

Python para RPi

Parte I: Introducción a GNU/Linux y Python

Manel Velasco,¹ PhD and Alexandre Perera,^{1,2} PhD

¹Departament d'Enginyeria de Sistemes, Automatica i Informatica Industrial (ESAI)
Universitat Politecnica de Catalunya

²Centro de Investigacion Biomedica en Red en Bioingenieria, Biomateriales y Nanomedicina
(CIBER-BBN)

Alexandre.Perera@upc.edu Manel.Velasco@upc.edu

Introduction to Python and Raspberry Pi
July, 2013

Outline

- 1 Content
- 2 Sobre los Autores
- 3 Introducción a GNU/Linux
 - Porqué GNU/Linux ?
 - Línea de comando
 - Estructura de un sistema GNU/Linux
 - Sistema de archivos
- 4 Comandos UNIX útiles
 - Usuarios y Permisos
 - Gestión de usuarios (vía terminal)
 - Ejecutando tareas como administrador
 - Instalando programas

Introducción al curso

Este curso se estructura en cinco partes:

- 1 Introducción a GNU/Linux
- 2 Python
- 3 Raspberry Pi
- 4 Python en la Raspberry Pi

Outline

- 1 Content
- 2 Sobre los Autores
- 3 Introducción a GNU/Linux
 - Porqué GNU/Linux ?
 - Línea de comando
 - Estructura de un sistema GNU/Linux
 - Sistema de archivos
- 4 Comandos UNIX útiles
 - Usuarios y Permisos
 - Gestión de usuarios (vía terminal)
 - Ejecutando tareas como administrador
 - Instalando programas

Scipy Lecture Notes

Algunas partes de este seminario contiene material de <http://scipy-lectures.github.com>'s **Scipy Lecture Notes**. Esta web es un proyecto de código abierto para la creación de material para la enseñanza de python, incluyendo teoría, ejercicios y herramientas.

Editors

- Valentin Haenel
- Emmanuelle Gouillart
- Gaël Varoquaux

Additional Contributors

- Akihiro Uchida
- Corey Farwell
- egens
- Lars Buitinck
- Olivier Verdier
- Virgile Fritsch

Authors

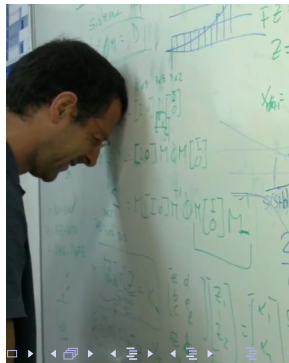
- Christopher Burns
- Adrian Chauve
- Robert Cimrman
- Christophe Combelles
- André Espaze
- Emmanuelle Gouillart
- Mike Müller
- Fabian Pedregosa
- Didrik Pinte
- Nicolas Rougier
- Gaël Varoquaux
- Pauli Virtanen
- Zbigniew Jędrzejewski-Szmek

About Us

Manel Velasco, PhD

Manel Velasco graduated in maritime engineering in 1999 and received the PhD degree in automatic control in 2006, both from the Technical University of Catalonia, Barcelona, Spain. He has been involved in research on artificial intelligence from 1999 to 2002 and, since 2000, on the impact of real-time systems on control systems. His research interests include artificial intelligence, real-time control systems, and collaborative control systems, especially on redundant controllers and multiple controllers with self-interacting systems.

Automatic Control Department
Universitat Politècnica de Catalunya
Pau Gargallo, 5
08028 Barcelona (Spain)
Phone: +34 93 401 1681
Fax: +34 93 401 7045
manel.velasco@upc.edu



Alexandre Perera, PhD

Alexandre Perera-LLuna graduated in physics at University of Barcelona at 1999 and in electrical engineering at 2001, he received a PhD degree in physics from the same university in 2003. He stayed as a postdoctoral fellow at Texas A&M University (USA) and in Universitat Politècnica de Catalunya(Spain) as a Ramon y Cajal Fellow from 2008-1012. His main area of expertise covers machine learning, statistical analysis, and data mining in biomedical systems, bioengineering and bioinformatics. He is an Associate Professor at Universitat Politècnica de Catalunya-BarcelonaTech (UPC).

Automatic Control Department
Universitat Politècnica de Catalunya
Pau Gargallo, 5
08028 Barcelona (Spain)
Phone: +34 93 401 6963
Fax: +34 93 401 7045
Alexandre.Perera@upc.edu



About this course

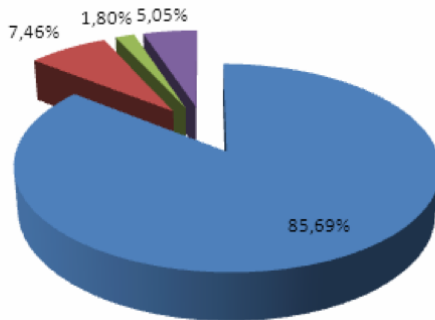
- Todo el material se ha preparado empleando \LaTeX y ediado en **Vim** (por desgracia de Alex).
- El código Python están empotrados en \LaTeX con la ayuda de **Pweave**, desarrollado por Matti Pastell.
- Todo el código Python se ha marcado en el documento mediante el paquete *minted*, desarrollado por Konrad Rudolph, el *python syntax highlighter* **Pygments**, y código bash propio.

Outline

- 1 Content
- 2 Sobre los Autores
- 3 **Introducción a GNU/Linux**
 - Porqué GNU/Linux ?
 - Linea de comando
 - Estructura de un sistema GNU/Linux
 - Sistema de archivos
- 4 Comandos UNIX útiles
 - Usuarios y Permisos
 - Gestión de usuarios (vía terminal)
 - Ejecutando tareas como administrador
 - Instalando programas

Uso de los sistemas operativos de escritorio

■ Microsoft Windows 98, 2000, 2003, XP y 7 ■ Mac OS X ■ Linux ■ Otros

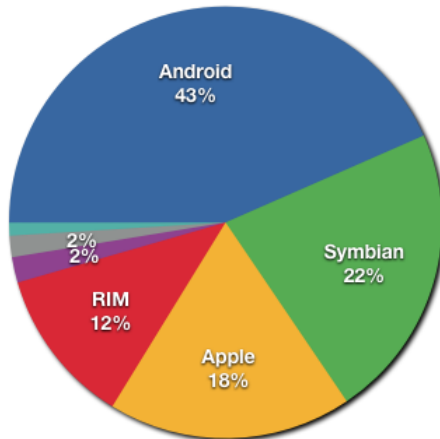


Sistemes operatius més usats en mòbils

Aunque no es estrictamente cierto....

Sistemas operatius més usats en mòbils

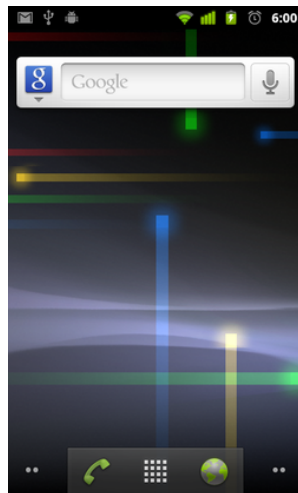
Aunque no es estrictamente cierto....



Android

Android es...

Un conjunto de programas para teléfonos móviles, que incluye un sistema operativo y un conjunto de librerías y aplicaciones. En 2005, Google Inc. compró Android a Android Inc. Android se basa en una versión modificada del kernel de Linux. Android Open Source Project (AOSP) tiene como objetivo mantener y desarrollar el sistema operativo Android. Android es la plataforma de sistemas móviles con más prevalente.



iOS (iphone)

Android és...

iOS es el sistema operativo del iPhone, iPod Touch, iPad, que es el mismo sistema operativo que el sistema de escritorio de Mac, OS X.

Mac OS X es una versión del sistema operativo que utilizan los ordenadores Macintosh, basados en un sistema operativo UNIX.



Usos de UNIX y GNU/Linux

Linux se usa en:

- U.S. Department of Defense
- Parlamento Francés
- Google
- IBM
- E*Trade (La bolsa de Nova York)
- Servidores de Bases de Datos
- Maquinaria industrial
- Routers ADSL
- 61 % de los Teléfonos móviles !!

Usos de UNIX y GNU/Linux

Linux se usa en:

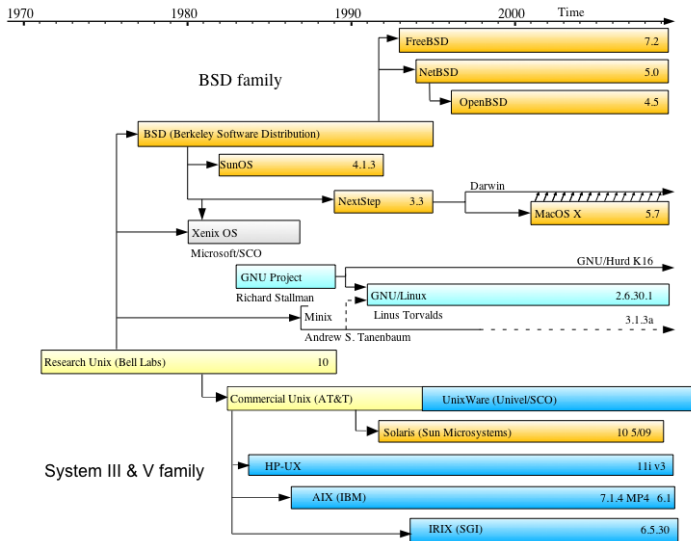
- U.S. Department of Defense
- Parlamento Francés
- Google
- IBM
- E*Trade (La bolsa de Nova York)
- Servidores de Bases de Datos
- Maquinaria industrial
- Routers ADSL
- 61 % de los Teléfonos móviles !!

y en

La Raspberry Pi !!

Historia de GNU/Linux y UNIX

Historia de GNU/Linux y UNIX



Propiedades de GNU/Linux

- Un sistema operativo de 32/64 bits clon de UNIX

Propiedades de GNU/Linux

- Un sistema operativo de 32/64 bits clon de UNIX
- Herramientas para la gestión avanzada y no interactiva de un sistema (sed, awk, bash, grep, etc)

Propiedades de GNU/Linux

- Un sistema operativo de 32/64 bits clon de UNIX
- Herramientas para la gestión avanzada y no interactiva de un sistema (sed, awk, bash, grep, etc)
- Compiladores: C, C++, Fortran, Ada, etc

Propiedades de GNU/Linux

- Un sistema operativo de 32/64 bits clon de UNIX
- Herramientas para la gestión avanzada y no interactiva de un sistema (sed, awk, bash, grep, etc)
- Compiladores: C, C++, Fortran, Ada, etc
- Interpretes: *Python*, java, R, octave, etc

Propiedades de GNU/Linux

- Un sistema operativo de 32/64 bits clon de UNIX
- Herramientas para la gestión avanzada y no interactiva de un sistema (sed, awk, bash, grep, etc)
- Compiladores: C, C++, Fortran, Ada, etc
- Interpretes: *Python*, java, R, octave, etc
- Multi-usuario, multi-tarea, multi-processorador

Propiedades de GNU/Linux

- Un sistema operativo de 32/64 bits clon de UNIX
- Herramientas para la gestión avanzada y no interactiva de un sistema (sed, awk, bash, grep, etc)
- Compiladores: C, C++, Fortran, Ada, etc
- Interpretes: *Python*, java, R, octave, etc
- Multi-usuario, multi-tarea, multi-processorador
- Puede coexistir con otros sistemas operativos (grub)

Propiedades de GNU/Linux

- Un sistema operativo de 32/64 bits clon de UNIX
- Herramientas para la gestión avanzada y no interactiva de un sistema (sed, awk, bash, grep, etc)
- Compiladores: C, C++, Fortran, Ada, etc
- Interpretes: *Python*, java, R, octave, etc
- Multi-usuario, multi-tarea, multi-processorador
- Puede coexistir con otros sistemas operativos (grub)
- Corre en diferentes plataformas (móviles, Raspberry Pi)

Propiedades de GNU/Linux

- Un sistema operativo de 32/64 bits clon de UNIX
- Herramientas para la gestión avanzada y no interactiva de un sistema (sed, awk, bash, grep, etc)
- Compiladores: C, C++, Fortran, Ada, etc
- Interpretes: *Python*, java, R, octave, etc
- Multi-usuario, multi-tarea, multi-processorador
- Puede coexistir con otros sistemas operativos (grub)
- Corre en diferentes plataformas (móviles, Raspberry Pi)
- Su código fuente es abierto, libre y pertenece a la comunidad.

Distribuciones de GNU/Linux

- Código fuente del kernel <http://www.kernel.org>
- Distribuciones:

Ranquing	Distribución	Ratio
1	Ubuntu	2239
2	Mint	2069
3	Fedora	1685
4	Debian	1305
5	Arch	1240
6	openSUSE	1221
7	PCLinuxOS	1040
8	CentOS	893
9	Puppy	819
10	Mandriva	716
11	Lubuntu	617

Cuadro: Clasificación per <http://distrowatch.com>

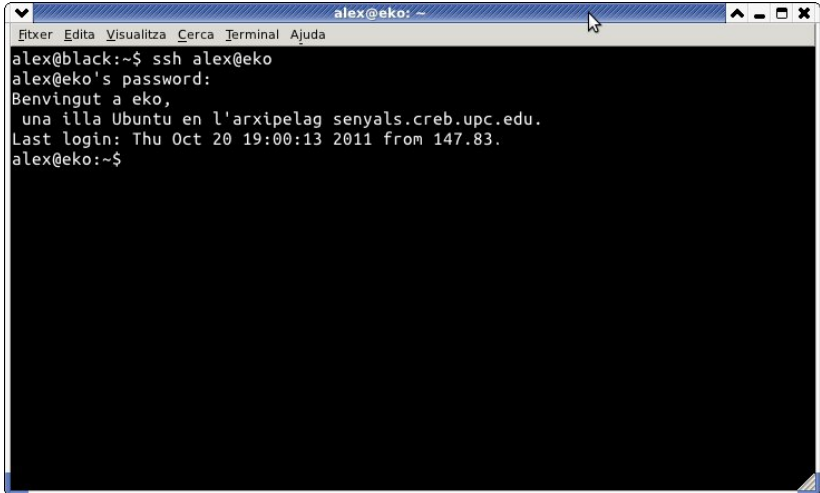
El terminal

- Antiguamente se accedía a los sistemas UNIX mediante un terminal.



Terminal vt100

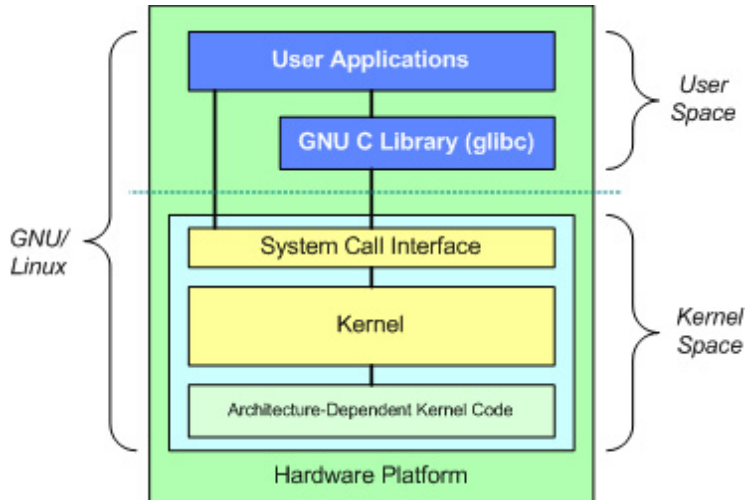
El terminal



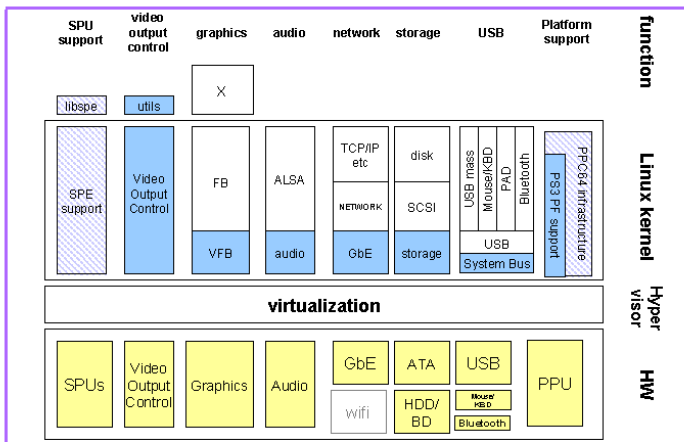
A screenshot of a terminal window titled "alex@eko: ~". The window has a menu bar with options: Fitxer, Edita, Visualitza, Cerca, Terminal, and Ajuda. The terminal content shows an SSH session from alex@black to alex@eko. The user enters the password, and the system responds with a welcome message and login details.

```
alex@black:~$ ssh alex@eko
alex@eko's password:
Benvingut a eko,
una illa Ubuntu en l'arxipelag senyals.creb.upc.edu.
Last login: Thu Oct 20 19:00:13 2011 from 147.83.
alex@eko:~$
```

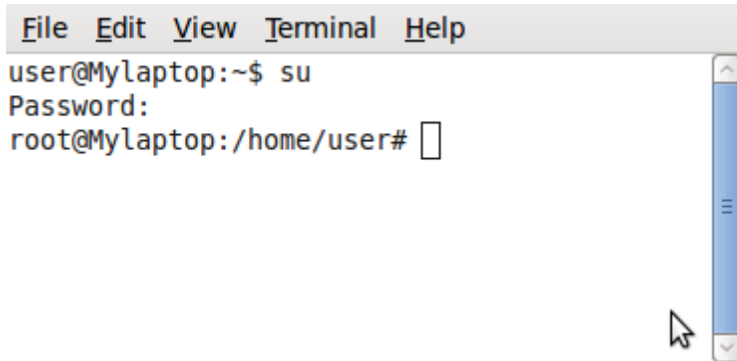
Estructura del Linux



Estructura del Linux



Estructura del Linux



The image shows a terminal window with a menu bar at the top containing 'File', 'Edit', 'View', 'Terminal', and 'Help'. The terminal text shows a user at 'user@Mylaptop:~\$' entering the command 'su'. After a 'Password:' prompt, the user has successfully switched to root, indicated by the prompt 'root@Mylaptop:/home/user#'. A mouse cursor is visible at the bottom right of the terminal area.

```
File Edit View Terminal Help
user@Mylaptop:~$ su
Password:
root@Mylaptop:/home/user#
```


Ficheros

Dos cosas importantes a tener en cuenta

- 1 Todo en UNIX son ficheros.

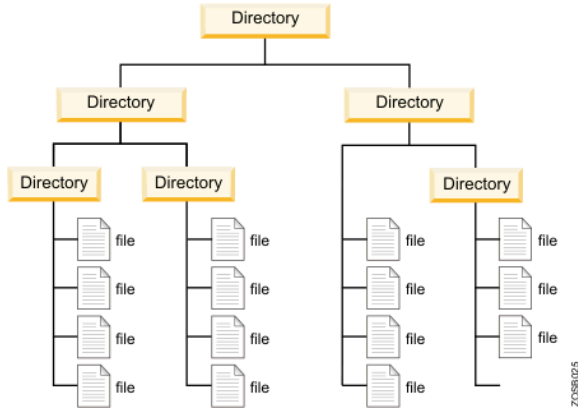
Ficheros

Dos cosas importantes a tener en cuenta

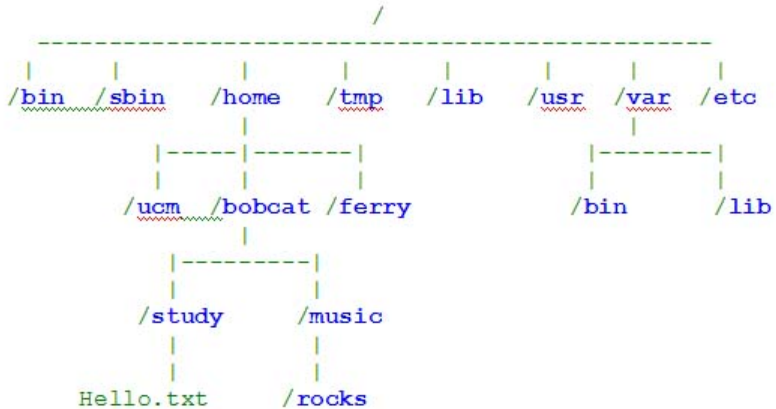
- 1 Todo en UNIX son ficheros.
- 2 Todo fichero pertenece a un usuario y a un grupo.

Sistema d'arxius

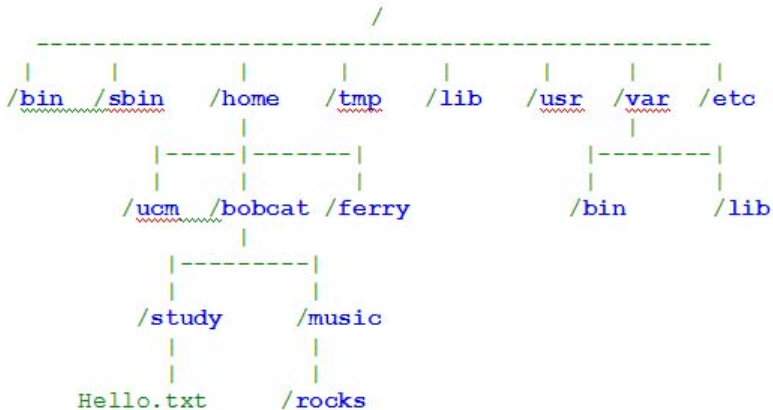
- Los ficheros en UNIX estan almacenados en estructura jerárquica, en forma de árbol con una sola raíz.
- Cada nodo “intermedio” son directorios.
- Cada nodo “hoja” son ficheros.



Sistema d'arxius

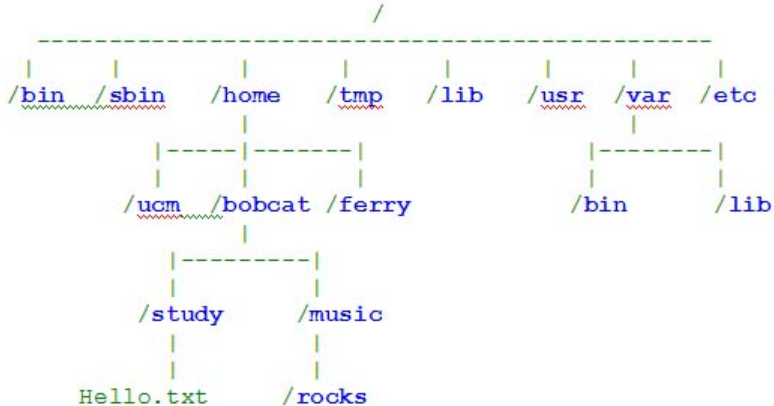


Sistema de archivos, *pathnames*



- El camino per acceder a un recurso se denomina *pathname*
- e.g. */home/bobcat/study/Hello.txt*

Sistema de archivos, *current pathname*



- El *shell* siempre está apuntando en un punto del árbol del sistema de archivos.
- Podemos usar *pwd* para obtener el *current pathname*

Nombres especiales en los fitxers

- / representa el nodo raiz, o *root directory*.

Nombres especiales en los fitxers

- / representa el nodo raiz, o *root directory*.
- . representa el directorio actual.

Nombres especiales en los fitxers

- / representa el nodo raiz, o *root directory*.
- . representa el directorio actual.
- .. representa el directorio anterior.

Nombres especiales en los fixers

- / representa el nodo raiz, o *root directory*.
- . representa el directorio actual.
- .. representa el directorio anterior.
- ~ representa el directorio del usuario en curso. Este directorio es diferente para cada usuario.

Ubicacions especiales de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)

Ubicacions especials de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema

Ubicacions especials de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema
- */sbin* i */usr/sbin* comandos de administración del sistema

Ubicacions especiales de fixters

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema
- */sbin* i */usr/sbin* comandos de administración del sistema
- */etc* directorio con las configuraciones del sistema que afectan a todos los usuarios.

Ubicacions especials de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema
- */sbin* i */usr/sbin* comandos de administración del sistema
- */etc* directorio con las configuraciones del sistema que afectan a todos los usuarios.
- */var* varios directorios per usos diferentes

Ubicacions especiales de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema
- */sbin* i */usr/sbin* comandos de administración del sistema
- */etc* directorio con las configuraciones del sistema que afectan a todos los usuarios.
- */var* varios directorios per usos diferentes
 - */var/log* contiene archivos que monitorizan la actividad del sistema. (actividad de apache, dispositivos, red, autorizaciones, ataques al sistema, etc).

Ubicacions especiales de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema
- */sbin* i */usr/sbin* comandos de administración del sistema
- */etc* directorio con las configuraciones del sistema que afectan a todos los usuarios.
- */var* varios directorios per usos diferentes
 - */var/log* contiene archivos que monitorizan la actividad del sistema. (actividad de apache, dispositivos, red, autorizaciones, ataques al sistema, etc).
 - */var/tmp* y */tmp* contienen archivos temporales.

Ubicacions especiales de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema
- */sbin* i */usr/sbin* comandos de administración del sistema
- */etc* directorio con las configuraciones del sistema que afectan a todos los usuarios.
- */var* varios directorios per usos diferentes
 - */var/log* contiene archivos que monitorizan la actividad del sistema. (actividad de apache, dispositivos, red, autorizaciones, ataques al sistema, etc).
 - */var/tmp* y */tmp* contienen archivos temporales.
 - */var/spool* contiene varios *spools* (carretes o bobinas), e.g. *cups*, *fax*, *mqueue* y otras.

Ubicacions especiales de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema
- */sbin* i */usr/sbin* comandos de administración del sistema
- */etc* directorio con las configuraciones del sistema que afectan a todos los usuarios.
- */var* varios directorios per usos diferentes
 - */var/log* contiene archivos que monitorizan la actividad del sistema. (actividad de apache, dispositivos, red, autorizaciones, ataques al sistema, etc).
 - */var/tmp* y */tmp* contienen archivos temporales.
 - */var/spool* contiene varios *spools* (carretes o bobinas), e.g. *cups*, *fax*, *mqueue* y otras.
- */dev* contiene ficheros que representan dispositivos.

Ubicacions especiales de fitxers

- */home* directorio donde estan todos los directorios *home* para cada usuario. Depende de la variante del sistema operativo (e.g. en OS X es */Users/*)
- */bin* i */usr/bin* comandos del sistema
- */sbin* i */usr/sbin* comandos de administración del sistema
- */etc* directorio con las configuraciones del sistema que afectan a todos los usuarios.
- */var* varios directorios per usos diferentes
 - */var/log* contiene archivos que monitorizan la actividad del sistema. (actividad de apache, dispositivos, red, autorizaciones, ataques al sistema, etc).
 - */var/tmp* y */tmp* contienen archivos temporales.
 - */var/spool* contiene varios *spools* (carretes o bobinas), e.g. *cups*, *fax*, *mqueue* y otras.
- */dev* contiene ficheros que representan dispositivos.
- */proc* conté ficheros especiales, relacionados con el sistema.

Outline

- 1 Content
- 2 Sobre los Autores
- 3 Introducción a GNU/Linux
 - Porqué GNU/Linux ?
 - Línea de comando
 - Estructura de un sistema GNU/Linux
 - Sistema de archivos
- 4 **Comandos UNIX útiles**
 - Usuarios y Permisos
 - Gestión de usuarios (vía terminal)
 - Ejecutando tareas como administrador
 - Instalando programas

Comandos UNIX útiles

- *cp fuente destino* copia el fichero de fuente a destino. Con el modificador *-r* podemos copiar recursivamente.

Comandos UNIX útiles

- *cp fuente destino* copia el fichero de fuente a destino. Con el modificador *-r* podemos copiar recursivamente.
- *mv fuente destino* mueve el fichero o directorio de fuente a destino.

Comandos UNIX útiles

- *cp fuente destino* copia el fichero de fuente a destino. Con el modificador *-r* podemos copiar recursivamente.
- *mv fuente destino* mueve el fichero o directorio de fuente a destino.
- *rm fichero* borra el fichero (*-r*).

Comandos UNIX útiles

- *cp fuente destino* copia el fichero de fuente a destino. Con el modificador *-r* podemos copiar recursivamente.
- *mv fuente destino* mueve el fichero o directorio de fuente a destino.
- *rm fichero* borra el fichero (*-r*).
- *mkdir directorio* crea un directorio.

Comandos UNIX útiles

- *cp fuente destino* copia el fichero de fuente a destino. Con el modificador *-r* podemos copiar recursivamente.
- *mv fuente destino* mueve el fichero o directorio de fuente a destino.
- *rm fichero* borra el fichero (*-r*).
- *mkdir directorio* crea un directorio.
- *rmdir directorio* borra un directorio.

Usuarios

- Todo en UNIX (persona o programas) que usen el sistema operativo lo hacen mediante un *usuario*.

Usuarios

- Todo en UNIX (persona o programas) que usen el sistema operativo lo hacen mediante un *usuario*.
- El sistema contempla diferentes usuarios, con diferentes privilegios. Existen usuarios especiales:

Usuarios

- Todo en UNIX (persona o programas) que usen el sistema operativo lo hacen mediante un *usuario*.
- El sistema contempla diferentes usuarios, con diferentes privilegios. Existen usuarios especiales:
- *root*: Todo sistema UNIX tiene un usuario root
 - Permite acceder a todos los ficheros del sistema (y por tanto todos los dispositivos)
 - Tiene su *home* en */root*.
 - En ubuntu, no tiene *password* asignado en una instalación estándar, pero realiza las tareas administrativas como *root* mediante *sudo* (Administración de programes (instalación)).
 - \$ - #

Usuarios

- Todo en UNIX (persona o programas) que usen el sistema operativo lo hacen mediante un *usuario*.
- El sistema contempla diferentes usuarios, con diferentes privilegios. Existen usuarios especiales:
- *root*: Todo sistema UNIX tiene un usuario root
 - Permite acceder a todos los ficheros del sistema (y por tanto todos los dispositivos)
 - Tiene su *home* en */root*.
 - En ubuntu, no tiene *password* asignado en una instalación estándar, pero realiza las tareas administrativas como *root* mediante *sudo* (Administración de programes (instalación)).
 - \$ - #
- *www-data*: Servidor web.
 - Tiene el *home* en */var/www*, que es donde se ubican las páginas web.

ls -l

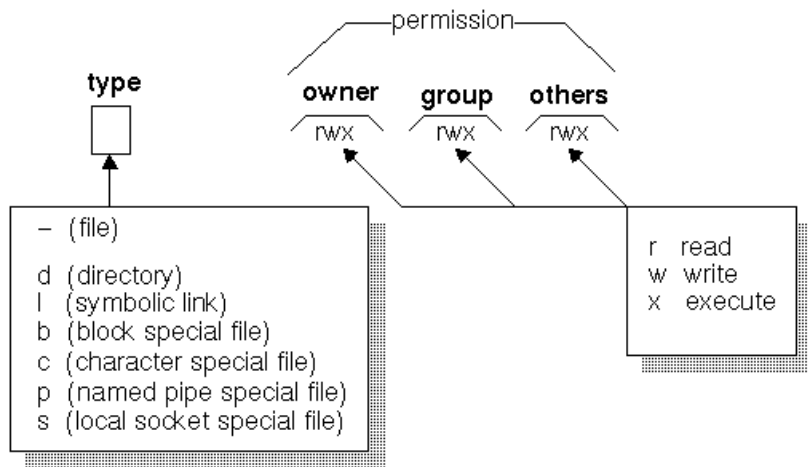
```
-rw----- 1 bshotts bshotts      576 Apr 17 1998 weather.txt
drwxr-xr-x 6 bshotts bshotts     1024 Oct  9 1999 web_page
-rw-rw-r-- 1 bshotts bshotts    276480 Feb 11 20:41 web_site.tar
-rw----- 1 bshotts bshotts      5743 Dec 16 1998 xmas_file.txt
```

File Name	Modification Time	Size (in bytes)	Group	Owner	File Permissions
weather.txt	Apr 17 1998	576	bshotts	bshotts	-rw-----
web_page	Oct 9 1999	1024	bshotts	bshotts	drwxr-xr-x
web_site.tar	Feb 11 20:41	276480	bshotts	bshotts	-rw-rw-r--
xmas_file.txt	Dec 16 1998	5743	bshotts	bshotts	-rw-----

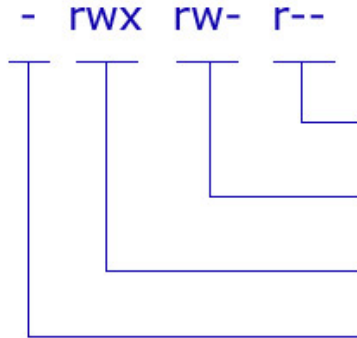
Entendiendo ls -l

- FileName: Nombre del fichero
- Modification Time: Fecha última modificación
- Size: Tamaño en *bytes*
- Group: Grupo propietario
- Owner: Usuario Propietario
- File Permissions: Una representación de los permisos.

Permisos en UNIX



Permisos en UNIX



Read, write and execute permissions for all other users

Read, write and execute permissions for members of the group owning the file

Read, write and execute permissions for the owner of the file

File type: "-" means a file.
"d" means a directory.

Permisos en UNIX

	u			g			o		
	754								
	/						\		
access	r	w	x	r	w	x	r	w	x
binary	4	2	1	4	2	1	4	2	1
enabled	1	1	1	1	0	1	1	0	0
	<hr/>			<hr/>			<hr/>		
result	4	2	1	4	0	1	4	0	0
	<hr/>			<hr/>			<hr/>		
total	7			5			4		

Modificando Permisos: chmod

- `chmod 600 test.txt` Modifica todos los permisos según un código binari.

(*) Podeis ver los cambios mediante `ls -l test.txt`. Podeis crear un archivo mediante un editor de texto o `touch`.

Modificando Permisos: chmod

- `chmod 600 test.txt` Modifica todos los permisos según un código binari.
- `chmod o+r test.txt` Modifica los permisos de *others*.

(*) Podeis ver los cambios mediante `ls -l test.txt`. Podeis crear un archivo mediante un editor de texto o `touch`.

Modificando Permisos: chmod

- `chmod 600 test.txt` Modifica todos los permisos según un código binari.
- `chmod o+r test.txt` Modifica los permisos de *others*.
- `chmod o-r test.txt` Deshabilita el bit de lectura para *others*

(*) Podeis ver los cambios mediante `ls -l test.txt`. Podeis crear un archivo mediante un editor de texto o `touch`.

Modificando Permisos: chmod

- `chmod 600 test.txt` Modifica todos los permisos según un código binari.
- `chmod o+r test.txt` Modifica los permisos de *others*.
- `chmod o-r test.txt` Deshabilita el bit de lectura para *others*
- `chmod g+w test.txt` Habilita el bit de escritura para el grupo.

(*) Podeis ver los cambios mediante `ls -l test.txt`. Podeis crear un archivo mediante un editor de texto o `touch`.

Modificando Permisos: chmod

- `chmod 600 test.txt` Modifica todos los permisos según un código binari.
- `chmod o+r test.txt` Modifica los permisos de *others*.
- `chmod o-r test.txt` Deshabilita el bit de lectura para *others*
- `chmod g+w test.txt` Habilita el bit de escritura para el grupo.
- `chmod u+x test.txt` Habilitat el bit de ejecución para el usuario.

(*) Podeis ver los cambios mediante `ls -l test.txt`. Podeis crear un archivo mediante un editor de texto o `touch`.

Modificar el propietario de un archivo o directorio

La orden `chown` altera la propiedad de un fichero

```
chown
```

```
chown alex logfile.txt
```

Hace que la propiedad de *logfile.txt* pase al usuario *alex*.

(*) Recordad que podeis ver los cambios con *ls -l test.txt*

Añadir y borrar usuarios

Añadir un usuario

```
adduser username
```

Añadir y borrar usuarios

Añadir un usuario

```
adduser username
```

Borrar un usuario

```
deluser username
```

Añadir y borrar grupos

Añadir un grupo

```
addgroup username
```

Añadir y borrar grupos

Añadir un grupo

```
addgroup username
```

Borrar un grupo

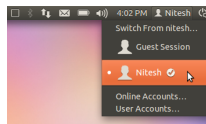
```
delgroup username
```

Añadir un usuario a un grupo

```
adduser username groupname
```

Gestión de usuarios a Ubuntu (GUI)

- Ubuntu contiene una herramienta gráfica para la gestión de usuarios
- Permite la gestión fácil de usuarios y grupos.



Quién puede ejecutar tareas como administrador (ubuntu)

- El usuario *root* no tiene contraseña.

Quién puede ejecutar tareas como administrador (ubuntu)

- El usuario *root* no tiene contraseña.
- En los sistemas *ubuntu* los usuarios que pertenecen al grupo *admin* pueden ejecutar tareas como administrador.

Quién puede ejecutar tareas como administrador (ubuntu)

- El usuario *root* no tiene contraseña.
- En los sistemas *ubuntu* los usuarios que pertenecen al grupo *admin* pueden ejecutar tareas como administrador.
- Se escalan privilegios mediante *sudo*.

Quién puede ejecutar tareas como administrador (ubuntu)

- El usuario *root* no tiene contraseña.
- En los sistemas *ubuntu* los usuarios que pertenecen al grupo *admin* pueden ejecutar tareas como administrador.
- Se escalan privilegios mediante *sudo*.
- Ejemplo *sudo visudo* edita el fichero que define quién puede escalar privilegios. (Intentad encontrar la entrada que determinará que el grupo *admin* puede utilizar *sudo*)

Desde la línea de comandos

apt-get (ubuntu)

Necesitamos saber el nombre del *paquete*, entonces *apt-get*.

Desde la línea de comandos

apt-get (ubuntu)

Necesitamos saber el nombre del *paquete*, entonces *apt-get*.

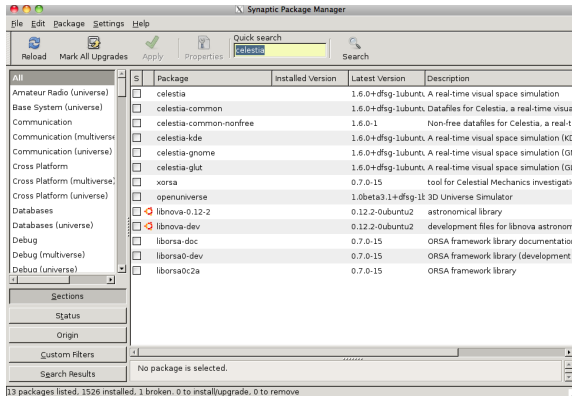
Actualización de un sistema

apt-get update
apt-get upgrade

Herramientas de actualización (GUI)

synaptic

synaptic es un programa para ayudarnos en la administración de programas.



Ejercicio

- Buscad e instalad los components ipython y ipython-notebook.

Ejercicio

- Buscad e instalad los components `ipython` y `ipython-notebook`.
- Buscad e instalad el editor para python *spyder*

Fin de la Parte I

