

# CS 3410: Distributed Systems

Spring 2024	Topics	Paper (due Wednesday)
Jan 8-12	<a href="#">Go, RPC</a>	1. Google File System
Jan 15-19 ( <i>MLK Day</i> )	<a href="#">Go examples</a>	2. Bigtable
Jan 22-26	Effective Go, replicated state machines	3. Paxos
Jan 29-Feb 2	TCP, sockets, clusters	4. Case study: Google
Feb 5-9	coherent caching, CAP	5. Chubby
Feb 12-16	transactions, 2-phase commit	6. Megastore
Feb 19-23 ( <i>President's Day</i> )	time, clocks, snapshots	7. Spanner
Feb 26-Mar 1	peer to peer	8. Chord
Mar 4-8	concurrency, actors	9. Case study: Facebook
Mar 11-15 ( <i>Spring Break</i> )	—	—
Mar 18-22	databases	10. Calvin
Mar 25-29	big data	11. MapReduce
Apr 1-5	SOA, microservices	12. Dynamo
Apr 8-12	eventual consistency	13. S3 Node
Apr 15-19		14. RDDs (Spark)
Apr 22-26 ( <i>Thursday last day</i> )		—

Changes to the schedule will be announced in class.

## Resources

- [Syllabus](#)
- [Examples from class](#)
- [Effective Go](#)
- Recommended book: [The Go Programming Language](#)
- [Go package docs](#)
- [Screencast on setting up Go and vim-go](#)
- TCP videos
  - [TCP service model \(16:27\)](#)
  - [The end-to-end principle \(10:33\)](#)
  - [Sliding window \(19:25\)](#)
  - [Retransmission strategies \(9:45\)](#)
- RPC demo app in Go
  1. [introduction \(8:22\)](#)
  2. [server RPC \(3:01\)](#)
  3. [client RPC \(4:50\)](#)
  4. [command-line flags \(13:58\)](#)
  5. [call function \(6:58\)](#)
  6. [client shell \(14:35\)](#)
  7. [actor \(15:23\)](#)
- [Paxos assignment slides](#)
- [RPC chat assignment](#)

## Papers

1. [The Google File System](#)
2. [Bigtable: A Distributed Storage System for Structured Data](#)
3. Paxos
  - [Paxos Made Simple](#)
  - [Paxos vs Raft: Have we reached consensus on distributed consensus?](#)
  - [Paxos in 25 lines](#)
4. Case study: Google
  - [Web search for a planet: The Google Cluster Architecture](#)
  - [Building Software Systems at Google and Lessons Learned \(video 1:22:44\)](#)
5. [The Chubby lock service for loosely-coupled distributed systems](#)

6. [Megastore: Providing Scalable, Highly Available Storage for Interactive Services](#)
  7. [Spanner: Google's Globally-Distributed Database](#)
  8. [Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications](#)
  9. Case study: Facebook
    - [Scale at Facebook](#) (video, 1 hour)
    - [Needle in a haystack: efficient storage of billions of photos](#) (details about one specific service)
  10. [Calvin: Fast Distributed Transactions for Partitioned Database Systems](#)
    - Recommended: skim this paper first: [The Case for Determinism in Database Systems](#)
  11. [MapReduce: Simplified Data Processing on Large Clusters](#)
  12. [Dynamo: Amazon's Highly-available Key-value Store](#)
  13. [Using Lightweight Formal Methods to Validate a Key-Value Storage Node in Amazon S3](#)
  14. [Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing](#)
- 

## Presentations

- [Managing Update Conflicts in Bayou, a Weakly Connected Replicated Storage System](#)
- [Practical Byzantine Fault Tolerance](#)
- [Impossibility of Distributed Consensus with One Faulty Process](#)
- [The Byzantine Generals Problem](#)
- [Session Guarantees for Weakly Consistent Replicated Data](#)
- [CAP Twelve Years Later: How the "Rules" Have Changed](#)
- [Distributed Snapshots: Determining Global States of Distributed Systems](#)
- [Life beyond Distributed Transactions: an Apostate's Opinion](#)
- [Scale and Performance in a Distributed File System \(AFS\)](#)
- [Petal: Distributed Virtual Disks](#) (Austin S, ??)
- [On Designing and Deploying Internet-Scale Services](#)
- [Dapper, a Large-Scale Distributed Systems Tracing Infrastructure](#)
- [PNUTS: Yahoo!'s hosted data serving platform](#)
- [Mesa: Geo-Replicated, Near Real-Time, Scalable Data Warehousing](#)
- [High-Availability at Massive Scale: Building Google's Data Infrastructure for Ads](#)
- [Twitter Heron: Stream Processing at Scale](#) (Lake I, Lexi P)
- [Large-scale Incremental Processing Using Distributed Transactions and Notifications](#)
- [F1: A Distributed SQL Database That Scales](#) (Calvin H, Jack W, Luke G)
- [Paxos Made Live—An Engineering Perspective](#)
- [Flexible Paxos: Quorum intersection revisited](#)
- [Large-scale cluster management at Google with Borg](#) (Thomas K, Gabe T, Jeremy H)
- [Time, Clocks, and the Ordering of Events in a Distributed System](#)
- [Exploiting virtual synchrony in distributed systems](#)
- [Conflict-free Replicated Data Types](#)

Here is another list of papers to draw from:

- [Foundational distributed systems papers](#)
- [Hall of fame awards](#). These are systems papers that have been recognized as especially important, though note that only some of them are distributed systems papers.