Storage Systems

NPTEL Course

Jan 2012

(Lecture 22)

K. Gopinath

Indian Institute of Science

Redundancy

- Critical structures replicated
  - Superblocks
  - Large scale systems: inodes also...
- May depend on volume managers
  - RAID
- To avoid corruption (eg. from hw faults)
  - Extensive self-consistency checks in code (just like in an OS)
  - eg: is buffer alloc safe? Invariant: alloc during syscall & free at end
    - Disk drive hw problem: cannot interrupt CPU: buf lost!
  - Best Effort Service
    - Soft errors (retry), hard errors
    - If error detected in spite of checks, try to move fs to some reasonable consistent state
    - Usually no non-starvation guarantees
- End-to-end checksums (across disks, HBAs, netw links) needed

NFS: dir lockup

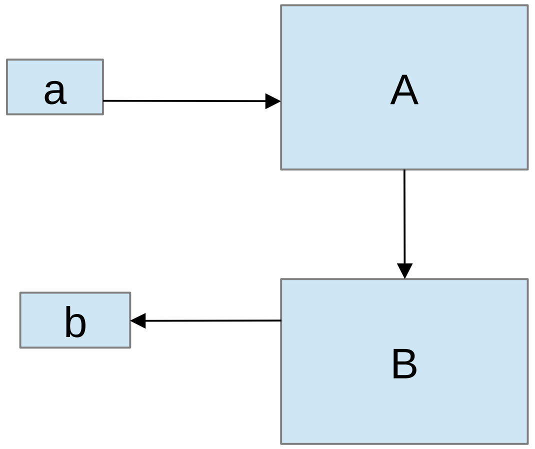
- Consider a “slow op” on a (locked) file due to NFS congestion
- VOP_LOOKUP on that file results in a lock on its dir that cannot be released until “slow op” finishes =>
  - cascade of locks upto root that hangs the system till “slow op” finishes
- lockd: similar problem but worse as lockd a user process

QoS

- BW
  - Minimize seeks/rotational delays (disks)
  - Aggregate requests for contiguity (disks/SCM)
- Latency
  - Scheduling requests to devices
  - Managing Parallelism across devices

Types of Designs

- Journaling FS
  - Transactional
- Soft updates
- Update in place?
  - Log structured FS
- Disk optimized?
- B-Tree based?
- Support triggers?



Abstractions Used

Buffer related

- getblk: given a filesystem number and disk block number, get its locked buffer
- brelse: given a locked buffer, wakeup waiting procs and unlock it
- bread: read a given disk block into a buffer
- breada: bread + asynch. read ahead
- bwrite: write a given buffer to a disk block
- alloc: allocate a free disk block and return buffer using getblk
- free: free a disk block

Inode related

- iget: get a locked inode (doing bread if necessary) given inode number
- iput: release an inode; if ref count 0, writes dirty inode
- bmap: given inode and byte offset, returns disk block num and offset
- namei: given a path, get the locked inode
- ialloc: assign a new disk inode for a newly created file
- ifree: free an inode (link count 0)