the Availability Digest

www.availabilitydigest.com
@availabilitydig

The High Availability Design Spectrum – Part 2

Dr. Terry Critchley
January 2017

[Editor's Note: In his book "High Availability IT Services," Dr. Terry Critchley lists twenty-three areas that can have an impact on the availability of IT business services. In this multipart series, and with his permission, we publish his observations. In Part 1 of this series, we reviewed his first four reflections - his Parts A through D. In this Part 2, we examine his next nine considerations – his parts E through M.]

Dr. Terry Critchley: Most of the documentation on HA/DR I have come across majors on hardware, mainly redundant or fault tolerant, and, to some extent, software. My thesis is that the spectrum of activity needed to design, implement and maintain a high availability business IT system and recover from failures small and large (DR) is much, much greater. Below, I have listed 23 areas (A to W) which can have an impact on the availability of business services which are IT-based. I am sure it will be evident that these areas can have a significant impact on the availability and non-availability of any service or system.

Remember, focusing on availability and focusing on avoidance of non-availability are not the same thing, if you think about it.

The book and chapter references following refer to 'High Availability IT Services': https://www.crcpress.com/High-Availability-IT-Services/Critchley/9781482255904.

E. Availability by Outside Consultancy

Several vendors and IT consultancies offer specialist services related to achieving high availability. These can cover availability assessments of current systems, availability design/modifications for new/current systems, remote proactive monitoring, creation of operational procedures for backup and recovery and so on.

Your vendors would be a good start for high availability assistance, but others specialize in this kind of work as well.

F. Availability by Vendor Support

To ensure rapid response by vendors to outages caused by their products, it will be necessary to have suitable service contracts with the vendors. This includes software and other vendors - not just hardware vendors. The chain of availability is only as strong, at theoretical best, as its weakest link. It is pointless having a 24 x 7 support contract for the hardware but only an 8 x 5 support contract for the database that supports the application. In addition, most IT vendors offer a remote monitoring and diagnostic service for proactive observation of system hardware and, in some cases, software. Ask your vendor(s) about these

facilities and make sure you understand the difference between bronze, silver, gold and platinum services.

G. Availability by Proactive Monitoring

Given that suitable tools are available, it is often possible to predict where failures might occur if corrective action is not taken in time. Failures include hardware, database overflow, and other 'full' conditions, which might lead to application non-availability, plus the question of physical and logical security.

There are a number of other tools which can help with proactive monitoring of over-utilized resources such as the network. Bottlenecks should be anticipated before they disrupt (change management) or degrade (capacity planning) the service.

Remember, poor performance can often be construed by users as an outage of their system. A form of performance degradation is the presence of 'rogue' or 'runaway' tasks which may hog CPU and I/O resources. It is desirable to have a process for detecting and 'killing' such tasks when necessary.

The key disciplines of Change, Capacity and Performance Management are key ingredients in baking a successful '*availability*' cake, along with other aspects of service management.

H. Availability by Technical Support Excellence

To ensure maximum availability, it is key that staff supporting the Data Center systems are adequately trained and work hand in hand with the incumbent vendor(s). Many outages can be attributed to 'fat finger trouble' by technical support staff as well as operators - the ever-present 'liveware' threat.

A paper by IDC[1] says that *'to maximize the benefits of systems and technology ... an organization's workforce must be high performing and well skilled*.' This phrase applies in this section, the next and, to a lesser extent, elsewhere. A study quoted in the paper indicates that the two main factors contributing to the success of the IT technology function are team skills (61%) and support by the technology vendor (19%). The figures quoted here are for IT managers' perceptions in looking at the key IT operating metrics:

- data and backup recovery
- endpoint/perimeter security
- high availability
- archiving and retrieval
- client [*user*] management

The paper is well worth some study, especially as it points out that skills lag the use and need for various technologies in a form of *hysteresis*. An example today would be the recognition of security and performance as key to maintaining proper service availability but with a possible skills lag.

The items following are further examples of areas where support and operational errors can occur and often have far reaching implications for availability. There are doubtless many others.

Summary of H: The major ongoing task here is *education* and maintaining currency of skills. See item U below to see the broader aspects of this skills requirement. It involves the organizational structure of IT and, believe it or not, some parts of the business. As availability targets are raised, the list of items that need consideration grows beyond just obvious hardware and software reliability. Remember, a little knowledge is dangerous.

[1] 'Information Security and Availability: The Impact of Training on IT Organizational Performance' http://eval.symantec.com/downloads/edu/Impact_of_Training_on_Organizational_Performance.pdf, sponsored by Symantec.

I. Availability by Operations Excellence

Many outages of applications can be attributed to poor operations and technical actions. Vendors through their installations can quote several instances of 'failure' caused by inadequately trained administration personnel.

One such instance is the recovery from failure of mirrored disks, where the good mirror in brought into synchronization with the corrupt mirror instead of vice versa. Another example is switching a machine off to recover from a 'stalled' situation. A further example is entering the wrong time/date, at some point causing havoc with the execution of '*cron*' jobs or applications which rely heavily on accurate times/dates for their correct functioning.

These incidents are all too common in many modern environments and are still occurring today in all probability in IT organizations called 'laggards' by some Gartner reports on various topics. See notes in H above.

First Class Runbook [2]

Information and technical procedures in the Runbook should provide support of the following areas:

- *System Configuration Information* - build and configuration information about the installation is documented with mechanisms for update.
- *Routine Housekeeping* - recommended actions to be performed on a regular basis.
- *Platform Tools* - guidelines and procedures showing how to use the toolsets which vendors supply.
- *Startup/Shutdown* - technical procedures for starting up and shutting down hardware and software
- *General Administration* - technical procedures to ensure that the platform is running in an optimum state such as checking logs, etc.
- *Maintenance* - technical procedures which upgrade or otherwise change the configuration of the platform such as adding new components or upgrading a running Cluster environment.
- *Troubleshooting* - information to help identify faults and ensure rapid escalation to the vendor(s).
- *Backup Procedures* - these should be at least semi-automated, but knowledge of them is needed if they don't run according to plan.
- *Recovery* - recovery procedures to be run following a variety of fault scenarios.
- *Documentation Library* - electronic and web based set of internal and vendor product and procedure manuals.

A Runbook should not attempt to replicate the information gained through training courses, etc. Rather, it should assume an appropriate level of skill of the operations and administration staff. It is not simply a prescriptive set of instructions - it should be a *living document*, updated regularly and containing wisdom from ongoing operations.

Software Level Issues

I have observed that often, when pieces of software interact with each other (for example, driving communications hardware, operating system, middleware and so on), there are sometimes ‘glitches' in this interface. What happens then is the software functions cease to communicate or cooperate, causing an embarrassing ‘wait’ between them but no tangible evidence of an outage. The key to this is to make sure interfacing software is at the level specified in the release documentation.

For example, if SW A only works with SW B at levels (0) and (-1), and you expect it to work with SW B at level (-2) or lower (earlier version), you may have trouble. This happened several years ago with a pair of IBM communications controllers (3705s) where contact was lost because each controller was waiting for the other to reply in some way ['*ac*k' (acknowledgement) or *nak*' (negative acknowledgement) or similar]. There was no indication of an outage except an embarrassing 'nothingness.'

[2] See Appendix 1 for a Runbook (operations manual) overview.

System Time

Some errors which result in non-availability of applications are independent of hardware failures. An example is the incorrect setting of system clocks by administrators. This can result in '*cron*'-related jobs being run at the wrong time or even wrong day. Some installations keep their system clocks synchronized to a reliable external clock, such as the Rugby (UK) clock. There are others which provide this facility.

Some installations reduce 'fat finger trouble' issues by using software to automate times and other things, such as job scheduling, backup of databases and so on. Previously, this was thought to lead to the nirvana of the unattended or 'dark' machine room, which is illustrated technically in item L. below.

Performance and Capacity

Rigorous *performance* and *capacity* management are essential if the system appears to the end user to have stalled because of very poor response times. To him or her, the application is in essence unavailable. These disciplines are not covered in any detail in this book because they are somewhat detailed and I believe that a little knowledge is more dangerous than none. I therefore recommend the use of other resources for these tasks.

Data Center Efficiency[3]

It goes without saying that in data centers, *'running a tight ship'* should be the watchword to provide against outages.

> 'Automation is typically the next step in the data center journey. Introducing higher levels of automation enables greater levels of flexibility and helps support even higher levels of availability. Greater reliance on automation tools and technologies offloads manually intensive tasks for system administrators, reduces error rates, and ensures the performance of applications against their SLAs' (from the referenced IBM report).

The major ongoing task here, as before, is *education* and maintaining *currency* of skills.

J. Availability by Retrospective Analysis

This is essentially the data analysis part of Root Cause Analysis (RCA - see Appendix 1) should it be required. When errors occur, it is sometimes possible to interrogate log files post-event (performance, change, operating system etc.) to ascertain what the cause of an outage was and any attendant circumstances which might be included in the RCA and operations documentation. Manual operations logs (handwritten or perhaps PC-based) can also offer clues to outages if they are kept as working documents.

DEC used to supply a leather-bound operations log book in the heady days of good business. These may or may not avoid a similar outage occurring again as can *deja vu* from observant technical support.

Exactly what can be done further here depends on what the relevant vendor and associated third parties can offer in the areas of monitoring. There is a philosophy of mine that says *'capture everything that changes or moves and worry about whether it is of any use later'*. If you haven't got it you can't do anything with it and you can always delete it when it is past its *'sell by'* or *'useful date'*.

K. Availability by Application Monitoring

In recent years, COTS (Commercial Off The Shelf) software, particularly ERP, has been subject to monitoring. This is often provided by the COTS vendor and often by vendors of systems management software, such as BMC, Tivoli and so on. In fact, some ERP packages provide HA functions for their software in conjunction with operating on, and cooperating with, hardware clusters and other HA facilities.

---

[3] **IBM report called:** ***'Data Center Operational Efficiency Best Practices'***
http://docs.media.bitpipe.com/io_10x/io_105085/item_542158/Data_Center_Study.pdf

The chain of events when an outage occurs in a system when failures or application non-availability arise are shown in Figure 43. The development of problem determination (t1), backout and recovery procedures, etc. (t2 and t3) and training are vital to increasing application availability by reducing downtime.

L. Availability by Automation

We have alluded to this earlier but the idea of automating activities is receiving acclaim for various reasons, including productivity, speed, and the elimination of many (not all) user errors. Keep an eye open for this topic, which is gaining ground and whose nirvana, full autonomic computing, is illustrated below. Torches with long-life batteries or fully automated procedures are needed to function in this environment.

Remember that automation has degrees of sophistication and effectiveness and is not yet fully autonomic computing.

There is still the need for:

- understanding of your IT operations and areas 'ripe' for automation. You might choose to run a Delphi session on the subject supported by a BIA (business impact analysis)
- correct choice of tools
- effective use of automation tool(s)
- ongoing operations skills to reduce 'fat finger trouble'
- other items specified in various places in this book

You will find an increasing number of articles about this topic, often accompanied by product descriptions.

There are numerous existing articles on automating backup and even disaster recovery, for example, at the URL below. For others, search on ' backup automation, disaster recovery.'

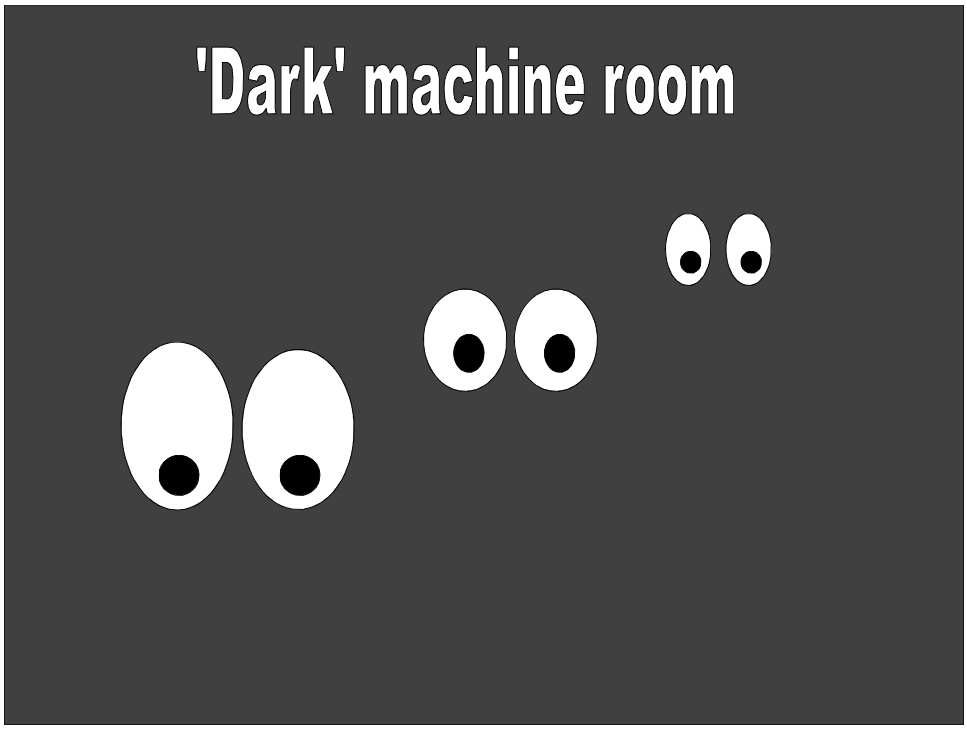

Figure 1 The Unattended ('Dark') Machine Room

http://www.itbusinessedge.com/blogs/infrastructure/time-to-automate-your-disaster-recovery-program.html

See Chapter 2, *'Automation. The Solution*?' under Availability and Change. M. Availability by Reactive

M. Availability by Reactive Recovery

The normal operational actions for a problem are - detect, locate, identify and fix. Recovery from failure is a key element in availability management. Recovery after a problem occurs is reactive but proactive monitoring can prevent problems from occurring in many instances. This latter might be called 'virtual recovery' whereas reactive recovery is very real. An example of proactive recovery would be the servicing or replacement of parts which show soft errors above a recommended threshold.

A second example might be a tuning exercise undertaken when a service's response time is increasing and might eventually break some clause in a Service Level Agreement (SLA).

When proactive actions do not avoid problems, then thorough, documented recovery procedures (often known as ***Runbooks***) need to be in place.

The example below illustrates the scope of problem determination and recovery. Procedures such as those listed in the example need to be 'living' procedures, modified in the light of experience of availability issues.

There are tools, usually tied to vendor products, for automating some aspects of Runbook development. Ideal Runbooks are online and maintained religiously. Procedures such as the example below will need to be developed for the other 'elements' in the service chain between user and the servers and within the servers.

It is obvious that online operations procedures are more effective when maintained **but** don't put them on the system(s) you are monitoring, for obvious reasons. Use a separate PC and duplicate it (see the 'Availability Monitoring' section below). This should be repeated for other parts of the chain of hardware and software from end user to the applications.

Operations Procedures 12.3 - Recovery

Application: Online Orders

System Element: Opsys1 operating system

Proactive Monitoring Information: Opsys1 console

Problem Determination: PROB09

Auto-recovery Facilities: Automatic reboot

Availability Architecture: Shadow OS

Business Impact: Delayed or lost orders

Impact (H/M/L): H

Recovery Procedures: REC07 if auto reboot fails

Estimated Recovery Time: 10 mins

Recovery Contacts: Service desk x3456, technical support x4567.

MTBF This Type: 1500 hours

Last Occurrence: 13:45 12/5/2013

Another perspective on recovery is to classify problem determination by domains, which were discussed earlier. For example. consider the following as usable domains:

- Remote server domain recovery
- LAN domain recovery
- WAN domain recovery
- External domain recovery, for example external power supply
- Server recovery
- Data recovery
- Application recovery
- Recovery of other vital components and/or peripherals

Recovery Times: The time to recover (MTTR) will be between two values**:**

$$T_{MAX} \text{ and } \sum T_i$$

where T_{MAX} is the longest component recovery time for completely overlapped recovery and $\sum T_i$ is the total of the individual recovery times (T_i) for totally non-overlapped recovery - end to end recovery. It is important that the service agreements for each block in the service configuration match the availability requirements. For example, it is not feasible to have a 7 x 24 hour service agreement for RDBMS databases and only a 5 x 8 hour agreement for the vendor's hardware or other components of the service configuration. This covers things like ATM switches, software, application cover, and so on.