Lecture 18: RAID

- I/O bottleneck
- JBOD and SLED
- striping and mirroring
- “classic” RAID levels: 1 – 5
- additional RAID levels: 6, 0+1, 10
- RAID usage

Impending I/O crisis

- CPU speed is growing exponentially
- Memory size is growing exponentially
- I/0 performance is increasing only slowly
- computer systems will become I/O dominated
  - Amdahl law - the system is only as fast as it’s slowest component

Can we get more disks?

- Can we get more disks and access them in parallel - *disk array*?
- Advantage: disk access speeds up
- problem: mean time between failures MTBF decreases!
  - MTBF(disk array) = MTBF(disk) / # of disks
- idea - controlled redundancy of the information in disk array improves the MTBF as well as keeping disk access fast:
  - RAID - Redundant Array of Inexpensive Disks
  - JOBD - Just a Bunch of Disks
  - SLED - Single Large Expensive Drive

RAID uses *striping* and *mirroring*

- Disks are divided into independent *reliability groups*
- Striping - information is written in *stripes,* each stripe spans multiple drives.
  - Can be
    - bit interleaving - every bit belongs to a different portion of the logical “volume”
    - sector interleaving - every sector belongs to a different portion of the logical volume
  - advantage: reads and writes can be done in parallel
  - disadvantage: one disk fails - the information is lost
- mirroring - information is copied into two different disks
  - advantage: reads can be done in parallel, fault-tolerant
  - disadvantage: have to get 2X disks
- depending on the combination of the two techniques RAID is classified into 5 levels

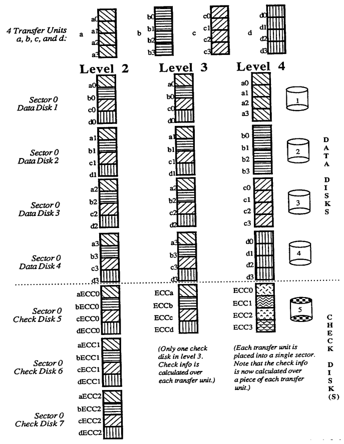
RAID levels 0, 1 and 2

- 0 – bit-interleaving striping
  - ok performance
  - low MTBF
- 1 - mirroring only
  - excellent reliability
  - high cost (must purchase 2X disks) -50% overhead
  - reads are slightly better (can read from either copy)
  - have to do two writes, can’t proceed until they complete
- 2 - use striping (with bit interleaving) and error correction code (ECC)
  - hamming ECC ensures that the error can be corrected, 20-40% overhead
  - reliable
  - performance is bad for small I/0 - have to read the whole stripe
  - can’t do I/O in parallel

RAID levels 3 and 4

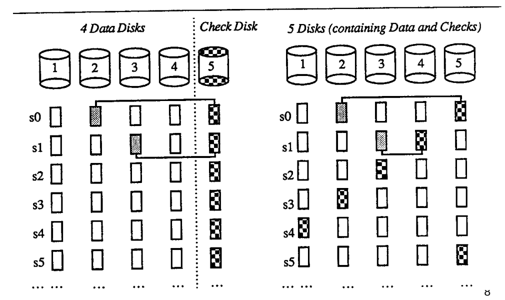
- disk controllers can recognize if the disk has failed!
- 3
  - only one parity disk per reliability group (4-10% overhead)
  - bit interleaving
  - reliability and performance is slightly better than 2 since we use fewer disks
- 4
  - use sector interleaving
  - large writes can go in parallel
  - independent / large reads can go in parallel
  - problem: parity disk is a bottleneck!

Comparison of RAID levels 2, 3 and 4

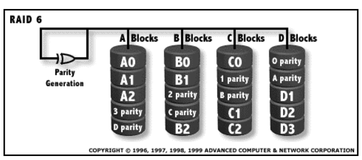


RAID level 5

- 5 - stripe and parity across all disks - no singe disk is a bottleneck

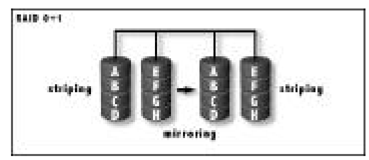


Raid level 6

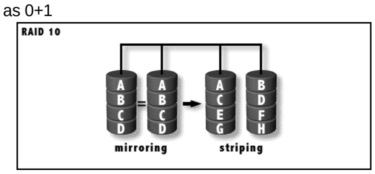


- two independent sets of parities (2-dimentional parity)
  - one - similar to RAID 5
  - two – across all disks for fault tolerance
- eval
  - can sustain 2 simultaneous disk crashes
  - second parity slows down writes, needs extra disk, expensive electronics to calculate parity

RAID Levels 0+1 and 10



- 0+1 – stripe then mirror
  - fault tolerance – can withstand single failure
  - performance – as good as mirroring and striping
- 10 – mirror than stripe
  - better fault tolerance than 0+1 (why?)
  - same performance as 0+1
- both techniques are expensive since 2X disks are needed



RAID applications

- *Hot spare* is maintained - if a “working” disk fails then the information is rewritten to hot spare
- RAID can be implemented in hardware or software
  - software RAID - OS calculates checksums and does writes to raids,
    - does not need special hardware
    - slow
    - not very fault-tolerant - OS crashes - RAID may go with it
  - hardware RAID - there is a separate CPU on the RAID; RAID’s CPU talks to the disks, calculates checksums and supplies the computer with “ready” data
    - faster
    - fault-tolerant
    - expensive?
- Most hardware RAIDs have “on-board” RAM to use as cache