

# **RICHIAMI DI CALCOLATORI ELETTRONICI**

Sistema Operativo:

- 1) Componenti
- 2) Servizi
- 3) Interfaccia verso l'utente e il programmatore

Dal punto 1:

Gestione dei processi:

(processo = programma in esecuzione)

- creare e cancellare processi
- sospendere e riattivare processi
- fornire meccanismi per la sincronizzazione dei processi
- fornire meccanismi per la comunicazione dei processi
- fornire meccanismi per il controllo dello stallo dei processi (deadlock)

Gestione della memoria principale:

- tenere traccia di aree di memoria occupate e il processo che le occupa
- decidere quali processi caricare in memoria
- allocare e deallocare la memoria

Gestione dei file:

- creare e cancellare file
- creare e cancellare directory
- supportare primitive per manipolare file e directory
- mappare i file sulla memoria secondaria
- backup

Gestione dispositivi I/O:

- caching, buffering, spooling
- interfaccia generica verso driver device
- driver dei dispositivi

Gestione memoria secondaria:

- gestire spazio libero
- allocare memoria
- (scheduling del disco)

Punto 2, servizi:

- operazioni di I/O
- manipolazione del file system
- comunicazione
- rilevazione errori

Dal punto 3:

System Call:

- controllo dei processi
- end / abort
- load /execute
- wait / signal event

File:

- create / dolete file
- open / close file
- read / write file

Dispositivi:

- request / release device
- read / write reposition
- get / set device attributes

Informazioni:

- set / get date
- set / get attributes

L'ultima categoria di system call è:

Comunicazione:

- create / dolete communication channel
- send / receive message

Struttura del sistema operativo

La struttura del sistema operativo può essere semplice, complessa, stratificata.

L'MS-DOS è nato con una struttura relativamente semplice con moduli comunicanti fra loro, il tutto non è stato progettato per il successo che poi avuto e quindi non era prevista una sua espansione, ma l'obiettivo era quello di realizzare un sistema funzionante nel minor spazio possibile.

Nell'MS-DOS abbiamo possiamo schematizzare 4 blocchi posti uno sopra l'altro.

- application program
- resident system program
- MS-DOS device drivers
- BIOS ROM device drivers

Ogni blocco comunica con quello sottostante.

In più application program può comunicare direttamente con il livello BIOS e lo stesso il livello resident system program.

Unix

Kernel

## System programs

Quella Unix è poco espandibile perché è quasi tutto contenuto nel Kernel e modificare una cosa significa andare a intaccare anche tutto il resto.

Struttura stratificata: la struttura stratificata a layers si può espandere facilmente ma può comportare un rallentamento del sistema.

Vi sono soluzioni intermedie, ad esempio quella dell'OS/2.

## Microkernel

Il kernel fa da tramite.

Per microkernel si intende il kernel ridotto alle dimensioni minime con le funzioni essenziali cercando di delegare tutto il resto.

Sul microkernel Mach è stato basato anche il sistema operativo Max OS X.

## Macchine virtuali

I primi passi sono stati i VM IBM VM = Virtual Machine

## JAVA

1995 – Java

Più che un linguaggio di programmazione è anche una tecnologia.

Comprende:

- 1) Specifiche del linguaggio
- 2) API
- 3) JVM Java Virtual Machine

Le ultime due costituiscono la Java Platform.

Dal punto 1:

- oggetti
- indipendente dalla piattaforma
- sicuro
- distribuito (classi presenti su vari calcolatori connessi fra loro)
- multithread

Dal punto 2:

- JAVA.LANG (insieme di classi che definiscono la struttura base del linguaggio)
- JAVA.IO / JAVA.NIO
- JAVA.AWT
- JAVA.UTIL

Seguono vari esempi e piccoli cenni su Java che attualmente non riporto come molti altri schemi / disegni sulle strutture dei sistemi operativi, il tutto sostanzialmente per mancanza di tempo.