

Planificación

Sesión 1 Introducción a la distibución de GNU
Familiarizarse con el escritorio GNU

Sesión 2 Creación de un USB con GNU
Gestión de archivos y carpetas

Sesión 3 Configuración de impresora y red
Compartir carpetas en la red

Sesión 4 Configuración básica de los repositorios. Actualizaciones

Creación de un USB con GNU

Sistemas Operativos Live

Creación de USB con `usb-creator.exe` en *windows*

Creación de USB con `usb-creator` en *ubuntu*

Creación de USB con `unetbootin`

Actividades

Proceso de arranque de un ordenador personal

- ▶ Después de realizar el Power On Self Test, la BIOS lee el primer sector del dispositivo que esté configurado como de arranque y le transfirere el control
- ▶ Normalmente está configurado como dispositivo de arranque un disco duro: El primer sector del disco (Master Boot Record MBR) lee el primer sector de la partición marcada como activa y le transfiere el control ... y así arranca el S.O. que está en la *partición activa*
- ▶ Si queremos instalar un S.O. a partir de un CD de instalación debemos configurar el ordenador para que arranque del CD, entrando en el *setup* de la BIOS o mediante la combinacion de teclas adecuada (normalmente nos informa de dicha combinación al arrancar).
- ▶ Si queremos instalar un S.O. a partir de un USB de instalación debemos configurar el ordenador para que arranque del USB
- ▶ Podemos usar esta característica para ejecutar un *S.O. live*

Sistemas Operativos live

- ▶ Llamamos *live* a los S.O. que se ejecutan en nuestro ordenador sin necesidad de instalación previa (no se escribe nada en el disco)
- ▶ Creamos el CD/DVD o USB live del S.O. y configuramos el dispositivo de arranque adecuado
- ▶ Con un *S.O. live* podemos familiarizarnos con su uso y capacidades sin necesidad de hacer ningun cambio permanente en nuestro ordenador
- ▶ Con un *S.O. live* podemos acceder a nuestros archivos en caso de que el S.O. que había instalado tenga algún problema y no pueda arrancar

Sistemas Operativos live

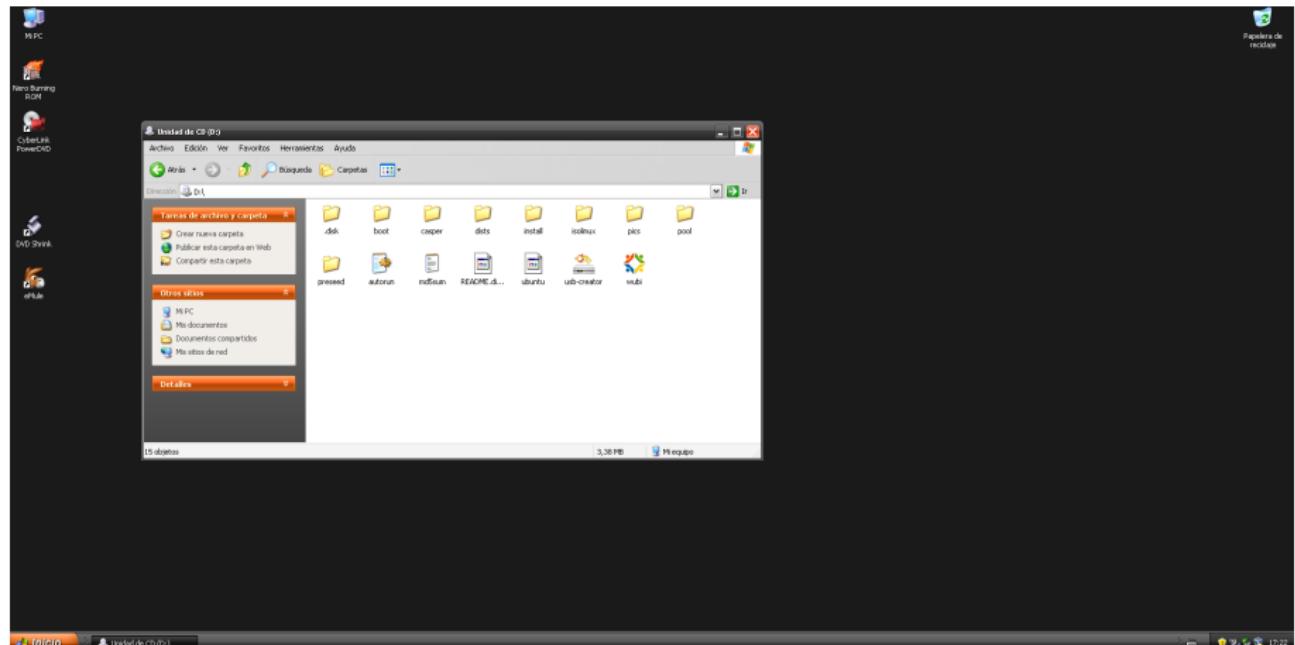
- ▶ Si el sistema *live* utilizado está basado en CD/DVD, lo que hagamos se pierde al apagar la máquina (a no ser que lo hayamos salvado específicamente en otro medio)
- ▶ Algunos sistemas *live* en USB permiten crear una zona del USB donde se guardarán nuestros archivos, teniendo de esta manera un S.O. completamente operacional en un USB
- ▶ Aunque existen varios sistemas linux *live*, nos centraremos en la creación de un USB live de la distribución de *ubuntu*
- ▶ Veremos tres métodos de creación del USB con GNU linux
 1. Desde *windows* con el programa `usb-creator.exe` que viene en el CD de instalación de *ubuntu*
 2. Desde *ubuntu* con el programa `usb-creator`
 3. Desde cualquier distribución de linux con el programa `unetbootin`

Creación del USB con *usb-creator.exe*

- ▶ Introducimos en *windows* el CD de instalación de *ubuntu*
- ▶ Abrimos la aplicación *usb-creator*. En donde aparecen las siguientes opciones
 - ▶ CD o imagen .iso de nuestra distribución
 - ▶ Unidad USB de destino
 - ▶ Espacio que queremos para nuestros ficheros

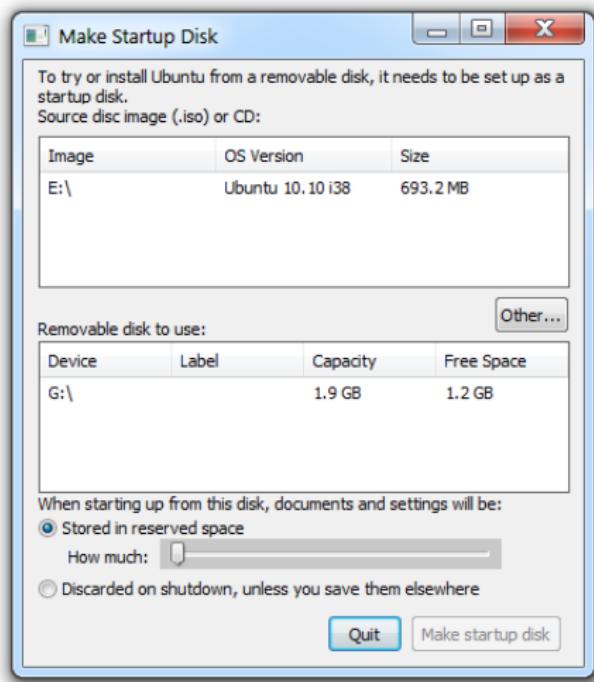
Creación del USB con usb-creator.exe

- ▶ Vista del CD instalación de *ubuntu* en un equipo *windows*



Creación del USB con usb-creator.exe

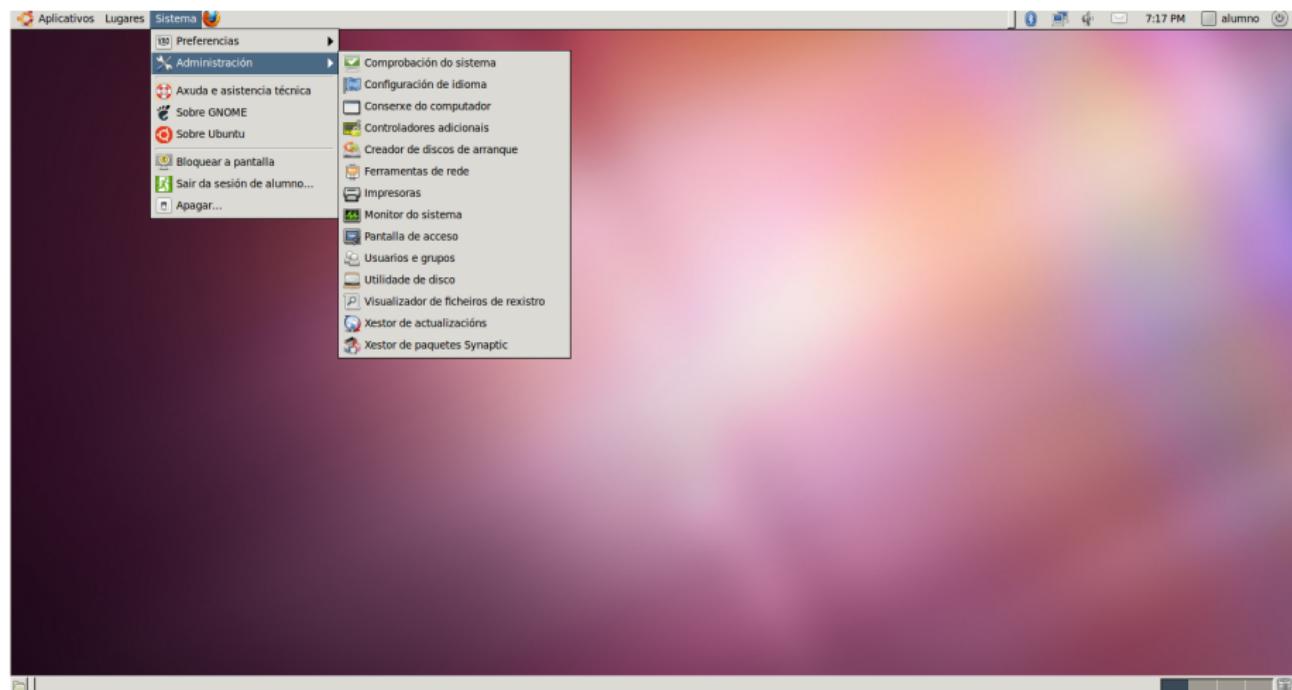
► Aplicación usb-creator.exe



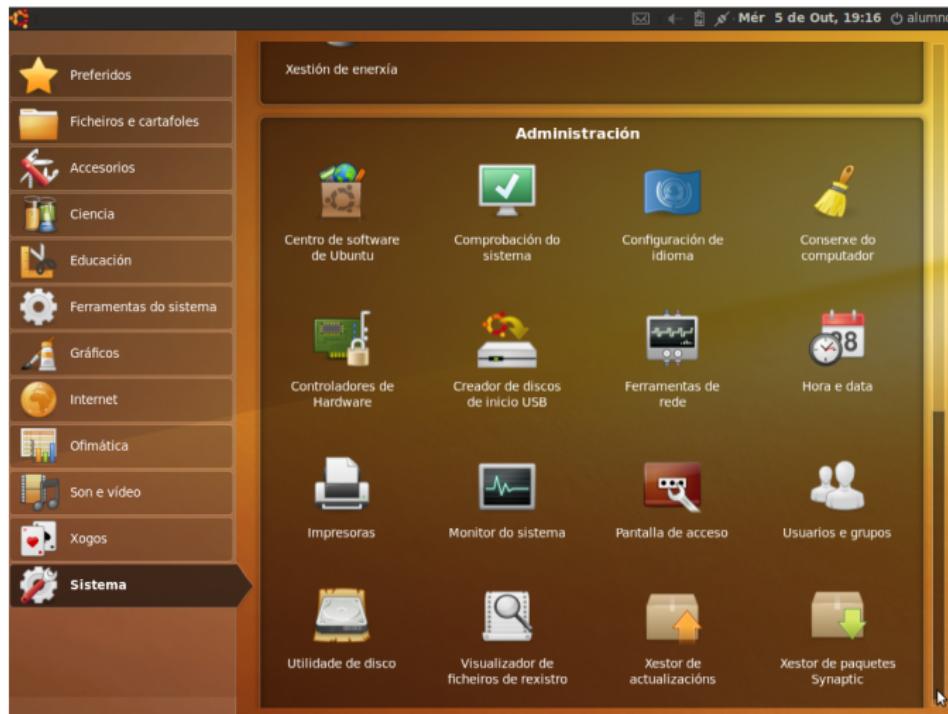
Creación del USB con *usb-creator* en *ubuntu*

- ▶ Desde un sistema *ubuntu* (*live* o instalado) podemos lanzar la aplicación *usb-creator* de cuaquiera de las siguientes maneras
 - ▶ Tecleando *usb-creator-gtk* en un terminal
 - ▶ En el escritorio *gnome*:
 - ▶ sistema-->administración-->creador discos de arranque
 - ▶ En el escritorio UNR
 - ▶ sistema-->creador discos de inicio USB
- ▶ La aplicación a la que se accede es la misma que en *windows* y de nuevo podemos seleccionar origen (CD o imagen iso), unidad USB de destino y espacio para los ficheros en la instalación *live* USB

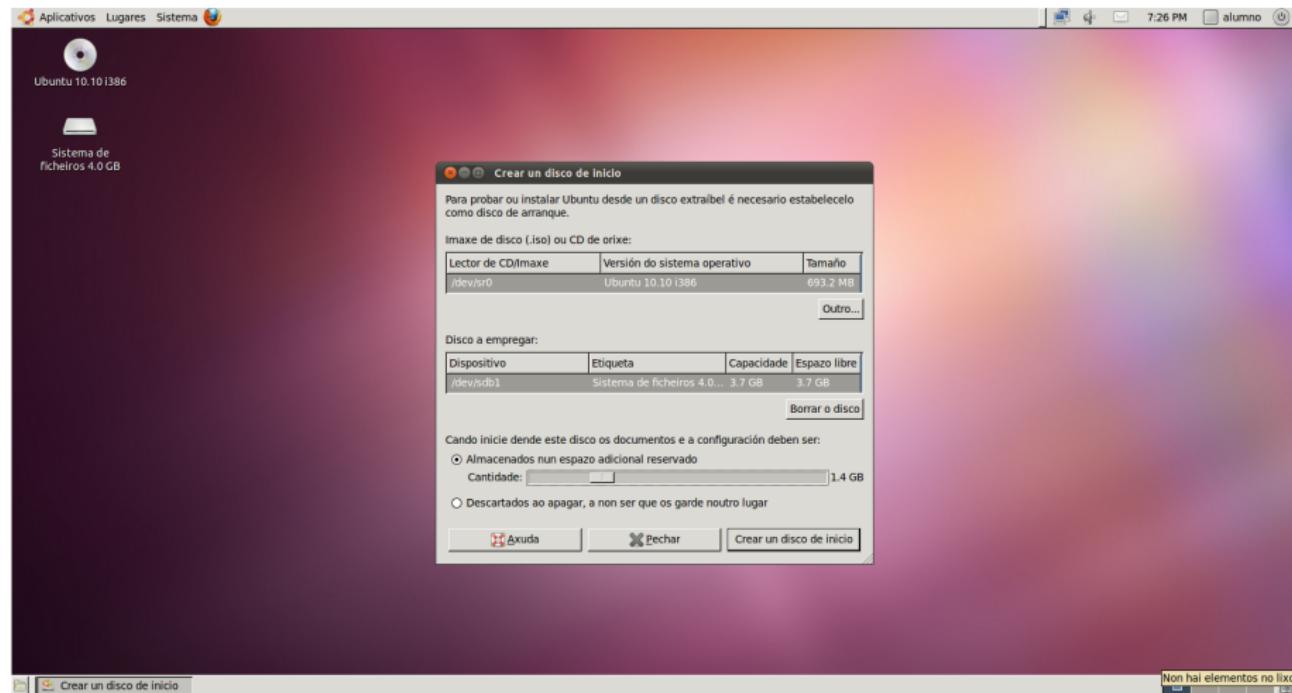
usb-creator en gnome



usb-creator en UNR



aplicación *usb-creator*



Creación del USB con unetbootin

- ▶ Es necesario tener instalada la aplicación unetbootin
- ▶ La aplicación unetbootin está disponible para muchas distribuciones de linux (existe tambien para windows) y permite crear USB de arranque de muchas distribuciones de linux (así como de otros S.O.)
- ▶ Podemos instalarla con el sistema de paquetes de nuestra distribución o desde <http://unetbootin.sourceforge.net/>
- ▶ Esta aplicación nos permite seleccionar
 - ▶ Qué distribución y que versión queremos instalar en el USB
 - ▶ Cual es la unidad USB de destino
 - ▶ En caso de crear un USB de ubuntu, tamaño para los ficheros personales en nuestro USB

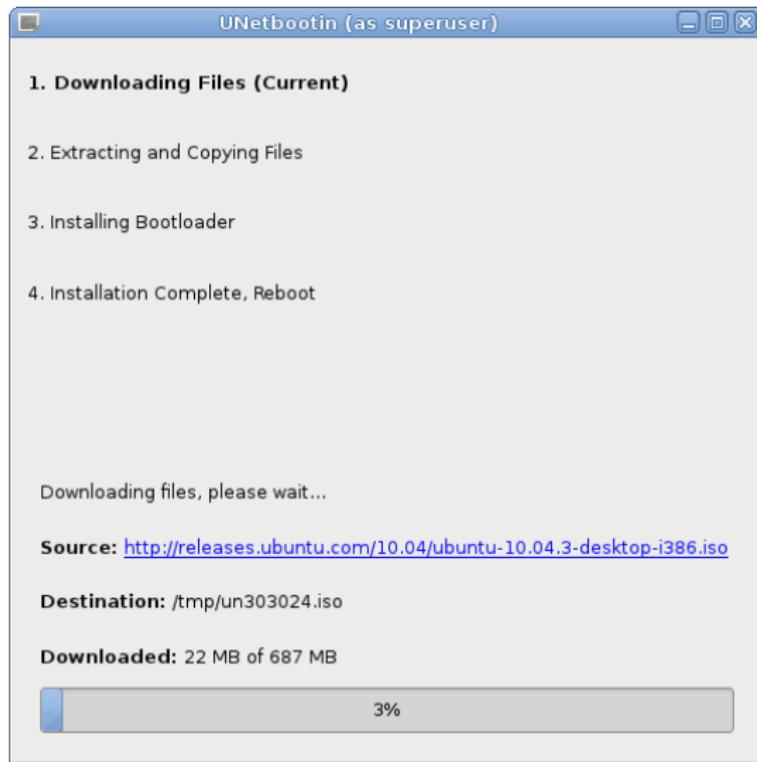
Creación del USB con unetbootin

- ▶ La aplicación se baja la imagen correspondiente de internet y crea el USB
- ▶ Si no optamos por una de las distribuciones que facilita unetbootin, o si no disponemos de conexión a internet (o no es lo suficientemente rápida) podemos suministrarle nosotros la imagen iso correspondiente
- ▶ Podemos arrancar la aplicación tecleando `unetbootin` en el terminal o desde `aplicaciones-->herramientas_del_sistema`
- ▶ Hay que tener en cuenta que es probable que necesitemos privilegios para ejecutar alguna de estas herramientas de creación de usb (puede pedirnos la contraseña del root o la contraseña de un usuario del grupo adm según el sistema donde estemos)

Creación del USB con unetbootin



Creación del USB con unetbootin



Actividades

- ▶ Realizar un USB de Ubuntu 10.04 con la aplicación `unetbootin`.
- ▶ Hacer que en el USB queden 300 MegaBytes para los archivos de usuario
- ▶ Cargar el S.O. desde dicho USB

Gestión de archivos y carpetas

Permisos

Acceso a archivos y carpetas

Archivos ocultos

Copiar, mover y borrar archivos y carpetas

Creación de archivos y carpetas

Cambio de permisos

Consideraciones sobre el uso del terminal

Unidades Extraibles

Actividades

Archivos y carpetas

- ▶ El sistema de archivos en linux se estructura como un árbol (grafo)
- ▶ En el tronco tenemos la carpeta (o directorio) /, del que cuelga el resto de los ficheros y carpetas del sistema
- ▶ El símbolo “/” se utiliza para separar los nombres de las carpetas (como “\” en los sistemas windows)
- ▶ Cada usuario tiene una carpeta personal que se encuentra dentro de la carpeta *home* (dentro de la carpeta raíz: /home). Así, los ficheros y carpetas del usuario *alumno* están en (/home/alumno)

Permisos de los archivos y carpetas

- ▶ Cada fichero (o carpeta) en el sistema tiene un usuario propietario y un grupo.
- ▶ Además tiene unos permisos que indican lo que pueden hacer con el fichero su propietario, el grupo o el resto de los usuarios del sistema.
- ▶ Solamente el propietario puede cambiar los permisos
- ▶ Para el usuario *root* no se tienen en cuenta los permisos de los ficheros
- ▶ El usuario *root* puede cambiar los permisos de cualquier fichero en el sistema

Permisos de los ficheros

- ▶ Los permisos de un fichero pueden ser

- r (*read*) Se puede leer
- w (*write*) Se puede escribir
- x (*eXecute*) Se puede ejecutar

Permisos de las carpetas

- ▶ El significado de los permisos para las carpetas es el siguiente
 - r Se pueden ver los contenidos de la carpeta (listar los ficheros que hay en ella)
 - w Se pueden modificar los contenidos de la carpeta (añadir o eliminar ficheros)
 - x Se puede acceder a los contenidos de la carpeta

Representación de los permisos

- ▶ Los permisos de un fichero se representan como una palabra de nueve letras con los tres permisos del propietario, los tres del grupo y los tres del resto de usuarios en el sistema (el guion “-” indica que no se tiene el permiso)
 - ▶ Si un fichero tiene permisos `rwxr-xr--` significa que el propietario puede leer escribir y ejecutar el fichero, los miembros del grupo pueden leer y ejecutar, y el resto de los usuarios sólo pueden leer
 - ▶ Si un fichero tiene permisos `rw-----` significa que solamente el propietario puede leer y escribir en el fichero

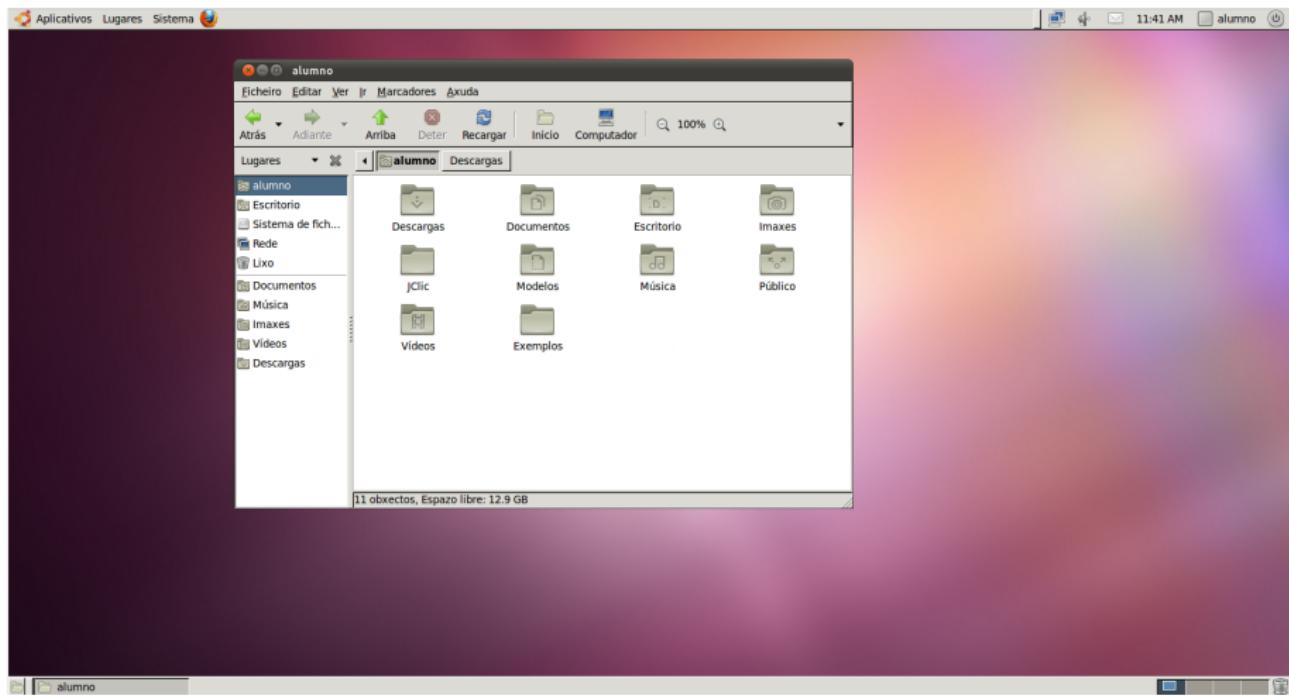
Representación de los permisos

- ▶ Algunos comandos del sistema entienden los permisos como un número de tres cifras: la cifra del propietario, la del grupo y la del resto.
- ▶ Cada cifra se obtiene sumando los permisos que tenga. A este efecto los valores de los permisos son: *read* es 4, *write* 2 y *execute* 1,
- ▶ Ejemplos
 - ▶ `rwxr-xr--` sería 754 (7 para el propietario, 4+2+1; 5 para el grupo 4+1 y 4 para el resto),
 - ▶ `rw-----` sería 600 (6 para el propietario, 4+2; 0 para el grupo y 0 para el resto)

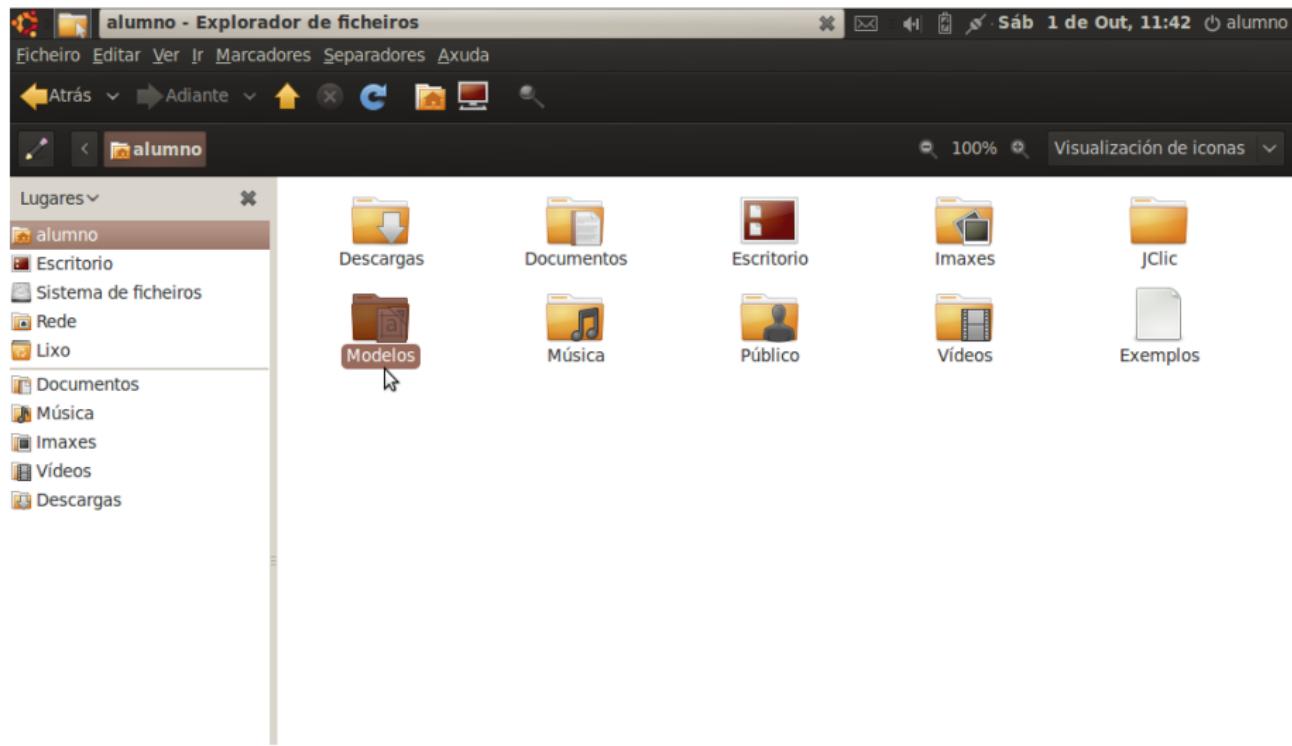
Acceso a archivos y carpetas

- ▶ En linux tenemos dos modos de acceder a los ficheros y a las carpetas
 - ▶ Usando la interfaz gráfica
 - ▶ ítem **ficheros y carpetas en UNR**
 - ▶ Menú lugares *gnome*
 - ▶ Usando el interfaz texto: terminal (común a los sistemas unix, incluido el *mac*)
 - ▶ aplicaciones-->accesorios-->terminal

Navegador carpetas gnome



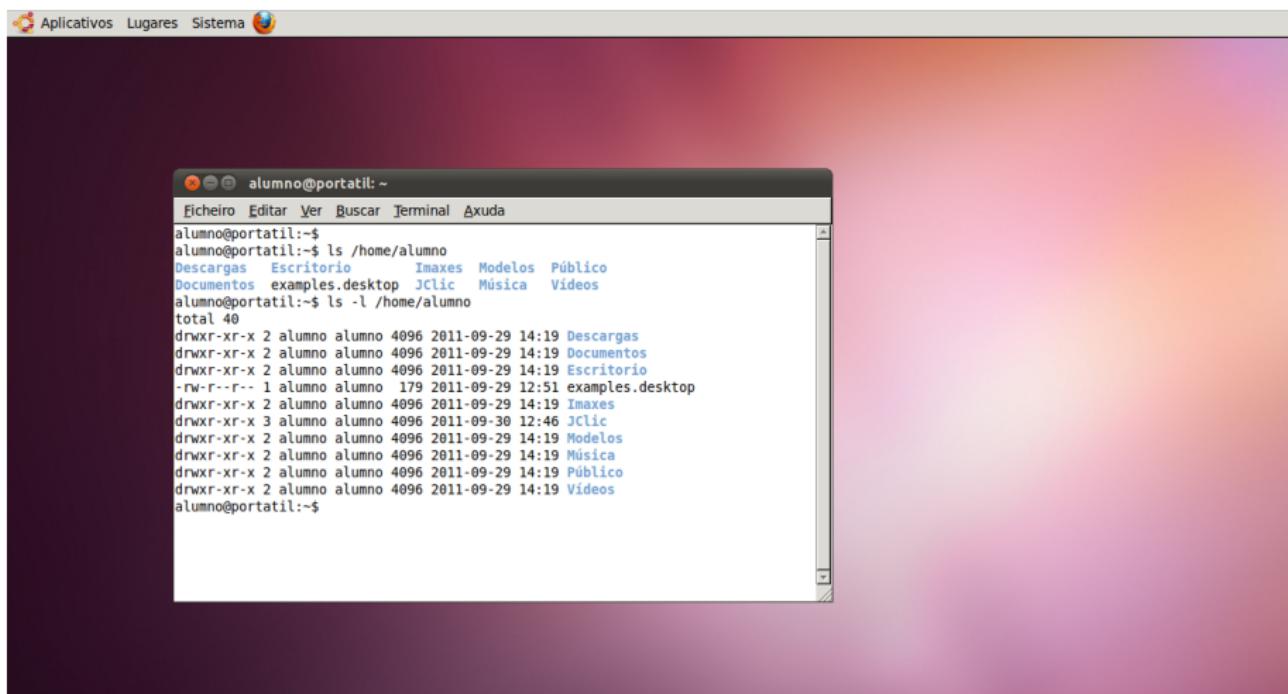
Navegador carpetas UNR



Uso de los equipos: Ver los contenidos de una carpeta

- ▶ En el navegador de archivos basta pinchar dos veces en la carpeta (aunque esto puede cambiarse para que sea un solo *click*)
- ▶ En la terminal el comando `ls` muestra los contenidos de una carpeta. (''ls nombre_carpeta''). Si no especificamos el nombre de la carpeta cuyos contenidos queremos listar, listará la *carpeta actual*
 - ▶ Para cambiar la *carpeta actual* usamos el comando `cd`. ''cd nueva_carpeta_actual'' o ''cd ..'' para volver a la carpeta anterior.
- ▶ ''ls -l nombre_carpeta'' realiza el listado largo, además de los nombres de los contenidos nos muestra el propietario, el grupo, el tamaño, los permisos, y la fecha de última modificación.
- ▶ Para mostrar esta información en el navegador de archivos pulsaríamos con el botón derecho y seleccionaríamos *propiedades*

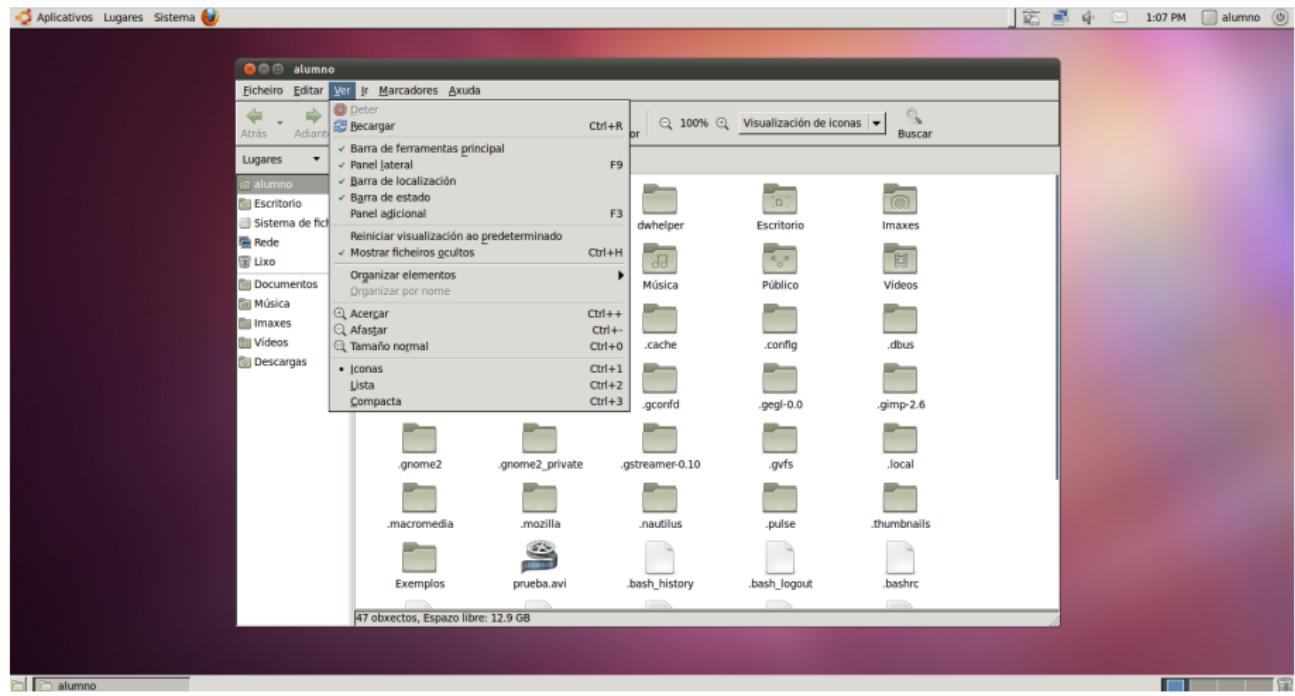
Uso de los equipos: Listado de archivos desde el terminal



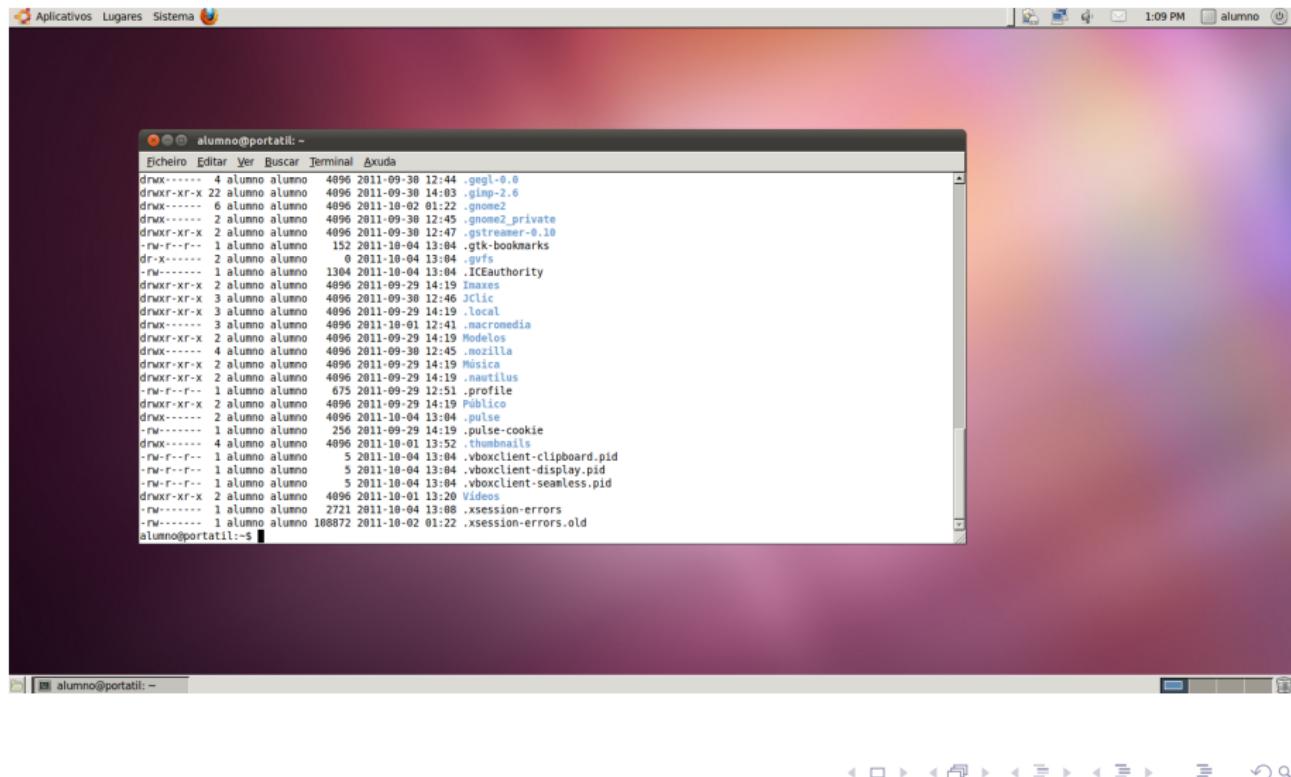
Archivos ocultos

- ▶ Los archivos cuyo nombre comienza por “.” no se muestran en los listados habituales (suele utilizarse esto para que los archivos de configuración y de opciones de los distintos programas no nos distraigan en los listados)
- ▶ Si queremos ver los archivos ocultos
 - ▶ Interfaz gráfica: opción “ver-->mostrar archivos ocultos” en en navegador de archivos
 - ▶ Terminal texto: comando ‘‘ls -a’’ o ‘‘ls -la’’

Archivos ocultos



Archivos ocultos desde el terminal



Copiar, mover y borrar

- ▶ Las opciones editar->copiar y editar->pegar, entre carpetas sirven para **copiar** archivos y carpetas de un lugar a otro
- ▶ Las opciones editar->cortar y editar->pegar, entre carpetas sirven para **mover** archivos y carpetas de un lugar a otro
- ▶ Arrastrar archivos y carpetas de una carpeta a otra tiene el siguiente comportamiento
 - ▶ Si las carpetas origen y destino están en la misma unidad de disco, *arrastrar mueve* dichos archivos de un lugar a otro. (Por ejemplo entre la carpeta *Escritorio* y la carpetas *Descargas*)
 - ▶ Si las carpetas origen y destino están en la distinta unidad de disco, *arrastrar copia* dichos archivos de un lugar a otro. (por ejemplo entre la carpeta *Descargas* y una unidad externa USB)

Copiar, mover y borrar

- ▶ Para eliminar marcamos los elementos a borrar, y luego con el botón derecho seleccionamos *mover a la papelera*
- ▶ Esto no elimina los archivos del disco (siguen ocupando espacio), simplemente los mueve a la papelera
- ▶ Si queremos eliminarlos definitivamente debemos vaciar la papelera

Copiar, mover y borrar desde el terminal

- ▶ Cuando operamos con archivos desde el terminal hay que especificar los archivos por su nombre.
- ▶ Si el nombre tiene *caracteres especiales* pueden ponerse entre comillas (‘ ‘) o precedido de símbolo \
- ▶ Su nombre puede expresarse
 - ▶ Completo
 - ▶ Con *caracteres comodín*

Copiar, mover y borrar desde el terminal: caracteres comodín

- ▶ **caracteres comodín** son los que nos permiten expresar abreviadamente el nombre de uno (o muchos) archivos. Los mas usados son
 - ▶ *: Representa cualquier carácter o grupo de caracteres. Ejemplos:
 - ▶ *.mp3 se refiere a todos los archivos cuyo nombre acabe en .mp3
 - ▶ a*k se refiere a todos los archivos cuyo nombre comienza por a y termina en k
 - ▶ ?: Representa cualquier carácter (solamente uno). Ejemplos:
 - ▶ ??? se refiere a todos los archivos cuyo nombre tiene tres caracteres
 - ▶ *..?.srt se refiere a todos los archivos cuyo nombre acaba en .srt y antes de .srt hay un punto y dos caracteres cualesquiera
 - ▶ [] Representa cualquier carácter de los que va entre los corchetes. Ejemplos:
 - ▶ document[12347].odt se refiere a los archivos document1.odt, document2.odt, document3.odt document4.odt y document7.odt
 - ▶ [a-z]*.avi se refiere a todos los archivos cuyo nombre comienza por una letra minúscula y acaba en .avi

Copiar, mover y borrar desde el terminal.

► Para copiar desde el terminal:

- ▶ `cp origen destino`. *origen* puede ser un archivo o un grupo de archivos, y *destino* puede ser un nombre de archivo o una carpeta
- ▶ ejemplo (desde un terminal)

```
cp viejo.avi nuevo.avi
```

copia el archivo *viejo.avi* y la copia se llama *nuevo.avi* (está en la misma carpeta)

```
cp *.mp3 /home/alumno/Descargas
```

copia todos los archivos cuyo nombre acaba en *.mp3* a la carpeta */home/alumno/Descargas*

- ▶ Si no especificamos nombre de carpeta en los archivos *origen*, se supone que están en la *carpeta actual* donde está el terminal
 - ▶ el comando `cd` permite cambiar de *carpeta actual* en un terminal
- ▶ Si queremos copiar una carpeta con sus contenidos (es decir, recursivamente) debemos emplear `cp -R`

Copiar archivos desde el terminal.

► Para mover desde el terminal:

- ▶ `mv origen destino`. *origen* puede ser un archivo o un grupo de archivos, y *destino* puede ser un nombre de archivo o una carpeta
- ▶ ejemplo (desde un terminal)

`mv viejo.avi nuevo.avi`

mueve el archivo *viejo.avi* a *nuevo.avi* (en realidad es un cambio de nombre)

`mv *.mp3 /home/alumno/Descargas`

mueve todos los archivos cuyo nombre acaba en *.mp3* a la carpeta */home/alumno/Descargas*

- ▶ Si no especificamos nombre de carpeta en los archivos origen, se supone que están en la *carpeta actual* donde está el terminal
 - ▶ el comando `cd` permite cambiar de *carpeta actual* en un terminal

Mover archivos desde el terminal

- ▶ Para borrar desde el terminal:

- ▶ `rm archivo_o_archivos_a_eliminar`
- ▶ ejemplo (desde un terminal)

`rm viejo.avi`

elimina el archivo `viejo.avi`

`rm /home/alumno/Descargas/*.mp3`

elimina todos los archivos cuyo nombre acaba en `.mp3` a la carpeta
`/home/alumno/Descargas`

- ▶ Si no especificamos nombre de carpeta en los archivos a eliminar, se supone que están en la *carpeta actual* donde está el terminal
 - ▶ el comando `cd` permite cambiar de *carpeta actual* en un terminal

Borrar archivos desde el terminal

- ▶ `rm` **elimina** los archivos del disco, no van a la papelera de reciclaje y por tanto no pueden ser recuperados
- ▶ dependiendo de los permisos de los archivos, `rm` puede pedir confirmación para borrar algunos archivos; `rm -f` los elimina sin pedir confirmación. Si queremos que pida confirmación antes de eliminar algo, usamos `rm -i`
- ▶ `rm` no borra carpetas, para borrar carpetas usamos `rmdir`
- ▶ `rmdir` solo elimina carpetas vacias. Si queremos borrar una carpeta que no está vacia `rm -r` y si queremos hacerlo sin que pida confirmación para nada (peligroso!) `rm -rf`

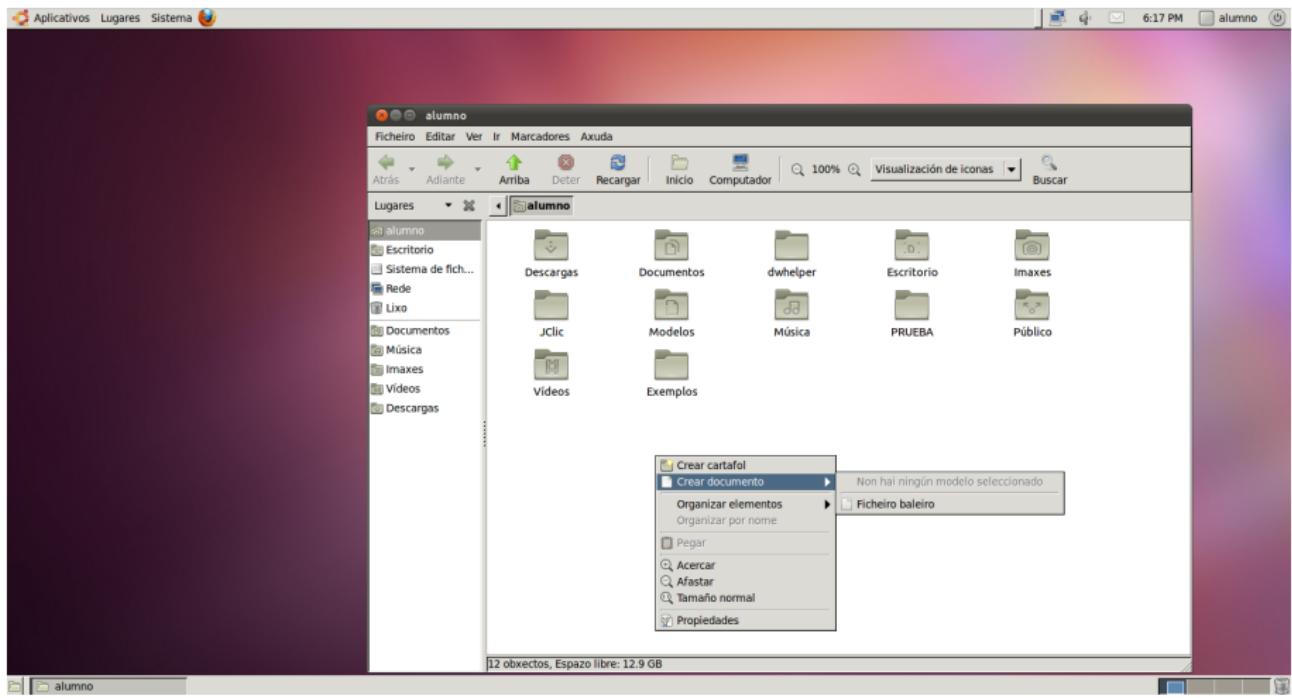
Creación de carpetas

- ▶ Para crear una carpeta
 - ▶ Desde el navegador de archivos
 - ▶ Pulsamos con el botón derecho sobre una zona en blanco
 - ▶ Seleccionamos Crear Carpeta y a continuación le ponemos el nombre
 - ▶ Desde el terminal
 - ▶ `mkdir nombre_nueva_carpeta`

Creación de ficheros

- ▶ Los archivos se crean normalmente desde las aplicaciones (p.e. al salvar los trabajos realizados) o al copiar otros archivos. No obstante, si queremos crear un archivo vacío
 - ▶ Desde el navegador de archivos
 - ▶ Pulsamos con el botón derecho sobre una zona en blanco
 - ▶ Seleccionamos Crear Documento-->Archivo vacío y a continuación le ponemos el nombre
 - ▶ Desde el terminal
 - ▶ `touch nombre_nuevo_archivo`

Creación archivos y carpetas

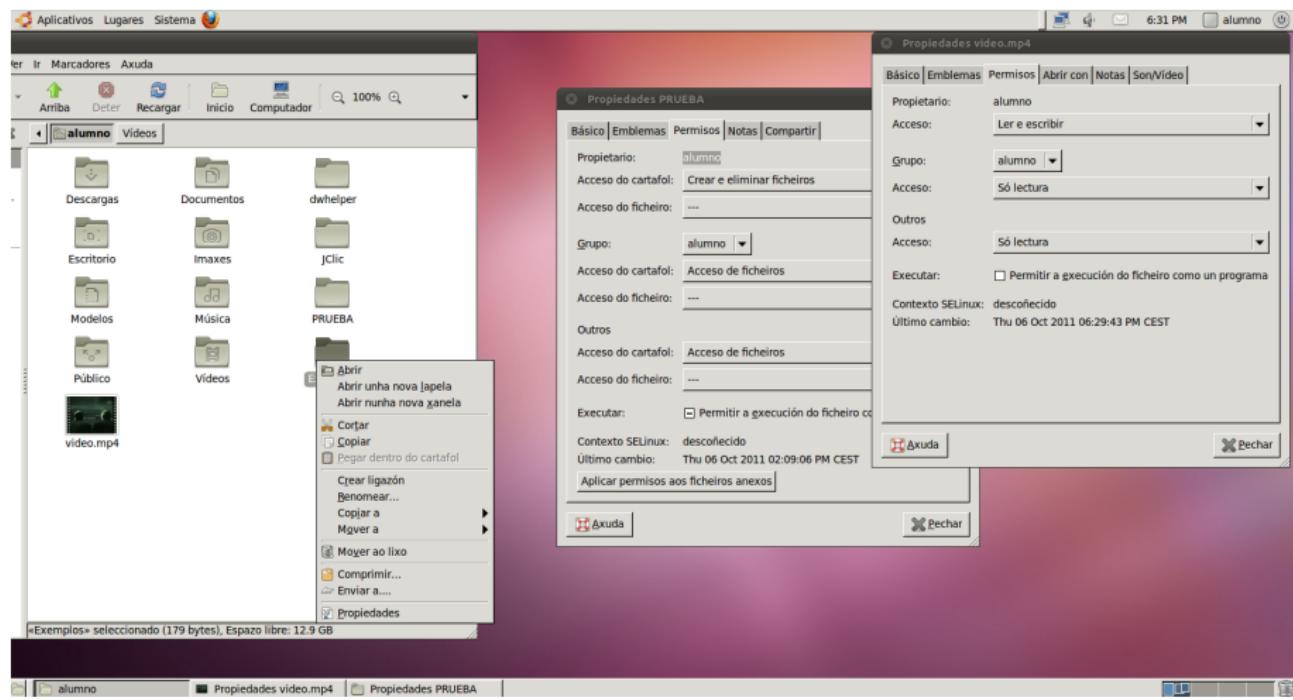


Cambiar permisos

- ▶ Desde el navegador de archivos

- ▶ Cuando pulsamos con el botón derecho sobre un archivo o carpeta en el navegador de archivos (tanto en el escritorio *gnome* como en el UNR nos aparece un menú. Seleccionando propiedades vamos al menú donde, a través de distintas pestañas, podemos cambiar los permisos
 - ▶ el nombre
 - ▶ el icono
 - ▶ los permisos
 - ▶ la aplicación usada para abrirlo
 - ▶ las opciones de compartición (para las carpetas)

Cambio de permisos



Cambiar permisos

- ▶ Desde el terminal
 - ▶ `chmod nuevos_permisos nombre_fichero.` *nuevos_permisos* es el número que representa los permisos y *nombre_fichero* puede contener caracteres comodín, de manera que se refiera a varios ficheros y/o carpetas.
 - ▶ Si queremos que el cambio de permisos afecte tambien a las subcarpetas de una carpeta podemos usar `chmod -R`. Ejemplos
 - ▶ `chmod 754 prueba.avi`
pone al archivo *prueba.avi* los permisos `rwxr-xr--`
 - ▶ `chmod -R 700 /home/alumno`
pone a la carpeta `/home/alumno` y a todo lo que ella contenga los permisos `rwx-----`

Consideraciones sobre el uso de terminal

- ▶ Todo lo que se puede hacer desde el interfaz gráfico puede realizarse desde el terminal.
- ▶ Los nombres de los archivos y carpetas pueden tener mayúsculas y minúsculas.
- ▶ Es sistema distingue entre mayúsculas y minúsculas. El fichero `Hola.txt` es distinto del fichero `hola.TXT`
- ▶ Los sistemas de archivos de *windows* no distinguen entre mayúsculas y minúsculas. Debemos tenerlo en cuenta si accedemos desde linux a un sistema de archivos de *windows* (p.e. una llave USB)

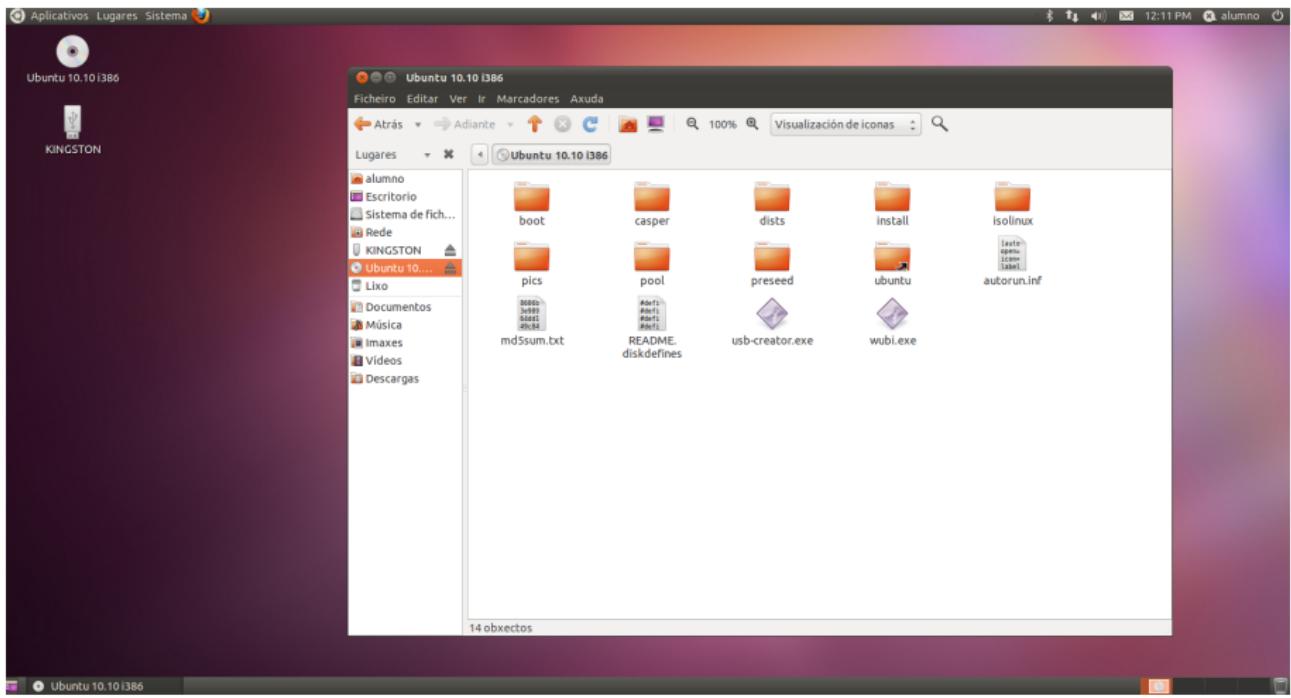
Consideraciones sobre el uso de terminal

- ▶ Los códigos de los caracteres que no son ASCII estándar (p.e. á, é, í, ó, ú ñ, ü ...) pueden ser distintos en sistemas *windows* y *linux* por lo que no es una buena idea utilizarlos para nombres de archivos si pensamos intercambiarlos entre sistemas
- ▶ La terminal dispone de ayuda en línea sobre los comandos. El comando para obtener ayuda es `man`. por ejemplo '`man ls`' nos informa de como se utiliza el comando `ls`
- ▶ '`man -k palabra`' nos proporciona una lista de comandos relacionados con *palabra*
- ▶ En la nomenclatura del terminal, una carpeta se denomina directorio (*directory* en inglés)

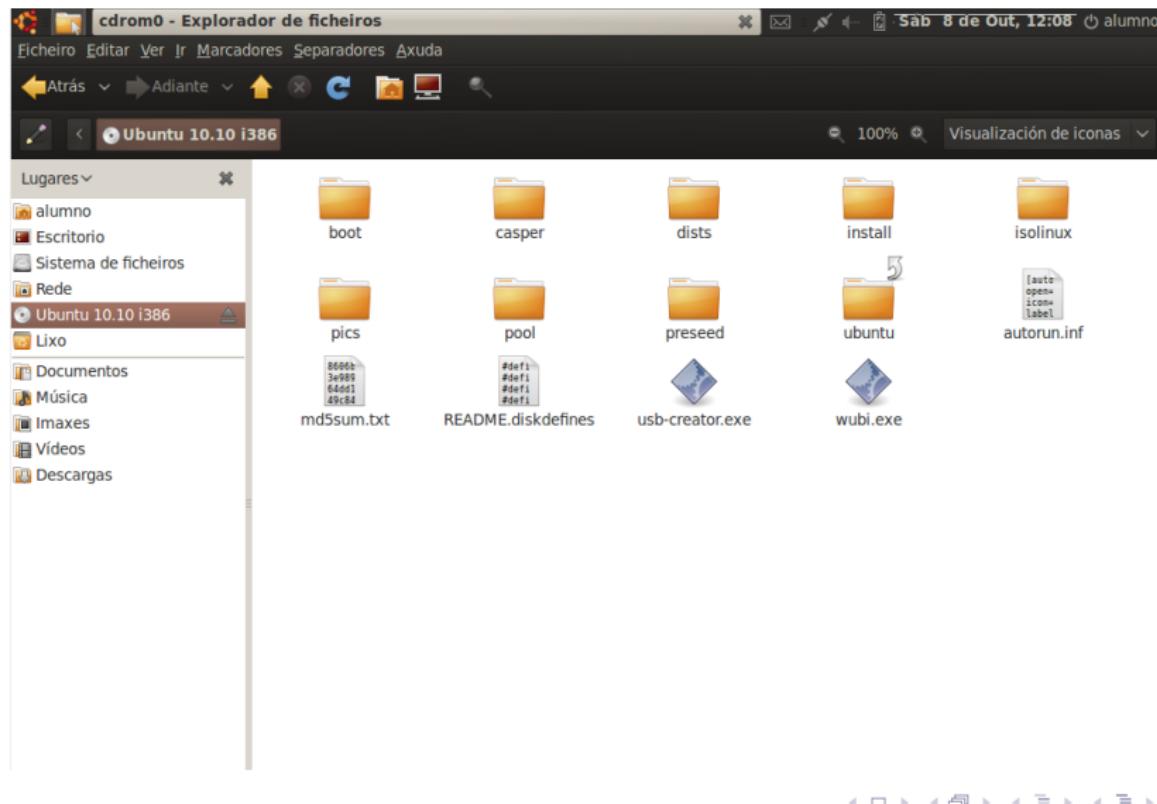
Unidades extraibles

- ▶ Al introducir un medio estraible (DVD, CD, USB ...), el entorno de escritorio lo detectará y procederá a *montarlo*
- ▶ Las distintas unidades en linux no se designan con otra letra, sino que aparecen *montadas* sobre un directorio. Los entornos de escritorio *gnome* y UNR, las colocan en el direcorio /media
- ▶ En el navegador de archivos, en la ventana nos aparece ahora el medio insertado, junto con la opción de expulsión
- ▶ Opcionalmente en *gnome* aparece sobre el escritorio un icono que nos permite acceder al medio.

Unidades extraíbles gnome



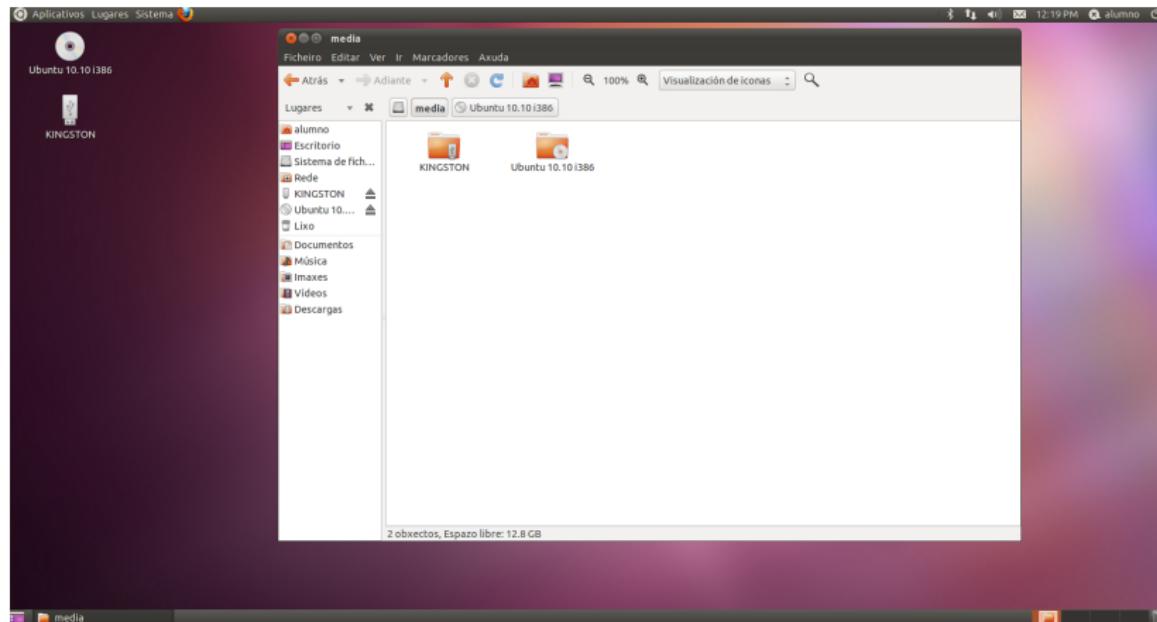
Unidades extraibles UNR



Unidades extraibles: acceso a archivos

- ▶ Para acceder a los archivos de los medios extraibles se procede igual que con los archivos de disco, puede hacerse tanto desde la terminal como desde la interfaz gráfica.
- ▶ En linux, los distintos sistemas de ficheros no se designan con distinta letra (como ocurre en *windows*) sino que aparecen en el directorio (carpeta) `/media`
 - ▶ En los medios WORM (Write Once Read Many) p.e. CDs y DVDs, no pueden copiarse archivos simplemente arrastrándolos y es preciso utilizar un programa de *masterización*
 - ▶ Los programas de *masterización* mas usuales son el *brasero* y *k3b*
 - ▶ Normalmente al introducir uno de estos medios vacío el entorno de escritorio nos dará la opción de arrancar uno de estos programas

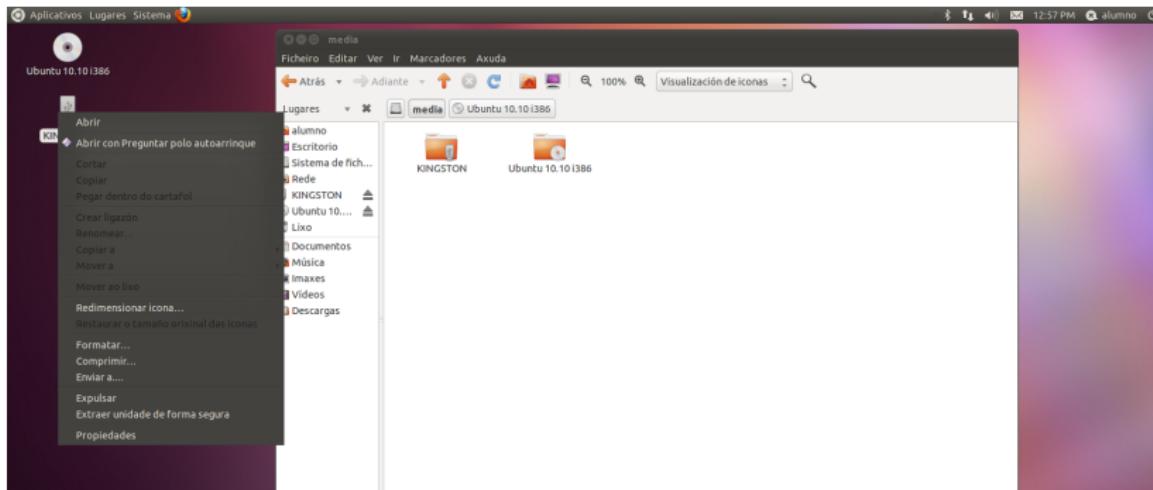
Unidades extraibles en directorio /media



Unidades extraibles: Expulsión y formateo

- ▶ Pulsando con el botón derecho sobre el icono de uno de los medios extraibles nos aparece la opción de *extraer unidad de forma segura* (algunas veces denominada desmontar). Esta opción
 - ▶ Hace que la unidad deje de estar accesible
 - ▶ **DEBEMOS** usarla siempre antes de retirar el medio
- ▶ La opción de *expulsar* es equivalente salvo que, si el medio puede ser expulsado (p.e. un CD), lo expulsa
- ▶ La opción de formatear nos permite formaterar un medio. Si queremos que el medio sea accesible en sistemas *windows* debemos usar formato FAT o FAT32

Unidades extraíbles, expulsión



Actividades I

- ▶ Crear una carpeta en nuestro directorio home a la que los otros usuarios del sistema no tenga acceso
- ▶ Crear una carpeta en nuestro directorio home donde el resto de los usuarios el sistema puedan acceder y añadir ficheros
- ▶ Crear una carpeta en nuestro directorio home en donde los otros usuarios puedan acceder a nuestros ficheros pero no puedan añadir ni borrar ficheros
- ▶ Hacer que otros usuarios del sistema puedan ver lo que tenemos en la carpeta *descargas* pero no puedan acceder a ello
- ▶ Realizar las tareas anteriores desde el terminal
- ▶ Insertar un USB y copiar sus contenidos a una carpeta situada en nuestro escritorio, de nombre *copiaUSB*

Actividades II

- ▶ Utilizar el terminal para mover parte de los contenidos del USB a la carpeta *descargas*
- ▶ Borrar la carpeta *copiaUSB* y recuperarla de la *papelera*
- ▶ Desde el terminal borrar todos los contenidos del USB. Intentar recuperarlos yendo a la *papelera*
- ▶ Restaurar los contenidos del USB utilizando la copia en *copiaUSB*
- ▶ Expulsar el USB