

Lecture 1

System Evaluation and Benchmarking

Plan for today

- What and why
- Overview of available strategies

Why do we care?

- As systems researchers, we want to build “better” computer systems.
- Evaluation serves (at least) two purposes:
 - Tells us where to focus our efforts (Amdahl's Law)
 - Tells us how we compare to existing systems
- Performance is key in design, purchase, and use of computer systems
 - Other important features (security, reliability, usability, etc.) still require performance

Big Picture: What do we need?

- Metrics - a measurable quantity that is the basis for comparison
 - Choosing a good metric requires deciding what factors are most important
 - Latency and bandwidth are common in computer systems
 - Give me some other ones...
- A system to measure
 - Model
 - Simulation
 - “Live”
- A set of tests to perform on the target system
 - Benchmarks
- What else?

Choosing Metrics

- What performance metric should be used to compare the following?
 - Two disk drives
 - Two transaction processing systems
 - Two packet retransmission algorithms
 - Two clock scaling algorithms for reducing energy usage

2227, Spring 2006

5

Choosing a system to measure

- Models
 - + cheap, fast
 - highly simplified
- Simulation
 - + easy to vary parameters, test assumptions
 - cost/time depends on level of detail, less detailed simulations may leave out important factors
- Live System
 - + real results, can't overlook contribution of other components
 - measurement can alter results
 - Can be hard to interpret
 - Expensive

2227, Spring 2006

6

Choosing experiments

- The performance of a system depends on the following three factors:
 - Garbage collection technique used (concurrent, stop and copy, none)
 - Type of workload (office desktop computing, database server, scientific computing)
 - Type of CPU (Pentium 4, POWER 4)

How many experiments are needed? How do you quantify the performance impact of each factor?

2227, Spring 2006

7

And then there's analysis

- Why performance analysis is an "art" not a "science":
- Given the following measurements of throughput:

System:	Workload 1	Workload 2
A	20	10
B	10	20

- What is a fair comparison?

2227, Spring 2006

8

Some possibilities...

- Absolute:

System	Workload1	Workload2	Average
A	20	10	15
B	10	20	15

- Performance of A relative to B:

System	Workload1	Workload2	Average
A	2x	0.5x	1.25x
B	1x	1x	1x

- Performance of B relative to A:

System	Workload1	Workload2	Average
A	1x	1x	1x
B	0.5x	2x	1.25x

2227, Spring 2006

9

How do you learn performance analysis?

- Learn the mathematical basics
 - Statistics!
- Read papers about real performance studies critically
 - Throughout this course, next week especially
- Practice – do a project
 - Think carefully about evaluation from the start

2227, Spring 2006

10

Common Mistakes

- No goals
 - Best to work with a clear goal in mind – too much generality leads to unneeded complexity
- Biased goals
 - How can we show that our system is better than their system?
- Unsystematic approach
 - Arbitrary selection of parameters to vary; clever theories about causes unsupported by data

“No amount of experimentation can ever prove me right; a single experiment can prove me wrong.” -Einstein

2227, Spring 2006

11

More common mistakes

- Not understanding the problem
 - A problem well stated is half solved
 - Why do you think defining a thesis topic is so hard?
- Incorrect performance metrics
 - E.g. compare MIPS (Millions of instructions per second) for CISC CPU against MIPS for a RISC CPU
 - Strive to identify relevant metrics, rather than the ones that are easy to collect
- Unrepresentative workload
 - Huge impact on results!
 - Very tempting to construct a favorable workload

2227, Spring 2006

12

YACMS

- Wrong evaluation technique
 - Measurement/simulation/modelling
 - Choose what works best for the problem, not what you are most comfortable with
- Overlooking important parameters
 - Make a list of everything (system and workload) that could affect performance
- Ignoring significant factors
 - What's the difference?
- Inappropriate design
 - How much of the parameter space is covered by the experiments?
 - Want to maximize new info for a given # of experiments

2227, Spring 2006

13

Final YACMS

- Inappropriate level of detail
 - Tradeoff between exploring alternatives and depth
- No analysis
 - You'll see a lot of this if you read enough systems papers!
- Erroneous analysis
- No sensitivity analysis
- Improper treatment of outliers
 - When is it appropriate to throw them out?
- Assuming no changes over time
- Ignoring variability
- Too complex analysis
- Improper presentation of results
- Omitting assumptions and limitations

2227, Spring 2006

14

Resources

- Papers for next week's discussion
 - Imbench: an extensible micro-benchmark suite
 - Carl Staelin
 - Scale and Performance in a Distributed File System
 - Howard et al.
- "Textbook" reference
 - "The Art of Computer Systems Performance Analysis", Raj Jain, 1991, Wiley & Sons
- Fun papers
 - "Brittle metrics in operating systems research", Jeffrey Mogul, HotOS-VII
 - "Should computer scientists experiment more? (16 excuses to avoid experimentation)", Walter F. Tichy, IEEE Computer, vol. 31, no. 5, May 1998

2227, Spring 2006

15