CS420: Operating Systems

RAID

James Moscola
Department of Engineering & Computer Science
York College of Pennsylvania

Based on Operating System Concepts, 9th Edition by Silberschatz, Galvin, Gagne

RAID

- **Redundant Array of Independent Disks (RAID)**
  - Combines multiple physical drives into a single logical volume
  - Originally created as an alternative to large expensive disks
    - Combine small inexpensive disks to achieve the same storage capacity at lower cost (was once 'Redundant Array of Inexpensive Disks')
  - Two main goals of RAID systems:
    - Increase reliability of storage through redundancy
    - Increase I/O performance by distributing the load

Improved Reliability Through Redundancy

- **All hard disks will fail eventually -- it's just a question of when**
- **With a single disk, if a disk failure occurs data is lost**
- **Reliability of data storage can be increased by using multiple disks**
  - If a single disk fails, one or more additional disks contain enough information to reconstruct the data from the lost disk
  - Multiple disks can be configured for redundancy in a couple of ways
    - Mirroring - one disk mirrors the data from another disk
    - Parity - a parity bit is stored for the bits located in the same location on each of the disks in a RAID

Improved Performance Through Striping

- **Mechanical disk drives are S L O W**
  - Approximately 8 ms seek time + some rotational latency

- **Speed of read and write operations can be improved by dividing up the data to be written and writing it to multiple disk -- striping**
  - Writing x bytes to a disk may take some time t
  - When striping data across n disks,
    - Only need to write x/n bytes to each disk
    - Time to read/write may be reduced by a factor of n

- **Striping may take place on different scales**
  - Write a bit each of file to a different disk; write a byte of each file to a different disk; write a block of each file to a different disk; etc.

Mean Time Between Failures

- **Mean Time Between Failures (MTBF) is a prediction of the amount of time between component failures (assumes that the failed component can be repaired/replaced)**
  - A measure of hardware reliability (e.g. disk reliability)
  - Does NOT mean that your hardware won't fail sooner
  - Does NOT mean that your disks won't all fail at the same time

- **Example of determining MTBF:**
  - Run 10,000 units for 1,000 hours each, count how many fail (e.g. 20 failures)
  - MTBF = (10,000 * 1000) / 20 = 500,000 hours ~= 57 years

- **If the MTBF of a single disk is 100,000 hours, then the MTBF of a RAID with 100 disks will be 100,000/100 = 1000 hours ~= 41 days**
  - This can be improved through redundancy

RAID Levels

- **RAID comes in a variety of different ‘levels’**
  - There are many standard levels of RAID
  - Different levels of RAID provide different advantages/disadvantages

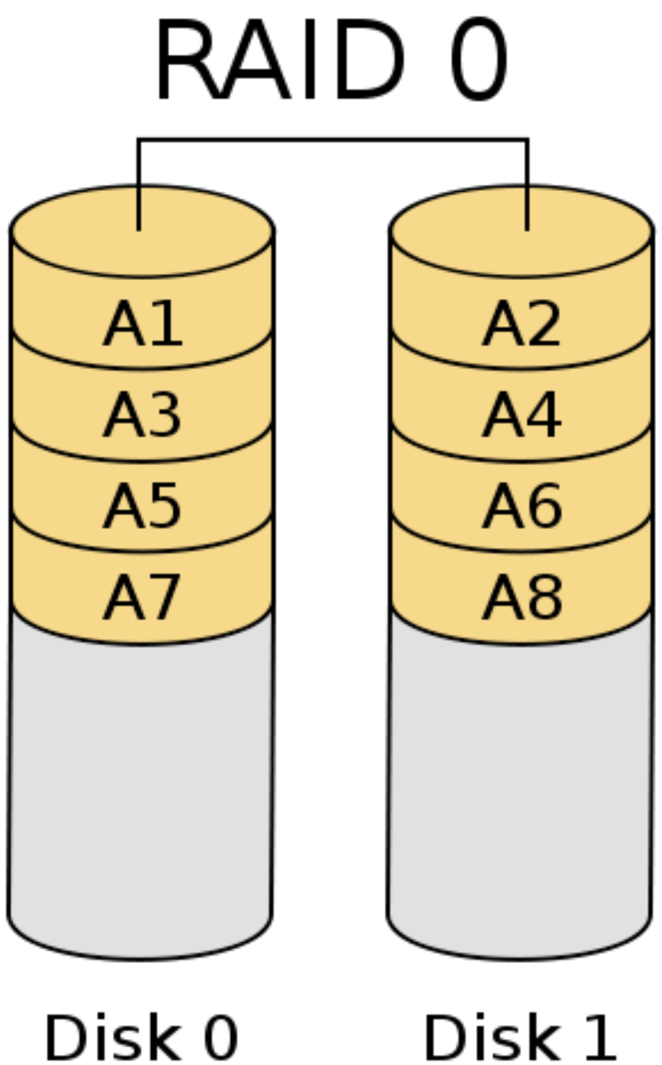
- **There are also a number of non-standard RAID implementations**
  - Typically developed by a company or research group that feels the standard levels of RAID do not meet the needs of all users

RAID Level 0 (Striping)

- **Block-level striping**
  - Simply provides the ability to combine multiple physical disk drives into a single volume
  - Improves the read and write performance by a factor of *n* where *n* is the number of disks in the array
  - Does NOT provide any redundancy -- data is NOT protected
  - A single disk failure means the contents of the entire array are lost

Array Failure Rate = $1 - (1 - r)^n$
where n is the number of disks in the array, and
r is the failure rate for each disk over some period of time

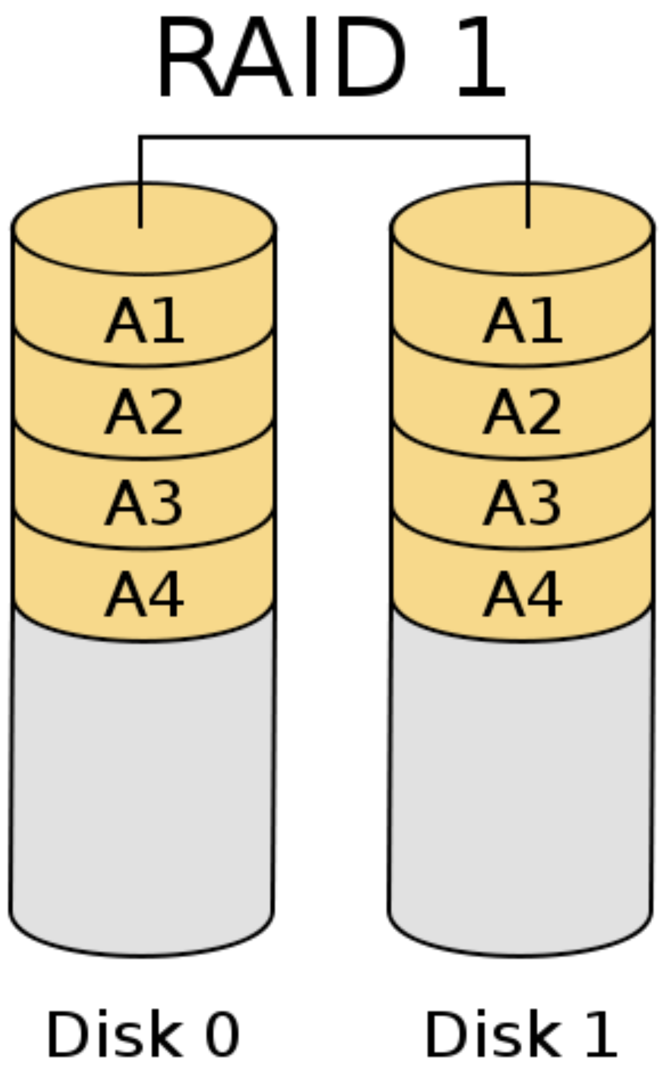
Example:
10 disks with 5% failure rate over 1 year
$1 - (1 - .05)^{10}$ = 40% chance array will fail by end of first year



RAID diagrams from Wikimedia Commons

RAID Level 1 (Mirroring)

- **Mirroring (no parity or striping)**
  - Each disk in the array has an exact copy somewhere else in the array (all data is written to *n* disks)
    - Provides redundancy through data replication
  - Can improve the read performance by a factor of *n* since data can be read from any of the *n* disks
    - No performance increase for writing data
  - Fault tolerant up to *n-1* disk failures
    - Read performance will degrade as disks fail
  - Can be expensive



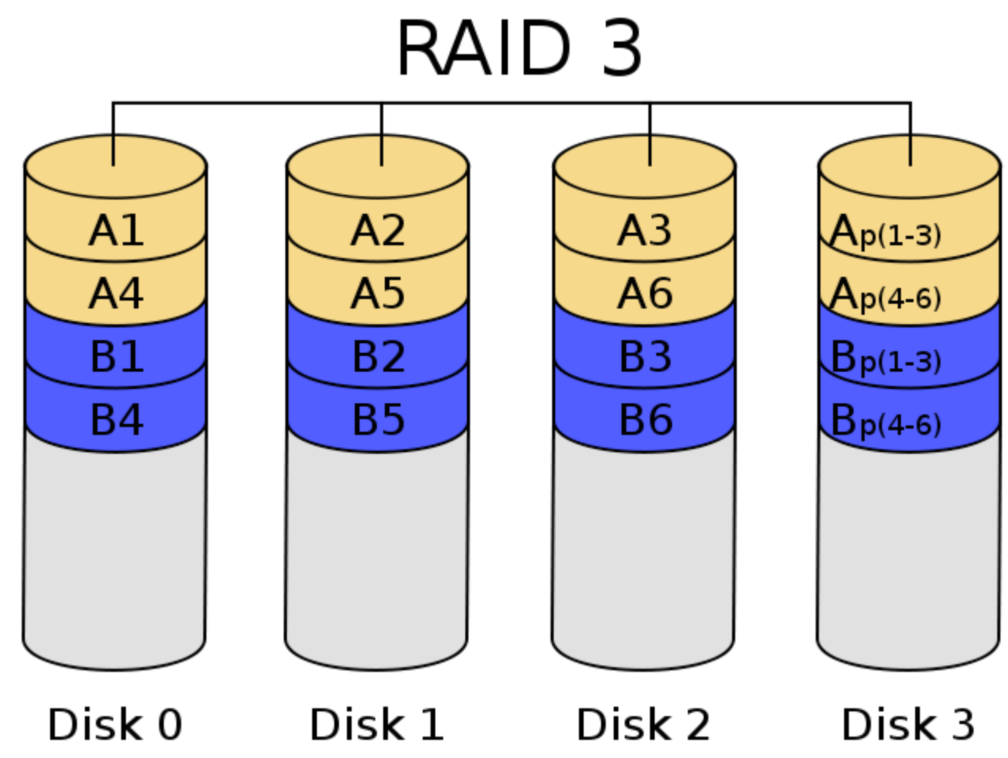
Array Failure Rate = r^n

where n is the number of disks in the array, and
r is the failure rate for each disk over some period of time

Example:
10 disks with 5% failure rate over 1 year
$.05^{10}$ = *very small chance of failure in first year*

RAID Level 3

- **Byte-level striping with dedicated parity disk**
  - Data is striped across *n-1* disks at the *byte* level, a single disk is used for parity
  - Only a single parity disk is required
  - Can improve the read performance by a factor of *n-1* since a portion of the data is read from each of the *n-1* disks
  - Can only tolerate a ***single*** disk failure
  - Since all disks are required for each I/O request, RAID 3 arrays can only service a single I/O request at a time
    - Read/write a block at a time
  - Parity computation can be expensive
    - Most RAID controllers contain specialized hardware to compute parity

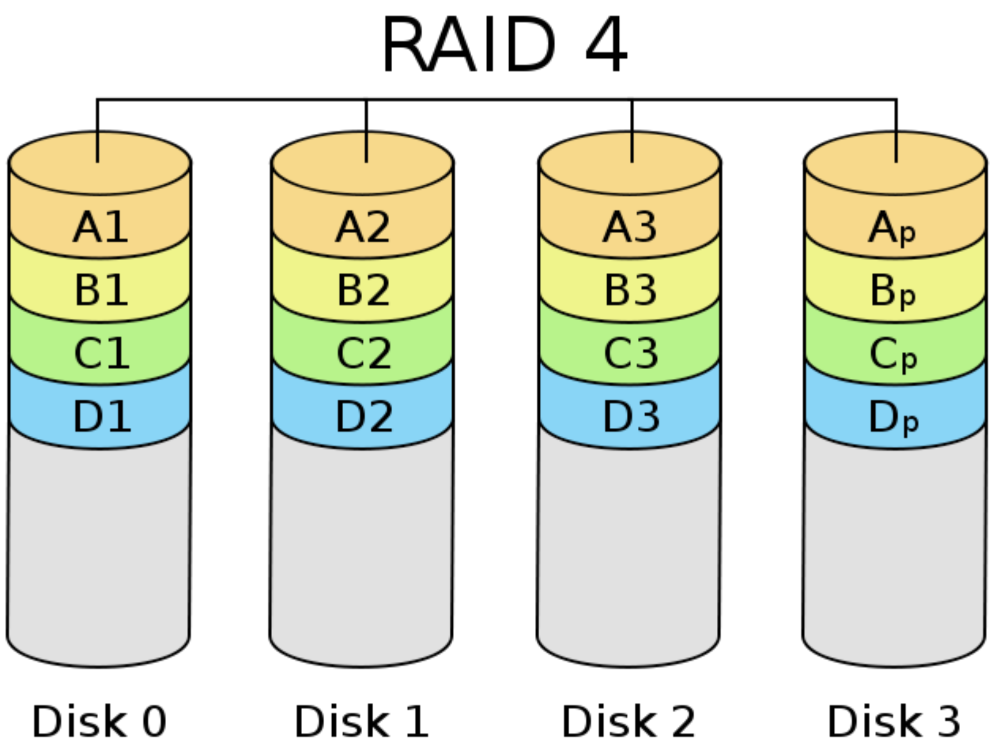


Array Failure Rate = $n(n-1)r^2$

Example:
10 disks with 5% failure rate over 1 year
$10(10-1) * .05^2 = 22.5\%$ chance of failure

RAID Level 4

- **Block-level striping with dedicated parity disk**
  - Data is striped across *n-1* disks at the *block* level, a single disk is used for parity
  - Only a single parity disk is required
  - Can improve the read performance by a factor of *n-1* since a portion of the data is read from each of the *n-1* disks
  - Can only tolerate a ***single*** disk failure
  - Can handle multiple reads in parallel when block sized reads are performed



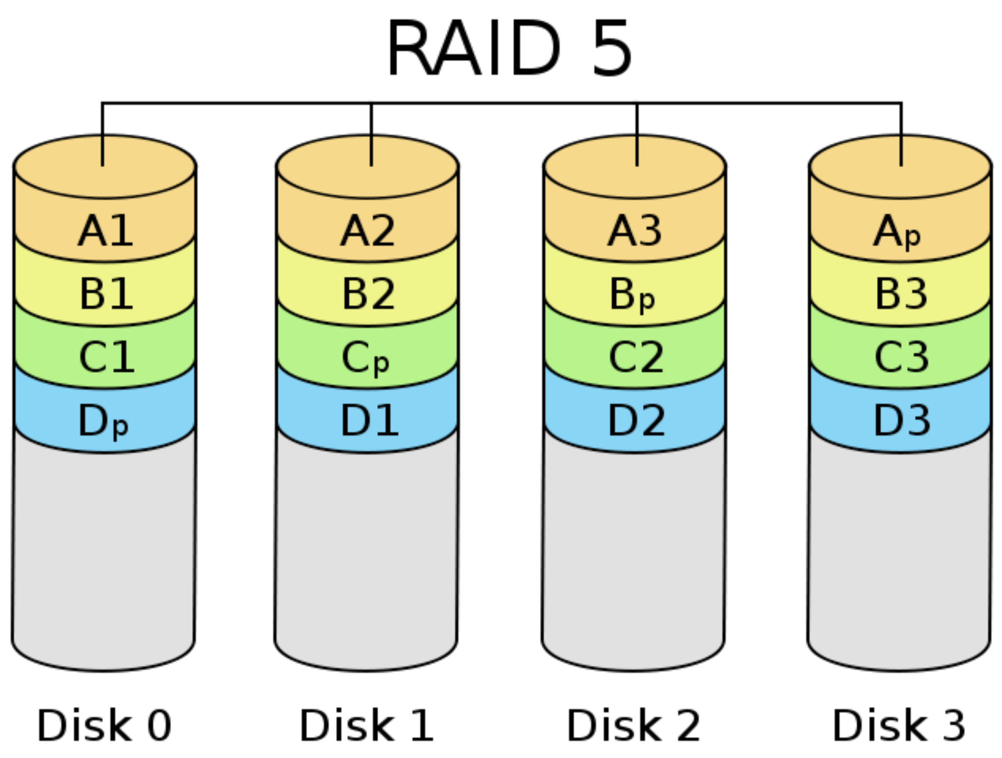
  - Parity computation can be expensive
    - Most RAID controllers contain specialized hardware to compute parity
  - Single parity disk can become a bottleneck
  - Writing data smaller than a full stripe is costly
    - Requires read/modify/write to compute new parity

Array Failure Rate = $n(n-1)r^2$

Example:
10 disks with 5% failure rate over 1 year
$10(10 - 1) * .05^2 = 22.5\%$ chance of failure

RAID Level 5

- **Block-level striping with distributed parity**
  - Similar to RAID Level 4, but distributes parity among all *n* disks in the array
  - Avoids overusing a single parity disk
    - Distributes parity load across all disks
    - Eliminates single parity disk bottleneck
  - Can only tolerate a single disk failure
  - Most popular RAID implementation since it provides good performance and redundancy is inexpensive



  - Suffers the same problems as RAID 4 when write is smaller than a stripe
    - Requires read/modify/write to compute new parity

Array Failure Rate = $n(n-1)r^2$

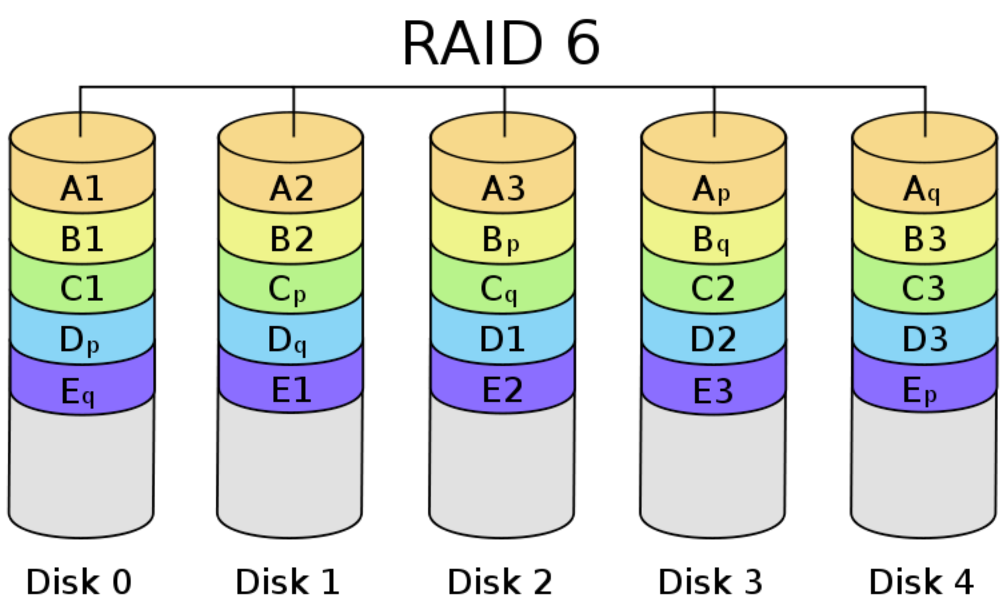
Example:

10 disks with 5% failure rate over 1 year

$10(10 - 1) * .05^2 = 22.5\%$ chance of failure

RAID Level 6

- **Block-level striping with <u>double</u> distributed parity**
  - Similar to RAID 5, but adds a second level of parity
    - Can tolerate two disk failures before data loss
  - Data and parity blocks are striped across all *n* disks in the array
  - Avoids overusing a single parity disk
    - Distributes parity load across all disks
    - Eliminates single parity disk bottleneck
  - Typically uses an ECC (error correction codes) such as Reed-Solomon coding instead instead of simple parity
  - Parity calculations are even more expensive than in previous levels of RAID
    - Use specialized hardware to compute ECC, typically included in RAID controllers

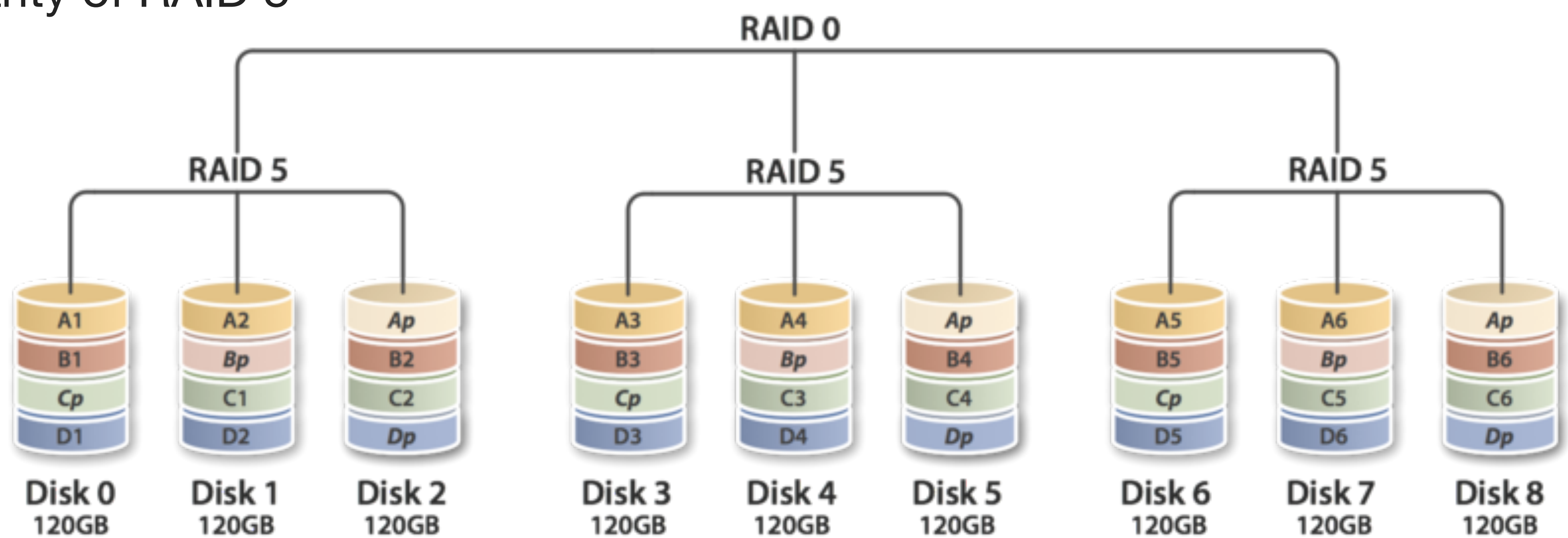


Array Failure Rate $= n(n-1)(n-2)r^3$

Example:
10 disks with 5% failure rate over 1 year
$10(10-1)(10-2) * .05^3 = 9\%$ chance of failure

Combining RAID Levels (Hybrid RAID)

- **RAID levels can be 'nested' to get benefits from two different levels**
- **Examples:**
  - RAID 0+1 - uses a second striped set to mirror the first striped set
  - RAID 5+0 (a.k.a RAID 50) - combines block level striping of RAID 0 with distributed parity of RAID 5



  - Many other possible nested RAID levels, RAID 10, 51, 60, 61, 100, etc.

Rebuilding an Array

- **When a disk failure occurs, it is important to replace the disk as soon as possible**
  - Many RAID levels can only sustain a single drive failure
  - Rebuilding the array can take **MANY** hours (10+ hours depending on disk sizes)
  - A secondary failure during array rebuild can cause total data loss :-(
  - A secondary failure during array rebuild becomes more likely as ALL disk are working continuously to rebuild array

- **RAID 6 provides better protection during array rebuild as it can sustain a second failure and still continue to rebuild the array**

Nonstandard RAID Implementations

- **Many nonstandard versions of RAID exist**
    - RAID-DP (double parity) - uses two bit parity on dedicated disks
    - BeyondRAID (Drobo devices)
    - RAID-Z (ZFS filesystem)
    - unRAID :-)
    - Many others