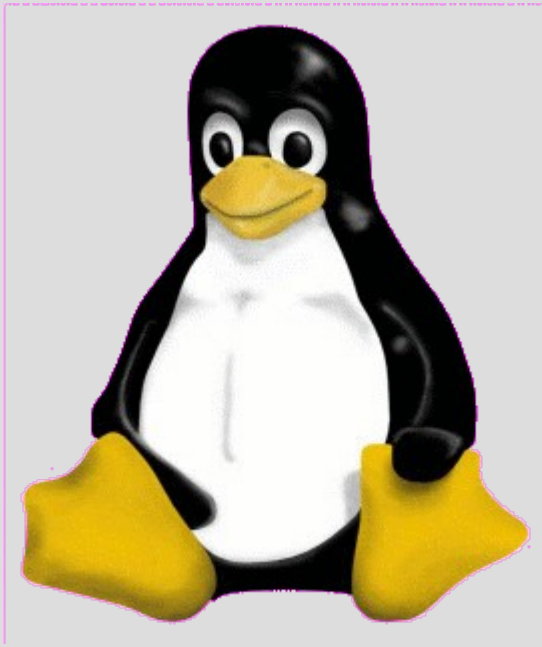
Introducción a Linux

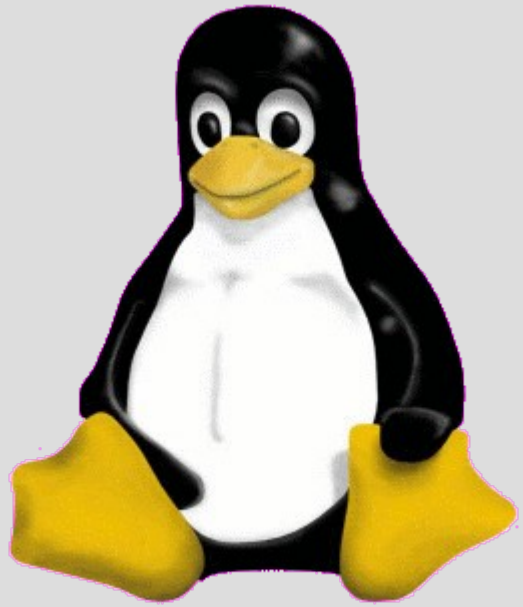
El regreso a los años 70: la consola (II)



Introducción a Linux

El regreso a los años 70: la consola (II)

- ✓ ¿Qué es y cómo acceder a una consola?
- ✓ Algunos comandos para moverse
- ▪ Sintaxis general de parámetros en BASH
- ▪ Encadenamiento de comandos
- ▪ Otros comandos GNU
- ▪ Editores de textos



Introducción a Linux

El regreso a los años 70: la consola (II)

Sintaxis general de parámetros en BASH

- Podemos adjuntar varios comandos seguidos, que se ejecutarán uno a continuación de otro, separándolos con “;”
- Cada comando acepta parámetros, separados por espacios. Si algún parámetro debe contener espacios, habrá que entrecomillarlo o “escapar” los espacios.
- Los espacios al inicio y al final del comando son descartados.

Introducción a Linux

El regreso a los años 70: la consola (II)

Sintaxis general de parámetros en BASH

- Cada comando ejecutado crea un nuevo proceso en el sistema. Normalmente, BASH espera que ese proceso termine de ejecutarse antes de volver al prompt.

- Si queremos volver inmediatamente al prompt mientras se ejecuta el comando, podemos usar el carácter “&” al final.

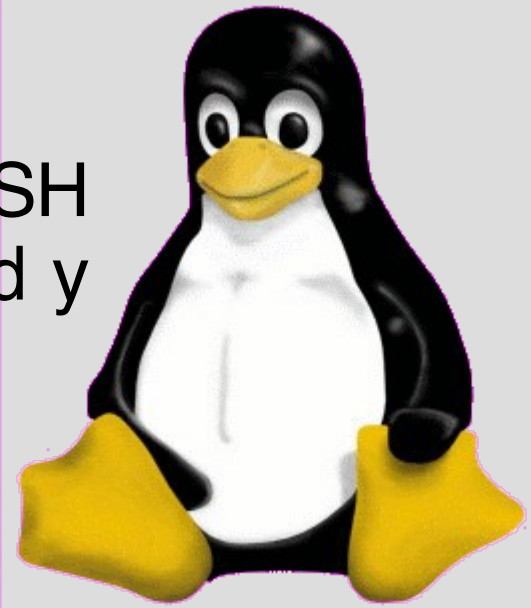
Introducción a Linux

El regreso a los años 70: la consola (II)

Sintaxis general de parámetros en BASH

- Cada proceso abierto equivale a una ventana en el modo gráfico. Pasar un proceso a “background” (con &) es como desactivar esa ventana.

- Disponemos de comandos internos de BASH para gestionar los procesos en background y reactivarlos o detenerlos.



Introducción a Linux

El regreso a los años 70: la consola (II)

Sintaxis general de parámetros en BASH

- Cada parámetro de un comando será una opción, su valor o un patrón de fichero.
- Al ir escribiendo un nombre de fichero BASH nos ayuda expandiendo automáticamente el nombre si pulsamos <TAB>
- Un patrón de fichero puede coincidir con 1 ó más nombres de ficheros. Disponemos de comodines para “agrupar” varios ficheros en un sólo parámetro.

Introducción a Linux

El regreso a los años 70: la consola (II)

Sintaxis general de parámetros en BASH

Tenemos los siguientes comodines:

- "**`*`**" Se sustituye por cero o más caracteres.
- "**`?`**" Se sustituye por exactamente 1 caracter.
- "**`[abc]`**" Se sustituye por 1 caracter de los indicados, ningún otro. Se pueden indicar rangos, ej: **[1-5] , [a-zA-Z]** …
- "**`[!abc]`**" Se sustituye por 1 caracter, cualquiera que no sea uno de los indicados.

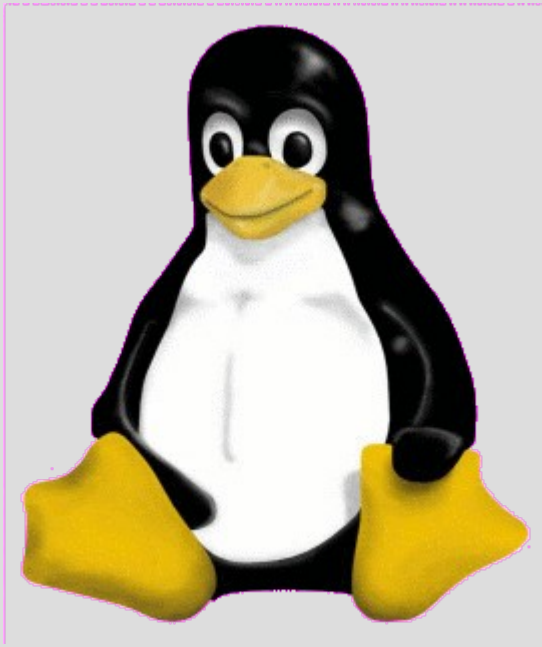
Introducción a Linux

El regreso a los años 70: la consola (II)

Sintaxis general de parámetros en BASH

Probemos…

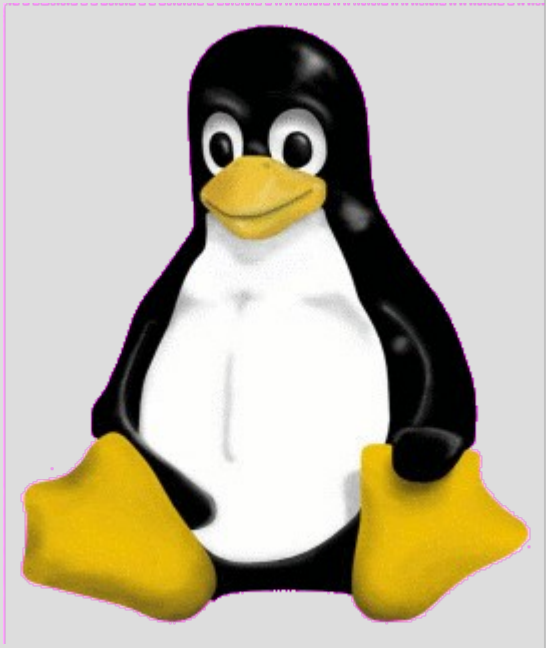
```
[usr@host ~]$ cd /usr/share/man/man1
[usr@host man1]$ ls *.gz
[usr@host man1]$ ls perl*
[usr@host man1]$ ls perl???.1
[usr@host man1]$ ls perl[abc]??.1
```



Introducción a Linux

El regreso a los años 70: la consola (II)

Encadenamiento de comandos



Introducción a Linux

El regreso a los años 70: la consola (II)

Encadenamiento de comandos

- Todos los comandos GNU procesan una entrada y generan una salida, adicionalmente un error.
- Salvo que indiquemos lo contrario mediante parámetros, el comando leerá de la entrada estándar y escribirá en la salida estándar.
- La entrada estándar es el teclado de la consola actual y la salida estándar es la pantalla de la consola actual.

Introducción a Linux

El regreso a los años 70: la consola (II)

Encadenamiento de comandos

La mayoría de comandos definen su propia sintaxis de parámetros para recibir entrada y generar salida, pero… podemos forzar una entrada y salida diferente mediante los caracteres:

- “>” , envía la salida del comando al fichero que indiquemos, si no existe se crea.
- “>>” , igual que el anterior, pero si ya existe el fichero, los datos se añaden al final.
- “<“ , envía el fichero indicado a continuación como entrada estándar del proceso.
- “|” , entuba la salida de un proceso a la entrada de otro.

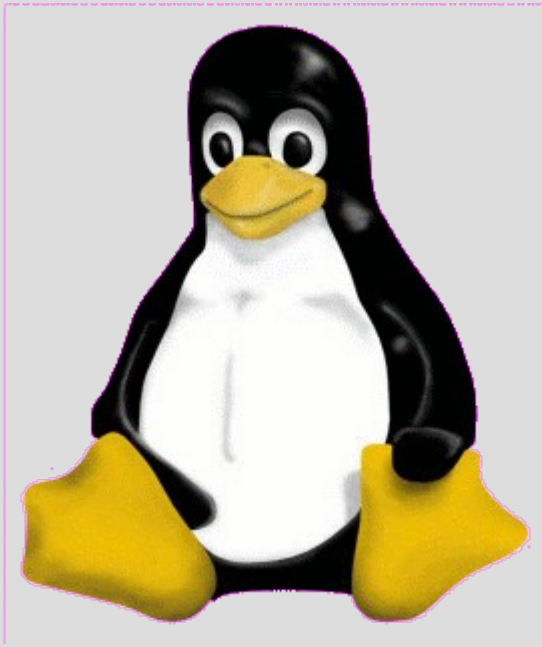
Introducción a Linux

El regreso a los años 70: la consola (II)

Encadenamiento de comandos

Probemos…

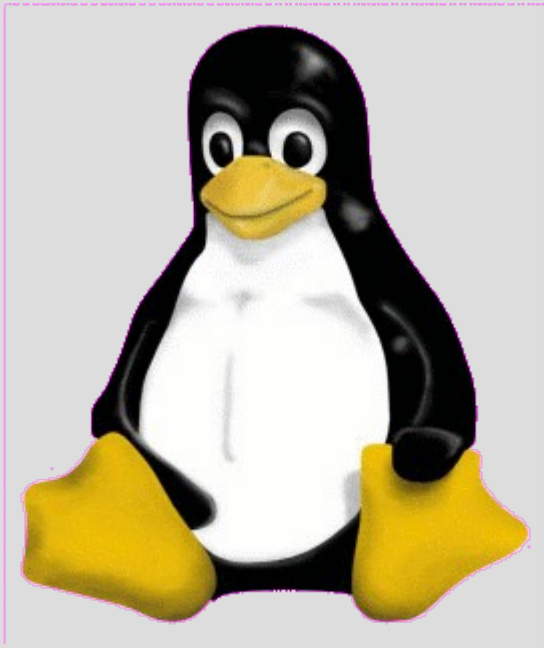
```
[usr@host ~]$ cd /usr/share/man/man1
[usr@host man1]$ ls *.gz > lista_gz
[usr@host man1]$ cat lista_gz
[usr@host man1]$ ls -l | more
[usr@host man1]$ ls *.gz | cat > lista_gz2
[usr@host man1]$ cat < lista_gz >> lista_gz2
```



Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

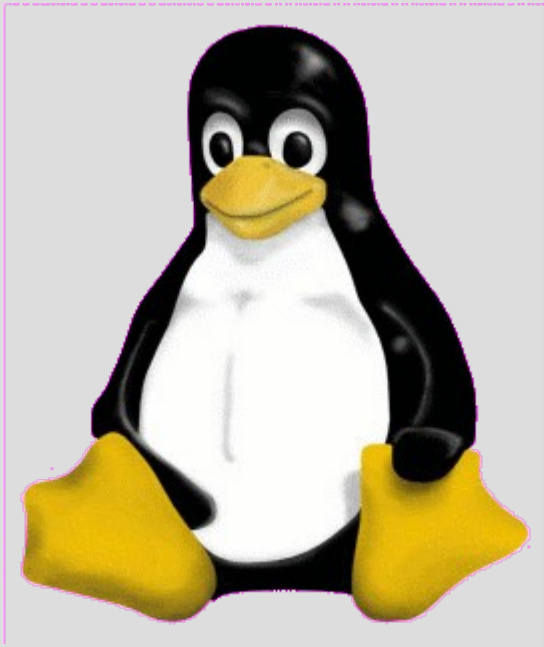


Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

- grep , egrep , fgrep , bzgrep , …
- tar , bzip2 , gzip
- su , sudo
- diff , cmp
- head , tail , zcat
- wc , df , du
- whatis , whereis , apropos



Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

grep y sus variantes.

Permite aplicar expresiones regulares a una entrada de texto y devuelve las lineas que coinciden.

Ejemplo:

```
[usr@host ~]$ ls -l | grep perl
```

mostraria todas las lineas del listado que contengan el texto “perl”

Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

tar , bzip2 , gzip

"tar" es una aplicación de archivado de ficheros, crea un único fichero a partir de varios.

"bzip2" y "gzip" son compresores, que sólo permiten un fichero por archivo comprimido.

típicamente se combinan ambos para, primero archivar todo, y luego comprimirlo.

Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

su , sudo

Permiten cambiar a modo superusuario.

“su” abre una nueva sesión, subproceso de la actual, con el usuario root. Nos pide contraseña, y a partir de ahí somos root hasta cerrar esa sesión.

“sudo” permite ejecutar un comando como root sin abrir una sesión, al terminar volvemos a nuestro usuario.

Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

diff , cmp

Permiten comparar ficheros.

“diff” se usa para ficheros de texto, devuelve un listado de cambios de un fichero respecto a otro.

“cmp” funciona a nivel de bytes, se usa más para binarios.

Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

head , tail , zcat

Similares a “cat”, muestran el contenido de los ficheros.

“head” muestra las primeras lineas.

“tail” muestra las últimas lineas.

“zcat” permite ver el contenido de un fichero comprimido con zip o gzip.

Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

wc , df , du

"wc", *word counter*, cuenta las palabras, lineas y caracteres de la entrada.

"df", *diskfree*, cuenta el espacio libre en las particiones.

"du", *disk usage,* cuenta el espacio usado en disco por los ficheros indicados.

Introducción a Linux

El regreso a los años 70: la consola (II)

Otros comandos GNU

whatis , whereis , apropos

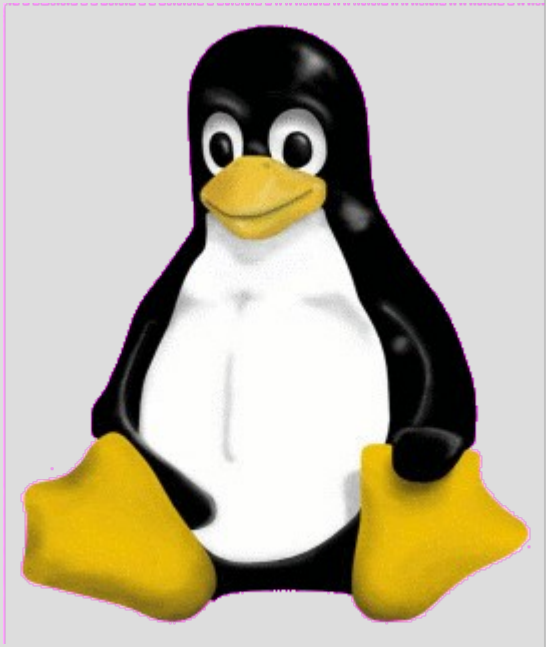
Comandos adicionales al man y que lo indexan para facilitar encontrar un comando que haga lo que necesitamos.

“whatis” da información resumida de lo que hace un comando.
“whereis” devuelve la ubicación física de un comando.
“apropos” permite buscar comandos por lo que hacen y no por el nombre.

Introducción a Linux

El regreso a los años 70: la consola (II)

Editores de texto

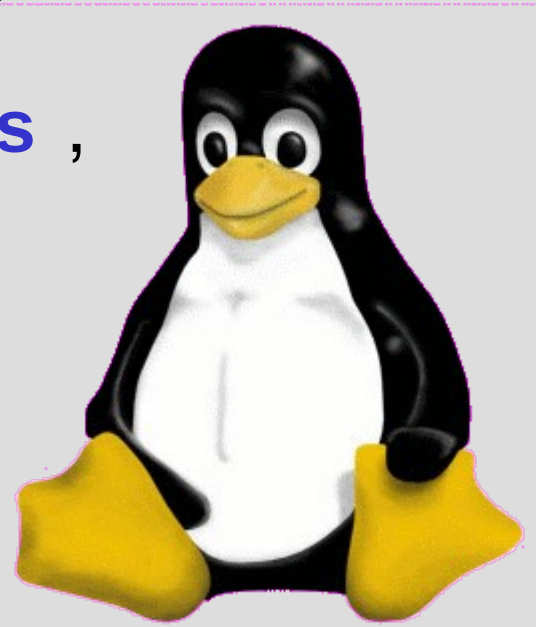


Introducción a Linux

El regreso a los años 70: la consola (II)

Editores de texto

- Hay una gran variedad de editores, cada uno con un modo diferente de trabajar al que deberemos adaptarnos para elegir el que más nos guste.
- Los más conocidos son: **vi / vim** , **emacs** , **nano** y **ed**.



Introducción a Linux

El regreso a los años 70: la consola (II)

Editores de texto

nano

Es un editor ligero y bastante parecido al modo de edición visual al que estamos acostumbrados.

Al abrir un fichero, en la parte inferior tendremos continuamente el menú de comandos disponible, todos son combinaciones de teclas.

Introducción a Linux

El regreso a los años 70: la consola (II)

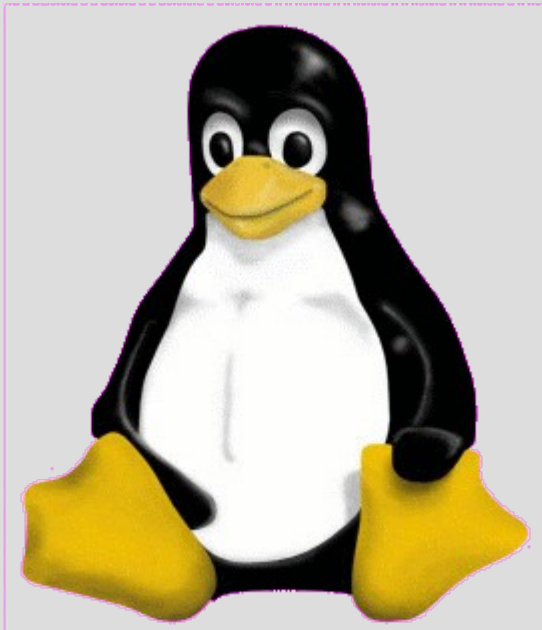
Editores de texto

vi ó **vim**

Este editor es muy comúnmente usado en entornos linux por su rapidez de edición.

Tiene 4 modos básicos disponibles:

- Edición rápida,
- Comando,
- Inserción y
- Reemplazo.



Introducción a Linux

El regreso a los años 70: la consola (II)

Editores de texto

vi ó **vim**

Por defecto, empezamos en modo edición rápida, los comandos son las pulsaciones del teclado.

Para cambiar a modo comando, pulsamos ":" seguido del comando e intro.

Para pasar a modo inserción/reeemplazo, pulsamos "INS", y luego "ESC" para salir.

Introducción a Linux

El regreso a los años 70: la consola (II)

Editores de texto

vi ó **vim**

Resumen de comandos básicos:

- **dd** , cortar la linea actual , **d<n><🡻>** corta la linea actual más n lineas inferiores.
- **yy** , copiar la linea actual, también y<n>
- **p** , pegar el buffer
- **u** , *undo*, deshacer.
- **:w** , *write*, guardar
- **:q** , *quit*, salir, igual q el anterior se puede forzar siguiéndolo con "**!**"
- **:x** , *write&quit*

Introducción a Linux

El regreso a los años 70: la consola (II)

Editores de texto

vi ó **vim**

Resumen de comandos básicos:

- **:<n>** , salta a la linea n del texto
- **/<texto>** , busca el texto
- **//** , repite la última búsqueda.