

## Standard Unix & Implementazioni

## Standard Unix: ANSI

- ▶ American National Standards Institute: venditori + utenti
- ▶ Membro dell' International Organization for Standardization
- ▶ 1989: ANSI C per la standardizzazione del linguaggio C
  - ▶ portabilità di programmi C ad una grande varietà di S.O. e non solo per UNIX
  - ▶ Definisce non solo la semantica e la sintassi ma anche una libreria standard divisa in 15 aree (individuate dagli header, vedi fig. 2.1)

## Standard Unix: IEEE POSIX

- ▶ Institute of Electrical and Electronic Engineers propose una famiglia di standard
  - ▶ Portable Operating System Interface
  - ▶ Lo standard 1003.1 relativo a **interfacce** di s.o.: definizione di servizi che un s.o. deve fornire per essere POSIX COMPLIANT
    - Definisce una interfaccia non una implementazione
    - Non è fatta distinzione tra system call e funzioni di libreria
    - Non prevede la figura di "superuser", ma certe operazioni richiedono appropriati privilegi

## Standard Unix: XPG3

- ▶ X/Open: gruppo di venditori di Computer
- ▶ Hanno prodotto 7 volumi di una guida di portabilità
- ▶ X/Open Portability Guide, Issue 3 1989
- ▶ Il II vol. definisce interfacce per un s.o. Unix-like, a partire da IEEE 1003.1, ma con variazioni (e.g. msg in varie lingue)

## Implementazioni: SVR4

- ▶▶ System V Release 4 è stato prodotto dalla AT&T
- ▶▶ SVR4 è conforme a POSIX e XPG3

## Implementazioni: 4.3+BSD

- ▶▶ Berkeley Software Distributions ( sono distribuite da UCB)
- ▶▶ Conforme allo standard POSIX
- ▶▶ Benché fosse inizialmente legato a codice sorgente AT&T e quindi alle sue licenze, è stata creata una versione (free) molto interessante per PC (intel based) FreeBSD

## Implementazioni: Linux

- ▶▶ Il primo kernel sviluppato da Linus Torvalds nel 1991
- ▶▶ Conforme a POSIX, ma include anche la maggior parte di funzioni di SVR4 e 4.3BSD
- ▶▶ Disponibile su Intel, Compaq (ex Digital), Alpha, Sparc, McIntosh e Amiga
- ▶▶ Free

## Distribuzioni Linux

- ▶▶ Slackware
- ▶▶ Debian
- ▶▶ RedHat
- ▶▶ SuSe
- ▶▶ Mandrake

## Linux: dove?

- ▶ [www.linux.org](http://www.linux.org)
- ▶ [www.kernel.org](http://www.kernel.org)
- ▶ [www.linux.it](http://www.linux.it)
- ▶ [www.suse.com](http://www.suse.com)
- ▶ [www.redhat.com](http://www.redhat.com)
- ▶ In edicola: Linux & C.
  - ▶ [www.oltrelinux.com](http://www.oltrelinux.com)

## Limiti

- ▶ Numeri magici e costanti definiti nelle varie implementazioni
- ▶ Tipi di limiti:
  - ▶ Opzioni e limiti in fase di compilazione
    - Possono essere definiti in **headers** che un qualunque programma può includere a tempo di compilazione
  - ▶ Limiti run-time
    - Prevede che il processo chiami una funzione per ottenere il valore del limite
    - Limiti non associati a file/dir → funzione **sysconf**
    - Limiti associati a file/dir → funzione **pathconf**
- ▶ Limiti Ansi C sono a tempo di compilazione (cf. `<limits.h>`)
- ▶ Limiti Posix e XPG3 dello stesso genere

## Tipi di dati primitivi

- ▶ Nel file header `<sys/types.h>` (come anche in altri header) sono definiti alcuni tipi di dati system-dependent chiamati **tipi di dati primitivi**
  - ▶ Sono definiti utilizzando **typedef**
  - ▶ I loro nomi finiscono in genere con **\_t**
- ▶ La tabella 2.8 mostra i principali tipi di dati primitivi che useremo
- ▶ Esercizio:
  - ▶ in Linux dove sono definiti?