

---

# Sobreviviendo a la línea de comandos

## Una introducción a GNU/Linux

Jorge Juan <[jjchico@gmail.com](mailto:jjchico@gmail.com)> 2014

Usted es libre de copiar, distribuir y comunicar públicamente la obra y de hacer obras derivadas bajo las condiciones de la licencia Attribution-Share alike de Creative Commons.

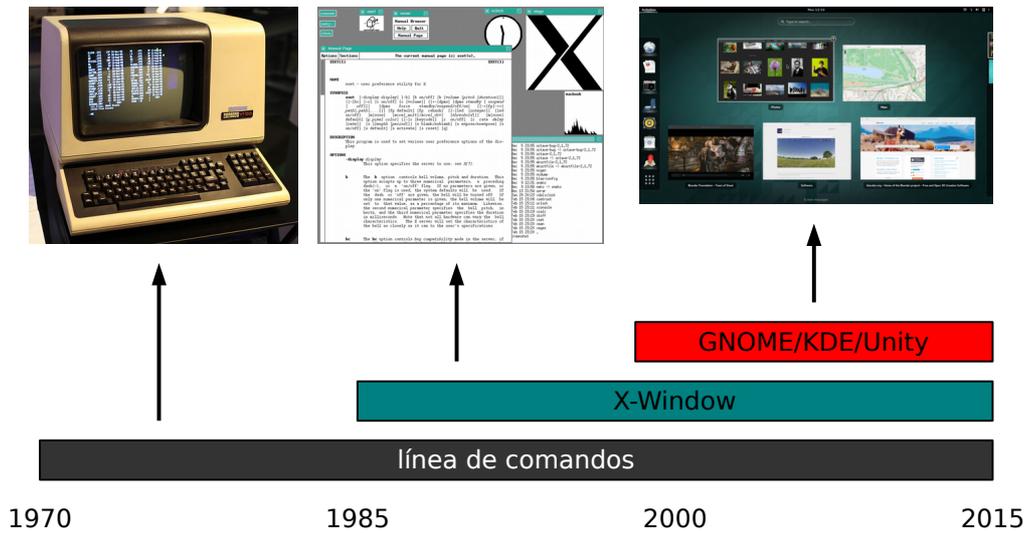
Puede consultar el texto completo de la licencia en <http://creativecommons.org/licenses/by-sa/3.0/>

## Contenidos

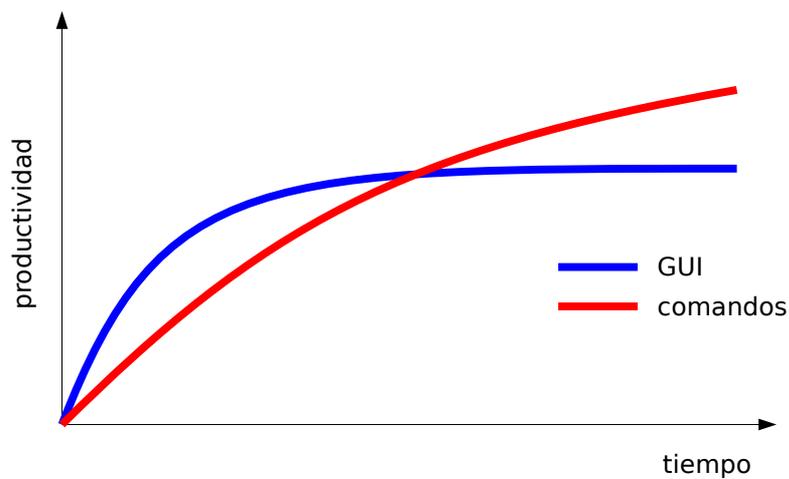
---

- **Generalidades**
- Redirección y tuberías
- Gestión de archivos
- Algunos comandos útiles
- Aplicaciones de terminal

# Interfases de usuario UNIX/GNU/Linux



# Curva de aprendizaje



# Interfaz de comandos

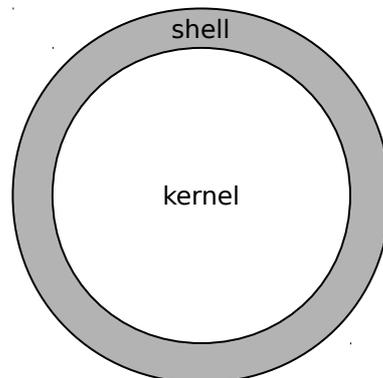
---

- Base para otras aplicaciones gráficas
- Alta productividad en muchas tareas
- Acceso a tareas exclusivas
- Curva de aprendizaje pronunciada

# Shell: intérprete de comandos

---

UNIX



# Shell: intérprete de comandos

---

- Tipos de shells:
  - /bin/sh: shell original de UNIX
  - **/bin/bash**: Bourne again shell. (/bin/bash)
  - /bin/csh: C shell
  - Otros: tcsh, ksh, ...
- Interpreta los comandos y ejecuta, devolviendo el resultado

# Formato de la línea de comandos

---

- Comando: archivo ejecutable
  - /bin, /usr/bin, /usr/local/bin, /sbin, /usr/sbin, /usr/local/sbin
- Formato:
  - <comando> [OPCIONES] [ARGUMENTOS]
- [OPCIONES] pueden ser:
  - Letras precedidas del símbolo '-'
  - Palabras precedidas de los símbolos '--'
- [ARGUMENTOS]: nombres de archivos o cualquier otro dato a pasar al programa.

```
$ ls -l /home
$ ls --human-readable /home
```

# Caracteres especiales

---

- Separadores
  - Espacio: separa parámetros
  - Enter: ejecuta comandos
- Comodines: expansión de nombres de archivos
  - \*: representa cualquier cadena de caracteres.
  - ?: representa cualquier carácter simple.
  - [<rango>]: cualquier carácter en <rango>.
- Escape:
  - \ " '
- Otros
  - # < > ( ) { } \$ & [[ ]]

# Comodines

---

```
$ ls
cap1 cap2 cap3 suma.c resta.c

$ ls c*
cap1 cap2 cap3

$ ls cap?
cap1 cap2 cap3

$ ls *.c
suma.c resta.c

$ cp "Mis Documentos" /tmp
$ cp 'Mis Documentos' /tmp
$ cp Mis\ Documentos /tmp
```

# Secuencias de control del terminal

---

- Permiten enviar señales los programas ejecutados bajo el control del terminal
  - Ctrl-C: Terminar
  - Ctrl-Z: Suspende
  - Ctrl-D: Fin de archivo
  - Ctrl-V: próximo carácter es literal (escapar sec. de control)

```
$ find / carta.txt  
<Ctrl-C>  
  
$ cat  
hola  
hola  
<Ctrl-D>
```

# Control de trabajos

---

- Control de trabajos
  - Ctrl-Z: (desde el terminal) detiene trabajo en curso
  - bg/fg: enviar a segundo/primer plano
  - '&': (al final del comando) ejecuta en segundo plano
  - jobs: lista de trabajos del terminal

# Comandos de ayuda

---

- man: páginas de manual. Información detallada sobre cada comando y sus opciones
  - 'q' para salir.
- info: es similar a 'man', más didáctico, GNU.
- whatis: breve descripción de cada comando.
- apropos <término>: comandos relacionados
- <comando> --help: ayuda básica

```
$ ls --help
...
$ man ls
...
$ info sed
...
$ whatis passwd
```

# Localización de comandos

---

- Comandos de propósito general
  - /bin, /usr/bin, /usr/local/bin
- Comandos de administración
  - /sbin, /usr/sbin, /usr/local/sbin
- Comandos del usuario
  - ~/bin (si existe)
- which: localiza un comando
- whereis: localiza un comando, página de manual, etc.

```
$ which ls
/bin/ls

$ ls /*bin /usr/*bin | wc -l
3294
```

# Trucos

---

- Copia/pega rápido en X-Window
  - Copiar: click/doble click izquierdo + arrastrar
  - Pegar: click medio en punto de destino
- Copia/pega en terminal
  - Copiar: Ctrl+Shift+C. Pegar Ctrl+Shift+V
- "cd" a carpeta desde el gestor de archivos
  - cd + arrastrar carpeta al terminal + ENTER
- Limpiar terminal
  - clear, Ctrl-L

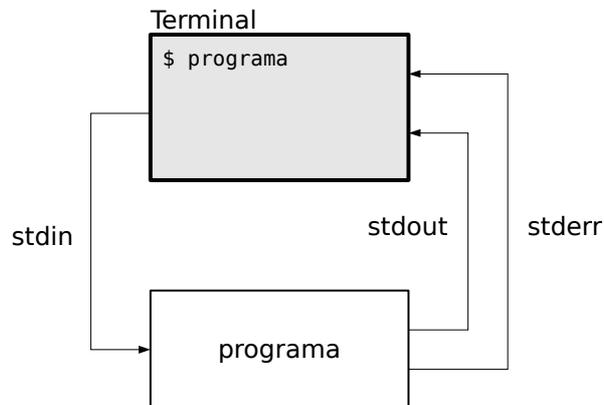
# Contenidos

---

- Generalidades
- **Redirección y tuberías**
- Gestión de archivos
- Algunos comandos útiles
- Aplicaciones de terminal

# Redirección Entrada/salida estándar

---



```
$ ls dir.txt list.txt
ls: no se puede acceder a list.txt: No existe el fichero ó directorio
dir.txt
```

## Redirección de la salida estándar a un archivo

---

- `>`: redirección a un archivo nuevo o borra el anterior
- `>>`: redirección a un archivo nuevo o añade al anterior
- `2>`, `2>>`: igual, pero para la salida de error
- `&>`, `&>>`: igual pero para ambas salidas
  
- `cat`: lee un archivo o la entrada estándar y copia a la salida estándar

## Redirección de la salida estándar a un archivo

---

```
$ ls /home > dir.txt
$ cat dir.txt
ana
jorge

$ ls /etc > etc.txt
$ less etc.txt

$ ls dir.txt list.txt 2> error.txt
dir.txt
$ cat error.txt
ls: no se puede acceder a list.txt: No existe el fichero ó directorio

$ ls dir.txt list.txt > out.txt 2> error.txt
$ cat out.txt
dir.txt
$ cat error.txt
ls: no se puede acceder a list.txt: No existe el fichero ó directorio
```

## Redirección de la salida estándar a un archivo

---

```
$ cat > lista.txt
leche
galletas
zumo
pañales
Ctrl-D
$ cat lista.txt
leche
galletas
zumo
pañales
```

```
$ cat >> lista.txt
chocolate
Ctrl-D
$ cat lista.txt
leche
galletas
zumo
pañales
chocolate
```

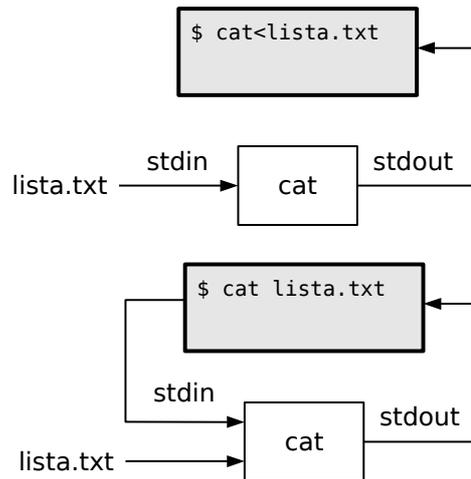
# Redirección de la entrada estándar desde un archivo

---

- <: redirección desde un archivo

```
$ cat < lista.txt
leche
galletas
zumo
pañales
chocolate

$ cat lista.txt
leche
galletas
zumo
pañales
chocolate
```



# Archivos especiales /dev/null y /dev/zero

---

- /dev/null
  - escritura: elimina lo que se escribe
  - lectura: proporciona "fin de archivo"
- /dev/zero
  - escritura: elimina lo que se escribe
  - lectura: proporciona infinitos "ceros"

# Archivos especiales /dev/null y /dev/zero

---

```
# Descarta la salida de un comando
$ ls /etc/fstab > /dev/null

# También la salida de error
$ ls /etc/fstab &> /dev/null

# Crea un archivo vacío o elimina su contenidos
$ cat /dev/null > dir.txt

# Mantiene ocupado al sistema
$ cat /dev/zero > /dev/null
Ctrl-C
```

## Tuberías (pipes)

---

- Conecta la salida estándar de un comando con la entrada estándar de otro
- Permite concatenar comandos sin crear archivos intermedios
- Una de las claves de UNIX
- Especialmente útil con filtros
  - sort: ordena archivos de texto plano
  - wc: cuenta líneas, palabras y caracteres
  - grep: muestra líneas que concuerdan con un patrón
  - ...

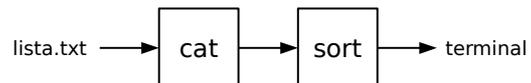
# Tuberías

---

```
$ cat lista.txt  
leche  
galletas  
zumo  
pañales  
chocolate
```

```
$ cat lista.txt | sort  
chocolate  
galletas  
leche  
pañales  
zumo
```

```
$ cat lista.txt | sort -r | less  
...
```



# Los mil usos de "cat"

---

```
$ cat > lista.txt  
$ cat >> lista.txt  
$ cat lista.txt  
$ cat bloque1 bloque2 bloque3 > bloques  
$ cat /dev/zero > /dev/sdd1  
$ cat /dev/cdrom > imagen.iso  
$ cat sound_of_silence.wav > /dev/dsp  
$ cat /dev/urandom > /dev/dsp  
$ cat lista.txt | tr A-Z a-z
```

# Contenidos

---

- Generalidades
- Redirección y tuberías
- **Gestión de archivos**
- Algunos comandos útiles
- Aplicaciones de terminal

## Gestión de archivos

---

- Carpeta de trabajo
  - pwd
- Ruta absoluta: respecto a la carpeta raíz
  - Comienza con "/"
- Ruta relativa: respecto a la carpeta de trabajo (\$PWD)
  - Comienza sin "/"
  - <ruta> = \$PWD/<ruta-relativa>
- Abreviaturas:
  - '.' carpeta actual
  - '..' carpeta superior
  - '~' carpeta del usuario
  - '~user' carpeta del usuario 'user'

# Gestión de archivos

---

```
$ pwd
/home/usuario
$ ls /etc/fstab
/etc/fstab
$ cd /etc
$ ls fstab
fstab
$ cd
$ ls ../../etc/fstab
../../etc/fstab
$ ls fstab
ls: no se puede acceder a fstab: No existe el fichero ó directorio
```

# Gestión de archivos

## ls

---

- Para listar los ficheros contenidos en una carpeta (directorio) se usa el comando 'ls' (list).
- Si no se le proporciona argumentos muestra contenido de la carpeta actual.
- La opción -a lista también los archivos ocultos.
- La opción -l ofrece información adicional.
- No confundir con "sl"

```
$ ls -la
...
```

# Gestión de archivos

## cd

---

- El comando 'cd' (*change directory*) permite cambiar la carpeta actual.

```
$ cd Documentos
$ cd Documentos/asignaturas/ced/tema01
$ cd /etc
$ cd ../tema02/ejercicios
$ cd -
$ cd ~jose/Imágenes
$ cd ~
$ cd $HOME
```

# Gestión de archivos

## cp, mv

---

- El comando 'cp' (*copy*) permite copiar archivos y carpetas
- El comando 'mv' (*move*) permite mover archivos y carpetas.
  - Si el destino es una carpeta: copia/mueve el archivo a la carpeta con el mismo nombre.
  - Si el destino no es una carpeta: copia el archivo cambiando el nombre (isi el destino existe es destruido!)

```
$ cp carta.txt documentos/
$ cp video.ogv Vídeos/vacaviones.ogv
$ cp -r practicas-pgpi archivo/curso_2014-15/
$ mv practicas-pgpi archivo/curso_2014-15/
$ mv carta.txt carta_ana.txt
```

# Gestión de archivos

## rm, rmdir, mkdir

---

- 'rm' (*remove*) borra archivos y carpetas.
- 'rmdir' borra carpetas.
- 'mkdir' (*make directory*) crea subcarpetas.

```
$ rm carta.txt
$ rm -r documentos/asignaturas/ced
$ rmdir Vídeos/vacaciones.old
$ mkdir copias_seguridad
$ mkdir -p documentos/asignaturas/ced
```

## Contenidos

---

- Generalidades
- Redirección y tuberías
- Gestión de archivos
- **Algunos comandos útiles**
- Aplicaciones de terminal

# Comandos útiles

---

- Principios de diseño
  - Un comando para cada tarea
  - Una tarea para cada comando
  - Todo comando lee/escribe en la entrada/salida estándar de forma predeterminada
  - Se facilita la interconexión de comandos
  - Si todo va bien, no se informa
  - El usuario es inteligente: (casi)nunca se pide confirmación
  - El usuario tiene prisa: nombres cortos
  - El usuario sabe mecanografía
- Do you speak UNIX?

# More or less

---

- more, less, hexdump (hd), strings
  - iii 'q' para salir !!!

```
$ more lista.txt
...

$ less lista.txt
...

$ find . -name '*.png' | less
...

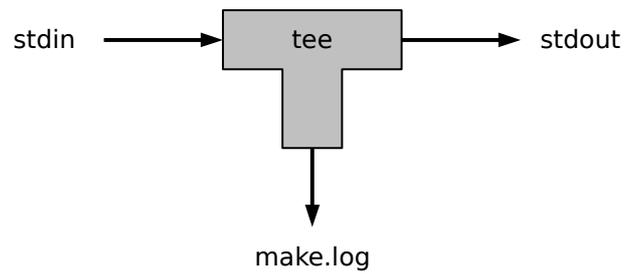
$ hd /bin/ls | less
...

$ strings /bin/ls | less
```

# Tee

---

```
$ make | tee make.log  
...
```



# Buscar

---

- find, locate

```
$ find . -name '*.txt' -a -newer lista.txt  
...  
$ locate '*.wav' | less  
...
```

# Comparar

---

- cmp, diff, patch

```
$ cmp lista1.txt lista2.txt
...

$ diff lista1.txt lista2.txt
...

$ diff -u lista.txt lista2.txt > diff2.txt
$ diff -u lista.txt lista3.txt > diff3.txt
...
$ patch lista.txt diff2.txt
...
$ patch lista.txt diff3.txt
...
$ cat diff2.txt diff3.txt | patch lista.txt
```

# Renombrar

---

- rename, ...

```
$ rename 's/JPEG$/jpg/' *.JPEG
...
```

# Comprimir/descomprimir

---

- gzip, gunzip, zcat, zless

```
$ gzip lista.txt

$ gzip -l lista.txt.gz
      compressed      uncompressed  ratio uncompressed_name
          165             197   30.5% lista.txt

$ zcat lista.txt.gz
leche
zumo
mantequilla
...

$ zless lista.txt.gz
...

$ gunzip lista.txt.gz
```

# Comprimir/descomprimir

---

- tar (tape archiver): empaqueta un conjunto de archivos en un archivo único (archivador) manteniendo la información de propiedades y permisos de cada uno.
  - Opción “z”: comprime con gzip
  - Opción “c”: crea un nuevo archivo
  - Opción “x”: extrae archivos

```
$ tar zcvf documentos.tar.gz docs
...

$ tar zxvf documentos.tar.gz
...
```

# Discos

---

```
$ mount
...

$ df -h
...

$ du Documentos
...

$ du -sh /home/paco
...
```

# Información del sistema

---

- whoami, who, w, uptime, uname

```
$ whoami
jjchico

$ who
jjchico :0          2014-11-18 09:35 (:0)
jjchico pts/6      2014-11-18 09:35 (:0)

jjchico@valhalla:~ $ w
12:08:01 up 3 days, 16:09, 5 users, load average: 0,08, 0,13, 0,14
USUARIO TTY      DE          LOGIN@  IDLE   JCPU   PCPU WHAT
jjchico :0       :0          09:35  ?xdm? 27:52  0.13s init --user
jjchico pts/6    :0          09:35  2:31m 0.07s 0.07s bash

jjchico@valhalla:~ $ uptime
12:08:04 up 3 days, 16:09, 5 users, load average: 0,08, 0,13, 0,14

$ uname -a
Linux valhalla 3.13.0-39-generic #66-Ubuntu SMP Tue Oct 28 13:30:27 UTC
2014 x86_64 x86_64 x86_64 GNU/Linux
```

## Cronometrar: time

---

```
pi@raspbmc:~$ time find /media/bigdisk/Películas -name "*.avi" | wc -l
103

real    0m2.098s
user    0m0.030s
sys     0m0.070s
```

## Monitorizar: watch

---

```
$ watch w
...

$ watch df -h
...

$ watch 'df -h | grep home'
...

$ watch
...

$ watch cat /proc/interrupts
...

$ watch -n 0 "cat /proc/interrupts | grep '^ *[0-9]*:'"
...
```

# Filtros

---

- sort
- grep
- tr
- uniq
- head / tail
- sed (avanzado)
- awk (avanzado)

```
$ wget http://www.gnu.org/licenses/gpl.txt
...
$ cat gpl.txt |
tr -sc A-Za-z '\012' |
sort |
uniq -c |
sort -nr |
head
 309 the
 210 of
 177 to
 171 a
 138 or
 106 you
  97 work
  91 that
  91 and
  76 in
```

# Filtros

## Expresiones regulares

---

- Especifican patrones para localizar texto en archivos.
- Usadas por filtros como grep, sed, awk, ...
- Expresiones
  - '.' cualquier carácter
  - '[...], [^...]' uno de varios caracteres o rangos
  - '\*' cero o más caracteres iguales al anterior
  - '+' uno o más caracteres iguales al anterior
  - '^' principio de la línea
  - '\$' fin de la línea
  - '\' operación especial
  - '\(...\)' establece marca
  - '\n' referencia a n-ésima marca anterior

# Filtros

## Expresiones regulares

---

```
$ wget http://www.gnu.org/licenses/gpl.txt
...
$ grep '[Ss]oftware' gpl.txt | wc -l
26

$ grep '^([aeiou])\.*\1$' /usr/share/dict/spanish
aarónica
...
ozonómetro

$ grep '^([\^aeiou])\.*\1$' /usr/share/dict/spanish
baobab
...
zarevitz

$ grep '^t.t.r.$' /usr/share/dict/spanish
tataré
...
tutora
```

## Control de procesos

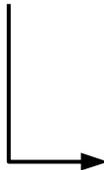
---

- Comando para el control de procesos
  - ps: listar procesos
  - nice: cambiar prioridad
  - kill/killall: enviar señales

# Administración: sudo

---

- En Ubuntu el acceso al root es “indirecto” por defecto
- Administradores: usuarios en grupo “sudo”
- Acceso: sudo, gksudo, ...



```
$ cat /etc/shadow
cat: /etc/shadow: Permiso denegado
jjchico@chalupa:~$ sudo cat /etc/shadow
[sudo] password for paco:
root:$6$kn8KsP2q$DVdBYsVX0WvGYvhf/:15826:0:99999:7:::
daemon*:14361:0:99999:7:::
...

$ sudo -s
[sudo] password for paco:
#
```

# Paquetes de software

---

- apt-get, apt-cache

```
$ sudo apt-get install iverilog gtkwave
...

$ sudo apt-get install git gitk git-gui
...

$ apt-cache search control versiones | less
...

$ apt-cache show gitk | less
...

$ apt-get remove gitk
...
```

# Administración

---

- adduser, addgroup: crear usuarios y grupos
- chown, chmod, chgrp: gestionar permisos
- mount/umount: conectar/desconectar sistemas de archivos.
- dmesg: muestra mensajes del kernel
- shutdown: detener/reiniciar el sistema

# dd: cat con esteroides

---

- Copia datos de una entrada a una salida
  - A menudo combinado con archivos de dispositivo

```
$ dd if=data.bin of=data_out.bin bs=1M count=10
...
$ dd if=/dev/zero of=zeros.bin bs=1k count=100
...
$ dd if=/dev/urandom of=gabrage.avi bs=1M count=10
...
$ dd if=/dev/cdrom of=mydvd.iso bs=4k
...
$ dd if=/dev/zero of=/dev/sdd bs=4k
...
```

# Fecha y hora

```
$ date
mar nov 18 11:20:09 CET 2014

$ ntpdate -q hora.rediris.es
...
18 Nov 11:22:50 ntpdate[7845]: ...

$ cal
    Noviembre 2014
su  lu  ma  mi  ju  vi  sa
                1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30
```

```
$ ncal
    Noviembre 2014
lu   3 10 17 24
ma   4 11 18 25
mi   5 12 19 26
ju   6 13 20 27
vi   7 14 21 28
sá  1  8 15 22 29
do   2  9 16 23 30

$ ncal -C feb 1981
    Febrero 1981
lu  ma  mi  ju  vi  sa  do
                1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28

$ ncal -C 1971 | lpr
```

# Calcular

- bc, python

```
$ bc -l
3*5
15
pi = 4*a(1)
pi
3.14159265358979323844
radio = 7
perimetro = 2 * pi * radio
area = pi * radio^2
perimetro
43.98229715025710533816
area
153.93804002589986868356
quit
```

```
$ python
Python 2.7.6 ...
>>> import math
>>> 3*5
15
>>> pi = math.pi
>>> pi
3.141592653589793
>>> radio = 7
>>> perimetro = 2 * pi * radio
>>> area = pi * radio**2
>>> perimetro
43.982297150257104
>>> area
153.93804002589985
>>> Ctrl-D
```

# Conexión remota

---

- ssh, scp, sftp, rsync, wget

```
desktop $ ssh server.home.net
...
server $ scp carta.txt paco@lin1.home.net:Documentos/cartas

server $ wget http://www.gnu.org/licenses/gpl.txt
server $ mv gpl.txt Documentos/licencias
server $ rsync -c -av Documentos backup.home.net:

$ exit
logout
Connection to server.home.net closed.

desktop $ ssh -X office.home.net
...
office $ gnome-calculator &
office $ lowriter
```

# Contenidos

---

- Generalidades
- Redirección y tuberías
- Gestión de archivos
- Algunos comandos útiles
- **Aplicaciones de terminal**

# Aplicaciones de terminal

---

- Editores de texto
  - Editor 'nano':
    - Intuitivo y fácil de utilizar
  - Editor 'vi':
    - Uso muy extendido en sistema UNIX/Linux.
    - Tiene varios modos: modo comando y modo de inserción y modo de línea.
    - Si no se conoce es difícil de utilizar.
    - Potente y rápido

## Vi esencial

---

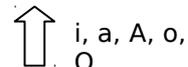
```
$ vi lista.txt  
...
```



```
galletas  
leche  
~  
~  
~  
~  
~  
~  
~  
"lista.txt" 2L, 15C 1,1
```

### MODO INSERCIÓN

Añadir texto  
Borrar: Retroceso, Supr  
Movimiento: cursores, ...



### MODO VISUAL

Movimiento: 0, \$, h, j, k, l, ESP, b, ...  
Borrar: x, dd, dw, d+mov, ...  
Buscar: /, n, N



### MODO COMANDOS

Salvar: w, wq, ...  
Salir: wq, q!

# Aplicaciones de terminal

---

- top, htop
  - Control de procesos e información del sistema
- lynx
  - Navegador web
- mutt, mail, mpack
  - Cliente de correo electrónico

# Grabar una sesión de terminal (script)

---

```
$ script
Script iniciado; el fichero es typescript
$ echo "patatas" >> lista.txt
$ echo "tomates" >> lista.txt
$ cat lista.txt
patatas
$ exit
Script terminado; el fichero es typescript

$ cat typescript
Script iniciado (dom 16 nov 2014 17:54:28 CET)
$ echo "patatas" >> lista.txt
$ echo "tomates" >> lista.txt
$ cat lista.txt
patatas
tomates
$ exit
Script terminado (dom 16 nov 2014 17:55:21 CET)
```

# Gestor de terminal (screen)

---

- Gestiona sesiones de terminal “virtuales”
- Posibilidad de desconectar y reconectar a sesiones desde distintos terminales.
- Posibilidad de tener varias sesiones en un mismo terminal
- Ejemplo: reconectar remotamente (ssh) a procesos largos.

```
# Inicia sesión
screen
# Sale de la sesión
exit
# Lista sesiones
screen -ls
# Desconecta una sesión
screen -d
# Conecta a una sesión
screen [-d] -r
```

```
# Secuencias de control
C-a d # desconecta de la sesión
C-a c # crea nueva sesión
C-a w # lista sesiones
C-a N # cambia a sesión N
```

# Gestor de terminal (screen)

---

```
$ screen
$ sudo apt-get dist-upgrade
[sudo] password for jjchico:
...
$
```

```
$ screen -d -r
$ sudo apt-get dist-upgrade
[sudo] password for jjchico:
...
...
...
```