I supercomputer oggi: applicazioni e architettura

Candidato
Pellinacci Marco

Relatore
Prof. M. Avvenuti

Classificare le architetture parallele

Tassonomia di Flynn

	SI *(Single Instruction stream)*	*MI* *(Multiple Instruction stream)*
SD *(Single Data stream)*	Macchine SISD	Macchine MISD
MD *(Multiple Data stream)*	Macchine SIMD	Macchine MIMD

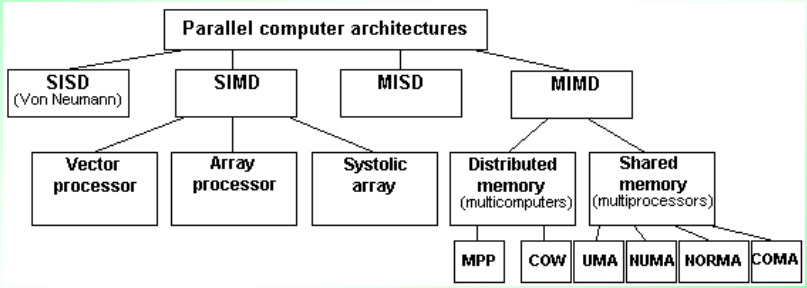
Guarda un sistema di elaborazione da 2 punti di vista:

- In base alla capacità di avere più flussi di istruzioni
- In base alla capacità di avere più flussi di dati

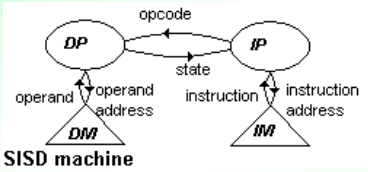
Limiti della tassonomia di Flynn

Classificazione incapace tuttavia di esprimere caratteristiche come la distinzione tra architettura a memoria distribuita e architettura a memoria condivisa

Tassonomia estesa

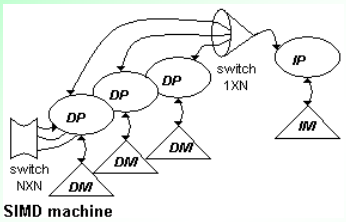


SISD



- Tradizionale architettura sequenziale (o di **Von Neumann**) usata da tutti i calcolatori convenzionali
- Un'unica istruzione è eseguita ad ogni passo temporale

SIMD



SIMD machine

- Macchine spesso dette *number crunching*
- Tante unità di elaborazione eseguono contemporaneamente la stessa istruzione lavorando però su insiemi di dati differenti
- Topologie di interconnessione regolari (torus, ipercubi, ecc.) o create *ad hoc* (in base alla struttura del problema, ecc.)
- Le comunicazioni regolari (cioè che rispettano la topologia fisica) non creano conflitti, sono efficienti e, dunque, poco costose
- Esempi più famosi di macchine SIMD: i **supercomputer vettoriali**, usati per particolari applicazioni (dove soprattutto si lavora su grandi matrici)

SIMD

- Modello di computazione usato: tipo sincrono (solo 1 Unità di Controllo)
- Una computazione è suddivisa in molteplici fasi, in ciascuna delle quali le computazioni possono essere partizionate per esplicitare **parallelismo temporale** o **parallelismo spaziale**

  Parallelismo temporale. Pipeline: parti diverse di un'unica istruzione sono eseguite in parallelo in differenti moduli connessi in cascata

  Parallelismo spaziale. I medesimi passi sono eseguiti contemporaneamente su un array di processori perfettamente uguali, sincronizzati da un solo controllore

SIMD

- Vector processor con caratteristiche pipeline (parallelismo temporale)
- Array processor (parallelismo spaziale)
- Systolic array (parallelismo temporale/spaziale)

SIMD
- Vector processor -

- Elevate prestazioni specialmente con calcoli vettoriali/matriciali
- Parallelismo realizzato all'interno del processore (non visibile a livello programmatore)

Architettura

- Memoria principale
- Unità di Controllo scalare
- Unità di Controllo vettoriale
- Registri scalari
- Registri vettoriali
- Più Unità Funzionali (scalari/vettoriali) connesse in pipeline (e capaci di funzionare in maniera concorrente)

Fattore critico: banda di memoria offerta alle Unità Funzionali (se non è elevata, è impossibile sfruttare pienamente la velocità delle Unità stesse)

SIMD
- Vector processor -

- **Diversità di funzionamento tra un processore scalare ed un processore vettoriale**
  Esempio
  ```
  c = a + b;
  ```
  Processore scalare: gli operandi sono praticamente dei numeri
  Processore vettoriale: gli operandi sono vettori

- **Compilatore vettoriale**
  Esempio
  ```
  int i = 0;
  for ( ; i < 10; i++ )
            c[i] = a[i] + b[i];
  ```
  Riconosce tutti quei cicli sequenziali trasformabili però in un'unica operazione vettoriale

SIMD
- Array processor -

- Elevate prestazioni **solo** nel caso in cui eseguano programmi costituiti per lo più da istruzioni vettoriali

Architettura

- Memoria "programma"
- Array di Elementi di Elaborazione
- Unità di Controllo. Preleva le istruzioni dalla memoria "programma" e distingue tra istruzioni scalari (eseguite direttamente) e istruzioni vettoriali (inviate in parallelo a tutti gli Elementi di Elaborazione dell'Array)

Sincronismo della computazione: l'Unità di Controllo non invia una nuova istruzione finché il processore più lento non ha terminato l'esecuzione della precedente istruzione

SIMD
- Systolic array -

- Usati in particolari ambiti (analisi numerica ed elaborazione dei segnali)

Architettura

- Un insieme di moduli identici:
  1. Ciascuno con la propria memoria locale
  2. Interconnessi da strutture semplici e regolari (alberi, mesh, ecc.) che corrispondono al grafo della computazione (comunicazioni regolari)

"Systolic": i dati si spostano in maniera ritmica (operazioni sincronizzate da un segnale di clock globale ed esterno ai nodi) lungo il percorso circolare *memoria_calcolatore-nodi-memoria_calcolatore* (modalità di funzionamento analoga a quella della circolazione del sangue)

MISD

- Più flussi di istruzioni (processi) lavorano contemporaneamente su un unico flusso di dati
- Categoria praticamente vuota (finora nessuna implementazione)

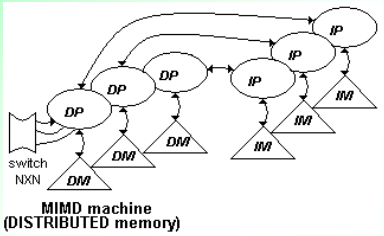
MIMD

- Tutti gli elaboratori con più unità di elaborazione (sia scalari che vettoriali)
- Più processi sono in esecuzione contemporaneamente su più processori ed utilizzano dati propri o condivisi

MIMD

- Modello di computazione in parallelo: tipo asincrono
- I processi eseguiti su un certo nodo fanno riferimento al clock del processore di quel nodo
- Non esistendo un “tempo globale” sono indispensabili tecniche che permettano la comunicazione/sincronizzazione tra i vari processi:
  1. Modello *message passing*
  2. Modello *shared memory*

MIMD a memoria distribuita



- Sostanzialmente tutte le reti di calcolatori
- Ogni coppia *IP-DP* (e relative memorie) costituisce in pratica una macchina SISD

Multicomputers

- Struttura di interconnessione (*switch NxN*): presi 2 nodi, la distanza tra essi è "piccola" e ciascun nodo ha "pochi" link di comunicazione
- Rete di interconnessione regolare e diretta (ipercubi, mesh, torus), attraverso cui i nodi si scambiano informazioni secondo il paradigma *message passing*
- Modello di comunicazione che si discosta dalla topologia dell'architettura **=>** un'operazione di *embedding* (cioè di mappatura del grafo delle comunicazioni in quello della topologia)

- Tra i nodi non c'è memoria condivisa e ogni nodo esegue indipendentemente insiemi multipli di istruzioni usando differenti insiemi di dati, memorizzati su spazi differenti
- Dunque è ottimale usare algoritmi ad elevata località
- Scalabilità elevata

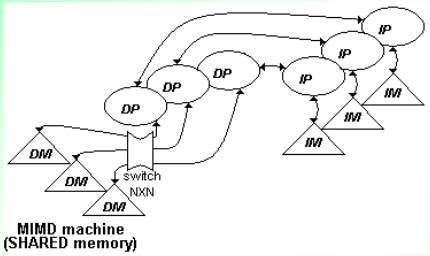
MIMD a memoria distribuita
- MPP (Massively Parallel Processing) -

- Elaborazione MPP: applicazioni scientifiche e particolari contesti di calcolo commerciale-finanziario
- Sistema MPP:
  1. Centinaia di nodi (CPU standard, ognuna con la propria memoria e la propria copia del SO)
  2. Rete di interconnessione custom molto potente (larga banda e bassa latenza)
- Affinché l'elaborazione MPP dia effettivi vantaggi, occorre disporre di software capace di partizionare il lavoro e i dati su cui opera tra i vari processori
- Questi sistemi sembrano un buon punto di partenza per arrivare a macchine parallele *general-purpose* (flessibilità d'uso + prestazioni elevate)

MIMD a memoria distribuita
- COW (Cluster Of Workstations) -

- Insieme di nodi (calcolatori *stand-alone*) che lavorano dando l'impressione che si abbia a che fare con un'unica risorsa di elaborazione
- Connessioni: Gigabit Ethernet, ATM, ecc.
- Caratteristiche:
  1. *High-availability*. Ogni nodo esegue una serie di programmi particolari (*cluster management software*) e può controllare 1 o più nodi: se dovesse presentarsi un malfunzionamento presso un nodo, il nodo che lo controlla può prendere i mezzi di memorizzazione del nodo malfunzionante e riavviare le applicazioni che erano in esecuzione
  2. *Load-balancing*. Le transazioni da eseguire sono indirizzate verso quei nodi che hanno il minor carico

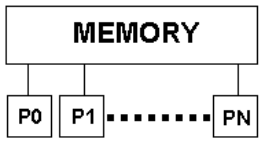
MIMD a memoria condivisa



Multiprocessors

- Comunicazione tra processori effettuata condividendo aree di memoria
- Dunque lo *switch NxN* deve essere molto efficiente
- Poiché il numero ***N*** di processori è "piccolo" (***N***<100), anche un collegamento *tutti-a-tutti* non è troppo costoso
- Scalabilità bassa

MIMD a memoria condivisa
- UMA (Uniform Memory Access) -



UMA machine

- Tutte le CPU hanno il medesimo tempo di accesso a tutta la **memoria** (allocazione dati non critica)
- Difficile aumentare il numero di processori (bassa scalabilità), causa il tempo di accesso alla memoria (il bus può diventare un *bottleneck*)
- Per diminuire il traffico sul bus si può ricorrere all'uso di memorie private e cache (per le varie CPU)

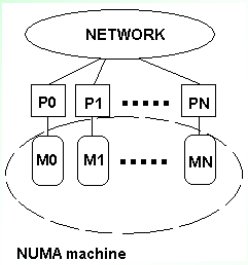
  Attenzione

  Se si opta per le cache, c'è da risolvere il *problema della coerenza di cache*
- *Problema della coerenza di cache*: se un certo blocco ad un dato istante è presente in più cache, si rischia il disallineamento delle copie

  Soluzione

  Le CPU rispettano opportuni protocolli (ad esempio *MESI*), specifici per garantire la coerenza di cache

MIMD a memoria condivisa
- NUMA (NonUniform Memory Access)-



NUMA machine

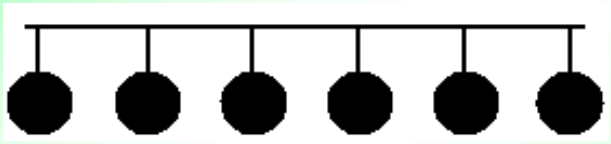
- Per ogni CPU il tempo di accesso alla memoria varia a seconda del modulo acceduto (allocazione dati critica)
- Memoria solo **logicamente condivisa**, ma **fisicamente distribuita**
- E' possibile aggiungere processori con estrema facilità (scalabilità)
- Lo schema NUMA (metà anni '90) cerca di risolvere i problemi del modello UMA (*memory bottleneck*)
- Vantaggio anche dal punto di vista della programmazione: poiché i processori hanno in comune un unico spazio di indirizzamento, per le applicazioni utente il sistema appare dotato di un'unica (e omogenea) area di memoria

Confronto tra SIMD e MIMD

- Le SIMD richiedono meno hw delle MIMD (un'unica Unità di Controllo)
- Le MIMD usano spesso processori *general-purpose*, dunque sono meno costose delle "classiche" SIMD
- Le SIMD usano meno memoria delle MIMD (una sola copia del programma)
- Le MIMD godono di una grande flessibilità in termini di modelli computazionali supportati

Topologie di interconnessione

Bus

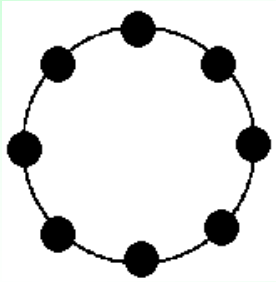


- Configurazione semplice e affidabile
- Grado: 1 (per tutti i nodi)
- Diametro: 1
- # totale di link: 1
- Competizione massima sull'accesso al mezzo

Linear array

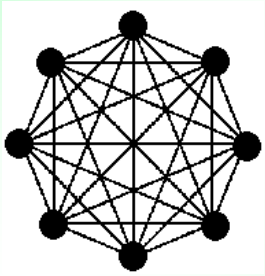
- Grado: per il "primo" e l' "ultimo" nodo è 1, mentre per i restanti nodi è 2
- Diametro: N-1
- # totale di link: N-1
- Competizione ridotta al minimo
- # comunicazioni in contemporanea (caso ideale): N/2
- Nodi capaci di offrire servizi di routing
- Tolleranza ai guasti pessima

Ring



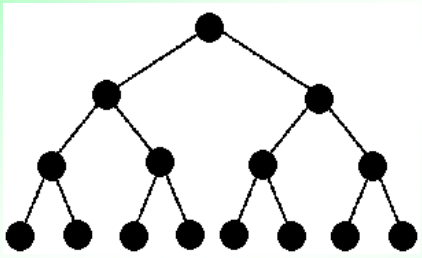
- Grado: 2 (per tutti i nodi)
- Diametro: $\lfloor N/2 \rfloor$
- # totale di link: N
- Tolleranza ai guasti minima

Connessione completa (*tutti-a-tutti*)



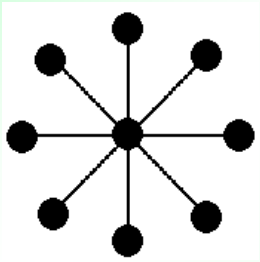
- Grado: N-1 (per tutti i nodi)
- Diametro: 1
- # totale di link: N*(N-1)/2 (il collegamento punto-punto non è scalabile)

B-Tree



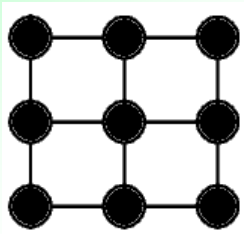
- Altezza albero (h): $h = \lceil \log_2 N \rceil$
- Grado: per la radice è 2; per le foglie è 1; per gli altri nodi è 3
- Diametro: 2*(h-1)
- # totale di link: N-1
- Rami alti congestionati (topologia non scalabile). Soluzione possibile: topologia a *fat-tree*
- Radice: potenziale “punto debole”

Star



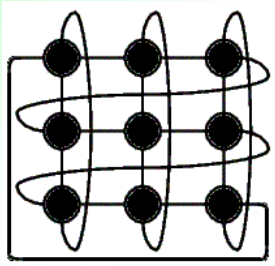
- Grado: per il nodo centrale è N-1, mentre per gli altri nodi è 1
- Diametro: 2
- # totale di link: N-1
- Tolleranza ai guasti fortemente dipendente dalla “robustezza” del nodo centrale

Mesh (2-D)



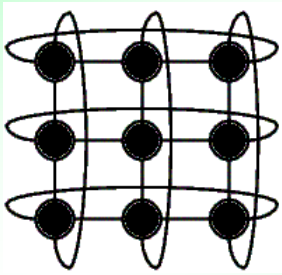
- r: radice quadrata di N
- Grado: per i nodi ai vertici è 2; per i nodi “centrali” ai lati è 3; per i restanti nodi è 4
- Diametro: 2*(r-1)
- # totale di link: 2*N-2*r
- Resistenza ai guasti buona

Illiac mesh



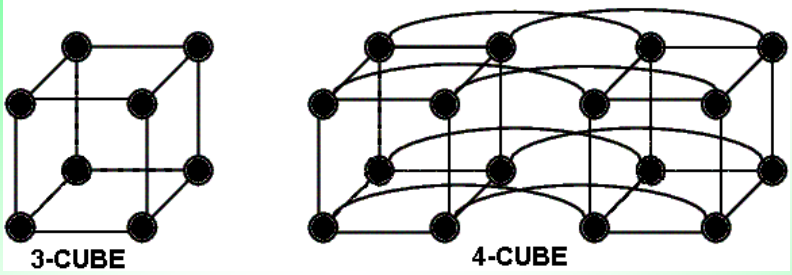
- Grado: 4 (per tutti i nodi)
- Diametro: r-1
- # totale di link: 2*N

Torus (2-D)



- Grado: 4 (per tutti i nodi)
- Diametro: $2*\lfloor r/2 \rfloor$
- # totale di link: 2*N
- Topologia ben scalabile e notevolmente resistente ai guasti

Ipercubo (d-CUBE)



- Dimensione ipercubo: d
- N: 2^d
- Grado: d (per tutti i nodi)
- Diametro: $\log_2 N = d$
- # totale di link: d*N/2
- Topologia scalabile solo con un numero di nodi potenza di 2
- Numerazione nodi: Codice Binario di Gray

Metriche di prestazione

Speed-up ed Efficienza

Speed-up (S)

$S = T_1/T_N$

- Fattore di velocità che si guadagna rispetto ad un *uniprocessore*
- Ideale: *speed-up* lineare con il numero di processori (N) usati nella macchina parallela
- Realtà: $S < N$
- Il valore dello *speed-up* dipende dalle applicazioni, ma anche dall'architettura: nelle SIMD spesso $S \approx N$, mentre nelle MIMD è difficile far crescere S (non è facile far lavorare pienamente tutte le CPU)

Efficienza (E)

$E = S/N$

- Misura direttamente collegata allo *speed-up*
- Ideale: $E = 1$
- Realtà: $E < 1$

Tempo sequenziale

- Tempo sequenziale (T_{seq}): tempo impiegato per eseguire istruzioni non parallelizzabili (operazioni di I/O, costrutti condizionali, algoritmi intrinsecamente sequenziali, ecc.)
- **Legge di Amdahl**: un parallelismo "perfetto" (nelle varie attività compiute da un calcolatore) non è **mai** raggiungibile poiché saranno **sempre** presenti sequenze di sw intrinsecamente seriale
- La **legge di Amdahl** ridefinisce lo *speed-up*:

$$S = T_1 / \{T_{seq} + [(T_1 - T_{seq})/N]\}$$

- Viene perciò posto un limite superiore per S: anche se $N \rightarrow \infty$, avremmo:

$$S = T_1 / T_{seq}$$

Esempio (algoritmo non parallelizzabile)

$f_{n+2} = f_{n+1} + f_n$ con $f_0 = f_1 = 1$ ed n = 0, 1, 2, …

Tempo impiegato per comunicare

- Altro aspetto da tenere in considerazione ai fini prestazionali, è il rapporto tra elaborazione ("pura") e comunicazione
- Ideale: 2 flussi tra loro indipendenti
- Sarebbe poi importante che il tempo impiegato complessivamente per comunicare (T_{com}) fosse irrilevante rispetto al tempo di elaborazione (T_{elab})
- T_{com} dipende da vari fattori:

$$T_{com} = (T_{su} + L_c * T_b) * N_c$$

- Questi ultimi aspetti rilevati sono particolarmente importanti quando si considera un multicomputer

Multitasking

- Di notevole importanza pure nelle macchine parallele per mantenere lo sfruttamento delle varie CPU altissimo
- Deve rispettare il seguente vincolo:

$$P >> N$$

Perché architetture parallele?

Premessa

Domanda

Perché investire nei supercomputer o, più in generale, nelle macchine parallele?

Cerchiamo di capirlo illustrando l'effettiva strategicità del *Calcolo ad Alte Prestazioni* (**HPC**, *High Performance Computing*)

Il settore del Calcolo ad Alte Prestazioni

- Caratterizzato da più discipline, afferenti nel loro complesso alle Scienze e Tecnologie dell'Informazione, e relative allo studio/realizzazione di sistemi di elaborazione hw/sw capaci di prestazioni che vanno dalle centinaia di GigaFlops ai TeraFlops

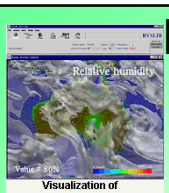
  Ricorda

  1 GigaFlops = 10^9 operazioni su numeri reali al secondo

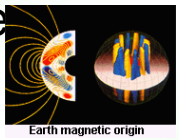
  1 TeraFlops = 10^3 GigaFlops

  1 PetaFlops = 10^3 TeraFlops
- **Scopo**: affrontare e risolvere le **Grand Challenges** (modelli climatici, turbolenza dei fluidi, modellazione superconduttori, visione e comprensione, ecc.)

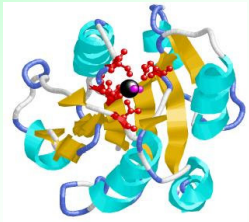
Il settore del Calcolo ad Alte Prestazioni
- Area Scientifica -

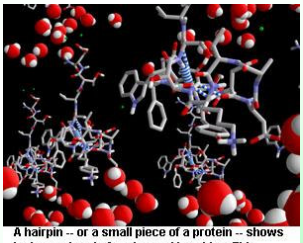


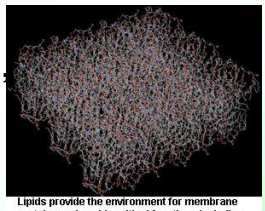
Visualization of meteorological simulation results

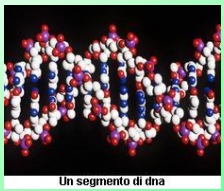

Earth magnetic origin

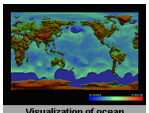
- Astrofisica
- Geoscienze e Geofisica
- Biologia, Chimica Farmaceutica, Biotecnologie e Medicina


Rappresentazione del sito attivo dell'Integrina umana β3, la proteina coinvolta nelle ischemie.

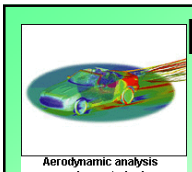

A hairpin -- or a small piece of a protein -- shows hydrogen bonds forming and breaking. This amounts to a microcosm of the folding process. The shape a protein folds into is determined by a balance of many different interactions, including hydrogen bonds and hydrophobic effects. Misfolding can lead to a variety of diseases, including mad cow disease and cystic fibrosis

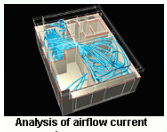

Lipids provide the environment for membrane proteins and enable critical functions including cell signalling and cell division. Studying lipids is crucial to understanding diseases related to these proteins, including muscular dystrophy and Alzheimer's. One third of all proteins in the human body -- and half of all drug targets -- are membrane proteins


Un segmento di dna

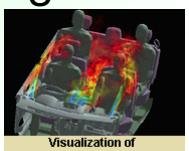

Visualization of ocean circulation flows

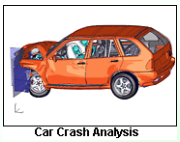
Il settore del Calcolo ad Alte Prestazioni
- Area Ingegneristica -

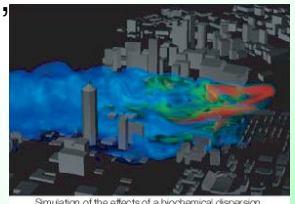

Aerodynamic analysis around an auto-body

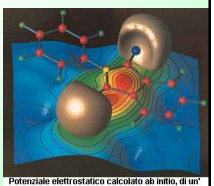

Analysis of airflow current in a room

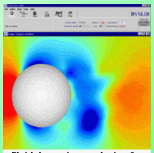
- Industria (Automobilistica, Aerospaziale, Energetica, Militare, Meccanica, Chimica, ecc.)
- Telecomunicazioni
- Automazione
- Server ad alte prestazioni

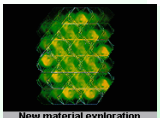

Visualization of air-conditioning flow inside a car


Car Crash Analysis


Simulation of the effects of a biochemical dispersion

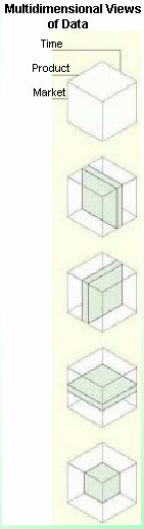

Potenziale elettrostatico calcolato ab initio, di un' immide
- Chimica Computazionale -


Fluid dynamics analysis of a golf ball motion


New material exploration by pseudo-atom-database as building blocks

Il settore del Calcolo ad Alte Prestazioni
- Area Economico-Finanziaria -

- *DSS, Decision-Support Systems*
- Grandi volumi di dati



Il settore del Calcolo ad Alte Prestazioni

Lo *HPC*

è indispensabile dunque quando siamo davanti a

problemi computational intensive e/o data intensive

Un po' di storia

Introduzione

- Anni '60: ci si rende conto che, per classi di problemi caratterizzate da grandi volumi di dati su cui eseguire ripetutamente le medesime operazioni, l'architettura di Von Neumann è totalmente inadatta (limiti prestazionali)
- Si inizia così a parlare di *supercomputer*

Supercomputer

- Uso di tecnologie circuitali all'avanguardia (i circuiti microelettronici devono essere "rapidi")
- Uso di molteplici Unità di Elaborazione, per passare ad un'esecuzione parallela delle istruzioni
- Uso di appositi algoritmi

Anni ’60: Pipeline e Legge di Moore

- **CDC** (*Control Data Corporation*): realizza il *CDC 6600* (predecessore del più potente *CDC 7600*)
- **IBM**: realizza il *360/91* (predecessore del più potente *360/370 195*)
- *CDC 6600* e *360/91* hanno un’architettura innovativa: si introduce il concetto di pipeline (viene così esteso il principio di sovrapposizione (*overlap*) delle istruzioni)

Prima legge di Moore (1965)

Il numero di transistor su di un chip raddoppia ogni 18 mesi

Conseguenze

  - Aumenta la capacità dei chip di memoria
  - Aumenta la capacità delle CPU
- A fine anni ’60 la **prima legge di Moore** già “si fa sentire”: i chip sono sempre più complessi:
  1. Memorie (istruzioni e dati) di transito ad alta velocità (*I-cache* e *D-cache*)
  2. Registri interni di varia tipologia
  3. Molteplici Unità di *Pre-Fetch* e *Fetch* (*Prefetch Unit* e *Fetch Unit*), di *Decode* (*Instruction Unit*) e di *Execute* (*ALU*)

Anni ’70: Supercomputer Vettoriali

- **Amdahl Corporation** (di Gene Amdahl): realizza evolutissimi mainframe
- **Cray Research Inc.** (di Seymour Cray): inaugura l’inizio del vero e proprio *supercomputing* con l’installazione, presso il *Los Alamos Scientific Laboratory*, della prima macchina vettoriale, cioè *Cray 1* (1976)

  Da allora c’è la possibilità di usare istruzioni vettoriali, le quali portano (fin da subito) a prestazioni elevate per 2 motivi principali:
  1. La *Instruction Unit* ha “poco lavoro” (il processore necessita di decodificare meno istruzioni) e la *bandwidth* di memoria è (conseguentemente) ridotta
  2. Le *ALU* vettoriali sono alimentate con un flusso costante di dati, visto che le coppie di operandi coinvolte in un’istruzione vettoriale sono posizionate in memoria in maniera regolare

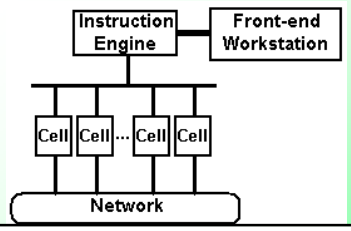
Anni ’80-’90: Sistemi Paralleli a Memoria Distribuita

- Seconda metà degli anni ’80: i sistemi *MPP* riscuotono un gran successo
- Irrompe sul mercato la **TMC** (*Thinking Machine Corporation*), in particolare con la *CM-1* (*Connection Machine-1*) prima, e con la *CM-2* (*Connection Machine-2*) poi
- Tali macchine confermano come i primi sistemi a parallelismo massiccio realizzati siano piuttosto eterogenei tra loro
- Esse hanno infatti una architettura che di fatto è una evoluzione del tipico modello di un array processor: ogni Elemento di Elaborazione dell’Array è ora una *cella*. Le varie celle sono interconnesse in un ipercubo
- Cella: (semplice) processore + memoria locale
- Concetto di “cella”: cade la caratteristica suddivisione tra processore e memoria

Anni ’80-’90: Sistemi Paralleli a Memoria Distribuita

Architettura della *CM-2*

- Unica Unità di Controllo (*Instruction Engine*) che invia in broadcast i microcomandi a tutte le celle
- Fino a 64K celle
- Cella: processore *bit-serial* (dunque semplice e compatto) dotato di *ALU* con 3 ingressi a singolo bit + memoria privata
- 2048 *Floating-Point Unit*
- Interconnessione ad ipercubo



Anni ’80-’90: Sistemi Paralleli a Memoria Distribuita

- Anni ’90: ci si orienta verso sistemi *Cluster*
- La distinzione tra supercomputer e sistemi convenzionali ormai non è più netta come in passato

What is a Supercomputer? A supercomputer is defined simply as the most powerful class of computers at any point in time.

(tratto da: www.cray.com/industry/)

Situazione attuale

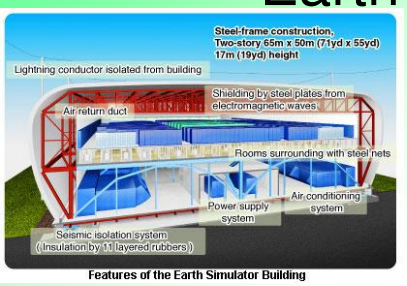
Earth Simulator

- Produttore: *NEC*
- Processore vettoriale (*AP*, *Arithmetic Processor*): NEC 500 MHz (discendente di *NEC SX-5*) da 8 Gflops. Chip *LSI*
- SO: SUPER-UX 64-bit (versione avanzata dello UNIX-based OS di *NEC*)
- Attuali usi

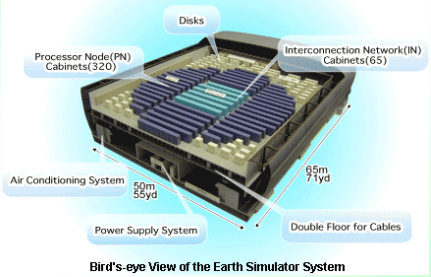
  Earth Simulator Center

  1. **Primo** nella *TOP500 List* di giugno 2004 con 35.86 Tflops (maggio 2002)
  2. Sistema di calcolo parallelo a memoria distribuita costituito da 640 *Processor Nodes*
  3. *PN* (*Processor Node*): 8 *AP* con memoria condivisa
  4. Prestazione di picco: 40 Tflops
  5. Memoria principale totale: 10 TB
  6. Uso:
     - Predizioni di variazioni atmosferiche, oceaniche e terrestri
     - Produzione di *dati rilevanti* al fine di proteggere l'uomo da disastri naturali
     - Promozione di simulazioni innovative in qualsiasi campo (industriale, energetico, delle bioscienze, ecc.)

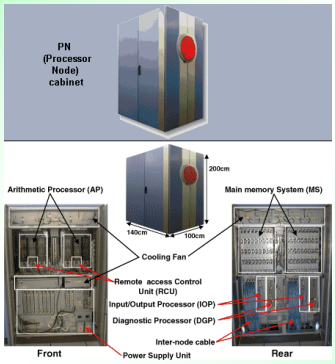
Earth Simulator

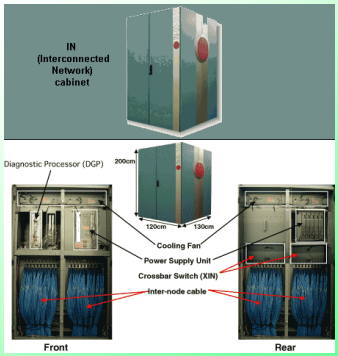


Features of the Earth Simulator Building



Bird's-eye View of the Earth Simulator System





Server high-end IBM eServer pSeries 690

- Produttore: *IBM*
- Processore: *64-bit POWER4+*, primo "server su un chip" (progetto di "SMP-on-a-chip", "Symmetric MultiProcessing-on-a-chip"). Chip *POWER4+*
- Architettura *MCM*, *MultiChip Module*: fino a 8 microprocessori su un singolo modulo *MCM* (distanza fisica tra i componenti ridotta **=>** spostamento più rapido delle informazioni)
- # processori per sistema: 8, 16, 24 oppure 32. Frequenze di clock: 1.5 GHz, 1.7 GHz o 1.9 GHz
- Tecniche avanzate di clustering, funzioni *RAS* (*Reliability*, *Availability and Serviceability*) e funzione *dynamic LPAR* (*Logical PARtitioning*)
- SO: AIX 5L (avanzato SO UNIX (aperto e scalabile) sviluppato da *IBM*)
- Attuali usi

  Research Centre Juelich
  1. **Ventunesimo** nella *TOP500 List* di giugno 2004 (5.5 Tflops)
  2. 41 *eServer p690*, ciascuno configurato a 32-vie con processori *POWER4+* da 1.7 GHz
  3. Prestazione di picco: 8.9 Tflops
  4. Uso: ricerche sulla materia, l'energia, l'*IT*, le *life sciences* e l'ambiente

  European Centre for Medium-range Weather Forecasts
  1. **Sesto** nella *TOP500 List* di giugno 2004 (8.9 Tflops)
  2. 68 nodi *p690+* (66 *compute nodes*, 1 *I/O node* ed 1 *networking node*), ciascuno con 32 processori *POWER4+* da 1.9 GHz
  3. Prestazione di picco: 16 Tflops
  4. Uso: meteorologia

  HPCx
  1. **Diciottesimo** nella *TOP500 List* di giugno 2004 (6.1 Tflops)
  2. 50 nodi *IBM POWER4+ Regatta*, ognuno con 32 processori *POWER4+* da 1.7 GHz
  3. Prestazione di picco: 10.8 Tflops
  4. Uso: fisica molecolare e atomica, biochimica, chimica computazionale, scienze ambientali

Server high-end IBM eServer pSeries 690

Forschungszentrum Jülich
View of the new supercomputer in the specially constructed machine room

The two IBM Cluster 1600 systems installed in ECMWF's Computer Hall

The HPCx platform

pSeries 690 high-end server with optional expansion frame

Cray X1

- Produttore: *Cray Inc.*
- Approccio misto *VP-MPP* (*Vector Processing-Massively Parallel Processing*)
- Processore: *MSP*, *MultiStreaming Processor* (picco prestazionale: 12.8 Gflops). Ha al suo interno 4 *SSP* (*Single-Streaming Processor*)
- Moderno set di istruzioni vettoriali
- Ogni nodo ha 4 *MSP* e memoria locale condivisa (indirizzabile globalmente)
- I singoli nodi supportano l'accesso *UMA* (*Uniform Memory Access*)
- *Cray X1* (nel complesso) è un sistema *NUMA* (*NonUniform Memory Access*)
- Interconnessioni a bassa latenza ed elevata ampiezza di banda per quanto concerne la memoria
- Configurazione massima (1024 nodi): 50 Tflops
- SO: UNICOS/mp. Il kernel è basato su quello del SO IRIX 6.5 (versione del SO UNIX System V prodotto *Silicon Graphisc, Inc.*)
- Attuali usi

  Center for Computational Sciences (presso l'Oak Ridge National Laboratory)

  1. **Ventesimo** nella *TOP500 List* di giugno 2004 (5.8 Tflops con 504 *MSP*)
  2. 512 *MSP*, ovvero 128 nodi
  3. Picco di prestazione (504 *MSP*): 6.4 Tflops
  4. Uso: climatologia, fusione, biologia

Cray X1

Air-cooled Cray X1 System

Liquid-cooled Cray X1 Systems

In sintesi

	Centro di SuperCalcolo				
	Earth Simulator Center	*ECMWF*	*HPCx*	*ORNL/CCS*	*Research Centre Juelich*
Paese	Giappone	Gran Bretagna	Gran Bretagna	Stati Uniti	Germania
Computer	Earth Simulator	eServer pSeries 690 (Power4+ 1.9 GHz)	eServer pSeries 690 (Power4+ 1.7 GHz)	Cray X1	eServer pSeries 690 (Power4+ 1.7 GHz)
N° Processori	5120	2112	1600	504	1312
Produttore	NEC	IBM	IBM	Cray Inc.	IBM
Prestazione di Picco (Gflops)	40960	16051	10880	6451	8921
Posizione nella TOP500 List (giugno 2004)	1	6	18	20	21
Prestazione con cui entra nella TOP500 List (Gflops)	35860	8955	6188	5895	5568

Futuro immediato

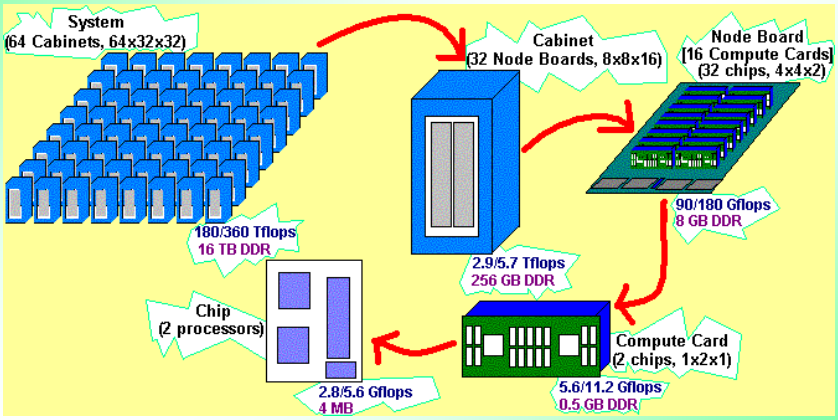
BlueGene

- Progetto *IBM* di supercomputing (fine 1999)
- Supercomputer che si distinguano per i loro ottimi risultati in termini di:
  1. Ampiezza di banda
  2. Scalabilità
  3. Capacità di gestire grandi volumi di dati a bassi costi
  4. Dimensioni fisiche drasticamente ridotte rispetto ad una qualsiasi macchina di potenza equiparabile
- Campo di applicazione primario: genetica avanzata
- *BlueGene/L*: primo sistema della famiglia *BlueGene*

BlueGene/L

- Produttore: *IBM*
- **Ultracomputer** scalabile
- Sistema *Massively Parallel* di (fino a) 65536 nodi computazionali (configurati come un torus 3-D 64x32x32)
- Nodo computazionale:
  1. Un'unica unità di calcolo *ASIC* (*Application-Specific Integrated Circuit*), basata sulla tecnologia *system-on-a-chip* di *IBM*. In particolare vi troviamo 2 *processing cores* di tipo *PowerPC 440*, ciascuno con una "doppia" unità Floating-Point a 64-bit (*PowerPC 440 FP2 core*). Frequenza di lavoro prevista: 700MHz (i 2 processori di un nodo offrono una potenza totale di 5.6 Gflops)
  2. Vari chip di memoria *SDRAM-DDR* (ogni nodo può supportare fino a 2 GB di memoria locale)
- Usando entrambi i processori di un nodo, il sistema raggiunge i 360 Tflops
- SO di controllo (*I/O nodes*): Linux
- SO custom molto essenziale (*compute nodes*): *BlueGene/L compute node kernel*
- La macchina *BlueGene/L* completa (64 rack) è in costruzione per il *Lawrence Livermore National Laboratory* e dovrebbe essere disponibile nel 2005 (o a fine 2004)

BlueGene/L



Considerazioni finali

"Rinascita" del supercomputing - Internet ed e-business on-demand -

- Server ad alte prestazioni: divenuto un campo di *HPC* in conseguenza dell' "esplosione" di Internet
- Sistemi server: caratterizzati per la loro abbondanza di risorse da gestire e devono (tipicamente) fornire diversi servizi (Web server, Mail server, ecc.) a (sempre più) client
- Oggi molte aziende cercano il successo nel mondo dell'*e-business*
- Occorre dunque una reale capacità di reazione efficace, efficiente e tempestiva (*on-demand*)
- Quindi tali aziende richiedono:
  1. Scalabilità eccellente
  2. Potenza (prestazioni) di calcolo notevoli
  3. Disponibilità ininterrotta di applicazioni e dati (affidabilità)
- Seguendo la logica dell'*on-demand*, *IBM* sta insistendo sul *supercomputing on-demand*

  Esempio

  Deep Computing Capacity on-Demand Center (Montpellier): permetterà ai vari clienti di disporre di una notevole capacità di calcolo senza dover acquistare alcuna infrastruttura di calcolo. Il concetto su cui si basa il servizio è quello dell' "utility computing"

“Rinascita” del supercomputing - Human Genome Project -

- Agli inizi del 2003 (con 2 anni di anticipo) il gruppo di scienziati internazionali impegnati nel difficile compito di completare la mappa dei geni alla base della vita umana, ha concluso i lavori: tale accelerazione è stata possibile grazie ai continui sviluppi delle tecnologie usate nell’ambito dei supercomputer
- Grande interesse dei produttori di macchine *HPC* per il campo delle bioscienze

  Esempio

  Stretta collaborazione tra il Centro di Calcolo ad Alte Prestazioni presso l’Università Tecnica di Dresda e *NEC* al fine di testare/fornire software bioinformatici che permettano di descrivere processi biologici su supercomputer vettoriali

Ricorda

E’ fondamentale che, a passi in avanti fatti dallo hw, corrispondano altrettanti passi in avanti fatti dal sw

“Rinascita” del supercomputing - Leadership -

- Settore dei supercomputer indubbiamente associato alla tecnologia americana
- Negli ultimi anni dello scorso secolo però gli Stati Uniti non si sono impegnati più di tanto nell’evoluzione dello *HPC* (orientamento verso sistemi clusterizzati poco costosi, ma pure poco esaltanti dal punto di vista prestazionale)
- 2002. *Earth Simulator* scalza dal primo posto della *TOP500 List* il sistema *IBM ASCI White*: gli Stati Uniti hanno ufficialmente perso la leadership in uno dei suoi settori strategici (primeggiare nel settore dei supercomputer è anzitutto simbolo di potenza economica)
- Il settore ritrova un nuovo slancio. In particolare *Cray* crea la serie *X1*

"Rinascita" del supercomputing
- Sfida ed Evoluzione Tecnologica -

- Macchine come i sistemi *X1* o *BlueGene/L* sono il punto di partenza per arrivare (**entro il 2010**) ad **1 PETAFLOPS**
- Secondo la *SIA* (*Semiconductor Industry Association*) nel giro di pochi anni in un solo chip potremo avere anche **più di 1 miliardo di transistor**
- Queste proiezioni, unite alle previsioni di crescita delle frequenze di clock (in meno di 10 anni si arriverà ai **10 GHz**), inevitabilmente influenzerà (e già ha influenzato) l'architettura dei calcolatori e, di conseguenza, le prestazioni e le possibilità applicative dei medesimi
- Il cosiddetto *Petascale Computing* non sembra così lontano
- Lo confermano anche analisi fatte esaminando l'evoluzione prestazionale dei supercomputer negli ultimi 10 anni: la potenza di **1 PETAFLOPS** sembrerebbe raggiungibile **già intorno al 2009**

Adesso che la *sfida al PETAFLOPS* è pienamente in atto, è certamente essa a tenere vivo il settore dei supercomputer e a garantire rapidi e continui miglioramenti