

Calcolo Parallelo e Distribuito II

Presentazione del corso - A.A. 2008-2009

Prof. Marco Lapegna

- tel. 081 675623
- studio 155 DMA (VI liv.)
- marco.lapegna@dma.unina.it

- sito web in fase di aggiornamento

Contenuti del corso

...Dal **Calcolo Parallelo** (e Distribuito mod.A)
al **Calcolo Parallelo e Distribuito** mod.B

Calcolo Parallelo e Distribuito I

Obiettivo del corso

Fornire idee, strumenti software e metodologie alla base della risoluzione computazionale di applicazioni in ambienti di **calcolo parallelo**

Calcolo parallelo e distribuito II

Obiettivo del corso

Fornire idee, strumenti software e metodologie alla base della risoluzione computazionale di applicazioni in ambienti di calcolo distribuito, su griglie computazionali e su CPU multicore

Prima domanda:

Che differenza c'è tra:

un **Calcolatore Parallelo**

e

un **Sistema** per il **Calcolo Distribuito**



Un calcolatore parallelo e'...



IBM Roadrunner

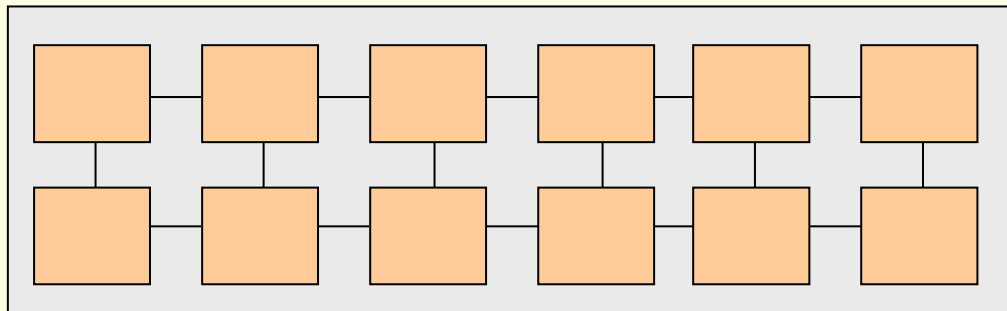
12960 processori IBM Cell

Performance sostenuta: **1.105** Pflops

Picco di performance: 1.456 Pflops

Los Alamos Nat. Lab

Calcolatore Parallelo



$$1 \text{ Pflops} = 10^{15} \text{ flop/sec}$$

Un calcolatore parallelo e'...



Jaguar

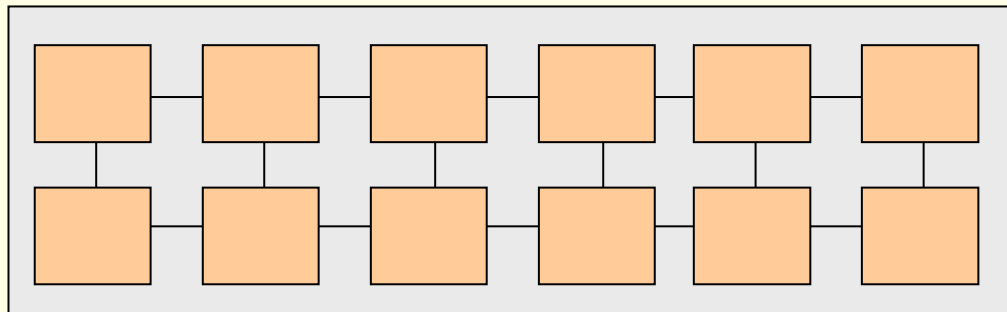
37538 processori AMD 4core

Performance sostenuta: 1.059 Pflops

Picco di performance: 1.381 Pflops

Oak Ridge Nat. Lab.

Calcolatore Parallelo





(<http://www.top500.org> - nov 2008)

Top five

	Cost.	computer	sito	Nprocs	Rete conn	Linpack bench. (Pflops)
1	IBM	Roadrunner	LANL (USA)	129600 core PowerXCell 3.2 GHz	Infiniband	1.105
2	CRAY	Jaguar	ORNL (USA)	150152 core AMD Opteron 4core 2.3 GHz	XT4 (custom)	1.059
3	SGI	Pleiades	NASA	51200 core Intel Xeon 4 core 3.0 GHz	Infiniband	0.487
4	IBM	Blue Gene/L	LLNL (USA)	212992 core Power PC 0.7 GHz	proprietary	0.478
5	IBM	Blu Gene/P	ANL (USA)	163840 core Power PC 0.85 GHz	proprietary	0.450

Alcune statistiche dalla Top500

Architettura:

- 410 cluster
- 88 MPP
- 2 altri

Rete:

- 282 Gbit Ethernet
- 141 Infiniband
- 42 proprietary
- 10 SP Switch
- 10 Myrinet
- 15 altri

Processori:

- 369 Intel Xeon
- 60 AMD Opteron
- 60 Power PC
- 9 Intel Itanium
- 2 altri

Num. Proc:

- 290 2049-4096
- 96 4097-8192
- 61 1025-2048
- 28 8193-16000
- 21 > 16000
- 4 < 1024

Un calcolatore parallelo e':

Un sistema di unità processanti **omogenee**

strettamente collegate

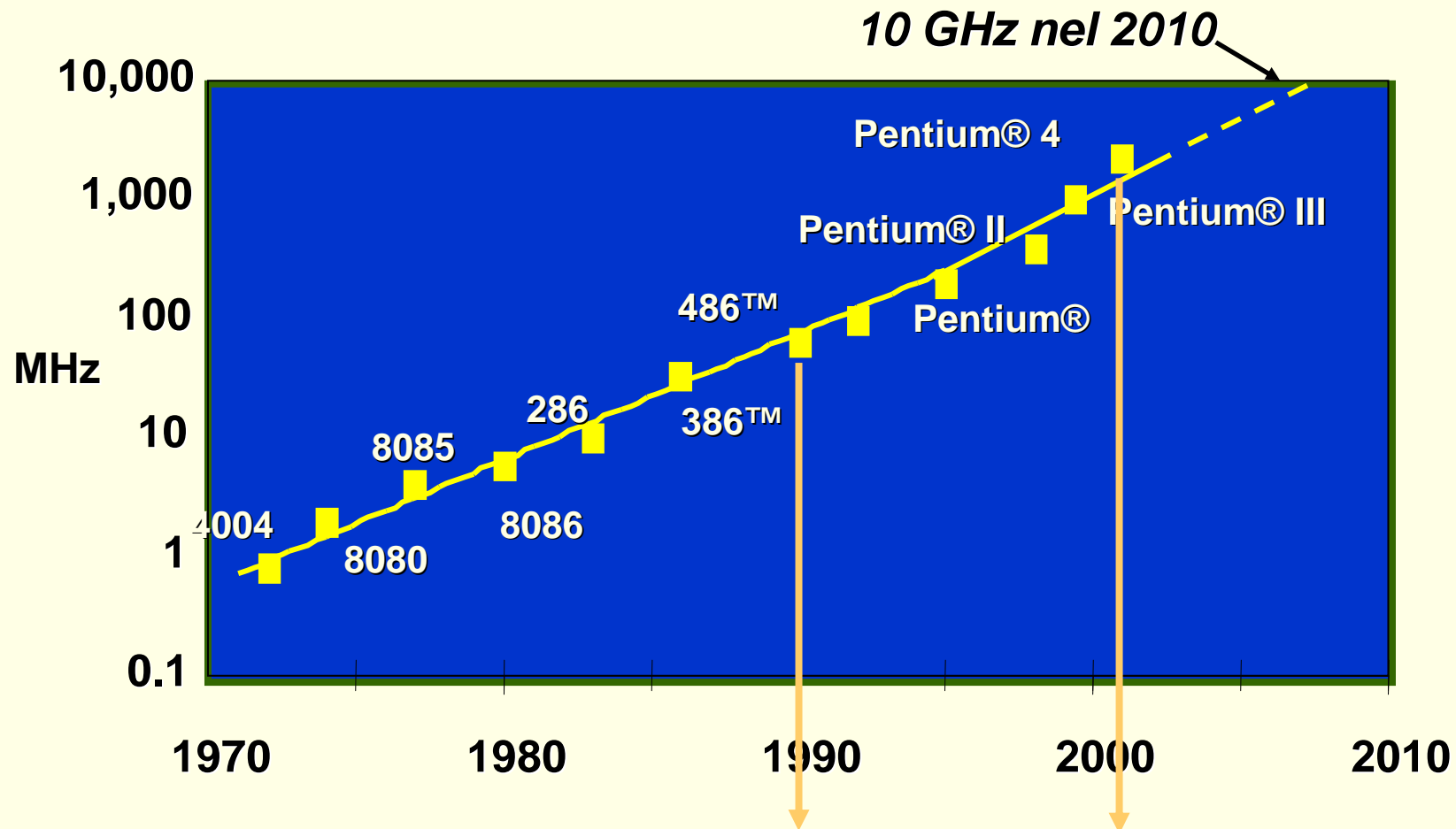
che **comunicano**

per risolvere problemi su larga scala

in maniera efficiente

***Risorse dedicate
e centralizzate***

Legge di Moore



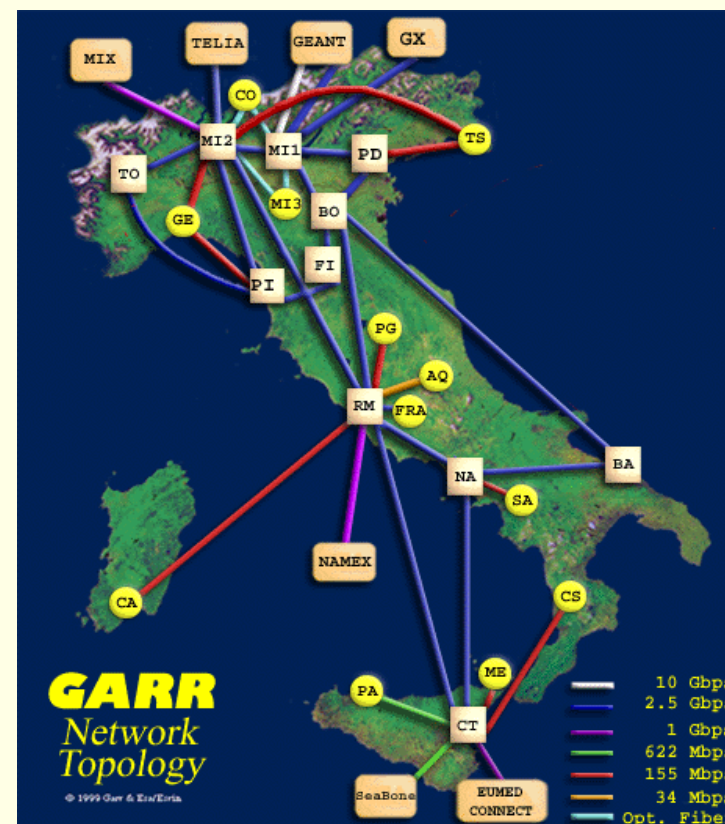
La velocità dei processori raddoppia ogni 18 mesi (crescita di 128 volte in 10.5 anni)

E il bandwidth delle reti?

Un esempio: La **rete GARR** e' la rete che connette tutte le istituzioni scientifiche italiane (universita', centri CNR, osservatori, ENEA, INFN)

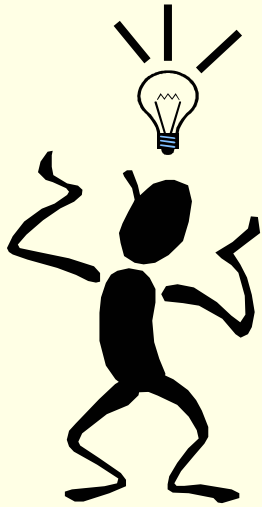
Evoluzione del bandwidth

GARR	1994	2Mbit/sec
GARR-2	1998	32Mbit/sec
GARR-B	2002	2.5Gbit/sec
GARR-G	2006	10Gbit/sec

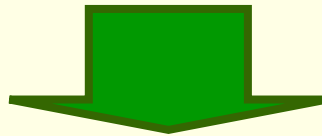


Crescita di
1000 volte in 8 anni
(x2 in 9 mesi)

IDEA

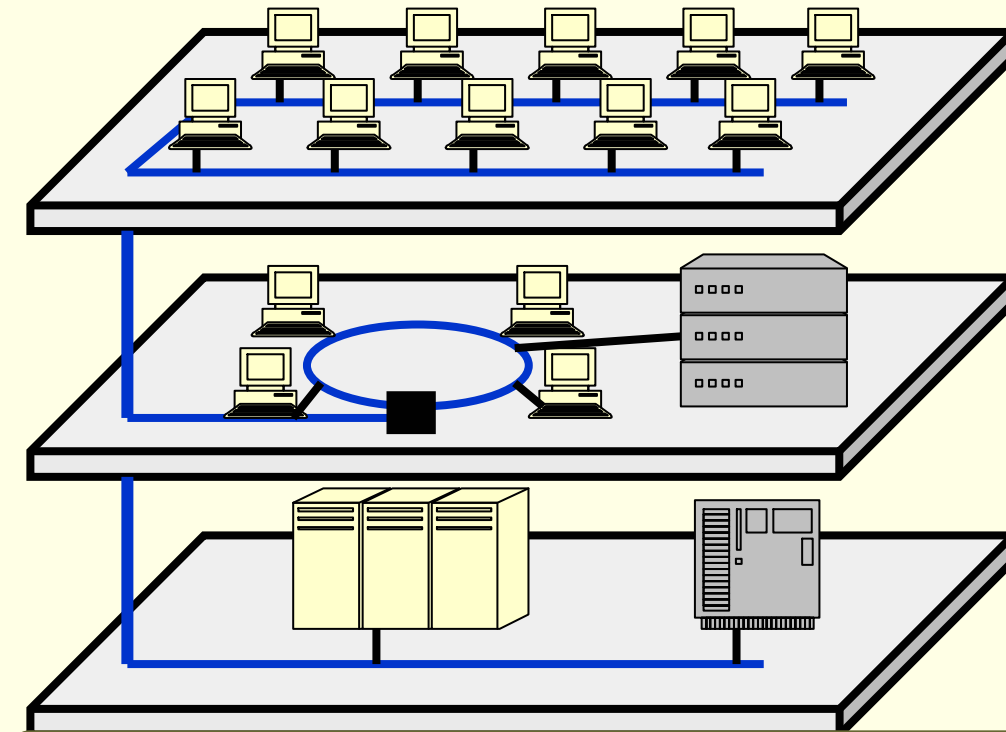


fattore di **crescita** della
velocita' delle **reti** di comunicazione
molto maggiore
del fattore di **crescita** della
velocita' dei **processori**



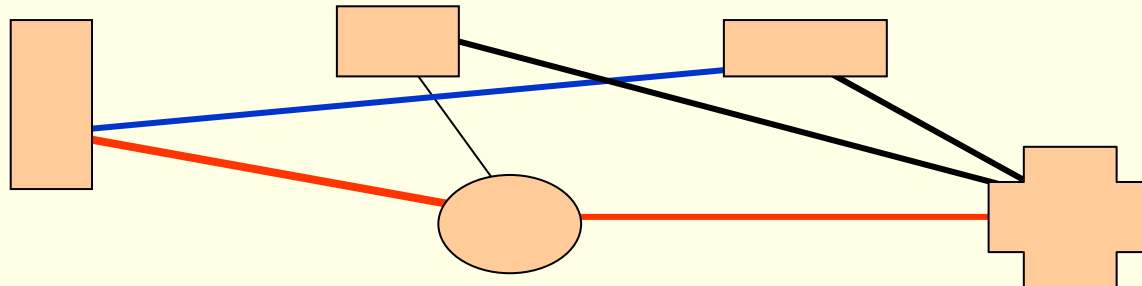
Posso considerare tutti i **calcolatori**
connessi ad una rete come una **unica**
risorsa di calcolo per risolvere problemi

un ambiente per il C.D. e' ...

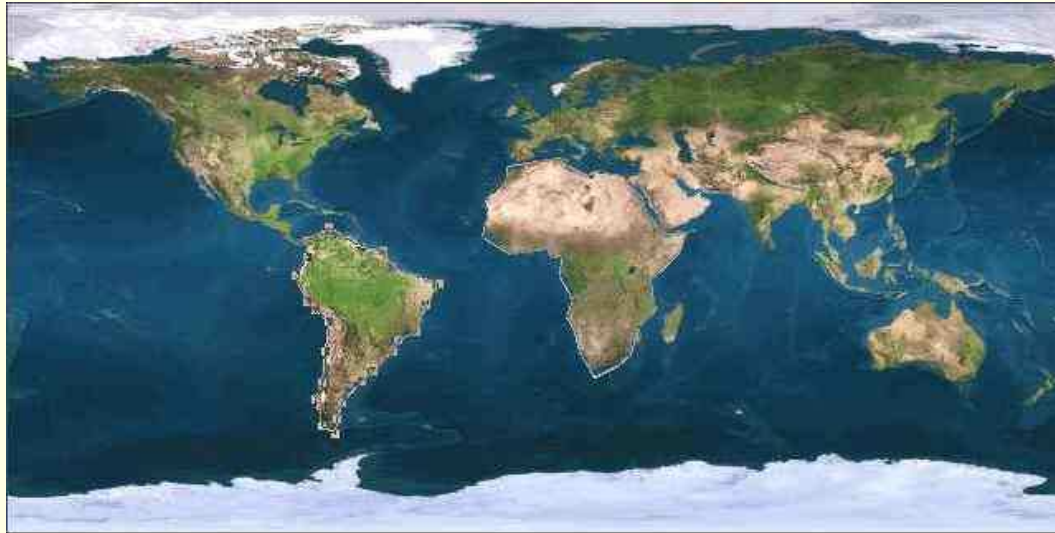


Una **Rete Aziendale** composta da **differenti calcolatori** collegati tra loro tra **reti differenti**

Calcolatore Distribuito

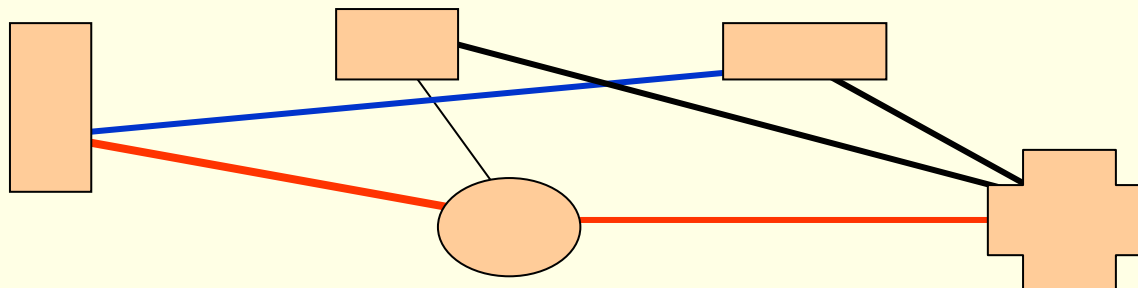


un ambiente per il C.D. e' ...



La **Rete** di tutti i
calcolatori
connessi tra loro
mediante **Internet**

Calcolatore Distribuito



Un ambiente per il C.D. e':

Un sistema di unità processanti **non omogenee**

autonome, indipendenti, geograficamente distribuite

che **sono aggregate**

per risolvere problemi su larga scala

in maniera efficiente

***Risorse condivise
e distribuite***

Prima risposta

Calcolatore parallelo

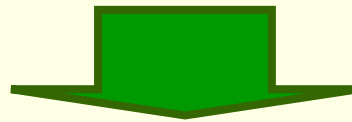


sistema di nodi collegati
da **switch specializzati e
dedicati**
(tightly coupled systems)

Sistema ad arch.
distribuita



sistema di nodi collegati
da **reti geografiche**
(loosely coupled systems)



La differenza e' nella **rete di connessione**

Caratteristiche di una rete di connessione

Tempo per trasmettere un messaggio di n bit

$$t_{comm} = \tau_0 + n \cdot \tau_{comm}$$

Tempo di latenza =
(tempo di start up)=
tempo di comunicazione
di un messaggio nullo

Tempo unitario di trasmissione =
Tempo per trasmettere un
messaggio di 1 bit

Velocita' di trasmissione

La **velocita' di trasmissione** di una rete e'
misurata mediante la
Bandwidth (ampiezza di banda) =
velocità di trasferimento di un messaggio
(bit/sec)

Es.: Bandwidth = 30 Mbit/sec



$$\tau_{comm} = \frac{1}{30 \cdot 10^6} \text{sec} = 0.33 \times 10^{-7} \text{sec}$$

esempio

	Latenza μ s	Bandwidth Gbit/sec	topologia
Conn. ADSL	1000	< 0.01 (eff. Disponib.)	
LAN Fast Ethernet	100	< 0.1 (eff. Disponib.)	bus
LAN Giga Ethernet	30	1	bus
Infiniband 4x	6	6.5	tree
Myrinet (E)	6	7	switch
QsNetII (R)	3	7	tree
IBM Blue Gene	1 - 5	20	toro 3D

Rete
lenta



Rete
veloce

Differenze tra le reti di connessione

	Switch dedicati	Reti LAN (Locali)	Reti WAN (Geografiche)
bandwidth	~ 10 Gb/sec	0.01 -0.1 Gb/sec	<0.001 Gb/sec
latenza	< 10 μ sec	10 - 100 μ sec	> 1000 μ sec

↓
•IBM BlueGene
•NEC ESC

↓
•cluster
•beowulf

↓
• Internet

← Parallelo → Distribuito →

Seconda domanda:

Che differenza c'è tra gli obiettivi di:

un Calcolatore Parallelo



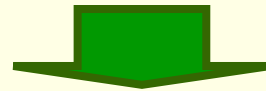
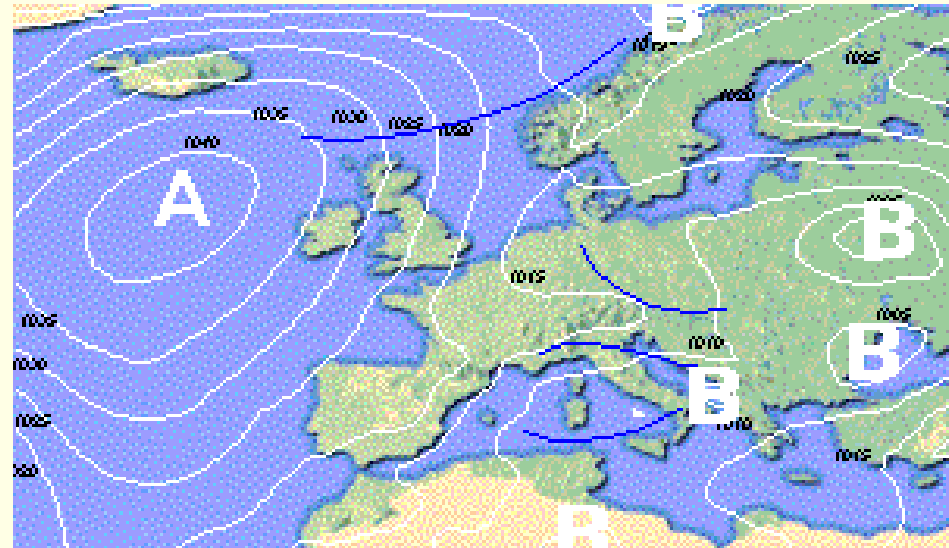
e



di un Sistema per il Calcolo Distribuito

Esempio: previsioni meteorologiche

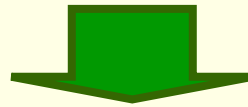
Superficie =
20 milioni di Km²
Altezza s.l.m. =
di 20 Km
Discretizzazione =
cubi di lato 100m
(1Km³ = 10³ cubi)



$$20 \times 10^6 \text{ Km}^2 \times 20 \text{ Km} = 4 \times 10^8 \text{ Km}^3$$
$$= 4 \times 10^8 \times 10^3 = 4 \times 10^{11} \text{ cubi}$$

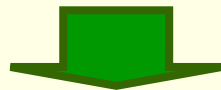
Esempio (cont.)

- previsioni per i prossimi 2 giorni (48 ore)
- ogni ora di simulazione in ogni cubo necessarie 10^2 flop



Intera superficie = $4 \times 10^{11} \times 10^2 \times 48 =$
 $\sim 2 \times 10^{15}$ operazioni

PC 1 Gflops



2×10^6 sec
(23 giorni ???)

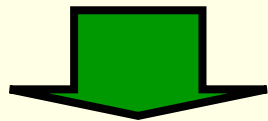
roadrunner



$\sim 10^2$ sec
(2 minuti !!!)

Calcolo parallelo ...al fine di ...

- Ridurre il tempo necessario alla risoluzione computazionale di un problema



Suddivisione del problema



Esempio : Progetto SETI

- Progetto per la ricerca di intelligenze extraterrestri
- Dati raccolti dal radiotelescopio di Arecibo
 - 2 anni di raccolta dati (1998-2000)
- Trovare sequenze regolari nelle frequenze all'interno dei segnali raccolti



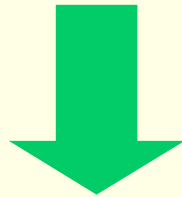
39 Tbytes =
1100 nastri da 35 Gbyte

Le dimensioni del progetto

- ogni nastro da 35 Gbyte e' diviso in oltre 150000 **workunit** di circa 350 Kb
- ogni workunit contiene i segnali raccolti in circa 100 sec su ognuna delle quali vengono eseguite **FFT con diversi campionamenti** ed analisi statistiche per circa **280 miliardi flops** (circa 40 ore su Pentium II 500 MHz)
- totale 165×10^8 blocchi di circa 100 sec., per un totale di 460×10^{19} flops
(quasi 1 mese su Roadrunner !!)

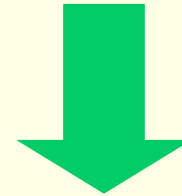
Altre caratteristiche

Le workunits sono
indipendenti



Predisposizione al
parallelismo

Non c'è urgenza
ad avere il
risultato



Assenza di vincoli
temporali

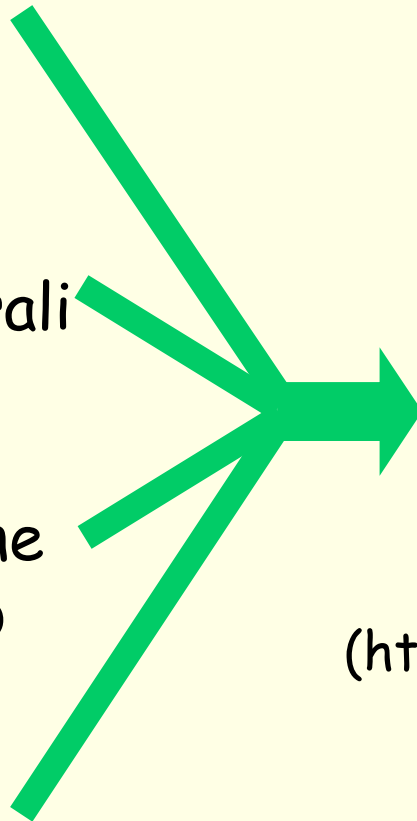
Driving forces

Grandi
dimensioni

Assenza di
vincoli temporali

Predisposizione
al parallelismo

Disponibilita'
di risorse



IDEA !!

Utilizzare i
personal computer
nel mondo
connessi ad Internet
e lasciati inutilizzati

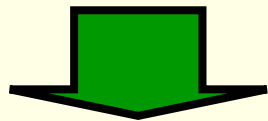
(<http://setiathome.ssl.berkeley.edu/>)

Come partecipare?

- collegarsi a (<http://setiathome.ssl.berkeley.edu/>)
- scaricare e installare un apposito software sul PC
- Quando il PC e' inutilizzato, in maniera analoga ad uno screensaver, il software si attiva
 - Scarica i dati (~350KBytes) dal server
 - li analizza (circa 10 ore CPU su Pentium4 a 2.0 GHz)
 - restituisce al server i risultati, che vengono combinati con quelli degli altri PC
- ~ 5000000 partecipanti → ~ 100 Tflops di media

Calcolo distribuito .. al fine di ..

- riutilizzare
"efficacemente"
risorse hardware e
software distribuite
geograficamente sul
territorio



Aggregazione di risorse



Seconda risposta

Calcolatore parallelo

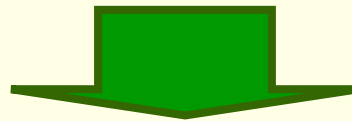


Principale obiettivo:
Riduzione efficiente dei
tempi di esecuzione

Sistema ad arch.
distribuita



Principale obiettivo:
Aggregazione efficiente di
risorse esistenti



La differenza e' nelle **esigenze**
delle applicazioni da risolvere

Terza domanda:

Che differenza c'è tra l'uso di:

un **Calcolatore Parallelo**



e

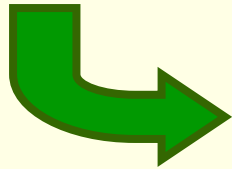


un **Sistema** per il **Calcolo Distribuito**

uso delle risorse

Calcolatore parallelo:

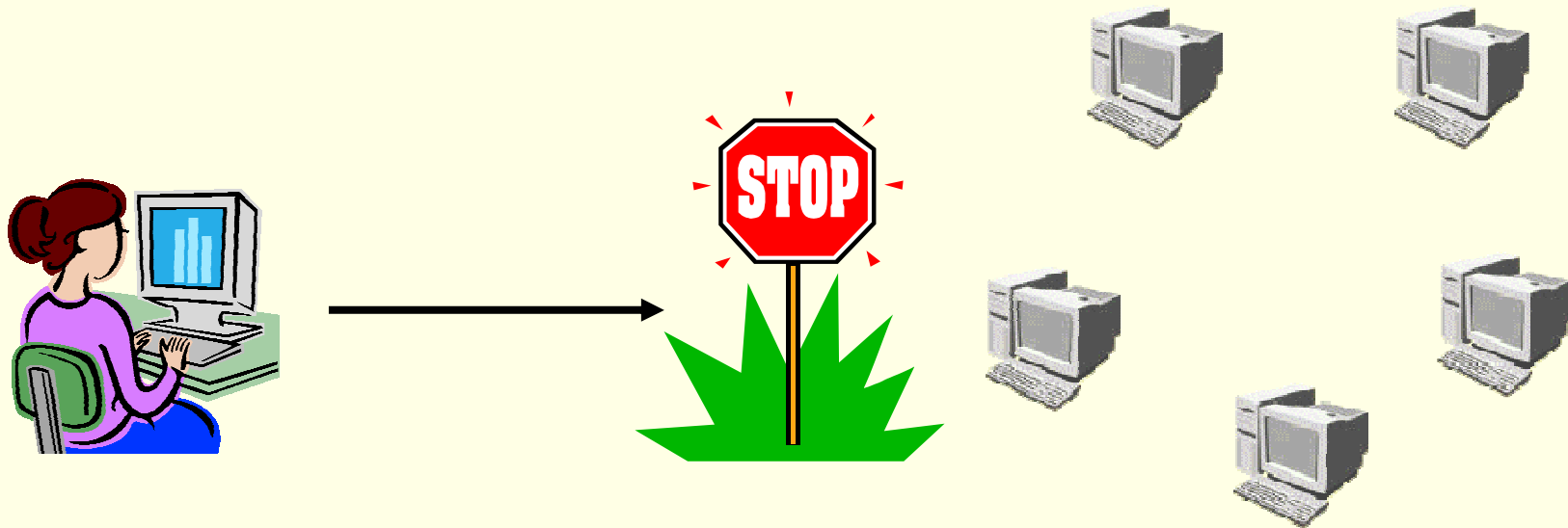
- risorse di calcolo omogenee
- risorse sotto controllo diretto



Uso di **librerie di comunicazione** (es. MPI)

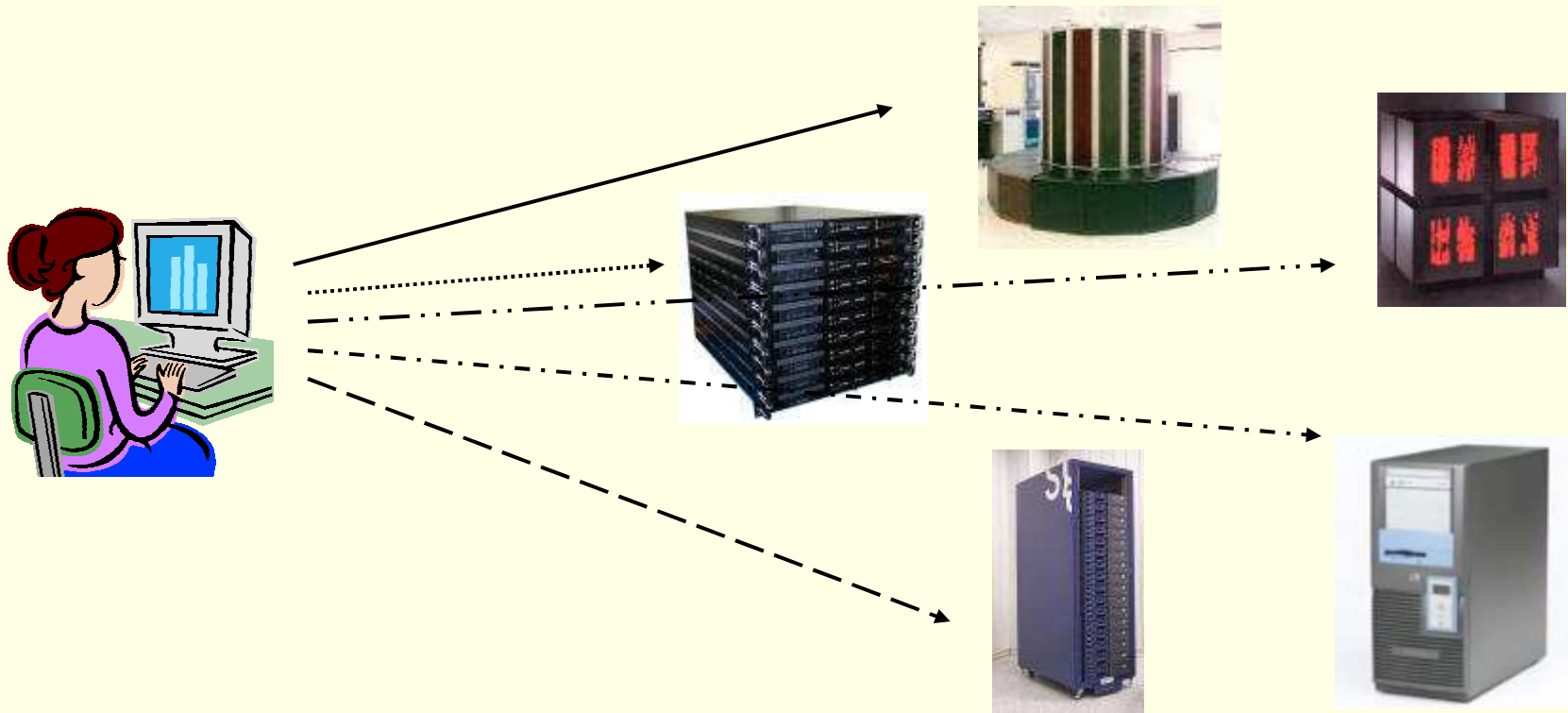
Quali sono i problemi nell'utilizzo di un
sistema per il **Calcolo Distribuito**?

In un ambiente per il C.D. ...



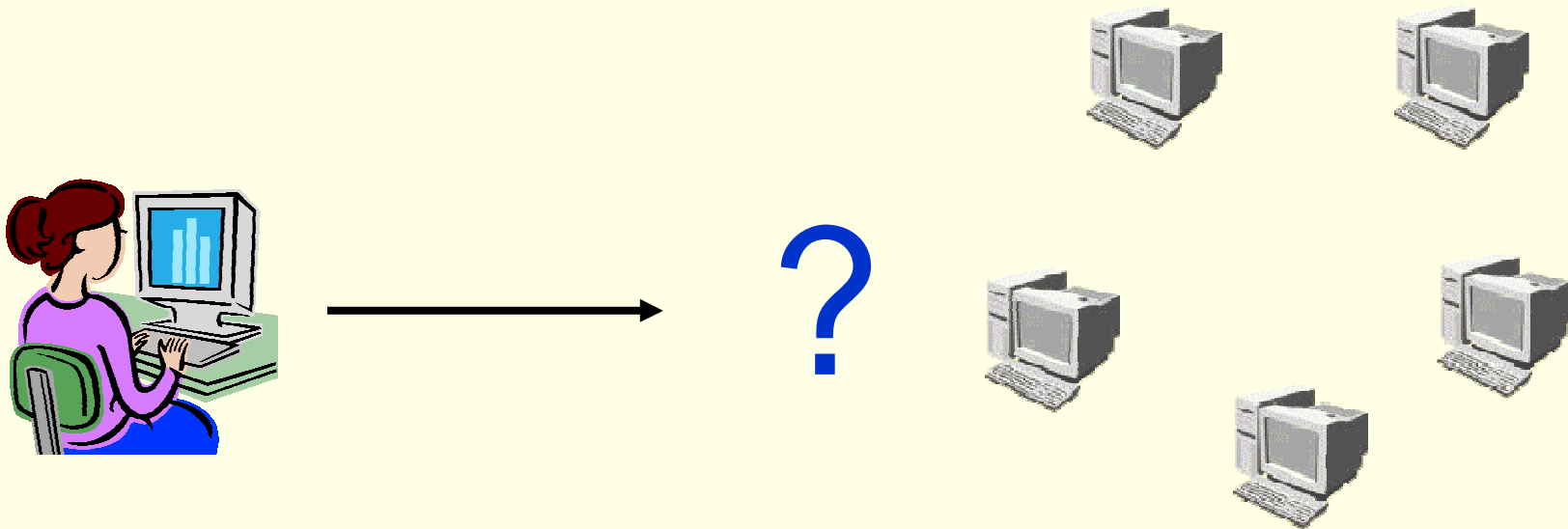
L'utente potrebbe non avere le **autorizzazioni** per accedere a tutte le macchine

In un ambiente per il C.D. ...



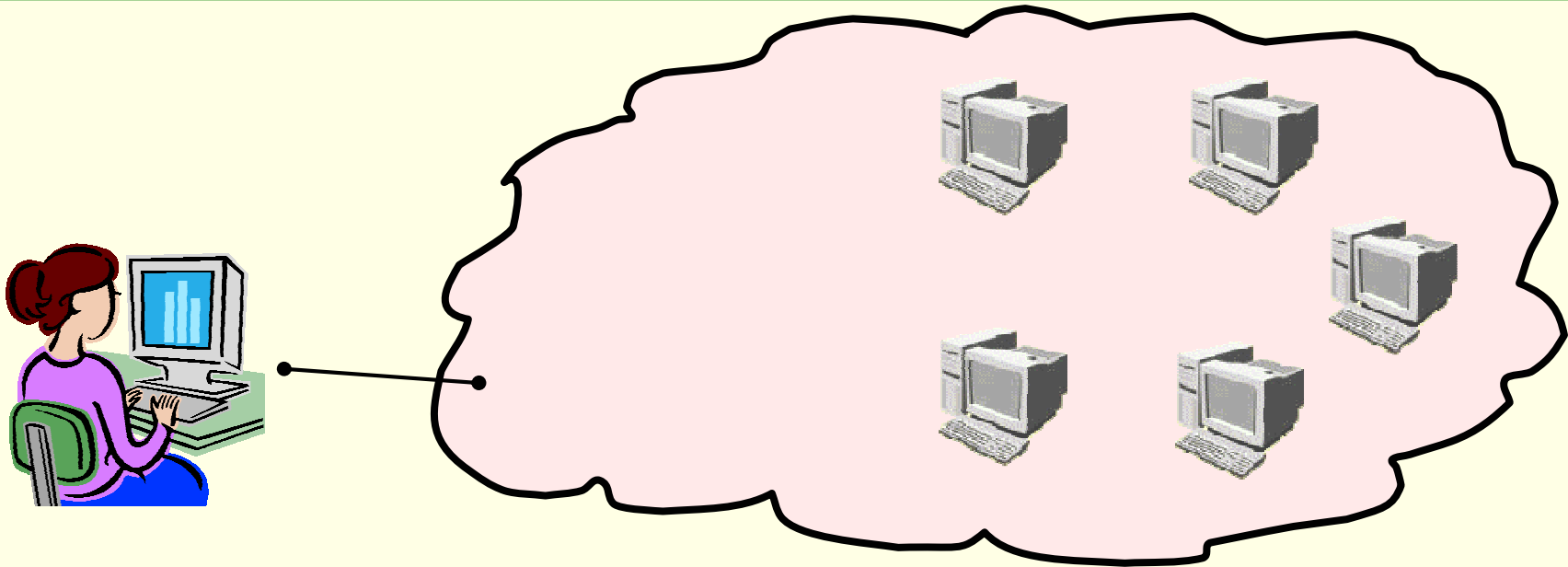
L'utente deve gestire di persona **l'eterogeneità** delle risorse (sistemi operativi, aritmetica f.p., rappresentazione dati, architettura,...)

In un ambiente per il C.D. ...



L'utente deve verificare di persona la **disponibilita'** delle risorse e le loro caratteristiche **statiche** (performance) e **dinamiche** (carico di lavoro)

Una soluzione

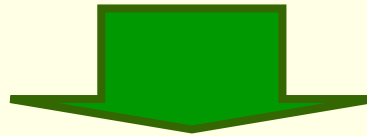


Utilizzare un **ambiente software** tra l'utente e le risorse,
in maniera che

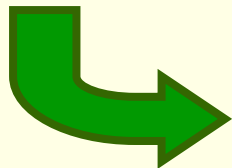
- all'utente appaia un **singolo sistema** di calcolo
- risolva i **problemi di eterogeneità, dinamicità e di sicurezza** dell'ambiente

Quindi ...

Uso di un Sistema per il Calcolo Distribuito



- risorse non dedicate
- risorse di calcolo non omogenee
- risorse su cui non si ha controllo diretto



Uso di **ambienti software** sofisticati

Terza risposta

Calcolatore parallelo



Compiti dell'ambiente sw:

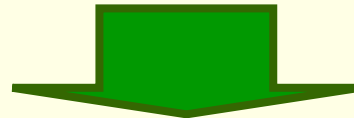
- trasf. dati tra le risorse
(overhead < 5%)

Sistema ad arch. distribuita



Compiti dell'ambiente sw.:

- autenticazione,
- gestione eterogeneita', dinamicita', asincronia
- selezione risorse
- trasf. dati
(overhead > 20%)



La differenza e' nel peso dell'**ambiente software**

Quarta domanda:

Che differenza c'è tra il costo di:

un **Calcolatore Parallelo**



e

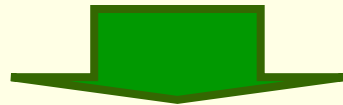


di un **Sistema** per il **Calcolo Distribuito**

Esempio: IBM BlueGene

L'investimento per la costruzione =
200 M USD ~ 170 M EUR
(senza contare la gestione quotidiana...)

Performance =
450 Tflops

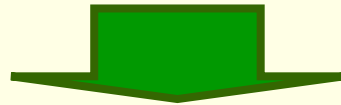


Costo di 1 Tflops = $170/450 =$
0.3 M USD

Esempio: SETI@home

L'investimento per i server =
0.5 M USD ~ 0.4 M EUR

performance =
100 Tflops



Costo di 1 Tflops = $0.4/100 =$
0.005 M USD = 4000 EUR !!

Quarta risposta

Calcolatore parallelo

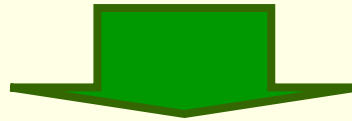


Costi hardware
notevoli

Sistema ad arch.
distribuita



Costi hardware
trascurabili



La differenza e' nell'efficienza dell'investimento

Parallelo vs distribuito

Calcolatore parallelo

- reti veloci
- risorse limitate
- risorse dedicate e omogenee
- applicazione gestisce le risorse
- costo hardware notevole
- overhead sw sistema < 5%
- presenza di vincoli temporali
- es. IBM Roadrunner
 - 16000 CPU
 - 1 Pflops

Ambiente per il C.D.

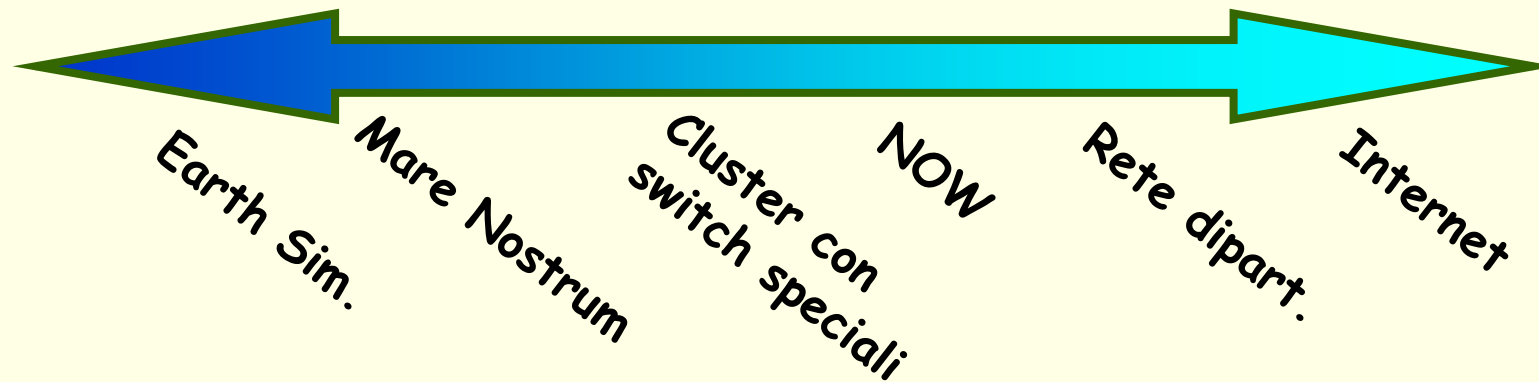
- reti lente
- risorse potenzialmente illimitate
- risorse condivise e disomogenee
- ambiente sw.gestisce le risorse
- costo hardware trascurabile
- overhead sw sistema > 20%
- assenza di vincoli temporali
- es. SETI@home
 - 5 milioni CPU
 - 100 Tflops

Molte differenze !!!

Parallelo vs distribuito

Calcolatore parallelo

Ambiente per il C.D.



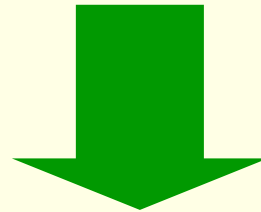
Ma non una differenza netta !!!

Definizione

Calcolo distribuito

=

Risoluzione efficiente di un problema su un sistema ad architettura distribuita



Studio delle metodologie di base, degli algoritmi e conoscenza dei relativi strumenti software

Sinonimi di calcolo distribuito

- network computing
- heterogenous computing
- cluster computing
- meta computing
- ...

Enfasi sui **diversi aspetti** del
Calcolo distribuito

Bibliografia e materiale

Generalita' sul calcolo parallelo e distribuito:

- A. Murli - Lezioni di Calcolo Parallelo
- I. Foster - Designing and Building Parallel Programs,
on-line textbook → <http://www-unix.mcs.anl.gov/dbpp/>
- Fox, Williams, Messina - Parallel Computing Works,
on-line textbook → <http://www.netlib.org/utk/lsi/pcwLSI/text/>

- copia dei lucidi
- link internet
- rapporti tecnici e lavori

distribuiti a lezione
e via e-mail

Per cominciare:

- Dongarra, Meuer, Simon and Strohmaier - High Performance Computing Today - <http://www.netlib.org/utk/people/JackDongarra/PAPERS/hpc-today.pdf>

Dalla precedente lezione

Calcolatore parallelo

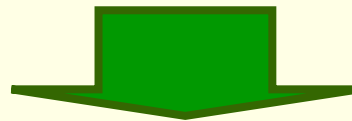


Principale obiettivo:
Riduzione efficiente dei
tempi di esecuzione

Sistema ad arch.
distribuita



Principale obiettivo:
Aggregazione efficiente di
risorse esistenti



Differenti obiettivi

Parallelo vs distribuito

Calcolatore parallelo

- reti veloci
- risorse limitate
- risorse dedicate e omogenee
- applicazione gestisce le risorse
- costo hardware notevole
- overhead sw sistema < 5%
- presenza di vincoli temporali
- es. Roadrunner
 - 16000 processori
 - 1 Pflops

Ambiente per il C.D.

- reti lente
- risorse potenzialmente illimitate
- risorse condivise e disomogenee
- ambiente sw.gestisce le risorse
- costo hardware trascurabile
- overhead sw sistema > 20%
- assenza di vincoli temporali
- es. SETI@home
 - 5 milioni processori
 - 100 Tflops

Molte differenze !!!

Domanda

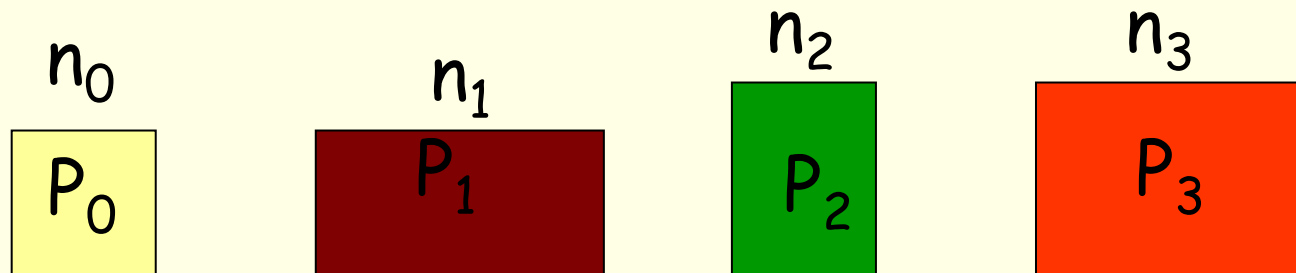
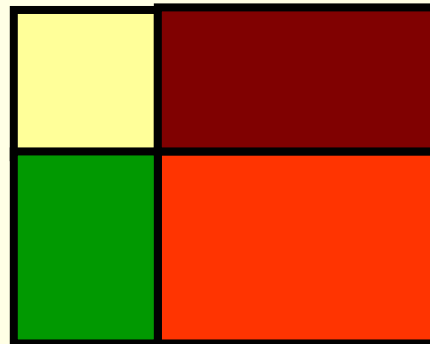
Quali sono le applicazioni
"predisposte"
al calcolo parallelo
(o *naturalmente parallele*) ?

Quelle che **riducono efficientemente il tempo di esecuzione !!**

Che vuol dire "naturalmente parallela?"

Supponiamo che il problema si possa decomporre in 4 sottoproblemi **indipendenti** ma di **complessità diversa** ...

N
Numero di operazioni
del problema completo



n_i = numero di operazioni del sottoproblema i -mo

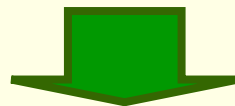
Assumiamo che

- i processori siano **omogenei**
 - abbiano lo **stesso carico** di lavoro
- (ipotesi non restrittive per un calcolatore parallelo)



Detto

t_i il tempo di esecuzione del i -mo problema
sul processore P_i



$$t_i = K n_i \quad (K \text{ costante per tutti i processori})$$

Inoltre, ...

detto T_1 il tempo di esecuzione del problema su 1 processore $\Rightarrow T_1 = KN$ cioè $N = T_1/K$

Supponiamo ora che:

$$n_0 = 0.2 N \quad n_1 = 0.1 N \quad n_2 = 0.2 N \quad n_3 = 0.5 N$$

$$t_0 = K n_0 = 0.2 K N = 0.2 T_1$$

analogamente $t_1 = 0.1 T_1 \quad t_2 = 0.2 T_1 \quad t_3 = 0.5 T_1$

$$T_4 = \max(t_i) = t_3 = 0.5 T_1$$

quindi

$$S_4 = \frac{T_1}{T_4} = \frac{T_1}{0.5T_1} = 2$$

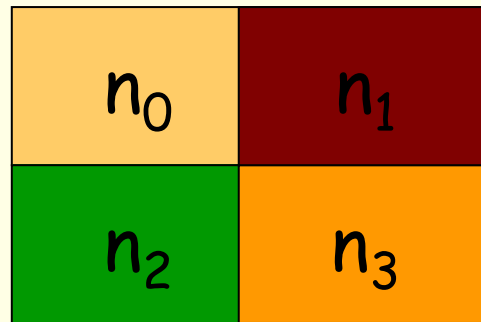
Anche se i sottoproblemi sono indipendenti
lo **speed up** sui 4 proc. è **estremamente basso** !



riduzione **non efficiente** del tempo
(**Problema non predisposto** al calcolo parallelo)

Se invece

Si riesce a decomporre il problema
in 4 sottoproblemi equivalenti



se $n_0 = n_1 = n_2 = n_3$



$$T_4 = t_0 = T_1 / 4$$

ovvero

$$S_4 = \frac{T_1}{T_4} = \frac{T_1}{T_1/4} = 4$$

riduzione **efficiente** del tempo



...in questo caso l'applicazione è
naturalmente parallela

In definitiva...

Un'applicazione è **naturalmente parallela** se



$$S_p \approx p \iff T_p \approx \frac{T_1}{p}$$

Domanda

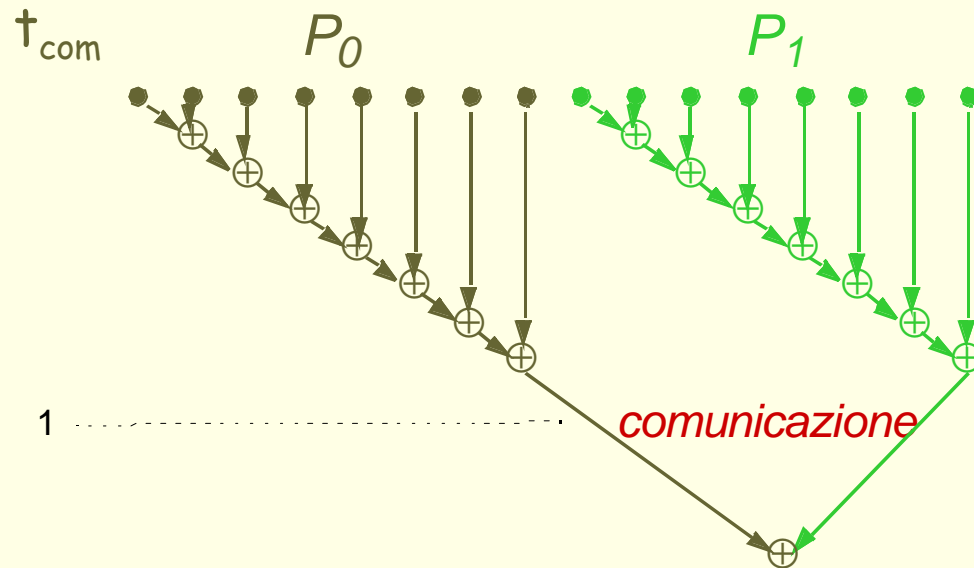
Quali sono le applicazioni
"predisposte"
al calcolo distribuito

Quelle che **aggregano efficientemente le risorse** di calcolo !!

Esempio: calcolo della somma di $n=16$ numeri

ALGORITMO PARALLELO

$p=2$



L'algoritmo richiede

8 addizioni e 1
comunicazione



$$T_2 = 8 t_{calc} + 1 t_{com}$$

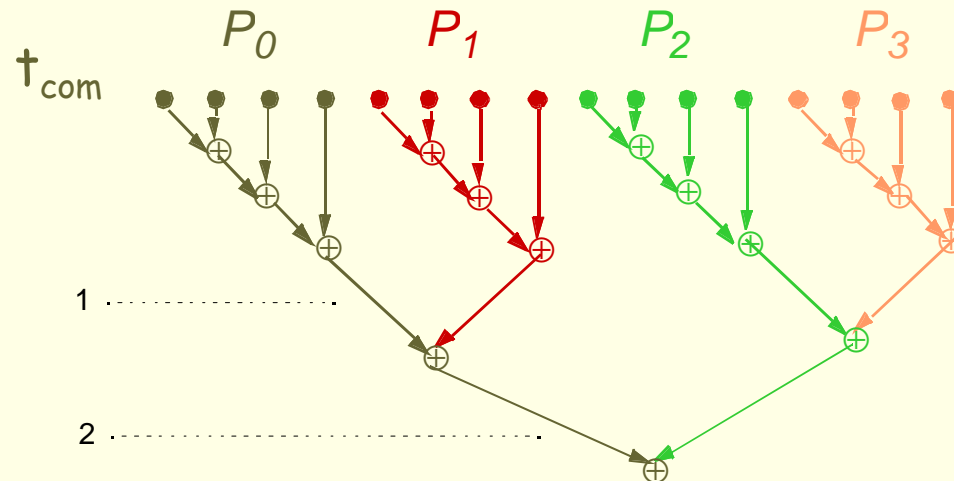
t_{calc} = tempo di esecuzione unitario (di 1 op. fl.p.)

t_{com} = tempo di comunicazione unitario (di 1 dato fl.p. = 8 bytes)

Esempio: calcolo della somma di $n=16$ numeri

ALGORITMO PARALLELO

$p=4$



L'algoritmo richiede 5
addizioni e 2
comunicazioni

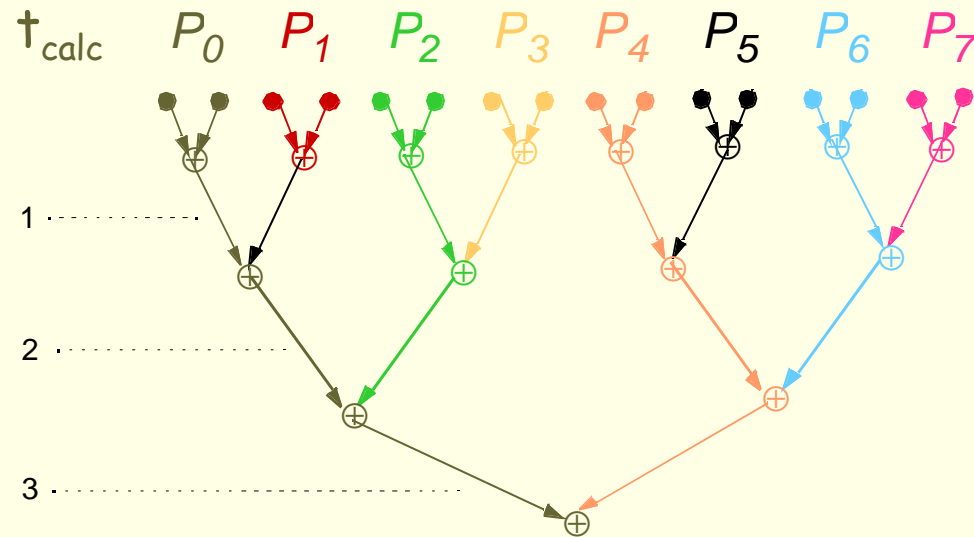


$$T_4 = 5 t_{calc} + 2 t_{com}$$

Esempio: calcolo della somma di $n=16$ numeri

ALGORITMO PARALLELO

$p=8$



L'algoritmo richiede 4
addizioni e 3
comunicazioni



$$T_8 = 4 t_{\text{calc}} + 3 t_{\text{com}}$$

Esempio: calcolo della somma di $n=16$ numeri

$$T_1 = 15t_{\text{calc}}$$

Supponendo

$$\frac{t_{\text{com}}}{t_{\text{calc}}} = 0$$

(trascurando la comunicazione)

p	T_p	S_p	E_p
2	$8t_{\text{calc}}$	1.88	0.94
4	$5t_{\text{calc}}$	3.00	0.75
8	$4t_{\text{calc}}$	3.75	0.47

Supponendo

$$\frac{t_{\text{com}}}{t_{\text{calc}}} = 2$$

p	T_p	S_p	E_p
2	$10t_{\text{calc}}$	1.50	0.75
4	$9t_{\text{calc}}$	1.67	0.42
8	$10t_{\text{calc}}$	1.50	0.19

Esempio: calcolo della somma di $n=16$ numeri

Supponendo

$$\frac{t_{\text{com}}}{t_{\text{calc}}} = 4$$

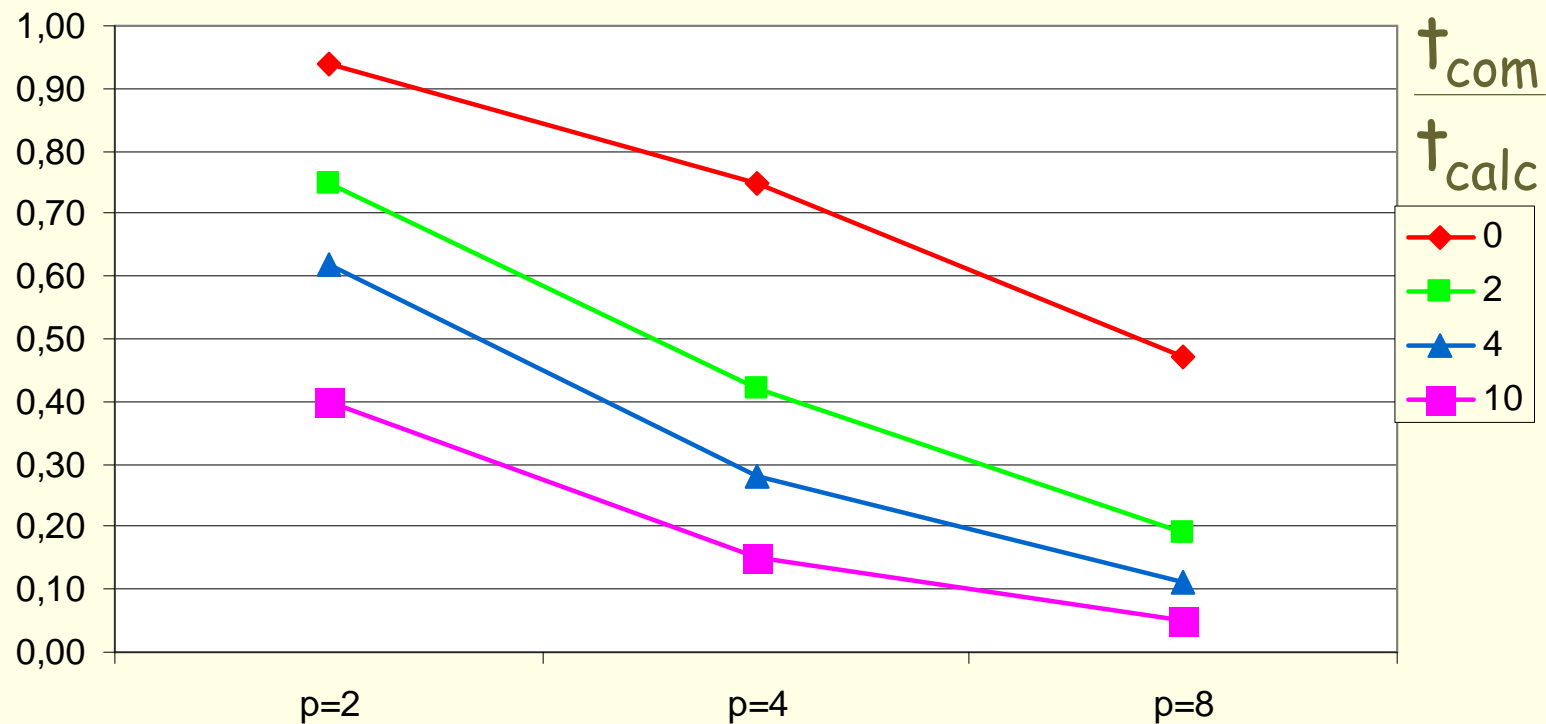
p	T_p	S_p	E_p
2	$12t_{\text{calc}}$	1.25	0.62
4	$13t_{\text{calc}}$	1.15	0.28
8	$16t_{\text{calc}}$	0.93	0.11

Supponendo

$$\frac{t_{\text{com}}}{t_{\text{calc}}} = 10$$

p	T_p	S_p	E_p
2	$18t_{\text{calc}}$	0.8	0.4
4	$25t_{\text{calc}}$	0.6	0.15
8	$34t_{\text{calc}}$	0.4	0.05

Efficienze




le prestazioni dell' algoritmo
possono cambiare notevolmente!

In generale...

Qual è l'**impatto della comunicazione**
sull'**efficienza** di
un algoritmo in ambiente parallelo/distribuito?

Impatto sull'efficienza

$$E_P = \frac{T_1}{PT_P} = \frac{T_1}{P(T_{comm} + T_{calc})}$$

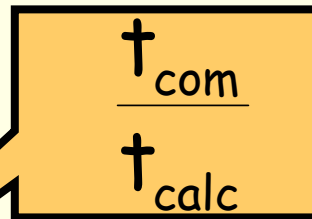
Poiche' $T_{calc} \geq \frac{T_1}{P}$  $E_P \leq \frac{T_{calc}}{T_{comm} + T_{calc}}$

Cioe': $E_P \leq \frac{1}{1 + \frac{T_{comm}}{T_{calc}}} = \frac{1}{1 + OC_P}$

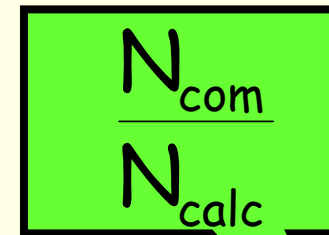
In GENERALE

Per l'overhead totale di comunicazione risulta ...

$$OC_p = \frac{T_p^{com}}{T_p^{calc}} =$$


$$\frac{t_{com}}{t_{calc}}$$

×


$$\frac{N_{com}}{N_{calc}}$$

Dipendenza
dall'ambiente (hw/sw)
di calcolo

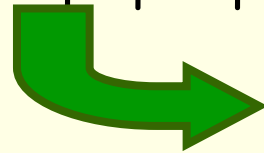
Dipendenza
dall'algoritmo

Quanto vale $t_{\text{comm}}/t_{\text{calc}}$?

IBM BlueGene

Bandw. = 2.8 GB/sec $\rightarrow t_{\text{comm}} = 1.4 \times 10^{-9}$ sec

P.P. = 2.4 Gflops per proc. $\rightarrow t_{\text{calc}} = 0.41 \times 10^{-9}$ sec

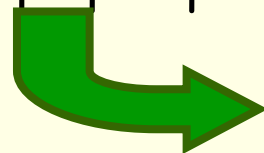


$$t_{\text{comm}}/t_{\text{calc}} = 3.4$$

Beowulf

Bandw. = 0.15 GB/sec $\rightarrow t_{\text{comm}} = 26 \times 10^{-9}$ sec

P.P. = 2 Gflops per proc. $\rightarrow t_{\text{calc}} = 0.5 \times 10^{-9}$ sec



$$t_{\text{comm}}/t_{\text{calc}} = 53.3$$

Ambiente C.D.

Bandw. = 0.001 GB/sec $\rightarrow t_{\text{comm}} = 4000 \times 10^{-9}$ sec

P.P. = 2 Gflops per proc. $\rightarrow t_{\text{calc}} = 0.5 \times 10^{-9}$ sec



$$t_{\text{comm}}/t_{\text{calc}} = 8000$$

Quanto vale $N_{\text{comm}}/N_{\text{calc}}$?

Per l'algoritmo della somma:

$$P=2 \rightarrow 1/8 = 0.12$$

$$P=4 \rightarrow 2/5 = 0.4$$

$$P=8 \rightarrow 3/4 = 0.75$$

Quanto vale OC_p ?

IBM BlueGene

$$P=2 \rightarrow OC_p = 3.4 \times 0.12 = 0.41$$

$$P=4 \rightarrow OC_p = 3.4 \times 0.4 = 1.36$$

$$P=8 \rightarrow OC_p = 3.4 \times 0.75 = 2.55$$

Beowulf

$$P=2 \rightarrow OC_p = 53.3 \times 0.12 = 6.4$$

$$P=4 \rightarrow OC_p = 53.3 \times 0.4 = 21.3$$

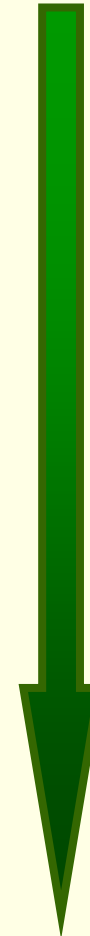
$$P=8 \rightarrow OC_p = 53.3 \times 0.75 = 40$$

Ambiente C.D.

$$P=2 \rightarrow OC_p = 8000 \times 0.12 = 960$$

$$P=4 \rightarrow OC_p = 8000 \times 0.4 = 3200$$

$$P=8 \rightarrow OC_p = 8000 \times 0.75 = 6000$$

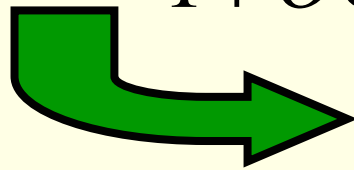


Overhead
basso

Overhead
alto

Quindi...

da $E_P < \frac{1}{1 + OC_P}$



	E_p
$OC=1$	< 0.5
$OC=10$	< 0.1
$OC=100$	< 0.01
$OC=1000$	< 0.001

Somma su
IBM BG

Somma su
beowulf

Somma in
ambiente C.D.

Impatto devastante!!



Aggregazione **non efficiente**
delle risorse

VICEVERSA



Che OC_p si puo' tollerare se si vuole una data efficienza?

Efficienza vs OC

$$\text{Da } E_P < \frac{1}{1 + OC_P} \Rightarrow OC_P < \frac{1 - E_P}{E_P}$$

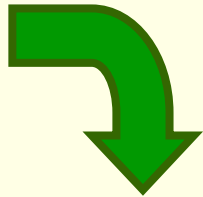
	OC
$E_p=0.95$	< 0.05
$E_p=0.9$	< 0.11
$E_p=0.8$	< 0.25
$E_p=0.5$	< 1

esempio: $E_p=0.8 \rightarrow OC_p < 0.25$

Poiche' in un ambiente distribuito

$$OC_p = \frac{T_p^{com}}{T_p^{calc}} = \boxed{\frac{t_{com}}{t_{calc}}} \times \frac{N_{com}}{N_{calc}}$$

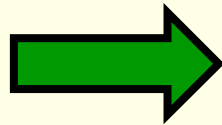
è maggiore di 8000


$$\frac{N_{com}}{N_{calc}} \approx 10^{-4}$$

In generale, se si vuole $E_p=0.8$

IBM BG

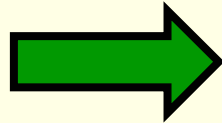
$$\frac{t_{\text{com}}}{t_{\text{calc}}} = 3.4$$



$$\frac{N_{\text{com}}}{N_{\text{calc}}} \approx 10^{-1}$$

beowulf

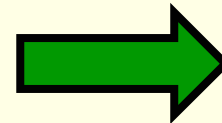
$$\frac{t_{\text{com}}}{t_{\text{calc}}} = 53.5$$



$$\frac{N_{\text{com}}}{N_{\text{calc}}} \approx 10^{-2}$$

Ambiente C.D.

$$\frac{t_{\text{com}}}{t_{\text{calc}}} = 8000$$



$$\frac{N_{\text{com}}}{N_{\text{calc}}} \approx 10^{-4}$$

Quindi

IBM BG

$$\frac{N_{\text{com}}}{N_{\text{calc}}} \approx 10^{-1}$$



Circa 1
comunicazione
ogni 10
operazioni f.p.

beowulf

$$\frac{N_{\text{com}}}{N_{\text{calc}}} \approx 10^{-2}$$



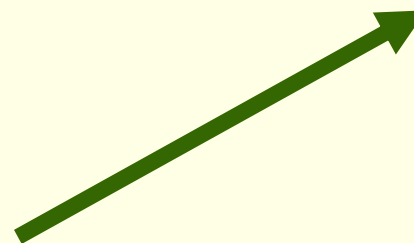
Circa 1
comunicazione
ogni 100
operazioni f.p.

Ambiente C.D.

$$\frac{N_{\text{com}}}{N_{\text{calc}}} \approx 10^{-4}$$



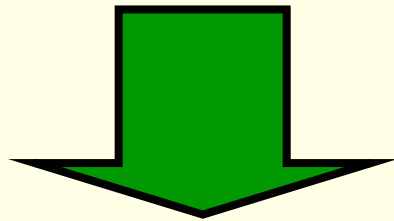
Circa 1
comunicazione
ogni 10000
operazioni f.p.



Numero di comunicazioni trascurabili!!!

In definitiva...

Quali sono le applicazioni "predisposte"
al calcolo distribuito ?



quelle per cui il **numero di comunicazioni**
è **praticamente trascurabile**
rispetto al numero di operazioni

Osservazione

Supponiamo che

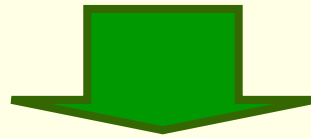
$t_{\text{comm}}=0$ (comunicazione gratis !!)

Possiamo risolvere un problema
con poche comunicazioni
in un ambiente di calcolo distribuito
ed avere una efficienza elevata?

Osservazione

Per un **Calcolatore Parallelo** abbiamo assunto che:

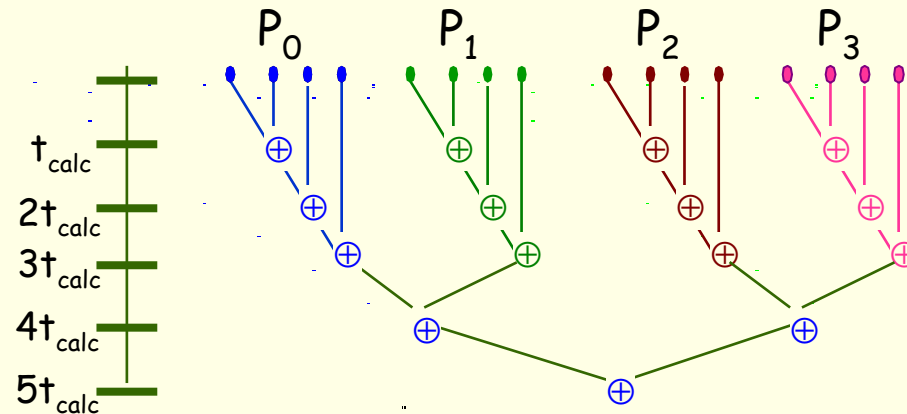
- i processori sono **omogenei**
- hanno lo **stesso carico** di lavoro



In un ambiente per il **Calcolo Distribuito**
tali ipotesi **non possono essere fatte**

Riprendiamo l'esempio della somma

Esempio: $P=4$



Assumendo $t_{\text{comm}}=0$
per un **calcolatore parallelo**
si ha $T_4=5 t_{\text{calc}}$

MA

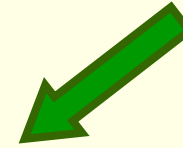
Cio' vale **solo** se si assume che il **tempo per una somma e' lo stesso per tutti i processori!!!!**

In un ambiente per il C.D.

Processori
eterogenei



Risorse non
dedicate

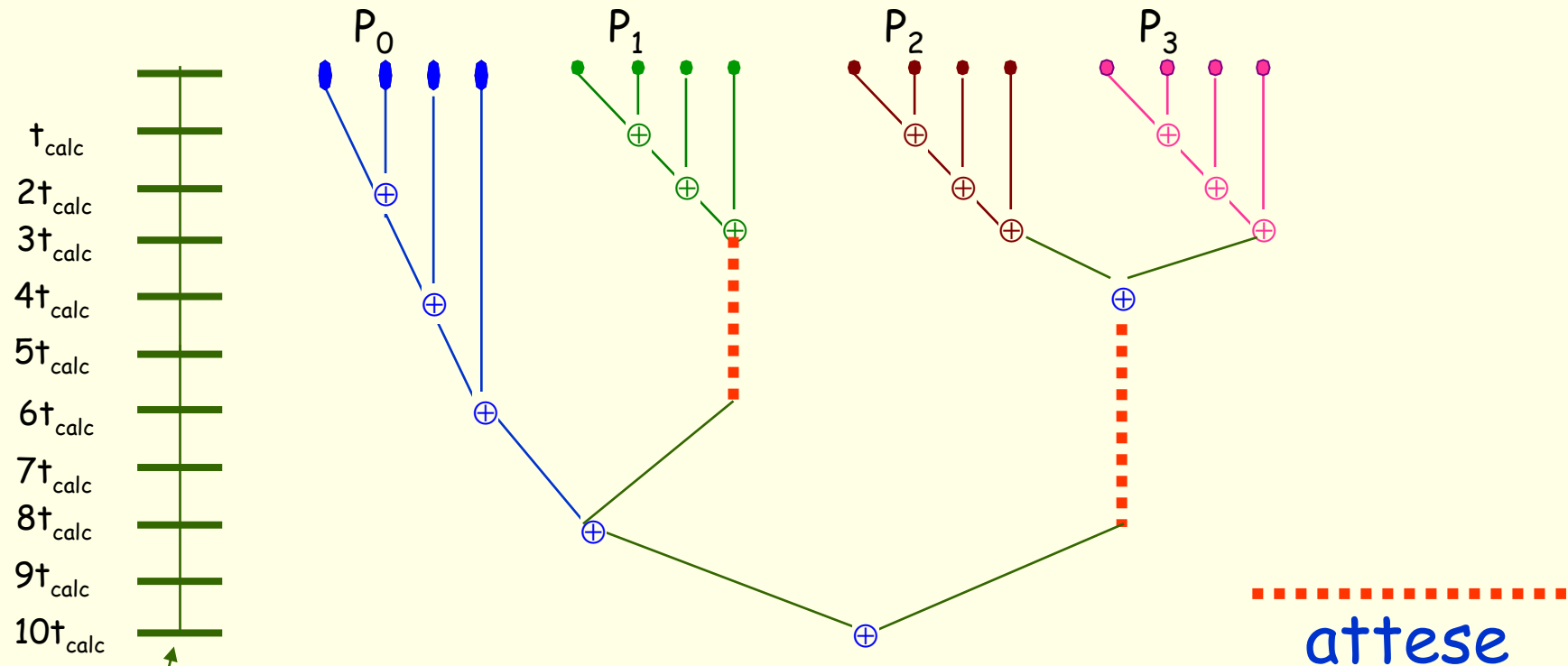


t_{calc} non e' lo stesso in tutti i processori

Assumiamo per esempio

$t_{\text{calc}}^{(0)} \sim 2 t_{\text{calc}}^{(i)}$ (tempo per la somma su P_0 circa
doppio rispetto agli altri)

Esecuzione in ambiente pr C.D.



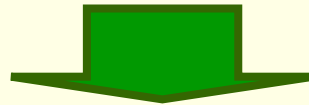
$$T_4 = 10t_{\text{calc}}$$



L'intera esecuzione e' rallentata
dal processore piu' lento
 $S_4 = T_1/T_4 = 15/10 = 1.5$

In definitiva

Le **comunicazioni** sono
punti di **sincronizzazione**
negli algoritmi



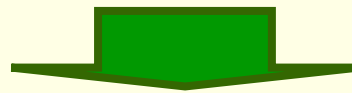
In un **ambiente distribuito**
(dove le **risorse non sono dedicate**)
la **presenza di comunicazioni** tra i
processori rendono **molto bassa**
l'efficienza anche se si assume

$$t_{\text{comm}} = 0 \quad !!!$$

Quindi ...



Un'applicazione predisposta al
Calcolo Distribuito e'
un'applicazione in cui e'
completamente assente
la comunicazione tra i processori



Aggregazione efficiente di risorse
geograficamente e amministrativamente
distribuite

Domanda:

Calcolatore parallelo

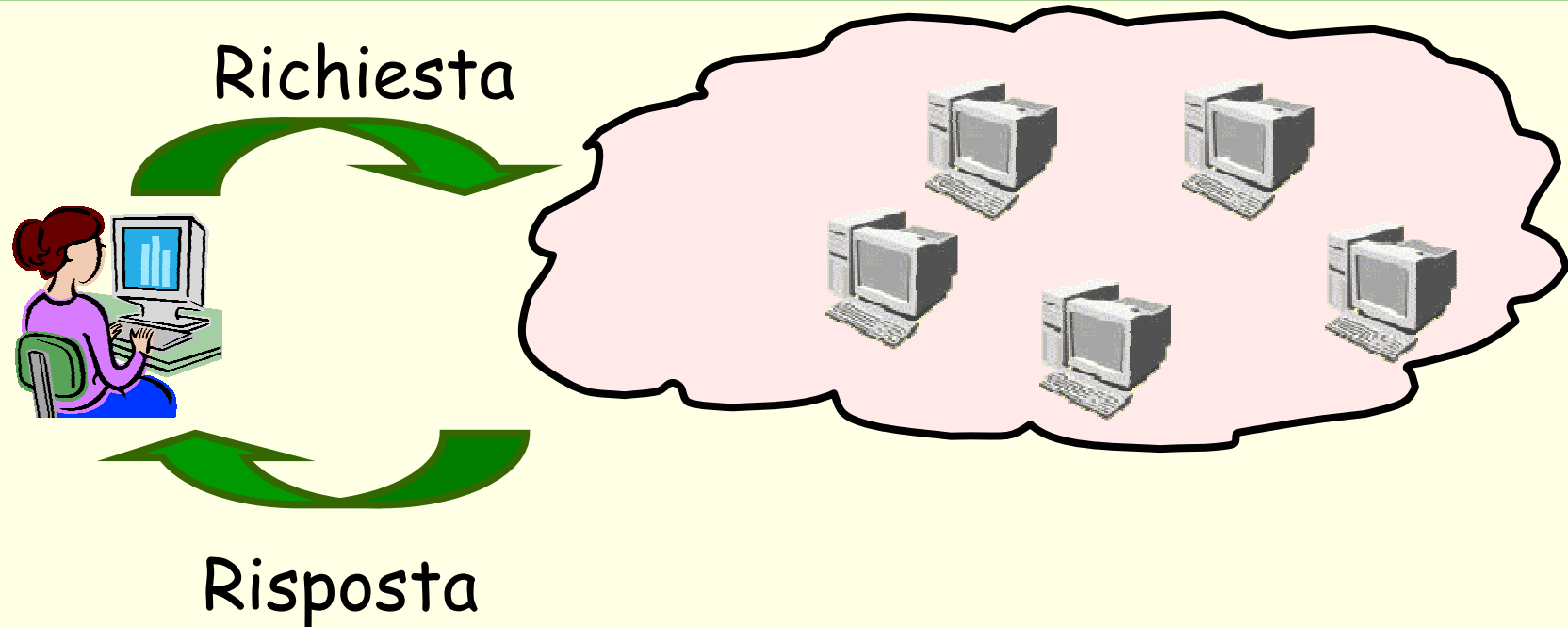
- risorse omogenee e dedicate
- reti di connessione veloci / memorie condivise



Sviluppo di applicazioni mediante
Message passing

Quale modello di programmazione
usare per sviluppare **applicazioni distribuite?**

Modello di programmazione per il C.D.

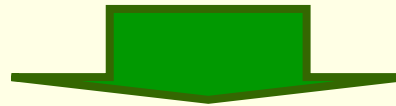


In un ambiente per il calcolo distribuito
l'utente dialoga con un ambiente software
richiedendogli un servizio
(modello client-server)

Modello client - server

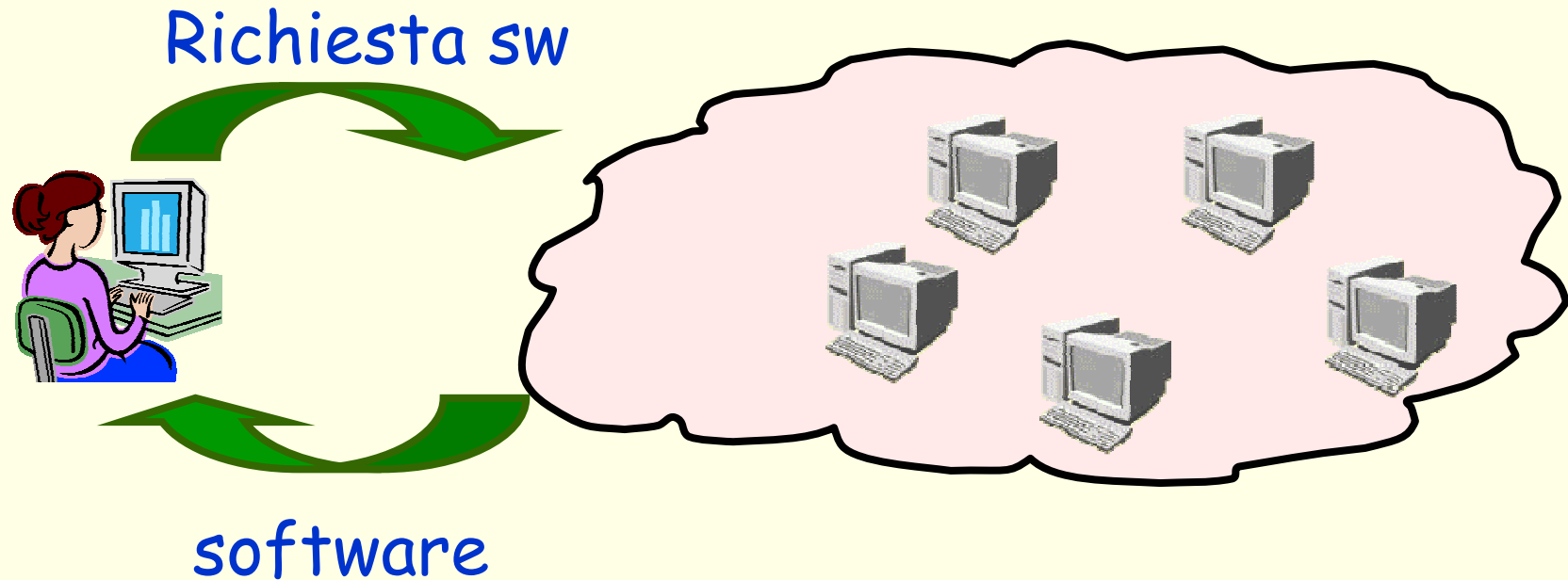
L'ambiente software puo' fornire:

- servizi software (server software)
- servizi hardware (server hardware)
- entrambi i servizi hw e sw



All'interno del modello client-server
si possono distinguere **differenti**
modalita' di esecuzione

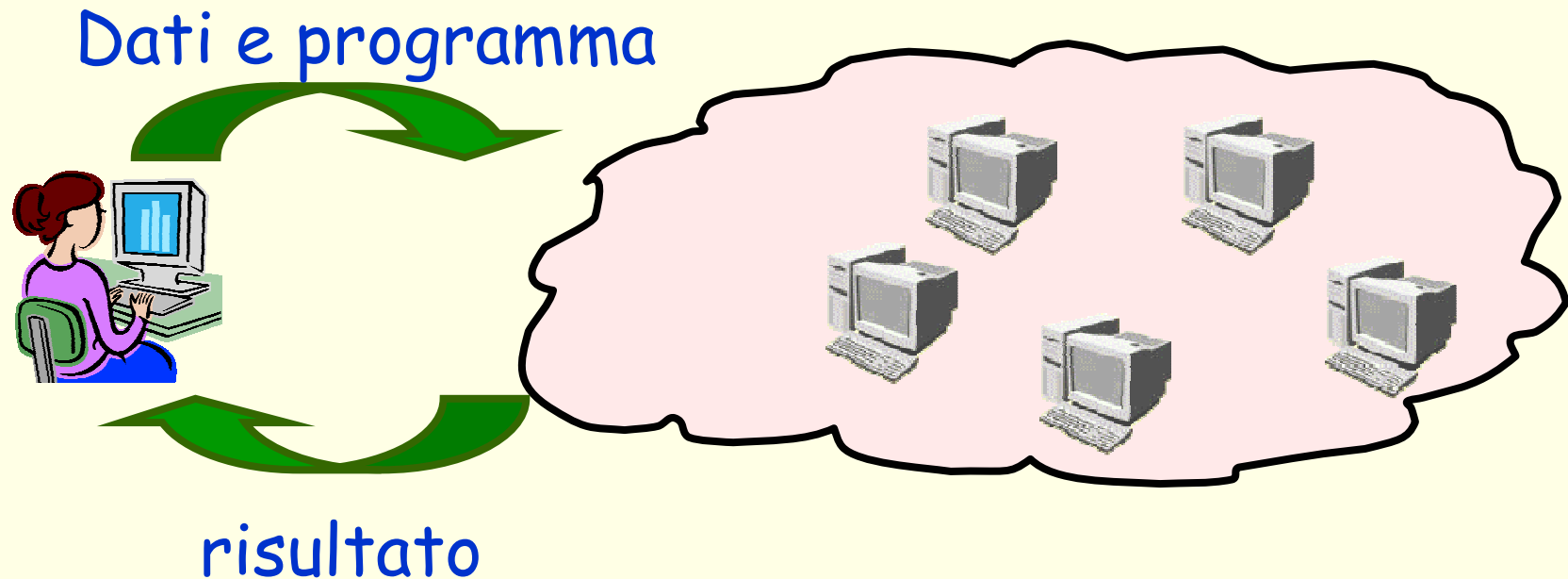
Esempio 1: server software



- il client richiede al server un programma
- il server fornisce al client il software richiesto
- il client esegue il calcolo su dati in suo possesso

(Code shipping)

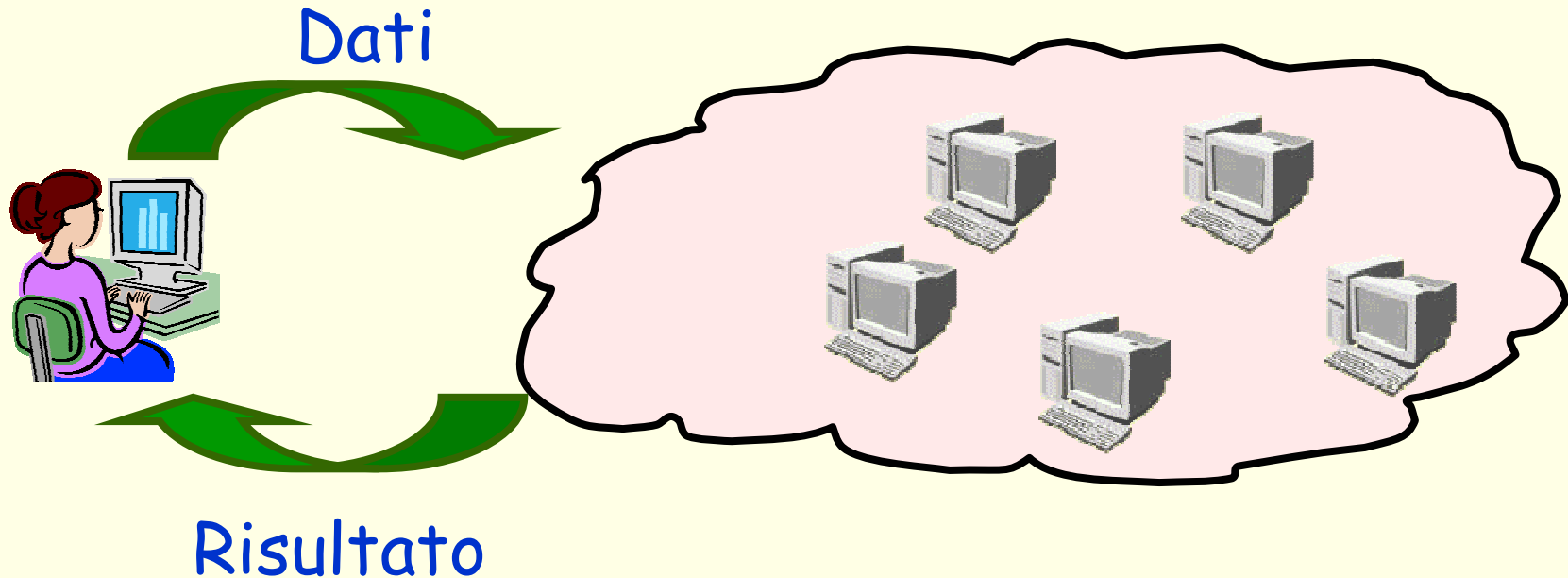
Esempio 2: server hardware



- il client fornisce al server dati e programma
- il server esegue il calcolo
- il server ritorna il risultato al client

(Proxy computing)

Esempio 3: server sw e hw



- il client manda al server i dati
- il server elabora i dati localmente
- il server manda il risultato al client

(remote computing)