RAID

The basic idea of RAID (Redundant Array of Independent Disks) is to combine multiple inexpensive disk drives into an array of disk drives to obtain performance, capacity and reliability that exceeds that of a single large drive. The array of drives appears to the host computer as a single logical drive.

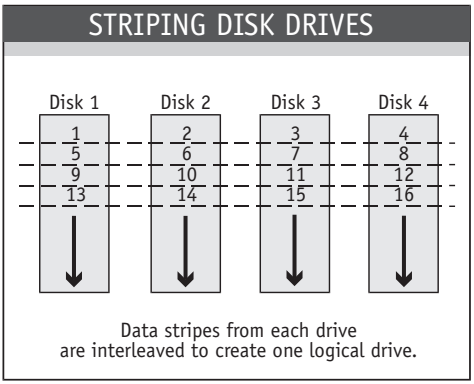
The Mean Time Between Failure (MTBF) of the array is equal to the MTBF of an individual drive, divided by the number of drives in the array. Because of this, the MTBF of a non-redundant array (RAID 0) is too low for mission-critical systems. However, disk arrays can be made fault-tolerant by redundantly storing information in various ways.

Five types of array architectures, RAID 1 through RAID 5, were originally defined, each provides disk fault-tolerance with different compromises in features and performance. In addition to these five redundant array architectures, it has become popular to refer to a non-redundant array of disk drives as a RAID 0 array.

Disk Striping

Fundamental to RAID technology is *striping*. This is a method of combining multiple drives into one logical storage unit. Striping partitions the storage space of each drive into stripes, which can be as small as one sector (512 bytes) or as large as several megabytes. These stripes are then interleaved in a rotating sequence, so that the combined space is composed alternately of stripes from each drive. The specific type of operating environment determines whether large or small stripes should be used.

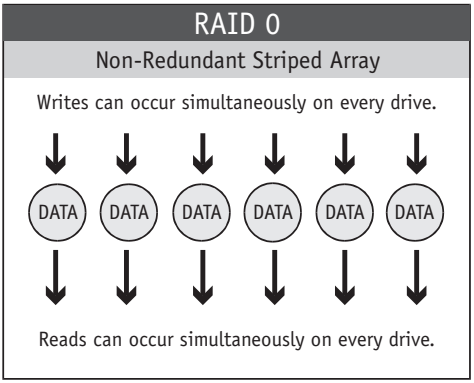
Most operating systems today support concurrent disk I/O operations across multiple drives. However, in order to maximize throughput for the disk subsystem, the I/O load must be balanced across all the drives so that each drive can be kept busy as much as possible. In a multiple drive system without striping, the disk I/O load is never perfectly balanced. Some drives will contain data files that are frequently accessed and some drives will rarely be accessed.



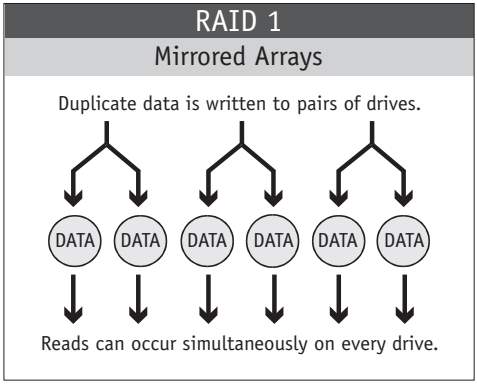
By striping the drives in the array with stripes large enough so that each record falls entirely within one stripe, most records can be evenly distributed across all drives. This keeps all drives in the array busy during heavy load situations. This situation allows all drives to work concurrently on different I/O operations, and thus maximize the number of simultaneous I/O operations that can be performed by the array.

Definition of RAID Levels

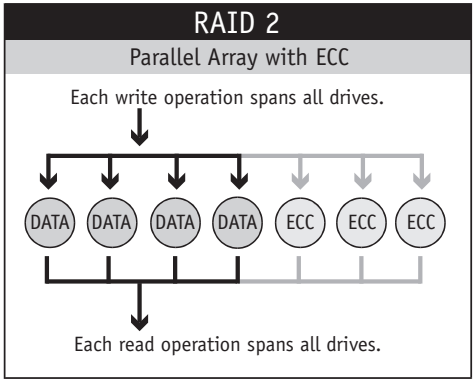
RAID 0 is typically defined as a group of striped disk drives without parity or data redundancy. RAID 0 arrays can be configured with large stripes for multi-user environments or small stripes for single-user systems that access long sequential records. RAID 0 arrays deliver the best data storage efficiency and performance of any array type. The disadvantage is that if one drive in a RAID 0 array fails, the entire array fails.



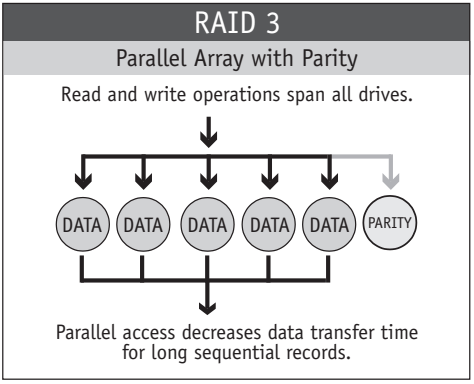
RAID 1, also known as *disk mirroring*, is simply a pair of disk drives that store duplicate data but appear to the computer as a single drive. Although striping is not used within a single mirrored drive pair, multiple RAID 1 arrays can be striped together to create a single large array consisting of pairs of mirrored drives. All writes must go to both drives of a mirrored pair so that the information on the drives is kept identical. However, each individual drive can perform simultaneous, independent read operations. Mirroring thus doubles the read performance of a single non-mirrored drive and while the write performance is unchanged. RAID 1 delivers the best performance of any redundant array type. In addition, there is less performance degradation during drive failure than in RAID 5 arrays.



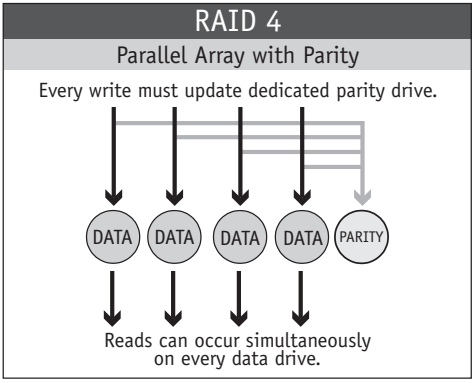
RAID 2 arrays sector-stripe data across groups of drives, with some drives assigned to store ECC information. Because all disk drives today embed ECC information within each sector, RAID 2 offers no significant advantages over other RAID architectures and is not supported by Adaptec RAID controllers.



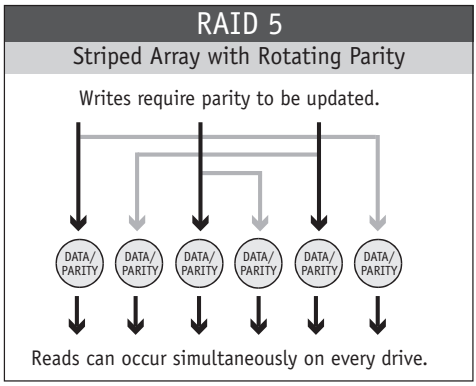
RAID 3, as with RAID 2, sector-stripes data across groups of drives, but one drive in the group is dedicated to storing parity information. RAID 3 relies on the embedded ECC in each sector for error detection. In the case of drive failure, data recovery is accomplished by calculating the exclusive OR (XOR) of the information recorded on the remaining drives. Records typically span all drives, which optimizes the disk transfer rate. Because each I/O request accesses every drive in the array, RAID 3 arrays can satisfy only one I/O request at a time. RAID 3 delivers the best performance for single-user, single-tasking environments with long records. Synchronized-spindle drives are required for RAID 3 arrays in order to avoid performance degradation with short records. Because RAID 5 arrays with small stripes can yield similar performance to RAID 3 arrays, RAID 3 is not supported by Adaptec RAID controllers.



RAID 4 is identical to RAID 3 except that large stripes are used, so that records can be read from any individual drive in the array (except the parity drive). This allows read operations to be overlapped. However, since all write operations must update the parity drive, they cannot be overlapped. This architecture offers no significant advantages over other RAID levels and is not supported by Adaptec RAID controllers.



RAID 5, sometimes called a Rotating Parity Array, avoids the write bottleneck caused by the single dedicated parity drive of RAID 4. Under RAID 5 parity information is distributed across all the drives. Since there is no dedicated parity drive, all drives contain data and read operations can be overlapped on every drive in the array. Write operations will typically access one data drive and one parity drive. However, because different records store their parity on different drives, write operations can usually be overlapped.



In summary:

- RAID 0 is the fastest and most efficient array type but offers no fault-tolerance. RAID 0 requires a minimum of two drives.
- RAID 1 is the best choice for performance-critical, fault-tolerant environments. RAID 1 is the only choice for fault-tolerance if no more than two drives are used.
- RAID 2 is seldom used today since ECC is embedded in all hard drives. RAID 2 is not supported by Adaptec RAID controllers.
- RAID 3 can be used to speed up data transfer and provide fault-tolerance in single-user environments that access long sequential records. However, RAID 3 does not allow overlapping of multiple I/O operations and requires synchronized-spindle drives to avoid performance degradation with short records. Because RAID 5 with a small stripe size offers similar performance, RAID 3 is not supported by Adaptec RAID controllers.
- RAID 4 offers no advantages over RAID 5 and does not support multiple simultaneous write operations. RAID 4 is not supported by Adaptec RAID controllers.
- RAID 5 combines efficient, fault-tolerant data storage with good performance characteristics. However, write performance and performance during drive failure is slower than with RAID 1. Rebuild operations also require more time than with RAID 1 because parity information is also reconstructed. At least three drives are required for RAID 5 arrays.

Dual-Level RAID

In addition to the standard RAID levels, Adaptec RAID controllers can combine multiple hardware RAID arrays into a single array group or *parity group*. In a dual-level RAID configuration, the controller firmware stripes two or more hardware arrays into a single array.

NOTE *The arrays being combined must both use the same RAID level.*

Dual-level RAID achieves a balance between the increased data availability inherent in RAID 1 and RAID 5 and the increased read performance inherent in disk striping (RAID 0). These arrays are sometimes referred to as RAID 0+1 or RAID 10 and RAID 0+5 or RAID 50.

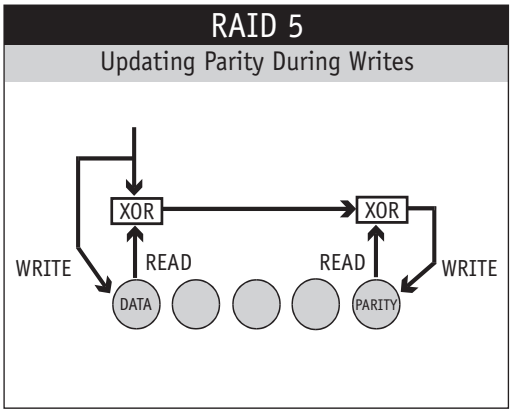
Creating Data Redundancy

RAID 5 offers improved storage efficiency over RAID 1 because only the parity information is stored, rather than a complete redundant copy of all data. The result is that three or more drives can be combined into a RAID 5 array, with the storage capacity of only one drive dedicated to store the parity information. Therefore, RAID 5 arrays provide greater storage efficiency than RAID 1 arrays. However, this efficiency must be balanced against a corresponding loss in performance.

The parity data for each stripe of a RAID 5 array is the XOR of all the data in that stripe, across all the drives in the array. When the data in a stripe is changed, the parity information is also updated. There are two ways to accomplish this:

The first method is based on accessing all of the data in the modified stripe and regenerating parity from that data. For a write that changes all the data in a stripe, parity can be generated without having to read from the disk, because the data for the entire stripe will be in the cache. This is known as *full-stripe write*. If only some of the data in a stripe is to change, the missing data (the data the host does not write) must be read from the disks to create the new parity. This is known as *partial-stripe write*. The efficiency of this method for a particular write operation depends on the number of drives in the RAID 5 array and what portion of the complete stripe is written.

The second method of updating parity is to determine which data bits were changed by the write operation and then change only the corresponding parity bits. This is done by first reading the old data which is to be overwritten. This data is then XORed with the new data that is to be written. The result is a bit mask which has a 1 in the position of every bit which has changed. This bit mask is then XORed with the old parity information from the array. This results in the corresponding bits being changed in the parity information. The new updated parity is then written back to the array. This results in two reads, two writes and two XOR operations. This is known as *read-modify-write*.



The cost of storing parity, rather than redundant data as in RAID 1, is the extra time required for the write operations to regenerate the parity information. This additional time results in slower write performance for RAID 5 arrays over RAID 1. Because Adaptec RAID controllers generate XOR in hardware, the negative effect of parity generation is primarily from the additional disk I/O required to read the missing information and write the new parity. Adaptec RAID controllers can generate parity using either the full- or partial-stripe write algorithm or the read-modify-write algorithm. The parity updated method chosen for any given write operation is determined by calculating the number of I/O operations needed for each type and choosing the one with the smallest result. To increase the number of full stripe writes, the cache is used to combine small write operations into larger blocks of data.

Handling I/O Errors

Adaptec RAID controllers maintain two lists for each RAID 5 array: a Bad Parity List, and a Bad Data List. These lists contain the physical block number of any parity or data block that could not be successfully written during normal write, rebuild or dynamic array expansion operations. These lists alert the controller that the data or parity in these blocks is not valid. If the controller subsequently needs data from a listed block and cannot recreate the data from existing redundant data, it returns an error condition to the host.

Blocks are removed from the Bad Parity List or the Bad Data List if the controller successfully writes to them on a subsequent attempt.

Degraded Mode

When a drive fails in a RAID 0 array, the entire array fails. In a RAID 1 array, a failed drive reduces read performance by 50%, as data can only be read from the remaining drive. Write performance is increased slightly because only one drive is accessed. A RAID array operating with a failed drive is said to be in *degraded mode*.

RAID 5 arrays synthesize the requested data by reading and XORing the corresponding data stripes from the remaining drives in the array. For RAID 5, the magnitude of the performance impact in degraded mode depends on the number of drives in the array. An array with a large number of drives will experience more performance degradation than an array with small number of drives.

Rebuilding a Failed Hard Drive

A failed drive can be replaced in a RAID 1 or RAID 5 array by physically removing the drive and replacing it or by a designated Hot Spare. Adaptec RAID controllers will rebuild the data for the failed drive onto the new drive or Hot Spare. This rebuild operation occurs online while normal host reads and writes are being processed by the array.

RAID 1 arrays are rebuilt relatively quickly, because the data is simply copied from the duplicate (mirrored) drive to the replacement drive. For RAID 5 arrays, the data for the replacement drive must be synthesized by reading and XORing the corresponding stripes from the remaining drives in the array. RAID 5 arrays that contain a large number of drives will require more time for a rebuild than a small array.