Today
- Dynamo
- End class: what have we learned?
  (wrap up)
- Admin
  - Class presentations
  - Midterm #2
  - Other

Dynamo
Problem: DBs of time ⇒ didn't scale
Goals:
  Scale (incremental)
Availability: at all costs
⇒ Brewer: "CAP" theorem ⇒ truism
  avail
  consistency partition tolerance
⇒ pick any two
Dynamo: availability
  - always writeable
  - conflict resolution on reads (later)
  - just accept writes

Conflicts:
  - app-specific (Bayou)

Performance:
  - latency = tail latencies (99.9th percentile)
    - predictable

Service Level Agreements
  - known h/w
  - admission control
System Arch

API: get/put "key value"
hash(key) => lookup

Partitioning:
"consistent hashing"

virtual nodes:
→ load balance under join/leave
→ hetero.
Replication: \( N \)-way

writes: \( \rightarrow \) coordinator

\[\text{replicate key/value } \rightarrow \text{ success}\]

\( \text{careful (N distinct phys modes)} \)

\( N = 3 \)

Versions: "eventual consistency"

\( W \) ......

\( \{ R \} \) may see new or old

\( R \)
e.g. shopping cart

how to map shop cart ops onto get/put?
   => op => put  add cart => put
     del cart => put

~ versions
=> vector clocks
   list of (node, counter) ... allows detection of
     conflicts, non-conflicts

concern: size => lists can get big
   limit: 10

Get/put:
=> routing
  2 options:
   -> load balancing tier
   -> client

\[-v_1\]
\[-v_2\]
\[-v_3\]
\[-v_4\]
\[-v_5\]
writes: coordinator -> replicate to first N-1 "healthy" nodes
quorum: N, R, W (read write)
"sloppy" quorums: 
  -> available
  replicate to N healthy nodes
  \[
  N \xrightarrow{\text{recover}} N_2 \rightarrow N_3
  \]
  list: eventually contact N_1 = replica "hinted handoff"

Replicas Sync.
  periodic nodes sync w/ each other
  merkle trees over key/values
  \[\Rightarrow \text{efficient sync}\]
Experience + Lessons

**Perf vs. Durability:**

- Write buffer: 5x tail latency
- Write to N nodes

\[ Q \ w=2 \]

Load Distribution

Fixed-size partition

M Trees
Background Tasks:
  replica sync, etc.
  => bg tasks can slow fg tasks

Schedule bg work:
  → driven by measurement
  → fg tail latency
  → throttle bg work accordingly

\[ \text{\underline{M}} \sim \text{TCP} \]

Conclude:
  Early NoSQL → Cassandra Risk

Interesting:
  wasn’t super popular inside Amazon (@first)
  → system, not service (includes ops.)
Presentations:

20 minute talk
slides, bring laptop

Structure:

1) Intro → Title, Who
   → Problem
   → Solution
   → Summary

2) Body

3) Conclusion
   → Summary
   → Thoughtful conclusion

Midterm: no Thanksgiving questions!

= take home typing OK

Sun a.m. →
Mon p.m. (6)

sleep
Project write up:

\[
\begin{array}{l}
\text{Short} \\
5\text{-pages (4)}
\end{array}
\]