the Availability Digest

www.availabilitydigest.com

What's Your Concern – MTR or MTBF?

November 2010

Fast recovery from a fault is what users look for. But frequent failures can drive the system operators to a frenzy. So what should we be seeking in a highly available system – fast recovery or infrequent failures? Let's look at this question further.

Is the Availability Barrier MTR?

Continuous-availability systems provide (nearly) 100% uptime by recovering from a fault so rapidly that no one notices that there has been outage (or at least no one is inconvenienced by it). In most of our discussions in the Availability Digest, we have for this reason focused on how to achieve very fast recoveries.

A leading example of a continuously available system is an active/active system,[1] in which all nodes are actively processing transactions and in which a transaction can be routed to any node for processing. Should a node fail, all that needs to be done is to reroute traffic to surviving nodes. With the right architecture, this can be done in seconds to subseconds.[2] Who will even notice?

Following this philosophy, we argued in our article entitled What Is the Availability Barrier?[3] that it is the recovery time, or MTR (mean time to recover), that is important to commercial data processing. While this is true for the user community, it ignores another very important community – the system operators.

What About MTBF?

At a recent Business Continuity SIG (Significant Interest Group) at HPTF (the HP Technology Forum), the HP liaison expressed some dismay at this interpretation of MTR being all important. He pointed out that to the system operators, an outage is an outage. Whether recovery is in seconds, minutes, or hours, the system must still be repaired; and that takes a lot of effort and knowledge on the part of the operations staff.

The rate of failures is the major impact on them. With every failure, they must scurry and correct the fault. The failure interval, or time between failures (MTBF – the mean time between failures),

[1] What is Active/Active?, *Availability Digest*; October 2006.
http://www.availabilitydigest.com/public_articles/0101/what_is_active-active.pdf.

[2] Achieving Fast Failover in Active/Active Systems – Parts 1 and 2, *Availability Digest*; August/Sept 2009.
http://www.availabilitydigest.com/public_articles/0408/user_redirection.pdf.
http://www.availabilitydigest.com/public_articles/0409/user_redirection_2.pdf.

[3] What Is the Availability Barrier, *Availability Digest*; March 2010.
http://www.availabilitydigest.com/public_articles/0503/availability_barrier.pdf.

© 2010 Sombers Associates, Inc., and W. H. Highleyman

may be more important to them than MTR. Even if the system is a redundant system, such as an active/active system with automated rapid recovery, the operations staff must correct the fault before a second fault can indeed take the system down for an extended period of time.

An Example

Consider a system with three 9s availability (it is up 99.9% of the time). If you are a user of this system, would you rather be down for:

- one second every seventeen minutes? (your PC probably does this).
- one minute once a day? (a minor aggravation).
- one hour once every six weeks? (a major aggravation).
- one day every three years? (perhaps major damage to your company).
- one week every 20 years? (would your company survive this?).

As an end-user, you might choose one second every seventeen minutes or one minute a day. You probably won't notice the one-second outage, and a one-minute outage might be nothing more than a little annoying. Certainly, neither will cause you to stop work and call the help desk to find out what is going on.

But what is going on behind the scenes? The one-second outage is causing the system operators to track down and fix some sort of problem four times an hour. This is probably intolerable – they may not be able to keep up, meaning that the system may be heading for disaster with a dual outage of some critical component.

The one-minute outage is a little better, but the staff must still work pretty quickly. What if a multi-hour recovery of a system is required? What if the repair requires a new part that may take a few days to acquire? Several of these incidents could well lead to a shaky system with several simultaneous faults under repair.

The staff will probably find an outage every six weeks quite acceptable. The workload to recover from faults will not be that great, and it is unlikely that there will be several failed components outstanding at the same time. But users will now surely be affected as they wait around for the system to come up.

What will management think? That depends upon the application. Certainly, the amount of downtime is important to them. But we have assumed a three-9s system, which by definition causes eight hours of downtime per year. Given that, the one-hour-every-six-weeks scenario might be a good choice. The impact of downtime may well be minimal – one hour of employees sitting around (assuming that this is not a critical application, in which case a three-9s system shouldn't be used anyway). The chance of multiple faults under repair will be minimal.

It is interesting to note that typical parameters today for a three-9s commodity Windows or Linux server are in the order of four hours of downtime every six months. This is what we have learned to live with.

Thus, MTR and MTBF are both important to some segment of the user community. One should not be optimized in the absence of consideration of the other.

Calculating MTBF

These considerations raise the question of how to calculate MTBF. If we have a system comprising several components, the failure of any one of which will cause the system to fail, and each with its own MTBF, what is the MTBF of the system? Obviously, it is not just a matter of adding the MTBFs.

© 2010 Sombers Associates, Inc., and W. H. Highleyman

Let us consider a simple system comprising a network, an application server, and a database server, as shown in Figure 1. We will use lower case to denote component parameters and upper case to denote system parameters.

The mtbf of the network is 24 months (two years). The mtbf of the application server is six months (one-half a year). The mtbf of the mirrored database is 60 months (five years). What is the MTBF that we can expect of the system?

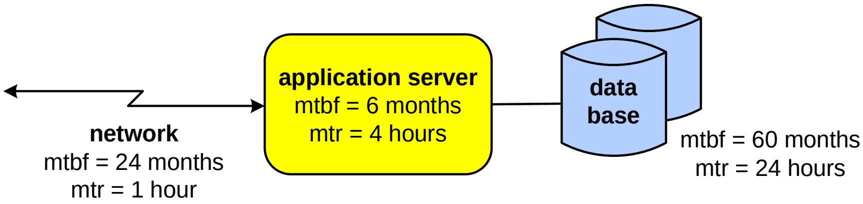

Figure 1: MTBF Example

We cannot add mtbfs. But we can add failure frequencies. Let us take a five-year period. The network will fail 2.5 times in five years. The application server will fail ten times in five years. The database server will fail once in five years. (They are all averages, of course). Thus, we can expect the system to fail 2.5+10+1 = 13.5 times in a five-year period. This is a failure every 4.44 months (the system's MTBF). Note that the MTBF of the system is less than that of its weakest link, the application server in this case.

For the mathematically inclined, let mtbf_i be the mtbf of component *i*. Then the system MTBF is the reciprocal of the sum of the reciprocals of the component mtbfs:

$$\text{MTBF} = \frac{1}{\dfrac{1}{\text{mtbf}_1} + \dfrac{1}{\text{mtbf}_2} + \dfrac{1}{\text{mtbf}_3} + \ldots} = \frac{1}{\sum\limits_i \dfrac{1}{\text{mtbf}_i}}$$

Conforming MTBF with MTR and Availability

Just how does this correlate with system MTR and system availability, A?

First of all, with the mtbf and mtr component parameters given in Figure 1, we can calculate the availability, *a*, of each component. Remembering that *a* = (mtbf-mtr)/mtbf, and using 720 hours per month, we find

a (network) = (24x720-1)/(24x720) = 0.999942
a (application server) = (6x720-4)/(6x720) = 0.999074
a (database server) = (60x720-24)/(60x720) = 0.999444

This is a serial system in which the failure of any component will cause a system failure. The system will be up if the network is up and if the application server is up and if the database server is up. Since the probability that a component will be up is its availability, the system availability, A, is the product of its component availabilities:

A = 0.999942 x 0.999074 x 0.999444 = 0.998461

The average recovery time for the system, MTR, can be found from the basic availability equation, A = (MTBF-MTR)/MTBF, or MTR = MTBF(1-A). Since we have calculated an MTBF of 4.44 months, we have

© 2010 Sombers Associates, Inc., and W. H. Highleyman

MTR = (4.44x720)(1-0.998461) = 4.9 hours

Thus, the system of Figure 1 has an MTBF of 4.44 months, an MTR of 4.9 hours, and an availability of 0.998461. In round terms, it has an MTBF of 4 ½ months, a five-hour MTR, and an availability of a little less than three 9s.

Summary

There are more than just the users of a system who are interested in the availability of a system. There are also the system operators and management.

Users will typically be interested in recovery time, MTR. The faster the recovery time, the less impact an outage has on them. System operators will typically be more interested in the failure interval (MTBF) since that defines their workload in terms of component repairs (though fast recovery helps them by minimizing the stress to get a component fixed). Management is interested in minimizing downtime and in balancing MTR and MTBF for the benefit of the enterprise.

Note that an active/active system can achieve all of these goals – recovery times measured in seconds, downtimes of seconds per year (six 9s is 30 seconds of downtime per year), and MTBFs of centuries.

© 2010 Sombers Associates, Inc., and W. H. Highleyman