

## Tabla de Contenidos

GNU/LINUX y sus DISTRIBUCIONES:.....	2
Una ENORME cantidad de posibilidades.....	4
El particionado del disco.....	6
Particiones Físicas.....	6
Montado de dispositivos de disco.....	7
Identificación de los discos desde /dev.....	7
Acceso a los discos mediante el montaje.....	8
Distribución y estructura lógica del file system.....	9
File Systems.....	9
Directorios importantes del Sistema Operativo GNU/LINUX.....	10
Configuraciones principales y troubleshooting.....	11
TroubleShooting a partir de los archivos de log.....	13
Archivos importantes:.....	13
Niveles de ejecución del sistema operativo.....	13
Entendiendo los conceptos del entorno gráfico.....	14
Servidor X.....	15
Entorno de escritorio.....	15
Display manager.....	16
Redirección de Servidor X:.....	16
Instalación de aplicaciones.....	16
Manejadores de paquetes.....	16
Nuevas versiones de apt-get.....	17
Uso correcto de aptitude o apt-get.....	17
Uso de apt-get o aptitude para actualizar todo el sistema operativo.....	18
Archivo de configuración de apt-get o aptitude.....	18
Uso especial de aptitude o apt-get para realizar un pasaje de una distribución completa a una más nueva.....	19
Uso de entorno gráfico para instalar y configurar aplicaciones.....	19

## GNU/LINUX y sus DISTRIBUCIONES:

Una Distribución no es más que el agrupamiento junto al Kernel de LINUX de un conjunto de aplicaciones de acuerdo al objetivo y al uso para el que se ha armado la distribución.

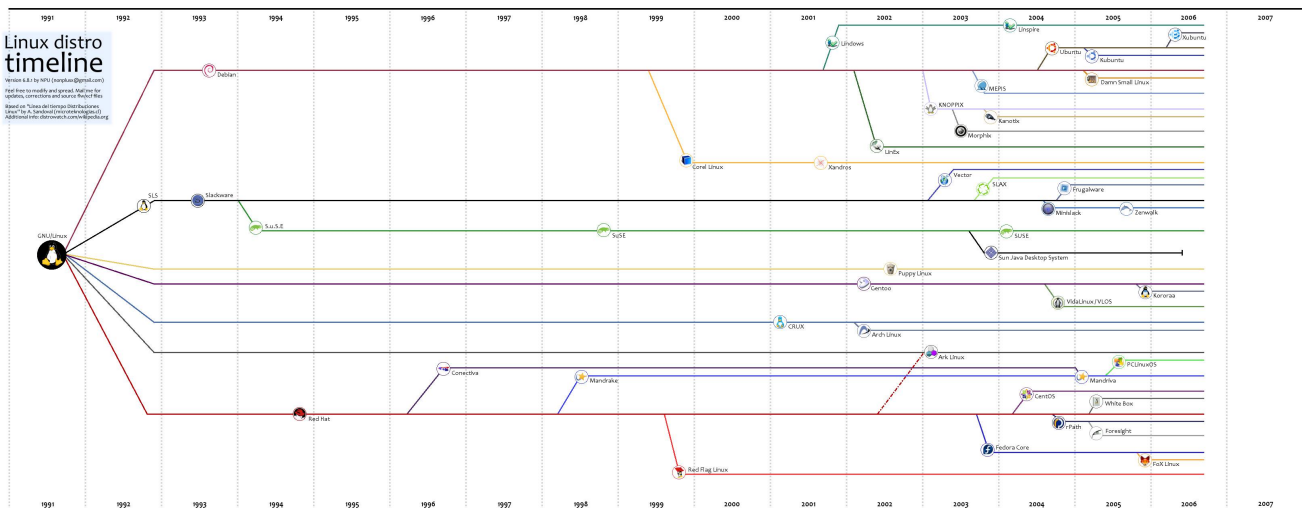
Las primeras distribuciones de las cuales surgieron muchas más son:

- Debian
- Slackware
- RedHat

UN poco más tardías y también originales son

- GENTO
- CRUX
- Puppe.

En el siguiente gráfico puede verse la línea de tiempo de las más principales y sus derivadas:



Como importante para nosotros es simplemente conocer en líneas generales cuales son las distribuciones más recomendables para diversos usos. Todas ellas son muy buenas y pueden utilizarse para todos los usos, pero algunas nos facilitan o vienen más orientadas a ciertos fines.

### DEBIAN:

Debian es una de las más antiguas, con un muy fuerte compromiso por el Software Libre, y un número muy importante de adeptos sobre todo en Europa. Es una de las distribuciones que tiene más aplicaciones disponibles y empaquetadas, así como también una de las revolucionarias en cuanto a la comodidad y facilidad de instalación. Por estas ventajas, ha sido fuente de gran cantidad de distribuciones, una de las cuales estaremos probando hoy en nuestro taller: Ubuntu.

Debian es una opción muy sólida y fuertemente recomendada para servidores. Sin embargo puede que no tenga los instaladores más “user friendly” que existen. En debian cada versión está fuertemente cuidada, y tenemos tres ediciones:

- stable (etch)
- testing (lenny)

- unstable (sid).

La diferencia entre estas tres no es más que el tiempo de prueba que llevan. Debian suele ser conocida como la “tortuga” de las distribuciones, en tanto no es demasiado rápido el pasaje de las aplicaciones a las versiones estables. Esto es por el terrible cuidado que pone en el testing de las mismas, para asegurar y garantizar su uso en servidores y aplicaciones críticas. Por ese motivo es posible que la versión “Estable” no tenga las últimas versiones de todas las aplicaciones, y lo que es más importante es posible que no tenga drivers para hardware muy reciente. La versión “Estable” es muy conservadora. Sin embargo en “Testing” tenemos una edición muy estable (mucho más estable que un windows por supuesto), que es super óptima para uso en Desktops, Notebooks y demás. Si bien las últimas versiones de drivers y aplicaciones no tienen un testing tan exhaustivo como “Estable” es lo suficientemente estable para un usuario final. De todas maneras no se recomienda pasar a “Testing” un servidor crítico productivo. La versión “Inestable” está solo indicada para usuarios con mucho conocimiento que desean probar antes que nadie las nuevas versiones de drivers y aplicaciones. Sin embargo, en tanto se trata de una versión de “desarrollo” pueden existir ciertas dificultades, especialmente en cuanto a las dependencias de librerías y demás por lo que no se recomienda para un usuario que no sea experto. Como ven, una vez más, aquí somos LIBRES de elegir como queremos usar nuestro OS.

Una de las principales ventajas de DEBIAN es su revolucionario sistema de paquetes de instalación de aplicaciones. Aún antes que Internet tuviera la extensión que tiene hoy en día, debian diseñó un manejador de paquetes que es perfectamente capaz de determinar todas las dependencias de una aplicación, y sin ninguna acción manual, encontrar las versiones correspondientes en Internet, bajarlas e instalarlas. Realmente si utilizan esta distribución, es muy difícil luego acostumbrarse a otra por la gran facilidad y eficiencia que este manejador de paquetes posee. Ya lo veremos más detenidamente.

### **MANDRIVA**

Mandriva es una de las primeras que intentó acercar GNU/LINUX al usuario final de equipos Desktop. Fue pensada desde el comienzo como una distribución para usuarios sin demasiado conocimiento técnico. El nombre original era Mandrake, pero por un tema de licencias tuvo que cambiar por Mandriva.

### **REDHAT**

RedHat fue la gran distribución de una época, sobre todo en los Estados Unidos. Acumuló una gran cantidad de fanáticos hasta que se formó la empresa REDHAT, con el objetivo de armar una distribución paga, especialmente diseñada para corporaciones y con una importante estructura de servicios de soporte. Tiene el mérito de permitir introducir LINUX en las corporaciones que siempre requieren soporte corporativo, por más que muchas veces el soporte libre de la red suele ser mucho más prolífico que el corporativo de unos pocos empleados pagos. Sin embargo al convertirse en una de las primeras versiones pagas conocidas a nivel mundial, perdió gran cantidad de adeptos que adscribían a la filosofía del software no solo libre y abierto, sino también gratuito. Sin embargo, si por un lado cerró y empaquetó una distribución para corporaciones “RedHAt Advance Server” que ya va por la versión 3, mantuvo el compromiso de sostener el desarrollo libre de una versión alternativa: Fedora.

El manejo de paquetes de RedHat, a pesar de no ser tan práctico como el de debian posee gran cantidad de seguidores: RPM

### **SLACKWARE**

Es una distribución de las más antiguas, mundialmente conocida por su solidez y estabilidad para servidores. Sin embargo es una de las más duras y la que fue más resistente a la inclusión de aplicaciones gráficas de configuración e instalación. Es una distribución recomendada especialmente para personas que tienen un alto conocimiento de GNU/LINUX. Es extremadamente sólida y está

preparada para ser una de las más sólidas distribuciones existentes. Tiene también su propio manejador de paquetes y la pequeña desventaja que suele ser un poco más difícil de configurar que las anteriores.

## **UBUNTU**

Merece una mención especial. A ella se aplican todas las cuestiones dichas más arriba sobre DEBIAN, excepto el manejo de ediciones que es levemente distinto. Sin embargo utiliza el mismo tipo de manejador de paquetes que DEBIAN e incluso es posible en ella usar los repositorios originales de DEBIAN. La mención especial se refiere fundamentalmente que surge como una distribución sostenida por un multimillonario, que se ha puesto como objetivo y se encuentra invirtiendo lo bastante como para que el Ideal Hacker de la información para las masas sea posible. El objetivo que se ha puesto la distribución es tomar una distribución sólida y práctica como DEBIAN pero ajustarla y acercarla mucho más al usuario final que su predecesora. El compromiso que mantiene con DEBIAN es que las mejoras introducidas por sus desarrolladores son puestas inmediatamente a consideración de los desarrolladores de DEBIAN los que si pasa los test de calidad correspondientes y revisiones pueden ser incorporadas inmediatamente en sus repositorios. De esta manera, la gran cantidad de personas pagas por este multimillonario para mejorar DEBIAN de alguna manera le retribuyen a su distribución madre el haberles proporcionados una distribución muy avanzada para comenzar a orientarla hacia nuevos fines. Hoy es una de las distribuciones más bajadas de Internet. De hecho en su sitio hasta se pueden pedir directamente los Cds, que son enviados de modo gratuito a quien lo solicite.

La única diferencia entre UBUNTU, KUBUNTU y XUBUNTU es el entorno gráfico que instala por defecto. Ya hablaremos de ello.

## **KNOPPIX:**

También merece una mención especial ya que, basada en Debian, fue una de las primeras distribuciones que introdujo las tecnologías suficientes como para que pudiera ser ejecutada directamente desde un CD sin tocar nada del equipo en el que corre. La tecnología de compresión/descompresión en memoria de las aplicaciones hizo que en un mínimo CD se pudieran incluir un número increíblemente grande de aplicaciones que permite con solo bootear de CD tener un equipo completamente operativo, con entorno de escritorio, servidor, web, servidor de Apache, Office, y un sinnúmero de aplicaciones. En otras palabras, un LINUX completamente utilizable sin tocar un solo byte del disco.

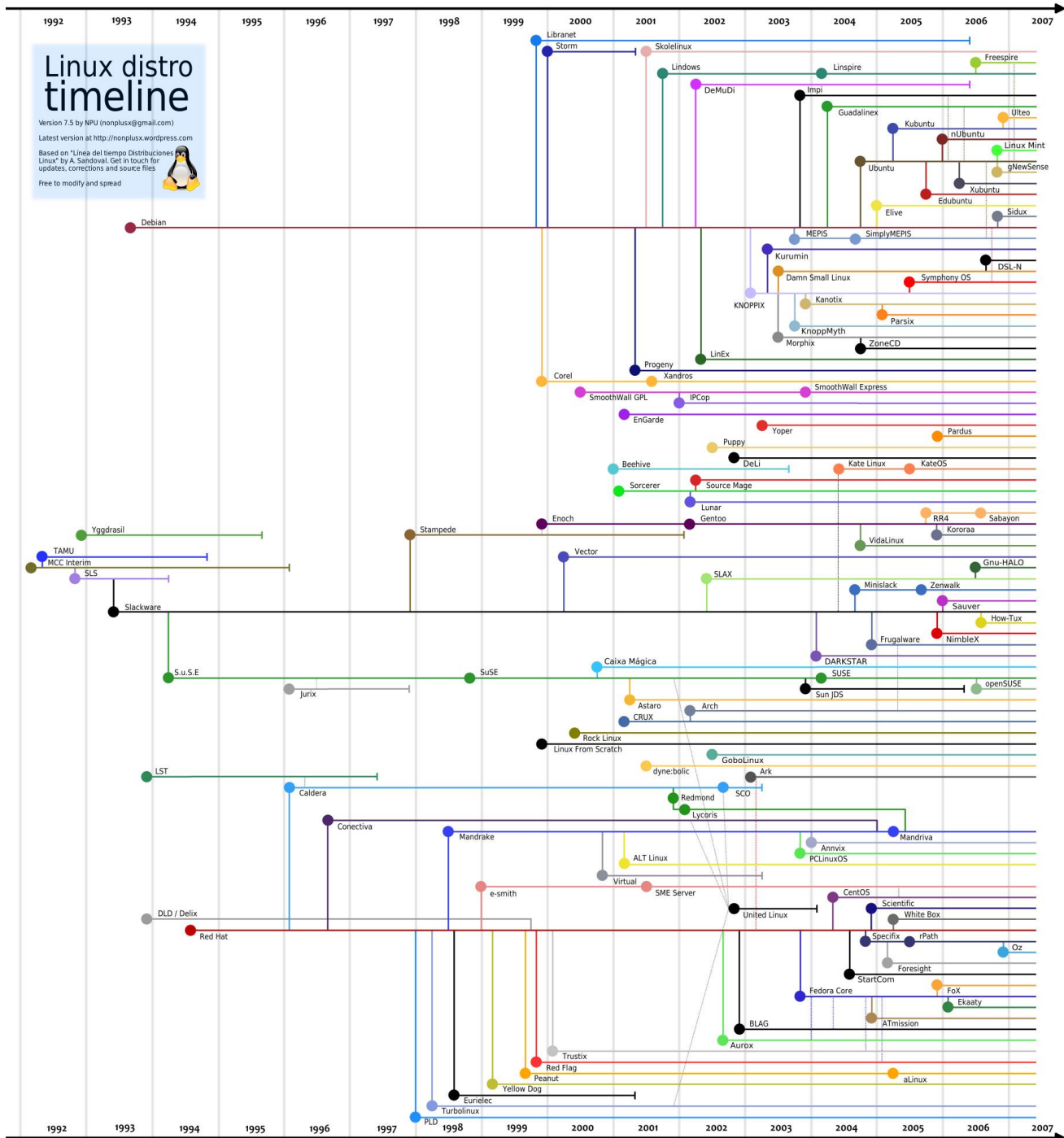
Esto permitió luego el uso de estas tecnologías por Slackware por ejemplo cuando sacó su primera versión de SLAX, que era un Slackware “live”, que además permitía cambiar, y guardar configuraciones en un disco removible o la misma Internet para poder levantar el OS desde cualquier máquina y que recuperar las configuraciones de la última vez que se utilizó. Asimismo SLAX tiene un esquema de módulos que permite hacer un Live-Cd a medida de acuerdo a requerimientos especiales, lo que abrió las puertas a un sinnúmero de aplicaciones realizada con la convergencia de todas estas tecnologías.

## **Damm Small LINUX (DSL LINUX)**

Basada en Debian con un kernel 2.4, con la ventaja de ser muy chiquita (64 MB) por lo que fue una de las favoritas para incluir en los primeros pen Drive que no eran tan grandes como actualmente.

## **Una ENORME cantidad de posibilidades**

Solo he mencionado unas pocas, ya que como se ve en el siguiente gráfico, son muchísimas más las disponibles, pero no podría abarcarlas a todas. Solo para dar cuenta del enorme crecimiento que han tenido se agrega como referencia el siguiente gráfico:



Tenemos distribuciones para un número enorme de usos, desde orientadas a seguridad, a recuperación de equipos (Windows o LINUX) luego de un desastre, a edición de música y video, a educación, a determinadas ciencias, live-cds que corren centrales telefónicas, distribuciones muy seguras para routers (IPCop / Coyote) , , etc, etc, etc....

### IPCop

Cabe mencionarla ya que es una distribución excelente y muy óptima para funcionar como router y firewall. Es muy sencilla de instalar y posee una interfase WEB que permite configurar opciones muy avanzadas de ruteo, filtro de paquetes, auditoria, log, y protección de red.

## **Coyote**

Es una muy linda distribución que entra en un simple diskette de 3 1/2, usa tecnología de compresión y se prepara el diskette solo con los drivers de red necesarios. Este diskette se coloca en una PC que ni siquiera necesita disco y le sobra con 128 Mb de memoria, y al bootear se convierte en un poderoso y práctico router-firewall, con interfase gráfica de administración y todo.

Podríamos estar varios días enumerando distribuciones y sus ventajas, y mientras las describimos ya surgirían nuevas que no conoceríamos. Para estar siempre al tanto de las últimas distribuciones y versiones que van saliendo:

<http://distrowatch.com/>

Incluso ofrece servicios de RSS que podemos incorporar sin problemas en nuestro desktop para estar siempre informados.

## **El particionado del disco**

En LINUX el particionado de disco es completamente flexible y cambiabile en cualquier momento. Desde el comienzo los sistemas operativos UNIX han tenido en mente que los discos se llenan, las necesidades cambian, y se necesita un modo flexible de poder ir cambiando discos sin necesidad de reinstalar todo el equipo. GNU/LINUX no es la excepción a esto.

A diferencia de Windows, en Linux no existen las famosas “Unidades de disco” (C: D:, etc). El concepto de unidad de disco es bastante incómodo y poco funcional cuando hablamos de equipamientos serios y de misión crítica.

En cualquier OS Unix tendremos un arbol de directorios que comienzan por el directorio raíz: “/” y por supuesto es extensible sin límites teórico-prácticos de niveles.

La diferencia fundamental y que lo hace tan flexible a la hora de cambiar discos o particiones, es que en cualquier “Directorio Vacío” podemos montar un disco entero o una partición. Para el usuario final es completamente transparente si en una carpeta se encuentra montado un disco o si la misma es una subcarpeta de una carpeta de nivel superior.

El líneas generales, en GNU/LINUX necesitamos al menos dos particiones (una de ellas es opcional pero altamente recomendable), una para la realización del swapping de memoria a disco si el OS se queda sin memoria, y otra para los archivos y el Sistema Operativo.

## **Particiones Físicas**

Sin embargo es recomendable armar más particiones que estas por un tema de orden y comodidad. En síntesis, una configuración típica recomendada es:

1. Partición de swap (invisible para el usuario final): Es recomendable que sea la primera partición del disco si es posible, y que tenga el doble de tamaño de la cantidad máxima de memoria que le vamos a colocar al equipo. En esta partición el OS baja temporariamente porciones de memoria si se queda sin ella en medio de la operatoria. (Esto es lo que realiza Windows en el archivo de swapping, que por el mal diseño que tiene Windows suele competir con el resto de los archivos del sistema operativo, fragmentando y estreizando el disco). Por supuesto como UNIX está pensado en serio, la partición de swapping, que tiene una actividad elevada si el equipo cuenta con poca memoria, es completamente independiente del resto de los archivos, con lo cual no nos estaría afectando la performance de la manera que lo hace el archivo de swapping de Windows, fragmentándonos cada día un poco más nuestro disco. Puede instalarse LINUX sin partición de Swap si el equipo tiene suficiente memoria pero no suele ser lo recomendado.

2. Partición Raíz (/): Es la partición en donde solemos colgar todos los archivos del sistema operativo, los archivos de booteo y las configuraciones de nuestro OS. No debería ser una partición utilizable para los usuarios finales. No deberían incluirse en esta partición archivos de usuario. En una instalación pequeña puede compartirse con todos los archivos de las aplicaciones (usualmente en /usr). Esta partición debería poder ser reformateada en el caso en que deseemos cambiar de Sistema Operativo o Distribución sin alterar ningún archivo de usuario.
3. Partición de Usuario (Usualmente /home) :En esta partición solo deberían colocarse archivos de usuario. Esto incluye por supuesto todos los documentos de trabajo del usuario, aplicaciones que no requieran ser utilizadas por otros usuarios del mismo equipo, y las configuraciones específicas de cada usuario. Por un tema de orden dentro de la carpeta /home, se crean tantas carpetas como usuarios utilicen el equipo, separando de esa manera los archivos de unos y otros. Cuando instalamos el OS Debian nos suele preguntar si queremos compartir archivos entre usuarios. Si respondemos por la afirmativa, todos los usuarios podrán leer los archivos de todos los demás, pero solo podrán modificar los propios o a los que se le haya dado permisos específicos. Si respondemos por la negativa, ningún usuario sin permisos adicionales podrá ver los archivos de otros usuarios.

Este particionado es el recomendable para una instalación pequeña o una instalación de una Notebook o un Desktop. En instalaciones más serias o críticas se suelen multiplicar las particiones utilizadas, incluso muchas veces se utilizan particiones externas, provenientes de algún equipo servidor de storage.

## Montado de dispositivos de disco.

Como mencionamos más arriba, el acceso a los datos de los discos lo debemos hacer luego de haber “montado” el dispositivo en un directorio libre del sistema operativo. (si lo montamos en un directorio ocupado, los archivos de ese directorio quedan ocultos). Por montado se entiende conectar un dispositivo de hardware accedido por /dev... a un directorio cualquiera utilizable por el usuario del árbol normal de directorios. ¿Cuándo debemos montar un disco?:

- En primer lugar las particiones en las que tenemos instalado nuestro sistema operativo deben estar montadas.
- Cuando insertamos un pen drive.
- Cuando introducimos un CD.
- Cuando necesitamos aumentar el espacio en disco a partir de cierto directorio, podemos pasar los archivos previamente a una partición nueva montada en cualquier lado, y luego montamos esa partición nueva en el directorio original. No es más que esto lo que hay que hacer en un UNIX si queremos ampliar nuestro espacio en disco a partir de un directorio en particular.

## Identificación de los discos desde /dev.

**PEN DRIVE:** se lo puede acceder desde los siguientes devices, según la instalación de hardware sobre la que estamos trabajando:

- /dev/sda1: Es el utilizado en la mayoría de los motherboards
- /dev/sdb1: Si por ejemplo nuestro disco principal es un Serial Ata, el dispositivo /dev/sda... estará reservado a él, pasando a /dev/sdb... o /dev/sdc... Según sea el caso.

**DISCO RIGIDO IDE:**se lo puede acceder desde los siguientes devices:

**/dev/hda1** (2 – 3 – 4) : Primera-(segunda-tercera-cuarta) partición, primer disco, primera controladora.

**/dev/hdc1** (2 – 3 - 4): Primera-(segunda-tercera-cuarta) partición, primer disco, segunda controladora.  
Como se ve: “hd” significa Hard Disk Ide, “a,b y c,d” son los primero y segundo discos de cada una de las controladoras, y “1,2,3,4” identifica la partición de cada disco.

Un disco IDE puede tener 4 particiones primarias, o 3 primarias y una extendida en que pueden configurarse particiones lógicas. GNU/LINUX es full compatible con el uso de particiones lógicas pero lo recomendable es usar siempre particiones primarias.

### **DISCO RIGIDO SCSI / SATA:**

Los discos SATA o SCSI tienen una nomenclatura similar:

**/dev/sda1** (2 – 3 – 4) identifica al primer disco que se levanta dinámicamente de acuerdo a las configuraciones del motherboard. **/dev/sdb1** el segundo, etc...

De esta manera también se identifican los sucesivos pen drive que puedan conectarse al equipo. El primer pen se levanta en el primer dispositivo scsi libre. Si solo tenemos discos ide, el primer pen drive será **/dev/sda1** (ya que normalmente los pen están formateados en una sola partición.), el segundo **/dev/sdb1**, etc....

Si alguno de los dispositivos scsi está siendo utilizado por el disco SATA entonces el primer pen drive será el primero que quede libre; **/dev/sdc /dev/sdd.....**

Al igual que para el caso de los ide, “sd” identifica Scsi Disk, “a,b,c,d”, primero, segundo tercer. Disco, “1,2,3,4...” particiones.

### **CDROM,**

- si se utiliza un kernel con emulación SCSI (modo antiguo de acceder a los CDROMS o DVDROMS), será el primer **/dev/sd....** libre.
- Si se usa un kernel moderno normalmente se identificará como un disco IDE más, con la nomenclatura **/dev/hd...**

### **Acceso a los discos mediante el montaje.**

Montar un disco es asociar el dispositivo correspondiente (descriptos en el apartado anterior) a una carpeta cualquiera del árbol de directorios de LINUX. Es importante que la carpeta no contenga datos, porque si los contiene, los mismos quedan ocultos hasta tanto se desmonte el dispositivo.

Normalmente tenemos un archivo: `/etc/fstab` que tiene cargados que dispositivos queremos montar automáticamente. Es muy importante tener mucho cuidado al editar este archivo ya que si nos equivocamos, puede ser que el equipo no bootee porque no tiene montadas las particiones mínimas requeridas (“/” y “/boot” si está aparte).

En este archivo también seteamos la posibilidad de que un usuario cualquiera pueda montar un pen drive o un cdrom. Siempre que accedemos a cualquier dispositivo “de bloque” como ser discos, cds, discos removibles, etc, debemos hacerlo a partir del punto de montaje.

El comando para montar una partición de disco es:

*Disco rígido formateado en ext3:*

```
mount -t ext3 /dev/hda1 /home
```

**-t** indica el tipo de archivo, **/dev/hda1** indica la primera partición del primer disco de la primera controladora, **/home** es el directorio en donde la estamos montando.



### **Disco rígido formateado en ntfs (Windows):**

```
mount -t ntfs /dev/hdc3 /media/discowindows
```

-t indica el tipo de archivo, /dev/hdc3 indica la tercera partición del primer disco de la segunda controladora, /media/discowindows es el directorio que hemos creado para montarla

### **Pen Drive formateado en VFAT :**

```
mount -t vfat /dev/sda1 /media/pendrive
```

-t indica el tipo de archivo, /dev/hdc3 indica la tercera partición del primer disco de la segunda controladora, /media/pendrive es el directorio que hemos creado para montarla.

### **Pen Drive formateado en VFAT :**

```
mount -t vfat /dev/sda1 /media/pendrive
```

-t indica el tipo de archivo, /dev/hdc1 indica la primera partición del primer disco Scsi, /media/pendrive es el directorio que hemos creado para montarla.

### **CDROM o DVD de datos:**

```
mount -t iso9660 /dev/hdc /cdrom0
```

-t indica el tipo de archivo, /dev/hdc indica el primer disco de la segunda controladora (iso9660 no arma por defecto particiones separadas, por eso no tiene número), /cdrom0 es el directorio que hemos creado para montarlo. Para el caso de los discos de música, el acceso es directo a través del /dev, por lo tanto no es necesario montarlo. Esto se debe al tipo de acceso que la extracción de sonido digital requiere.

### **Aclaraciones adicionales:**

Muchas veces los entornos gráficos, de acuerdo a la distribución y la instalación son capaces de reconocer cuando un dispositivo removible se conecta al equipo y lo montan automáticamente, luego de lo cual nos abren una ventana con los archivos del mismo. En este caso (es el caso de ubuntu por ejemplo) el usuario ni se entera que se tuvo que montar el dispositivo.

Normalmente el montado de dispositivos nuevos solo lo puede hacer el usuario root. Si el dispositivo se encuentra registrado en el archivo /etc/fstab y tiene la opción "user" el mismo puede ser montado por cualquier usuario del sistema. Este es el modo en que frecuentemente los instaladores del sistema operativo dejan configurados la unidad de cdrom y a veces el primer dispositivo scsi para los pendrives.

Para mayor información acerca de como montar una partición o configurarla en el archivo /etc/fstab puede tipearse en una consola:

```
man mount
```

```
man fstab
```

Todos los comandos de GNU/LINUX y los principales archivos de configuración tienen su correspondiente "man" manual en donde se pueden profundizar todas las opciones que posee. Algunos poseen además su página "info" invocada con

```
info fstab o info mount
```

## **Distribución y estructura lógica del file system**

### **File Systems**

Cada partición física que configuremos, deberá ser formateada con algún file system. En el caso de la

partición de swapping, el filesystem es uno especialmente diseñado para este uso, más rápido y menos complejo que el resto, ya que solo se usa para almacenamiento temporario.

El resto de las particiones podemos tenerlas formateadas o formatearlas en diversos File System, que van desde los de Windows (NTFS 4, NTFS 5, FAT, FAT32) hasta opciones más específicas de los entornos UNIX: ext2, reiserfs, ext3, hpfs, etc.

El formato más común y recomendado para todas las particiones que deben ser leídas únicamente por LINUX es el ext3. El ext3 puede ser montado como ext2 a los fines de recovery. La diferencia es que cuando está montado como ext3, tiene una protección conocida como “journaling” que permite recuperar a un estado consistente a un disco si en medio de una escritura importante se corta la alimentación. En un server productivo esto es deseable que no ocurra nunca ya que deberían estar protegidos por UPS, sin embargo para un desktop hogareño puede pasar que se corte la energía y alguna escritura quede inconsistente. Ext3 es una tecnología que me permite recuperar al FS de esas inconsistencias.

## **Directorios importantes del Sistema Operativo GNU/LINUX**

Cualquiera sean los FS que utilicemos en nuestras particiones de disco, necesariamente para acceder a la info almacenada en ellos deberemos “montar” el FS en la estructura de árbol de acuerdo a lo que mencionamos más arriba.

En GNU/LINUX somos libre de hacer lo que queramos con nuestro OS, pero al mismo tiempo la gente de Linux es RACIONAL; por lo tanto harán las cosas de acuerdo a criterios que permitan entender de manera ordenada lo que tenemos almacenado en nuestro equipo.

**/boot:** Es el directorio en donde se encuentra la imagen del kernel que estamos utilizando y todo lo necesario para cargarla, como el disco virtual que generan en memoria algunas instalaciones al arranque para determinar los dispositivos del equipo dinámicamente. En síntesis, de los archivos que tenemos en esta carpeta dependen tanto la carga del sistema operativo como la carga del boot.loader (Ver más abajo).

**/etc:** Es el directorio en donde guardamos todas las configuraciones a nivel de sistema para las aplicaciones instaladas y el sistema operativo mismo. (Las configuraciones de aplicaciones a nivel usuario se guardan en el home de cada usuario). Esta carpeta es la que de manera tan desastrosa pretende reemplazar la registry de Windows. Para el caso de cualquier UNIX, las configuraciones se guardan en archivos de texto solo modificables por el SuperUsuario root del equipo. Estos archivos son fácilmente accesibles para recuperación ante desastres que no sean rotura de disco. Se guardan en este directorio también dos archívitos que guardan los usuarios y las passwords encriptadas por supuesto.

**/home:** En esta carpeta en GNU/LINUX se crean todos los directorios “home” de cada usuario. Es de esperar que en este directorio se encuentran tantas carpetas como usuarios tenga cargados el equipo. En general los nombres de las carpetas coinciden con el loginid del usuario al que pertenece. De acuerdo a configuraciones puede establecerse que todos los usuarios ven los archivos de todos los usuarios o solo los propios y los del sistema.

**/usr:** Es en donde se guardan la mayoría de las aplicaciones instaladas en el equipo.

**/usr/bin:** Es en donde deberían almacenarse los binarios compilados en general para carga dinámica de librerías externas, de todas las aplicaciones instaladas en el sistema. Algunas aplicaciones colocan los binarios en el propio directorio bin, creado en la carpeta raíz de la aplicación.

**/usr/sbin:** Almacena binarios compilados estáticamente. Esto quiere decir que no necesitan librerías externas para ejecutarse. Suele tratarse de aplicaciones o comandos importantes del sistema operativo, usualmente solo con permisos de ejecución para root.

**/sbin:** Igualmente almacena binarios compilados estática o monolíticamente. En este caso se trata de comandos críticos para configuración inicial del equipo, recuperación de disco, diagnóstico, etc.

**/dev:** Este directorio es muy especial y distinto a todos los demás. En general a través de este directorio, el núcleo del sistema operativo accede a todos los dispositivos de hardware del equipo. No suelen ser accedidos directamente por el usuario, ya que no se trata de archivos normales, sino de archivos especiales que interactúan directamente sobre el hardware. Por ejemplo /dev/dsp suele ser el acceso que tiene el núcleo para el procesador de sonido digital propio de la placa de sonido. En esta carpeta también encontramos el acceso directo a todos los discos físicos instalados en el equipo, pero no podemos acceder desde aquí directamente a los datos. Debemos previamente montar este dispositivo en algún directorio normal del sistema operativo. Solo accedemos directamente a los discos a través de este directorio con herramientas especiales como el fsck para reparar un disco.

Aquí también el núcleo interactúa con las consolas de pantalla, video, dispositivos de DVD y CDROM, discos removibles, etc. Lo máximo que realiza el usuario en este directorio es, con usuario root, eventualmente da permisos para acceder a algún dispositivo que haya quedado mal configurado.

**/root:** Es el home del usuario root. No debe confundirse con “/” que es la carpeta raíz del sistema operativo. En GNU/LINUX root es el único usuario cuyo home no se encuentra en /home

## **Configuraciones principales y troubleshooting.**

Todos los archivos de configuración del sistema operativo y las aplicaciones a nivel general se encuentran en el directorio **/etc**. De este directorio es importante destacar los siguientes archivos:

### **fstab**

Este archivo permite definir los discos a montar automáticamente (los que tienen la opción “auto”) y los que pueden ser montados a pedido por cualquier usuario (“noauto,user”) así como también si se montan en solo lectura o en lectura-escritura (“ro” o “rw”). Como se aprecia en el fragmento siguiente, este archivo tiene todos los datos requeridos en el comando mount. La diferencia es que este archivo es utilizado por el sistema para montar/desmontar discos/particiones:

```
# /etc/fstab: static file system information.
#
# <file system> <mount point> <type> <options> <dump> <pass>
proc /proc proc defaults 0 0
/dev/sda2 / ext3 errors=remount-ro 0 1
/dev/sda4 /home ext3 defaults 0 2
/dev/sda3 /usr ext3 defaults 0 2
/dev/sda1 none swap sw 0 0
/dev/hda /media/cdrom0 udf,iso9660 user,noauto 0 0
```

Las columnas del archivo son Dispositivo (disco), Punto de montaje, Tipo de Archivo, Opciones. Las otras dos columnas son para algunas aplicaciones de backup y recuperación. Las últimas dos, si no se trata de una partición crítica del sistema deberían ser 0 y 0.

**Troubleshooting:** Este archivo hay que mirarlo para verificar cuando nuestro sistema operativo no arranca porque no encuentra alguna partición. Puede haber ocurrido que se hayan cambiado los discos de controladoras o bien que se haya modificado incorrectamente este archivo. Si no ocurren esas dos cosas es difícil que el OS no arranque para nada

### **passwd**

En este archivo se guardan los usuarios, el grupo principal al que pertenecen, cual es su “shell” de consola cuando se loguea y la carpeta home de ese usuario.

## **Shadow**

Es un archivo paralelo al passwd en donde se guardan las contraseñas de los usuarios encriptadas. Solo root puede ver este archivo

## **group**

Es en donde se guardan los grupos secundarios de los usuarios cargados en passwd.

Si bien el usuario final es posible que nunca deba acceder a estos archivos directamente es interesante conocer donde se encuentran todos los datos de usuarios y archivos. Seguridad no está cubierto por este taller, pero no está demás conocer estos archivos. Un usuario final generalmente modificará este archivo desde los utilitarios gráficos.

**Troubleshooting:** El archivo group puede servir para verificar los grupos a los que pertenece un usuario. Muchas veces algún usuario no puede usar el cdrom, el pendrive o no tiene sonido en el entorno gráfico, el problema puede ser que no esté incluido en los grupos audio, video, cdrom, plugdev...

## **Configuración del servidor gráfico.**

El modo recomendado de configurar un entorno gráfico en debian y sus derivadas es:

```
dpkg-reconfigure xserver-xorg
```

o si usamos xfree:

```
dpkg-reconfigure xserver-xfree86
```

Esto nos va presentando una serie de pantallas que nos van preguntando las características de la placa de video, monitor, teclado, mouse, etc. Todas las configuraciones de xserver se guardan en los archivos:

```
/etc/X11/xorg.conf (si es Xserver-xorg)
```

```
/etc/X11/XF86Config-4 (si es Xserver-xfree86)
```

Troubleshooting: El entorno gráfico es uno de los típicos problemas del que se inicia en el camino de Linux. El servidor X suele estar compilado para aprovechar al máximo las capacidades de la placa de video. Ejemplos de servidores gráficos compilados específicamente son:

<i>xserver-xorg-video-i810</i>	- X.Org X server -- Intel i8xx, i9xx display driver
<i>xserver-xorg-video-imstt</i>	- X.Org X server -- IMSTT display driver
<i>xserver-xorg-video-intel</i>	- X.Org X server -- Intel i8xx, i9xx display driver
<i>xserver-xorg-video-mga</i>	- X.Org X server -- MGA display driver
<i>xserver-xorg-video-nv</i>	- X.Org X server -- NV display driver
<i>xserver-xorg-video-sis</i>	- X.Org X server -- SiS display driver
<i>xserver-xorg-video-sisusb</i>	- X.Org X server -- SiS USB display driver
<i>xserver-xorg-video-tdfx</i>	- X.Org X server -- tdfx display driver
<i>xserver-xorg-video-tga</i>	- X.Org X server -- TGA display driver
<i>xserver-xorg-video-trident</i>	- X.Org X server -- Trident display driver

Los siguientes son dos servidores estándar siempre listos para cuando no nos arranca el entorno gráfico:

<i>xserver-xorg-video-vesa</i>	- X.Org X server -- VESA display driver
<i>xserver-xorg-video-vga</i>	- X.Org X server -- VGA display driver

Los esenciales siempre estarán instalados, pero si por ejemplo queremos instalar alguno especial debemos hacer:

```
aptitude install xserver-xorg-video-intel (Ver más abajo para detalles acerca de como instalar
```

aplicaciones)

### **Troubleshooting:**

-Estos servers se levantan cuando seleccionamos “vga” o “vesa” en la selección del driver de video en el dpkg-reconfigure del que hablamos hace un instante. Cuando tenemos problemas para levantar el entorno gráfico y no tenemos seguridad acerca de cual es el driver a utilizar, seleccionando “vesa” normalmente nos funciona ya que es compatible con la mayoría de las placas de video actuales

-Otro motivo por el cual puede no levantar el entorno gráfico es porque el mouse o el teclado están mal configurados.

-Cuando no alcanzamos a ver los errores que han ocurrido siempre podemos analizar el archivo:

**/var/log/Xorg.0.log** (o su equivalente Xfree86)

En este archivo seguramente encontraremos algún indicio acerca de lo que está pasando.

### **TroubleShooting a partir de los archivos de log**

GNU/Linux, al igual que todos los UNIX, están programados para loguear toda la actividad que sea posible a fin de permitir encontrar al experto cuando existe un problema. Todos los archivos de logs del sistema los encontramos en:

**/var/log**

En este directorio encontramos los principales archivos de log del sistema. Estos archivos pueden consultarse tanto desde un editor gráfico como así también desde un editor de consola. Lo que hay que tener en cuenta es que en algunos casos se requieren privilegios de root para poder leerse. Los comandos de consola que podemos encontrar muy cómodos para mirar estos archivos son:

**tail** (Muestra las últimas 10 líneas, si le paso -n40 muestra las 40 últimas, etc)

**less** (Muestra todo el archivo. Con la tecla “b” voy hacia atrás (back) con la barra de espacio hacia adelante. Las nuevas versiones permiten hacer lo mismo con PgUp y PgDn

### **Archivos importantes:**

**/var/log/syslog:** Archivo principal de logueo. Todo lo que pasa en el OS lo podemos ver desde este archivo. Es el primer lugar a donde iremos a ver cuando ocurra un error y no sabemos de donde viene

**/var/log/daemon:** Archivo en el que se guardan los logs de los servicios que corren en segundo plano.

**/var/log/kern.log:** Archivo que loguea todos los errores o información del kernel.

**/var/log/Xorg.0.log:** Muestra los eventos, warnings o errores del último intento de arranque del servidor X o entorno gráfico.

Para ver los últimos eventos importantes o errores críticos en el arranque se puede utilizar el comando:

**dmesg**

Este comando me permite mostrar la info que aparece en el archivo **/var/log/messages**

### **Niveles de ejecución del sistema operativo.**

Cuando arrancamos nuestro sistema GNU/LINUX, el arranque va pasando por diversas instancias en que se levantan y bajan diferentes cosas llamados “Niveles de Ejecución”

El estándar LINUX utiliza 5 niveles de ejecución diferentes (RunLevel). Un nivel de ejecución es completamente configurable para levantar o bajar determinados servicios o aplicaciones. En su origen fueron pensados para tener diversos modos de levantar el sistema operativo según necesidades. Esto es muy útil cuando se prueban y desarrollan drivers o aplicaciones para que si alguno tiene algún

problema que hace que el sistema operativo no arranque se pueda utilizar otro nivel, corregir el problema o eventualmente deshabilitar la aplicación o driver defectuoso.

Para la mayoría de los usos que podemos hacer de un GNU/LINUX con dos niveles nos alcanza. De hecho Debian y Ubuntu solo utilizan por defecto dos niveles de ejecución:

1. Level 1: Single user. Lo esperable es que esté configurado con los menores servicios posible, los mínimos como para acceder a los discos y eventualmente la red. En este nivel de ejecución ningún otro usuario puede iniciar una nueva sesión, y el mismo usuario solo puede utilizar una. Este nivel se utiliza fundamentalmente para una especie de “safe mode” para cuando ocurren problemas que impiden levantar los restantes niveles.
2. Level 2: Multi User. Este nivel en los LINUX tradicionales era el nivel en que se levantaban todos los servicios necesarios, pero sin el entorno gráfico. Este nivel permite a varios usuarios conectarse simultáneamente y al mismo tiempo levanta todos los servicios que se le han instalado, como ser web server, mysql, Asterisk, etc. Este nivel era el adecuado para cuando las máquinas venían con poca memoria, para no estar utilizando memoria con aplicaciones pesadas como el entorno gráfico en un equipo que solo funciona como server.

En Debian y Ubuntu sin embargo, el entorno gráfico se configura por defecto también en este nivel y los restantes 3 no se utilizan.

3. Level 5: Tradicionalmente este nivel venía configurado de manera similar al runlevel 2, pero con el agregado del entorno gráfico. Hoy varias distribuciones utilizan aún este esquema.

El resto de los niveles se pueden configurar si los requerimientos del usuario lo necesita. Hoy los sistemas operativos pensados para un equipo de escritorio suelen levantar siempre en el máximo nivel con entorno gráfico y todo pensando en que el usuario ni siquiera tiene que pensar en que nivel de ejecución está.

Para verificar el nivel en el que estamos debemos tipear en una consola:

```
runlevel
```

Para pasar de un nivel a otro se puede usar:

```
init 1, init 2, init 5...
```

(Solo pueden ejecutarse con root)

La realidad es que hoy en día, desde el punto de vista del usuario final, no importa demasiado conocer esto, solo saber que hay una opción de emergencias, que normalmente se muestra en el menú de arranque como “Single User” en el que solo contará con la consola para resolver problemas. En ese caso, al igual que lo hace cuando un Windows no arranca, llamará a un técnico especializado.

También existen los runlevel 0 para apagar y 6 para reiniciar.

## ***Entendiendo los conceptos del entorno gráfico***

A diferencia de los OS Windows y MAC, en los OS UNIX, se entiende que el sistema operativo es completamente independiente de los entornos gráficos. Esto mejora la estabilidad y la flexibilidad a la hora de diseñar soluciones o solucionar problemas.

El entorno gráfico en GNU/LINUX se encuentra separado en tres módulos con funciones distintas:

- Servidor X
- Display manager
- Entorno de escritorio

## Servidor X

Es el encargado de realizar todas las interfases con el hardware de video/multimedia y los programas que quieren utilizar ventanas. De esta manera se dan las condiciones para armar una interfase standard de “servicio de ventanas” que facilita el desarrollo de entornos gráficos al enmascarar la complejidad del acceso al hardware.

En GNU/LINUX disponemos de dos servidores X principales (ambos técnicamente semejantes):

- Xserver-xfree86: es el más antiguo y tradicional de los servidores X. Funciona bien pero tiene algunas restricciones de patentes que no agradan a los creadores exquitos de distribuciones libres.
- Xserver-xorg: Es el más moderno y surge como un “branch” del anterior que se propone reescribir todo el server, mejorando y corrigiendo problemas del anterior y completamente libre de patentes propietarias. Este es el hoy utilizado por las distribuciones como Debian o Ubuntu por defecto.

Al ser técnicamente semejantes y completamente compatibles con todos los entornos graficos, podemos instalar en nuestra distribución el que más nos agrada, independientemente de cuál instale por defecto.

## Entorno de escritorio

Es la contraparte del servidor X que interactúa con el usuario. Aquí tenemos muchas opciones todas técnicamente buenas y orientadas a diversos fines. La selección del entorno de escritorio que tomemos definirá el modo en que el usuario verá las “ventanas” en nuestro equipo. Las hay más sencillas para el usuario final, más livianas para equipos con bajos recursos y básicamente diferencias de gusto estético. Los entornos más utilizados son:

- KDE: Muy similar en el aspecto a Windows XP. Indicado para usuarios que acaban de migrar desde Windows. Tiene gran cantidad de aplicaciones para los más diversos usos. Es por supuesto bastante pesado y corre con al menos 128 Mb de RAM, siendo 256 el mínimo recomendado.
- GNOME: Es un entorno un poco distinto de aquellos a los que nos ha acostumbrado Windows. Los menus de inicio salen de arriba y la barra de tareas también esta ubicada arriba. Es un excelente entorno y con muchísimo cuidado en la estética y la calidad de sus aplicaciones. Es tan pesado como KDE, por lo que para correrlo lo recomendable es tener al menos 256 Mb de RAM. (aunque corre con 128).
- FLUXBOX: es estéticamente agradable pero terriblemente simple y liviano. Corre en equipos con al menos 32 Mb de memoria. Es muy liviano pero el uso puede no ser del todo cómodo para un usuario final.
- XFCE4: es la mejor opción de balance entre uso de recursos, comodidad y estética. Es un avanzado y novedoso entorno de escritorio que utiliza muy pocos recursos y nos ofrece muchísimas facilidades adicionales de configuración y optimización. Sin embargo, en tanto el estilo es un tanto diferente a todo lo conocido no es lo recomendable para usuarios finales un poco conservadores.

### ***Beryl, Compitz, como manejador de ventanas.***

Cada entorno de escritorio tiene su propio “Manejador de ventanas”; que es el que se encarga de realizar los movimientos, la apertura, el cierre de las mismas, etc. Recientemente gracias al impulso dado por Novel a Compitz, ha surgido el entorno “Beryl” que es perfectamente configurable con cualquiera de los anteriores y agrega una cantidad importante de efectos, como movimientos raros,

escritorio cúbico 3D, marcas de agua, etc. Si bien puede ser complejo dejarlo configurado, hoy hay distribuciones como Ubuntu que prácticamente lo traen por defecto. Es importante destacar que no se trata de un entorno gráfico, sino de un manejador de ventanas específico que puede interactuar con los entornos gráficos antes mencionados u otros.

## Display manager

Si bien con el Servidor X y el entorno de escritorio es suficiente para levantar un entorno gráfico, existe un tercer intermediario que nos permite una pantalla gráfica de login. Es decir una pantalla en la que el usuario pueda ingresar su usuario y su password para luego seleccionar qué entorno gráfico desea utilizar.

Los más comunes son:

- XDM: Es el genérico del servidor X. Posee muy pocas posibilidades de customización. Básicamente no es más que una pantalla en la que se pueden ingresar usuario y password para luego al presionar aceptar abrir el entorno gráfico configurado para el usuario.
- KDM: Es el análogo pero en este caso del entorno gráfico KDE. Es altamente customizable. Muestra una lista de usuarios a seleccionar a la manera de Windows XP y permite cambiar antes de ingresar el entorno gráfico a utilizar o abrir en modo consola. Adicionalmente puede configurarse para permitir apagar el equipo.
- GDM: Es el análogo de GNOME. No muestra una lista de usuarios sino que despliega una ventana en la que primero debe escribirse el usuario y luego la password. Asimismo tiene un menú inferior que permite seleccionar el entorno gráfico con el que se desea ingresar, apagar la máquina, etc. Es altamente customizable como el KDM.

Este módulo en cualquiera de sus variantes, tiene como fundamental función validar el usuario y levantar el entorno gráfico deseado. Como se sospechará, puedo tener varios entornos gráficos instalados y usuarios que usen uno u otro alternativamente.

## Redirección de Servidor X:

El diseño modular de esta solución permite por ejemplo redireccionar el servidor X hacia un equipo remoto sin ningún tipo de software adicional. Esto hace que conectándonos por la simple consola desde un equipo remoto, podamos lanzar y utilizar interactivamente en el equipo remoto aplicaciones que corren localmente en el equipo que funciona como server. En otras palabras, sin instalar nada, el servidor X puede ser utilizado tanto local como remotamente, lo que lo hace una opción similar a la de acceso remoto de Windows, con la diferencia que esta facilidad solo redirecciona la ventana de la aplicación ejecutada. Esto está desde el origen como posibilidad en el servidor X. Por supuesto su funcionamiento está determinado por el ancho de banda de conectividad entre ambos equipos, ya que no posee grandes opciones de compresión y funcionamiento como si posee por ejemplo el NXServer, que es la opción favorita para acceso remoto gráfico en GNU/LINUX

## Instalación de aplicaciones

### Manejadores de paquetes

Un manejador de paquetes es un programa que facilita realiza la instalación de aplicaciones. El manejador de paquetes principal de cualquier distribución basado en Debian es el

***dpkg***

Esta aplicación me permite desde la consola instalar cualquier paquete que yo tenga disponible en alguna carpeta. Es el responsable de extraer, copiar y generar todos los archivos de configuración que



la aplicación necesita. El uso típico es:

***dpkg -i alien.deb*** (instala el paquete grabado en disco como lien.deb)

***dpkg -r alien*** (desinstala el paquete cuyo nombre es alien)

Esto lo podemos usar con paquetes ya instalados o con paquetes que ha hemos descargado a nuestra máquina, cd, o pendrive. Hasta aquí no hay nada original. Funciona casi de la misma manera que por ejemplo el programa **rpm** de redhat. La cosa comienza a cambiar cuando esto mismo lo podemos hacer escribiendo:

***apt-get install alien***

Este comando no solo llama al anterior (dpkg) para instalar, sino que:

1. Nos busca en Internet todos los programas o librerías que el programa que quiero instalar necesita;
2. Nos baja todos los paquetes necesarios de Internet;
3. Nos instala y configura la aplicación deseada.

De esta manera, sea cual fuere la complejidad de la aplicación a cargar, instalarla en debian o sus derivadas se limita a : ***apt-get install xxxxxx...*** y listo, no hay que ni siquiera preocuparse por ir a buscar las aplicaciones.

## **Nuevas versiones de apt-get**

A partir de la edición Sarge de Debian, se introdujo de manera estable una nueva aplicación de manejo de paquetes mucho más potente que la anterior y prácticamente con la misma sintaxis si la queremos usar a la manera antigua: **aptitude**

Entonces, donde decíamos:

***apt-get install abiword***

Podríamos usar:

***aptitude install abiword***

Para remover una aplicación podemos usar:

***apt-get remove abiword*** o bien ***aptitude remove abiword***

Lo recomendable es utilizar siempre **aptitude** en lugar de **apt-get** por el principal motivo que **aptitude** nos revisa en cada momento nuestro sistema y si encuentra librerías que no se utilizan más para ninguna aplicación nos ofrece removerlas, con lo que mantiene nuestro sistema siempre limpio.

Si ejecutamos **aptitude** sin ningún argumento nos abre una pantalla pseudo-gráfica en donde nos muestra todas las aplicaciones que tenemos instaladas, las que pueden actualizarse y las que tenemos disponibles para instalar.

## **Uso correcto de aptitude o apt-get.**

Aptitude o apt-get utilizan un catálogo común, almacenado localmente en nuestro equipo para saber en que lugar puede ir a buscar una aplicación para descargar e instalar. Este catálogo no se actualiza automáticamente, por lo tanto debemos siempre tratar de tenerlo perfectamente actualizado antes de instalar nada. Para ello siempre debemos hacer:

1. ***apt-get update*** o ***aptitude update*** (Esto solo actualiza el catálogo, no instala nada en nuestro equipo)
2. ***apt-get install xxxxx*** o ***aptitude install xxxxxx*** (Esto nos busca, descarga e instala la última versión disponible de la aplicación xxxxxx)

Con estos simples dos pasos nos aseguramos que nos estamos instalando la última versión disponible para nuestra distribución y edición.

## Uso de apt-get o aptitude para actualizar todo el sistema operativo

1. *apt-get update* o *aptitude update* (actualiza el catálogo)
2. *apt-get upgrade* o *aptitude upgrade* o *aptitude safe-upgrade* (NOS ACTUALIZA TODAS LAS APLICACIONES DE NUESTRO EQUIPO CON LAS ÚLTIMAS VERSIONES DISPONIBLES PARA LA EDICION O DISTRIBUCIÓN QUE ESTAMOS CORRIENDO. OBIAMENTE PUEDE DEMORAR YA QUE PUEDE TENER VARIOS MEGAS PARA DESCARGAR DE Internet.)

## Archivo de configuración de apt-get o aptitude

El archivo a partir del cual estas aplicaciones arman en catálogo es el :

*/etc/apt/sources.list* (o sus equivalentes dentro de */etc/apt/sources.list.d/\** )

En estos archivos cargamos las urls de los repositorios públicos de aplicaciones para nuestra distribución/edición. Por ejemplo, en DEBIAN podemos encontrar:

***deb http://ftp.us.debian.org/debian/ etch main non-free contrib***

Esta línea indica:

- El repositorio de la distribución/edición estable (etch)
  - para los paquetes core y mantenidos por los desarrolladores Debian (main)
  - para los paquetes que tienen alguna restricción de patente o licencia (non-free)
  - para los paquetes extra mantenidos por terceros o “contributors) (contrib)

***deb http://ftp.us.debian.org/debian/ lenny main non-free contrib***

Esta línea indica lo mismo que la anterior, excepto que en este caso se trata de la distribución testing de debian (lenny).

Todas las líneas que comienzan con “deb” son repositorios de binarios. Las que comienzan con deb-src son repositorios de código fuente, que podemos bajarnos, modificar y compilar por nuestra cuenta por supuesto

Si incluyo ambas líneas en el archivo, apt o aptitude siempre intentarán bajarme e instalarme la última versión, aparezca en la distribución que aparezca.

Si bien ubuntu no utiliza exactamente el concepto de distribuciones stable, testing y unstable, la sintaxis es exactamente igual con algunas variantes en los argumentos finales de la línea.

Lo sumamente interesante de esto es que tanto debian entiende perfectamente los repositorios de ubuntu, como ubuntu los de debian. Es decir, si alguna vez no tenemos una aplicación en alguno pero si en el otro, podemos temporariamente agregar el repositorio del otro para instalar lo que necesitamos (luego es recomendable sacarlo). De todas maneras casi nunca hace falta hacer esto, ya que ambos están muy sincronizados.

Veamos un ejemplo de archivo de urls del repositorio de ubuntu:

***deb http://ar.archive.ubuntu.com/ubuntu/ hardy main restricted***

***deb-src http://ar.archive.ubuntu.com/ubuntu/ hardy main restricted***

Estos son los repositorios del core del sistema, por supuesto con full support de la distribución. Como

ven aparece el nombre/edición de la distribución: hardy y luego los repositorios del core (main y restricted) del Sistema Operativo

***deb http://ar.archive.ubuntu.com/ubuntu/ hardy-updates main restricted***

Esta línea tiene los “security” updates de la distribución.

***deb http://ar.archive.ubuntu.com/ubuntu/ hardy universe***

Esta línea tiene los paquetes generados y soportados por terceros, que no tienen restricción de patentes

***deb http://ar.archive.ubuntu.com/ubuntu/ hardy multiverse***

Aquí tenemos el “non-free” de Debian. En este caso se trata de aplicaciones desarrolladas por terceros pero que pueden tener alguna restricción de licencia en la medida que pueden ser aplicaciones que utilizan algoritmos, íconos, o porciones de código registradas bajo licencias no totalmente libres.

## **Uso especial de aptitude o apt-get para realizar un pasaje de una distribución completa a una más nueva.**

Cada x meses, tanto ubuntu como debian actualizan sus distribuciones, lo que estaba en “testing” pasa a ser “estable”, y lo que estaba en “inestable” es corregido y pasado a “testing”. En ese momento, nuestro equipo quedaría una distribución atrás. Por ese motivo, existe la opción dist-upgrade que nos pasa de una distribución a otra. Pero hay que tener ciertos cuidados al utilizarla. Un modo correcto de hacerlo es:

1. El archivo sources.list debe tener siempre las distribuciones nombradas con su nombre (lenny, etch, sid) ya que si las llamamos stable, testing, o unstable, cuando lo testing pase a stable y lo sid pase a testing tendremos todo desfasado
2. Antes de cambiar el nombre para la nueva distribución, conviene actualizar la actual, para esto hacemos:
  - a) ***aptitude update***
  - b) ***aptitude upgrade*** o ***aptitude safe-upgrade***
3. Una vez que la opción upgrade funcione bien (sin conflictos irresueltos o actualizaciones truncas)
  - a) cambio en los repositorios los nombres de las distribuciones, por ejemplo, si estaba en “etch” y quiero pasar a “lenny”, donde decía “etch” pongo “lenny”, y así con todas las líneas en que haga referencia a alguna de ellas.
  - b) Ejecuto: ***aptitude update***
  - c) Ejecuto: ***aptitude dist-upgrade***

Espero que se baje todo lo nuevo y lo termine de instalar y listo, ya tengo mi distribución completamente actualizada.

## **Uso de entorno gráfico para instalar y configurar aplicaciones.**

La opción más tradicional gráfica para instalar aplicaciones es Synaptic. Synaptic realiza todas las acciones que son capaces de realizar apt-get y aptitude, con la única diferencia que me muestra todo en un entorno gráfico en donde solo tengo que seleccionar paquetes y apretar botones, sin tener que recordar ninguna sintaxis. Para instalarnos synaptic en nuestro equipo ya sabrán como hacerlo:

***aptitude update***

***aptitude install synaptic***

Una vez descargado e instalado lo podemos acceder desde el menu del entorno gráfico y hasta podemos olvidarnos de aptitude si no nos gusta mucho la consola.

Asimismo, Ubuntu y Kubunto traen nuevas aplicaciones gráficas para instalar y configurar aplicaciones y hasta un “live update” que nos avisa cuando tenemos nuevas versiones para actualizar.

Para descargarse la última versión de este instructivo, ingresar en:  
[http://www.marianoacciardi.com.ar/textos\\_gnulinux/PrimerosPasos.pdf](http://www.marianoacciardi.com.ar/textos_gnulinux/PrimerosPasos.pdf)  
<http://www.linuxreloaded.org/PrimerosPasos.pdf>