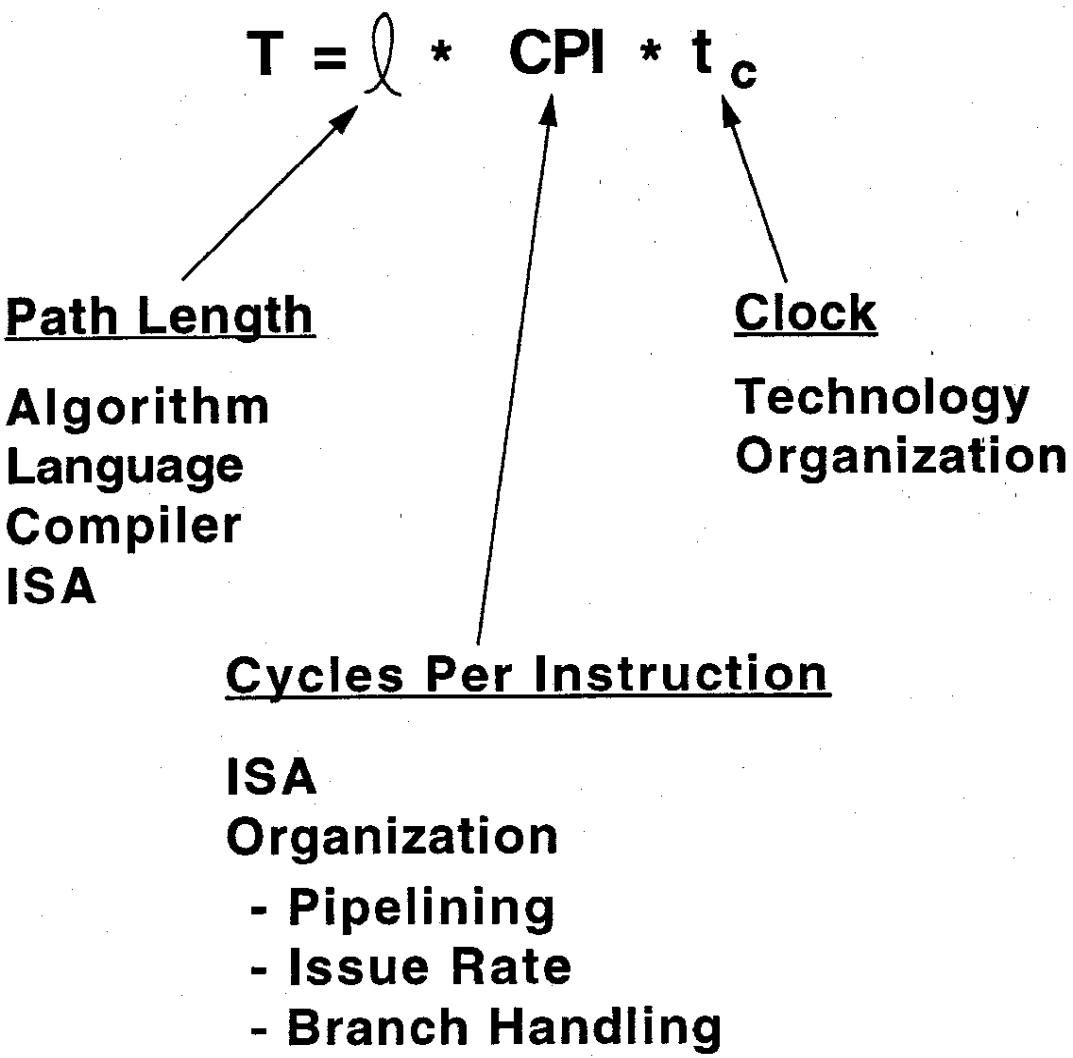
The Standard Performance Equation



Means

* **Arithmetic Mean**

$$A = \frac{1}{n} \sum_{i=1}^{n} P_i$$

* **Geometric Mean**

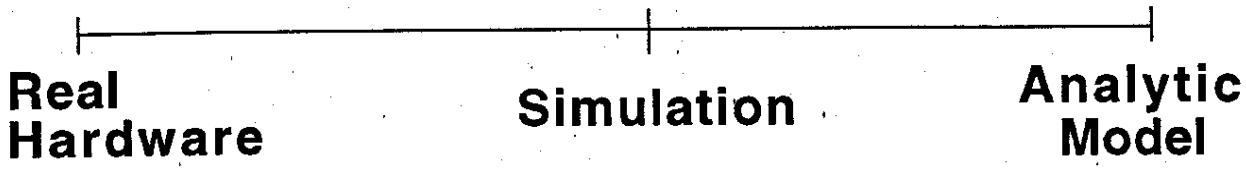
$$G = \sqrt[n]{\prod_{i=1}^{n} P_i}$$

* **Harmonic Mean**

$$H = \frac{1}{\frac{1}{n} \sum_{i=1}^{n} \frac{1}{P_i}}$$

How Do We Measure

Degree of Santizing



Real Hardware

- "Gotchas" Have a chance to get in the way
- Least Flexible
- Fast for doing thorough job

Simulation

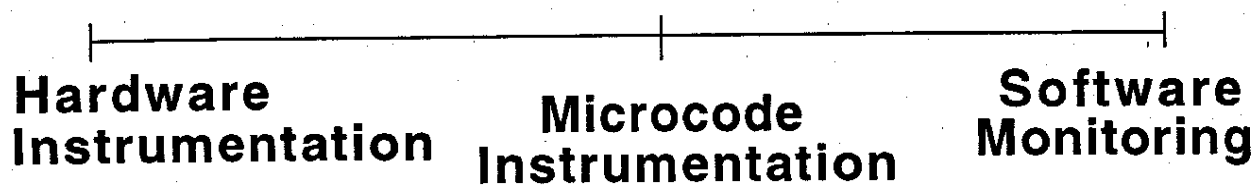
- Some effects are missing
- Most Flexible
- Slowest

Analytic Model

- Good for gross effects
- Must be validated

How Do We Measure (Continued)

Invasiveness



Hardware Instrumentation

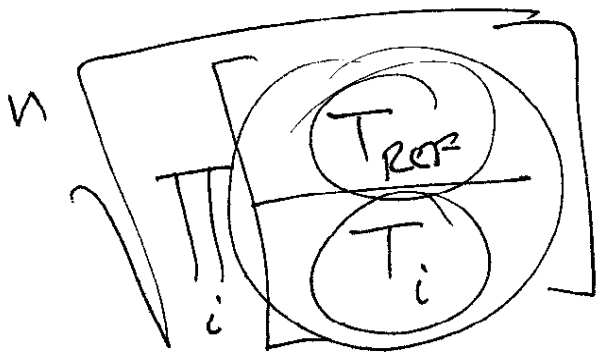
- Most Expensive
- Non-Invasive
- Least Flexible

Microcoded Instrumentation

- Best of Both Worlds
- SPAM

Software Monitoring

- Cheap
- Very Invasive
- Most Flexible



~~VAX 11-780~~

?

GEOMETRIC

~~MEAN~~

SPEC mark

Benchmarks

Rationale: Find a set of programs or program fragments representative of the workload you will be requiring of the machine

Types:

1. The ADD instruction - very old
2. Instruction MIX - Old (Gibson MIX, 1959)
3. Kernels
   - e.g., Livermore Loops
4. Synthetic Benchmarks
   - Parameterized
   - Careful: RRW is not RWR
5. Toy Benchmarks
   - Easy to hand-compile
   - Pretty much in disrepute today
     e.g., Towers of Hanoi
6. SPEC Suite (Systems Performance Evaluation Co-operative)
   - At least common agreement, I Guess!!
7. Real Workload

Bad Ways to Measure Performance (... and each has been used and reported in the Open Literature)

* **Apples & Oranges**
  - A Lightly Loaded VAX vs. Counting Simulated Cycles
* **Who Gets the Credit**
  - The Architecture or the Compiler
  - Example: Berkeley Pascal vs VMS Pascal
  - Algorithm Optimizations
  - Instruction set or register windows (Colwell)
* **Choice on Benchmarks**
  - Selective
    * Overstates significance of one feature
      e.g. Regularity (Fl. Pt.)
      e.g. Procedure Call Intensive
      e.g. No Floating Point
  - Small
    * 100% Cache, TB Hits
    * No I/O, Context Switch

★ Play with Statistics

	Program A	Program B
Machine 1:	1 unit	2 units
Machine 2:	2 units	1 unit

Machine 1 is $\frac{2}{1}$ on A, $\frac{1}{2}$ on B

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

★ Too Focused on Frequency

	Frequency	Execution Time
Calls	2.5%	21.6%
MOVL	12.4%	6.8%