CSE 536 (Adv OS)

Agenda
1. Dist. Sys. Theory Quorum Systems
2. GFS cont.

- We have seen how consistent replicas can be maintained.
- Replicas?
  - fault tolerance
  - improving performance.
- Quorum system - generalization of majority rule. (sec. 10.3.5)

Nr: num of replicas.
Qrd: read quorum - number of replicas you read from
Qwr: write quorum - nr of replicas the app. writes to

Qrd, Qwr ≤ Nr, Qrd + Qwr > Nr

Remember we need to guarantee consistency (r-w consistency)
e.g. UNIX semantics:

- a read/write op is complete only if Qrd/Qwr replicas have been performed
- for read Qrd values should match
- for write Qwr writes should have completed

S13.1
As we will see, however, the system may not be solved analytically.

One important point to remember is:

It is not a part of a fractal theorem.

Diff. choose a $q_i, r_i, a_i, c_i$ and give different characteristics.

From within:

As read and alone (carry)

$N = 5$

$\frac{3}{2}$

$\frac{1}{2}$

$\frac{1}{2}$

$\frac{1}{2}$

$\frac{1}{2}$

$\frac{1}{2}$

$\frac{1}{2}$

$\frac{1}{2}$
Index File

1. More likely scrubbed and/or dirty
2. Lots of comment - don't account
3. Likely benchmark
- many files are in a directory
- random write is in progress
- most updates are append

I. Fix me - large
-\textit{unix - keep user-priv}

\underline{Workload Characterization:

- for write, no def.
- performance: cf. experience for a frame recycled data
- interface: non-posix
- design is for failure
- concurrency,rich semantics
- huge files, nearly append ops
- graf,overload - (6.5 dot)
- NFS/unix - keep user-privated

\texttt{GF}
Affinity & concurrency matters.
- Ocean doesn't matter for append, but
  - log-based
  - use ftlo or producer consumed group
  - f/c (map/reduce)

Capacity matters for shuffle.
- Expand or prevent failures.
Read Protocol

I

- App
- Master
- Cache

Metadata is cached!
- Why? To avoid making master a bottleneck.

But Data is not cached!

II

Write Protocol

Primary Replic: Holds the current lease
Secondary Replic: Loose
- All updates go thru' primary replica
- Secondary replica follows the primary in the same order
- Primary decides on the order of updates & secondary replicas do the same
- Lease is used for fault tolerance.

- Master selects the chunkservers and grants it lease for a chunk