

1.- Software de un sistema informático.

Sistema informático: conjunto de elementos que hacen posible el tratamiento automatizado de la información. El software de un sistema informático está formado por programas, estructuras de datos y documentación asociada. está distribuido en el ordenador, los periféricos y el subsistema de comunicaciones(sistemas operativos, paquetes ofimáticos, compresores, editores de imágenes)

1.1.- Requisitos e instalación: Determinación del equipo necesario.

En todo proceso de instalación se han de seguir unos pasos que describiremos a continuación.

1. Determinación del equipo necesario.
2. Ejecución del programa de instalación.
3. Configuración de la aplicación.

Determinación del equipo necesario

Debemos conocer que características o requisitos tendrá nuestro sistema informático. Las app están enfocadas a plataformas concretas, con unas necesidades de hardware y software para que funcione. No se pueden instalar en otra distinta o si no se cumplen unos requisitos mínimo. Antes hay que conocer esos requisitos y verificar que puede instalarse y pueden ser de naturaleza hardware:

- Plataforma hardware: PC, Mac, etc.
- Procesador: fabricante, velocidad, generalmente se indica el inferior posible de la gama con el que la aplicación funciona adecuadamente.
- Memoria RAM mínima.
- Espacio mínimo disponible en el soporte de almacenamiento: por ejemplo, en disco duro o unidad de almacenamiento externa para aplicaciones portables.
- Tarjeta gráfica: la memoria gráfica necesaria para el buen funcionamiento de la aplicación.
- Resolución recomendada del monitor

Y de carácter software:

- Plataforma software: sistema operativo
- Otros paquetes software adicionales necesarios, (actualizaciones de seguridad para el SO, la JVM (máquina virtual de Java), el Flash Player, etc. Para instalar el editor de imágenes de Microsoft te indica que debes tener instalado varios componentes de Microsoft.

Teniendo en cuenta lo visto anteriormente, los fabricantes de aplicaciones informáticas suelen establecer tres niveles de requisitos para la instalación de sus aplicaciones:

- Equipo básico.
- Equipo opcional.
- Equipo en red.

1.2.- Requisitos e instalación: Ejecución del programa de instalación.

Ejecución del programa de instalación

Por la instalación de un programa/aplicación informática entendemos el conjunto de pasos que nos van a permitir copiar los archivos necesarios, configurar, implantar y poner en funcionamiento una aplicación en un sistema informático. Dos niveles en función de los conocimientos del usuario:

Instalación básica: usuarios con pocos conocimientos informáticos. El programa realizará una instalación en función de los elementos que detecte en el equipo y según unos parámetros básicos establecidos por defecto por el fabricante.

Instalación personalizada o avanzada: usuario experto incluir o eliminar elementos de la aplicación con el fin de optimizar los recursos del sistema informático, instalando sólo aquellos elementos de la aplicación que se van a utilizar.

La aplicación se encuentra normalmente en formato comprimido. El traspaso del programa al soporte de almacenamiento se realiza a través del programa de instalación (su nombre puede ser setup, install, instalar, etc.), y es el encargado de extraer los bloques de la aplicación de los discos, descomprimiéndolos si es necesario; crear la estructura de directorios necesaria, ubicar los archivos de la aplicación donde corresponda, y, si fuera necesario, modificar el registro del sistema. En la actualidad, distribuyen DVD, CD o con posibilidad de descarga de los archivos de instalación o en imágenes ISO

1.3.- Requisitos e instalación: Configuración de la aplicación.

Configuración de la aplicación

Configurar las opciones de la aplicación, a veces también del sistema operativo, y configurar el entorno de trabajo. Se pueden modificar los parámetros establecidos por defecto para la aplicación. Algunas aplicaciones pueden generar una serie de archivos de configuración con los datos introducidos por los usuarios. El usuario debe realizar por ultimo la configuración del entorno de trabajo para definir una serie de

Tema 4.- Introducción al Software de base de un Sistema Informático

parámetros de funcionamiento que se adecuen a las exigencias del usuario. Se realizará en el caso que no sea satisfactoria la configuración establecida por defecto. (Ajuste y distribución de la pantalla y Definición de directorios de trabajo)
Tras este último paso de configuración la aplicación ya está lista para empezar a funcionar adecuadamente.

1.4.- Tipos de aplicaciones informáticas.

Podemos decir que las aplicaciones informáticas pueden clasificarse en dos tipos, en función del ámbito o la naturaleza de uso:

Aplicaciones de propósito general:

Se emplean para el desempeño de funciones no específicas (informes, documentos, presentaciones, gráficos, hojas de cálculo, etc.). Se suelen comercializar en paquetes integrados denominados suites, tales como: Microsoft Office, OpenOffice, StarOffice, Lotus SmartSuite, etc. y se componen de:

- Gestión de texto:
 - o Editores de texto (no permiten formato, como por ejemplo Notepad).
 - o Procesadores de texto (Microsoft Word, Writer de OpenOffice).
 - o Programas de autoedición, maquetación y diseño: Microsoft Publisher.
- Hoja de cálculo (Microsoft Excel, Calc de OpenOffice, Lotus 1-2-3).
- Asistente personal: agenda, calendario, listín telefónico.
- Generador de presentaciones (Microsoft PowerPoint, Impress de OpenOffice).
- Herramientas de acceso y gestión de bases de datos (Microsoft Access, Base de OpenOffice).
- Editores de XML y HTML (Microsoft FrontPage).

Otras aplicaciones de propósito general son:

- Herramientas para la comunicación: GroupWare o Trabajo en grupo como gestores de e-mail, servicio de mensajería instantánea, etc. Gestión de FAX.
- Utilidades y herramientas: como antivirus, navegadores web, gestores de archivos, compresores de archivos, visores de archivos.

Aplicaciones de propósito específico:

Por otro lado, las aplicaciones de propósito específico se utilizan para el desempeño de funciones específicas, científicas, técnicas o de gestión, tales como:

- Administración, contabilidad, facturación, gestión de almacén, RRHH: por ejemplo ContaPlus.
- Entorno gráficos de desarrollo: Visual Studio, Borland Builder C++, etc.
- Herramientas de administración de bases de datos: Oracle, phpMyAdmin, etc.
- Herramientas de gestión de red: Tivoli, NetView, etc.
- Herramientas “ad-hoc” especializadas: OCR/OMR, monitores bursátiles, gestión empresarial ERP, etc.
- Herramientas de diseño gráfico y maquetación: Corel Draw, Adobe PhotoShop, PaintShop, etc.
- Herramientas de ingeniería y científicas utilizadas en ámbitos de investigación, en universidades, etc.

1.5.- Licencias software.

Las licencias software nos sirven para establecer un contrato entre el autor de una aplicación software (sometido a propiedad intelectual y a derechos de autor) y el usuario. En el contrato se definen con precisión los derechos y deberes de ambas partes, es decir, los “actos de explotación legales”. Por otra parte, entendemos por derecho de autor o copyright la forma de protección proporcionada por las leyes vigentes en la mayoría de los países para los autores de obras originales incluyendo obras literarias, dramáticas, musicales, artísticas e intelectuales, tanto publicadas como pendientes de publicar. Distinguimos varios tipos de software o licencias en función de lo limitadas que estén las acciones del usuario sobre el mismo:

Software propietario

Redistribución o modificación prohibidos o necesitan una autorización. Los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones), o su código fuente no está disponible, o el acceso a éste se encuentra restringido. Usuario solo tiene derecho de uso de la aplicación. Una persona física o jurídica posee los derechos de autor sobre un software, negando o no otorgando, los derechos de usar el programa con cualquier propósito (estudiar como funciona, adaptarlo, distribuir copias, mejorarlo y publicarlo). Un software sigue siendo no libre aún si el código fuente es hecho público, cuando se mantiene la reserva de derechos sobre el uso, modificación o distribución (por ejemplo, el programa de licencias Shared source, de código abierto para uso académico de Microsoft).

Software libre

Proporciona al usuario las cuatro libertades siguientes:

- Utilizar el programa, para cualquier propósito.
- Estudiar cómo funciona el programa y adaptarlo a tus necesidades, debe proporcionarse las fuentes, directa o indirectamente, pero siempre de forma fácil y asequible.
- Distribuir copias, Mejorar el programa y hacer públicas las mejoras a los demás.

Tema 4.- Introducción al Software de base de un Sistema Informático

Si no incorpora estas libertades se considera no libre o semilibre. La mayor parte de las licencias de software libre surgen de la FSF. Suele estar gratis, o al precio de costo de la distribución a través de otros medios; Pero no hay que asociar software libre a "software gratuito" (freeware), ya que libre, puede ser distribuido comercialmente.

Software de dominio público

No está protegido con copyright y que no requiere de licencia, derechos de explotación son para toda la humanidad. El autor lo dona a la humanidad o los derechos de autor han expirado (en un plazo contado desde la muerte del autor, generalmente 70 años). En caso de que el autor condicione el uso de su software bajo una licencia, por muy débil que sea, ya no se consideraría software de dominio público.

Software con copyleft

Es el software libre cuyos términos de distribución no permiten a los redistribuidores agregar ninguna restricción adicional cuando lo redistribuyen o modifican, o sea, la versión modificada debe ser también libre.

Existen otros tipos de software, tales como:

- **Freeware:** Programa totalmente gratuito. Es posible que requiera que nos registremos, pero siempre de forma gratuita.
- **Shareware (Demo):** Se trata de una versión reducida del programa, con algunas funciones desactivadas para que podamos probarlo y decidir si lo vamos a comprar o no.
- **Shareware (Versión limitada por tiempo):** Se trata de una versión totalmente funcional por un cierto número de días (normalmente 30, pero puede variar según la compañía) tras la cual no lo podremos usar o se verá reducida su funcionalidad. Su objetivo es poder probar la aplicación y luego decidir si la compraremos o no.

Veamos algunos ejemplos de aplicaciones de software libre:

- **Sistemas Operativos:** Debian GNU/Linux, Ubuntu, Linex, Guadalinex, MAX, etc.
- **Entornos de escritorio:** GNOME, KDE, etc.
- **Aplicaciones de oficina:** OpenOffice, KOffice, LATEX, etc.
- **Navegación web:** FireFox, Konqueror, etc.
- **Aplicaciones para Internet:** Apache, Zope, etc.

2.- Sistemas Operativos.

El sistema operativo es un conjunto de programas que se encarga de gestionar los recursos hardware y software del ordenador, por lo que actúa como una interfaz entre los programas de aplicación del usuario y el hardware puro.

2.1.- Concepto y objetivos de los sistemas operativos.

Los principales objetivos de los sistemas operativos son:

- **Abstraer** al usuario de la complejidad del hardware: El sistema operativo hace que el ordenador sea más fácil de utilizar.
- **Eficiencia:** Permite que los recursos del ordenador se utilicen de la forma más eficiente posible.
- Permitir la **ejecución de programas:** El sistema operativo realiza todas las tareas necesarias para ello, tales como cargar las instrucciones y datos del programa en memoria, iniciar dispositivos de entrada/salida y preparar otros recursos.
- **Acceder a los dispositivos** entrada/salida: El sistema operativo suministra una interfaz homogénea para los dispositivos de entrada/salida para que el usuario pueda utilizar de forma más sencilla los mismos.
- **Proporcionar una estructura** y conjunto de operaciones para el sistema de archivos.
- **Controlar el acceso al sistema y los recursos:** en el caso de sistemas compartidos, proporcionando protección a los recursos y los datos frente a usuarios no autorizados.
- **Detección y respuesta ante errores:** prever todas las posibles situaciones críticas y resolverlas, si es que se producen.
- **Capacidad de adaptación:** Puede evolucionar a la vez que surgen actualizaciones hardware y software.
- **Gestionar las comunicaciones en red:** permitir manejar con facilidad todo lo referente a la instalación y uso de las redes
- Permitir a los usuarios **compartir recursos y datos:** sistema operativo el papel de gestor de los recursos de una red.

Evolución histórica de los sistemas operativos.

Primera generación (1945-1955)

En un principio **no existían** sistemas operativos, se **programaba directamente sobre el hardware**. Los programas estaban hechos directamente en **código máquina** y el control de las funciones básicas se realiza mediante **paneles enchufables**. La mayoría de los programas **usaban rutinas de E/S** y un programa cargador (automatizaba la carga de programas ejecutables en la máquina) esto constituía una forma rudimentaria de sistema operativo.

2ª Generación (1955-1965)

La programación se realizaba en **lenguaje ensamblador y en FORTRAN sobre tarjetas perforadas**. Otro aspecto importante de esta generación es el **procesamiento por lotes**, en el cual mientras el sistema operativo está ejecutando un proceso, **éste último dispone de todos los recursos hasta su finalización**. La **preparación de los trabajos** se realiza a través de un lenguaje de control de trabajos conocido como

Tema 4.- Introducción al Software de base de un Sistema Informático

JCL. El **sistema operativo residía en memoria** y tenía un **programa de control que interpretaba las tarjetas de control**, escritas JCL. Dependiendo del contenido de la tarjeta de control el sistema operativo realizaba una acción determinada. Este programa de control es un antecedente de los modernos intérpretes de órdenes. Como mejora del procesamiento por lotes surgió el procesamiento fuera de línea (off-line), en el cual las operaciones de carga de datos y salida de resultados de un proceso podían realizarse de forma externa y sin afectar al tiempo que el procesador dedicaba a los procesos. A esto ayudó la aparición de las cintas magnéticas y las impresoras de líneas. FMS (Fortran Monitor System) y IBSYS.

3ª Generación (1965-1980)

La característica principal de esta generación fue el desarrollo de la **multiprogramación y los sistemas compartidos**. En los sistemas multiprogramados se cargan varios programas en memoria simultáneamente y se alterna su ejecución. Esto maximiza la utilización del procesador. Como evolución aparecen los sistemas de tiempo compartido donde el tiempo del procesador se comparte entre programas de varios usuarios pudiendo ser programas interactivos. OS/360, CTSS, MULTICS y UNIX.

4ª Generación (1980-hasta hoy)

Aparecen los **ordenadores personales** y ejemplos de sistemas operativos de los primeros ordenadores personales son MS-DOS, desarrollado por Microsoft, Inc., para el IBM PC y MacOS de Apple Computer, Inc. Steve Jobs, cofundador de Apple, apostó por la primera **interfaz gráfica** basada en ventanas, iconos, menús y ratón a partir de una investigación realizada por Xerox. Siguiendo esta filosofía aparecería MS Windows. Durante los **90 apareció Linux** a partir del núcleo desarrollado por Linus Torvalds.

Los sistemas operativos evolucionan hacia sistemas interactivos con una interfaz cada vez más amigable al usuario. Los sistemas Windows han ido evolucionando (Windows 3.x, 98, 2000, XP, Vista, 7, 10, Windows Server 2003, 2008, 2012, etc), al igual que lo han hecho Linux (con multitud de distribuciones, Ubuntu, Debian, RedHat, Mandrake, etc) y los sistemas Mac (Mac OS 8, OS 9, OS X, Mac OS X 10.6 "Snow Leopard", Mac OS X 10.10, entre otros). **Desarrollo de redes de ordenadores a mediados de los años 80** que ejecutan sistemas operativos en red y sistemas operativos distribuidos. En un sistema operativo en red los usuarios tienen conocimiento de la existencia de múltiples ordenadores y pueden acceder a máquinas remotas y copiar archivos de un ordenador a otro. En un sistema distribuido los usuarios no saben dónde se están ejecutando sus programas o dónde están ubicados sus programas, ya que los recursos de procesamiento, memoria y datos están distribuidos entre los ordenadores de la red, pero todo esto es transparente al usuario.

Existen sistemas operativos integrados, para una gran diversidad de dispositivos electrónicos, tales como, teléfonos móviles, PDAs (Personal Digital Assistant, Asistente Digital Personal u ordenador de bolsillo), otros dispositivos de comunicaciones e informática y electrodomésticos. Ejemplos de este tipo de sistemas operativos son PalmOS, WindowsCE, Android OS, etc.

2.2.- Tipos de sistemas operativos.

Ahora vamos a clasificar los sistemas operativos en base a su estructura, servicios que suministran y por su forma.

Tipos de sistemas operativos		
Por estructura	Por sus servicios	Por su forma
Monolíticos	Monousuario	Sistema operativo en red
Jerárquicos	Multiusuario	
Máquina Virtual	Monotarea	Sistema operativo distribuido
Microkernel o Cliente-Servidor	Multitarea	
Monolíticos	Monoprocesador	
	Multiprocesador	

Sistemas operativos por su estructura

Monolíticos: Es la estructura de los primeros sistemas operativos. Un solo programa desarrollado con rutinas entrelazadas que podían llamarse entre sí. Por lo general, eran sistemas operativos hechos a medida, pero difíciles de mantener.

Jerárquicos: Mayor organización del software del sistema operativo, dividiéndose en partes más pequeñas, diferenciadas por funciones y con una interfaz clara para interoperar con los demás elementos. Un ejemplo de este tipo de sistemas operativos fue MULTICS.

Máquina Virtual: El objetivo de los sistemas operativos es el de integrar distintos sistemas operativos dando la sensación de ser varias máquinas diferentes. Presentan una interfaz a cada proceso, mostrando una máquina que parece idéntica a la máquina real subyacente. Estas máquinas no son máquinas extendidas, son una réplica de la máquina real, de manera que en cada una de ellas se pueda ejecutar un sistema operativo diferente, que será el que ofrezca la máquina extendida al usuario. VMware y VM/CMS son ejemplos de este tipo de sistemas operativos.

Microkernel o Cliente-Servidor: El modelo del núcleo de estos sistemas operativos distribuye las diferentes tareas en porciones de código modulares y sencillas. El objetivo es aislar del sistema, su núcleo, las operaciones de entrada/salida, gestión de memoria, del sistema de archivos, etc. Esto incrementa la tolerancia a fallos, la seguridad y la portabilidad entre plataformas de hardware. Algunos ejemplos son MAC OS X o AIX.

Tema 4.- Introducción al Software de base de un Sistema Informático

Sistemas operativos por sus servicios

Monousuario: Son aquellos que soportan a un usuario a la vez, sin importar el número de procesos o tareas que el usuario pueda ejecutar en un mismo instante de tiempo. Ejemplos de sistemas operativos de este tipo son MS-DOS, Microsoft Windows 9x y ME, MAC OS, entre otros.

Multiusuario: Son capaces de dar servicio a más de un usuario a la vez, ya sea por medio de varios terminales conectadas al ordenador o por medio de sesiones remotas en una red de comunicaciones. No importa el número de procesadores en la máquina ni el número de procesos que puede ejecutar cada usuario simultáneamente. Algunos ejemplos serán UNIX, GNU/Linux, Microsoft Windows Server o MAC OS X.

Monotarea: Sólo permiten una tarea a la vez por usuario. Se puede dar el caso de un sistema multiusuario y monotarea, en el cual se admiten varios usuarios simultáneamente pero cada uno de ellos puede ejecutar sólo una tarea en un instante dado. Ejemplos de sistemas monotarea son MS-DOS, Microsoft Windows 3.x y 95 (estos últimos sólo simulan la multitarea).

Multitarea: Permite al usuario realizar varias tareas al mismo tiempo. Algunos ejemplos son MAC OS, UNIX, Linux, Microsoft Windows 98, 2000, XP, Vista y 7.

Monoprocesador: Es aquel capaz de manejar sólo un procesador, de manera que si el ordenador tuviese más de uno le sería inútil. MS-DOS y MAC OS son ejemplos de este tipo de sistemas operativos.

Multiprocesador: Un sistema operativo multiprocesador se refiere al número de procesadores del sistema, éste es más de uno y el sistema operativo es capaz de utilizarlos todos para distribuir su carga de trabajo. Estos sistemas trabajan de dos formas: simétricamente (los procesos son enviados indistintamente a cualquiera de los procesadores disponibles) y asimétricamente (uno de los procesadores actúa como maestro o servidor y distribuye la carga de procesos a los demás).

Sistemas operativos por su forma

Sistemas operativos en red: Estos sistemas tienen la capacidad de interactuar con los sistemas operativos de otras máquinas a través de la red, con el objeto de intercambiar información, transferir archivos, etc. La clave de estos sistemas es que el usuario debe conocer la ubicación de los recursos en red a los que desee acceder. Los sistemas operativos modernos más comunes pueden considerarse sistemas en red, por ejemplo: Novell, Windows Server, Linux, etc.

Sistemas operativos distribuidos: Abarcan los servicios de red, las funciones se distribuyen entre diferentes ordenadores, logrando integrar recursos (impresoras, unidades de respaldo, memoria, procesos, etc.) en una sola máquina virtual que es a la que el usuario accede de forma transparente. En este caso, el usuario no necesita saber la ubicación de los recursos, sino que los referencia por su nombre y los utiliza como si fueran locales a su lugar de trabajo habitual. MOSIX es un ejemplo de estos sistemas operativos.

2.3.- Servicios de los sistemas operativos.

El sistema operativo necesita administrar los recursos para tener control sobre las funciones básicas del ordenador. Los principales recursos que administra el sistema operativo son:

- El procesador.
- La memoria.
- Los dispositivos de entrada/salida.
- El sistema de archivos.

Núcleo

Para gestionar todos estos recursos. El núcleo normalmente representa sólo una pequeña parte de todo lo que es el sistema operativo, pero es una de las partes que más se utiliza. Por esta razón, el núcleo reside por lo general en la memoria principal, mientras que otras partes del sistema operativo son cargadas en la memoria principal sólo cuando se necesitan. Parte principal del código de un sistema operativo y se encarga de controlar y administrar los servicios y peticiones de recursos. Para ello se divide en distintos niveles:

- Gestión de procesos
- Gestión de memoria
- Gestión de la entrada/salida (E/S)
- Gestión del Sistema de archivos

3.- Gestión de procesos.

Entre las principales tareas del sistema operativo está la de administrar los procesos del sistema. Un proceso es un programa en ejecución. Un proceso simple tiene un hilo de ejecución (o subproceso), en ocasiones, un proceso puede dividirse en varios subprocesos. Un hilo es básicamente una tarea que puede ser ejecutada en paralelo con otra tarea. Por lo que los hilos de ejecución permiten a un programa realizar varias tareas a la vez. En los sistemas operativos modernos los procesos pueden tener diferentes estados, según el momento de creación, si están en ejecución, si se encuentran a la espera de algún recurso, etc. Pero podemos hacer una simplificación, y un proceso, en un instante dado, puede estar en uno de los tres estados siguientes: Listo, En ejecución y Bloqueado.

Los procesos en estado listo son los que pueden pasar a estado de ejecución si el planificador del sistema operativo los selecciona, esto es, cuando llegue su turno (según el orden de llegada o prioridad). Los procesos en estado de ejecución son los que se están ejecutando en el procesador en un momento dado. Los procesos que se encuentran en estado bloqueado están esperando la respuesta de algún otro proceso para poder continuar con su ejecución, por ejemplo una operación de entrada/salida.

Tema 4.- Introducción al Software de base de un Sistema Informático

El sistema operativo sigue la pista de en qué estado se encuentran los procesos, decide qué procesos pasan a ejecución, cuáles quedan bloqueados, en definitiva, gestiona los cambios de estado de los procesos. Los procesos pueden comunicarse entre sí o ser independientes. En el primer caso, los procesos necesitarán sincronizarse y establecer una serie de mecanismos para la comunicación; por ejemplo, los procesos que pertenecen a una misma aplicación y necesitan intercambiar información. En el caso de procesos independientes estos, por lo general, no interactúan y un proceso no requiere información de otros.

3.1.- Planificación del procesador.

En la planificación del procesador se decide cuánto tiempo de ejecución se le asigna a cada proceso del sistema y en qué momento. Si el sistema es monousuario y monotarea no habrá que decidir, pero en el resto de los sistemas multitarea esta decisión es fundamental para el buen funcionamiento del sistema, ya que determinará la correcta ejecución de los distintos programas de aplicación que se estén ejecutando. El sistema operativo almacena en una tabla denominada tabla de control de procesos con la información relativa a cada proceso que se está ejecutando en el procesador. Ésta es:

- Identificación del proceso.
- Identificación del proceso padre.
- Información sobre el usuario y grupo que lo han lanzado.
- Estado del procesador. El contenido de los registros internos, contador de programa, etc. Es decir, el entorno volátil del proceso.
- Información de control de proceso.
- Información del planificador.
- Segmentos de memoria asignados.
- Recursos asignados.

Una estrategia de planificación debe buscar que los procesos obtengan sus turnos de ejecución de forma apropiada (momento en que se le asigna el uso de la CPU), junto con un buen rendimiento y minimización de la sobrecarga (overhead) del planificador mismo. En general, se buscan cuatro objetivos principales:

1. Todos los procesos en algún momento obtienen su turno de ejecución o intervalos de tiempo de ejecución hasta su terminación con éxito.
2. El sistema debe finalizar el mayor número de procesos por unidad tiempo.
3. El usuario no percibirá tiempos de espera demasiado largos.
4. Evitar el aplazamiento indefinido, los procesos deben terminar en un plazo finito de tiempo. Esto es, el usuario no debe percibir que su programa se ha parado o “colgado”.

La carga de trabajo de un sistema informático a otro puede variar considerablemente, esto depende de las características de los procesos. Nos podemos encontrar:

- Procesos que hacen un uso intensivo de la CPU.
- Procesos que realizan una gran cantidad de operaciones de Entrada/Salida.
- Procesos por lotes, procesos interactivos, procesos en tiempo real.
- Procesos de menor o mayor duración.

En función de cómo sean la mayoría de los procesos habrá algoritmos de planificación que den un mejor o peor rendimiento al sistema.

3.2.- Planificación apropiativa y no apropiativa.

La planificación no apropiativa (en inglés, no preemptive) es aquella en la que, cuando a un proceso le toca su turno de ejecución, ya no puede ser suspendido; es decir, no se le puede arrebatar el uso de la CPU, hasta que el proceso no lo determina no se podrá ejecutar otro proceso. Este esquema tiene sus problemas, puesto que si el proceso contiene ciclos infinitos, el resto de los procesos pueden quedar aplazados indefinidamente. Otro caso puede ser el de los procesos largos que penalizarían a los cortos si entran en primer lugar.

La planificación apropiativa (en inglés, preemptive) supone que el sistema operativo puede arrebatar el uso de la CPU a un proceso que esté ejecutándose. En la planificación apropiativa existe un reloj que lanza interrupciones periódicas en las cuales el planificador toma el control y se decide si el mismo proceso seguirá ejecutándose o se le da su turno a otro proceso.

En ambos enfoques de planificación se pueden establecer distintos algoritmos de planificación de ejecución de procesos. Algunos de los algoritmos para decidir el orden de ejecución de los procesos en el sistema son los siguientes:

- Round Robin
- Por Prioridad
- El trabajo más corto primero
- El primero en llegar, el primero en ejecutarse
- El tiempo restante más corto

4.- Gestión de memoria.

Hemos visto en la gestión de procesos que el recurso compartido es el procesador. Sin embargo, para que un proceso se pueda ejecutar no sólo requiere tiempo de procesamiento sino también estar cargado en memoria principal. Esto es así, porque ningún proceso se puede activar antes de que se le asigne el espacio de memoria que requiere. Así, la memoria se convierte en otro recurso clave que tendrá que gestionar el sistema operativo y la parte encargada de ello se denomina gestor de memoria.

La función principal del gestor de memoria es la de asignar memoria principal a los procesos que la soliciten. Otras funciones serán:

Tema 4.- Introducción al Software de base de un Sistema Informático

- Controlar las zonas de memoria que están asignadas y cuáles no.
- Asignar memoria a los procesos cuando la necesiten y retirársela cuando terminen.
- Evitar que un proceso acceda a la zona de memoria asignada a otro proceso.
- Gestionar el intercambio entre memoria principal y memoria secundaria en los casos en que la memoria principal está completamente ocupada, etc.

De este modo, la gestión de memoria va a tener que cubrir los siguientes requisitos:

- Reubicación: En un sistema multitarea la memoria va a estar compartida entre varios procesos, el gestor de memoria debe decidir qué zonas de memoria asigna a cada proceso y que zonas descarga.
- Protección: El gestor de memoria debe evitar que los procesos cargados en memoria interfieran unos con otros accediendo a zonas de memoria que no les corresponden. Para ello, se comprueba que las referencias a la memoria generadas por un proceso durante su ejecución sólo hacen referencia a la zona de memoria asignada a ese proceso y no acceden a zonas prohibidas, áreas de memoria donde estén otros procesos.
- Control de memoria: El sistema operativo, a través del gestor de memoria, tiene que controlar las zonas de memoria libres y las asignadas, además de saber las zonas de memoria que corresponden a cada proceso.
- Controlar y evitar en lo posible casos de fragmentación de la memoria: Existen dos tipos de fragmentación de la memoria principal, la fragmentación interna y la externa. La fragmentación interna sucede al mal gastarse el espacio interno de una partición cuando el proceso o bloque de datos cargado es más pequeño que la partición. Por el contrario, la fragmentación externa sucede cuando la memoria externa a todas las particiones se divide cada vez más y van quedando huecos pequeños y dispersos en memoria difícilmente reutilizables.
- Organización lógica y física: En ocasiones la memoria principal no es suficiente para proporcionar toda la memoria que necesita un proceso o para almacenar todos los procesos que se pueden ejecutar. Entonces los procesos pueden ser intercambiados a disco y más tarde, si es necesario, vuelven a cargar en memoria. Por lo que el gestor de memoria se encarga de gestionar la transferencia de información entre la memoria principal y la secundaria (disco).

Para llevar a cabo estas funciones nos encontramos con seis técnicas utilizadas por el administrador de memoria:

- partición fija
- partición dinámica
- partición simple
- segmentación simple
- Memoria virtual paginada
- Memoria virtual segmentada

La forma más común de administración de la memoria implica crear una memoria virtual; con este sistema, la memoria del ordenador aparece, para cualquier usuario del sistema, mayor de lo que es. El sistema de gestión de la memoria que se use dependerá del ordenador y sistema operativo en particular que se tenga.

5.- Gestión de la entrada/salida.

Anteriormente, vimos que una de las funciones del ordenador era procesar la información, dicha información la obtiene y muestra a través de los periféricos. La parte del sistema operativo que se encarga de este proceso es la gestión de la E/S (entrada/salida). En la primera unidad estudiamos los periféricos y recordamos que se clasificaban en periféricos:

- De entrada: son periféricos que reciben información y la transmiten al ordenador para su procesamiento, por ejemplo: el ratón, el teclado, el escáner, etc.
- De salida: periféricos que presentan la información procesada por el ordenador, por ejemplo: la impresora, el plóter (para impresión de planos y cartografía), etc.
- De entrada y salida: Añoran ambas funciones, por ejemplo: el monitor, el disco duro, unidad de lectura y grabación de DVD, etc.

El sistema operativo hace que los dispositivos se conecten al sistema y realicen sus funciones de forma adecuada y eficiente. El sistema operativo abstrae de la complejidad y peculiaridad hardware de cada periférico para que las aplicaciones de usuario puedan hacer uso de los periféricos de una manera estandarizada y más sencilla. El sistema operativo actúa pues como intermediario entre ellos, gracias a los controladores de dispositivo.

¿Cómo pueden entenderse los programas de aplicación con los dispositivos periféricos? Hay multitud de tipos y fabricantes de periféricos, esto conlleva que tanto el sistema operativo como los fabricantes de periféricos deben estandarizar el acceso a los dispositivos utilizando lo que se denominan controladores de dispositivos (device drivers).

Un periférico siempre tiene dos partes: un controlador, se encarga de la comunicación con la CPU y un dispositivo mecánico, electromecánico o electromagnético. El controlador es un software, generalmente, suministrado por el fabricante del dispositivo o bien por el desarrollador del sistema operativo. De esta manera, estos controladores actúan como interfaz entre los programas y el hardware.

Las técnicas más utilizadas por los sistemas operativos para gestionar las entradas / salidas son dos:

- Spools: Los datos de salida se almacenan de forma temporal en una cola situada en un dispositivo de almacenamiento masivo (spool), hasta que el dispositivo periférico requerido se encuentre libre. De este modo se evita que un programa quede retenido porque el periférico no esté disponible. El sistema operativo dispone de llamadas para añadir y eliminar archivos del spool. Se utiliza en dispositivos que no admiten intercalación, como ocurre en la impresora, ya que no puede empezar con otro hasta que no ha terminado.

Tema 4.- Introducción al Software de base de un Sistema Informático

• **Buffers:** Es para dispositivos que pueden atender peticiones de distintos orígenes. En este caso, los datos no tienen que enviarse completos, pueden enviarse porciones que el buffer retiene de forma temporal. También se utilizan para acoplar velocidades de distintos dispositivos. Así, si un dispositivo lento va a recibir información más rápido de lo que puede atenderla se emplea un buffer para retener temporalmente la información hasta que el dispositivo pueda asimilarla. Esto ocurre entre una grabadora de DVD y el disco duro, ya que la primera funciona a una menor velocidad que el segundo.

Las distintas formas de funcionamiento de la E/S en los sistemas operativos según la intervención de la CPU tenemos:

E/S programada: la CPU tiene todo el protagonismo ya que inicia y lleva a cabo la transferencia. Esta técnica repercute en la velocidad de proceso del ordenador porque la CPU debe dejar todo lo que está haciendo para ocuparse del proceso de entrada/salida.

E/S por interrupciones: la CPU ejecuta la transferencia pero el inicio es pedido por el periférico que indica así su disponibilidad. La CPU no pregunta a los dispositivos sino que son estos los que la avisan cuando es necesario.

Acceso directo a memoria (DMA): la transferencia es realizada por un controlador especializado. Esta técnica acelera enormemente el proceso de la E/S y libera a la CPU de trabajo. Lo habitual es que los datos que se quieren escribir en el dispositivo o que son leídos del dispositivo provengan o vayan a la memoria del ordenador, pues bien en este caso, la CPU inicia el proceso, pero luego este continúa sin necesitar a la CPU, con lo que se acelera mucho el proceso de entrada/salida y se libera a la CPU del proceso.

6.- Gestión del sistema de archivos.

Esta parte del sistema operativo gestiona el servicio de almacenamiento, por lo que permite crear, modificar, borrar archivos y directorios y para ello utiliza generalmente una estructura jerárquica. Cada sistema operativo utilizará su propio sistema de archivos, no obstante las operaciones que se pueden realizar sobre el sistema de archivos son bastante similares. Así, todos los sistemas de archivos actuales utilizan los directorios o carpetas para organizar a los archivos. El sistema de archivos es el software que provee al sistema operativo, a los programas de aplicación y a usuarios de las funciones para operar con archivos y directorios almacenados en disco proporcionando mecanismos de protección y seguridad.

Los objetivos más importantes en la implementación de un sistema de archivos son:

- **Optimizar el rendimiento** mediante un acceso rápido para recuperar la información contenida en archivos: No se debe ralentizar el sistema en general por una deficiente gestión de los medios de almacenamiento, discos duros.
- **Fácil actualización:** Los cambios (añadir, borrar y modificar) no deben suponer una tarea complicada para el usuario y las aplicaciones.
- **Economía de almacenamiento:** Intentar que los archivos desperdicien la menor cantidad de espacio en disco posible. Es muy importante evitar la fragmentación de los discos.
- **Mantenimiento sencillo:** Evitar las operaciones complicadas a usuarios y programas, ocultando los detalles y proporcionando un acceso estandarizado a los archivos.
- **Fiabilidad para asegurar la confianza en los datos:** Deben proveer sistemas que aseguren que los datos escritos o leídos (entradas/salidas) sean correctos y fiables. También se debe minimizar o eliminar la posibilidad de pérdida o destrucción de datos.
- **Incorporar mecanismos de seguridad y permisos:** Esto es especialmente importante en sistemas de archivos de sistemas operativos multiusuario. Se debe poder proteger los archivos de un usuario del acceso de los demás usuarios. Por ejemplo estableciendo permisos de escritura, lectura o ejecución.
- **Control de concurrencia:** Se debe controlar y asegurar el acceso correcto a los archivos por parte de varios usuarios a un tiempo, posiblemente bloqueando el archivo en uso hasta que termine la operación de modificación en curso.

6.1.- Organización lógica y física.

Se suele diferenciar entre la organización de discos a nivel físico (hardware) y lógico (software). El nivel físico de almacenamiento de datos en un disco duro consiste en el formato en pistas, sectores, cilindros y platos. Pero esto es muy dependiente del hardware concreto que se esté usando y además funciona a muy bajo nivel.

NTFS	FAT32	EXT4
Windows 7	Windows 2000	distribuciones Linux

Los sistemas de archivos deben proveer una capa de abstracción que oculte los detalles puramente hardware al usuario y le permita utilizar el medio de almacenamiento (disco) de una forma intuitiva y cómoda, por supuesto más cercana a los hábitos humanos de organización de la información. Éste es el nivel lógico del sistema de archivos y naturalmente en el que estamos más interesados.

A esto se le llama organización del sistema de archivos y suele coincidir en todos los sistemas de archivos actuales, utilizando el esquema de almacenamiento en archivos y la organización en carpetas o directorios.

¿En qué consisten los archivos y carpetas? y ¿cómo los gestiona el sistema de archivos?

Archivos: Es el elemento central de la mayoría de programas de aplicación. Los archivos o ficheros son estructuras de datos en disco donde se almacena la información y los programas de un ordenador. Pueden tener diversas estructuras y ello dependerá del sistema de archivos de nuestro sistema operativo y de la extensión del mismo.

Cada archivo de un sistema tendrá unas características, o atributos, que lo identifican y le sirven al sistema de archivos y al sistema operativo para manejarlo correctamente. Los atributos pueden variar de un sistema a otro, pero suelen coincidir al menos en los siguientes:

- o **Nombre:** Identificador principal del archivo para el usuario. Cada sistema operativo establece las reglas para nombrar a los archivos, en cuanto a longitud y caracteres permitidos.
- o **Extensión:** La extensión de un archivo son los caracteres que se colocan al final del nombre de un archivo para especificar su tipo de contenido. Por ejemplo, la extensión “.TXT” indica que el archivo es de texto o la extensión “.EXE” indica que el archivo es un programa ejecutable.

Tema 4.- Introducción al Software de base de un Sistema Informático

- o Permisos: El sistema de archivos debe controlar qué usuarios están autorizados a utilizar cada archivo y que operaciones pueden realizar. Por ejemplo un archivo puede tener permiso de lectura y escritura para un usuario y en cambio otro usuario solo podrá utilizar el archivo en modo de lectura.
- o Creador: Identificador del usuario que creó el archivo.
- o Propietario: Identificador del usuario que es el propietario actual del archivo.
- o Fecha de creación: Fecha y hora de la creación del archivo.
- o Fecha del último acceso: Fecha y hora del último acceso al archivo.
- o Fecha de la última modificación: Fecha y hora de la última modificación al archivo.
- o Tamaño actual: Número de bytes que ocupa el archivo en el disco duro del ordenador.
- o Directorios: También denominados carpetas, son archivos especiales que cumplen la función de almacenar y organizar en su interior a archivos y otros subdirectorios. Son estos los que permiten mantener una cierta organización en el sistema de archivos. La organización en directorios mantiene forma de árbol invertido que comienza por un directorio principal llamado raíz y se va ramificando en otros directorios que pueden contener archivos y otros directorios. Respecto a los atributos de un directorio, como archivos que son coinciden con los atributos de estos.

6.2.- Operaciones soportadas por un sistema de archivos.

Las operaciones básicas sobre archivos que la mayoría de los sistemas de archivos soportan son:

- o Crear: Los archivos se crean sin datos y después el usuario o alguna aplicación los van llenando.
- o Borrar: Si un archivo ya no es necesario debe eliminarse para liberar espacio en disco.
- o Abrir: Antes de utilizar el archivo se debe abrir para que el sistema conozca sus atributos, tales como el propietario, fecha de modificación, etc.
- o Cerrar: Tras realizar las operaciones deseadas sobre el archivo, éste puede cerrarse para asegurar su integridad y liberar recursos de memoria que tuviera asignados.
- o Leer: Los datos se leen del archivo; quien hace la llamada (programa) debe especificar la cantidad de datos necesarios y proporcionar un buffer para colocarlos.
- o Escribir: Los datos se escriben en el archivo. El tamaño del archivo puede aumentar si se agregan datos nuevos o no si lo que se hace es actualizar los existentes.
- o Renombrar: Permite modificar el atributo nombre de un archivo ya existente.

Los sistemas de archivos también suministran un conjunto de operaciones para los directorios, las más comunes son: crear, borrar, abrir, cerrar, renombrar y leer. Además existen otras dos operaciones sobre archivos y directorios como son la de crear un enlace y eliminarlo. La operación de crear un enlace se utiliza para poder acceder a un archivo o directorio desde distintos puntos de la organización de directorios del sistema sin tener que duplicar o copiar el archivo o directorio en cuestión.

6.3.- Rutas de acceso.

Los sistemas de archivos necesitan una forma de determinar la localización exacta de un archivo o directorio en la estructura del árbol de directorios. La ruta de acceso a un archivo o directorio se indica nombrando todos los directorios y subdirectorios que tienen que atravesarse hasta llegar al elemento concreto. Dependiendo del sistema operativo con el que se trabaje cambiará la forma de establecer la ruta de acceso. Por ejemplo, en Windows se utiliza la barra “\” para separar los directorios y en Linux se utiliza la barra “/”.

Existen dos tipos de rutas de acceso:

- 1. Ruta de Acceso Absoluta:** Se comienza desde el directorio raíz y se va descendiendo en la estructura de directorios hasta llegar al archivo o directorio buscado. En las rutas de acceso absolutas se conoce la ubicación exacta.
- 2. Ruta de Acceso Relativa:** Se utiliza junto con el concepto de directorio de trabajo o directorio activo, que es aquel donde estamos situados en un momento dado. Consiste en escribir la ruta a partir del directorio activo, esto se indica con un punto (.) que hace referencia a la localización actual donde nos encontramos; y al elemento padre (o directorio de nivel superior) se le referencia con dos puntos seguidos (..), que se utilizan para subir en la jerarquía. En las rutas de acceso relativas no se conoce la ubicación exacta.

7.- Mecanismos de seguridad y protección.

El sistema operativo debe protegerse activamente a sí mismo y a los usuarios de acciones accidentales o malintencionadas. Cada vez es más necesaria la seguridad en los sistemas, ya que actualmente la mayoría de los ordenadores se encuentran conectados en red y el número de usuarios y recursos compartidos ha aumentado considerablemente. Vamos a diferenciar entre seguridad y protección. Por seguridad nos referimos a una política donde se deciden qué accesos están permitidos, qué usuarios pueden acceder, en que forma y a qué recursos. Por otro lado, la protección hace referencia al mecanismo que se utiliza para llevar a cabo la política de seguridad.

Los requisitos que debe cumplir un sistema operativo son:

Confidencialidad: Los elementos del sistema sólo serán visibles por aquellos usuarios o grupos autorizados

Integridad: Los elementos del sistema sólo serán modificados por los usuarios o grupos autorizados.

Disponibilidad: Los elementos del sistema sólo estarán disponibles para usuarios y grupos autorizados.

Los elementos amenazados son:

Hardware.

Software.

Datos.

Líneas de comunicación.

Tema 4.- Introducción al Software de base de un Sistema Informático

Seguridad informática, detección de acciones accidentales o malintencionadas sobre activos informáticos.			
Elemento amenazado	Confidencialidad	Integridad	Disponibilidad
Hardware			Robo o sobrecarga de equipos, eliminando el servicio.
Software	Realización de copias no autorizadas del software.	Alteración de un programa en funcionamiento haciéndolo fallar durante la ejecución o haciéndolo que realice alguna tarea para la que no está programado.	Eliminación de programas, denegando el acceso a los usuarios.
Datos	Lecturas de datos no autorizadas. Revelación de datos ocultos de manera indirecta por análisis de datos estadísticos.	Modificación de archivos existentes o invención de nuevos.	Eliminación de archivos, denegando el acceso a los usuarios.
Líneas de comunicación	Lectura de mensajes. Observación de la muestra de tráfico de mensajes.	Mensajes modificados, retardados, reordenados o duplicados. Invención de mensajes falsos.	Destrucción o eliminación de mensajes. Las líneas de comunicación o redes no se encuentran disponibles.

Para hacer frente a estas acciones el sistema operativo agrupa la seguridad según tres aspectos:

1. Seguridad en el uso de recursos y servicios y control de acceso: Utilizar un mecanismo de control de acceso a los recursos que tan sólo permita el acceso si existe el permiso correspondiente. Se establecerán políticas de permisos para acceder y operar con recursos y servicios.
2. Seguridad en el acceso al sistema: Asegurar que sólo entran los usuarios autorizados. Para ello podrán utilizarse un sistema de contraseñas eficaz con niveles de acceso diferentes.
3. Seguridad en el uso de redes: Evitar que se puedan producir escuchas y alteraciones en los datos que viajan por la red. Se aplicarán técnicas de cifrado y descifrado de las comunicaciones a través de la red.

8.- Documentación y búsqueda de información técnica.

Todo software con una cierta complejidad suele venir acompañado de una documentación, ésta puede ser en formato digital o papel. Esta documentación toma forma en manuales, tutoriales y demás guías de referencia que sirven para mostrar al usuario cómo se implanta y utiliza una aplicación. A continuación, veremos los tipos de documentación nos podemos encontrar:

Manual de usuario (con distintos niveles: básico, intermedio, avanzado): Explica en detalle la forma de operar con la aplicación, las explicaciones de texto suelen venir acompañadas de capturas de pantalla para hacer que el seguimiento sea más fácil y captar la atención del lector.

Manual de Instalación y Configuración del programa: Dedicado por lo general a la persona encargada de la puesta en funcionamiento del programa. Conlleva la explicación de los pasos de instalación, configuración, carga inicial de datos, si fuera necesaria, y demás pruebas de aceptación antes de que el programa pase a la fase de explotación (cuando comienza a ser utilizada por el usuario final). Este manual puede encontrarse incluido en el manual del administrador que veremos a continuación. En empresas donde se deben poner en marcha aplicaciones en red que requieren ciertos conocimientos técnicos en la configuración de aplicaciones el perfil de la persona que implanta la aplicación y el del usuario final está claramente diferenciado. Sin embargo, en otras muchas ocasiones la persona que instala, configura y utiliza el programa suele ser la misma, sobretudo en aplicaciones de escritorio.

Manual del Administrador: Documentación que va dirigida a la persona responsable del correcto funcionamiento, seguridad y rendimiento de la aplicación. Esta persona es, en muchos casos, la misma que instala y configura la aplicación.

Guía de referencia rápida: Contiene las funciones básicas imprescindibles para instalar, con las opciones por defecto, y comenzar a utilizar una aplicación.

En ocasiones podemos encontrarnos con problemas o dudas técnicas sobre la instalación, configuración o utilización de un programa que no quedan claramente resueltas en la anterior documentación. En esas situaciones existe la posibilidad de buscar información adicional utilizando otros medios, como por ejemplo:

- Consulta al soporte técnico del desarrollador software, vía web, email o teléfono.
- Consulta en foros de expertos.
- Consulta en bases de conocimiento.
- Consulta en FAQs (Frequently Asked Questions – Preguntas Frecuentes).
- etc.

1.HISTORIA

En los años setenta, dos investigadores de los Laboratorios Telefónicos Bell (Bell Telephone Labs o BTL) llamados Dennis Ritchie y Ken Thompson desarrollaron un sistema operativo al que llamaron Unix.

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 fue usada por las universidades, que consiguieron acceso a un gran sistema operativo y a su código fuente. Una de las universidades que adquirió una licencia de Unix fue la Universidad de California en Berkeley. Al poco tiempo, la gente de Berkeley había leído el código fuente y había escrito varios programas adicionales para Unix que otros investigadores podrían encontrar útiles.

La Universidad decidió entonces distribuir este código a la comunidad y llamó a sus distribuciones BSD (Berkeley Software Distribution). Al principio estas distribuciones BSD consistían en el Unix de BTL y algunas herramientas propias de la Universidad, pero muy pronto comenzaron a cambiar la forma en que el propio sistema operativo funcionaba, comenzaron a modificar el código fuente del propio Unix. Entre otras cosas implementaron el manejo de memoria virtual y programaron el soporte para los protocolos de Arpanet que luego se convertiría en el conocido Internet. Todos estos cambios eran recopilados y lanzados como distribuciones BSD basadas en Unix de BTL.

A mediados de los años ochenta, Richard Stallman, entonces en el Instituto Tecnológico de Massachusetts (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 y 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, que había salido del ámbito puramente universitario. Entre otras empresas, Microsoft distribuía Xenix, una versión de Unix para el procesador 80386 y Sun Microsystems utilizaba BSD como base para su SunOS.

Como resultado de la finalización del juicio antimonopolio, BTL fue desmembrada en varias compañías, una de ellas AT&T, que, de acuerdo con los nuevos arreglos legales ya podía comercializar Unix. Pero cuando AT&T quiso vender Unix comercialmente se dio cuenta que ya existían varias variaciones de su Unix que se estaban comercializando. AT&T inmediatamente lanzó una demanda legal contra todas estas compañías y sobre sus sistemas operativos.

AT&T también decidió demandar a la Universidad de California en Berkeley por distribuir código basado en el Unix de AT&T sin licencia. La Universidad de California a su vez, demandó a AT&T argumentando que BSD incorporaba muchísimas mejoras y que estas mejoras habían sido introducidas por AT&T dentro de Unix. En el mercado convivían en ese momento el Unix de AT&T junto con muchas distribuciones basadas en BSD como el 386BSD y el FreeBSD. Todas estas distribuciones fueron detenidas por el Juez mientras se celebraban los diversos juicios que se habían abierto y se prohibió su uso.

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, o usado comercialmente y el sistema se encontraba muy limitado en muchísimos aspectos.

Un estudiante de Finlandia, Linus Torvalds, al ver que no le era posible modificar 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. En realidad, Linus Torvalds no creó un sistema operativo completo, sino su pieza más importante, el núcleo.

El proyecto GNU que Stallman había iniciado hacía ya casi diez años había producido un sistema casi completo a excepción del núcleo, que no conseguía hacer funcionar correctamente. Linus unió su núcleo con GNU formando un sistema operativo completo al que llamó Linux.

Richard Stallman insiste que el sistema operativo resultante debiera ser llamado GNU/Linux, ya que incluye más código del proyecto GNU que del proyecto Linux. En la actualidad, el proyecto GNU puede ser instalado sin usar el núcleo de Linux, sino con su propio núcleo conocido como Hurd. Sin embargo, dicho núcleo es excesivamente inestable y casi todas las instalaciones de GNU se realizan con el Kernel de Linux.

El éxito inmediato que tuvo Linux se basó en una variedad de factores. Por un lado, es un núcleo realmente bueno, llegó justo en el momento en que GNU necesitaba un núcleo, y coincidió con el boom de Internet, lo que permitió que se creara una comunidad alrededor de dicho núcleo, tanto para desarrollarlo como para distribuirlo, usarlo y mantenerlo.

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 de AT&T y lanzaría una última distribución de BSD totalmente libre de problemas de licencias.

Esta distribución fue el 4.4-BSD Lite2. Quien quisiera seguir trabajando sobre BSD debería basar su distribución en 4.4-BSD Lite2 para no tener problemas legales. Hoy en día, existen varias distribuciones del BSD basadas en esa versión.

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. Usando el código de FreeBSD se creó el sistema operativo Darwin BSD, que a su vez fue usado por Apple para crear su Mac OS, que a su vez fue la base para crear el sistema iOS usado en la actualidad en todos los iPhone, iPod, iPad, etc.

Tema 4.- Introducción al Software de base de un Sistema Informático

NetBSD: el énfasis de este grupo es la portabilidad del sistema operativo. Actualmente existen sistemas NetBSD para casi cualquier plataforma.

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 libres con una especial preocupación en que no puedan ser espiados ni forzados.

Existen numerosas distribuciones Linux (también conocidas como "distros"), ensambladas por individuos, empresas y otros organismos, conjuntando el núcleo Linux, el sistema GNU y todas las herramientas y utilidades que deseen. Ahí compañías que crean sus propias herramientas y hay compañías que se limitan a usar software libre disponible. Los sistemas Linux funcionan sobre más de 20 diferentes plataformas de hardware, desde un PC con arquitectura x86 hasta una consola de videojuegos.

2.DEBIAN

Debian o más concretamente Debian GNU/Linux es una distribución Linux que basa sus principios y fin en el software libre y en luchar contra el software propietario o cerrado. Creado por Debian Project en el año 1993, dicha organización es la responsable de la creación y mantenimiento de la distribución, centrada en el núcleo Linux y en las utilidades GNU. También mantienen y desarrollan otros sistemas operativos GNU basados en los núcleos Hurd, llamado Debian GNU/Hurd, y NetBSD, llamado Debian GNU/NetBSD.

Debian nace como una apuesta por separar en sus versiones el software libre del software propietario. El modelo de desarrollo es independiente de 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 respete su licencia.

Algunas de sus características principales son:

- Disponibilidad en varias plataformas hardware. Debian 7 está disponible para más de 10 plataformas distintas.
- Una amplia colección de software disponible. La versión 7 cuenta con más de 37.500 paquetes (programas).
- Un grupo de herramientas gráficas para facilitar el proceso de instalación y actualización del software en las últimas versiones.
- Su compromiso con los principios y valores involucrados en el movimiento del Software Libre. Es la distribución que más en serio se toma estos principios, llegando incluso a ser tachada de intransigente por otras distribuciones.
- No tiene preferencia sobre ningún entorno gráfico en especial ya sea GNOME, KDE, XFCE, LXDE, FluxBox... Cualquier entorno puede funcionar en Debian, dado que dicha distro le da una importancia crucial al hecho de permitir que el usuario tenga libertad para elegir sus propias interfaces.

Debian es una distribución sobre la que se han generado una gran cantidad de distribuciones propias. Algunas de ellas son Augustux, Catux, Gnoppix, Guadalinex, Knoppix, Kanotix, Linex, Linspire, MEPIS, Progeny, SkoleLinux, Ubuntu, UserLinux, Xandros, Mint, etc. Debian va por su versión 9.

2.1 RAMAS DE DESARROLLO DE DEBIAN

Cada versión de Debian establece 4 fases distintas

Debian estable (stable), es la versión estabilizada de esta distribución. Cuenta con el apoyo del Equipo de seguridad de Debian y es la recomendada para uso en producción.

Debian en pruebas (testing). En esta versión se encuentran paquetes que han estado previamente en la versión Inestable, pero que contienen muchos menos fallos. Además, deben de poder instalarse en todas las arquitecturas para las cuales fueron construidas. Es la versión más utilizada como sistema de escritorio por aquellos que buscan tener el software más actualizado, aunque se pierde en estabilidad. De aquí saldrá la futura versión Estable.

En Debian inestable (unstable), es donde tiene lugar el desarrollo activo de Debian. Es la rama que usan los desarrolladores del proyecto. La rama inestable de Debian siempre tiene como nombre en clave Sid.

Cuando la versión de pruebas (testing) llega a un nivel aceptable de fallos, entonces se "congela", lo que significa que ya no se aceptan nuevos paquetes desde la versión inestable. A continuación, se trabaja para pulir el mayor número de bugs posibles, para así liberar la versión Estable. Ese periodo puede durar varios meses debido a que no se fija una fecha de lanzamiento.

Existe un repositorio de Debian llamado **experimental**. Esto no es una rama de desarrollo, sino un repositorio que cuenta con los últimos paquetes aparecidos.

Debian no será liberada como estable en tanto sus desarrolladores no consideren que está preparada y es estable. Esa estabilidad se mide basándose en el registro de errores de software o Bug Tracking. Cuando se alcanza un nivel aceptable se le asigna un número de versión, acordado previamente, y se libera como versión estable, solo las versiones estables cuentan con número de versión. La anterior versión estable es clasificada como old-stable, se mantendrá soporte por un periodo, generalmente un año, y posteriormente será archivada.

3.ARCH LINUX

Arch Linux es una distribución Linux para computadoras x86. El enfoque de diseño se centra en la simplicidad, la elegancia, la coherencia de código y el minimalismo. Arch Linux define simplicidad como «...una ligera estructura base sin agregados innecesarios, modificaciones, o complicaciones, que permite a un usuario individual modelar el sistema de acuerdo a sus propias necesidades». La simplicidad de su estructura no implica sencillez en su manejo. (Más bien al contrario).

Tema 4.- Introducción al Software de base de un Sistema Informático

Inspirado por CRUX, otra distribución minimalista, Judd Vinet creó Arch Linux en marzo de 2002. Arch Linux utiliza un modelo de rolling release, de tal manera que una actualización regular del sistema operativo es todo lo que se necesita para obtener la última versión del software; las imágenes de instalación son simplemente «capturas» de los principales componentes del sistema. No se puede hablar por tanto de una versión 1.0 o 2.3, Arch Linux se va actualizando continuamente. Arch Linux no posee herramientas de configuración automática, compartiendo así la misma filosofía que otras distribuciones, como por ejemplo Slackware, por lo que para poder llegar a instalar y configurar el sistema se necesita un grado de conocimiento más que básico.

4.RED HAT

Red Hat Linux es una distribución Linux creada por Red Hat, la cual fue una de las más populares en sus inicios. La versión 1.0 fue presentada el 3 de noviembre de 1994. 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 muchas otras distribuciones, tales como Mandrake, Fedora o Yellow Dog Linux.

Desde el 2003, Red Hat ha desplazado su enfoque hacia el mercado de los negocios con la distribución Red Hat Enterprise Linux (RHEL). Esta versión ha tenido mucho éxito comercial, dado que aúna las ventajas del software libre con el soporte comercial de una gran empresa.

Ahora mismo Red Hat se encarga de los sistemas operativos Red Hat Enterprise Linux, Fedora y CentOS (Estos dos últimos mediante supervisión, apoyo y patrocinio, ya que cuentan con equipos de desarrollo propios). RHEL es su apuesta para empresas, Fedora para usuarios normales y CentOS sirve para ambos cometidos.

5.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. Utiliza el sistema de paquetes RPM (RedHat package manager) aunque no guarda relación con esta distribución. También, al igual que Red Hat, ha establecido una compañía dedicada a dar soporte a empresas, aunque su éxito ha sido relativo.

6.UBUNTU

Ubuntu es una distribución GNU/Linux basada en Debian GNU/Linux. Proporciona un sistema operativo actualizado y estable para el usuario, con un fuerte enfoque en la facilidad de uso y de instalación del sistema. Al igual que otras distribuciones se compone de múltiples paquetes de software normalmente distribuidos bajo una licencia libre o de código abierto. Está patrocinado y financiado por Canonical Ltd., una compañía británica propiedad del empresario sudafricano Mark Shuttleworth.

Cada seis meses se publica una nueva versión de Ubuntu la cual recibe soporte por parte de Canonical durante dieciocho meses por medio de actualizaciones de seguridad, parches para bugs críticos y actualizaciones menores de programas. Las versiones LTS (Long Term Support), que se liberan cada dos años normalmente, reciben soporte durante tres años en los sistemas de escritorio y cinco para la edición orientada a servidores.

Ubuntu soporta oficialmente dos arquitecturas de hardware: Intel i386 y AMD64. A partir de Ubuntu 9.04 (abril de 2009) se empezó a ofrecer soporte oficial para procesadores ARM. Esta distribución ha sido y está siendo traducida a más de 130 idiomas, y cada usuario es capaz de colaborar voluntariamente a esta causa, a través de Internet.

Canonical intenta desde hace varios años que Ubuntu de beneficios. Buscando una monetización del proyecto, canonical intentó que Ubuntu fuera utilizado en teléfonos, tablets, televisores, etc. Un problema que esta política ocasionó es que Canonical intentó unificar el escritorio de todos estos dispositivos mediante un interfaz de control conocido como Unity. Este interfaz es bastante incomodo de utilizar en PC y tiene unos requerimientos de proceso bastante elevados.

En la actualidad Ubuntu ha abandonado este enfoque debido a su nula aceptación por el mercado y ha vuelto a un escritorio más adecuado al PC, eliminado el Unity y utilizando Gnome, que es el entorno de escritorio habitual en la mayoría de las distribuciones Linux. Ubuntu en la actualidad es una opción muy interesante para desarrollos especiales como “cloud computing” o “big data” donde cuenta con soluciones automatizadas bastante interesantes, pero ha perdido bastante empuje en su versión escritorio.

7.GUADALINEX

Guadalinex es una distribución Linux promovida por la Junta de Andalucía para fomentar el uso del software libre en su comunidad autónoma. Está inspirada en Gnu LinEx, un proyecto similar de la Junta de Extremadura basada en Debian. Inicialmente por lo tanto Guadalinex estuvo basada en Debian GNU/Linux debido al acuerdo inicial entre la Junta de Andalucía y la de Extremadura, pero desde la versión 3.0 se basa en Ubuntu.

Existen varios «sabores» de Guadalinex, según a qué público esté orientada:

Guadalinex Base: de propósito general, se publica una vez al año.

Guadalinex BIB: para Bibliotecas (en desarrollo).

Guadalinex CDM: para los Centros de Día de Mayores.

Guadalinex EDU: para los centros educativos, mantenida por el CGA (Centro de Gestión Avanzado).

Guadalinex Guadalinfo: para los centros Guadalinfo.

Guadalinex Mini: para ordenadores antiguos, lleva IceWM como gestor de ventanas.

Guadalinex UCA: para la Universidad de Cádiz. La última versión liberada es la Guadalinex 9

4. WINDOWS

Windows 10 es la última versión del sistema operativo de Microsoft. En esta unidad veremos en qué consiste este S.O. y sus diferencias con otros como Linux. Veremos todas las características necesarias tanto hardware como de otros componentes del sistema que sean necesarias para instalarlo. Windows 10 combina las características de sus dos versiones anteriores para que sea más fácil de usar tanto en ordenadores de sobremesa como en portátiles y dispositivos móviles. A continuación, veremos los fundamentos de Windows 10.

1.1 Windows VS Linux. ¿En qué se diferencian?

Tanto Windows como Linux son dos sistemas operativos que se usan tanto a nivel empresarial en estaciones de trabajo o servidores, como a nivel doméstico. Sin embargo difieren en una serie de características. La principal diferencia entre ambos es la licencia propietaria de Windows y la licencia de software libre de Linux. Otra de las principales diferencias es el sistema de fichero de Windows. En Linux todo eran ficheros (dispositivos, directorios, entrada/salida, los propios ficheros...) ordenados en una estructura en árbol, donde el directorio raíz puede considerarse el inicio del sistema de archivos, y en él se ramifican otros subdirectorios. La raíz se indica con una barra inclinada '/'. En cambio, en Windows, los archivos se almacenan en carpetas en diferentes unidades de datos como C: D: E: Al ser Linux un sistema operativo de código abierto, permite que los usuarios puedan cambiar el código fuente según las necesidades. Esto ha provocado que surjan distintas distribuciones de Linux (distros) orientadas cada una a distintas necesidades y mantenidas por empresas o por una comunidad. En cambio, el sistema operativo Windows se trata de un S.O. comercial, por lo que el código fuente es cerrado y el usuario no puede acceder al código. Windows utiliza diferentes dispositivos como C, D, E para guardar los ficheros y directorios, en cambio, en Linux se utiliza un árbol de directorios.

En Windows se consideran las impresoras, discos HDD o SSD, unidades de CD-ROM como dispositivos, sin embargo en Linux los periféricos son tratados como ficheros. Los tipos de usuarios también son diferentes entre ambos sistemas operativos, en Windows tenemos tres tipos: Administrador, Estándar e Invitado, pero en Linux tenemos la cuenta de usuario, el administrador o root y cuentas de servicios, creadas principalmente por los paquetes del sistema.

En Windows no puede haber dos ficheros con el mismo nombre en la misma carpeta, aunque esté escrito de forma diferente entre mayúsculas y minúsculas. Linux por otra parte, es sensible a las mayúsculas y minúsculas y por lo tanto sí permite tener dos ficheros como "ejemplo" y "Ejemplo" en el mismo directorio. El kernel de Windows es híbrido, es decir, tiene dos capas, una de usuario y otra del sistema, donde se ejecutan los componentes principales del sistema. En cambio Linux utiliza un kernel monolítico, donde todo se ejecuta en una misma capa (aunque internamente esté dividida en subcapas).

Windows también se trata de uno de los sistemas operativos más utilizados tanto a nivel doméstico como empresarial, con una cuota de mercado a nivel mundial en el primer semestre de 2020 del 87.54% según Statista. Por esta razón desarrollar software y aprender a administrar estos sistemas operativos, tiene una gran compatibilidad con dispositivos en todo el mundo. También como ventajas con respecto a otros sistemas operativos, tiene integración directa con los servicios y herramientas de Microsoft, esta es una de las principales razones por las que se utiliza en las empresas en todo el mundo, como Microsoft Azure o Office 365. Cabe destacar también, que en los últimos años Microsoft está incluyendo en su sistema operativo el kernel completo de Linux mediante el WSL (Windows Subsystem for Linux), permitiendo utilizar Linux desde la propia terminal de Windows y poder correr ejecutables de Linux nativamente en Windows 10 y Windows Server 2019.

1.2 Historia de Windows

1.2.1 MS-DOS

El sistema operativo original creado por Microsoft para IBM, MS-DOS, era el sistema operativo estándar para los ordenadores personales de IBM. Se trataba de un S.O. muy simple, cuyas versiones posteriores se fueron satisficando a medida que incorporaban características de los sistemas operativos para miniordenadores.

1.2.2 Windows 1.0

Presentado en 1985, Microsoft Windows 1.0 recibió su nombre debido a las cajas informáticas o "ventanas" que representaban un aspecto fundamental del sistema operativo. En lugar de teclear los comandos de MS-DOS, Windows 1.0 permitía a los usuarios apuntar y hacer clic para acceder a las ventanas.

1.2.3 Windows 95

Windows 95 fue lanzado, como su nombre indica, en 1995, siendo una importante actualización porque se introdujeron mejoras significativas entre las que se pueden mencionar los profundos cambios en la interfaz gráfica y el pasar de una arquitectura multitarea cooperativa de 16 bits a usar una arquitectura multitarea apropiativa de 32 bits.

1.2.4 Windows 98

Lanzado el 25 de junio de 1998, se sigue tratando de un sistema operativo híbrido de 16 y 32 bits, pero con soporte mejorado para FAT32. Tenía un arranque basado en MS-DOS. Se le añadieron características prácticas a la interfaz de usuario como poder minimizar las ventanas haciendo clic en su botón de la barra de tareas o abrir inicio con un clic y los botones de navegación (atrás/adelante) en el explorador de Windows. También desde el punto de vista del hardware se añade soporte para DVD, puerto AGP para tarjetas gráficas, Unidades IDE y SCSI y admitía muchos medios de red, incluido Ethernet.

1.2.5 Windows 2000

Todos los sistemas operativos anteriores se encontraban basados en MS-DOS, sin embargo Microsoft también desarrolló desde 1993 otra línea de S.O. orientado a la empresa y la oficina llamado Windows NT. Windows 2000 se trataba de la 5 versión de esta línea de sistemas operativos, llamado originalmente como Windows NT 5.0. Lanzado en el 2000, fue la última versión de esta línea de Windows destinada únicamente a estaciones de trabajo y servidores. En 1999, Microsoft empezó a trabajar en los proyectos Neptune y Odyssey. El primero iba a ser la primera versión NT para consumidor, mientras que el segundo iba a ser el sucesor de Windows 2000. Sin embargo, en enero de 2000, Microsoft decidió cancelar los dos proyectos para fusionarlos en uno solo, Whistler, que acabaría siendo Windows XP en 2001. Desde entonces, Windows NT se distribuye tanto para consumidores como para empresas.

1.2.6 Windows XP

Como se ha mencionado anteriormente, se trata del sucesor de Windows 2000, perteneciente a la línea de Windows NT, pero a diferencia de las anteriores versiones, estaba orientado tanto a nivel empresarial como doméstico. Windows XP introdujo una serie de categorías nuevas, como

4. WINDOWS

pueden ser:

- Un ambiente gráfico más agradable.
- Secuencias más rápidas de inicio y de hibernación.
- Capacidad de desconectar un dispositivo externo sin necesidad de reiniciar el sistema.
- Uso de varias cuentas de usuario permitiendo que un usuario guarde el estado actual y las aplicaciones que tiene abiertas, haciendo que si otro usuario inicia sesión no se pierda esa información.
- Escritorio remoto (permitiendo abrir una sesión a través de la red).

Como curiosidad, se trata de la primera versión de Windows que utiliza la activación del producto para reducir la piratería del software.

1.2.7 Windows Vista

Tras el lanzamiento de Windows XP, Microsoft empezó el desarrollo del proyecto Longhorn, un proyecto muy ambicioso ideado originalmente como una actualización menor de Windows XP, pero nunca llegó a ver la luz en su versión final. Tuvieron que abandonar muchas de las innovadoras ideas que pretendían llevar a cabo, convirtiéndolo en Windows Vista. Para muchos considerado el peor sistema operativo de Windows, principalmente por sus requisitos hardware, que hacían que la novedosa interfaz Windows Aero hiciese que los tiempos de respuesta perjudicasen la experiencia de usuario en equipos modestos. Pero también consta de una serie de avances positivos, como por ejemplo la introducción de DirectX 10, unas APIs que permitían mejorar tanto en el rendimiento como la calidad de los juegos.

1.2.8 Windows 7

A diferencia del salto de Windows Vista en cuanto arquitectura y características con respecto a Windows XP, Windows 7 fue inicialmente una actualización incremental y focalizada de Windows Vista, permitiendo mantener un cierto grado de compatibilidad con aplicaciones y hardware en los que ya era compatible su antecesor. En el desarrollo de Windows 7, se dio importancia a la mejora de la interfaz gráfica para volverla más accesible e incluir nuevas características que permitan realizar las tareas de forma fácil y rápida, la mismo tiempo que hacerlo más rápido, ligero y estable. Añade una serie de características nuevas como el soporte para sistemas con múltiples tarjetas gráficas de distintos proveedores, soporte para discos duros virtuales, rendimiento mejorado en procesadores multinúcleo, mejor rendimiento de arranque y mejoras en el núcleo.

1.2.9 Windows 8

La siguiente versión de Windows (Windows 8), añade soporte de microprocesadores ARM, además de los tradicionales x86 de Intel y AMD. Por esta razón y por añadir compatibilidad con un gran número de dispositivos, incluido aquellos con pantallas táctiles, se cambió la interfaz gráfica de Windows eliminando el menú de inicio, existente desde Windows 95 como un estándar a la hora de presentar aplicaciones en interfaces gráficas, recibió muchas críticas al respecto hasta que el 2 de abril de 2014 Microsoft reconoció el error y anunció la implementación del menú de inicio en su siguiente versión. El 18 de octubre de 2013, Microsoft lanzó una actualización gratuita al sistema: Windows 8.1. Mientras que el 29 de julio de 2015, presentó su sucesor, Windows 10, orientado a integrar de una mejor forma el sistema operativo en todos los dispositivos, desde ordenadores, tabletas y hasta teléfonos inteligentes, destacando el regreso de uno de sus elementos más característicos, el ausente Menú Inicio.

1.2.10 Windows 10

La última versión de Windows hasta el momento, perteneciente a la línea de Windows NT. Ha traído grandes avances y cambios hasta el momento, como por ejemplo la inclusión del kernel completo de Linux de forma nativa, permitiendo que en un mismo S.O. se puedan ejecutar aplicaciones tanto de Linux como de Windows. Para la versión empresarial, Windows 10 Enterprise, se ofrece características de seguridad adicionales, como pueden ser establecer normativas para el cifrado de datos automático y bloquear selectivamente las solicitudes de acceso a los datos cifrados. Windows 10 también ofrece Device Guard, una característica que permite a los administradores reforzar la seguridad mediante el bloqueo de la ejecución de software que no está firmado digitalmente por un proveedor de confianza o Microsoft. Se introduce también con esta versión el navegador Edge basado en Chromium sustituyendo a Internet Explorer como el navegador predeterminado en Windows.

1.2.11 Windows 11

En la nueva versión de Windows 11 existe un cambio de estética notable a diferencia de su antecesor. Nuevo diseño y colocación del menú de inicio, rediseño de las aplicaciones preinstaladas, también en sus iconos, bordes redondeados. Permitirá instalaciones de aplicaciones de Android integradas en Microsoft Store. Las aplicaciones clásicas de Windows no estarán instaladas en el sistema operativo sino mediante la Microsoft Store. Tendrá una nueva aplicación de terminal, en la que podrás tener diferentes pestañas con el símbolo de sistema o PowerShell, todo en uno, y pudiendo personalizar su diseño. Cambio en el menú contextual más simple y con menos contenido. Nos permitirá personalizar escritorios virtuales, renombrarlos y diferenciarlos con distintos fondos de pantalla. Explorador de Windows más optimizado, más limpio y menos denso. Nos permitirá crear más de dos divisiones de pantallas. Mejora en la experiencia de juego añadiendo funcionalidades como AutoHDR, Direct Storage y acceso a Xbox. Refuerzo de seguridad a través de "Zero Trust" que protege los datos y al acceso en todos los dispositivos. Es más rápido y seguro para los profesionales de TI gracias a Microsoft Endpoint Manager, configuración en la nube, Windows Update para empresas, y Autopilot, que permite una integración natural de Windows 11. Por último mejor accesibilidad para personas con discapacidad y mejora en la velocidad eficiencia y experiencia de uso gracias a la interacción táctil e instrucciones de voz.

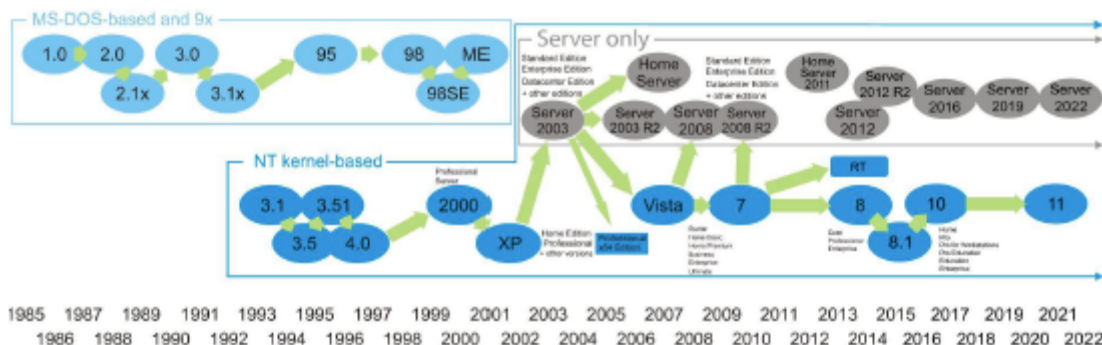


Figura 1.1: Cronología de los S.O. Windows²

4. SISTEMA DE ARCHIVO

1.Sistemas de archivo

Los sistemas de archivo (files systems en inglés) estructuran la información guardada en una unidad de almacenamiento (normalmente el disco duro). Para poder almacenar los datos en un disco, éstos se han de guardar respetando una serie de normas y restricciones que vienen impuestas por el sistema de archivos. Cada sistema operativo utiliza su propio sistema de archivos, aunque hay sistemas de archivos que son compatibles en diferentes versiones. El tipo de sistema de archivos se determina en el proceso de dar formato (formateo del disco). En un sistema de archivos hay dos tipos fundamentales de objetos: los directorios y los archivos.

Los archivos son los objetos encargados de contener los datos, mientras que los directorios (también llamados carpetas en Windows) son los objetos cuya misión principal es permitir una mayor organización de los archivos dentro del dispositivo. Un directorio es un contenedor que puede albergar archivos y, a su vez, otros directorios dentro de él. De esta forma, se puede llegar a crear una jerarquía en forma de árbol que simplifica enormemente la tarea de organizar y estructurar los archivos dentro de un disco. Los sistemas de archivos disponen de métodos para crear, mover, renombrar y eliminar tanto archivos como directorios.

1.1 Archivos

Los archivos son un mecanismo de abstracción. Es la forma de almacenar información en el disco y poder volverla a leer más adelante sin que el usuario tenga que preocuparse por la forma y lugar físico de almacenamiento de la información así como del funcionamiento real de los discos. Las reglas para nombrar los archivos varían de un sistema de archivos a otro, pero en general, todos los sistemas operativos permiten cadenas de hasta ocho caracteres como nombre de archivo aunque en la mayoría de los sistemas actuales se permiten longitudes mayores (255). Algunos de estos sistemas de archivo diferencian entre mayúsculas y minúsculas (como es el caso de Linux) mientras que, para otros (como Windows), no existe tal diferencia. La mayoría de los sistemas operativos utilizan nombres de archivos con dos partes separadas por un punto (por ejemplo: documento.doc). A la parte posterior se le denomina extensión de archivo y, por lo general, indica el tipo de archivo que representa. Así pues la estructura típica de nombre de archivo se podría asociar al siguiente formato: Nombre.extensión

Junto con el nombre de archivo, el sistema operativo almacena también diferente información referida al propio archivo como los atributos, la fecha y hora de creación o modificación, el tamaño, los permisos que definen su funcionamiento, etc.

En las ventanas capturadas en la siguiente página pueden apreciarse algunas características y permisos de utilización de archivos tanto en Windows como en Linux Ubuntu.

Los comodines.

En cualquier sistema de archivos existen formas de recortar y facilitar las cosas más usuales, entre las que se encuentra facilitar la selección de ficheros. Para ello, se disponen de los comodines que son de dos tipos:

1. `"*"`. Sustituye a TODOS los caracteres, delante, detrás o en medio del nombre.
2. `"?"`. Sustituye a un CARÁCTER para que coincida con el resto que esté escrito.

EJEMPLOS:

`*data` Se refiere a cualquier nombre de archivo terminado en 'data'.

`note*` Se refiere a cualquier archivo que comience con 'note'.

`*pazos*` Alude a cualquier nombre de archivo que contenga el texto 'pazos' en cualquier lugar del archivo.

`memo?` Identifica cualquier nombre de archivo que conste de 'memo' seguido por un carácter cualquiera (por ejemplo, 'memo1', pero no 'memoria').

`*old?` Identifica cualquier nombre de archivo que finalice en 'old' seguido por un carácter cualquiera.

Tipos de Archivo.

Los archivos se pueden dividir en dos grandes grupos: los ejecutables y los no ejecutables. La diferencia fundamental entre ambos es que los primeros están creados para funcionar por sí mismos y los segundos almacenan información que tendrá que ser utilizada con ayuda de algún programa. Por ejemplo, WINWORD.EXE es el archivo ejecutable que nos permite lanzar Word para editar texto, y DOCUMENTO.DOCX sería el archivo de datos que necesita del Word para poder ser visualizado o modificado.

Los archivos se pueden clasificar por categorías dependiendo de la temática o clase de información que almacenen. Entre dichas categorías se encuentran:

- SISTEMA. Son archivos necesarios para el funcionamiento interno del sistema operativo. No es recomendable moverlos, editarlos o variarlos de ningún modo porque pueden afectar al buen funcionamiento del sistema (pueden contener cat, ini, inf, msi, sys como extensiones)
- AUDIO. Son todos los que contienen sonido (no solo música). Entre sus extensiones se encuentran mp3, wma, cda.
- VÍDEO. Los formatos de vídeo no sólo contienen imágenes sino también el sonido que las acompaña. Son archivos de vídeo los que utilizan como extensión mp4, avi, mpeg, wmv, etc.
- COMPRIMIDOS. Los formatos de compresión son de gran utilidad a la hora de almacenar la información, ya que hacen que ocupe el menor espacio posible y agrupar varios ficheros en uno solo. Muy práctico también para enviar información por la web. Entre sus extensiones se encuentran rar, zip, tar, tgz.
- IMÁGENES. Las imágenes y sus formatos utilizan unos métodos de representación distinto, ofreciendo algunos mejor calidad que otros. Son archivos de imágenes los que tienen como extensión jpg, bmp, pcx, tif, gif, etc.
- TEXTO. Dentro de los documentos de texto hay que diferenciar entre el texto plano y el enriquecido. Es decir, entre los formatos que sencillamente guardan las letras (txt, log...) y los que permiten asignarles un tamaño, fuente, color, etc. (doc, pdf, ...).
- IMÁGENES de CD/DVD. Se utilizan para guardar en un archivo único todo lo incluido dentro de un CD/DVD. Entre sus extensiones se encuentran iso o img.

Los permisos.

Cuando se establecen los permisos sobre un archivo, se define el acceso de un usuario o de un grupo de usuario a dicho archivo.

Estos permisos sólo pueden establecerlos y cambiarlos el administrador, el propietario o aquel usuario que haya recibido el permiso correspondiente.

Operaciones comunes con archivos.

4. SISTEMA DE ARCHIVO

Entre las operaciones comunes que se pueden realizar con los archivos se encuentran:

- CREAR. Con esta operación se añade un nuevo archivo. En este momento se le deberá asignar un nombre que no podrá ser igual a otro que ya exista y que no podrá tener determinados caracteres: \, /, :, *, ?, ", <, >, |.
- ABRIR. En esta operación se identifica y localiza un archivo existente para que los usuarios o el propio sistema pueda trabajar con él.
- CERRAR. Esta operación se utiliza para indicar que se va a dejar de utilizar un archivo determinado.
- COPIAR. Consiste en crear una copia del archivo en otra ubicación quedando también el archivo en su lugar de origen.
- MOVER. Con esta operación se quita el archivo del lugar de origen y se lleva a otra ubicación de destino.
- RENOMBRAR. Consiste en cambiar el nombre al archivo.
- ELIMINAR. Es borrar el archivo de la ubicación en la que se encuentre.

1.2. Los Directorios.

Los directorios son una división lógica de almacenamiento de archivos o de otros subdirectorios. Los directorios constituyen una estructura jerárquica en forma de árbol. En cualquier momento, el usuario se encuentra en un determinado directorio y, a menos que se indique otra cosa, todos los archivos se buscan o se crean en ese directorio.

Trayectorias o rutas de acceso.

En los sistemas de archivos jerárquicos se declara normalmente la ubicación precisa de un archivo con una cadena de texto llamada ruta (path). Una ruta viene dada por una sucesión de nombres de directorios y subdirectorios, ordenados jerárquicamente de izquierda a derecha, separados por un carácter especial ("/" en Linux y "\" en Windows) y que termina con el nombre del archivo. En todo sistema de archivos hay un directorio especial llamado raíz [root] que es el directorio que contiene todos los demás directorios y archivos. Se identifica por la barra inclinada (/).

Desde este directorio es desde el que se parte cuando se busca un archivo mediante una ruta de acceso absoluta. Cuando se usa una ruta de acceso relativa, el archivo se busca partiendo del directorio en el que se esté trabajando o directorio activo.

Las rutas relativas carecen de carácter inicial y, por tanto, no parten del directorio raíz, sino del activo. Por ejemplo bin/passwd o local/bin, frente a sus correspondientes rutas absolutas /bin/passwd o /usr/local/bin.

Cada directorio o carpeta, una vez que es creado, está vacío; es decir, no tiene ni subdirectorios ni archivos o ficheros que cuelguen de él. Pero al crear un directorio, el sistema operativo genera dentro de él dos subdirectorios automáticamente que son dos entradas especiales. Son las siguientes:

. [punto]. Este subdirectorio hace referencia al contenido del propio directorio.

.. [punto, punto]. Este subdirectorio hace referencia al directorio padre. El directorio padre es aquel que se encuentra jerárquicamente por encima del directorio en el que estamos situado. El directorio raíz no contendrá este fichero porque encima de él no hay ningún otro directorio en la jerarquía.

Una trayectoria es la que se construye partiendo desde donde estamos y que baja por la estructura jerárquica hasta llegar al directorio o fichero deseado. En las siguientes capturas se muestran dos ejemplos de enrutamientos absolutos y relativos respectivamente.

Los permisos.

Cuando se establecen los permisos sobre un directorio, se define el acceso de un usuario o de un grupo de usuario a dicho directorio y sus archivos. Estos permisos sólo pueden establecerlos y cambiarlos el administrador, el propietario o aquel usuario que haya recibido el permiso correspondiente.

Una vez establecidos los permisos, afectarán a los archivos y subdirectorios que dependan de él, tanto los que se creen posteriormente como los que ya existían previamente. A este hecho se le denomina herencia. Si no se desea que se hereden los permisos, deberá indicarse expresamente cuando se indiquen los permisos.

Operaciones comunes con directorios.

Entre las operaciones comunes que se pueden realizar con los directorios se encuentran:

- CREAR. Con esta operación se añade un nuevo directorio. En este momento se le deberá asignar un nombre que no podrá ser igual a otro que se encuentre en la misma ubicación y que no podrá tener determinados caracteres: \, /, :, *, ?, ", <, >, |.
- COPIAR. Consiste en crear una copia del directorio en otra ubicación quedando también el directorio en su lugar de origen.
- MOVER. Con esta operación se quita el directorio del lugar de origen y se lleva a otra ubicación de destino.
- RENOMBRAR. Consiste en cambiar el nombre al directorio.
- ELIMINAR. Es borrar el directorio de la ubicación en la que se encuentre.
- DESPLAZARSE POR EL ÁRBOL DE DIRECTORIOS. Con esta operación se puede desplazar de un directorio a otro.

1.3. Tipos de Sistemas de Archivos.

Como se acaba de observar, el sistema de archivos es la estructura lógica más adecuada y eficiente para el manejo de archivos y directorios.

El diseño del sistema de archivos tiene una gran influencia en la eficacia, rendimiento, seguridad, flexibilidad y capacidad de crecimiento de los almacenamientos en disco, y, por tanto, en el rendimiento del propio sistema operativo.

Todo sistema operativo incorpora un sistema de archivo que determina la forma en que se almacena la información en el soporte y, en consecuencia, la manera en que va a ser recuperada con posterioridad. Así, diferentes sistemas operativos pueden utilizar también diferentes sistemas de archivo. Existen grandes diferencias entre cada uno de ellos, sobre todo en cuanto a seguridad y a la capacidad de rendimiento.

Los tradicionalmente más utilizados son:

- **FAT - [File Allocation Table].** Es el sistema de archivos tradicional de MS-DOS y las primeras versiones de Windows. Por esta razón, es considerado como un sistema universal, aunque padece de una gran fragmentación³ y es un poco inestable. Los sistemas de archivo FAT imponían a los usuarios la limitación de 8 + 3 caracteres para los nombres de archivo y directorio y no distingue entre mayúsculas y minúsculas en los nombres de archivos/directorios. Desde la aparición de Windows 95, con su sistema de archivo FAT ampliado [V-FAT o FAT 32], ya no existe la limitación en cuanto a la longitud de los nombres admitiéndose archivos de hasta 256 caracteres. Pero ni la seguridad ni la estabilidad del sistema se pudieron mejorar. Sólo admite particiones de 4 GB como máximo.
- **NTFS - [NT File System]** Es el nuevo sistema de Windows, usado a partir de Windows NT y extendido su uso a partir de Windows 2000, XP, Vista, Windows 7 y 8. Es muy estable y seguro aunque necesita un cierto tiempo de acceso a los archivos. El problema es

4. SISTEMA DE ARCHIVO

que es privativo, con lo cual otros sistemas operativos no pueden acceder a él de manera transparente. Desde Linux sólo se recomienda la lectura, siendo la escritura en estas particiones un poco arriesgada. Admite particiones de hasta 256 TB.

- **ext2** - Hasta hace poco era el sistema estándar de Linux. Tiene una fragmentación bajísima, aunque es un poco lento manejando archivos de gran tamaño.
- **ext3** - Es la versión mejorada de ext2, con previsión de pérdida de datos por fallos del disco o apagones. En contraprestación, es totalmente imposible recuperar datos borrados. Es compatible con el sistema de archivos ext2. Actualmente es el más difundido dentro de la comunidad GNU/Linux y considerado como el estándar. Admite particiones de hasta 16 TB
- **ext4** - Es un sistema de archivos con registro por diario (en inglés Journaling) en el que se almacena la información necesaria para restablecer los datos en caso de pérdidas de información. La principal novedad en ext4 es Extent, o la capacidad de reservar un área contigua para un archivo; esto puede reducir y hasta eliminar completamente la fragmentación de archivos. Es el sistema de archivos por defecto para los sistemas Linux actuales. Admite particiones de hasta 1 EB(Exabyte).

El problema de almacenamiento no contiguo de archivos se denomina fragmentación, se produce debido al almacenamiento de archivos en dispositivos como disco duro y memoria RAM por el uso del computador. La fragmentación es un problema que surge debido al ordenamiento interno de los datos en algunos sistema de archivos. Se da muy comúnmente en el sistema operativo Windows aunque también afecta a otras plataformas pero en una escala mucho menor.

- **swap** - Es el sistema de archivos para la partición de intercambio de Linux. Todos los sistemas Linux necesitan una partición de este tipo para cargar los programas y no saturar la memoria RAM cuando se excede su capacidad. En Windows, esto se hace con el archivo pagefile.sys en la misma partición de trabajo, con los problemas que conlleva.

En cuanto al tamaño de esta partición, es una costumbre extendida que ésta sea igual que el tamaño de la memoria RAM disponible. Por ejemplo, si tenemos 256 MB de memoria RAM, nuestra partición de intercambio será de 256 MB. Sin embargo esto únicamente es aplicable a tamaños de memoria de hasta 2GB. Si tenemos más memoria tenemos que utilizar la siguiente regla: entre 2 GB y 4 GB, utilizaremos como tamaño del swap la mitad del valor de la RAM; mientras que si tenemos más de 4 GB, utilizaremos una swap de sólo 2 GB.

Otros sistemas de archivos de uso común se reflejan en la siguiente tabla:

MacOS: HFS (Sistema de Archivos Jerárquico)/ MFS (Sistemas de Archivos Macintosh)
OS/2 HPFS (Sistema de Archivos de Alto Rendimiento)
Sun Solaris UFS (Sistema de Archivos Unix)
CD-ROM ISO 9660/Joliet

2. Secuencia de Arranque de un ordenador (BIOS) , Particionado del Disco y Sectores Críticos (MBR y BR) , y Gestores de Arranque.

No solo para comprender correctamente la secuencia de arranque del ordenador sino para preparar un proceso de instalación adecuado es necesario conocer la estructura lógica del disco que viene determinada por tres conceptos fundamentales:

a) Particiones.

La Partición es el concepto más elemental en el trabajo con discos. Una partición hace de un disco o de una determinada parte de él lo que realmente debe ser un medio de almacenamiento de datos. El disco duro no es 'nada' sin particiones. La partición hace del producto físico del disco duro una unidad de almacenamiento del ordenador. Se distinguen dos tipos de particiones básicas posibles:

1. **Particiones PRIMARIAS:** Son reconocidas por la BIOS [Sistema Básico de Entrada/Salida] como capaces de realizar un inicio. Es decir, el sistema puede iniciarse desde esa unidad. Las particiones primarias no pueden dividirse y existen como máximo cuatro en cada disco duro, de las que sólo una puede estar activa. Ésta, la partición activa, será la que determine el sistema operativo que se cargará tras el postrealizado por la BIOS.
2. **Particiones EXTENDIDAS:** Deben hacerse cargo de las áreas del disco duro que no contienen particiones primarias y que tampoco deben permanecer desocupadas. La partición extendida puede continuar subdividiéndose en subgrupos más pequeños denominados UNIDADES LÓGICAS. Cada disco duro sólo permite una partición extendida de un área continua. De utilizarse esta partición, el número de primarias disponibles por disco disminuye en una unidad, es decir se puede tener como máximo una partición extendida y tres primarias.
3. **Unidades LÓGICAS:** Las unidades lógicas dividen particiones extendidas. Se pueden crear tantas unidades lógicas como letras dispone el alfabeto, exceptuando las de las unidades de disquete [A/B], lectores CD/DVD [D] y las particiones primarias ya creadas (como la del Sistema: C). Con las unidades lógicas se pueden dividir los discos duros grandes para mantener de forma sencilla las estructuras de los directorios.

b) Espacio sin particionar o No Asignado.

Se denomina espacio sin particionar al que no se ha asignado a ninguna partición y que por lo tanto NO puede utilizarse para el almacenamiento de la información.

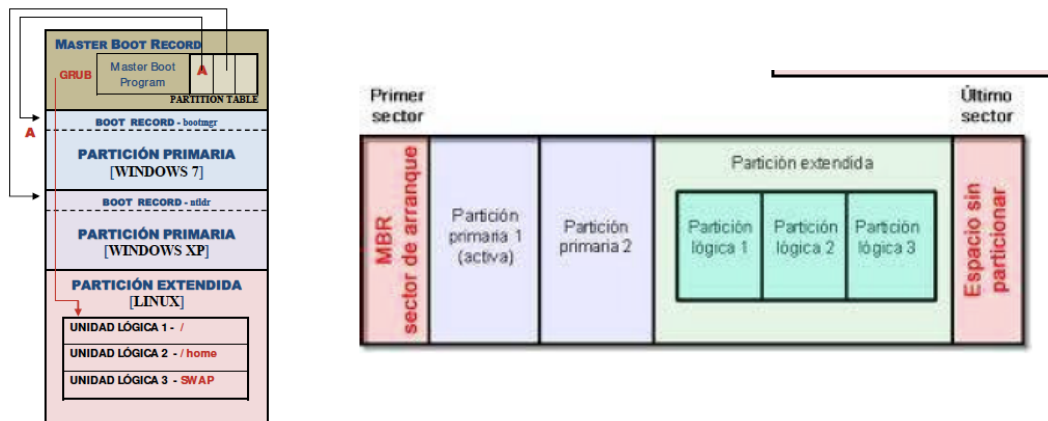
c) Sectores críticos.

Al margen de las particiones, los discos duros disponen de unos Sectores Críticos que deben tenerse en cuenta durante el Proceso de Arranque:

1. **SECTOR DE ARRANQUE MAESTRO (Master boot Record ® MBR):** Se crea cuando se genera la primera partición de un disco duro. Está en el primer sector del disco ocupando siempre la pista 0, cabeza 0 y sector1. Contiene la tabla de Particiones del disco y una pequeña cantidad de código ejecutable. Éste examina la tabla de particiones e identifica la partición ACTIVA del sistema, carga en memoria una copia de su sector de arranque y transfiere el control a la misma.

4. SISTEMA DE ARCHIVO

2. SECTOR DE ARRANQUE DE LA PARTICIÓN (Boot Record ® BR). Sólo disponible en las particiones primarias. Contiene información sobre el sistema de Archivos usados en esa partición (FAT, HTFS, HPFS, etc.) y almacena el IPL (Initial Program Load) que se utiliza para cargar los archivos del kernel del sistema operativo en memoria.



A partir de estos conceptos, deben ser tenidos en cuenta los siguientes aspectos, en relación con la Administración de los discos duros:

ADMINISTRADORES DE INICIALIZACIÓN. Gestores de Arranque.

Cada sistema operativo se puede activar durante el proceso de inicio [BOOT] a través del llamado BootManager (Administrador de Inicialización). Cada sistema dispone de su programa de arranque y un archivo de configuración donde se encuentran los parámetros que indican al cargador la información a mostrar:

SISTEMA OPERATIVO	GESTOR DE ARRANQUE	ARCHIVO DE CONFIGURACIÓN
Windows XP	ntldr	boot.ini
Windows Vista/7/8	bootmgr	bcd.log
Linux	grub	/boot/grub/grub.cfg

BCD [Boot Configuration Data] es un archivo binario y en consecuencia NO puede ser editado con un editor/procesador de textos. El sistema pone a disposición del usuario el archivo BCDEDIT.EXE que se encuentra en C:>Windows\System32 . Pero este archivo es un comando a ejecutar desde una consola DOS con muchos parámetros que lo convierte en una alternativa altamente compleja. Se recomienda la modificación de bcd.log mediante el programa gratuito: [easybcd](#) .

La secuencia de arranque de un ordenador consiste en iniciar el sistema desde un estado de apagado del PC. Este proceso se denomina bootstrapping o booting. La secuencia que se sigue se corresponde con la descripción de los siguientes pasos:

1. En primer lugar, desde una memoria no volátil del tipo EEPROM o memoria flash se carga el programa BIOS (Basic Input/Output System). Este programa comprueba el estado del hardware y determina su configuración. Este proceso se denomina POST (Power On Self Test - Prueba automática de encendido).
2. Posteriormente, si no hay errores, se carga en memoria un código llamado bootstrap o boot loader, que se encuentra almacenado en un CD/DVD, en un dispositivo USB o en el MBR (Master Boot Record) que se localiza al principio del disco duro (pista 0, sector 0).
3. Si se arranca desde el disco duro, el código del MBR examina la tabla de particiones, identifica la partición activa, lee el sector de arranque de la partición, y ejecuta el código almacenado en ese sector de arranque que posibilita la carga del sistema operativo en memoria. Si se iniciase desde otro dispositivo diferente al disco duro, se accederá directamente a este programa cargador almacenado al principio de la partición de arranque.
4. Una vez cargado el sistema operativo, este comienza con un test del sistema de archivos, crea las estructuras de datos internas necesarias para el funcionamiento del sistema y comienzan a arrancar los procesos del sistema operativo.

Una vez arrancados estos procesos el equipo ya está en funcionamiento y en espera de que el usuario lo utilice y empiece a ejecutar sus propios procesos.

4.MAQUINAS VIRTUALES

1.1. Definición.

Una máquina virtual es un programa o aplicación que emula a un ordenador, lo que permite utilizar más de un sistema operativo al mismo tiempo en el mismo equipo.

La máquina virtual está formada por una BIOS y un conjunto de recursos hardware (memoria, procesador, disco duro virtual, etc.) que se utilizan como si fuera la máquina física. Dentro de una máquina virtual se puede instalar cualquier sistema operativo, siempre y cuando el programa para virtualizar soporte este sistema operativo. Desde las máquinas virtuales se puede imprimir, usar los dispositivos USB, navegar por la red, etc. Así, las máquinas virtuales se pueden utilizar para:

- Probar sistemas operativos nuevos antes de instalarlos realmente sobre la máquina física.
- Para instalar aplicaciones que no soporta nuestro sistema operativo actual, pero que la queremos probar sin tener que cambiar de sistema operativo.
- Si se ha cambiado a una versión superior de un sistema operativo, pero se desea seguir utilizando una aplicación que ya no funciona en el nuevo entorno por problemas de compatibilidad.

Las máquinas virtuales son una forma más sencilla y más flexible de instalar varios sistemas operativos en un mismo equipo informático. Además, sobre una máquina real puede haber varias máquinas virtuales ejecutándose simultáneamente.

Si bien, cuando se habla de virtualización, hay que tener en cuenta la diferencia entre emulador y máquina virtual. El ordenador puede trabajar bajo ambos conceptos que, aunque puedan parecer similares, no son lo mismo:

- Una máquina virtual es un software que permite emular un sistema operativo y, por tanto, ejecutar programas usando los componentes y periféricos de la máquina real.
- Un emulador es un software que permite ejecutar programas en una plataforma distinta para la que fueron creados.

Cuando se trabaja con máquinas virtuales, se deben tener claro dos conceptos en lo que a sistemas operativos se refiere:

1. Sistema Operativo ANFITRIÓN [Host]. Es el sistema operativo que está instalado en la máquina real. Sobre el sistema operativo anfitrión se instalará el programa o aplicación que permitirá que se instalen otros sistemas operativos.
2. Sistema Operativo HUÉSPED o INVITADO [Guest]. Es el sistema operativo que se instala en el programa de la máquina virtual. Sobre un mismo sistema operativo anfitrión se pueden tener instalados varios sistemas operativos invitados.

La instalación de una máquina virtual sobre un sistema operativo anfitrión no afecta a éste. Para el sistema operativo anfitrión una máquina virtual se comporta como una aplicación más instalada en él.

Características.

Algunas características de las máquinas virtuales son:

- Las distintas máquinas virtuales pueden ejecutar cualquier sistema operativo o aplicación sin que la ejecución afecte a otras máquinas virtuales.
- El sistema operativo que está instalado sobre el ordenador (físico) se llama anfitrión y el sistema operativo que está instalado en la máquina virtual se denomina invitado. La forma de instalar un sistema operativo invitado es igual que cuando se instala sobre una máquina física.
- Las máquinas virtuales tienen discos duros virtuales que para la máquina anfitriona (es decir, para la máquina real) son simplemente ficheros de datos que se pueden copiar, mover, eliminar, etc.
- El software que sirve para crear máquinas virtuales permite ejecutar varios sistemas operativos dentro del mismo hardware, compartiendo simultáneamente los recursos sin interferencias.
- La virtualización permite que en un solo ordenador haya máquinas virtuales con distintos servicios o aplicaciones. Por ejemplo, el correo electrónico, DNS, DHCP, etc. Gracias a la virtualización se aprovechan los ordenadores y se deja a un lado el principio de una aplicación por ordenador.

1.2. Ventajas e Inconvenientes.

Ventajas

- Permiten probar sistemas operativos, aplicaciones y configuraciones sin interferir con otras aplicaciones o sistemas operativos existentes. La alternativa a la creación de una máquina virtual, sería crear una partición con el arranque de su propio sistema operativo.
- No es necesario cargar e instalar un gestor de arranque dual para ejecutar varios sistemas operativos, se pueden ejecutar distintos sistemas simultáneamente.
- El sistema operativo virtualizado en la máquina virtual es totalmente independiente del sistema operativo de la máquina anfitriona.
- Sus reinicios son más rápidos que los de las máquinas reales.
- Ofrecen mayor facilidad para establecer copias de seguridad y recuperarlas tras un desastre. Basta con copiar y pegar un fichero.
- El sistema operativo virtualizado se puede tratar como un sistema operativo portátil, ya que se puede llevar en un dispositivo USB y arrancarlo en cualquier ordenador.
- Permiten guardar el estado en el que se encuentran y recuperarlo en otro momento gracias a las snapshots o instantáneas.
- Ofrecen mayor facilidad para crear entornos de pruebas

4. MÁQUINAS VIRTUALES

Desventajas

- Se necesitan más recursos hardware, sobre todo memoria RAM y espacio de almacenamiento de disco duro ya que van a ser compartidas por el sistema anfitrión y cada una de las máquinas virtuales en uso. Además se recomienda un microprocesador y una placa base compatibles con virtualizaciones.
- Se vuelven inestables a nivel de velocidad de ejecución si funcionan varias máquinas virtuales a la vez, aunque mantienen su estabilidad a nivel de resultados.
- Ralentizan el sistema porque se ejecutan de forma más compleja, ya que hay que pasar a través de la máquina real para ejecutar los programas.

Software [Propietario y libre] para crear máquinas virtuales.

Hay muchas empresas que ofrecen productos para crear y utilizar máquinas virtuales, algunos productos son gratuitos y se ajustan a una licencia freeware y otros son de pago con licencia de software propietario. Algunos de estos programas se comentan a continuación:

a) SOFTWARE PROPIETARIO:

1. VMware. Es un sistema propietario de virtualización. Los productos VMware son los más utilizados por empresas en todo el mundo, la mayoría son de pago y tienen licencia propietaria. VMware dispone de varios productos:
 - VMware Workstation. Es una versión de pago de las más utilizadas que soporta cerca de 200 sistemas operativos incluidos Windows 7, Ubuntu, Open Suse, etc.
 - VMware Player. Es un producto gratuito que permite correr máquinas virtuales creadas con otros productos de VMware, pero no permite crearlas él mismo.
 - VMware Server ESXi. Es el líder del mercado en la virtualización de servidores y soporta sistemas con hasta 32 procesadores y 64 Gb de memoria.
2. Virtual PC: Es un programa propietario desarrollado por Connectix y comprado por Microsoft para crear máquinas virtuales en equipos de sobremesa o portátiles que no sean servidores. Existen versiones para instalar en sistemas operativos anfitriones Windows de Microsoft y Mac OS X de Macintosh, pero no existe ninguna versión para ser instalado en sistemas operativos anfitriones Linux.

b) SOFTWARE LIBRE:

- Virtual Box: Desarrollado por Innotek GMBH, fue adquirido en 2008 por Sun Microsystems Inc, que a su vez fue comprado por Oracle en 2010. VirtualBox se puede ejecutar en multitud de sistemas operativos como Windows, Linux, OS X, Solaris, etc. Existen dos versiones de VirtualBox:
 1. Versión ejecutable, Oracle VM VirtualBox, es propietaria y gratuita para uso personal, de evaluación y académico.
 2. Versión de software libre (Open Source) llamada VirtualBox OSE (Open Source Edition, Edición de código abierto) que tiene licencia GNU pero con algunas limitaciones.
- Qemu: Es un emulador libre y gratuito de procesadores que también dispone de capacidades de virtualización dentro de un sistema operativo. Admite como sistemas operativos anfitriones Windows y Linux. El programa no dispone de interfaz gráfica, pero existe otro programa llamado QEMU manager que hace las veces de interfaz gráfica para Windows.

Creación de máquinas virtuales.

Se analizará con detenimiento el uso de VirtualBox instalándolo sobre un sistema anfitrión Windows 7 ya que sería demasiado tedioso mostrar el funcionamiento de todas las máquinas virtuales comentadas en el apartado anterior. En cualquier caso, tanto el proceso de instalación, como de configuración e instalación del sistema operativo huésped es muy parecido. Por otro lado se valora muy positivamente el hecho de que se trate de software libre y de su amplia distribución y uso en el sector.

1.5.1. Instalación.

Lo primero será descargar el programa, que es gratuito. Para proceder a la instalación del programa se inicia el proceso haciendo doble clic sobre el archivo descargado. Aparecerá como primera ventana, la pantalla de bienvenida. En la actualidad, los procesos de instalación de cualquier programa suelen ser muy sencillos y, la mayoría de las veces, basta con ir haciendo clic en el botón CONTINUAR (NEXT si se está utilizando una versión en inglés). Entre las pocas decisiones que habrá que tomar durante la instalación de este programa figuran determinar dónde queremos que se instale y qué aplicaciones del programa deseamos instalar. Por lo general, la instalación se realiza en la ubicación propuesta por VirtualBox (en Program Files). La siguiente decisión a tomar será si se desea que el programa instale iconos en el escritorio y en la barra de tareas para que se pueda acceder más rápidamente a VirtualBox. Si se desea disponer de estas pequeñas ayudas, se marcarán las opciones pertinentes. En las siguientes pantallas se deberá contestar afirmativamente a todas las cuestiones. Finalmente, el programa mostrará una ventana donde indica que ha terminado el proceso de instalación y propondrá que se empiece a utilizar.

1.5.2. Creación.

Una vez instalada la aplicación, cuando se ejecuta VirtualBox, aparece la ventana principal sobre la que se debe hacer clic en el botón NUEVA para crear una máquina virtual. A continuación, mediante un pequeño pero preciso asistente, se irá introduciendo los parámetros de la máquina virtual que se desea crear para simular una determinada máquina real:

1. Primero se define el sistema operativo que va a utilizar la máquina virtual, su versión y el nombre que se le quiere asignar a la máquina. Por cada sistema operativo que se desee emular en VirtualBox, se debe crear una máquina virtual. Tampoco es apropiado intentar usar máquinas virtuales de 64 bits sobre equipos físicos de 32, pues surgirán problemas de compatibilidad. Al elegir el sistema operativo se está indicando qué parámetros de configuración (memoria, disco duro...) se asignarán por defecto. Esto es importante, ya que no necesitarán los mismos valores un sistema operativo DOS que un Ubuntu 11.10 o un Windows 7. De todas maneras, estas opciones que se indican por defecto pueden modificarse en las siguientes ventanas de creación de la máquina virtual.

4. MÁQUINAS VIRTUALES

2. Luego, se pide seleccionar la cantidad de memoria RAM que se va a reservar para la máquina virtual. El mínimo requerido es 512 MB, pero esta cantidad puede ser modificada hasta el máximo disponible por el anfitrión, aunque en este caso se colapsaría el sistema real al dejarlo sin memoria. Existe una barra que cambia su color gradualmente de verde a rojo para indicar el grado de disponibilidad que tendría el sistema anfitrión con cada configuración. Cuanto más nos adentremos en la zona roja, peor será el rendimiento del anfitrión, pues no tendrá memoria suficiente y perderá por ello velocidad.
3. A continuación se procede a dotar de disco duro al equipo virtual. Se debe tener en cuenta que los discos duros que utiliza la máquina virtual no son reales, sino ficheros que VirtualBox gestiona. Se podrá seleccionar uno creado con anterioridad o crear uno nuevo para la ocasión, si bien no hay que fiarse demasiado de la recomendación que se nos plantea sobre el tamaño del disco duro virtual, pues suele ser insuficiente. Para facilitar la tarea, el proceso de creación de una máquina virtual dispone de un asistente para crear el disco duro:
 - Lo primero será elegir si se desea Crear un disco duro virtual ahora o Usar un archivo de disco duro virtual existente. Normalmente se seleccionará la primera alternativa para asignar un disco duro nuevo a la máquina que se está creando.
 - Luego preguntará por el tipo de disco duro virtual. Se deberá elegir una de las siguientes opciones:
 - o VDI (VirtualBox Disk Image). Para usar este disco en sistemas VirtualBox.
 - o VMDK (Virtual Machine Disk). Que es compatible con VMware.
 - o VHD (Virtual Hard Disk). Compatible con VirtualPC.
 - o Parallels. Que es compatible con las máquinas virtuales del mismo nombre.
 - o QED y QCOW. Compatibles con máquinas virtuales QEMU.

Como no hay intención de compartir nuestro disco con otras máquinas virtuales, se elegirá la primera alternativa que es la que aparece seleccionada por defecto.

- En la siguiente ventana pregunta por la forma de gestionar el espacio de almacenamiento del disco. Existen dos posibilidades:
 - **Reservado dinámicamente.** El fichero real asociado al disco duro ocupa muy poco espacio y se va incrementando a medida que se introducen en él más datos y programas.
 - **Tamaño Fijo.** El fichero asociado ocupa desde el principio todo el espacio del disco duro asignado.

Lo adecuado, por lo general, es usar discos duros virtuales de tamaño dinámico, pues irán creciendo a medida que se necesita espacio, mientras que los de tamaño fijo tienen un valor constante y no modificable que desperdicia todo el espacio reservado que no se usa.

-El tamaño se escoge en la última ventana del asistente para la creación del disco duro de la máquina virtual, donde también se determina el nombre del fichero en el que se va a guardar.

- Si una vez definido el tamaño del disco e instalado el sistema operativo, se observa que hace falta aumentar el espacio asignado, se procederá de la siguiente manera:

- Con la máquina virtual a la que pertenece la imagen apagada, abrir la consola de Windows (INICIO/cmd) e ir a la carpeta donde esté instalado VirtualBox.
- Una vez allí hay que ejecutar el siguiente comando: `VBoxManage modifyhd <path> --resize <tamaño>`

Donde <path> es la ruta completa al fichero .vdi que representa al disco duro que queremos redimensionar, y <tamaño> es el tamaño (en Mb) al que queremos redimensionar el disco. Por defecto VirtualBox almacena los discos duros virtuales en la carpeta 'VirtualBox VMS' del usuario que crea la máquina virtual. Aquí se genera otra carpeta con el mismo nombre que la máquina creada donde ya aparece el disco .vdi. Por ejemplo si tuviéramos una imagen de disco llamado WindowsXP_Jose.vdi en "C:\Users\Administrador\VirtualBox VMS\WindowsXP_Jose" y quisiéramos redimensionarla a 12 GB el comando a ejecutar sería:

Si todo ha ido OK, se mostrará el progreso de cómo va aumentando el tamaño del disco virtual.

De momento VirtualBox solo permite redimensionar imágenes de disco que se hayan creado con tamaño dinámico. A partir de aquí, ya está creada la máquina virtual aunque sin sistema operativo.

1.6. Configuración de máquinas virtuales.

Una vez creada la máquina virtual pueden llevarse a cabo modificaciones de las opciones elegidas inicialmente, si se considera que las configuradas por defecto no son las adecuadas. Además de las principales características de la máquina, también hay otros detalles que se pueden cambiar después, como por ejemplo los dispositivos físicos que se desean que estén disponibles en la máquina virtual: audio, USB, tarjeta de red, etc.

Para cambiar la configuración de una máquina virtual en VirtualBox, después de seleccionar la máquina virtual que se desea modificar (pinchando en ella adquiere un fondo azul que indica que esa máquina es la seleccionada en ese instante, se debe hacer clic en el icono CONFIGURACIÓN. Entonces se mostrará una ventana con todas las configuraciones posibles a aplicar sobre la máquina seleccionada. Son muchos los parámetros configurables, a continuación se establece una relación de los que se consideran más útiles:

- En GENERAL, se da la posibilidad de decidir si compartir el portapapeles, de manera que sea el mismo para la máquina real y la virtual. Esto permitirá realizar el tan manido Copiar y pegar de los programas de ventanas.
- Otra de las posibilidades que se ofrece es habilitar la virtualización de la máquina hardware (virtualización por hardware), que permite trabajar sin tener que preocuparse por el fabricante del procesador que se utilice. Esta opción no siempre es posible, ya que algunas placas base no lo permiten, pero a veces esto puede solucionarse actualizando la BIOS de la placa en la web del fabricante o, en otros casos, activando en esa misma BIOS esa propiedad que, por lo general, viene desactivada como medida de seguridad.

La Virtualización en sí, es un proceso que puede ser emulado por software, con una carga sobre el procesador que influye en el rendimiento final, tanto del equipo Host como del equipo Guest (el sistema virtualizado). Para reducir esa carga extra e innecesaria, los fabricantes de procesadores Intel y AMD, han incluido una serie de instrucciones nuevas en sus procesadores durante los últimos años. Esas instrucciones están agrupadas bajo el nombre VT-x para Intel, y AMD-V para AMD.

Como puede verse en la imagen del software CPU-Z, el procesador de Intel que está en el equipo Host, posee el juego de instrucciones habilitado, por ello, es posible habilitar las opciones de Virtualización que responden al Hardware.

4. MAQUINAS VIRTUALES

¿Cuál es la ventaja/desventaja de estos mecanismos? Desventajas, más bien pocas. Un pequeño impacto en el rendimiento general del hardware del equipo Host. ¿Ventajas? Muchas. Desde el manejo del equipo virtual mucho más fluido, acceso más veloz a los dispositivos, y menos recursos utilizados por el Sistema Operativo del equipo Host gracias a que la mayoría de las instrucciones van directo al procesador y no pasan previamente por el Sistema Operativo del Host. La virtualización por Hardware ha demostrado enormes ventajas cuando se trata de esquemas de Virtualización de alto nivel.

También desde la configuración de SISTEMA en la ficha Placa Base, se puede cambiar el orden de arranque de la máquina como se muestra en la siguiente imagen: Es importante seleccionar el Chipset que corresponda con el de la Placa Base real del equipo (por defecto no siempre se incluye el correcto) para que funcionen componentes integrados como el audio o los USB.

Otro de los cambios relevantes que seguramente haya que hacer es activar una tarjeta de red. Se puede simular hasta cuatro tarjetas de red para cada máquina virtual instalada. Al usar el asistente de creación de la máquina virtual, se habilita el primer adaptador de red y se establece el protocolo NAT (Network Address Translation) que significa traslación de direcciones de red.

De esta manera, la máquina virtual puede conectarse al mundo exterior usando la red de la máquina anfitriona, aunque el resto de las máquinas físicas conectadas a la red no serán capaces de verla.

VirtualBox permite escoger entre los siguientes modos de conexión.

1. **No conectado.** VirtualBox muestra un adaptador de red pero sin conexión (cable desconectado).
2. **"Network Address Translation" (NAT):** Permite funcionalidad básica desde el sistema operativo Huésped. Navegar por internet acceder al correo, descargar ficheros, pero tiene bastantes limitaciones si tenemos que establecer conexiones con la máquina virtual. Toma la misma dirección IP que el equipo Anfitrión de forma que no será reconocible por otros equipos desde el entorno de red.
3. **Adaptador puente** Simula una conexión física real a la red, asignando una IP al sistema operativo Huésped. Esta IP se puede obtener por DHCP o directamente configurándola en el Sistema Operativo huésped.
4. **Red interna** Similar al Adaptador puente, se puede comunicar directamente con el mundo exterior con la salvedad de que ese mundo exterior está restringido a las máquinas virtuales conectadas en la misma red interna. Esta limitación viene justificada por seguridad y velocidad.
5. **Adaptador sólo-anfitrión** Es una mezcla entre los tipos "Adaptador puente" e "interna".

Por defecto aparece configurado el modo NAT. Esta configuración no es la más adecuada, ya que no tenemos "visibilidad" del sistema operativo huésped desde el sistema operativo anfitrión, y nos resulta muy útil tener una IP diferente para cada uno (aunque sea asignada por DHCP).

Elegimos la 2ª opción (Adaptador puente). En el campo nombre seleccionamos el interfaz que vamos a utilizar: Ethernet, AirPort, etc.

Es importante establecer que este dispositivo debe estar conectado para el correcto funcionamiento de la red entre el sistema operativo anfitrión y el huésped. Si elegimos Ethernet y no nos conectamos a una red mediante el cable no se podrán establecer conexiones entre los sistemas operativos.

- También se puede capturar instantáneas para su almacenamiento. Las instantáneas no son más que el estado en un momento determinado de las variables del sistema entre las que se encuentran los programas instalados. Esto permite volver a ese determinado estado cuando sea necesario. Es una posibilidad interesante, pero consume mucha memoria del disco real si creamos instantáneas cada poco tiempo. Para recuperar una instantánea creada, basta con seleccionarla desde el botón INSTANTÁNEAS y pulsar otra vez sobre el botón 'Restaurar Instantánea'. Se tomará referencia de tal circunstancia en el nombre de la máquina virtual de la lista de la izquierda y cuando se inicie se visualizará la instantánea capturada.
- A la hora de proceder a la instalación del sistema operativo se puede elegir entre montar el dispositivo físico o montar una imagen ISO. Esta última posibilidad es muy útil puesto que casi todas las distribuciones de cualquier sistema operativo pueden encontrarse en Internet en forma de imagen. Así, se pueden descargar y montar la imagen en el DVD e instalar el sistema operativo sin necesidad de utilizar físicamente el DVD de la máquina anfitrión.
- Para añadir una entrada USB a la máquina virtual hay que seleccionar la máquina virtual a configurar y hacer clic en el apartado USB. Probablemente indicará "Filtros de dispositivos: 0" porque todavía no se han instalado filtros. Se abre la ventana de configuración de USB haciendo dos veces clic.

Desde la nueva ventana desplegada primero se habilitan todos los controladores USB seleccionando las casillas correspondientes y luego hay que hacer clic en el símbolo del lápiz con la cruz verde para añadir un dispositivo USB. VirtualBox ofrecerá la lista de dispositivos que ha encontrado y habrá que seleccionar el que se corresponda con el dispositivo a montar. Luego se pulsa sobre ACEPTAR para guardar la configuración y se quita de forma segura el dispositivo USB del ordenador.

Por último, solo faltaría seleccionar el sistema Huésped en el cuadro izquierdo para arrancarlo pulsando una vez en la fecha INICIAR o dos veces sobre su nombre. Una vez iniciado el sistema, volver a conectar el lápiz USB al ordenador.

Desde la pestaña 'Dispositivos' del menú, seleccionar 'Dispositivos USB' y el dispositivo correspondiente. Se debe montar el dispositivo y abrirse una ventana con su contenido.

- Instalar "Guest Additions". Este paquete de configuración es un software provisto por Virtual Box que proporciona una mejor integración entre el sistema anfitrión y el "invitado". El Guest Additions ofrece:
 - Integración del cursor del mouse
 - Carpetas compartidas
 - Mejor soporte de vídeo
 - Ventanas integradas
 - Sincronización horaria
 - Portapapeles compartido
 - Los inicios de sesión automáticos

Es un paquete que depende del sistema operativo instalado en la máquina virtual por lo que no es posible instalarlo hasta no tener instalado y corriendo un sistema operativo. Una vez que el sistema se ha iniciado, clicar DISPOSITIVOS (en el menú superior) y seleccionar "instalar «guest additions»...".

4. MAQUINAS VIRTUALES

Su instalación no requiere tomar ninguna decisión, basta con pinchar en la opción de instalación y el asistente se encargará por sí mismo de realizar todos los pasos sin preguntar nada. Para que tengan efecto habrá que reiniciar la máquina virtual. Si no fuese así y tras hacer clic sobre la opción oportuna no se lanzase la aplicación, abriría que montar el CD/DVD con la imagen VBoxGuestAdditions.iso disponible en la carpeta 'C:\Program files\Oracle\VirtualBox\' y ejecutar manualmente la aplicación desde EQUIPO.

Se iniciará el instalador en el lado de la máquina virtual tal y como se muestra en la captura de ventana de la derecha. La instalación es rápida y fácil, y una vez terminada pedirá reiniciar el equipo virtual para disponer de todas las ventajas incluidas.

- Tan solo resta apagar la máquina virtual. Existen varias posibilidades:
 1. Desde el propio sistema operativo huésped, eligiendo la opción de apagado que ofrezca dicho sistema. Evidentemente éste (el Sistema Operativo) tiene que estar corriendo sobre la máquina virtual.
 2. Desde el menú MÁQUINA existen hasta cuatro alternativas:
 1. **Pausar**: Detiene momentáneamente la ejecución de la máquina virtual para reanudarla cuando sea necesario. Tendría el mismo efecto que un 'Pause' en cualquier reproductor de audio.
 2. **Reiniciar**: Reinicia el sistema operativo actualmente seleccionado en la máquina virtual. Se corresponde con una pulsación del botón RESET de la caja del ordenador.
 3. **Apagado ACPI**: Desconecta el ordenador transmitiendo una orden de apagado al sistema operativo que esté ejecutándose. Tiene el mismo efecto que el producido cuando se pulsa sobre el botón POWER o de apagado de la máquina física real.
 4. **Cerrar**: Equivale a un cierre organizado de la máquina virtual y, para ello mostrará una ventana con las tres siguientes opciones:
 1. **Guardar el estado de la máquina**: Con esta opción VirtualBox congela el sistema copiando en disco el estado actual del sistema. Cuando se inicie de nuevo se recupera como si el tiempo no hubiera pasado. Es la opción adecuada, y una de las grandes ventajas de las máquinas virtuales, para interrumpir un trabajo que está a medias sin tener que repetir todo el proceso.
 2. **Enviar señal de apagado**: Con esta opción Virtual-Box envía una señal de apagado al sistema de la máquina virtual quien, normalmente, procederá a un apagado controlado. Como si hubieras realizado tú mismo la instrucción de apagado desde dentro del sistema. Es la forma adecuada de terminar una sesión de trabajo.
 3. **Apagar la máquina**: En este modo de apagado, el sistema se comporta como si quitáramos el cable de corriente a la máquina virtual; se apaga de forma brusca sin salvar ningún cambio y pudiendo dañarse el sistema de archivos del sistema, por lo que se debe de utilizar de forma cuidadosa. Su uso no tiene mucho sentido si no se utiliza con la opción restauración de una instantánea o snapshot anterior de forma que se recupere el estado que teníamos en el momento de dicha instantánea. Es extremadamente útil cuando se están haciendo pruebas y desgraciadamente éstas no salen como se hubiera deseado.

1.7. Instalación de un Sistema Operativo en una máquina virtual.

Una vez creada la máquina virtual, como se comentó en el punto 1.5, sólo faltaría iniciar la instalación de un sistema operativo y configurarla para su rendimiento óptimo.

Para ello, una vez seleccionada la máquina virtual sobre la que se va a efectuar la instalación, seleccionar 'Almacenamiento', luego la unidad de CD/DVD ROM en el 'Árbol de almacenamiento' para tener la opción desde 'Atributos' de introducir un disco de instalación en la unidad de CD/DVD del equipo Host, y leerla directamente de ahí, o seleccionar una imagen ISO y montarla directamente para acceder desde la máquina virtual. Si se decide utilizar una imagen ISO habrá que seleccionar la ubicación donde se encuentre, de lo que se deduce que, evidentemente, hay que disponer previamente de dicha imagen en el disco duro anfitrión.

En la imagen de muestra (capturada en la parte superior del texto) se pueden observar las diferentes alternativas. Así se puede optar por elegir la unidad anfitrión «E» que se corresponde con el lector CD/DVD y que obligará a incluir un disco con el sistema operativo a instalar en dicha unidad, o seleccionar una imagen ya disponible (como las que se muestran) o seleccionar la imagen ISO de la ubicación adecuada (la primera de las opciones). Ahora ya se puede INICIAR la máquina virtual haciendo clic sobre el botón adecuado para que comience el proceso de instalación del sistema operativo elegido.

Una vez finalizado el proceso de instalación, se accederá al primer inicio de nuestro sistema virtual recién instalado: Se puede apreciar que el Windows Virtual funciona tal cual una versión usada en un equipo físico real.

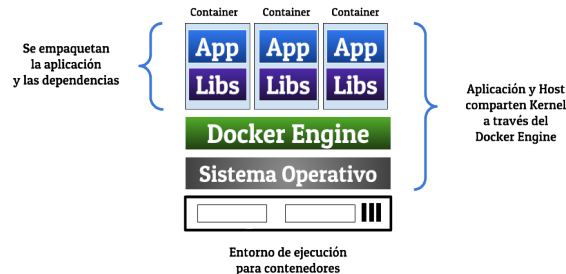
Cuando se hace clic dentro de la máquina virtual, VirtualBox despliega un mensaje informando sobre la 'integración del ratón'. Con las últimas versiones no es necesario liberar el ratón y éste se desplazará por la máquina huésped o por el anfitrión indistintamente. Si estuviere capturado por la máquina virtual, sólo podría utilizarse en su entorno y no rebasarlo para acceder a la máquina real. Para ello debería pulsarse la tecla CTRL (la ubicada a la derecha del teclado y que es conocida como tecla 'Host') al mismo tiempo que se desplaza el ratón por el escritorio.

4.DOCKER

Docker es una tecnología de virtualización "ligera" cuyo elemento básico es la utilización de **contenedores** en vez de máquinas virtuales y cuyo objetivo principal es el despliegue de aplicaciones encapsuladas en dichos contenedores. Es una evolución de la virtualización. Los contenedores (Linux Containers, LXC) se conocen en Linux desde hace mucho tiempo, pero una compañía (Docker) ha refinado esta tecnología simplificándola y reduciendo la carga que cada uno de estos contenedores representan para el sistema host.

Un contenedor es una forma de aislar un grupo de procesos dentro de un sistema Linux. Se consigue utilizando una serie de funcionalidades incorporadas en el kernel que permiten que estos procesos cuenten con una visión virtualizada del sistema cada uno con su propio sistema de ficheros y sus propios interfaces de red.

La gran diferencia entre una máquina virtual y un contenedor es que mientras la máquina virtual nos permite ejecutar un SO completo, un contenedor está diseñado para ejecutar únicamente una serie de procesos limitados, como una aplicación, un servidor web, etc. Su arquitectura general:



Esta tecnología también tiene algún inconveniente: Los contenedores son **más frágiles que las MMVV** y en ocasiones se quedan en un estado desde el que no podemos recuperarlos. No es algo frecuente pero ocurre y para eso hay soluciones como la orquestación de contenedores.

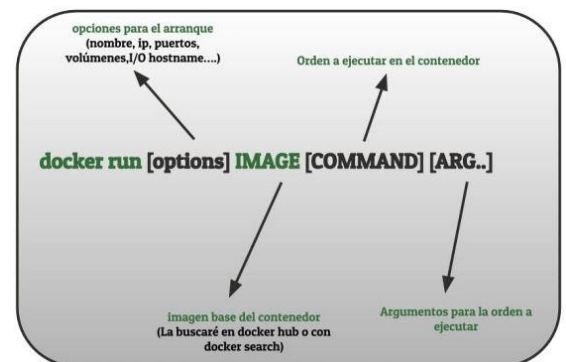
Componentes de Docker:

- **Docker Engine:** Es un demonio que corre sobre cualquier distribución de Linux y que expone una API externa para la gestión de imágenes y contenedores.
- **Docker Client:** Es el cliente de línea de comandos (CLI) que nos permite gestionar el Docker Engine. El cliente docker se puede configurar para trabajar con un Docker Engine local o remoto.
- **Docker Registry:** La finalidad de este componente es almacenar las imágenes generadas por el Docker Engine. Nos permite distribuir nuestras imágenes. Podemos instalar un registro privado, o hacer uso de uno público como Docker Hub.

Acciones con Docker:

- **docker pull nombre_imagen:versión** Descargará desde el repositorio una imagen con la versión indicada o la última versión (latest) si no indicamos versión.
- **docker run nombre_imagen:versión** Ejecutará un contenedor que usa como base una imagen, si no tenemos ésta, se descargará de manera automática. **docker run** tiene una sintaxis sencilla pero con multitud de opciones:

- e para establecer variables de entorno en la ejecución del contenedor.
- help para obtener ayuda de las opciones de docker.
- i para mantener la STDIN abierta en el contenedor.
- ip si quiero darle una ip concreta al contenedor.
- name para darle nombre al contenedor.
- net para conectar el contenedor a una red determinada.
- p para conectar puertos del contenedor con los de nuestro host.
- restart que permite reiniciar un contenedor si este se "cae".
- rm que destruye el contenedor al pararlo.
- t para que el contenedor que vamos a ejecutar nos permita un acceso a un terminal para poder ejecutar órdenes en él.
- u para establecer el usuario con el que vamos a ejecutar el contenedor.
- w para establecer el directorio de trabajo en un contenedor.



docker exec [opciones] nombre_contenedor orden [argum] Ejecutará órdenes en el contenedor. Es necesario que el contenedor esté en ejecución. Dos opciones: mediante terminal o por orden directa contra el contenedor. Opciones importantes:

- it si vamos a querer tener interactividad con el contenedor.
- u si quiero ejecutar la orden como si fuera un usuario distinto a root.
- w si quiero ejecutar la orden desde un directorio concreto.

docker ps Servirá para obtener información de los contenedores y sus arrancados. Opciones importantes: Los contenedores parados tienen el estado (Status) EXITED y los que están en ejecución (UP).

docker stop Servirá para detener el contenedor, ya sea por nombre o por ID.

docker rm Servirá para borrar el contenedor, ya sea por nombre o por ID. Si está en ejecución no se puede borrar salvo que usemos la opción -f.

docker start Servirá para iniciar un contenedor que estaba parado previamente, ya sea por nombre o por ID.

docker restart Servirá para reiniciar un contenedor que estaba en ejecución previamente.

docker images Servirá para mostrar por pantalla una lista de las imágenes que tenemos en nuestro sistema.

docker image rm Servirá para borrar una imagen que tenemos en nuestro sistema, ya sea por nombre o por ID.

Docker rmi Servirá también para borrar una imagen que tenemos en nuestro sistema, ya sea por nombre o por ID.

4.INSTALACION DE APLICACIONES LINUX

1.1 PAQUETES

Como ya se expresó anteriormente una distribución de Linux ofrece básicamente un conjunto de paquetes que contienen aplicaciones, utilidades, documentación y el propio kernel del sistema.

Estos paquetes no son más que ficheros con cierto formato que, manipulados por un comando u otra aplicación son capaces de instalarse en el sistema de acuerdo a las especificaciones que determinó su fabricante. Entre las especificaciones puede citarse a los ficheros y directorios que crea el paquete, el tamaño que ocupa una vez instalado, una descripción breve y otra más amplia de su utilidad, las dependencias que puede haber respecto a otros paquetes, etc.

Algunos paquetes son dependientes de plataforma como es el caso del que contiene al kernel, otros por el contrario se pueden instalar en cualquier arquitectura como son los que agrupan documentación. Usualmente los programas fuentes también se agrupan en paquetes independientes de los compilados.

En Linux no se utiliza un único sistema de paquetes, sino que podemos encontrar varias alternativas. Veamos ahora las más usadas:

1. **Paquetes RPM.** Los paquetes RPM (Red Hat Package Manager), deben su nombre a que fueron creados por Red Hat, aunque más adelante fue adoptándose por otras distribuciones como SUSE, Mandriva o Fedora.

Los paquetes se identifican por la siguiente sintaxis:

nombre-versión_aplicación-versión_paquete.arquitectura.rpm

Para instalar paquetes RPM en modo texto se usa el comando rpm.

Por ejemplo si se desea descargar un programa como Virtual Box se podría utilizar el siguiente comando:

```
rpm -i VirtualBox-4.1.4.1.18_78361_fedora17-1.x86_64.rpm
```

2. **Paquetes DEB.** Creados por la distribución Debian y usadas en muchas otras, sobre todo Ubuntu. Están mantenidos por una gran comunidad de desarrolladores y el proyecto está patrocinado por la empresa Canonical Ltd., que ofrece soporte y servicios para Ubuntu. Estos paquetes se gestionan mediante los comandos apt, aptitude y dpkg. También se puede instalar alguna aplicación gráfica de gestión de paquetes como Synaptics, que permite instalar, desinstalar, actualizar o ver si está instalado un paquete.
3. **Paquetes tgz.** Contienen el código fuente del programa con los archivos necesarios para compilar e instalar el programa mediante los comandos tar, para empaquetar, y gzip o bzip para comprimir. Este tipo de paquetes, propio de la distribución Slackware, se suele usar en casi todas las distribuciones. El paquete está en un tipo de fichero .tar.gz o .tar.bz2.

RPM

Existen varios tipos de formato para definir un paquete. Red Hat introdujo la forma RPM (RedHat Package Manager) la cual ha sido adoptada por otras distribuciones como SuSE y Mandrake. En cambio Debian y Slackware hacen sus paquetes siguiendo otra técnica. Los paquetes RPM por lo general poseen extensión rpm. Estos pueden tener un grado mayor o menor de compatibilidad con las distribuciones que emplean el mismo formato.

En Red Hat la herramienta por excelencia para administrar paquetes es el comando rpm que posee muchísimas opciones. Internamente rpm se basa en una base de datos con información acerca de lo instalado, las dependencias existentes, etc. Esta base de datos es actualizada transparentemente por rpm cuando se instala, actualiza o desinstala un paquete. También puede ser consultada para conocer información diversa acerca de lo ya instalado en el sistema.

De forma general el comando rpm brinda las siguientes posibilidades:

- Instalar, actualizar y desinstalar paquetes.
- Construir y compilar paquetes.
- Inicializar y reconstruir la base de datos.
- Hacer consultas sobre paquetes instalados o no.
- Verificar dependencias.
- Adicionar firmas y chequearlas (mecanismo que permite asegurar la validez e integridad de un paquete).

A continuación se listan las opciones que se emplean más a menudo:

- -i : permite instalar los paquetes. Como argumento se colocan los nombres de los ficheros rpm a instalar. Internamente el comando chequea las dependencias funcionales, y de ser posible, reordena los ficheros rpm para que se satisfagan dichas dependencias durante la instalación.
- -e : permite desinstalar los paquetes. Como argumento se indican los nombres de los paquetes a desinstalar. El nombre de un paquete una vez instalado puede tener dos formas: una breve y otra más amplia que incluye el número de la versión. Algunos paquetes permiten que se instalen simultáneamente distintas versiones del mismo. Tal es el caso del paquete que contiene al kernel cuyo nombre breve siempre es kernel, mientras que pueden haber varios nombres ampliados como kernel-2.4.2-2 ó kernel-2.4.3-12.
- -U : permite actualizar los paquetes de versiones inferiores a superiores. Al igual que en la instalación, se colocan como

4.INSTALACION DE APLICACIONES LINUX

argumentos los nombres de los ficheros rpm que se desean actualizar. También aquí el comando realiza el ordenamiento más adecuado para satisfacer dependencias. De no existir una versión inferior previamente instalada, el proceso sería equivalente a una instalación de la versión superior.

- -F : es igual a la opción anterior salvo en que hace la actualización sólo si hay una versión inferior del paquete previamente instalada.
- -q : permite hacer consultas diversas a los paquetes instalados. Para ello se emplean como argumento los nombres de los paquetes: de forma breve o no. El tipo de consulta se expresa a través de una segunda opción. Las más importantes son:
- -a : lista los nombres (incluyendo la versión) de todos los paquetes instalados en el sistema.
- -i : brinda información acerca de un paquete: nombre, versión, tamaño, descripción, fabricante, fecha de instalación, licencia, clasificación, etc.
- -f : indica dado un fichero del file system, el nombre del paquete que lo creó.
- -l : lista los nombres de todos los ficheros que instaló un paquete.
- -p : permite hacer consultas a un paquete no instalado a partir del fichero rpm que lo contiene. Se puede combinar con las opciones anteriores -i y -l.

Otras opciones interesantes:

- -v : activa el modo explicativo durante la instalación o actualización. Si se indica dos veces, o sea -vv entonces se imprime información detallada de todas las operaciones efectuadas en la base de datos de RPMs durante el proceso.
- -h : muestra hasta 50 caracteres “#” que expresan el progreso del proceso de instalación o de actualización. Usualmente se combina con -v.
- -R : es una opción de consulta que muestra las dependencias que tiene un paquete.
- -d : es una opción de consulta que lista los ficheros de documentación que instala un paquete.
- -changelog : es una opción de consulta que muestra los logs que expresan los cambios efectuados en el paquete a partir de su versión anterior.
- -scripts : es una opción de consulta que muestra los scripts que se ejecutan al instalar y desinstalar un paquete.

1.2 Manipulación de paquetes en DEBIAN

En el principio existían los .tar.gz. Los usuarios tenían que descomprimir, destaar y compilar cada programa que quisieran usar en su sistema GNU/Linux. Cuando Debian fue creado, se decidió que el sistema incluyera un programa que se encargara de manejar los paquetes (programas) instalados en el ordenador. Este programa se llamó **DPKG**.

Así fue como nació el primer "paquete" en el mundo GNU/Linux, aún antes de que RedHat decidiera crear su propio sistema de paquetes "rpm".

Rápidamente llegó un nuevo dilema a las mentes de los creadores de GNU/Linux. Ellos necesitaban un modo fácil, rápido y eficiente de instalar programas, que manejara automáticamente las dependencias (programas que dependen de otros) y se hiciera cargo de la configuración mientras se actualizan. Nuevamente Debian fue pionera y creó el **APT**, Herramienta Avanzada de Empaquetamiento (Advanced Packaging Tool).

Como parte de su funcionamiento, APT utiliza un archivo que contiene las direcciones de varios servidores (repositorios) que se encuentran en Internet que contienen los paquetes candidatos a ser instalados. También indicamos en este fichero si vamos a cargar paquetes desde medios locales como un cdrom, un directorio compartido de la red, etc. Este archivo es /etc/apt/sources.list. Veamos el contenido de un archivo sources.list:

Cada línea en este fichero es un repositorio de software, que almacena paquetes que podemos descargar e instalar.

La primera palabra en cada línea, deb o deb-src, indica el tipo del archivo que se almacena en el repositorio: ya sea que contenga paquetes binarios (deb), esto es, los paquetes precompilados que normalmente se usan, o los paquetes fuente (deb-src), que son los códigos originales o fuentes.

A continuación de esta palabra viene la dirección donde se encuentra el repositorio, que puede ser bien una URL de red o un indicador como cdrom.

A continuación, viene el nombre de la distribución de Linux que queremos manejar con el repositorio. Normalmente un repositorio almacena paquetes para varias versiones de Linux, así por ejemplo el fichero anterior indica al repositorio que queremos los paquetes de la versión squeeze (Debian 6.0).

También se indica el tipo de repositorio, que puede ser normal, de actualización, de seguridad, etc.

Cada línea termina indicando el tipo de repositorio que queremos usar. Vemos que en el ejemplo todos nuestros repositorios son main (general).

En Debian normalmente se pueden indicar repositorios para stable, testing, unstable y experimental, que son las distintas versiones de desarrollo ofrecidas por Debian.

En Ubuntu podemos indicar main, restricted, universe y multiverse, dependiendo de si nos queremos bajar los paquetes oficiales y libres de Ubuntu, los paquetes no enteramente libres, paquetes no oficiales, etc.

Siempre que se modifica el archivo sources.list, hay que ejecutar el comando **apt-get update**. Debe hacer esto para permitir a APT obtener la lista de paquetes desde las fuentes que especificamos.

4.INSTALACION DE APLICACIONES LINUX

Si queremos utilizar el CD-ROM actual introducido para instalar los paquetes o para actualizar el sistema con APT, lo podemos agregar al archivo `sources.list`. Para hacerlo, podemos utilizar el programa `apt-cdrom` así: `apt-cdrom add`

Si tenemos en la unidad de `cdrom` un `cd` con paquetes `debian`, esta instrucción lo montará, y buscará la información de los paquetes en el `CD`.

El sistema de paquetes utiliza una base de datos para llevar un control sobre los paquetes instalados, los no instalados y cuales están disponibles para su futura instalación. El programa **apt-get** utiliza esta base de datos para averiguar cómo instalar los paquetes que son requeridos por el usuario y para indagar sobre que paquetes adicionales serán requeridos para que el paquete seleccionado funcione correctamente. Esta base de datos se actualiza con la orden **apt-get update**.

1.3 INSTALAR PAQUETES

Con el archivo `sources.list` listo y la lista de paquetes disponibles al día, todo lo que necesitamos es ejecutar `apt-get` para tener el paquete que queremos instalar. Por ejemplo, si ejecutamos:

```
apt-get install dopewars
```

APT buscará en su base de datos para encontrar la versión más reciente del paquete `dopewars` y lo descargará del servidor correspondiente especificado en `sources.list`. Si este paquete necesitara otro para funcionar APT resolverá las dependencias e instalará los paquetes necesarios.

Como vemos hemos indicado que queremos instalar el paquete `3dchess`, `apt` se ha puesto en contacto con el repositorio indicado en `sources.list`, le ha pedido información sobre dicho paquete y ha encontrado que tiene una dependencia no resuelta, es decir, que le hace falta un paquete para funcionar que no tenemos instalado en nuestro sistema (en el caso del ejemplo, el que falta es el paquete `xaw3dg`).

Vemos como `apt` automáticamente marca dicho paquete para instalarlo y nos pide confirmación. APT se encargará de bajar ambos paquetes, descomprimirlos, instalarlos en el sistema y configurarlos para que funcionen juntos.

APT sólo pregunta por confirmación cuando se van a instalar paquetes que no fueron especificados en la línea de comando.

Las siguientes opciones de `apt-get` podrían ser útiles:

- `h` Ayuda de la orden.
- `d` Solo descarga los paquetes, no instala nada en el sistema.
- `f` Continúa aunque haya fallos de integridad en los ficheros bajados.
- `y` Asume SI a todas las preguntas.
- `u` Muestra una lista de paquetes modificados.

Pueden seleccionarse varios paquetes para instalar en una sola línea. Los archivos descargados son almacenados en el directorio `/var/cache/apt/archives` para su instalación posterior.

Si en lugar de un paquete queremos descargarnos el código fuente del paquete podemos hacerlo usando el comando **apt-get source** paquete.

1.4 Reinstalar un paquete

Si queremos reinstalar un paquete, (por qué se haya estropeado, hayamos tocado la configuración y yano funcione, etc.) podemos usar la opción `--reinstall`: `apt-get install paquete --reinstall`

1.5 Eliminar un paquete

Si ya no necesitamos utilizar cierto paquete, podemos eliminarlo de nuestro sistema utilizando APT. Para realizar esta tarea sólo debemos escribir: `apt-get remove paquete`

1.6 Actualizar un paquete

Las actualizaciones de los paquetes pueden realizarse con tan sólo un comando: **apt-get upgrade**.

Podemos utilizar esta opción para actualizar los paquetes de la distribución actual, o bien para actualizar a una nueva distribución, aunque el comando **apt-get dist-upgrade** es una mejor opción.

Es muy útil utilizar este comando con la opción `-u`. Esta opción muestra la lista completa de paquetes que APT actualizará. Sin ella, se estaría actualizando a ciegas. APT descargará las versiones más recientes de cada paquete y las instalará de la manera más apropiada. Es muy importante ejecutar siempre **apt-get update** antes de probar esto.

1.7 Buscar un paquete

Existen algunas interfaces para el APT que lo hacen más fácil de utilizar. Pero nuestro objetivo aquí es aprender a manejar APT puro. Así que, ¿cómo podríamos saber el nombre de un paquete que queremos instalar?

4.INSTALACION DE APLICACIONES LINUX

Tenemos numerosos recursos para realizar esa tarea. Empezaremos con **apt-cache**. Este programa es utilizado por APT para mantener su base de datos. Nosotros sólo veremos un poco de sus aplicaciones. Por ejemplo, supongamos que queremos encontrar un emulador de la nintendo DS, y después bajar algunos juegos. Podríamos realizarlo así:

Veremos que la orden nos devuelve una cantidad muy grande de paquetes relacionados de alguna u otra manera con nintendo. Para refinar nuestra búsqueda, podríamos utilizar una orden como la siguiente:

1.8 Manejo de paquetes

Parece que hay un paquete prometedor, el desmume. Para ver información sobre dicho paquete usamos el comando **apt-cache show**

Como vemos obtenemos mucha información sobre el paquete, como la descripción, el tamaño, las dependencias, etc. Una vez comprobado que es el paquete que realmente queremos, bastaría con instalarlo con un `apt-get install`.

Comando dpkg	Explicación.
<code>dpkg -i paquete.deb</code>	A veces podemos bajar directamente un paquete desde internet, normalmente con la extensión <code>.deb</code> . Para instalar uno de estos paquetes basta con usar <code>dpkg</code> con la opción <code>-i</code> .
<code>dpkg -r paquete</code>	Elimina un paquete (remove).
<code>dpkg -P paquete</code>	Elimina un paquete, y además elimina también todos sus archivos de configuración, temporales, etc.
<code>dpkg -l</code>	Orden importante. Nos muestra un listado de todos los paquetes instalados en nuestro sistema.
<code>dpkg -L paquete</code>	Nos muestra información de un paquete ya instalado en el sistema, indicando que ficheros se instalaron y donde.

Un comando que se suele usar a menudo es `dpkg-reconfigure`. Este comando nos permite reconfigurar un paquete completo del sistema. Algunos usos de este comando habituales suelen ser:

Comando dpkg-reconfigure	Explicación.
<code>dpkg-reconfigure locales</code>	Configuración del idioma usado en los terminales de Debian.
<code>dpkg-reconfigure xserver-xorg</code>	Configura el sistema gráfico.
<code>dpkg-reconfigure -f noninteractive tzdata</code>	Configura el time zone (la zona horaria).

4. PAQUETES TGZ

Comando	Formato y Uso
	<pre>root@PC-Ubuntu-Jose:/home/jose# add-apt-repository ppa:otto-kesselgulasch/gimp This PPA is for Ubuntu >=12.04 and Linux Mint >=13</pre> <pre>root@PC-Ubuntu-Jose:/home/jose# sudo apt-get update</pre> <pre>root@PC-Ubuntu-Jose:/home/jose# apt-get install gimp Leyendo lista de paquetes... Hecho Creando árbol de dependencias</pre> <p>Y para lanzar el programa instalado:</p> <pre>root@PC-Ubuntu-Jose:/home/jose# gimp</pre> <ul style="list-style-type: none">Para desinstalar paquetes ya instalados, se ejecutaría de la siguiente manera: <pre>root@PC-Ubuntu-Jose:/home/jose# apt-get remove gimp Leyendo lista de paquetes... Hecho</pre>
aptitude	Es una versión mejorada del comando apt que se puede utilizar con las mismas opciones que éste. No viene instalado por defecto.
dpkg	[Debian Package Management System] También se utiliza para instalar, desinstalar, y proporcionar información sobre los paquetes .deb , pero NO resuelve las dependencias . Sólo trabaja sobre los archivos con la extensión .deb . <pre>dpkg [opciones] paquete.deb</pre> <p>Algunas de las opciones soportadas por el comando son:</p> <ul style="list-style-type: none">-i / -install Para instalar paquetes .deb.-l Lista los paquetes conocidos por dpkg.--purge Permite la desinstalación de paquetes .deb.

• Comandos para comprimir y empaquetar: paquetes tgz.

Son quizás los de utilización más antigua. Las primeras distribuciones de **GNU/Linux** los utilizaban para instalar el software, y aún varias distribuciones los usan (por ejemplo, Slackware) y algunos UNIX comerciales. Son una combinación de ficheros unidos por el comando **tar** en un único fichero **.tar**, que luego ha sido comprimido por la utilidad **gzip**. Por eso suele aparecer con la extensión **.tgz** o bien **.tar.gz**. Asimismo, hoy en día es común encontrar los **tar.bz2** que utilizan en lugar de **gzip** otra utilidad llamada **bzip2**, que en algunos casos consigue mayor compresión del archivo.

Este tipo de paquete no contiene ningún tipo de información de dependencias, y puede presentar tanto contenido de aplicaciones en formato binario como en código fuente. Se puede considerar como una especie de colección de ficheros comprimida.

El proceso básico con estos paquetes consiste en:

Descomprimir el paquete: **tar -zxvf fichero.tar.gz** (o fichero **.tgz**)

Donde la opción **z** descomprime, **x** extraer ficheros, **v** muestra el proceso y **f** hace referencia al fichero a tratar.

También se puede hacer por separado (sin la **z** del **tar**) para descomprimir y luego extraer el fichero del paquete: **gunzip fichero.tar.gz** (deja un fichero **tar**) **tar -xvf fichero.tar**

4. PAQUETES TGZ

Si se trata de un paquete binario, la instalación suele ser bastante fácil ya que será ejecutable directamente desde donde se encuentre ubicado, o traerá algún instalador propio para dicho propósito. Otra posibilidad será que haya que hacerlo manualmente, con lo que bastará con copiar (**cp -r**, copia recursiva) o mover (comando **mv**) el directorio a la posición deseada.

Otro caso es el formato de código fuente. Entonces, antes de instalar el software tendremos que ejecutar un proceso de compilación. Para esto habrá que leer con cierto detalle las instrucciones asociadas al programa. Pero la mayoría de desarrolladores usan un sistema de GNU llamado **autoconf** (de autoconfiguración), en el que habitualmente se usan los siguientes pasos (si no aparecen errores):

- **./configure**: se trata de un script que configura el código para poder ser compilado en la máquina verificando la existencia de las herramientas adecuadas. La opción **-prefix = directorio** permite especificar dónde se instalará el software.
- **make**: compilación propiamente dicha.
- **make install**: instalación del software a un lugar adecuado, normalmente especificado previamente como opción al configure o asumida por defecto.

En caso de querer borrar el software instalado, habrá que utilizar el desinstalador si nos lo proporcionan. En otro caso habrá que borrar directamente el directorio o ficheros que se instalaron, teniendo cuidado de posibles dependencias.

2. También puede interesar realizar el proceso inverso, es decir empaquetar y comprimir ficheros o directorios. Así, los paquetes **tgz** son bastante habituales como mecanismo de *backup* en tareas de administración, por ejemplo, para guardar copias de datos importantes, hacer copias de cuentas de usuario, o guardar datos que no se sabe si se volverán a necesitar: históricos. Suponiendo que se quiere crear copia de seguridad del directorio **'empleados'**, habría que seguir el siguiente proceso: **tar -cvf empleados.tar empleados**

Donde la opción **c** compacta el directorio **empleados** generando un fichero empaquetado denominado **empleados.tar**. **gzip empleados.tar**

Para comprimir el fichero resultante del proceso anterior, obteniéndose un fichero **empleados.tar.gz**.

También se podían haber aunado los dos comandos en uno sólo mediante la ejecución de la instrucción:

tar -zcvf empleados.tgz empleados

El resultado será un fichero **empleados.tgz** resultante de la compresión [z] y la compactación [c] de la carpeta original.

4. PAQUETES TGZ

Para comprobar cómo funciona **tar**, se puede crear en primer lugar dos archivos vacíos en el directorio SOM de la siguiente manera:

```
root@PC-Ubuntu-Jose:/home/jose# mkdir SOM; cd SOM
root@PC-Ubuntu-Jose:/home/jose/SOM# touch notas1 notas2
root@PC-Ubuntu-Jose:/home/jose/SOM# ls -l
total 0
-rw-r--r-- 1 root root 0 may 17 15:47 notas1
-rw-r--r-- 1 root root 0 may 17 15:47 notas2
root@PC-Ubuntu-Jose:/home/jose/SOM#
```

Ahora, se empaquetan (se guardan) los ficheros en un archivo **.tar** con el nombre *notasempaquetadas.tar*. Es conveniente incluir en el nombre del archivo la extensión **.tar** para indicar que se ha empaquetado en él determinados ficheros y/o directorios.

```
root@PC-Ubuntu-Jose:/home/jose/SOM# tar -cvf notasempaquetadas.tar notas1
notas1
notas2
```

Analizando la sintaxis del comando, se observa que:

- La primera opción establece la acción que se llevará a cabo: **-c** indica que se creará un archivo **.tar** empaquetado.
- La segunda opción, **-v [Verbose]**, lista los ficheros que están siendo procesados. Sólo se utiliza si se desea visualizar esta información en pantalla.
- La última opción, **-f**, indica el fichero que se usará como salida del comando. en este caso, el archivo a crear (*notasempaquetadas.tar*).
- El primer argumento es el archivo **.tar** a crear.
- Al final, se especificarán los ficheros y directorios a archivar (*notas1* y *notas2* en el ejemplo).

Si se lista el directorio SOM (el actual) se observará el fichero **.tar** con su característico color rojo:

```
root@PC-Ubuntu-Jose:/home/jose/SOM# ls -l
total 12
-rw-r--r-- 1 root root 0 may 17 15:47 notas1
-rw-r--r-- 1 root root 0 may 17 15:47 notas2
-rw-r--r-- 1 root root 10240 may 17 21:02 notasempaquetadas.tar
root@PC-Ubuntu-Jose:/home/jose/SOM#
```

Al desempaquetar el archivo **.tar**, los ficheros y directorios se situarán en la ruta de acceso con la que fueron archivados. Por este motivo, se debe ser cuidadoso al especificar las rutas de acceso de los ficheros *notas1* y *notas2*, en este caso.

- Si lo que interesa es visualizar el contenido de un archivo **.tar**, se usará de nuevo el comando **tar** con la opción **-t**:

```
root@PC-Ubuntu-Jose:/home/jose/SOM# tar -tf notasempaquetadas.tar
notas1
notas2
```

- Para añadir información al fichero *notasempaquetadas.tar*, se creará un nuevo directorio en la carpeta del usuario denominada MAN, donde se creará un fichero vacío denominado *notas3*:

```
root@PC-Ubuntu-Jose:/home/jose/SOM# mkdir ../MAN; touch ../MAN/notas3
```

Ahora, se añade el directorio completo al fichero empaquetado utilizando el comando **tar** con la opción **-r**:

```
root@PC-Ubuntu-Jose:/home/jose/SOM# tar -rvf notasempaquetadas.tar ../MAN
tar: Eliminando la '../' inicial de los nombres
../MAN/
../MAN/notas3
```

Así si mostramos el contenido del fichero *notasempaquetadas.tar* se mostrará:

```
root@PC-Ubuntu-Jose:/home/jose/SOM# tar -tf notasempaquetadas.tar
notas1
notas2
MAN/
MAN/notas3
```

4. PAQUETES TGZ

- Para desempaquetar archivos se utiliza la opción **-x** del comando **tar**:

```
root@PC-Ubuntu-Jose:/home/jose/SOM# cd ..
root@PC-Ubuntu-Jose:/home/jose# tar -xvf SOM/notasempaquetadas.tar
notas1
notas2
MAN/
MAN/notas3
root@PC-Ubuntu-Jose:/home/jose# ls -l
total 64
drwxr-xr-x 2 jose jose 4096 abr 23 22:14 Descargas
drwxr-xr-x 2 jose jose 4096 abr 23 22:14 Documentos
drwxr-xr-x 2 jose jose 4096 may 17 21:25 Escritorio
-rw-r--r-- 1 jose jose 8980 abr 23 21:57 examples.desktop
-rw-rw-r-- 1 jose jose 18 may 14 22:06 fichero
drwxr-xr-x 2 jose jose 4096 abr 23 22:14 Imágenes
drwxr-xr-x 2 root root 4096 may 17 21:36 MAN
drwxr-xr-x 2 jose jose 4096 abr 23 22:14 Música
-rw-r--r-- 1 root root 0 may 17 15:47 notas1
-rw-r--r-- 1 root root 0 may 17 15:47 notas2
drwxr-xr-x 2 jose jose 4096 abr 23 22:14 Plantillas
drwxr-xr-x 2 root root 4096 may 14 21:58 prueba
drwxrwxr-x 2 jose jose 4096 may 14 22:02 prueba2
drwxr-xr-x 2 jose jose 4096 abr 23 22:14 Público
drwxr-xr-x 2 root root 4096 may 17 21:51 SOM
drwxr-xr-x 2 jose jose 4096 abr 23 22:14 Vídeos
```

El sistema sitúa el contenido del paquete en el directorio actual porque el contenido fue archivado indicando rutas relativas al directorio actual. Si se hubiesen usado rutas absolutas se podría tener verdaderos problemas a la hora de desempaquetar el archivo, pues podrían depender de nombres de directorios que no existen si se hubiesen efectuado cambios en el sistema de archivos o se estuviese desempaquetando en otro equipo.

El programa **gzip** reduce el tamaño de los ficheros. Se mantienen los mismos permisos, propietarios y fecha de modificación de los archivos comprimidos, aunque se sustituye cada fichero por su correspondiente comprimido con extensión **.gz**.

Para comprobar cómo funciona se empleará este comando para comprimir el archivo **notasempaquetadas.tar** para que ocupe menos, por ejemplo, al enviarlo por correo electrónico:

```
root@PC-Ubuntu-Jose:/home/jose/SOM# ls -l
total 12
-rw-r--r-- 1 root root 0 may 17 15:47 notas1
-rw-r--r-- 1 root root 0 may 17 15:47 notas2
-rw-r--r-- 1 root root 10240 may 17 21:52 notasempaquetadas.tar
root@PC-Ubuntu-Jose:/home/jose/SOM# gzip -v notasempaquetadas.tar
notasempaquetadas.tar: 98.5% -- replaced with notasempaquetadas.tar.gz
```

Primero se lista en formato largo [ls -l] el directorio SOM donde se encontraba almacenado el fichero empaquetado **notasempaquetadas.tar**. Luego se ejecuta el comando **gzip** con la opción **-v** para obtener información sobre el proceso de compresión aplicado al fichero especificado.

Si ahora se vuelve a listar el directorio se observará que el archivo **notasempaquetadas.tar** ha añadido la extensión **.gz**. Es decir, ahora ya es un archivo comprimido, que además ocupa bastante menos espacio en el dispositivo: antes 10240 bytes y ahora 190.

```
root@PC-Ubuntu-Jose:/home/jose/SOM# ls -l
total 4
-rw-r--r-- 1 root root 0 may 17 15:47 notas1
-rw-r--r-- 1 root root 0 may 17 15:47 notas2
-rw-r--r-- 1 root root 190 may 17 21:52 notasempaquetadas.tar.gz
```

Para descomprimir el archivo **.tar.gz**, se utilizará el comando **gzip -d** como se muestra en el ejemplo:

```
root@PC-Ubuntu-Jose:/home/jose/SOM# gzip -d notasempaquetadas.tar.gz
root@PC-Ubuntu-Jose:/home/jose/SOM# ls -l
total 12
-rw-r--r-- 1 root root 0 may 17 15:47 notas1
-rw-r--r-- 1 root root 0 may 17 15:47 notas2
-rw-r--r-- 1 root root 10240 may 17 21:52 notasempaquetadas.tar
```