How much energy can you save with a multicore computer for web applications?

Peter Strazdins
Computer Systems Group,
Department of Computer Science,
The Australian National University

seminar at
Green ICT 2008: Learning to be Sustainable with Computers and Broadband

(slides available from http://cs.anu.edu.au/~Peter.Strazdins/seminars)

1 Overview

- background in modern processor design
  - instruction level parallelism,
  - shared memory and multicore processors
- characteristics of web server applications
- traditional server design
- multicore/multithread server design (T2)
  - ANU's T2 configuration
- comparison on SPECWeb benchmarks
- discussion
- virtualization technology for server consolidation
- conclusions

2 Background: Pipelining (Instruction-Level Parallelism)

- (within a single thread) the execution of each instructions is broken into a k stages $\Rightarrow$ can get $\leq k$-way parallelism
- generally, the circuitry for each stage is independent
- e.g. ($k = 5$): stages FI = Fetch Instr'n., DI = Decode Instr'n., FO = Fetch Operand, EX = Execute Instr'n., WB = Write Back

(branch):	FI	DI	FO	EX	WB				
(delay slot:)		FI	DI	FO	EX	WB			
(guess)			FI	DI	FO	EX	WB		
(guess)				FI	DI	FO	EX	WB	
(sure)					FI	DI	FO	EX	WB

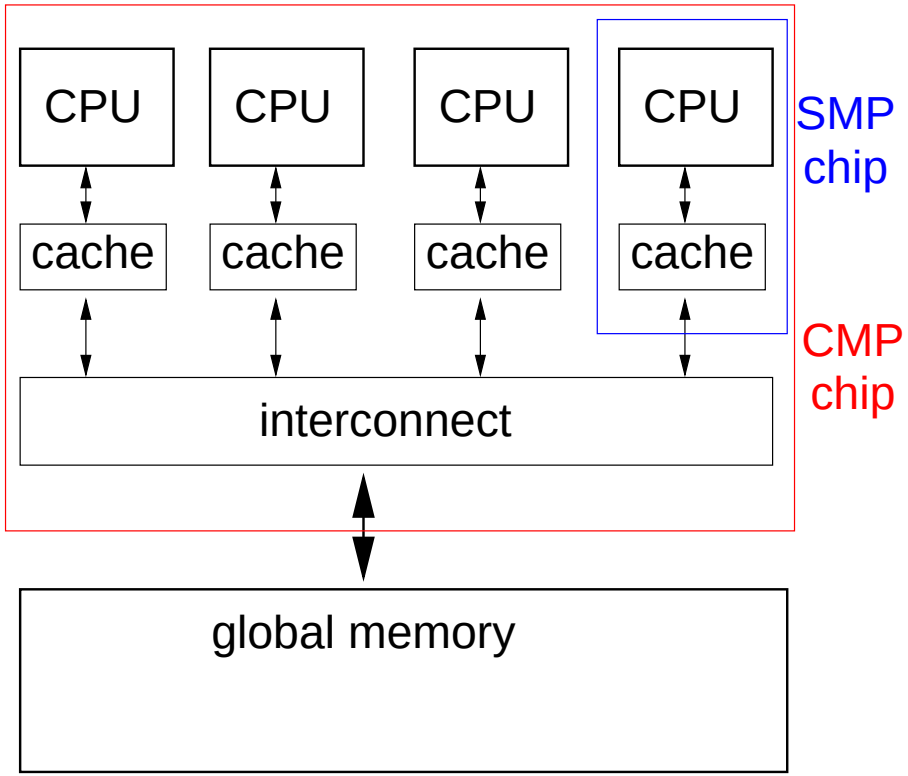
- dependencies between instructions, memory accesses, mispredicted branch instructions may stall the pipeline
- modern processors have $k \geq 13$
- we have reached the end of the line to increasing clock speed! (by this or any other technique)

3 Background: Superscalar Execution (also ILP)

- (within a single thread), issue a (small) group of (w) instructions in a single cycle
- each group can be made up of different types of instructions

  e.g. UltraSPARC III ($w = 4$): $\left\{ \begin{array}{l} \leq 2 \text{ different floating point} \\ \leq 1 \text{ load / store} \ ; \leq 1 \text{ branch} \\ \leq 2 \text{ integer / logical} \end{array} \right\}$ instr'ns
- have $\leq w$-way parallelism over *different* types of instr'ns
  - generally, the circuitry for (EX stage) for each type of instruction is independent
- with pipelining, achieves $\leq kw$-way parallelism but amplifies problem of dependencies between instructions
  - solution: out-of-order execution: buffer large numbers of instructions well in advance, form groups in any order that do not violate dependencies
  - very complex (expensive), and how often is this worthwhile?

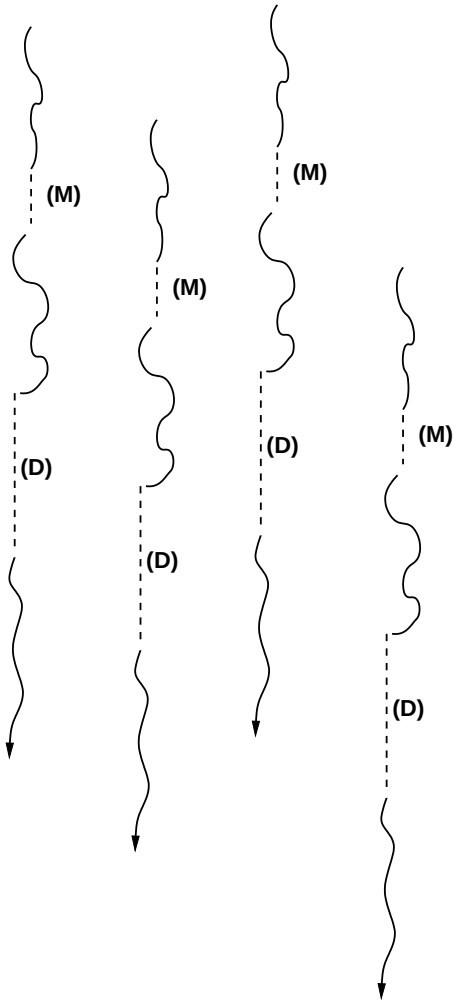
4 Background: Shared Memory and Chip Multiprocessing



- multiple CPUs can execute multiple programs in parallel
- to speed up memory accesses, cache memory is needed
- SMP (Symmetric Multiprocessor): time/power of interconnect much greater!
- (CMP) multicore processor generally has simpler CPUs

- a single application can have multiple threads of execution on each CPU
- each such unit generally needs own disk(s) and network interfaces
- to gain more processing power, can have clusters of these connected via (fast) Ethernet

5 Web Server Applications

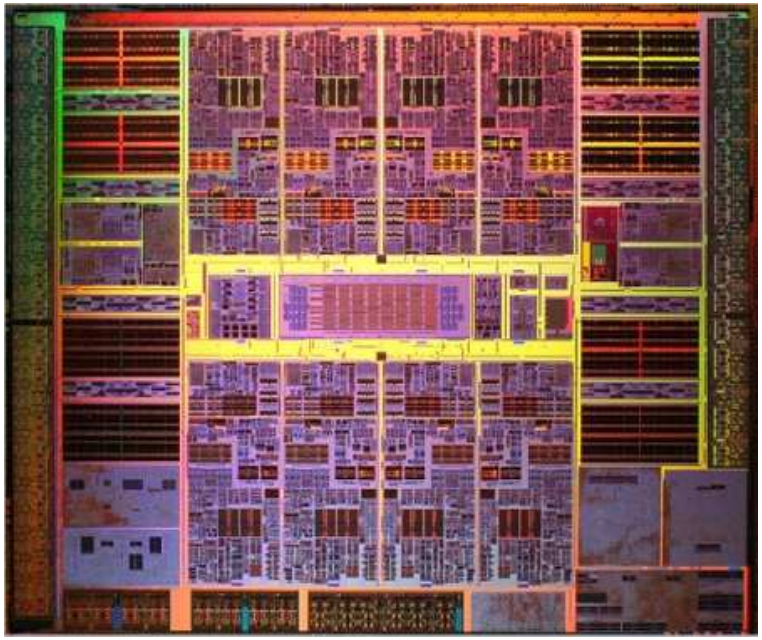


- each client has one (or more) threads of execution
- potentially very many threads at once (under heavy load)
- these threads seldom interact (update same part of server's data)
- each thread has frequent stalls (memory and disk/network accesses)
- each thread has very limited instruction level parallelism
  - mainly integer and memory access instructions
  - high degree of dependencies; unpredictable branches
- speed can however come from trying to execute many threads (pseudo-) simultaneously!

6 Traditional Server Design

- made up of general-purpose commodity processors
  - scientific applications *can* benefit from high clock frequency, ILP, branch prediction, multiple floating point units
  - mostly wasted on web server apps – still consumes power!!
    - power usage increases quadratically with frequency . . .
  - generally, cannot tolerate well delays (memory access)
- more powerful systems are usually made of small-to-medium scale SMPs, possibly connected into clusters
  - expensive interconnects (SMP and cluster levels)
  - requires more disks than really needed
- traditional software requires one logical server to be implemented in one physical system
  - must have enough power for peak demand! (only 5% of the time)

7 Multicore/Multithread Server Design: UltraSPARC T2



- 8 cores, with 8-way hardware threading
  - enables (pseudo-) parallel execution of 64 threads!
- each core has a simple CPU (shallow pipeline, no superscalar), with low clock frequency
- shared memory interconnect on chip
- worst case power usage 84W (1.4 GHz)
- CoolThreads:low cooling requirements
- typical system breakdown:
  T2 chip 25%, I/O 22%, disk 4%, DRAM 22%, AC/DC 15%, fans 10%
- power throttling: can turn cores, DRAM on/off
- system fits within a single rack unit
- hardware support for virtualization

8 ANU's T2 Configuration

- recently donated T5120 UltraSPARC T2 processor (Niagara-2), called mavericks
- exported (via a logical domains virtualization software) to the CS student systems, as wallaman
  - wallaman uses 1 of the 3 network interfaces, 1 of the 2 disks
  - isolates research & teaching uses; needed for security!
- integrated into the Computer Systems teaching program
  - first use in COMP2310 Concurrent & Distributed Systems, 10/08
  - basis of practical work in COMP8320 Multicore Computing, for S2'09

9 Performance (Power) for SPECweb2005 Benchmark

- CoolThreads T5220 model: 1.4 GHz, T2+ chip, 64GB RAM, 2 × 146GB (+ 4 × StorageTek 3510) disks, 2(+2) network cards, 1 PCIe card
  - SPECweb2005 rating: 41847
  - 227W idle / 426W loaded (SPECjbb2005, rating = 192055)
    - from Sun's Power Calculator
- Sun Fire X4240: 2 × 4-core Opterons, similar system
  - SPECweb2005 rating: 32288
  - 231W idle / 411W loaded (SPECjbb2005, rating = 372467)
- IBM Power-5 550, 2 × 2 core 1.9 GHz Power-5 chips (post-RISC)
  - SPECweb2005 rating: 7881 (would 4 GHz be 2× better?)
- Dell PowerEdge 2950: 2 × 2 core 3GHz Xeon 5160
  - SPECweb2005 rating: 14495
- HP ProLiant DL585 G5 4 × 2 core 3GHz Xeon 5160
  - SPECweb2005 rating: 43854

10 Web Server Performance: Further Remarks

- SPECweb2005 results:
  - made up of banking, e-commerce and support sub-benchmarks
  - requires complex configurations, heavily subject to HW/SW tuning
- Sun claims for T5240 (latest 2-chip system) for SPECjbb and AppServer:
  - 2–4 $\times$ performance
  - 2-5 $\times$ performance / watt
  - 2–4 $\times$ space savings

  over ‘comparable’ competitor servers (x86 / IA-64 / post-RISC based chips)

11 What was Virtualization again?

- can put several logical servers (operating systems) on a single physical system
- allocate resources (CPUs, memory) to each; can vary dynamically
- typically share disks and network interfaces
  - each virtual machine will be given a different 'partition'

- T2 via logical domains can support up to 64!
  - H/W support reduces overhead
- each logical server can be for a different application / client
  - high degree of security / isolation possible
- many other flavors of virtualization available (e.g. Xen, VMware, KVM)

12 Conclusions and Outlook

- the days of the gas-guzzling SMPs are over! (IA-64?)
  - revenge of the CMPs/SMPs over small-scale clusters
- the question: moderate multicore/multithread vs aggressive
  - aggressive *ought* to have large performance /watt advantages . . .
    - *never* underestimate the engineering in x86!
  - watch out for multicore with transactional memory!
- the T2 systems arguably represent the cutting edge for green servers
  - especially well-suited to web applications

  but all other vendors are following close behind!