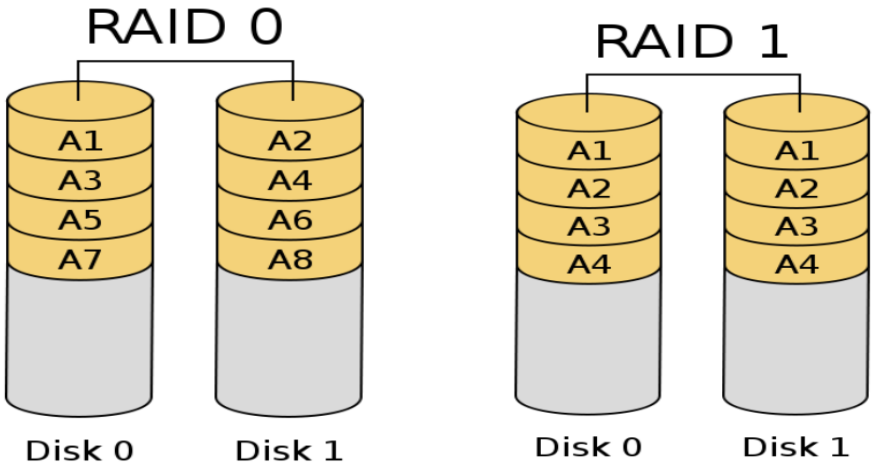
RAID 0 & RAID 1

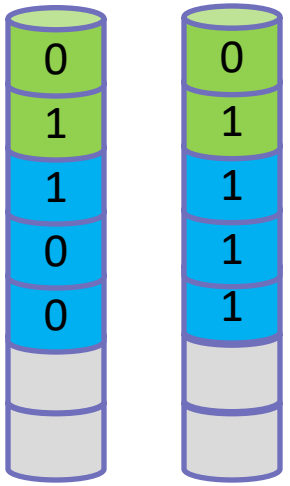
- RAID 0: non-redundant **striping**
  - Improve **performance** via **parallelism**
  - I/O bandwidth is proportional to the striping count
    - **Both read and write BW increase by N times (N is the number of disks)**
- RAID 1: **Mirrored** disks
  - Provide **reliability** via **redundancy**
    - **Read BW increases by N times**
    - **Write BW remains the same**



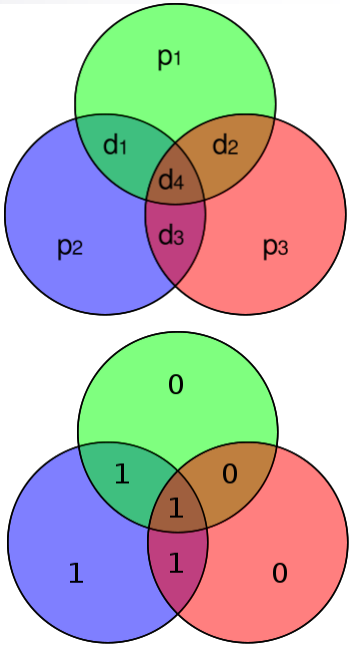
RAID0 Example

File1: 0011

File2: 110101



RAID 2: Hamming code

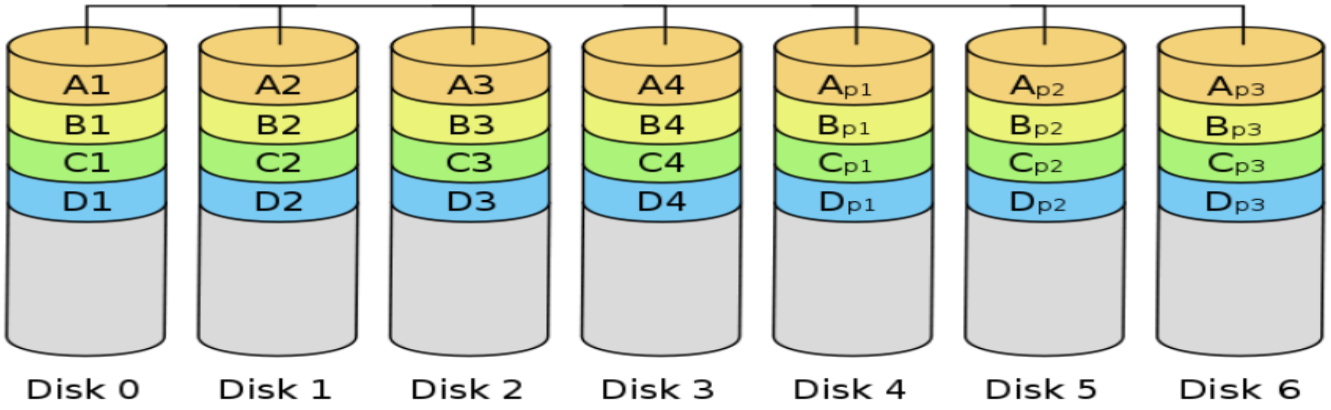


- E.g.: Hamming code(7,4)
  - 4 data bits (on 4 disks) + 3 parity bits (on 3 disks)
  - Each parity bit is linear code of 3 data bits

☺Recover from any single disk failure

  - Can detect up to two disks(i.e. bits) error
  - But can only "correct" one bit error

☺Better space efficient than RAID1 (75% overhead)



Hamming code reference: http://en.wikipedia.org/wiki/Hamming_code

RAID 3 & 4: Parity Bit

- Disk controller can detect whether a sector has been read correctly

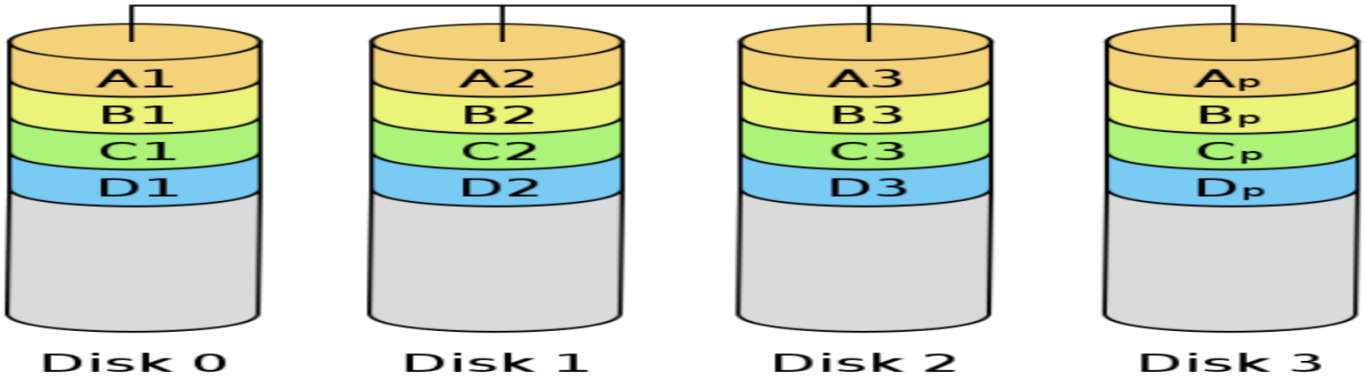
➔ a single parity bit is enough to correct error from a single disk failure

- **RAID 3**: Bit-level striping; **RAID 4**: Block-level striping

☺ Even better space efficiency (33% overhead)

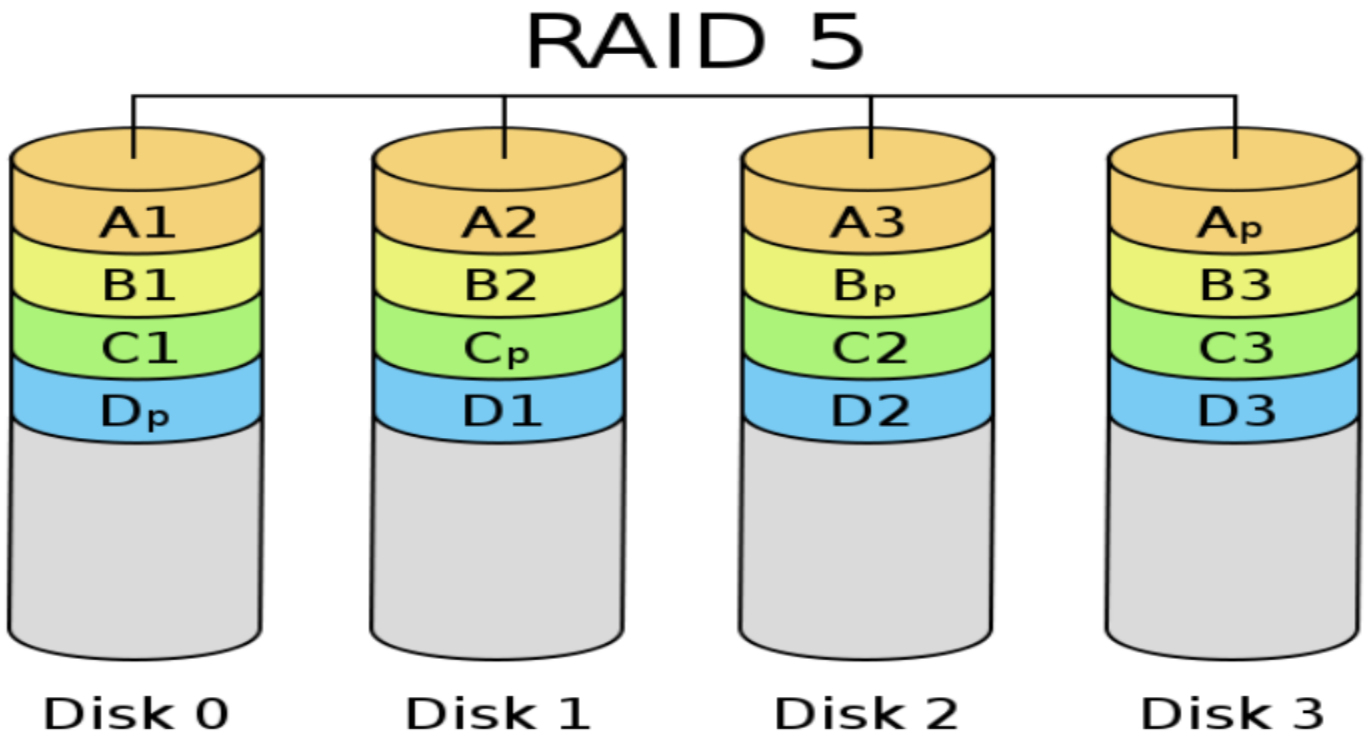
☹ Cost to compute & store parity bit

- RAID4 has higher I/O throughput, because controller does not need to reconstruct block from multiple disks



RAID 5: Distributed Parity

- Spread data & parity across all disks
- Prevent over use of a single disk (e.g. RAID 3,4)

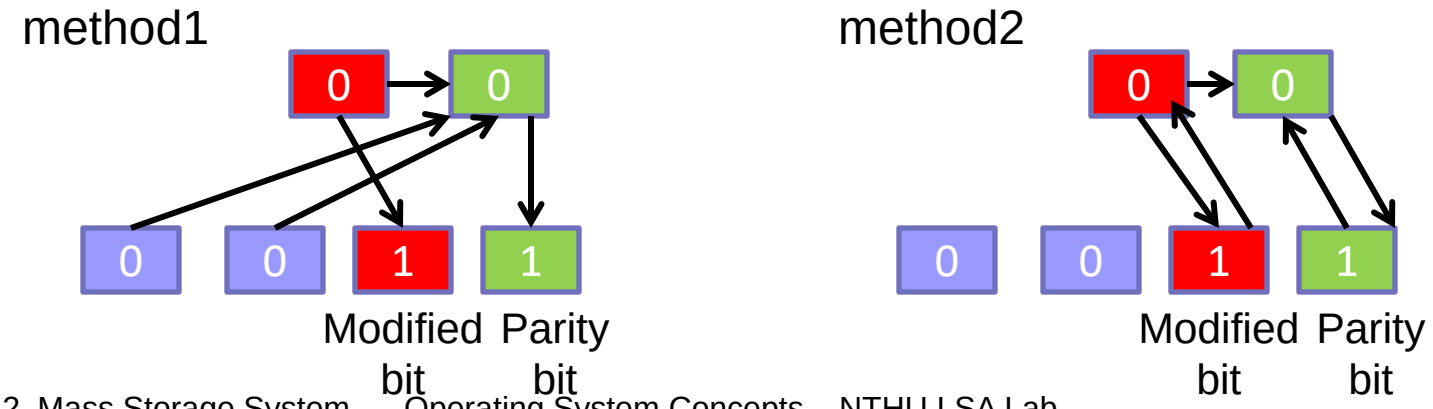


RAID 5: Distributed Parity

- Read BW increases by N times, because all four disks can serve a read request
- Write BW
  - Method1: (1)read out all unmodified (N-2) data bits. (2) re-compute parity bit. (3) write both modified bit and parity bit to disks.

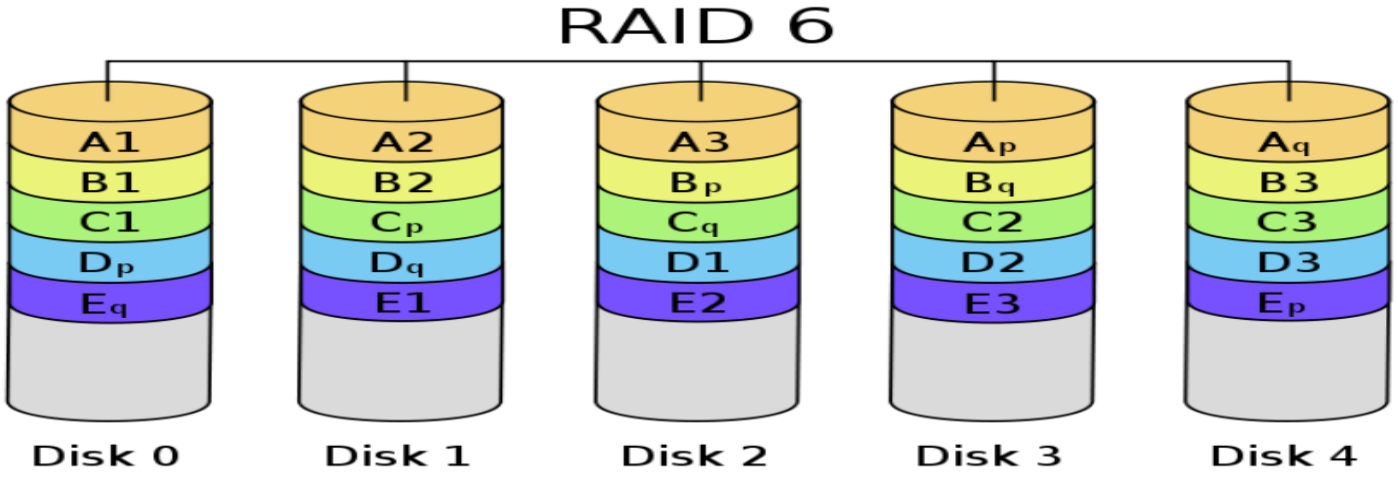
    ➔ write BW = N / ((N-2)+2) = 1 ➔ remains the same
  - Method2: (1)only read the parity bit and modified bit. (2) re-compute parity bit by the difference. (3) write both modified bit and parity bit.

    ➔ write BW = N / (2+2) = N/4 times faster



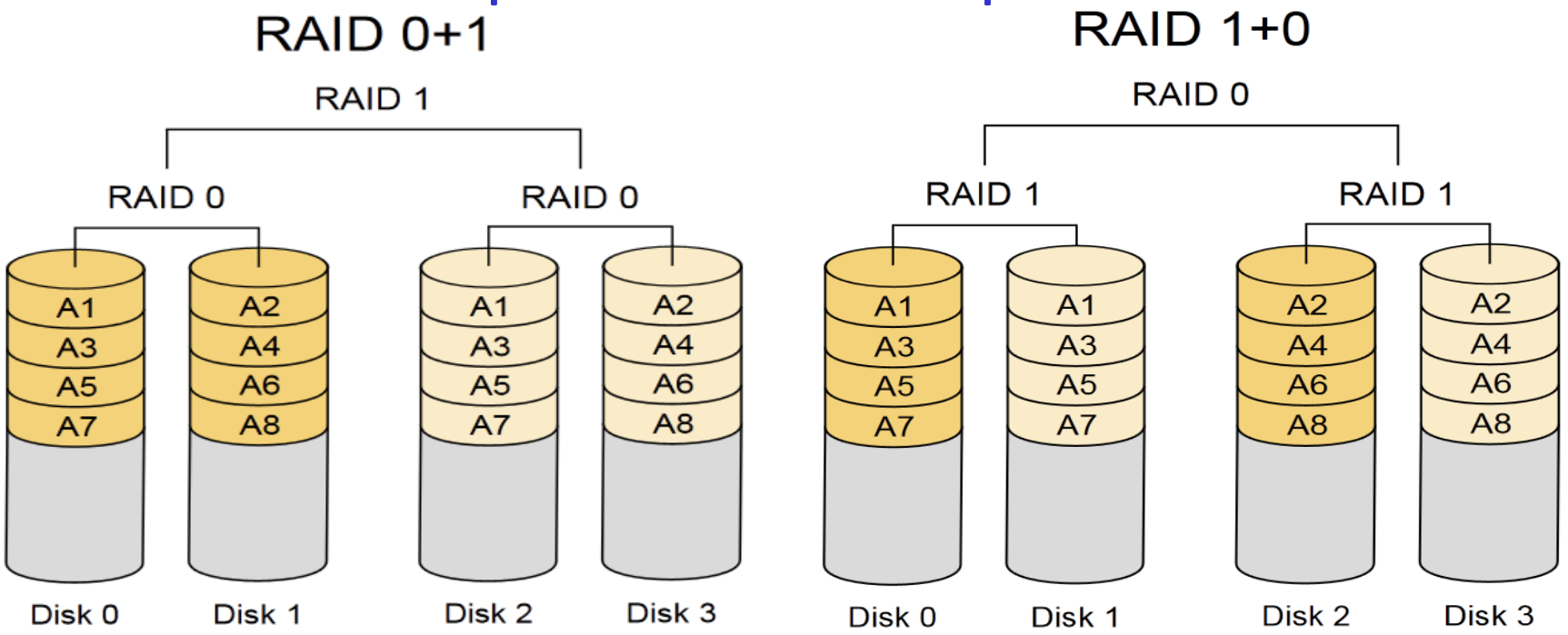
RAID 6: P+Q Dual Parity Redundancy

- Like RAID 5, but stores extra redundant information to guard against **multiple disk failure**
- Use ECE code (i.e. Error Correction Code) instead of single parity bit
- Parity bits are also striped across disks



Hybrid RAID

- RAID 0+1: Stripe then replicate
- RAID 1+0: Replicate then stripe



*First level often control by a controller. Therefore, RAID 10 has better fault tolerance than RAID 01 when multiple disk fails
http://www.thegeekstuff.com/2011/10/raid10-vs-raid01/