CSci599: Modern Distributed Systems
Tuesday/Thursday 5-6:50 in Room TBD
Wyatt Lloyd

Last update: 2015-5-27

Course Description:

Distributed systems explore how to use multiple computers to build bigger and better systems than are possible on a single computer. Single computers can fail, are limited in their processing capacity, and can only be in one physical place. This course will cover how distributed systems overcome all of these limitations. It will then show how distributed systems enable large web services like Facebook, Google, and Amazon.

This is a graduate course in distributed systems. We will cover the foundations of distributed systems and their modern, practical applications. Topics will include logical time, consistency, transactions, fault tolerance, quorums, replicated state machines, atomic commit, Paxos, peerto-peer systems, and the Google and Facebook stacks.

Each class will include a subset of the following: a lecture that covers the fundamentals of an area, presentation of supplementary papers by class members, a discussion of an assigned paper on the topic, and a discussion of research project related to the topics and paper discussed. In addition, students will put their knowledge into practice by building a large distributed system over the course of the semester.

The class has four major components: paper readings, paper presentations, programming assignments, and exams.

Paper Readings:

Students will be expected to read and master one paper per class. Summaries of some of the papers will be assigned, class participation grades will be based heavily on in-class discussion of papers, and paper techniques and insights will be included on exams.

Paper Presentations:

There will be many supplementary papers for each class. Over the course of the semester, students will chose at least one paper to read, summarize, and present to the class. The paper summary will be made available to other students in the class.

The presentation of the paper will be 8 minutes short. The presentations will cover the paper at a high level for the rest of the class. Final versions of the presentations are due 2 days before their assigned class.

Programming Assignments:

Students will develop a cumulative project over the course of the semester. The project is spread over 4 assignments, each of which must be coded individually. Students are allowed to discuss the assignments with each other, but they are not allowed to look at others' solutions from this or previous years. Writing or looking at pseudo-code together is also not allowed. Viewing general information on the web, i.e., StackOverflow, is permitted.

Assignments will be coded in Go, synced regularly to a git repository, and submitted via that git repository. Students who take the class are assumed to be proficient in C++ and familiar with git. Assignments will be graded mainly on correctness with some points allocated to programming style and performance.

Exams:

There will be two in-class exams. Each exam will cover roughly half the material, will be closed book, and will test topics covered in lectures and required paper readings.

Grading:

Paper Summaries: 10%
Paper Presentation: 10%
Participation: 5%
Exams: 35%
Programming Assignments: 40%

Office Hours:

Held weekly on TBD from TBD in SAL 210.

Assignments:

There are 4 programming assignments. Each assignment builds upon the previous assignment and all must be completed. Assignments are due at 11:59pm on the specified day.

Assignment 1: Local MapReduce + Go Intro	due Sep 4
Assignment 2: Primary/Backup Key/Value Service	due Sep 18/25
Assignment 3: Paxos-based Key/Value Service	due Oct 23/30
Assignment 4: Sharded Key/Value Service	due Nov 13/20

In the first assignment you will be introduced to Go and write a MapReduce implementation that runs on your local file system. In the second assignment you will create a key/value service that uses primary/backup replication for fault tolerance to ensure correctness even if your server is killed at arbitrary times. In the third assignment you will make your key/value service fault tolerant using Paxos, which overcomes many limitations of the primary/backup approach. In the final assignment you will shard the key/value service so it can scale out as you as more (logical) machines to the system.

Class Structure and Schedule:

Intro
Aug 25 – Intro to Distributed Systems
Aug 27 – Intro to Go, Assignment Overview

Sep 1 – Background: MapReduce & Reading Papers in Depth
Sep 3 – Background: Remote Procedure Calls & Numbers Everyone Should Know
(Sep 4 – Assignment 1 due)
Sep 8 – Background: Logical Time & Decomposing Safety and Liveness

Fault Tolerance

Sep 10 – Fault Models & Replicated State Machines
Sep 15 – Primary Backup Replication
Sep 17 – FLP (Impossibility of Consensus)
(Sep 18 – Assignment 2a due)
Sep 22 – Atomic Commit
Sep 24 – Paxos
(Sep 25 – Assignment 2b due)
Sep 29 – Paxos Optimized
Oct 1 – Byzantine Fault Tolerance

Oct 6 – SOSP – Guest Lecture TBD
Oct 8 – Exam 1

Scalability, Consistency, and Transactions

Oct 13 – Background: The Datacenter as a Computer
Oct 15 – Distributed Hash Tables / Consistent Hashing
Oct 20 – Strong Consistency (Linearizability / Sequential Consistency)
Oct 22 – Weaker Consistency (Eventual) and CAP
(Oct 23 – Assignment 3a due)
Oct 27 – Weaker Consistency (Causal)
Oct 29 – Background: Transactions
(Oct 30 – Assignment 3b due)
Nov 3 – Distributed Transactions
Nov 5 – Distributed Transactions Optimized

Modern Marvels

Nov 10 – Untrusted Servers
Nov 12 – Bitcoin
(Nov 13 – Assignment 4a due)
Nov 17 – Google Stack Day
Nov 19 – Facebook Stack Day
(Nov 20 – Assignment 4b due)
Nov 24 – Pushing Systems to their limits
Nov 26 – THANKSGIVING
Dec 1 – Exciting / Relevant Paper from SOSP 2015

Dec 3 – Exam 2

Academic Responsibilities:

Grading

Final letter grades will be determined using a "modified" curve. I will assign grades of C and below to individuals who do not perform satisfactorily in the above areas (i.e., you should not assume that because this is a graduate class you will get a B- or even C if you perform unsatisfactorily.).

We will not assign incompletes unless it is for a documented medical reason (in accordance with USC policy).

Diversity Statement

The diversity of the participants in this course is a valuable source of ideas, problem solving strategies, and engineering creativity. I encourage and support the efforts of all of our students to contribute freely and enthusiastically. We are members of an academic community where it is our shared responsibility to cultivate a climate where all students and individuals are valued and where both they and their ideas are treated with respect, regardless of their differences, visible or invisible.

Statement on Academic Conduct and Support Systems

Academic Conduct

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Section 11, *Behavior Violating University Standards* https://scampus.usc.edu/1100-behavior-violating-university-standards-and-appropriate-sanctions. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct.

Discrimination, sexual assault, and harassment are not tolerated by the university. You are encouraged to report any incidents to the *Office of Equity and Diversity* http://equity.usc.edu or to the *Department of Public Safety* http://capsnet.usc.edu/department/department-public-safety/online-forms/contact-us. This is important for the safety of the whole USC community. Another member of the university community – such as a friend, classmate, advisor, or faculty member – can help initiate the report, or can initiate the report on behalf of another person. *The Center for Women and Men* http://www.usc.edu/student-affairs/cwm/ provides 24/7 confidential support, and the sexual assault resource center webpage http://sarc.usc.edu describes reporting options and other resources.

Support Systems

A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the *American Language Institute* http://dornsife.usc.edu/ali, which sponsors courses and workshops specifically for international graduate students. *The Office of Disability Services and Programs* http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency makes travel to campus infeasible, *USC Emergency Information* *http://emergency.usc.edu* will provide safety and other updates, including ways in which instruction will be continued by means of blackboard, teleconferencing, and other technology.