

CS 525: Advanced Distributed Systems

Course Overview

Over the past few decades, the functions of “traditional” operating systems have been scattered out to the edges of distributed systems. Peer-to-peer systems (think: Kazaa), sensor networks, the Grid, PlanetLab, the Internet and the Web are examples. More recently, these functionalities are being consolidated in datacenters and computing clouds (e.g., AWS, Azure, Google Compute Engine, and many private clouds, etc.). These infrastructures are being used to store and process various kinds of Big Data. This course focuses on three case study areas, spanning the classical to bleeding edge: **cloud computing**, **peer-to-peer systems**, and **sensor networks**. We will study efficient protocol design and evaluation, as well as learn high-level system issues. Research in these areas also tends to be scattered across disjoint sets of researchers and conferences, and the course attempts to study commonalities. Projects can be one of two types - **research projects** or **entrepreneurial projects**.

Prerequisites

Basic Computer Science and basic computer programming skills are essential. Knowledge of Operating Systems (CS 241 or CS 423), or Networks (CS 438), or Distributed Systems (CS 425), or an equivalent course, or instructor consent, is required.

Course Website <http://courses.engr.illinois.edu/cs525/>

Timings

Class: Tuesday and Thursday, 3:30 PM - 4:45 PM, 1310 DCL (Digital Computer Laboratory).
Professor Office Hours (class days only): Tuesday and Thursday 4.45 PM - 5.30 PM, 3112 SC.

Course Staff

Dr. Indranil Gupta (Professor)
3112 SC, indy@illinois.edu, 265-5517

Mainak Ghosh (TA)
mghosh4@illinois.edu

Course Content

The first few weeks of lectures introduce ground basics in cloud computing, peer to peer systems, distributed algorithms and sensor networks. Subsequently, over 70 research papers in various areas of distributed systems are presented, discussed, and debated by the students. The selection includes papers from industry and successful companies, as well as classical and contemporary papers from conferences including, but not limited to, SOSP, OSDI, SOCC, FAST, NSDI, PODC, VLDB, Middleware, Usenix, Infocom, SIGCOMM, SASO, etc., as well as ACM and IEEE journal papers. Rather than running through all the papers in a few of such proceedings and journals, we will pick and choose publications appropriate to the stated goals of this course.

An essential component of this course is a project that could be either a research project or an

entrepreneurial project (your choice, but you must make this choice at the semester start).

A research project will solve a cutting-edge research problem with innovative ideas, build systems solutions (real implementation required!), and evaluate it experimentally in a deployment scenario. The goal of this project is a conference-quality paper submission and acceptance; many past course projects have been published in conferences and journals.

An entrepreneurial project will explore a new idea for a startup or non-profit company. You will have to build the system or working prototype (real implementation required!) and evaluate it experimentally. You will also have to write a Business Plan. Think about who your potential customers are.

In all projects (research or entrepreneurial), the core ideas have to come from the students. The instructor will work with you in refining your ideas. In some cases, the instructor might offer initial ideas for projects.

To get ideas for projects, please DO be courageous and read ahead in the course schedule! Your project topic is expected to be distinct from your “regular” research project (i.e., you can’t type-cast the main project you’re doing with your advisor research as your CS525 project.). However, exceptions can be made in some cases - please talk to Indy if you need an exception.

It is highly recommended that you work in groups. Teams are suggested to be 2 to 3 students. All projects will have to submit three versions of reports (survey, midterm, and final reports) - for research projects, these submissions should read like research papers (or parts thereof), while for entrepreneurial projects, these submissions should be readable documentations of ideas, systems, implementations and (a required) business plan.

At the end of the course, a few “best” projects will be selected. All promising projects can be continued after the semester, e.g., for conference submission or at the Illinois EnterpriseWorks startup incubator.

This course’s projects in past semesters have been extremely successful. Research papers have been accepted in conferences (e.g., ICDCS, Infocom, Middleware, MMCN, SASO, GRID, etc.), as well as appear in several top journals (e.g., ACM TAAS, ACM TOSN, IEEE TNSM, JSS, Distributed Computing, etc.). Some course project papers have won a Best Paper Award.

About the Class

The initial few weeks of class will consist of lecturing, with the intent of building up common knowledge and grounding for the latter half of the course. We will then transition to student-led presentations of papers. Once student-led presentations start, students who are not presenting are expected to write short reviews (1-2 pages total) for the “Main Papers” in that session. Active class participation is required, even in the initial part of the course!

Class Evaluation: Project (survey + midterm + final), reviews of papers, presentation, and class participation (may include a quiz). Tentative splits are 50% (5% + 15% + 30%), 20%, 25%, 5% respectively.

Abridged list of Topics (see course website for more comprehensive list): Cloud computing, theory, peer to peer systems, sensor networks, overlays, cloud scheduling, cloud programming, storage, key-value stores, NoSQL, stream processing, consistency, adaptivity, reliability, graph processing, epidemics, in-network processing, membership, industrial systems, low latency, measurement studies, structure of networks.