

# **UNIVERSIDAD NACIONAL DEL NORDESTE**

**Facultad de Cs. Exactas y  
Naturales y Agrimensura**

**Cátedra: SISTEMAS OPERATIVOS.**

**Adscripta: KUBSKI, MARIANA.**

**Año: 2004**

# LINUX



- ◆ Qué es Linux?
- ◆ Antecedentes.
- ◆ Licencia.
- ◆ Características.
- ◆ Entorno de Trabajo.
- ◆ Estructura General.
- ◆ Sistema de Ficheros.
  - ◆ Tipos.
  - ◆ Path.
  - ◆ Permisos de Acceso.
- ◆ Distribuciones Comerciales.
  - ◆ Elementos Claves.
- ◆ Recomendaciones de Instalación.

# Qué es Linux?



- ◆ Linux es un clónico del sistema operativo UNIX. Es una implementación de UNIX distribuida **gratuitamente** en los términos de la Licencia GPL.
- ◆ Lo que hace a Linux tan diferente es que es una **implementación gratuita**.
- ◆ Además fue y aún es desarrollado por un grupo de voluntarios, principalmente en Internet, intercambiando código, comentando fallos, y arreglando los problemas en un entorno abierto.

# Antecedentes



- ◆ Inicialmente fue desarrollado por **Linus Torvalds** en la Universidad de Helsinki, en Finlandia.
- ◆ Se inspiraba en Minix, un pequeño UNIX desarrollado por Andy Tanenbaum. Dirigido a aquellos usuarios de Minix que querían algo más.
- ◆ A su desarrollo se sumaron muchos programadores y expertos de UNIX a lo largo y ancho del mundo, a través de Internet.
- ◆ Hoy Linux es ya un clónico de UNIX completo.

# Licencia GPL



- ◆ La **Licencia Pública General** permite a cualquiera trabajar en Linux. Puede ser vendido, como así también copiado sin costo o restricción alguna.
- ◆ Este tipo de licencia de software libre requiere que si se realiza un cambio o agregado al código GPL, éste debe permanecer bajo los mismos términos de GPL, de manera que ningún desarrollador gane alguna ventaja encima de otros contribuyentes del desarrollo.
- ◆ Bajo la GPL el derecho de propiedad de Linux puede ser sostenido por Linus Torvalds y otros pero ellos no tienen ningún otro derecho para restringir el uso de él.

# Características



- ◆ Multitarea.
- ◆ Multiusuario.
- ◆ De planificación mixta.
- ◆ Casi todas las implementaciones son de memoria virtual.
- ◆ Multiplataforma: fácilmente portable a diferentes tipos de ordenadores.
- ◆ Librerías compartidas de carga dinámica (DLL's).
- ◆ Libre acceso para desarrollar o modificar módulos.
- ◆ Escritorios virtuales múltiples que posibilitan varias sesiones de login.



- ◆ Protección de memoria entre procesos, de manera que uno de ellos no pueda colgar el sistema.
- ◆ Código fuente disponible, incluyendo el núcleo completo, herramientas de desarrollo y programas de usuario. Todo se puede distribuir libremente.
- ◆ Núcleo usualmente empaquetado con varios ejecutables que comprenden un sistema operativo funcional. Estos paquetes son llamados “distribuciones”.
- ◆ Implementa todo lo necesario para trabajar en red con TCP/IP. Soporta protocolos DNS, HTTP, SMTP, FTP, TNP, Y PPP/SLIP.
- ◆ Fabricantes de Hardware como IBM y COMPAQ brindan su apoyo.

# Entorno de Trabajo



- ◆ Existen dos maneras de trabajar con Linux:
  - ◆ Utilizando el *Shell*. Interpreta los comandos que introduce el usuario y realiza las acciones oportunas en respuesta.
  - ◆ Utilizando el sistema *X-Window*. Controla y maneja la interfaz de usuario. Permite que el usuario tenga un control absoluto de la representación de los elementos gráficos.

# Estructura General

Se puede dividir en varios componentes bien diferenciados:

- ◆ **Núcleo o Kernel:** es el corazón del sistema, posee un gran número de funciones de sistema que son invisibles para el usuario, llamadas **system calls**. Realiza todas las labores propias de un sistema operativo.
- ◆ **Caparazón o Shell:** actúa como intérprete de comandos. Es un programa que lee las órdenes suministradas, las decodifica y comunica al núcleo para realizar la acción especificada.
- ◆ **Programas de utilidad.**



# Sistema de Ficheros



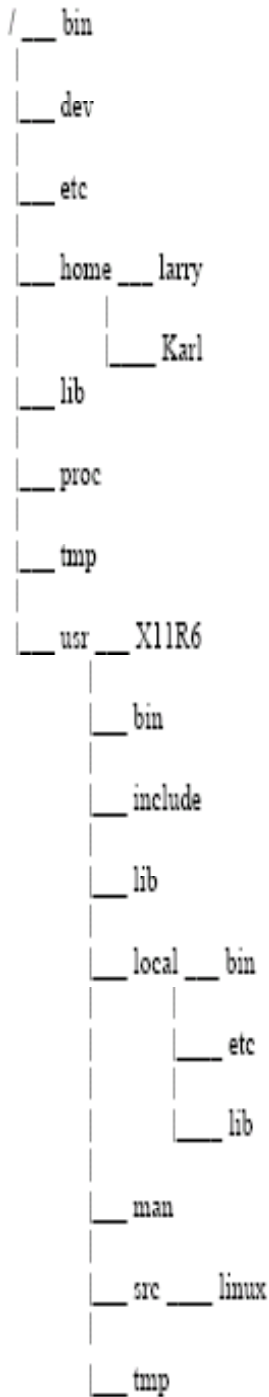
- ◆ Un fichero es un conjunto de información al que se le da un nombre.
- ◆ No existe, a nivel de usuario, el concepto de volumen, ni de dispositivo físico. El usuario no sabe en qué disco están los ficheros.
- ◆ Existen tres tipos:
  - ◆ *Ordinarios*: son cadenas de bytes. Pueden ser texto, objetos, ejecutables, bibliotecas de módulos, entre otros.
  - ◆ *Directorios*: contienen nombres de ficheros y su dirección física. Puede pensarse en ellos como carpetas.
  - ◆ *Especiales*: asociados a dispositivos de entrada /salida. Contienen referencias a los drivers. Pueden ser de tipo bloque y tipo carácter.



# Sistema de Ficheros (cont.)

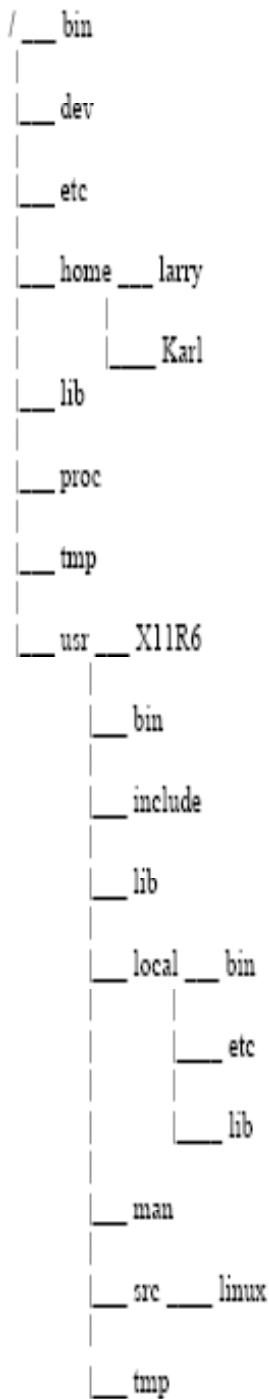
- ◆ Convenciones para el nombre:
  - ◆ Nombre: entre 1 y 255 caracteres.
  - ◆ Se puede utilizar cualquier caracter excepto /
  - ◆ No es recomendable emplear los caracteres con significado especial en Linux:  
= ' " ` \* ; - ? [ ] ( ) ! \ & (Si se los utiliza, introducir el nombre del fichero entre comillas.)
  - ◆ Las letras mayúsculas y minúsculas se consideran diferentes.
- ◆ Enlace: tipo de archivo ordinario.
  - ◆ Objetivo: crear un nuevo nombre para un archivo determinado. Especialmente útil cuando se quiere que un grupo de personas trabaje sobre un mismo fichero, dado que permite compartir el fichero pero centraliza las modificaciones.

# Sistema de Ficheros: Camino o Path



- ◆ Origen único: **raíz o root /**. Bajo este directorio se encuentran todos los ficheros a los que puede acceder el sistema operativo.
- ◆ **/ dev** Ficheros especiales de dispositivos conectados al ordenador.
- ◆ **/ lib** Bibliotecas del sistema. Necesarios para ejecutar los programas que residen en **/bin**.
- ◆ **/bin** Órdenes más empleadas.
- ◆ **/etc** Ficheros de configuración del sistema. No debe aparecer ningún programa.
- ◆ **/home** Directorio de usuarios.
- ◆ **/tmp** Ficheros temporales.
- ◆ **/proc** Ficheros especiales que, o bien reciben, o bien envían información al kernel del sistema.
- ◆ **/sbin** Programas que son únicamente accesibles al superusuario o root.

# Sistema de Ficheros: Camino o Path (cont.)

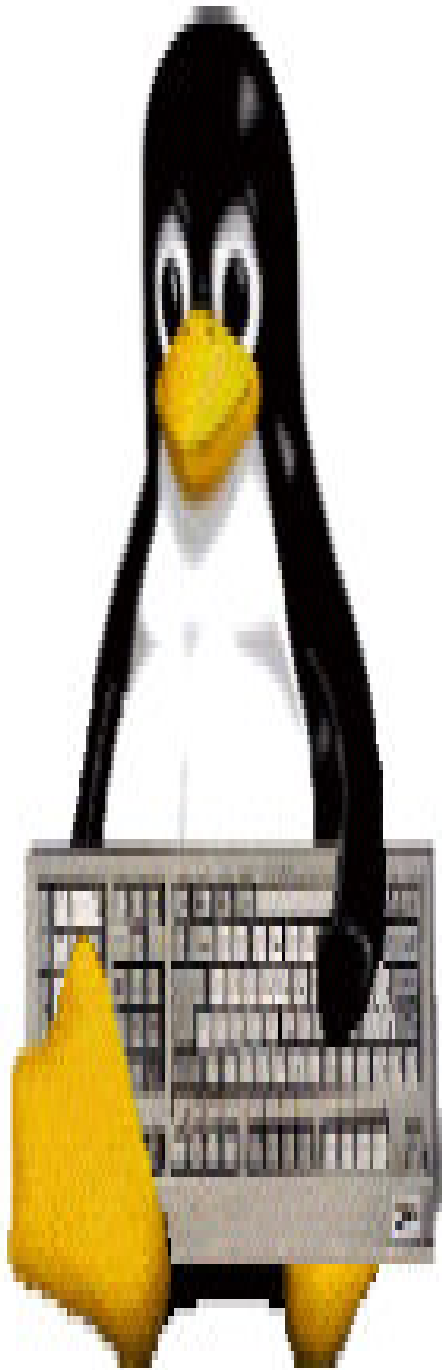


- ◆ **/usr** órdenes, bibliotecas y programas adicionales.
- ◆ **/usr/bin** Programas de uso general, incluye el compilador de C/C++.
- ◆ **/usr/include** Ficheros de cabecera de C/C++.
- ◆ **/usr/lib** Librerías generales de los programas.
- ◆ **/usr/man** Manuales accesibles con el comando man.
- ◆ **/usr/doc** Documentación general del sistema.
- ◆ **/usr/etc** Ficheros de configuración generales.
- ◆ **/usr/info** Ficheros de información de GNU.
- ◆ **/usr/sbin** Programas de administración del sistema.
- ◆ **/usr/src** Código fuente de programas.
- ◆ Cada usuario tiene un directorio **HOME** que es el directorio asignado a ese usuario para que almacene sus ficheros.

# Sistema de Ficheros: Permisos de Acceso



- ◆ Linux mantiene un sistema de permisos de acceso muy estricto, con el fin de controlar qué es lo que se puede hacer y quién lo puede hacer.
- ◆ Estos permisos se identifican con letras y son:
  - ◆ **r**: permiso de lectura en el fichero.
  - ◆ **w**: permiso de escritura en el fichero.
  - ◆ **x**: permiso de ejecución del fichero.
  - ◆ **s**: permiso para cambiar el propietario del fichero.
- ◆ Los programas ejecutables de Linux no están marcados por una determinada extensión (.exe) sino por un atributo, el permiso de ejecución **x**.



# Distribuciones Comerciales

- ◆ Una **Distribución** es un agrupamiento del núcleo del sistema operativo Linux y otra serie de aplicaciones de uso general.
- ◆ Las distribuciones más conocidas son SuSe, RedHat, Debian, Slackware, Caldera, entre otros.
- ◆ Elementos claves:
  - ◆ **Boot Manager:** Administrador de inicio (LILO).
  - ◆ **Interfaz de Usuario:** Escritorios gráficos (CDE, KDE, GNOME).
  - ◆ **Sistema X-Window:** Subsistema gráfico que soporta una interfaz de usuario gráfica. (Xfree86).

## Distribuciones Comerciales

# Elementos Claves (cont.)



- ◆ **Servicio de Internet:** soporta TCP/IP y protocolos más comunes. (Servidor Apache).
- ◆ **Servicios de Impresión de Archivos:** permiten al sistema acceder a los recursos de una red. (NFS; Samba).
- ◆ **Aplicaciones.**
- ◆ **Administradores de Paquetes:** herramientas para instalar, desinstalar y actualizar aplicaciones. (Debian Package Management System; Red Hat Package Manager).
- ◆ **Herramientas del Programador:** diferentes herramientas de programación (lenguajes de programación, eliminadores de fallas, otros).
- ◆ **Bibliotecas de vínculo dinámico:** GNU Libc y Libc5.

# Recomendaciones de Instalación



- ◆ La instalación no es un proceso sencillo. Se debe considerar una serie de aspectos fundamentales:
  - ◆ Particionamiento del disco.
  - ◆ Gestor de arranque.
  - ◆ Drivers de monitor y placa de video.
- ◆ **No aceptar las opciones defaults a menos que se tenga la seguridad de que sea la opción correcta.**

# Conclusión



Linux se percibe como un sistema confiable. No se lo pensó como algo fácil de emplear, sino como un sistema sumamente flexible.

Es un sistema multiplataforma, por lo que permite trabajar en un amplio rango de hardware.

Soporta casi todos los protocolos de red e internet.

Su naturaleza abierta permite configurarlo según los gustos personales.

# Bibliografía



- ◆ **Unix.**

Extracto de Operating Systems: Internals and Design Principles, Cuarta Edición de William Stallings. Prentice Hall. 2000.

- ◆ **Aprenda Linux Como si Estuviera en Primero.** Javier García de Jalón; Iker Aguinaga; Alberto Mora. Escuela Superior de Ingenieros Industriales de San Sebastián – Universidad de Navarra. Enero 2000.

- ◆ **Manual de Unix.**

Jonathan Noel Tombs; Jorge Chavez Orzáez; Rev. 2.4. Noviembre 1999.

- ◆ **Apunte Unix Avanzado.**

Centro de Computación - Facultad de Ciencias Físicas y Matemáticas. Universidad de Chile. Julio 1998.



**GRACIAS POR  
SU ATENCIÓN !!**