

Introduzione a Linux

Roberto De Virgilio

Sistemi operativi - 9 Ottobre 2017

OBIETTIVI

- ◆ *Fornire le informazioni per installare, configurare e amministrare un sistema Linux*
- ◆ *Fornire un buona preparazione “sistematica” utile a gestire Linux sysadmin and utenti Unix in generale*
- ◆ *Confrontare Linux con altri sistemi operativi*

Prerequisiti

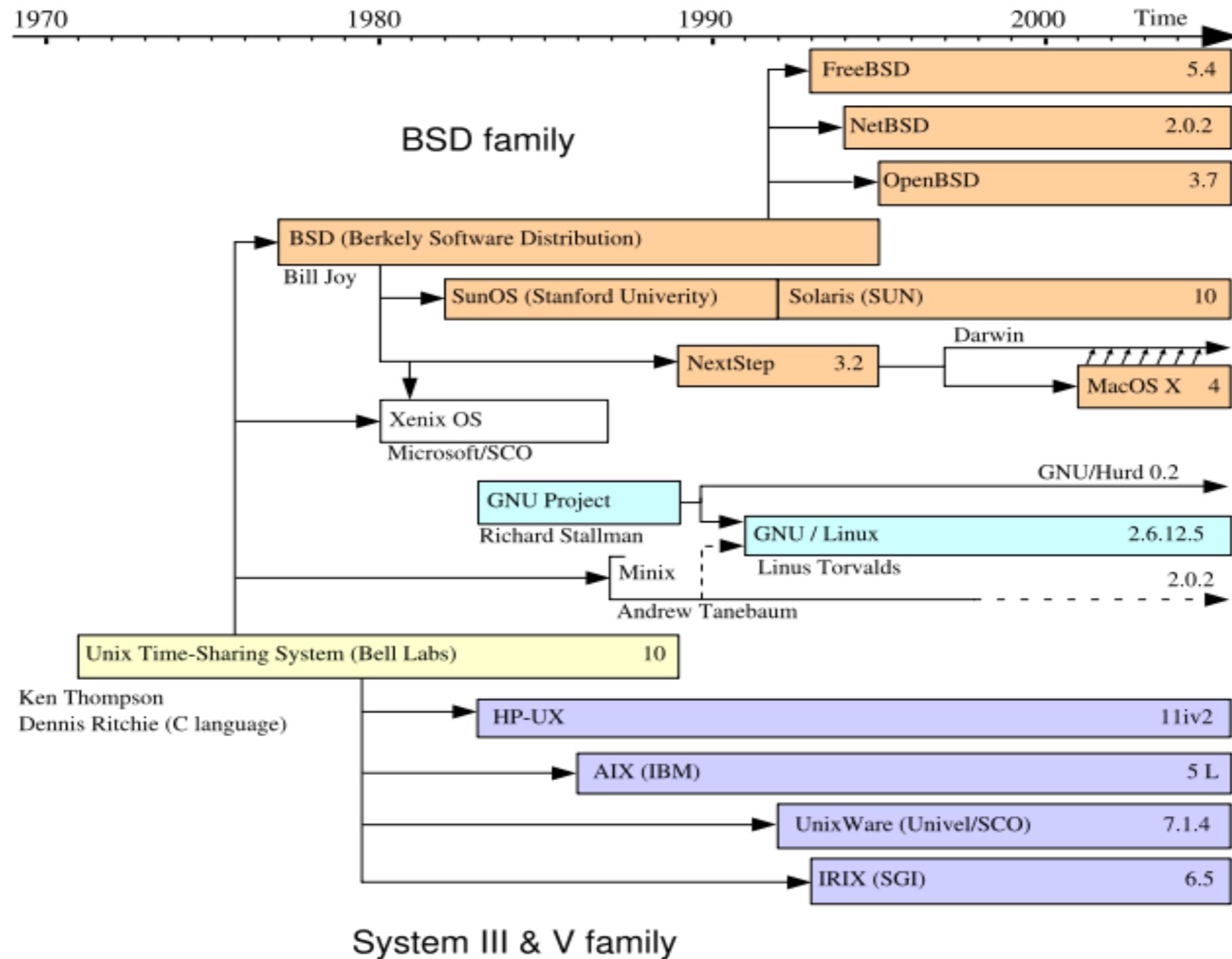
- ◆ *Buona Alfabetizzazione Informatica (Hardware, Software, Internet)*
- ◆ *Conoscenza base del protocollo TCP/IP*
- ◆ *Buona conoscenza del “Vocabolario Inglese”*

Un passo indietro: UNIX

- ◆ *Unix è un sistema operativo, nato nel 1969 presso i Bell Labs ad opera di Dennis Ritchie, Ken Thompson, Brian Kernighan ed altri programmati.*
- ◆ *Inizialmente chiunque fosse interessato e possedesse l'hardware occorrente, poteva chiedere ad un costo irrisorio un nastro del software ed i relativi manuali stampati.*
- ◆ *Questo accadeva prima dell'avvento dei personal computer, pertanto si trattava in genere di università e centri di ricerca. I singoli centri modificavano il codice sorgente ampliando e personalizzando il sistema in base alle loro necessità*



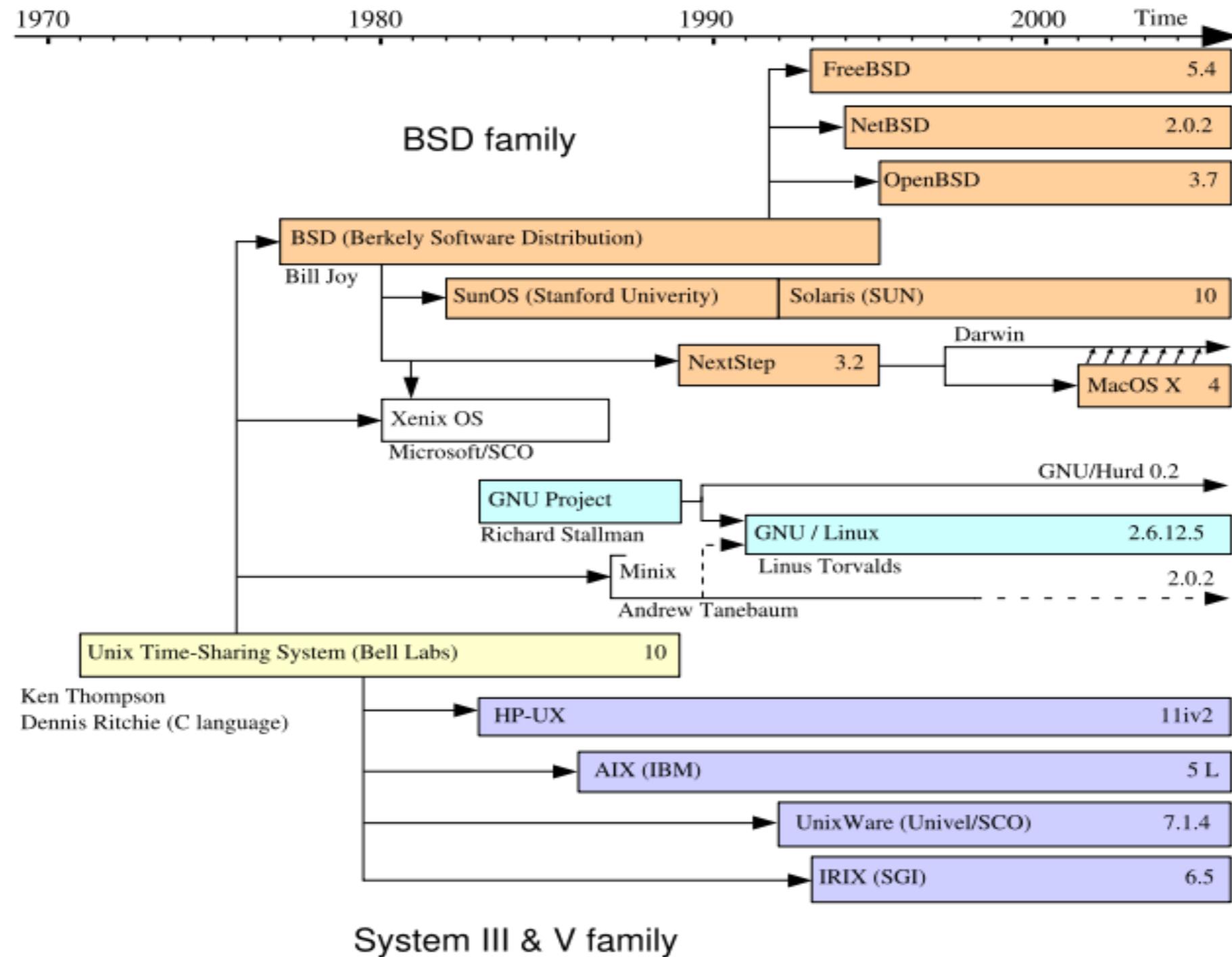
UNIX: evoluzioni



UNIX: evoluzioni

- ✿ Una tappa importante è stata raggiunta alla fine degli anni '70 con la realizzazione della versione **BSD** (*Berkley System Distribution*) ad opera di alcuni esperti di informatica dell'Università della California di Berkley, che apportarono diverse migliorie, la più importante fu l'adozione del protocollo **TCP/IP**.
- ✿ Il *codice sorgente* venne reso pubblicamente disponibile con una licenza che ne consentiva la distribuzione con o senza codice sorgente, a condizione che queste parti del codice venissero **ascritte** a Berkley

UNIX: evoluzioni



UNIX: evoluzioni



- ◆ *Alcune società commerciali adottarono la modalità di distribuzione del software senza sorgenti, ponendo le basi di un nuovo **modello di sviluppo proprietario***
- ◆ *Nel 1984 **Richard Stallman**, invece di cedere a questa nuova tendenza, decise di dare vita ad un nuovo sistema operativo di tipo Unix il cui codice sorgente potesse essere liberamente copiato e modificato. Nacque il progetto **GNU** (**GNU is Not Unix**)*
- ◆ *Il nuovo modello di sviluppo prese il nome di **Software Libero** (free software).*
- ◆ *Venne scritta una licenza specifica **GNU General Public License** (nota come **GPL**, <http://www.gnu.org/copyleft/gpl.txt>) che aggirasse i limiti imposti dai diritti d'autore e consentisse a chiunque di copiare e modificare un lavoro, seppur nel rispetto di condizioni e termini rigorosi.*

- ◆ Alcune persone senza scrupoli cercavano di appropriarsi del software.
- ◆ Nel 1985 Richard Stallman decise di dare alle persone il diritto di potersse servire con il software che volevano. Not Unfree Software.
- ◆ Il nuovo software era chiamato GNU.
- ◆ Venne scritto sotto la licenza GPL, una licenza che ti dà il diritti di usare il software nel rispetto dei diritti degli altri.



software

za, decise gente (GNU is software).

nota come ti dai o, seppur

Open Source o Sorgente Aperto

- ◆ Il termine **Open Source** venne introdotto nel 1998 quale termine commerciale per indicare il “**software libero**”. Si voleva enfatizzare il fatto che il software Open Source non vuol dire necessariamente **GRATUITO**.
- ◆ E' stata fondata la **OSI** (*Open Source Initiative*) allo scopo di promuovere il software conforme alla *Open Source Directive*.
- ◆ **Libero** indica diverse “**libertà**” ad esempio:
 - ▶ **Ridistribuzione** Libera del software e del codice
 - ▶ Il Codice Sorgente è **Aperto** e disponibile per lettura, modifiche, analisi, controllo
 - ▶ Prodotti derivati da prodotti GPL devono avere licenza GPL
 - ▶ **Copyright** dell'autore

Linux



- ◆ *Agli inizi degli anni '90, **Linus Torvald**, uno studente finlandese in scienze dell'informazione iniziò ad apportare variazioni a **Minix**, un sistema operativo di tipo Unix per personal computer allora utilizzato nei corsi universitari sui sistemi operativi*
- ◆ *Torvald decise di migliorare il componente principale del software alla base di Minix, chiamato **kernel**, e di scriverne uno proprio.*

Linux: fasi dello sviluppo



Luglio 1991

Linus Benedict Torvalds, ancora un giovane studente dell' Università di Helsinki, inizia a lavorare al suo hobby: Linux

5 Ottobre 1991

Nello stesso anno viene rilasciata la versione 0.02. Il post su usenet che ne annuncia la presenza è diventato un classico.

Gennaio 1992

Viene rilasciata la versione 0.12. Risulta relativamente stabile e supporta vario hardware

Aprile 1992

Rilasciate la versione 0.95 e 0.96. Il salto è diretto dalla 0.12. Nascono le prime distribuzioni: la MCC Linux e la SLS.

Linux: fasi dello sviluppo



1994

Viene rilasciata la prima versione definitiva 1.0. Nascono RedHat, Debian, SUSE ad oggi le distribuzioni più diffuse. Linux, che resta copyrighted by Linus Torvalds, diventa ufficialmente un software aperto, abbracciando in pieno la General Public License (GPL) del movimento GNU Open Source.
Nascono i primi LUGs (Linux User Groups), ormai diffusi anche in Italia.

1995

Compaiono sul mercato nuove distribuzione commerciali come Caldera Linux. Kernel 1.2 out in Marzo. Dal kernel 1.3 in sviluppo si passerà direttamente al 2.0

Linux: fasi dello sviluppo



1996

Rilasciata la versione 2.0.
Compaiono le prime versioni tradotte in più lingue.
Linux necessita di una mascotte: nasce TUX, il

1999

Dopo lunga attesa il kernel 2.2 vede la luce

2001

Agli inizi dell'anno, dopo varie pre-version, su kernel.org appare l'immagine da 19.788.626 byte del 2.4.0 La prima release di un altro stable thread

Linux: fasi dello sviluppo



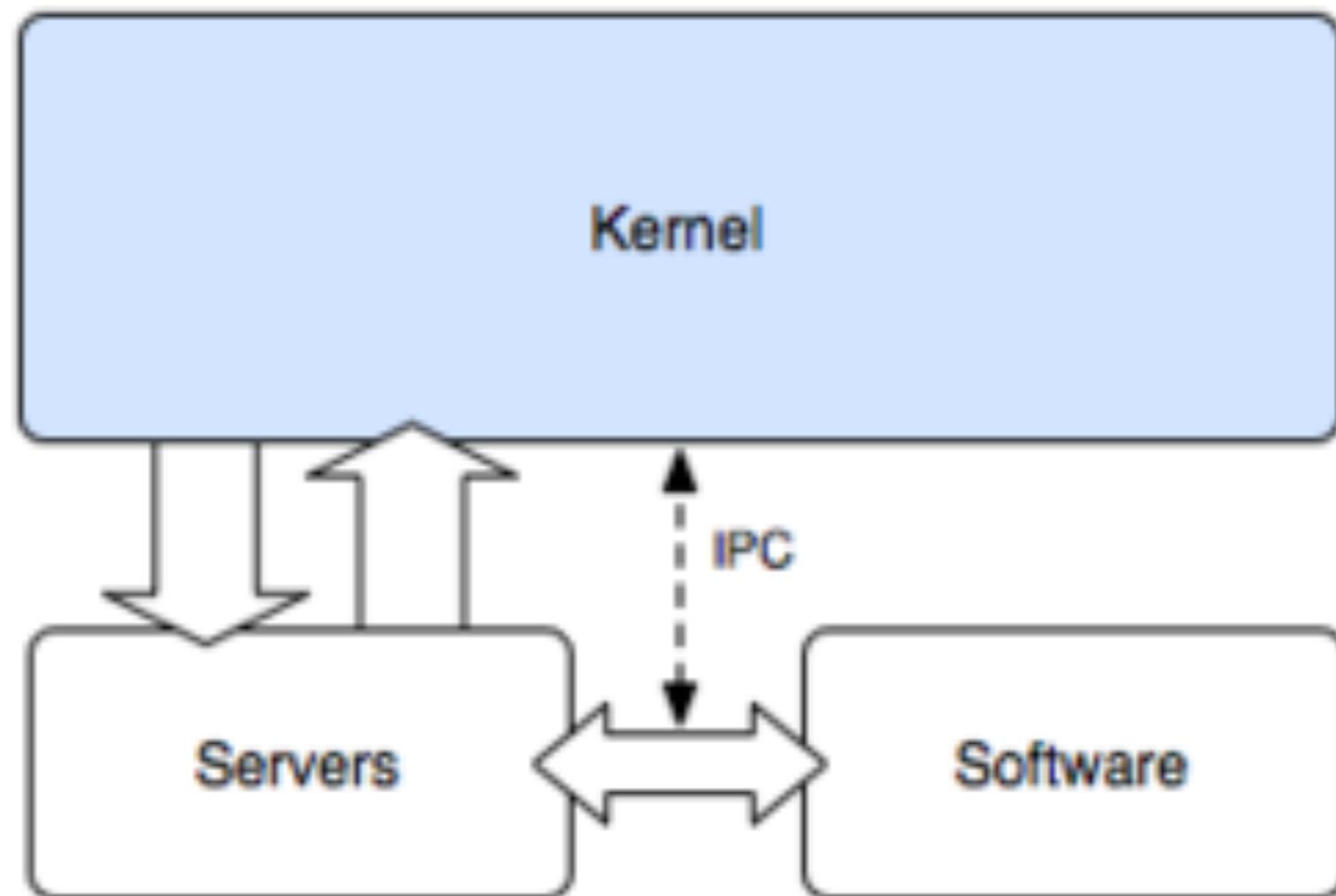
Oggi

Linux è una reale alternativa al mondo Microsoft e Unix, si ritrova milioni di utenti, migliaia di sviluppatori e un mercato in espansione. E' presente in sistemi integrati, è usato per il controllo di dispositivi robotizzati e ha volato a bordo dello shuttle, praticamente gira su oggetti elettronici di tutti i tipi, dai palmari alle workstation Alpha, risultando l'OS in assoluto più soggetto a porting

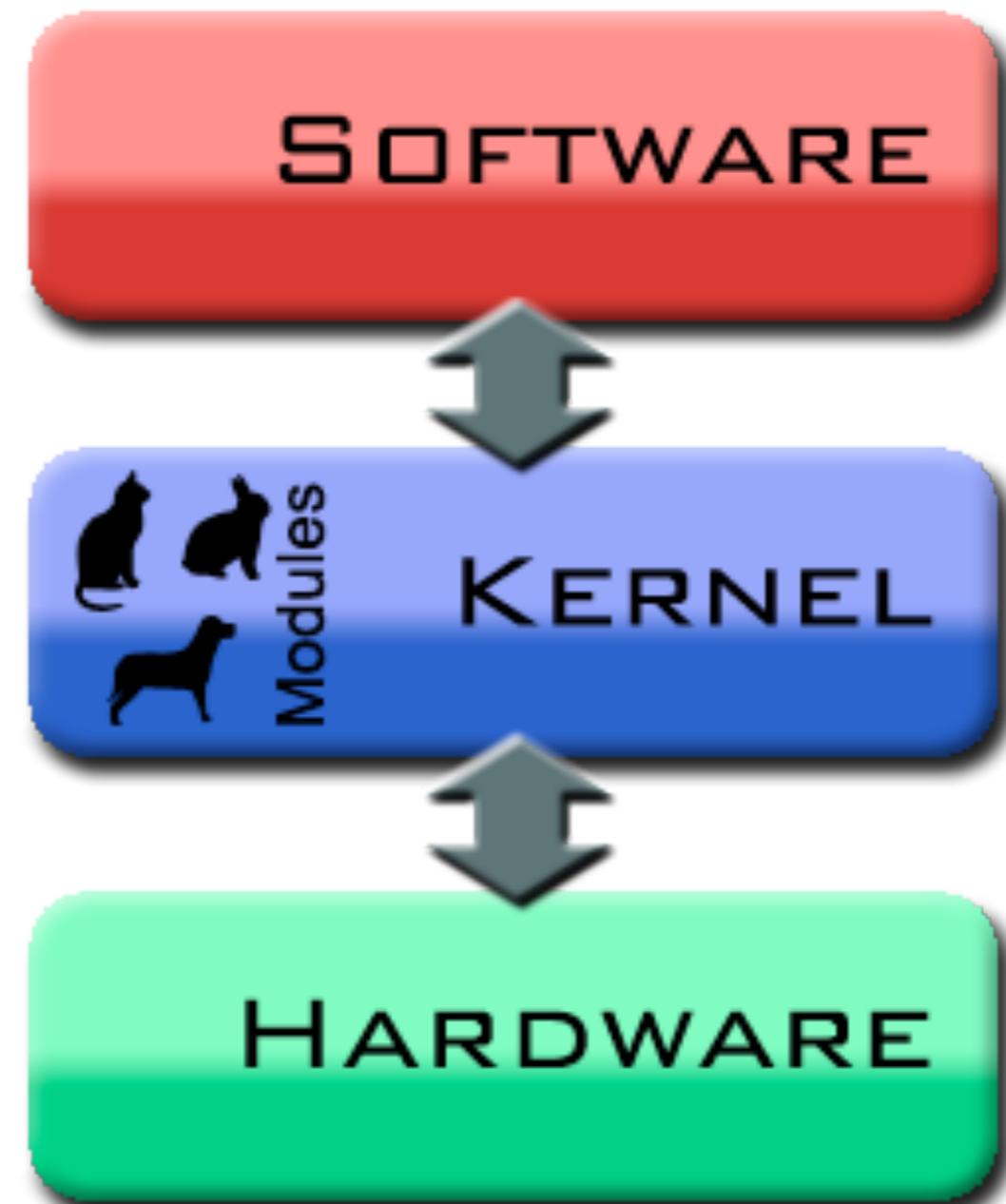
Kernel

- ◆ Per **kernel** si intende il **cuore** di un sistema operativo.
 - ▶ Il **codice** che gestisce le risorse presenti sul sistema e le rende disponibili alle applicazioni.
- ◆ Il kernel si occupa principalmente di **gestire**:
 - ✓ le **comunicazioni** con l'hardware del sistema (device driver)
 - ✓ i **file system** e la **memoria**
 - ✓ l'**accesso** alle risorse da parte dei processi (le applicazioni eseguite sul sistema)

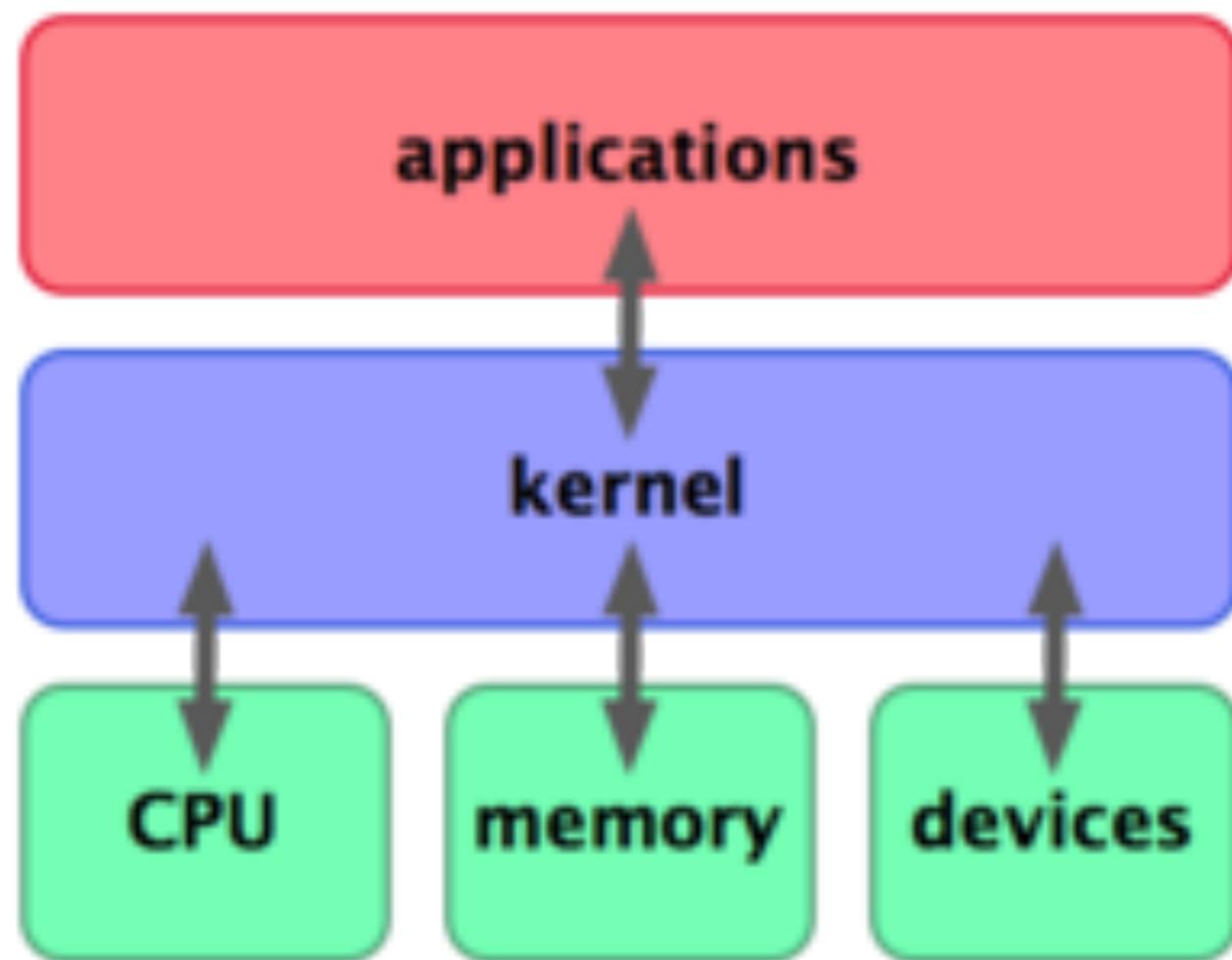
Kernel



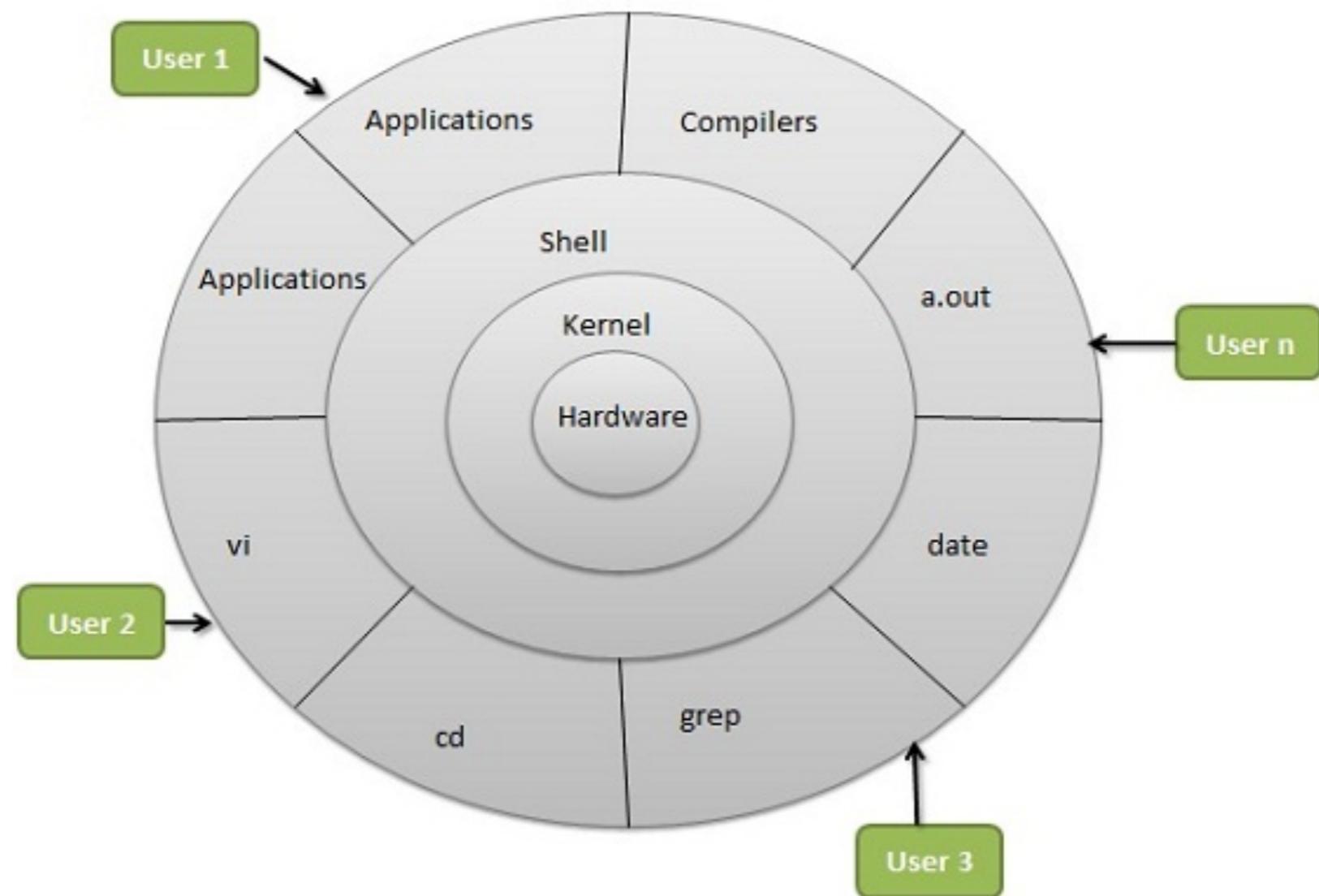
Kernel



Kernel



Kernel



Versioni del Kernel

- ◆ *Le versioni del kernel Linux sono identificate con **numeri** dal significato ben preciso. Per esempio il kernel **2.6.22** uno delle ultime versioni rilasciate nel 2007 ha:*

2	Major Number
6	Minor Number
22	Revision (patch)

Versioni del Kernel

- ✿ *Il Major Number identifica il valore più alto della revisione del kernel.*
- ✿ *Il rilascio di un kernel con un Major Number successivo rappresenta un'evoluzione “notevole” in termini di funzionalità e/o di architettura rispetto il precedente.*
- ✿ *I kernel della serie 1.x sono ormai piuttosto vecchi e non più usati.*



Versioni del Kernel

- ✿ Il **Minor Number** indica se il kernel è in versione **stable** o in versione **development**.
- ✿ Se **Pari** il kernel viene considerato **stable** e pronto per sistemi in produzione,
- ✿ Se **Dispari** si considera in **development** da usare con cautela o per sperimentazione. Le release stable sono sempre figlie delle devel precedenti.
- ✿ Per esempio il kernel stable è la versione 2.6.22, mentre quello in sviluppo è la 2.7.x (da cui deriverà la 2.8.x o direttamente la 3.0.x).
- ✿ Solitamente nei kernel **stable** si tende a fare **maintenance** e ad implementare solo le **features strettamente necessarie**, lasciando a quello in **development** lo sviluppo di **nuove funzionalità**.



Versioni del Kernel

- ◆ *Il Revision Number indica la revisione (patch) corrente.*
- ◆ *E' un numero progressivo che parte da 0. Tra due revisioni successive possono passare da pochi giorni a varie settimane.*
- ◆ *Esistono inoltre varie patch temporanee, anche non di Torwalds stesso (comuni sono le -ac patch, di Alan Cox) che rappresentano stadi intermedi **STABILI** prima della release di una revisione definitiva. Ha senso utilizzarle subito solo in caso di utilizzo di kernel con gravi problemi di sicurezza o stabilità.*



Principali distribuzioni Linux

Distribuzione	Note	Web
RedHat	Orientata ad applicazioni di tipo Enterprise	www.redhat.com
Fedora	Versione Open Source derivata da RedHat	fedoraproject.org
Mandriva	ex Mandrake ha dovuto cambiare nome per problemi di copyright	www.mandriva.org
SUSE	Acquisita da Novell. Tramite Novell Open enterprise Server supporta i servizi Netware	www.suse.com
Slackware	Una delle prime amata dai “puristi”	www.slackware.co
Debian	Storica	www.debian.com
Ubuntu	Preinstallata	www.ubuntu.com
Kubuntu	KDE come Desktop Environment	www.kubuntu.com
Linspire	Pubblicizzata come la più semplice da utilizzare	www.linspire.com

Principali distribuzioni Linux

- ✿ *E TANTE TANTE TANTE ALTRE*
- ✿ *Andate su <http://futurist.se/gldt/>*

Confronto



Linux

vs



Macintosh

PICODRILL.COM

vs



Confronto



Linux

“Linux users have the benefit of having low resource requirements as well. Linux can be installed on a PC just having the bare minimum resources. In fact, Linux is so versatile that almost any type of computer and console can probably support it, except Macintosh computers, which come preloaded with Mac OS X.”

Confronto



Macintosh

IPV2011.COM

“Macintosh computers come preloaded with the Mac OS X, and this OS can only be installed on the Mac. These computers are considerably more expensive than other PCs. Even an entry level Mac can cost almost twice the amount required to assemble a Windows PC.”

Confronto



“The Windows operating system is pretty versatile, and can be installed on PCs having variable amounts of resources. There are versions of Windows that can be installed on PCs having as little resource as 233 MHz processor and 64 MB RAM”

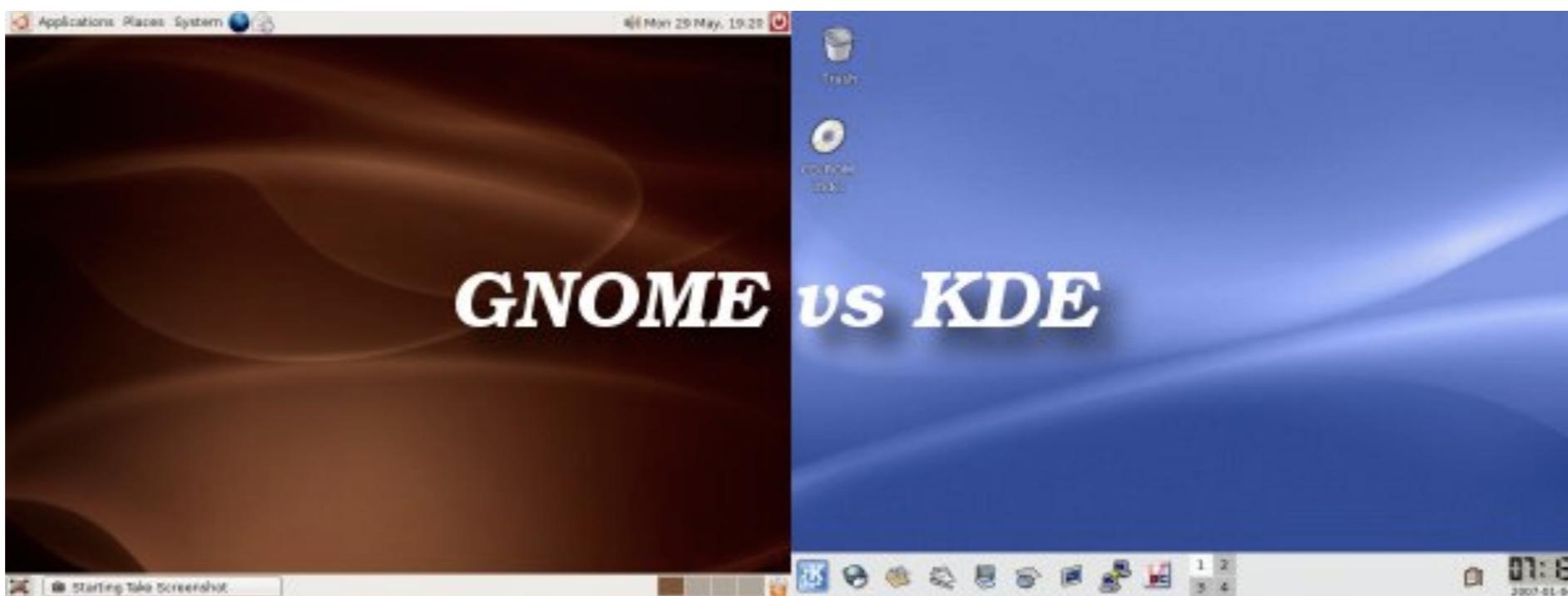
Utenti

- ◆ In ambienti Unix e quindi anche su Linux esistono **differenze** fra i **vari utenti**, definite dai **permessi** e dall'accesso ai **file** e **comandi** che un'utente può lanciare.
- ◆ E' convenzione che i **semplici utenti** possono **scrivere, leggere e modificare file** solo all'interno del loro ambiente (**home**) e lanciare semplici comandi che non influiscono sulla configurazione del sistema.
- ◆ Per poter accedere completamente alle risorse del sistema bisogna accedere al sistema come **superuser** ovvero come utente **root**.
- ◆ In fase di installazione di una macchina Linux si consiglia di scegliere una password di root piuttosto complicata (ma ricordabile!) e di creare immediatamente un normale utente con il quale operare per tutte le attività di tipo **NON AMMINISTRATIVO**.



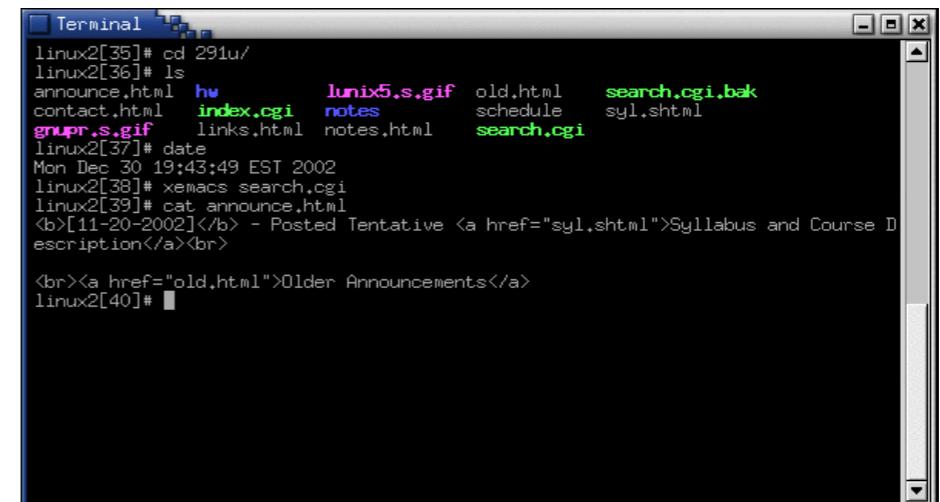
Interfacce UNIX

- ◆ ***Graphical User Interfaces (GUIs)***: si può essere seduti davanti a un PC e tutti i comandi vengono poi gestiti localmente su quel computer. Quando si accede in questo modo si dispone di un ambiente GUI completo



Interfacce UNIX

- ◆ **Command Line Interface:** è possibile connettersi in remoto ad un server UNIX (sia da casa o presso laboratori). Questo è spesso il modo in cui vengono gestiti i progetti da sviluppare in quanto sono classificati sui vari server. Quando si accede in questo modo si dispone di una linea di comando (o ambiente testuale). È anche possibile aprire una riga di comando su macchine locali.



The screenshot shows a terminal window titled "Terminal". The session starts with the user navigating to a directory named "291u" and listing its contents. The user then runs the "date" command to check the current date and time. Finally, the user opens a text editor named "xemacs" and edits a file named "search.cgi". The terminal window has a dark background with light-colored text and uses color-coded syntax highlighting for file extensions like .html, .gif, and .cgi.

```
linux2[35]# cd 291u/
linux2[36]# ls
announce.html    hw      linux5.s.gif  old.html   search.cgi.bak
contact.html     index.cgi notes    schedule   syl.shtml
gnupr.s.gif      links.html notes.html  search.cgi
linux2[37]# date
Mon Dec 30 19:43:49 EST 2002
linux2[38]# xemacs search.cgi
linux2[39]# cat announce.html
<b>[11-20-2002]</b> - Posted Tentative <a href="syl.shtml">Syllabus and Course Description</a><br>
<br><a href="old.html">Older Announcements</a>
linux2[40]#
```

Programming tools and Utilities

■ Text Editors

- Xemacs
- Emacs
- Pico
- vi

■ Compilers

- C compiler - gcc
- C++ compiler - g++
- Java compiler & Java Virtual Machine - javac & java

■ Debuggers

- C / C++ debugger - gdb

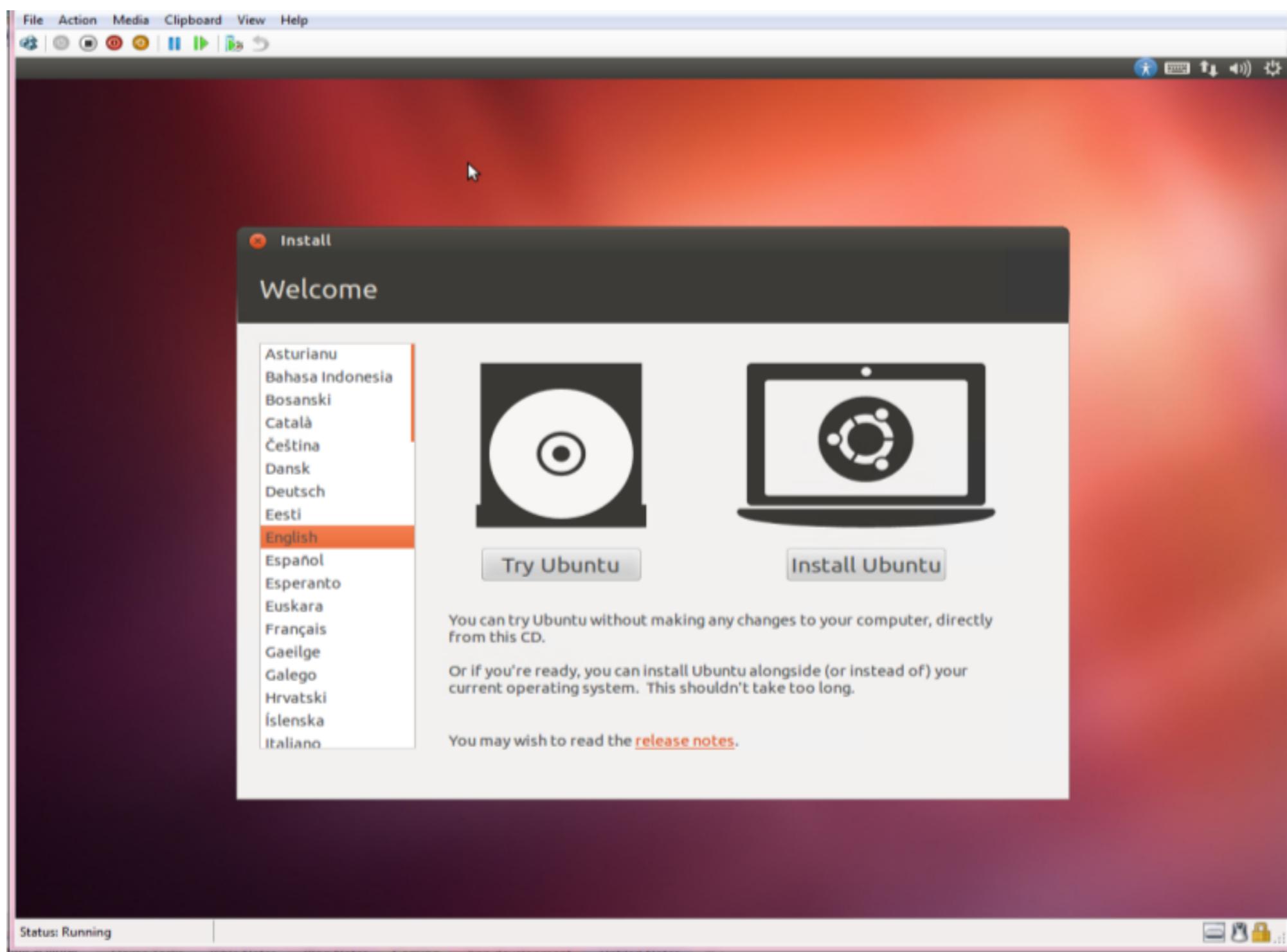
■ Interpreters

- Perl - perl
- Tcl/Tk - tcl & wish

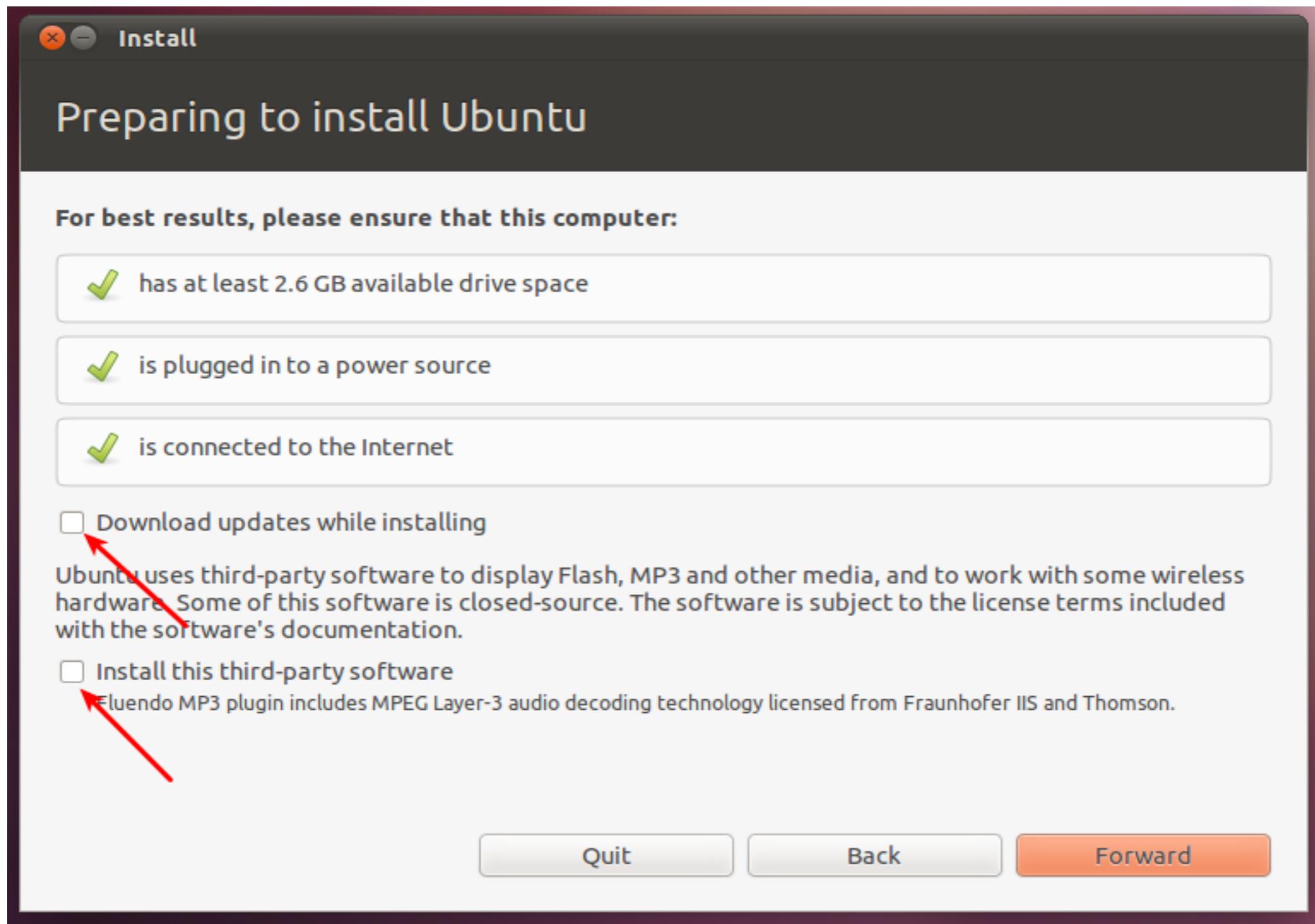
■ Miscellaneous

- Web Browsers - Mozilla, Netscape, Firefox, and Lynx (lynx is text based)
- Instant Messengers - Gaim
- Email - Netscape is there, but we will learn Pine

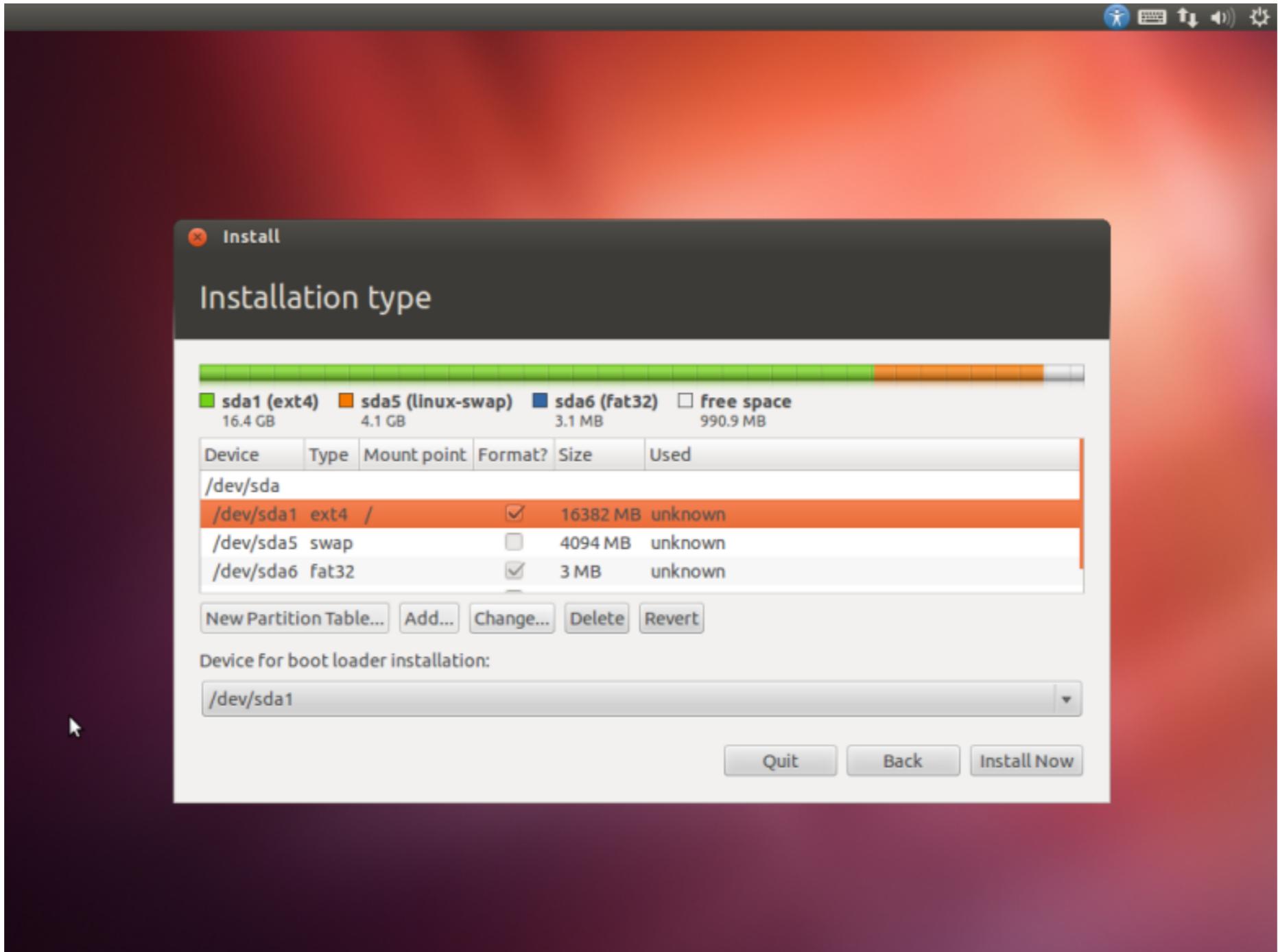
Istallando Linux (ad esempio Ubuntu)



Istallando Linux (ad esempio Ubuntu)



Istallando Linux (ad esempio Ubuntu)



Installando Linux (ad esempio Ubuntu)

Install

Who are you?

Your name:

Your computer's name:
The name it uses when it talks to other computers.

Pick a username:

Choose a password: Fair password

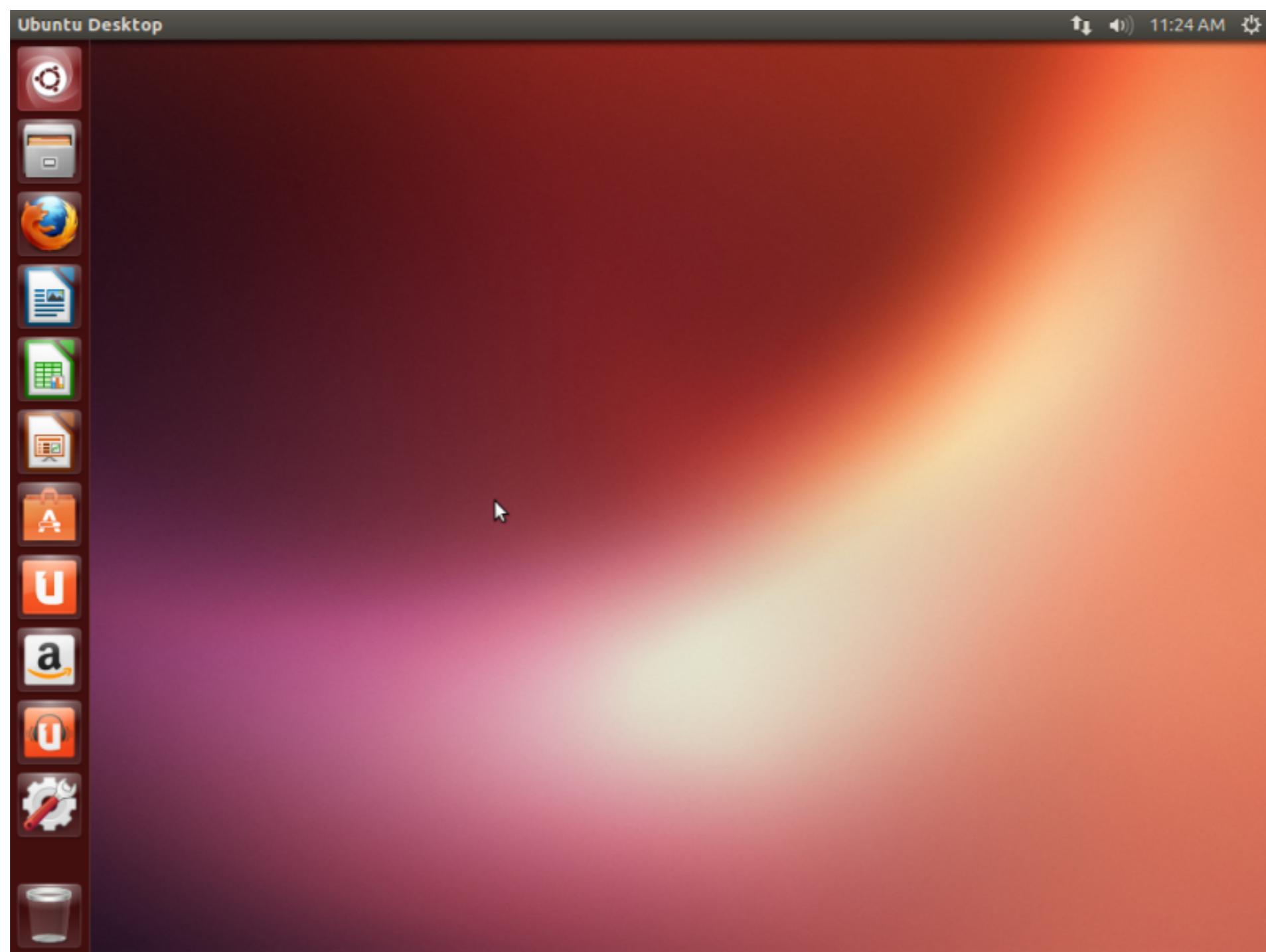
Confirm your password:

Log in automatically
 Require my password to log in
 Encrypt my home folder

[Back](#) [Continue](#)

► Almost finished copying files...

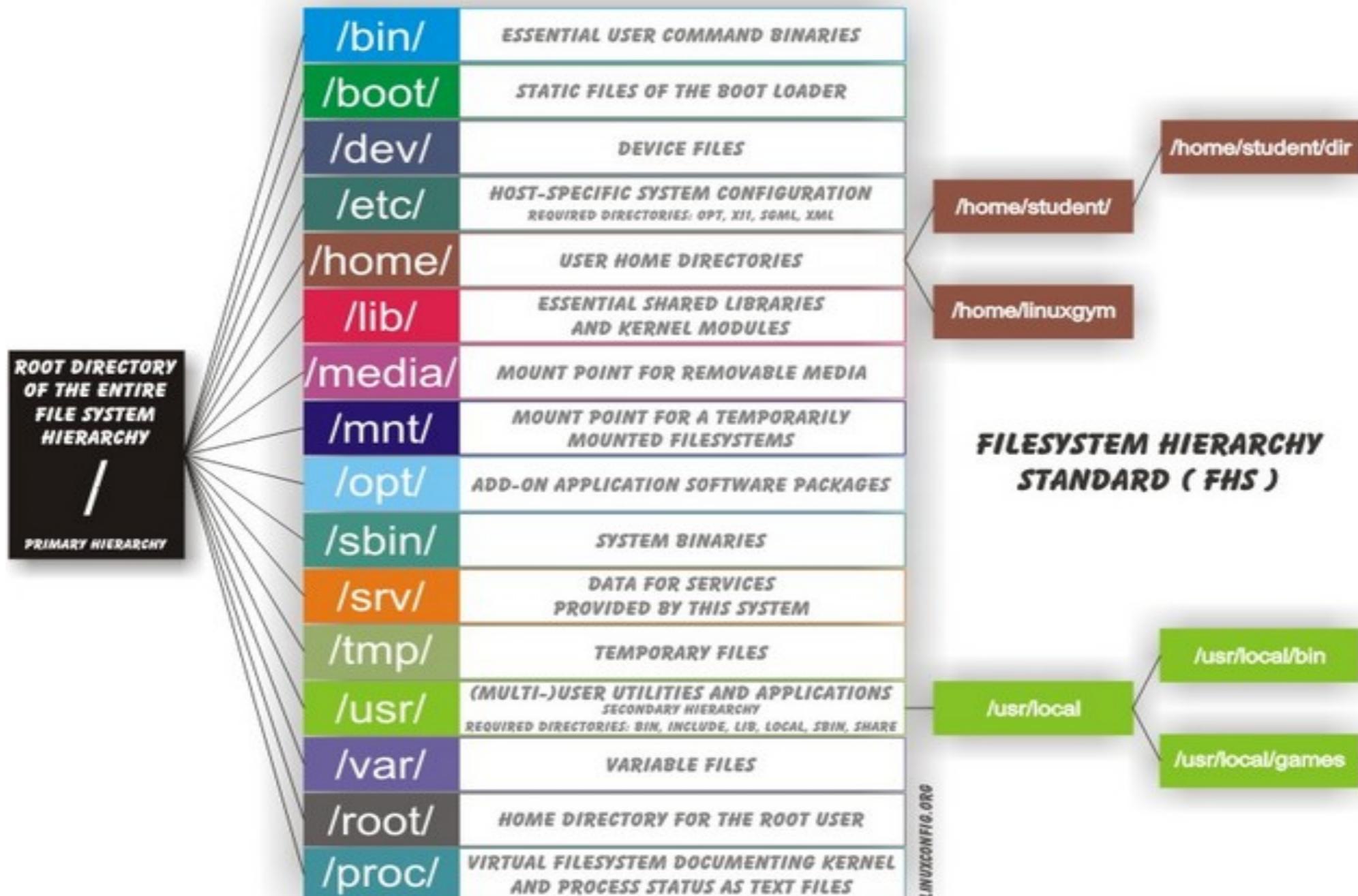
Installando Linux (ad esempio Ubuntu)



File system

- ◆ Per **file system** si intende l'astrazione (metodi e protocolli) con cui si organizzano i file su un supporto fisico ad accesso casuale (floppy, cdrom, hard-disk, ecc)
- ◆ Le informazioni riguardanti un oggetto (file o directory) di un file system sono contenute in un **inode**, che viene identificato da un numero progressivo e descrive le caratteristiche base di un oggetto: permessi, data di modifica, tipo, posizione ecc.
- ◆ Un sistema Linux, come ogni sistema Unix, ha una **directory principale**, chiamata **root** ed indicata con / sotto la quale si trovano **TUTTE** le altre directory e **TUTTI** gli altri file system eventualmente montati sul sistema (floppy, cdrom, condivisione di rete ecc.).
- ◆ Il principio è radicalmente diverso da quello presente nel mondo Windows, dove ogni device o risorsa ha un suo nome o lettera identificativa al cui interno si trovano le directory del relativo file system.
- ◆ Il file system usato in Linux è l' **ext2**. Dal kernel 2.4.x è disponibile un'evoluzione dell'**ext2**, l'**ext3** che ha il vantaggio di essere un journal file system

File system



File system

/ Radice (root)

/root Home dell'utente root

/boot Contiene tutte le immagini del kernel e file indispensabili al bootstrap

/etc Contiene i file di configurazione del sistema e dei programmi installati

/home Contiene le home degli utenti tranne quella di root

/usr Contiene binari, documentazione, librerie e sorgenti della maggior parte dei programmi (e i sorgenti del kernel)

/var File contenenti informazioni dinamiche (log, pid file, directory di spool).

/proc File system virtuale. Contiene, come se fossero file e directory, dati dinamici sul sistema e sui processi

/dev Contiene i file per la gestione dei dispositivi sia a blocchi (hdd) che a carattere (tty) oltre a file speciali (/dev/null)

File system

/sbin	Contiene comandi e programmi riservati a root (altri comandi sono in /usr/sbin/)
/bin	Contiene comandi e programmi base per tutti gli utenti (altri comandi sono in /usr/bin/)
/lib	File delle <i>librerie</i> di sistema utilizzate dai programmi
/tmp	Contiene i file temporanei
/usr/tmp	Altra directory che contiene file temporanei
/usr/doc	Documentazione sul sistema
/mnt	Directory sotto la quale vengono montati altri file system (floppy, cdrom, chiavi USB, partizioni NTFS, ecc)
/media	Come /mnt si trova in diverse distribuzioni, ma non fa parte della struttura standard (è usata dal demone automount).

Utenti e gruppi

- ◆ **UTENTI:** Si possono creare più account che vengono memorizzati nel file **/etc/passwd**. Ogni riga del file passwd è del tipo:

*username:passwd:**UID:GID:user_data:home_dir:shell***

UID =	User IDentifier è un valore numerico univoco
GID =	Group Identifier è un valore numerico univoco
user_data =	Nome e Cognome dell'utente, telefono, ecc...
home_dir =	directory di lavoro dell'utente
shell =	shell che utilizzerà l'utente

es: root:passwd:0:0:root:/root:/bin/bash

Utenti e gruppi

- ◆ **GRUPPI:** *Linux gestisce anche gruppi di utenti, per condividere file e per favorire la collaborazione. Ogni utente deve appartenere ad almeno un gruppo.*
- ◆ *I gruppi sono definiti nel file **/etc/group**, ogni riga ha la forma del tipo:*

group_name:passwd:GID:users_list

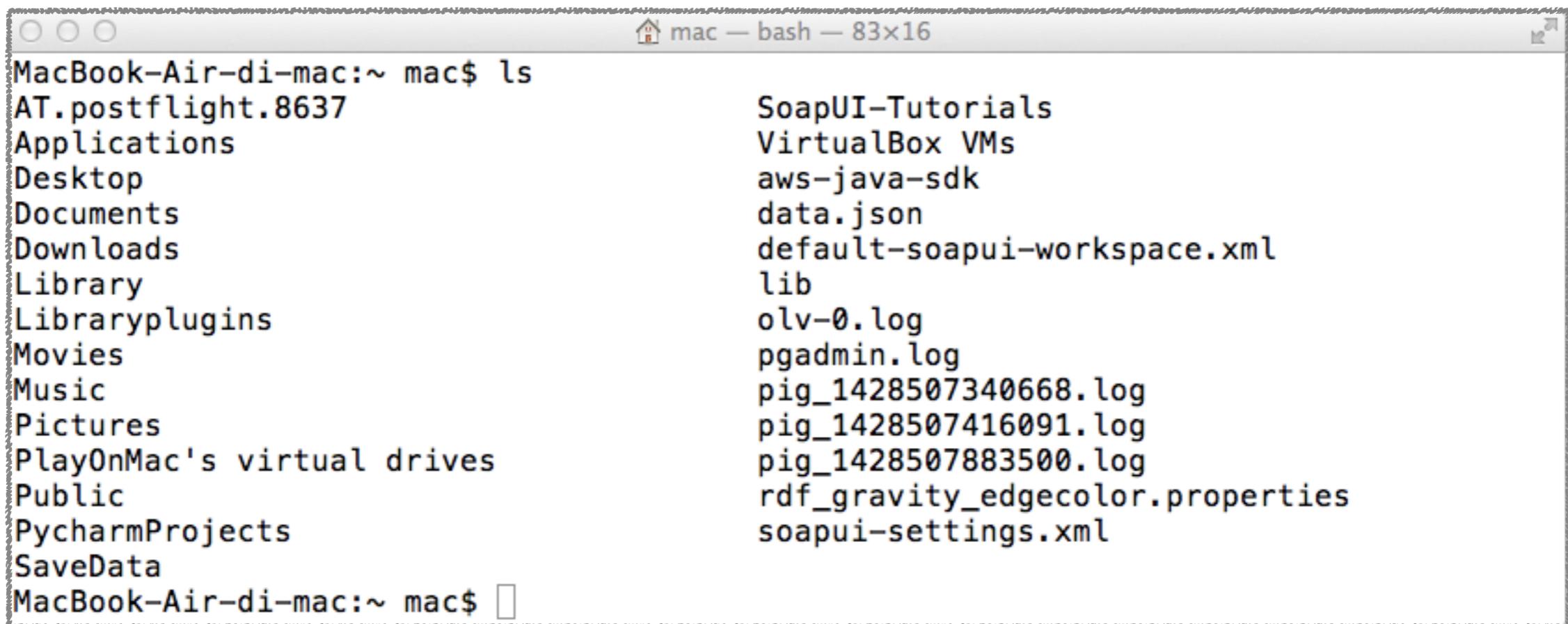
es: adm:passwd:4:root,adm,daemon

Comandi di uso frequente

```
Last login: Sat Oct 10 17:45:00 on ttys000
MacBook-Air-di-mac:~ mac$
```

Comandi di uso frequente

- ◆ *ls: visualizza il contenuto della directory corrente*

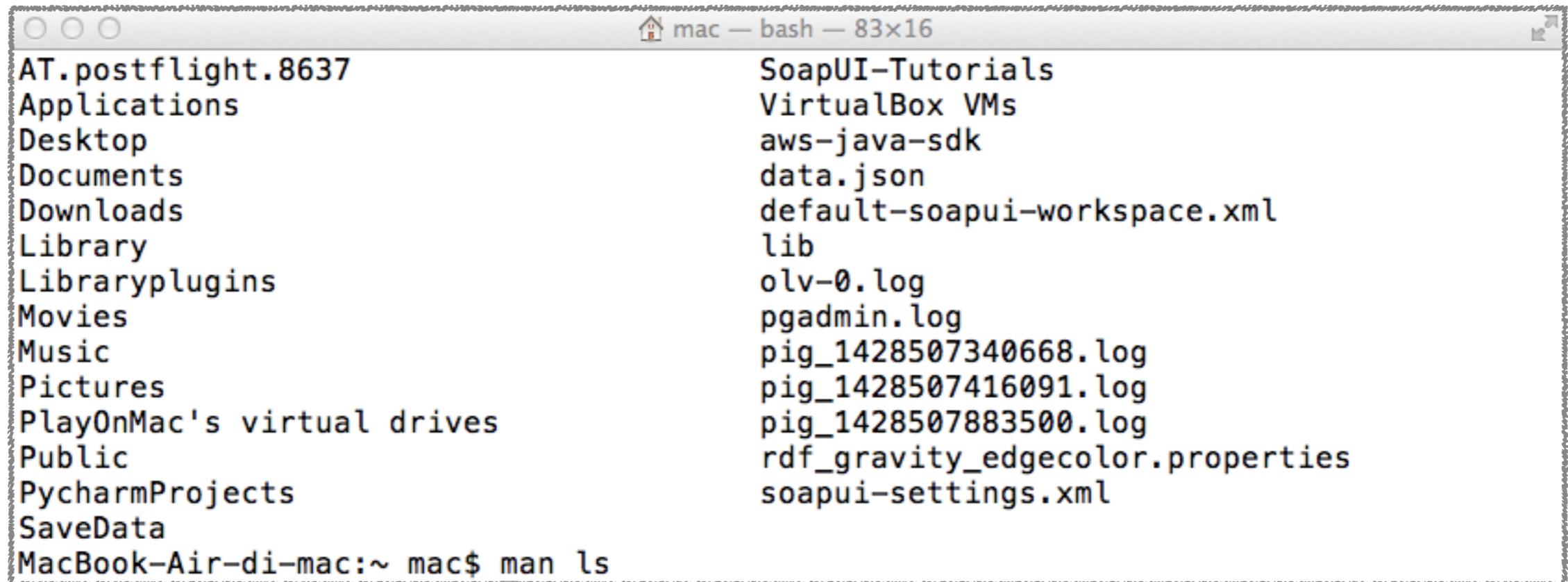


The screenshot shows a terminal window titled "mac — bash — 83x16". The command "ls" is run, displaying the contents of the current directory. The output is as follows:

```
MacBook-Air-di-mac:~ mac$ ls
AT.postflight.8637
Applications
Desktop
Documents
Downloads
Library
Libraryplugins
Movies
Music
Pictures
PlayOnMac's virtual drives
Public
PycharmProjects
SaveData
MacBook-Air-di-mac:~ mac$ SoapUI-Tutorials
VirtualBox VMs
aws-java-sdk
data.json
default-soapui-workspace.xml
lib
olv-0.log
pgadmin.log
pig_1428507340668.log
pig_1428507416091.log
pig_1428507883500.log
rdf_gravity_edgecolor.properties
soapui-settings.xml
```

Comandi di uso frequente

- ◆ *man [comando]*: visualizza la pagina del manuale relativa a “comando”



```
mac — bash — 83x16
AT.postflight.8637
Applications
Desktop
Documents
Downloads
Library
Libraryplugins
Movies
Music
Pictures
PlayOnMac's virtual drives
Public
PycharmProjects
SaveData
MacBook-Air-di-mac:~ mac$ man ls
SoapUI-Tutorials
VirtualBox VMs
aws-java-sdk
data.json
default-soapui-workspace.xml
lib
olv-0.log
pgadmin.log
pig_1428507340668.log
pig_1428507416091.log
pig_1428507883500.log
rdf_gravity_edgecolor.properties
soapui-settings.xml
```

Comandi di uso frequente

- ◆ *man [comando]*: visualizza la pagina del manuale relativa a “comando”

```
mac — less — 83x27
LS(1)                               BSD General Commands Manual      LS(1)

NAME
  ls -- list directory contents

SYNOPSIS
  ls [-ABCDEFGHJKLMNOPRSTUW@abcdefghijklmnopqrstuvwxyz] [file ...]

DESCRIPTION
  For each operand that names a file of a type other than directory, ls
  displays its name as well as any requested, associated information.
  For each operand that names a file of type directory, ls displays the
  names of files contained within that directory, as well as any
  requested, associated information.

  If no operands are given, the contents of the current directory are
  displayed.  If more than one operand is given, non-directory operands
  are displayed first; directory and non-directory operands are sorted
  separately and in lexicographical order.

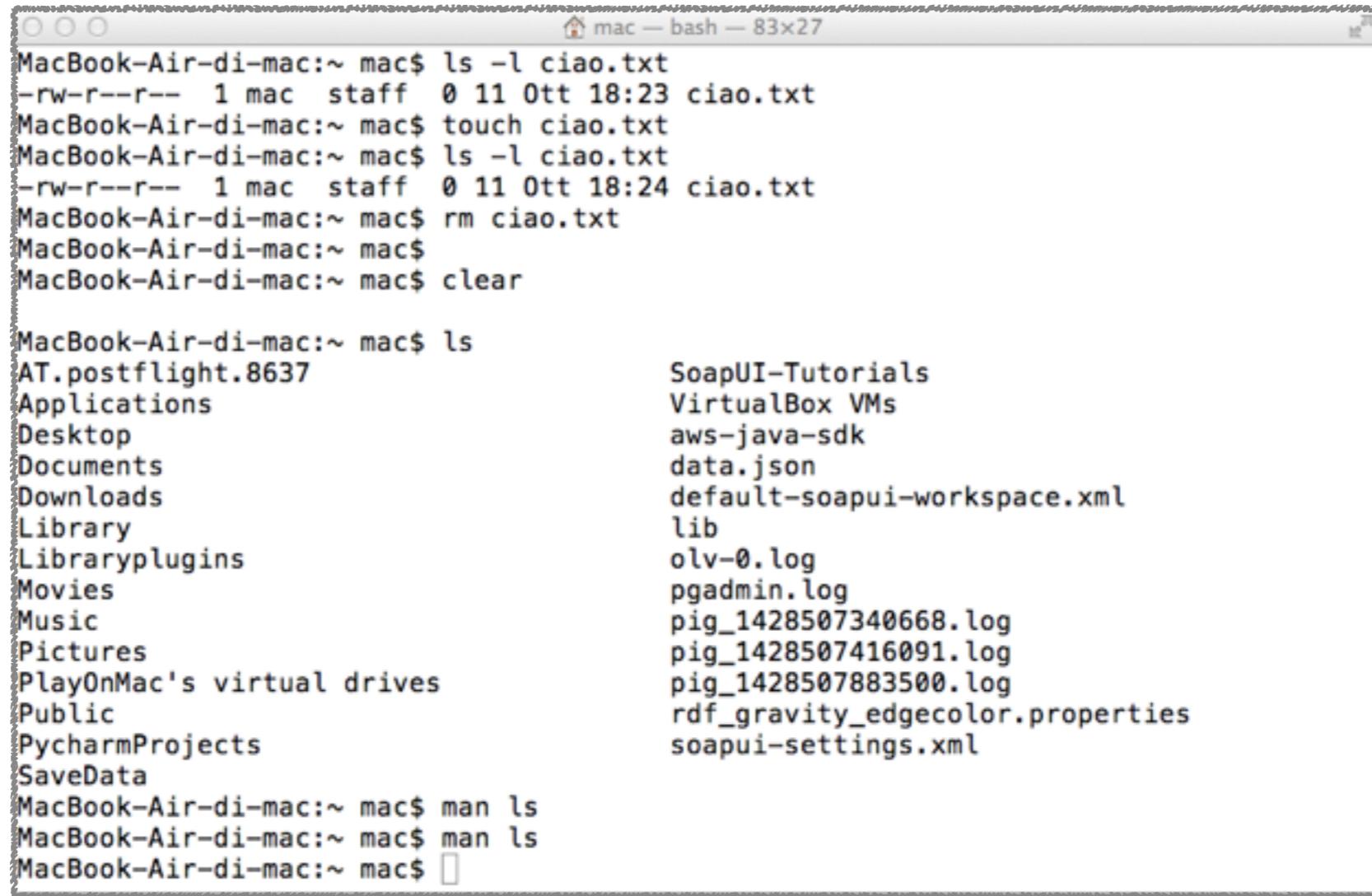
  The following options are available:

    -@      Display extended attribute keys and sizes in long (-l) out-
            put.

:
```

Comandi di uso frequente

- ◆ *man [comando]*: visualizza la pagina del manuale relativa a “comando”; digitare **q** per tornare al terminale



The screenshot shows a terminal window titled "mac — bash — 83x27". The window displays the following command history and file listing:

```
MacBook-Air-di-mac:~ mac$ ls -l ciao.txt
-rw-r--r-- 1 mac staff 0 11 Ott 18:23 ciao.txt
MacBook-Air-di-mac:~ mac$ touch ciao.txt
MacBook-Air-di-mac:~ mac$ ls -l ciao.txt
-rw-r--r-- 1 mac staff 0 11 Ott 18:24 ciao.txt
MacBook-Air-di-mac:~ mac$ rm ciao.txt
MacBook-Air-di-mac:~ mac$
MacBook-Air-di-mac:~ mac$ clear

MacBook-Air-di-mac:~ mac$ ls
AT.postflight.8637
Applications
Desktop
Documents
Downloads
Library
Libraryplugins
Movies
Music
Pictures
PlayOnMac's virtual drives
Public
PycharmProjects
SaveData
MacBook-Air-di-mac:~ mac$ man ls
MacBook-Air-di-mac:~ mac$ man ls
MacBook-Air-di-mac:~ mac$
```

On the right side of the terminal window, there is a vertical list of files and folders:

- SoapUI-Tutorials
- VirtualBox VMs
- aws-java-sdk
- data.json
- default-soapui-workspace.xml
- lib
- olv-0.log
- pgadmin.log
- pig_1428507340668.log
- pig_1428507416091.log
- pig_1428507883500.log
- rdf_gravity_edgecolor.properties
- soapui-settings.xml

Comandi di uso frequente

- ✿ ***touch nome file [INVIO]***: aggiorna la data e l'ora di un file, se il file non esiste crea un file vuoto
- ✿ ***mkdir nome directory [INVIO]***: crea una directory
- ✿ ***mkdir -p work/completed/2001 [INVIO]***: crea la directory 2001 e le directory superiori se non esistono.
- ✿ ***cd nome directory [INVIO]***: cambia directory
- ✿ ***cd [INVIO]***: rende come corrente la directory home
- ✿ ***pwd [INVIO]***: visualizza il PATH in cui ci si trova
- ✿ ***ls -R [INVIO]***: visualizza il contenuto della directory corrente e delle sotto directory.

Comandi di uso frequente

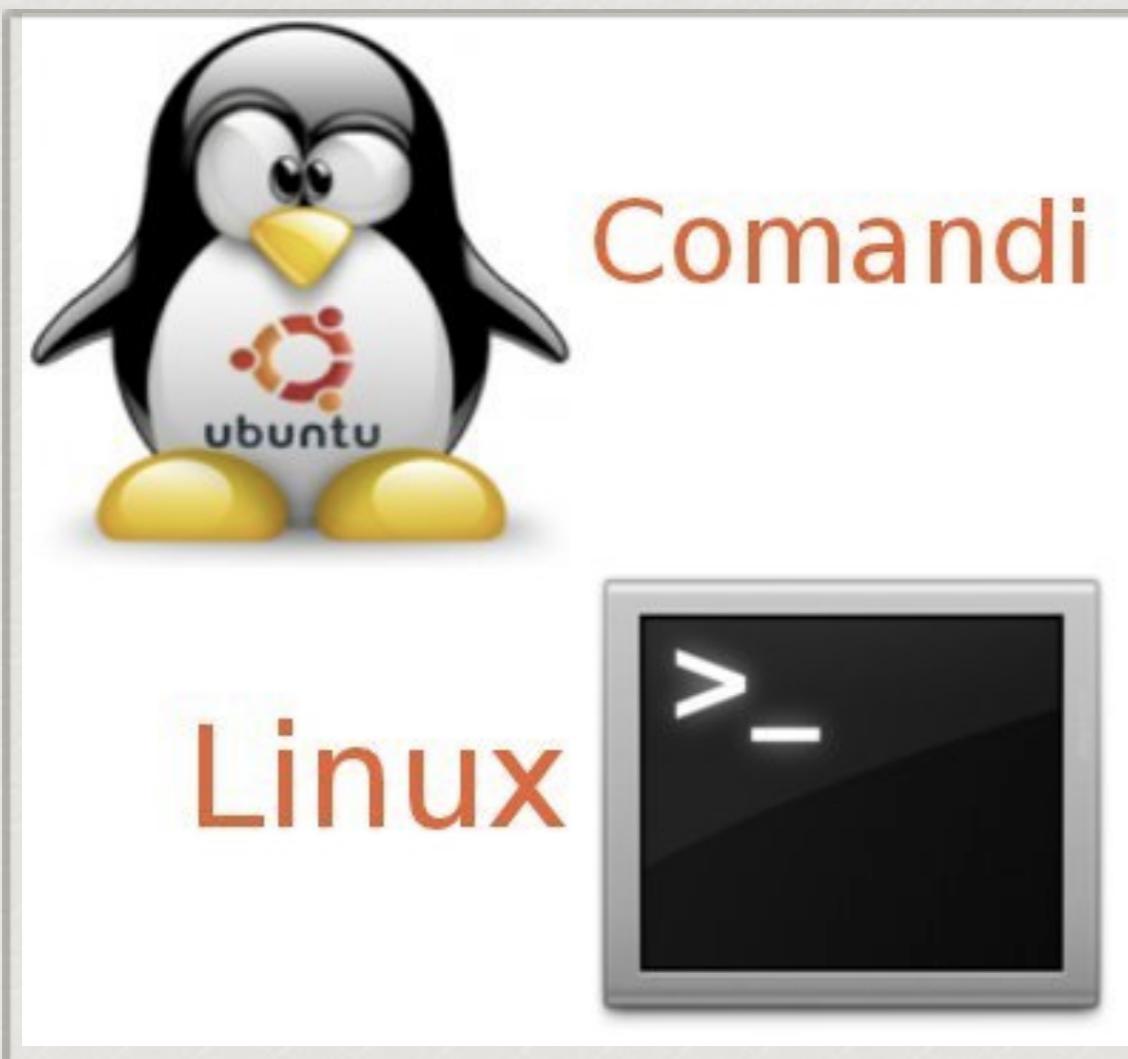
- ◆ ***cp file_sorgente file_destinazione [INVIO]***: copia il *file_sorgente* creando il nuovo *file_destinazione*
- ◆ ***mv file_sorgente file_°destinazione [INVIO]***: rinomina o sposta il *file_sorgente* in *file_destinazione*
- ◆ ***ls -a***: visualizza i file nascosti quelli in cui il primo carattere del nome è un punto (.)

E' possibile utilizzare dei caratteri speciali all'interno del nome dei file:

- * Corrisponde ad una serie di nessuno o più caratteri.
* da solo indica tutti i file
.txt indica i soli file con estensione txt
- ? Simile ad * ma sostituisce un generico carattere
cas? equivale a tutti i file di 4 caratteri la aventi le prime tre uguali a cas
- ~ rappresenta la home directory dell'utente
~ = /home/pippo
cd ~/prova equivale al comando: cd /home/pippo/prova

Comandi di uso frequente

- ✿ ***find percorso -name nome_file [INVIO]***: visualizza tutti i file che si trovano sotto “percorso” aventi “nome_file”.
- ✿ ***find /tmp -size +1000ok [INVIO]***: visualizza i file memorizzati sotto /tmp aventi dimensione maggiore di 1000ok
- ✿ ***find /home -user topolino [INVIO]***: visualizza i file memorizzati sotto /home di proprietà di topolino.
- ✿ ***which programma [INVIO]***: visualizza il percorso completo di dove si trova il comando
- ✿ ***grep nome [INVIO]***: filtra le righe che contengono la parola “nome”
- ✿ ***more programma [INVIO]***: interrompe la visualizzazione quando si riempie lo schermo ed attende la pressione di un tasto per proseguire
- ✿ ***clear [INVIO]***: ripulisce il terminale



```
Current conditions at Pescara, Italy (LIBP) 42.26N 014.12E ILM (LIBP)
Last updated Feb 10, 2012 - 02:50 PM EST / 2012.02.10 1950 UTC
Temperature: 1 C
Relative Humidity: 80%
Wind: from the W (270 degrees) at 15 MPH (13 KT) gusting to 45 KPH
Weather: light snow grains
Sky conditions: overcast
Su Mo Tu We Th Fr Sa      Su Mo Tu We Th Fr Sa
feb 29e 30 31 01 02 03 04 | mar 04 05 06 07 08 09 10
  05 06 07 08 09 10e 11 | r11e 12 13 14 15 16 17
  12 13 14 15 16 17 18 | 18 19 20 21 22 23 24
  19 20 21e 22e 23 24 25 | 25 26 27 28 29 30 31
mar 26 27 28 29 01 02 03 | apri01e 02 03 04 05 06e 07
silvio@Stan:~$ cd Video
silvio@Stan:~/Video$ movgrab http://vimeo.com/27998081
Formats available for this Movie: flv
Selected format item:flv
Progress: 61.47% 15.4M of 25.1M 693.6k/s
```

