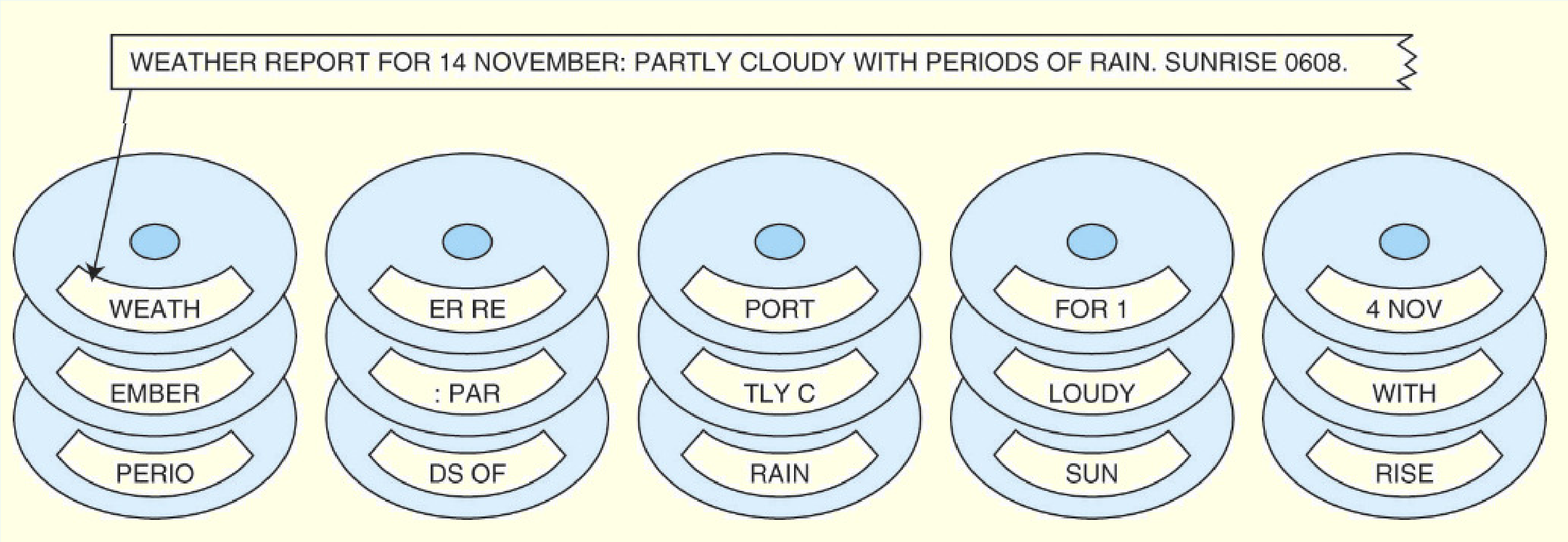
RAID

- RAID, an acronym for *Redundant Array of Independent Disks* was invented to address problems of disk reliability, cost, and performance.
- In RAID, data is stored across many disks, with extra disks added to the array to provide error correction (redundancy).
- The inventors of RAID, David Patterson, Garth Gibson, and Randy Katz, provided a RAID taxonomy that has persisted for a quarter of a century, despite many efforts to redefine it.

RAID 0: Striped Disk Array

- RAID Level 0 is also known as *drive spanning*
  - Data is written in blocks across the entire array



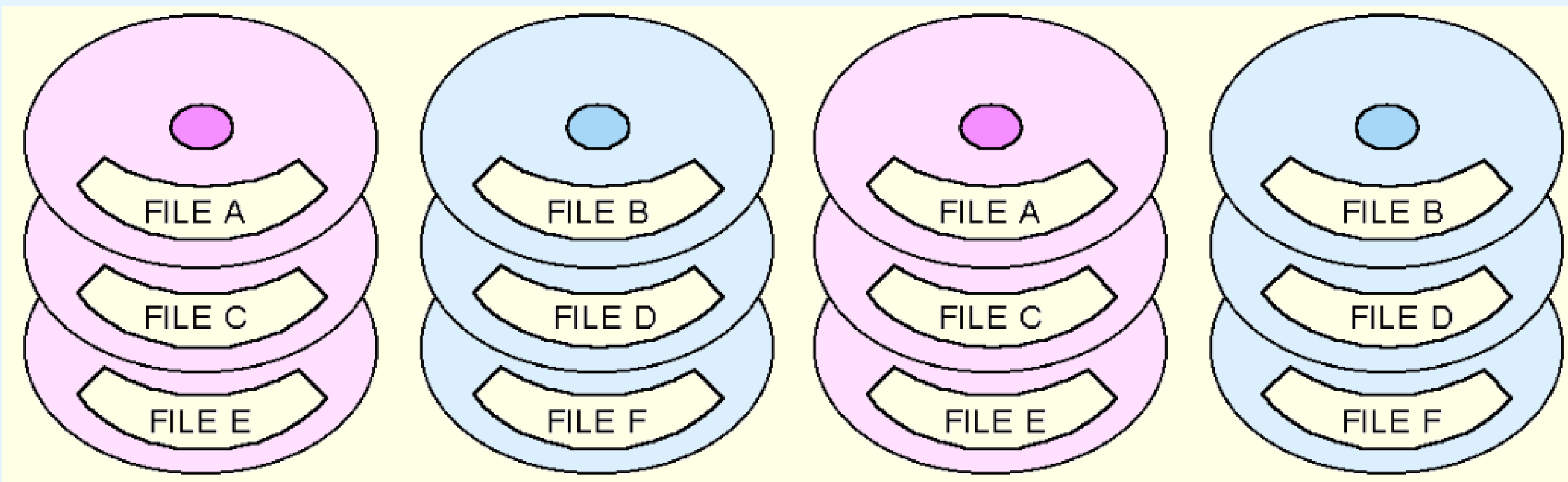
•

RAID 0

- Recommended Uses:
  - Video/image production/edition
  - Any app requiring high bandwidth
  - Good for non-critical storage of data that needs to be accessed at high speed
- Good performance on reads and writes
- Simple design, easy to implement
- No fault tolerance (no redundancy)
- Not reliable

RAID 1: Mirroring

- RAID Level 1, also known as *disk mirroring*, provides 100% redundancy, and good performance.
  - Two matched sets of disks contain the same data.

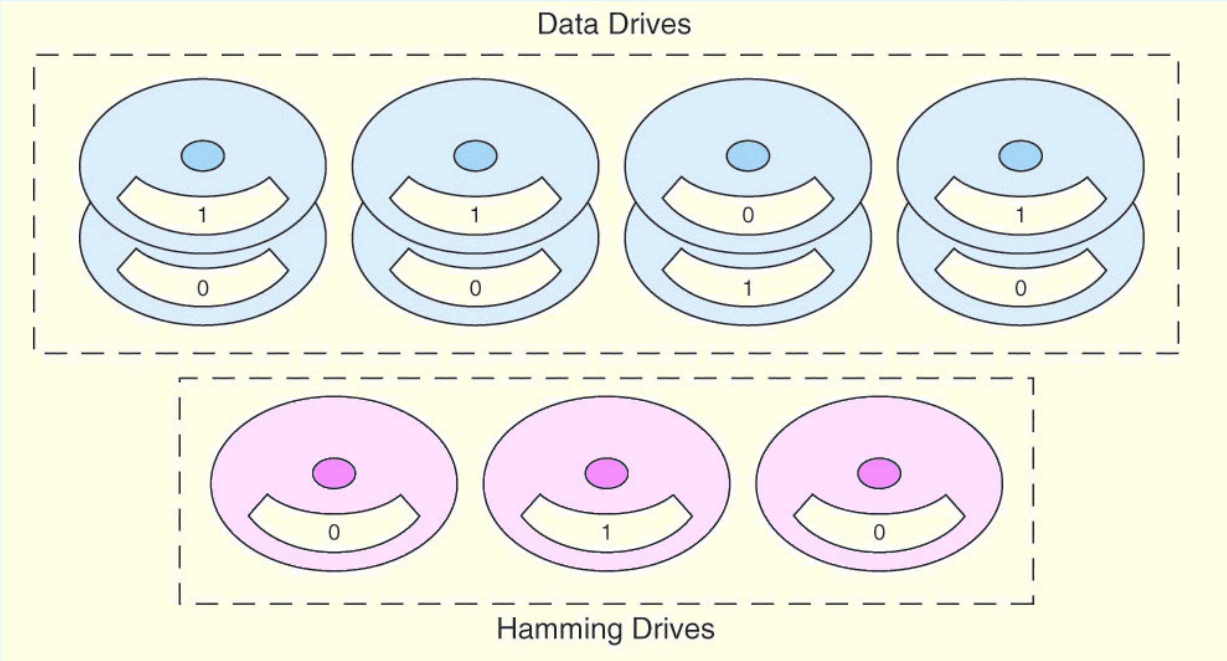


RAID 1

- Recommended Uses:
  - Accounting, payroll, financial
  - Any app requiring high reliability (mission critical storage)
- For best performance, controller should be able to do concurrent reads/writes per mirrored pair
- Very simple technology
- Storage capacity cut in half
- S/W solutions often do not allow “hot swap”
- High disk overhead, high cost

RAID 2: Bit-level Hamming Code ECC Parity

- A RAID Level 2 configuration consists of a set of data drives, and a set of Hamming code drives.
  - Hamming code drives provide error correction for the data drives.

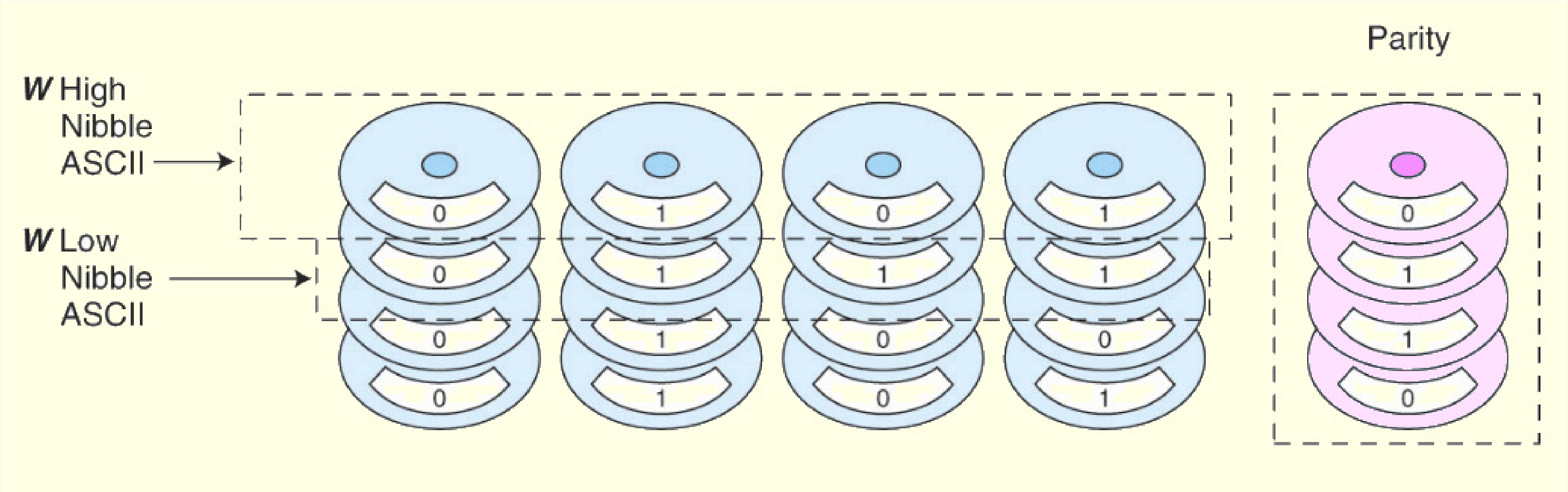


RAID 2

- Provides on the fly error correction
- Can have very high data xfer rates
- Somewhat complex
- High ratio of ECC disks to data disks
- Startup costs are high
- Not commercially viable

RAID 3: Bit-level Parity

- RAID Level 3 stripes bits across a set of data drives and provides a separate disk for parity.
  - Parity is the XOR of the data bits.

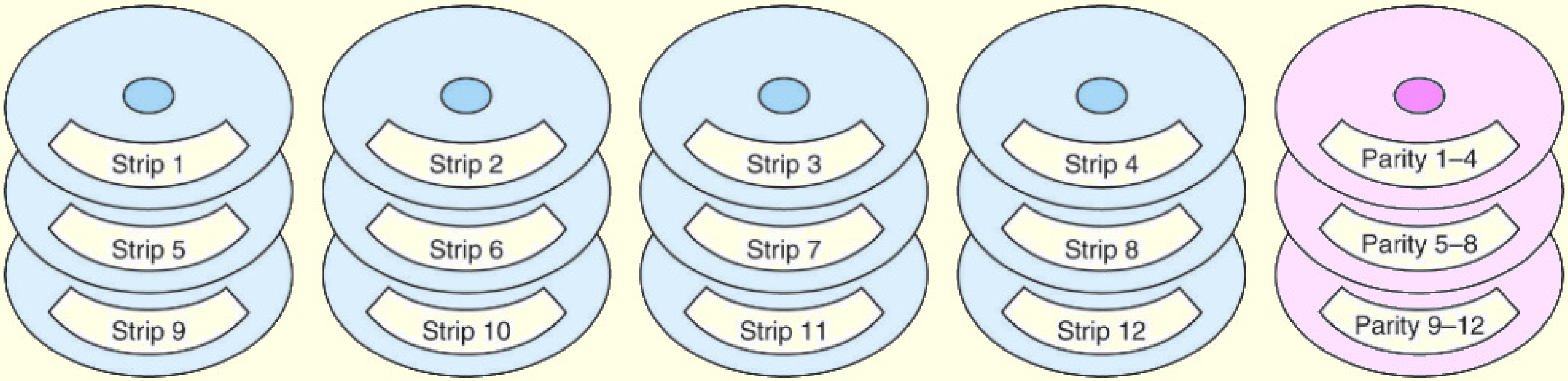


RAID 3

- Good for video production, image editing, any app needing high throughput, but not very common
- Very high data xfer rates, both read and write, for large data xfers
- Disk failure has little impact
- Low ratio of parity to data
- Resource intensive
- Complex controller (must be implemented in h/w)
- Performance is slower for random, small I/O operations

RAID 4: Block-level Parity

- RAID Level 4 is like adding parity disks to RAID 0.
  - Data is written in blocks across the data disks, and a parity block is written to the redundant drive.



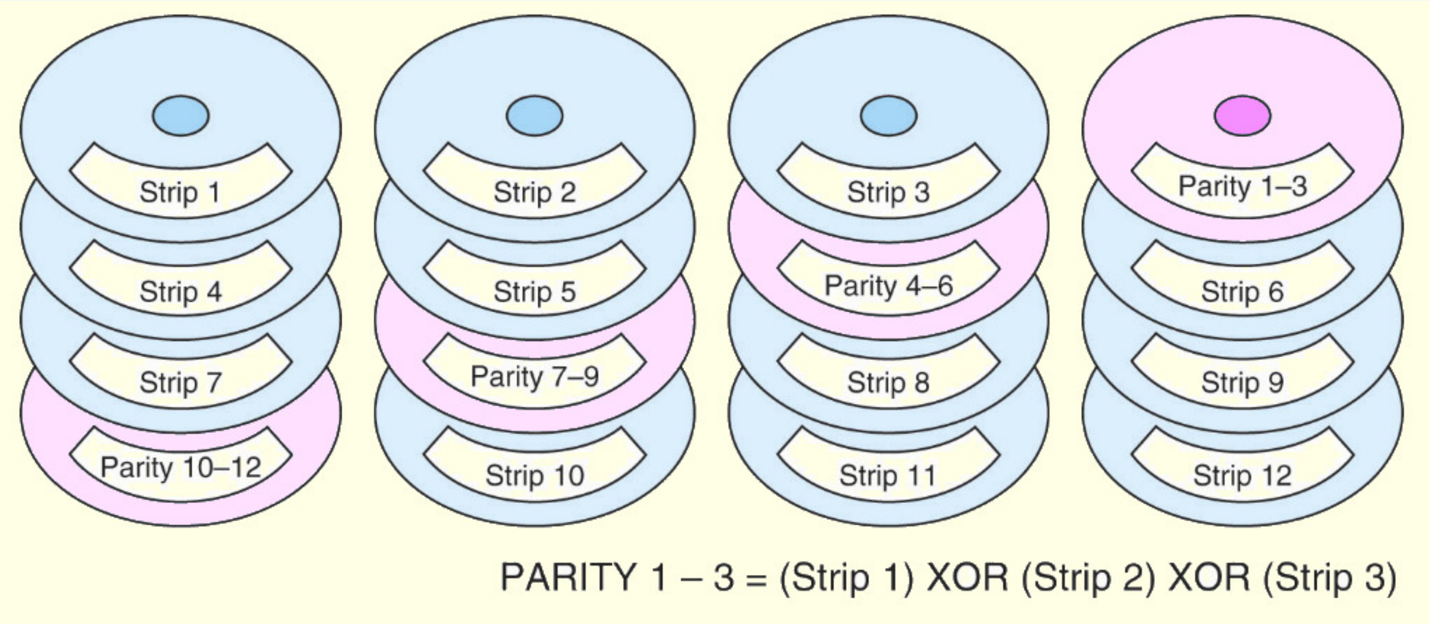
PARITY 1 – 4 = (Strip 1) XOR (Strip 2) XOR (Strip 3) XOR (Strip 4)

RAID 4

- Very high read rate
- Low ratio of parity to data disks
- Complex controller
- Very poor write rate
- If fails, difficult to rebuild data
- Not very common

RAID 5: Block-level Distributed Parity

- RAID Level 5 is RAID 4 with distributed parity.
  – With distributed parity, some accesses can be serviced concurrently, giving good performance and high reliability.



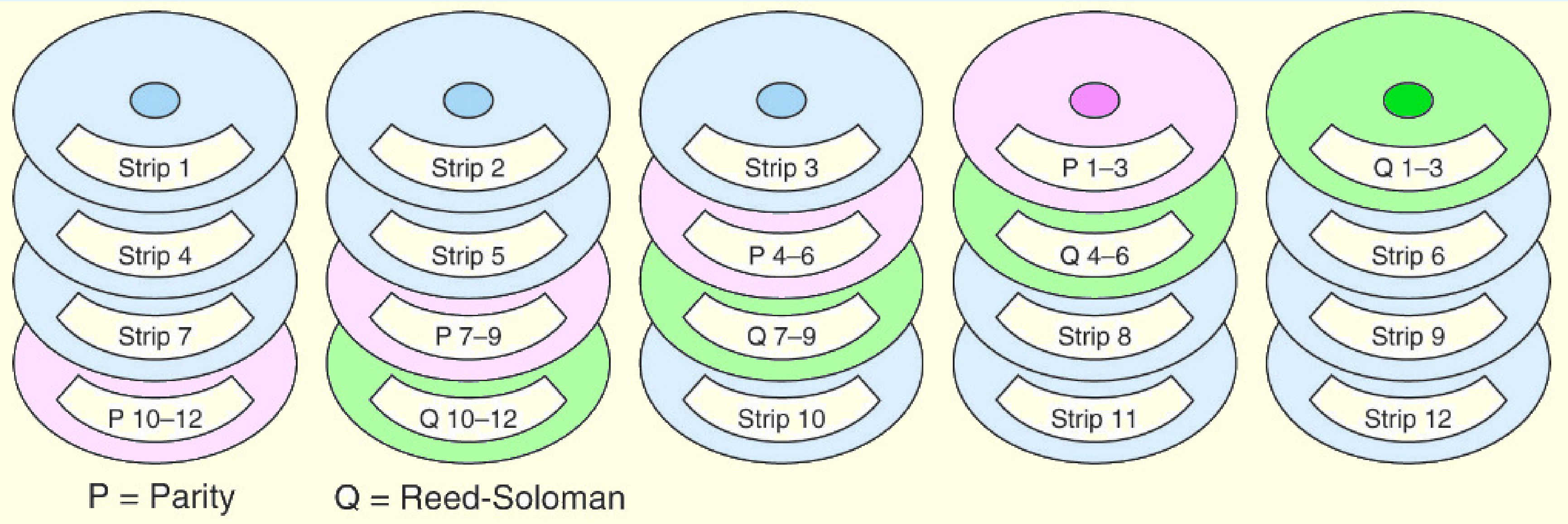
  – RAID 5 is used in many commercial systems and quickly replaced 3 and 4.

RAID 5

- Good for file and application servers (database, WWW, email, news)
- Most versatile RAID level, most common
- Good balance of performance, cost, reliability
- Reads are very fast; writes are slower due to calculating parity
- Failure has medium impact (on throughput)
- Complex controller design

RAID 6: 2 Distributed Parity Schemes

- RAID Level 6 carries two levels of error protection over striped data: Reed-Soloman and parity.
  - It can tolerate the loss of two disks.



RAID 6

- Extension of Level 5 with additional fault tolerance
- Perfect solution for mission critical apps
- Complex controller, significant overhead
- Write intensive

Combined (nested) RAID Levels

- Keep in mind that a higher RAID level does not necessarily mean a “better” RAID level. It all depends upon the needs of the applications that use the disks.
- Large systems consisting of many drive arrays may employ various RAID levels, depending on the criticality of the data on the drives.
  - A disk array that provides program workspace (say for file sorting) does not require high fault tolerance.
- Critical, high-throughput files can benefit from combining RAID levels

RAID 0+1

- Also called RAID 01
- Mirrored configuration of two striped sets (striping done first, then mirrored)
- Would divide 6 disks into 2 sets of 3
  - Stripe drives 1, 2, and 3 into RAID set
  - Mirror the set using disks 4, 5, and 6
  - Now have 2 sets of 3 disks

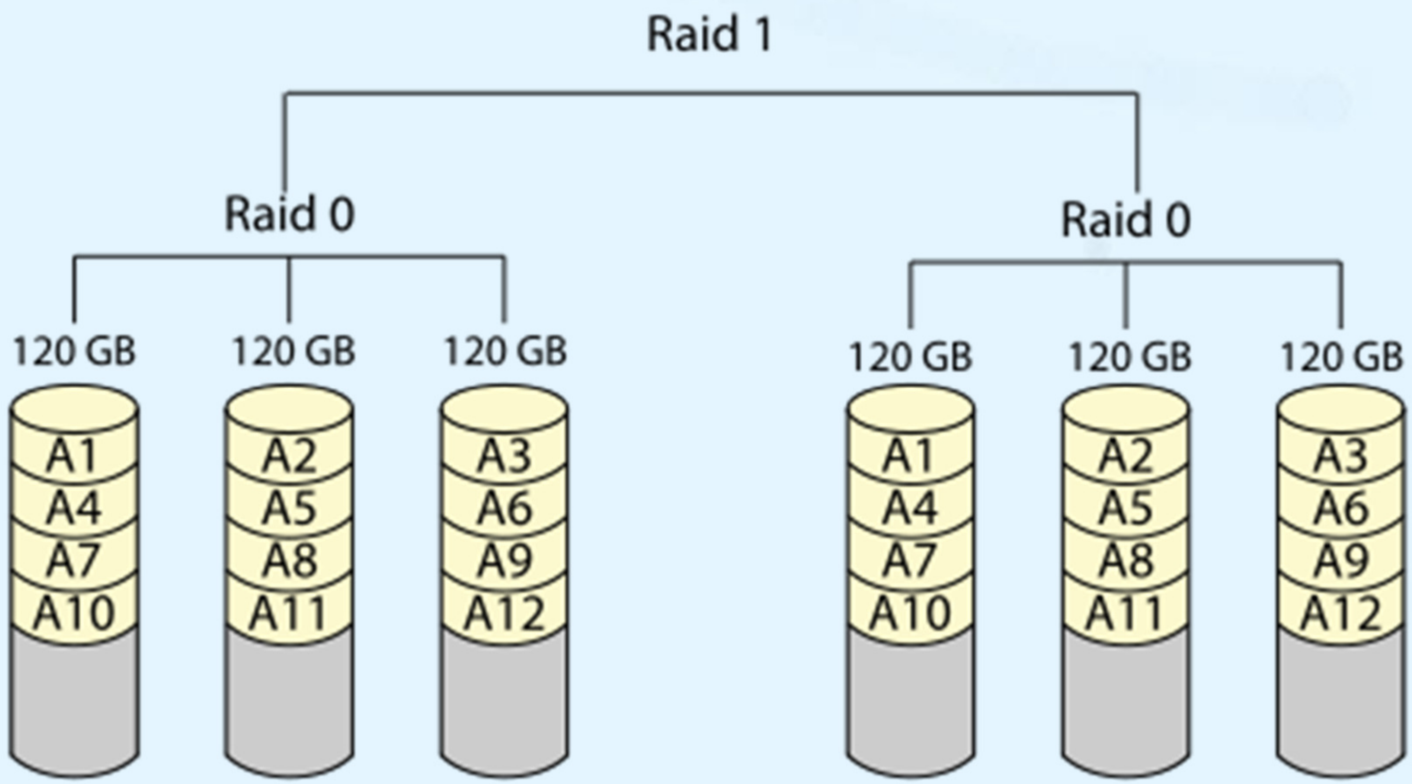


Image from: http://en.wikipedia.org/wiki/Nested_RAID_levels

RAID 1+0

- Also called RAID 10
- Stripe across a number of mirrored sets (mirroring done first, then striped)
- Would divide 6 disks into 3 sets of 2
  - Each set is a RAID 1 array
  - Have 3 sets each with 2 disks

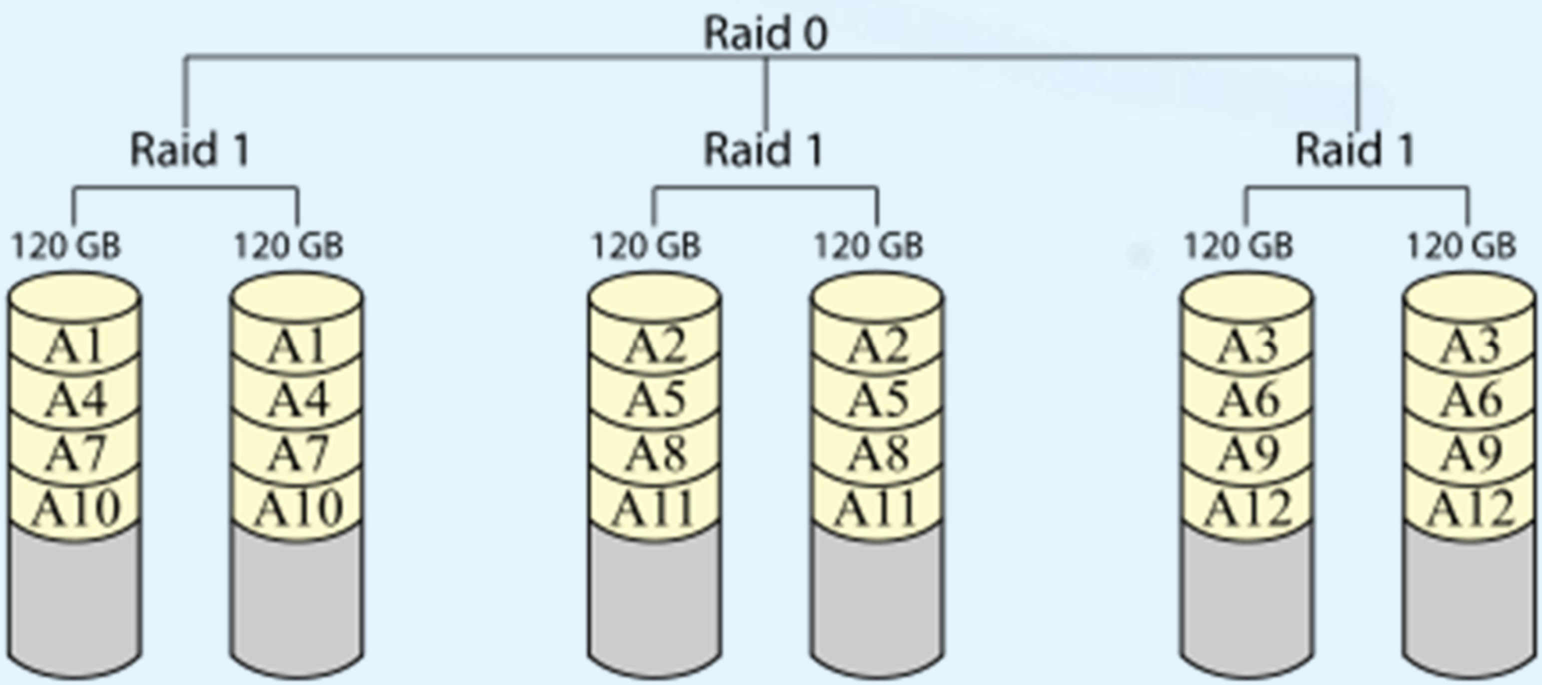


Image from: http://en.wikipedia.org/wiki/Nested_RAID_levels

RAID 5+0

- Combines block-level striping of RAID 0 with distributed parity of RAID 5
- Below: 3 collections of RAID 5s are striped together

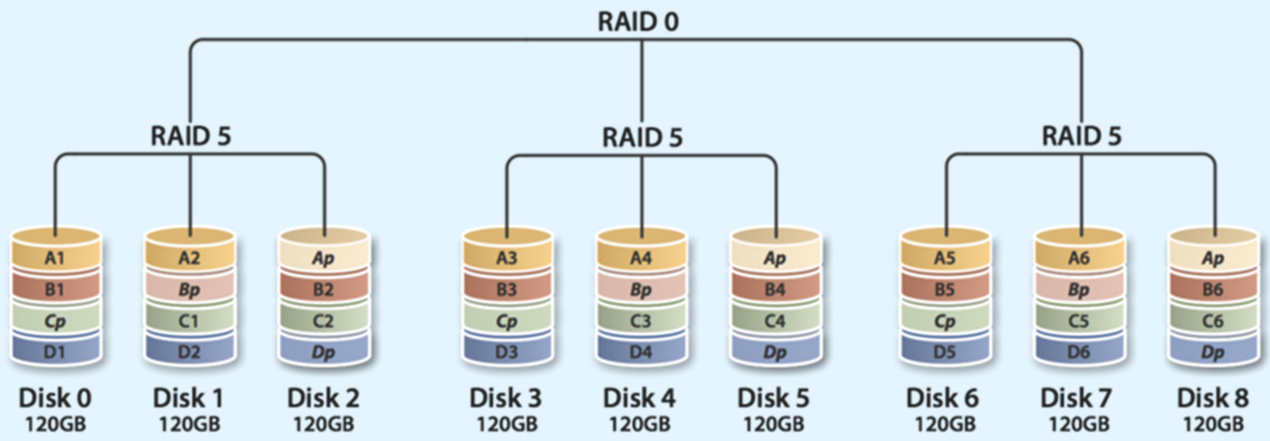


Image from: http://en.wikipedia.org/wiki/Nested_RAID_levels

RAID 5+1

- Two RAID 5's that are mirrored

Image from: http://en.wikipedia.org/wiki/Nested_RAID_levels

Other Common Nested Levels

- RAID 0+3
- RAID 30
- RAID 100 (1+0+0 or 10+0) is a stripe of RAID 10s
- RAID 51
- RAID 0+5
- RAID 53
- RAID 60
- RAID DP (double parity RAID 6)
- RAID 1.5

End of Chapter 7