ECE560: Computer Systems Performance Evaluation

Lecture #2 –

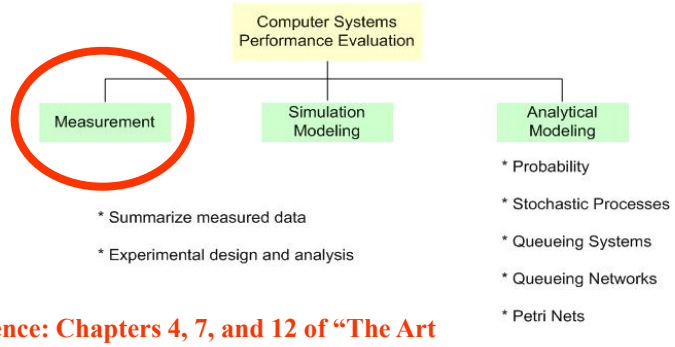
Measurement Techniques & Tools

Instructor: Dr. Liudong Xing

Administration Issues (1/24)

- Last day to add/drop: **January 29, Monday**
- Project team setup by **January 29, Monday**
  - Two students per team

Today's Topic



Reference: Chapters 4, 7, and 12 of "The Art of Computer Systems Performance Analysis", by R. Jain

Measurement Technique

- Designing an experiment and performing the experiment
- Measuring performance parameters while the system is being subject to a particular workload
- Data analyzed by statistical techniques to draw meaningful conclusions

Key Issues in Measurement Technique

- **Types of Workloads**
- Performance Monitoring
- Summarizing Measured Data
- Experimental Design (L#3)

Types of Workloads (1)

- Workload: what is stressing the system
- Test workload: any workload used in performance studies
  - Addition instruction
  - Instruction mixes
  - Kernels
  - Synthetic programs
  - Application benchmarks

Addition Instruction

- Performance of CS was synonymous with that of processor, historically.
- Addition instruction is the most frequently-used among few instructions, initially
- faster addition ≈ better performance
- Metric: addition time

Instruction Mixes (IM)

- Measuring relative frequencies of various instructions and using them as weighting factors to get an average instruction time become necessary!
- Specification of various instructions coupled with their usage frequency
- Metric: average instruction time
- Measuring performance of processor, which may or may not reflect the entire system performance!

Processing Kernels

- A generalization of IM
- A function or a service (consisting of a set of instructions) provided by CPU
- Commonly used for specialized applications: sorting, matrix inversion, tree searching, puzzle

Synthetic Programs (Exerciser Loop)

- Can measure performance of I/O and OS by making a specified number of system calls and I/O requests
- Advantages
  - Can be quickly developed
  - Require no real data files that may contain proprietary information
  - Can be easily modified by adjusting control parameters
  - Can build in measurement capabilities so that the measurement process is automated
- An example (refer to Relevant Reading on website)

Application Benchmarks

- A representative subset of functions for a particular application (e.g. banking)
- Make use of almost all the resources

- Benchmark Examples from SPEC (Standard Performance Evaluation Corporation)

http://www.spec.org/benchmarks.html

A Note

- Benchmarks
  - The workloads used in the measurements (5 types)

- Benchmarking
  - The process of performance comparison for two or more systems by measurements

Agenda

Key issues in measurements:

- ✓ Types of Workloads
- **Performance Monitoring**
- Summarizing Measured Data
- Experiment Design (L#3)

Use/Goals of Monitors

- To find frequently-used segments of SW and optimize their performance
- To measure resource utilizations and to find performance bottleneck
- To tune the system, etc.

Monitor Terminology

- Event: a change in system state
- Trace: a log of events (time, type…)
- Overhead: consumption of system resources (storage, CPU time).
- Domain: set of activities observable by the monitor
- Input rate: maximum frequency of events that a monitor can correctly observe
- Input width: number of bits of information recorded on a event
- Resolution: coarseness of information observed

Monitoring Techniques (1)

- Concerning mechanism that triggers the monitor into action
  - **Event-driven**: activated only by occurrence of certain events
  - **Timer-driven**: activated at fixed time intervals. Sampling frequency is determined by event frequency and desired resolution
- Concerning the result displaying ability
  - **On-line**: display result either continuously or at frequent intervals
  - **Batch**: collect data that can be analyzed later using a separate analysis program

Monitoring Techniques (2)

- Concerning the level at which a monitor is implemented
  - Hardware
  - Software

Hardware Monitoring Technique

- Employ additional monitoring hw (counters, timers) that is interfaced with system in a non-intrusive way
- Pros & cons
  - Without interference with normal functioning of monitored system
  - Can capture fast events
  - High resolution (10ns)
  - Overhead is little
  - Expensive
  - Low flexibility
- Appropriate for measuring device utilization, cache hit rate etc.

Software Monitoring Technique

- Use measurement code
  - embedded in existing software or as a separate set of routines
- Pros & cons
  - Seriously interfere with normal functioning of monitored system
  - Cannot capture fast occurring events
  - Resolution is lower than hw (10-16ms)
  - Overhead is usually high
  - Cost is lower than hw
  - flexibility
- Appropriate for measuring OS/user program related information (time spent executing a particular routine)
- A software monitor example in Relevant Reading

Agenda

Key issues in measurements:

✓ Types of Workloads

✓ Performance Monitoring

- **Summarizing Measured Data**
- Experiment Design (L#3)

Summarizing Measured Data

- Due to the randomness of the workloads/inputs, and thus the outputs, no single observation from the system would give us a reliable indication of system performance.
- Usually multiple observations (several hundred or even millions)
- Some statistical ways are needed to summarize the data

Summarizing Data By a Single Number (1)

- An average of the data, representative of a major part of the data set
- Called "**indices of central tendencies**" by statisticians
- Measures: sample (arithmetic) mean, sample median, sample mode, geometric mean, etc.

Summarizing Data By a Single Number (2)

Given a sample $\{x_1, x_2, \ldots, x_n\}$ of n observations

- **Sample Mean:**

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Summarizing Data By a Single Number (3)

Given a sample $\{x_1, x_2, \ldots, x_n\}$ of n observations

- **Sample Median**: x_i is the ith observation in the sorted set (in an increasing order).

$$\text{Sample median} = \begin{cases} \frac{1}{2}(x_{\frac{n}{2}} + x_{\frac{n}{2}+1}) & \text{n is even} \\ x_{\frac{n+1}{2}} & \text{n is odd} \end{cases}$$

- **Example:** find the median for the sample of {1.3, 1.39, 1.37, 1.41, 1.1, 1.42}

Summarizing Data By a Single Number (4)

- **Sample Mode:** the observation that occurs most often
  - Usually for describing "Categorical data"
  - Example: most used resource in a system
  - Unlike mean and median, mode of a sample may not exist, and if existing, it is not necessarily unique!

Summarizing Data By a Single Number (5)

- **Geometric Mean:**
  - Used when the product of the observations is a quantity of interest
  - Examples (working in a multiplicative manner):
    - Cache hit/miss ratios over several levels of caches
    - Percentage performance improvement between successive versions
    - Average error rate per hop on a multi-hop path in a network

$$\dot{x} = (\prod_{i=1}^{n} x_i)^{1/n}$$

Summarizing Variability (1)

- Given a data set, summarizing it by a single average number is rarely enough!
- Given two systems with the same mean performance, one would prefer one whose performance doesn't vary much from the mean (low variability!)
- Called "**indices of dispersion**" by statisticians
- Measures: range, sample variance, sample standard deviation, sample coefficient of variation (COV), percentiles, etc.

Summarizing Variability (2)

Given a sample $\{x_1, x_2, \ldots, x_n\}$ of n observations

- **Range of a sample:** difference between the maximum and the minimum
- **Sample Variance:**

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 \quad \text{where,} \quad \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- **Sample Standard Deviation:** the square root of sample variance

Summarizing Variability (3)

- **Sample Coefficient of Variation (COV):**

$$COV = \frac{s}{x}$$

- **Percentiles:** the pth percentile observation of a sample is the number such that p% of the values do not exceed it and (100-p)% are greater.
  - Estimated by sorting the observations and taking the [(n-1)*p/100+1]th element in the ordered set
  - [ ]: rounding to the nearest integer.
  - Q_1: the 25th percentile or lower quartile
  - Q_2: the 50th percentile or middle quartile
  - Q_3: the 75th percentile or upper quartile

Summarizing Variability (4)

- **Interquartile Range:** the range between Q_3 and Q_1, i.e.,

$$Q_3 - Q_1 = x_{0.75} - x_{0.25}$$

- **Semi-Interquartile Range (SIQR):**

$$\frac{Q_3 - Q_1}{2} = \frac{x_{0.75} - x_{0.25}}{2}$$

- **Example:** In an experiment which was repeated 7 times, the measured CPU time was found to be {2.7, 1.9, 2.8, 2.9, 3.1, 3.7, 3.3}. Find the 10^{th} and 90^{th} percentile?

Hands-On Problems (1)

- In an experiment that was repeated 6 times, the measured CPU time in milliseconds was found to be {2.8, 1.9, 3.2, 4.1, 2.7, 2.8}.
- For this set of measured data, compute the following:
  - sample mean,
  - sample median,
  - sample standard deviation, and
  - Semi-InterQuartile Range (SIQR).

Hands-On Problems (2)

- The performance improvements in the latest version of 7 layers of a new networking protocol was measured separately for each layer. What is the average improvement per layer?

  (**Hint:** the improvements in the 7 layers work in a multiplicative way)

Layer	1	2	3	4	5	6	7
Improvement (%)	18	13	11	8	10	28	5

Summary of Lecture #2

- 3 key issues related to performance measurements are presented
  - 5 types of workloads that have been commonly used to compare CS
    - Addition instruction, Instruction mixes, Kernels, Synthetic programs, Application benchmarks
  - HW/SW/FW performance monitoring techniques
  - Summarizing measured data
    - By a single number: mean, median, mode, geometric mean
    - Variability: range, variance, COV, percentiles, SIQR

Things To Do

- Read the software monitor example available in "Lecture Notes" → "Relevant Reading" of course website

https://xing560.sites.umassd.edu/

Next Topic

- Experimental Design and Analysis