

Universitatea de Vest din Timisoara, Facultatea de Matematica si Informatica

ARHITECTURA CALCULATOARELOR

Informatica , an I, 2021-2022

Dr. Maftciu-Scai Liviu Octavian

Curs 5

STRUCTURA CALCULATORULUI PARALEL

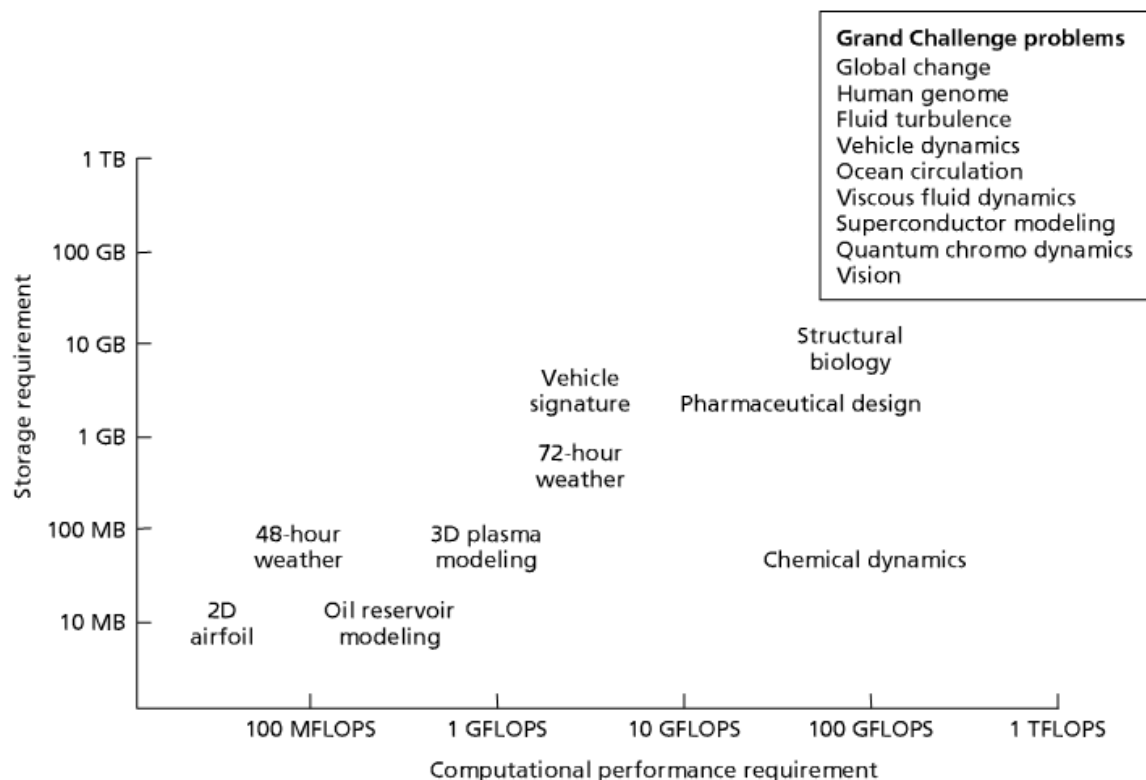
Definire

Calculator paralel = o multime de elemente de prelucrare (oarecum tot calculatoare, carora le lipseste clasicul I/O) care comunica intre ele in scopul rezolvarii mai rapide decat cu un calculator secvential, a unor probleme cu grad mare de complexitate si/sau de dimensiuni mari.

Exemple in care calculatorul paralel este necesar/indispensabil:

- Modelarea mediului, **prognoza meteo**
- Explorarea spatiului
- **Cercetare fundamentala si simulare in fizica, chimie, biologie**
-
- *sa invinga un campion mondial de sah neinvins (1999: IBM Deep Blue vs. Kasparov)*

Probleme pe care (doar) calculatorul paralel le rezolva in timp util:



FLOPS (floating point operations per second)

- NU e unitate in SI dar poate fi aprox. cu 1/s

- Oper. in virgula mobila consuma mai multe resurse/ timp decat cele in virgula fixa

- **ENIAC 1947: 500FLOPS vs Sunway TaihuLight 2016: 93PFlops**

Probleme pe care (doar) calculatorul paralel le rezolva in timp util:

Realizarea de filme “digitale”

Randarea (Rendering) este pasul in care informatia din fisierele animatiei, precum iluminarea, texture, umbrirea sunt aplicate la modele 3D pentru a genera imagini 2D care constituie cadre ale filmului. „Calculul paralel (in paralel) este esential pentru a genera numarul necesar/minim de cadre (24 fps) pentru filme de mediu sau lung metraj.

1995 - Pixar: Toy Story: complet generat pe calculator: 100 de masini de calcul fiecare cu procesor dual. [?]

1999 – Pixar: Toy Story 2: un sistem cu 1400 de procesoare – rezultatul=>detalii mult imbunatatite in ceea ce priveste texturile, imbracamintea, efectele atmosferice. [?]

2001: Monsters Inc. : un sistem cu 250 servere fiecare cu 14 procesoare

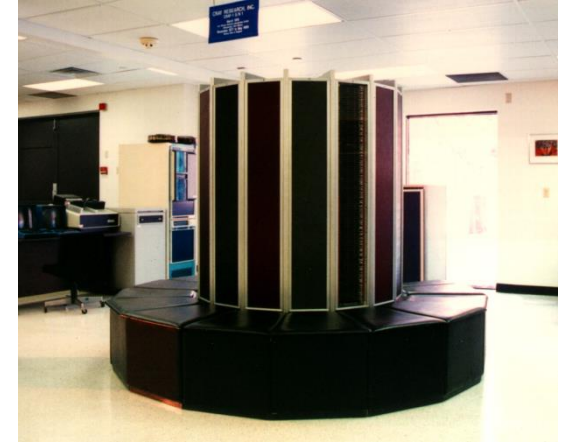
Si bineinteles multe alte prelucrari grafice complexe si complicate

Cum arata un calculator paralel

1970 Cray I

Los Alamos National Laboratory) US, New Mexico →

.....



2007 IBM Blue Gene /P

(in 2011, one “came” to the West University of Timisoara)



.....

→
2016 Sunway TaihuLight, China



Alte sisteme capabile de calcul paralel



Creierul uman: 100 miliarde neuroni, 150,000-180,000 km de fibre acoperite cu mielină, 15×10^{13} sinapse, 20W



CPU (Intel Xeon E7):
10 miezuri, 2.4 GHz,
130W, \$5000



GPU (NVIDIA Titan):
2688 miezuri, 6 GB
RAM, 1.5 Tflops,
250W, \$1000



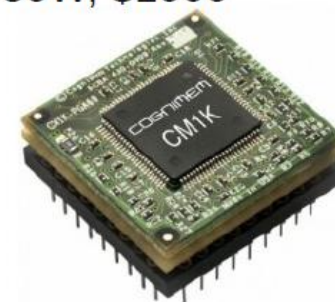
Coprocresoare CPU
(Intel Xeon Phi 7100):
61 miezuri, 1.2 GHz, 16
GB RAM, 1.2 Tflops,
300W, \$4000



Supercomputer Tianhe-2: 3,120,000 miezuri,
1375 TiB RAM, consum
33.8 Pflops, 17.6 MW,
\$390 mil.



FPGA (Field
Programmable Gate
Array): circuite logice
reconfigurabile, cai de
procesare paralele



Retea neuronală
implementată
hardware
(CogniMem CM1K):
4X1024 neuroni
artificiali, ~0.15
TFlops, \$3000

Legea lui Moore (Gordon E. Moore in 1965) (Cofondator Intel Co.)

Cresterea performantei/vitezei microprocesorului cu un factor de 2 la fiecare 18 luni (sau aprox. 60% pe an). (determinata empiric in 1965)

Cresterea performantelor calculatorului se datoreaza in principal:

- cresterii complexitatii si performantelor cipurilor VLSI, adica a CPU
- aparitiei si cresterii performantelor altor componente hardware si software: memorii cache pe cip, buffere mari pentru instructiuni, mai multe instructiuni per ciclu, multithreading, executia de instructiuni in avans, etc
- Legea a fost extrapolata pentru a afirma ca, cantitatea de putere de calcul la un cost dat se dubleaza la aproximativ fiecare 18 luni.
- Limitele fizice sunt impuse de viteza de propagare a semnalelor pe un cablu, argument cunoscut ca si **argumentul vitezei luminii (sau limita)**
- Pentru a motiva calculul paralel, acest argument este cunoscut ca si **argumentul puterii de calcul**

Argumentul vitezei luminii

Viteza luminii este de **299.792.458 m/s** adica aprox. 30 cm/ns.

Semnalele electrice circula printr-un cablu la o fractiune din vitezei luminii.

Considerand diametrul unui cip/procesor de 3 cm, un calcul care implica transmiterea de semnale intre extremitatile cipului nu poate fi executat mai rapid decat 10^{-3} secunde.

Daca reducem distantele cu un factor de 10 sau 10^2 (datorita limitarilor fizice in realizarea circuitelor integrate) se constata o limitare superioara si a vitezei de transmitere a semnalelor , adica se pare ca nu putem depasii 10^{12} calcule pe secunda.

Se poate continua cu rationamentul daca estimam distanta pe care trebuie sa o parcurga semnalele in cadrul unui ciclu de instructiune si ajungem la o estimare a valorii limita in MIPS sau FLOPS

Se va constata ca in prezent suntem destul de aproape de limita superioara a performantelor unui calculator, aceasta limita fiind impusa de viteza de propagare a semnalelor electrice precum si de alte limite fizice.

Consecinta

Argumentul vitezei luminii sugereaza ca odata ce limitele mentionate anterior au fost atinse, **singura modalitate** de a creste performantele de calcul consta in utilizarea simultana a mai multor procesoare, adica **calculatoare paralele**.

E drept, acelasi argument poate fi invocat si pentru a determina o limita superioara a comunicarii procesoarelor intr-un calculator paralel, dar deoarece comunicatiile nu sunt atat de frecvente ca si calculele, aceasta limitare nu ar trebui sa ne ingrijoreze in viitorul apropiat.

Scurta istorie a calculatorului paralel

1960 **ILLIAC IV**, Universitatea din Illinois, US

- Primul calculator paralel de dimensiuni considerabile;
- Arhitectura de interconectare tip grila 2D
- A fost proiectat pentru 256 procesoare (4 cadre cu 64 de procesoare fiecare).
- A fost construit numai un cadru de 64-procesoare
- A demonstrat clar fezabilitatea calculatoarelor paralele
- Dar, a aratat si dificultatile in programarea si utilizarea calculatoarelor paralele



Scurta istorie a calculatorului paralel

Calculul paralel a fost resuscitat in anii '80:

1990: o alta recesiune datorata:

- terminarii razboiului rece (fonduri guvernamentale mai mici)
- dezvoltarea rapida a procesoarelor din punct de vedere al raportului performanta/cost si drept urmare calculatoarele paralele au ramas in urma in ceea ce priveste acest raport.

⇒ multe companii noi ce activau in domeniul calculatoarelor paralele au dat faliment sau si-au schimbat activitatea.

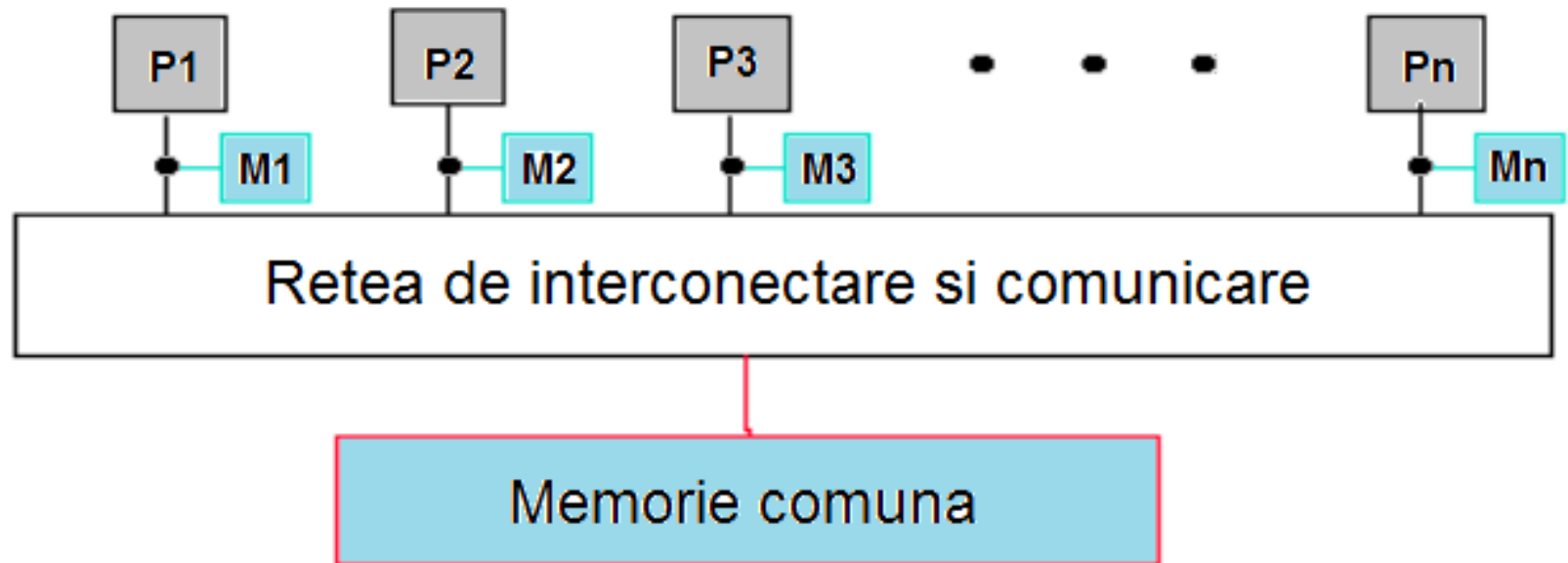
1993: listaTop500

1994: SGI, Digital & Sun au inceput vanzarea modelelor cu multiprocesoare simetrice (SMP) in familia statiilor de lucru pentru clienti industriali (ex. IBM SP2)

In prezent: 

Dar, portarea unui cod spre o arhitectura paralela e mai mult decat a pune un cod existent pe un calculator nou. Masinile paralele sunt fundamental diferite de celelalte, ceea ce impune reformularea codului de baza si a structurilor de date, precum si o regandire a reprezentarilor pentru procese si dinamica acestora.

Arhitectura generala a unui calculator paralel

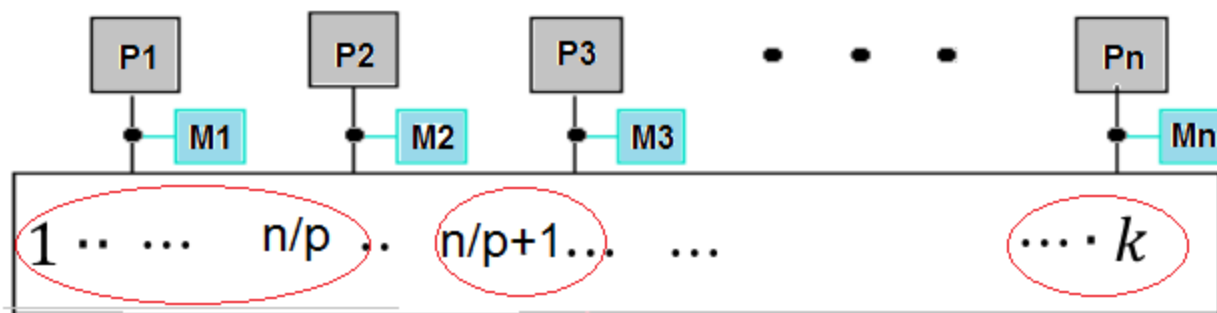


Exemplu de paralelizare a unui calcul

Calculul factorialului: $k! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot k$

Presupunem ca n (nr. procesoare) divide k

Vom avea n taskuri, adica fiecarui procesor ii revine k/n nr. de inmultit, adica n produse pariale



Avem doua seturi de date:

- partajate: valorile i si *produsul final*;
- private: produsele pariale si rezultatele pariale (individuale) ale fiecarui procesor in parte

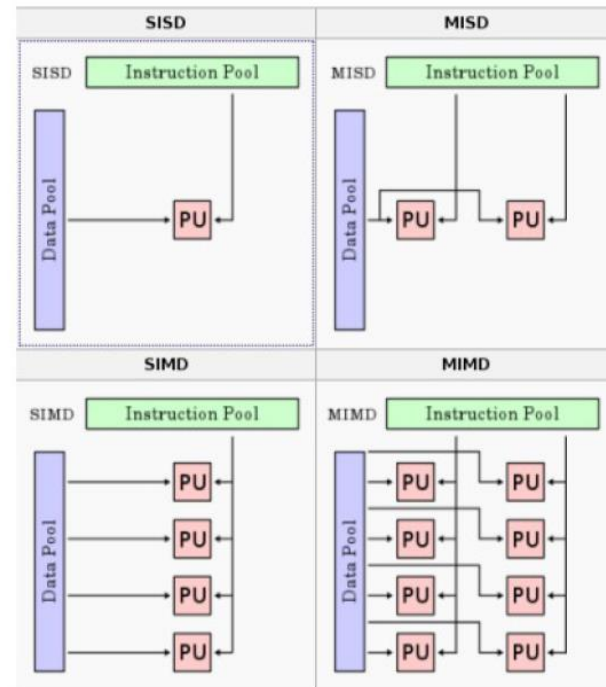
Clasificarea Flynnnschen:

- **SISD**: flux de instrucțiuni singular, flux de date singular (single instruction stream, single data stream)(O singura instructiune ,O singura Data)

Prelucrare
secventiala

- **MISD** : fluxuri de instrucțiuni multiple, flux de date singular

Paralelism
redutant



- **SIMD**: flux de instrucțiuni singular, fluxuri de date multiple
- **MIMD**: fluxuri multiple de instrucțiuni, fluxuri multiple de date

Paralelism

O clasificare a calculatoarelor paralele

-SIMD (Single Instruction Multiple Data):

- Procesoare matriciale;
- Procesoare vectoriale;
- Procesoare sistolice VLSI.

-MIMD (Multiple Instruction Multiple Data):

- Multiprocesoare (MIMD cu memorie partajată):
 - UMA (Uniform Memory Access):
 - UMA cu magistrale;
 - UMA cu comutatoare.
 - NUMA (Non-Uniform Memory Access):
 - CC-NUMA (Cache Coherent NUMA);
 - NC-NUMA (Non Coherent NUMA).
 - COMA (Cache Only Memory Access).
- Multicalculatoare (MIMD cu transfer de mesaje):
 - MPP (Massively Parallel Processors):
 - MPP cu grilă;
 - MPP hipercub.
 - COW (Cluster Of Workstations).

Clasificarea calculatoarelor paralele dupa Taxonomia lui Wang

Organizare matricială a datelor, adica o matrice de dimensiunea $m \times n$ are m cuvinte, fiecare cuvânt avand n biți. Clasificarea se face dupa gradul de paralelism în procesarea datelor:

- **WSBS** (Word Serial-Bit Serial): se lucrează pe un singur cuvânt, fiecare cuvânt fiind prelucrat serial bit cu bit
- **WSBP** (Word Serial-Bit Paralel): se lucrează pe un singur cuvânt , biții fiecărui cuvânt fiind prelucrați simultan.
- **WPBS** (Word Paralel-Bit Serial): se lucrează pe un singur bit la toate cuvintele simultan
- **WPBP** (Word Paralel-Bit Paralel): se lucrează simultan pe toate cuvintele și pe toți biții

Note:

- WSBS nu reprezinta o prelucrare paralela.
- WSPB și WPBS sunt parțial paralele.
- WPBP este complet paralela.

? Fot fi asimilate cele doua clasificari?

Arhitectura hardware a calculatoarelor paralele

Nota: In prezent, toți marii producători comercializează procesoare cu mai multe unități eficiente de calcul (cores) împachetate într-un singur chip -> practic și un PC sau un smartphone sunt mini-sisteme paralele. Totuși, în cele ce urmează, **nu** ne vom referi la această ultimă categorie ci la sisteme hardware complexe cu mult mai multe procesoare și evident mult mai multă putere de calcul.

HPC – High Performance Computing = soluții tehnologice care permit integrarea unui număr foarte mare (>100 K) de unități de calcul (procesoare + memorii +...)

Domeniul este susținut și de dezvoltarea tehnologiilor IT de comunicare (Ethernet gigabyte, Infiniband, comunicații pe fibră optică) care permite un transfer eficient de date între unități de calcul distincte.

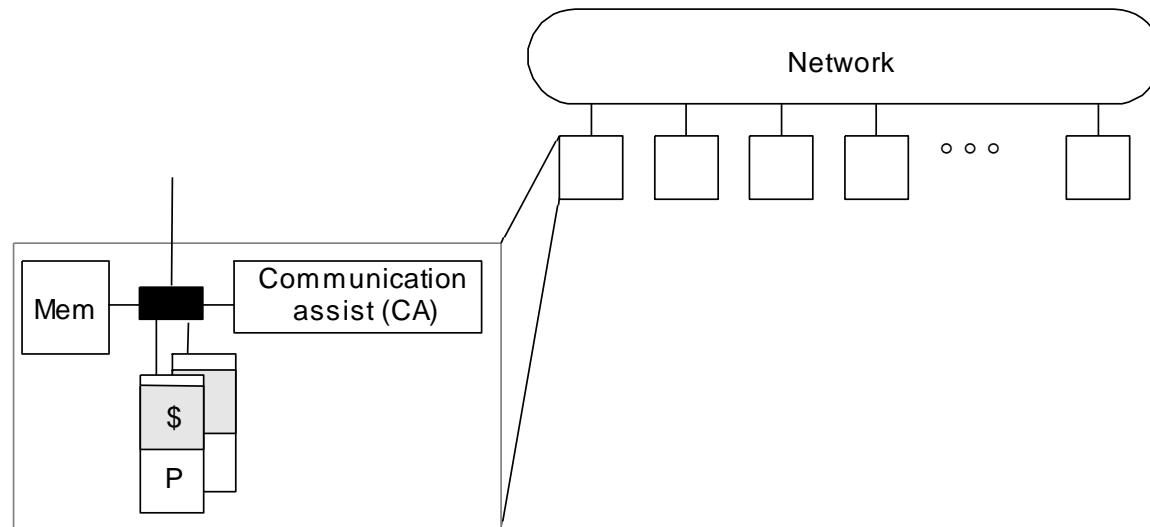
Supercomputer = mai multe procesoare (foarte multe) integrate într-un sistem/computer complex, care folosesc o memorie comună/centrală și care funcționează concomitent și coordonat în scopul atingerii unor performanțe globale sporite.

Modul de calcul al unui supercomputer se numește "**calcul paralel**".

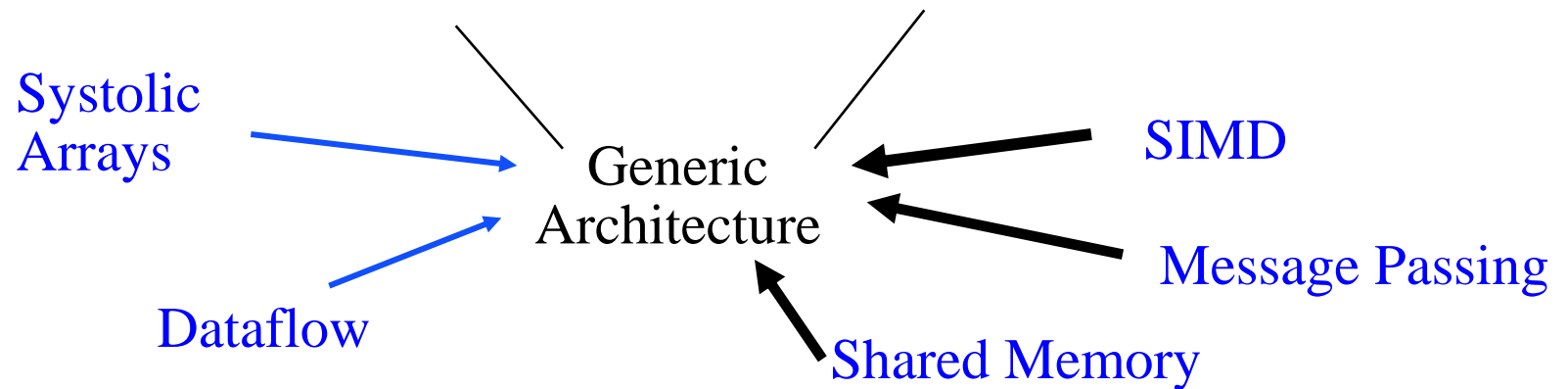
Arhitectura hardware a calculatoarelor paralele

Arhitectura hardware = F(destinatie)

Forma generala a arhitecturii unui calculator paralel, din care deriva sau spre care converg cazurile particulare/concrete de sisteme paralele:

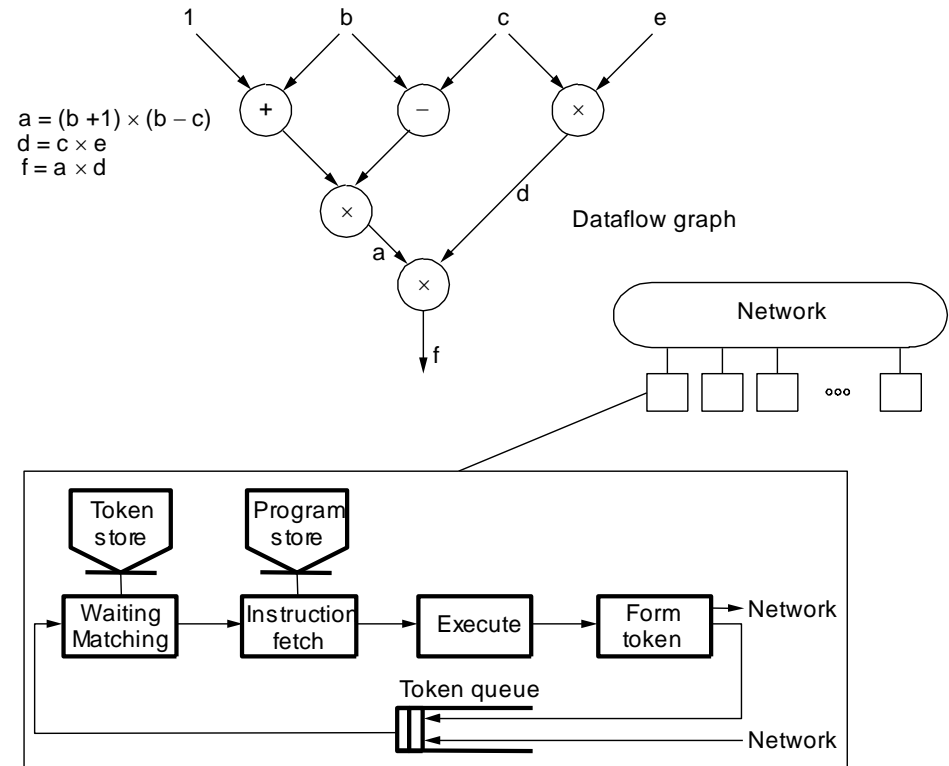


Arhitectura hardware a calculatoarelor paralele – tipuri de arhitecturi



Arhitectura hardware a calculatoarelor paralele

Arhitectura Dataflow

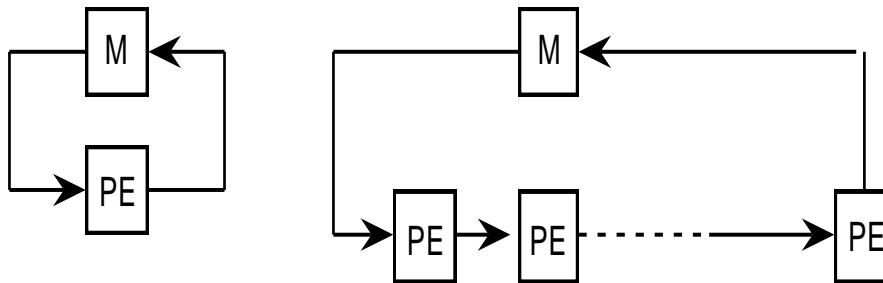


- Reprezentare sub forma de graf
- Procesorul logic din fiecare nod este activat de disponibilitatea operanzilor.
- Mesajul care transporta si instructiunea urmatoare este transmis urmatorului procesor

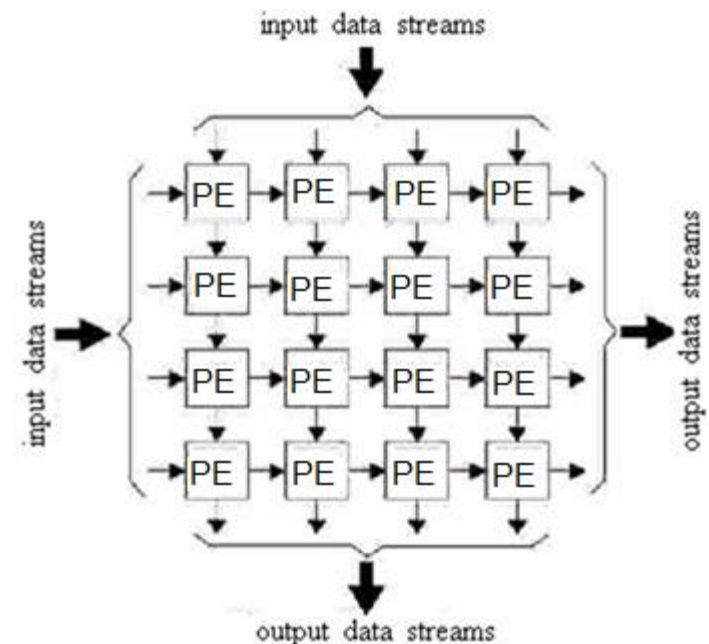
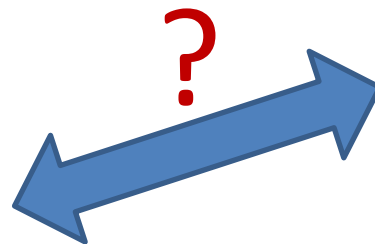
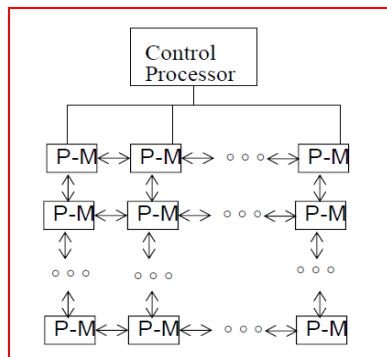
Arhitectura hardware a calculatoarelor paralele

Arhitecturi Systolice

- VLSI a permis realizarea de cip-uri dedicate ieftine
- Ideea de baza: reprezentarea algoritmilor direct prin cip-uri conectate in pattern-uri regulate. (vector, matrice, tor...) adica inlocuirea unui procesor cu o arie regulata/uniforma de elemente de procesare.
- Avantajul major: acces putin la memorie, deoarece datele se transfera de la procesor la procesor prin intermediul registrilor proprii



arhitectura SIMD



Arhitectura hardware a calculatoarelor paralele – alte clasificari-

- 1) Sincron/Asincron:** refera modul in care snt executate instructiunile pe un microprocesor in functie de gradul de executie al instructiunilor pe celelalte procesoare.
- 2) Mod comunicare informatie:** prin partajarea unui spatiu de memorie sau prin transmiterea de mesaje (ex. MPI, OpenMP)

Arhitectura hardware a calculatoarelor paralele – nivele de paralelism -

- 1) Paralelism la nivel de instructiune:** instructiunile multiple sunt executate in paralel daca sunt independente;
- 2) Paralelism la nivel de date:** datele sunt distribuite uniform intre procesoare; - in special folosit in domeniul calculelor stiintifice (rez. sist.ec.mari-metode Newton)
- 3) Loop paralelism:** - daca instructiunile dintr-o bucla for nu depind una de cealalta, se pot executa in paralel; - sequential loop → parallel loop;
- 4) Paralelism functional:** - multe programe secventiale contin parti de cod care sunt independente si pot fi rulate paralel (ex. : bucle, apeluri de functii);

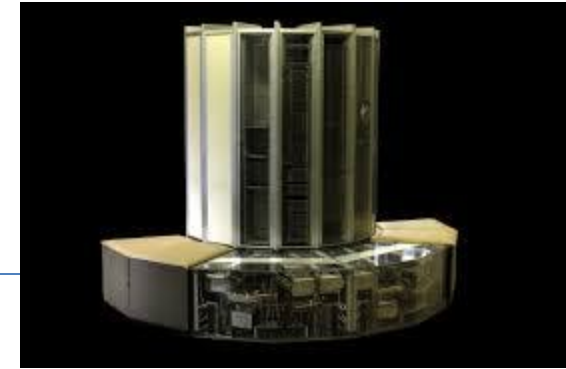
Arhitectura hardware a calculatoarelor paralele -exemple-

Cray I, 1975 (Los Alamos National Laboratory, US)

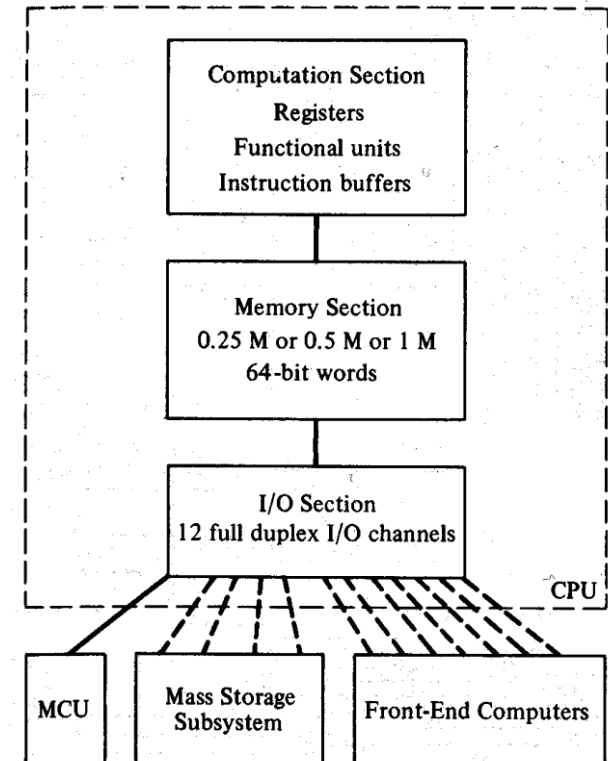
Primul supercomputer rapid: 80 Mhz clock speed

138 Mflops/s

64-bit word size, Integer and floating-point arithmetic, 12 functional units (pipelined processors), 128 basic instructions

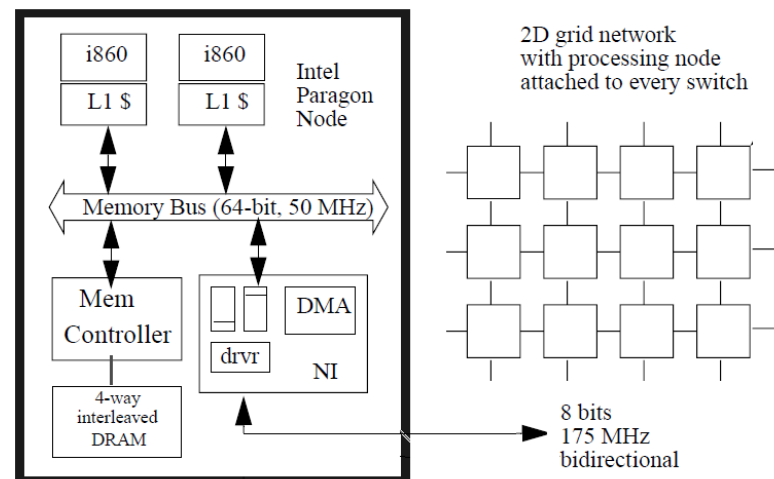


Racirea: densitate mare de componente=> multa caldura disipata. Astfel, pt. 1024K, memorie se disipa 115 KW caldura.
Mentenananta: in medie aprox. 50 ore functionare continua



Arhitectura hardware a calculatoarelor paralele -exemple-

Intel Paragon, 1990



PROCESOR DE BAZA: Intel 860 40Mhz, 32/64 bits, 32-bit ALU si 64-bit FPU

INTEL PARAGON SUPERCOMPUTER"

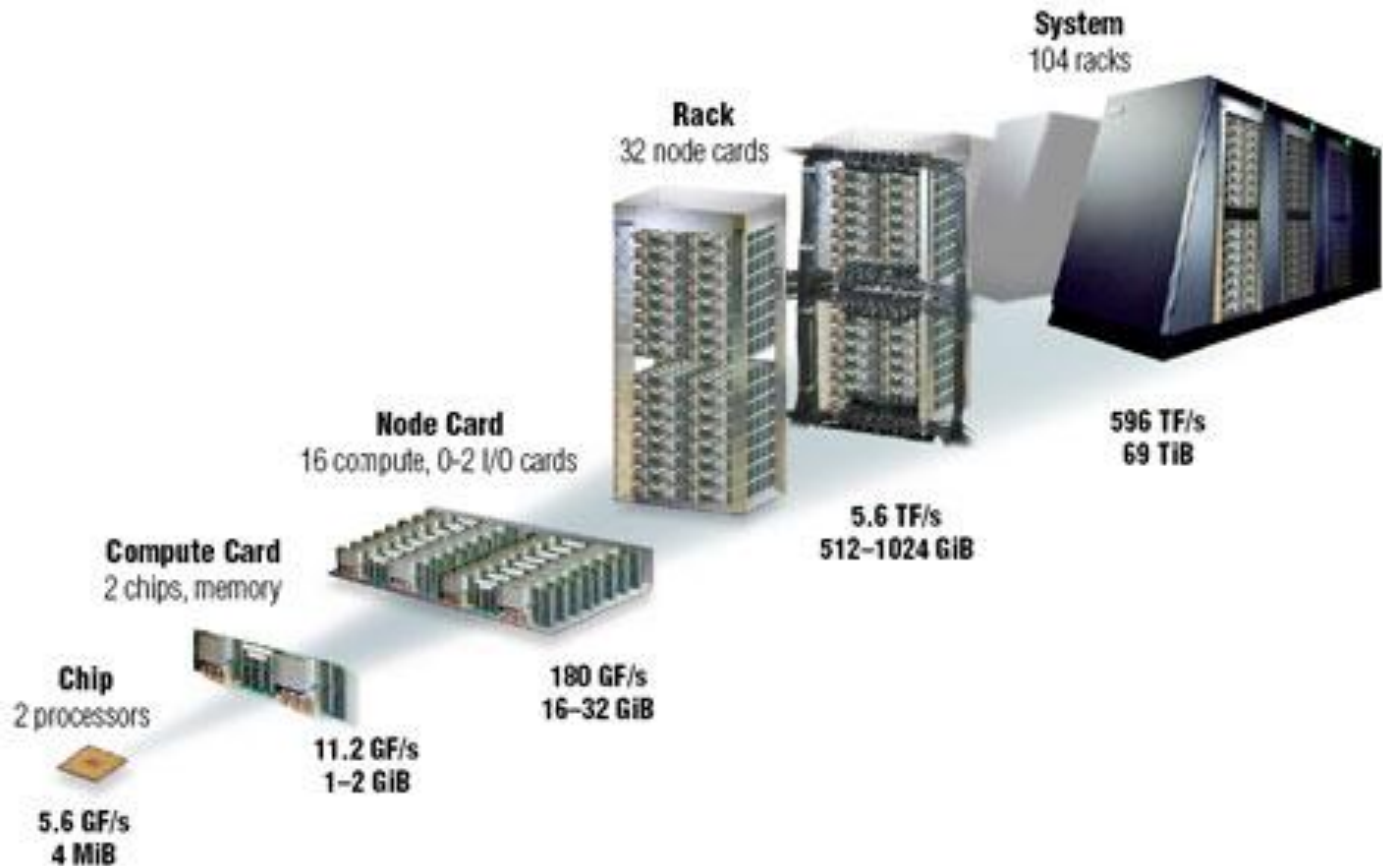
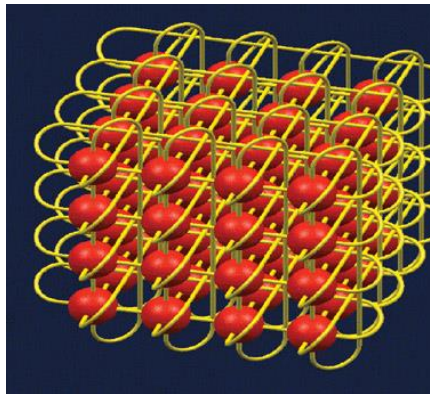
- 1824 noduri (max 4000), configurate ca si o matrice de 16x114,
- 128 Gb memorie principala
- comunicare: 200Mb/s
- 300 Gflops/s/procesor
- sistem operare: OSF/1, SunMos
- compilatoare: Fortran 77, ADA

Arhitectura hardware a calculatoarelor paralele -exemple-

IBM Blue Gene /L

2004, primul model construit special pentru cercetari in biologie
131000 procesoare (dual cores), 69 Tbytes memorie, 478 TFlops/s

32 x 32 x 64 3D torus:

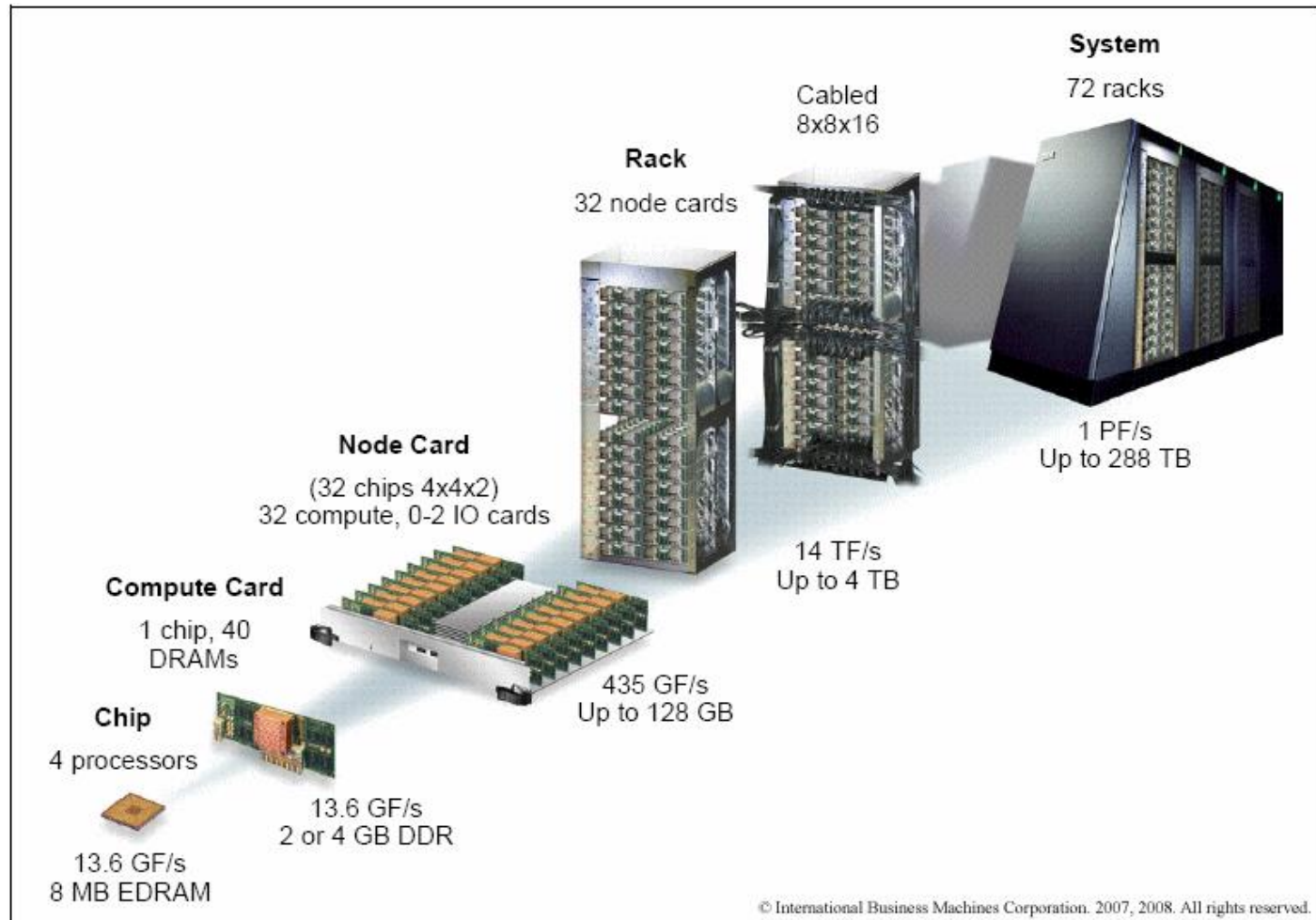


Arhitectura hardware a calculatoarelor paralele -exemple-

IBM Blue Gene /P

2007

163,840 procesoare (quad-cores) (PowerPC 450 4C la 850MHz), 557.056 TFlops/s

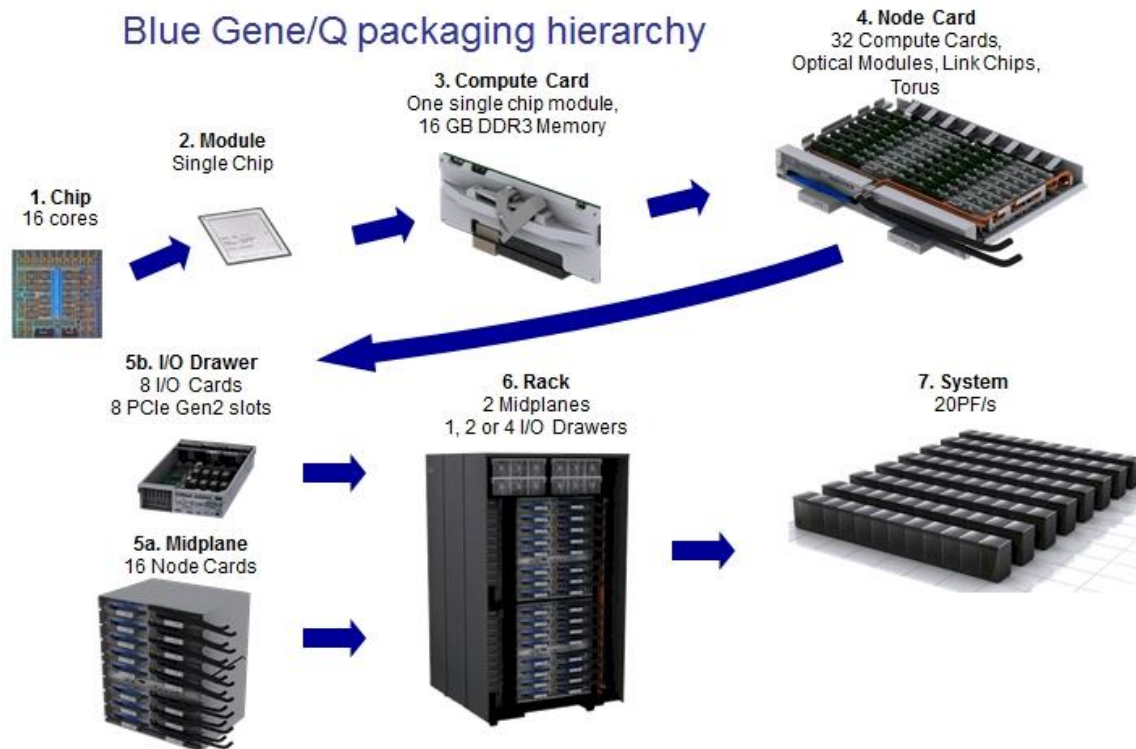


Arhitectura hardware a calculatoarelor paralele -exemple-

IBM Blue Gene /Q

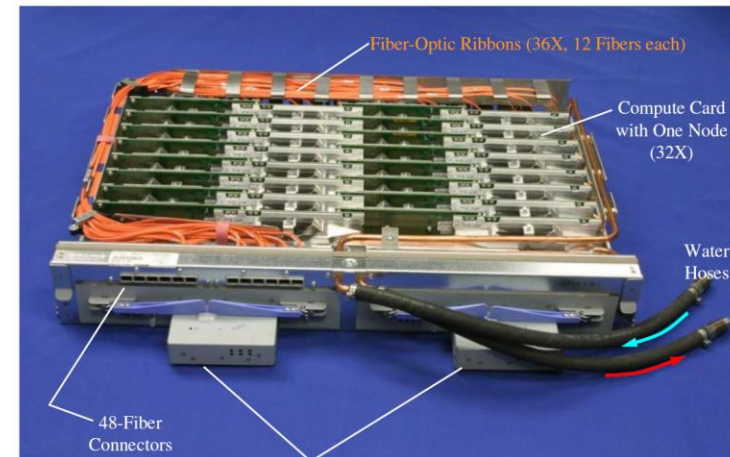
2011

Blue Gene/Q packaging hierarchy



Blue Gene/Q node board: 32 compute nodes

IBM



Arhitectura hardware a calculatoarelor paralele -exemple-

Sunway TaihuLight

2016, Top500

93 Pflops/s (Tiahne avea “doar” 34PF)

64 bit RISC procesor la 1,5 GHz, 16-core chip, 40960 noduri,

32 GBytes/nod memorie (DDR3) adica un total de 1,3 Pbytes

Consum energie 15,3 Mwatts (1 calorifer electric pe ulei aprox. 1,5Kw adica supercomputerul consuma cam cat 10000 de calorifere!)

Pret: 273 mil. \$ US



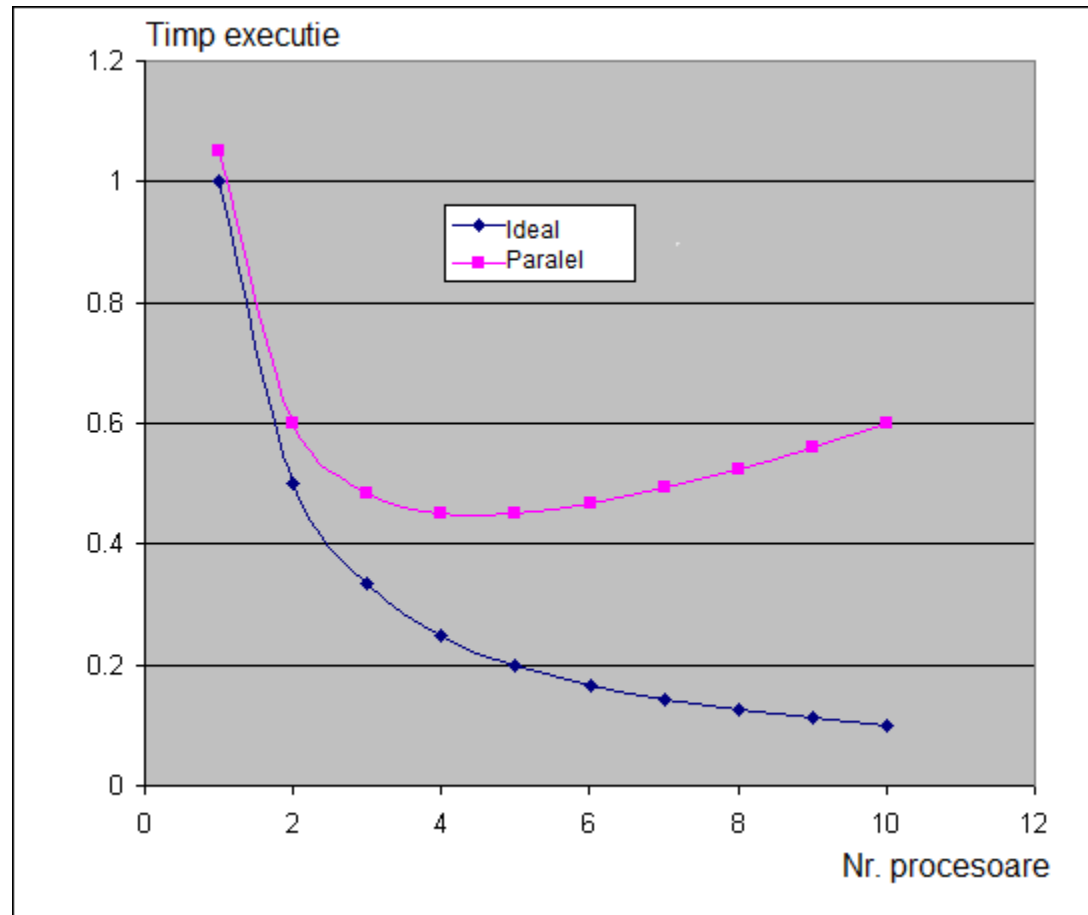
Sunway TaihuLight – arhitectura generala-



Cum stau lucrurile pe un calculator paralel?

Timpul de rulare pentru un program in paralel (mai multe procesoare) este mai mare decat timpul ideal \Leftrightarrow exista un punct (nr. procesoare) de la care chiar daca se adauga procesoare nu creste eficienta in timp, dimpotriva aceasta scade.

Comp. cu o echipa de oameni



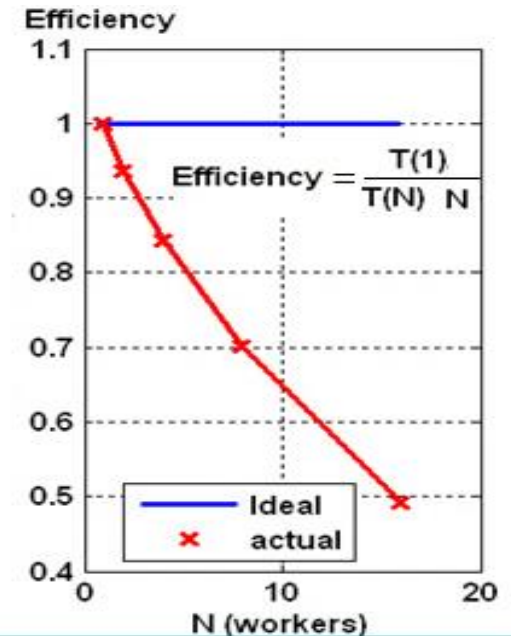
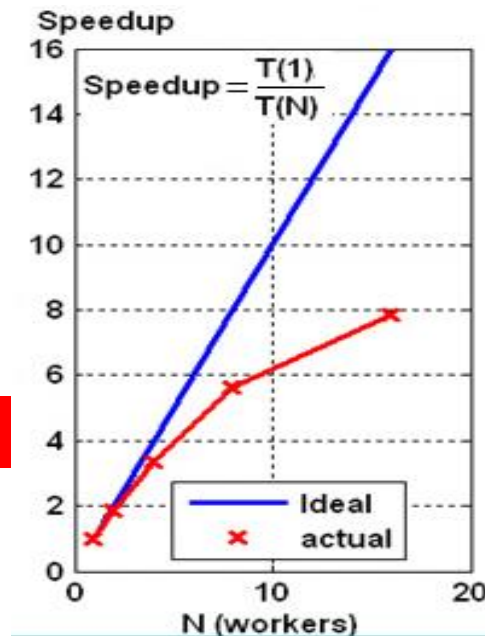
Metrici pentru performantele calculatoarelor paralele

Gradul de paralelism DOP („degree of parallelism”) = numarul de procesoare dintr-un calculator paralel utilizate intr-un anumit interval de timp.

Factorul de accelerare S (Speedup) = raportul dintre timpii necesari pentru rezolvarea aceleiași probleme pe un sistem monoprocesor/secvential și respectiv pe sistemul paralel.

- linear
- superlinear (superspeedup)
- sublinear

Serializare ? => RelativeSpeedup



Eficiența paralelizării E (Efficiency) = raportul dintre Speedup și numarul de procesoare

Este paralelizarea soluția ideală ?

Teoretic da, dacă programele ar putea fi paralelizate în mod optim ar apărea o creștere **liniară** a vitezei cu numărul de procesoare.

Dar foarte puține probleme permit o paralelizare optimă !

Legea lui Amdahl: creșterea performanței (raportul dintre timpul serial și cel paralel) folosind P procesoare este data de

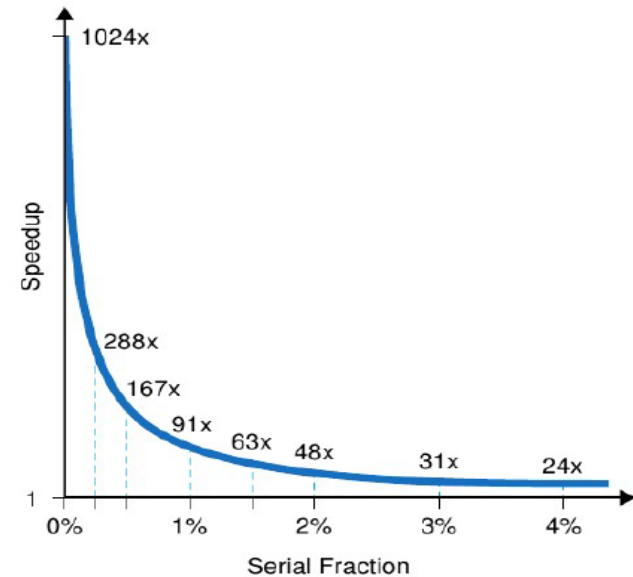
$$S(P) = \frac{1}{f + (1-f)/P} \quad S(P) = \frac{T_s}{T_p}$$

pentru o problema de dimensiune dată (unde f este fracția din program care trebuie efectuată serial).

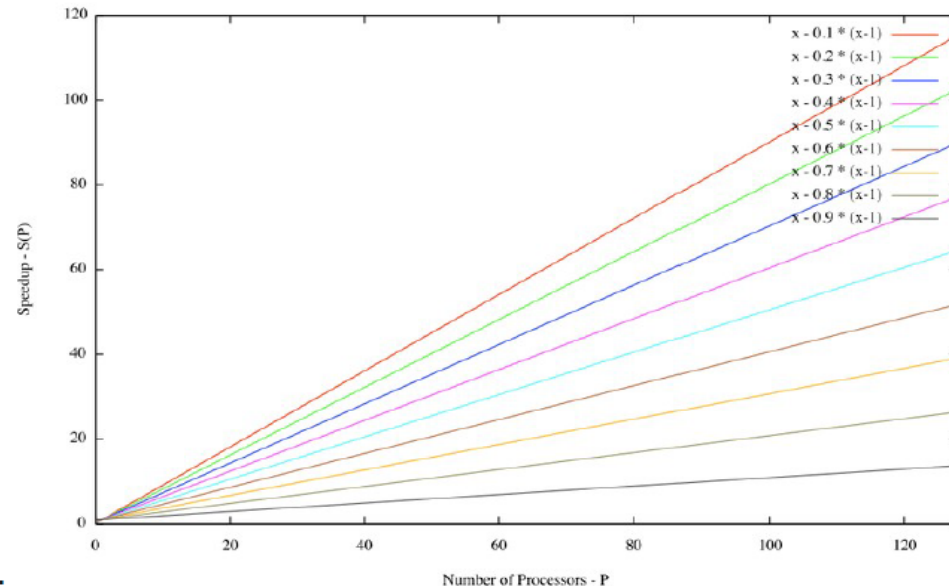
Legea lui Gustafson:

$$S(P) = \frac{a + Pb}{a + b}$$

Păstrarea timpului de calcul fix dar creșterea dimensiunii problemei abordate. Această abordare este de multe ori validă deoarece efortul paralel crește mai repede decât cel serial o dată cu creșterea dimensiunii problemei. P = nr. de procesoare, a = timpul serial, b = timpul paralel de pe fiecare procesor.



Gustafson's Law: $S(P) = P \cdot a^*(P-1)$



Metrice pentru performantele calculatoarelor paralele

Cateva observatii se impun pentru inceput:

1. Performanta unui algoritm in cazul unui computer serial (von Neumann) serial nu implica neaparat si performanta in cazul unui calculator paralel => modificarea/adaptarea algoritmului initial pentru cazul paralel. ! Nu intotdeauna (cel mai adesea) este posibila aceasta translatie asa ca trebuie proiectat un nou algoritm!
2. Ca si consecinta la pct. 1, cel mai adesea si programele trebuie rescrise tinand cont in principal de urmatoorii factori:
 - organizarea memoriei;
 - topologia de interconectare;
 - numar de procesoare disponibile;
3. Rescrierea programelor este adeseori necesara si cand se trece de la o arhitectura paralela la alta, factorii principali de influenta fiind aceiasi ca si pct. 2
4. Daca in cazul serial, timpul si memoria sunt criterii suficiente de evaluare a performantelor, in cazul paralel, mult mai multe metrice sunt necesare, ele avand uneori interpretari si valori diferite in functie de arhitectura paralela (adica nu se prea poate vorbi de performanta alg. paralel de inm. A 2 matrici in general, ci eventual doar de cea pe un sistem dat)

Metrici pentru performantele calculatoarelor paralele – **Speedup**-

In general reprezinta raportul dintre timpul de executie serial cu un singur procesor si cel paralel folosind n procesoare:

$$\frac{\text{serial execution time}}{\text{parallel execution time}}$$

Acest indicator nu este fix, el depinzand de numarul de procesoare folosite, dar nu numai. Exista totusi cateva abordari diferite in functie de cum anume este considerat timpul de executie serial:

1. *RelativeSpeedup*: timpul serial este timpul de executie pe acelasi computer (paralel) folosind un singur procesor, in cazul folosirii aceluiasi program Q si aceiasi instanta I :

$$RelativeSpeedup(I, P) = \frac{\text{time to solve } I \text{ using program } Q \text{ and 1 processor}}{\text{time to solve } I \text{ using program } Q \text{ and } P \text{ processors}}$$

2. *MaximumRelativeSpeedup*: speedup depinde si de configuratia si mai ales dimensiunea n a instantei (dimensiunea datelor de intrare) (de ex. la sortare):

$$MaximumRelativeSpeedup(n, P) = \max_{|I|=n} \{RelativeSpeedup(I, P)\}$$

Metrici pentru performantele calculatoarelor paralele – **Speedup**-

3. *RealSpeedup*: timpul paralel este comparat cu timpul obtinut de cel mai rapid algoritm serial folosit in practica:

$$RealSpeedup(I, P) = \frac{\text{time to solve } I \text{ using best serial program and 1 processor}}{\text{time to solve } I \text{ using program } Q \text{ and } P \text{ processors}}$$

4. *AbsoluteSpeedup*: timpul paralel este comparat cu timpul obtinut de cel mai rapid algoritm serial pe cel mai rapid calculator secvential:

$$AbsoluteSpeedup(I, P) = \frac{\text{time to solve } I \text{ using best serial program and 1 fastest processor}}{\text{time to solve } I \text{ using program } Q \text{ and } P \text{ processors}}$$

Definitia poate fi usor extinsa la *MaximumAbsoluteSpeedup*.

Metri pentru performantele calculatoarelor paralele –Efficiency-

! O masura “inrudita” cu Speedup ! adica

$$E=S/p$$

unde p este numarul de procesoare.

In fapt, masoara eficienta de utilizare a procesoarelor

Metri precum *RelativeEfficiency* sunt usor de dedus din *RelativeSpeedup* respectiv ceilalti membrii ai clasei de metri Speedup.

Metrici pentru performantele calculatoarelor paralele – Scalability-

Aceasta metrica reflecta capacitatea unui calculator paralel de a-și mări viteza totală de calcul odata cu cresterea numarului de procesoare. In practica, aceasta metrica nu atinge valoarea ideala, adica proportional cu numarul de procesoare implicate, ci scade odata cu cresterea numarului de procesoare de la un anumit numar al acestora, in principal datorita cresterii ponderii proceselor de intercomunicare.

Intuitiv, un sistem este scalabil, daca performantele cresc odata cu cresterea numarului de procesoare din componenta sa. In practica se folosesc: Assymptotic Scalability, Isoefficiency, Isospeed

Se mai folosesc deasemenea urmatoarele metrici:

1. **Latency**: timpul maxim necesar pentru ca un mesaj “mic” sa “calatoreasca” intre doua procesoare.
2. **Overhead**: cantitatea de timp consumata de un procesor pentru a receptiona sau transmite un mesaj, timp in care procesorul nu poate fi angajat in alta operatie.
3. **Gap**: timpul minim necesar unui procesor intre doua trimiteri/receptionari de mesaje.



Top 10 supercomputer list

Rank	System	Site (country)	RMAX (pflop/s)*	Top 10 countries	Supercomps In Top500
1	Sunway TaihuLight	China	93.0	China	167
2	Tianhe-2 (MilkyWay-2)	China	33.9	US	165
3	Titan	US	17.6	Japan	29
4	Sequoia - BlueGene/Q	US	17.2	Germany	26
5	K computer	Japan	10.5	France	18
6	Mira	US	8.6	UK	12
7	Trinity	US	8.1	India	9
8	Piz Daint	Switzerland	6.3	S Korea	7
9	Hazel Hen	Germany	5.6	Russia	7
10	Shaheen II	S Arabia	5.5	Poland	6

*RMAX score is used to rank supercomputers based on their performance using the LINPACK Benchmark. Peflop is a unit of computing speed equal to one thousand million floating-point operations per second

Source: www.top500.org AFP

Raspunsul Americii la *Sunway TaihuLight*

2018 BIG BLUE IBM promite 200 PFlops prin *IBM Summit*
(proiect demarat in 2014)



ANEXA 1

- **Lungimea cuvântului**

Calculatoarele lucrează cu ajutorul cuvintelor de cod a căror cantitate de informație este măsurată în biți. Numărul de biți reprezintă lungimea unui cuvânt și este multiplu de doi. Un cuvânt poate reprezenta: - o instrucțiune; - un segment de date. Într-un calculator, lungimea cuvântului se identifică cu numărul de biți ai instrucțiunii. Cele mai noi calculatoare au instrucțiuni pe 64 biți. Aceasta este o caracteristică principală a UC. Nu este obligatoriu ca lungimea cuvântului să fie aceeași cu dimensiunea magistralei de memorie pe care se aduc instrucțiunile din UM în UC. Un calculator poate avea, de exemplu, lungimea instrucțiunii de 64 biți și lățimea magistralei de 32 biți; pentru aducerea unei instrucțiuni din memorie sunt necesare, în acest caz, două apeluri la memorie.

ANEXA 2

- **Frecvența ceasului**

Orice calculator are un generator de impulsuri, numit ceasul unității centrale.. Acesta este realizat cu un cuarț care emite impulsuri cu frecvență fixă. Ceasul inițial suferă două tipuri de operații:

- operația de divizare a ceasului, ceea ce înseamnă că ceasul inițial, cu cuarț, suferă modificarea frecvenței sale;
- operația de amplificare a semnalului de ceas. Pe acest ceas, care este inima calculatorului, au loc toate evenimentele hard din UC.

ANEXA 3

- **Numărul de instrucțiuni executate în unitatea de timp**

Dacă notăm cu f frecvența ceasului (Hz), cu N numărul mediu de ceasuri în care se execută o instrucțiune

atunci $n=f/N$, n fiind numărul de instrucțiuni executate într-o secundă:

Exemplu: pentru un calculator cu frecvența ceasului de 2GHz și care execută două instrucțiuni pe ceas, numărul de instrucțiuni executate în unitatea de timp este 4 miliarde instrucțiuni pe secunda.

In prezent se folosește MIPS (milioane de instrucțiuni executate într-o secundă)