

**Latinamerican School for  
Computational Materials  
Science.**



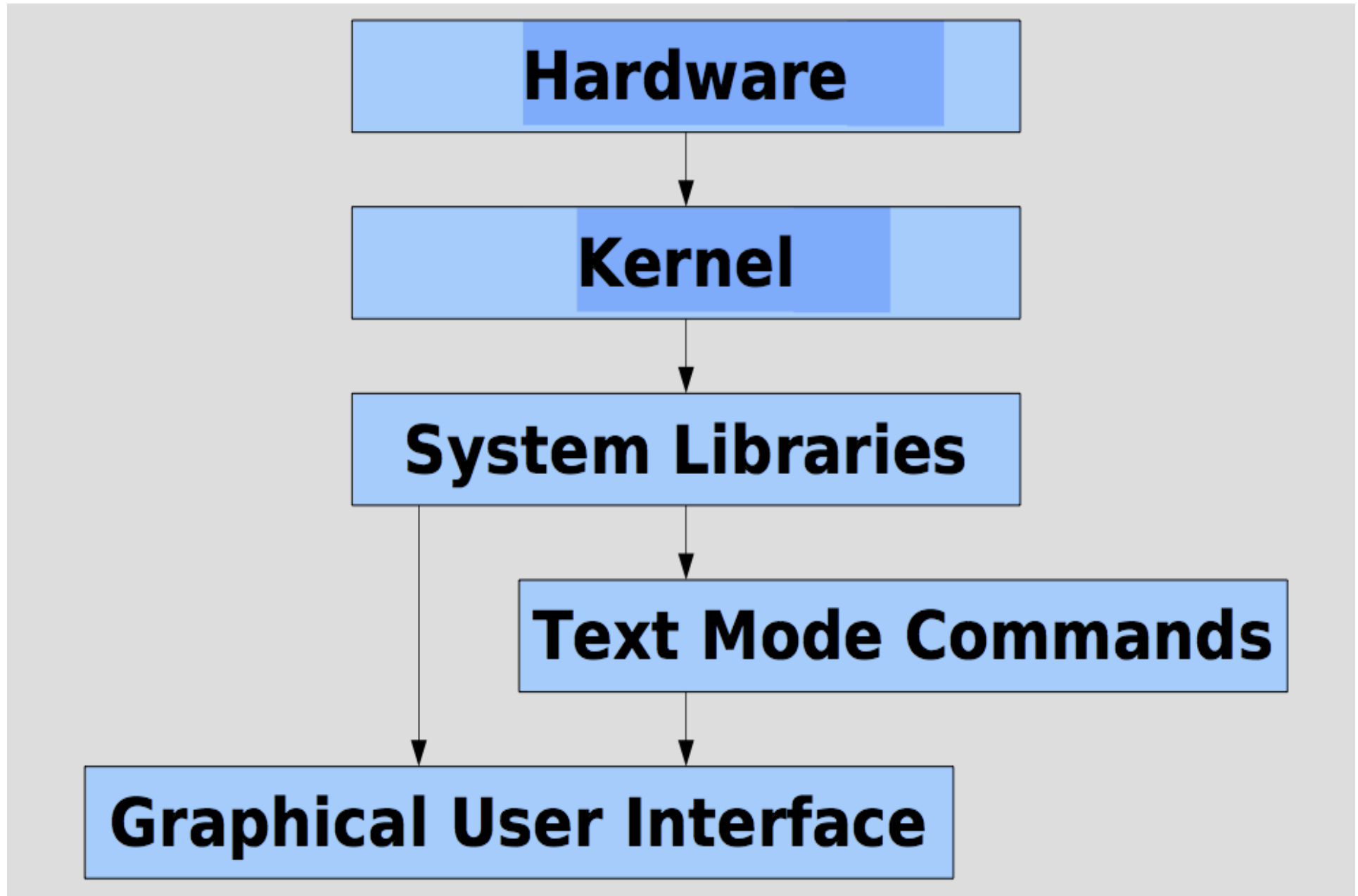
# What is Linux ?

**Stefano Cozzini**

CNR-INFM DEMOCRITOS, Trieste



# Operating system overview



# Agenda

- Introduction: what is e-science ?
- High Performance Computing:
  - introduction/ concepts /definitions
- Parallel computers
- Clusters:
  - definitions and some other funny things
- Grids
- Wrap-up

# Agenda

- Introduction: what is e-science ?
- High Performance Computing:
  - introduction/ concepts /definitions
- Parallel computers
- Clusters:
  - definitions and some other funny things
- Grids
- Wrap-up

# some linux definition

- Linux
  - By itself is only the Linux kernel, low level interface between hardware and software
  - Ex.: 1.0.9, 1.2.24, 2.6.18, 2.6.25
- Linux Distribution:
  - Prepackaged collection of Applications bundled with a Linux kernel and matching libraries
  - Ex.: Fedora, Ubuntu, Debian, Red Hat, SuSE,...
- Linux Architecture:
  - Linux running a specific hardware in a specific mode:
  - Ex.: i386, x86\_64, ia64, alpha, ppc5, ...
  - Compiled software only runs on one architecture

# Linux is everywhere

- Desktop Computer
- Laptop Computer
- Computer Cluster
- Smartphones
- Appliances (firewalls, webserver, wireless router, digital video recorder)
- Supercomputers (lightweight compute kernel)
- iPod
- ...

# Linux vs. Windows vs. MacOS X

- All have very different kernels
- MacOS and Linux based on similar (Unix) concepts (BSD vs. Linux kernel); similar interface with system library (POSIX);
- Windows different.
- Graphical UI separate from text mode
- Cross platform programming through libraries that hide differences. (Ex. Qt, tcl), or text mode only (Fortran, ANSI-C/C++)
- GUIs can have multiple “appearances” and be configured to look similar to each other.

# File System

- Linux treats practically everything as file
- All “files” are organized in a global, hierarchical file system tree, starting at '/'.
  - Storage devices are “mounted” into the tree
  - “Raw” devices can be accessed through special files in /dev/
  - Kernel status and settings can be read and modified via pseudo file systems: /proc/, /sys/
  - Data on file systems can be differently organized: ext3, fat, tmpfs
  - Access to devices, files through permissions



# Multiuser OS

- Linux is by construction a multi-user OS
- Each user has a name and a number (uid)
- Each task or process has a unique ID (pid) and has to be owned by a user
- A user can be in one or more groups (gid)
- All files are owned by a user and a group
- For a user (or group) to read, write, or execute a file, permission flags are required.
- Permissions are given to “user”, “group”, or “other” and thus files can be “protected”
- The system user “root” has all permissions.

# memory management

- Kernel translates between physical location of data in memory and its logical address
- Each process has its own address space
- Physical memory has to be mapped to a logical address to be written to.
- Access to memory outside of the mapped area => Segmentation fault, Signal 11
- Physical memory can be in (fast) RAM or on a (slow) disk and the kernel memory management can “swap” them as needed.

=> swap partition.