ECE437/CS481

M06F: RAID

CHAPTER 10.7

Xiang Sun

The University of New Mexico

MULTIPLE DISKS: RAID

❑ Motivation

- Moore's law: CPU speed doubles every 18 month
- SRAM speed increases by 40-100% a year
- However, disk seek time only improves 5-10% a year

❑ Solution: Redundant Array of Independent Disks (RAID)

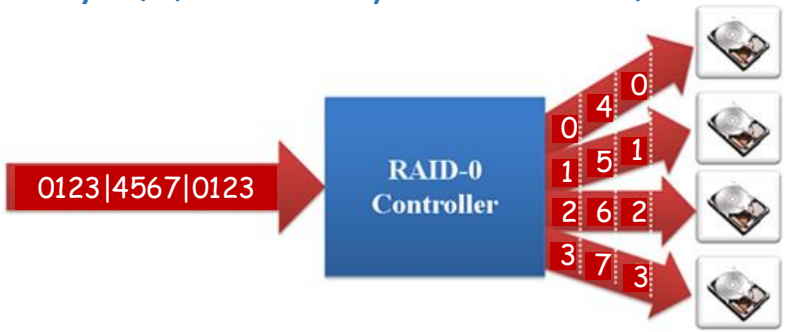
- Originally, known as Redundant Array of Inexpensive Disks (RAID), to combine multiple, small inexpensive disks drive into an array of disk drives which yields performance exceeding that of a Single, Large Expensive Drive(SLED).
- The array of drives appear to the computer as a single logical storage unit or drive.

MULTIPLE DISKS: RAID

❑ Parallel disk systems

- How RAID improves performance as compared to a single drive?

- Solution—parallel disk systems, which is achieved by data stripping
  - ✓ Fundamental to RAID
  - ✓ A method of concatenating multiple drives into one logical storage unit
  - ✓ Splitting the bits of each byte across multiple disks: bit – level striping
    For example, an array of four disks, write bit i of each byte to disk i%4



  - ✓ The data transferring rate of the RAID is four times of a single drive
  - ✓ Similarly for splitting the blocks of a file across multiple disks: block-level striping

MULTIPLE DISKS: RAID

❑ Reliability via redundancy

- As the number of disks per component increases, the probability of failure also increases .
  - ✓ Suppose the mean time to failures (MTTF) of a single disk is 100,000 hrs.
  - ✓ What is the MTTF if there are N disks in a RAID system?

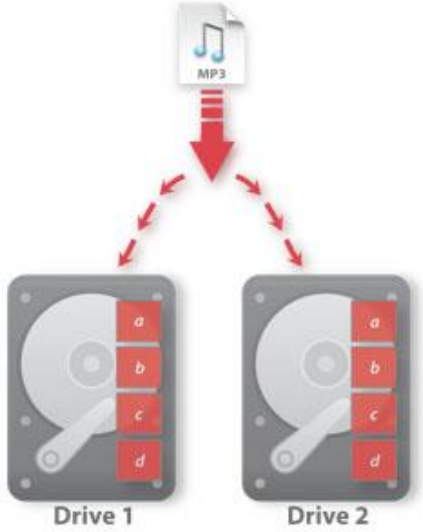
$$\text{MTTF}=\frac{1}{1-\left(1-\frac{1}{100{,}000}\right)^{N}}$$

  - ✓ If N=100, then MTTF≈ 1000.5 hrs =41.6875 days !! ---Not enough
  - ✓ If N=2, then MTTF≈ 50,000.25 hrs ≈2084 days ≈ 5.7 years

- Solution?
  - ✓ Redundancy!

MULTIPLE DISKS: RAID

- Redundancy achieved by mirroring
  - Duplicate every disk.
  - Logical disk consists of two physical disks
  - Every write is carried out on both disks
  - If one of the disk fails, data read from the other
  - Data permanently lost only if the second disk fails before the first failed disk is replaced

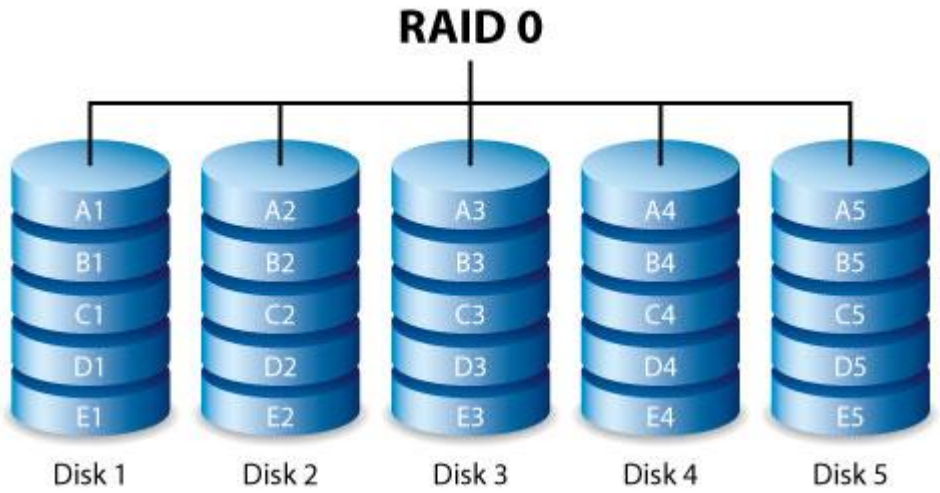


- The probability of failure for a mirrored disk system significantly reduces.
  - Suppose the MTTF of a single disk is 100,000 hrs, i.e., the probability of failure for a disk=$\frac{1}{100,000}$ per hr.
  - Assume that the mean time to replace a failure disk in a mirrored disk system is 10 hrs.
  - Thus, the probability of failure for a mirrored disk system p= $2 \times \left(\frac{1}{100,000}\right)^2 \times 10$
  - Thus, MTTF of the mirroring RAID system=$\frac{1}{p} = 57{,}000\ years$
- Main disadvantage: Most expensive approach.

MULTIPLE DISKS: RAID

- Design of RAID-six levels representing six alternative designs

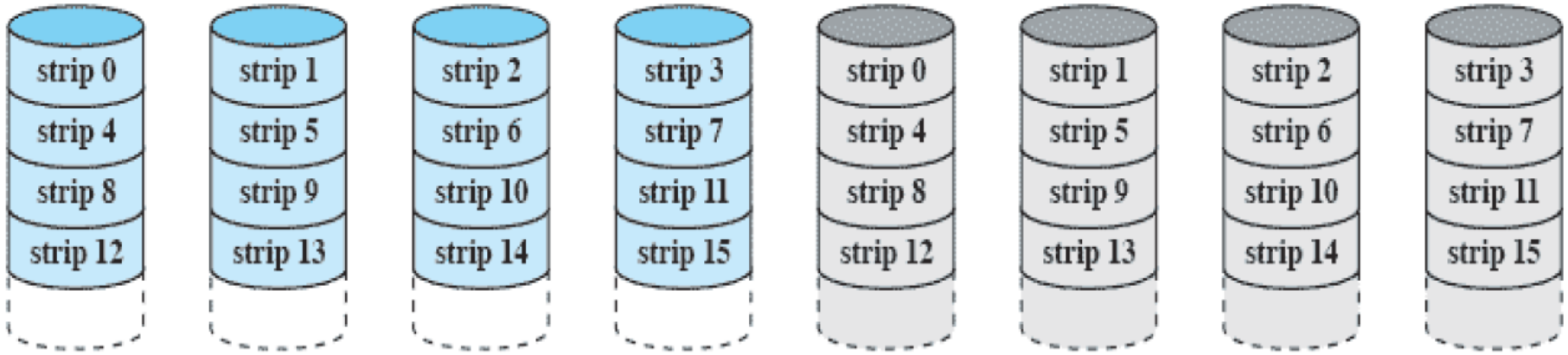
  - RAID Level 0: Non-redundant Disk Striping
    - ✓ Multiple disks are connected to a single disk controller.
    - ✓ Data is striped to spread segments across multiple drives
    - ✓ Improvement of I/O performance, not data redundancy
      - provide a high transfer rate, by overlapping disk read and write
      - possible interspersed read/write
      - if one of drives fails, the whole system may fail.



MULTIPLE DISKS: RAID

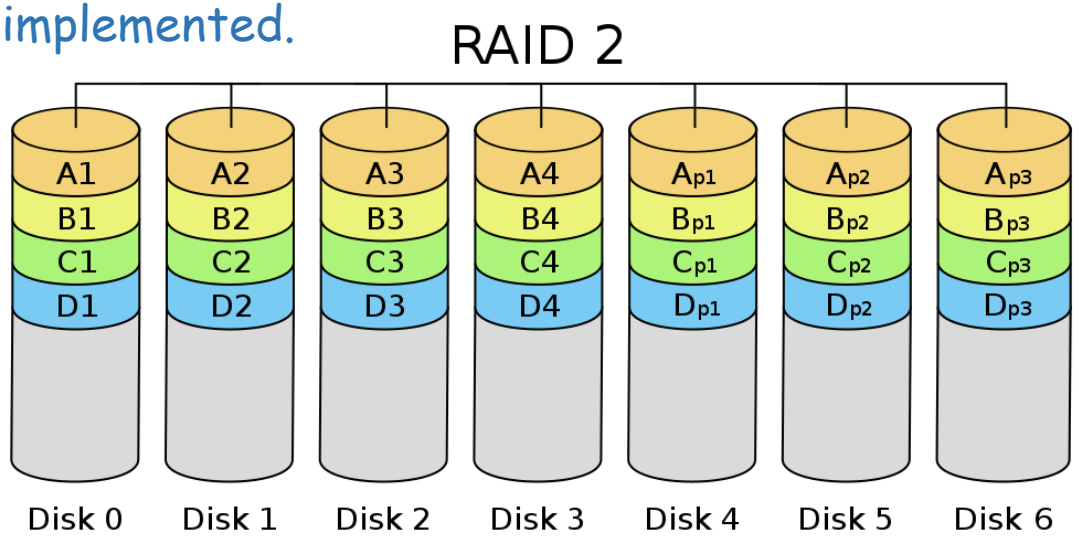
- Design of RAID-six levels representing six alternative designs

  - RAID Level 1: Disk Mirroring (or shadowing)
    - ✓ RAID 0 plus duplicated data stored in the identical drives, i.e., every main drive has a minor drive, backup/mirror.
    - ✓ Improvement of not only performance but also data redundancy.
      - expensive way to achieve data redundancy.
      - good reliability, if a single drive fails, its mirror drive takes over.



MULTIPLE DISKS: RAID

- Design of RAID-six levels representing six alternative designs
  - RAID Level 2: Bit Interleaving of Data with HECC (Hamming Error Correcting Code)
    - ✓ Requires fewer disks than Level-1 to provide redundancy, but still needs quite a few more disks
      - $m+n<=2^n$
      - m=4 data disks need n=3 check disks.
    - ✓ Write request requires all the disks.
    - ✓ Given high reliability of disks with the expense of complicated coding algorithm.
      - RAID Level 2 is an overkill and is never implemented.

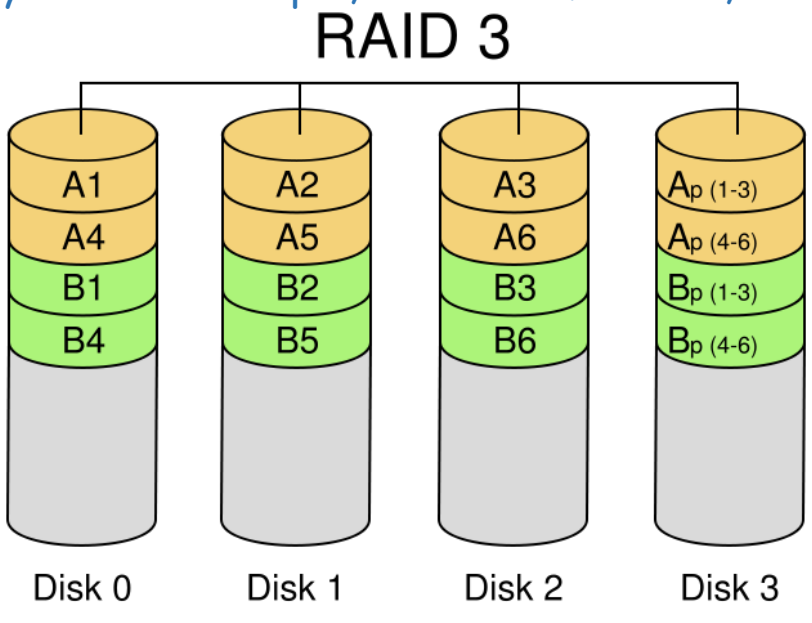


MULTIPLE DISKS: RAID

❑ Design of RAID-six levels representing six alternative designs

➢ RAID Level 3: Bit Interleaving of Data with Parity (Parallel Disk Array)

- ✓ One parity disk is used to do on-the-fly parity generation and parity checking, capable of correcting any single and self-identifying failure.
  - Suppose we have 3 data disks and one parity disk. The sample bits in the data disks are: 0,1,1.
  - The parity bit is the XOR of these three data bits, which can be calculated by adding them up and writing a 0 if the sum is even and a 1 if it is odd. Here, the sum of Disk 0 through Disk 2 is "2", so the parity is 0.
  - Now if we attempt to read back this data, and find that Disk 2 gives a read error, we can reconstruct Disk 2 by conducting XOR of all the other disks, including the parity. In the example, the sum of Disk 0, 1 and Parity is "1", so the data on Disk 2 must be 1.
- ✓ RAID Level 3 requires a minimum of 3 drives to implement.
  - # of redundant disks is 1 (constant, no matter how many drives).
  - works only if it is known which disk fails.
  - to access a single file block of data, must access all the disks. This allows good parallelism for a single file access, but doesn't allow multiple I/Os (i.e., multiple file access at a time).



MULTIPLE DISKS: RAID

❑ Design of RAID-six levels representing six alternative designs

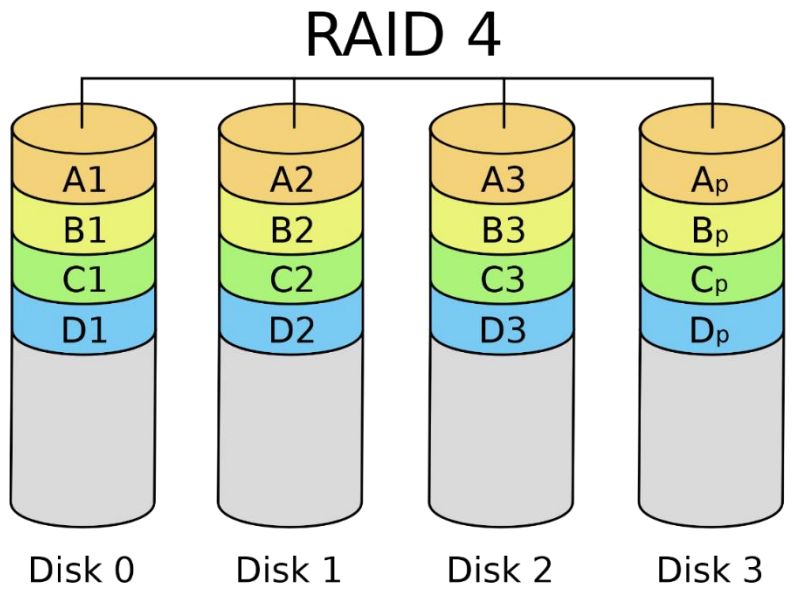
➢ RAID Level 4: Block Interleaving of Data with Parity (Parallel Disk Array)

- ✓ Similar to RAID Level 3, except that data is interleaved across disks of striping unit (striping width) rather than in bits.
- ✓ Still use a single disk for parity; however, the parity is calculated over data from multiple striping units.

  - If an error detected, we may have to read other striping units on other disks to correct data.
  - What if we modify a data bit in a striping unit? Do we need to read striping units from other disks and recalculate the parity data?—No, can use the following formula.

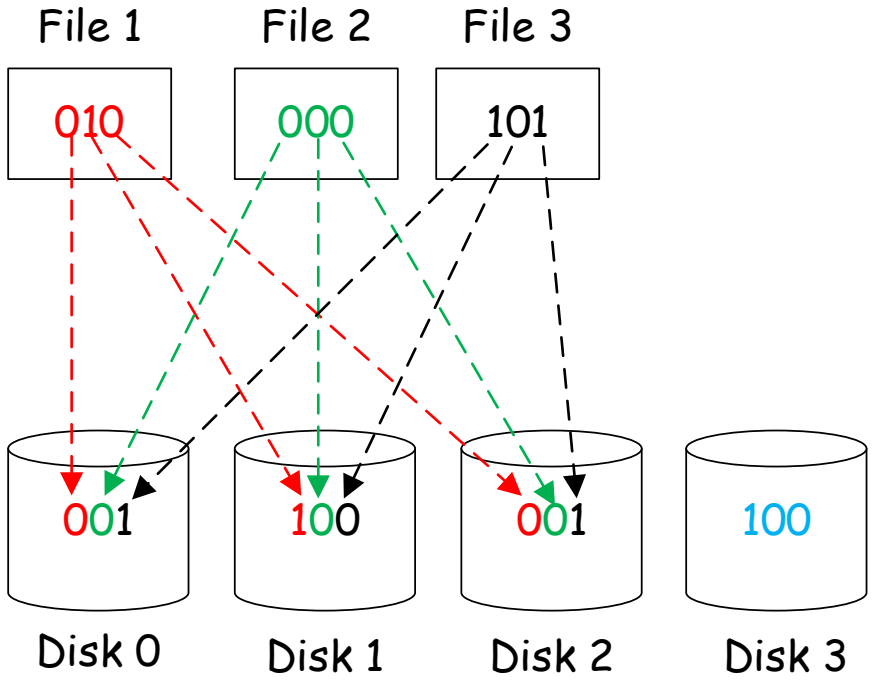
    new parity = (old data xor new data) xor old parity

    which requires 2 reads and 2 writes.
  - Allow multiple file reads at a time but may not allow parallelism for a single file access.
  - Only one write is allowed at a time.

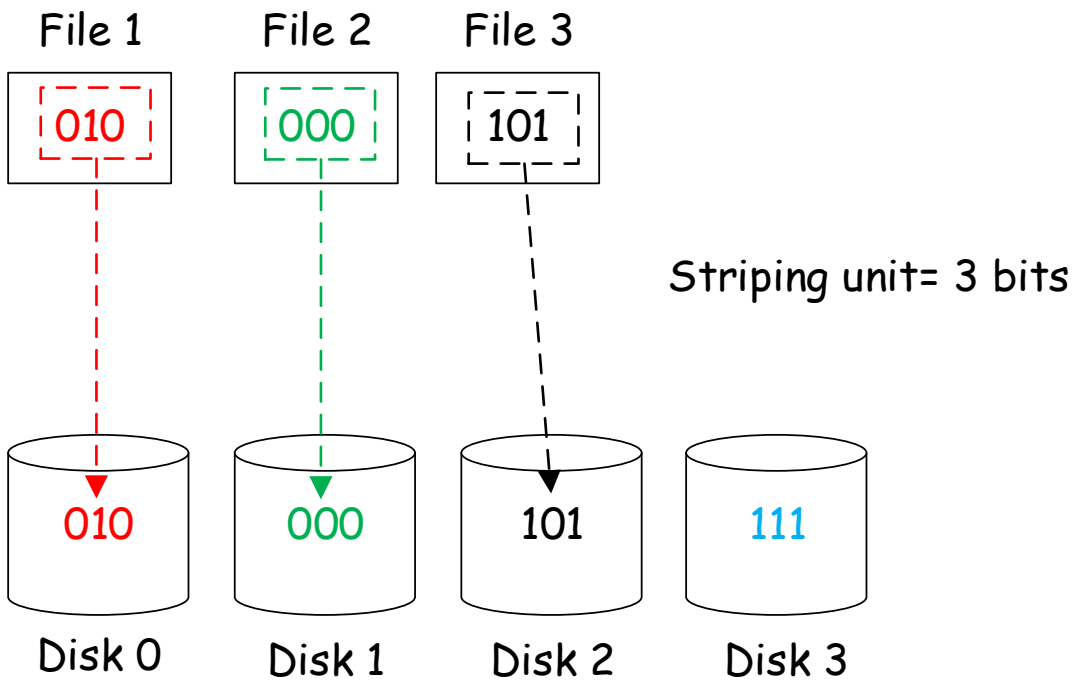


MULTIPLE DISKS: RAID

❑ Comparison between RAID Level-3 and Level-4



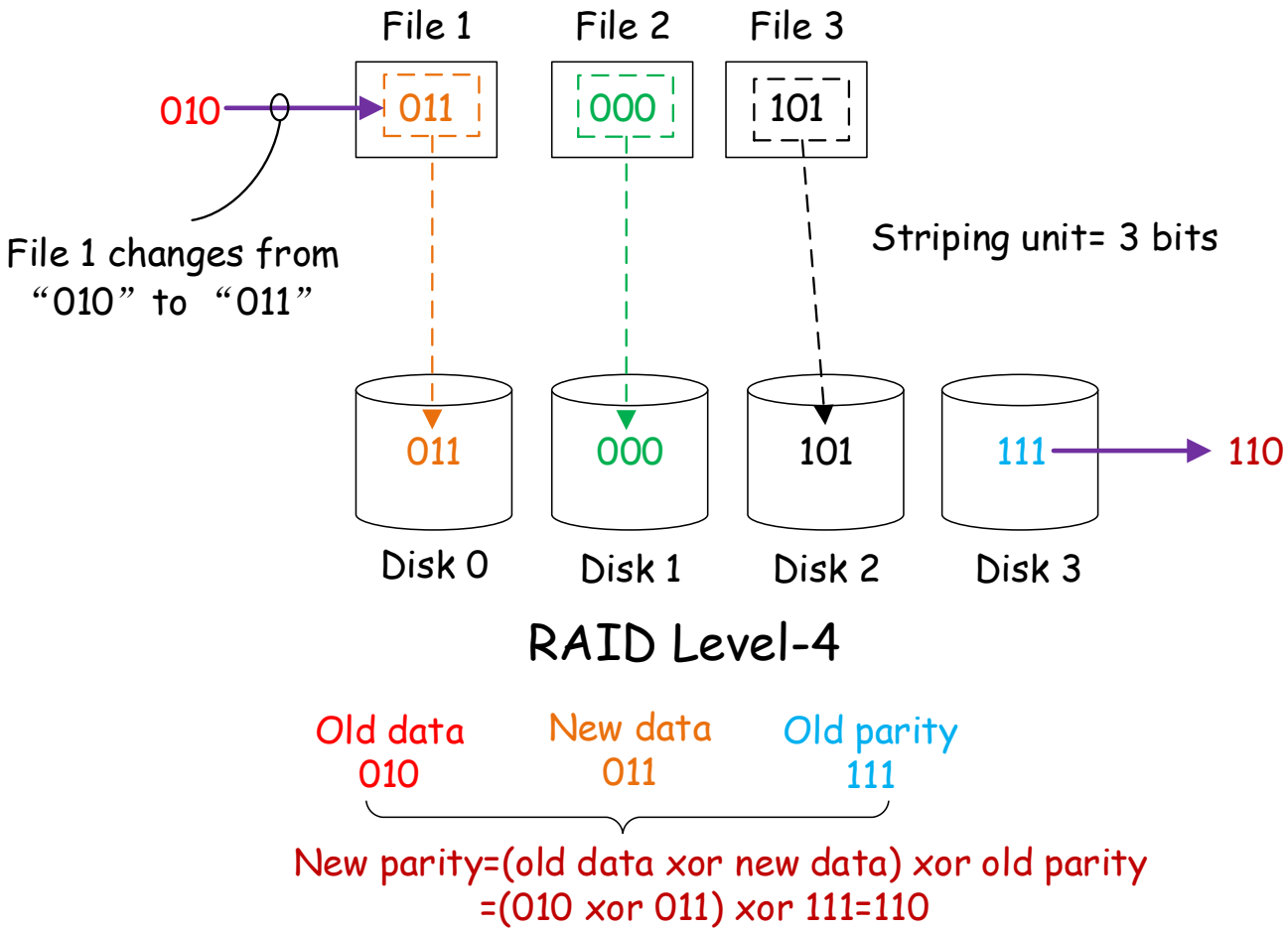
RAID Level-3



RAID Level-4

MULTIPLE DISKS: RAID

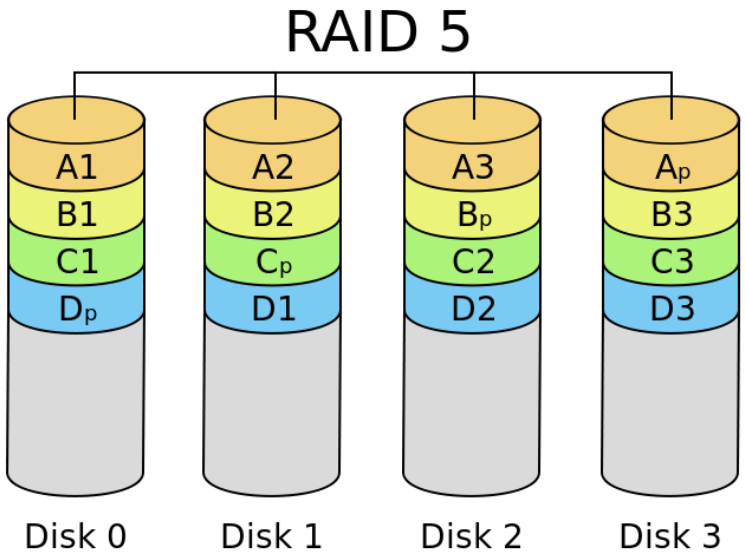
❑ Write on a striping unit in RAID Level-4



MULTIPLE DISKS: RAID

- Design of RAID-six levels representing six alternative designs

  - RAID Level 5: Block Interleaving with Distributed Parity
    - ✓ Level 5 is to solve the problem of one write being allowed at a time (bottleneck in parity disk) in Level 4.
      - e.g., writing a file on Disk 0 and writing on Disk 1 both require a write to the parity disk
    - ✓ In Level 5, there is no dedicated parity drive in Level 5. The parity bits are across all disks based on a round-robin manner.
      - e.g., writing on A1 and writing on B2 can be proceeded simultaneously.



MULTIPLE DISKS: RAID

❑ RAID Level 10: Combine Level 0 and Level 1

- ✓ Using four disks as an example, RAID 10 creates two RAID 1 segments, and then combines them into a RAID 0 stripe.
- ✓ How about eight disks?