

A few comments on Measurements

Outline

- *The Basic Equation (and what is wrong with it)*
- *How do we measure*
 - *Real Hardware, Simulator, Analytic model*
 - *Hardware instrumentation, Microcode, Software monitoring*
- *What do we measure (i.e., benchmarks)*
 - *Synthetic code*
 - *Kernels*
 - *Toy Benchmarks*
 - *SPEC*
 - *The Perfect Club*
 - *Your relevant workload*
- *Serious Abuses*

Why do we measure?

- ***Before the fact***
 - ***So we will know what to build***
- ***After the fact***
 - ***So we will know what to build next time***

The Standard Performance Equation

$$***T = L \times CPI \times t***$$

L = no. of instructions executed (ISA)

CPI = Cycles per instruction (ISA, Microarchitecture)

- ***Pipelining, Issue rate, branch handling***

t = Clock (technology, marketing)

How do we measure? (Degree of Sanitizing)

- ***Real Hardware***
 - *Gotchas have a chance to get in the way (a good thing!)*
 - *Least flexible*
 - *Fast for doing a thorough job*
- ***Simulation***
 - *Some effects are missing*
 - *Most flexible*
 - *Slowest*
- ***Analytic Model***
 - *Good for gross effects*
 - *Must be validated*

How do we measure (Invasiveness)

- ***Hardware instrumentation***
 - ***Most expensive***
 - ***Non-invasive***
 - ***Least flexible***
- ***Microcoded instrumentation***
 - ***Best of both worlds (e.g., performance counters, SSMT)***
 - ***SPAM***
- ***Software monitoring***
 - ***Cheap***
 - ***Very invasive***
 - ***Most flexible***

Benchmarks

- ***Why benchmarks?***
 - ***Find a set of programs or program fragments
REPRESENTATIVE of the WORKLOAD you need the machine for***
- ***Types***
 - ***The ADD instruction***
 - ***Instruction MIX (Gibson Mix, 1959)***
 - ***Kernels (Livermore Loops, Berkeley's 13 dwarfs)***
 - ***Synthetic Benchmars (Allows parameterization, but RRW is not RWR)***
 - ***Toy benchmarks (easy to hand-compile, pretty much in dispute today)***
 - ***SPEC (Systems Performance Evaluation Co-operative), Agreement!***
 - ***Real workload***

A few of my concerns

- *One number: SpecMARK (Better than ADD time)*
- *SimplScalar (the bar to entry, bugs)*
- *In the literature (1.85 IPC max, Issue width does not matter)*
- *400 floating point ops or 1 LLC miss*
- *Power models*

BAD ways to measure performance (...and each has been used and published)

- ***Apples and Oranges***
 - ***RISC A lightly loaded VAX vs. Counting Simulated Cycles***
- ***Who should get the credit***
 - ***The architecture or the compiler (Berkeley Pascal or VMS Pascal)***
 - ***Algorithm optimizations (disallowed by SPEC, determin concat)***
 - ***Instruction set or Register Windows (Bob Colwell)***
- ***Choice of Benchmarks***
 - ***Overstates significance of a feature (procedure call, no floating point)***
 - ***Small size (100% fits in cache, TLB hits, no I/O)***

BAD ways to measure performance (continued)

- ***Play with Statistics (Which machine is better)***

	<i>Program A</i>	<i>Program B</i>
<i>Machine 1:</i>	<i>1 unit</i>	<i>2 units</i>
<i>Machine 2:</i>	<i>2 units</i>	<i>1 unit</i>

Machine 1 is twice as fast on A, half as fast on B

Speedup is $\frac{1}{2} (2 + \frac{1}{2}) = 1.25$