RAID Level :

RAID level ○ uses a striped disk array with no fault tolerance and hence has no redundancy. The disk contains data only & there is no ECC data.

In level ○, multiple reads & writes are possible.

The striping level (size) is a block.

Level ○ is sometimes not considered as RAID as it has no redundancy (No ECC).

2 RAID Level1 (Mirroring)

This level employs disk mirroring (shadowing) to provide redundancy, so each disk in the array is duplicated. Stripes are not implemented in level1 & hence multiple access for the same file is only possible on reading but not on writing. On writing, the same data has to be written on both disks (the original & the mirror) but on reading, 2 different parts of the same file can be read at the same time from the original & mirror disks. The mirror technology enhances reliability & restricted multiple access but doubling the cost.

3 RAID Level2

RAID level2 arrays are striped at the bit level, so each strip stores one bit. This means that adjacent bits of file are stored on different disks. Level2 arrays are not mirrored, which reduces the storage overhead incurred by Level1.

The fault tolerance is achieved here by using hamming error correcting codes (hamming ECCs). The error code bits are stored on separate disks (parity disks).

Of course, each stripe of one bit size on data disks has a stripe of one bit size on parity disks. This means that each group of data bits has a corresponding group of ECC bits.

The clear problem here is that if the OS wants to write few bits of the group (stripe), it has to read all data stripes first & then modify the necessary data bits & then calculate the ECC bits & at last store the new data & ECC bits. This is called "read-modify-write" cycle.

From the above, we notice that the storage overhead is decreased compared to mirror system but the multiple access for several files is not possible as all disks will be occupied for one file request (remember that the adjacent file bits are distributed among the disks, also, it is necessary to read the parity disks).

✓ ***Note:*** In hamming, we can correct one data bit error. The number of ECC bits are as follows:

Number of data bits	**number of ECC bits**
3	
11	4
26	5
27	

It is clear that the larger data stripe, the better & hence the more data disks in the array (stripe) are the better.

4 RAID Level3

RAID level3 stripes data at the bit or byte level but use parity checks for fault tolerance instead of Hamming. In parity check, we use only one bit (even or odd parity) & this bit does not locate the place of error (as incase of Hamming) but indicates only its existence. When the error occurs, the OS will inform the user immediately who has to find out the erroneous disk and replace it. The data on the faulty disk can be regenerated automatically with the help of other data disks & the parity disk. The advantages of such system:

1- Large storage.
2- Fault tolerance with one extra disk (one parity disk).
3- Multiple access for one file is possible and hence fast access time.

Of course, multiple access for several files is not possible as any file request will occupy all of the disks.

5 RAID Level4

RAID Level4 systems are striped using fixed size blocks (typically much larger than a byte) & use one disk for parity (even or odd parity). The difference with level3 is that the file may occupy fraction of disks & not all of them (remember the coarse grained stripes) & hence multiple requests for multiple files may be possible at the same time. Here, we should remember that when reading data from disks, it is not always necessary to read parity bits as these bits are stored not for error detection but mainly for error correction (of one bit –usually one disk may be faulty-).

✓ ***Note:*** Multiple writes is not possible because the parity disk will be occupied for one write.

✓

6 RAID Level5

RAID Level5 arrays are striped at the block level & a parity check (even or odd) is used like in level4. The difference with level4 is that the parity bits are not located on one disk but distributed throughout the arrays of disks. This means that disk1 carries parity bit1, disk2 carries parity bit2, & so on. The advantage of this level is that multiple writes (writes for several files) are possible because the parity bits are not stored on one disk as the case of level4.

7 Other RAID Levels

There are other levels such as:

- RAID Level6
- RAID Level10+1
- RAID Level10
- RAID Level10+3, 0+5, 50, 1+5, etc.

8 comparison of RAID Levels

The properties of different RAID levels can be summarized as follows:

From multiple files

RAID Level	**Read concurrency**	**Write concurrency**	**Redundancy**	**Striping Level**
0	Yes	Yes	None	Block
1	Yes	No	Mirroring	None
2	No	No	Hamming ECC	Bit
3	No	No	Even or odd parity	Bit/Byte
4	Yes	No	Even or odd parity	Block
5	Yes	Yes	Distributed even or odd parity	Block

✓ ***Notes:***

1- When striping level is bit or byte. This means that the file data are distributed on all the disk arrays & hence it is not possible to service multiple requests for several files, however, the single file will be read/write very quickly i.e. fast access (fast data transmission).

When block level is used then the file may occupy few disks of the array (stripe) & the others may be for other file, & hence multiple files may be serviced.

2- The main purpose of redundancy is not to detect errors but to correct it & hence, it is not always necessary to read parity disks during read cycle & this allows multiple read for several files when block (coarse grained striping) striping is used.

3- Hamming ECC is not necessary as the faulty disk may be discovered by other means & hence one parity (even or odd) bit is enough for error detection & correction. In other words, the parity bit will inform us about the existence of error and then by other means (electrical, mechanical) we can find out the faulty disk & hence regenerates its data from knowing the other data bits on the data disks & the parity bit from the parity disk.

✓ ***Note:*** in RAID, we discover that there are errors by using parity check. Then by some means we find the faulty disk & then we regenerate the data on the faulty disk by making parity of all other data disks (except the faulty one) & of the parity disk itself.

Conclusions;

The parity check can be used to correct data if the erroneous bit location is known.

Once we know this location, we can find the missing bit by making parity of all other data bits & the parity bit.

In RAID, we know location by knowing the faulty disk by different means & this starts by finding parity check error.

From the above we notice that RAID systems features are:

- Large storage volume as it uses arrays of disk.
- Fast data transmission as many heads work together (Independent Disks).
- Fault tolerance with low overhead storage by using parity check (redundancy).