificultades con la "live" y a los que prefieren una instalación más clásica.

Para conseguir Guadalinex V 3.0 puedes dirigirte a:

- ▶ http://www.guadalinex.org/descargador/index.php?nombre=guadalinex_v3_0_live.iso
- ▶ http://www.guadalinex.org/descargador/index.php?nombre=guadalinex_v3_0_nolive.iso

Estos archivos también están disponibles en la red P2P BitTorrent:

- ▶ CD Live: http://www.cica.es/torrents/guadalinex_v3.0_live.iso.torrent
- ▶ CD No Live: http://www.cica.es/torrents/guadalinex_v3.0_nolive.iso.torrent

Las principales mejoras introducidas en esta versión V 3.0 son:

- ▶ Instalador mejorado con procedimientos automatizados para los escenarios más frecuentes y versátil como siempre con la opción de particionado manual.
- ▶ Opción (por defecto) de partición separada para "/home", mejorando la seguridad y facilitando la actualización.
- ▶ Detección de dispositivos extraíbles conectados en caliente muy mejorada gracias a Hermes.
- ▶ Kernel 2.6.12 con un mejor soporte para los equipos y dispositivos recientes
- ▶ Escritorio Gnome 2.12

- ▶ OpenOffice.org 2.0, paquete ofimático muy mejorado y ahora con aplicación de base de datos.
- ▶ Evolution 2.4 cliente de correo y organizador personal.
- ▶ Últimas versiones de programas para Internet.
- ▶ Asistentes para configuración de acceso por red ethernet, wifi, módem y ADSL
- ▶ Navegador Mozilla Firefox
- ▶ Gaim y Amsn para mensajería instantánea
- ▶ Clientes para redes P2P
- ▶ Lectores de titulares RSS y de "podcast"
- ▶ Receptor de radio por Internet
- ▶ Aplicaciones gráficas actualizadas
- ▶ Reproductores multimedia
- ▶ Grabador de CD y DVD

Todo lo demás que decidas instalar desde Internet, ahora más fácil con la preselección de aplicaciones útiles que presenta la opción guiada "Añadir programas".

(Toda esta información esta sacada de www.guadalinex.org, desde donde directamente se pueden solicitar que te envíen gratuitamente los CDs a casa).

1.2 Software libre. Licencias GPL y BSD.

La licencia de un software nos indica a que nos comprometemos cuando adquirimos o conseguimos dicho software. En el caso del "software propietario" la licencia nos impide distribuir el software, instalarlo en más de una maquina, modificarlo de ninguna manera, etc.

En los sistemas operativos conocidos como "software libre" que hemos referenciado en el anterior punto, se usan dos tipos distintos de licencia, las licencias GPL y BSD. Los puntos principales de estas licencias consisten en la necesidad de distribuir el software junto con los fuentes del mismo, la posibilidad que tiene el usuario para realizar todas las copias que quiera y distribuir las, y el derecho que se le da al usuario para que modifique cualquier programa a su antojo.

La licencia del tipo BSD es la que utilizan todos los productos desarrollados o basados por la **BSD** (Berkeley Software Distribution). Sin embargo, todos los productos desarrollados o basados en GNU utilizan una licencia desarrollada por la FSF denominada **GPL**.

No vamos a entrar a explicar en este artículo detalladamente todas las diferencias entre ambas licencias, tan solo vamos a centrarnos en un punto que tiene la GPL y que no comparte la licencia ni la comunidad BSD. Es el conocido "virus de la GPL", es decir, la obligación de que todo programa derivado de un software GPL debe estar así mismo bajo la GPL. Esta obligación no existe en la licencia BSD y por lo tanto muchos programa propietarios contienen código BSD que se paga con un mero párrafo en la documentación del mismo.

Esto hace que muchas grandes compañías desarrollen software propietario que se basa en parte en software distribuido bajo la licencia BSD. Sin embargo, si una compañía utiliza aunque solo sea una pequeña parte de código GPL esta forzada a distribuir su aplicación bajo la licencia GPL.

Software libre es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre suele estar disponible gratuitamente en Internet, o a precio del coste de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente.

Análogamente, el software gratuito (denominado usualmente Freeware) incluye en algunas ocasiones el código fuente; sin embargo, este tipo de software no es libre en el mismo sentido que el software libre, al menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

No debemos confundir software libre con software de dominio público. Este último es aquel por el que no es necesario solicitar ninguna licencia y cuyos derechos de explotación son para toda la humanidad porque pertenece a todos por igual. Cualquiera puede hacer uso de él, siempre con fines legales y consignando su autoría original. Este software sería aquel cuyo autor lo dona a la humanidad o cuyos derechos de autor han expirado tras transcurrir 70 años de la muerte de su autor. Si un autor condiciona su uso bajo una licencia, por muy débil que sea, ya no es dominio público. En resumen, el software de dominio público es la pura definición de la libertad de usufructo de una propiedad intelectual que tiene la humanidad porque así lo ha decidido su autor o la ley tras 70 años de la muerte de éste.

Entre los años 60 y 70 del Siglo XX, el software no era considerado un producto sino un añadido que los vendedores de los grandes computadores de la época (los mainframes) aportaban a sus clientes para que éstos pudieran usarlos. En dicha cultura, era común que los programadores y desarrolladores de software compartieran libremente sus programas unos con otros. Este comportamiento era particularmente habitual en algunos de los mayores grupos de usuarios de la época, como DECUS (grupo de usuarios de computadoras DEC). A finales de los 70, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia.

En 1984, Richard Stallman comenzó a trabajar en el proyecto GNU, y un año más tarde fundó la Free Software Foundation (FSF). Stallman introdujo una definición para free software y el concepto de "copyleft", el cual desarrolló para dar a los usuarios libertad y para restringir las posibilidades de apropiación del software

De acuerdo con tal definición, el software es "libre" si garantiza las siguientes libertades:

- ▶ "libertad 0", ejecutar el programa con cualquier propósito (privado, educativo, público, comercial, etc.)
- ▶ "libertad 1", estudiar y modificar el programa (para lo cuál es necesario poder acceder al código fuente)
- ▶ "libertad 2", copiar el programa de manera que se pueda ayudar al vecino o a cualquiera
- ▶ "libertad 3", mejorar el programa, y hacer públicas las mejoras, de forma que se beneficie toda la comunidad.

Es importante señalar que las libertades 1 y 3 obligan a que se tenga acceso al código fuente. La "libertad 2" hace referencia a la libertad de modificar y redistribuir el software libremente licenciado bajo algún tipo de licencia de software libre que beneficie a la comunidad.

1.3 Instalación de Linux.

La instalación de Linux es muy parecida a la que hemos visto en temas anteriores para otros sistemas operativos. La mejor manera de instalar Linux es arrancar directamente la máquina con el CD de Linux o con el disquete de instalación de Linux en caso de que no podamos arrancar desde el CD.

Un problema que nos vamos a encontrar, es que no existe una "instalación normal" de Linux. Cada distribución de Linux dispone de su instalador propio, siendo distintos los instaladores de SuSe y de Red Hat, por poner un ejemplo. Incluso varían los instaladores de una versión a otra de la misma distribución. Aunque los conceptos básicos de la instalación siempre serán los mismos, la forma de trabajar varia.

En general, veremos que los pasos básicos de la instalación serán mas o menos estos:

1. Arrancar la máquina con el CD de Linux.
2. Detección del hardware básico para la instalación (CD, HD, teclado, pantalla, ratón, tarjeta de red, etc.)
3. Elegir el tipo de instalación (normalmente, siempre es conveniente escoger personalizada, manual o experta).
4. Crear una partición para instalar en ella Linux, y seleccionar dicha partición.
5. Montar la partición creada en un punto de montaje.
6. Crear una partición para utilizarla como swap.
7. Escoger los paquetes que queremos instalar.
8. Preparar un disquete de arranque de Linux, para poder arrancar el sistema en caso de que no funcione el arranque desde HD.
9. Copiar archivos.
10. Reiniciar la máquina para ejecutar ya nuestro Linux.

En estos pasos, he hablado de dos puntos que aun no hemos visto; puntos de montaje y partición swap.

Puntos de montaje.

Normalmente estamos acostumbrados a la forma en que los sistemas operativos de Microsoft denominan a los medios de almacenamiento secundarios, asignando una letra a cada volumen, de modo que la disquetera es la A: la primera partición del primer disco duro es la C:, la siguiente la D:, etc. En Linux todo esto cambia.

En primer lugar, veamos como Linux referencia a los propios discos duros. Así, el primer disco duro de nuestra máquina en Linux se conoce como `/dev/hda`.

► `/` nos indica el root o raíz del árbol de Linux (En Linux solo existe un árbol)

- ▶ **dev** nos indica el directorio donde se almacenan todos los dispositivos (devices)
- ▶ **/hda** nos indica que nos referimos al Hard Disk (hd) con la letra a.

/dev/hda - Dispositivo maestro en la IDE 1.

/dev/hdb - Dispositivo esclavo en la IDE 1.

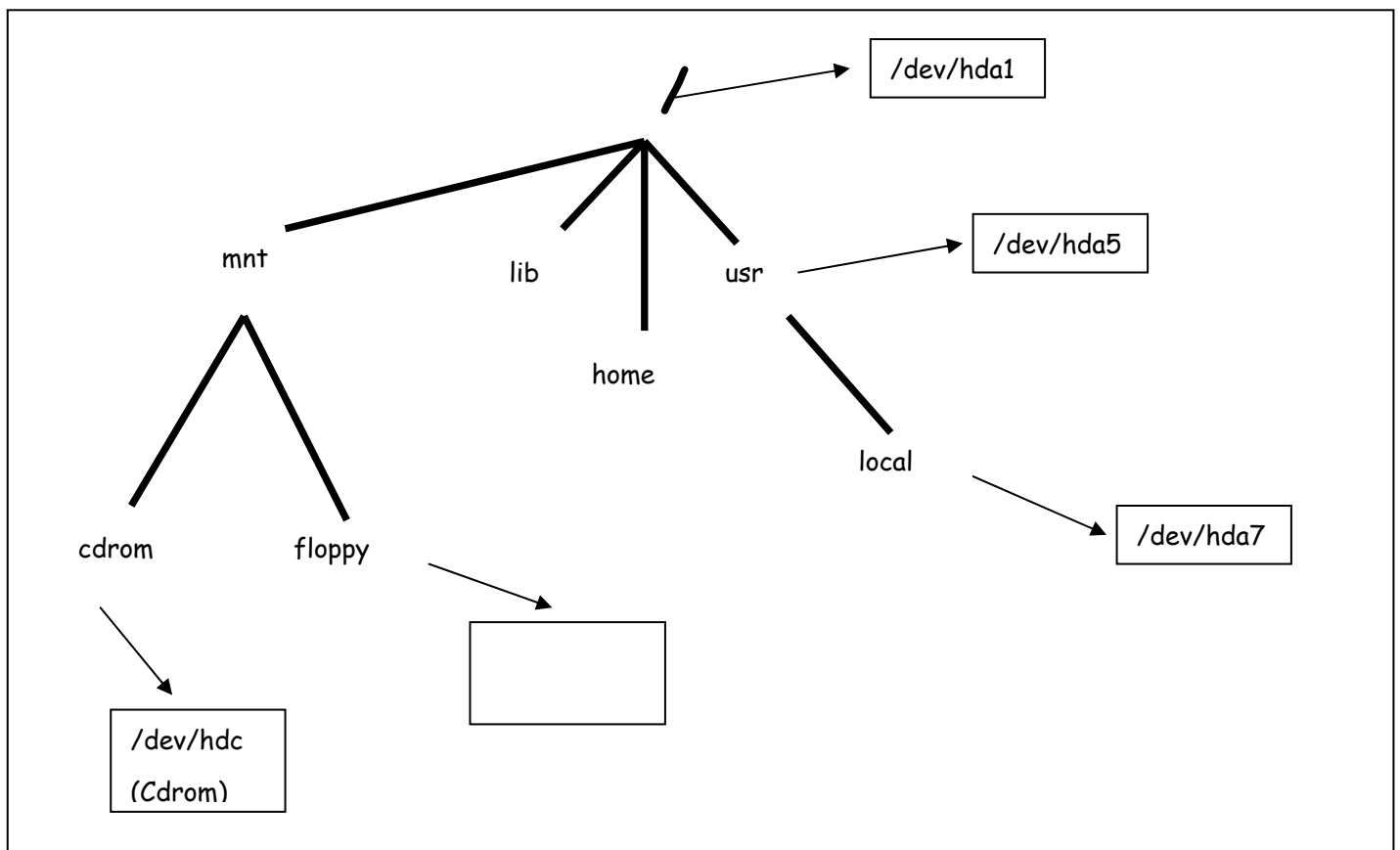
/dev/hdc - Dispositivo maestro en la IDE 2.

/dev/hdd - Dispositivo esclavo en la IDE 2.

Si nuestro disco duro en lugar de ser IDE es SCSI, veríamos que en vez de ser nombrado /hda seria nombrado /sda.

Cuando reverenciamos particiones, se utiliza un número a continuación del nombre del disco duro. Este número representa la partición. Así, /dev/hda2 nos indica que nos referimos a la segunda partición del disco duro maestro de IDE 1. Como en un disco duro no pueden existir más de cuatro particiones primarias, estas reciben números del 1 al 4. Si creamos una partición extendida, esta no recibe ningún numero (igual que en Windows no se le asigna una letra) y a las unidades lógicas que se crean dentro de dicha partición extendida se le asignan números a partir del 5. Veamos algunos ejemplos.

/dev/hdb1 - Primera partición primaria del disco duro 2 (esclavo en el 1º canal IDE).



/dev/hda5 - Primera unidad lógica del disco duro 1 (maestro en el 1º canal IDE).

Como comentamos anteriormente, Linux no utiliza "letras" para acceder a las particiones que creamos, así que ¿Cómo podemos grabar un archivo por ejemplo en /dev/hdb6? En Windows nos limitaríamos a grabar el archivo en D: o E: o la letra que asignemos a esa partición, pero esto no se hace así en Linux.

En Linux, cada dispositivo de almacenamiento (partición, disquete, CD) que utilicemos debe ser montado en un punto de montaje. En Linux solo existe un espacio de almacenamiento, un único árbol que empieza en el raíz (root) y que contiene todo lo que tenemos en nuestro sistema. Esto se consigue asociando cada partición a un directorio mediante un proceso denominado montaje. Montar una partición hace que su espacio de almacenamiento se encuentre disponible accediendo al directorio especificado (conocido como punto de montaje).

Por ejemplo, si montamos la partición /dev/hda5 en /usr, significa que todos los ficheros y directorios a partir de /usr residen físicamente en /dev/hda5. Por lo tanto, el fichero /usr/doc/FAQ/txt/Linux-FAQ estará almacenado en /dev/hda5, cosa que no ocurre con el fichero /etc/X11/gdm/Sessions/Gnome.

Continuando con nuestro ejemplo, también es posible que uno o más directorios dentro de /usr sean puntos de montaje de otras particiones. Por ejemplo, se puede montar una partición (por ejemplo, /dev/hda7) en /usr/local, lo que significa que, por ejemplo, /usr/local/man/whatis estará en /dev/hda7, no en /dev/hda5.

(Como hemos visto en los ejemplos, en Linux no se utiliza la contrabarra (\) para indicar directorios, sino la barra normal (/).

Es absolutamente obligatorio montar al menos el root o raíz (/) durante la instalación.

Partición Swap.

Vimos cuando tratamos los temas sobre conceptos de Sistemas Operativos, que existía una técnica conocida como paginación de memoria, que nos permitía ofrecer a los programas más memoria de la que existe físicamente en la maquina. En Windows esta técnica utiliza un archivo de intercambio que es gestionado normalmente por Windows, se suele llamar algo como pagefile.sys y tener un tamaño de muchos megas. Linux no crea ningún archivo de intercambio, sino que utiliza una partición entera para este fin, conocida como partición Swap. Simplemente tenemos que crearla en la instalación de Linux, y el sistema se encarga de usarla, sin tener nosotros que montarla ni nada por el estilo.

El tamaño que se le suele dar a una partición Swap, es el doble de la memoria RAM que tengamos instalado en nuestro sistema, aunque esta cifra es simplemente una recomendación. (Lo que si se recomienda es utilizar siempre Swap, a menos que tengamos cantidades de memoria RAM cercanas al Gigabyte).

1.4 Carga de Linux.

Otro que reviste mucha importancia si se tiene otro sistema operativo previamente instalado, es la forma básica en que se carga un sistema operativo Linux. Para ello la vía más simple es un pequeño programa llamado LILO (LIinux LOader) que permite cargar tanto Linux como cualquier otro sistema operativo instalado en la misma computadora.

La parte inicial del LILO se puede ubicar en dos zonas diferentes del disco duro:

1. En el MBR (Master Boot Record) del primer disco duro. En esta forma el LILO tomará el control una vez finalizada la ejecución del programa del BIOS, siendo el encargado de iniciar la carga de todos los sistemas operativos instalados en la máquina. (Atención. Si en la instalación escogemos esta opción, eliminaremos cualquier gestor de arranque que este instalado ya en el MBR de nuestra maquina, como puede ser el gestor de arranque de Windows XP o de Windows 2003).
2. En el sector de arranque de la partición raíz de Linux o en el de la partición /boot de existir esta. En este caso no será el LILO el programa inicial de carga sino otro que lo llame como puede ser el cargador de Windows XP o Windows 2003. Solo se recomienda esta forma cuando existen otros sistemas operativos instalados previamente como Windows XP o 2003 y se desea conservar el arranque mediante el programa que estos proveen.

Cuando se utilice el sector de arranque de una partición para colocar al LILO se deberá crear un disquete de carga del sistema durante la instalación, pues sino será imposible acceder al mismo sin alguna herramienta de recuperación.

En la actualidad, el gestor de arranque LILO esta siendo substituido a marchas forzadas por otro gestor mucho mas potente: GRUB (GRand Unified Bootloader), que veremos en un tema posterior.

Configuración del arranque dual.

No existe actualmente ningún problema en instalar un sistema operativo Linux en un disco duro donde exista ya instalado un sistema operativo Windows. Tanto Lilo como GRUB son perfectamente capaces de arrancar estos sistemas Windows. Dedicaremos un tema a ver como configuramos tanto Lilo como sobre todo GRUB.

1.5 Sistema de Ficheros en Linux.

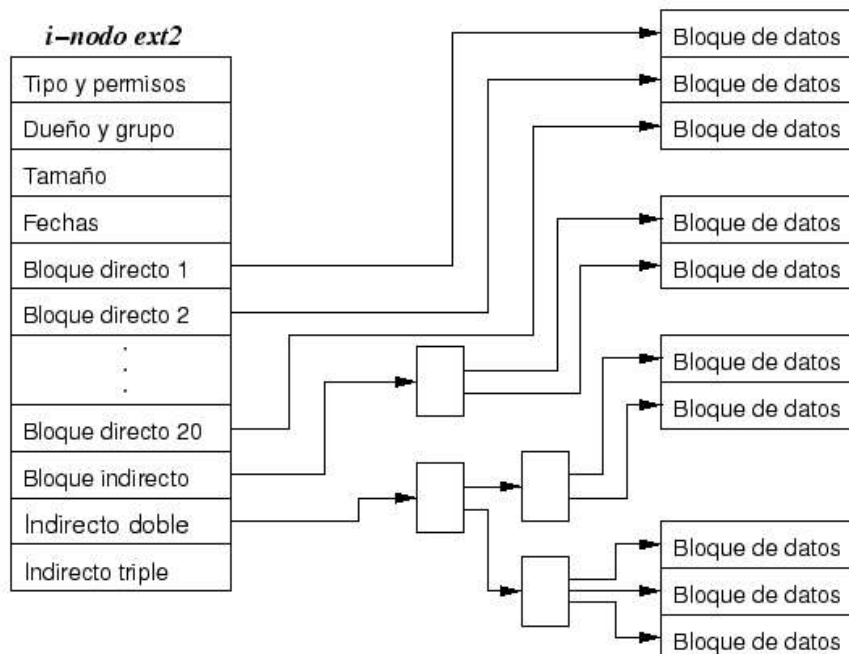
Una de las características más importantes de Linux es que soporta una gran cantidad de sistemas de ficheros diferentes. Esto le permite convivir e interactuar con otros sistemas operativos. Ejemplos de sistemas de ficheros soportados en Linux son: ext, ext2, ext3, reiserfs, msdos, vfat, xia, minix, umsdos, proc, smb, ncp, iso9660, sysv, hpfs, affs y ufs. El sistema de ficheros propio de Linux es el conocido como **ext2** y más recientemente, el **ext3** (en realidad, ext3 es exactamente igual que ext2, pero se le incorporan técnicas de journaling).

Físicamente un fichero no se almacena de forma continua en los distintos dispositivos de almacenamiento. Aunque los sistemas de ficheros de Linux tienden a la continuidad física en la medida que se empleen más frecuentemente siempre existe la posibilidad de que el contenido de un fichero se fragmente en pequeños bloques de longitud fija no continuos en el disco. Debido a esto es necesaria la existencia de una estructura adicional que direcciona estos bloques para cada fichero. Esta estructura en Unix se conoce como i-nodo y almacena, entre otras características, las direcciones físicas de los bloques que constituyen un fichero. Cada i-nodo ocupa un área fija (4k),

pequeña y continua, y se identifica a través de un número entero. Su estructura está formada por los siguientes campos:

- ▶ El tipo del fichero y sus permisos.
- ▶ El identificador del usuario dueño y del grupo del fichero.
- ▶ La cantidad de bytes del fichero.
- ▶ Los tiempos de: último acceso al fichero (lectura o ejecución), última modificación del fichero (escritura), última modificación del i-nodo.
- ▶ Las direcciones de los 20 bloques directos del fichero, del indirecto de primer nivel, del indirecto de segundo nivel y del indirecto de tercer nivel.

En la figura se representa el esquema de un i-nodo.



El sistema de ficheros de Linux de forma lógica posee una estructura arbórea cuya raíz es única y se representa por / que además constituye el carácter separador en el camino (Path) de cualquier recurso del sistema. En la figura se muestra un esquema general de los principales directorios presentes en el sistema de ficheros de Linux.

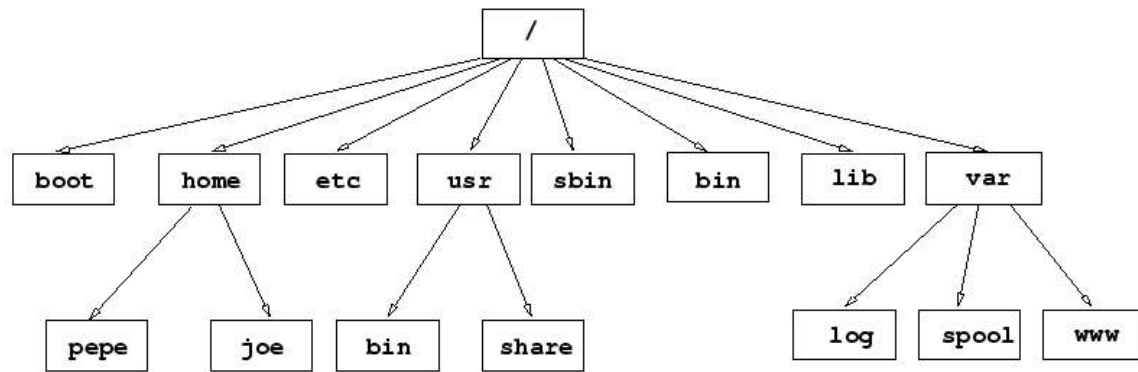


Figura: Estructura lógica en el sistema de ficheros de Linux

Para nombrar los ficheros es posible emplear una cantidad no mayor de 256 caracteres que pueden ser de cualquier tipo con excepción de /, por ser el separador de caminos (Path). De todos modos es conveniente no incluir caracteres especiales que son interpretados por el shell de otra manera. No existe el concepto de extensión asociada a un fichero, aunque en muchos casos se utiliza para dar una idea acerca de su contenido.

Linux es un sistema "case sensitive" (distingue mayúsculas y minúsculas), o sea no es lo mismo el fichero ``document.ps'' que ``DocumenT.ps''. Los ficheros ``ocultos'' en Linux son aquellos cuyo nombre comienza con el carácter punto. (.)

Al igual que en los sistemas de ficheros de Microsoft, en todos los directorios existen dos ficheros ocultos especiales nombrados . y .. el primero se refiere al directorio actual y el segundo, al directorio padre.

Esta estructura de directorios viene fijada en el momento de la instalación de nuestro Linux, y debemos respetarla. Cada directorio tienen una función concreta, y los programas (paquetes) que instalemos en nuestro sistema usaran dicha estructura. Así, cuando en Windows instalamos un programa lo normal es que nos pregunte el directorio donde queremos instalarlo, sin embargo veremos que en Linux no es así. Automáticamente el programa instalara sus archivos de documentos en el directorio /usr/share/doc por ejemplo, sus paginas de manual en /usr/share/man, su fichero ejecutable en /usr/sbin, sus ficheros de configuración en /etc/ y sus ficheros logs en /var. (Estas rutas son de ejemplo, cada programa se instala en diversos directorios que no tienen por que ser estos).

Esta forma de trabajar resulta muy funcional cuando te acostumbras, pero choca al principio a los usuarios de Windows. También implica que en ningún caso debemos borrar directorios, modificar sus permisos, etc. La única rama donde podemos trabajar sin problemas es /home/nombre de usuario.