## **CS 3410: Distributed Systems**

Topics	Paper (due Wednesday)
Go, RPC	1. Google File System
<u>Go examples</u>	2. Bigtable
Effective Go, replicated state machines	3. Paxos
TCP, sockets, clusters	4. Case study: Google
coherent caching, CAP	5. Chubby
transactions, 2-phase commit	6. Megastore
time, clocks, snapshots	7. Spanner
peer to peer	8. Chord
concurrency, actors	9. Case study: Facebook
-	—
databases	10. Calvin
big data	11. MapReduce
SOA, microservices	12. Dynamo
eventual consistency	13. S3 Node
	14. RDDs (Spark)
	—
	Topics Go. RPC Go examples Effective Go, replicated state machines TCP, sockets, clusters coherent caching, CAP transactions, 2-phase commit time, clocks, snapshots peer to peer concurrency, actors  databases big data SOA, microservices eventual consistency

Changes to the schedule will be announced in class.

## **Resources**

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

## **Papers**

- 1. <u>The Google File System</u>
- 2. <u>Bigtable: A Distributed Storage System for Structured Data</u>
- 3. Paxos
  - <u>Paxos Made Simple</u>
  - 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. <u>The Chubby lock service for loosely-coupled distributed systems</u>

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

## **Presentations**

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

Here is another list of papers to draw from:

- Foundational distributed systems papers
- <u>Hall of fame awards</u>. These are systems papers that have been recognized as especially important, though note that only some of them are distributed systems papers.