Paxos wrapup

Doug Woos
Logistics notes

Whence video lecture?

Problem Set 3 out on Friday
Paxos Made Moderately Complex Made Simple
When to run for office

When should a leader try to get elected?

- At the beginning of time
- When the current leader seems to have failed

Paper describes an algorithm, based on pinging the leader and timing out

If you get preempted, don’t immediately try for election again!
Reconfiguration

All replicas *must* agree on who the leaders and acceptors are.

How do we do this?
Reconfiguration

All replicas *must* agree on who the leaders and acceptors are

How do we do this?

- Use the log!
- Commit a special reconfiguration command
- New config applies after WINDOW slots
Replicas

WINDOW=2

Leader

Replica

slot_out

reconfig(L, A) Put k1 v1

App k2 v2

slot_in

Reconfiguration

What if we need to reconfigure *now* and client requests aren’t coming in?
Reconfiguration

What if we need to reconfigure now and client requests aren’t coming in?

- Commit no-ops until WINDOW is cleared
Other complications

State simplifications

- Can track much less information, esp. on replicas

Garbage collection

- Unbounded memory growth is bad
- Lab 3: track finished slots across all instances, garbage collect when everyone is ready

Read-only commands

- Can’t just read from replica (why?)
- But, don’t need their own slot
Data center architecture

Doug Woos
The Internet

Theoretically: huge, decentralized infrastructure

In practice: an awful lot of it is in Amazon data centers
  - Most of the rest is in Google’s, Facebook’s, etc.
The Internet
The Internet
Data centers

10k - 100k servers

100PB - 1EB storage

100s of Tb/s bandwidth
  - More than core of Internet

10-100MW power
  - 1-2% of global energy consumption

100s of millions of dollars
Servers in racks

19” wide
1.75” tall (1u)
(convention from 1922!)
~40 servers/rack
- Commodity HW
Connected to switch at top
- ToR switch
Racks in rows
Rows in hot/cold pairs
Hot/cold pairs in data centers
Where is the cloud?

Amazon, in the US:

- Northern Virginia
- Ohio
- Oregon
- Northern California

Why those locations?
Early data center networks

3 layers of switches

- Edge (ToR)
- Aggregation
- Core
Early data center networks

3 layers of switches
- Edge (ToR)
- Aggregation
- Core
Early data center limitations

Cost

- Core, aggregation routers = high capacity, low volume
- Expensive!

Fault-tolerance

- Failure of a single core or aggregation router = large bandwidth loss

Bisection bandwidth limited by capacity of largest available router

- Google’s DC traffic ~doubles every year!
Clos networks (1953)

How can I replace a big switch by many small switches?
Clos networks (1953)

How can I replace a big switch by many small switches?
Figure 3: Simple fat-tree topology. Using the two-level routing tables described in Section 3.3, packets from source 10.0.1.2 to destination 10.2.0.3 would take the dashed path.

To reduce costs, thin out top of fat-tree
Multipath routing

Lots of bandwidth, split across many paths
Round-robin load balancing between any two racks?
  - TCP works better if packets arrive in-order
ECMP: hash on packet header to determine route
Data center scaling

“Moore’s Law is over”
- Moore: processor speed doubles every 18 mo
- Chips still getting faster, but more slowly
- Limitations: chip size (communication latency), transistor size, power dissipation

Network link bandwidth still scaling
- 40 Gb/s common, 100 Gb/s coming
- 10-100 µs cross-DC latency

Services scaling out across the data center
Local storage

Old: magnetic disks — “spinning rust”
Now: solid state storage (flash)
Future: NVRAM
Persistence

When should we consider data persistent?

- In DRAM on one node?
- On multiple nodes?
- In same data center? Different data centers?
- Different switches? Different power supplies?
- In storage on one node? etc.