

# Uso elemental de GNU/Linux

Alejandro Sztrajman

Métodos numéricos multidimensionales en fluidos  
Departamento de Física  
Facultad de Ciencias Exactas y Naturales  
Universidad de Buenos Aires

16 de agosto de 2011

## Agenda

Navegación y  
manejo de  
archivos

Bash

Navegación

Atributos y ejecución  
de archivos

Manejo de archivos

Sistema de  
usuarios

# Agenda

- 1 Navegación por el sistema de archivos
- 2 Manejo de archivos
- 3 Sistema de usuarios

## Agenda

Navegación y  
manejo de  
archivos

Bash

Navegación

Atributos y ejecución  
de archivos

Manejo de archivos

Sistema de  
usuarios

# Agenda

- 1 Navegación por el sistema de archivos
- 2 Manejo de archivos
- 3 Sistema de usuarios

## Agenda

Navegación y  
manejo de  
archivos

Bash

Navegación

Atributos y ejecución  
de archivos

Manejo de archivos

Sistema de  
usuarios

# Agenda

- 1 Navegación por el sistema de archivos
- 2 Manejo de archivos
- 3 Sistema de usuarios

## Agenda

Navegación y  
manejo de  
archivos

Bash

Navegación

Atributos y ejecución  
de archivos

Manejo de archivos

Sistema de  
usuarios

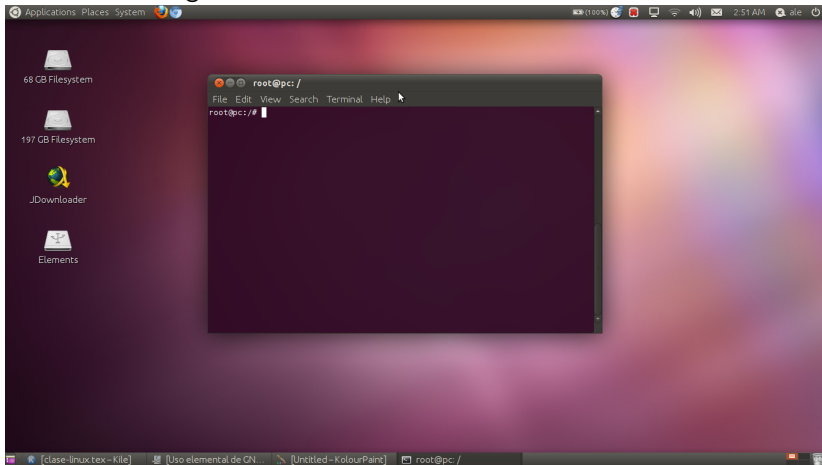
# Agenda

- 1 Navegación por el sistema de archivos
- 2 Manejo de archivos
- 3 Sistema de usuarios

Tambien vamos a ver  
diferencias con otros  
sistemas operativos



ALT + F2 → `gnome-terminal`



# Navegación por el sistema de archivos

**ls** Este comando nos muestra el contenido (archivos y directorios) del directorio actual. Si agregamos la opción **ls -lh** obtenemos además información sobre cada elemento.

Para movernos entre directorios tenemos el comando `cd <directorio>`.

`<directorio>` puede ser el nombre de un directorio, o una indicación relativa:

- `/` es el directorio principal del disco (conocido como raíz).
- `.` es el directorio actual.
- `..` es el directorio externo al actual.
- `~` es sinónimo del directorio home de nuestro usuario. Este es el directorio en el que van a estar alojados nuestros archivos, los que podemos modificar. En general, si mi usuario es alejandro, el directorio home está en `/home/alejandro/`.

# Navegación por el sistema de archivos

**ls** Este comando nos muestra el contenido (archivos y directorios) del directorio actual. Si agregamos la opción **ls -lh** obtenemos además información sobre cada elemento.

Para movernos entre directorios tenemos el comando **cd <directorio>**. **<directorio>** puede ser el nombre de un directorio, o una indicación relativa:

- / es el directorio principal del disco (conocido como raíz).
- . es el directorio actual.
- .. es el directorio externo al actual.
- ~ es sinónimo del directorio **home** de nuestro usuario. Este es el directorio en el que van a estar alojados nuestros archivos, los que podemos modificar. En general, si mi usuario es **alejandro**, el directorio **home** está en `/home/alejandro/`.



# Navegación por el sistema de archivos

**ls** Este comando nos muestra el contenido (archivos y directorios) del directorio actual. Si agregamos la opción **ls -lh** obtenemos además información sobre cada elemento.

Para movernos entre directorios tenemos el comando **cd <directorio>**.

**<directorio>** puede ser el nombre de un directorio, o una indicación relativa:

- / es el directorio principal del disco (conocido como raíz).
- . es el directorio actual.
- .. es el directorio externo al actual.
- ~ es sinónimo del directorio **home** de nuestro usuario. Este es el directorio en el que van a estar alojados nuestros archivos, los que podemos modificar. En general, si mi usuario es **alejandro**, el directorio **home** está en `/home/alejandro/`.

# Navegación por el sistema de archivos

**ls** Este comando nos muestra el contenido (archivos y directorios) del directorio actual. Si agregamos la opción **ls -lh** obtenemos además información sobre cada elemento.

Para movernos entre directorios tenemos el comando **cd <directorio>**.

**<directorio>** puede ser el nombre de un directorio, o una indicación relativa:

- / es el directorio principal del disco (conocido como raíz).
- . es el directorio actual.
- .. es el directorio externo al actual.
- ~ es sinónimo del directorio **home** de nuestro usuario. Este es el directorio en el que van a estar alojados nuestros archivos, los que podemos modificar. En general, si mi usuario es **alejandro**, el directorio **home** está en `/home/alejandro/`.

# Navegación por el sistema de archivos

**ls** Este comando nos muestra el contenido (archivos y directorios) del directorio actual. Si agregamos la opción **ls -lh** obtenemos además información sobre cada elemento.

Para movernos entre directorios tenemos el comando **cd <directorio>**.

**<directorio>** puede ser el nombre de un directorio, o una indicación relativa:

- / es el directorio principal del disco (conocido como raíz).
- . es el directorio actual.
- .. es el directorio externo al actual.
- ~ es sinónimo del directorio **home** de nuestro usuario. Este es el directorio en el que van a estar alojados nuestros archivos, los que podemos modificar. En general, si mi usuario es **alejandro**, el directorio **home** está en **/home/alejandro/**.

# Filesystem

El directorio raíz (/) contiene muchos directorios que pueden ser confusos.

/media

En este directorio se encuentra el contenido de los discos (en subdirectorios). En Linux, a diferencia de Windows, no existen las unidades (del tipo c:).

/etc

Aloja los archivos de configuración de la mayor parte del sistema.

/home

Adentro están los directorios personales de los usuarios.

# Filesystem

El directorio raíz (/) contiene muchos directorios que pueden ser confusos.

## /media

En este directorio se encuentra el contenido de los discos (en subdirectorios). En Linux, a diferencia de Windows, no existen las unidades (del tipo c:).

## /etc

Aloja los archivos de configuración de la mayor parte del sistema.

## /home

Adentro están los directorios personales de los usuarios.

# Filesystem

El directorio raíz (/) contiene muchos directorios que pueden ser confusos.

## /media

En este directorio se encuentra el contenido de los discos (en subdirectorios). En Linux, a diferencia de Windows, no existen las unidades (del tipo c:).

## /etc

Aloja los archivos de configuración de la mayor parte del sistema.

## /home

Adentro están los directorios personales de los usuarios.

# Filesystem

El directorio raíz (/) contiene muchos directorios que pueden ser confusos.

## /media

En este directorio se encuentra el contenido de los discos (en subdirectorios). En Linux, a diferencia de Windows, no existen las unidades (del tipo c:).

## /etc

Aloja los archivos de configuración de la mayor parte del sistema.

## /home

Adentro están los directorios personales de los usuarios.

# Atributos y ejecución de archivos

- En Linux, los archivos tienen tres atributos: leible (r), escribible (w) y ejecutable (x).
- Los atributos se pueden ver con el comando `ls -lh`.
- Con `ls` los ejecutables se ven en verde.
- Si queremos hacer que un archivo sea ejecutable, lo podemos hacer con el comando `chmod +x <archivo>`.
- A diferencia de Windows, los archivos ejecutables no tienen una extensión determinada (.exe por ejemplo). Los programas de Windows no funcionan en Linux. Pero con `wine` podemos correr algunos de ellos.
- Si queremos ejecutar un archivo que está en el directorio actual, usamos `./<nombre>`.
- Si estamos ejecutando un programa de ventana, es conveniente agregar un `&` al final de `<nombre>`, para seguir trabajando en la terminal.



# Atributos y ejecución de archivos

- En Linux, los archivos tienen tres atributos: leible (r), escribible (w) y ejecutable (x).
- Los atributos se pueden ver con el comando `ls -lh`.
- Con `ls` los ejecutables se ven en verde.
- Si queremos hacer que un archivo sea ejecutable, lo podemos hacer con el comando `chmod +x <archivo>`.
- A diferencia de Windows, los archivos ejecutables no tienen una extensión determinada (.exe por ejemplo). Los programas de Windows no funcionan en Linux. Pero con `wine` podemos correr algunos de ellos.
- Si queremos ejecutar un archivo que está en el directorio actual, usamos `./<nombre>`.
- Si estamos ejecutando un programa de ventana, es conveniente agregar un `&` al final de `<nombre>`, para seguir trabajando en la terminal.

# Atributos y ejecución de archivos

- En Linux, los archivos tienen tres atributos: leible (r), escribible (w) y ejecutable (x).
- Los atributos se pueden ver con el comando `ls -lh`.
- Con `ls` los ejecutables se ven en verde.
- Si queremos hacer que un archivo sea ejecutable, lo podemos hacer con el comando `chmod +x <archivo>`.
- A diferencia de Windows, los archivos ejecutables no tienen una extensión determinada (.exe por ejemplo). Los programas de Windows no funcionan en Linux. Pero con `wine` podemos correr algunos de ellos.
- Si queremos ejecutar un archivo que está en el directorio actual, usamos `./<nombre>`.
- Si estamos ejecutando un programa de ventana, es conveniente agregar un `&` al final de `<nombre>`, para seguir trabajando en la terminal.

# Atributos y ejecución de archivos

- En Linux, los archivos tienen tres atributos: leible (r), escribible (w) y ejecutable (x).
- Los atributos se pueden ver con el comando `ls -lh`.
- Con `ls` los ejecutables se ven en verde.
- Si queremos hacer que un archivo sea ejecutable, lo podemos hacer con el comando `chmod +x <archivo>`.
- A diferencia de Windows, los archivos ejecutables no tienen una extensión determinada (.exe por ejemplo). Los programas de Windows no funcionan en Linux. Pero con `wine` podemos correr algunos de ellos.
- Si queremos ejecutar un archivo que está en el directorio actual, usamos `./<nombre>`.
- Si estamos ejecutando un programa de ventana, es conveniente agregar un `&` al final de `<nombre>`, para seguir trabajando en la terminal.

# Atributos y ejecución de archivos

## Agenda

### Navegación y manejo de archivos

#### Bash

#### Navegación

#### Atributos y ejecución de archivos

#### Manejo de archivos

### Sistema de usuarios

- En Linux, los archivos tienen tres atributos: leible (r), escribible (w) y ejecutable (x).
- Los atributos se pueden ver con el comando `ls -lh`.
- Con `ls` los ejecutables se ven en verde.
- Si queremos hacer que un archivo sea ejecutable, lo podemos hacer con el comando `chmod +x <archivo>`.
- A diferencia de Windows, los archivos ejecutables no tienen una extensión determinada (.exe por ejemplo). Los programas de Windows no funcionan en Linux. Pero con `wine` podemos correr algunos de ellos.
- Si queremos ejecutar un archivo que está en el directorio actual, usamos `./<nombre>`.
- Si estamos ejecutando un programa de ventana, es conveniente agregar un `&` al final de `<nombre>`, para seguir trabajando en la terminal.

# Atributos y ejecución de archivos

- En Linux, los archivos tienen tres atributos: leible (r), escribible (w) y ejecutable (x).
- Los atributos se pueden ver con el comando `ls -lh`.
- Con `ls` los ejecutables se ven en verde.
- Si queremos hacer que un archivo sea ejecutable, lo podemos hacer con el comando `chmod +x <archivo>`.
- A diferencia de Windows, los archivos ejecutables no tienen una extensión determinada (.exe por ejemplo). Los programas de Windows no funcionan en Linux. Pero con `wine` podemos correr algunos de ellos.
- Si queremos ejecutar un archivo que está en el directorio actual, usamos `./<nombre>`.
- Si estamos ejecutando un programa de ventana, es conveniente agregar un `&` al final de `<nombre>`, para seguir trabajando en la terminal.

# Atributos y ejecución de archivos

- En Linux, los archivos tienen tres atributos: leible (r), escribible (w) y ejecutable (x).
- Los atributos se pueden ver con el comando `ls -lh`.
- Con `ls` los ejecutables se ven en verde.
- Si queremos hacer que un archivo sea ejecutable, lo podemos hacer con el comando `chmod +x <archivo>`.
- A diferencia de Windows, los archivos ejecutables no tienen una extensión determinada (.exe por ejemplo). Los programas de Windows no funcionan en Linux. Pero con `wine` podemos correr algunos de ellos.
- Si queremos ejecutar un archivo que está en el directorio actual, usamos `./<nombre>`.
- Si estamos ejecutando un programa de ventana, es conveniente agregar un `&` al final de `<nombre>`, para seguir trabajando en la terminal.

# Manejo de archivos

**mkdir** <directorío>

crea el directorío <directorío>.

**cp** <origen> <destino>

copia el archivo <origen> al archivo o directorío <destino>.

**mv** <origen> <destino>

mueve el archivo <origen> al archivo o directorío <destino>.

**rm** <archivo>

borra el archivo <archivo> (guarda que no hay undelete).

**rmdir** <directorío>

borra el directorío <directorío> (pero tiene que estar vacío).

Si queremos borrar un directorío con todo su contenido, usamos **rm -r**  
<directorío>.

# Manejo de archivos

`mkdir <directorio>`

crea el directorio `<directorio>`.

`cp <origen> <destino>`

copia el archivo `<origen>` al archivo o directorio `<destino>`.

`mv <origen> <destino>`

mueve el archivo `<origen>` al archivo o directorio `<destino>`.

`rm <archivo>`

borra el archivo `<archivo>` (guarda que no hay undelete).

`rmdir <directorio>`

borra el directorio `<directorio>` (pero tiene que estar vacío).

Si queremos borrar un directorio con todo su contenido, usamos `rm -r <directorio>`.



# Manejo de archivos

**mkdir** <directorio>

crea el directorio <directorio>.

**cp** <origen> <destino>

copia el archivo <origen> al archivo o directorio <destino>.

**mv** <origen> <destino>

mueve el archivo <origen> al archivo o directorio <destino>.

**rm** <archivo>

borra el archivo <archivo> (guarda que no hay undelete).

**rmdir** <directorio>

borra el directorio <directorio> (pero tiene que estar vacío).

Si queremos borrar un directorio con todo su contenido, usamos **rm -r** <directorio>.

# Manejo de archivos

**mkdir** <directorio>

crea el directorio <directorio>.

**cp** <origen> <destino>

copia el archivo <origen> al archivo o directorio <destino>.

**mv** <origen> <destino>

mueve el archivo <origen> al archivo o directorio <destino>.

**rm** <archivo>

borra el archivo <archivo> (guarda que no hay undelete).

**rmdir** <directorio>

borra el directorio <directorio> (pero tiene que estar vacío).

Si queremos borrar un directorio con todo su contenido, usamos **rm -r** <directorio>.

# Manejo de archivos

**mkdir** <directorio>

crea el directorio <directorio>.

**cp** <origen> <destino>

copia el archivo <origen> al archivo o directorio <destino>.

**mv** <origen> <destino>

mueve el archivo <origen> al archivo o directorio <destino>.

**rm** <archivo>

borra el archivo <archivo> (guarda que no hay undelete).

**rmdir** <directorio>

borra el directorio <directorio> (pero tiene que estar vacío).

Si queremos borrar un directorio con todo su contenido, usamos **rm -r** <directorio>.

# Manejo de archivos

`mkdir <directorio>`

crea el directorio `<directorio>`.

`cp <origen> <destino>`

copia el archivo `<origen>` al archivo o directorio `<destino>`.

`mv <origen> <destino>`

mueve el archivo `<origen>` al archivo o directorio `<destino>`.

`rm <archivo>`

borra el archivo `<archivo>` (guarda que no hay undelete).

`rmdir <directorio>`

borra el directorio `<directorio>` (pero tiene que estar vacío).

Si queremos borrar un directorio con todo su contenido, usamos `rm -r <directorio>`.

# Sistema de usuarios

Linux funciona con un sistema estricto de usuarios.

- Los usuarios pertenecen a grupos de privilegios.
- `root` es el super user, tiene privilegios de administrador.
- Las restricciones sobre los permisos de los usuarios hicieron que los virus no prosperaran.
- No se debe usar `root` para las tareas cotidianas.
- Para cambiar el password de nuestra cuenta, podemos usar el comando `passwd`.

# Sistema de usuarios

Linux funciona con un sistema estricto de usuarios.

- Los usuarios pertenecen a grupos de privilegios.
- **root** es el super user, tiene privilegios de administrador.
- Las restricciones sobre los permisos de los usuarios hicieron que los virus no prosperaran.
- No se debe usar **root** para las tareas cotidianas.
- Para cambiar el password de nuestra cuenta, podemos usar el comando `passwd`.

# Sistema de usuarios

Linux funciona con un sistema estricto de usuarios.

- Los usuarios pertenecen a grupos de privilegios.
- **root** es el super user, tiene privilegios de administrador.
- Las restricciones sobre los permisos de los usuarios hicieron que los virus no prosperaran.
- No se debe usar **root** para las tareas cotidianas.
- Para cambiar el password de nuestra cuenta, podemos usar el comando `passwd`.

# Sistema de usuarios

Linux funciona con un sistema estricto de usuarios.

- Los usuarios pertenecen a grupos de privilegios.
- **root** es el super user, tiene privilegios de administrador.
- Las restricciones sobre los permisos de los usuarios hicieron que los virus no prosperaran.
- No se debe usar **root** para las tareas cotidianas.
- Para cambiar el password de nuestra cuenta, podemos usar el comando `passwd`.



# Sistema de usuarios

Linux funciona con un sistema estricto de usuarios.

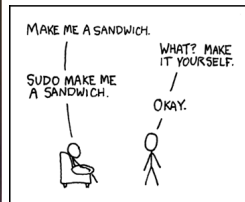
- Los usuarios pertenecen a grupos de privilegios.
- **root** es el super user, tiene privilegios de administrador.
- Las restricciones sobre los permisos de los usuarios hicieron que los virus no prosperaran.
- No se debe usar **root** para las tareas cotidianas.
- Para cambiar el password de nuestra cuenta, podemos usar el comando `passwd`.

Algunos comandos requieren permisos de administrador para ejecutarse. Para usarlos, no es necesario cambiar al usuario **root**. Lo más cómodo es anteponer el comando **sudo** al comando. (para que esto funcione, nuestro usuario debe tener permisos de sudo)

# Sudo

Algunos comandos requieren permisos de administrador para ejecutarse. Para usarlos, no es necesario cambiar al usuario **root**. Lo más cómodo es anteponer el comando **sudo** al comando. (para que esto funcione, nuestro usuario debe tener permisos de sudo)

```
ale@pc: /usr
File Edit View Search Terminal Help
ale@pc: /usr$ mkdir nuevodir
mkdir: cannot create directory `nuevodir': Permission denied
ale@pc: /usr$ sudo mkdir nuevodir
ale@pc: /usr$
```



# Finalmente

Cosas que se pueden bajar de la página web de la materia:

- Esta presentación, en su versión **presentación** o **handout** (formato para imprimir).
- La presentación anterior, hecha en prezi ([www.prezi.com](http://www.prezi.com)).
- El archivo `helloworld.cpp`.