

Programação concorrente e Distribuída

Aula 1: Introdução

Prof: Álvaro L. Fazenda
[\(alvaro.fazenda@unifesp.br\)](mailto:alvaro.fazenda@unifesp.br)

5º Termo em Bac. Ciência da Computação

Bibliografia

- Ben-Ari, M. Principles of concurrent and distributed programming.
2nd Ed. Addison-Wesley, 2006.
- Andrews, G.R. Foundations of Multithreaded, Parallel, and
Distributed Programming, Addison-Wesley, 1999.
- Dongarra, J., Foster, I., Fox, G.C., Gropp, W., Kennedy, K., Torczon,
L., White, A. The Sourcebook of Parallel Computing. Morgan
Kaufmann, 2002
- Grama, A., Karypis, G., Kumar, V., Gupta, A. Introduction to
Parallel Computing. 2nd Ed., Addison Wesley, 2003
- Pacheco, P. An Introduction to Parallel Programming. Morgan
Kaufmann, 2011

Programas Concorrentes

- *Um programa concorrente é um conjunto de programas sequenciais (ou de processos) que podem ser executados em paralelo (M. Ben-Ari, "Principles of Concurrent and Distributed Programming")*
 - "podem ser executados em paralelo" não significa necessariamente que serão paralelos e sim que existe paralelismo em potencial

Concorrência vs Paralelismo

- **Paralelo**
 - Fisicamente ao mesmo tempo
- **Concorrente**
 - Logicamente ao mesmo tempo mas pode ser implementado sem dispositivo que permita paralelismo
 - Exemplo: SO multitarefas, aproveitamento de recurso ocioso de CPU, etc.
- O curso aborda concorrência como uma forma de obter paralelismo eficiente

Estudar Concorrência

- Motivação:
 - Mundo real não é sequencial!
 - Eventos ocorrem em paralelo
 - Lidamos com assuntos diversos em paralelo
 - Pessoas concorrem (cooperando ou não) para atingir um objetivo comum
 - Sistemas computacionais atuais permitem concorrência:
 - Tarefas podem compartilhar dispositivos comuns
 - Diversas tarefas podem ser executadas de forma intercalada
 - **Paralelismo de tarefas: muitos sistemas paralelos atualmente (*Multicores/manycores*)**

Concorrência: origens

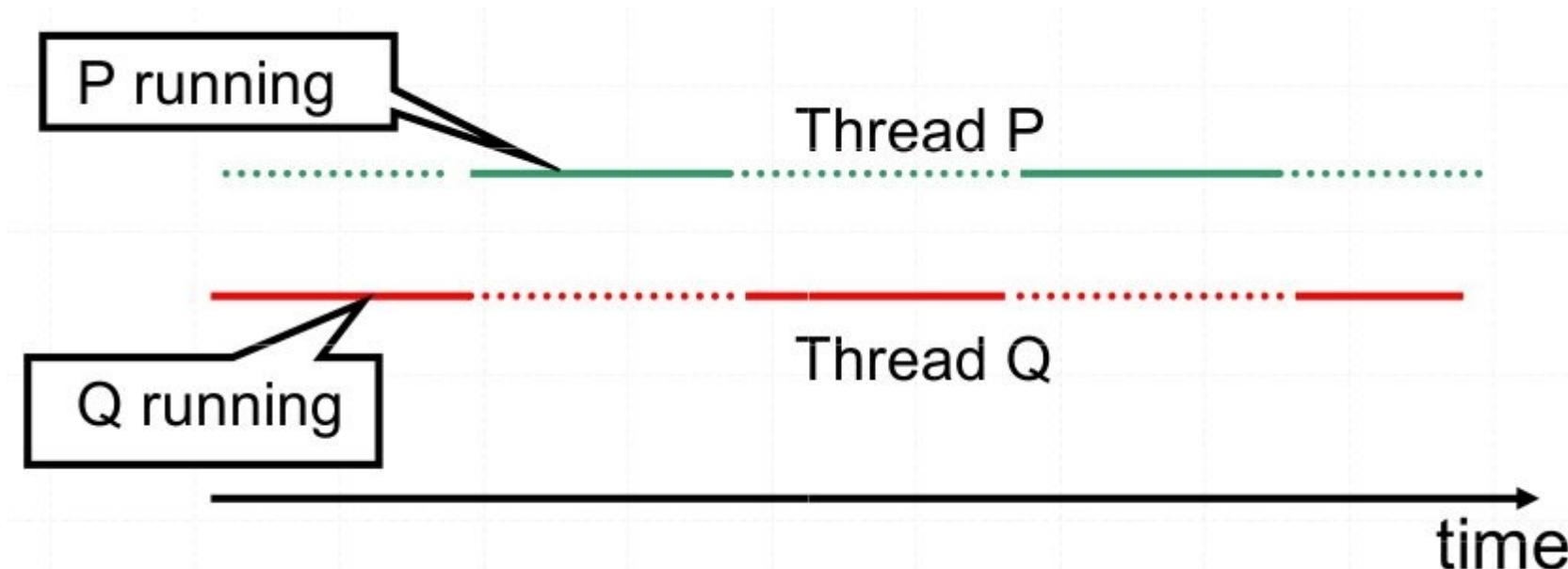
- Primórdios da computação:
 - Inexistência de SO
 - Execução de um único programa por vez usando todos os recursos do Hardware
- Surgimento dos SOs
 - Permitiu executar mais de um programa por vez, os quais são considerados processos
 - Executa isoladamente de forma protegida
 - Fatias de tempo pra cada processo
 - Compartilham recursos do hardware
 - Podem comunicar-se de diversas maneiras
 - Memória, mensagens, sockets

Concorrência: motivação

- Otimizar a utilização de recursos do hardware
 - Exemplos:
 - Recursos com I/O podem impor atrasos na computação. O tempo ocioso de espera da CPU pode ser alocado a outras tarefas
 - Sistemas multiusuário podem compartilhar recursos de forma justa, impondo fatias de tempo a cada utilizador
- Permitir paralelismo de fato

Concorrência em sistemas computacionais

- SOs multitarefas permitem que tarefas distintas sejam executadas concurrentemente em intervalos regulares (*interleaving*):
 - SO gerencia a troca de contexto na CPU
 - *Scheduling* (escalonamento) de tarefas
 - Aparência de que executam paralelamente



Gerenciamento de Processos no S.O.

- A tarefa de escalonamento de aplicações concorrentes é feita pelo SO através do *Scheduler* em tempo de execução
 - Existem muitos diferentes métodos para fazer o escalonamento de tarefas:
 - Cooperativo (Cada tarefa define o seu momento de dispensar o processador. Exemplo: esperando por *I/O*, terminação, *stand-by*)
 - Pré-emptivo (Interrompe-se uma tarefa para executar outra. Posteriormente pode-se retornar a antiga)
 - Combinações de ambas são permitidas
 - Priorizações de certos processos idem

Paralelismo demanda concorrência

- Velocidade e poder computacional de CPUs crescem a velocidade assombrosa
 - Base: Lei de Moore (será visto em seguida)
- Requisitos das aplicações (usuários) crescem mais rapidamente do que a velocidade das CPUs
- **Acoplar múltiplas CPUs (Máquinas paralelas) é uma forma viável de reduzir a distância entre as necessidades dos usuários e a velocidade de uma única CPU**
 - Exige algoritmos paralelos

Por que aumentar o número de processadores?

- Necessário onde a computação seqüencial não gera resultados em tempos aceitáveis
 - Melhorar o desempenho das aplicações!
 - Usuários continuarão a demandar maior potência computacional
 - Processamento paralelo
 - Algoritmos implementados seguindo um paradigma de programação paralela em máquinas adaptadas a estas características
 - Divisão de um problema “grande” e “complexo” em partes menores, as quais são resolvidas em paralelo

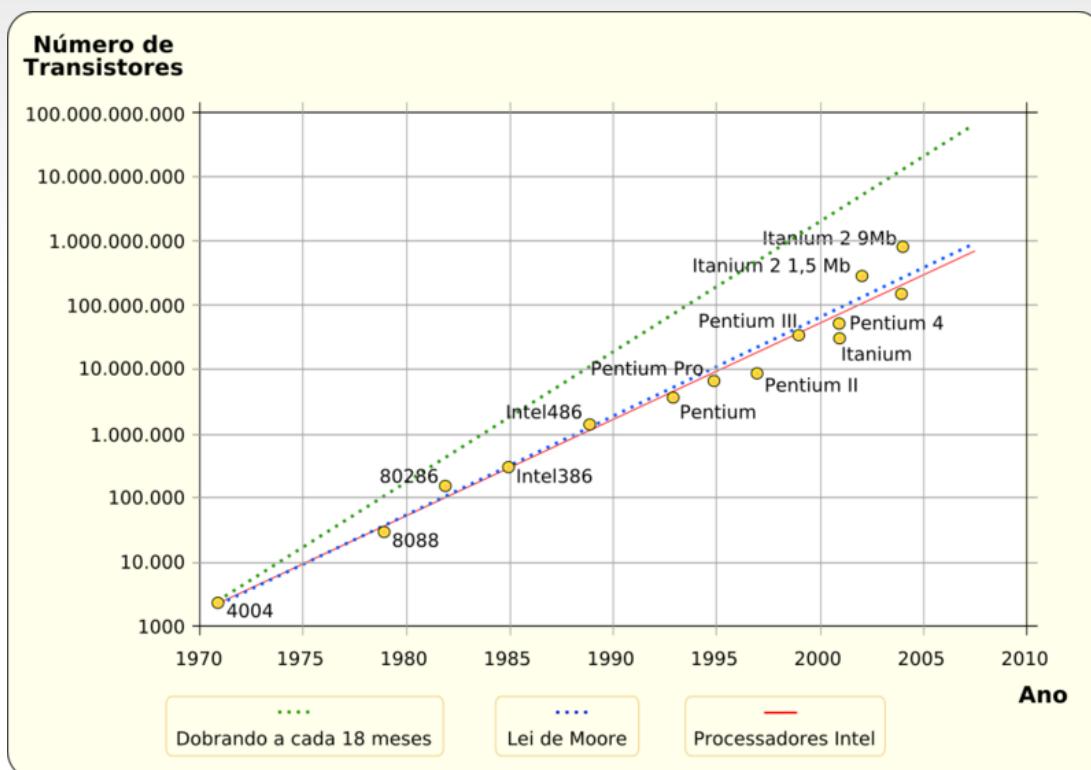
Nova ruptura em hardware ocorre...

- Aumento da velocidade de uma única CPU não é suficiente
 - Aumento da freqüência da CPU traz problemas de aquecimento e energia
 - I.L.P. é altamente explorado mas apresenta limitações
- Solução: Uso de múltiplas CPUs: **multicores e manycores**
- Mudança no hardware (“rupturas”) que requer re-escrita de software:
 - *Concurrency is the next major revolution in how we write software (Herb Sutter, Dr. Dobb's Journal, 30(3), March 2005.)*

O que são Processadores Multicores?

- Múltiplas Unidades Centrais de Processamento (CPUs) encapsuladas dentro de um mesmo Chip
 - Usualmente dois ou mais *Cores* (CPUs)
 - Grande possibilidade de aumento da quantidade de cores no futuro próximo
- Frequentemente encontrado em máquinas *COTs* (*Commercial Off-The-Shelf*), ou seja, em máquinas comuns do dia a dia
 - Tendência forte na indústria de CPUs (Intel, AMD)

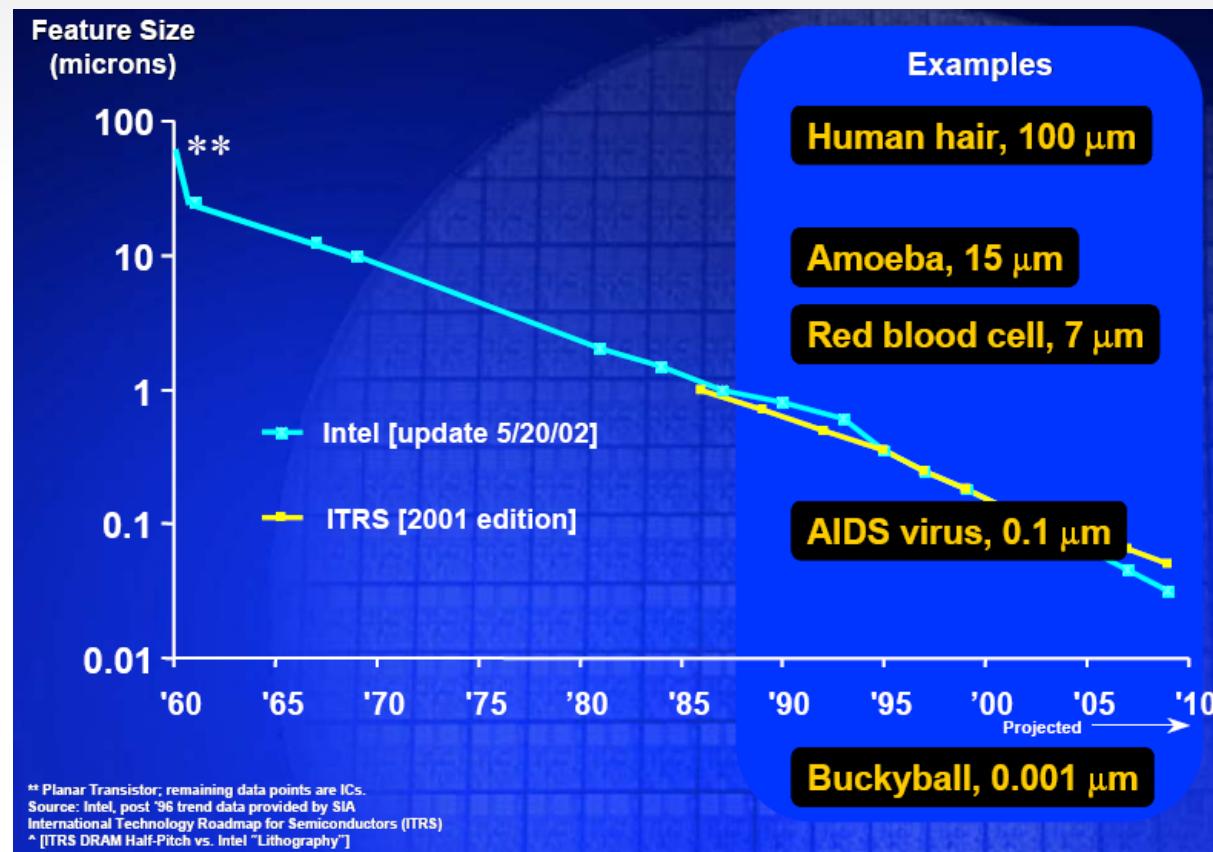
Entendendo o Multicore: Lei de Moore



- Gordon Moore:
“Cramming more components onto integrated circuits”,
Electronics v.38, N.8,
1965
- Número de transistores integrados em Chips (CPUs inclusive) dobra a cada 18/24 meses

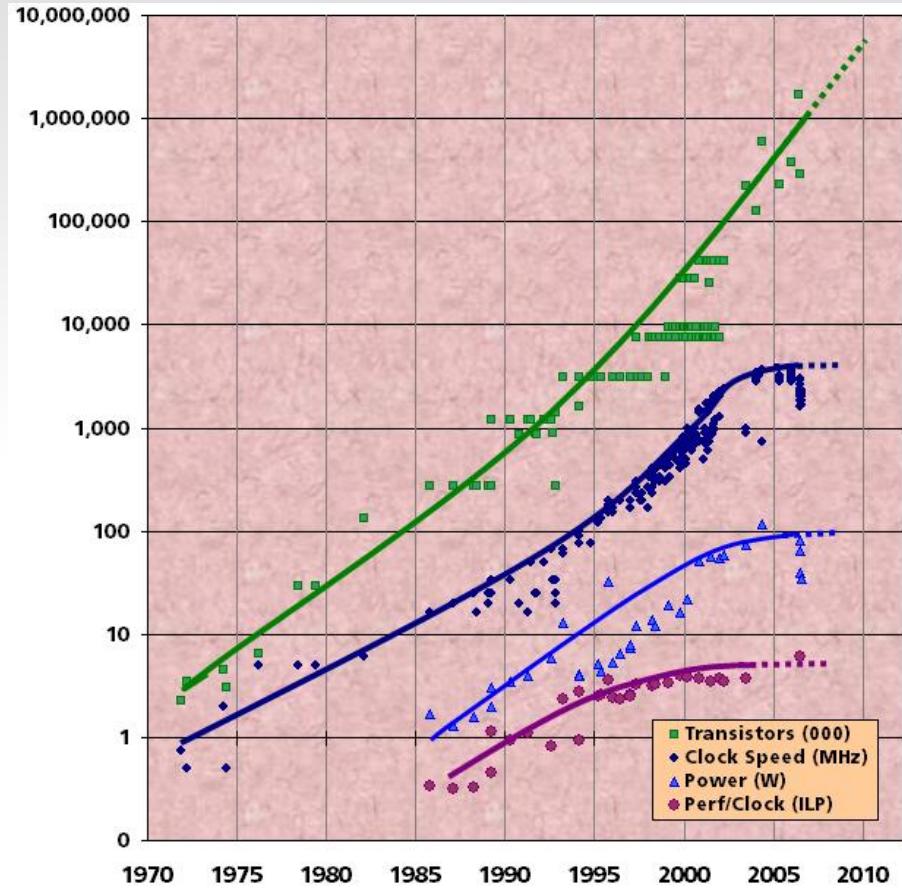
Em que se baseia a Lei de Moore

- Tecnologia de Litografia constantemente reduz tamanho de componentes em CI



Fonte: G. Moore, palestra convidada ISSCC 2003

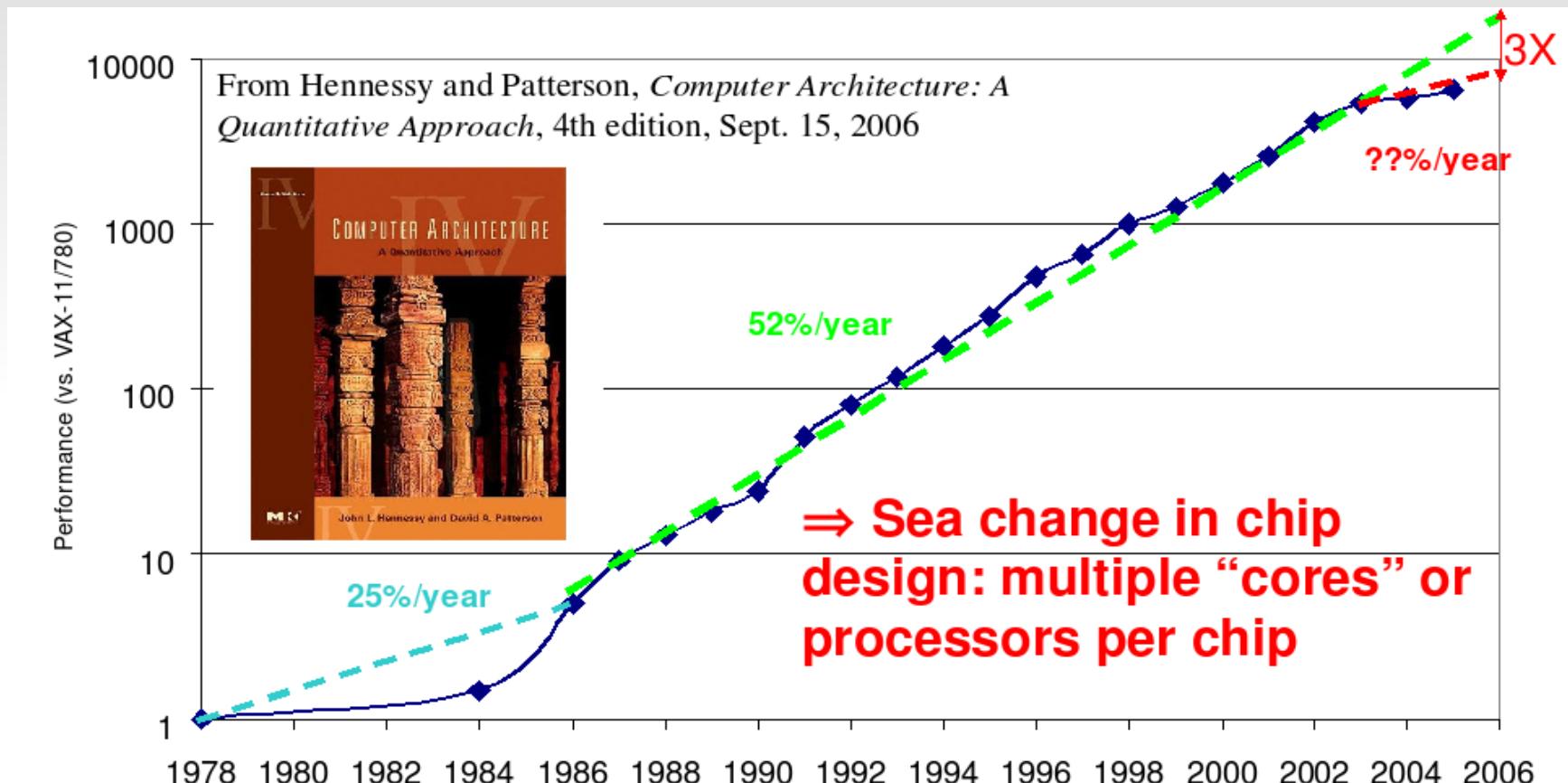
Lei de Moore na era do Multicore



- Válida para integração de componentes
- Freqüência de *clock* e desempenho parecem não mais acompanhar
- Reinterpretação:
 - Número de cores aumenta a cada 18-24 meses

Fonte: Shalf, J., SCIDAC, 2007

Desempenho com Multicore Histórico do SPECINT



- VAX : 25%/year 1978 to 1986
- RISC + x86: 52%/year 1986 to 2002
- RISC + x86: ??%/year 2002 to present

Sumarizando até o momento...

- O aumento relativo de desempenho de um dado processador é bem menor que o aumento relativo no número de componentes integrados em um CI para um mesmo período de tempo.
- **Motivo: Existem barreiras para a melhoria no desempenho!!**

Barreira imposta pelo consumo de energia



Power Cost of Frequency

- Power \propto Voltage² x Frequency (V²F)
- Frequency \propto Voltage
- Power \propto Frequency³

	Cores	V	Freq	Perf	Power	PE (Bops/watt)
Superscalar	1	1	1	1	1	1
"New" Superscalar	1X	1.5X	1.5X	1.5X	3.3X	0.45X
Multicore	2X	0.75X	0.75X	1.5X	0.8X	1.88X

(Bigger # is better)

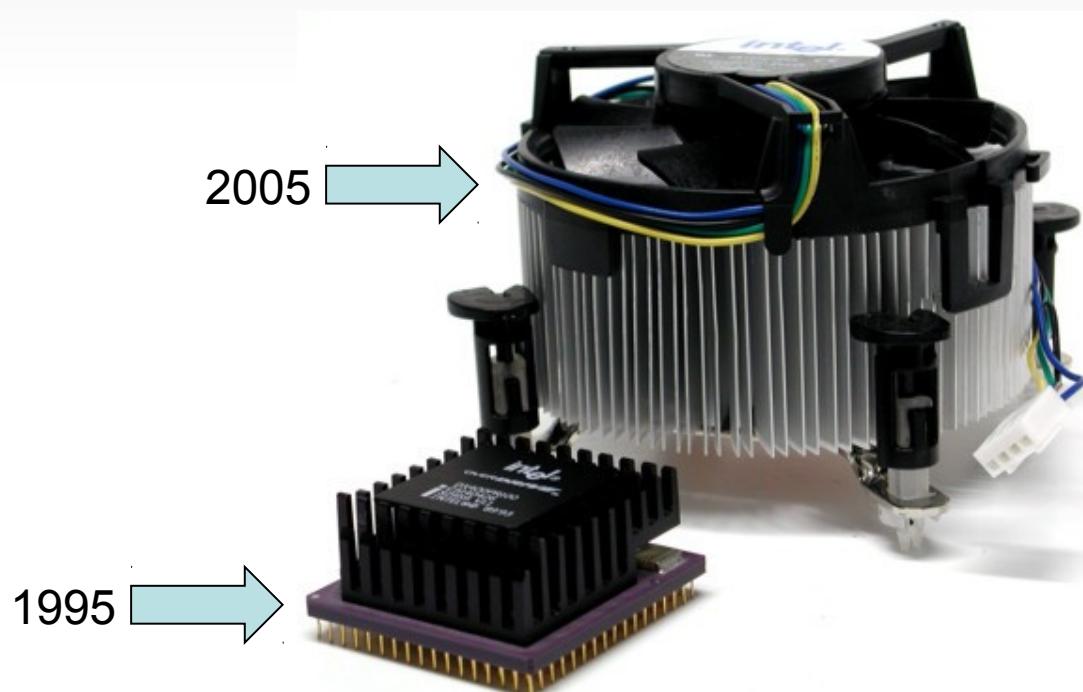
50% more performance with 20% less power

Preferable to use multiple slower devices, than one superfast device

Fonte: Jack Dongarra, 2008 – SIAM Anual Meeting

Consumo de energia e Dissipação térmica

- Níveis insustentáveis!!!
- Aumentar a freqüência é cada vez mais difícil



Fonte: Ed Davis, Intel

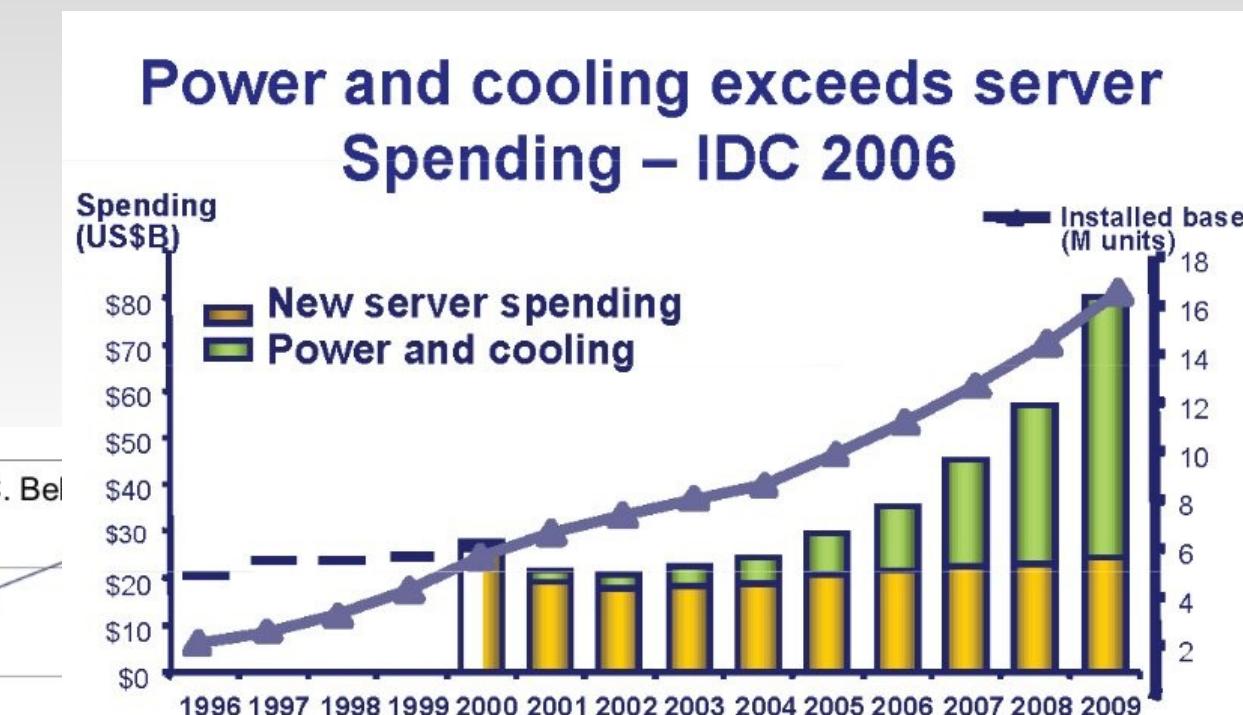
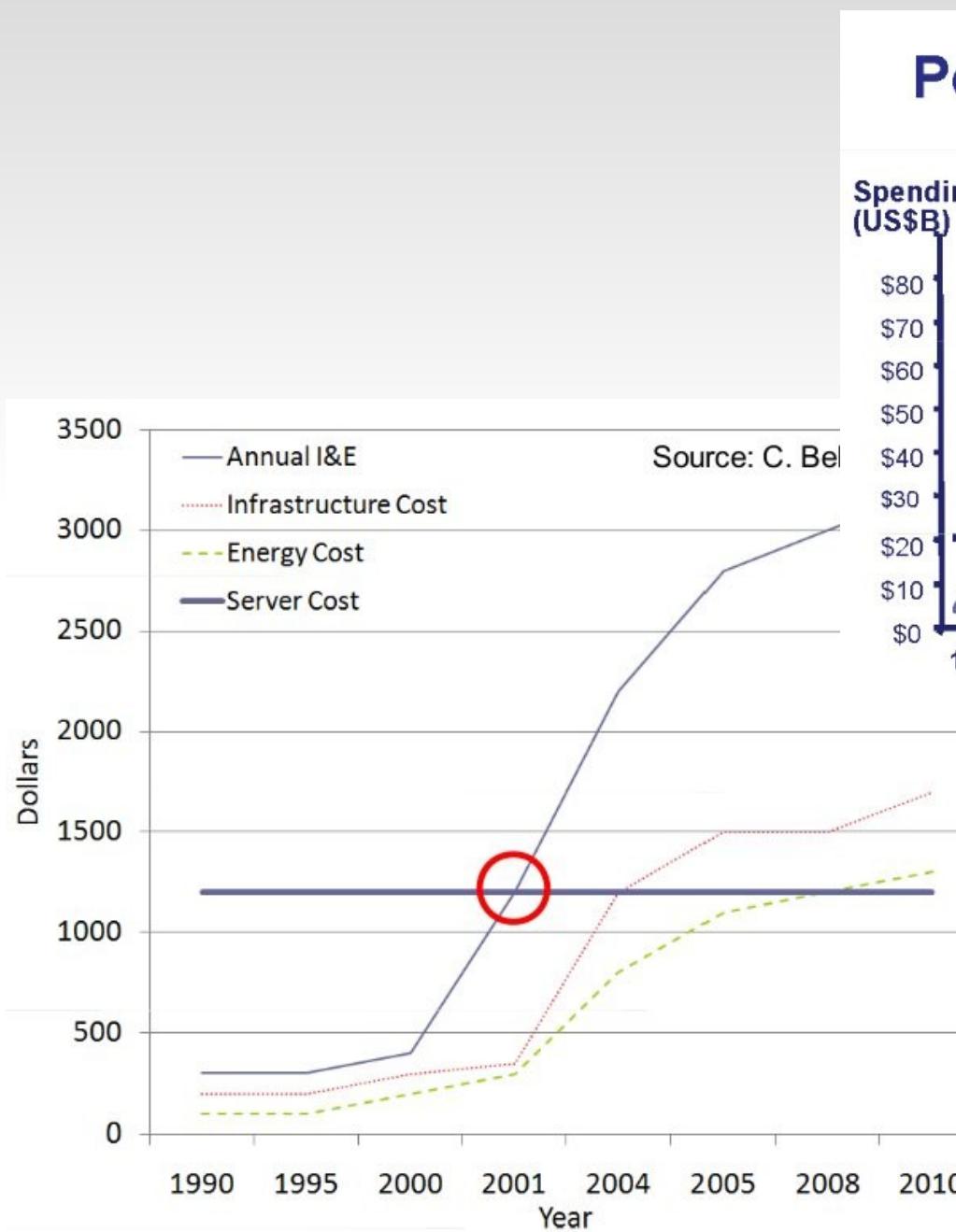
Consumo de energia é sério problema (*power wall*)

The screenshot shows a news article from CNET News.com. The header includes the CNET logo and 'NEWS.com'. Navigation links include 'Today on CNET', 'News', 'Reviews', 'Compare prices', 'How-to', and 'Downloads'. Below the header are categories like 'Today on News', 'Business Tech', 'Cutting Edge', 'Access', 'Threats', 'Media 2.0', 'Markets', and 'Digital Life'. A search bar and login links ('Log in | Sign up') are also present. The main headline reads 'Power could cost more than servers, Google warns'. The byline is 'By Stephen Shankland, Staff Writer, CNET News.com'. The date is 'Published: December 9, 2005, 4:00 AM PST' and 'Last modified: December 9, 2005, 9:55 AM PST'. There are links for 'TalkBack', 'E-mail', and 'Print'. To the right is a small advertisement featuring a silver car.

**“Hiding in Plain Sight, Google Seeks More Power”,
by John Markoff, June 14, 2006 The New York Times**

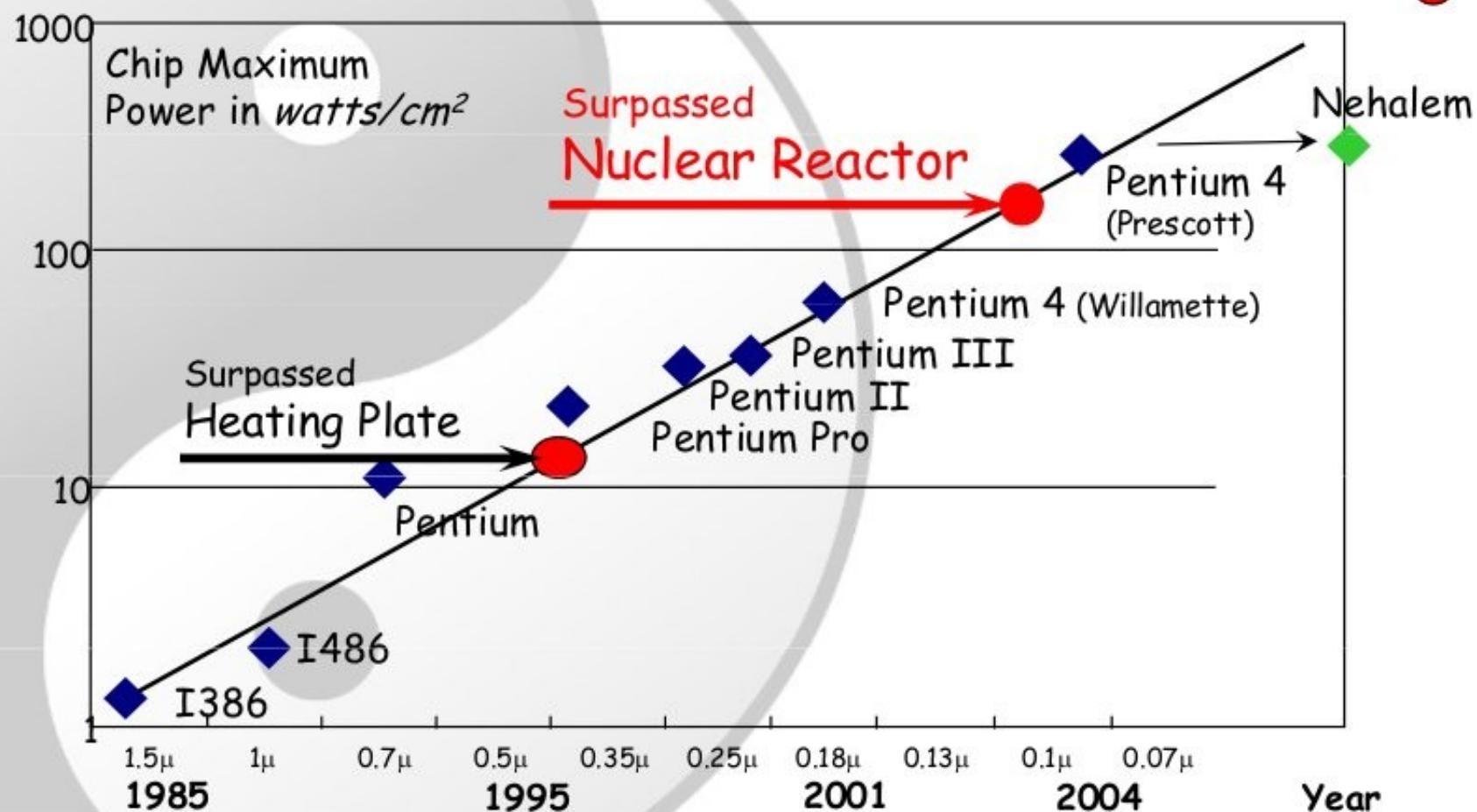


Power Wall (II)



Moore's Law for Power Consumption

Rocket
Nozzle



Resposta da Indústria

Power and Heat: Intel Embraces Multicore

May 17, 2004 ... Intel, the world's largest chip maker, publicly acknowledged that it had hit a "thermal wall" on its microprocessor line. As a result, the company is changing its product strategy and disbanding one of its most advanced design groups. Intel also said that it would abandon two advanced chip development projects ...

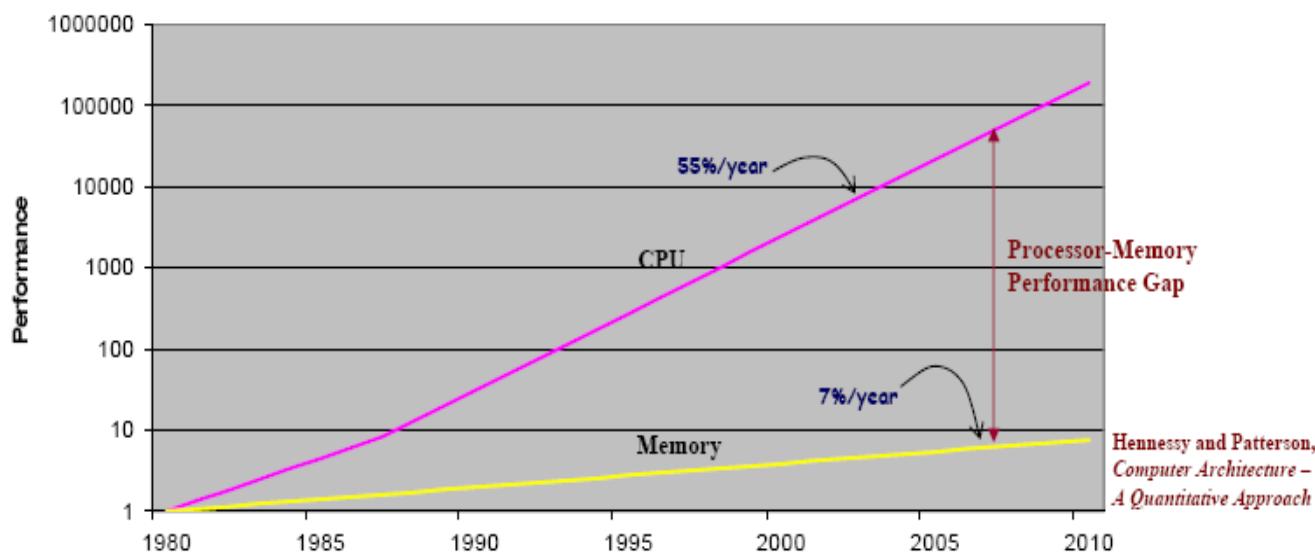
Now, Intel is embarked on a course already adopted by some of its major rivals: obtaining more computing power by stamping multiple processors on a single chip rather than straining to increase the speed of a single processor ... Intel's decision to change course and embrace a "dual core" processor structure shows the challenge of overcoming the effects of heat generated by the constant on-off movement of tiny switches in modern computers ... some analysts and former Intel designers said that *Intel was coming to terms with escalating heat problems so severe they threatened to cause its chips to fracture at extreme temperatures...*

New York Times, May 17, 2004

Barreira imposta pelo acesso a memória

MEMORY WALL

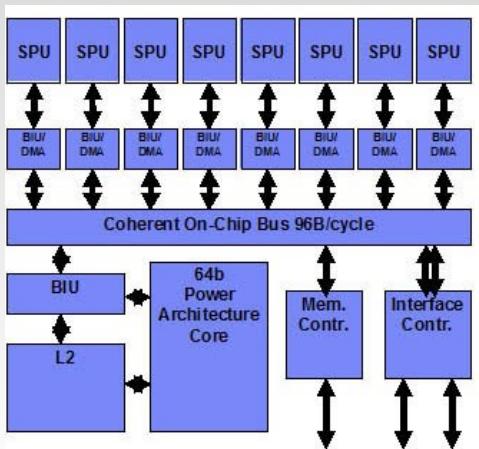
- A velocidade de acesso à memória escala mais lentamente que a velocidade da CPU, ao longo dos anos
 - Acesso à memória torna-se o gargalo da eficiência



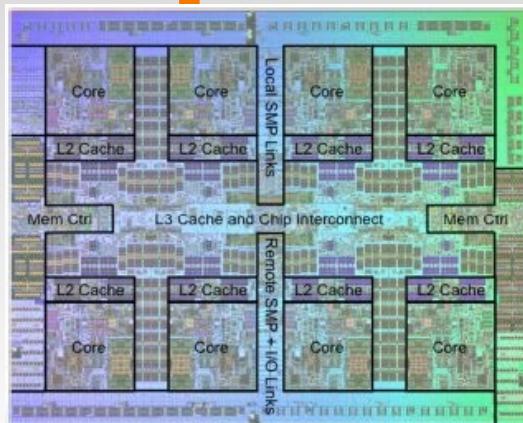
Em síntese:

- Como usar maior número de componentes para gerar máquinas mais rápidas, uma vez que a Lei de Moore permanece válida?
 - Aumentar a freqüência não é mais possível devido as barreiras apresentadas
- **Tendência clara: Múltiplas CPUs de menor freqüência no mesmo chip (Multicores)**

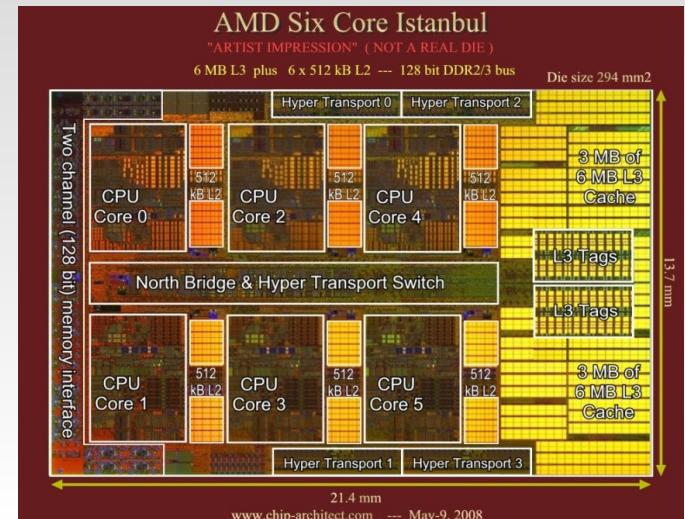
Multicore (CMP-Chip Multiprocessor)



IBM Cell (9 cores)



IBM Power 7



2009



Intel® Xeon® 5500 Platform (previously Tylersburg-EP)

Intel® Xeon® processor 5500 series

(Nehalem-EP)

Intel® 5520 Chipset

Technologies

Quad-Core / Dual Core

DDR3 1333/1066/800 memory

Intel® Turbo Boost Technology,

Intel® VT

Intel® QuickPath Interconnect (QPI)

Intel® Hyper-Threading Technology

Lead and halogen free platform kit¹

2010

Next-generation Intel® Xeon® processor

(Westmere-EP^{*})

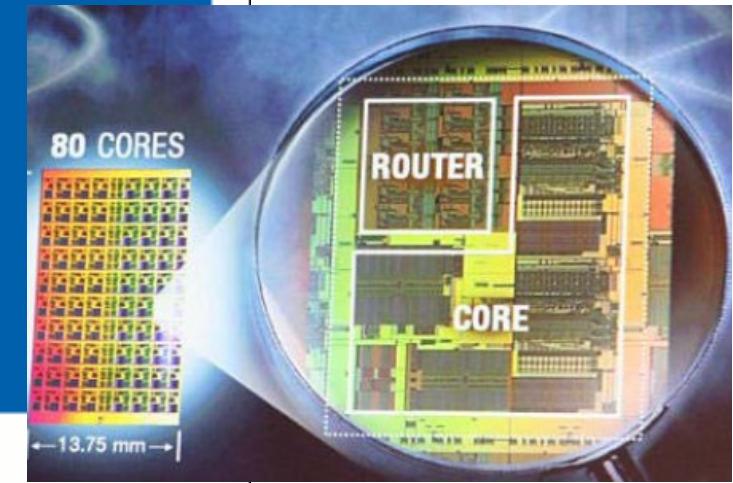
Intel® 5520 Chipset

Technologies

Up to 6 cores/12 Threads

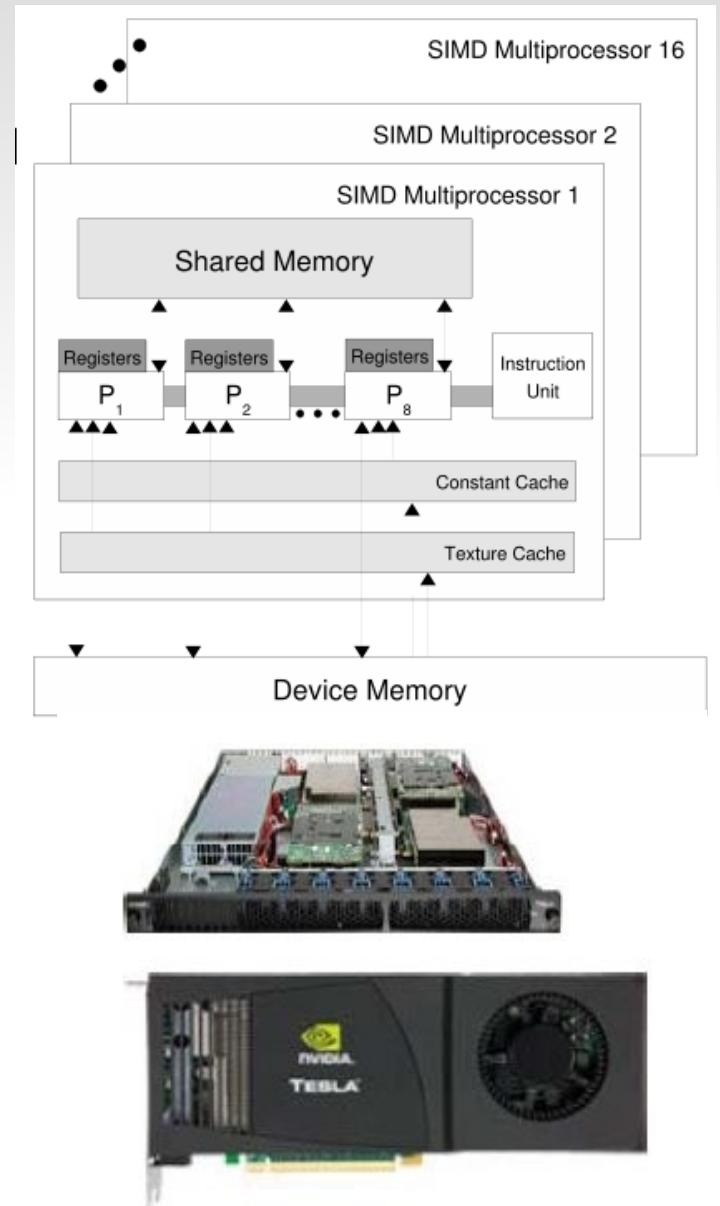
Improved Energy Efficiency

Cache Enhancements



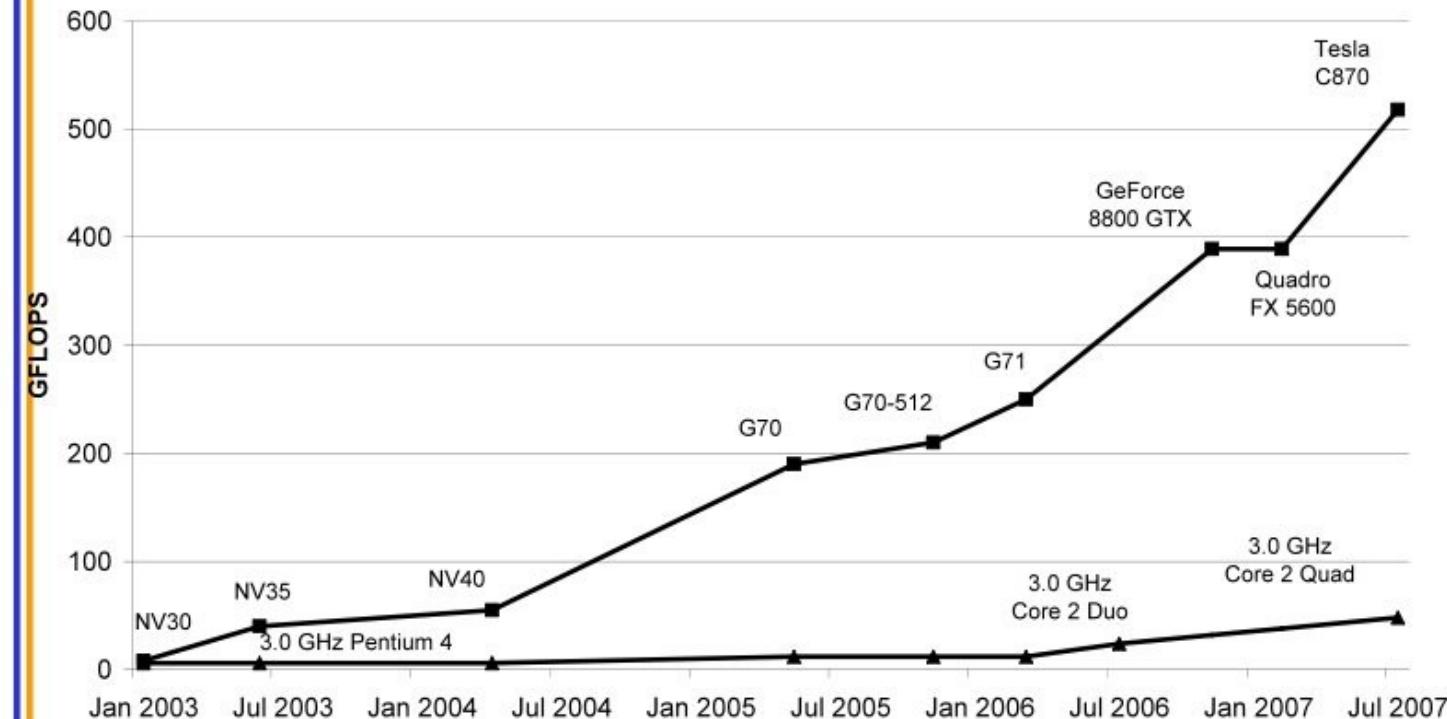
Multicores GPU - SIMD

- Paralelismo massivo: muitas unidades SIMD
 - NVIDIA's Tesla T10P
 - 240 cores; 1.5 GHz
 - Tpeak 1/0.1 Tflop/s – 32/64 bit floating point
- Biblioteca NVIDIA CUDA C/C++
- Portland Group desenvolveu compilador C/C++/Fortran-90
- Ótimo custo/benefício:
US\$ 2000/TeraFlop



Avanços de GPUs

What is driving the many-cores?



¹ Based on slide 7 of S. Green, "GPU Physics," SIGGRAPH 2007 GPGPU Course. <http://www.gpgpu.org/s2007/slides/15-GPGPU-physics.pdf>

Arquitetura interna

Design philosophies are different.

- The GPU is specialized for compute-intensive, massively data parallel computation (exactly what graphics rendering is about)
 - So, more transistors can be devoted to data processing rather than **data caching** and **flow control**



- The fast-growing video game industry exerts strong **economic pressure** for constant innovation

Avanços para supercomputação (I)

Unclassified

Livermore computing

Adopting disruptive technologies can dramatically reduce costs

BlueGene/L

Purple

Peak Speed = 93 Teraflops

Footprint = 12,000 sq. ft.

Power = 4.5MW

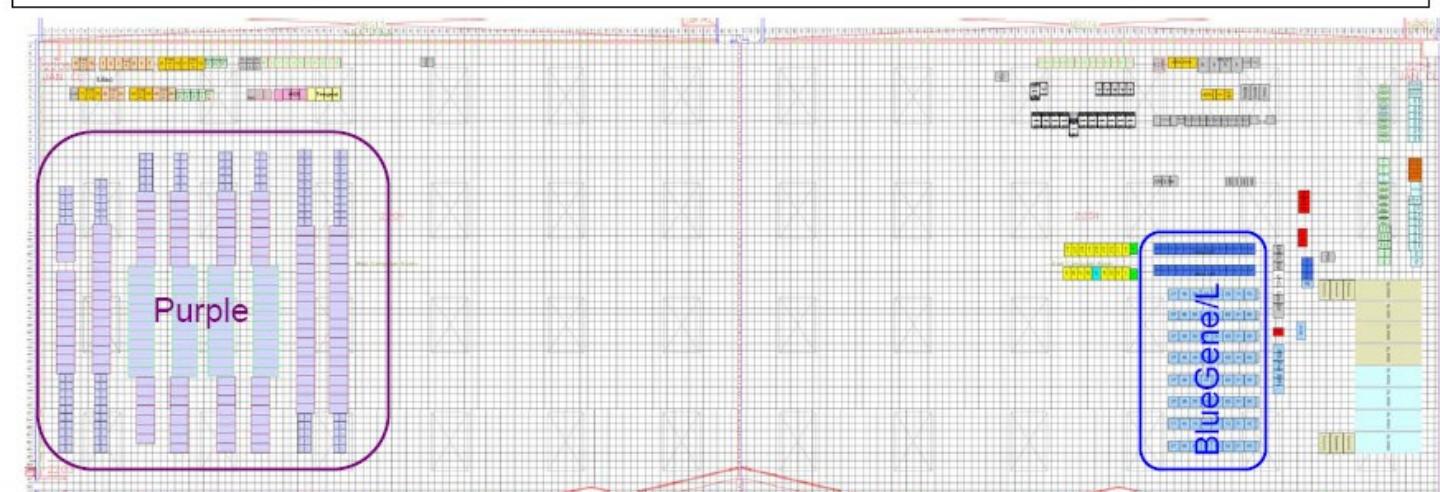
Peak Speed = 180/360 Teraflops

Footprint = 2,500 sq. ft.

Power = 1.5MW

Over a 4 year period Purple will cost \$12M more for power than BG/L

Cost per Teraflop for BG/L was much less than Purple's



Computation's Mini- DRC 4/20/06

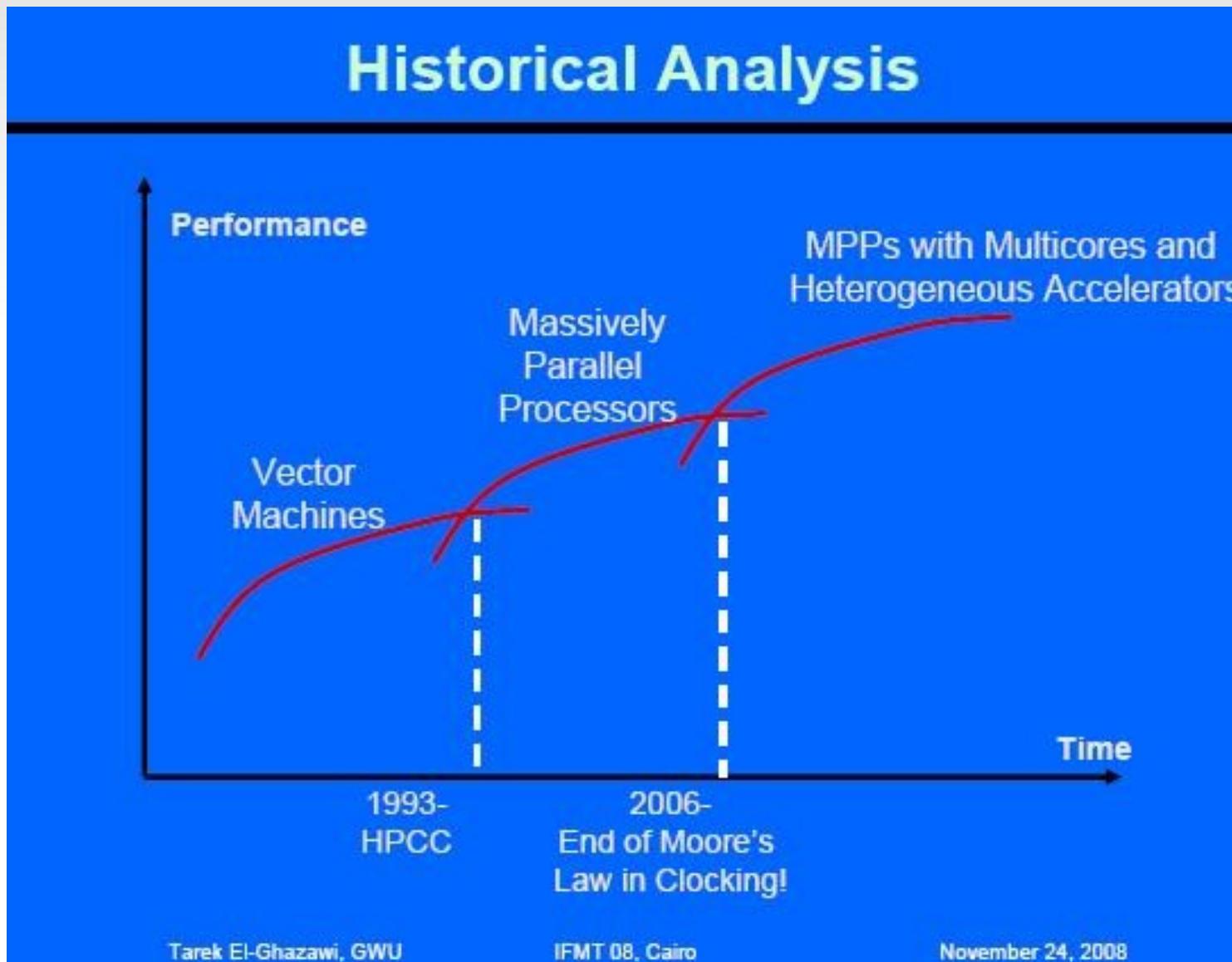
Unclassified

Document #7

IBM Power 5
12.208 Processadores

IBM Blue Gene
131.072 Processadores

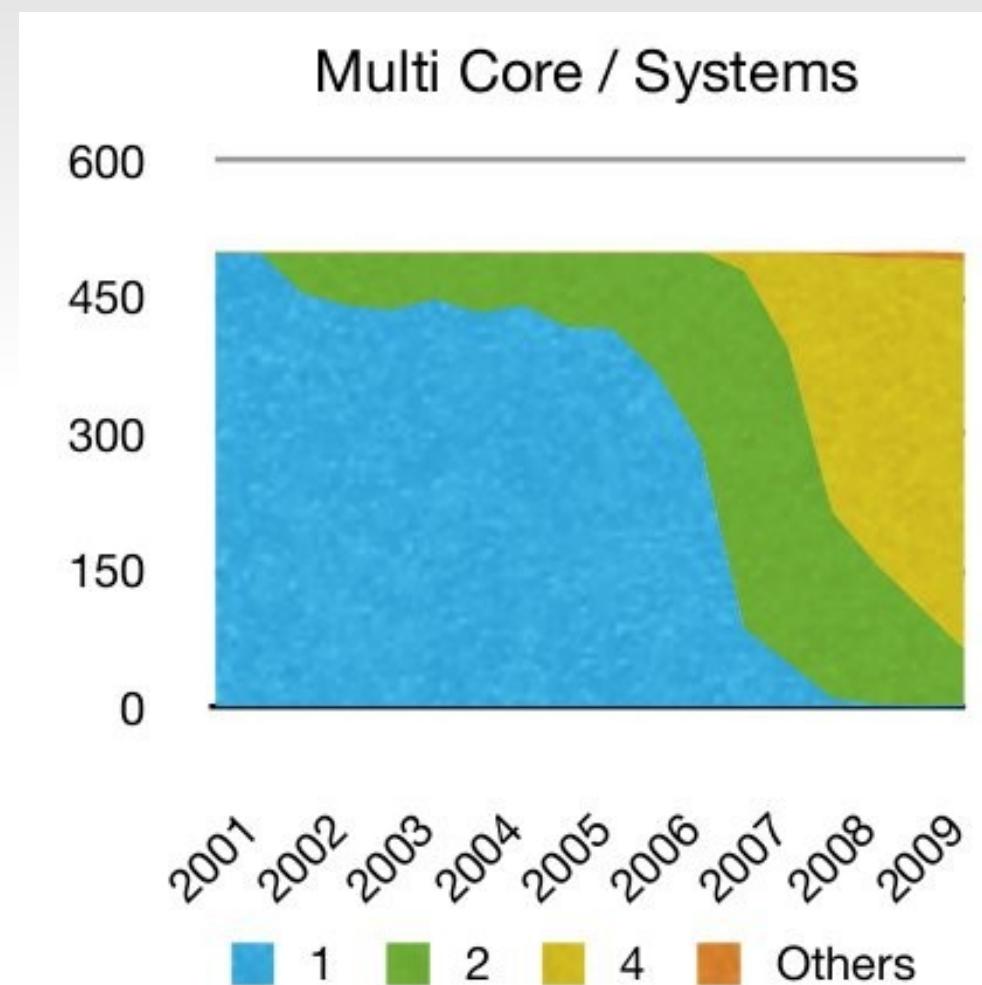
Avanços para supercomputação (II)



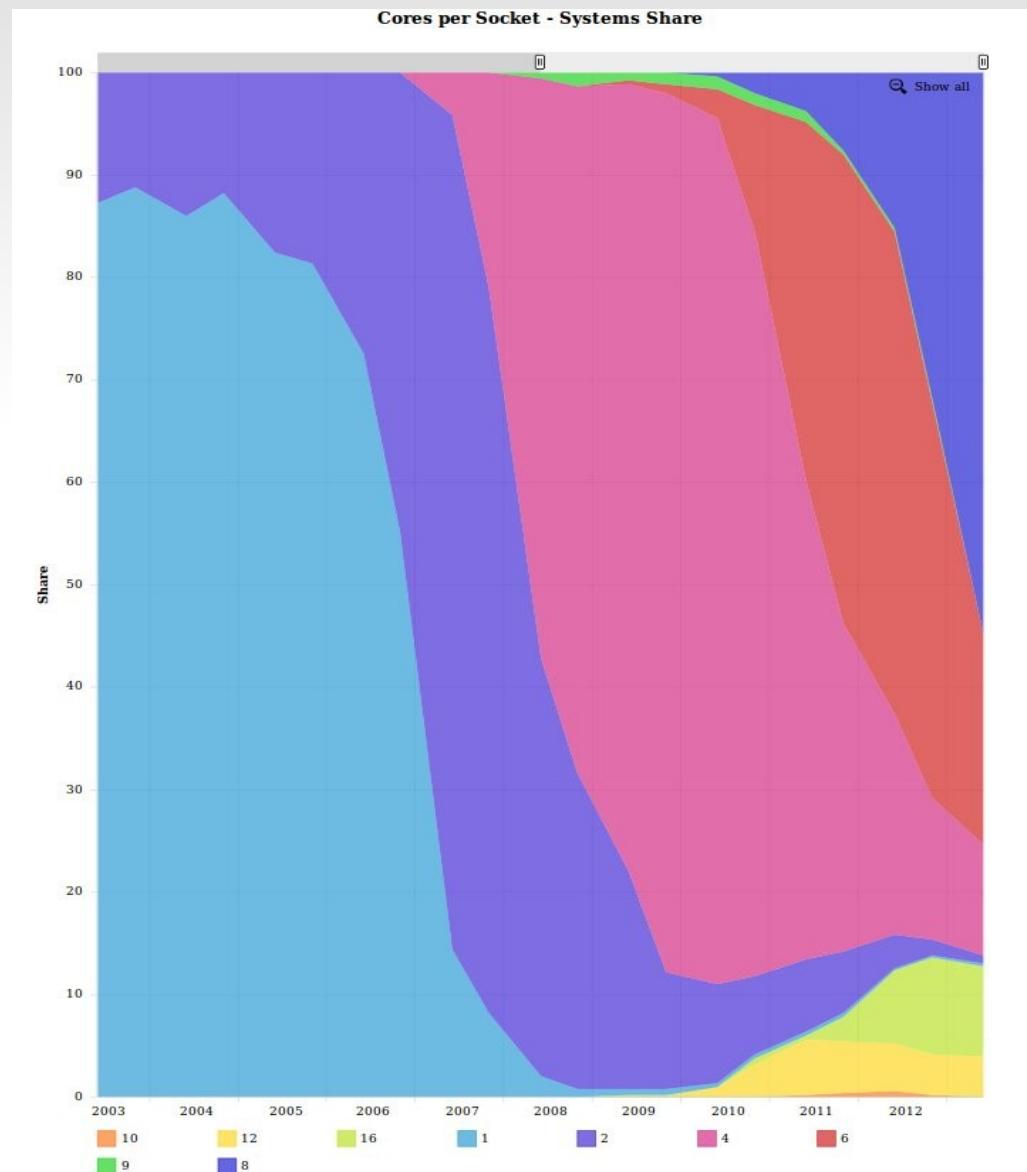
- Processadores “Quad-core” são usados em 85,4% (426) dos sistemas, geralmente acompanhado de aceleradores de hardware.

Exemplo:

- Tianhe-1 (China, No.5): Cada nó consiste de 2 AMD GPUs (aceleradores) anexados a 2 Intel Xeon
- Single-core em apenas 4 sist.



- Processadores 6 núcleos ou mais são usados em 88,8% dos sistemas
- Processadores com 8 ou mais núcleos são usados em 67% dos sistemas
- 54 sistemas usam aceleração por dispositivo externo a CPU:
 - 39 com GPU NVIDIA
 - 3 com GPU AMD/ATI
 - 11 com Intel Xeon Phi





Junho/2016

Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	Summit - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband , IBM DOE/SC/Oak Ridge National Laboratory United States	2,282,544	122,300.0	187,659.3	8,806
2	Sunway TaihuLight - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway , NRCPC National Supercomputing Center in Wuxi China	10,649,600	93,014.6	125,435.9	15,371
3	Sierra - IBM Power System S922LC, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband , IBM DOE/NNSA/LLNL United States	1,572,480	71,610.0	119,193.6	
4	Tianhe-2A - TH-IVB-FEP Cluster, Intel Xeon E5-2692v2 12C 2.2GHz, TH Express-2, Matrix-2000 , NUDT National Super Computer Center in Guangzhou China	4,981,760	61,444.5	100,678.7	18,482
5	AI Bridging Cloud Infrastructure (ABCi) - PRIMERGY CX2550 M4, Xeon Gold 6148 20C 2.4GHz, NVIDIA Tesla V100 SXM2, Infiniband EDR , Fujitsu National Institute of Advanced Industrial Science and Technology (AIST) Japan	391,680	19,880.0	32,576.6	1,649
6	Piz Daint - Cray XC50, Xeon E5-2690v3 12C 2.6GHz, Aries interconnect , NVIDIA Tesla P100 , Cray Inc. Swiss National Supercomputing Centre (CSCS) Switzerland	361,760	19,590.0	25,326.3	2,272
7	Titan - Cray XK7, Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x , Cray Inc. DOE/SC/Oak Ridge National Laboratory United States	560,640	17,590.0	27,112.5	8,209
8	Sequoia - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom , IBM DOE/NNSA/LLNL United States	1,572,864	17,173.2	20,132.7	7,890
9	Trinity - Cray XC40, Intel Xeon Phi 7250 68C 1.4GHz, Aries interconnect , Cray Inc. DOE/NNSA/LANL/SNL United States	979,968	14,137.3	43,902.6	3,844



TOP 500 #1 - Summit

- 122.300 petaflop/s no Linpack benchmark
- Processador IBM POWER9 22C 3.07GHz
- Total de 2.282.544 cores
- Consumo de energia de 8.805 MW
- Localização: DOE/SC/Oak Ridge National Laboratory (USA)



TOP 500 #2 - Sunway TaihuLight

- 93.014 petaflop/s no Linpack benchmark
- 40,960 nós
- Processador Sunway SW26010 260C 1.45GHz
- Total de 10,649,600 cores
- Consumo de energia de 15.37 MW
- Localização: National Supercomputing Center – Wuxi, China



TOP 500 #4 – Tianhe-2A

- 33.86 petaflop/s no Linpack benchmark
- 16,000 nós
- Cada nó com 2 Intel Xeon IvyBridge e 3 co-processadores Xeon Phi
- Total de 3,120,000 cores
- Consumo de energia de 17.8 MW
- Localização: National Super Computer Center – Guangzho, China



TOP 500 #7 - Titan

- 17.59 petaflop/s no Linpack benchmark
- Processador Opteron 6274 16C 2.2GHz
- Total de 560,640 cores
- Consumo de energia de 8.209 MW
- Localização: DOE/SC/Oak Ridge National Laboratory – Oak Ridge, EUA



Brasil (Junho 2018)

BC1 - Lenovo C1040, Xeon E5-2673v4 20C 2.3GHz, 40G Ethernet

Site:	Cloud Provider
Manufacturer:	Lenovo
Cores:	38,400
Memory:	491,520 GB
Processor:	Xeon E5-2673v4 20C 2.3GHz
Interconnect:	40G Ethernet
Performance	
Linpack Performance (Rmax)	1,123.15 TFlop/s
Theoretical Peak (Rpeak)	1,413.12 TFlop/s
Nmax	7,010,880
Power Consumption	
Power:	
Software	
Operating System:	Linux



Santos Dumont GPU

- 456.8 teraflop/s no Linpack benchmark
- Processador Intel Xeon E5-2695v2 12C 2.4GHz
- Total de 10,692 cores
- Localização: Laboratório Nacional de Computação Científica - Petrópolis, Brasil
- Desligado por alto custo de energia
<http://agenciabrasil.ebc.com.br/geral/noticia/2016-06/alto-custo-de-energia-poe-em-risco-funcionamento-do-supercomputador-brasileiro>



Santos Dumont Hybrid

- 363.225 teraflop/s no Linpack benchmark
- Processador Intel Xeon E5-2695v2 12C 2.4GHz
- Total de 24,732 cores
- Localização: Laboratório Nacional de Computação Científica – Petrópolis, Brasil
- Desligado por alto custo de energia
<http://agenciabrasil.ebc.com.br/geral/noticia/2016-06/alto-custo-de-energia-poe-em-risco-funcionamento-do-supercomputador-brasileiro>



Santos Dumont CPU

- 321.178 teraflop/s no Linpack benchmark
- Processador Intel Xeon E5-2695v2 12C 2.4GHz
- Total de 18,144 cores
- Localização: Laboratório Nacional de Computação Científica - Petrópolis, Brasil
- Desligado por alto custo de energia
<http://agenciabrasil.ebc.com.br/geral/noticia/2016-06/alto-custo-de-energia-poe-em-risco-funcionamento-do-supercomputador-brasileiro>



CIMATEC Yemoja

- 405.365 teraflop/s no Linpack benchmark
- Processador Intel Xeon E5-2690v2 10C 3GHz
- Total de 17,200 cores
- Localização: SENAI CIMATEC – Salvador, Brasil



Ranking the World's Most ENERGY-EFFICIENT SUPERCOMPUTERS

TOP500			Cores	Rmax (TFlop/s)	Power (kW)	Power Efficiency (GFlops/watts)
Rank	Rank	System				
1	359	Shoubu system B - ZettaScaler-2.2, Xeon D-1571 16C 1.3GHz, Infiniband EDR, PEZY-SC2 , PEZY Computing / Exascaler Inc. Advanced Center for Computing and Communication, RIKEN Japan	794,400	857.6	47	18.404
2	419	Suiren2 - ZettaScaler-2.2, Xeon D-1571 16C 1.3GHz, Infiniband EDR, PEZY-SC2 , PEZY Computing / Exascaler Inc. High Energy Accelerator Research Organization /KEK Japan	762,624	798.0	47	16.835
3	385	Sakura - ZettaScaler-2.2, Xeon E5-2618Lv3 8C 2.3GHz, Infiniband EDR, PEZY-SC2 , PEZY Computing / Exascaler Inc. PEZY Computing K.K. Japan	794,400	824.7	50	16.657
4	227	DGX SaturnV Volta - NVIDIA DGX-1 Volta36, Xeon E5-2698v4 20C 2.2GHz, Infiniband EDR, NVIDIA Tesla V100 , Nvidia Corporation United States	22,440	1,070.0	97	15.113
5	1	Summit - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband , IBM DOE/SC/Oak Ridge National Laboratory United States	2,282,544	122,300.0	8,806	13.889
6	19	TSUBAME3.0 - SGI ICE XA, IP139-SXM2, Xeon E5-2680v4 14C 2.4GHz, Intel Omni-Path, NVIDIA Tesla P100 SXM2 , HPE GSIC Center, Tokyo Institute of Technology Japan	135,828	8,125.0	792	13.704
7	287	AIST AI Cloud - NEC 4U-8GPU Server, Xeon E5-2630Lv4 10C 1.8GHz, Infiniband EDR, NVIDIA Tesla P100 SXM2 , NEC National Institute of Advanced Industrial Science and Technology Japan	23,400	961.0	76	12.681



Multicores requerem reescrita de software

- Características do software para multicore:
 - **Granularidade fina**
 - **Execução de threads assíncrona**

Parallel Linear Algebra Software for Multicore Architectures (PLASMA)			
Software/Algorithms follow hardware evolution in time			
LINPACK (70's) (Vector operations)		Rely on - Level-1 BLAS operations	
LAPACK (80's) (Blocking, cache friendly)		Rely on - Level-3 BLAS operations	
ScaLAPACK (90's) (Distributed Memory)		Rely on - PBLAS Mess Passing	
PLASMA (00's) New Algorithms (many-core friendly)		Rely on - a DAG/scheduler - block data layout - some extra kernels	
Those new algorithms	<ul style="list-style-type: none">- have a very low granularity, they scale very well (multicore, petascale computing, ...)- removes a lots of dependencies among the tasks, (multicore, distributed computing)- avoid latency (distributed computing, out-of-core)- rely on fast kernels		
	Those new algorithms need new kernels and rely on efficient scheduling algorithms.		

Fonte: Jack Dongarra, 2008 – SIAM Anual Meeting

Do que estamos falando...

Boys don't make passes at female smart-asses - Letty Cottin Pogrebin

Sun 23 Mar 2008

the INQUIRER

News, reviews, facts and friction

Search

Go

RSS

WEEK TO DATE
EARLIER
REVIEWS



SOFTWARE
MEMORY
MICROPROCESSORS
MOBILE/PDA
GRAPHICS
SERVERS
CHANNEL
EMISSION GOALS
GUTTER WATCH
STORAGE
SECURITY

Parallel programmers not prepared for the glorious revolution

Intel goes post-Communist with multicore

By Wily Ferret: Tuesday, 27 November 2007, 12:28 PM

INTEL RECKONS

barely one per cent of software programmers are prepared to face the challenge of parallel programming, which the hardware giant (unsurprisingly) reckons is the future of development.

The HP StorageWorks 1200 All-in-One Storage System
Powered by the Dual-Core Intel® Xeon® Processor

TechFinder

Search for Software,
Hardware & Services



Newsletter signup

InqBOT Weekly Newsletter

Submit

Existing user

lenovo™

Preocupação na indústria

Biting the hand that feeds IT Cash 'n' Carrion Events

The Register®

Hardware Software Music & Media Comms Security Public Sector Management Science Odds & Sod

PCs & Chips | Servers | Storage | Data Networking | Semi-Coherent Computing | Virtualization

Gartner SYMPOSIUM ITXPO® 2008 Emerging Trends

April 6-10 Las Vegas, NV gartner.com/us/emergingtrends REGISTER NOW

The Register » Hardware » PCs & Chips »

Intel and Microsoft dump \$20m on researchers to avert software crisis

To your Ivory Towers!

By Ashlee Vance in Mountain View → [More by this author](#) Page: 1 2 [Next >](#)
Published Tuesday 18th March 2008 18:42 GMT

[Nail down your security priorities. Ask the experts and your peers at The Register Security Debate, April 17, 2008](#)

Microsoft and Intel have put their money where their fear is. The two companies have shelled out \$10m each to the University of California, Berkeley and the University of Illinois to fund research around advanced software programming techniques for PCs and mobile devices.

Symantec Backup Exec™ 12

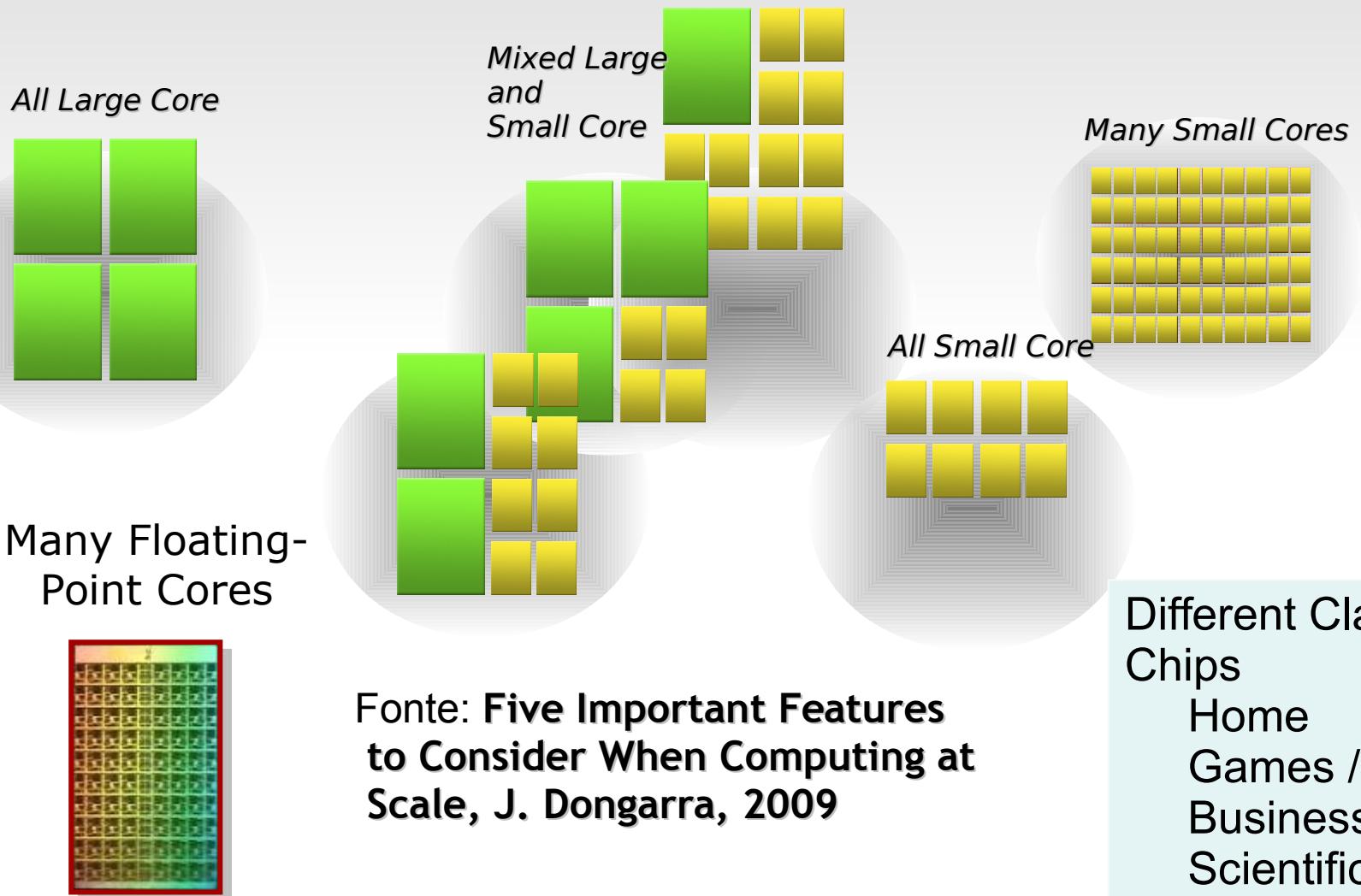
See for yourself:



Sério problema atual e futuro!

- Cores heterogêneos
 - Aceleradores de desempenho específicos
 - GPGPUs (General-Purpose computation on Graphics Processing Units) massificados em P.A.D.
- Multi-core para Many-core ???
 - Many-core englobam vários (ordem de dezenas a centenas) processadores menos complexos
 - Cada core individual consome muito menos energia
 - Exigirá mais paralelismo para melhorar throughput
- Linguagens de programação que expressam mais facilmente paralelismo?
- Técnicas/Algoritmos mais adaptados a paralelismo
- Ferramentas de medição de desempenho adequadas?

Futuros possíveis Manycores



Intel® XEON PHI

Intel® Xeon Phi™ Coprocessor 3100 Product Family

Performance

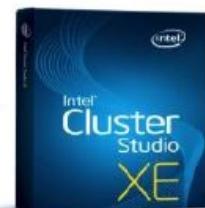
> 1000
Gigaflop double-
precision (peak)



6GB GDDR5
240 GB/s Bandwidth

Programmability

C, C++, Fortran
Intel and 3rd party tools



Intel and 3rd party
tools

Applications

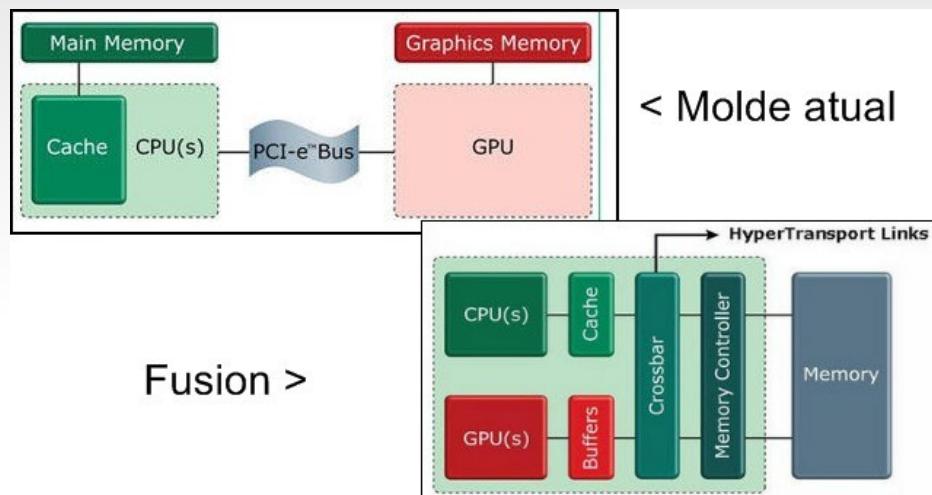
Compute Bound
workloads



Ideal for MonteCarlo,
Black-Sholes, HPL,
Life Sciences



The AMD Fusion™ Family of APUs (Accelerated Processing Unit)



- União de CPU tradicional (x86) com Cores de GPUs (ATI) no mesmo Chip.
- Minimiza tempo gasto para transferir dados da CPU para GPU
- Programação em OpenCL

ARM TEGRA

- Integra processador ARM com GPU NVIDIA no mesmo Chip



and
ARM

- Tegra is a FANTASTIC MID and Mobile Computing solution
- GREAT USER EXPERIENCE
 - high-performance visual computing
 - exceptional battery life

This is just the beginning!

- NVIDIA is also an ARM Cortex™-A9 licensee ... stay tuned



NVIDIA Fermi



GPU	G80	GT200	Fermi
Transistors	681 million	1.4 billion	3.0 billion
CUDA Cores	128	240	512
Double-Precision Floating Point	None	30 FMA ops per clock	256 FMA ops per clock
Single-Precision Floating Point	128 MADD ops per clock	240 MADD ops per clock	512 FMA ops per clock
Warp Schedulers per Streaming Multiprocessor (SM)	1	1	2
Special Function Units per SM	2	2	4
Shared Memory per SM	16KB	16KB	Configurable 48KB or 16KB
L1 Cache per SM	None	None	Configurable 16KB or 48KB
L2 Cache	None	None	768KB
ECC Memory Protection	No	No	Yes
Concurrent Kernels	No	No	Up to 16

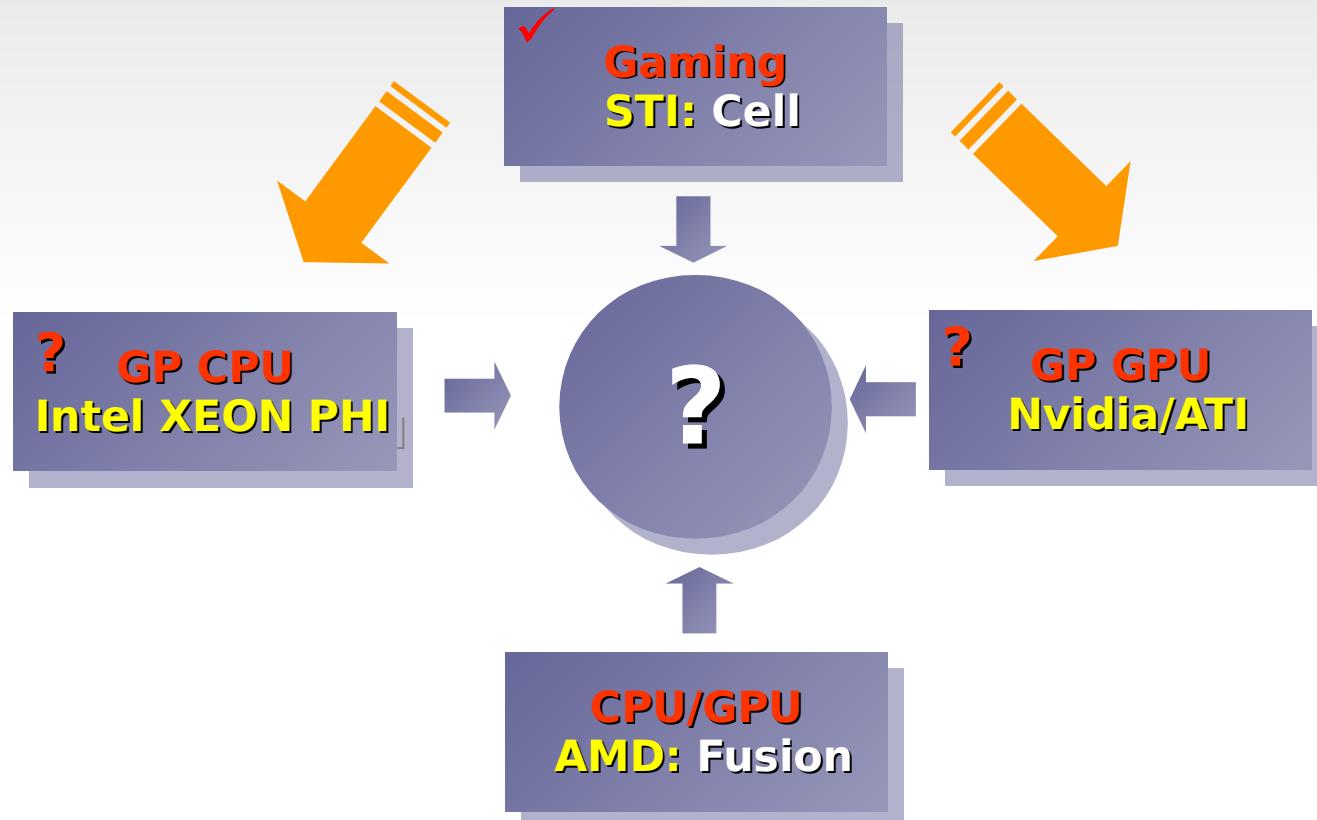
NVIDIA KEPLER



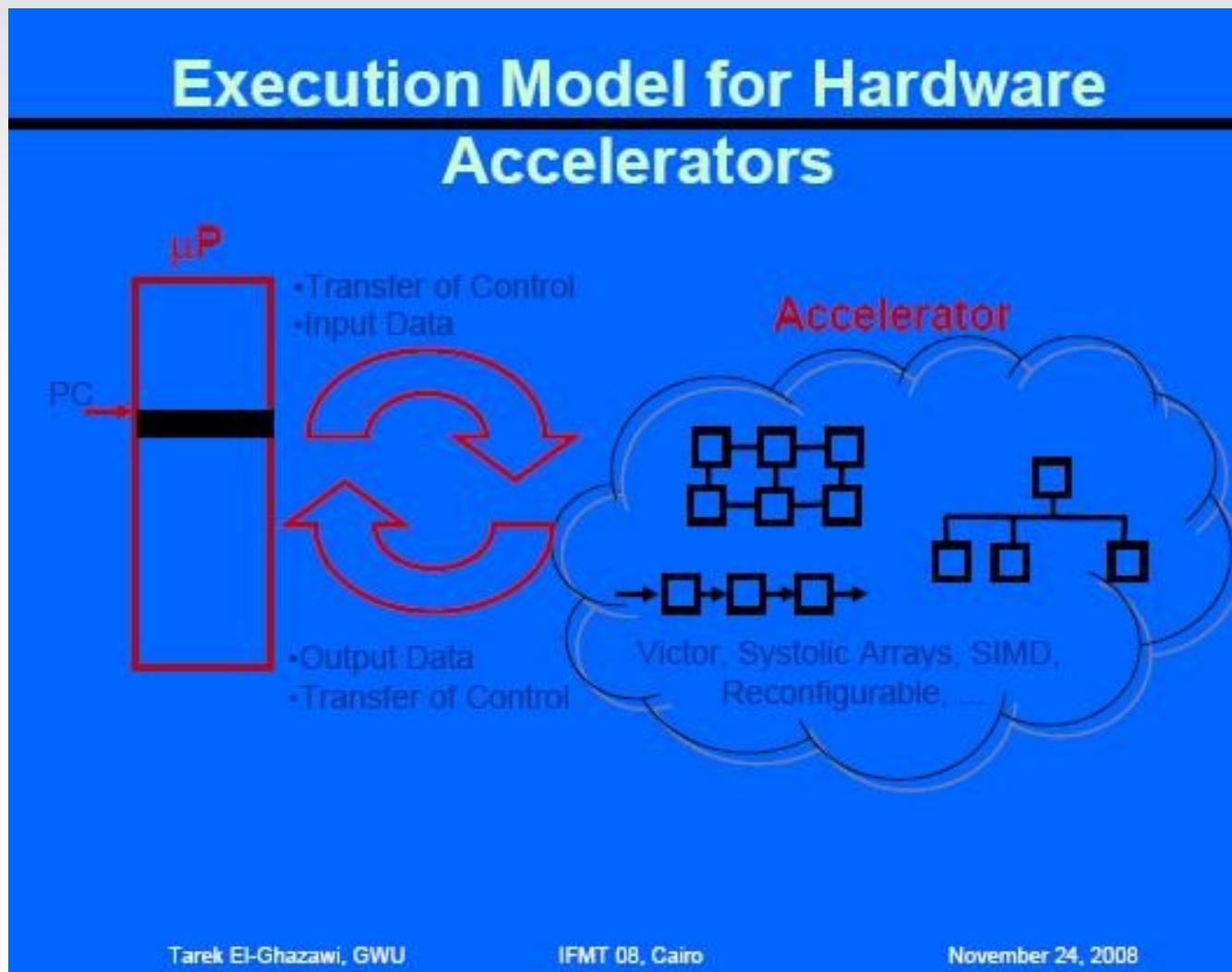
TECHNICAL SPECIFICATIONS	TESLA K10 ^a	TESLA K20	TESLA K20X
Peak double precision floating point performance (board)	0.19 teraflops	1.17 teraflops	1.31 teraflops
Peak single precision floating point performance (board)	4.58 teraflops	3.52 teraflops	3.95 teraflops
Number of GPUs	2 x GK104s	1 x GK110	
Number of CUDA cores	2 x 1536	2496	2688
Memory size per board (GDDR5)	8 GB	5 GB	6 GB
Memory bandwidth for board (ECC off)	320 GBytes/sec	208 GBytes/sec	250 GBytes/sec
GPU computing applications	Seismic, image, signal processing, video analytics	CFD, CAE, financial computing, computational chemistry and physics, data analytics, satellite imaging, weather modeling	
Architecture features	SMX	SMX, Dynamic Parallelism, Hyper-Q	
System	Servers only	Servers and Workstations	Servers only

Futuro do Manycore

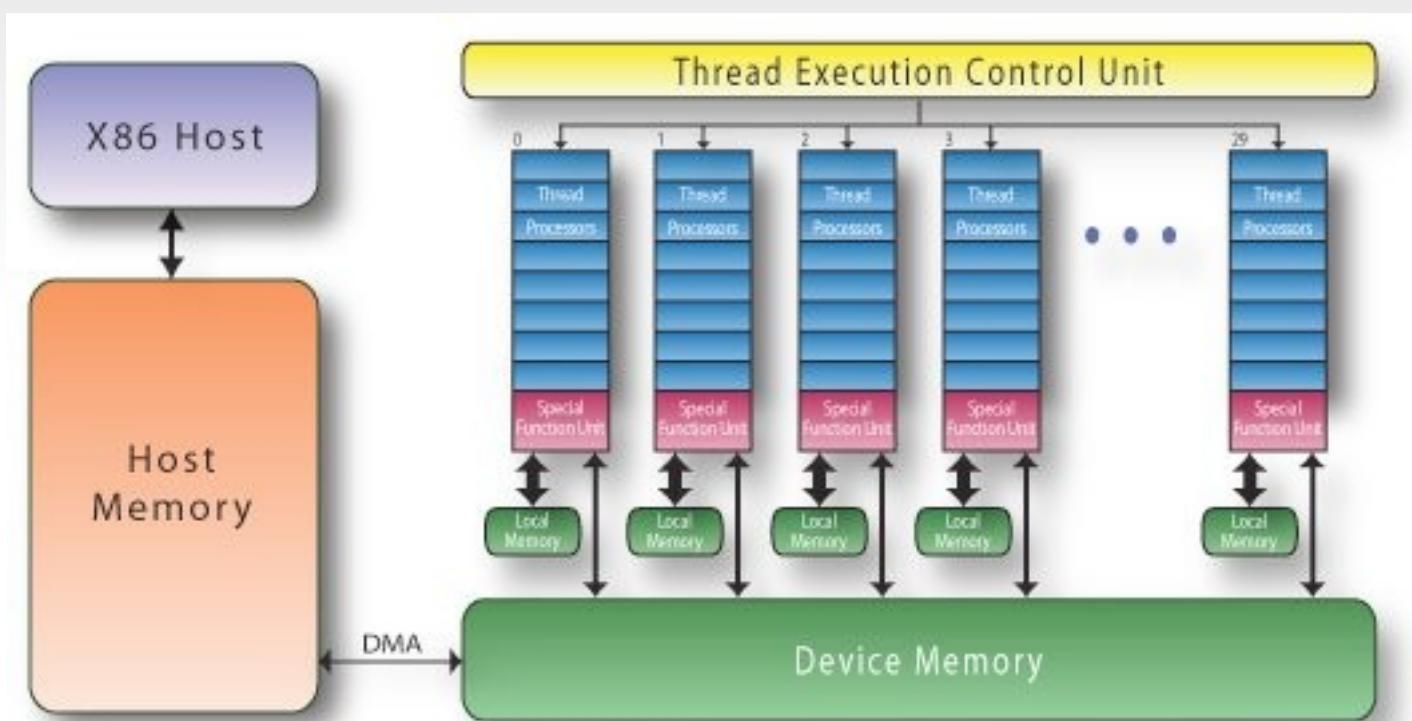
?



Aceleradores de hardware (I)



Aceleradores de hardware (II)

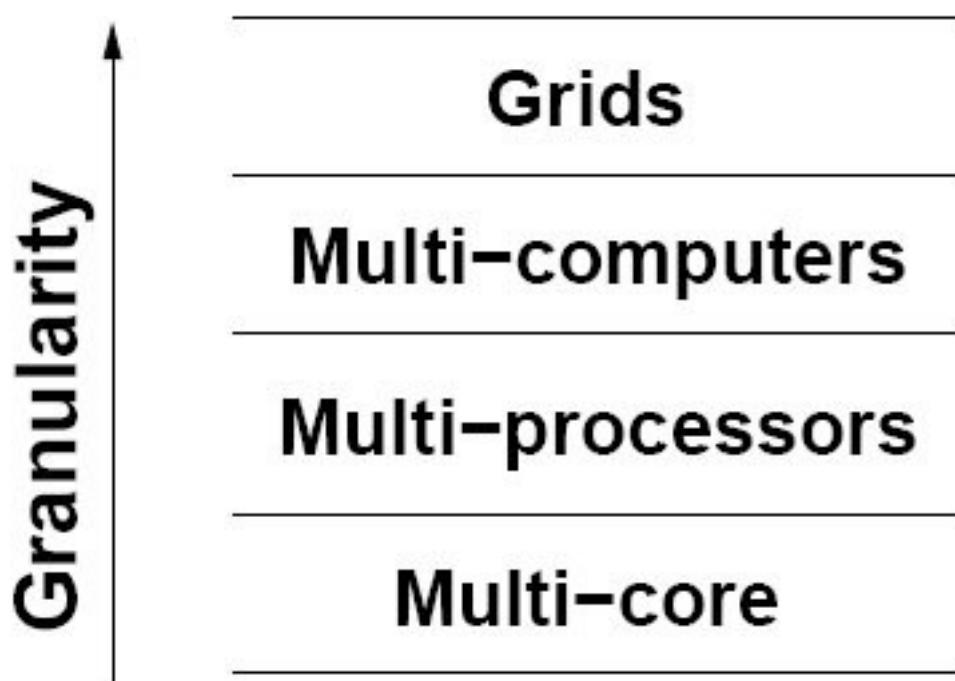


Máquinas híbridas implicam em paralelismo de múltiplos níveis

- Em **todos os níveis**, as arquiteturas provêm **paralelismo** importante.
- Linguagens de programação / algoritmos não são concebidos para aproveitar este paralelismo
 - Algoritmos: o modelo de referência é a máquina de Turing...
 - Modelos paralelos não capturam todos os parâmetros ou são impraticáveis
 - PRAM (memória compartilhada infinita)
 - BSP / CSM (fortemente síncrono)
 - LogP (poucos resultados)
- Só se sabe explorar paralelismo “trivial”
 - Mestre/escravo, Task Farm, ...

Granularidade no paralelismo de múltiplos níveis

Granularidade ou tamanho do grau em paralelismo define a carga de trabalho (*workload*) executada entre nós paralelos em relação ao tempo/carga de comunicação entre eles

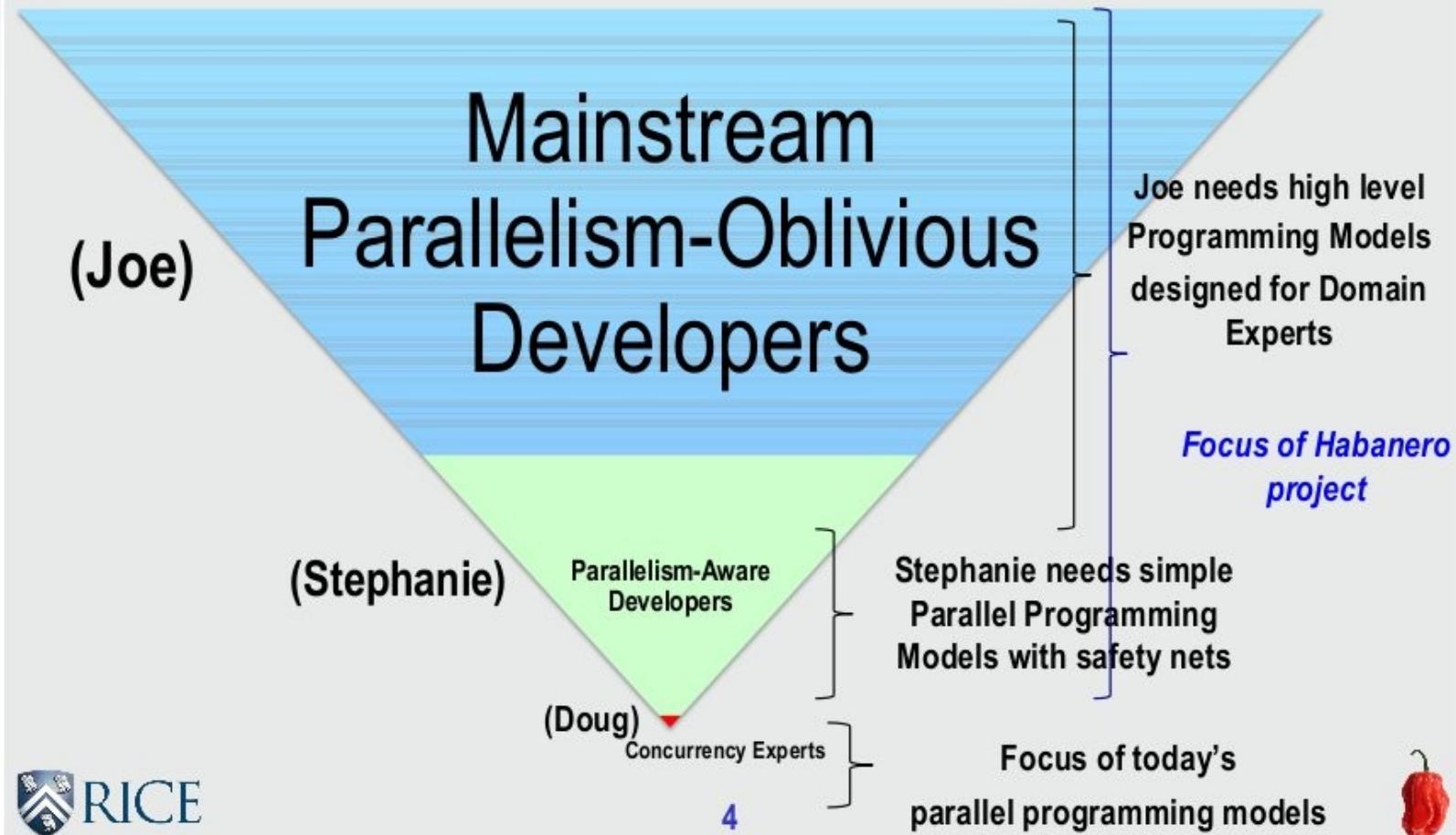


Software??

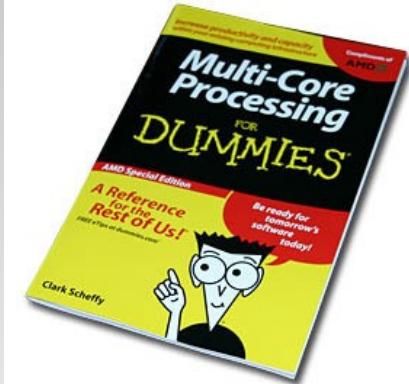
- Pequena parcela de programadores sabem lidar com paralelismo (*Shanghai Manycore Workshop, 2008*)
 - *Harnessing high degrees of thread-level parallelism is the next big challenge in computer systems*
 - *Using traditional CPUs with accelerators (GPUs, Cell, FPGAs, etc..) is the only way to significantly advance future system performance*
- “*CPU/GPU programming skill is the biggest hole they have. They can't find students who can do it well.*” - Prof. Blair MacIntyre

Conhecimentos para programação eficiente

Parallel Software Challenge & Inverted Pyramid of
Parallel Programming Skills



Ponto crucial abordado: Programação



- É necessário gerar paralelismo para utilizar múltiplos “cores” por “chip” com eficiência
 - Uso eficaz deste poder de processamento é desafio central em Ciência da Computação para a próxima década
 - Requer mudar forma tradicional de Programação
 - Atenção nas Particularidades: *multi-core is not “the new SMP”* (J. Dongarra, CT Watch Quarterly, 2009)
 - Compartilhamento de recursos (Cache, TLB, BUS)
 - Iniciativas de estudo existentes:
 - Projeto multi-disciplinar do Depto. Ciência da Computação e Engenharia Elétrica da Universidade da Califórnia em Berkeley
 - S.B.C.: Grandes Desafios – Computação no Brasil 2006-2016

Otimizando desempenho (I): Especial atenção na programação

- 1) Paralelismo em nível de processos, identificando e otimizando “gargalos” nos códigos seriais
- 2) Competição por memória entre seus processos independentes
 - a) Otimizar o acesso a todos os níveis de memória, em especial nas memórias caches (mitigar efeitos de *Memory Wall*)
- 3) Explorar I.L.P.
 - a) Vetorização de instruções em nível de hardware (Intel SSE e AMD 3DNow)

Otimizando desempenho (II): Otimização fina do código

- Ajuste fino de opções de compilação
- Instrumentação do código
 - Utilização de ferramentas de análise de desempenho:
 - Intel VTUNE Performance Analyzer, Intel Thread Profiler, Intel Thread Checker, TotalView, MPIP, PAPI, Oprofile, etc.,
 - PAPI e Oprofile, constituem-se de APIs que permitem inspeção aos chamados contadores especiais de hardware (“Hardware Counters”), presentes na maioria dos processadores citados
 - Manter reprodutibilidade na versão otimizada (“*Correctness*”)

Modelos de Prog. paralela para memória compartilhada

- Modelo Fork-Join;
- Threads: POSIX Pthreads, Java Threads, Win 32 threads, .NET System threading, etc.;
- Diretivas aplicadas a linguagem de programação comum: OpenMP, CAPS HMPP, PGI Accelerator Programming Model;
- Funções de biblioteca: “*Intel Thread Building Blocks*”;
- Funções de biblioteca paralela para operações matemáticas comumente utilizadas em aplicações científicas: Intel MKL, LAPACK, FFTW, etc.;
- Linguagem de programação paralela: HPF - “*High Performance Fortran*”, CHARM, Cilk, Intel Ct;
 - Modelo PGAS (Partitioned Global Address Space): Chapel, Fortress, X10, CAF – “Coarray Fortran”, UPC – “Unified Parallel C”, Titanium - Java Dialect for High Performance Computing
 - Streamming Programming: *OpenCL*
 - *CPU off loading*: CUDA C, CUDA Fortran, BROOK+
- Paralelizações automáticas
- MPI com otimizações para memória compartilhada
- TRANSACTIONAL MEMORY (STM ou HTM)

Possível modelo de computação para multicore

- Características desejáveis:
 - Desempenho: escalável para milhares de processadores
 - Alta Produtividade
 - Tratamento dinâmico do comportamento da aplicação
- Virtualização com migração de threads
 - Extrapolar número de processos
 - Mapeamento dinâmico