

Peter's Solaris Zone

Raid and reliability

Individual disk drives are moderately reliable, having lifetimes measured in years. A fairly standard MTBF of 500,000 hours is 57 years.

Applying simple procedures such as mirroring can dramatically improve overall reliability. But how much?

So, here are some quick calculations. There's some terminology here:

Variable	Meaning
F	Mean time between failure
R	Mean time to repair
N	Number of data disks
L	Mean time to data loss

Single disk

This is easy. The drive fails.

$$L = F$$

Mirror

Again, this is pretty easy. The expected life is that of either drive (which is half that of one drive), divided by the probability that the second drive will fail before the first one is replaced - which is just R/F .

$$L = (F/2) / (R/F) = F^2 / 2R$$

As (normally) F is much greater than R , mirroring dramatically improves your availability.

Stripe

The lifetime here is reduced because you lose all the data as soon as any drive fails, so your lifetime is reduced by the number of drives.

$$L = F/N$$

Mirrored Stripe

Also known as RAID-1+0. There are $2N$ drives in total, so the time to the first failure is $F/2N$. However, you only lose data if the matching mirrored drive fails in the repair window.

$$L = (F/2N) / (R/F) = F^2 / (2N \cdot R)$$

As you can see, the overall reliability is reduced from the simple mirror by the number of disks in the stripe.

RAID-5

You need $N+1$ drives for data, so the time to the first failure is $F/(N+1)$. However, any failure of the remaining N drives causes total data loss.

$$L = F * F / (N * (N+1) * R)$$

Note that a RAID-5 is much less reliable than a mirrored stripe. In some ways this is obvious - the mirrored stripe has more drives and therefore ought to be safer - but isn't often appreciated.

Note also that the reliability of a raid-5 system decreases quite rapidly as more disks are added. What this actually means is that splitting a big raid-5 into 2 smaller ones and then putting those together will double your overall reliability.

Some example times

Just for fun, I assume a MTBF of 500,000 hours and call that 60 years, and assume for stripes and the like that I have 10 data drives. I then calculate the expected time to data loss for 2 scenarios - hotspare with MTTR of 5 hours, and service call with MTTR of 50 hours (or two days).

Type of storage	Life with hotspare	Life waiting for service
Plain disk	60yrs	60yrs
Mirror	3 million years	300000 years
Stripe	6yrs	7 months
Striped Mirror	300,000 years	30,000 years
raid-5	50,000 years	5000 years

Let's try that again, with cheap desktop drives that may only have a life of 100,000 hours:

Type of storage	Life with hotspare	Life waiting for service
Plain disk	12yrs	12yrs
Mirror	120,000 years	12,000 years
Stripe	14 months	6 weeks
Striped Mirror	12,000 years	1,200 years
raid-5	2,000 years	200 years

Ouch!

Availability

Availability is a different calculation again. The fraction of time that the data is unavailable is the time it takes to restore the system (rebuild the storage plus restore the data) divided by the time it takes for the system to fail. I assume that the rebuild time is equal to the mean time to repair and the restore time is proportional to the number of disks (more data) and that it takes an hour per disk. In the case of the single-disk and single-disk mirrors, I assume there are N of them.

(This gives an advantage to the single-disk setups - they might not be any more reliable, overall, but being smaller they're quicker to get the data back after failing.)

Then the availability is the unavailability subtracted from 1.

Type of storage	Unavailability with hotspare	Unavailability waiting for service
Plain disk	0.000114 (99.9886)	0.00097 (99.90)
Mirror	2e-9 (99.999998)	2e-7 (99.99998)
Stripe	0.00028 (99.972)	0.0114 (98.86)
Striped Mirror	6e-9 (99.999994)	6e-7 (99.99994)
raid-5	3e-8 (99.999997)	1.37e-6 (99.99986)

Again, with cheap desktop drives that may only have a life of 100,000 hours:

Type of storage	Unavailability with hotspare	Unavailability waiting for service
Plain disk	0.00057 (99.943)	0.00485 (99.515)
Mirror	5e-8 (99.999995)	5e-6 (99.9995)
Stripe	0.0014 (99.86)	0.057 (94.3)
Striped Mirror	1.5e-7 (99.999985)	1.5e-5 (99.9985)
raid-5	7.5e-7 (99.999925)	3.4e-5 (99.996)

Commentary

These numbers don't have any close relationship to reality, of course.

One thing that is almost certainly true, in my experience, is that disk failures tend to be highly correlated. There are several reasons this is to be expected. For starters, you tend to get a lot of early failures when systems are new, then a low trough, then a ramp up as the system gets old. Also, some arrays fail drives much more often than others (some manufacturing or transport gremlin at work?). Then there may be some external cause, such as an environmental fluctuation (temperature spikes, for example, or a power cut) or a change in usage that hits the drives hard. Then there's the simple fact that there's extra strain on the surviving disk(s) when the redundancy has failed - not only is it having to handle more load anyway, but the reconstruction process can be intensive. And that doesn't allow for multiple events due to controller or cable failure, or to bad devices causing interference.

The main conclusion you can draw from these numbers is that **redundancy is essential**, which ought to be obvious. Essentially any level of redundancy gives reasonable reliability.

The secondary conclusion is that having hotspares available gives a significant improvement. It reduces the window of vulnerability from days to hours. It's not just how long it takes to ship a drive, either - somebody has to spot it's failed and be around to do the work, whereas a hotspare kicks in automatically and immediately.

Another comment is in order. Given the difference between redundant and non-redundant configurations, you should **never break a mirror to do backups**.