CAS CS 561: Data Systems Architectures

Boston University

Spring 2023

Class Syllabus

Course Description: Data is everywhere. As scientists, users, and citizens we are both generating and exploiting large, ever-growing, diverse sets of data. For several applications – ranging from scientific discovery to business analysis, governance, and everyday activities – we are directly using and indirectly affecting hundreds of data systems! The big challenge is to turn data into useful knowledge, and to do so quickly, in order to increase the impact of the new insights. Achieving these goals comes with a number of technical challenges. How to exploit the continuously evolving hardware (storage, computation, network)? How to collect all incoming data efficiently? How to query dynamic collections of data that keep accumulating incoming data? How to parallelize query processing from one core to a few (scale-up), and then to thousands (scale out)? What are the needs of evolving workloads (hybrid transactional/analytical processing, graph analytics, Internet-of-Things, micro-payments, monitoring)? In this course, we will discuss how to design data systems that can address these challenges. We will see in detail the two driving forces behind innovation in data systems: hardware and workloads, and we will discuss recent and future trends of both. We will use examples from several data management areas including relational systems, distributed database systems, key-value stores, newSQL and NoSQL systems, data systems for machine learning (and machine learning for data systems), interactive analytics, and data management as a service. In a quickly moving industry and research landscape, such skills are essential.

Prerequisites: The class requires familiarity with database systems at the level **CS460/660**, and with algorithms, data structures, computer systems, and system programming at the level of **CS210**. Please see the instructor if you are not sure about the level of your preparation.

Instructor: Subhadeep Sarkar (ssarkar1@bu.edu)
Office hours: Tu/Th, 2-3pm
Office: CCDS 929 (@665 Comm. Ave.)

Teaching Assistants: Zichen Zhu (OH details available on class website)

Meeting Times and Places
Lectures: Tu/Th, 12:30-1:45 pm, room MCS B37
Lab: F, 9:05-9:55 am, room CAS 116

Course Website: https://bu-disc.github.io/CS561/

Textbooks (not required): There is no textbook that covers cutting edge research, however, the data management community has produced top quality textbooks that can serve as reference to provide background material. The class is based on recent research papers which will be available to you through the BU network.

- R. Ramakrishnan and J. Gehrke. *Database Management Systems*. Third Edition. McGraw-Hill 2002.

An excellent collection of classic papers in the database field is the following:

- Readings in Database Systems. P. Bailis, J. Hellerstein, M. Stonebraker, editors.

Other good background material is:

- Architecture of a Database System. By J. Hellerstein, M. Stonebraker and J. Hamilton. Foundations and Trends in Databases, 2007
- The Design and Implementation of Modern Column-store Database Systems. By D. Abadi, P. Boncz, S. Harizopoulos, S. Idreos, S. Madden. Foundations and Trends in Databases, 2013
- Massively Parallel Databases and MapReduce Systems. By Shivnath Babu and Herodotos Herodotou. Foundations and Trends in Databases, 2013

Course Requirements

Project 0: The first requirement for the class is a small implementation project at the beginning of the semester. *Project 0* will be carried out by **each student independently** during the first three weeks of the semester. Its goal is to prepare you for the semester project by sharpening your development skills.

Project 1: After Project 0, the class will have another short project that aims to expose students to different types of data systems and their internal APIs. *Project 1* will be carried out by a **group of 3 students**. Its goal is to further prepare you for the semester project by exposing you to real-life data systems.

Paper Presentation: After the initial 5 classes all students will take turns presenting papers. In each class, we will discuss one (or two) main paper(s) (and there will be a few background papers), and each student will present once in the semester, either alone or as a group of two students). The student(s) presenting will be responsible to outline the strong and the weak points of the paper and proposing at least one idea for improving the approach presented in the paper. **All students** will read the presented paper.

Paper Reviews & Technical Questions: All students should read all papers. *Reading the paper and writing a review is very important to help the students prepare for the class presentation and discussion.* Every student is expected to deliver a **review for 3 papers** and **answer 9 technical questions from (a subset of) all other papers**. Each paper will be clearly marked as a paper for review or a paper for a technical question (which will be provided well before the class). Every review/answer of TQ for a given paper has to be submitted **before the class,** having the class starting time as a hard deadline.

A ***review*** consists of a few paragraphs answering the following questions: (i) what is the problem, (ii) why it is important, (iii) why is it hard, (iv) why older approaches are not enough, (v) what is the key idea and why it works (a list of at least three key points), (vi) what might be missing and how can we improve this idea (a list of at least three key points), (vi) an evaluation as to whether the paper supports its claims, and (vii) possible next steps of the work presented in the paper. The ideal size of the review is about 1 page, single column, 10pt font, 1-inch margin (and it may only exceed 1 page if the student wants to elaborate on how to improve the ideas on the paper).

Class Project: Finally, this class requires a semester-long project and a final report in the style of a conference paper. The project will be either implementation-heavy or research-oriented. Students will work in **groups of 3** (for the implementation or the research project) and after the first two weeks, each team will have been associated with a specific project. Students can propose their own research project upon approval by the instructor.

The overall grade will be based on the following policy:

- Class participation: 5%
- Project 0: 10%
- Project 1: 15%
- Paper reviews: 5%
- Technical questions: 10%
- Paper presentation: 15%
- Project proposal: 5%
- Mid-semester project progress report: 5%
- Project: 30%

Topics: Throughout the class, we will cover data systems design principles from the following different angles.

1. What affects new data systems designs (data and applications, emerging hardware, and new workloads)
2. Traditional Data Systems for Modern Hardware
3. Distributed Database Systems
4. Scale-out Systems: from Map-Reduce to SQL-on-Hadoop
5. NoSQL, NewSQL, and Key-Value stores

Important Dates for all classes (more info at: https://www.bu.edu/reg/calendars/semester/)
February 1st, last day to add a class
February 23rd, last day to drop (without a "W")
March 31st, last day to drop (with a "W")

Important Dates for CS 561
February 4th, last day to select a project
February 18th, submit the project proposal
March 25th, submit the mid-way project report
April 30th, submit final project report

Tentative Schedule

Lecture 1	**Introduction to Data Systems and CS561**
Lecture 2	**Data Systems Architectures Essentials – Part 1**
Lecture 3	**Data Systems Architectures Essentials – Part 1**
Lecture 4	**Class Project Overview**
Lecture 5	**Storage Layouts:** Row-Stores vs. Column-Stores
Student talk 1	**Storage Layouts:** Adaptive & Hybrid Layouts
Lecture 6	**Modern Storage Engines:** LSM-Trees
Lecture 7	**Modern Storage Engines:** Deletes in LSMs
Lecture 8	**Modern Storage Engines:** Robust LSM Designs
No Class, Monday replacement day	
Lecture 9	**Modern Storage Engines:** Bloom Filters in LSM-Trees
Lecture 10	**Indexing:** Introduction to Indexing, Trees & Tries
Student talk 2	**Indexing:** Adaptive Radix Trees
No Class – Spring Break	
Student talk 3	**Indexing:** Adaptive Indexing & Cracking
Lecture 11	**Indexing:** Sortedness-Aware Indexing
Lecture 12	**Modern Hardware:** Modern Hardware Trends
Lecture 13	**Modern Hardware:** Parametric I/O Model
Student talk 4	**Modern Hardware:** HTAP Systems
Student talk 5	**Modern Hardware:** SSD-Conscious Designs
Student talk 6	**Modern Hardware:** Data Processing with GPUs
Lecture 14	**Modern Hardware:** Relational Memory
Student talk 7	**Serverless Computing:** The Paradigm
Student talk 8	**Serverless Computing:** Execution
Lecture 15	**ML for Data Systems:** Learned Indexes
Lecture 16	**ML for Data Systems:** Cosine
Lecture 17	**ML for Data Systems:** TBD
Project 1	**Project Presentations A**
Project 2	**Project Presentations B**

Collaboration Policy

You are strongly encouraged to collaborate with one another in studying the lecture materials and preparing for reviews and presentations.

You may discuss ideas and approaches to the projects with others (provided that you acknowledge doing so in your solution), but such discussions should be kept at a high level and should not involve actual details of the code or of other types of answers. However, **Project 0 is an individual project**, and all students must complete it on their own. **You must also complete the reviews and technical questions on your own.**

Academic Misconduct
We will assume that you understand BU's Academic Conduct Code:
http://www.bu.edu/academics/policies/academic-conduct-code

Prohibited behaviors include:

- copying all or part of someone else's work, even if you subsequently modify it; this includes cases in which someone tells you what you should write for your solution
- viewing all or part of someone else's work
- showing all or part of your work to another student
- consulting solutions from past semesters, or those found online or in books
- posting your work where others can view it (e.g., online).

Incidents of academic misconduct will be reported to the Academic Conduct Committee (ACC). The ACC may suspend/expel students found guilty of misconduct. ***At a minimum, students who engage in misconduct will have their final grade reduced by one letter grade (e.g., from a B to a C).***