

[www.dis.uniroma1.it/~midlab](http://www.dis.uniroma1.it/~midlab)

---

# **Sistemi Operativi II**

## **Corso di Laurea in Ingegneria Informatica**

**Prof. Roberto Baldoni**

### **Introduzione**



# OS=Astrazione

---

- Dare l'illusione all'applicazione di memoria infinita, CPU infinita, unico "worldwide" computer ecc.;
- Quando si vuole aumentare il livello di astrazione, tre problemi devono essere considerati:
  - Coordinazione
  - Sicurezza
  - Comunicazione
- Ognuno dei precedenti aspetti viene risolto attraverso algoritmi/protocolli (implementati attraverso librerie e/o demoni) che devono rappresentare un compromesso tra semplicità e prestazioni

# OS=Piattaforma abilitante

- Un sistema operativo rappresenta di fatto una piattaforma abilitante per le applicazioni;
- Poiche le applicazioni cambiano nel tempo e si stratificano, ne segue che la nozione di astrazione non e' un concetto statico ma cambia nel tempo
- Ciò che è applicazione oggi puo' potenzialmente diventare parte del sistema operativo di domani.
  - Esempi: NFS in solaris, Browser in windows, smtp in unix
- Diventare parte di un SO significa diventare parte di una piattaforma abilitante per le applicazioni

# Astrazione



# OS: Fasi storiche ('40-'79)

- Fase (i): batch processing
  - Ogni lavoro viene caricato, processato e terminato attraverso funzioni di libreria direttamente inserite all'interno del programma;
  - Non è permesso multitasking
  - E.g. univac
- Fase (ii): interactive timesharing
  - Multitasking abilitato attraverso algoritmi di scheduling, di gestione memoria e del file system, e dall'overlapping tra I/O e processing abilitato dalla tecnologia DMA; tutti questi algoritmi formano, insieme alle librerie di funzioni richiamabili da utente, il SO
  - SO proprietari
  - gli utenti interagiscono con il sistema attraverso terminali "stupidi"
  - E.g., Mainframe, minicomputer e workstation (sistemi molto costosi)
  - L'evoluzione di tali SO proprietari porterà verso la realizzazione delle prime versioni di unix per abbattere i costi e per l'interoperabilità rispetto all'hardware

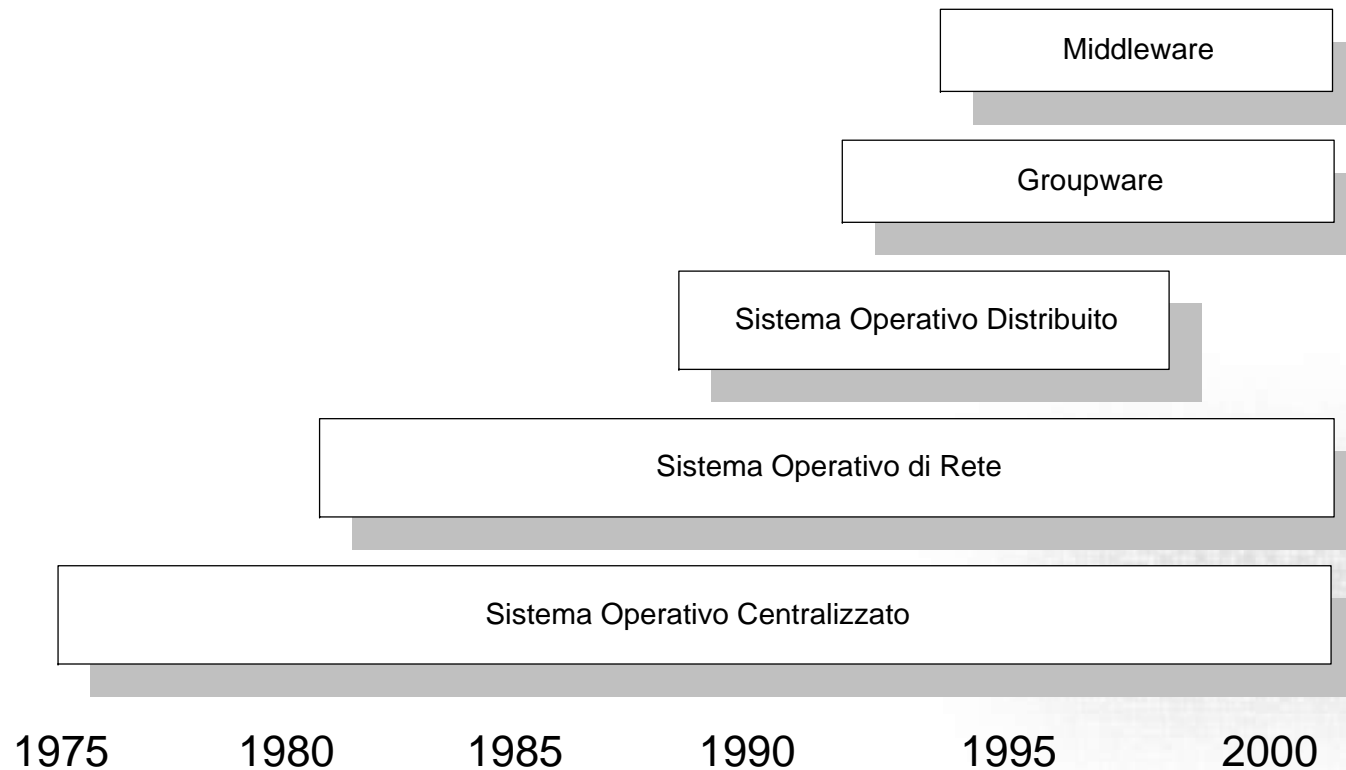
# OS: Fasi storiche "la rivoluzione dei PC" '80 ad oggi

- Fase (iii): PC-prima generazione
  - Abbattimento dei costi
  - Ogni lavoro viene caricato, processato e terminato attraverso funzioni di libreria direttamente inserite all'interno del programma....si ripete la storia
  - Non è permesso multitasking (solo attraverso particolari programmi chiamati TRS)
  - E.g. MSDOS
- Fase (iv): PC-seconda generazione
  - Multitasking abilitato attraverso algoritmi di scheduling, di gestione memoria e del file sistem; tutti questi algoritmi formano, insieme alle librerie di funzioni richiamabili da utente, il SO
  - Iniziano a sentirsi i problemi di sicurezza (windows 2k)
  - E.g., Mac-OS, Windows 2K, Linux

# OS: Fasi storiche

- Fase (v): Sistemi distribuiti
  - Inserimento all'interno del SO di quelle funzioni di libreria e demoni che permettono
    - Coordinamento
    - Sicurezza
    - Comunicazione
  - tra gruppi di PC attraverso una rete di elaboratori
  - e.g., RPC, middleware etc.
- Fase (vi): Sistemi pervasivi
  - Inserimento all'interno del SO di quelle funzioni di libreria e demoni che permettono
    - Coordinamento
    - Sicurezza
    - Comunicazione
  - tra gruppi di sistemi (PC,PDA, sensori, attuatori ecc.)
  - Convergenza di tecnologie telefoniche e di rete wireless e wired
  - E.g. Ambient intelligence, internet everywhere

# SO: Evoluzione legata alla distribuzione



# SO: Evoluzione legata alla distribuzione

Architettura	Caratteristiche Principali	Obiettivi
<i>Sistema operativo centralizzato</i>	Gestione dei processi Comunicazioni tra processi locali Gestione della memoria Gestione Input/Output Gestione del file system	virtualizzazione
<i>Sistema operativo di rete</i>	Comunicazione tra processi remoti Scambio di informazioni remote Ricerca di informazioni remote	Interoperabilità a livello di trasporto (TCP/IP)
<i>Sistema Operativo distribuito</i>	Visione di un insieme di elaboratori connessi in rete come un singolo elaboratore. Visione globale: del file system, dello spazio dei nomi, del tempo, della sicurezza e della potenza computazionale. In alcuni sistemi visione globale anche della memoria.  Una parte di tali sistemi si sono evoluti nei cluster di PC	Trasparenza (interoperabilità a livello di trasporto opzionale)

# SO: Evoluzione legata alla distribuzione

<i>Groupware</i>	Visione di un insieme di processi (gruppo) come una unica entità. Comunicazione tra entità uno-a-molti, gestione dei membri del gruppo.	Affidabilità (attraverso ridondanza software)
<i>Middleware</i>	Visione di una applicazione distribuita come un insieme di entità cooperanti al di sopra di SO eterogenei. Servizi per la localizzazione di una entità, servizi per la scoperta delle operazioni eseguibili da una entita, servizi per la sincronizzazione temporale, per la sicurezza etc. Esempio di applicazioni include l'integrazione di sistemi informativi aziendali.	Autonomia e cooperazione (interoperabilità a livello di sistemi operativi locali)

# Evoluzione di un SO commerciale

