Designing Modern Web-Scale Applications

Ashvin Goel

Electrical and Computer Engineering
University of Toronto

ECE1724
Topics

• Overview of the course
• Class format
• Introduction to the course
My Research Background

• Systems software
  • Operating systems
  • Storage systems
  • Dependable systems
  • Distributed systems

• Recent focus
  • Distributed storage systems
  • Big data analytics
  • Course reflects this focus
What are Web-Scale Apps?

- Applications that are hosted in massive-scale computing infrastructures such as data centers
- Used by millions of geographically distributed users
  - Via web browsers, mobile clients, etc.
- Produce, store, consume massive amounts of data
  - Scale is hard to comprehend
Focus of Course

• Web-scale applications are large scale systems
  • They require massive infrastructure for storing their data and for their computation needs

• Course focuses on
  • Infrastructure needed for web-scale applications
  • Big data computation models and analytics

• Core concerns
  • Efficiency, scalability, availability, reliability, consistency, programmability, flexibility
Key Issues

• How to store data at scale
• How to serve data with low latency
• How to index and analyze data at scale
  • Unstructured and structured data
  • Streaming data
  • Graph data
  • Model training data
Course Goals

• Understand challenges in designing systems and infrastructure for web-scale applications
• Understand the design of data storage systems
• Understand the design of data analytics applications
• (Optional) Gain experience with system development with a large software project
Relation to Other Courses

• ECE1779: Intro to cloud computing teaches you to be a cloud application developer
  • Use Microsoft Azure, Google App Engine, Amazon AWS Lambda, etc.
  • Lots of jobs available

• ECE1724: This course teaches you to be the cloud provider’s application developer
  • Understand the design of the provider’s infrastructure
  • Use it to design big data applications
  • In-demand jobs
Industrial Relevance

• Many papers in the reading list are from industry
  • GFS, MapReduce, Bigtable, Spanner, Twine, Millwheel, Pregel, TensorFlow (Google)
  • Dynamo (Amazon)
  • Spark, Spark Streaming (Databricks)
  • Zookeeper (Yahoo)

• Similarly, for optional reading list
  • Chubby, Omega, Borg (Google)
  • Azure, Apollo, Quincy (Microsoft)
  • Storm (Twitter)
  • Akkio, Flighttracker, SVE (Facebook)
Course Prerequisites

• Distributed systems
• Operating systems
• Preferably taken courses in database systems, networking
• For course project
  • Developed large software project
  • Languages like Java, C, C++
Main Topics

- Consensus and coordination
- Distributed data stores
- Data parallel frameworks
- Scheduling and resource management
- Stream processing
- Graph processing
- Machine learning systems
Class Format
Overview

• Course web site
• Course readings
• Course project and presentation
• Quizzes
• No assignments
• No final exam
• Grading policy
Course Website

- Course website available from my home page
  - [http://www.eecg.toronto.edu/~ashvin](http://www.eecg.toronto.edu/~ashvin)
  - This website contains reading material, project ideas, etc.

- Quercus
  - Course announcements on Quercus Announcements
  - Please ask general questions on Quercus Discussions
  - Quizzes

- Email
  - If you want to contact me directly
Course Readings

• Advanced, discussion-oriented seminar course
  • Course discusses seminal and recent research papers
• Background in distributed systems, databases, OS, networking
• Typically discuss 2 papers per week
  • Unless marked optional, all papers are required reading
• Will take about 4-6 hours per week
• Allows discussion in class
  • It will show if you don't do the reading ...
• Significant component of grade is based on class participation
Course Project and Presentation

• Choose a project based on topics covered
• Sample topics will be posted on website

• Options
  • Implement and evaluate a system
  • Evaluate existing system
  • Write a research paper

• Write up your work
  • 8-10 pages

• Present your work
Quizzes

• Three quizzes
• Roughly every 4 weeks
• Quiz will cover topics discussed in previous 4 weeks
• Questions will vary, some straightforward, some open ended, **will assess conceptual and critical understanding of course content**
• Exact format and dates will be announced later
  • Likely a take home (open notes) Quercus quiz
Grading Policy

• Two options
  • Course Project
  • All Quizzes
Grading Policy: Course Project

• Quizzes: 25%
  • Take any two of four quizzes

• Course project: 50%
  • Description: 5%
  • Mid-term report: 10%
  • Final report: 35%
  • Feedback and mark in each report

• This option is required for MASc and PhD students
Grading Policy: All Quizzes

- Quizzes: 10%
  - Take four quizzes
- This option is suggested for MEng students