CS147

CS 147: puter Systems Performance Analysis Course Introduction

CS 147: Computer Systems Performance Analysis Course Introduction

Overview

Class Introduction

Administrivia

Projects

Overview

Break

Issues in Performance Analysis

Introduction

What You Need to Know to Analyze Performance

In Summary



Basic Course Information

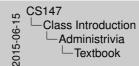
- ► Prerequisites: CS70, Math 35
 - ► Highly recommend CS 105, 140, etc.
- Web page: www.cs.hmc.edu/geoff/cs147
- ► AIM: ProfKuenning
- Office hours on Web page
 - ▶ Will be changed in first few weeks of term



Prerequisites: CS70, Math 35
Highly recommend CS 105, 140, etc.
Web page: www.cs.hmc.edu/geoffics147
AMN-Profiscenting
Office Nucsin on Web case

asic Course Informatio

- ► The Art of Computer Systems Performance Analysis, by Raj Jain
 - Readings assigned weekly
 - ► First assignment: Chapters 1-2
- You are expected to find and read materials required to perform projects
- Recommended: The Visual Display of Quantative Information, by Edward Tufte

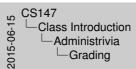


ok

The Art of Computer Systems Performance Analysis, by Rej Jain
Readings assigned weekly
First assignment: Chapters 1-2

Frost assignment: Crapers 1-2
 You use exposited to find and read materials required to parform projects
 Recommended: The Visual Display of Quantative Information, by Edward Tuffs

- ► Project 50%
- ► Midterm 20%
- ► Homework 25%
- ► Evaluation of other projects 5%
- ► Grading criteria may change!



Project - 50%

Michaem - 20%

Homework - 25%

Evaluation of other projects - 5%

Grading oriteria may change!

Project Information

- Design and perform evaluation of a real software system
- Present plans in class
- Present results in class
- ► Final written report
- Evaluate others' projects



- Operating systems or their components (file systems, I/O subsystems, process handling, etc.)
- Compilers
- Databases
- ► Real-time applications
- Large application packages (e.g., email, web tools)
- Networks/networking systems
- The Web

CS147
Class Introduction
Projects
Suitable Subjects for Projects

se Subjects for Projects

 Operating systems or their components (file systems, I/O subsystems, process handling, etc.)

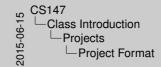
Compilers
Databases

Real-time applications
 Large application packages (e.g., email, web tools)

Networks/networking systems
 The Web

Project Format

- Group projects
 - ▶ Prefer 2-3 people per group
 - Groups chosen by you
- Project topic chosen by the group
- ▶ All group members must participate in all group activities





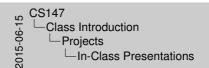
Written Materials for Project

- ▶ Project proposal (1-2 pages), due February 20
- ► Project design (3-6 pages), due Mar 9
- ► Final report (10+ pages), due May 4
- All due dates are 5 PM
 - ► E-mail, give to me during class, or slide under my door



In-Class Presentations

- Detailed presentation of project designs (March 5) 15 minutes per group
- Presentation of results (April 23 & 25) 30-40 minutes per group
- All group members expected to present



ss Presentations

 Detailed presentation of project designs (March 5) - 1 minutes per group

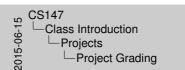
group

Project Grading

Several criteria:

- ▶ Proper design of the experiment
- ► Care and thoroughness of its execution
- Completeness of analysis
- Quality of data presentation
- Insight gained from experiment

Insight is the most important



ect Grading

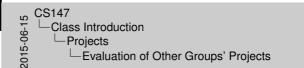
Several criteria:

Proper design of the experiment

Care and thoroughness of its execution
 Completeness of analysis

 Insight gained from experiment Insight is the most important

- Submitted by each student individually
- ▶ 1-page critique of each group's proposed experiment
 - Due Mar 9, 5 PM
- ▶ 1-page critique of each group's results
 - ▶ Due May 4, 5 PM
- Graded on basis of insight into strengths and flaws of each project

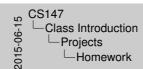




- Submitted by each student individually
 1-page critique of each group's proposed experiment
- Due Mar 9, 5 PM
- 1-page critique of each group's results
 Due May 4, 5 PM
 Graded on basis of insight into strengths and flaws of each

Homework

- ► Approximately 5 homework sets worth 5% each
- Assigned Wednesday each of 3rd-7th weeks
- ► Due Wednesday of the following week

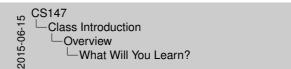


Homework

Approximately 5 homework sets worth 5% each
Assigned Wildersday and 69 57 th weeks
Dis Wildersday dark bill 500 sing weeks

What Will You Learn?

- You'll learn:
 - ▶ How to design & perform software experiments
 - ► How to analyze & present data
 - ► How to critique others' experiments and data
- ► But not:
 - Basic systems software principles
 - Systems software modeling
 - Queueing theory
 - Simulation techniques





- ► Introduction (1½ classes)
- ► Review of probability & statistics (3½ classes)
- Analysis of sample system (1 class)
- Presentation of project designs (1 class)

CS147
Class Introduction
Cocycle
Class Outline (1)

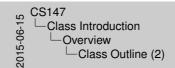
Introduction (1½ classes)
Review of probability & statistics (3½ classes)
Analysis of sample system (1 class)
Presentation of project designs (1 class)

Presentation of project designs (1 class)

Technology

**Techno

- Graphical presentation (2 classes)
- System measurement techniques and tools (3 classes)
- Experimental design and analysis (4 classes)
- Design and analysis of sample experiments (1–2 classes)
- Presentation of project results (1–2 classes)



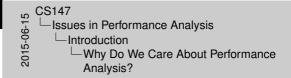
Presentation of project results (1-2 classes)

CS147 2015-06-15 -Class Introduction └─Break

Let's take a break

Let's take a break

- ▶ Performance almost always key issue in software
 - Especially in system software
- Everyone wants best possible performance
- Cost of achieving performance also key
- Reporting performance necessary in many publication venues
 - Both academic and industry



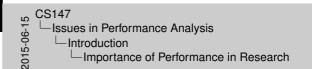
Thy Do We Care About Performance Analysis?

Performance almost always key issue in software
 Especially in system software

Everyone wants best possible performance
 Cost of achieving performance also key

Reporting performance necessary in many publication venues

- ▶ Performance is key in almost all CS research
- A solution that doesn't perform well isn't a solution at all
- Successful research must prove performance characteristics to a skeptical community





- Generally regarded as poor
- Many systems have little performance data presented
- Many systems are measured by improper criteria
- Many experiments are poorly designed
- Many results are badly or incorrectly presented
- Replication not generally respected

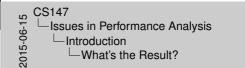


ate of Performance Evaluation in the Field

- Generally regarded as po
- Many systems have little perform
- Many systems are measured by improper criteria
 Many experiments are poorly designed
- Many experiments are poorly designed
 Many results are badly or incorrectly presented
 Replication not generally respected

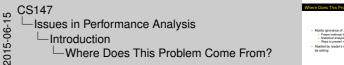
What's the Result?

- ► You can't always trust what you read in a research paper
- Authors may have accidentally or intentionally misled you
 - Overstating performance
 - Hiding problems
 - Not answering the important questions



You can't always trust what you read in a research paper
 Authors may have accidentally or intentionally misled you
 Overstaing performance.
 Hiding problems

- Mostly ignorance of:
 - Proper methods for measuring performance
 - Statistical analysis
 - Ways to present results
- Abetted by reader's ignorance of what questions they should be asking



- ▶ People are taking performance measurement more seriously
- Quality of published experiments is increasing
- Yours had better be of high quality, too
 - Publishing is tough
 - Business competition is tough
 - ► So be at the top of the heap of papers



People are taking performance measurement more seriously

Quality of published experiments is increasing
 Yours had better be of high quality, too

Things Are Improvin

urs had better be of high quality, too

Publishing is tough

Business competition is tough

So be at the too of the heap of papers

- How to select appropriate experiment characteristics
- Proper performance measurement techniques
- Proper statistical techniques
- Proper data presentation techniques

2015-06-1

CS147

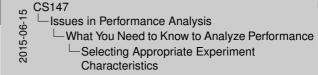
-Issues in Performance Analysis

-What You Need to Know to Analyze Performance

What Do You Need To Know to Analyze Performance?

Selecting Appropriate Experiment Characteristics

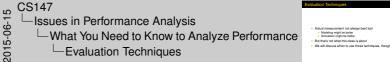
- Evaluation techniques
- Performance metrics
- Workloads



Performance metrics

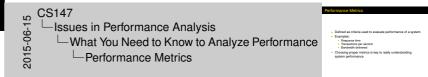
Evaluation Techniques

- Actual measurement not always best tool
 - Modeling might be better
 - Simulation might be better
- But that's not what this class is about
- We will discuss when to use those techniques, though

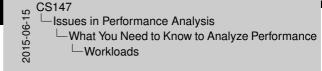


Performance Metrics

- Defined as criteria used to evaluate performance of a system
- Examples:
 - Response time
 - Transactions per second
 - Bandwidth delivered
- Choosing proper metrics is key to really understanding system performance



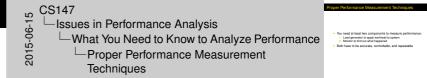
- Defined as requests users make of a system
- ► If you don't evaluate with proper workload, you aren't measuring what real users will experience
- ► Typical workloads:
 - Types of queries
 - Jobs submitted to an OS
 - Messages sent through a protocol



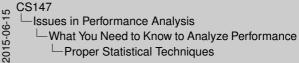
rkloads

- Defined as requests users make of a system
 If you don't evaluate with proper workload, you aren't measuring what real users will experience
- Typical workloads:
 Types of queries
 Jobs submitted to an OS
 Messages sent through a protocol

- ▶ You need at least two components to measure performance:
 - 1. Load generator to apply workload to system
 - 2. Monitor to find out what happened
- ▶ Both have to be accurate, controllable, and repeatable



- Most computer performance measurements not purely deterministic
- Most performance evaluations weigh effects of different alternatives
- How to separate meaningless variations from vital data in measurements?
- Requires proper statistical techniques
- We'll spend a fair amount of time on this issue



oper Statistical Techniques

 Most computer performance measurements not deterministic
 Most conformance analysticas weight offcom of

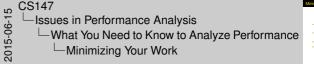
alternatives

How to separate meaningless variations from vital

Requires proper statistical techniques
 We'll energy a fair amount of time on this issue

30/35

- Unless you design carefully, you'll measure a lot more than you need to
- A careful design can save you from doing lots of measurements
- Should identify critical factors
- And determine smallest number of experiments that gives "sufficiently" accurate answer



nizing Your Work

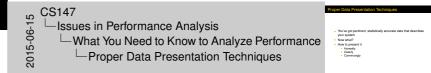
Unless you design carefully, you'll measure a lot more the you need to
 A careful design can save you from doing lots of

measurements

Should identify critical factors

And determine smallest number of experiments that gives

- You've got pertinent, statistically accurate data that describes your system
- Now what?
- ► How to present it
 - Honestly
 - Clearly
 - Convincingly



Why Is Performance Analysis Difficult?

- ▶ It's an art it's not mechanical
- Can't just apply a handful of principles and expect good results
- Must understand your system
- Must properly select measurement techniques and tools
- Must be careful and honest

CS147

In Summary

Why Is Performance Analysis Difficult?

Summary

Example

- Suppose you've built OS for a special-purpose Internet browsing box
- ► How well does it perform?
- Indeed, how do you even begin to answer that question?



In Summary

Starting on an Answer

- ▶ What's the OS supposed to do?
- ▶ What demands will be put on it?
- What hardware will it work with, and what are that hardware's characteristics?
- What performance metrics are most important?
 - Response time?
 - Delivered bandwidth?
 - Something else?

