

De 1986 até 2002 o desempenho dos microprocessadores aumentava na média 50% a cada ano.
Causas?

- Aumento da velocidade do relógio (mais ciclos no mesmo tempo), otimizações no fluxo de execução (mais trabalho por ciclo): pipelining, etc
- Memórias e HDs mais rápidos

Novo estilo de programação: se precisar mais desempenho vá pra praia e volte daqui a meio ano!

Demanda constante por hardware mais rápido: Aplicações mais ambiciosas ou simplesmente software menos eficiente.

Cada vez mais difícil aumentar a velocidade de relógio: aquecimento, consumo de energia (Power Wall). No outono de 2004 Intel abandonou oficialmente os planos de produzir um chip a 4GHz

Em lugar de aumentar a velocidade de relógio, os fabricantes começaram a colocar múltiplos processadores em um mesmo circuito integrado.

O caminho para o aumento do desempenho está na direção do paralelismo.

Computação paralela é uma forma de computação em que vários cálculos são realizados ao mesmo tempo, operando sob o princípio de que grandes problemas geralmente podem ser divididos em problemas menores, que então são resolvidos concorrentemente.

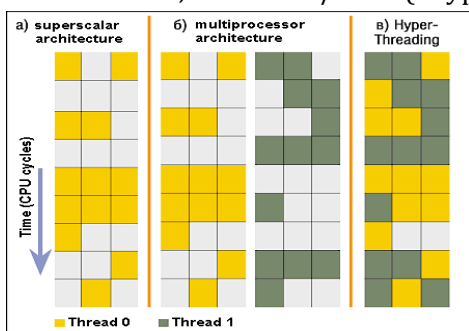


The Free Lunch Is Over: A Fundamental Turn Toward Concurrency in Software

<http://www.gotw.ca/publications/concurrency-ddj.htm>

Simultaneous MultiThreading/Hyperthreading: Técnica de Hardware

- Processadores Superscalares podem executar múltiplas instruções independentes. O hyperthreading duplica os registros para manter uma segunda thread executando. O speedup depende da aplicação.
- Hyperthreading (termo Intel), **Simultaneous multithreading**, ou SMT, permite a uma CPU executar múltiplos fluxos de controle (hardware).
- Implica em múltiplas cópias dos registros e o controlador de interrupções, compartilhando o restante do hardware.
- Um processador hyperthreaded pode decidir a troca de fluxo a cada ciclo.
- Melhor uso dos recursos **em alguns casos**: ex. thread bloqueada esperando por dados.
- Melhora do desempenho depende da situação mas deve ser menor que o 25% anunciado.
- SPARC, Pentium 4/Xeon ("Hyperthreading"), Power 5.



A maior parte dos SOs modernos são multitarefas: suportam a execução (aparentemente simultânea) de múltiplos programas. Sistemas mono processados: timeslicing (cada processo executa durante uma fatia de tempo -> multiplex de tempo)

**Precisamos de programas paralelos. É possível executar programas seriais em paralelo?
Aplicações precisam ser reprogramadas para aproveitar as novas plataformas!!**

- ▣ Paralelismo de Tarefas

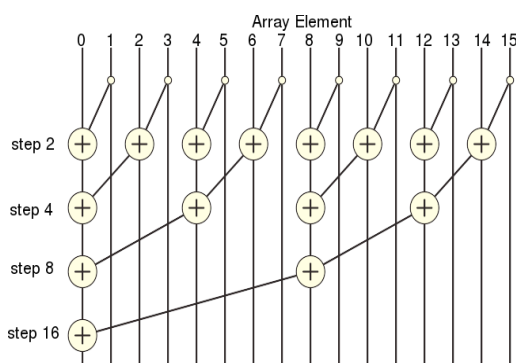
- ▣ Particiona as tarefas entre os cores para resolver o problema.

- ▣ Paralelismo de Dados

- ▣ Particiona os dados entre os cores para resolver o problema.

Cada core executa operações similares em sua porção dos dados.

Precisamos pensar em paralelo! Eliminar o modelo mental em que a máquina executa as instruções sequencialmente.



Nem todos os problemas são inerentemente paralelizáveis

Pró: 9 grávidas Vs. 1 para formar um time de baseball

Contra: 9 grávidas durante 1 mes não dão a luz uma criança. (*The Mythical Man-Month: Essays on Software Engineering*)

Problemas? Programas seriais são determinísticos: eles sempre retornam a mesma resposta para a mesma entrada. Programas paralelos?

Bibliografia

An Introduction to Parallel Programming, Peter Pacheco

Structured Parallel Programming: Patterns for Efficient Computation. [James Reinders](#), [Arch Robison](#), [Michael Mccool](#)

Computer Systems: A Programmer's Perspective (3da Edição) [Randal E. Bryant](#), [David R. O'Hallaron](#)

Programming Massively Parallel Processors: A Hands-on Approach, 2nd Edition [David B. Kirk](#), [Wen-mei W. Hwu](#)
[+artigos, etc](#)