

Laboratorio Linux/FOSS

Alberto Cammozzo
anno accademico 2005-2006

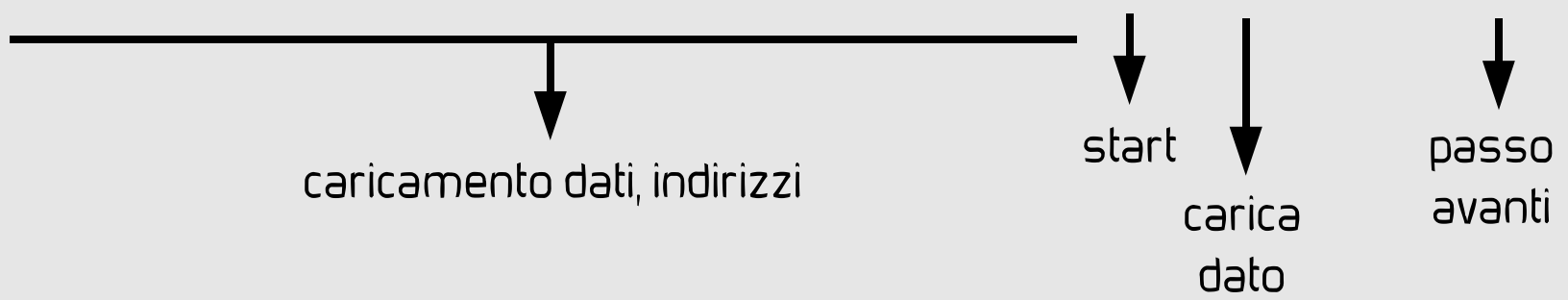
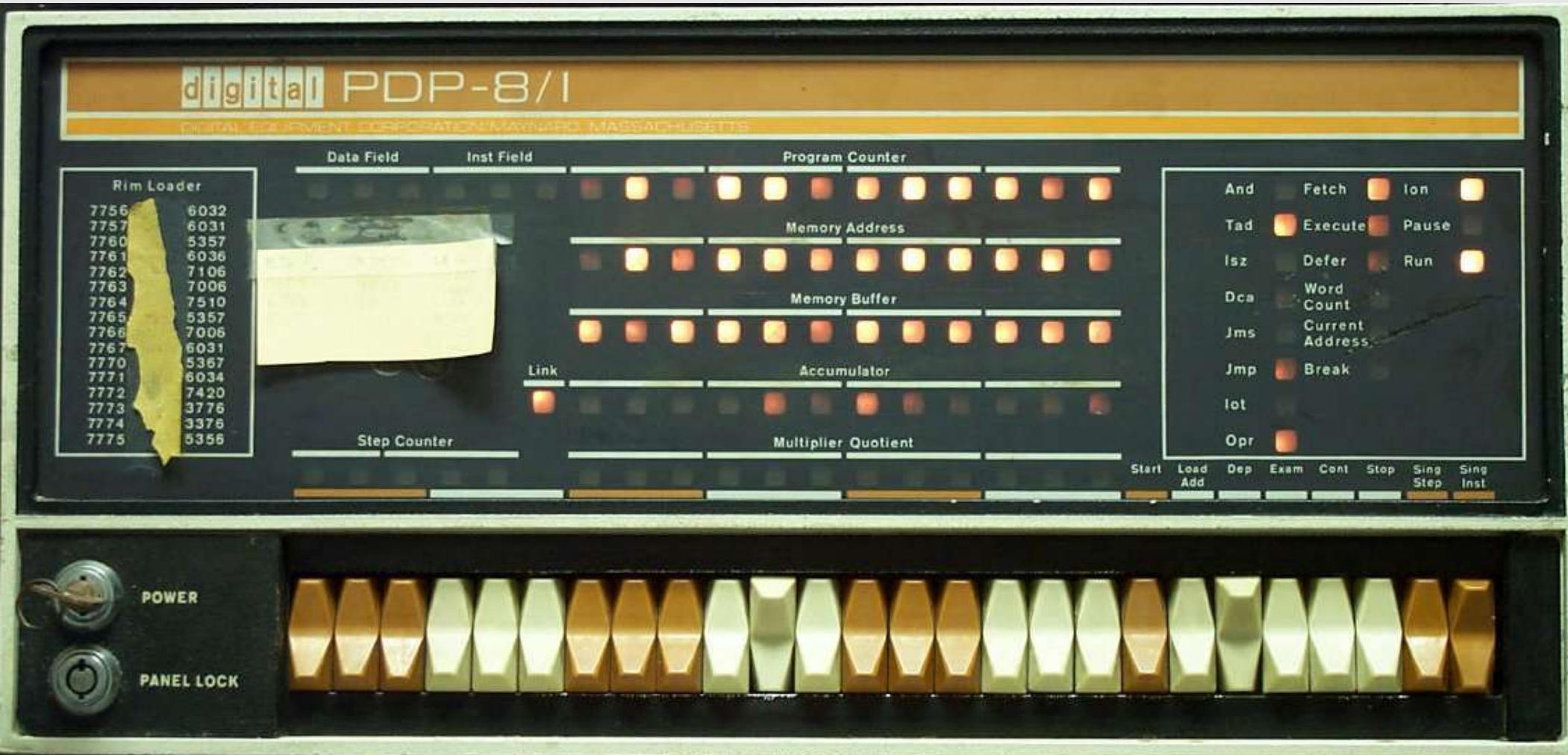
Parte Prima

Sistemi operativi

Indice

- Cenni storici
- Struttura e alcuni modelli di sistema operativo
 - tipo di kernel
 - finalità operative
- Funzioni del sistema operativo
 - raccordo e organizzazione
 - il nucleo
- GNU/Linux

PDP 8 Digital, 1970



Storia



1960	mainframe	VMS, OS360
1970	minicomputer	Unix, CP/M
1980	micro, PC	DOS, Windows, OS2
1990	PC	Linux, Plan9
2000	?	MacOS X, ?



Nascita del S.O.

- Nessun SO



- Batch

- resident monitor



- Multiprogrammazione

- time sharing
- memoria virtuale
- CPU multiple



scheduling

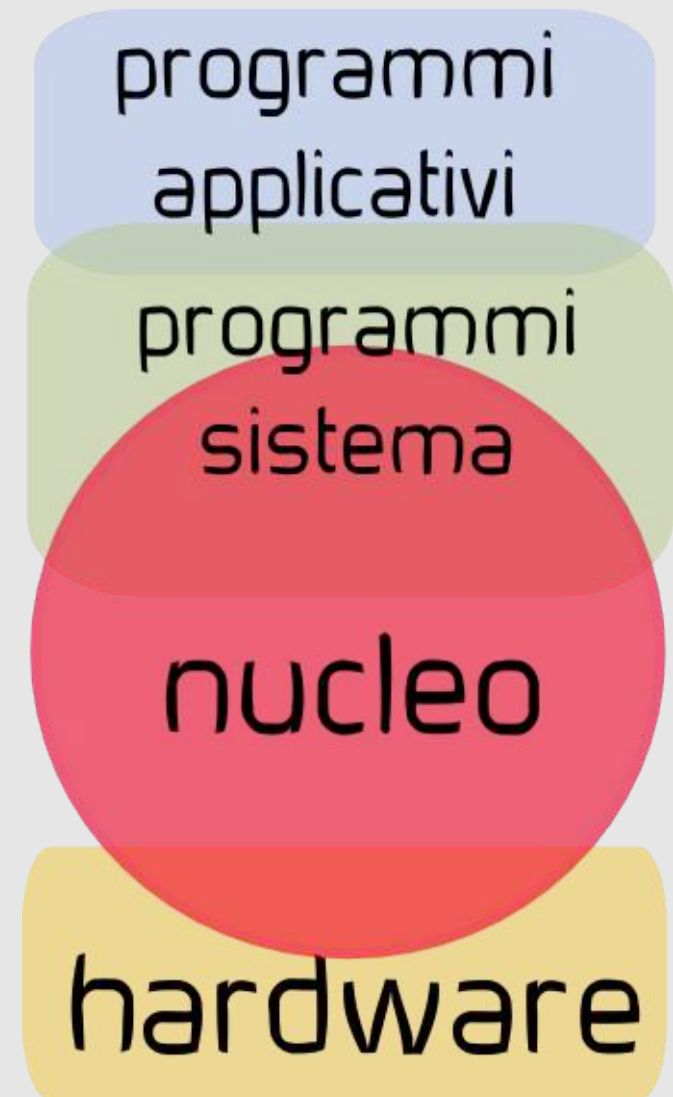
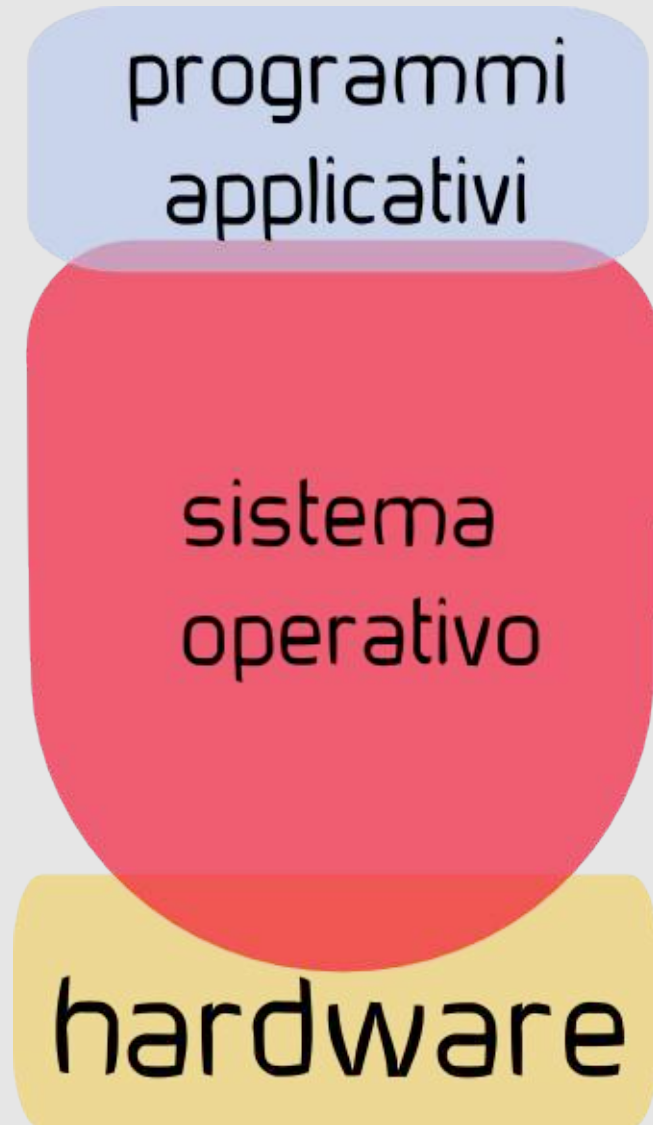
- razionalizzare l'accesso alla CPU da parte dei programmi
 - minimizzare i tempi morti
- assegnare porzioni di tempo a seconda delle necessita' e di *politiche* definite
 - privilegio per i processi interattivi
 - privilegio per i processi di input/output
 - *real-time*

Evoluzione

- Pervasive computing
 - *embedded*
 - controllo sensori
 - *entertainment*
- Più funzioni
 - rete, wifi
 - interfacce utente complesse
 - database? motori di ricerca?



Struttura



una importante distinzione

nucleo (kernel, core)

- prevalente funzione:
 - protezione
 - raccordo con l'hardware: i driver
 - *scheduling*
- inaccessibile direttamente all'utente

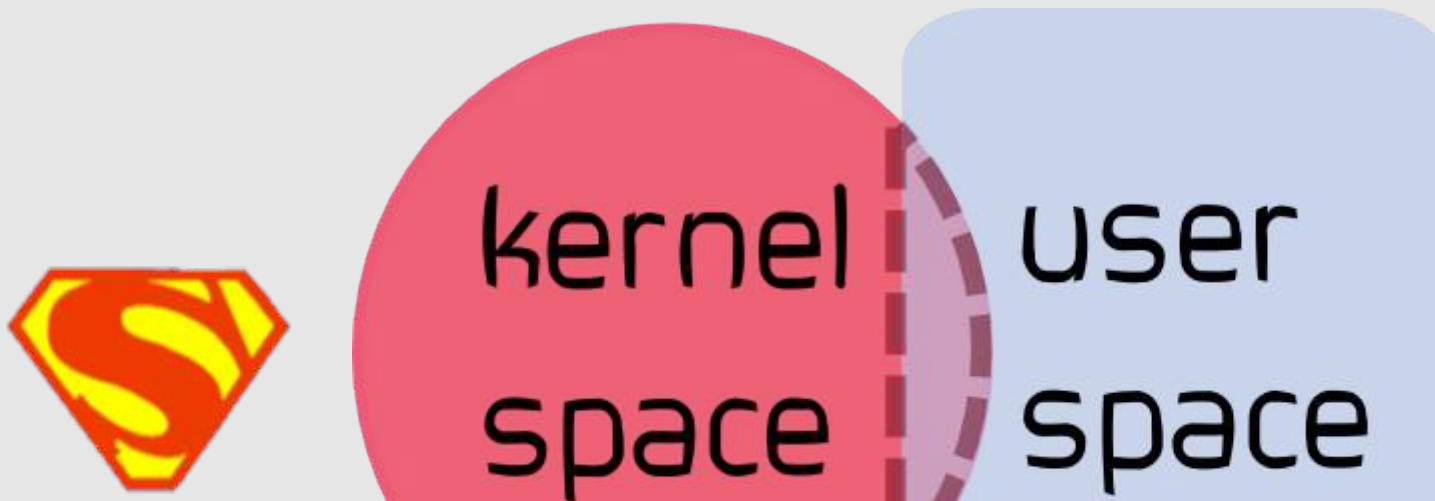
programmi di sistema

- prevalente funzione:
 - raccordo con l'utente
 - raccordo con altri calcolatori
- supporto ai programmi applicativi
- gestione dei dati

kernel

- *multiplexing* (commutazione)
e *astrazione* delle risorse fisiche
- sistemi operativi convezionali:
 - porzione *privilegiata* del s.o.
 - memoria *protetta*
 - CPU in *supervisor mode*
- quali servizi mettere dentro al kernel?
- che grado di astrazione offrire?

kernel - user



Privileged mode:

- *CPU in protected mode*
- *accesso a tutta la memoria*

Unprivileged mode:

- *CPU in modo user*
- *memoria protetta*

system call + librerie sistema

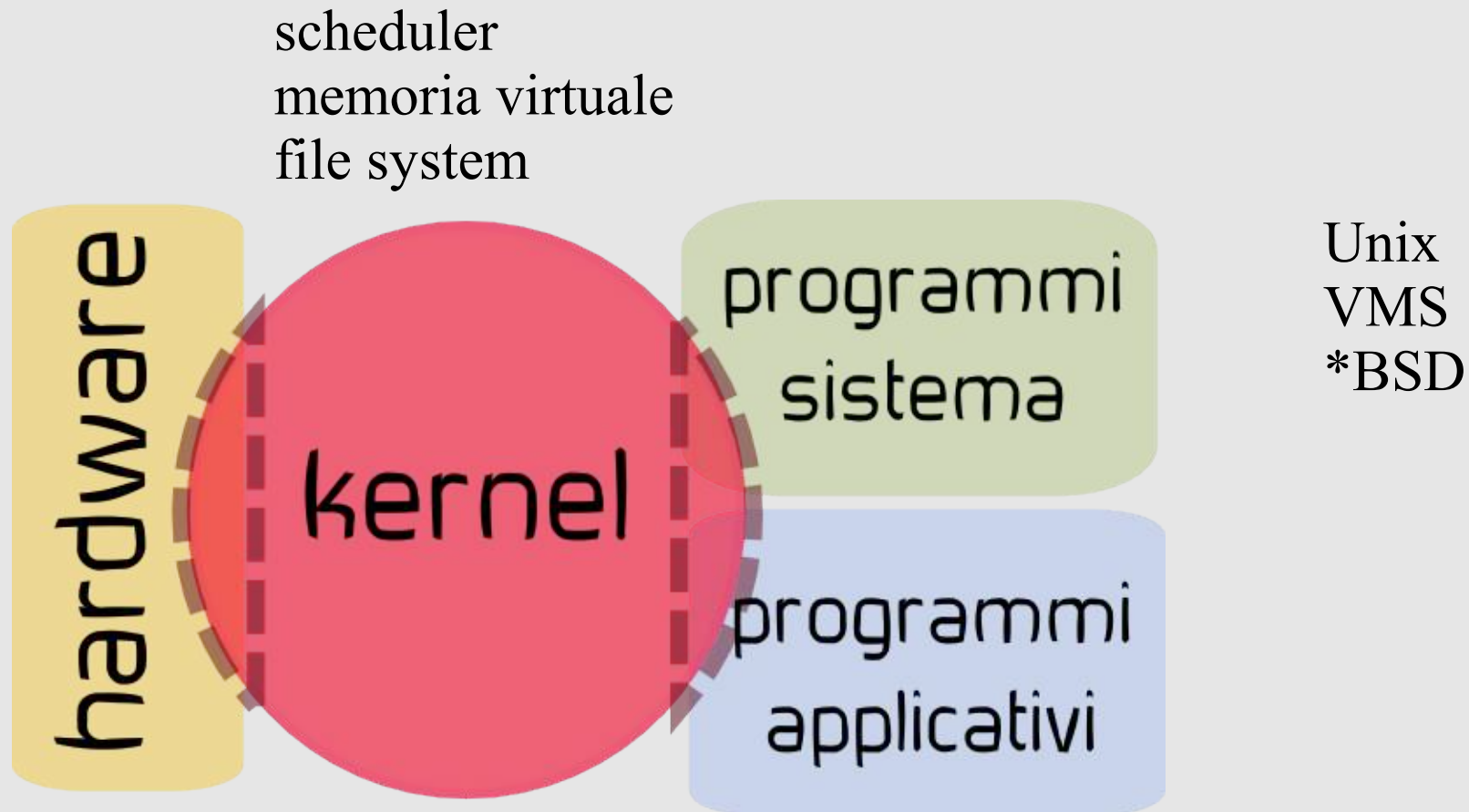
Funzione dei programmi di sistema

- interazione con l'utente
 - interfaccia a linea comandi
 - interfaccia grafica
- supporto ai programmi
 - *librerie* di funzioni di raccordo con il kernel
 - gestione permessi e privilegi di file
 - *boot*
 - eventi temporizzati, allarmi, etc...
 - ...

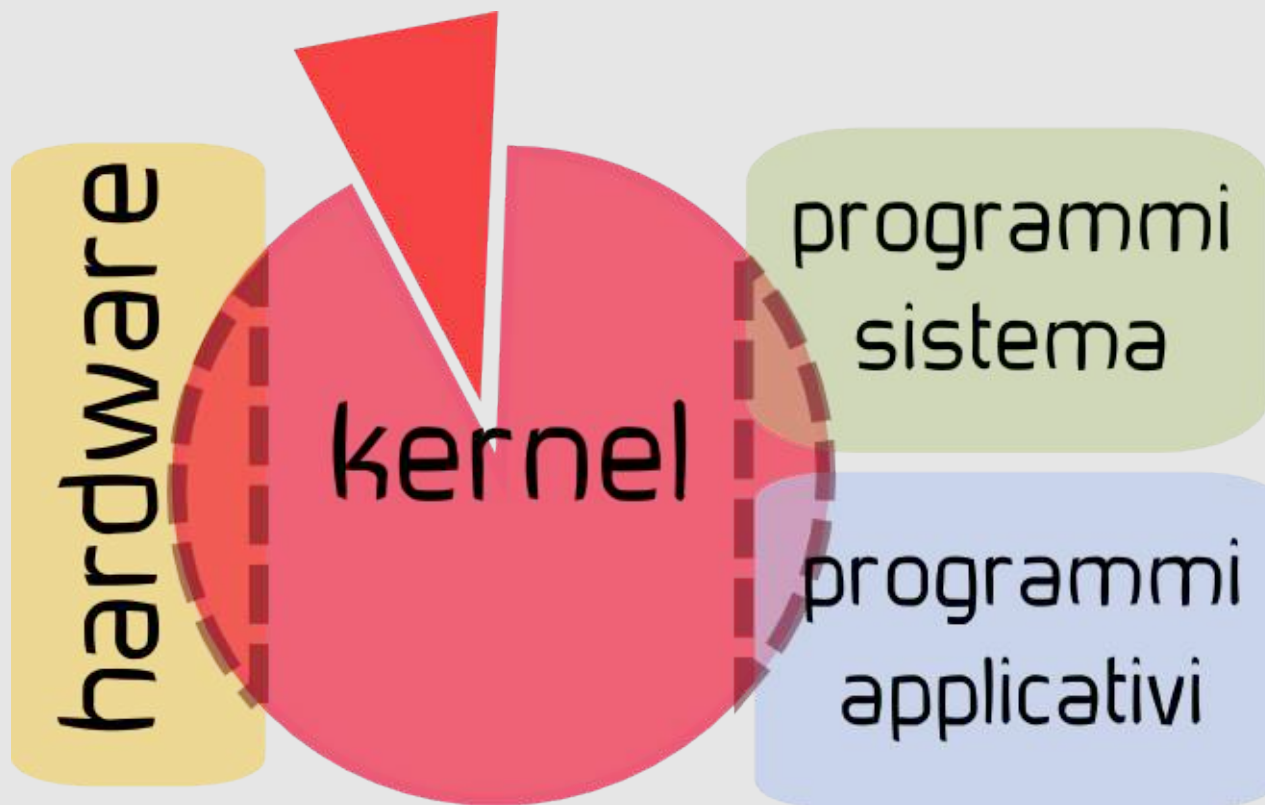
modelli di kernel

- monolitico
 - tutti i servizi dentro al kernel
 - alto livello di astrazione
- modulare
 - monolitico, a moduli
- micro
 - poche cose nel kernel
- exo
 - tutto fuori dal kernel

monolitico



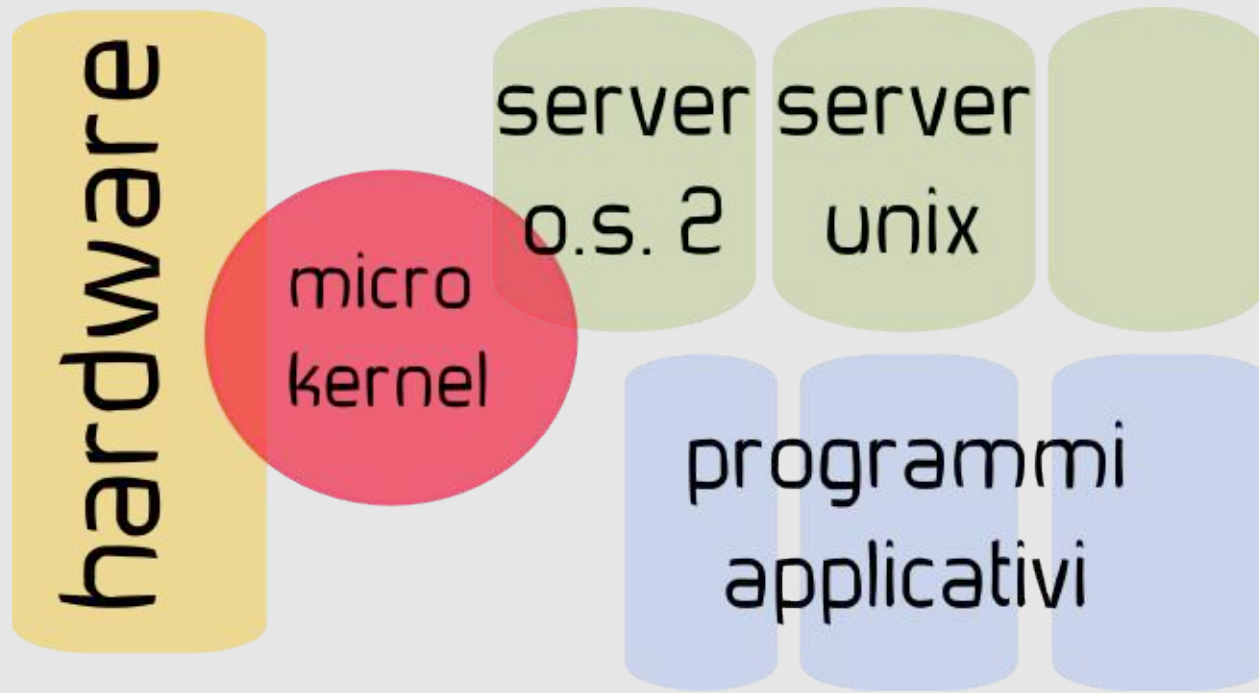
monolitico modulare



Unix,
Linux

microkernel

filesystem
virtual memory



Mach
Chorus
Amoeba
Coyotos
Spring
Minix
SymbianOS
QNX

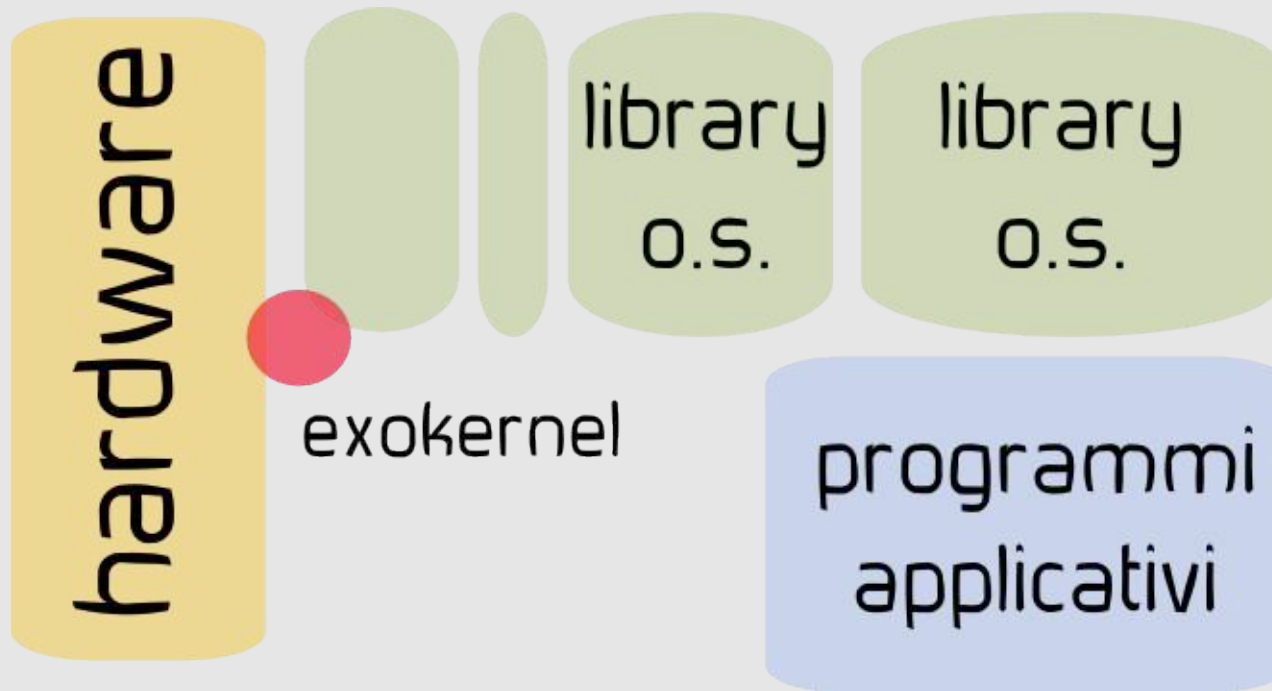
MacOS X
Windows 2000

scheduler

inter-process-communication

exokernel

filesystem
virtual memory
scheduler



MIT XOK
IBM K42

multiplexer

SO: funzioni tipiche

- Processi:
 - caricamento, assegnazione alla CPU, *scheduling*
- Input/output (I/O)
 - interazione con periferiche (inclusa la rete)
 - gestione file
- Prevenzione dei blocchi critici
- Gestione della memoria
- Interazione con l'utente
- Supporto alla programmazione

Funzione

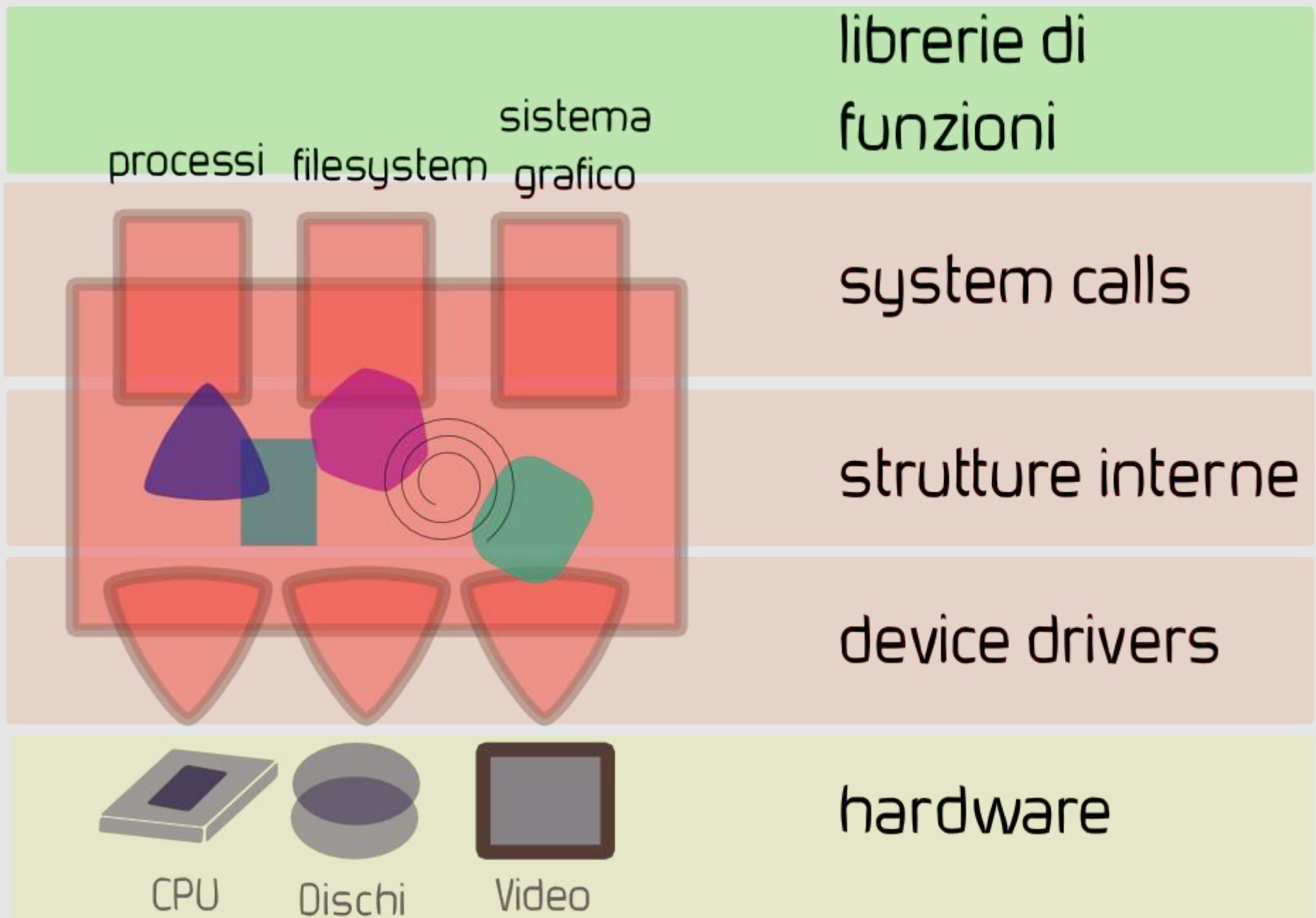
A) Raccordo

1. con l'hardware: *astrazione e virtualizzazione*
2. con l'utente: *interfaccia* linea comandi, grafica
3. tra computer: reti, calcolo parallelo e distribuito

B) Organizzazione

1. memoria: fisica e virtuale
2. programmi in esecuzione: *processi*
3. *dati*: memorizzazione, organizzazione, interscambio

A.1 La funzione di astrazione



A.1 virtualizzazione

- *cluster*
 - multiple istanze fisiche, singola istanza virtuale
 - memory management globale
 - process management globale
 - file system globale
- segmentazione o partizionamento
 - singola istanza fisica, multiple istanze virtuali
 - uso efficiente delle risorse

A.2 interfaccia utente

- linea comandi
 - mainframe, Unix shell, ambiente DOS
 - comandi → funzioni di sistema
- grafica: finestre, mouse, menu,
 - Xerox Parc, Apple, Windows, X
 - *event loop*
- tastiere ridotte, sensori di movimento, vocali, ...
- stampanti

A.3 reti di computer/1

- centralizzato
 - host – terminale
 - servizi web
- client – server
 - *storage server*
 - *application server*
 - *remote procedure call (RPC), web services*

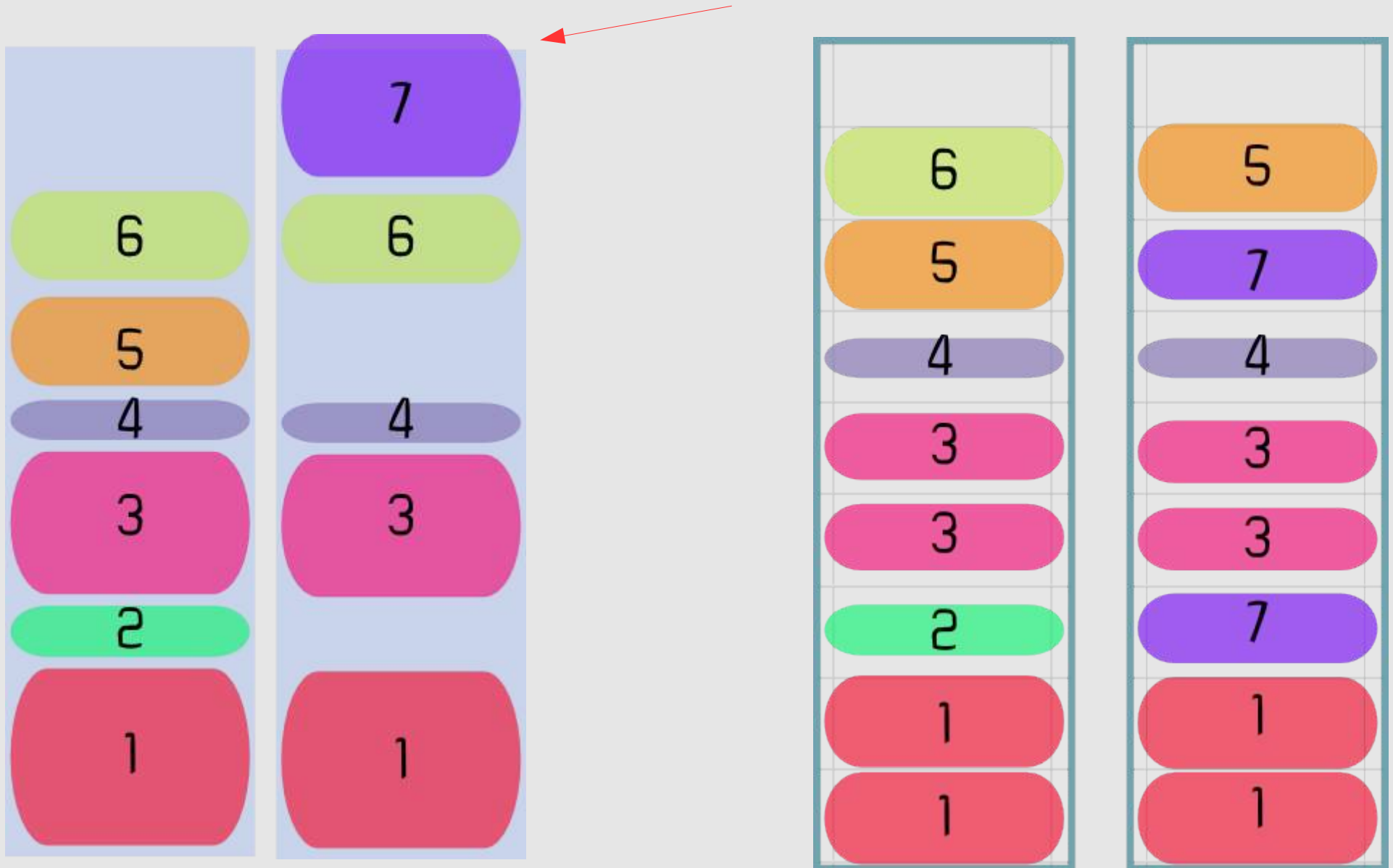
PROTOCOLLI

A.3 reti/2

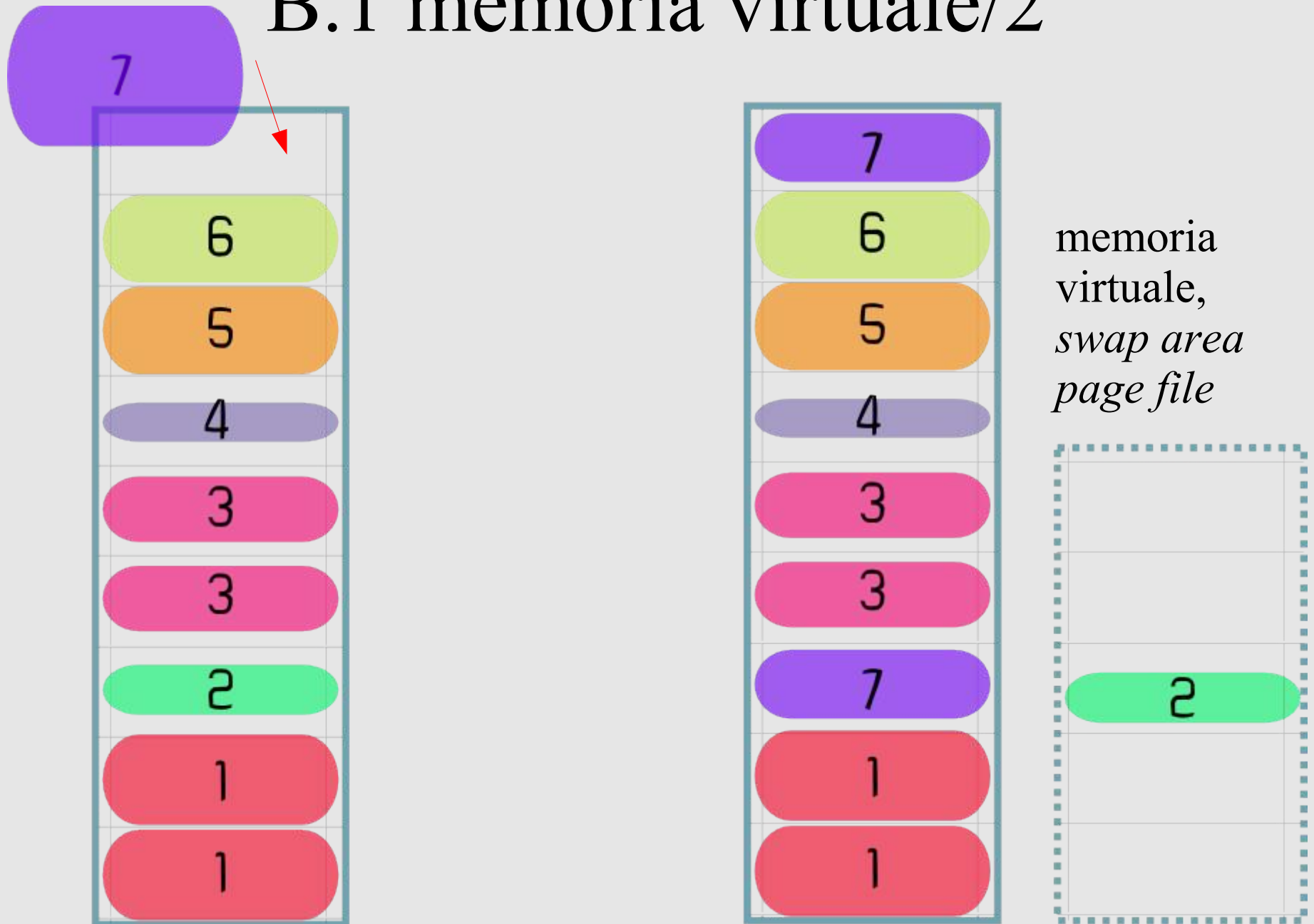
- *peer to peer*
 - *data sharing*
- *distributed* computing:
 - *clusters*: stretta cooperazione tra computer
 - alta affidabilita'
 - calcolo
 - *grid*: cluster virtuale in una rete di telecomunicazioni
 - calcolo

PROTOCOLLI
formati

B.1 memoria virtuale/1



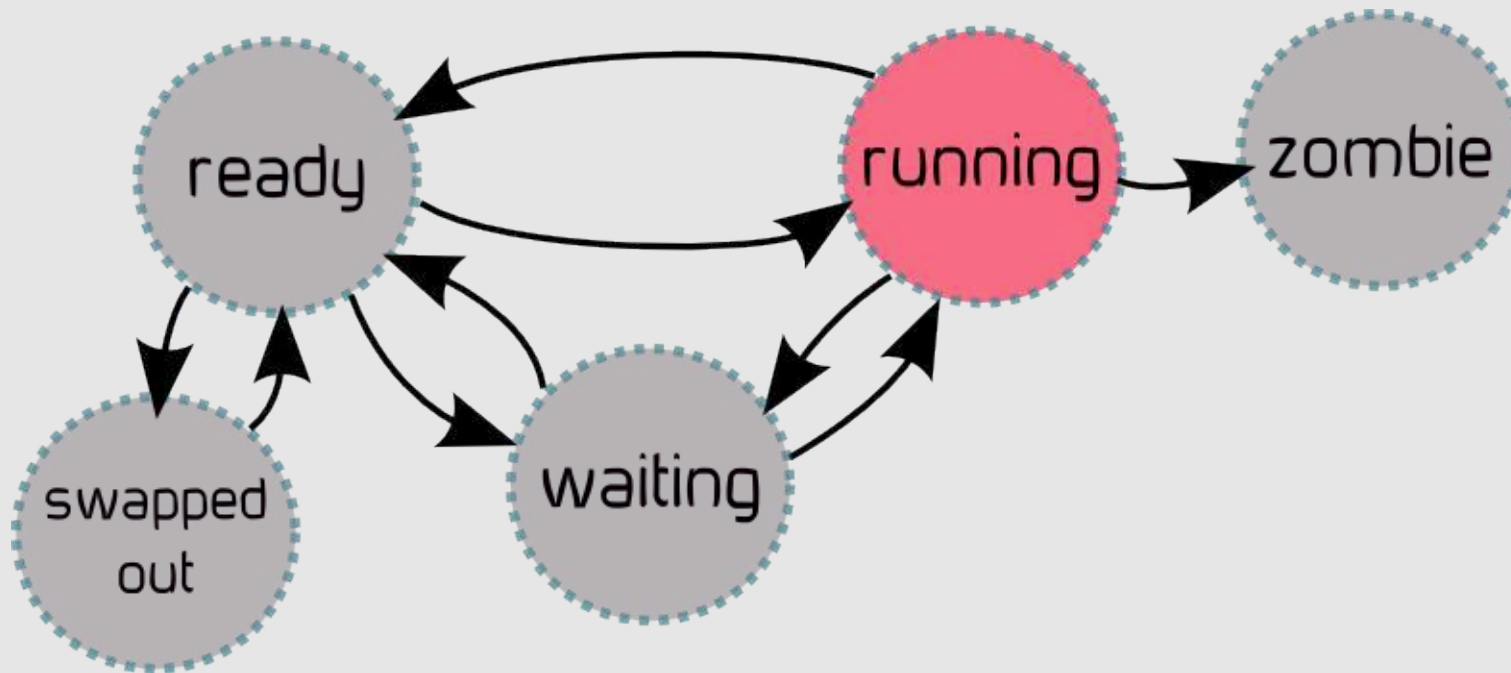
B.1 memoria virtuale/2



B.2 Processi

- Istanze di programmi
 - in un proprio spazio di indirizzamento
 - con proprie variabili e stati
- Multitasking:
 - esecuzione “contemporanea”
 - *preemptive* (con prelazione) o *cooperative*
- *Execution thread* (filo)
 - spazio di indirizzamento condiviso con altri *thread*

B.2 processi



tipico diagramma di stato dei processi

B.2 processi

- code: algoritmi e politiche di *scheduling*
- gerarchia e famiglie
 - processi padre e figlio
 - memoria, segnali
- sicurezza
 - identificativi di utenti, privilegi
 - aree di memoria non sovrapponibili
- temporizzazioni, *timeout* e allarmi
- statistiche

B.3 o.s. e i dati

- memorizzazione
 - filesystem
 - database
- organizzazione
 - attribuzione ai dati di permessi e proprietà
 - trattamento (copia, cancellazione, trasferimento, ...)
 - backup e ripristino
- interscambio
 - importa/esporta da formati diversi

File system

- astrazione della periferica di memorizzazione
 - dischi fisici, di rete, rimovibili, ...
 - facilita' di interazione
- persistentenza
- organizzazione
 - struttura gerarchica, informazioni su ogni file
- protezione
 - attribuzione di utenti e permessi

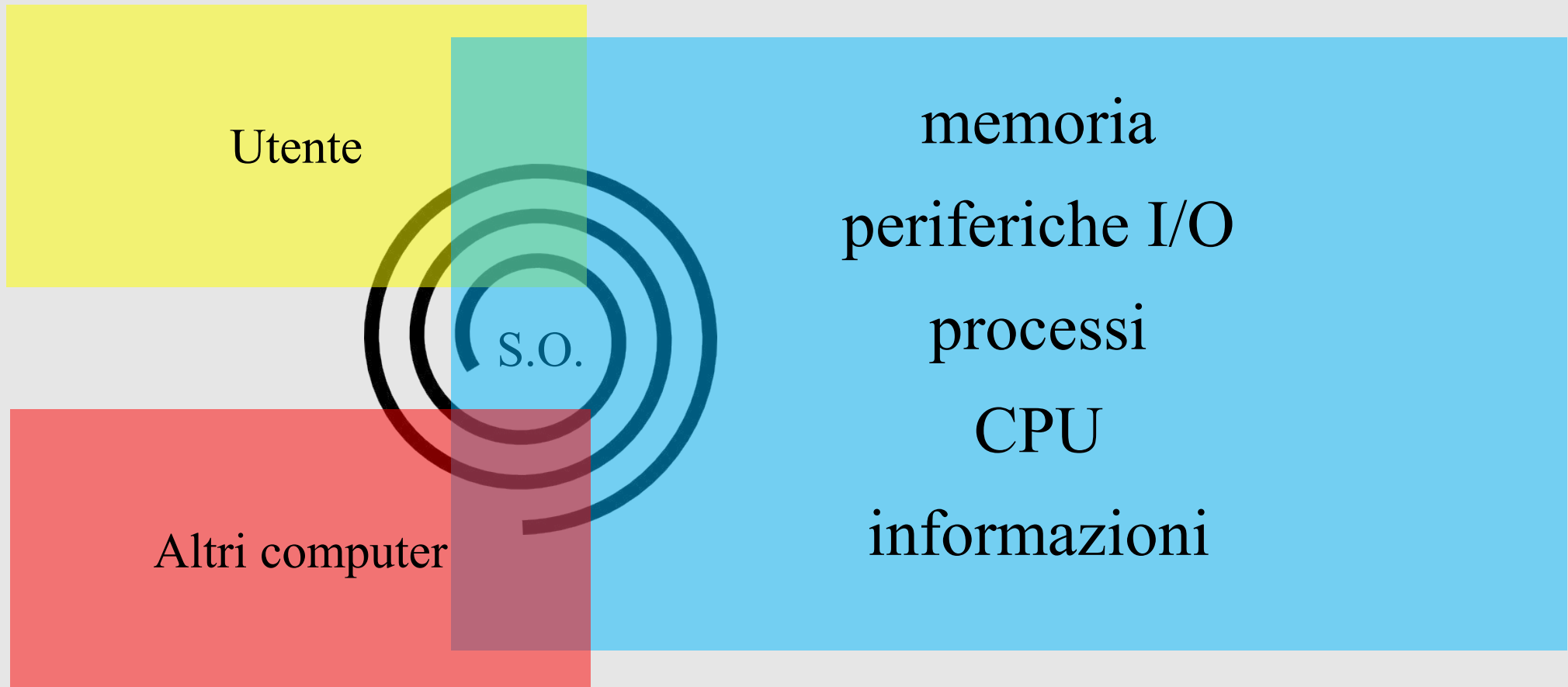
(due parole sui dati)

- i dati come patrimonio:
 - Accessibilità
 - Indipendenza dall'architettura
 - Persistenza

Legge 82/05 (codice amministrazione digitale), art. 68, comma2:

*Le pubbliche amministrazioni nella predisposizione o nell'acquisizione dei programmi informatici, adottano soluzioni informatiche che assicurino **l'interoperabilità** e la cooperazione applicativa, secondo quanto previsto dal decreto legislativo 28 febbraio 2005, n. 42, e che consentano la rappresentazione dei dati e documenti in più formati, di cui almeno uno di tipo **aperto**, salvo che ricorrano peculiari ed eccezionali esigenze.*

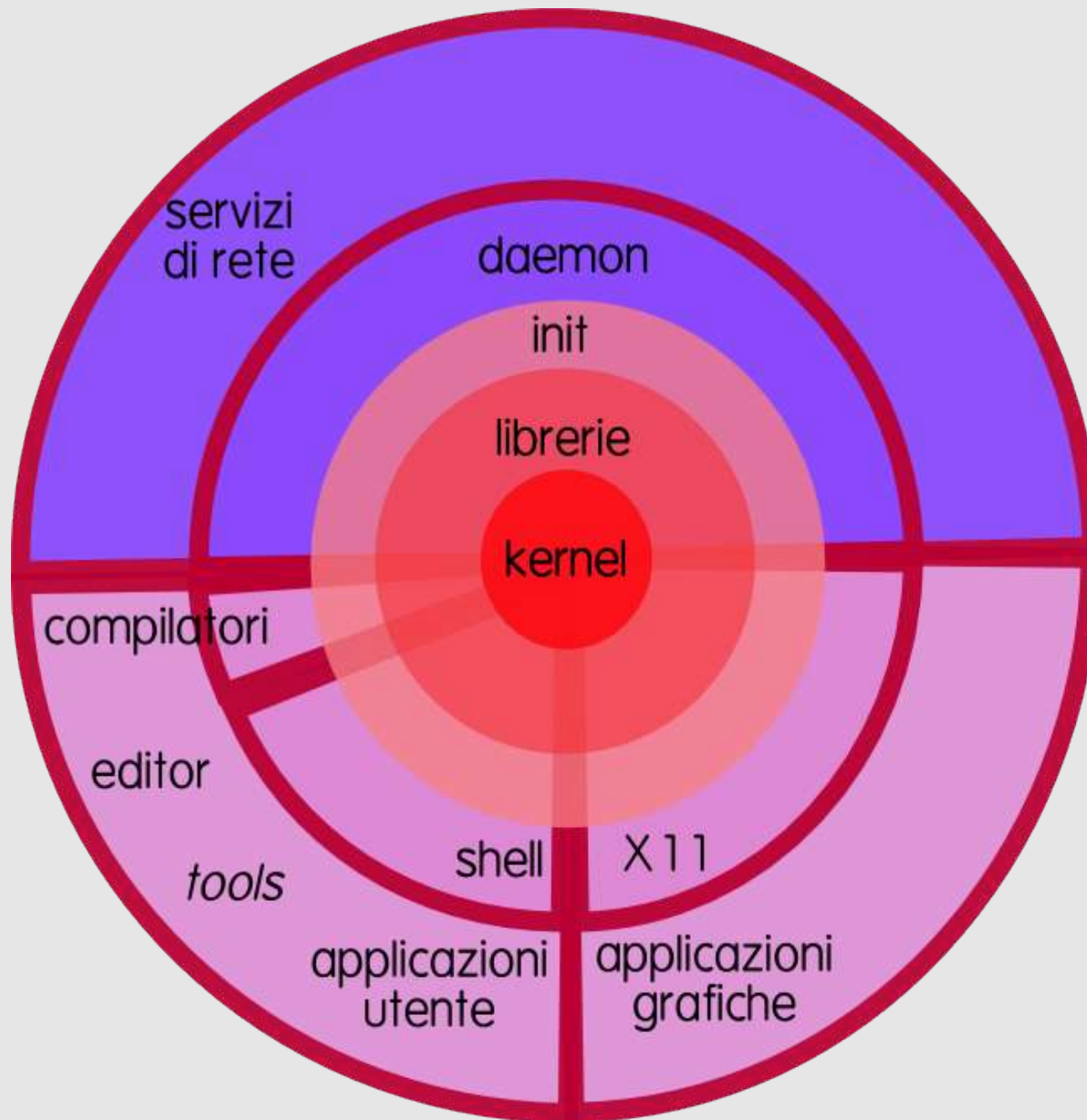
O.S. come resource manager



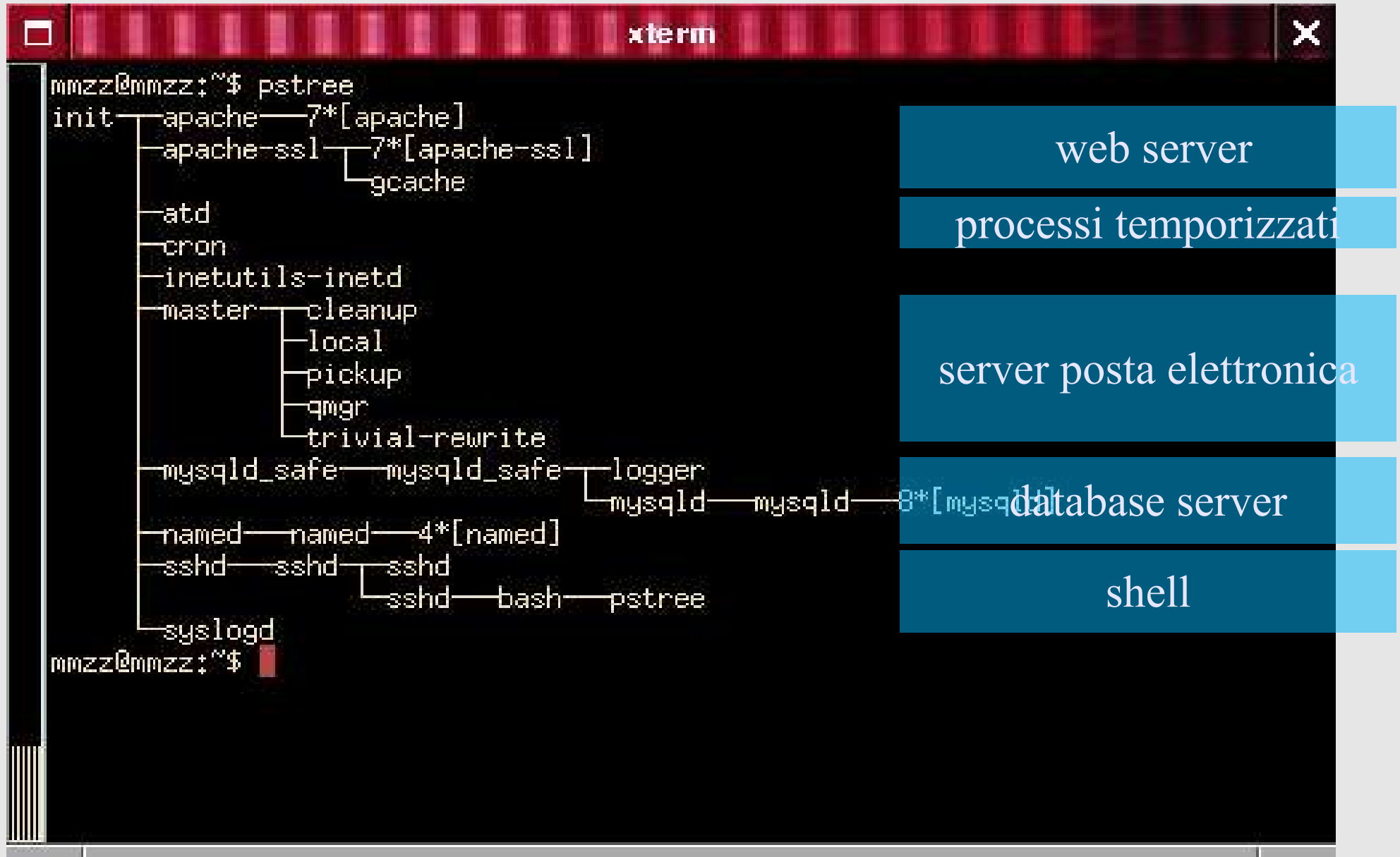
tipologie di s.o.

- specializzati
 - *real time*
 - *embedded*
 - *distributed*
 - *network*
 - database
 - affidabili
- general purpose
- multiutente
- *multitasking*

Unix/Linux: struttura



gerarchia dei processi



GNU/Linux

- funzioni
 - versatile, non specializzato (*general purpose*)
 - multiutente, multitasking
- kernel
 - monolitico, modulare
- non-kernel:
 - varie distribuzioni
 - commerciali: redhat, novell
 - non commerciali: debian, slackware, fedora, ...

Kernel Linux

- www.kernel.org
- configurabile
 - server, desktop, embedded, cluster, ...
- modulare:
 - driver di periferiche
 - protocolli di rete
- versioni 2.4 e 2.6

Per approfondire

Testi sui sistemi operativi e Unix

- Andrew S. Tanenbaum, *Operating Systems: design and implementation*, Prentice Hall, 1987
- Abraham Silberschatz, Peter Galvin, *Operating Systems Concepts*, Addison-Wesley, 1994
- P.Ancilotti, M.Boari, A. Ciampolini, G.Lipari, *Sistemi operativi*, McGraw-Hill 2004
- Maurice J. Bach, *The Design of Unix Operating System*, Prentice Hall 1986

Materiale di corsi di sistemi operativi, in italiano:

- <http://www.dimi.uniud.it/~miculan/Didattica/SOB04/>
- <http://www.cs.unibo.it/~montreso/so/index.shtml>
- <http://www.disi.unige.it/person/DelzannoG/SO1/AA0405/aa0405.html>

Gruppo di discussione comp.os.research

- FAQ: datate al 1996 <http://www.faqs.org/faqs/os-research/part1>
- Interfaccia google al newsgroup <http://groups.google.it/group/comp.os.research>