

# CS 6410: Advanced Systems

Fall 2013

Instructor: Hakim Weatherspoon

TA: Erluo Li and Qin Jia

# Who am I?

- Prof. Hakim Weatherspoon
  - (Hakim means Doctor, wise, or prof. in Arabic)
  - Background in Education
    - Undergraduate University of Washington
      - Played Varsity Football
        - » Some teammates collectively make \$100's of millions
        - » I teach!!!
    - Graduate University of California, Berkeley
      - Some class mates collectively make \$100's of millions
      - I teach!!!
  - Background in Operating Systems
    - Peer-to-Peer Storage
      - Antiquity project - Secure wide-area distributed system
      - OceanStore project – Store your data for 1000 years
    - Network overlays
      - Bamboo and Tapestry – Find your data around globe
    - Tiny OS
      - Early adopter in 1999, but ultimately chose P2P direction

# Goals for Today

- Be brief!
- Why take this course?
- How does this class operate?
- Class details

# Why take this course

- Learn about systems abstractions, principles, and artifacts that have had lasting value,
- Understand attributes of systems research that is likely to have impact,
- Become comfortable navigating the literature in this field,
- Gain experience in thinking critically and analytically about systems research, and
- Acquire the background needed to work on research problems currently under study at Cornell and elsewhere.

# Coverage

- The course is about the cutting edge in computer systems – the topics that people at conferences like ACM Symposium on Operating Systems Principles (SOSP) and the Usenix Conference on Operating Systems Design and Implementation (OSDI) love
- We look at a mix of topics:
  - Classic insights and classic systems that taught us a great deal or that distilled key findings into useable platform technologies
  - Fundamental (applied theory) side of these questions
  - New topics that have people excited right now

# Systems: Three “arcs” over 40

**Risk: Cool theory but impractical result that can't be deployed . even the model is unrealistic!**

**Advantage: At massive scale your intuition breaks down. Just doing it is a major**

**Risk: Totally unprincipled spaghetti**

- In the e



- Today, these are separated
- Some people get emotional over which is best!

e and more

# Who is the course “for”?

- Most of our CS6410 students are either
  - PhD students (but many are from non-CS fields, such as ECE, CAM, IS, etc)
  - Undergraduates seriously considering a PhD
- A small subset are MEng students
  - Some MEng students are ok pretending to be PhD students and have the needed talent and background
  - MEng students not fitting this profile won't get permission to take the course
  - CS5410 was created precisely to cover this kind of material but with more of an MEng focus and style

# Why take this course

- Satisfy systems breadth requirement

		Research Styles		
		Theoretical	Systems	Applied
Areas	Algorithms/Theory	68xx		
	AI	676x		67xx except 676x
	Systems		632x, 64xx	
	PL	6110		
	Sci. Comp. and Apps			62xx, 65xx, 66xx



# How class operates and class detail

# How this class operates

- Instructor: Hakim Weatherspoon
  - [hweather@cs.cornell.edu](mailto:hweather@cs.cornell.edu)
  - Office Location: 4105C Upson
- TA: Erluo Li and Qin Jia
  - [el378@cs.cornell.edu](mailto:el378@cs.cornell.edu) and [qinjia@cs.cornell.edu](mailto:qinjia@cs.cornell.edu)
- Lectures:
  - CS 6410: Tu, Th: 10:10 – 11:25 PM, 407 Phillips Hall

# Course Help

- Course staff, office hours, etc:
  - <http://www.cs.cornell.edu/courses/cs6410/2013fa>
- Research projects
  - <http://fireless.cs.cornell.edu>

# CS 6410: Overview

- Prerequisite:
  - Mastery of CS 4410 material
    - Fundamentals of OS design
    - How parts of the OS are structured
    - What algorithms are commonly used
    - What are the mechanisms and policies used
- Class Structure
  - Papers Readings (whole semester)
  - Paper Presentations (whole semester)
  - Labs (first 1/8)
  - Research Project (second 7/8)

# CS 6410: Topics

- Operating Systems
  - Concurrency, file systems, VM, I/O, etc.
- Distribution/Networking
  - RPC, clusters, pub/sub, mobility, etc.
- Fault Tolerance
  - Replication, consensus, transactions, etc.

# CS 6410: Paper Readings

- Required reading is always *two* papers
  - Different approach, competition, criticism,...
  - Papers pulled from, best journals and conferences
    - TOCS, SOSP, OSDI, ...
  - 27 lectures, 54 (required) papers!
- Read papers before each class and bring notes
  - takes ~3 to 4 hrs per paper, write notes and questions
- Write a review and turn in *at least one hour* before beginning of class
  - Turn on online via Course Management System (CMS)
  - ***No late reviews will be accepted***

# CS 6410: Writing Reviews

- Each student is *required* to prepare notes on each paper before class and to bring them to class for use in discussion.
- Your notes should list assumptions, innovative contributions and criticisms. Every paper in the reading list has at least one major weakness.
- Turn paper reviews in online before class via CMS
  - Be succinct—One paragraph per paper
    - Short summary of paper (two or three sentences)
    - Two to three strengths/contributions
    - and at least one weaknesses
  - One paragraph to compare/contrast papers
  - In all, turn in two to three paragraphs

# CS 6410: Paper Presentations

- Each person will present a paper one or two times, depending on class size
  - Read and understand both required and suggested papers
- Two and a half weeks ahead of time
  - Meet with professor to agree on ideas to focus on
- One and a half weeks ahead of time
  - Have presentation prepared and show slides or “chalk talk” to professor
- One week ahead of time
  - Final review / do a number of dry-runs



# CS 6410: Class Format

- 45 minutes presentation,  
30 minutes discussion/brainstorming.
  - In that order, or mixed.
- All students are required to participate!
- Counts in final grading.

# CS 6410: Labs

- Labs (first 1/8 of semester)
  - 2 labs
  - Using Amazons EC2/S3 infrastructure
  - Building a proxy using events (instead of threads)

# CS 6410: Research Project

- One major project per person
  - Or two persons for a very major project
- Initial proposal of project topic – due mid-September
- Survey of area (related works)—due begin of October
  
- Midterm draft paper – due begin of November
- Peer reviews—due a week later
  
- Final demo/presentation—due begin of December
- Final project report – due a week later

# CS 6410: Project Suggestions

- <http://fireless.cs.cornell.edu/projects>
- Networks
  - Software Routers and Packet Processors
  - Netslice, FwP, Fmeter
- Data Center Networking and Network Measurements
  - Software Defined Network Adapter (SoNIC)
  - Cornell NLR Rings Testbed
- Cloud Storage
  - User controlled computation: xCloud-- <http://xcloud.cs.cornell.edu>
  - User controlled storage: Redundant Array of Cloud Storage (RACS)
- File Systems
  - Local and wide-area file systems enhancements
    - Reliability, consistency, performance

# CS 6410: Project Suggestions

- Global-scale datacenters
  - Utilization, Low-energy file systems, Virtual machines, etc
  - High bandwidth-delay product networks enhancements
    - Cluster of servers, Netslice, RouteBricks, FWP, Maelstrom, etc
  - Exploit parallelism in multicore processors
    - Thread vs events, operating system, network process architectures
- P2P
  - Cloud storage @ home, etc
- I have more ideas, but you can also talk to other faculty for more ideas:

Professors Birman, Sizer, Schneider, Van Renesse, Gehrke, Myers, or Foster

# CS 6410: Project Infrastructure

- Fractus: our very own (mini) cloud
- Amazon's Cloud Infrastructure EC2/S3
- Emulab
- PlanetLab
- Cornell's Center for Advanced Computing (CAC)
- ...

# Important Project Deadlines

~9/19	Submit your topic of interest proposal
~10/3	Submit 2-3 pages survey on topic
~10/4	Discuss project topic with me
~11/5	Midterm draft paper of project
~12/5	Final demo/presentation of project
~12/12	Final paper on project

# CS 6410: Grading

- Class Participation ~ 40%
  - lead presentation, reading papers, write reviews, participation in class discussion
- Project ~ 50%
  - Proposal, survey, draft, peer review, final demo/paper
- (maybe) Labs ~ 5%
- Subjective ~ 5%
- This is a rough guide



# Academic Integrity

- Submitted work should be your own
- Acceptable collaboration:
  - Clarify problem, C syntax doubts, debugging strategy
  - You may use any idea from any other person or group in the class or out, provided you **clearly state what you have borrowed and from whom.**
  - If you do not provide a citation (i.e. you turn other people's work in as your own) that is *cheating*.
- Dishonesty has no place in any community
  - May NOT be in possession of someone else's homework/project
  - May NOT copy code from another group
  - May NOT copy, collaborate or share homework/assignments
  - University Academic Integrity rules are the general guidelines
- Penalty can be as severe as an 'F' in CS 6410

# Stress, Health and Wellness

- Need to pace yourself to manage stress
  - Need regular sleep, eating, and exercising
- Do ***not*** come to class sick (with the flu)!
  - Email me ***ahead*** of time that you are not feeling well
  - People not usually sick more than once in a semester

# Before Next time

- Sign up twice to present (first *and* second half)
- Read *two* papers below and write review
  - *End-to-end arguments in system design*, J.H. Saltzer, D.P. Reed, D.D. Clark. *ACM Transactions on Computer Systems* Volume 2, Issue 4 (November 1984), pages 277--288.  
<http://portal.acm.org/citation.cfm?id=357402>
  - *Hints for computer system design*, B. Lampson. *Proceedings of the Ninth ACM Symposium on Operating Systems Principles* (Bretton Woods, New Hampshire, United States) 1983, pages 33--48.  
<http://portal.acm.org/citation.cfm?id=806614>
- Lab 0
  - Using Amazon's EC2/S3 infrastructure
- Check website for updated schedule