Designing Modern Web-Scale Applications

Ashvin Goel

Electrical and Computer Engineering
University of Toronto

ECE1724

Topics

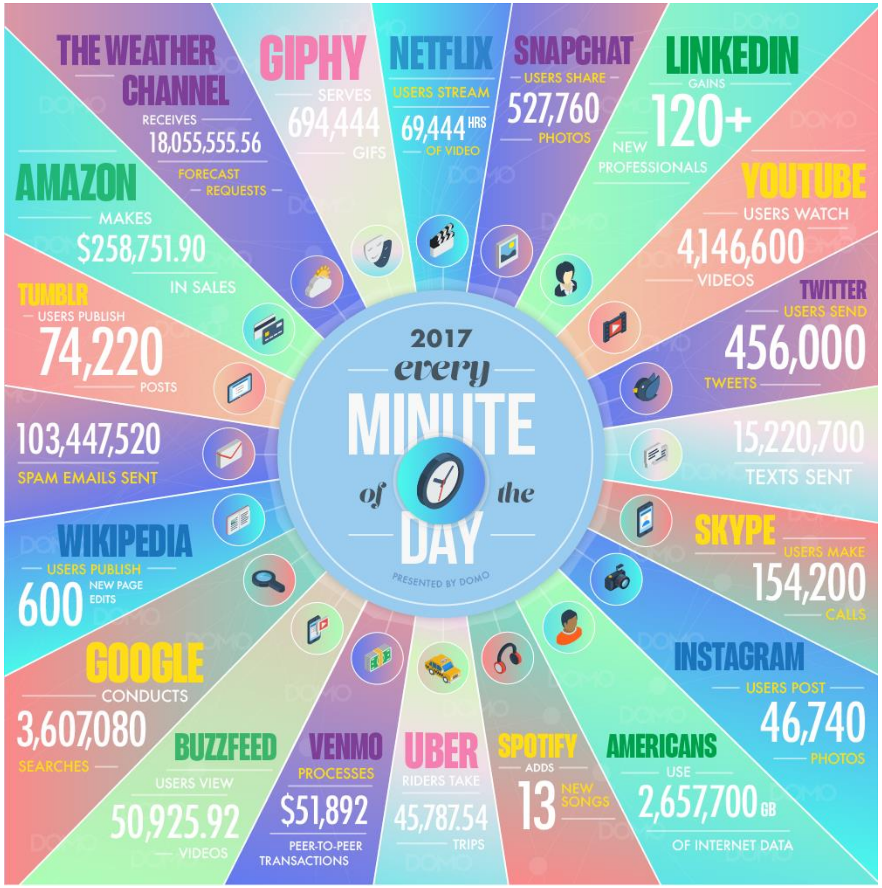
- Overview of the course
- Class format
- Introduction to the course

My Research Background

- Systems software
  - Operating systems
  - Storage systems
  - 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
  - Cloud-scale storage systems
  - Cloud-scale computation models
- 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

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

- ECE1724H: Advanced web development
  - Teaches you to develop interactive web and mobile apps
  - Uses modern web development frameworks such as React
- ECE1779: Intro to cloud computing
  - Introduces you cloud computing platforms
  - Gain experience with cloud technologies
  - Use Azure, Google App Engine, Amazon AWS Lambda, etc.
- ECE1724: Designing modern web-scale applications
  - 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

Industrial Relevance

- Many papers in the reading list are from industry
  - GFS, MapReduce, Bigtable, Spanner, Twine, Millwheel, Pregel (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

- Operating systems
- Distributed 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
- Cluster storage systems
- Wide-area storage systems
- Data parallel frameworks
- Scheduling and resource management
- Stream processing systems
- Graph processing systems
- Machine learning systems (if time permits)

Class Format

Overview

- Course web site
- Course readings
- Course project and presentation
- Quizzes
- Grading policy

Course Website

- Course website available from my home page
  - 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
- Email
  - If you want to contact me directly

Course Readings

- Advanced, discussion-oriented course
    - Course discusses seminal and some recent research papers
- Background in distributed systems, databases, OS
- Typically, we will discuss 2 papers per week
- Expect to spend 4-6 hours reading these papers

Course Project and Presentation

- Choose a project based on topics covered
- Sample topics will be posted on website
- Background reading material available 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

- Four quizzes
- Roughly every 3 weeks
- Quiz will cover topics discussed in previous 3 weeks
- Questions will vary, some straight forward, some open ended, will assess conceptual and critical understanding of course content
- Format
  - Typically, 1.5 hours
  - In class, on paper, but outside class hours
- Dates and location will be announced on Quercus

Grading Policy

- No assignments
- No final exam

- Two options
  - Course Project
  - All Quizzes

Grading Policy: Course Project

- Take two out of four quizzes, each quiz is 25%
- Course project: 50%
  - Description: 5%
  - Mid-term report: 10%
  - Final report: 35%
  - Feedback and mark provided for each report
- This option is required for MASc and PhD students
- Please make sure to discuss your potential project ideas with the instructor

Grading Policy: All Quizzes

- Take four quizzes, each quiz is 25%
- This option is
  - Required for MEng students that have not taken an undergrad course in distributed systems
  - Suggested for other MEng students