



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES  
ACATLAN

**"SISTEMAS OPERATIVOS  
OPEN SOURCE: CASO DE ESTUDIO LINUX"**

**TESINA**



QUE PARA OBTENER EL TITULO DE:  
LICENCIADO EN MATEMATICAS APLICADAS Y COMPUTACION

**PRESENTA:**

**Román Martínez Alejandro**

**ASESOR:**

**M. en C. Sara Camacho Cancino**



NAUCALPAN EDO. DE MEXICO ENERO, 2004



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**

**Tesis Digitales**

**Restricciones de uso**

**DERECHOS RESERVADOS ©**

**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## **Agradecimientos**

En estas líneas quiero expresar mi agradecimiento a toda aquellas personas que han contribuido en mi formación como persona y como profesionista.

Agradezco a mis padres por haberme apoyado todos estos años en los que realice mis estudios. Gracias por haber creído en mi. A mis hermanas Maribel, Leticia, Ofelia, Celia y Margarita por haberme apoyado.

A mi asesora la M. C. Sara Camacho Cancino quien me brindo su atención y tiempo en el trayecto de todo el proceso de la realización de mi tesina. Gracias por brindarme su asesoría y su valioso conocimiento.

Agradezco a los integrantes del jurado Ing. Rubén Romero Ruíz, Ing. Francisco Javier Patiño Donnadieu, Lic. Oscar Gabriel Caballero Martínez, Ing. Gabriel Díaz Mirón Mac Donough.

Agradezco a la U.N.A.M. por abrirme las puertas al conocimiento y por formarme como profesionista.

Y muy en especial a Lizbeth por estar siempre en mi corazón.

## INDICE

	Pagina
<b>INTRODUCCIÓN</b> .....	7
<b>METODOLOGÍA</b> .....	9
<b>MARCO TEÓRICO</b> .....	11

### CAPÍTULO UNO SISTEMAS OPERATIVOS

1.1 ¿Qué es un Sistema Operativo?.....	13
1.2 Historia de los Sistemas Operativos.....	13
1.3 Funciones y Características de los Sistemas Operativos.....	14
1.4 Clasificación de los Sistemas Operativos.....	16
1.5 Sistemas Operativos por Lotes.....	16
1.6 Sistemas Operativos de Tiempo Real. ....	17
1.7 Sistemas Operativos de Multiprogramación.....	18
1.8 Sistemas Operativos de Tiempo Compartido.....	19
1.9 Sistemas Operativos Distribuidos.....	20
1.10 Sistemas Operativos de Red. ....	21
1.11 Sistemas Operativos Paralelos.....	21

### CAPÍTULO DOS LICENCIA OPEN SOURCE

2.1 Conceptos Relacionados con el Software Libre.....	26
2.2 Proyecto GNU.....	28
2.3 Implicaciones Legales.....	31



## CAPÍTULO TRES

SISTEMAS OPERATIVOS FUNDAMENTADOS EN OPEN SOURCE: CASO DE  
ESTUDIO LINUX

3.1 ¿Elementos de Linux?.....	48
3.2 ¿Que es un Sistema Operativo Basado en Software Libre?.....	48
3.3 ¿Que características Analizaremos de los S.O Linux?.....	49
3.4 Distribuciones de Sistemas Operativos basados en Software Libre.....	50
3.5 Distribución Redhat Linux.....	53
3.6 Distribución Debian.....	56
3.7 Distribución S.u.S.E.....	59
3.8 Distribución Linux-Mandrake.....	63
3.9 Pasos para instalar un Sistema Operativo Linux.....	66

## CAPÍTULO CUATRO

## INTERFASE DE LOS SISTEMAS OPERATIVOS CON EL SOFTWARE LIBRE

4.1 Lenguajes de Programación.....	79
4.2 Bases de datos.....	85
4.3 Aplicaciones de Escritorio.....	88
4.8 Redes.....	91

## CAPÍTULO CINCO

USO Y TENDENCIA DE LOS SISTEMAS OPERATIVOS OPEN SOURCE EN  
MEXICO.

5.1 Uso de los Sistemas Operativos Open Source en el Sector publico.....	99
5.2 Impactos principales de los Sistemas Operativos Open Source en la Administración Pública.....	100
5.3 Dificultades de adopción de los Sistemas Operativos Open Source.....	103
5.4 Papel del Sector Público en el mundo de los Sistemas Operativos Open Source.....	104
5.5 Instituciones Públicas que utilizan Sistemas Operativos Open Source.....	109
5.6 Uso de los Sistemas Operativos Open Source en el Sector Privado.....	109
5.7 Instituciones Privadas que utilizan Sistemas Operativos Open Source.....	110

CONCLUSIONES.....	113
BIBLIOGRAFÍA.....	115
REFERENCIA DE SITIOS EN INTERNET.....	117
GLOSARIO.....	121

## INTRODUCCIÓN

El software de la computadora puede dividirse de modo general en dos clases: los programas que manejan la operación de la computadora misma y los que manejan las aplicaciones que se utilizan. El programa mas importante es el Sistema Operativo ya que controla todos los recursos de la computadora y ofrece la base sobre la cual se escriben los programas de aplicación. La utilización de un Sistema Operativo es indispensable para la interfaz entre el ser humano y el hardware, la necesidad adquirir un Sistema Operativo y adaptarlo a las exigencias de las actividades que se llevan a cabo en el área de sistemas es algo relevante ya que requiere de un análisis detallado de selección y si éste no se realiza con cuidado los costos tienden a ser muy altos. Actualmente en el mercado los Sistemas Operativos más populares son manufacturados en serie y el usuario debe ajustarse a ellos quiera o no, por ejemplo, solo se pueden utilizar las licencias que se compran, no es factible modificar el código fuente para adaptarlo a las necesidades que se requieren y se debe actualizar constantemente si se desea recibir apoyo y mantenimiento. Conforme avanza la tecnología en los Sistemas Operativos es necesario crear nuevas respuestas para poder establecer un modelo que se adapte a las necesidades de cada usuario, esto parece muy general y complejo, pero de lo que se trata aquí es de probar que la utilización de Sistemas Operativos de distribución libre permite lograr ventajas contundentes ante el Sistemas Operativos obtenidos bajo licenciamientos; esta ventaja deberá verse reflejada en la libertad de la constitución del sistemas operativo y el costo total de la inversión.

El objetivo general de la tesina presentada es mostrar las alternativas que existen en cuanto al uso de Sistemas Operativos, utilizar el caso de Linux para señalar las bondades y virtudes de los Sistemas Operativos Open Source así como la problemática del uso del Software Libre, revisar los estatutos legales del Software Libre y mostrar las aplicaciones que existen para los Sistemas Operativos que se estan tratando.

## INTRODUCCION

---

Cuando se habla de Sistemas Operativos fundamentados en Software Libre, se desea resaltar que su principal cualidad es la capacidad de modificar su código fuente y poder ajustarlo como un traje a la medida para satisfacer las necesidades del usuario en el ámbito de cómputo.

Para lograr su objetivo esta tesina esta estructurada en cinco capítulos:

En el primer capítulo se exponen los elementos, las definiciones y funciones de los Sistemas Operativos, también se revisaran sus limites y alcances, la razón de ser de este capítulo es introducir al lector en los conceptos fundamentales de un Sistema Operativo.

En el segundo capítulo se analiza que es el Open Source, sus límites y alcances y se presenta la licencia de los Sistemas Operativos Open Source.

En el tercer capítulo se estudian las principales características de los Sistemas Operativos Open Source fundamentados en código Linux que existen en el mercado.

En el cuarto capítulo se muestran las aplicaciones que existen para los Sistemas Operativos Open Source, ya que son las que le otorgan la funcionalidad para que se convierta en una herramienta útil de cómputo.

Y en el último capítulo se analizan los usos, interacciones y tendencias de los Sistemas Operativos Open Source en nuestro país para mostrar los fundamentos de este trabajo.

Al final del trabajo afirmaremos o refutaremos la viabilidad de utilizar los Sistemas Operativos basados en Open Source.

## METODOLOGIA

La hipótesis de la tesina es demostrar que los Sistemas Operativos basados en Open Source pueden ser utilizados en cualquier rama del área de sistemas y su uso tiene ventajas importantes.

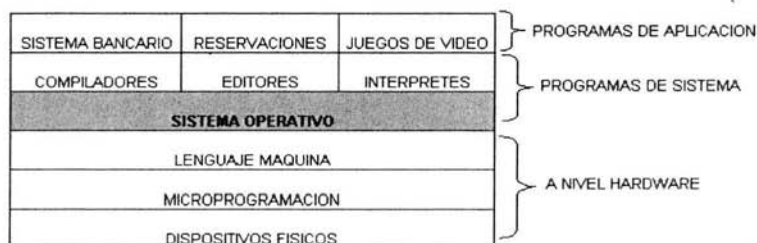
Todos los inventos con los que ha avanzado la humanidad con el paso del tiempo se transforman y se hacen de libre distribución y no se tiene que estar pagando los derechos para usar digamos la rueda, el modelo básico de un automóvil, el teorema de Pitágoras, la medicina, en fin todos los inventos que han hecho avanzar al hombre en su camino hacia el entendimiento de las cosas y cuyo uso ha contribuido a mejorar la calidad de vida del hombre, y precisamente el origen de este trabajo se basa en que el software también es patrimonio de la humanidad, y por lo tanto debe ser de libre distribución.

Ahora adentrándonos en los Sistemas Operativos ¿Qué pasos se seguirán para investigar este tema? Primero se explicara que es un Sistema Operativo después se revisará cuales son las legislaciones vigentes para analizar el marco legal que sustenta este trabajo posteriormente, se evaluarán las distribuciones más representativas de Linux, se estudiarán las aplicaciones y se mostrarán las características de cada Sistema Operativo, no con la finalidad de decidir cual es el mejor, sino con el deseo de mostrar las cualidades de cada Sistema Operativo y así cuando un usuario decida instalar alguno este estudio le sirva como base y le ayude a seleccionar el mas adecuado a sus necesidades.

## MARCO TEORICO

Lo primero que se recomienda cuando se es niños es aprender a caminar antes que correr, lo mismo sucede cuando se investiga un tema, primero debemos conocer y comprender los términos que se van a tratar.

Los Sistemas Operativos son fundamentales para el funcionamiento de una computadora, su estudio es obligatorio e importante en varias universidades, pero es dinámico y va cambiando conforme transcurre el tiempo, en la figura 1 se muestra el esquema de un sistema de computo a fin de entender su ámbito.



**Fig. 1. Esquema de un sistema de cómputo**

El objetivo fundamental de un Sistema Operativo es conseguir un alto rendimiento del hardware y software ofreciéndole, a las aplicaciones las facilidades necesarias para compartir datos y recursos entre las aplicaciones y, un modelo de programación sencillo que evite que el programador de la aplicación tenga que rescribir procesos del Sistema Operativo<sup>1</sup>.

Las características se van a investigar en los Sistemas Operativos Open Source son:

- ✓ Conveniencia.
- ✓ Eficiencia.

<sup>1</sup> ANDREWS, TANENBAUM., Sistemas Operativos Modernos, México, Prentice Hall, 1993, Pág. 12.

- ✓ Habilidad para evolucionar
- ✓ La forma de administrar el hardware.
- ✓ Como relacionan dispositivos (gestión a través del kernel).
- ✓ La manera de organizar los datos para accesos rápidos y seguros.
- ✓ Su manejo de comunicaciones.
- ✓ Su manejo de entradas y salidas.

Y a un nivel mas específico en esta investigación se verificara que los Sistemas Operativos cumplan con los siguientes requisitos:

- ✓ Multitarea.
- ✓ Multiusuario.
- ✓ Multiplataforma.
- ✓ Multiprocesador
- ✓ Protección de la memoria entre procesos.
- ✓ Carga de ejecutables por demanda.
- ✓ Archivos compartidos ejecutables.
- ✓ Código fuente disponible.
- ✓ Conectividad.

Cuando se recibe una computadora nueva lo que aparece en el escritorio es un botón de inicio con una barra de aplicaciones, tal vez un usuario se logre introducir al registro y personalizar su interfaz pero no podrá cambiar la estructura de las ventanas ni las aplicaciones especificadas del Sistema Operativo debido a que solo se tienen los ejecutables del programa, claro que para hacerlo es necesario conocer fundamentos básicos de programación, lenguajes de programación, saber recompilar el núcleo, ensamblar una red, e instalar el Sistema de Computo y aplicaciones de oficina; en pocas palabras, crear computo en vez de solo utilizarlo algo que un egresado de la Licenciatura en Matemáticas Aplicadas y Computación con dedicación, tiempo y esfuerzo debe poder llevar a cabo, y es precisamente por esto es que es posible aventurarse a proponer esta tesina. ¿Por qué conformarse con usar algo que existe tal y como un tercero lo diseño y programo?, ¿Por qué si se puede adaptarlo a la experiencia, gusto y necesidad del usuario?.

# **CAPÍTULO 1**

## **SISTEMAS OPERATIVOS**

### **1.1 ¿Qué es un Sistema Operativo?**

Un Sistema Operativo es el software básico que controla una computadora. El Sistema Operativo tiene tres grandes funciones: coordina y manipula el hardware de la computadora (como la memoria, las impresoras, las unidades de disco, el teclado o el ratón), organiza los archivos en diversos dispositivos de almacenamiento (como discos flexibles, duros, o compactos, DVD's y cintas magnéticas) y gestiona los errores de hardware y la pérdida de datos.

### **1.2 Historia de los Sistemas Operativos.**

Para tratar de comprender los requisitos de un Sistema Operativo y el significado de las principales características de un Sistema Operativo contemporáneo, es útil conocer como ha sido su evolución:

En los 40's, se introducen los programas bit a bit, por medio de interruptores mecánicos. El programador interactuaba de manera directa con el hardware de la computadora, era un Sistema Operativo rudimentario. Las primeras computadoras utilizaban bulbos, la entrada de datos y los programas se realizaban a través del lenguaje máquina (bits) o a través de interruptores.

Durante los años 50's y 60's.- A principio de los 50's, la compañía General's Motors implanto el primer Sistema Operativo para su IBM 170. Empiezan a surgir las tarjetas perforadas que permiten que los usuarios (que en ese tiempo eran programadores, diseñadores, capturistas, etc.), se encarguen de modificar sus programas. Los usuarios establecían o apartaban tiempo, metían o introducían sus programas, corregían y depuraban sus programas en su tiempo. A esto se le llama trabajo en serie.



Todo esto se traduc a en p rdida de tiempo y tiempos de ejecuci n excesivos.

En los a os 60's y 70's se genera el circuito integrado, se organizan los trabajos y se generan los procesos Batch (por lotes), lo cual consiste en determinar los trabajos comunes y realizarlos juntos. En esta  poca surgen las unidades de cinta y el cargador de programas, considerado el primer tipo de Sistema Operativo.

A finales de los a os 80's comienza el gran auge y evoluci n de los Sistemas Operativos. Se descubre el concepto de multiprogramaci n que consiste en tener al mismo tiempo, varios trabajos cargados en memoria, tema principal de los Sistemas Operativos actuales.

En los 90's se introduce a la era de la computaci n distribuida y del multiprocesamiento a trav s de m ltiples redes de computadoras, aprovechando el ciclo del procesador. Se tiene una configuraci n din mica con un reconocimiento inmediato de dispositivos y software que se a ade o elimina de las redes a trav s de procesos de registro y localizadores<sup>2</sup>.

La conectividad se facilita gracias a est ndares y protocolos de sistemas abiertos por organizaciones como la Organizaci n Internacional de Normas, todo encaminado a ser controlado por los protocolos de comunicaci n OSI y por la red de servicios digital ISDN.

### 1.3 Funciones y caracter sticas de los Sistemas Operativos.

Las principales funciones de un Sistema Operativo son:

- **Control de recursos:** Coordina y manipula el hardware de la computadora, como la memoria, las impresoras, las unidades de disco, el teclado o el Mouse.
- **Manejo de dispositivos de E/S:** Organiza los archivos en diversos dispositivos de almacenamiento, como discos flexibles, discos duros, discos compactos o cintas magn ticas.
- **Manejo de errores:** Gestiona los errores de hardware y la p rdida de datos.

<sup>2</sup> A.SILBERTSCHATZ, J. PETERSON, P. GALVIN, Sistemas Operativos, conceptos fundamentales, M xico, 3  Edici n, Addison Wesley ,727p.

- **Secuencia de tareas:** El Sistema Operativo debe administrar la manera en que se reparten los procesos.
- **Definir el orden:** Indica que tarea va primero y cual va después.
- **Protección:** Evita que las acciones de un usuario afecten el trabajo que esta realizando otro usuario.
- **Multiacceso:** Un usuario se puede conectar a otra máquina sin tener que estar cerca de ella.
- **Contabilidad de recursos:** Establece el costo que se le cobra a un usuario por utilizar determinados recursos.

En general, se puede decir que un Sistema Operativo tiene las siguientes características:

- **Conveniencia.** Un Sistema Operativo hace más conveniente el uso de una computadora.
- **Eficiencia.** Un Sistema Operativo permite que los recursos de la computadora se usen de la manera más eficiente posible.
- **Habilidad para evolucionar.** Un Sistema Operativo deberá construirse de manera que permita el desarrollo, prueba o introducción efectiva de nuevas funciones del sistema sin interferir con el servicio.
- **Encargado de administrar el hardware.** El Sistema Operativo es responsable de manejar los recursos de la computadora de la mejor manera en cuanto a hardware se refiere, esto es, asignar a cada tarea una parte del procesador para poder compartir los recursos.
- **Relacionar dispositivos (gestionar a través del kernel).** El Sistema Operativo se debe encargar de comunicar a los dispositivos periféricos, cuando el usuario así lo requiera.
- **Organizar datos para acceso rápido y seguro.**
- **Manejar las comunicaciones en red.** El Sistema Operativo permite al usuario manejar con facilidad todo lo referente a la instalación y uso de las redes de computadoras.
- **Procesamiento por bytes de flujo a través del bus de datos.**

- **Facilitar las entradas y salidas.** Un Sistema Operativo debe facilitar al usuario el acceso y manejo de los dispositivos de Entrada/Salida de la computadora<sup>3</sup>.

#### 1.4 Clasificación de los Sistemas Operativos.

Con el paso del tiempo, los Sistemas Operativos se clasificaron de diferentes maneras, dependiendo de su uso o de su aplicación. A continuación se mostrarán diversos tipos de Sistemas Operativos que existen en la actualidad con algunas de sus características:

#### 1.5 Sistemas Operativos por lotes.

Los Sistemas Operativos por lotes, procesan una gran cantidad de trabajos con poca o ninguna interacción entre los usuarios y los programas en ejecución. Se reúnen todos los trabajos comunes para realizarlos al mismo tiempo, evitando la espera de dos o más trabajos como sucede en el procesamiento en serie. Estos sistemas son de los más tradicionales y antiguos, y fueron introducidos alrededor de 1956 para aumentar la capacidad de procesamiento de los programas.

Cuando estos sistemas son planeados correctamente pueden, tener un tiempo de ejecución muy alto, porque el procesador se utiliza mejor y los Sistemas Operativos pueden ser simples, debido a la secuenciabilidad de la ejecución de los trabajos.

Algunos ejemplos de Sistemas Operativos por lotes exitosos son el SCOPE, del DC6600, el cual está orientado a procesamiento científico pesado, y el EXEC II para el UNIVAC 1107, orientado a procesamiento académico.

Algunas características de los Sistemas Operativos por lotes son:

- Requiere que el programa, datos y órdenes al sistema sean remitidos en forma de lote.
- Permiten poca o ninguna interacción usuario/programa en ejecución.
- La utilización de recursos es mejor que en el procesamiento serial simple.

---

<sup>3</sup> ANDREWS, TANENBAUM, *Sistemas Operativos Modernos*, México, Prentice Hall, 1993.  
A.S.TANENBAUM, A.S.WOODHULL, *Sistemas Operativos. Diseño e Implementación*. México., Prentice-Hall, 1997

- Se recomienda para programas con largos tiempos de ejecución (Ejemplo, análisis estadísticos, nóminas de personal, etc.).
- Se encuentran en muchas computadoras personales combinados con procesamiento serial.
- La planificación del procesador es sencilla, típicamente las tareas son procesadas en orden de llegada.
- La planificación de la memoria es sencilla, generalmente se divide en dos: parte residente del S.O. y programas transitorios.
- No requieren gestión crítica de dispositivos en el tiempo.
- Suelen proporcionar gestión sencilla de manejo de archivos: se requiere poca protección y ningún control de concurrencia para el acceso.

### **1.6 Sistemas Operativos de tiempo real.**

Los Sistemas Operativos de tiempo real son aquellos en los cuales los procesos tienen mayor importancia que el usuario. Por lo general, sus recursos están subutilizados con la finalidad de prestar atención a los procesos en el momento que lo requieran; se utilizan en entornos donde son procesados un gran número de sucesos o eventos.

Muchos Sistemas Operativos de tiempo real son construidos para aplicaciones muy específicas como control de tráfico aéreo, bolsas de valores, control de refinerías, control de laminadores. También en el ramo automovilístico y de la electrónica de consumo, las aplicaciones de tiempo real están creciendo muy rápidamente. Otros campos de aplicación de los Sistemas Operativos de tiempo real son los siguientes:

- Control de trenes.
- Telecomunicaciones.
- Sistemas de fabricación integrada.
- Producción y distribución de energía eléctrica.
- Control de edificios.
- Sistemas multimedia.

Algunos ejemplos de Sistemas Operativos de tiempo real son: VxWorks, Solaris, Lyns OS y Spectra. Los Sistemas Operativos de tiempo real, cuentan con las siguientes características:

- Se deben utilizar en entornos en donde deben ser aceptados y procesados una gran cantidad de sucesos, la mayoría externos al sistema computacional, en breve tiempo o dentro de ciertos plazos.
- Se utilizan en control industrial, conmutación telefónica, control de vuelo, simulaciones en tiempo real, aplicaciones militares, etc.
- Su objetivo es proporcionar tiempos rápidos de respuesta.
- Procesa ráfagas de miles de interrupciones por segundo sin perder un solo suceso.
- El proceso se activa tras ocurrencia de suceso, mediante interrupción.
- El proceso de mayor prioridad expropia recursos.
- Generalmente se utiliza planificación expropiativa basada en prioridades.
- Usualmente los procesos son residentes permanentes en memoria.
- La población de procesos es estática en gran medida.
- Existe poco movimiento de programas entre almacenamiento secundario y memoria.
- La gestión de archivos se orienta más a la velocidad de acceso que a la utilización eficiente del recurso.

### **1.7 Sistemas Operativos de Multiprogramación (o Sistemas Operativos de multitarea).**

Se distinguen por su habilidad para soportar la ejecución de dos o más trabajos activos (que se están ejecutando) al mismo tiempo. Esto trae como resultado que la Unidad Central de Procesamiento (CPU) siempre tenga alguna tarea que ejecutar, aprovechando su utilización al máximo.

Su objetivo es tener varias tareas en la memoria principal, de manera que cada uno está usando el procesador, o un procesador distinto, es decir, involucra máquinas con más de un CPU.

Los Sistemas Operativos como UNIX, Windows 95, Windows 98, Windows NT, MAC-OS, OS/2, soportan la multitarea.

Las características de un Sistema Operativo de multiprogramación o multitarea son las siguientes:

- Mejora la productividad del sistema y la utilización de recursos.
- Multiplexa recursos entre varios programas.
- Generalmente soportan múltiples usuarios (multiusuarios).
- Proporcionan facilidades para mantener el entorno de usuarios individuales.
- Requieren validación de usuario para seguridad y protección.
- Proporcionan contabilidad del uso de los recursos por parte de los usuarios.
- Soportan la ejecución simultánea de múltiples tareas sobre diferentes procesadores.
- Cuentan con múltiples programas activos compitiendo los recursos del sistema: procesador, memoria, dispositivos periféricos.

### **1.8 Sistemas Operativos de tiempo compartido.**

Permiten simular que el sistema y sus recursos son para cada usuario. El usuario hace una petición a la computadora, esta la procesa tan pronto como le es posible, y la respuesta aparecerá en la terminal del usuario.

Los principales recursos del sistema (el procesador, la memoria, dispositivos de E/S), continuamente son utilizados entre los diversos usuarios, dando a cada uno la ilusión de que tiene el sistema dedicado para él mismo. Esto trae como consecuencia una gran carga de trabajo al Sistema Operativo, principalmente en la administración de memoria principal y secundaria.

Algunos ejemplos de Sistemas Operativos de tiempo compartido son Multics, OS/360 y DEC-10 y sus características son:

- Representantes populares de sistemas multiprogramados multiusuario, ejemplo: sistemas de diseño asistido por computador, procesamiento de texto, etc.
- Dan la ilusión de que cada usuario tiene una máquina para sí.
- La mayoría utilizan el algoritmo de reparto circular.
- Los programas se ejecutan con prioridad rotatoria que se incrementa con la espera y disminuye después de que el servicio se ha concedido.
- Evitan la monopolización del sistema asignando tiempos de procesador (time slot).
- La gestión de memoria proporciona protección a programas residentes.
- La gestión de archivo debe proporcionar protección y control de acceso debido a que pueden existir múltiples usuarios accediendo a un mismo archivo.

### 1.9 Sistemas Operativos Distribuidos.

Permiten distribuir trabajos, tareas o procesos, entre un conjunto de procesadores. Puede ser que este conjunto de procesadores esté en uno o diferentes equipos, en este caso es transparente para el usuario. Existen dos esquemas básicos de éstos. Un sistema fuertemente acoplado es aquel que comparte la memoria y un reloj global, cuyos tiempos de acceso son similares para todos los procesadores. En un sistema débilmente acoplado los procesadores no comparten ni memoria ni reloj, ya que cada uno cuenta con su memoria local.

Los sistemas distribuidos deben de ser muy confiables, ya que si un componente del sistema se descompone, otro componente debe de ser capaz de reemplazarlo.

Entre los diferentes Sistemas Operativos distribuidos que existen tenemos los siguientes: Sprite, Solaris-MC, Mach, Chorus, Spring, Amoeba, Taos, etc.

Características de los Sistemas Operativos distribuidos:

- Es una colección de sistemas autónomos capaces de comunicación y cooperación mediante interconexiones hardware y software.
- Gobierna la operación de un Sistema de Cómputo y proporciona abstracción de máquina virtual a los usuarios.

- Su objetivo clave es la transparencia.
- Generalmente proporcionan medios para compartir los recursos globalmente.
- Cuenta con servicios añadidos: denominación global, sistemas de archivos distribuidos, facilidades para distribución de cálculos (a través de comunicación de procesos internodos, llamadas a procedimientos remotos, etc.).

#### **1.10 Sistemas Operativos de red.**

Son aquellos sistemas que mantienen dos o más computadoras unidas a través de algún medio de comunicación (físico o no), con el objetivo primordial de poder compartir los diferentes recursos y la información del sistema.

El primer Sistema Operativo de red estaba enfocado a equipos con un procesador Motorola 68000, pasando posteriormente a procesadores Intel como Novell Netware.

Los Sistemas Operativos de red más ampliamente usados son: Novell Netware, Personal Netware, LAN Manager, Windows NT Server, UNIX, LANtastic.

#### **1.11 Sistemas Operativos paralelos.**

En estos tipos de Sistemas Operativos se pretende que cuando existan dos o más procesos que compitan por algún recurso se puedan realizar o ejecutar al mismo tiempo.

En UNIX existe también la posibilidad de ejecutar programas sin tener que atenderlos en forma interactiva, simulando paralelismo (es decir, atender de manera concurrente varios procesos de un mismo usuario). Así, en lugar de esperar a que el proceso termine de ejecutarse (como lo haría normalmente), regresa a atender al usuario inmediatamente después de haber creado el proceso.

Ejemplos de estos tipos de Sistemas Operativos son: Alpha, PVM, la serie AIX, que es utilizado en los sistemas RS/6000 de IBM.



Sus características son:

- El procesamiento paralelo exige la ejecución concurrente en la computadora de muchos programas. Ello contrasta con el procesamiento secuencial. Es un medio coste-efectivo para mejorar el rendimiento del sistema mediante la realización de actividades concurrentes en la computadora.
- El nivel más alto de procesamiento paralelo se aplica a trabajos y programas múltiples a través de la multiprogramación, el tiempo compartido y el multiprocesamiento. Este nivel exige el desarrollo de algoritmos procesables en paralelo. La implementación de algoritmos paralelos depende de la asignación eficaz de recursos limitados software - hardware a los múltiples programas que están siendo utilizados para resolver un extenso problema de cálculo.
- Descomposición de un programa en múltiples tareas.
- Explotación de la concurrencia entre múltiples instrucciones. Con frecuencia, se realiza un análisis de dependencia de datos para revelar paralelismos entre instrucciones.

El procesamiento paralelo puede abordarse en cuatro niveles de programación:

- Nivel de programación o trabajos.
- Nivel de procedimientos o tareas.
- Nivel de interinstrucciones.
- Nivel intrainstrucción.

El procesamiento paralelo es ciertamente un campo combinado de estudios. Requiere amplios conocimientos y experiencia en todos los aspectos de algoritmos, lenguajes, hardware, software, evaluación del rendimiento y alternativas de computación.

El procesamiento paralelo y el procesamiento distribuido están estrechamente relacionados. En algunos casos, utilizamos ciertas técnicas distribuidas para conseguir paralelismo. Conforme la tecnología de las comunicaciones de datos progresa, la distinción entre procesamiento paralelo y procesamiento distribuido se hace más y más pequeña. En este sentido podemos imaginar el procesamiento distribuido como una forma de procesamiento paralelo en un entorno especial.

La mayoría de los fabricantes de computadoras comenzaron por el desarrollo de sistemas con un solo procesador central, denominados sistemas monoprocesador.

Los sistemas monoprocesadores tienen un límite en la consecución de alto rendimiento. La potencia de cálculo en un monoprocesador puede incrementarse si se permite el uso de elementos de procesamiento paralelo bajo el mando de un controlador. También se puede ampliar la estructura del computador para incluir procesadores múltiples con espacio de memoria y periféricos compartidos bajo control de un sistema operativo integrado, a este tipo de computador se le denomina sistema multiprocesador.

Para concluir con este capítulo es necesario tomar en cuenta que los Sistemas Operativos surgen para satisfacer las necesidades de cómputo de acuerdo a las características de la problemática que surge en el contexto del manejo de la información.

## CAPÍTULO 2

### LICENCIA OPEN SOURCE

En este capítulo se analizará el marco legal del software de código abierto, el significado de términos como software de código abierto y shareware. También se estudiarán las legislaciones aplicables en el país y como afecta o no el uso de Software Libre.

Lo primero que se hará es definir los tipos de Software que existen; su clasificación se muestra en la figura 2.1.

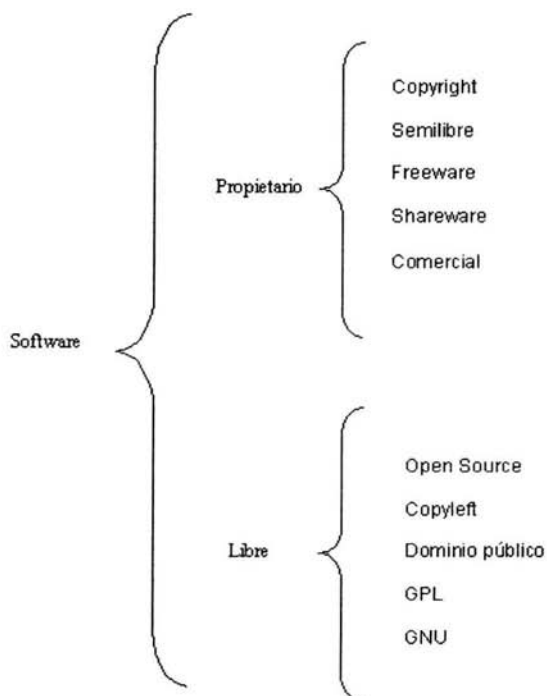


Fig. 2.1 Clasificación del Software

## 2.1 Conceptos Relacionados con el Software

### Software Libre

El Software Libre es aquel que viene con autorización para que cualquiera pueda usarlo, copiarlo y distribuirlo, ya sea literal o con modificaciones, gratis o mediante una gratificación. Si un programa es libre, entonces puede ser potencialmente incluido en un Sistema Operativo libre tal como GNU, o sistemas GNU/Linux libres<sup>4</sup>.

### Software de Open Source

El término software de "Open Source" es usado por algunas personas para dar a entender que existe la libertad por parte de los creadores para que cualquier individuo pueda modificar el programa en el código en el cual se escribió la aplicación.

### Software Copyright

El software Copyright es aquel que tiene la siguiente leyenda: Copyright © año inicial-año terminal Nombre de la Compañía. Derechos Reservados. Este programa esta protegido por las leyes de derechos de autor y otros tratados internacionales. La reproducción o distribución ilícitas de este programa o cualquier parte del mismo, esta penada por la ley con severas sanciones civiles y penales y será objeto de todas las acciones judiciales que correspondan.

### Software protegido con copyleft

El software protegido con copyleft es Software Libre cuyo término de distribución no permite agregar restricciones adicionales al redistribuir o modificar el software. Esto significa que cada copia del software, aun si ha sido modificado, debe ser Software Libre.

---

<sup>4</sup> CHRIS DIBONA, SAM OCKMAN, MARK STONE, Open Sources: Voices from the Open Source Revolution, U.S.A., O'Reilly, 1999, 280 p.

**Software de dominio público**

El software de dominio público es software que no está protegido con copyright.

**Software semilibre**

El software semilibre es software que no es libre, pero tiene autorización para ser usado, copiado, distribuido (ejemplo: el Internet Explorer de Microsoft) sin fines de lucro. PGP (Pretty Good Privacy: es un programa que sirve para encriptar mensajes) es un ejemplo de un programa semilibre. El software semilibre no se puede usar en un Sistema Operativo completamente libre.

**Software propietario**

El software propietario es software que no es libre ni semilibre. Su redistribución o modificación está prohibida, y para usarlo se requiere autorización (Licencia).

**Freeware**

El término "freeware" no tiene una definición clara, pero se usa comúnmente para paquetes que permiten la redistribución pero no su modificación; su código fuente no está disponible.

**Shareware**

El shareware es software que viene con autorización de redistribuir copias, y su uso se autoriza por un tiempo determinado, pero si después de ese tiempo se continua usando, se deberá pagar un cargo por licencia (Ejemplo WinZIP<sup>5</sup>).

El shareware no es Software Libre, ni siquiera semilibre debido a:

- 1) El código fuente no esta disponible.
- 2) No se puede usar por tiempo indefinido sin pagar una cantidad por licencia, ni aún para particulares involucrados en actividades no lucrativas.

---

<sup>5</sup> WINZIP. Software utilizado para comprimir archivos.

## Software Comercial

El software comercial es el que se desarrolla por una entidad que tiene la intención de hacer dinero por medio de las personas o instituciones que hacen uso del software. "Comercial" y "Propietario" no es lo mismo, la mayoría del software comercial es propietario, pero hay Software Libre comercial y hay software no libre no comercial.

## 2.2 Proyecto GNU

### ¿Qué es GNU?

GNU es un proyecto que tiene por objetivo desarrollar un Sistema Operativo completo tipo Unix que sea Software Libre: el sistema GNU (GNU es un acrónimo recursivo para "GNU No es Unix"; y se pronuncia "gnu", tal y como se escribe).

### Historia de GNU

En enero de 1984 Richard Stallman renuncia a MIT y comienza a escribir software GNU. Él considero necesario abandonar MIT, para que el MIT no interfiriera con la distribución de GNU como Software Libre (si hubiese continuado como parte del staff, el MIT podría haber reclamado propiedad sobre su trabajo, y podría haber impuesto sus propios términos de distribución, o incluso podría haberlo transformado en un paquete de software propietario) y enfocarse a una nueva misión: crear una nueva comunidad para compartir software; el Profesor Winston, entonces a cargo del laboratorio de IA del MIT, le invitó amablemente a que continuara utilizando las instalaciones del laboratorio.

Poco después de comenzar en el proyecto GNU, Stallman escucho acerca del Free University Compiler Kit [Kit de Compilador de la Universidad Libre], también conocido como VUCK. Se trataba de un compilador diseñado para manejar múltiples lenguajes, C y Pascal entre ellos, y para admitir múltiples máquinas destino. Stallman escribió a su autor para consultarle si GNU podría utilizarlo. Él le respondió burtonamente, que la universidad era libre, pero el compilador no. Por lo tanto, Stallman decidió que su primer programa para el proyecto GNU sería un compilador multilenguaje, multiplataforma.

Con la esperanza de evitar tener que escribir él mismo, todo el compilador, obtuvo el código fuente del compilador Pastel, que era un compilador multiplataforma desarrollado en el «Lawrence Livermore Lab». Admitía, y estaba escrito en una versión extendida de Pascal, diseñada para usarse como lenguaje de programación a nivel de sistema. Le agrego un front end para C, y comenzó a transportarlo a la computadora Motorola 68000, pero tuvo que abandonar la idea al descubrir que el compilador necesitaba varios megabytes de espacio en la pila, y los sistemas Unix basados en 68000 sólo permitían 64 kbytes.

Fue entonces cuando se dio cuenta que el compilador Pastel funcionaba analizando el archivo de entrada completo y transformándolo en un árbol sintáctico, luego convertía todo el árbol sintáctico en una cadena de «instrucciones» y generaba el archivo entero de salida, y en ningún momento liberaba el espacio ocupado. En ese momento llegó a la conclusión de que debería escribir un nuevo compilador partiendo desde cero. Ese nuevo compilador se conoce ahora como GCC; no hay nada del compilador Pastel en él, pero se las arregló para adaptar y usar el front end que había hecho para C.

### **¿Es libre el programa para cualquier usuario?**

Si un programa es Software Libre cuando abandona las manos de su autor, esto no significa que será Software Libre para todos los que tienen una copia de él. Por ejemplo, el software de dominio público (software que no está sujeto al copyright de nadie) es Software Libre; pero cualquiera puede hacer una versión modificada propietaria a partir de él. En ese mismo sentido, muchos programas libres están sujetos a copyright pero se distribuyen mediante licencias permisivas que admiten las versiones modificadas propietarias.

El ejemplo paradigmático de este problema es el X Window System. Desarrollado en el MIT, y entregado como Software Libre con un licencia permisiva, fue rápidamente adoptado por varias compañías de computación. Éstas agregaron X a sus sistemas Unix propietarios, sólo en formato binario, y lo hicieron propietario.

Los desarrolladores del X Window System no consideraban que esto fuese un problema esperaban y buscaban que esto sucediese, su meta no era la libertad, sólo el «éxito», definido como «tener muchos usuarios». No les preocupaba si buscaban la libertad del software, sólo que fueran muchos.

Esto conlleva a una situación paradójica en la cual dos maneras distintas de contabilizar la cantidad de libertad dan por resultado dos respuestas distintas a la pregunta “¿Este programa es libre?”. Si se juzga en base a la libertad que se proporcionaba con los términos de distribución de la entrega del MIT, diría que X es Software Libre. Pero si se mide la libertad del usuario promedio de X, diría que X es software propietario. La mayoría de los usuarios de X usan las versiones propietarias que vienen con los sistemas Unix, no la versión libre.

### **Copyleft y la GNU GPL**

La meta de GNU era dar libertad a los usuarios, no sólo ser popular. Por lo tanto, debía usar términos de distribución que impidieran que el software GNU se transformara en software propietario. El método utilizado se denomina «copyleft»<sup>6</sup>

La idea central del copyleft es que cualquiera posee el permiso para correr el programa, copiar, modificar y redistribuir versiones modificadas pero no tiene permiso para agregar restricciones propias. De esta manera, las libertades cruciales que definen al “Software Libre” quedan garantizadas para cualquiera que tenga una copia.

Para que el copyleft sea efectivo, las versiones modificadas deben ser también libres. Esto asegura que todo trabajo basado en el Software Libre quedara disponible para toda la comunidad si se publica. Cuando los programadores que tienen trabajo como programadores se ofrecen como voluntarios para mejorar un software GNU, el copyleft impide que digan: “no puedo compartir esos cambios, porque se quieren usar para hacer una versión propietaria del programa”.

El requerimiento de que los cambios deben ser libres es esencial si se quiere asegurar la libertad para cada usuario del programa. Las compañías que privatizaron el X Window System en general realizaron algunos cambios para transportarlo a sus sistemas y hardware. Estos cambios fueron pequeños comparados con el gran tamaño de X, pero no fueron triviales. Un tema relacionado trata la combinación de un programa libre con código no libre. Tal combinación será inevitablemente no-libre; cualesquiera libertades que falten a la parte no-libre, le faltarán también al todo. Si se permiten tales combinaciones se tendrá un grave problema. Por ello, un requerimiento

---

<sup>6</sup> El copyleft usa la ley de copyright, pero la da vuelta para servir a lo opuesto de su propósito usual: en lugar de ser un medio de privatizar el software, se transforma en un medio de mantener libre al software.



crucial para el copyleft es garantizar que cualquier cosa agregada a o combinada con un programa bajo copyleft debe ser también libre y bajo copyleft.

La implementación específica de copyleft que se usa para la mayoría del software GNU es la Licencia Pública General de GNU (GNU General Public License) de la cual se hablara a continuación.

### 2.3 Implicaciones Legales

La transcripción de la licencia completa se muestra a continuación.

#### LICENCIA PÚBLICA GENERAL GNU

##### TRADUCCION DE CORTESIA DE LA LICENCIA PÚBLICA GENERAL GNU

*Version 2, junio de 1991. Copyright (C) 1989 Free Software Foundation, Inc. 075 Mass Ave., Cambridge, MA 02139 USA.*

*Se permite copiar y distribuir copias exactas de este documento de licencia pero no está permitido hacerle cambio alguno.*

#### **I. Preámbulo**

*Para la mayoría de los software se idean licencias que le quitan a usted la libertad de compartirlos y cambiarlos. En cambio la licencia Pública General GNU tiene como objetivo garantizarle a usted la libertad de compartir y cambiar Software Libre, asegurarse de que el software es libre para todos sus usuarios. Esta licencia Pública General es aplicable a la mayoría del software de la Free Software Foundation así como a cualquier otro programa cuyos autores se comprometan a usarlo. ( Algunos de los programas de la Free Software Foundation están cubiertos por la Licencia Pública General para Bibliotecas GNU). Usted puede aplicarlo a sus programas también.*

*Cuando hablamos de Software Libre, nos referimos a la libertad y no al precio. Nuestras Licencias Públicas Generales están pensadas para asegurar que usted tiene la libertad de distribuir copias del Software Libre (y cobrar por el servicio si así lo desea)*

*a) que pueda usted recibir el código fuente de los programas o bien que lo obtenga si lo quiere usted, que pueda usted cambiar el software o bien usar partes del mismo en nuevos programas libres y que sepa usted que puede hacer todas estas cosas.*

*Para proteger sus derechos, es necesario que impongamos restricciones que impidan que le nieguen a usted dichos derechos o bien que le pidan que renuncie*

*usted a los mismos. Estas restricciones se traducen en ciertas responsabilidades que debe usted asumir si distribuye usted copias del software o si lo modifica usted.*

*Por ejemplo, si distribuye usted copias de un programa de esa naturaleza, ya se gratis o por una cuota, deberá usted conceder a los receptores todos los derechos que usted tiene. Deberá asegurarse, asimismo, de que reciban o puedan recibir el código fuente. Y deberá mostrarles estas estipulaciones a fin de que conozcan sus derechos.*

*Protegemos sus derechos mediante dos medidas:*

*(1) con el copyright del software y*

*(2) proporcionándole esta licencia que le permite legalmente copiar, distribuir y/o modificar el software.*

*Asimismo, como una manera de protección de cada autor y de la nuestra propia, queremos estar seguros de que todos entienden que no existe garantía para este Software Libre. Si el software es modificado por otra persona y lo hace circular, queremos que los que lo reciben sepan que lo que obtienen no es el original, de modo que los problemas que hayan introducido otras personas no repercutan en el prestigio de los autores.*

*Por último, cualquier programa libre se ve amenazado constantemente por las patentes de software. Deseamos evitar el peligro de que los redistribuidores de un programa libre obtengan individualmente licencias patentadas que los convierta en realidad en los propietarios del programa. Para impedirlo, hemos dejado muy claro que cualquier patente debe autorizarse para el uso libre de todo el mundo o no autorizarse.*

A continuación detallamos los términos y las condiciones precisas para copiar, distribuir o modificar.

## **II. LICENCIA PÚBLICA GENERAL GNU. TERMINOS Y CONDICIONES PARA COPIAR, DISTRIBUIR O MODIFICAR.**

*Esta Licencia es aplicable a cualquier programa u otra obra que contenga una leyenda, colocada ahí por el dueño del copyright, que especifique que puede ser distribuido de acuerdo con los términos de esta Licencia Pública General. El "Programa", más abajo mencionado, se refiere a cualquier programa y obra de esa naturaleza, y una "obra basada en el Programa" significa, ya sea el Programa o cualquier obra derivada bajo la ley copyright: es decir, una obra que contenga el Programa o una porción del mismo, ya sea una copia exacta o que contenga modificaciones y/o una traducción a otra lengua, (Más adelante, se incluye sin limitación alguna la traducción en el término "modificación"), Todo aquel que recibe la licencia será denominado "usted".*

*Otras actividades que no sean la de copiar, distribuir y modificar no están cubiertas por esta Licencia, quedan fuera de su alcance. No está restringido el acto de correr el Programa, y el producto del Programa está cubierto únicamente si su contenido constituye una obra basada en el Programa (independientemente de que se haya hecho corriendo el Programa). Si ello es cierto o no depende de lo que hace el Programa.*

*1. Usted puede copiar y distribuir copias exactas del código fuente del Programa tal y como lo recibe, en cualquier medio, con tal que publique usted de manera conspicua y apropiada en cada copia una leyenda apropiada de copyright y de no garantía; de que mantenga usted intactas todas las advertencias referentes a esta Licencia y al hecho de la ausencia de cualquier garantía; y de que proporcione usted a cualquier otro receptor del Programa una copia de esta Licencia junto con el Programa. **Usted puede cobrar una cuota** por el acto físico de transferir una copia, y puede usted, como una opción, ofrecer protección de garantía a cambio de el dinero que recibe.*

*2. Puede usted modificar su copia o copias del Programa o de una parte del mismo formando así una obra basada en el Programa y copiar y distribuir dichas*

*modificaciones o bien trabajar de acuerdo con los términos especificados en la Sección 1, a condición de que cumpla usted asimismo con estas condiciones:*

*a) Debe usted hacer que los archivos modificados lleven una advertencia prominente en que se asiente que cambió usted los archivos e incluya la fecha en que se hizo cualquier cambio.*

*b) Debe usted hacer que cualquier obra que distribuya usted o que publique, que, ya sea totalmente o en parte, contenga o se derive del Programa o de alguna de sus partes, sea también una licencia que pueda ser utilizada total o parcialmente por terceras personas sujetándose a los términos de esta Licencia.*

*c) Si el programa modificado normalmente lee las órdenes interactivamente al ser ejecutado, debe usted hacer que, al correr el programa para dicho uso interactivo de la manera más corriente, desde el inicio imprima usted o despliegue un anuncio que incluya una advertencia apropiada de copyright y una advertencia de no garantía (o bien, especificando que usted proporciona la garantía) y que los usuarios pueden redistribuir el programa bajo estas condiciones, y diciéndole al usuario cómo puede ver una copia de esta Licencia. (Excepción: si el Programa original en sí, es interactivo pero normalmente no imprime un anuncio de esa naturaleza, su obra basada en el Programa no tendrá que cumplir con el requisito de la impresión de un anuncio).*

*Estos requisitos se aplican a la obra modificada en su conjunto. Si algunas secciones identificables de dicha obra no se derivan del Programa, y pueden ser consideradas como obras razonablemente independientes y aparte en sí mismas, entonces esta Licencia y los términos de la misma no se aplicarán a esas secciones cuando las distribuya usted como obras aparte, Empero, cuando distribuya usted las mismas secciones como parte de un todo, que es una obra basada en el Programa, la distribución de ese todo deberá conformarse a los términos de esta Licencia, cuyos permisos concedidos a otros detentores de la licencia se extienden al conjunto total y, por consiguiente, a cualquier parte sin importar quien la haya escrito.*

*Por ende, no es la intención de esta sección reclamar derechos o impugnar sus derechos sobre obra escrita completamente por usted; más bien, lo que intenta es*

*ejercer el derecho de control sobre la distribución de obras derivadas o colectivas que se basan en el Programa.*

*Es más, el hecho único de agregar otra obra que no está basada en el Programa (o una obra basada en el Programa) dentro de un medio de almacenamiento o de distribución no implica que esa otra obra quede dentro del alcance de esta Licencia.*

*3. Puede usted copiar y distribuir el Programa (o una obra basada en él, según lo especificado en la Sección 2) en código de salida o en forma ejecutable de acuerdo con los términos expresados en las Secciones 1 y 2 a condición de que satisfaga usted una de las siguientes condiciones:*

- a) Acompañarlo del código fuente completo correspondiente en forma legible por máquinas, el cual deberá ser distribuido bajo los términos de las Secciones 1 y 2 arriba mencionadas utilizando un medio usado habitualmente en el intercambio de software; o,*
- b) Acompañarlo de un ofrecimiento escrito, válido por lo menos por tres años, de proporcionar a cualquier tercero interesado mediante el cobro exclusivamente del gasto hecho al proceder físicamente a la colocación en un medio y distribución de la fuente, una copia completa, en forma legible por máquinas, del correspondiente código fuente, para su distribución bajo los términos especificados en las Secciones 1 y 2 arriba mencionadas, en un medio utilizado habitualmente para el intercambio de software; o,*
- b) Acompañarlo de la información que recibió usted con respecto al ofrecimiento de distribución del correspondiente código fuente. (Esta alternativa es permitida únicamente en la distribución no comercial y solamente si recibió usted el programa*
- c) en código de salida o en forma ejecutable junto con dicho ofrecimiento, de acuerdo con la Subsección b arriba mencionada).*

*El código fuente de una obra significa la forma preferida de la obra para hacerle modificaciones. Para una obra ejecutable, el código fuente completo significa todo el código fuente para todos los módulos que contiene, más cualquier archivo de definición asociado a la interfase, más los documentos utilizados para controlar la*

*compilación y la instalación del ejecutable. Sin embargo, como excepción especial, el código fuente distribuido no necesariamente debe incluir algo de lo que normalmente es distribuido (ya sea en forma fuente o en forma binaria) con los principales componente (compilador, kernel, etc.) del Sistema Operativo en el que corre el ejecutable, a menos que ese componente sea parte del ejecutable. Si la distribución del ejecutable o del código de salida se hace ofreciendo el acceso para copiar desde un lugar designado, ofreciendo en seguida acceso equivalente para copiar el código fuente desde ese mismo lugar cuenta como distribución del código fuente, aunque terceras personas no son obligadas a copiar la fuente junto con el código de salida.*

*4. No puede usted copiar, modificar, sublicenciar o distribuir el Programa si no es siguiendo las disposiciones que rigen expresamente en esta Licencia. Cualquier intento de copiar, modificar, sublicenciar o distribuir el Programa de otra manera será nulo y automáticamente pondrá fin a sus derechos bajo esta Licencia. Sin embargo, los que hayan recibido copias, o derechos de usted bajo esta Licencia no perderán sus licencias mientras sigan cumpliendo totalmente las obligaciones aquí impuestas.*

*5. No se le exige que acepte usted esta Licencia, ya que no la ha firmado. Sin embargo, ninguna otra cosa le otorga permiso para modificar o distribuir el Programa o sus obras derivadas. Estas acciones están prohibidas por la ley si no acepta usted esta Licencia. Por consiguiente, al modificar o distribuir el Programa (o cualquier obra basada en el Programa) está usted indicando que acepta esta Licencia para hacerlo, así como todos sus términos y condiciones para el copiado, la distribución o modificación del Programa o de obras basadas en él.*

*6. Cada vez que redistribuye usted el Programa (o cualquier obra basada en el Programa), el receptor recibe automáticamente una licencia para copiar, distribuir o modificar el Programa de acuerdo con estos términos y condiciones que el otorgante original de la licencia concede. No puede usted imponer ninguna otra restricción al ejercicio de los derechos otorgados aquí al receptor. No es usted responsable de hacer cumplir esta Licencia a terceros.*

*7. Si como consecuencia de una sentencia de un tribunal o de un alegato de violación de patentes o de cualquier otro motivo (no limitado a cuestiones*

*patentarías) se le imponen a usted condiciones (ya sea por orden de un tribunal, por acuerdo o de otro modo) que contradicen las condiciones de esta Licencia no por ello queda usted excusado de las condiciones de esta Licencia. Si no puede usted distribuir sin cumplir simultáneamente con sus obligaciones impuestas por esta Licencia, así como con otras obligaciones pertinentes, entonces, consecuentemente, no puede usted distribuir el Programa. Por ejemplo, si una licencia patentaría no permitiera la redistribución del Programa, libre de regalías, a aquellos que recibieran copias directa o indirectamente a través de usted, en ese caso la única forma de cumplir con esa licencia y con esta Licencia sería abstenerse por completo de distribuir el Programa. Si alguna porción de esta sección, por alguna circunstancia particular, es juzgada nula o imposible de ejecutar, el resto de la sección es aplicable y la sección como un todo también está pensada como aplicable en otras circunstancias. Esta sección no se propone inducirle a usted a infringir patentes u otros reclamos de derechos de propiedad o a impugnar la validez de cualquiera de esas reclamaciones; esta sección no tiene más objetivo que el de proteger la integridad del sistema libre de distribución de software, el cual se lleva a cabo mediante reglas de licencias públicas. Muchos son los que han contribuido generosamente a la amplia extensión de software distribuido a través de ese sistema, confiando en la aplicación consistente de dicho sistema; es el autor/donante quien habrá de decidir si él o ella desea distribuir software a través de otro sistema diferente y la persona que obtiene la licencia no puede imponer esa decisión. Esta sección intenta dejar completamente claro lo que es considerado como una consecuencia del resto de esta Licencia.*

*8. Si la distribución y/o uso del Programa se halla restringida en algunos países, ya sea por patentes o por interfases con copyright, el detentor original del copyright que pone el Programa bajo esta Licencia puede añadir una limitación explícita de distribución geográfica excluyendo a dichos países, de modo que se permita la distribución sólo en y entre los países que no hayan sido excluidos. En tales casos, esta Licencia incorpora la limitación como si se hallara escrita en el cuerpo de esta Licencia.*

*9. La Free Software Foundation puede publicar versiones revisadas o nuevas de la Licencia Pública General de vez en cuando. Dichas versiones nuevas serán esencialmente similares a la presente versión, pero pueden variar en los detalles al*

*abordar problemas e intereses nuevos. Cada versión recibe un número distintivo de versión, Si el Programa especifica un número de versión de esta Licencia que se le aplica y a "cualquier versión posterior" tiene usted la opción de adherirse a los términos y condiciones ya sea de esa versión o de cualquier otra versión posterior que publique la Free Software Foundation.*

*Si el Programa no especifica un número de versión de esta Licencia, podrá usted escoger cualquier versión que haya publicado la Free Software Foundation.*

*10. Si desea usted incorporar partes del Programa en otros programas libres cuya distribución se somete a condiciones distintas, escriba usted al autor pidiendo su autorización. Para el software cuyo copyright pertenece a la Free Software Foundation, escriba usted a Free Software Foundation; a veces hacemos excepciones. Nuestra decisión se guiará por las dos metas que tenemos la de conservar el estatus libre de todos los derivados de nuestro Software Libre y la de promover el compartimiento y reutilización del software en general.*

#### **NO GARANTIAS.**

*11. COMO EL PROGRAMA TIENE UNA LICENCIA LIBRE DE GASTOS, NO HAY GARANTIA PARA EL PROGRAMA, HASTA DONDE LO PERMITE LA LEY APLICABLE. CON EXCEPCION DE LOS CASOS EN LOS QUE POR ESCRITO SE DECLARE LO CONTRARIO. LOS DETENTORES DEL COPYRIGHT Y/O OTROS PARTICIPES PROPORCIONAN EL PROGRAMA "TAL CUAL" SIN GARANTIA DE NINGUNA CLASE, YA SEA EXPLICITA O IMPLICITA, INCLUYENDO, PERO NO LIMITADO A, LAS GARANTIAS DE COMERCIALIZACION IMPLICITAS Y LA ADECUACION A UN PROPOSITO PARTICULAR. ES USTED QUIEN TIENE QUE CORRER TODO EL RIESGO EN LA CALIDAD Y EL RENDIMIENTO DEL PROGRAMA. EN EL CASO DE QUE EL PROGRAMA RESULTE DEFECTUOSO, ASUMIRA USTED EL COSTO DEL SERVICIO, REPARACION O CORRECCION.*

*12. EN NINGUN CASO, A MENOS QUE ASI LO REQUIERA LA LEY APLICABLE O SE HAYA ACORDADO POR ESCRITO, UN DETENTOR DEL COPYRIGHT O CUALQUIER OTRO QUE PUEDA MODIFICAR Y/O REDISTRIBUIR EL PROGRAMA TAL Y COMO ESTA PERMITIDO, SERA RESPONSABLE ANTE USTED POR DAÑOS, INCLUSO POR DAÑOS GENERALES, ESPECIALES, INCIDENTALES O*



*CONSECUENTES SURGIDOS POR EL USO O LA INCAPACIDAD DE USO DEL PROGRAMA (INCLUYENDOSE, PERO NO LIMITANDOSE A, LA PERDIDA DE DATOS O A DATOS QUE RESULTAN INEXACTOS O A PERDIDAS SUFRIDAS POR USTED O POR TERCEROS O BIEN PORQUE EL PROGRAMA NO PUEDA OPERAR CON NINGUN OTRO PROGRAMA), AUN CUANDO EL POSEEDOR DEL COPYRIGHT O LA OTRA PARTE HAYAN SIDO ADVERTIDOS DE LA POSIBILIDAD DE DICHOS DAÑOS.*

*FIN DE LOS TERMINOS Y CONDICIONES.*

**Apéndice: Como aplicar estos términos a sus nuevos programas.**

**Si crea usted un programa nuevo y desea que se generalice lo más posible su uso, la mejor manera de lograrlo es haciéndolo un Software Libre que cualquiera pueda redistribuir y cambiar bajo estos términos.**

**Para hacerlo, incluya las siguientes advertencias en el programa. Lo más seguro es ponerlas en el inicio de cada archivo fuente con el objeto de comunicar lo más eficazmente posible la exclusión de garantía; y cada archivo debería tener por lo menos la línea "copyright" y un indicador de dónde se encuentra la advertencia completa.**

< una línea para proporcionar el nombre del programa y una breve idea de lo que es capaz de hacer>

Copyright (C) año < nombre del autor>

Este programa es un Software Libre; puede usted redistribuirlo y/o modificarlo de acuerdo con los términos de la Licencia Pública General GNU publicada por la Free Software Foundation: ya sea en la versión 2 de la Licencia, o (a su elección) en una versión posterior.

Este programa es distribuido con la esperanza de que sea útil, pero SIN GARANTIA ALGUNA; incluso sin la garantía implícita de COMERCIABILIDAD o DE ADECUACION A UN PROPOSITO PARTICULAR. Véase la Licencia Pública General GNU para mayores detalles.

Debería usted haber recibido una copia de la Licencia Pública General GNU junto con este programa; de no ser así, escriba a Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA02139, USA.

Agregue también información sobre cómo entrar en contacto con usted por correo electrónico y correo ordinario.

Si el programa es interactivo, haga que salga una corta advertencia como ésta cuando empiece el modo interactivo.

Ejemplo:

```
Gnomovision versión 09, Copyright (C) 19yy nombre del autor
Gnomovision no contiene GARANTIA ALGUNA: para mayores detalles
escriba 'show w'.
Este es un Software Libre y puede usted redistribuirlo bajo ciertas
condiciones; escriba 'show c' para más detalles.
```

Las hipotéticas órdenes "show w" y "show c" deberán mostrar las partes apropiadas de la Licencia Pública General. Por supuesto, las órdenes que use usted pueden llamarse de otra manera, incluso pueden ser tictacs del ratón o bien artículos de un menú, cualquier cosa que sea apropiada para su programa. Deberá también hacer que su empleador (si trabaja usted como programador) o su escuela, si la tiene, firme una "renuncia de copyright" para el programa, en caso necesario. He aquí un ejemplo; cambie los nombres:

Yoyodyne, Inc., por este medio renuncia todo interés en el copyright del programa 'Gnomovision' (que le hace pases a los compiladores) escrito por Joe Doe.

< firma de TyCoon> , 1º de abril de 1989 TyCoon, Presidente de Vice

La Licencia Pública General no permite la incorporación de su programa en programas patentados. Si su programa es una subrutina bibliotecaria puede que considere usted más útil permitir la vinculación de las aplicaciones patentadas con la biblioteca. Si es esto lo que quiere hacer, use la Licencia Pública General Bibliotecaria GNU en lugar de esta Licencia.

**Análisis de Legislaciones Vigentes Acerca de Derechos de Autor**

Una vez revisada la licencia GNU se analizara que tanto las Leyes y Reglamentos aplicables en nuestro territorio podrían afectar la libre distribución del software.

En la **Ley Federal de Derechos de Autor Título Primero Artículo 3** (Ultima reforma aplicada 19/05/1997).

*Las obras protegidas por esta Ley son aquellas de creación original susceptibles de ser divulgadas o reproducidas en cualquier forma o medio.*

Por lo que podemos decir que esta disposición ampara nuestro derecho a la libertad de distribución de software de código abierto<sup>7</sup>.

En la **Ley Federal de Derechos de Autor Título Primero Artículo 5** ( Ultima reforma aplicada 19/05/1997).

*La protección que otorga esta Ley se concede a las obras desde el momento en que hayan sido fijadas en un soporte material, independientemente del mérito, destino o modo de expresión. **El reconocimiento de los derechos de autor y de los derechos conexos no requiere registro ni documento de ninguna especie ni quedará subordinado al cumplimiento de formalidad alguna.***

En la **Ley Federal de Derechos de Autor Título Primero Artículo 7** (Ultima reforma aplicada 19/05/1997).

Los extranjeros autores o titulares de derechos y sus causahabientes gozarán de los mismos derechos que los nacionales, en los términos de la presente Ley y de los tratados internacionales en materia de derechos de autor y derechos conexos suscritos y aprobados por México.

En la **Ley Federal de Derechos de Autor Capítulo III Artículo 27** (Ultima reforma aplicada 19/05/1997).

---

<sup>7</sup> MÉXICO, Ley Federal de Derechos de Autor, México, McGraw-Hill Interamericana, 1998, 113 p.

Los titulares de los derechos patrimoniales podrán **autorizar o prohibir**:

*I. La reproducción, publicación, edición o fijación material de una obra en copias o ejemplares, efectuada por cualquier medio ya sea impreso, fonográfico, gráfico, plástico, audiovisual, electrónico u otro similar;*

*II. La comunicación pública de su obra a través de cualquiera de las siguientes maneras:*

*a) La representación, recitación y ejecución pública en el caso de las obras literarias y artísticas;*

*b) La exhibición pública por cualquier medio o procedimiento, en el caso de obras literarias y artísticas, y*

*c) El acceso público por medio de la telecomunicación;*

*III. La transmisión pública o radiodifusión de sus obras, en cualquier modalidad, incluyendo la transmisión o*

*retransmisión de las obras por:*

*a) Cable;*

*b) Fibra óptica;*

*c) Microondas;*

*d) Vía satélite, o*

*e) Cualquier otro medio análogo;*

*IV. La distribución de la obra, incluyendo la venta u otras formas de transmisión de la propiedad de los soportes materiales que la contengan, así como cualquier forma de transmisión de uso o explotación. Cuando la distribución se lleve a cabo mediante venta, este derecho de oposición se entenderá agotado efectuada la primera venta, salvo en el caso expresamente contemplado en el artículo 104 de esta Ley;*

*V. La importación al territorio nacional de copias de la obra hechas sin su autorización;*

*VI. La divulgación de obras derivadas, en cualquiera de sus modalidades, tales como la traducción, adaptación, paráfrasis, arreglos y transformaciones, y*

*VII. Cualquier utilización pública de la obra salvo en los casos expresamente establecidos en esta Ley.*

Con lo que se entiende que los derechos de libre distribución son posibles para la aplicación de software de código abierto.

En la **Ley Federal de Derechos de Autor Capítulo VI Artículo 102** ( Última reforma aplicada 19/05/1997).

Los programas de computación se protegen en los mismos términos que las obras literarias. Dicha protección se extiende tanto a los programas operativos como a los programas aplicativos, ya sea en forma de código fuente o de código objeto. **Se exceptúan aquellos programas de cómputo que tengan por objeto causar efectos nocivos a otros programas o equipos.**

Este artículo dice que el Software Libre tiene los mismos derechos que el no libre por lo que el shareware queda protegido a excepción de lo que dice el siguiente artículo.

En la **Ley Federal de Derechos de Autor Capítulo VI Artículo 103** ( Última reforma aplicada 19/05/1997).

*Salvo pacto en contrario, los derechos patrimoniales sobre un programa de computación y su documentación, cuando hayan sido creados por uno o varios empleados en el ejercicio de sus funciones o siguiendo las instrucciones del empleador, corresponden a éste. Como excepción a lo previsto por el artículo 33 de la presente Ley, el plazo de la cesión de derechos en materia de programas de computación no está sujeto a limitación alguna.*

En la **Ley Federal de Derechos de Autor Capítulo III Artículo 33** ( Última reforma aplicada 19/05/1997).

**A falta de estipulación expresa, toda transmisión de derechos patrimoniales se considera por el término de 5 años. Sólo podrá pactarse excepcionalmente por más de 15 años cuando la naturaleza de la obra o la magnitud de la inversión requerida así lo justifique.**

Por lo que un usuario y un desarrollador se deben sentir amparados por la Ley Federal de Derechos de Autor en términos de que tenemos la libertad de la libre distribución establecida desde sus inicios como lo de la licencia GNU.

En la **Ley Federal de Derechos de Autor Capítulo VI Artículo 104** (Ultima reforma aplicada 19/05/1997).

*Como excepción a lo previsto en el artículo 27 fracción IV, el titular de los derechos de autor sobre un programa de computación o sobre una base de datos conservará, aún después de la venta de ejemplares de los mismos, el derecho de autorizar o prohibir el arrendamiento de dichos ejemplares. Este precepto no se aplicará cuando el ejemplar del programa de computación no constituya en sí mismo un objeto esencial de la licencia de uso.*

En la **Ley Federal de Derechos de Autor Capítulo VI Artículo 105** (Ultima reforma aplicada 19/05/1997).

El usuario legítimo de un programa de computación podrá realizar el número de copias que le autorice la licencia concedida por el titular de los derechos de autor, **o una sola copia de dicho programa siempre y cuando:**

*I. Sea indispensable para la utilización del programa, o*

*II. Sea destinada exclusivamente como resguardo para sustituir la copia legítimamente adquirida, cuando ésta no pueda utilizarse por daño o pérdida. La copia de respaldo deberá ser destruida cuando cese el derecho del usuario para utilizar el programa de computación.*

En estos dos últimos artículos se llega a la conclusión de que es posible conservar los derechos de libertad en cuanto a los Sistemas Operativos Open Source en nuestro país en términos de la ley Federal de derechos de Autor.

En términos del acceso al dominio público del software de código abierto se encontró

En la **Ley Federal de Derechos de Autor TITULO VI Capítulo III Artículo 152** (Ultima reforma aplicada 19/05/1997).

***Las obras del dominio público pueden ser libremente utilizadas por cualquier persona, con la sola restricción de respetar los derechos morales de los respectivos autores.***

Por lo que se esta en derecho de distribuir libremente los productos elaborados, y para terminar este capítulo consultaremos la **Ley de la Propiedad Industrial** la cual menciona en **Capítulo I Artículo 19.**

*No se considerarán invenciones para los efectos de esta Ley:*

*I.- Los principios teóricos o científicos;*

*II.- Los descubrimientos que consistan en dar a conocer o revelar algo que ya existía en la naturaleza, aún cuando anteriormente fuese desconocido para el hombre;*

*III.- Los esquemas, planes, reglas y métodos para realizar actos mentales, juegos o negocios y los métodos matemáticos;*

***IV.- Los programas de computación;***

*V.- Las formas de presentación de información;*

*VI.- Las creaciones estéticas y las obras artísticas o literarias;*

*VII.- Los métodos de tratamiento quirúrgico, terapéutico o de diagnóstico aplicables al cuerpo humano y los relativos a animales, y*

*VIII.- La yuxtaposición de invenciones conocidas o mezclas de productos conocidos, su variación de uso, de forma, de dimensiones o de materiales, salvo que en realidad se trate de su combinación o fusión de tal manera que no puedan funcionar separadamente o que las cualidades o funciones características de las mismas sean modificadas para obtener un resultado industrial o un uso no obvio para un técnico en la materia*

**De esto se puede concluir que el software que no puede ser patentado es libre para distribuirse en los términos que mas convengan al usuario y al desarrollador.**

En cuanto la licencia GNU se puede sintetizar como la libertad que tienen los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

- I. La libertad de usar el programa, con cualquier propósito (libertad 0).
- II. La libertad de estudiar cómo funciona el programa, y adaptarlo a sus necesidades (libertad 1).
- III. La libertad de distribuir copias (libertad 2).
- IV. La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie (libertad 3).

Un programa es Software Libre si los usuarios tienen todas estas libertades. Así pues, se debe tener la libertad de distribuir copias, sea con o sin modificaciones, sea gratis o cobrando una cantidad por la distribución, a cualquiera y a cualquier lugar. El ser libre de hacer esto significa (entre otras cosas) que no se tiene que pedir o pagar permisos.

Pero debemos tomar en cuenta que el fundamento legal es una de las partes esenciales que sustentan el Software Libre por ende los Sistemas Operativos, en la historia de la humanidad el hombre siempre ha creado reglas de convivencia para poder avanzar en el progreso como lo conocemos, estas reglas siempre se van modificando conforme el hombre toma conciencia de que todos los elementos que se han creado sirven para un bien común. En el caso de los fundamentos legales todavía no se ha dicho la última palabra, y en estos momentos hay una batalla legal acerca del uso de código UNIX de parte de SCO hacia toda la comunidad Linux, la legislación acerca del uso del software esta en sus inicios y hace falta interés de parte de los legisladores de nuestro país en analizar los tópicos que se refieren en éste trabajo.



## CAPÍTULO 3

### SISTEMAS OPERATIVOS FUNDAMENTADOS EN OPEN SOURCE: CASO DE ESTUDIO LINUX

La idea original que esta detrás de Linux surgió a principios de los años 90, en la Helsinki University Technology en Finlandia, de manos de un estudiante sueco llamado Linus Torvalds. Lo que empezó en 1991 como un proyecto para suministrar una alternativa al Sistema Operativo Minix (Es un clon de Unix distribuido junto con su código fuente y desarrollado por Andrew S. Tanenbaum.), se transformo en algo parecido a un movimiento cuando los primeros sistemas Linux Versión ALFA se publicaron en Internet<sup>8</sup>.

En pocos años el equipo de desarrollo de Linux, se había expandido para incluir desarrolladores de controladores, desarrolladores de Kernel y Software, muchos entusiastas trabajaron para portar el código fuente abierto de UNIX a Linux.

Lo que se consideraba un sistema Linux típico creció rápidamente hasta el punto de ser pesado y difícil de mantener. Con el tiempo los programadores tuvieron la idea de crear distribuciones para instalar Linux. Las distribuciones Slackware y SLS se hicieron muy populares, al principio se instalaban en disquetes pero con la disminución de precios de las unidades de CD-ROM, los discos ópticos se convirtieron en el medio preferido y los disquetes ya solo se utilizan para la recuperación del sistema en caso de daños.

La mayor parte de las distribuciones de Linux se diferenciaba por el conjunto de software que incluían pero a medida que pasaba el tiempo sus ofertas se diversificaron, añadiendo algunas veces software escrito específicamente para las propias distribuciones (como utilidades de empaquetamiento e interfaces graficas en un esfuerzo por diferenciarse del resto).

---

<sup>8</sup> DAVID BANDIEL Y ROBERT NAPIER, Edicion Especial Linux 6ª Edicion, Pearson Educación, Madrid, 2001

### 3.1 Elementos de Linux

La figura 3.1 muestra como esta conformado un Sistema Operativo Linux.

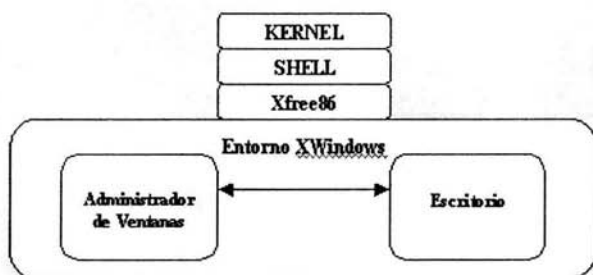


Fig. 3.1 Elementos de Linux

**Kernel:** Es el núcleo del sistema, sus responsabilidades más importantes son la administración de procesos y la administración de archivos, sin embargo el Kernel esta encargado de muchas otras cosas como la administración de la E/S, que es esencialmente la comunicación entre todos los dispositivos periféricos y la CPU.

**El Shell:** Es un interprete de comandos, a través de el usuario se comunica con la computadora y ejecuta todas las instrucciones. El más utilizado dentro de la comunidad Linux es Bash (GNU Bourne - Again Shell) que se ha convertido prácticamente en estándar y es compatible con otros shell como PDKSH (Korn de dominio publico) y TCSH (versión avanzada de un shell Unix denominado C).

**XFree86:** No es solamente un entorno grafico; es tambien un sistema cliente-servidor muy potente, diseñado para ejecutarse dentro de una red. Entorno XWindows: Es un generico que quiere decir "entorno de ventanas", que el caso de Linux se ha desarrollado gracias a un proyecto del MIT (instituto Tecnologico de Massachusetts) denominado X11 ([www.x11.org](http://www.x11.org)). Una implementacion de X11 denominada Xfree86.

### 3.2 ¿Que es un Sistema Operativo basado en Software Libre?

Un Sistema Operativo basado en Software Libre es aquel que tiene su código fuente disponible para todo publico, cuando adquirimos un Sistema Operativo como Windows que no es libre, lo único que se puede hacer con el es utilizarlo y ajustarse

a lo que ofrece, no se puede adecuarlo a las necesidades del usuario y del programador por que solo se tienen los ejecutables, en el Software Libre se cuenta con los fuentes y se puede modificar según las inquietudes del usuario y del programador.

### 3.3 ¿Que características analizaremos de los Sistemas Operativos Linux?

Después de estudiar las peculiaridades de los Sistemas Operativos en el capítulo Uno, se llega a la conclusión de que se deben evaluar las características que se muestran abajo sin embargo debemos aclarar que no todos los Sistemas Operativos cuentan con ellas a continuación se indicara lo que nos interesa estudiar de cada una:

- Multitarea.
- Multiusuario.
- Multiplataforma.
- Multiprocesador.
- Protección de la memoria entre procesos.
- Carga de ejecutables por demanda.
- Archivos compartidos ejecutables.
- Código fuente disponible.
- Conectividad.

**Multitarea:** Posee la habilidad de ejecutar varios programas al mismo tiempo, siendo el Sistema Operativo el encargado de ceder el tiempo de procesador a cada programa.

**Multiusuario:** Permite que un equipo pueda ser compartido por varios usuarios, cada uno con diferentes "privilegios" y sin acceso a la información de los demás, si así se desea. Así, cada uno puede trabajar como si se encontrara en un equipo diferente.

**Multiplataforma:** Esto implica que un Sistema Operativo cuenta con soporte para plataformas y servidores de distintas arquitecturas.

**Multiprocesador:** Esta característica se presenta cuando un Sistema Operativo tiene capacidad para más de un procesador.

**Protección de la memoria entre procesos:** Esta característica resalta que ningún proceso que ejecute el Sistema Operativo puede bloquearlo.

**Código fuente disponible:** Esta característica muestra que se cuenta con el código completo incluyendo el núcleo, los drivers, las herramientas de desarrollo y todos los programas de usuario; además que todo se pueda distribuir libremente.

**Conectividad:** Esta característica garantiza levantar todos los servicios para la comunicación entre usuarios

**Características Particulares:** Se describen las características propias de un Sistema Operativo que lo hace único y presenta una ventaja/desventaja sobre otros.

**Requisitos de instalación:** Son los requisitos mínimos para instalar el Sistema Operativo en una maquina.

### 3.4 Distribuciones de Sistemas Operativos Basados en Software Libre.

Existe una gran cantidad de distribuciones de Sistemas Operativos Basados en Software Libre, y conforme transcurre el tiempo surgen nuevos proyectos debido a la naturaleza de sus códigos fuentes disponibles a cualquier persona que cuente con una PC y conocimientos suficientes para modificarlos por ejemplo:

- ASP Linux
- Astaro
- BearOps
- Beehive
- Best Linux
- Blue Linux
- BluePoint
- BYO Linux
- Caldera

- ClarkConnect
- CLE
- Conectiva
- Corel (xandros)
- Cosix
- CRUX
- Debian
- ELX
- EnGarde
- e-smith
- Gentoo
- Hancom
- Happy Linux
- HispaFuentes
- Icepack
- Immunix
- IPCop
- JBLinux
- Kondara
- KRUD
- LASER5
- Libranet
- Lindows
- Linpus
- Linux From Scratch
- LinuxPPC
- Lycoris
- MadeinLinux

- Mandrake
- Miracle
- MIZI Linux
- MSC.Linux
- Omoikane
- OpenNA
- Owl
- Peanut
- PLD
- Progeny
- Red Flag
- Red Hat
- ROCK
- ROOT
- Slackware
- SmoothWall
- Sorcerer
- Stampede
- SuSE
- ThizLinux
- Trustix
- TurboLinux
- Vector Linux
- Voodoo
- WOWLinux
- Xandros
- Xteam
- Yellow Dog

Conforme pasa el tiempo van surgiendo nuevas distribuciones. Linux no es una moda ya que empieza a ser utilizado por empresas de gran tamaño (IBM, Oracle, Informix etc), pero de eso se hablará más adelante, las distribuciones más representativas que serán analizadas en este trabajo son:

1. Red hat
2. Debian
3. S.U.S.E
4. Mandrake

### 3.5 DISTRIBUCIÓN REDHAT LINUX

Versión 7.2



La historia de Red Hat forma parte de un grupo de programadores con base en Carolina del Norte. Su objetivo era facilitar el uso de Linux. Al igual que otros grupos parecidos, su finalidad era la reunir todos los "bits" necesarios en una distribución completa y simplificar los aspectos complejos que se presentan durante el arranque del nuevo Sistema Operativo. Su diferencia con otras instalaciones es que en vez de ser una fotocopia de un disco duro o un conjunto de disquetes a partir de los cuales se pueden formar las diferentes partes de un Sistema Operativo, esta distribución está basada en paquetes. Cada paquete proporciona una parte diferente del software, completamente comprobado, configurado y listo para ser ejecutado.

El hecho de que este Sistema Operativo se funda en un conjunto de paquetes, conllevaba otra ventaja: la distribución de Linux se puede actualizar fácilmente. Brinda el soporte para cualquier tipo de plataforma hardware, mayor fiabilidad del sistema y el uso creciente de Linux por parte de numerosas empresas en todo el mundo. Red Hat Linux funciona principalmente sobre tres plataformas: PC Intel compatibles, computadora Digital Alpha y Sun SPARC los códigos unificados y las ventajas aportadas por la tecnología RPM (RPM Package Management). Lo cual implica que es posible hacer disponible Red Hat Linux para cualquier tipo de sistema con un esfuerzo mínimo. Esto en la práctica permite gestionar y transferir el software entre estas plataformas de manera sencilla.

La mayoría de los sistemas Red Hat Linux son fáciles de instalar ya que se puede configurar en un idioma nativo y tiene soporte para autoreconocer el hardware, puntos que trataremos más adelante.

### **Características**

#### **a) Multitarea**

Definimos la multitarea como la capacidad para realizar muchas tareas simultáneamente. Por lógica, todas las tareas deberían realizarse exactamente a un tiempo, pero al disponer solo de un solo procesador las tareas no se traslapan, sino que se ejecutan de forma secuencial, pero en fracciones tan pequeñas de tiempo que da una apariencia de uniformidad, por lo que podemos decir que Red Hat satisface esta característica ampliamente al ejecutar procesos simultáneamente.

#### **b) Multiusuario**

Red hat es totalmente multiusuario. el número de usuarios simultáneos que se pueden utilizar dependen de la velocidad del procesador y de la memoria RAM.

#### **c) Multiplataforma**

Puede cargarse en cualquier plataforma de Intel x86, Intel Itanium, AMD, Cirix, 386, Alpha, PowePC, SPARC.

#### **d) Multiprocesador**

Red hat es multiprocesador soporta hasta 4 gigabytes de usuarios procesando información, elimina el limite de bus PCI y hace memorias virtuales efectivas soportando hasta 64 gigabytes de RAM.

#### **e) Protección de la memoria entre procesos**

Red hat utiliza los mecanismos de protección de memoria del procesador para evitar que el proceso acceda a la memoria del núcleo del sistema o de otros procesos. Esto es un importante logro con respecto a la seguridad del sistema. De esta forma, un programa erróneo no podrá bloquear el sistema.



**f) Código fuente disponible**

Todo los códigos fuentes están disponibles en la red, si tenemos una conexión rápida podemos descargar directamente del sitio <ftp://ftp.us.kernel.org/pub/redhat> o en <http://www.redhat.com/apps/download/>

**g) Conectividad**

Red Hat puede conectarse con otros Linux o con cualquier otro sistema usando cableado en serie, cableado paralelo, módems convencionales, tarjetas RDSI, Frame Relay, redes locales ethernet y token ring, cuanto a protocolos de red tiene TCP/IP, pero puede acceder (como cliente o como servidor, según los casos) a redes basadas en IPX (Novell), AppleTalk (Macintosh) y SMB (red LanManager para conectar con Windows para trabajo en grupo, Windows 95 y Windows NT).

**h) Características Particulares**

Red Hat ofrece guías completas, paquetes de programas y servicios. Entre sus principales contenidos y características, destacan:

- Soporte para dispositivos USB.
- XFree86 4.0.3, con gráficos en 3D.
- GNOME 1.2 y KDE 2.1.1 actualizados.
- Protocolo SSL.
- FireWalls
- Sistema de configuración de las impresoras locales y de red.

En cuanto a servidores cuenta con las siguientes características.

- Aplicaciones habilitadas para el protocolo SSL.
- Capacidad de cortafuegos ampliada y mejorada.
- Servidor DNS BIND y WU-FTP.
- Servidor de Web Apache.
- Soporte SMP de hasta un máximo de 8 procesadores y 64GB RAM.
- Herramienta de configuración gráfica para la configuración y gestión simplificada de los servicios de impresión, DNS y Apache.

- Servidor de Web Tux 2.0 de Red Hat. Récord de prestaciones en el SPECweb99TM para 1, 2, 4 u 8 procesadores.

#### i) Requisitos de instalación

Las características mínimas que debe tener un equipo son: un procesador x86 con un mínimo recomendado de 32MB de RAM, y 500MB de espacio libre en disco duro; así como Unidad de disco de 3,5" o de CD-ROM con autoarranque. Red Hat Linux soporta la mayoría de los sistemas PC Intel compatibles, incluidos los procesadores Pentium III y IV.

Red Hat Linux 7.2 es un Sistema Operativo modular construido de más de 800 paquetes individuales. Estos paquetes, sin tener en cuenta las aplicaciones y CD-ROMs extra, constituyen la base del Sistema Operativo.

### 3.6 DISTRIBUCIÓN DEBIAN

Versión 2.2



Debian fue fundada en agosto de 1993 por Ian Murdock, entonces estudiante de la Universidad de Purdue (en Indiana, EEUU), quien tuvo claro desde el primer momento que la construiría de acuerdo a las necesidades de la comunidad de usuarios del sistema y al modelo distribuido de Linux, desvinculado de intereses comerciales particulares. Unos meses después, en enero de 1994 el proyecto ya contaba con docenas de entusiastas y publicaba un manifiesto fundacional con su declaración de intenciones (el "Debian Linux Manifiesto"). En el Manifiesto, se auguraba por un lado que las distribuciones iban a adquirir un papel estratégico en la difusión de sistemas GNU/Linux, como así ha sido, y se constataban por otro lado las carencias cada vez mayores que padecían las distribuciones comerciales: sistemas descuidados y cada vez más inconsistentes, mezcla de Software Libre y no libre, dificultad para que los usuarios pudieran tomar decisiones o interviniesen en su mejora, etc. Lo cierto es que la idea de Debian de crear un Sistema Operativo libre, basado en Unix, tiene su origen varios años antes, con el proyecto GNU de la Free Software Foundation (FSF). Debian compartió desde su origen tanto los

fundamentos estratégicos del proyecto GNU como su concepto de "Software Libre" (no en el sentido de precio, sino de libertad de copia, modificación y redistribución).

Hasta 1996 aparece la versión 1.0 de Debian (en realidad, la 1.1) pues se dedicaron bastantes esfuerzos a la consolidación del proyecto no solo a nivel público y legal sino organizativa y técnicamente, para permitir un crecimiento escalar en desarrolladores y número de paquetes. Nunca hubo versión 1.0, para evitar la confusión provocada por un redistribuidor de CDs de Debian que etiquetó erróneamente una versión congelada previa como Debian 1.0-final. Ese fallo llevó a crear el concepto de "imagen oficial" del CD-ROM de Debian con idea de ayudar a los vendedores a evitar este tipo de errores. A partir de ese momento se suceden las versiones con regularidad, y cada nueva versión adopta el nombre de un personaje de la película Toy Story (hasta ahora Buzz, Rex, Bo, Hamm, Slink y Potato). Esto último fue idea de Bruce Perens, que entonces trabajaba en Pixar, la compañía que produjo la película de Walt Disney. En enero de 1998, Ian Jackson sustituye a Bruce Perens, poco antes de la liberación de Debian 2.0; quien posteriormente es sustituido por Wichert Akkerman (enero de 1999).

A juicio de muchos analistas, Debian ha alcanzado en este último año el número suficiente de usuarios y desarrolladores para garantizar su continuidad: diez distribuciones comerciales basadas en Debian lo avalan, sin embargo, Debian es construida por sus propios usuarios y es la comunidad la que dirige su estrategia: nadie puede garantizar nada en Debian, como nadie puede garantizar nada en el Software Libre por la naturaleza del contrato bajo el cual fue creado. Y sin embargo se tiene una distribución de una calidad extraordinaria, que en nada tiene que envidiar, sino todo lo contrario, a distribuciones comerciales como RedHat, Mandrake, SuSE o Caldera.

### **Características**

#### **a) Multitarea**

Debian permite ejecutar varios procesos simultáneamente. No hay ningún problema de dejar correr servidores de web, de mail, de FTP y otras aplicaciones al mismo tiempo en una sola computadora. Esto permite aprovechar el hardware eficientemente.

**b) Multiusuario**

Debian GNU/Linux permite a más de un usuario trabajar a la vez en la misma computadora (desde diferentes terminales). El sistema separa estrictamente los datos personales de los usuarios.

**c) Multiplataforma**

Debian soporta un gran número de arquitecturas diferentes. Aparte de Intel y compatibles hay versiones para Alpha, ARM, Motorola 68k, MIPS, MIPSEL, PowerPC, Sun Sparc/UltraSparc, IBM s/390 y HP PA-RISC.

**d) Multiprocesador**

Cuenta con soporte multiprocesador pero para utilizar múltiples procesadores, hay que sustituir el núcleo estándar Debian. Encontraremos instrucciones sobre cómo hacerlo en la sección "General" de la configuración del núcleo.

**e) Protección de la memoria entre procesos**

Debian al igual que todas las distribuciones de Linux cuenta con el sistema de protección de memoria, que no permite que ningún proceso detenga el sistema, ni acceda a zonas de memoria que no le pertenecen (si lo intenta, se aborta al proceso y el resto sigue como si nada, sin enterarse).

**f) Código fuente disponible**

Todo los códigos fuentes están disponibles en la red, si tenemos una conexión rápida podemos descargar directamente del sitio <http://packages.debian.org/stable/>.

**g) Conectividad**

Debian se conecta a cualquier otro sistema usando cableado en serie, cableado paralelo, módems convencionales, tarjetas RDSI, Frame Relay, redes locales ethernet y token ring, cuanto a protocolos de red tiene TCP/IP, pero puede acceder (como cliente o como servidor, según los casos) a redes basadas en IPX (Novell), AppleTalk (Macintosh) y SMB.

**h) Características Particulares**

- Kernel 2.2.20 (por defecto) y 2.4 (opcional)
- XFree86 4.1 (por defecto) y 3.3.6 (opcional)
- Gnome 1.4
- KDE 2.2
- Apache 1.3.26
- Squid 4.6-2
- Zope 2.5.1-1
- PHP 4.1-1
- cdrecord 1.10-7
- MySQL 3.23.49-8
- PostgreSQL 7.2.1-2

**i) Requisitos de instalación**

Debian GNU/Linux se instala en una i486 (o compatible) con 12 Mbyte de RAM. En el disco duro se necesitan por lo menos 64 Megabites, con ambiente gráfico 300 Megabites. Una instalación amplia ocupa alrededor de 800 Megabites.

**3.7 DISTRIBUCIÓN S.u.S.E**

Versión 8.0



SuSE Linux, tiene una plantilla de más de 340 cuarenta personas, es una compañía Linux líder internacional. La compañía ofrece un extenso abanico de servicios de consultoría cualificada y soporte para el uso de Linux en entornos empresariales, así como software comercial para Linux, y sistemas Linux completos. Más de 50.000 clientes de empresa y 1.000.000 de clientes particulares usan SuSE Linux en todo el mundo por su alta calidad y estabilidad. SuSE tiene representaciones en Alemania, los EEUU, el Reino Unido, la República Checa, e Italia. SuSe recibió el premio a la

"Mejor distribución Linux" en la exposición LinuxWorld 2000, en Nueva York. SuSE contribuye de forma importante al desarrollo de proyectos Linux tales como el propio núcleo de Linux, glibc, XFree86TM, KDE, ISDN4Linux, ALSA (Arquitectura Avanzada de Sonido de Linux) y USB (Bus Serie Universal) y en la actualidad es la única distribución que tiene certificaciones.

#### **a) Multitarea**

S.u.S.E 8.0 es totalmente multitarea.

#### **b) Multiusuario**

S.u.S.E es un entorno dedicado a las redes, por tanto es una de las mejores distribuciones para trabajar en entornos compartidos.

#### **c) Multiplataforma**

IBM Mainframe.

IBM S/390.

Intel x86.

Intel® Itanium2® Processor Family.

AMD64.

#### **d) Multiprocesador**

S.u.S.E puede soportar en la plataforma en x86 hasta 32 CPUs con un crecimiento de la potencia prácticamente lineal. El equilibrio de red, la alta disponibilidad y posibilidad de clustering acentúan aún más la escalabilidad y el rendimiento todas estas características se adecuarían a la plataforma que nos interese manejar.

#### **e) Protección de la memoria entre procesos**

En S.u.S.E la protección de la memoria, la multitarea preventiva y la gestión optimizada de la memoria o multiproceso simétrico están colocados para tener un mejor rendimiento en la red.

**f) Código fuente disponible**

El código fuente así como los ejecutables se pueden bajar de <http://www.suse.de/de/private/download/>

**g) Conectividad**

S.u.S.E soporta conexiones mediante Módem, PPP, FTP, configuración, cliente, servidor virtual, controles de acceso, APACHE, configuración, protocolos servidores virtuales, permisos, reenvío, errores. PROXY SQUID, E-MAIL, SMTP, comandos SMTP, MTA, MUA, MDA, Sendmail, configuración de Sendmail, POP3, Instalación y configuración de POP3, Pine Majordomo, listas de correo, configurar Majordomo, Usuarios y listas.

**h) Características Particulares***INTERNET*

- Netscape 6.2, Mozilla, Konqueror, Opera, etc.
- AOL Instant Messenger, IRQ, ICQ, Kinkatta, etc.
- KMail, Evolution, GNOME-Meeting, etc.
- Squid, Apache, sendmail, postfix, News, VShop, Zope, SAP DB, PostgreSQL, MySQL, etc.

*PAQUETES DE OFICINA*

- StarOffice 5.2, KOffice, KOrganizer, GNUCash (compatible con Quicken), Acrobat Reader y editores HTML (por ejemplo Quanta+), etc.
- teTeX, Latex, Ghostscript, etc.
- GIMP 1.2.3, ImageMagick, xfig, Kontour, Moonlight 3D, RIQ-convert Light, etc.
- XSane, Kooka, etc.

*Audio y Video*

- Reproductores CD, MOD, MP3 y RealPlayer, etc.

- Herramientas MIDI, sintetizador, DJ Mixer GDAM, compresión archivos de sonido Ogg/Vorbis, ALSA 0.9, etc.
- MainActor (versión de evaluación), Broadcast2000/Motv, Kwintv, etc.
- Redes
- TCP/IP, IPsec, NFS, UUCP, SLIP, IPX, netatalk (conexión a Macintosh), Samba (conexión a Windows), Token-ring, ARCnet, IPv6, DNS, radioaficionado, telefónica en Internet (H.323), etc.
- ADSL, RDSI, PPP Dial-in, DHCP, etc.

### PROGRAMACIÓN

- Programación: C/C++, Fortran, Pascal, Java (IBM, Sun), Perl, Python, PHP, Kdevelop, SWI-Prolog, etc.
- Editores: Emacs, XEmacs, Kate, vi, joe, nedit, etc.
- Administración del sistema : awk, sed, grep, find, rcs, make, groff, bison, flex, perl, m4, etc.
- Borland's Kylix Open Edition, versión 2. Entorno de desarrollo RAD completo.

### i) Requisitos de instalación

*Procesador:* Intel: Pentium® hasta Pentium® 4, AMD: K6/II/III, Duron(tm), Athlon(tm), Athlon XP/MP(tm)

*Memoria principal:* Para la instalación con YaST2 en modo gráfico se requieren como mínimo 64 MB. Para el uso de las aplicaciones ofimáticas o de Crossover Office se recomiendan 256 MB.

*Disco duro:* Al menos 2,5 GB para una instalación estándar. Es posible reducir la cantidad de memoria necesaria con una selección de software adaptada. Se recomiendan al menos 3 GB para el uso de Crossover Office y de las aplicaciones Microsoft soportadas. Se soportan los discos duros LBA48.

*Tarjetas gráficas:* SuSE Linux Office Desktop incluye controladores para las tarjetas de uso más extendido, por ejemplo: ATI: Radeon 7500/8500, FireGL 8700/8800, FireGL 2/4 Matrox: G450/G550, nVidia: GeForce 2/3/4, nForce.



**Controladoras Raid:** Es posible realizar la instalación en controladoras Raid. Se recomienda informarse en su base de datos de hardware sobre el grado de soporte del componente de hardware deseado.

**Tarjetas ISA Plug & Play** Es posible que las tarjetas ISA Plug & Play tengan que ser configuradas manualmente.

**Tarjetas de sonido:** SuSE Linux Office Desktop soporta las tarjetas de sonido de uso más extendido, incluyendo las siguientes tarjetas de sonido profesionales: Soundblaster: Audigy Terratec: EWX 24/96 (Stereo E/S analógica y digital), EWS 88 MT (8 canales analógicos), EWS 88D (10 canales digitales en formato ADAT)

### 3.8 DISTRIBUCIÓN LINUX-MANDRAKE

Versión 7.2



Linux Mandrake está basado en Linux, el Sistema Operativo multitarea de 32 bits. Actualmente funciona en la mayoría de las arquitecturas Intel y compatibles. (Pentium, Pentium Pro, Pentium MMX, Pentium II, Pentium III y todos sus clones). Linux Mandrake combina el poder de Unix con la flexibilidad del poderoso entorno gráfico KDE. También incluye otros entornos gráficos como Gnome, AfterStep, WindowMaker, IceWM.

Incluye el Servidor de Web Apache, el Programa Gnu de Manipulación de Imágenes Gimp, Netscape Communicator y muchas otras excelentes aplicaciones. Linux Mandrake se puede obtener y usar en forma, también su redistribución es libre.

Linux Mandrake es 99% compatible con RedHat además, la distribución se mantiene actualizada constantemente con las últimas correcciones y los nuevos kernels de Linux.

#### a) Multitarea

Linux Mandrake está basado en Linux, el Sistema Operativo multitarea de 32 bits.

**b) Multiusuario**

Mandrake es completamente multiusuario.

**c) Multiplataforma**

Intel Pentium®, AMD Athlon®, PowerPC; y procesadores de 64 bits como Intel Itanium®, SPARC®.

**d) Multiprocesador**

Soporta multiprocesadores para computadora personales basados en PPC, así como en IA-64 y AMD-64.

**h) Protección de la memoria entre procesos**

En Mandrake la protección y la gestión optimizada de la memoria o multiproceso simétrico esta disponible para un desempeño optimo.

**i) Código fuente disponible**

Los fuentes y los ejecutables están en la siguiente dirección:

<http://www.mandrakeLinux.com/es/ftp.php3>

**j) Conectividad**

Cuenta con todos los servicios que ofrecen las distribuciones Linux opara conectarse a otras redes, tiene ProFTP, servidor de correo Postfix (también disponible Sendmail), servidor OpenSSH, servidor DHCP, servidor Bind DNS y servidor NFS entre todos los servicios.

**k) Características Particulares**

Mandrake Linux 7.2 utiliza el siguiente software:

*INTERNET*

- Mozilla 1.3

*PAQUETES DE OFICINA*

- OpenOffice.org 1.0.2

- KOffice 1.2.1

#### *PROGRAMACIÓN*

- Kernel 2.4.21
- XFree 4.3
- Glibc 2.3.1
- GCC 3.2.2
- Apache 2
- Samba 2.2.7
- MySQL 4. 0.11
- ProFTPD 1.2.7
- Postfix 2.0.6
- OpenSSH 3.5

#### *ESCRITORIOS*

- GNOME 2.2
- IceWM 1.2
- WindowMaker 0.8
- KDE 3.1
- Enlightenment 0.16.5

#### **i) Requisitos de instalación**

Requiere un procesador tipo x86 o superior. Esto incluye Intel Pentium I/II/III/IV, AMD K6/II/III, AMD Duron, AMD Athlon/XP/MP. También soporta máquinas multiprocesador.

*Memoria:* requeridos al menos 64 MB, 128 MB recomendados.

*Discos Duros:* Soporta discos IDE y SCSI. También soporta la mayoría de controladores RAID y SCSI.

*Tamaño de disco duro:* requiere al menos 100 MB, 1 GB recomendado.

*USB y USB 2.0:* soporta la mayoría de dispositivos USB.

*Lectoras y grabadoras CD-ROM y DVD-ROM:* soporta la mayoría de los modelos IDE y SCSI.

*Tarjetas Gráficas:* soporta la mayoría de las tarjetas ISA, PCI y AGP, incluyendo hardware de aceleración 3D.

*Adaptadores de red:* soporta la mayoría de tarjetas de red ISA, PCI, USB y la mayoría de módems, RNIS y dispositivos DSL.

*Impresoras:* soporta la mayoría de impresoras de inyección de tinta y Láser.

*Tarjetas de Sonido:* todas las tarjetas de sonido compatibles con SoundBlaster son soportadas.

*Laptops:* soporta la mayoría de laptops.

Soporta muchas tarjetas de TV, escáneres, webcams, cámaras digitales y dispositivos FireWire.

### **3.9 Pasos para instalar un Sistema Operativo Linux**

Para este proyecto se realizaron varias instalaciones primero se instalo Red Hat 7.0 y se tuvo algunos problemas con Windows debido a que tenían dos Sistemas Operativos en un solo disco duro, por lo que se tuvo que eliminar, posteriormente se trato de Instalar SUSE 8.0 pero el software no lo permitió debido a las carencias del hardware. Después de analizar varios Sistemas Operativos se opto por la instalación de Linux Mandrake 7.2, debido a que se trabajo en una laptop Compaq Presario 1245 que solo tiene 64 megas de ram y otra versión causaría lentitud en el hardware.

#### **A) Arranque desde el CD**

Consiste en colocar el primer cd de Mandrake en la lectora y rebootear la máquina cuya secuencia de booteo debe tener como primera unidad al CD-Rom. La secuencia de booteo se controla desde el Setup del BIOS que aparece al presionar Del o Supr al iniciar.

**B) Lanzar la instalación a partir de Win95/98/Me**

Se inserta el CD-Rom en la lectora y se espera a que aparezca la ventana con opciones. Si no aparece se ejecutará `/dosutils/autorun.exe`. En esta ventana en la versión 7.2 aparece la opción `Linux for windows installation` y `Complete Installation` que permite hacer la instalación de Linux sobre una partición `fat` o `fat 32` y la otra una instalación estándar.

*Lanzar la instalación desde MS-DOS*

En las versiones 7.1 y 7.2 se puede ejecutar `dosutils/autoboot/autoboot.bat` para lanzar el instalador desde `ms-dos`.

*Instalación en red*

Hacer una instalación por red, con equipos PCMCIA (PCCard) o a partir del disco rígido, se necesita crear un diskette de arranque con alguna de las imágenes listadas en el manual de instalación y que se ubican en `/images/`.

*El diskette de arranque de Linux*

Además de servir como lanzador del instalador gráfico de Linux Mandrake, DrakX, tiene varias opciones útiles que ayudarán en caso de que no se logre utilizar el modo de instalación gráfica definido por defecto o se necesite utilizarlo para reparar el sistema. Las opciones que aparecen al presionar `<F1>` son:

*vgalo:* Para instalación gráfica de baja resolución.

*text:* Para instalación de texto en lugar de la gráfica.

*Linux:* Para instalación gráfica estándar a resolución normal.

*expert, vgalo expert o text expert:* Para desactivar la detección automática del hardware.

*rescue:* Nos permite, en conjunto con el primer cd, reparar el sistema que ya está instalado.

**C) Instalando Linux Mandrake 7.2****Paso 1:** Selección del idioma.

Spanish o algún otro idioma según nos convenga.

**Paso 2:** Clase de instalación.

El sistema cuenta con dos opciones según el nivel de conocimiento y experiencia que se tenga en instalar GNU/Linux: Recomendada y Experto. En este caso se selecciono experto para hacer más ilustrativo el ejemplo, finalmente se eligió Instalar.

**Paso 3:** Detección del disco rígido.

DrakX intentará buscar adaptadores SCSI PCI. Si lo encuentra y sabe que controlador usar, lo instalará automáticamente. En caso contrario preguntará: ¿Tiene alguna interfase SCSI?, en este caso se responde "no".

**Paso 4:** Configuración del ratón.

Se elige el puerto en que está conectado y el modelo, en éste caso es "estándar". La cual nos da la posibilidad de probar su funcionalidad, sino se esta conforme se elige otro ratón y listo.

**Paso 5:** Elegir teclado.

Si se tiene un teclado EEUU internacional, se elige EEUU con acentos. Si es español se elige este y se continúa.

**Paso 6:** Seguridad.

Se elige Seguridad Media ya que la PC va a estar conectada directamente al Internet y no va a funcionar como servidor. En Linux Mandrake 7.2 esta opción esta agrupada bajo el ítem "Preguntas Varias".

**Paso 7:** Configuración del Sistema de Archivos.

DrakX presenta la aplicación DiskDrake para el manejo de las particiones. Esta tiene muchas opciones que pueden llegar a confundir a un usuario inexperto, pero en este caso sólo se van a utilizar las correspondientes a la elección del punto de montaje y formateo. En la barra superior se muestra un esquema gráfico con las particiones correspondientes al disco rígido. Inmediatamente por debajo se tienen a la izquierda un menú con comandos y a la derecha detalles o información sobre la partición seleccionada. Se elige cada una y selecciona el comando Punto de Montaje. Ahora se ingresa el directorio que se usara para acceder a datos de esa partición desde el árbol de directorios de la partición raíz. El sistema por defecto aplica el viejo sistema de archivos Ext2 (id 83) a cada una de las particiones Linux Nativo y swap para la partición de intercambio (id 82), pero a partir de la serie 2.4 se tiene la posibilidad de formatear las particiones con el sistema de archivos ReiserFS.

*Características de ReiserFS*

Después de esto se elige el comando formatear, teniendo cuidado de NO formatear las particiones que contienen datos que queremos resguardar, tales como /mnt/windows. Ahora se selecciona (Done) y el sistema mostrará un cuadro de diálogo donde preguntará que particiones queremos formatear. Como ya fueron formateadas, se deselectan todas y se presiona OK.

**Paso 8:** Elegir los paquetes a instalar.

Es hora de elegir los paquetes a instalar según el perfil de nuestro sistema. En nuestro caso se tiene una Workstation. Entonces como se muestran las aplicaciones o paquetes se dividen en grupos y estos a su vez se agrupan según el uso que le de a nuestra máquina (Servidor, Workstation, etc.). Si deseamos elegir los paquetes en forma individual seleccionamos la casilla de la parte inferior y tendremos que navegar entre más de 1200 paquetes aproximadamente.

Se selecciona OK y se espera a que la instalación de paquetes finalice. Ahora los paquetes son instalados y se informa en todo momento sobre el progreso de la instalación de paquetes y el tiempo restante que varía según la cantidad de paquetes instalados y la velocidad de nuestra máquina. Si el instalador necesita los

paquetes que están en el segundo CD, abre la bandeja de la unidad de CD-Rom para que se ingrese.

**Paso 9: Establecer la Contraseña de root**

Root es el administrador del sistema y, entre otras cosas, es el único usuario autorizado para modificar la configuración de todo el sistema. Por eso es importante elegirla correctamente. Para eso demos seguir las siguientes pautas:

- a) Mínimo 8 caracteres.
- b) Los caracteres deben ser una combinación de letras mayúsculas y minúsculas y números.
- c) No debe ser escrita en ningún lado y por lo tanto debe ser fácil de recordar. Nota: Como estamos instalando el sistema no es necesario seguir estas pautas en este momento al pie de la letra, ya que todavía el sistema no funciona y no tenemos documentación importante que proteger<sup>9</sup>.

**Paso 10: Agregar un usuario**

Aunque sea el único usuario del sistema se debe crear una cuenta de usuario regular ya que ingresar como root para el uso cotidiano del sistema es un riesgo de seguridad elevado (el hacer que el sistema quede totalmente inoperante muchas veces está al alcance de un error de tecleo). En conclusión, root debe usarse únicamente para las tareas de administración del sistema que así lo requieran. A continuación se llenan las casillas con los datos propios para crear un nuevo usuario.

Ahora aparece una nueva ventana que nos da la posibilidad de ingresar al sistema gráfico X window como un usuario normal.

---

<sup>9</sup> JOHNSON, MICHAEL K. The Linux Kernel Hacker's Guide. Guía del hacker del kernel Linux, Traducción: German Castro Calderon.



**Paso 11:** Configurar la red

Ahora el proceso nos indica que va a configurar automáticamente el hardware de red e Internet, se elige "aceptar". ¿Cómo quiere conectarse a Internet? En este caso particular se va a realizar una conexión por módem normal. No obstante, se puede realizar la conexión por RDSI, DSL o ADSL, cable, red local. Se elige conexión normal por módem y se selecciona el puerto al que está conectado (ttyS3 o COM4 en ms-dos). Finalmente se llenan los campos del cuadro Opciones de llamada por módem con los datos que dio el proveedor y que también figuran en el acceso de la carpeta acceso telefónico a redes en Windows. DrakX nos avisa que la configuración de la red finalizó y los cambios van a ser aplicados.

**Paso 12:** Sumario

Aquí aparecen agrupadas varios ítems de configuración que en versiones anteriores de Mandrake estaban separados:

- a) Mouse.
- b) Teclado.
- c) Zona Horaria.
- d) Impresora.

Si el sistema detectó una placa de sonido y otra para ver televisión, serán mostradas en la parte inferior, pero no se tendrá la posibilidad de configurarlas. Lamentablemente no detectó la placa de sonido, así que se tendrá que configurar posteriormente. Si se tiene una impresora se enciende y se selecciona el sistema de impresión a utilizar. A partir de esta versión se tiene a CUPS (Common Unix Printing System) como sistema de impresión por defecto. Anteriormente era lpr. Ahora se espera a que el sistema de impresión sea extraído desde el CD y se instale en el disco. La impresora está conectada localmente, esto significa que está conectada directamente a la computadora. Las demás opciones sirven para configurar una impresora remota (en otra máquina). Se da un nombre a la impresora, se deja el nombre por defecto (lp ya que tenemos una sola impresora). En caso contrario hay que leer la ayuda. En caso de que se autodetecte solo hay que seleccionar el

modelo exacto y hacer una prueba de impresión para analizar si la impresión con ese driver es correcta.

En este caso se tiene una impresora HP Deskjet 690c. Si la impresión no sale bien se vuelve a configurar la impresora. Igualmente se puede seguir avanzando con la instalación de Mandrake y configurarla posteriormente. En este caso la prueba salió perfecta, la página de prueba en colores 300x300 dpi de resolución en hoja tamaño A4 y sin recortes.

#### **Paso 13: Servicios al Inicio**

En este paso se eligen los servicios que se desean que se inicien al arrancar. La ayuda nos da consejos básicos de seguridad sobre lo peligroso de tener servicios innecesarios ejecutándose en nuestra máquina (especialmente servidores de red). Se debe seleccionar la opción de xinetd en lugar de inetd, el superservidor inetd a partir de esta versión ya no es seleccionado por omisión y ahora aparece activado el xinetd. La opción de rawdevice asigna dispositivos de bloque (como las particiones de los discos), para el uso de aplicaciones como Oracle.

#### **Paso 14: Crear un Disquete de arranque**

Un diskette de arranque personalizado da la posibilidad de iniciar el equipo bajo GNU/Linux sin depender del cargador de arranque normal. Esto puede ser útil si no desea instalar Lilo (o GRUB) en su sistema, o si otro Sistema Operativo lo borra, o si Lilo no funciona con su configuración de hardware. Un diskette de arranque personalizado también puede usarse con la imagen de rescate de Mandrake, haciendo así mucho más fácil la recuperación en caso de fallo grave del sistema. Se inserta el diskette y se escribe la etiqueta para identificarlo rápidamente. Luego se protege contra escritura y se guarda en un lugar seguro.

#### **Paso 15: Instalar el cargador de inicio (bootloader)**

Como se tiene a Windows en el sistema y Linux Mandrake 7.1 es necesario instalarlo para poder bootear con cualquiera de ellos al inicio de la PC. En la instalación tipo Experto nos da la posibilidad de elegir entre 3 bootloaders distintos: GRUB, Lilo con menú gráfico y Lilo con menú de texto. Elegimos GRUB pero es un

poco más difícil de configurar si se tiene que hacerlo posteriormente. Ambos bootloaders pueden iniciar S.O. pasados los 1024 cilindros. Se elige /dev/hda para que se instale en el MBR (Master Boot Record) del primer disco rígido.

En Opciones Avanzadas se puede activar el modo lba si el BIOS lo soporta y el tamaño de la memoria RAM instalada en caso de que no sea correctamente detectada también lo soporta. Nota: tanto GRUB como Lilo pueden ser instalados en distintas particiones o discos, pero normalmente se instala en el MBR del disco rígido principal.

#### **Paso 16:** Configuración de X

El sistema X Window es un potente entorno gráfico que se permite, junto con el gestor de ventanas que se elija, acceder a programas y archivos con sólo saber su representación gráfica. X Window no es sólo una interface gráfica para ejecutar programas, sino también un sistema cliente/servidor para la interacción entre los programas y los dispositivos de terminales, un protocolo de red, varias herramientas de software que podrán ser usadas para crear utilidades basadas en X Windows y una colección de aplicaciones. En este paso se debe configurar el servidor X para que esté disponible luego de la instalación. DrakXos da a elegir entre 3 servidores según las características de nuestra placa de video<sup>10</sup>:

- a) XFree 4.0.3 con aceleración por hardware 3D.
- b) XFree 3.3.6 con aceleración por hardware 3D.
- c) XFree 3.3.6

Se elige la última opción ya que no se tiene tarjeta para aceleración de gráficos en video. Ahora DrakX intenta detectar nuestra placa de video y las características del monitor. También se puede optar por ejecutar X Windows al momento de ingreso al sistema. Esto hace que cuando se reinicia la máquina aparezca una pantalla de login (registro) en modo gráfico, pero la ayuda recomienda sólo aceptar esta opción en caso de haber probado la configuración de X anteriormente.

---

<sup>10</sup> GREENFIELD, LARRY. Guía de Linux para el Usuario. Traducción: Proyecto LUCAS.

**Paso 17:** Salir de la instalación.

El sistema está listo para reiniciarse y poder usarse. Se saca el disquete booteable y presionamos OK, o presionamos en Opciones Avanzadas para tener opciones adicionales:

- a) Crear el disquete de instalación automática. Esto graba las opciones elegidas durante la instalación. Estas opciones pueden ser usadas en otras máquinas que necesitan ser configuradas de la misma manera.
- b) Guardar la selección de paquetes. Guarda la selección de paquetes para poder restaurarla exactamente en caso de reinstalar el sistema y de esa forma ganar tiempo.

Cuando aparece GRUB o Lilo se selecciona con las flechas el Sistema Operativo a iniciar y se procede a registrar en el sistema el usuario root y password, o también con el nombre de usuario creado y otro password. Se puede entrar por cualquiera de las 6 consolas virtuales, ttys, presionando Alt+F1 a Alt+F6.

Se ha comprobado que cualquiera de los Sistemas Operativos Open Source analizados tiene las características, de madurez y robustez de un Sistema Operativo UNIX por lo que podemos aventurarnos a decir que pueden sustituir a los Sistemas Operativos amparados en una licencia por lo tanto es viable en cualquier empresa que desee utilizarlos en el ámbito informático.

## **CAPÍTULO 4**

### **INTERFASE DE LOS SISTEMAS OPERATIVOS CON EL SOFTWARE LIBRE**

Un problema común de muchas organizaciones es elegir las herramientas indicadas para resolver sus problemas en el ámbito de cómputo pero por desgracia, la mayoría de los usuarios las eligen de acuerdo a la mercadotecnia, la moda o la “educación”. Para exhibir esta problemática planteada se utilizarán las estadísticas de principales Sistemas Operativos considerados en los planes de estudio de las licenciaturas en tecnologías de información y comunicaciones, 1998/1999 y 1999/2000 (figura 4.1). La enseñanza de la utilización de Sistemas Operativos en el caso de Linux es nula en nuestro país éste dato se obtuvo consultado los indicadores sobre las tecnologías de información<sup>11</sup>, con respecto a los manejadores de bases de datos considerados en los planes de estudio de las licenciaturas en tecnologías de información y comunicaciones, 1998/1999 y 1999/2000 (figura 4.2) vemos que la participación del Software Libre no se registra o podría estar en otros por lo que tenemos la certeza de que su participación es mínima y por último se muestran tipos de software de aplicación considerados en los planes de estudio de las licenciaturas en tecnologías de información y comunicaciones, 1998/1999 y 1999/2000 (figura 4.3) la cual indica que las aplicaciones de escritorio son consideradas en los planes de estudio de las Licenciaturas en tecnologías de la información por lo que se puede aconsejar que una solución en Software Libre de Escritorio es viable, pero este punto será tratado posteriormente.

---

<sup>11</sup> Fuente Instituto Nacional de Geografía Estadística e Informática

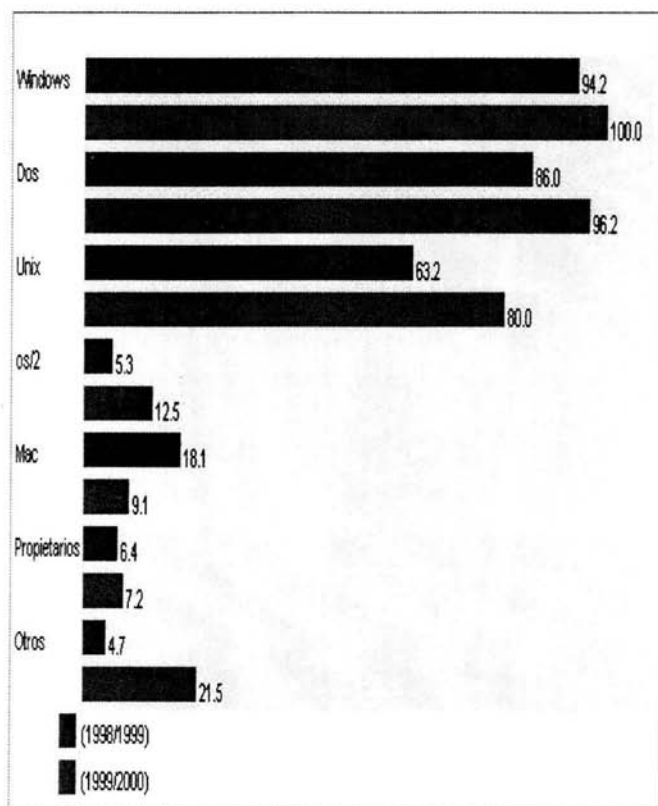


Principales sistemas operativos considerados en los planes de estudio de las licenciaturas en tecnologías de información y comunicaciones, 1998/1999 y 1999/2000

(Por ciento)

Opciones

[Imprimir](#)



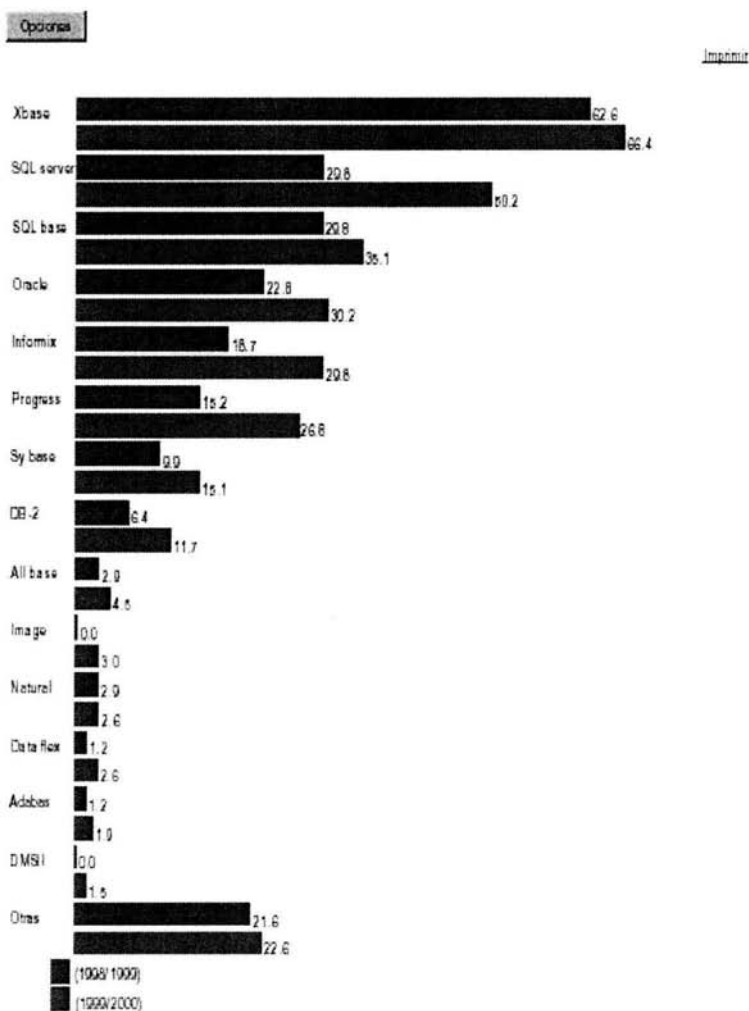
Fuente: INEGI. Encuesta de Formación de Recursos Humanos en Informática, (varios años).

**Figura 4.1**



Principales manejadores de bases de datos considerados en los planes de estudio de las licenciaturas en tecnologías de información y comunicaciones, 1998/1999 y 1999/2000

(Por ciento)



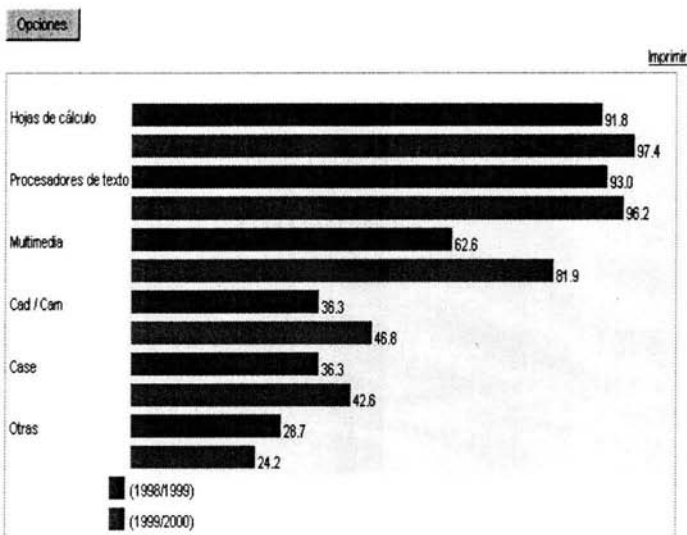
Fuente: INEGI Encuesta de Formación de Recursos Humanos en Informática (varios años).

Figura 4.2



Otros tipos de software de aplicación considerados en los planes de estudio de las licenciaturas en tecnologías de información y comunicaciones, 1998/1999 y 1999/2000

(Por ciento)



Fuente: INEGI. Encuesta de Formación de Recursos Humanos en Informática (varios años).

**Figura 4.3**

Todas estas estadísticas nos muestran que las tecnologías de información en nuestro país están regidas por aplicaciones de cómputo de licencia.

A continuación se explicará cada una de las aplicaciones de cómputo que tienen la capacidad de interactuar con el Software Libre y crear soluciones de cómputo:

1. *Lenguajes de programación.*
2. *Bases de datos.*
3. *Aplicaciones de escritorio.*
4. *Redes.*



## 4.1 Lenguajes de programación

### ¿Qué es un lenguaje de programación?

El hombre desde sus orígenes ha debido comunicarse, ya sea de forma oral, escrita o por lenguaje corporal. Para lograr esto se ha tenido que adaptar, buscar o crear formas de lenguaje que le permitan describir situaciones, solucionar problemas matemáticos, físicos o lógicos. La necesidad de explicación y solución de problemas principalmente matemáticos hizo necesario definir una notación genérica, clara y sencilla que permitiera describir un proceso, sus entradas, sus salidas, su inicio y su fin. También ha sido necesario definir lenguajes que le permitan comunicarse con la computadora, diseñados para realizar procesos internos con base a entradas y salidas de datos a esto le llamamos lenguaje de programación.

De manera simple podemos definir que el lenguaje de programación es un sistema notacional para describir cálculos en una forma legible para la máquina y para el humano.

El proceso de cómputo se define formalmente por medio del concepto matemático de la máquina de Turing, que es un modelo de computadora cuya operación es simple para que pueda ser descrita con precisión.

Para que un lenguaje sea legible para una máquina se requiere:

1. Que tenga una estructura simple, y
2. Un algoritmo de traducción.

Para que un lenguaje de programación sea legible para el humano se requiere que provea abstracciones de las acciones de la computadora que sean fáciles de entender.

La consecuencia de esto es que los lenguajes de programación tienden a parecerse a los lenguajes naturales. Para manejar el desarrollo de programas grandes se requieren ambientes de desarrollo de software, como plataformas de apoyo para el trabajo con el lenguaje de programación.

**Características de los lenguajes de programación.**

Cualquier notación para describir algoritmos y estructuras de datos se puede calificar como un lenguaje de programación, pero principalmente nos interesan los implementados para computadoras. Se han diseñado e implementado cientos de lenguajes de programación distintos. Ya en 1969, se elaboró una lista con 120 lenguajes de programación. Sin embargo, casi ningún programador se aventura a usar más de unos cuantos lenguajes, y muchos restringen por completo su programación a uno o dos. La tecnología de computadoras más antigua data de la época justo antes de la Segunda Guerra Mundial, de finales de los años treinta a principios de los cuarenta.

Hay muchas razones por las cuales un lenguaje de programación llega a tener éxito, algunos son por motivos externos al lenguaje en sí, como el caso de COBOL o Ada que eran promovidos para ser utilizados en ciertas áreas del gobierno de los EEUU, o como el caso de FORTRAN, que era ampliamente soportado por varias compañías que habían invertido en él y en su documentación para obtener implementaciones sofisticadas, sin embargo finalmente es el programador quien hace, ya sea directa o indirectamente, que un lenguaje se mantenga con vida o muera. Existen varias razones por las cuales un programador podría preferir entre un lenguaje u otro, he aquí 11 puntos que se toman en cuenta para que un lenguaje sea considerado de calidad<sup>12</sup>:

**1.- Claridad, simplicidad**

La sintaxis de un lenguaje afecta considerablemente la facilidad con la que un programa puede ser escrito, probado y sobretodo entendido tiempo después de haber sido creado, es por eso que la legibilidad es un punto muy importante, el

---

<sup>12</sup> Para la elaboración de estos criterios se tomó en cuenta la siguiente bibliografía

1. Pags. 544-559, Roger S. Pressman, Ingeniería de software. Un enfoque práctico, ed Mc Graw Hill, 1997.
2. S.J. Young; Lenguajes en Tiempo Real; Ed. Paraninfo 8, Madrid Caps. 1,9
3. G. Booch; Software Engineering with Ada; The Benjamin/Cummings Publishing Company 8 Caps 1-5.
4. S.Bennet; Real Time Computer Control; Ed. Prentice-Hall 9 Cap. 5.
5. Leslie b. Wilson & Robert G. Clark, Comparative Programming Languages. U.S.A, Second Edition. Addison-Wesley 1993.

lenguaje debe manejar conceptos claros y simples para crear algoritmos. Lo mejor es que existe un número pequeño de conceptos y que las reglas de combinación sean tan simples y regulares como se pueda.

La claridad de un programa en cualquier lenguaje de programación se debe lograr casi en la misma forma que para cualquier texto escrito, para ello se requiere:

- a) Una separación lógica del texto en partes comprensibles (capítulos, secciones, etc.) que reflejen la distinción entre los temas que describen, y su presentación en una secuencia lógica que refleje las relaciones entre ellas;
- b) Una selección cuidadosa de las características del lenguaje, usadas en cada parte para expresar su sentido propuesto tan precisamente como sea posible;
- c) Una selección cuidadosa de las palabras usadas para denotar los objetos y conceptos involucrados;
- d) La inclusión de comentarios y preámbulos para clarificar el texto principal cuando sea necesario;
- e) Un aprovechamiento de los dispositivos para presentación de textos, tales como las líneas en blanco y la sangría, para enfatizar la relación entre partes componentes de textos.

## **2.- Ortogonalidad**

El término ortogonalidad se refiere al atributo capaz de combinar varias características de un lenguaje en todas las combinaciones posibles, de manera que todas tengan significado. Por ejemplo, supóngase que un lenguaje dispone de una expresión que puede producir un valor, y de un enunciado condicional que evalúa una expresión para obtener un valor de verdadero o falso. Estas dos características del lenguaje, la expresión y el enunciado condicional, son ortogonales si se puede usar (y evaluar) cualquier expresión dentro del enunciado condicional.

Cuando las características de un lenguaje son ortogonales, es más fácil aprender el lenguaje y escribir los programas porque hay menos excepciones y casos especiales que recordar. El aspecto negativo de la ortogonalidad es que un programa suele compilar sin errores a pesar de contener una combinación de características que son lógicamente incoherentes o cuya ejecución es en extremo ineficiente. A causa de estas cualidades en oposición, la ortogonalidad como atributo de un diseño de lenguaje es todavía motivo de controversia, pues no todos están de acuerdo con ellas.

### **3.- Rango de aplicabilidad**

Este concepto implica que el lenguaje es suficientemente flexible y extensible para usar en casi cualquier tarea, desde tareas minúsculas de una línea hasta programas grandes y elaborados.

### **4.- Ambientes de Programación**

Existen dos tipos de ambientes que maneja el lenguaje texto o GUI.

### **5.- Documentación:**

La documentación se refiere a que existen los suficientes manuales de referencia para disipar las dudas que surjan conforme se vaya utilizando el lenguaje.

### **6.- Editores**

Que tipo de editores y que características tiene cada lenguaje evaluado.

### **7.- Depuradores**

Los depuradores son pausas que sirven para revisar los errores del programa.

### **8.- Portabilidad**

Con el concepto de portabilidad significa a que el mismo código puede llevarse de una arquitectura a otra sin problemas. Hay, por supuesto, algunas funciones que son dependientes de la plataforma como manejo de señales, saltos de línea, ordenamiento de bytes, largo de números y archivos.

### 9.- Ejecución

Son los requisitos mínimos para poder ejecutar un entorno de programación

### 10.- Creación y desarrollo

Es el costo en el desarrollo en el lenguaje.

### 11.- Mantenimiento

El mantenimiento esta fijado por:

*Control de la complejidad:* Pueden medirse examinando los estatutos condicionales en el programa.

*Complejidad de los datos:* Complejidad de las estructuras de datos e interfaces de componentes.

*Longitud de los nombres de los identificadores:* Nombres largos implican menos legibilidad.

*Comentarios de programas:* Quizás más comentarios permiten un mantenimiento más fácil.

### ¿Que es un lenguaje de programación basado en Software Libre?

Un lenguaje de programación basado en Software Libre es aquel que funciona con todos los Sistemas Operativos de libre distribución y este puede ser modificado para ajustarlo a las necesidades del proyecto que se este construyendo.

### Lenguajes de programación basados en Software Libre

Existen muchos lenguajes de programación basados en Software Libre (figura 4.4) la diversidad de estos se debe en gran parte al hecho de que estos son la herramienta más importante para desarrollar el software.

Lenguaje	Sistema Operativo
GNAT (compilador de Ada)	Linux
Gawk	Linux
Brandy Basic V Interpreter	Linux
Gambas	S.U.s.E Linux
Mole Basic	Linux
SmallBasic	Linux
Xbasic	Linux
GCC	Linux
Clip	Linux, FreeBSD y Solaris
SmartEiffel	Linux
Erlang	Linux, FreeBSD
lina	Linux Intel
Unicon	Red Hat Linux
Java	Linux
Jikes	Linux
Persistent Modula-3	Linux, Solaris y Alpha/OSF
Free Pascal	Linux, FreeBSD
Kylix Open Edition	Linux
ActivePerl	Linux , Solaris
Perl	Linux
GNU Prolog	Linux
ActivePython	Linux, Solaris

**Cuadro 4.4 Lenguajes Open Source**

Conforma avanza el tiempo se van liberando lenguajes que eran reservados, además de los investigados por la comunidad que se dedica a producir Software Libre por lo que posiblemente la lista no este completa pero el cuadro 4.4 muestra la importancia de estos proyectos.

#### 4.2 Bases de datos

### ¿Que es una base de datos?

Por definición una base de datos es un conjunto de datos organizados y relacionados entre sí, recolectados y explotados por los sistemas de información de una empresa.

Todos los conceptos referentes a las bases de datos hoy en día están muy claros y definidos formalmente, la tecnología de gestión de bases de datos se halla en una etapa muy madura. Las bases de datos han evolucionado durante los pasados 30 años desde sistemas de archivos rudimentarios hasta sistemas gestores de complejas estructuras de datos que ofrecen un gran número de posibilidades.

### Objetivos de las bases de datos:

Los objetivos de las bases de datos pueden variar de un sistema a otro pero generalmente son<sup>13</sup>:

**Independencia lógica y física de los datos:** Se refiere a la capacidad de modificar una definición de esquema en un nivel de la arquitectura.

**Redundancia mínima:** Se trata de usar la base de datos como repositorio común de datos para distintas aplicaciones.

**Acceso concurrente por parte de múltiples usuarios:** Control de concurrencia mediante técnicas de bloqueo o cerrado de datos accedidos.

**Integridad de los datos:** Se refiere a las medidas de seguridad que impiden que se introduzcan datos erróneos. Esto puede suceder tanto por motivos físicos (defectos de hardware, actualización incompleta debido a causas externas), como de operación (introducción de datos incoherentes).

**Seguridad de acceso y auditoria:** Se refiere al derecho de acceso a los datos contenidos en la base de datos por parte de personas y organismos. El sistema de auditoria mantiene el control de acceso a la base de datos, con el objeto de saber qué o quién realizó una determinada modificación y en qué momento.

---

<sup>13</sup> C. J. DATE, An Introduction to Database Systems, U.S.A., Addison-Wesley, 1994.

**Respaldo y recuperación:** Se refiere a la capacidad de un sistema de base de datos de recuperar su estado en un momento previo a la pérdida de datos.

**Acceso a través de lenguajes de programación estándar:** Se refiere a la posibilidad ya mencionada de acceder a los datos de una base de datos mediante lenguajes de programación ajenos al sistema de base de datos propiamente dicho.

Una base de datos típica conlleva la existencia de tres tipos de usuario con relación a su diseño, desarrollo y uso:

1. El administrador de bases de datos (DBA: Database Administrator): diseña y mantiene la DB.
2. El desarrollador de aplicaciones (programador): implementa las transacciones e interfaces.
3. Los usuarios finales: consultan y editan los datos de la DB mediante un lenguaje de consulta de alto nivel.

No cabe duda de que la función más importante es la realizada por el DBA. A él le corresponde la elección de un determinado modelo de datos y el diseño de la DB. La etapa de diseño es la más importante, ya que es ahí donde se refleja la información contenida en la DB a través del denominado esquema conceptual. En general, se puede decir que el propósito de una base de datos es doble: responder a consultas sobre los datos que contiene, y ejecutar transacciones.

Una consulta (query) se expresa como una expresión lógica sobre los objetos y relaciones definidos en el esquema conceptual; el resultado es la identificación de un subconjunto lógico de la base de datos. Una transacción consiste en un número de consultas y operaciones de modificación o actualización sobre un subesquema.

Las transacciones son atómicas por definición: todos los pasos de una transacción deben ser debidamente ejecutados y confirmados como requisito previo para que la



transacción pueda ser llevada a cabo en su conjunto, en caso contrario debe ser invalidada<sup>14</sup>.

### ¿Que es un sistema gestor de base de datos (DBMS)?

El sistema gestor de bases de datos (DBMS) es el que se encarga de realizar todas las operaciones con los datos (consultas y transacciones), de forma que al DBA no le atañe la manera en que los datos se encuentran almacenados físicamente, pudiéndose concentrar en los aspectos conceptuales en cuanto a diseño, desarrollo y mantenimiento. Un DBMS típico integra los siguientes componentes:

- Un lenguaje de definición de datos (DDL: Data Definition Language).
- Un lenguaje de manipulación de datos (DML: Data Manipulation Language)
- Un lenguaje de consulta (QL: Query Language).

Casi todos los Gestores de Bases de Datos cumplen con las tres anteriores características, otros como Oracle complementan su gestor con sentencias de control de transacciones y el lenguaje de control de datos.

### BASES DE DATOS BASADAS EN SOFTWARE LIBRE

Existen tres Sistemas Manejadores de bases de datos en el mercado:

1. **MySQL**
2. **PostgreSQL**
3. **InterBase**

#### MySQL

MySQL no cumple con todas las normas del SQL Standard debido a que no soporta subconsultas, no soporta procedimientos almacenados (triggers) ni llaves foráneas, todas estas características son prescindibles para agilizar los procesos en la base de datos. Los mecanismos de seguridad en MySQL aun están incompletos dado que no permite definir perfiles de usuarios, locks, reglas y auditorías de transacciones realizadas. Si se desea implementar algún mecanismo de seguridad este debe

---

<sup>14</sup> IAN MARTEENS, *La cara oculta del Delphi 4*, DANYSOFT, España, Madrid, 1998, 912 p.

realizarse en algún lenguaje externo al manejador. El sistema de password y privilegios es flexible y seguro dado que cuenta con un sistema de encriptación, además tiene el soporte de OpenSSL, no soporta vistas, estas características se contemplan hasta la versión 6 (MySQL Reference Manual). Sin embargo estas carencias se pueden remediar utilizando una interfase de lenguaje de programación.

### **InterBase**

InterBase cumple con las normas del SQL 92, proporciona integridad de datos a nivel de tabla, permitiendo asignar a cada usuario, los permisos que le permitan consultar, agregar, modificar y eliminar datos en cada tabla. Su seguridad está basada en una base de datos central llamada isc4.gdb. Esta base de datos almacena los usuarios que tienen acceso al RDBMS. Soporta la creación de vistas actualizables y los procedimientos que están almacenados dentro de la misma base de datos (en este caso, en el archivo .gdb) los cuales pueden ejecutar acciones sobre los datos de la base de datos, y devolvernos algún resultado.

### **PostgreSQL**

No cumple con todas las reglas de SQL 92 en parte debido a que esta es una base de datos orientada a objetos. La autenticación de cliente es una característica central para PostgreSQL sin ella podría sacrificarse la conectividad remota y permitir que cualquiera se conecte a su base de datos y por lo tanto consultar y modificar sus datos. PostgreSQL tiene diferentes tipos de autenticación de cliente. Permite la creación y manejo de vistas, la replicación no esta completamente instalada pero se puede implementar con paquetes externos. PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL, PostgreSQL soporta las claves foráneas y es la única base de datos que soporta la orientación a objetos.

## **4.3 Aplicaciones de Escritorio**

### **¿Qué es una aplicación de escritorio?**

Una aplicación de escritorio es un conjunto de herramientas que sirven para aumentar la productividad de un negocio, oficina u hogar, generalmente vienen

acompañas de un procesador de texto, una hoja de calculo, un presentador de diapositivas, un generador de diagramas de flujo, etc.

### **Características de las aplicaciones de escritorio basadas en Software Libre**

Las características que se analizarán son simples esperando sean de utilidad para los usuarios que normalmente utilizan un procesador de texto para realizar sus escritos, una hoja de calculo para llevar su contabilidad, un presentador de aplicaciones para vender sus productos, una agenda para escribir sus citas etc.

#### **koffice**

Koffice es una suite de oficina que es soportada por Sistemas Operativos como Linux y FreeBSD (con máquinas de 32 y 64 bits), Solaris de Sun, IRIX de SGI, HPUNIX de HP. Koffice es parte del escritorio del KDE y forma parte de muchas aplicaciones que vienen incluidas en el KDE, Koffice esta conformado por:

- KWord: Procesador de texto.
- KPresenter: Presentador de diapositivas.
- KSpread: Hoja de calculo.
- KChart: Presentador de graficas.
- Kontour: Aplicación de dibujo de vectores.
- Kivio: Paquete para realizar diagramas.
- KFormula: Editor de formulas cientificas.
- Kugar: Herramienta generado de reportes de calidad e negocio.
- koscript: Interprete de script de componentes de koffice.
- koshell: Ventana multidocumentos para los componentes del Koffice.
- KOrganizer - Un organizador/agenda/calendario.
- KLyx - Un transporte a KDE del famoso editor de texto Lyx.
- KMail - Un cliente de correo electrónico.
- KRN - Un lector de news.

Los programas de Koffice se basan en el modelo componente de KParts, lo que los hace interactuar entre si debido a que cualquier componente de KOffice se puede encajar en otro documento de KOffice. Por ejemplo, podemos insertar una tabla que se creó en KSpread directamente en un documento del KWord. De esta manera, podemos crear documentos compuestos usando los programas de KOffice<sup>15</sup>.

### OpenOffice

OpenOffice.org 1,1 es una aplicación de oficina basada en Software Libre, la interfaz utilizada y la funcionalidad son muy similares a otros productos en el mercado como Office de Microsoft y Lotus SmartSuite.

- Word Processor. Es un procesador de texto intuitivo.
- Spreadsheet. Cuenta con todas las funcionalidades de una hoja de Excel.
- Presentation Tool. Es un presentador de diapositivas.
- Drawings and Diagrams. Es una herramienta para crear diagramas de flujo.
- Database Access. Es una herramienta que accesa, analiza datos y crea informes.
- One-click Export to PDF. Exporta a formato PDF.
- Export to Flash (.SWF). Exporta a formato de Flash.
- Open XML File Format. Tiene formato nativo en XML.
- ActiveX Control. Soporta controles Activex creados en Visual Basic o Delphi.
- API and File Format Compatible. Es compatible con StarOffice 7 y Office XP.

**GNOME Office:** Es un paquete de varias aplicaciones, como: Gimp, Gnumeric, OpenCalc, Abiword, Impress, GNUMCash. Para el entorno Gnome (aunque funcionan en KDE, teniendo instaladas las librerías GTK). En cuanto a manejo/administración

---

<sup>15</sup> DAVID BANDIEL Y ROBERT NAPIER, Edicion Especial Linux 6ª Edicion, Pearson Educación, Madrid, 2001

de proyectos, esta el Toutdoux, que genera todo en formato XML, al igual que DIA, un software que sigue muy de cerca al MS Visio. Si lo que se busca es un software para administrar finanzas personales, GNUCash puede ser la solución perfecta. Para todo lo que es comunicaciones y Faxes, se puede utilizar GFax, otro software que permite hasta el uso de scanner bajo Linux, utilizando SANE.

El último GNumeric abre y trabaja archivos MS Excel 2000 sin ningún tipo de problemas. Impress es una excelente herramienta muy similar a PowerPoint, que además de trabajar con este tipo de archivos permite utilizar XML (Impress es parte de OpenOffice). Disponible en <http://www.gnome.org/>

También es posible encontrar aplicaciones no integradas en un paquete de oficina como Abiword que es un procesador de textos gratuito que esta acompañado por su código fuente. No sólo permite trabajar con documentos .doc, sino que también lo hace con Latex, HTML, WML, XML, PDB (Palm Document), etc. Disponible en <http://www.abisource.com/>

#### **4.4 Redes**

##### **¿Qué es una red?**

Una red informática es un conjunto de computadoras autónomas interconectadas capaces de intercambiar información. Una computadora autónoma es cualquier dispositivo capaz de procesar información.

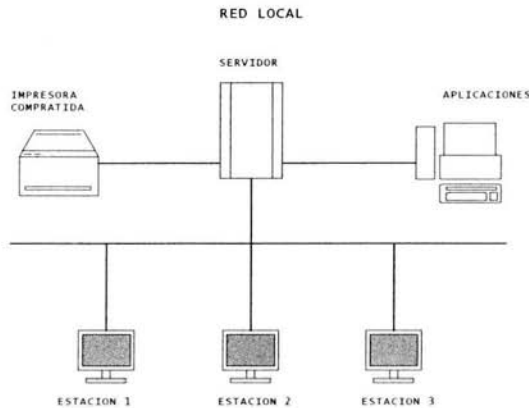
##### **Objetivos de una red**

- Intercambio de información.
- Compartir recursos.
- Mayor fiabilidad.
- Mayor versatilidad (más fácil ampliar el sistema).

El hecho de poder compartir información es muy importante para la cualquier organización, empresa o negocio. La manera de hacerlo es a través de una red. No todas las empresas se basan en una red estandarizada, ya que cada una tiene necesidades diferentes. Por esto, la elección del material (computadora, cableado,

tarjetas de red, software, etc.) irá en función del resultado de las necesidades de la empresa.

La importancia que hoy tienen las redes de empresa es el resultado de la aparición de varios elementos que han contribuido a formar, en primera instancia, la red local, evolucionando más tarde hacia la red de empresa. En una red de empresa podemos distinguir principalmente las técnicas de transmisión de información sobre los medios, los concentradores de cableado que gestionan esos medios, los gestores de redes locales sobre servidores de red, las técnicas de interconexión de red.



**Fig. 4. 5 Esquema de una red**

La comunicación entre el servidor de la red y las terminales de usuario era gestionada por el gestor de red local y se materializaba a través de un medio físico (cableado que podía estructurarse en diferentes formas o topologías de la red local). Para que la conexión al medio fuera posible, se introducía una tarjeta de red que disponía de un conector específico y gestionaba los mecanismos de acceso y conexión de todos los elementos de la red con el medio.

La segunda generación de gestores de redes locales no sólo permitía compartir discos e impresoras, sino también ciertas aplicaciones y datos. Sin embargo, los datos permanecían en el servidor y sólo eran accesibles mediante ciertas aplicaciones. En el siguiente paso se logró que las aplicaciones que gestionan los datos fueran tratadas por el propio servidor, y las estaciones de trabajo accedieran a

estos datos a través de aplicaciones del servidor. Este auge dio lugar a las redes locales paralelas, que en determinados momentos veían la necesidad de interconectarse para compartir los recursos así como datos de la misma organización. Estas redes podían provenir de distintas arquitecturas o diseños, de técnicas más o menos diferentes y/o actuales (como hemos visto en la anterior evolución) por lo que se necesita de mecanismos adecuados que permitan la realización de las comunicaciones. Por otra parte, el número de elementos que se incorporan o se desplazan, incrementaban la complejidad de la estructura así como la posibilidad de rupturas en el cableado<sup>16</sup>.

Los concentradores permiten conectar y asegurar el cableado entre varias redes locales, los puentes (Bridges) permiten que una red se pueda desarrollar a distancia y tamaño al recortar una red en segmentos o en anillos secundarios. Los puentes filtran las comunicaciones de modo que sólo el tráfico destinado a otro segmento pueda cruzar el puente

Una red con puentes funciona como un conjunto que constituye una red única pero con un gran tamaño y múltiples caminos de acceso entre dos puntos cualesquiera de la red, lo cual puede presentar un problema al aumentar las posibilidades de comunicación.

Los Routers aparecieron para racionalizar las interconexiones de redes. Un Router permite crear subredes que correspondan a áreas o grupos específicos de trabajo dentro de una gran estructura. Además, permiten mejorar la accesibilidad de la red al ofrecer la posibilidad de volver a dirigir los datos en función de la carga y la disponibilidad de los enlaces.

---

<sup>16</sup> DANIEL A. MENASCE, DANIEL SCHWABE, Redes de computadores, Aspectos técnicos y operacionales, Madrid, Paraninfo, 1988, 168 p.

### Componentes de una Red

Los elementos que configuran una red son:

**Servidor:** Es la computadora principal de la red que soporta el sistema gestor de la misma, proporcionando todos los servicios a los terminales conectados a la red.

**Estaciones de trabajo:** Componen los nodos terminales de la red, pudiendo ser tan sólo terminales o bien PC's, con sus propios dispositivos físicos.

**Tarjetas de Red:** Es el elemento interfaz que permite la conexión de una computadora a la red, constituyendo el pilar sobre el que se sostiene toda la red

**El medio:** Constituido por el cableado y los conectores que enlazan los componentes de la red.

**Dispositivos adicionales:** Son todos aquellos componentes de la red que son utilizados y compartidos por sus usuarios; son recursos tales como: impresoras, unidades de almacenamiento, etc.

### Cobertura y topología de las redes.

Es posible establecer una clasificación de las redes en función de su tamaño o radio de acción, así como dependiendo de su localización geográfica. Podemos enumerar las siguientes tipos de redes:

**Redes de Área Local (LAN):** Son pequeñas redes con un número reducido de nodos (siempre inferior al centenar), habitualmente localizadas en un mismo edificio o planta.

**Redes Interconectadas (InterNetWork):** Como su propio nombre indica, corresponde a la interconexión de redes del tipo LAN. Es habitual en una empresa, conectar todas sus subredes en todas las ubicaciones de la organización.

**Redes Metropolitanas (MAN):** Una red metropolitana comprende un espacio geográfico mayor, como puede ser un campus universitario o una ciudad.



**Redes de Gran Alcance (WAN):** En este tipo de redes es habitual que su ámbito operativo supere fronteras internacionales, siendo las conexiones habituales las líneas telefónicas de alta velocidad, satélites de comunicaciones o antenas de microondas. Corresponden a grandes organizaciones con centros u oficinas dispersos en varios países.

TAMAÑO DE LA RED	Distancia	Nombre
Sistema	0.1 - 1 m.	Multiprocesador
Sala edificio campus	10 m, 100 m, 1 km	LAN (Local Area Network)
Ciudad	10 km	MAN (Metropolitan)
País, continente	> 10 km	WAN

El proceso de comunicación se lleva a cabo siempre en función del nivel en que nos encontremos.

### Modelo OSI

El conjunto de normas que deben cumplir dos aplicaciones de nivel 7 para comunicarse es el protocolo nivel 7.

- 1) El nivel físico.** Define las especificaciones eléctricas, mecánicas, procedurales y funcionales del enlace físico entre sistemas. Características que preocupan: niveles de tensión, los conectores a emplear, tiempo de transición entre cambios de nivel, tipos de conductores de la información, velocidad de transmisión, distancia máxima del enlace. El nivel físico trabaja con bits.
- 2) El nivel de enlace de datos.** Su misión es proporcionar al nivel superior un canal físico fiable para la transmisión de datos. Intenta detectar y corregir los errores. Otra función de este nivel es el control de flujo lo cual implica que trata de evitar que se manden datos a una velocidad mayor a la que el receptor es capaz de recibir. También agrupa la información en tramas (agrupación de bits) que es la unidad con la que trabaja.
- 3) El nivel de red.** Se encarga de la operación de la subred, su principal misión es el encaminamiento, esto origina el problema de la congestión que también este nivel trata de corregir (existen algoritmos que controlan la congestión. Trabaja con paquetes). El encaminamiento determina el camino que debe seguir la información del host origen al host destino.

- 4) **El nivel de transporte.** Se encarga de separar los niveles superiores de la tecnología de los niveles inferiores. Otra función de este nivel es multiplexar de modo transparente para los niveles superiores varias conexiones de transporte sobre una misma conexión de red. Lo que proporciona es una conexión extremo a extremo libre de errores y una a la recepción de datos en el mismo orden en que se han originado en la emisión. Este nivel controla los errores que se pueden producir a través de una subred a diferencia del nivel de enlace que controla los errores que se producen en un enlace físico (un cable). Este nivel trabaja con mensajes.
- 5) **El nivel de sesión.** Se dedica a establecer, finalizar y gestionar las sesiones de diferentes máquinas. Una sesión es el diálogo entre dos o más entidades.
- 6) **El nivel de presentación.** Se encarga de la interpretación de los datos, que la información que envía un sistema sea legible por el sistema interlocutor de éste. También encripta los datos y los comprime.
- 7) **El nivel de aplicación.** Es el más próximo al usuario e incluye una serie de protocolos para uso común como mensajería electrónica.

### Modelo General de Red bajo Linux

Las distribuciones de Linux disponen de los dos protocolos principales de red para sistemas UNIX: TCP/IP (Transmission Control Protocol/Internet Protocol) y UUCP. (UNIX-to-UNIX Copy) TCP/IP es un conjunto de protocolos de red que permite a sistemas de todo el mundo comunicarse en Internet. Mediante estos protocolos podemos levantar servicios de correo electrónico, noticias (USENET news), transferencias de archivos con FTP y mucho más.

La mayoría de las redes TCP/IP usan Ethernet como tipo de red física de transporte. Linux da soporte a muchas tarjetas de red Ethernet e interfaces para computadoras personales, incluyendo el adaptador Ethernet D-Link de bolsillo para computadoras portátiles.

Linux también proporciona SLIP (Serial Line Internet Protocol), el cual permite conectarse a Internet a través de un módem. Para poder usar SLIP, necesitamos tener acceso a un servidor de SLIP, una máquina conectada a la red que permite acceso de entrada por teléfono. Muchas empresas y universidades tienen servidores

SLIP disponibles. De hecho, si tenemos que Linux dispone de conexión Ethernet y de módem, podemos configurarlo como servidor de SLIP para otros usuarios. NFS (Network File System) permite compartir fácilmente archivos con otras máquinas de la red. FTP (File Transfer Protocol) permite la transferencia de archivos entre máquinas.

Linux proporciona la interfase estándar de programación por "sockets", lo que permite que cualquier programa que use TCP/IP pueda ser llevado a Linux. El servidor Linux de X también soporta TCP/IP, permitiendo ver aplicaciones que están corriendo en otros sistemas sobre nuestra pantalla.

Linux permite UUCP que es un mecanismo usado para transferir archivos, correo electrónico y noticias entre máquinas UNIX.

Linux soporta una implementación completa de los protocolos de red TCP/IP (Transport Control Protocol/Internet Protocol). TCP/IP ha resultado ser hasta ahora el mejor mecanismo de comunicación entre computadora de todo el mundo. Con Linux y una tarjeta Ethernet podremos introducir nuestra máquina en una red local o (si se tienen las conexiones apropiadas) a la Internet, la red TCP/IP de ámbito mundial.

Poner en marcha una pequeña red local de máquinas UNIX es fácil. Sólo requiere una tarjeta Ethernet en cada máquina y los cables adecuados así como hardware accesorio (terminadores, etc.). Si una empresa tiene acceso a Internet, podrá insertar una máquina Linux en esta red.

La implementación actual de TCP/IP y los protocolos relacionados para Linux se llama "NET-2". No tiene que ver con la versión NET-2 para BSD. En realidad, se refiere a que es la segunda implementación que se hace para Linux.

NET-2 de Linux soporta también SLIP (Serial Line Internet Protocol). SLIP permite acceder a la Internet con un módem. Si la empresa proporciona accesos por SLIP, podremos llamar desde casa al servidor SLIP y conectarnos así a la Red. Recíprocamente, si tenemos Linux, una tarjeta de Red y un módem se tiene la capacidad de configurar un servidor SLIP en la red.

Para obtener más información de configuración de TCP/IP en Linux, puede leer el documento NET-2 HOWTO, disponible mediante FTP en <http://www.infor.es/LINUX>. Se trata de una guía completa de configuración, que incluye conexiones mediante Ethernet y SLIP. Otro documento relacionado es el Ethernet HOWTO, que se centra en cómo configurar diversos modelos de tarjetas Ethernet. Encontrará más ayuda en el libro TCP/IP Network Administration, de Craig Hunt.

### **Hardware requerido**

Se puede utilizar el TCP/IP para Linux sin hardware de red. Y el dispositivo "loopback" para conectarse. Aunque parezca poco serio, hay algunos programas que necesitan conexiones de red "loopback" para funcionar.

Hemos visto que Linux posee todas las características para ser un Sistema Operativo Completo sin carecer de aplicaciones para el hogar, la oficina o la empresa, el problema deriva de la complejidad intrínseca que acompaña al Sistema Operativo, las corporaciones que han logrado un éxito comercial en algún software lo han hecho debido a la facilidad de uso de sus aplicaciones, el reto de la comunidad de Linux es hacer accesible el Sistema Operativo a todo el mundo que desee ocuparlo pero eso no está dentro de los alcances del presente trabajo. El fin que perseguimos es mostrar la viabilidad de un Sistema Operativo Open Source dado existe un universo completo de aplicaciones que pueden ser utilizadas para complementar el trabajo de los Sistemas Operativos y vemos que no hay limitante con respecto a lo que un desarrollador, un DBA, administrador de red, empresario, oficinista o cualquier persona desee llevar a cabo en el mundo del Open Source.

## **CAPÍTULO 5**

### **USO Y TENDENCIA DE LOS SISTEMAS OPERATIVOS OPEN SOURCE EN MEXICO.**

En este capítulo se mostrará el campo de aplicación de los Sistemas Operativos Open Source, las perspectivas futuras y las dificultades para utilizar este tipo de sistemas como una alternativa de desarrollo en nuestro país.

#### **5.1 Uso de los Sistemas Operativos Open Source en el Sector publico**

Los gobiernos actuales han evolucionado, del estado paternalista hacia una administración dinámica y preocupada por no aumentar los déficits públicos (tener perdidas), pero además de eso, es necesario resaltar la importancia social del gobierno como rector y responsable de garantizar que el desarrollo integral del país sea sustentable. Entre esas actividades necesarias para garantizar el desarrollo sostenido está la educación, respeto a la pluralidad cultural, la seguridad pública y es ahí donde el uso del Software Libre ayudaría considerablemente, no sólo en el costo de las licencias, sino en la posibilidad de que con el conocimiento generado por el uso de los algoritmos mas óptimos (puestos al dominio público a través de las licencias compatibles con GNU), genere mejores profesionistas, mayores posibilidades de desarrollo para la población del país entre otros enormes beneficios. Quizá lo más importante es que el gobierno NO ES UNA EMPRESA, pero debería manejarse como tal y así evitar perdidas (déficits), pero además de eso, es necesario resaltar la importancia social del gobierno como precursor, reformador y estandarizador de las actividades que norman a la sociedad. Por otro lado, el gobierno no está para desarrollar sistemas, sino que para administrar los bienes, en lugar de contratar desarrollos sobre plataformas cerradas, debe de estar obligado a forzar que sean sobre plataformas abiertas para evitar la dependencia de proveedores.

En el sector público, es muy importante la adopción y promoción de tecnologías. La administración pública es una de las principales candidatas al uso de los Sistemas Operativos Open Source.

## **5.2 Impactos principales de los Sistemas Operativos Open Source en la Administración Pública**

Los principales aspectos que se pueden analizar para mostrar las ventajas del uso los Sistemas Operativos Open Source en el sector público son los siguientes:

**El Aprovechamiento adecuado de los recursos.** Muchos Sistemas Operativos Open Source pueden ser promovidos por la administración pública y utilizados por muchos otros sectores de la sociedad. Por ello, cualquier inversión pública en el desarrollo de un producto libre que le interese, redundara beneficios no solo en la propia administración, sino en toda la gente que podrán usar ese producto para sus tareas informáticas.

**Gastos del Erario Público.** Cuando se usan Sistemas Operativos propietarios, todo lo gastado en licencias va directamente al fabricante del producto, y además esa compra redundo en el fortalecimiento de su posición. Lo cual no es necesariamente perjudicial, pero es poco eficiente para la región vinculada a la administración, si analizamos la alternativa de usar un programa libre. En este caso, las empresas locales podrán competir proporcionando servicios (y el propio programa) a la administración, en condiciones muy similares a cualquier otra empresa. Se puede decir que, de alguna manera, la administración esta allanando el campo de juego, y haciendo mas fácil que cualquiera pueda competir en el, las empresas locales tendrán la posibilidad de aprovechar sus ventajas competitivas (mejor conocimiento de las necesidades del cliente, cercanía geográfica, etc.).

**Independencia de proveedor.** Es obvio que cualquier organización preferirá depender de un mercado en régimen de competencia que de un solo proveedor que puede imponer las condiciones en las que proporciona su producto. Sin embargo, en el mundo de la administración, esta preferencia se convierte en requisito fundamental, y hasta obligación legal en algunos casos. La administración no puede, en general, elegir contratar con un suministrador dado, sino que debe especificar sus

necesidades de forma que cualquier empresa interesada, que cumpla ciertas características técnicas, y que proporcione el servicio o el producto demandado con una cierta calidad, pueda tener opción a un contrato. De nuevo, en el caso del software propietario, para cada producto no hay más que un proveedor (aunque use una variedad de intermediarios). Si se especifica un producto dado, se está decidiendo también que proveedor contratara con la administración. Y en muchos casos es prácticamente imposible evitar especificar un cierto producto, cuando estamos hablando de programas de computadora. Razones de compatibilidad dentro de la organización, o de ahorros en formación y administración, u otros muchos, hacen habitual que una administración decida usar un cierto producto. La única salida a esta situación es que el producto especificado sea libre. En ese caso, cualquier empresa interesada podrá proporcionarlo, y también cualquier tipo de servicio sobre el (sujeto únicamente a sus capacidades y conocimientos del producto). Además, en caso de contratar de esta manera, la administración podrá en el futuro cambiar de proveedor si así lo desea, inmediatamente, y sin ningún problema técnico, pues aunque cambie de empresa, el producto que usara será el mismo.

**Adaptación a las necesidades exactas.** Aunque la adaptación a sus necesidades exactas es algo que necesita cualquier organización que precisa de la informática, las peculiaridades de la administración hacen que este sea un factor muy importante para el éxito de la implantación de un sistema informático. En el caso de usar un Sistemas Operativos Open Source, la adaptación puede hacerse con mucha mayor facilidad, y lo que es más importante, sirviéndose de un mercado con competencia, si hace falta contratarla. Cuando la administración compra un Sistema Operativo propietario, modificarlo pasa normalmente por alcanzar un acuerdo con su productor, que es el único que legalmente (y muchas veces técnicamente) puede hacerlo. En esas condiciones, es difícil realizar buenas negociaciones, sobre todo si el productor no está interesado en el mercado que le ofrece la administración en cuestión. Sin embargo, usando un Sistemas Operativos Open Source, la administración puede modificarlo, si dispone de personal para ello, o contratar externamente la modificación. Esta contratación la puede realizar en principio cualquier empresa que tenga los conocimientos y capacidades para ello, por lo que es de esperar la

conurrencia de varias empresas. Eso, necesariamente, tiende a abaratar los costes y a mejorar la calidad.

**Escrutinio publico de seguridad.** Para una Administración Pública poder garantizar que sus sistemas informáticos hacen solo lo que esta previsto que hagan es un requisito fundamental. No son pocas las veces que esos sistemas manejan datos privados, en los que pueden estar interesados muchos terceros (pensemos en datos fiscales, penales, censales, electorales, etc.). Dificilmente si se usa una aplicación propietaria, sin código fuente disponible, puede asegurarse que efectivamente esa aplicación trata esos datos como debe. Pero incluso si se ofrece su código fuente, las posibilidades que tendrá una institución pública para asegurar que no contiene código extraño serán muy limitadas. Solo si se puede encargar ese trabajo de forma habitual y rutinaria a terceros, y además cualquier parte interesada puede escrutarlos, podrá estar la administración razonablemente segura de cumplir con ese deber fundamental, o al menos de tomar todas las medidas en su mano para hacerlo.

**Disponibilidad a largo plazo.** Muchos datos que manejan las administraciones, y los programas que sirven para calcularlos, han de estar disponible dentro de decenas de años. Por ejemplo en este año Windows 98 ha dejado de ser respaldado por Microsoft por que su licencia de cinco años ya caduco, es muy difícil asegurar que un programa propietario cualquiera pueda estar a disponible cuando hayan pasado esos periodos de tiempo, y mas si lo que se quiere es que funcione en la plataforma habitual en ese momento futuro. Por el contrario, es muy posible que su productor haya perdido interés en el producto, y no lo haya portado a nuevas plataformas, o que solo este dispuesto a hacerlo ante grandes contraprestaciones económicas. En el caso de los Sistemas Operativos Open Source, por el contrario, la aplicación esta disponible, con seguridad, para que cualquiera la porte y la deje funcionando según las necesidades de la administración. Si eso no sucede de forma espontánea, la administración siempre puede dirigirse a varias empresas buscando la mejor oferta para hacer el trabajo. De esta forma puede garantizarse que la aplicación, y los datos que maneja, estarán disponibles cuando haga falta.



### 5.3 Dificultades de adopción de los Sistemas Operativos Open Source

Pero aunque las ventajas de uso de los Sistemas Operativos Open Source en las administraciones sean muchas, también son muchas las dificultades cuando se enfrentan a su puesta en práctica. Entre ellas pueden destacarse:

**Desconocimiento y falta de decisión política.** El primer problema que se encuentran los Sistemas Operativos Open Source para su introducción en las administraciones es uno que, sin duda, comparte con otras organizaciones: es aun muy desconocido entre quienes toman las decisiones. Afortunadamente, este es un problema que tienden a minimizarse con el paso del tiempo, pero aun en muchos ámbitos de las administraciones el software libre es percibido como algo extraño, y tomar decisiones en la línea de usarlo implica asumir ciertos riesgos. Unido a esto suele encontrarse un problema de decisión política. La principal ventaja del software libre en la administración no es el costo (siendo esta, de todas formas, importante, sobre todo cuando se habla de despliegues para gran cantidad de puestos). Las principales ventajas, como ya hemos visto, son sobre todo de fondo, estratégicas. Y por tanto quedan en gran medida en el ámbito de las decisiones políticas, no técnicas. Sin voluntad política de cambiar los sistemas informáticos y la filosofía con que se contratan, es difícil avanzar en la implantación de software libre en la administración.

**Falta de estrategia de implantación** En muchos casos, los Sistemas Operativos Open Source comienzan a usarse en una administración simplemente porque el costo de adquisición es mas bajo. Es muy habitual en estos casos que el producto en cuestión se incorpore al sistema informático sin mayor planificación, y en general sin una estrategia global de uso y aprovechamiento de software libre. Esto causa que la mayor parte de las ventajas del software libre se pierdan en el camino, ya que todo queda en el uso de un producto mas barato, cuando ya se ha visto que en general las mayores ventajas son de otro tipo. Si a esto unimos que el uso de software libre, si no se diseña el cambio adecuadamente, puede generar costos de transición considerables, hace que experiencias aisladas, y fuera de un marco claro, de uso de software libre en la administración puedan resultar fallidas y frustrantes.

#### **5.4 Papel del Sector Público en el mundo de los Sistemas Operativos Open Source**

Las administraciones públicas actúan sobre el mundo de los Sistemas Operativos Open Source al menos de tres formas:

1. Comprando programas y servicios relacionados con ellos. Las administraciones, como grandes usuarios de informática, son un actor fundamental en el mercado de los Sistemas Operativos Open Source.
2. Promoviendo de diversas formas el uso (y la adquisición) de ciertos programas en la sociedad. Esta promoción se hace a veces ofreciendo incentivos económicos (desgravaciones fiscales, incentivos directos, etc.), a veces con información y recomendaciones, a veces por "efecto ejemplo"
3. Financiando (directa o indirectamente) proyectos de investigación y desarrollo que están diseñando el futuro de la informática.

En cada uno de estos ámbitos los Sistemas Operativos Open Source pueden presentar ventajas específicas (además de las ya descritas en el apartado anterior), interesantes tanto para la administración como para la sociedad en general.

#### **¿Como satisfacer mejor las necesidades de las administraciones publicas?**

Las administraciones públicas son grandes consumidores de informática. En lo que al software se refiere, compran habitualmente tanto productos de consumo masivo como sistemas a medida. Desde este punto de vista, son fundamentalmente grandes centros de compras, similares a las grandes empresas, aunque con sus propias peculiaridades. Por ejemplo, en muchos ámbitos se supone que las decisiones de compra de las administraciones públicas no han de tener en cuenta simplemente parámetros de costo frente a funcionalidad, sino que otros, como impacto de la compra en el tejido industrial y social, o consideraciones estratégicas a largo plazo, pueden también ser importantes.

En cualquier caso, lo habitual hoy día en lo que se refiere a software de consumo masivo es utilizar los productos propietarios líderes del mercado. La cantidad de recursos públicos que se gastan los ayuntamientos, municipios, delegaciones, el Gobierno del Distrito Federal etc. en comprar licencias de Windows, Office u otros productos similares es ciertamente considerable. Pero cada vez penetran más las soluciones libres en este mercado. Cada vez se consideran más soluciones basadas en software libre para los servidores, y productos como OpenOffice, y GNU/Linux con GNOME o KDE en el escritorio, un ejemplo claro de esto es la pagina Web de la Secretaria de Gobernación.

### **El desarrollo local**

En el caso del software propietario el usuario esta habilitado par ejecutar un programa, pero no para inspeccionarlo ni modificarlo, entonces no pude aprender de el, se vuelve dependiente de una tecnologia que no solo no comprende sino que le esta expresamente vedada. Los profesionales de su entorno, que podrían ayudarlo a alcanzar sus metas, están igualmente limitados: funcionamiento secreto del programa, inspección prohibida, no se puede modificar. De esa manera los profesionales locales ven sus posibilidades de ofrecer valor agregado cada vez mas limitadas, y sus horizontes laborales se estrechan junto con sus oportunidades de aprender más. Con el software libre se neutraliza enormemente estas desventajas del software propietario.

### **Mayores fuentes de trabajo**

Con el software libre se libera mano de obra existente en el pais que estaba enfrascada como consecuencia de la dependencia tecnologica del software propietario. Ahora se asignarian recursos de los usuarios (en esta caso las instituciones del Estado) para mantenimiento y soporte informático del software libre.

### **Iniciativas legislativas**

En nuestro país surgió la Ley de Adquisiciones y Obra Pública, pero esta ley se modificó en noviembre de 1999 la parte en la que un funcionario se podía meter en un problema si compraba un software cerrado si existía la alternativa abierta. Es decir, no se podía comprar Oracle, porque estaban MySQL, PostgreSQL y demás.

Tampoco se podían pedir desarrollos en Visual Basic o cualquier otro lenguaje atado a una arquitectura en particular, esta parte fue separada de la Ley de Adquisiciones y la Ley de Obra Pública. Pero el espíritu de la ley sobrevive y la intención es precisamente que la independencia y autonomía del país no pueda depender de terceros. Ese es el fundamento y es de hecho, en lo que se basa la filosofía de la Secretaría de la Defensa Nacional.

Hay factores que limitan al desarrollo de los Sistemas Operativos Open Source, como el temor a que las aplicaciones no funcionen correctamente, que no exista soporte técnico adecuado; la incompatibilidad de archivos y la inversión de trabajo requerida. Todos estos elementos se ven reflejados en las estadísticas del uso de Linux en nuestro país en el cual podemos ver la Proporción de las dependencias y entidades de la administración pública por uso de Sistemas Operativos en computadoras personales (figura 5.1)

Proporción de las dependencias y entidades de la administración pública por uso de sistemas operativos en computadoras personales			
(Por ciento)			
Sistema operativo	1998	1999	2000
Windows 95	75.0	92.0	78.0
Dos	64.0	58.0	49.0
Windows NT	24.0	34.0	34.0
Windows 98	16.0	44.0	71.0
Novell	11.0	6.0	9.0
Unix	7.0	6.0	4.0
Mac	4.0	ND	ND
Linux	ND	3.0	6.0
Windows 2000	ND	ND	22.0
Otros	4.0	8.0	11.0
NOTA: 1998 Información de 14 dependencias, 84 entidades y 7 Estados			
1999 Información de 17 dependencias, 94 entidades y 15 Estados			
2000 Información de 15 dependencias, 59 entidades y 8 Estados			
ND No disponible.			
FUENTE: INEGI, Encuesta Informática de la Administración Pública (varios años).			

Fecha de actualización: Martes, 24 de Junio de 2003

**Figura 5.1**

En la figura 5.2 se muestra la proporción de las dependencias y entidades de la administración pública por uso de Sistemas Operativos en estaciones de trabajo.

Proporción de las dependencias y entidades de la administración pública por uso de sistemas operativos en estaciones de trabajo

(Por ciento)				
Sistema operativo	1998	1999	2000	
Unix	33.0	30.0	24.0	
Solaris	31.0	33.0	35.0	
Windows NT	24.0	36.0	41.0	
Windows 95	12.0	9.0	12.0	
HP-UX	12.0	14.0	15.0	
Novell	10.0	9.0	9.0	
VMS	7.0	4.0	ND	
AIX	7.0	6.0	4.0	
IRIX	5.0	10.0	9.0	
Ultron	5.0	ND	ND	
MPEIX	3.0	ND	ND	
Windows 98	ND	6.0	11.0	
Linux	ND	4.0	9.0	
Windows 2000	ND	ND	9.0	
Otros	9.0	9.0	6.0	
NOTA: 1998 Información de 14 dependencias, 84 entidades y 7 Estados				
1999 Información de 17 dependencias, 94 entidades y 15 Estados				
2000 Información de 15 dependencias, 59 entidades y 8 Estados				
ND No disponible.				
FUENTE: INEGI. Encuesta Informática de la Administración Pública (varios años).				

Figura 5.2

Esta aumenta un poco debido a que Linux empieza a ser reconocido como una estación de trabajo viable, con respecto a la proporción de las dependencias y entidades de la administración pública por uso de Sistemas Operativos en minicomputadoras (figura 5.3) está siendo mínima.

Proporción de las dependencias y entidades de la administración pública por uso de sistemas operativos en minicomputadoras

(Por ciento)				
Sistema operativo	1998	1999	2000	
Unix	25.0	47.0	49.0	
Windows NT	15.0	32.0	34.0	
HP-UX	9.0	20.0	22.0	
Novell	9.0	7.0	9.0	
VMS	6.0	7.0	7.0	
Solaris	5.0	13.0	16.0	
AIX	5.0	9.0	8.0	
MPE	5.0	13.0	8.0	
OS400	4.0	14.0	11.0	
VS732	2.0	ND	ND	
Windows 95	2.0	5.0	ND	
Windows 98	ND	3.0	ND	
Linux	ND	ND	7.0	
Windows 2000	ND	ND	4.0	
Otros	7.0	5.0	7.0	
NOTA: 1998 Información de 14 dependencias, 84 entidades y 7 Estados				
1999 Información de 17 dependencias, 94 entidades y 15 Estados				
2000 Información de 15 dependencias, 59 entidades y 8 Estados				
ND No disponible.				
FUENTE: INEGI. Encuesta Informática de la Administración Pública (varios años).				

Fecha de actualización: Martes, 24 de Junio de 2003

Figura 5.3

La tendencia mundial nos muestra que México está en desventaja desde el punto de vista de la economía global debido a que tiene que comprar a un precio muy elevado la tecnología, México no ha participado como líder en el desarrollo tecnológico y tiene que pagar el costo de esta dependencia a precios muy altos pero puede revertir esta tendencia tomando en cuenta las siguientes ventajas que ofrecen los Sistemas Operativos Open Source:

- **Independencia tecnológica:** Mediante el uso de software libre, el estado deja de tener sus sistemas controlados por una entidad externa (con frecuencia empresas extranjeras). De esta forma rompe la dependencia tecnológica que lo tiene actualmente atado y obtiene las libertades que el software libre otorga.
- **Control de la información:** Esto sale directamente como consecuencia directa de las libertades del software libre. Al tener la libertad de inspeccionar el mecanismo de funcionamiento del software y la manera en que almacena los datos, y la posibilidad de modificar (o contratar a alguien que modifique) estos aspectos, queda en manos del estado la llave del acceso a la información (en vez de quedar en manos privadas)
- **Confiabilidad y estabilidad:** El software libre, al ser público, esta sometido a la inspección de una multitud de personas, que pueden buscar problemas, solucionarlos, y compartir la solución con los demás. Debido a esto, y a lo que se llama "el principio de Linus" (dada la suficiente cantidad de ojos, cualquier error del software es evidente), los programas libres gozan de un excelente nivel de confiabilidad y estabilidad, requerido para las aplicaciones críticas del estado.
- **Seguridad:** Este es uno de los puntos clave para el Estado. Mucha de la información que el Estado maneja puede ser peligrosa en manos incorrectas. Es por esto que es crítico que el Estado pueda fiscalizar que su software no tenga puertas de entrada traseras, voluntarias o accidentales, y que pueda cerrarlas en caso de encontrarlas; tal inspección sólo es posible con el software libre

Hay que recalcar que en México aún no se cuenta con leyes y políticas oficiales con respecto a la posición y uso de Software Libre, pero podemos mencionar que el Diputado Jacobo Bonilla de la Asamblea Legislativa está realizando algunas acciones para promover una legislación para el uso del Software Libre.

Existe un interés creciente por el uso de Software Libre en la Presidencia, el cual es motivado por ahorros económicos, independencia en el desarrollo de sistemas, libertad para compartir sin restricciones los desarrollos entre diferentes entidades de la sociedad, la adquisición de capacidades técnicas para el desarrollo de sistemas y las inquietudes del atraso en el desarrollo de México con respecto a Estados Unidos en las aplicaciones de Software Libre. No deben excluirse otros Sistemas Operativos, sino analizar en qué casos conviene la aplicación de cada uno. Hay que tomar en cuenta los casos de otros países como Perú y Colombia, así como de Alemania y China que han emitido normas legales y la India que desarrolla sistemas informáticos para la exportación basados en software propietario o libre.

### **5.5 Instituciones Públicas que utilizan Sistemas Operativos Open Source**

El Gobierno del Distrito Federal, el INEGI, la Presidencia de la República, e-México y del subsistema e-Gobierno utiliza Sistemas Operativos Open Source, la Cámara de Senadores utiliza SuSE con una base de datos Informix, el PAN utiliza servidores Linux. Hay que utilizar en México las herramientas de Software Libre y propietario con que se cuenta para crear sistemas que proporcionen un alto valor agregado, no tratar de volver a hacer herramientas con las que ya se cuenta. Hay que fomentar el desarrollo de software en el país, ya que este es necesario para no depender de empresas transnacionales, lo cual no debe señalarse solamente como una cuestión de nacionalismo, sino por cuestiones de eficiencia de los procesos.

### **5.6 Uso de los Sistemas Operativos Open Source en el Sector Privado**

En México las empresas son todavía precavidas en el uso de Linux, por lo que buscan mantener su inversión en MS-Windows, pero el interés por Linux ha crecido así como el soporte en el mercado, que aún se requiere de un análisis muy cuidadoso en cada ambiente aplicativo para emplearlo más allá de los nichos que

ocupa actualmente. Linux se ha convertido en una opción aceptable para el escritorio.

Instrumentar el uso de Software Libre a través de Linux, se presenta como un valor agregado en los negocios. Hay que señalar que bajo el esquema planteado los clientes pueden utilizar Linux para consolidar sus equipos de trabajo, soluciones de aplicación y de infraestructura, hay que mencionar que IBM estableció en febrero de este año en Guadalajara, Jalisco, un Programa Nacional de Promoción para el Desarrollo de Software Libre en México.

Es importante tener presente que existen riesgos de que los desarrolladores introduzcan funcionalidades indeseables código malicioso en los sistemas de código abierto. La mayoría de las aplicaciones de Linux no han sido auditadas oficialmente.

### **5.7 Instituciones Privadas que utilizan Sistemas Operativos Open Source**

#### **A nivel nacional**

En el caso de las empresas privadas podemos mencionar a Grupo Hosting de México que utiliza servidores Linux para alojar páginas Web, México-Spin utiliza Red Hat Linux 9.0 y Red Hat 7.3, e-smartservice.net utiliza Red Hat 9.0, Bufete Consultor de México, Piensa Technology, Matías Software Group, Internet de alta Calidad, Alternativa Digital, Open Source de México, S.C, Fernández Consultores, ABL Consultores, 8.1.8 Computación Gráfica de México, LG Internet Solutions, PC Soporte y Mantenimiento, HPTC System Solutions S. A. de C. V., el panorama de las empresas que utilizan Software Libre va creciendo conforme se hace publicidad de Linux podemos mencionar a grupo Aullox los cuales fueron muy amables en mandar información de lo que hacen y los clientes que atienden utilizando tecnología Open Source

- Secretaría de Finanzas del Estado de Jalisco.
- Instituto Estatal Electoral del Estado de Oaxaca.
- Secretaría de Vialidad y Transporte del Estado de Jalisco
- Universidad Autónoma de Guadalajara (UAG).
- Tec de Monterrey Campus Guadalajara.



- Universidad Panamericana (UP).
- Universidad Cuauhtemoc.
- Centro de Estudios Tecnológicos Industriales (CETI).
- IBM Guadalajara, S.A. de C.V.
- Farmacia Guadalajara, S.A. de C.V.
- COPAMEX Higiene Infantil, S.A. De C.V.
- SIEMENS VDO, S.A. de C.V.
- IKON Copiroyal, S.A. de C.V.
- Grupo Omnilife de México, S.A. de C.V.
- Grupo MORSA, S.A. de C.V
- Digital PCS, S.A. de C.V. (Telcel)

#### **A nivel Multinacional**

En este nivel encontramos VA Linux, LinuxCare, RedHat, TurboLinux

#### **En otros países**

Alcove (Unión Europea, basado en Francia), SATEC (España), Andago (España).

El Software Libre es una revolución en plena marcha, cuyo alcance está mucho más allá de los teclados y los monitores. Es una revolución cuyo impacto alcanza a las finanzas de las naciones y las ansias de libertad de los pueblos. Y México (Y el mundo) hoy cuenta con los revolucionarios que generosamente han donado su tiempo y conocimientos a la humanidad. Faltan revolucionarios que quieran tomar el trabajo que ya está hecho, y comprueben lo ya probado: que el software producido como libre tiene tanta calidad como el software propietario, y con mas beneficios que el mas evidente: el costo del propietario. Y para los que dicen "a nadie han despedido por comprar a..." refiriéndose luego a marcas propietarias, yo respondería: "Muchos han triunfado por usar Software Libre". Es cuestión de conformarse, o tomar el riesgo, hacer propio el reto y ganar... Hoy tenemos en las manos el poder de tomar eso y llevarlo al mundo como bandera para tener un mundo mejor. ¿Nos atreveremos a hacerlo?.

## CONCLUSIONES

El presente trabajo ha sido un viaje que nos ha ilustrado acerca del uso y aplicaciones de los Sistemas Operativos Open Source, la razón de ser del proyecto ha dado como resultado el poder afirmar que un Linux es una alternativa razonable para poder utilizar un Sistema Completo de Computo prescindiendo del Software Propietario; esto como todas las cosas en el universo tiene sus ventajas y desventajas, pero la balanza se inclina mas hacia las bondades, una de las bondades de este sistema es la capacidad de modificarlo de acuerdo a nuestras necesidades, la desventaja es no tener los conocimientos para hacerlo, una ventaja contundente es el costo de las licencias, la desventaja seria que debemos contar con los recursos humanos capacitados para ello, los cuales no son tan accesibles para los negocios de pocos ingresos.

A un nivel empresarial los programas de origen abierto ofrecen otras ventajas, además de las descritas. Por ejemplo, evitan depender de un productor de software determinado y de la planificación de sus productos. Así, las empresas pueden intercambiar los proveedores de servicios para Apache o Linux, ya que ninguno de ellos tiene el control exclusivo sobre este software. Actualmente los distribuidores de Software Libre se encuentran bajo la presión de la innovación, para realizar extensiones o modificaciones reales a sus productos lo más rápido posible.

Una problemática para las grandes empresas es que tienen que dar su código fuente cuando distribuyen sus productos y esto es riesgoso por que su competencia lo puede utilizar.

Debemos tomar en cuenta que cualquier comunidad con iniciativa puede empezar su propia distribución y el límite para realizar cualquier proyecto es su imaginación.

Existen toda clase de aplicaciones para Linux, la dificultad estriba en hacer las instalación, configuración y mantenimiento tan accesibles que cualquier tenga la capacidad de maniobrarla.

Con este trabajo podemos afirmar que Los Sistemas Operativos Open Source son una alternativa de uso, por ser una solución integral en el ámbito de los Sistemas de Computo.

## BIBLIOGRAFÍA

1. ANDREW S. TANENBAUM, Sistemas Operativos Modernos, México, Prentice Hall, 1993.
2. A.SILBERTSCHATZ, J. PETERSON, P. GALVIN, Sistemas Operativos. conceptos fundamentales, México, 3ª Edición, Addison Wesley ,727p.
3. A.S.TANENBAUM, A.S.WOODHULL, Sistemas Operativos. Diseño e Implementación. México., Prentice-Hall, 1997
4. C. J. DATE, An Introduction to Database Systems, U.S.A, Addison-Wesley, 1994.
5. CHRIS DIBONA, SAM OCKMAN, MARK STONE, Open Sources: Voices from the Open Source Revolution, U.S.A, Orreally, 1999, 280 p.
6. DANIEL A. MENASCE, DANIEL SCHWABE, Redes de computadores. Aspectos técnicos y operacionales, Madrid, Paraninfo, 1988, 168 p.
7. DAVID BANDIEL Y ROBERT NAPIER, Edicion Especial Linux 6ª Edicion, Pearson Educación, Madrid, 2001
8. DEITEL, H. M., Sistemas Operativos, México, Addison\_Wesley Iberoamericana, 1993.

11. FINKEL, R. A., Fundamentos de Sistemas Operativos. Un Estudio Actual de los Sistemas Operativos, México, Anaya Multimedia, 1990.
12. G. BOOCH; Software Engineering with Ada; U.S.A, The Benjamin/Cummings Publishing Company, Ed. Prentice-Hall, 1994.
13. GREENFIELD, LARRY. Guía de Linux para el Usuario, Traducción: Proyecto LUCAS.
14. IAN MARTEENS, La cara oculta del Delphi 4, DANYSOFT, España, Madrid, 1998, 912 p.
15. JACK TACKETT, DAVID GUNTER Y LANCE BROWN, Linux, México, edición especial, Prentice Hall Hispanoamericana, 1996, 869 p.
16. JOHNSON, MICHAEL K. The Linux Kernel Hacker's Guide. Guía del hacker del kernel Linux, Traducción: German Castro Calderon.
17. LESLIE B. WILSON & ROBERT G. CLARK, Comparative Programming Languages. U.S.A, Second Edition. Addison-Wesley 1993.
18. M. CARLING, STEPHEN DEGLER, JAMES DENNIS, Guía Avanzada Administración de Sistemas Linux, Madrid, Prentice-Hall, 1999, 327 p.
19. MEXICO Ley de la Propiedad Industrial, Editorial Porrúa, México, 1997.
20. MÉXICO, Ley Federal de Derechos de Autor, México, McGraw-Hill Interamericana, 1998, 113 p.
21. ROGER S. PRESSMAN, Ingeniería de software. Un enfoque práctico, México, 3a. edición, Ed. Mc Graw Hill, 1997.
22. S.J. YOUNG; Lenguajes en Tiempo Real; Ed. Paraninfo 8, Madrid, 1987.
23. S.Bennet; Real Time Computer Control; Ed. Prentice-Hall,USA, 9 Cap. 5.

## REFERENCIA DE SITIOS EN INTERNET

[1]<http://www.InterBase.com.mx/articulos/procedimientos.php>

[2]<http://aquiles.uca.es/mateLinux/Linux.html>

[3]<http://aquiles.uca.es/mateLinux/software.html>

[4]<http://bulmalug.net/body.phtml?nIdNoticia=622>

[5][http://es.geocities.com/fisicas/enlaces\\_web/Linux.htm](http://es.geocities.com/fisicas/enlaces_web/Linux.htm)

[6]<http://es.tooLinux.com/lininfo/referencias/libre/index.htm>

[7]<http://ftp.gul.uc3m.es/gul/raxi/curso/node1.html>

[8]<http://ftp.gul.uc3m.es/gul/raxi/curso/node141.html>

[9]<http://geneura.ugr.es/~jcarpio/cursos/comercio/ppframe.htm>

[10]<http://jjps.kipelhouse.com/Linux/apache/ssl/index.phtml>

[11]<http://lucas.hispaLinux.es/htmls/manuales.html>

[12]<http://mundoLinux.cjb.net/>

[13]<http://slackware.i8.com/slakes/generinf/generinf.html>

[14]<http://technet.oracle.com/tech/Linux/content.html>

- [15]<http://www.beosfera.com.ar/interface.html>
- [16]<http://www.borland.com/>
- [17][http://www.borland.com/kylix/openedition/kylixopenedition\\_keyanddown\\_steps.html](http://www.borland.com/kylix/openedition/kylixopenedition_keyanddown_steps.html)
- [18]<http://www.caldera.com/support/docs/openLinux/1.3/espanol/contents.html>
- [19]<http://www.debian.org/international/Spanish.es.html>
- [20]<http://www.ciberdroide.com/misc/donde/desarrollo.html>
- [21]<http://www.elprisma.com/apuntes/apuntes.asp?page=11&categoria=602>
- [22]<http://www.en-linea.net/num-actual/software/redhat7/redhat7.htm>
- [23][http://www.es.corel.com/PRESSROOM/august\\_10\\_1999.html](http://www.es.corel.com/PRESSROOM/august_10_1999.html)
- [24]<http://www.europe.redhat.com/documentation/rhl6.2/gsg-es/>
- [25]<http://www.geocities.com/tallerLinux/enlaces.htm>
- [26]<http://www.hispafuentes.com/productos/hf80.php><http://www.Linux-mandrake.com/es/heliosfeat.php3>
- [27]<http://www.hotfeet.ch/~gemi/LDT/>
- [28][http://www.iac.com.mx/computo\\_98/licencia.html](http://www.iac.com.mx/computo_98/licencia.html)
- [29]<http://www.InterBase.com.mx/index.html>
- [30]<http://www.lindows.com/>

- [31]<http://www.Linux-es.com/>
- [32]<http://www.Linux-es.com/distribuciones.html>
- [33]<http://www.Linuxiso.org/distro-battle-2.html>
- [34]<http://www.oracle.com/es/>
- [35]<http://www.planetaLinux.com.ar/article.php?aid=54>
- [36]<http://www.sun.com/>
- [37]<http://www.suse.de/es/>
- [38][http://www.suse.de/es/products/suse\\_Linux/i386/support.html](http://www.suse.de/es/products/suse_Linux/i386/support.html)
- [39]<http://www.todo-Linux.com/>
- [40]<http://www.webestilo.com/php/php00.phtml>
- [41]<http://www2.sep.gob.mx/indautor/>
- [42][www.hispafuentes.com/productos/#manuales-hf8](http://www.hispafuentes.com/productos/#manuales-hf8)
- [43]<http://www.cddhcu.gob.mx/leyinfo/122/>
- [44][www.revistamundopyme.cl](http://www.revistamundopyme.cl)
- [45][http://www.pyme.com.mx/articulos\\_pyme/todoslosarticulos/la\\_pyme\\_motor\\_de\\_desarrollo.htm](http://www.pyme.com.mx/articulos_pyme/todoslosarticulos/la_pyme_motor_de_desarrollo.htm)
- [46]<http://www2.netexplora.com/tombrad/empre1.html>
- [47]<http://lightning.prohosting.com/~rodoval/lenguajes.html>

[48] <http://www.codigoabierto.gob.mx/>

[49] [http://www.usabilitynet.org/tools/r\\_international.htm#9126-1](http://www.usabilitynet.org/tools/r_international.htm#9126-1)

[50] <http://www.cse.dcu.ie/essiscope/sm2/9126ref.html>



## GLOSARIO

**ADSL.-** Asymmetric Digital Subscriber Line. Pág. 58,67.

**API:** (Interfaz para programas de aplicación) Una serie de reglamentos y acuerdos que nos definen la manera en cómo llamar determinado servicio desde cierto programa. Pág. 86.

**Archivo:** Datos estructurados que pueden recuperarse fácilmente y usarse en una aplicación determinada. El archivo no contiene elementos de la aplicación que lo crea, sólo los datos o información con los que trabaja el usuario. Pág. 18, 25, 31,84.

**BOOT:** Arranque, iniciar el funcionamiento de la computadora. Hacer que el la computadora inicie la ejecución de instrucciones. Las computadoras personales contienen instrucciones preinstaladas en un chip ROM, las cuales se ejecutan automáticamente cuando se enciende el computador. Tales instrucciones buscan el sistema operativo, lo cargan y le transfieren el control. En una computadora de mayor tamaño, el procedimiento suele requerir una secuencia más elaborada de ingresos por teclado y precisiones de botones. El término se deriva de bootstrap (tirador de botas); al igual que estos tiradores ayudan en la acción de calzarse las botas, el ejecutar el boot ayuda al computador a recibir sus primeras instrucciones. Pág. 69.

**Cliente/Servidor:** Modelo lógico de una forma de proceso cooperativo, independiente de plataformas hardware y sistemas operativos. El concepto se refiere más a una filosofía que a un conjunto determinado de productos. Generalmente, el modelo se refiere a un puesto de trabajo o cliente que accede mediante una combinación de hardware y software a los recursos situados en un ordenador denominado servidor. Pág. 69.

**Clustering:** Agrupamiento. Las técnicas de clustering permiten el crecimiento de un sistema mediante la adición de procesadores o CPUs a la unidad primitiva. Pág. 57.

**DHCP:** (Dynamic Host Configuration Protocol). Protocolo de configuración dinámica de host que usan las computadoras para obtener información de configuración. El DHCP permite asignar una dirección IP a una computadora sin requerir que un administrador configure la información sobre la computadora en la base de datos de un servidor. Pág. 58,61.

**Distribución.-** Una distribución es un conjunto de programas recopilados a lo largo y ancho de sitios en Internet, organizados de tal manera que ofrezcan una solución particular o general hacia él o los usuarios. Pág. 49, 50, 53, 56, 59, 60,103.

**DSL.-** Digital Subscriber Line. Pág. 62,67.

**DNS:** (Sistema de Nombres de Dominio) Se trata de un servicio de búsqueda de los datos de uso general, que se distribuyen y multiplican. Se utilizan principalmente para buscar direcciones IP en sistemas anfitriones (hosts) tomando como base los nombres de éstos. Los dominios se clasifican según el rubro de las organizaciones o empresas que los utilicen, entre los más importante se encuentran: ".com" (comercial-empresas), ".edu" (educación, universidades, etc.), ".org" (organización sin fines de lucro), ".net" (operación de la red), ".gov" (Gobiernos en general) y ".mil" (ejercito de Estados Unidos). La mayoría de los países cuentan con su propio dominio. Por ejemplo, ".es" (España), ".de" (Alemania). Pág. 52, 58,61.

**FAT:** (File Allocation Table). Tabla de asignación de archivos. Parte del sistema de archivos del DOS y OS/2 que lleva un seguimiento de la ubicación de los datos almacenados en un disco. Cuando el disco se formatea a alto nivel, el FAT se registra dos veces y contiene una tabla con una entrada para cada cluster (conglomerado) en disco. Pág. 63.

**FTP:** (File Transfer Protocol) Protocolo de Transferencia de Archivos. Pág. 52, 54, 57, 61,92.

**Fortran:** (FORmula TRANslator) Traductor de fórmulas. Primer lenguaje de programación de alto nivel y compilador, desarrollado en 1954 por IBM.

Originalmente fue diseñado para expresar fórmulas matemáticas, y aunque en ocasiones se emplea para aplicaciones comerciales, es aún el lenguaje que más se usa para problemas científicos, de ingeniería y matemáticos. Pág. 58.

**FireWalls.-** Es un sistema de defensa que se basa en la instalación de una "barrera" entre la PC y la Red, por la que circulan todos los datos. Este tráfico entre la Red y la PC es autorizado o denegado por el firewall (la "barrera"), siguiendo las instrucciones que le hayamos configurado. Pág. 51.

**GNOME:** GNu Object Model Environment, uno de los GUIs más seguidos de UNIX. Pág. 51, 57, 61,86.

**GPL:** General Public License. Pág. 26.

**GCC:** Originalmente, estaba manejado como el compilador de C del GNU, pero ahora maneja muchos diversos lenguajes de programación además de C. Escrito originalmente por Richard Stallman en 1987, el GCC ahora es mantenido por un variado grupo de programadores alrededor del mundo. El GCC se ha adoptado como el principal compilador usado para sistemas como GNU/Linux, el DEB, OS X del mac, NeXTSTEP, y BeOS. Pág. 25, 61,80.

**GUI:** (Interfaz Gráfica de Usuario). Pág. 78.

**IBM.-** International Business Machines. Pág. 11,19,49,54,56,58,100.

**IDE:** (Integrated Drive Electronics) Electrónica de unidades integradas. Disco duro que contiene un controlador incorporado. Las unidades IDE se utilizan ampliamente en computadoras personales y su capacidad varía de 40MB a 1GB. La unidad se conecta a través de un cable de tipo cinta plana de 40 líneas a un adaptador de computador central IDE (con frecuencia llamado controlador IDE), que se enchufa en una ranura de expansión en la computadora personal. El adaptador de esta computadora controla hasta dos unidades IDE, pero los adaptadores avanzados y los adaptadores IDE ampliados controlan hasta cuatro. Pág. 62.

**Internet Protocol:** (Protocolo Internet) Se trata de una serie de normas que regulan la transferencia de paquetes de información a través de Internet. Actualmente se

desarrolla la versión 6 (IPv6), que va a permitir mejores prestaciones dentro del concepto QoS (Quality of Service). Pág. 92,93.

**IPX:** (Internet Packet Exchange). Protocolo de intercambio de Paquetes entre Redes y paquetes interredes. Protocolo de comunicaciones NetWare que se utiliza para encaminar mensajes de un nodo a otro. Los paquetes IPX incluyen direcciones de redes y pueden enviarse de una red a otra. Ocasionalmente, un paquete IPX puede perderse cuando cruza redes, de esta manera el IPX no garantiza la entrega de un mensaje completo. La aplicación tiene que proveer ese control o debe utilizarse el protocolo SPX de NetWare. IPX provee servicios en estratos 3 y 4 del modelo OSI (capas de red y transporte). Pág. 51,55, 58.

**ISA:** (Industry Standard Architecture). Bus de 8 bits instalado en los primeros PC fabricados por IBM, que se amplió posteriormente a 16 bits en los PCs AT. El bus permite la conexión de diferentes dispositivos al sistema a través de ranuras de expansión. Arquitectura industrial estándar. El bus de expansión es comúnmente utilizado en computadoras personales. Acepta tarjetas de conexión que controlan la presentación de video, disco y otros periféricos. La mayor parte de las tarjetas de expansión de las computadoras personales en el mercado son tarjetas ISA. Pág. 69,62.

**ISDN:** (Red Digital de Servicios Integrados --RDSI) La gran ventaja de este tipo de tecnología es que permite que los datos, conexiones de voz, etc. viajen a través de un solo cable. Actualmente este servicio es ofrecido por las principales compañías telefónicas a nivel mundial y goza de gran aceptación. Pág. 12.

**KDE:** Acrónimo que significa ?K Desktop Environment?, uno de los GUI más importantes para sistemas UNIX. Pág. 51, 55, 56, 60, 62, 85, 86.

**Kernel:** Núcleo. Parte fundamental de un programa, por lo general de un sistema operativo, que reside en memoria todo el tiempo y que provee los servicios básicos. Es la parte del sistema operativo que está más cerca de la máquina que puede activar el hardware directamente o unirse a otra capa de software que maneja el hardware. Pág. 10, 13, 32, 43,51,55,61.

**MIT:** Massachusetts Institute of Technology. Pág. 24, 25,44.

**NFS:** (Network File System). Sistema de archivos distribuidos para un entorno de red de área local. Posibilita que distintos sistemas conectados a una misma red accedan a ficheros remotos como si se tratara de locales. Originalmente desarrollado por Sun Microsystems Inc. Pág. 58, 61,92.

**OSI:** (Interconexión de Sistemas Abiertos) Se trata de un modelo de referencia que fue diseñado por el ISO con la finalidad de que se conviertan en estándares a nivel mundial con respecto a la arquitectura de redes y computadoras. Pág. 12, 91.

**PCI:** (Peripheral Component Interface) Término inglés que significa Conexión de Componentes Periféricos. Se trata de un tipo de ranura de conexión para tarjetas de ampliación que se encuentran en la placa base del ordenador. Pág. 51, 62,64.

**PCMCIA:** Asociación Internacional de Tarjetas de Memoria para Ordenadores Personales) Es un dispositivo electrónico en forma de tarjeta de crédito que se emplea en nuestra computadora. Una de sus principales funciones es la transmisión de datos, faxes, etc. utilizando teléfonos portátiles y laptops. Es ideal para la transmisión de datos digitales ya que no se ve afectada por los ruidos e interferencias. Alcanza prestaciones de 64.000 bps hasta 128.000 bps (si se usan los dos canales). Pag. 64.

**PGP:** (Privacidad Bastante Buena) Como una medida para proteger el contenido enviado en los correos electrónicos fue desarrollado por Phil Zimmermann este programa de distribución libre que por medio de técnicas criptográficas, evita que los correos puedan ser leídos por otros usuarios. También se puede utilizar para firmar electrónicamente un documento o archivo, para de esta manera realizar la autenticación del autor. Pág. 23.

**PHP.-** Acrónimo de Personal Home Page. Pág. 55,58.

**POP:** (Protocolo de Oficina de Correos) Se traduce como un protocolo que está diseñado para permitir al usuario leer el correo electrónico almacenado en un servidor. la versión actual es la POP3 y es definida por el RFC 1725. Pág. 57.

**RDSI:** Red Digital de Servicios Integrados. Es un tipo de red que agrupa distintos servicios anteriormente distribuidos a través de soportes distintos, siempre que se utilice tecnología digital: telefonía (con centralitas digitales), videoconferencia, teleinformática, videotex, mensajería electrónica, sonido, datos, imágenes, etc. Naturalmente, esto implica el uso de protocolos idénticos y redes físicas de banda ancha. Pág. 51, 55, 58,67.

**Procesos Batch.-** El procesamiento por lotes es un ejemplo del procesamiento FIFO de la ejecución de programas. Es utilizado sobre todo en las aplicaciones de administración y se puede describir como sigue:

Programa procesamiento\_por\_lotes

Inicio

Repetir

Leer un trabajo;

Ejecutar el trabajo;

Imprimir los resultados;

Sin fin

Fin.

Pág. 11.

**LAN:** Red de área local (LAN) desarrollada por Xerox, Digital e Intel. Es el método de acceso LAN que más se utiliza (seguido por Token Ring). Ethernet es una LAN de medios compartidos. Todos los mensajes se diseminan a todos los nodos en el segmento de red. Ethernet conecta hasta 1,024 nodos a 10 Mbits por segundo sobre un par trenzado, un cable coaxial y una fibra óptica. Pág. 19,90.

**SCO:** (Santa Cruz Operation). Se trata de una firma norteamericana especializada en sistemas Unix. Pág. 42.

**SCSI:** Small Computer System Interface. Es un interfase hardware de tipo serie para periféricos muy común. Pág. 62,64.

**SMTP:** (Protocolo Simple de Transferencia de Correo) Dicho protocolo es definido en STD 10, RFC 821, y se usa para la transferencia de correo electrónico entre

computadoras. Es un protocolo de servidor a servidor, de forma que para poder leer los mensajes se deben utilizar otros protocolos. Pág. 57.

**SSL:** (Capa de Conexión Segura) Este protocolo fue desarrollado por Netscape para permitir la transmisión encriptada y segura de información por Internet. Pág. 51,52.

**USB:** (Universal Serial Bus). Bus serie universal. La característica principal de este bus reside en que los periféricos pueden conectarse y desconectarse con el equipo en marcha, configurándose de forma automática. Conector externo que llega a transferencias de 12 millones de bits por segundo. Totalmente PnP, sustituirá al puerto serie y paralelo, gracias a la posibilidad de conectar 127 dispositivos. Pág. 51, 52,62.

**UUCP:** (Copia de UNIX a UNIX) En un principio se utilizaba para la transmisión de información entre los sistemas operativos UNIX, posteriormente se aplicó para describir a la red global que usaba el protocolo UUCP para el envío de información (noticias, correo electrónico, etc.). Pág. 58, 92,93.

**XML:** (Lenguaje Extensible de Marcado) Se desarrolló este sistema para la promoción del uso del lenguaje SGML en Internet. XML no es un lenguaje sino un metalenguaje, esto es que, sirve para crear lenguajes. No se trata de una extensión ni un componente de HTML. Pág. 86,87.

**Zope.-** Zope es Servidor Web que permite a equipos colaborar en la creación y la gerencia de los usos de negocio dinámicos tales como intranets y portals. Pág. 55,57.