



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Corso introduttivo pluridisciplinare di Informatica¹

Mattia Monga

Dip. di Informatica
Università degli Studi di Milano, Italia
mattia.monga@unimi.it

a.a. 2013/14

¹ © 2014 M. Monga. Creative Commons Attribuzione-Condividi allo stesso modo 3.0 Italia License.
<http://creativecommons.org/licenses/by-sa/3.0/it/>.



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

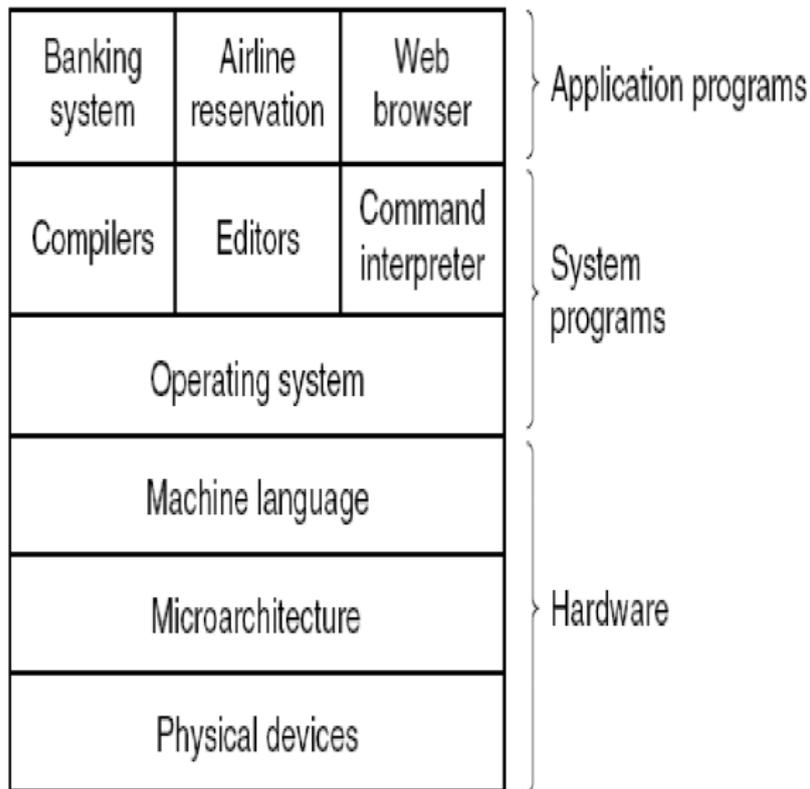
Overlay
Paginazione
Gestione dei page fault
Algoritmo

Cos'è un sistema operativo

Un insieme di programmi che:

- Gestisce e **protegge** le risorse di un calcolatore;
- Facilita a programmatori ed utenti finali l'uso della sottostante macchina hardware, fornendo le **astrazioni** di base

The onion model



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Con software di base si indica il sw che:

- Abilita l'uso del computer e delle sue componenti ad un utente
- Gestisce le risorse del sistema
- Facilita l'uso delle stesse ai programmatori di applicazioni

Fanno parte del sw di base:

- **Sistema operativo**
- Compilatori e interpreti
- DBMS
- Sistemi operativi di rete

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Con **software applicativo** si indica il sw che:

- Soddisfa le specifiche esigenze degli utenti finali (utilizzatori) in merito all'uso del calcolatore nelle loro attività

Fanno parte del sw applicativo:

- Video-scrittura, Foglio elettronico
- Contabilità, Fatturazione
- WWW, Posta elettronica, News
- ...

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Cos'è un sistema operativo



PAS

Mattia Monga

Sistema Operativo

Un s.o. è un programma che rende conveniente l'uso dello hardware

- fornendo astrazioni che semplificano l'uso delle periferiche e della memoria
- gestendo opportunamente le risorse fra tutte le attività in corso

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call
Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Le principali sono:

- System call
- Memoria virtuale
- Processo
- File
- Shell

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Le componenti fondamentali di un SO

- Inizializzazione del sistema
- Gestione dei processi
- Gestione della memoria
- Gestione delle periferiche I/O
- Apparati di comunicazione
- Gestione dei file
- Gestione Syscall/interrupt
- Protezione
- System Utility

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni

Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



Le astrazioni sono realizzabili efficientemente grazie a dispositivi di calcolo piú complessi del classico modello della Macchina di Von Neumann (*fetch-decode-execute* con memoria sequenziale). In particolare:

- Il processore ha diversi “modi” (o *ring*) di funzionamento
- Il processore è in grado di rispondere a “richieste di interruzione” (IRQ)
- Gli indirizzi della memoria sono ottenuti tramite un dispositivo programmabile (MMU)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

- Il s.o. è l'unico programma che esegue con il totale controllo delle risorse hardware (**kernel mode**).
- Gli altri programmi si appoggiano unicamente sui servizi del s.o. e la loro esecuzione è gestita e controllata dal s.o. (**user mode**)
- In molti processori questa separazione è imposta via hardware

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



La protezione è gestita tramite segnali di interrupt

Interrupt

È un segnale elettrico inviato da un dispositivo esterno (o da istruzioni particolari (*int. software*) al microprocessore, più precisamente all'interrupt controller

L'interrupt permette al processore di interrompere le attività in corso e di eseguirne altre

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Il meccanismo fondamentale di protezione è la **Chiamata di sistema**

- Le system call sono generalmente incapsulate in librerie. Un programma applicativo che vuole accedere ad una syscall deve solitamente:
 - Chiamare la routine di libreria collegata
 - Questa provvede a formattare i parametri opportunamente e a richiamare la syscall effettiva
- In realtà **non** sono normali *chiamate di procedura*, ma **interruzioni software** a cui il s.o. fa fronte svolgendo il servizio richiesto

1
2
3

```
mov eax, 1  
int 0x80
```

PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

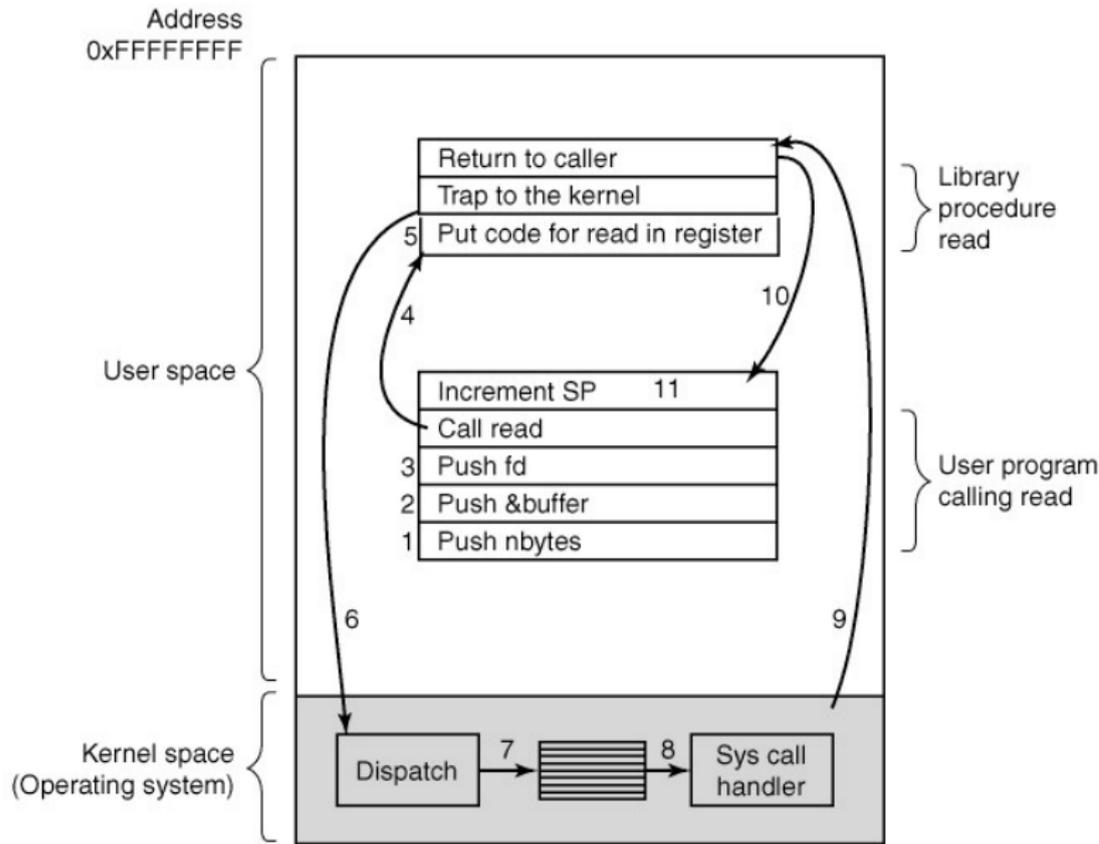
System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Syscall in un s.o. monolitico



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Il sistema operativo è il gestore esclusivo di tutte le risorse di un sistema al fine di:

- Garantire una gestione ottimale delle stesse
- Evitare collisioni
- Fornire modalità semplificate per il loro uso

Per usare una risorsa occorre rivolgersi al s.o. tramite una **chiamata di sistema** (syscall)

- Il formato delle syscall differisce tra le varie implementazioni dei sistemi operativi
- In ambito UNIX è stato fatto uno sforzo di standardizzazione: POSIX

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Kernel monolitici e micro



PAS

Mattia Monga

Kernel monolitico

Il kernel è un **unico** grande programma che gira in kernel mode. Le strutture dati e le funzionalità sono accessibili da tutto il codice del kernel

Microkernel

Il codice che gira in *kernel mode* è ridotto al minimo. Molte funzionalità (file system, process management, network, device drivers) sono realizzate da moduli *user mode* che mandano messaggi al microkernel.

I microkernel nascono per gestire meglio la complessità, ma si ha una perdita di efficienza.

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

- La memoria centrale è una risorsa fondamentale di un sistema di calcolo
- L'accesso a memoria centrale è una delle operazioni più frequenti
 - Accesso ai dati e al codice
- La tecnica usata per la gestione della memoria ha un effetto determinante sulle prestazioni del sistema

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

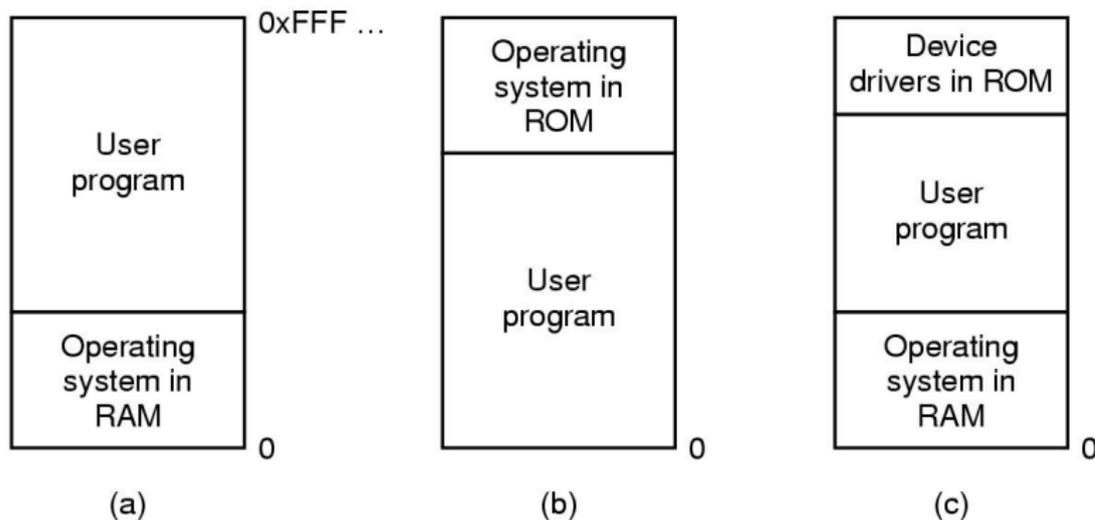
Overlay
Paginazione
Gestione dei page fault
Algoritmo

Sistemi monoprogrammati



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi monoprogrammati

Sistemi multiprogrammati

Paginazione

Memoria virtuale

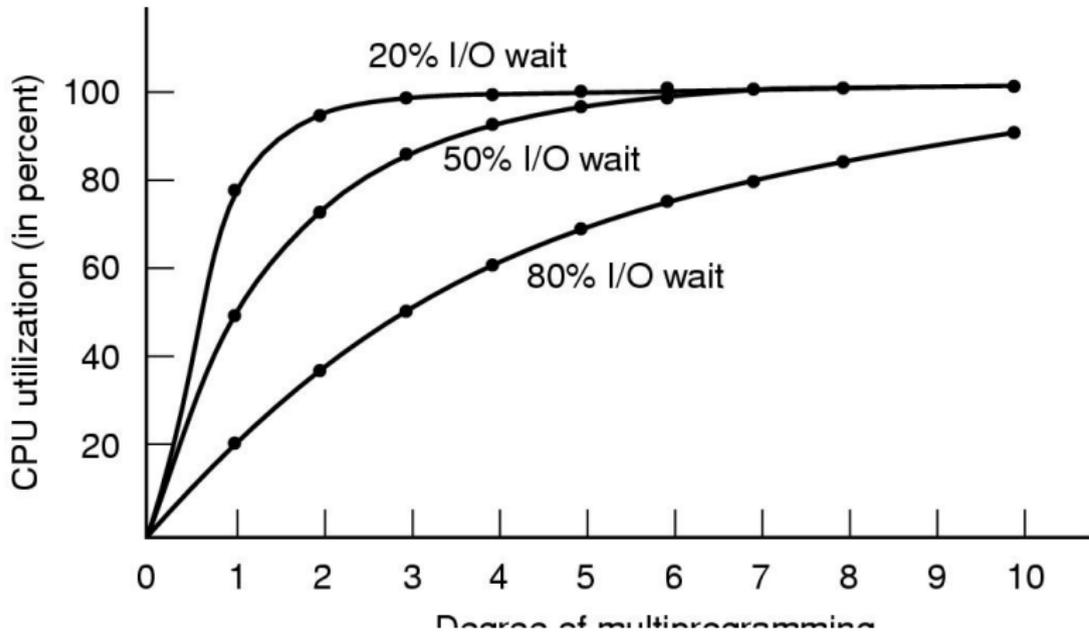
Overlay

Paginazione

Gestione dei page fault

Algoritmo

Prestazioni dei sistemi multiprogrammati



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



PAS

Mattia Monga

- Se i programmi sono I/O bound, molti programmi devono essere in esecuzione per raggiungere un elevato utilizzo di CPU
- Aumentare la memoria centrale di un sistema monoprogrammato non serve: per aumentare l'utilizzo di CPU occorre incrementare il livello di multiprogrammazione
 - L'efficienza dipende dal livello corrente di I/O (Pendenza della curva di utilizzo)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-

programmati

Sistemi multi-

programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



La gestione dell'esecuzione di piú programmi può essere ottenuta in due modi

1. Mantenere la stessa politica di gestione della memoria e ogni volta che un processo deve fare I/O effettuare uno swapping da memoria centrale a disco
2. Modificare la politica di gestione della memoria e consentire la presenza in memoria di piú programmi
 - sistemi multiprogrammati

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-

programmati

Sistemi multi-

programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

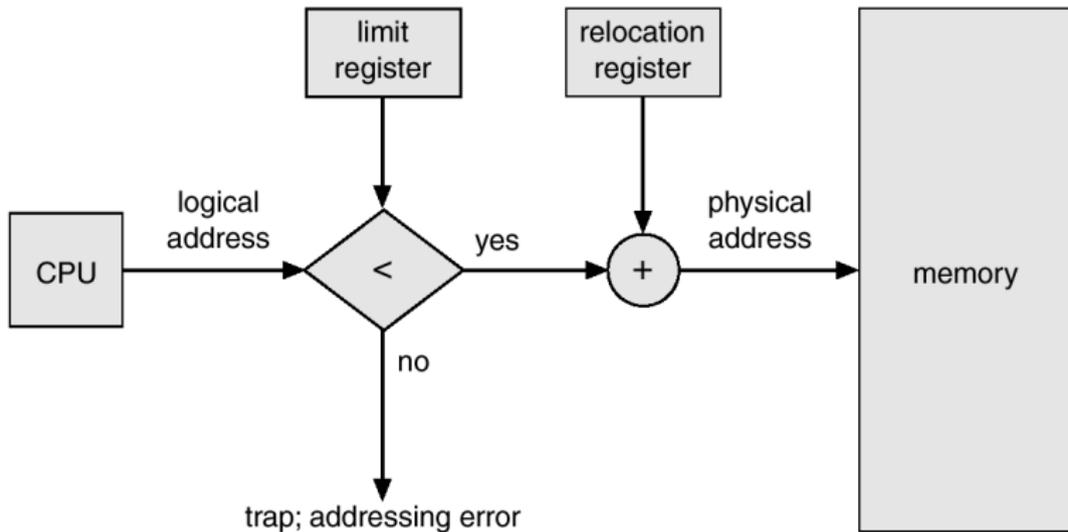
Algoritmo

Protezione hw



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati

Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Strategie di allocazione per l'intera memoria



Il pericolo è il frazionamento della memoria libera in piccoli pezzi inutilizzabili (frammentazione esterna).

First Fit il primo buco sufficientemente grande per contenere il processo

Next Fit si tiene traccia dell'ultima allocazione e si ricerca il prossimo buco sufficientemente grande

Best Fit il più piccolo buco sufficientemente grande per contenere il processo

Worst Fit il più grande buco che possa contenere il processo

Quick Fit mantenere delle liste separate per le richieste delle dimensioni più frequenti

Nessuna strategia è ottimale

PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati

Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Con gli schemi a partizione è possibile caricare in memoria un processo **solo quando si trova una porzione di memoria contigua**, che possa contenere l'intero processo è il processo di allocazione può richiedere parecchio tempo

Paginazione

La memoria fisica viene suddivisa in blocchi (**page frame**) di dimensioni costante (p.es. 4K). Porzioni di memoria logica di uguale dimensione (**pagine**) vengono caricate nei frame: la memoria di un processo è fisicamente contigua solo all'interno di una stessa pagina.

La paginazione soffre di **frammentazione interna**

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

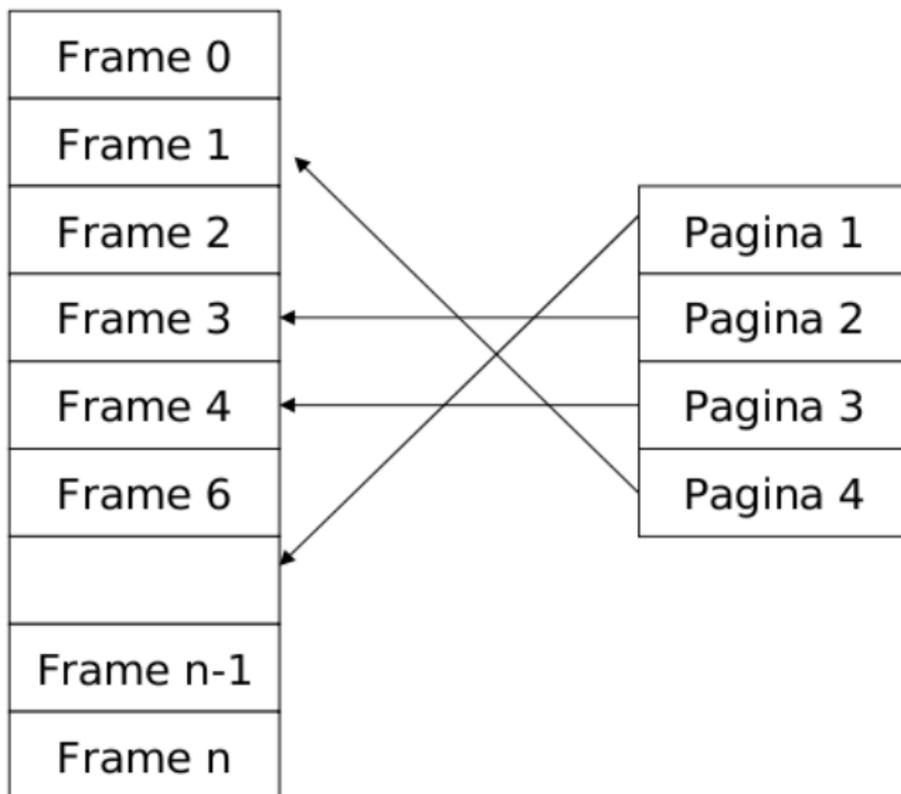
Algoritmo

Paginazione



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



In un sistema paginato gli indirizzi logici di memoria sono espressi da una coppia (p, d) , a cui va associato in fase di esecuzione un indirizzo fisico *lineare*, che esprime la locazione esatta del dato o istruzione in memoria

Example

Pagine di 256 byte, indirizzo logico 26251

$$p = 26251 / 256 = 102$$

$$d = 26252 \% 256 = 139$$

Per calcolare l'indirizzo fisico i bisogna conoscere in quale frame f è stata caricata la pagina 102

$$i = f * 256 + d$$

PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



Con pagine di dimensione 2^k , per il calcolo degli indirizzi fisici si sfrutta l'indirizzo logico

- I k bit meno significativi dell'indirizzo logico individuano lo spiazzamento all'interno della pagina
- I restanti bit più significativi individuano il numero della pagina
- Il numero di pagina si usa come indice nella **tabella delle pagine**
- Il valore trovato in corrispondenza è il numero di page frame associato alla pagina
- Se la pagina è presente in memoria, il numero di page frame è copiato nei bit alti del registro di output e concatenato ai k bit bassi dell'offset

Example

26251: 000001100110 10001011

PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

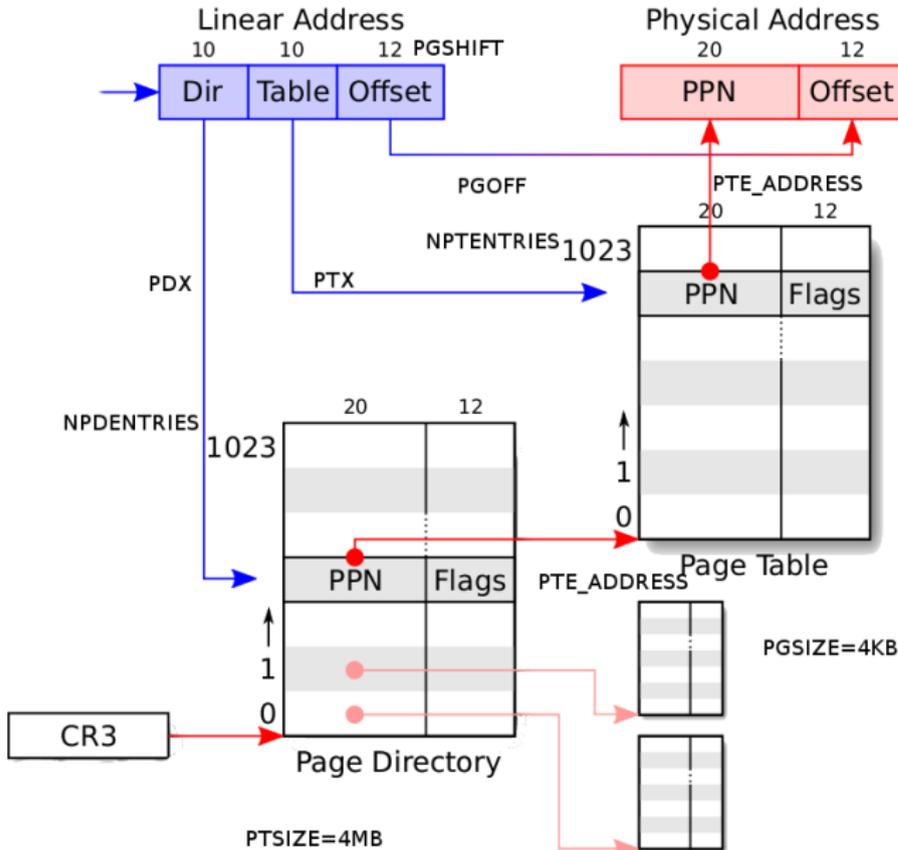
Overlay

Paginazione

Gestione dei page fault

Algoritmo

Tabella delle pagine



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmi



La memoria è *virtuale* in almeno due sensi diversi:

- Gli indirizzi usati nei programmi fanno riferimento allo spazio di indirizzamento del processo, anziché all'indirizzo fisico assoluto in memoria centrale.
 - Lo stesso indirizzo numerico si riferisce a parole di memoria differenti in differenti processi
- Gli indirizzi potrebbero fare riferimento a porzioni di memoria *non ancora caricate in memoria centrale*: il caricamento avviene al momento dell'esecuzione dell'istruzione
 - La stessa parola di memoria è utilizzata da due processi differenti (molto probabilmente riferendovisi con due indirizzi numerici differenti)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Il *late-binding* fra indirizzo e parola di memoria centrale può comunque funzionare in maniera ragionevolmente efficiente grazie al principio di località dei programmi

Località

- **Località temporale** se un elemento x viene referenziato all'istante t , la probabilità che x venga referenziato anche all'istante $t + t'$, cresce al tendere di $t' \rightarrow 0$
- **Località spaziale** se un elemento x di posizione s viene referenziato all'istante t , la probabilità che venga referenziato un elemento x' di posizione s' tale che $|s' - s| \leq \delta$ all'istante $t + t'$, cresce al tendere di $t' \rightarrow 0$

Non è quindi necessario caricare un programma interamente in memoria per poterlo eseguire, ma è sufficiente caricarlo “località per località”

PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



- Meccanismo molto primitivo per realizzare la memoria virtuale: affidato al programmatore
- Il programmatore suddivide il programma in porzioni la cui esecuzione non si sovrappone nel tempo
- Ogni porzione è chiamata **overlay**
- Gli overlay **vengono caricati in istanti successivi**

PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay

Paginazione
Gestione dei page fault
Algoritmo



- La paginazione può essere utilizzata per gestire la virtualizzazione della memoria: **paginazione a richiesta**
- La risoluzione delle pagine logiche in page frame viene sfruttata per caricare in memoria centrale una pagina e scaricarne un'altra nella memoria secondaria *swapping*
- **Lazy swapping** Si carica la prima pagina del programma da eseguire e solo quando nuova pagina viene referenziata la si carica in memoria

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



PAS

Mattia Monga

- La precedente strategia richiede però che sia possibile stabilire a priori la presenza o meno di una pagina in memoria
- Uso un bit present/absent (validità) nella tabella delle pagine
- L'evento di pagina non trovata in memoria è denominato **page fault**
- Gestire il page fault significa preoccuparsi di recuperare la pagina da disco e portarla in memoria

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



PAS

Mattia Monga

Un sistema che offre la paginazione deve gestire le seguenti strutture dati

- Tabella dei frame
- Tabella delle pagine, che definisce l'associazione pagina → frame
 - una per ogni processo in esecuzione o in attesa di esecuzione
- Per ogni riferimento a memoria è **necessario accedere alla tabella della pagine**

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

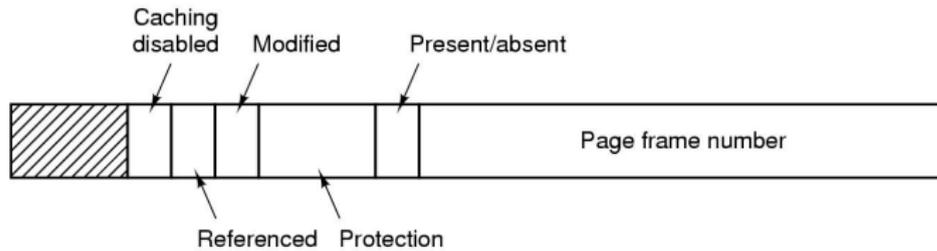
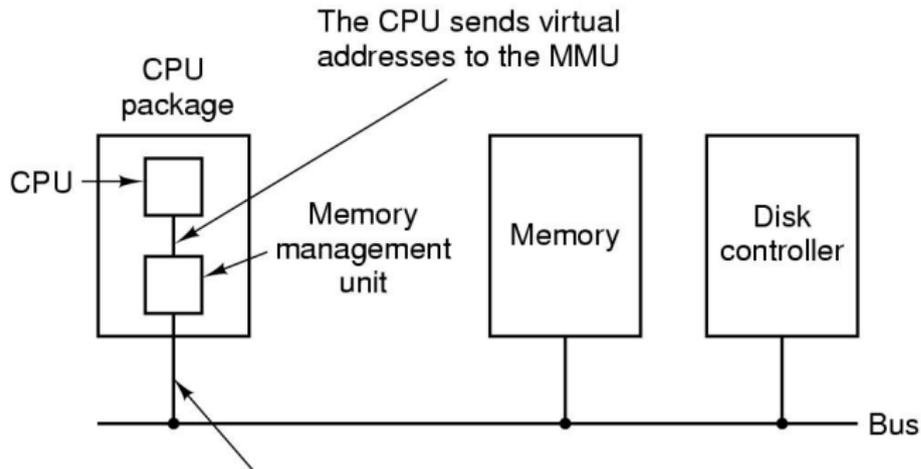
Overlay

Paginazione

Gestione dei page fault

Algoritmo

MMU



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



PAS

Mattia Monga

- La dimensione della tabella può essere molto elevata
 - Con pagine di 4K:1 milione di pagine con indirizzi di 32 bit, quindi 1 milione di elementi nella tabella
- L'associazione pagina-frame deve essere molto efficiente perché viene eseguita per ogni riferimento alla memoria
 - Un'istruzione richiede sempre almeno 1 accesso

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

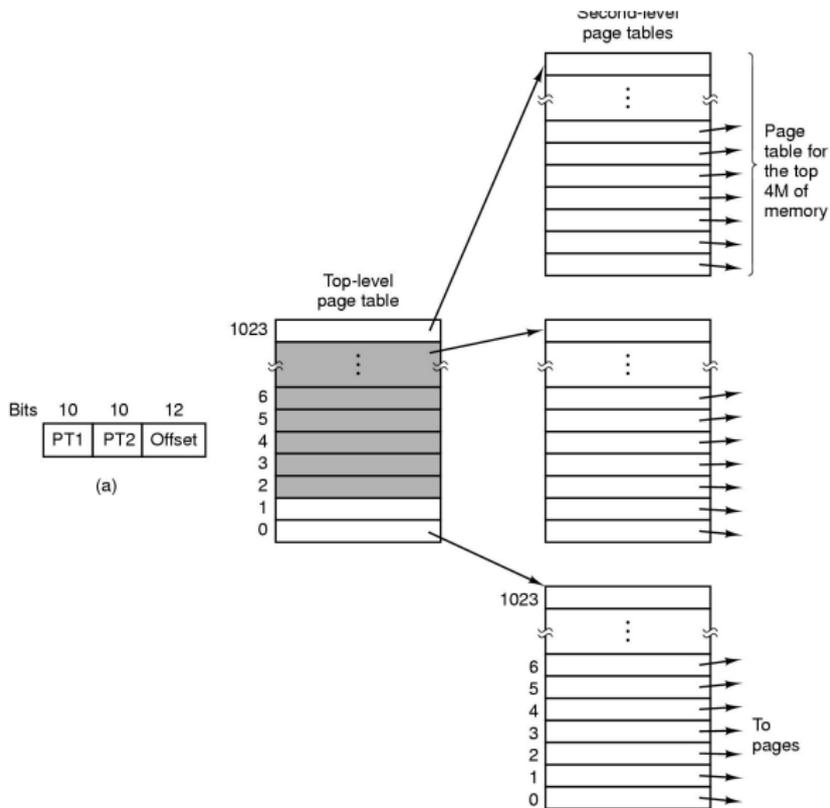
Overlay

Paginazione

Gestione dei page fault

Algoritmo

Tabella delle pagine a 2 livelli



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

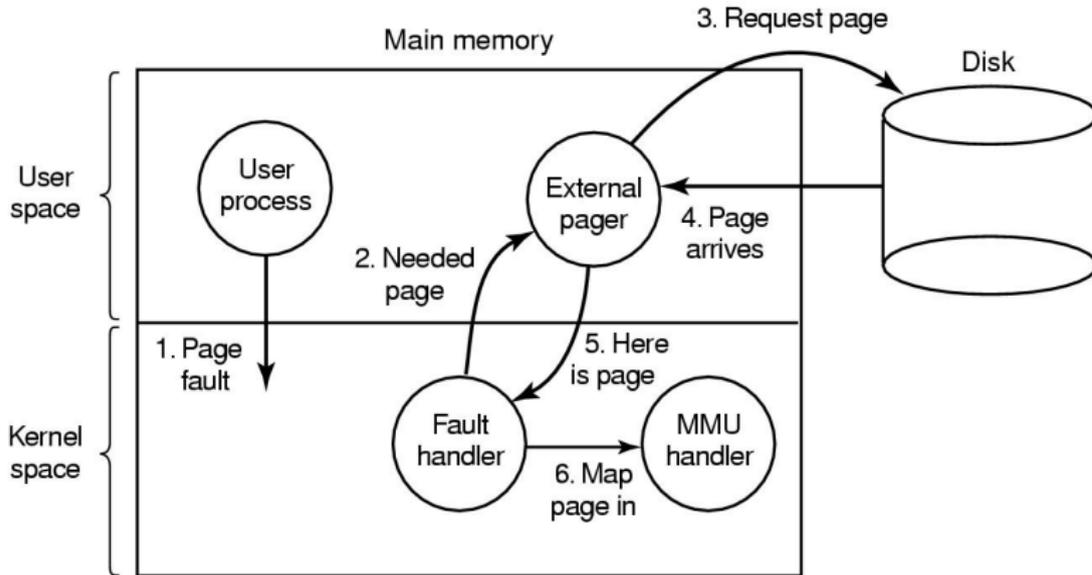
Overlay
Paginazione
Gestione dei page fault
Algoritmo

Page fault



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione

Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
**Gestione dei
page fault**

Durante la gestione di un page fault può essere necessario:

- Individuare le pagine da scaricare su disco
- Salvare le pagine modificate durante l'esecuzione, quelle non modificate possono invece essere sovrascritte



La strategia ottima sarebbe quella di *sostituire, tra quelle presenti in memoria, la pagina che sarà referenziata il piú tardi possibile*

- Ottimale ma non realizzabile
- Possibili stime
 - Tenere traccia dell'uso passato delle pagine e inferire il futuro dal passato
 - Anche questa euristica è molto difficile da realizzare praticamente

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



- Per ogni pagina lo hardware gestisce un *Reference bit* e un *Modified bit*
- I bit vengono settati quando la pagina è acceduta e/o modificata
- Le pagine vengono suddivise in quattro classi
 - ① not referenced, not modified
 - ② not referenced, modified
 - ③ referenced, not modified
 - ④ referenced, modified
- NRU rimuove la pagine casualmente a partire dalla classe con l'indice piú basso non vuota

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

- Mantiene una lista delle pagine che rispetta l'ordine con cui le stesse sono state caricate in memoria (le pagine più *anziane* sono all'inizio della lista)
- Le pagine più anziane sono le prime ad essere sostituite
- Si rischia di sostituire una pagine che è referenziata spesso (principio di località)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

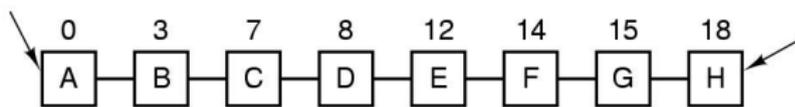
Second chance



PAS

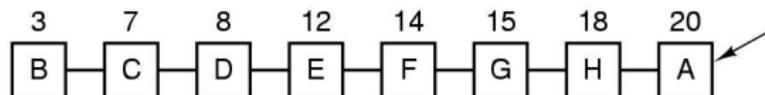
Monga

Page loaded first



(a)

Most recently loaded page



(b)

A is treated like a newly loaded page

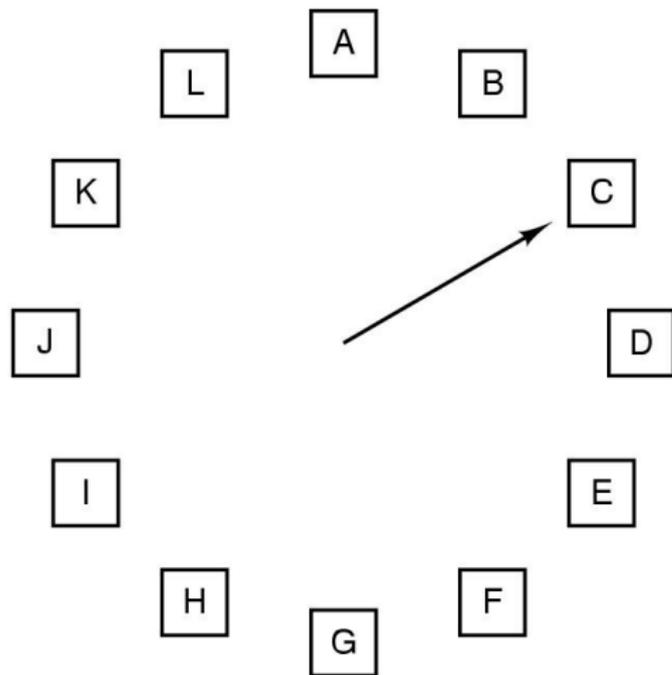
- Le pagine sono ordinate FIFO con un bit d'uso R
- Le pagine selezionate per il rimpiazzo vengono rimesse in coda se R è settato

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



When a page fault occurs, the page the hand is pointing to is inspected. The action taken depends on the R bit:

R = 0: Evict the page

R = 1: Clear R and advance hand

Come seconda chance, ma si usa una lista circolare: la lancetta punta alla pagina candidata

- Una buona approssimazione dell'algoritmo ottimale
- Assume che le pagine usate recentemente saranno usate anche in un futuro prossimo
 - Elimina le pagine che non sono usate da più tempo
- Mantiene una lista linkata di pagine
 - Quelle usate più recentemente all'inizio,
 - È necessario aggiornare questa lista ad ogni riferimento a pagina

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

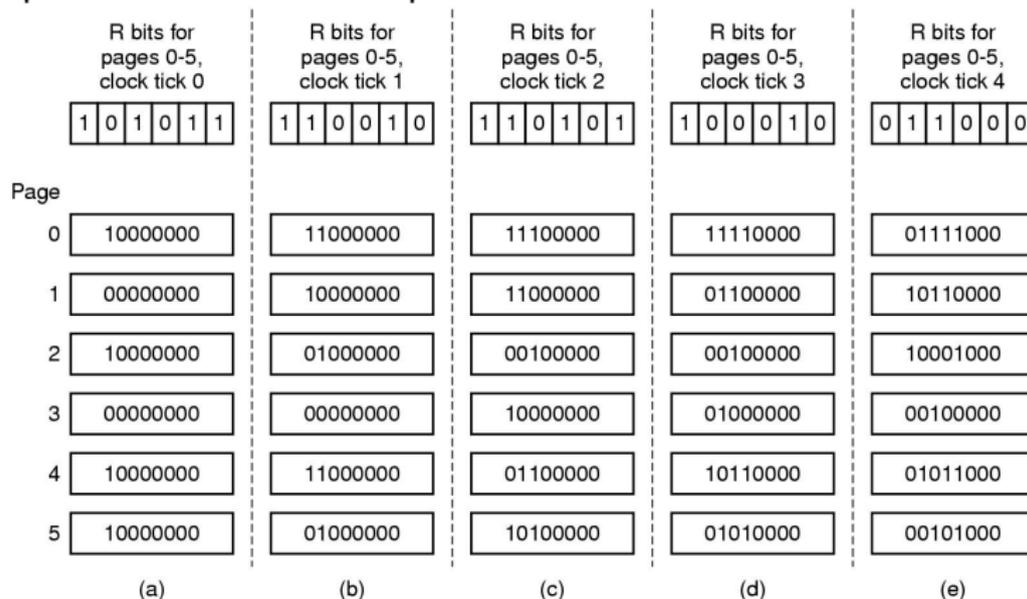
Simulazione LRU tramite aging



PAS

Mattia Monga

Alternativamente è possibile mantenere un contatore per pagina che si incrementa ad ogni tick: la pagina vittima è quella con il contatore piú basso



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



Affinché lo schema funzioni, è necessario disporre di:

- Routine di gestione degli interrupt
- Moduli per la gestione dei programmi sospesi e di quelli pronti all'esecuzione
- Moduli per la gestione delle periferiche
- più processi possono richiedere l'uso della stessa risorsa
- Moduli per la gestione della memoria

Tutte queste funzionalità sono accorpate nel sistema operativo

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Multiprogrammazione



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Prog. 3	
Prog. 2	
Prog. 1	
Gestione Proc.	Gestione Mem.
Gestione I/O	Gestione Inter
Interfaccia HW/SW	



- Nei sistemi batch la CPU di un sistema svolgeva un'attività fino al suo completamento e solo allora avviava un'altra attività
- Grazie alla multiprogrammazione la CPU di un elaboratore è impegnata nella gestione di più attività *in parallelo*
- Il sistema operativo gestisce **processi**, fornendo l'astrazione dello **pseudo-parallelismo**, dando all'utente l'impressione che esistano più processori.

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

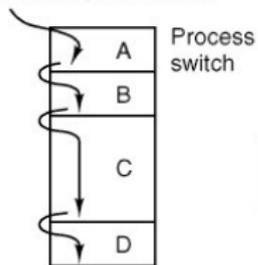
Pseudo-parallelismo



PAS

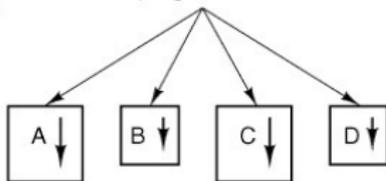
Mattia Monga

One program counter

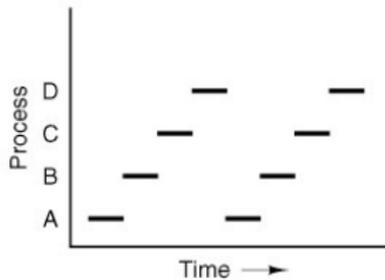


(a)

Four program counters



(b)



(c)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

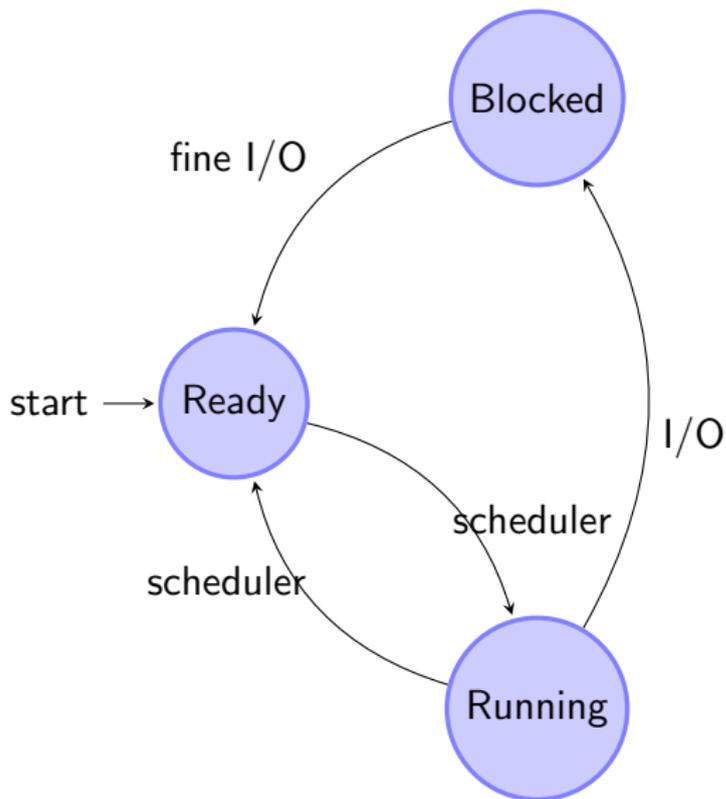
Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

- Un **processo** è un'entità che rappresenta un programma in esecuzione: conserva istruzioni, input, output, e lo *stato* dell'attività.
- Occorre definire con precisione:
 - come viene creato;
 - come evolve;
 - come termina.

Evoluzione dei processi



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

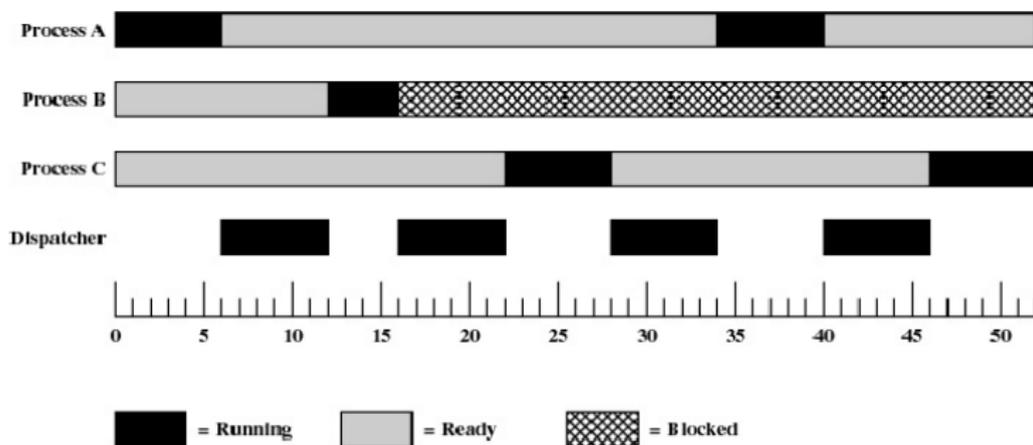
Overlay
Paginazione
Gestione dei page fault
Algoritmo

Evoluzione dei processi



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



PAS

Mattia Monga

- Il passaggio della CPU da un processo ad un altro è realizzato attraverso l'operazione di **context switch** o **process switch**
- Durante questa operazione i dati significativi del processo in esecuzione devono essere salvati, e sostituiti da quelli del processo che deve andare in esecuzione
- **L'operazione di context switch è interrupt driven**

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

- In alcuni sistemi è possibile che un processo si articoli in più thread sequenziali (*lightweight processes*)
- Sono sotto-attività di un processo: ne condividono le risorse (in particolare la memoria)



PAS

Mattia Monga

- I processi devono poter comunicare tra loro in modo strutturato e preciso
 - Per passarsi informazioni
 - Per non disturbarsi reciprocamente quando svolgono attività critiche
 - Per ordinare correttamente le reciproche esecuzioni in presenza di dipendenze
- Problema analogo con i thread tranne il primo caso
 - Condividono la memoria

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

- Due o più processi sono **concorrenti** se la loro esecuzione è *logicamente* sovrapposta nel tempo
- In un sistema monoprocesso l'unico modo per eseguire in modo "concorrente" dei processi è l'**interleaving delle istruzioni**
- Non potendo fare alcuna assunzione temporale, un sistema che supporta l'esecuzione concorrente di processi deve garantire la sua correttezza **indipendentemente dall'interleaving eseguito**
- L'interleaving è **non deterministico** rispetto all'informazione che ha il programmatore di processi concorrenti

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo

Interazione concorrente



PAS

Mattia Monga

Processi concorrenti possono avere la necessità di interagire tra loro per poter evolvere

Comunicazione due o più processi si scambiano informazioni

Contesa due o più processi competono per l'uso della stessa risorsa

Cooperazione due o più processi collaborano alla soluzione di uno stesso problema

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



PAS

Mattia Monga

Si hanno corse critiche (race condition, o alee) quando

- Due o più processi leggono e scrivono dati condivisi
- I risultati finali dipendono dalla particolare sequenza di esecuzione (interleaving)
 - Trovare l'errore è difficile perché si manifesta solo in presenza di particolari temporizzazioni dell'esecuzione

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Basato sull'uso di due primitive di sincronizzazione

- `send(destination, &message)`
 - invia un messaggio ad un processo destinatario
- `receive(source, &message)`
 - riceve un messaggio da un processo.
 - se il messaggio non è ancora disponibile:
 - Il processo si mette in attesa
 - Ritorna un codice d'errore

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Meccanismo di scambio messaggio **sincrono**

- Se send viene effettuata prima di receive, il processo che ha eseguito la send si blocca in attesa che il destinatario effettui la receive
- Lo stesso avviene in caso di receive “anticipata”
- Sender/receiver obbligati ad un'esecuzione di tipo *lockstep*

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



In un sistema di scambio messaggi è fondamentale l'identificazione dei processi coinvolti:

- Un identificatore univoco per ogni processo
- **mailbox**: buffer gestiti dai processi e usati nelle send e receive in sostituzione degli identificatori di processo, una receive su mailbox vuota è bloccante e allo stesso modo un send su mailbox piena.
- le *pipe* di UNIX e MINIX sono mailbox dove i messaggi non hanno dimensione predefinita.

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

La maggior parte dei sistemi per la gestione delle informazioni deve essere in grado di soddisfare i seguenti requisiti:

- Memorizzare grosse quantità di dati
- Garantire che le informazioni memorizzate sopravvivano al processo che le ha generate
- Consentire l'accesso alle informazioni a più processi indipendentemente dal processo che le ha generate

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

- Le informazioni sono memorizzate permanentemente su dischi o dispositivi analoghi in insiemi di dati detti **file**
- I file sono gestiti dal s.o.
- La componente del s.o. dedicata alla gestione dei file è detta **file system**
- In alcune circostanze il termine **file system** è usato anche per indicare l'insieme dei file contenuti in un dato sistema

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

All'utente il file system fornisce la possibilità di

- Creare, cancellare, leggere, modificare, eseguire un file
- Eseguire le operazioni di back-up e recovery
- Definire schemi per controllare chi accede ai propri file
- Strutturare i propri file in base alle proprie necessità
- Spostare e condividere dati tra file

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

- Garantire l'integrità dei dati memorizzati
- Ottimizzare l'uso dello spazio sulla memoria di massa
- Ottimizzare in termini di tempo le prestazioni dei dispositivi
- Supportare dispositivi di memorizzazione diversi
- Minimizzare o eliminare i rischi per la perdita di dati

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Campo Elemento base che contiene un singolo valore. È caratterizzato dal tipo di dato e dalla dimensione

Record Insieme di campi opportunamente correlati che vengono trattati come una singola entità logica

- Esempi: record studente, impiegato, cittadino

File Insieme di record omogenei opportunamente correlati identificato da un nome logico

Database Collezione di record non necessariamente omogenei e delle relazioni tra loro esistenti

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



Il primo problema che il file system deve risolvere proporre agli utenti uno schema attraverso il quale identificare i propri file all'interno di un sistema

- I nomi solitamente possono contenere un qualunque carattere alfanumerico e alcuni fra i caratteri di interpunzione
- I nomi variano in lunghezza (max 8-255 caratteri)
- Alcuni sistemi non distinguono tra maiuscole e minuscole
- È consuetudine usare un suffisso per indicare il tipo di dati memorizzati

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Estensione di un file



PAS

Mattia Monga

Extension	Meaning
file.bak	Backup file
file.c	C source program
file.gif	Compuserve Graphical Interchange Format image
file.hlp	Help file
file.html	World Wide Web HyperText Markup Language document
file.jpg	Still picture encoded with the JPEG standard
file.mp3	Music encoded in MPEG layer 3 audio format
file.mpg	Movie encoded with the MPEG standard
file.o	Object file (compiler output, not yet linked)
file.pdf	Portable Document Format file
file.ps	PostScript file
file.tex	Input for the TEX formatting program
file.txt	General text file
file.zip	Compressed archive

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

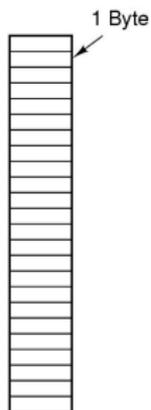
Overlay
Paginazione
Gestione dei page fault
Algoritmo

Organizzazione dei dati

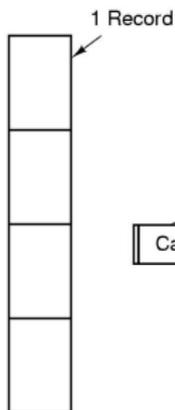


PAS

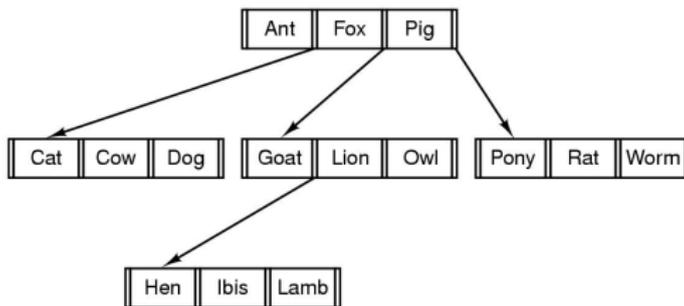
Mattia Monga



(a)



(b)



(c)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



Un file system può supportare diversi tipi di file che differiscono tra loro per l'uso che il sistema ne fa

- Regular file: contengono informazioni degli utenti
 - ASCII
 - Binary
- Link simbolici, Directory: usati dal file system
- Character special file: usati per la gestione di dispositivi ad accesso seriale
- Block special file: usati per la gestione di dispositivi a blocchi

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Le modalità di accesso determinano come un'applicazione può accedere ai dati memorizzati in un file

Accesso Sequenziale

- Legge tutti i bytes/records dall'inizio
- Non è possibile effettuare salti all'interno del file
- È possibile ritornare all'inizio o ripartire dall'ultima operazione di lettura

Random access

- bytes/records possono essere letti in qualunque ordine
- La *read* può avere due diverse forme:
 - 1 Move file marker (seek), then read (file)
 - 2 Read (file, n.ro rec.), Read (file, key)
- Fondamentale per supportare data base

PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Sono informazioni associate al file ed usate dal sistema per svolgere attività di gestione e manutenzione del file:

- Controllo degli accessi
- Back-up
- Gestione degli spazi
- accounting

Sono dati molto dipendenti dal sistema operativo

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Metadati o attributi



Attribute	Meaning
Protection	Who can access the file and in what way
Password	Password needed to access the file
Creator	ID of the person who created the file
Owner	Current owner
Read-only flag	0 for read/write; 1 for read only
Hidden flag	0 for normal; 1 for do not display in listings
System flag	0 for normal files; 1 for system file
Archive flag	0 for has been backed up; 1 for needs to be backed up
ASCII/binary flag	0 for ASCII file; 1 for binary file
Random access flag	0 for sequential access only; 1 for random access
Temporary flag	0 for normal; 1 for delete file on process exit
Lock flags	0 for unlocked; nonzero for locked
Record length	Number of bytes in a record
Key position	Offset of the key within each record
Key length	Number of bytes in the key field
Creation time	Date and time the file was created
Time of last access	Date and time the file was last accessed
Time of last change	Date and time the file has last changed
Current size	Number of bytes in the file
Maximum size	Number of bytes the file may grow to

PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Directory



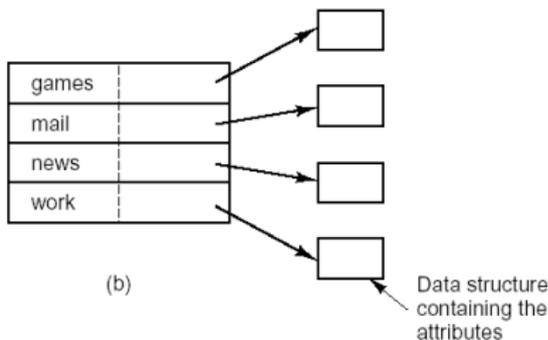
PAS

Mattia Monga

- File utilizzati dal s.o. per la gestione dei file
- Di fatto costituiscono un elenco dei file presenti sul sistema
- Esistono diversi modi per organizzare le directory all'interno di un file system

games	attributes
mail	attributes
news	attributes
work	attributes

(a)



(b)

Data structure containing the attributes

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

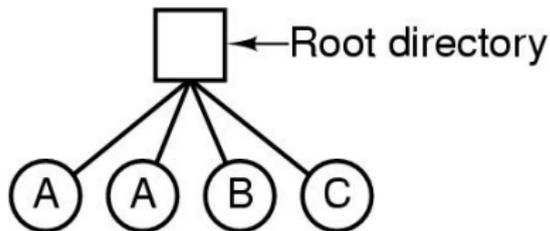
Paginazione

Gestione dei page fault

Algoritmo

Un sistema con directory ad un livello

- contiene 4 file
- di proprietà di 3 persone, A, B, and C
- Ovviamente non possono coesistere due file con lo stesso nome



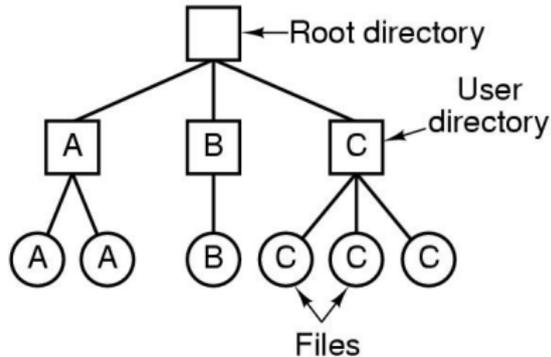
Two-level



PAS

Mattia Monga

Quando un utente accede ad un file, il s.o. deve sapere chi è al fine di individuare il corretto file



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

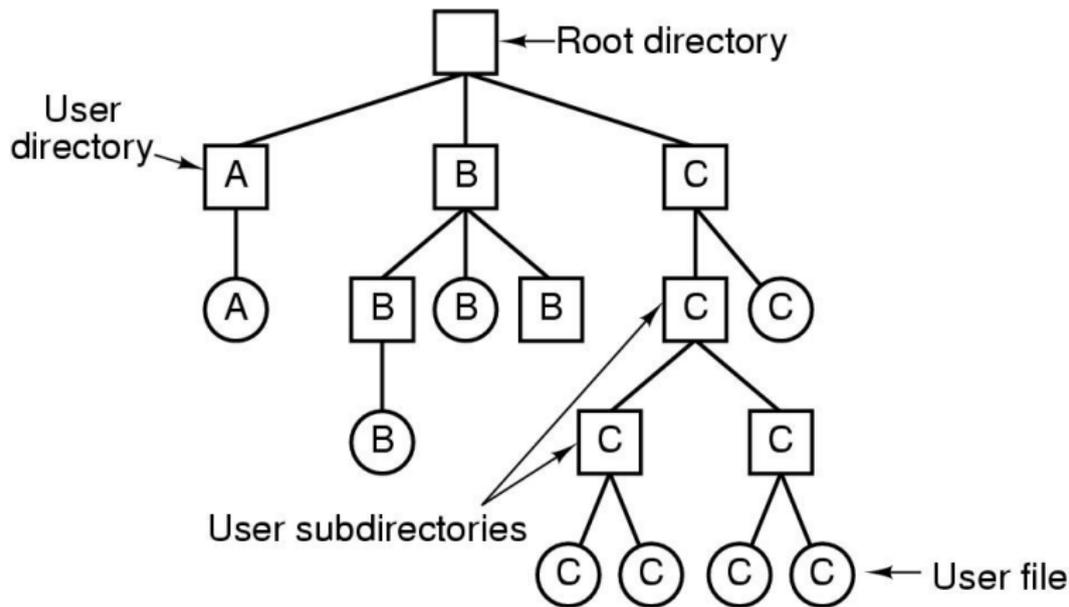
Overlay
Paginazione
Gestione dei page fault
Algoritmo

Gerarchico



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Nei casi di directory con strutture gerarchiche il nome dell'utente e del file non sono piú sufficienti per individuare univocamente un file

- Si usano i **path name**

Assoluti partono dalla radice (root) iniziano sempre con / (\)

Relativi sono computati a partire da una *working directory* (ogni processo ne possiede una)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

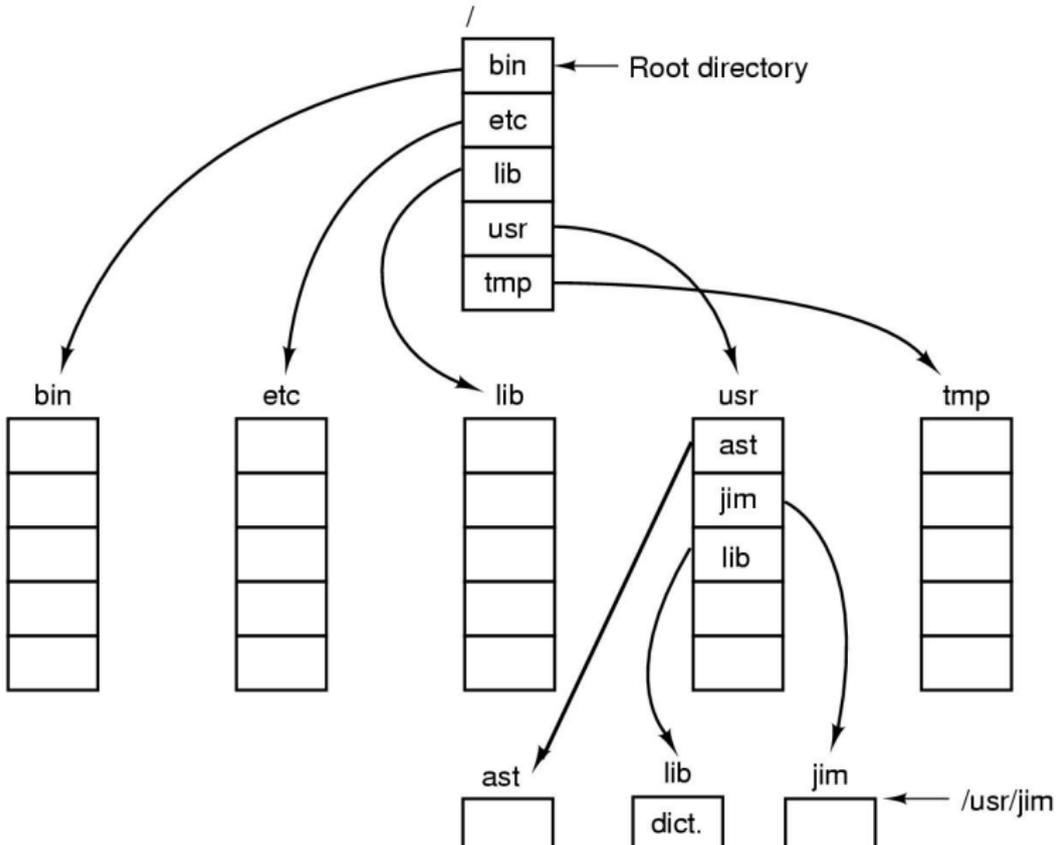
Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Path name



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

- Create
- Delete
- Opendir
- Closedir
- Readdir
- Rename
- Link
- Unlink

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Il FS deve preoccuparsi di tenere traccia dei contenuti dei diversi blocchi del disco, così come dei blocchi liberi, questa attività presuppone però che sia stata definita la modalità di allocazione dei blocchi liberi ai file

- Contigua
- Lista linkata
- FAT
- I-node

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

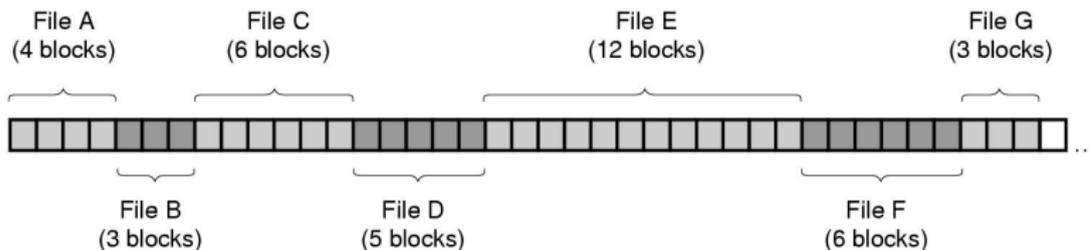
Overlay
Paginazione
Gestione dei page fault
Algoritmo

Allocazione contigua

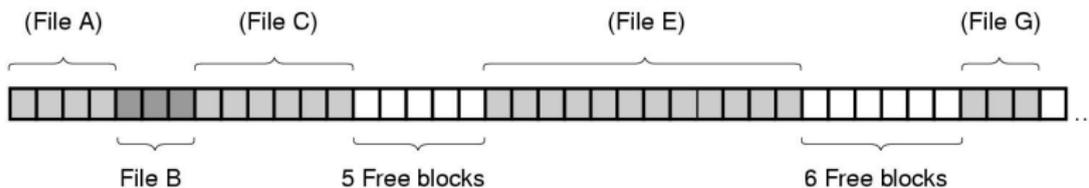


PAS

Mattia Monga



(a)



(b)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

MS-DOS FAT



PAS

Mattia Monga

Block size	FAT-12	FAT-16	FAT-32
0.5 KB	2 MB		
1 KB	4 MB		
2 KB	8 MB	128 MB	
4 KB	16 MB	256 MB	1 TB
8 KB		512 MB	2 TB
16 KB		1024 MB	2 TB
32 KB		2048 MB	2 TB

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

MINIX i-node



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

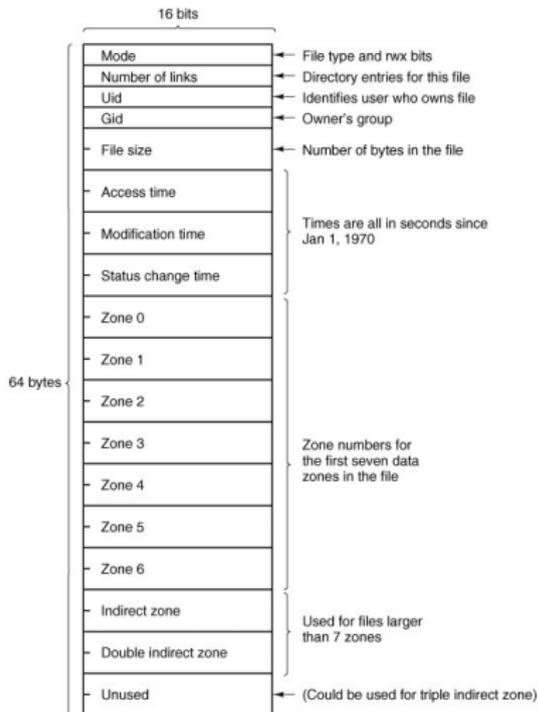
Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Con blocchi da 4KB, zone da 32 bit:

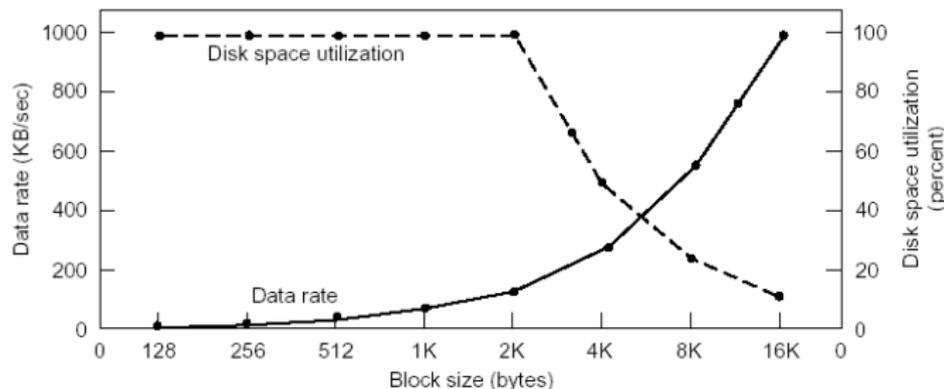
- Qual è la dimensione massima di un file con un solo blocco di overhead?
- Qual è la dimensione massima di un file?
- Quanti blocchi di overhead sono necessari per un file da 100MB di dati?

Gestione dello spazio



PAS

Mattia Monga



Bisogna trovare un compromesso che permetta di usare bene gli spazi e garantisca un data rate accettabile.

Nel 1984 la dimensione media di un file era 1K, nel 2005 la mediana 2475 byte.

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

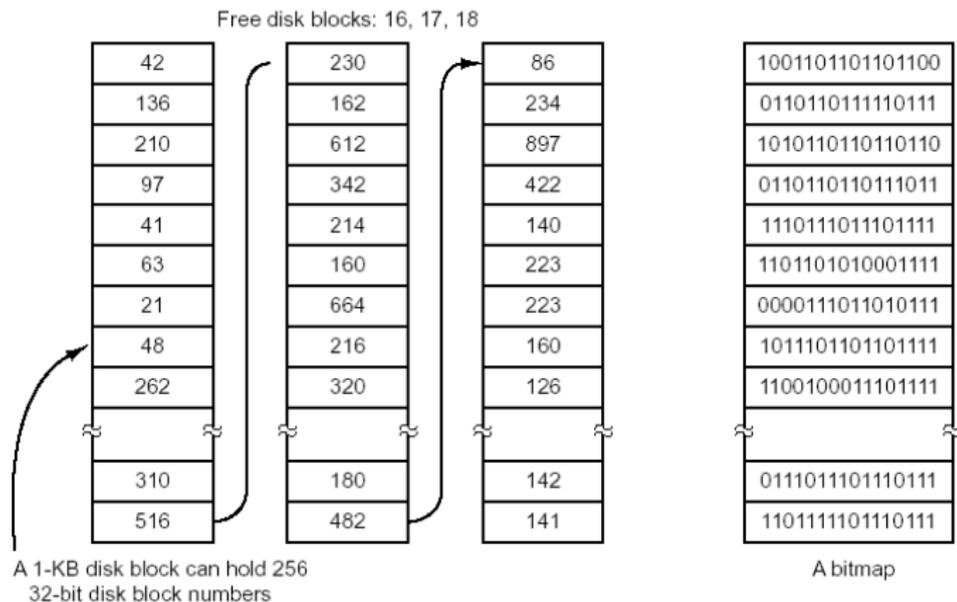
Overlay

Paginazione

Gestione dei page fault

Algoritmo

Free list



Per un disco da 256GB con blocchi da 1K ho bisogno di
 $\frac{2^{28}}{255} = 1052689$ blocchi nella free list. Con la bitmap
 $\frac{2^{28}}{2^3 \cdot 2^{10}} = 32768$ blocchi sono sufficienti.

PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

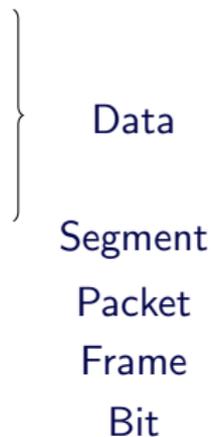
Il modello di riferimento OSI



PAS

Mattia Monga

Application
Presentation
Session
Transport
Network
Data link
Physical



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni
Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Stack dei protocolli Internet



PAS

Mattia Monga

Un modello semplificato (*TCP/IP Illustrated*, W. Stevens)

Application	Telnet, FTP, e-mail, etc.
Transport	TCP, UDP
Network	IP, ICMP, IGMP
Link	device driver and interface card

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

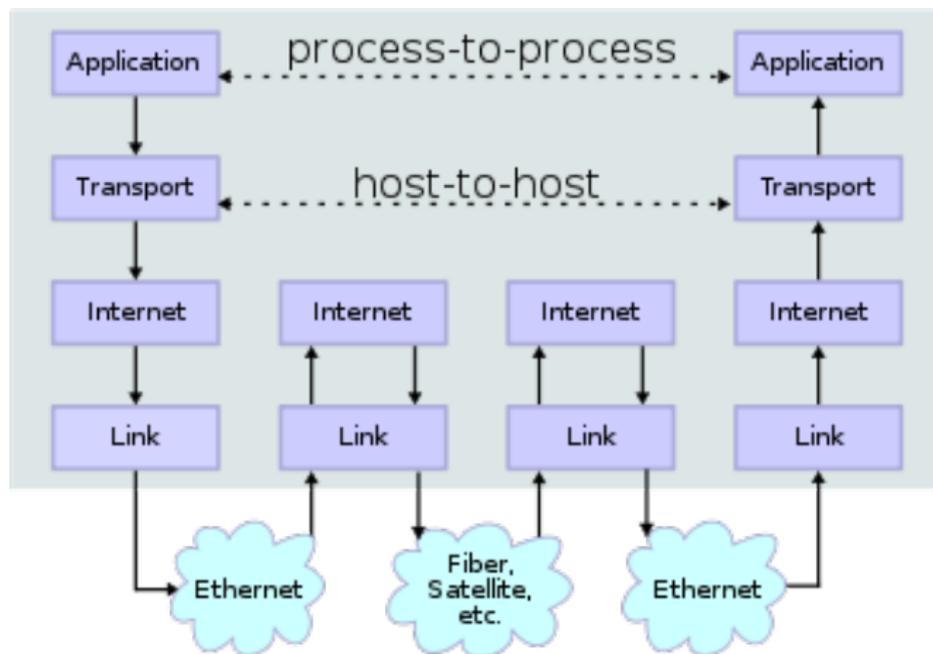
Algoritmo

Stack dei protocolli Internet



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

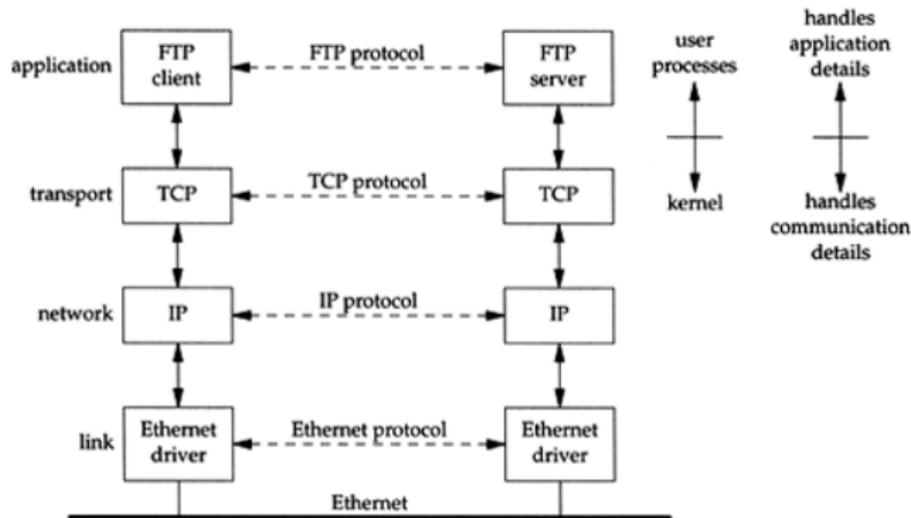
Overlay
Paginazione
Gestione dei page fault
Algoritmo

Stack dei protocolli Internet



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo

Stack dei protocolli Internet



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

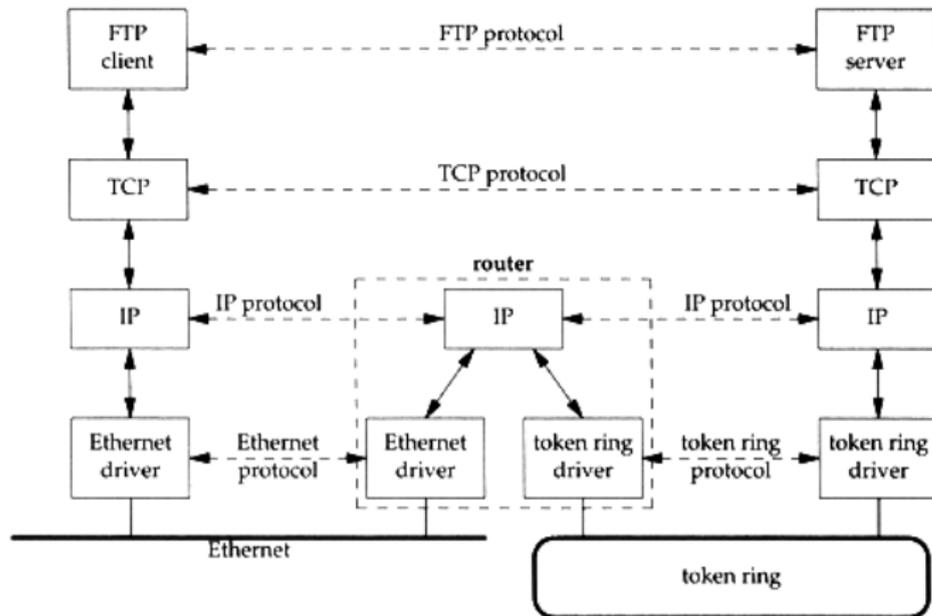
Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo





PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

end-to-end principle L'*intelligenza* ai vertici della rete, che
trasmette i dati nella maniera piú efficiente;
robustness approach Conservatori nel mandare, liberali nel
ricevere.

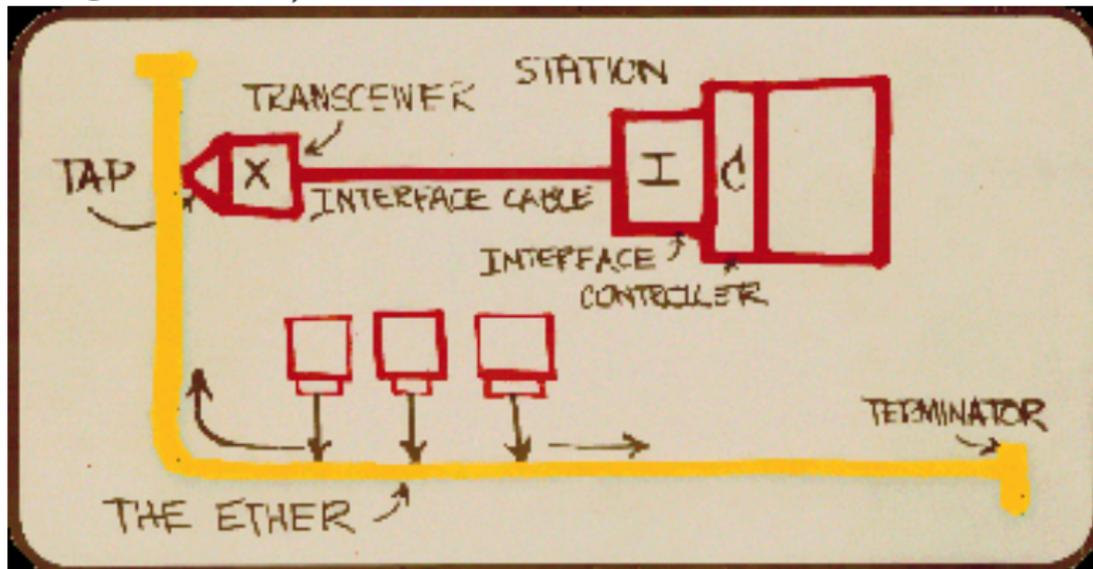
Ethernet



PAS

Mattia Monga

comunicare tramite un medium condiviso (analogo al famigerato *etere*)



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Caratteristiche del protocollo



PAS

Mattia Monga

- Carrier Sense, Multiple Access with Collision Detection
- indirizzi a 48 bit
- maximum transmission unit (MTU): 1500 bytes
- ... ma c'è anche una dimensione minima: 46 byte (ciò costringe al *padding*)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo

Tutti i nodi connessi (LAN) ricevono **tutti** i frame: scartano quelli non diretti a loro.

Hub e switch



PAS

Mattia Monga

L'estensione della rete si amplia con:

Hub semplici ripetitori di segnale
(tutt'al più aiutano nella
collision detection producendo
i *jam frame*)



Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

Hub e switch



L'estensione della rete si amplia con:

Switch definiscono diversi *collision domain*: logicamente LAN differenti, che non condividono fra loro il medium



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Le schede di rete sono identificate da un numero seriale, che viene utilizzato come indirizzo all'interno della LAN.

- 48 bit, i primi 24 identificano il produttore
- notazione esadecimale (MAC: 00:23:a2:d6:f2:15 (Motorola Mobility, Inc.))

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Occorre instradare i pacchetti fra media differenti.

- Ogni nodo è identificato da un **numero IP** da 32 bit (IPv4), tradizionalmente scritto come 4 ottetti (notazione in base 256)
- L'istradamento (*routing*) avviene tramite nodi **gateway** che si interfacciano con due o piú LAN

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Classi di indirizzo



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

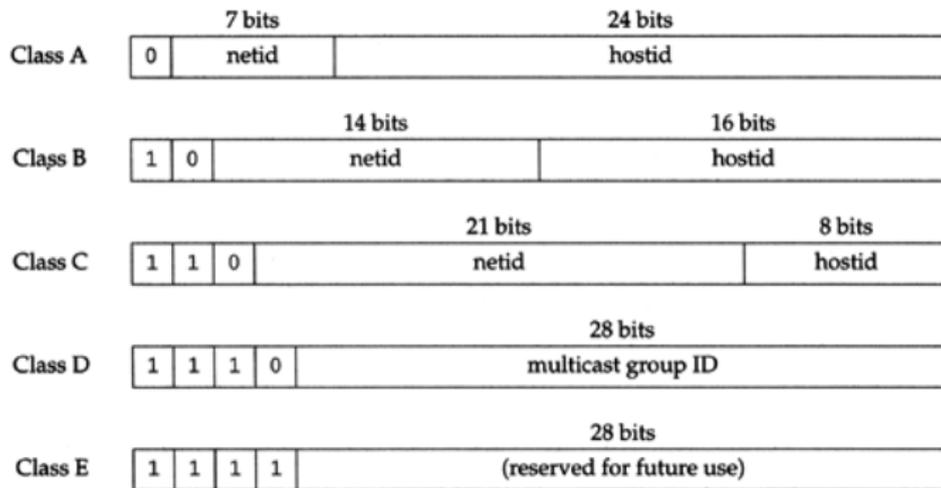
Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Classi di indirizzo



PAS

Mattia Monga

Classe	intervallo	uso
A	0.0.0.0–127.255.255.255	reti tradizionali
B	128.0.0.0–191.255.255.255	reti tradizionali
C	192.0.0.0–223.255.255.255	reti tradizionali
D	224.0.0.0–239.255.255.255	multicast
E	240.0.0.0–255.255.255.255	altri usi speciali

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Classi di indirizzo



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo

MAP OF THE INTERNET
THE IPv4 SPACE, 2006



THIS CHART SHOWS THE IP ADDRESS SPACE ON A PLANE USING A FRACTAL MAPPING WHICH PRESERVES GROUPING--ANY CONSECUTIVE STRING OF IPs WILL TRANSLATE TO A SINGLE COMPACT, CONTIGUOUS REGION ON THE MAP. EACH OF THE 256 NUMBERED BLOCKS REPRESENTS ONE /8 SUBNET (CONTAINING ALL IPs THAT START WITH THAT NUMBER). THE UPPER LEFT SECTION SHOWS THE BLOCKS SOLD DIRECTLY TO CORPORATIONS AND GOVERNMENTS IN THE 1990s BEFORE THE RIRs TOOK OVER ALLOCATION.

0 1 14 15 16 19 →
3 2 13 12 17 18
4 7 8 11
5 6 9 10



UNALLOCATED BLOCK



Classe	intervallo	uso
A	10.0.0.0–10.255.255.255	intranet
B	172.16.0.0–172.31.255.255	intranet
C	192.168.0.0–192.168.255.255	intranet

Secondo le specifiche i router devono *scartare* (o manipolare. . .) i pacchetti contrassegnati con questi indirizzi.

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

La **netmask** è una sequenza di 32 bit che identifica quali bit sono comuni negli IP all'interno di una LAN (sottorete)

01110111 01110111 01110111 11110111 119.119.119.247

7 bit per i nodi $2^7 = 128$

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



Normalmente si usano i primi bit (non obbligatorio), quindi è comoda la notazione CIDR (Classless InterDomain Routing)

159.149.30.0/24 24 bit per le sottoreti, $32 - 24 = 8$ per gli host

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

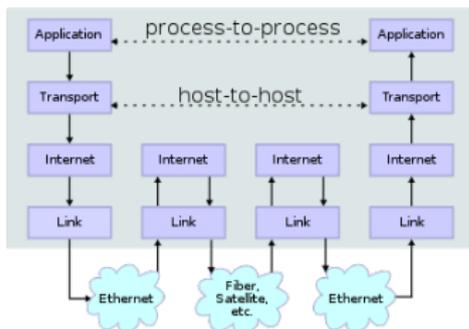
Algoritmo

Il livello di trasporto



PAS

Mattia Monga



Poiché a livello applicativo la comunicazione avviene fra **processi**, a livello trasposto occorre identificare **nodi** e **processi**.

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

Un segmento di scambio fra due processi necessita di **4** numeri

$$\langle ip_1, n_1 : ip_2, n_2 \rangle$$



Port

n_1, n_2 (0–65536) si dicono **porte**: quelle lato server devono essere note al client e rappresentano quindi il punto *d'accoglienza*.

Nota: il **client** è il nodo che **inizia** la connessione con il **server**.

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



da <http://www.iana.org/assignments/port-numbers>

```
discard 9/tcp sink null
discard 9/udp sink null
ftp-data 20/tcp
ftp 21/tcp
ssh 22/tcp # SSH Remote Login Protocol
ssh 22/udp
telnet 23/tcp
smtp 25/tcp mail
domain 53/tcp # name-domain server
domain 53/udp
finger 79/tcp
www 80/tcp http # WorldWideWeb HTTP
pop3 110/tcp pop-3 # POP version 3
nttp 119/tcp readnews untp # USENET News Transfer Protocol
ntp 123/udp # Network Time Protocol
irc 194/tcp # Internet Relay Chat
https 443/tcp # http protocol over TLS/SSL
printer 515/tcp spooler # line printer spooler
# ...
```

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Porte ben note



PAS

Mattia Monga

```
# Non fissate da IANA
socks 1080/tcp # socks proxy server
openvpn 1194/tcp
openvpn 1194/udp
rmiregistry 1099/tcp # Java RMI Registry
# ...
```

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

Ricordare sempre che le porte sono numeri **convenzionali**
(concordate con IANA per i numeri ≤ 1024)

- in generale **non** identificano un servizio, ma la possibilità di stabilire una connessione.



Transmission Control Protocol

- **connection-oriented**: è necessario uno handshake preliminare
- **full-duplex**
- lo “stato” è conservato interamente nei nodi (+ timer)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

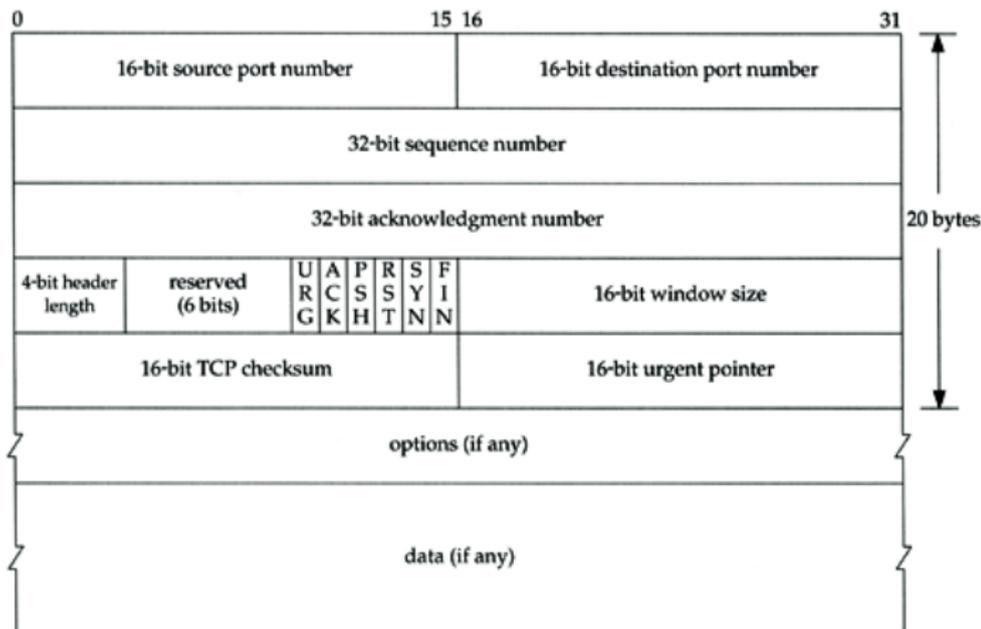
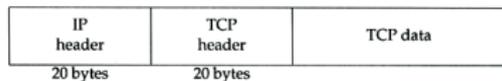
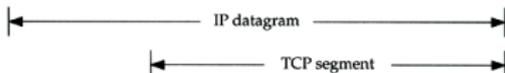
System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

TCP segment



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



- SYN** richiesta di connessione, sempre il primo pacchetto di una comunicazione
- FIN** indica l'intenzione del mittente di terminare la sessione in maniera concordata
- ACK** conferma del pacchetto precedente, sia esso dati, SYN o FIN

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



- RST reset della sessione
- PSH operazione di push, i dati vengono subito inviati al destinatario senza bufferizzarli
- URG dati urgenti (es. CTRL+C) vengono inviati con precedenza sugli altri

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

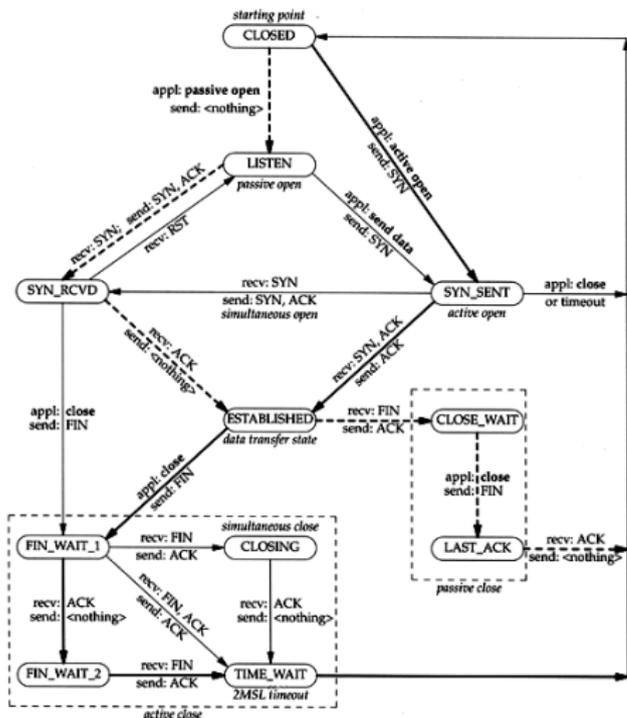
Overlay
Paginazione
Gestione dei page fault
Algoritmo

State diagram



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



User Datagram Protocol

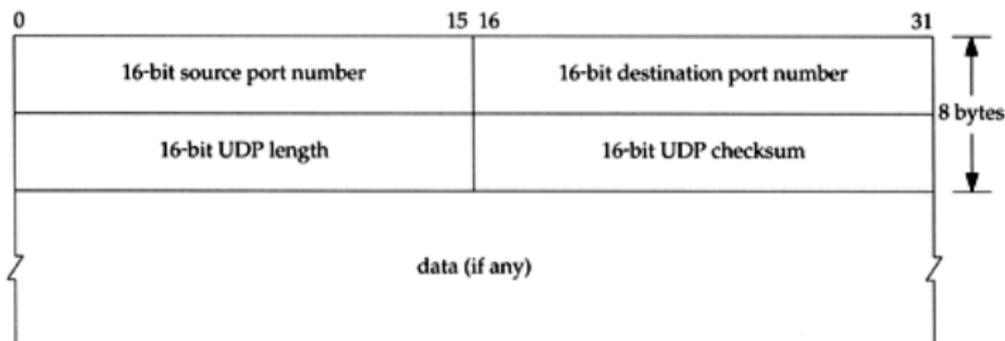
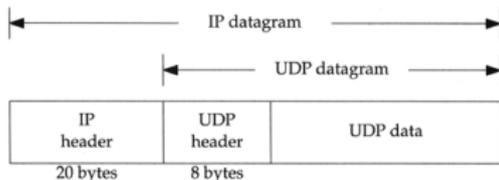
- Protocollo di trasporto “minimo”, senza connessione, senza stato
- minimo overhead (TCP: +20B, UDP: +8B)

UDP segment



PAS

Mattia Monga



Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

Sia TCP che UDP portano nei segmenti un checksum.

Attenzione: ha lo scopo di proteggere solo dagli **errori di trasmissione**, non dalle alterazioni maligne!



PAS

Mattia Monga

L'accesso ai servizi critici è controllato

- **Autenticazione:** **chi è l'agente** (che opera in nome di un *principal*)
- **Autorizzazione:** l'agente autenticato **ha il permesso?**
- **Confidenzialità:** un dato è confidenziale se può essere conosciuto solo dagli agenti autorizzati.

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Software di
sistema e
applicativo

Le astrazioni
del s.o.

Astrazioni
Componenti

Modo kernel e
utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-
programmati
Sistemi multi-
programmati
Paginazione

Memoria
virtuale

Overlay
Paginazione
Gestione dei
page fault
Algoritmo

Autenticazione

Autenticare significa verificare **l'identità** di un soggetto (non necessariamente umano)



Modalità di base per l'autenticazione (di Alice) tramite rete:

- 1 **password** (ossia la conoscenza di un segreto)
- 2 **locazione** (logica o fisica) da cui proviene la richiesta di autenticazione
- 3 per mezzo di operazioni crittografiche su dati forniti dall'autenticatore (Bob).

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo



PAS

Mattia Monga

Alcune vulnerabilità sono intrinseche:

- Le password possono essere **indovinate**
- Le locazioni possono essere **millantate**
- I dati crittografici possono essere **intercettati e riutilizzati** (replay attack)

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.
Sistemi mono-programmati
Sistemi multi-programmati
Paginazione

Memoria virtuale

Overlay
Paginazione
Gestione dei page fault
Algoritmo

Challenge/Response



PAS

Mattia Monga

Software di sistema e applicativo

Le astrazioni del s.o.

Astrazioni Componenti

Modo kernel e utente

Interruzioni

System call

Arch. di un s.o.

Sistemi mono-programmati

Sistemi multi-programmati

Paginazione

Memoria virtuale

Overlay

Paginazione

Gestione dei page fault

Algoritmo



- L'autenticazione non è reciproca: qualcuno potrebbe sostituirsi a Bob.
- Offline-guessing di K_{AB} è possibile intercettando R e $K_{AB}\{R\}$