

Sistemi a processori multipli

Sommario

- Classificazione e concetti di base
 - Sistemi multi-processore
 - Sistemi multi-computer (cluster)
 - Sistemi distribuiti
- Obiettivo comune
 - Risolvere problemi di dimensioni considerevoli, a volte intrattabili in architetture tradizionali (scale-up)
 - Risolvere problemi trattabili anche con architetture tradizionali, ma in minor tempo (speed-up)

CLASSIFICAZIONI E CONCETTI DI BASE

Sistemi con processori multipli

- Varie classificazioni possibili
- Determinate da:
 - organizzazione dei processori
 - organizzazione delle memorie
 - struttura della relativa interconnessione (HW)
 - accoppiamento tra SW e HW sui vari processori
- Diverse problematiche per i relativi S.O.!

CLASSIFICAZIONE HW

Classificazioni HW

- Classificazione HW
 - Tassonomia di Flynn
- Classificazione dei MIMD
 - Tassonomia di Tannenbaum
 - Tassonomia di Stallings

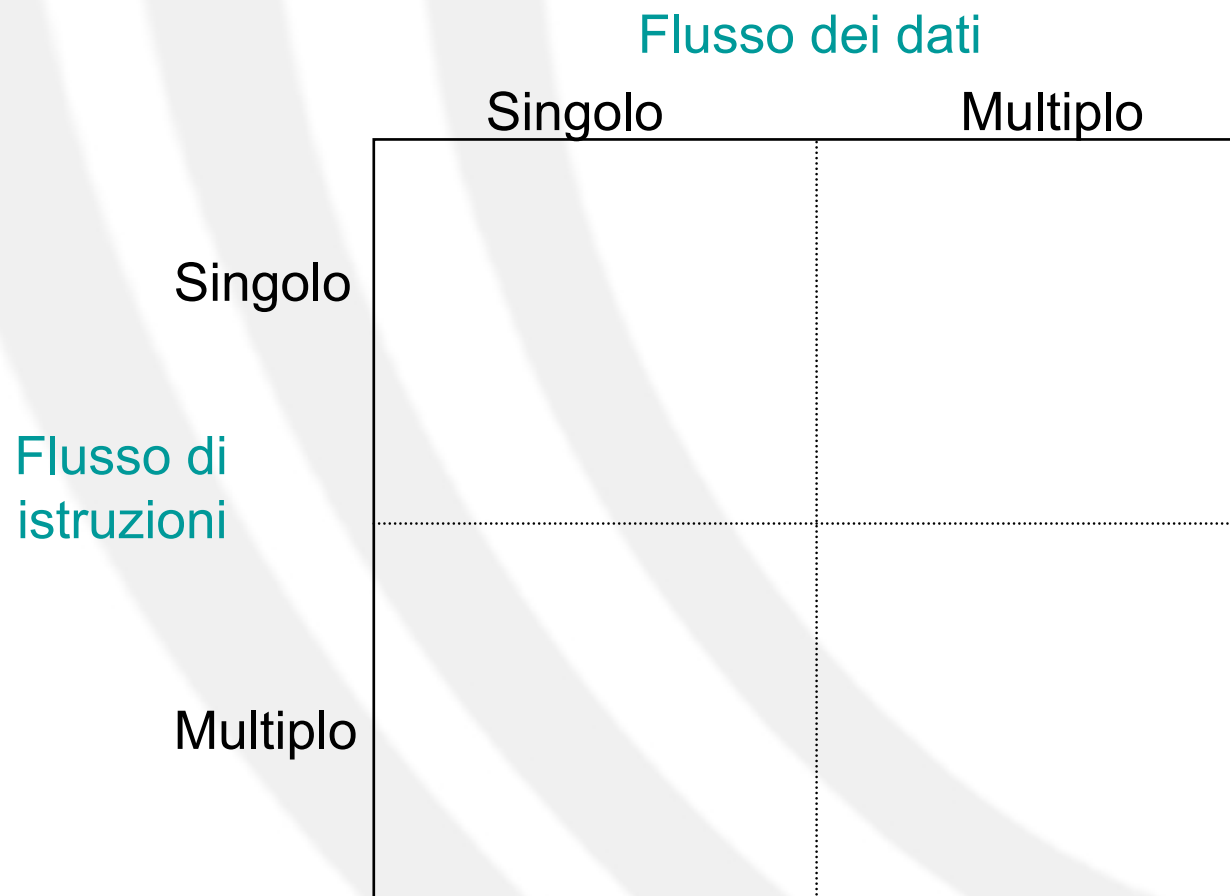
Tassonomia di Flynn (1966)

Flusso dei dati

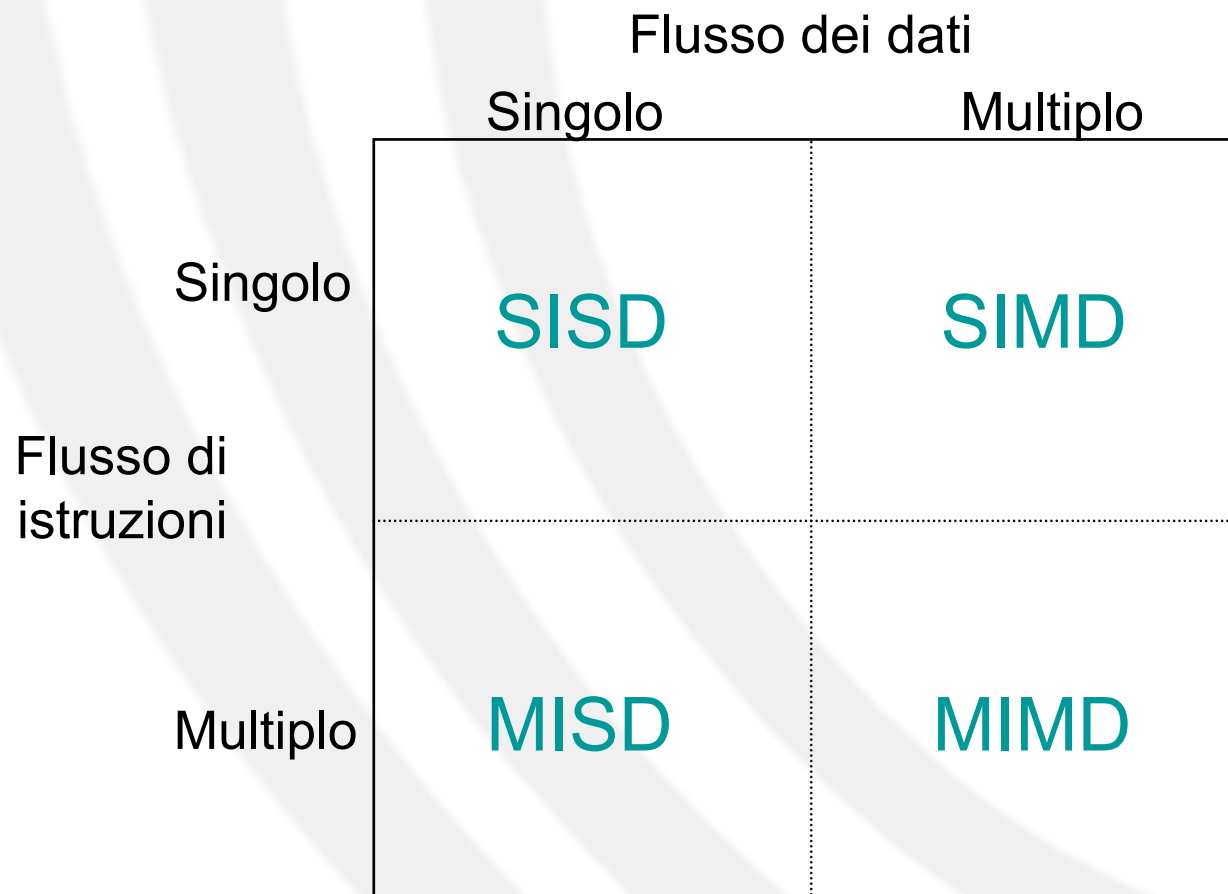
Flusso di
istruzioni



Tassonomia di Flynn



Tassonomia di Flynn

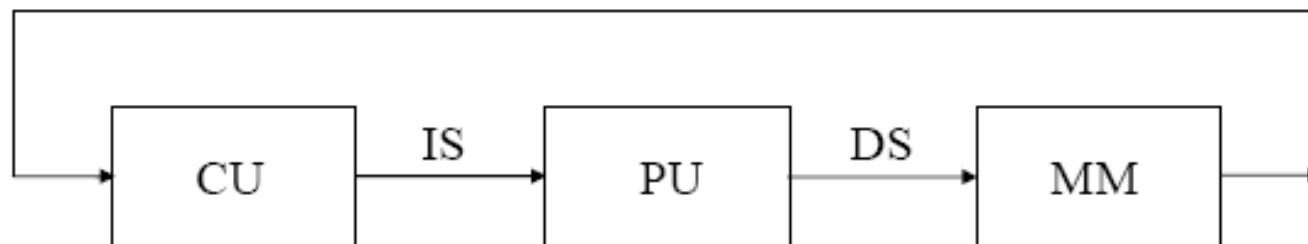


Tassonomia di Flynn

- Gli elementi fondamentali di un calcolatore sono
 - Flusso di istruzioni (IS):
 - programma da eseguire
 - Flusso di dati (DS):
 - operandi e risultati delle istruzioni
 - Unità di controllo (CU):
 - esegue prelievo e decodifica dell'istruzione
 - Unità di elaborazione (PU):
 - esegue le istruzioni (ALU+registri)
 - Memoria principale (MM):
 - contiene istruzioni e dati

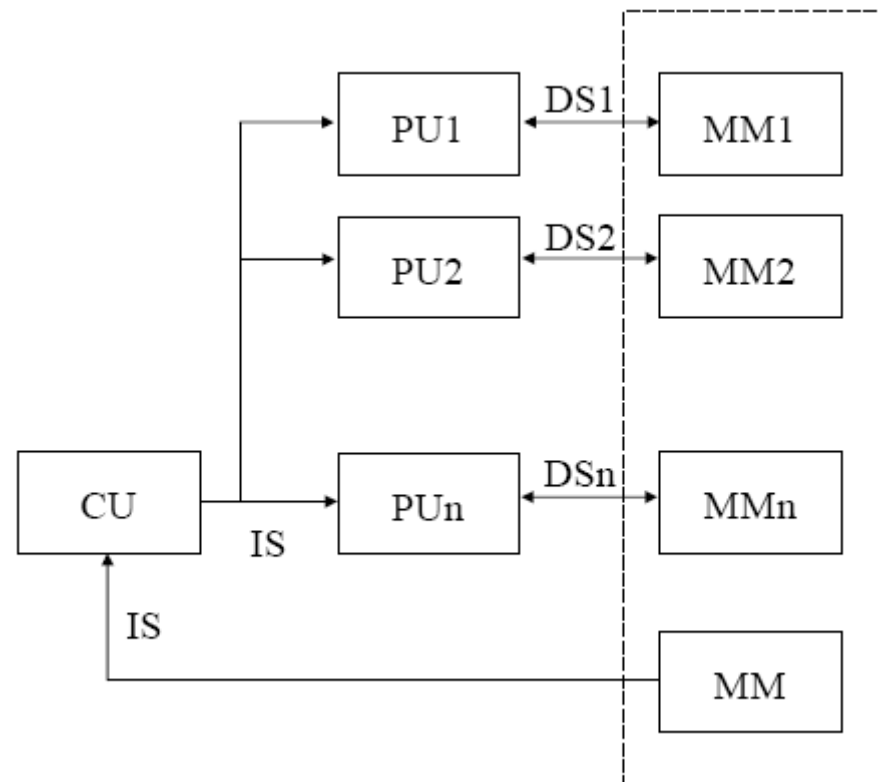
Tassonomia di Flynn

- SISD
 - Un processore
 - Una memoria
 - Un flusso di istruzioni, ogni istruzione applicata ad un dato alla volta
 - Si tratta della macchina di Von Neumann



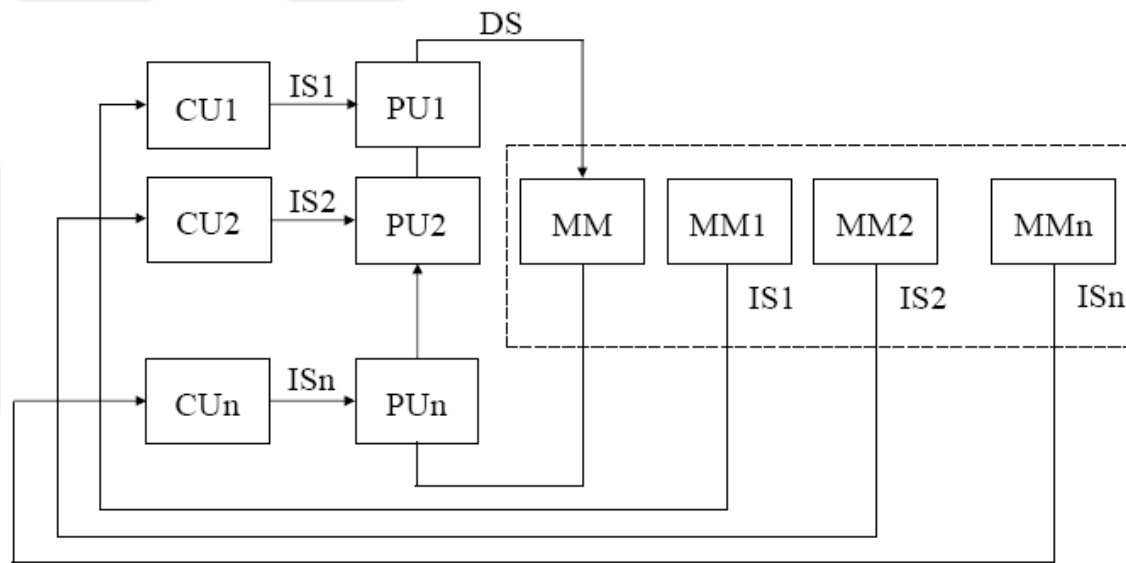
Tassonomia di Flynn

- SIMD
 - Più processori
 - Ognuno con la “propria memoria”
 - Un flusso di istruzioni, applicato simultaneamente a vari dati
 - Es.: array processor



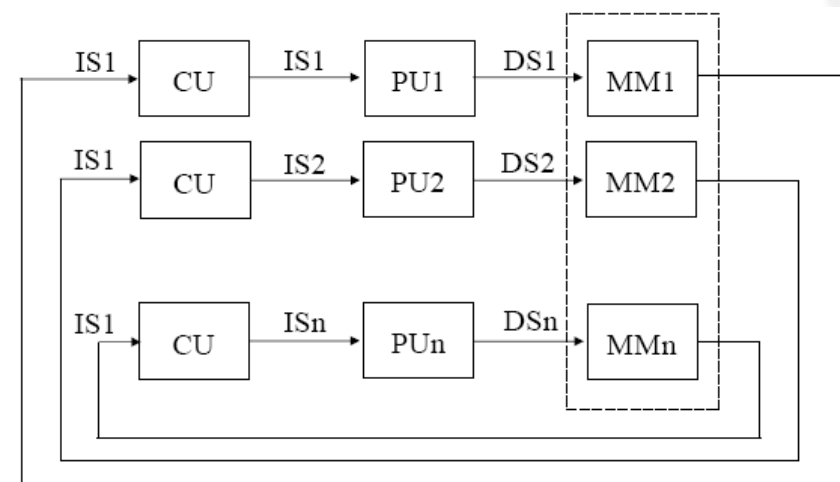
Tassonomia di Flynn

- MISD
 - Più processori
 - Una memoria
 - Un flusso dati, sul quale sono applicati i vari flussi di istruzioni
 - MAI implementato



Tassonomia di Flynn

- MIMD
 - Più processori
 - Più memorie
 - Più flussi dati, sui quali vengono eseguite le diverse sequenze di istruzioni
 - Caso “interessante”
 - Tutti i sistemi distribuiti sono MIMD



Classificazione dei MIMD

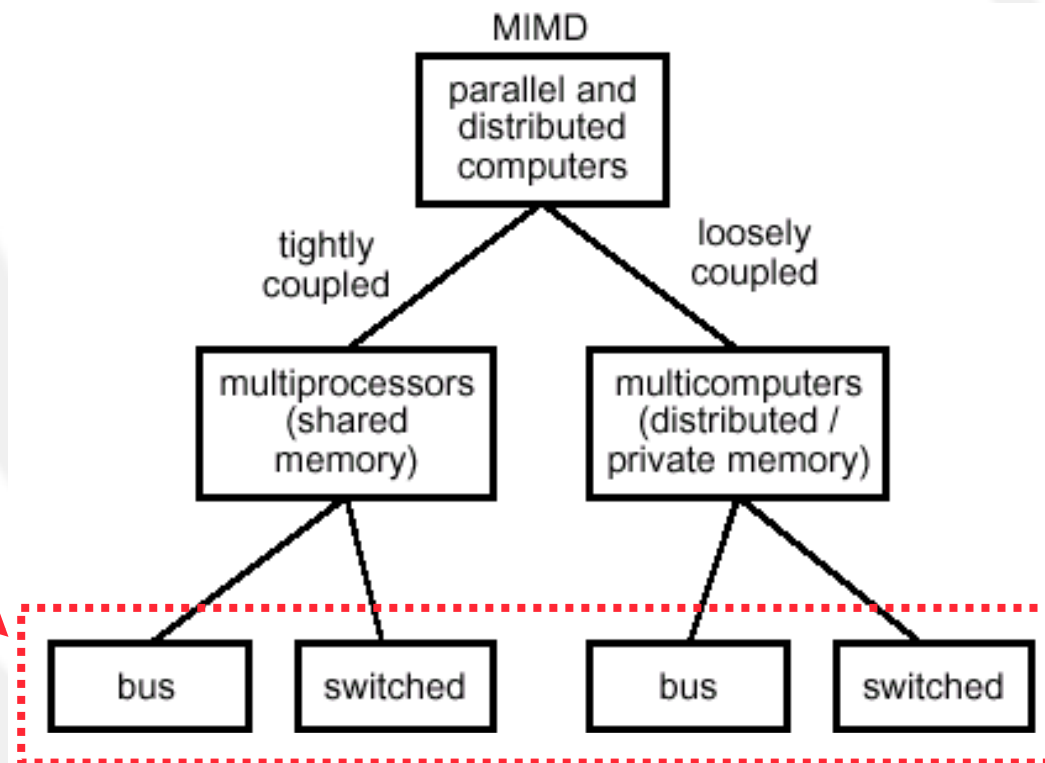
- Con memoria condivisa (multiprocessori)
 - Detti anche “*tightly coupled*”
 - Condividono clock e memoria
 - Eseguono lo stesso S.O.
 - Comunicazioni “frequenti” e veloci
 - Velocità di comunicazione = velocità della memoria
 - Adatti a lavorare in parallelo su un singolo problema

Classificazione dei MIMD

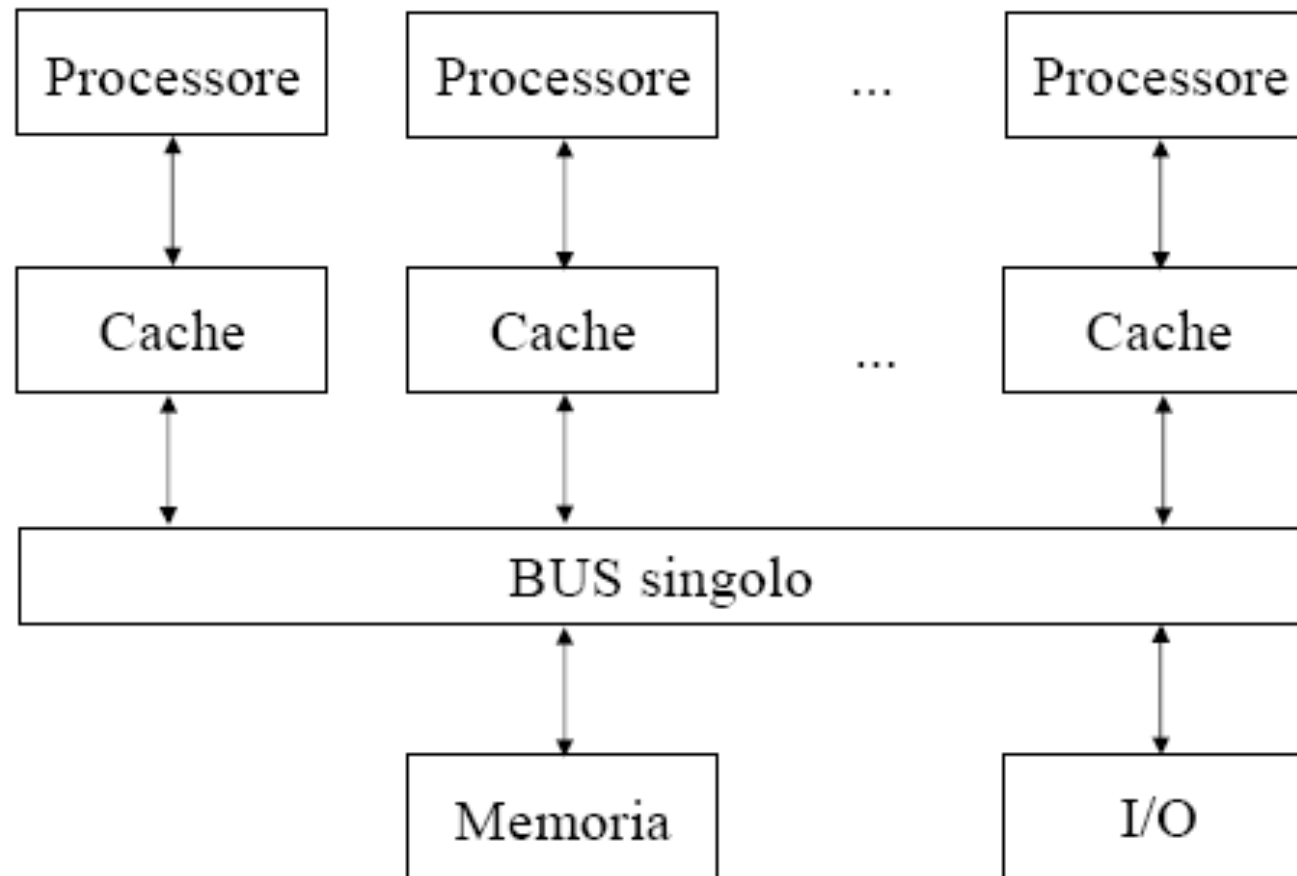
- Con memoria distribuita (multicalcolatori)
 - Detti anche “loosely coupled”
 - Memoria locale ad ogni macchina
 - Eseguono diversi S.O.
 - Comunicazioni “occasional” e lente
 - Velocità comunicazione = velocità della rete
 - Adatti a lavorare su molti problemi differenti contemporaneamente

Tassonomia di Tannenbaum

- In base al tipo di “connessione fisica”



Sistemi multiprocessore a bus



Sistemi multiprocessore a bus

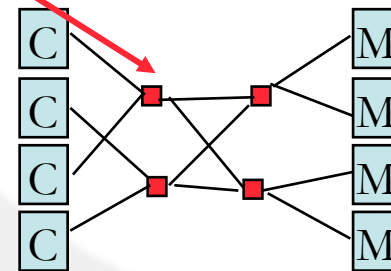
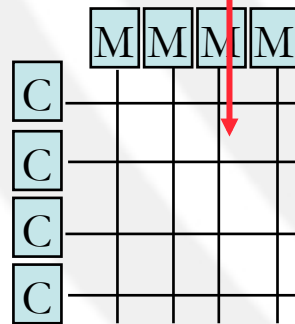
- N CPU connesse ad un singolo bus e a un'unica memoria
- Architettura semplice, permette broadcast, ma poco scalabile a causa del bus
 - Ogni CPU ha una cache per evitare il collassamento del bus
 - Problema: coerenza delle cache con la memoria
 - Soluzione: es.: snoopy write-through cache
- SMP (Symmetric MultiProcessor)

Sistemi multiprocessore a bus

Nome	Massimo numero di proc.	Nome del proc.	Freq. di clock del proc. (MHz)	Memoria max per sistema (GB)	Banda passante max per sistema (MB/sec)
Compaq ProLiant 5000	4	Pentium Pro	200	2	540
Digital AlphaServer 8400	12	Alpha 21164	440	28	2150
HP 9000 K460	4	PA-8000	180	4	960
IBM RS/6000 R40	8	PowerPC 604	112	2	1800
SGI Power Challenge	36	MIPS R10000	195	16	1200
Sun Enterprise 6000	30	UltraSPARC 1	167	30	2600

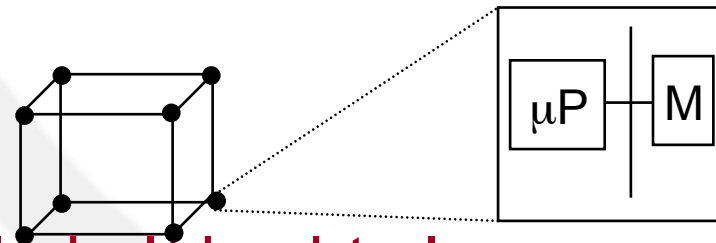
Sistemi multiprocessore a commutazione

- Sistemi multiprocessore a commutazione
 - N processori con N memorie
 - Collegamento basato su switch
 - A griglia (crossbar switch, 2 livelli) [n^2 switch] (costosi)
 - Omega network (switch 2x2) [$n \log n$ switch] (costosi e lenti)
 - Gerarchici (NUMA, Non-Uniform Memory Access) (allocazione SW complessa)

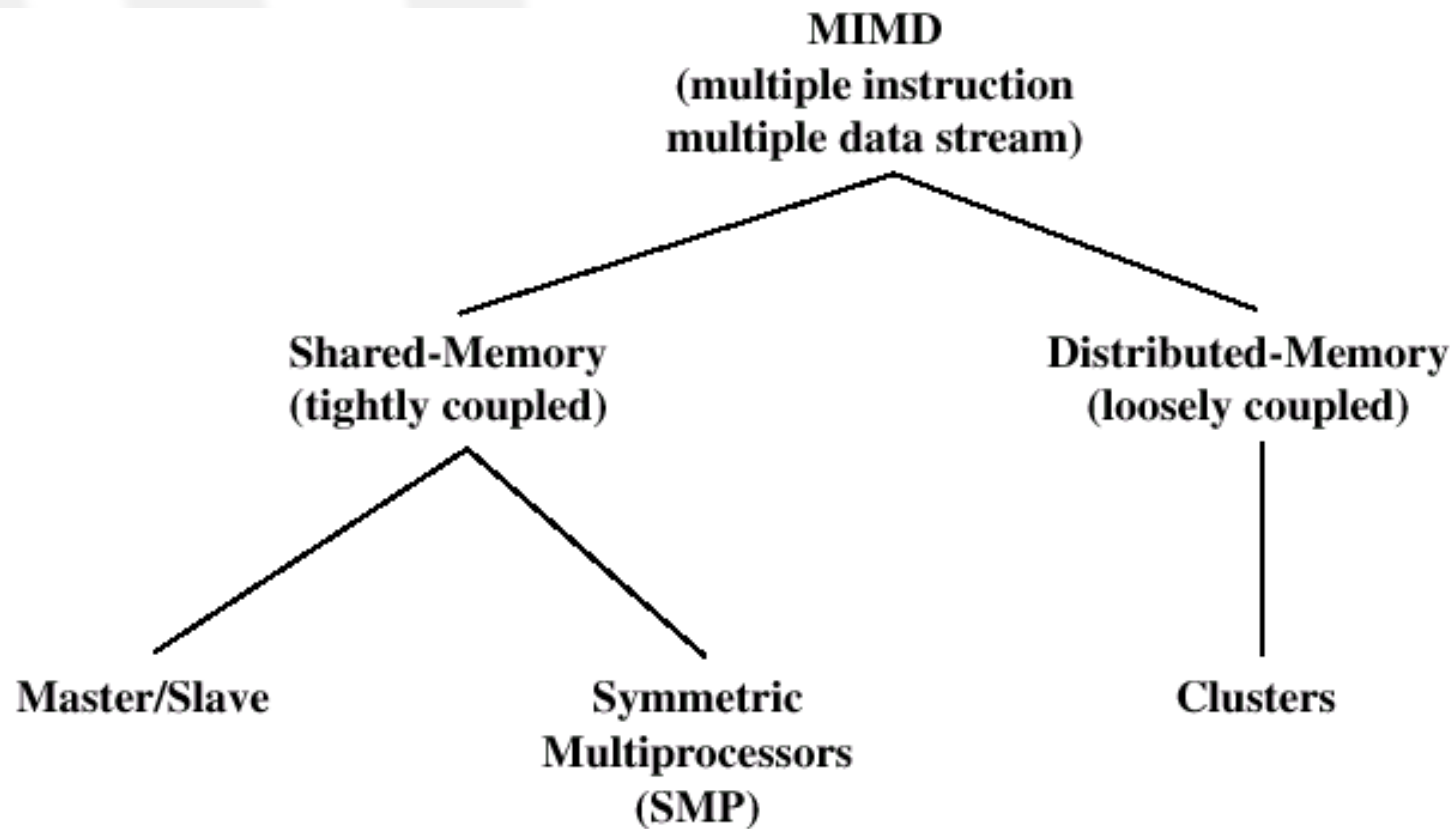


Classificazione dei MIMD

- Sistemi multiprocessore grossi costano e sono difficili da realizzare (max 64 CPU)
- Sistemi “multicomputer” (cluster)
 - Ogni macchina ha la propria memoria
 - Facili da costruire (fino a 16000 CPU)
 - A bus (host collegati su LAN)
 - A commutazione
 - Griglie
 - Ipercubi
 - Es.: <http://www.cs.vu.nl/~bal/das.html>



Tassonomia di Stallings



Tassonomia di Stallings

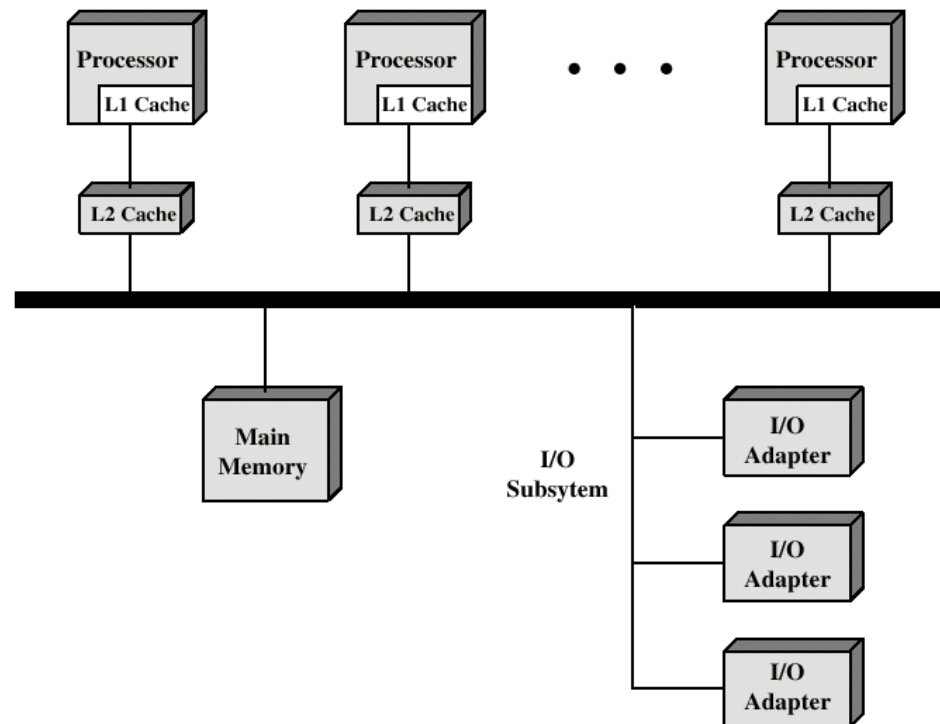
- Master/Slave
 - Asimmetrico
 - Un processore (Master) esegue il S.O.
 - Schedula i processori slave
 - Effettua tutti i servizi del kernel
 - Ha pieno controllo di tutte le risorse
 - Gli altri processori (Slave) eseguono programmi utente ed eventuali utility
 - Master può essere un collo di bottiglia e un punto critico di guasto

Sistemi multiprocessore

- Symmetric Multiprocessor (SMP)
 - Simmetrico
 - Tutti i processori sono uguali
 - Architettura a bus
 - S.O. + difficile da progettare rispetto ai sistemi master/slave

SMP

- Importante la modularità!



SMP

- Il kernel può eseguire su qualsiasi processore
 - Tipicamente in parallelo, thread multiple su processori diversi
 - Ogni processore è auto-schedulante
- Problemi chiave nel progetto di S.O. per SMP
 - Conflitti di scheduling tra processori
 - Sincronizzazione accessi a memoria condivisa
 - Gestione dell'esecuzione concorrente di servizi del S.O.
 - Fornire affidabilità e tolleranza ai guasti

Cluster

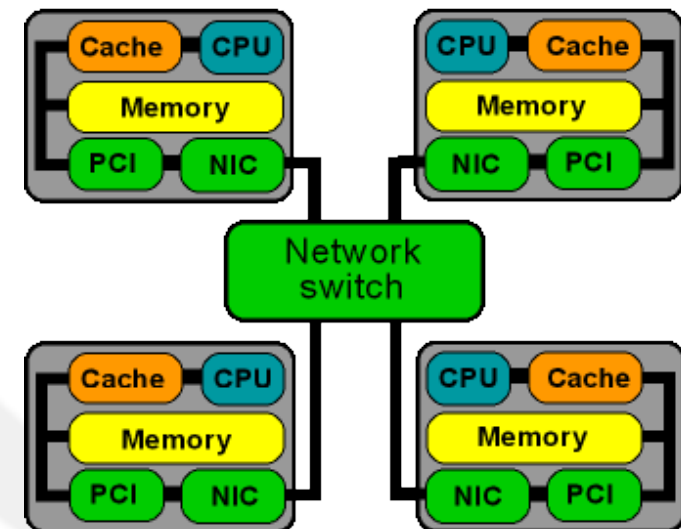
- Generica architettura multicomputer
 - Alternativa a SMP
 - Calcolatori “poco” eterogenei
 - No memoria condivisa
 - Appaiono come un unico calcolatore
- In generale diverso da un insieme di calcolatori su una rete locale
 - Molti processori
 - Connessione dedicata (tipo LAN)

Cluster

- Es: C-Plant Phase III
 - 1536 nodi (466 MHz Compaq DS10L)
 - 100MB/s
 - Linux



4 node PC/workstation cluster



Cluster vs. SMP

- SMP
 - più facili da gestire e configurare
 - occupa meno spazio e consuma meno
 - sono ormai stabili
- Cluster
 - sono superiori rispetto alla scalabilità
 - permettono una maggior ridondanza

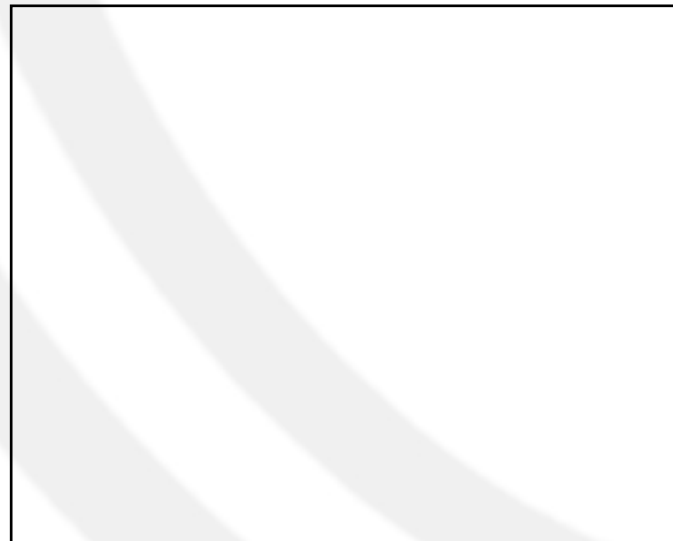
CLASSIFICAZIONE SW

Classificazione SW

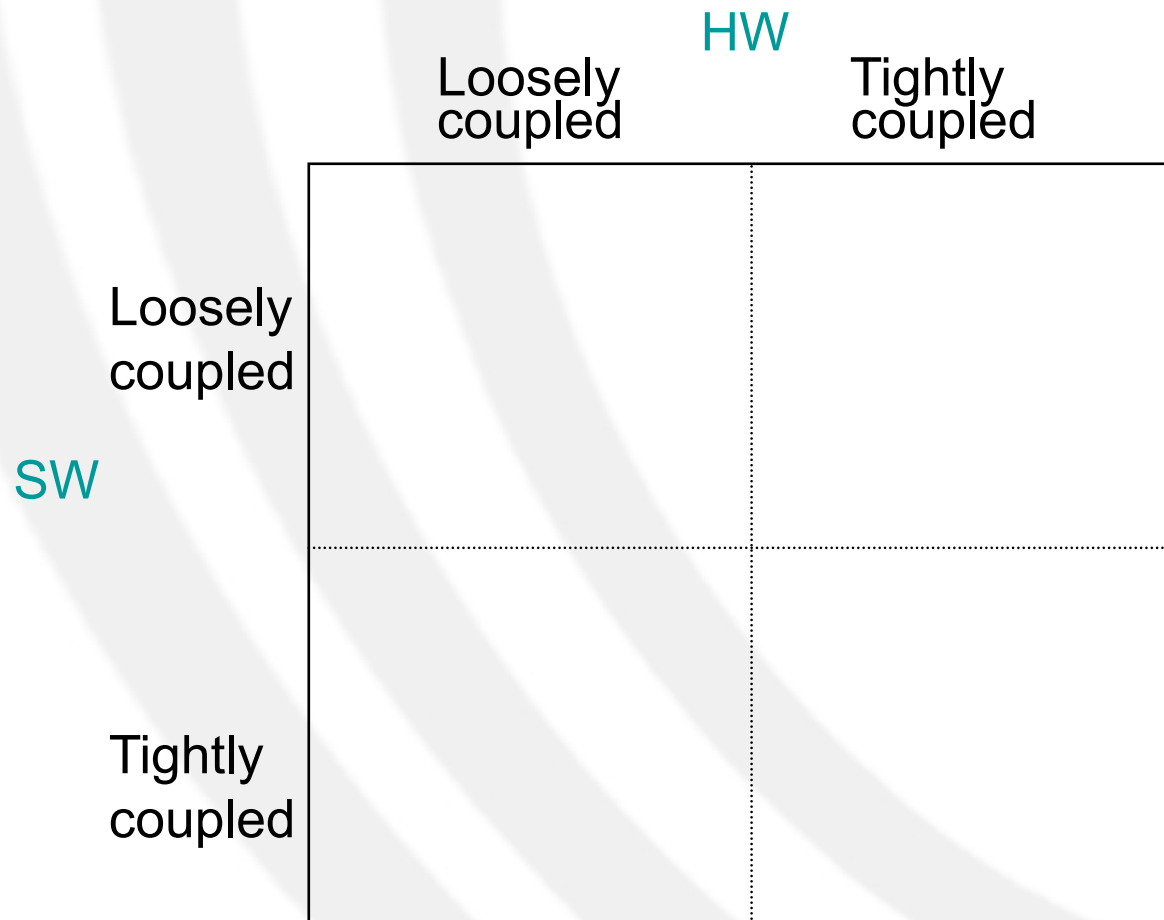
- HW vs. SW definisce il tipo di S.O.

HW

SW



Classificazione SW



Classificazione SW

		HW	
		Loosely coupled	Tightly coupled
SW	Loosely coupled	Network O.S. (NOS)	—
	Tightly coupled	Distributed O.S. (DOS)	Multiprocessor O.S. (MOS)

S.O. multiprocessore (MOS)

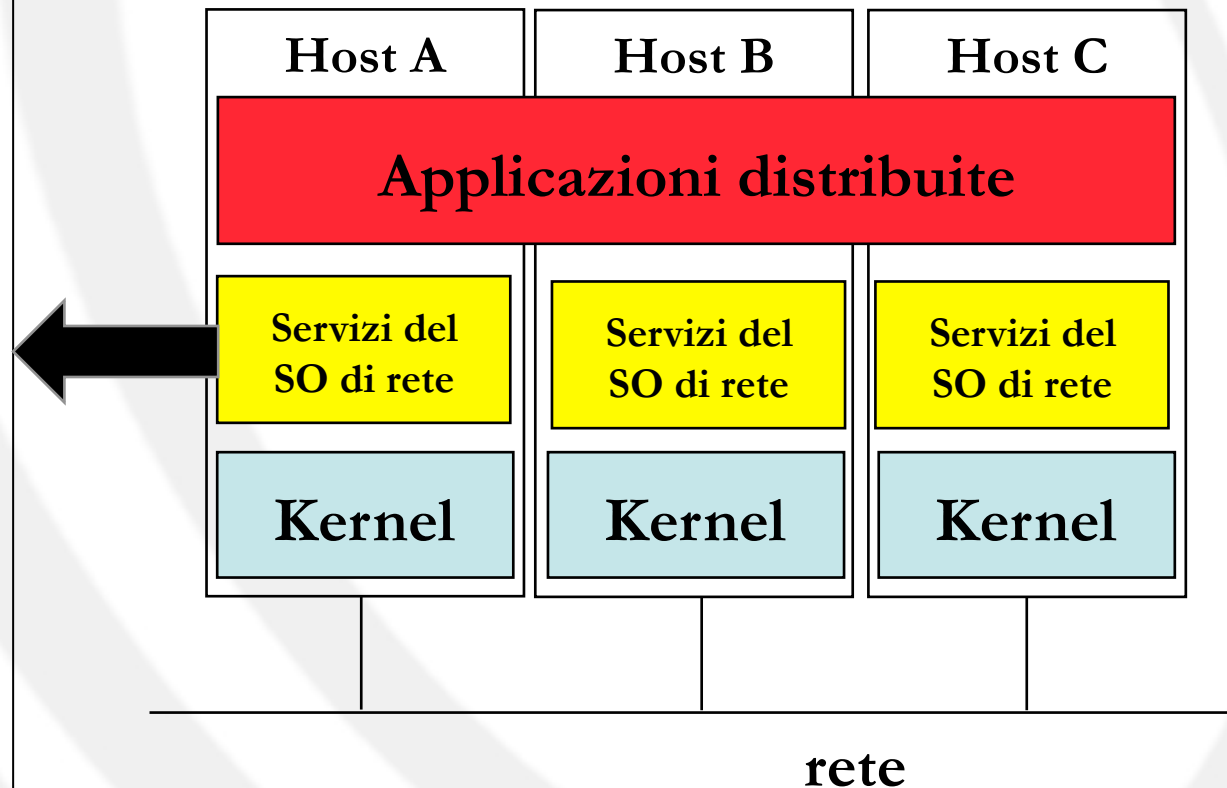
- Un processo può essere eseguito su qualunque CPU
- Tutte le CPU eseguono lo stesso S.O.
- File system simile al caso uniprocessore
- Memoria condivisa
 - Problematica → consistenza memoria/cache

S.O. di rete (NOS)

- Ogni macchina esegue il suo S.O.
- La molteplicità delle macchine è visibile agli utenti
- L'accesso alle risorse avviene in modo esplicito:
 - Login remoto
 - Trasferimento dati (es. Via FTP)
- “Esempio” di NOS:
 - NFS (network file system)
 - Usato per condivisione di file system
 - Non propriamente un S.O.!

Architettura NOS

- Necessario usare comandi ad hoc per usare le risorse distribuite in rete
- Manca trasparenza

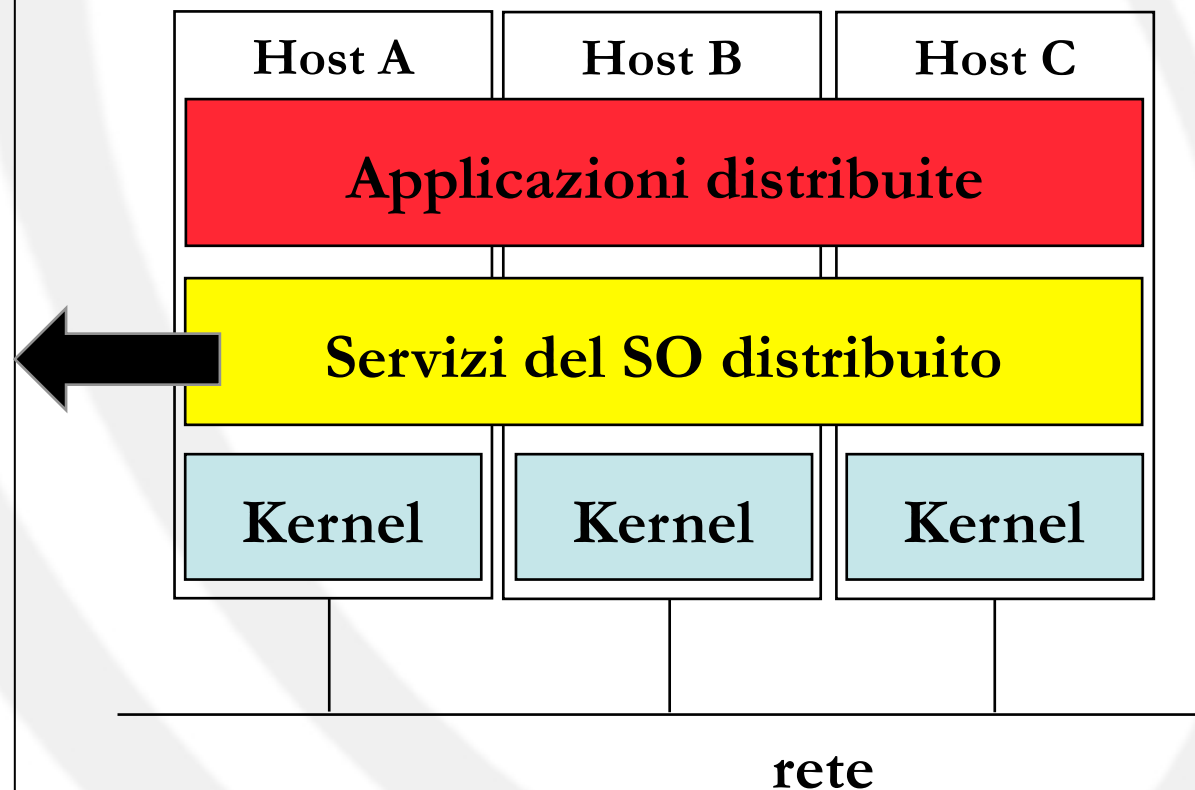


S.O. distribuiti (DOS)

- La molteplicità delle macchine NON è visibile agli utenti
- L'accesso alle risorse remote simile a quelle locali
- Macchine sono in qualche modo, ma non completamente, autonome
- Tutte le macchine eseguono lo stesso S.O.

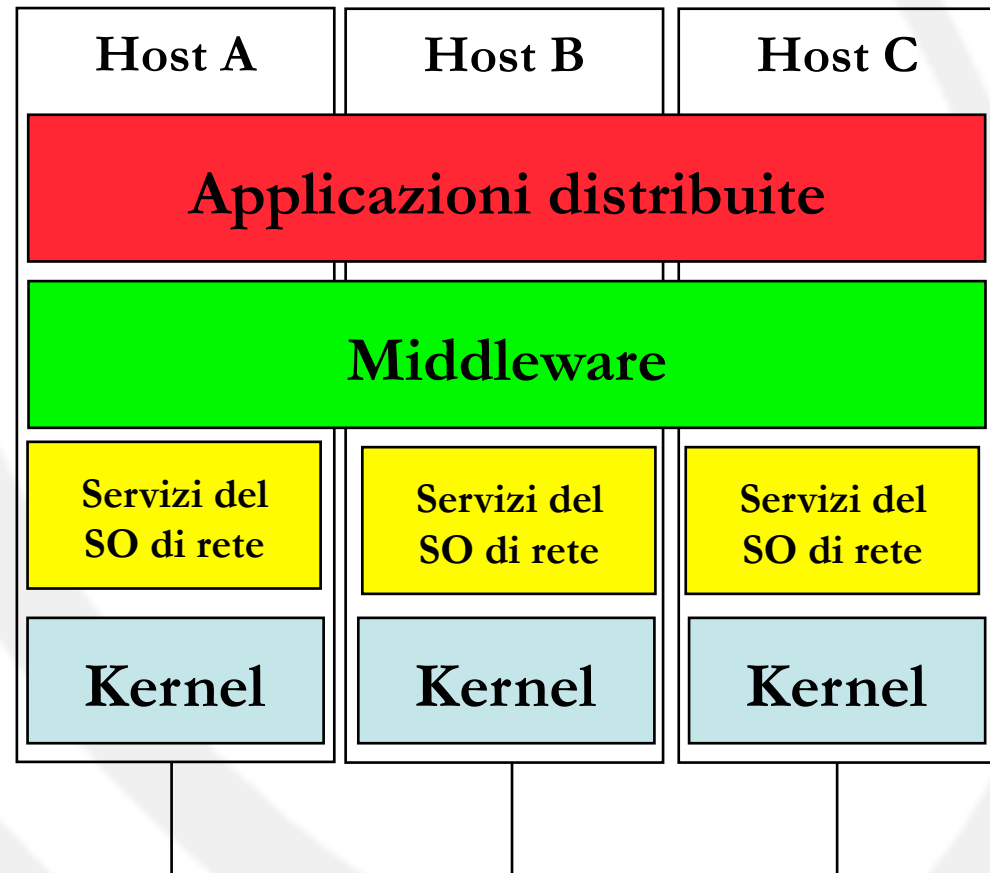
Architettura DOS

- Memoria condivisa via SW
- Assegnamento task ai processori
- Interprocess communication
- Mascheratura di guasti HW
- ...



Middleware

- DOS → trasparenza ma poca scalabilità, no eterogeneità
- NOS → non trasparenza ma molta scalabilità/eterogeneità
- **Middleware:**
Strato SW che unisce i benefici di DOS e NOS
- Primitive di comunicazione ad alto livello (es.: RPC)
- File system distribuito
- Naming
- ...



rete

Classificazione

	NOS	DOS	Middleware	MOS
Uniprocessore virtuale	No	Si	No	Si
CPU eseguono lo stesso S.O.	No	Si	No	Si
Quante copie di S.O.?	N	N	N	1
Comunicazione	Condivisione file/messaggi	Messaggi	Dipende dal middleware	Memoria condivisa
Unica coda di esecuzione per i processi?	No	No	No	Si
Semantica di condivisione dei file	No, in genere	Si	Si	Si
Grado di trasparenza	Basso	Alto	Alto	Molto alto
Scalabilità	Si	Moderata	Dipende	No
Gestione risorse	Locale	Globale, distribuita	Locale	Globale, centralizzata

Obiettivo

- Analizzare le problematiche che riguardano i sistemi operativi per le varie categorie di sistemi
- Priorità: sistemi distribuiti
 - Presentano le problematiche più interessanti
 - Significativamente diverse da S.O. centralizzati
 - Diffusione più ampia
 - In parte, problematiche comuni a sistemi multicomputer/cluster

Perché usare DOS?

- Prezzo/Prestazioni
 - Una rete di workstation fornisce + MIPS di un grosso mainframe
- Prestazioni più elevate
 - N processori forniscono idealmente N volte il lavoro di un processore
- Condivisione delle risorse
 - Risorse costose non devono essere replicate

Perché usare DOS?

- Scalabilità
 - Una struttura modulare permette di aggiungere/rimpiazzare risorse in modo + semplice
- Affidabilità
 - La ridondanza di risorse permette tolleranza ai guasti