CS 6410: ADVANCED SYSTEMS

PROF. HAKIM WEATHERSPOON

Fall 2017

A PhD-oriented course about research in systems
About me (Hakim)
Goals for Today

- What is CS6410 “about”?  
  - What will be covered, and what background is assumed?  
  - Why take this course?  
  - How does this class operate?  
  - Class details

- Non-goal: We won’t have a real lecture today  
  - This is because our lectures are always tied to readings
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
Course Overview

- First and foremost: Attend every class, participate
  - You’ll need to do a quite a bit of reading.
  - You’ll write a short (1 paragraph) response each time
    - Either response to a posted question
    - Or, summary of the papers
  - Whoever presents the paper that day grades these (\( -\), \( \sqrt{\ }, \sqrt{\sqrt{\ }}\))
  - You can skip up to 5 of them, whenever you like. Hand in “I’m skipping this one” and the grader will record that. But not more than 5.

- You’ll have two “miniprojects” during first six weeks
  - Cloud-based miniproject: start your own cloud
  - Build a block chain!: Initially single threaded, then multi-threaded and/or event based

- Then will do a semester-long independent project
  - Project can be done in pairs, or
  - Project can be part of a larger research project with an advisor

- Students need to present a paper. Required
Time Consideration

- You can definitely take one other class too
- But, should not take more than two courses

- Not so much that it is “hard” (by and large, systems isn’t about hard ideas so much as challenging engineering), but it definitely takes time
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,
- Learn to present papers in a classroom setting,
- 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.
- Advance your research agenda: Find a research advisor and project
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)
  - Two year MS students who might switch into PhD
  - Undergraduates seriously considering a PhD (need professor’s permission)

- Fall 2016: Too big to allow MEng students.
  - MEng program offers lots of other options;
  - CS6410 has a unique role for the core CS PhD group
CS6410 versus just-read-papers

- A paper might just brag about how great it is, how well it scales, etc

- Reality is often complex and reflects complex tensions and decisions that force compromises

- In CS6410 our goal is to be honest about systems: see what the authors had to say, but think outside of the box they were in when they wrote the papers
Details

- **Instructor:** Hakim Weatherspoon
  - hweather@cs.cornell.edu
  - Office Location: 427 Gates

- **TA:** Kevin Sekniqi

- **Lectures:**
  - CS 6410: Tu, Th: 10:10 – 11:25 PM, 114 Gates / Bloomberg 061
Course Help

- Course staff, office hours, announcements, etc:
  - [http://www.cs.cornell.edu/courses/cs6410/2017fa](http://www.cs.cornell.edu/courses/cs6410/2017fa)

- Please look at the course syllabus: the list of papers is central to the whole concept of this class

- Research project ideas are also listed there
Prerequisite:

- Mastery of CS3410, CS 4410 material
  - Fundamentals of computer architecture and OS design
  - How parts of the OS are structured
  - What algorithms are commonly used
  - What are the mechanisms and policies used

- Some insights into storage systems, database systems "helpful"

- Some exposure to networks, web, basic security ideas like public keys
CS 6410: Topics:

- Operating Systems
  - Core concepts, multicore, virtualization, uses of VMs, other kinds of “containment”, fighting worms/viruses.

- Cloud-scale stuff
  - Storage systems for big data, Internet trends, OpenFlow

- Foundational theory
  - Models of distributed computing, state machine replication and atomicity, Byzantine Agreement.
  - Impact of social networks, P2P models, Self-Stabilization

- A few lectures will focus on new trends: RDMA, BitCoin (a distributed protocol!), etc
CS 6410: Readings

- Required reading for each lecture: 1 or 2 papers
  - Reflecting contrasting approaches, competition, criticism,…
  - Papers pulled from, best journals and conferences
    - TOCS, SOSP, OSDI, …
  - 26 lectures, 26 to 54 (required) papers + “recommended” papers!

- Read papers before each class and bring notes
  - Takes ~2 to 3 hrs per paper, write notes and questions

- Write a review/question response and turn in at least one hour before class
  - Turn on online via Course Management System (CMS)
  - No late reviews will be accepted, but you can skip 4 of them
  - Graded by the person doing that lecture on a simple √-, √, √+ basis plus written comments.
Mini-Projects

- New, early part of semester

- Two of them
  - Hands on experience with multicore parallelism in C or C++
  - Hands on experience with cloud computing on EC2
CS 6410: Two small projects

- Goal: Get the rust off your systems skills!

- Mini-project one: start your own cloud

- Mini-project two: Build a multi-threaded, multicore version of a block chain. Make it really, really fast!
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.
  - Don’t channel the authors: your job is to see the bigger questions!

- Turn paper reviews or response question in online before class via CMS
  - Be succinct—One paragraph per paper
    - Respond to question, or
    - Short summary of paper (two or three sentences)
      - Two to three strengths/contributions
      - and at least one weaknesses
  - One paragraph to compare/contrast papers
Ideally, each person will present a paper, depending on the stable class size

- Read and understand both required and suggested papers
- Learning to present a paper is a big part of the job!
- The presenting person also grades the essays for that topic

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

- 35-45 minutes presentation,
- 30 minutes discussion/brainstorming.
  - In that order, or mixed.
- All students are required to participate!
- Counts in final grading.
One research project per person
  - Can work individually or in pairs
  - Further, often can turn research agenda in separate research area into a systems 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–last day of class in Dec/Nov
Final project report – due a week later
CS 6410: Project Suggestions

- Supercloud/X-containers related projects
- New cloud-scale computing services, perhaps focused on applications such as the smart power grid, smart self-driving cars, internet of things, smart homes
- Disaggregated datacenter related
- Operating system features to better leverage RDMA
- Software defined network infrastructure on the systems or network side (as distinct from Nate’s focus on the PL side)
- Study the security and distributed systems properties of BitCoin
- New systems concepts aimed at better supporting “self aware” applications in cloud computing settings (or even in other settings)
- Building better memory-mapped file systems: current model has become outmoded and awkward
- Tools for improving development of super fast multicore applications like the one in mini-project one.
- … and you can invent more of your own!
## Important Project Deadlines

<table>
<thead>
<tr>
<th>Date</th>
<th>Task Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>9/7</td>
<td>Submit your topic of interest proposal</td>
</tr>
<tr>
<td>9/21</td>
<td>Submit 2-3 pages survey on topic</td>
</tr>
<tr>
<td>(Oct)</td>
<td>Discuss project topic with Zhiming/Hakim</td>
</tr>
<tr>
<td>11/2</td>
<td>Midterm draft paper of project</td>
</tr>
<tr>
<td>11/30</td>
<td>Final demo/presentation of project</td>
</tr>
<tr>
<td>12/7</td>
<td>Final paper on project</td>
</tr>
</tbody>
</table>
CS 6410: Grading

- **Class Participation ~ 40%**
  - lead presentation, reading papers, write reviews, participation in class discussion

- **Projects ~ 50%**
  - Probably 10% will be the two mini-projects, 40% the big term one
  - Proposal, survey, draft, peer review, final demo/paper

- **Subjective ~ 10%**

- **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

- Don’t miss class... but....

- 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

- Rank-order 2 papers to present (first and second half)
- Read first papers below and write review

- Miniproject0
  - Using Amazon’s EC2/S3 infrastructure
- Check website for updated schedule