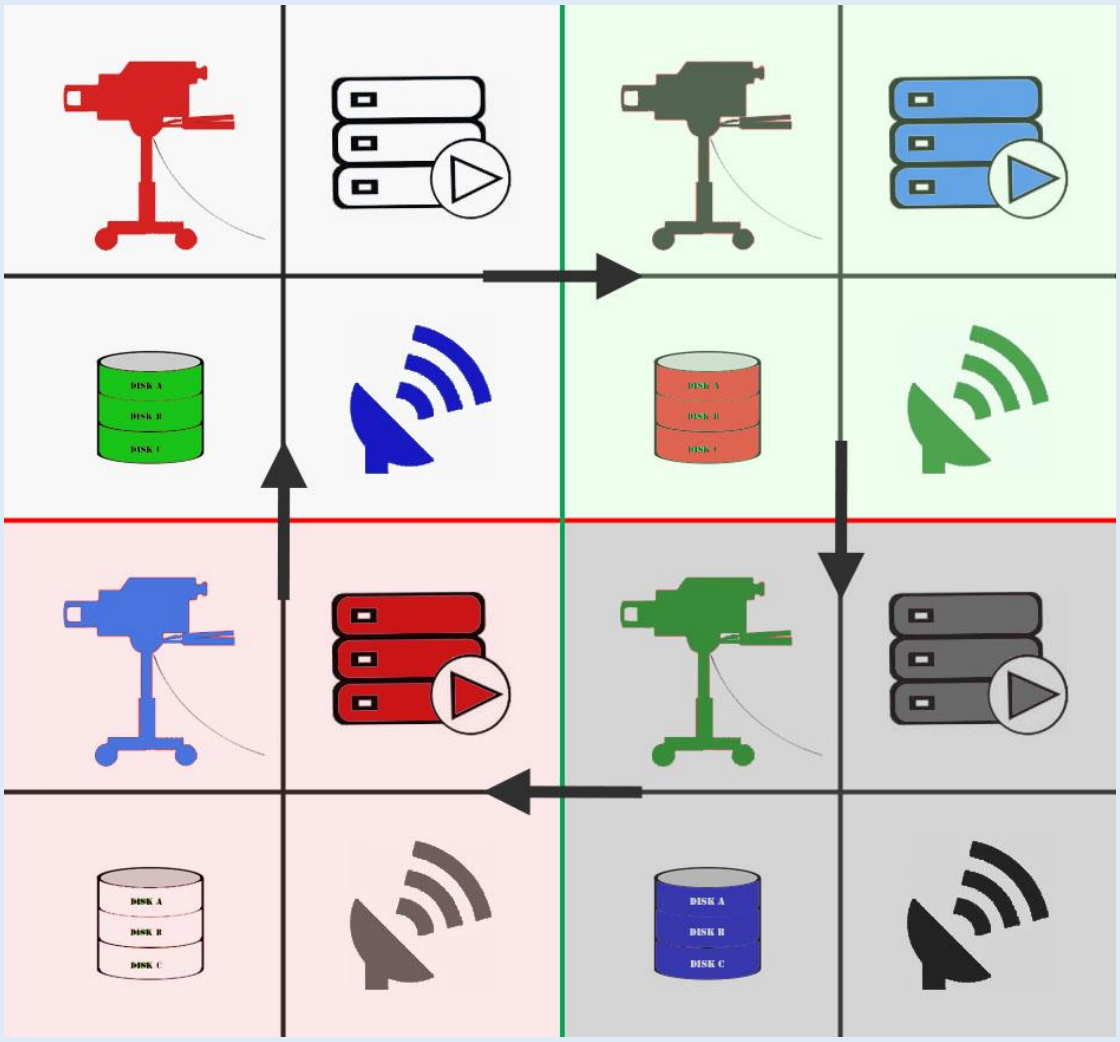
Achieving High Availability in Television Broadcast Systems



Optimizing the Availability of Your Broadcast Systems to Fit Your Business Model

By Jay Bergman
Dec 1, 2021

BACKGROUND

"Are we still on-air?" is a question we all dread when a failure occurs. A typical broadcast system is active or on-air 24 hours a day, 7 days a week. This includes linear playout systems such as broadcast networks and streaming channels as well as on-demand systems where consumers expect instant access to media content. We strive to maximize system uptime while minimizing capital and expense costs. Ideally every piece of equipment, whether it is on-premise or in the cloud, has at least one redundant hot backup but this is usually not feasible. Financially, a broadcast system cannot be treated like corporate e-mail or a web site in the sense that any broadcast downtime can result in lost revenues. If e-mail or internet access becomes inaccessible for a short period of time, it may be an inconvenience and in some cases a web-site service disruption is an opportunity lost for potential sales. Broadcasters must implement all measures possible, within reason, to keep systems on-line as revenue depends on keeping these systems operational.

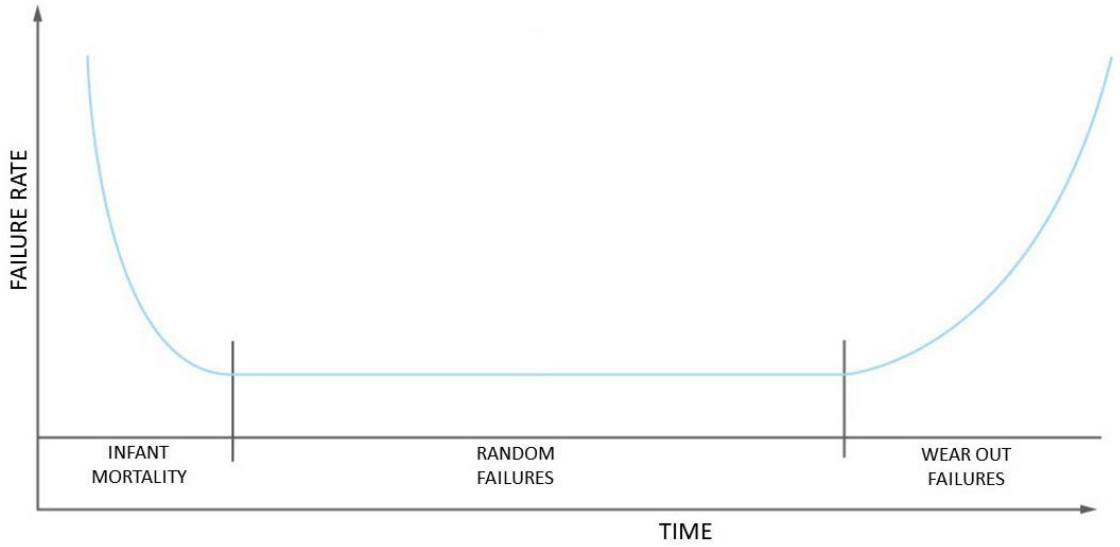
A broadcast system is made up of various technologies, each with its own set of considerations. Most facilities still have traditional audio, video and plant reference systems including routing, processing, mixing and switching equipment. File-based workflows move and process media such as programs, promos or commercials from edit environments to play-to-air systems or from near or off-line storage to video servers. These systems are less susceptible to short outages as their workflows are usually completed well ahead of their scheduled broadcasts. While this paper will not discuss the impact of different levels of impairment to the business in detail, in the examples of these workflows, a component or system failure of several seconds, minutes or even hours could be tolerated depending upon how soon before air-time the failure occurs.

It is therefore the "real-time" systems that need to be protected from failures. This would include the baseband or streaming playout systems, the live contribution, distribution and support systems that keep the content on-line and functioning properly. Managing failures is a very powerful method of reducing downtime which goes hand-in-hand with deciding where to best implement failure mitigation. This paper will demonstrate how redundant components or systems, when implemented correctly, can significantly increase the system availability or uptime.

FAILURE TIMELINE

All equipment will eventually fail. We purchase equipment with the hope that the theoretical failure will occur after that equipment has been replaced with the next generation of hardware but this is unrealistic. Most equipment failures follow what is called a Bathtub Curve in Figure 1. There is an initial higher rate of failure called infant mortality followed by a low (random) failure rate during the useful lifetime followed by a higher rate again as the equipment wears out and reaches its end of life. Random failures occurring during the normal lifetime may be a result of defects, design flaws, high stress, environmental and human factors.

FIGURE 1 – BATHTUB CURVE



REDUNDANCY TECHNIQUES

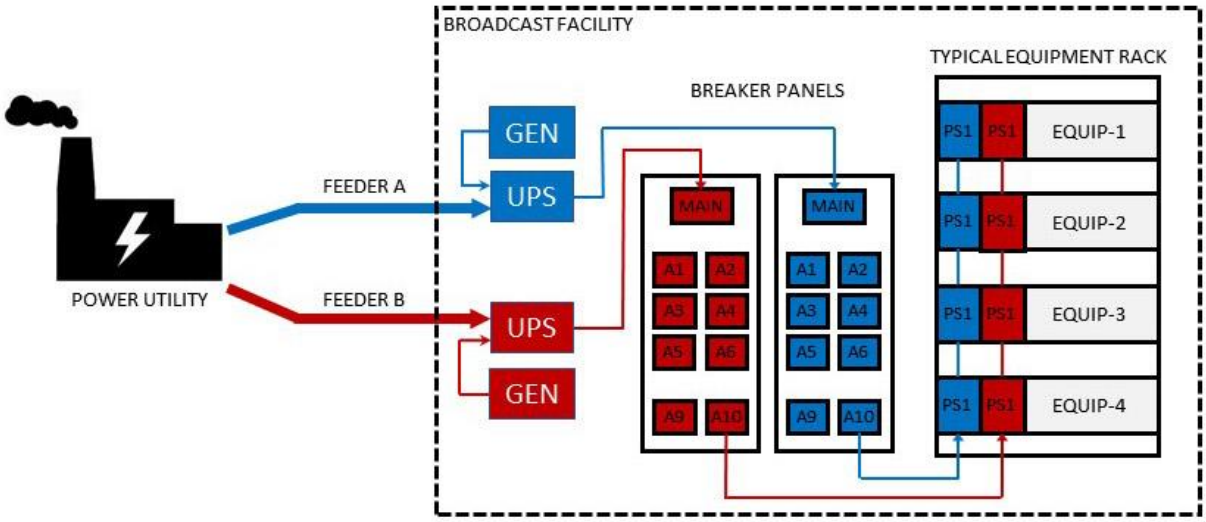
How much statistical uptime can a broadcaster hope to achieve? In theory we would like to get as close to 100% as possible. Without redundancies, a system is only as good as its weakest component. System outages can be placed into two categories – planned and unplanned. While a planned outage is still downtime, it is scheduled and it therefore causes minimal, if any revenue loss. The key to increasing our uptime and mitigating revenue loss from unplanned outages lies in implementing system redundancies. The following are design techniques on this topic which will increase system uptime.

Power System

We often take the infrastructure power for granted until there is an outage. Most resilient systems have several levels of power redundancy. A large building or datacenter may have multiple power feeders from the power utility. One or more of these critical feeders may be backed up by a generator and a UPS which will provide a backup for an outage from the utility. In order to be safe, the health of the generator and the UPS should be regularly checked by taking the utility off-line on a regular schedule and utilizing the generator during overnight periods. The UPS batteries should also be checked and replaced as per the manufacturer's specifications. Power redundancy should also be included in the service that cloud vendors provide together with the use of multiple availability zones.

Additional levels of power redundancy are achieved by wiring each equipment rack with at least two power circuits. Each of these circuits should originate from a different generator/UPS. Most equipment has at least two power supplies where each supply can support the equipment's load on its own. By wiring each piece of equipment with circuits from different power sources, it is protected from a major utility feeder outage, a generator/UPS failure as well as a circuit breaker or power supply failure.

FIGURE 2 – REDUNDANT POWER LAYERS



One of the most prevalent components to fail is the power supply. Utilizing redundant power supplies helps to ensure equipment uptime. For optimal protection, for an equipment rack with two circuits, each circuit should be able to handle the entire rack's equipment load at approximately 70% of the breaker's rating. Therefore, if one circuit fails, the other circuit can take the entire load. Otherwise, the entire equipment rack will lose power should one of its circuits fail. When planning the equipment loads, do not use the power supply rating. This is always rated well above the actual power draw and you will overdesign if you use this as your design metric. The best method is to measure the actual current draw when the equipment is under maximum load and add about 10% for headroom.

Storage

Computer storage can be purchased with many different types of redundancy. This is usually denoted by a level of RAID (Redundant Array of Inexpensive Disks) which can utilize multiple drives allowing for one or more drive failures. While this paper will not go into the details of storage redundancy, some popular examples are shown in Table 1. Each RAID mode requires a minimum number of drives and allows for a maximum number of failures while maintaining the integrity of the data. There are many other standard and non-standard storage redundancy schemas, each with its own benefits and drawbacks. The more protection, the more availability, but each comes with a cost.

TABLE 1 – POPULAR RAID MODES

RAID MODE	DESCRIPTION	MINIMUM DISKS	ALLOWABLE DRIVE FAILURES
RAID 0	DATA STRIPED ACROSS DISKS FOR SPEED	2	0
RAID 1	MIRRORED DISKS	2	1
RAID 5	STRIPED DISKS WITH PARITY	3	1
RAID 6	STRIPED DISKS WITH DUAL PARITY	4	2

Computer NICs

The interface between a computer and the network infrastructure is commonly called the Network Interface Card (NIC). In addition to the computer host (server or workstation), the NIC is also a single point of hardware failure for a software application which communicates or receives data from another network device. Multiple NICs in a single host can provide redundancy when using protocols such as Link Aggregation. This is a method of increasing the reliability and redundancy and providing load balancing by forcing the NICs in a Link Aggregation Group (LAG) to act in parallel. In order to achieve Link Aggregation, all NICs must connect to ports residing on the same logical Ethernet switch. While this would normally appear to be another single point of failure, use of stacked switches acting as a single logical switch would solve this problem.

NTP Time Source

All Windows based systems have a time service built into the domain controller. If setup to synchronize on a domain hierarchy, all computers in the forest will use Active Directory's (AD) domain hierarchy to find a reliable source for synchronizing time via the Network Time Protocol (NTP). In Linux systems, a host can be similarly configured as a central NTP server. For maximum reliability, the time service should be pointed at main and backup NTP beacons. Whether in Windows or Linux, these NTP servers should be setup to revert to the local clock should the NTP source become unavailable. There is broadcast and non-broadcast equipment available which provides the NTP source time for these servers. This equipment is usually locked to the Global Positioning System (GPS) and has a highly stable internal crystal should it need to free-run when failure circumstances cause it to unlock from GPS.

PTP Time Source

With the advent of the ST2110 Professional Media Over Managed IP Networks standards suite, accurate timing has become a critical system component which utilizes the Precision Time Protocol (PTP) multicast reference clock. This provides clock accuracies better then 1 microsecond for audio, video and data synchronization by utilizing a common PTP clock source. Unlike Serial Digital Interface (SDI) and MPEG transport streams, the ST2110 audio and video streams are carried separately on an IP network, can follow different processing workflows and must be synchronized at the destination. This is accomplished by time-stamping the packets during the sampling process. A PTP infrastructure must be designed to very specific constraints and will utilize Grandmaster, Boundary and Transparent clocks to achieve accurate clock distribution on the network. Not all Ethernet switches are capable of supporting PTP. For redundancy, multiple PTP generators (Grandmasters) should be utilized but extra care must be taken in the system design.

Sync Generators

Most traditional broadcast equipment requires synchronization signals such as reference black, sync and/or time code. Utilization of a main and a backup sync generator with an automatic changeover is standard practice. Be careful of the transition when performing a changeover as this discontinuity can affect the normal operation of broadcast equipment. For

use with ST2110 systems (or future-proofing for ST2110) these generators must either be capable of locking to a PTP source (or multiple PTP sources) or serving as the PTP network source (Grandmaster).

Network Architecture

The network architecture plays a major role in the availability of the overall broadcast system. Depending upon how large the system is and how many other systems that it interfaces with, designing a reliable network can be a complex undertaking. Redundancy and resiliency can be incorporated on multiple layers of the network OSI model. This can take the form of redundant wiring, application's use of the transport layer and the implementation of various switching and routing protocols. It is very important to understand that designing and maintaining a network utilizing these protocols is complex and requires a well-trained and experienced staff. The more complex the system, the more difficult it is to recover from failures.

Switch Redundancy

Adding load-balanced, loop-free, layer 2 redundancy in a three-tier switching environment provides multiple paths between edge, aggregation and core switches as well as in two-tier leaf-spine architectures. This delivers protection from link and switch failures. The Spanning Tree Protocol (STP) creates a loop-free network by turning off (blocking) switch ports. STP only allows for single paths but will enable blocked ports upon link failures. Switch vendors replace STP by providing multiple active path functionality via protocols such as Multi-Chassis Link Aggregation Group (MLAG), Split Multi-Link Trunking (SMLT) and Virtual Switching System (VSS), some of which are vendor proprietary. They also greatly reduce latency when recovering from failures.

Router Redundancy

There are multiple methods to provide layer 2 router redundancy depending upon the selected vendor and router. One such protocol is the Virtual Router Redundancy Protocol (VRRP) which is an election protocol that dynamically assigns the virtual IP to one of the routers in a VRRP group. Each network uses this virtual IP address as its gateway. If the active router fails, the protocol routes all traffic to another router in the group. This protocol may be setup for load balancing depending upon the selected vendor and hardware.

Routing Redundancy

Layer 3 routing provides paths to other networks. As static routing does not protect against routing changes and failures, standardized routing protocols such as Open Shortest Path First (OSPF) and Border Gateway Protocol (BGP) utilize algorithms to provide the best path to another network. OSPF is an Interior Gateway Protocol (IGP) operating within a single autonomous system while BGP is an Exterior Gateway Protocol (EGP) exchanging routing information among autonomous systems. OSPF is often used within a corporate environment. BGP is the Internet routing protocol and is often used as the protocol within a corporate

environment interfacing between independent departments. These routing protocols automatically adjust paths due to changing network conditions and therefore provide system redundancies.

Application Redundancy

Applications can be designed with a main/backup failure mode. This needs to be programmed into the application with specific services such as utilizing heartbeat communications between the servers. Virtualized applications or modern applications running as microservices in containerized environments are designed for load balancing, high loads and for quick recoveries. Cloud environments provide virtually unlimited resources for high availability requirements by manually or automatically spinning up processors or containers when required.

DNS Failover

In IT systems where computer hosts communicate with other hosts within and across networks, utilizing the Domain Name System (DNS) can be very useful. Often IP addresses are hard-coded into applications when systems are configured. Loss of communication may cause a host to failover to a backup by using a secondary hard-coded IP address. A safer approach would use the destination host name and a DNS server to resolve the IP address. Among other services, a DNS server can provide a host failover to multiple backup IP addresses using health checks and can be utilized as a round-robin load-balancer. Note that health checks cannot provide instantaneous failovers as DNS records are cached for the length of the Time-To-Live (TTL) property and cannot be set to zero. DNS providers and products also do not all provide the same services.

Geographical Redundancy

Once again, the more redundancy in a system, the more uptime is achieved but there is a price to pay for this. The ultimate redundancy is utilizing one or more entire duplicate systems running as hot standbys in multiple locations. This protects the owner from major location-based system outages and can be achieved with on-premise systems, cloud-based systems or a combination of both. A compromise, for example, would utilize an on-premise system backed up with a cloud-based system in a warm or cold standby configuration which can be spun-up when required. This would be more cost efficient but there would be a delay in bringing the backup on-line and therefore, utilizing this method must be a business decision.

A multi-cloud solution is also a viable option which utilizes multiple cloud vendors. This can mitigate for a specific cloud vendor system, hardware or software failure but it requires staff knowledge of the architectures, configurations and operations of multiple cloud environments.

TABLE-2 – AVAILABILITY CONFIGURATIONS

STANDBY MODE	AVAILABILITY	FAILOVER TIME
Cold	▪ Upon failure, standby must be powered up, configured and loaded with the latest software, metadata and media ▪ Requires a manual cutover	Hours
Warm	▪ Standby is already powered up, configured and ready for service ▪ May require software updates and synchronization with the latest metadata and media ▪ Requires a manual cutover	Minutes
Hot	▪ Standby is already running and synchronized with the current metadata and media ▪ Automatic cutover upon failure of primary	Immediate

IT Security

While not covered in this paper, IT security can play a major role in avoiding system outages by preventing malware, viruses, ransomware and hackers from taking some or all of the components and systems off-line. These systems, however, must be scheduled out of service in order to implement the required processes such as patching and scanning. IT security is a layered process utilizing best practices such as authentication, authorization, logging, encryption, scanning and keeping up with security patch updates.

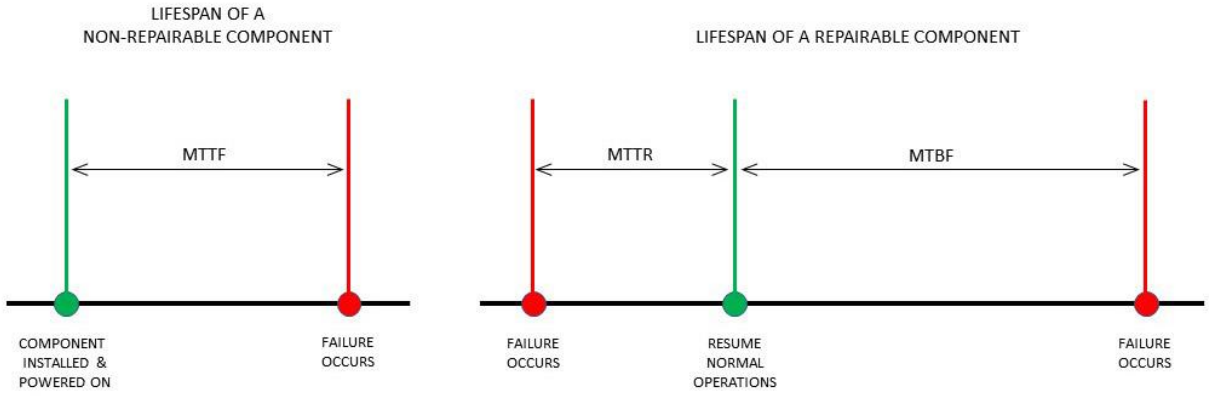
FAILURE METRICS

The three basic metrics measuring failure statistics are: Mean Time to Failure (MTTF), Mean Time Between Failures (MTBF) and Mean Time to Repair (MTTR). MTTF represents the length of time a component is expected to last in operation until it needs to be replaced. It attempts to estimate the average lifespan of a component that cannot be repaired. MTBF is a measure of the reliability of a component and MTTR can tell us the efficiency of the maintenance staff. For example, the MTTR for software would be the time it takes to detect the failure plus the reboot time. For a hardware component this would be the time it takes to detect and diagnose the failure, the administrative and logistic delay in receiving or retrieving a spare part and the time required to repair, test and bring the system back on line. A complex system can also delay the troubleshooting process.

Interestingly, the advent of hot-swappable components such as power supplies, fans and circuit boards, significantly decreased the MTTR metric. Repairing equipment by simply swapping out a failed component takes a significantly shorter time than un-racking the equipment, opening up the chassis, replacing the failed component and reversing the process. After system testing, if it is determined that the wrong component was replaced, the above process must be repeated.

Is it worth keeping spare parts and repairing the equipment in-house or would it be more expedient to purchase an on-site support contract with the vendor? While a four-hour support contract would be expensive, would it provide a lower MTTR and would it make sense for the business? This is a business decision that must be considered before making any purchases as it also has ramifications on the required on-site labor and training.

FIGURE 3 – COMPONENT LIFESPAN TIMELINE



TESTING

It is very important to test redundant systems on a regular basis. If a backup system or component has not been put into service for several years, what are the odds that it will function as required when brought on-line? Will it have the latest security patches, passwords, code, operating system updates or files that the primary system has? Were its ports blocked after a recent manual or automated firewall audit? An untested system can significantly increase the MTTR if it always needs to be repaired or updated prior to usage. A good method of maintaining a backup system is to use it online as the primary on a regular basis.

MONITORING

There are other methods which can reduce your downtime which are less costly than building complete multi-level redundant systems. When deciding between multiple products from various vendors, consider the real-time information provided to the staff in a dashboard or via an email or a text alert which would allow for quicker repair times and therefore lower MTTRs. For example, an alert for a failed redundant hot-swappable power supply would allow for a repair with zero downtime before the remaining power supply fails. An alert for an impaired process or a failed application in a workflow chain would allow for a reboot or repair possibly preventing a catastrophic event. Most on-premise and cloud-based virtual systems can automate this functionality when appropriately configured. Many products also provide machine learning algorithms which can detect component or system failures or congestion before they happen given historical learned behaviors.

AVAILABILITY

Availability can be defined as the degree to which a system is in an operable state when called upon to perform a task or function. There are several methods used to denote availability, one of which refers to this as a percentage. An availability of 99.999% (or .99999)

is also commonly referred to as five nines and is based on an annualized time frame. For example, a device would have an availability of 0.99999 x 365 days/year = 364.99635 days or a downtime of 365 – 364.99635 = 0.00365 days = 5.26 minutes per year. This does not mean that the device will be out of service for 5.26 minutes per year. Statistically it will be in a failed state in some manner for a total of about 5 minutes per year but it may not fail at all or it may fail for a much longer aggregate period of time. In the broadcast business, failing at certain times may be much more costly than at other times. For example, a failure during a sports playoff commercial is orders of magnitude more costly than a failure during a typical overnight period. Advertising rates vary depending upon the content, the time of the day and the day of the week. When you are considering an availability of five nines, think about the impact of a failure during a Superbowl commercial.

There are various terms for availability. A relevant one to consider here is steady state availability. From the Bathtub curve diagram in Figure 1 in a previous section, we can deduce that the availability will initially be reduced until we get past infant mortality. There will also be a learning curve for the maintenance staff. What specializations, amount of personnel and training will be required? Understanding how new components and new systems function and fail as well as what spare parts to stock locally will also have a negative affect on the initial availability. Steady state availability reflects the long-term availability after the system stabilizes.

TABLE 3 – LEVELS OF AVAILABILITY

AVAILABILITY	LEVEL	DOWNTIME (PER YEAR)
99.9999%	6 nines	32 seconds
99.999%	5 nines	6 minutes
99.99%	4 nines	53 minutes
99.9%	3 nines	9 hours
99%	2 nines	4 days
90%	1 nine	37 days

If it is not already provided by the manufacturer, once you know the component MTBF metrics, you can calculate the availability as follows:

$$Availability = \frac{MTBF}{(MTBF + MTTR)} x100\%$$

Calculating Availability

When considering on-premise systems, a manufacturer usually provides availability statistics for their specific component as an MTBF. The repair data would be customer driven. Statistics for a component such as storage or compute in a cloud environment can be reported as availability because the uptime and repair time are both under the cloud provider's control.

This metric is often difficult to obtain and may be included in an SLA commitment.

The equations for calculating a system's availability are similar to those used in electronic circuit calculations. A system can be reduced to serial and parallel components regardless of what the components consist of.

A simple serial configuration is when two or more components are required to be available at the same time for the system to be available. If either component fails, the system is unavailable.

The general formula for adding n serial components is the product of all of the availabilities:

$$\text{Serial Availability} = \prod_{i=1}^{n} \text{Availability}_i$$

A simple parallel configuration has redundant components. If one component fails, the system will still be available.

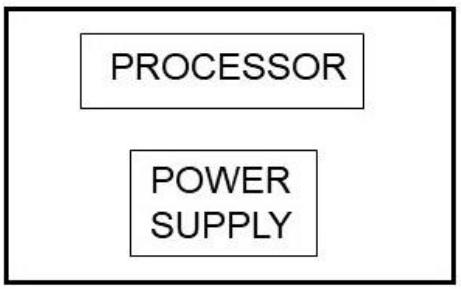
The general formula for adding n parallel components is:

$$\text{Parallel Availability} = 1 - \prod_{i=1}^{n} (1 - \text{Availability}_i)$$

Calculating System Availability

The manufacturers (or service providers) derive the MTBF (or availability) from the expected failure rate of each component within the architecture of the system. A simple serial availability example would be the following system in Figure 4 which consists of a processor and a power supply where the system becomes unavailable if either component fails.

FIGURE 4 – SYSTEM 1



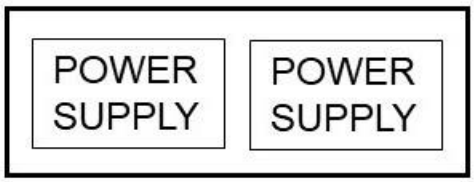
Assuming a processor availability of 0.999933 and a power supply availability of 0.999950, as these two parts are both necessary and both must be functional in order for the equipment to operate correctly, they have a serial relationship.

Availability = 0.999933 x 0.999950 = 0.9999

Note that the system availability is less than the availability of each of its components. This is because the system availability is dependent upon both components being active at the same time. This system is only as available as its weakest, or least available component.

A simple parallel availability example system would be a redundant power supply configuration which, in Figure 5, is made up of two parallel power supplies where the power supply system remains available if either component fails.

FIGURE 5 – SYSTEM 2



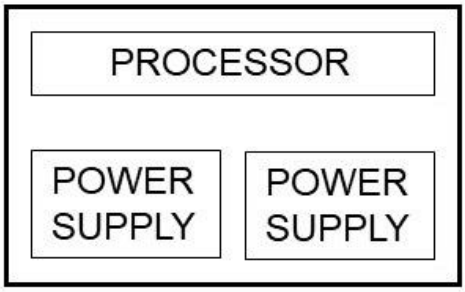
Availability = 1 – ((1 - 0.999950) x (1 - 0.999950)) = 0.9999999975

Note that the system availability is greater than the availability of each of its components.

Hybrid Configurations

The next example in Figure 6 shows the availability when adding a redundant power supply from Figure 5 to the first example in Figure 4. Here the loss of a single power supply will not cause the system to fail but a failure of the processor will cause a system failure. This is a hybrid configuration where the processor and the power supplies are in a serial configuration and the power supply is itself in a redundant parallel configuration.

FIGURE 6 – SYSTEM WITH 2 POWER SUPPLIES



System Availability = Processor Availability x Redundant Power Supply Availability
Power Supply Availability = 1 – ((1 - 0.999950) x (1 - 0.999950)) = 0.9999999975

System Availability = 0.999933 x 0.9999999975 = 0.99993

Note: The above 0.999950 and 0.999933 availabilities were taken from the previous examples

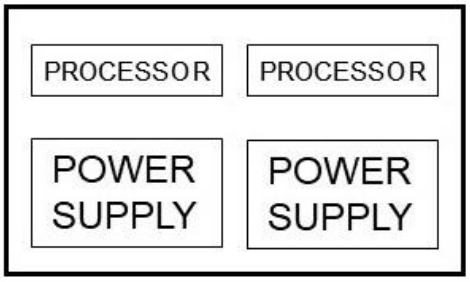
Notice that this result is slightly higher than the system availability without the redundant

power supplies (0.9999) from a previous example. The processor is still a single point of failure and the overall system availability cannot be greater than each of its serial components. Here, the overall availability seems to be approaching the availability of the processor as the power supply becomes more available.

The Effect of Additional Redundancy on Availability

The next example in Figure 7 shows the additional reliability when using a configuration with a redundant power supply and a redundant processor.

FIGURE 7 – SYSTEM WITH 2 POWER SUPPLIES AND 2 PROCESSORS



Parallel Calculations:
Redundant Power Supplies
Availability = 1 – ((1 - 0.999950) x (1 - 0.999950)) = 0.9999999975

Redundant Processors
Availability = 1 – ((1 - 0.999933) x (1 - 0.999933)) = 0.9999999955

Serial Calculation
Redundant Power Supplies with Redundant Processors
Power supply availability = 0.9999999975 x 0.9999999955 = 0.999999993

If we calculate the downtime per year as we did previously, we can compare the results of the additional system redundancies. While adding a second power supply seemed to make a big difference, removing any single points of failure has a dramatic effect on the overall system availability.

TABLE 4 – REDUNDANT CONFIGURATION AVAILABILITY COMPARISON

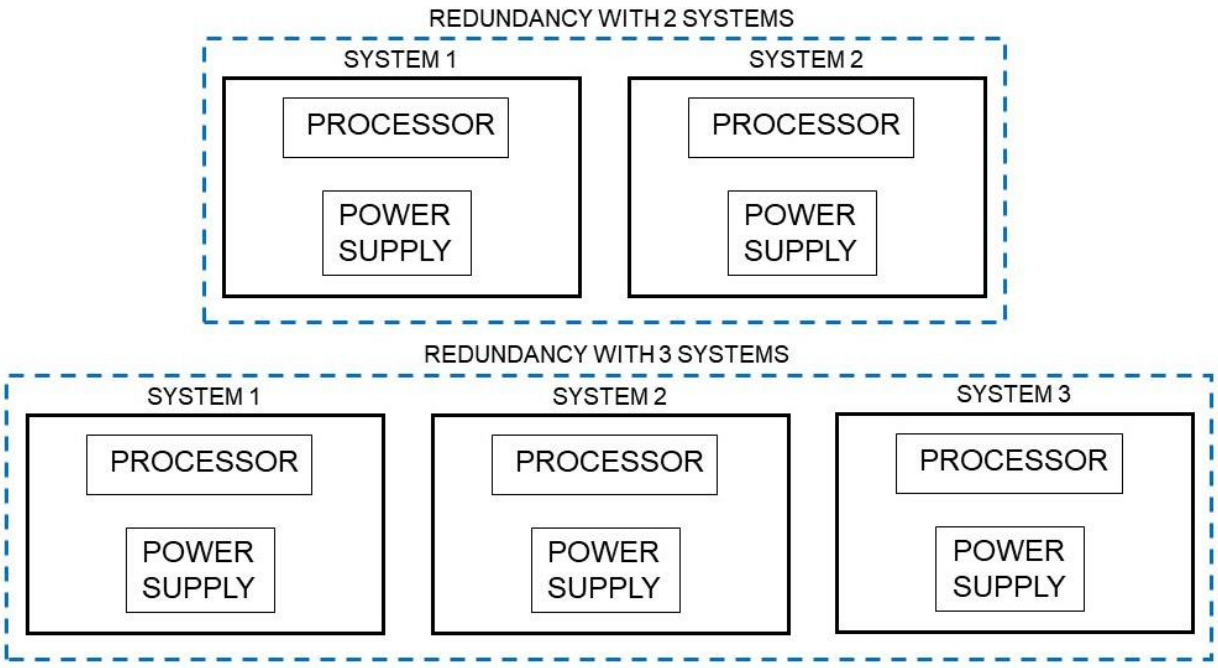
CONFIGURATION	AVAILABILITY	DOWNTIME (PER YEAR)
One Processor + One Power Supply	99.99%	53 minutes
One Processor + Two Power Supplies	99.993%	37 minutes
Two Processors + Two Power Supplies	99.9999993%	0.2 seconds

The Power of Parallel Systems

Another interesting example of the power of redundancy is calculating the availability of multiple low-cost, less-available systems in parallel. Similar to the concept of RAID storage where we can lose one or more inexpensive drives in a storage system and remain on-line, here we use multiple low-cost systems to achieve higher availability.

In Figure 8 below, we again use the example system from Figure 4 above with no redundant components but here we use multiple systems in parallel.

FIGURE 8 – LOWER AVAILABLE SYSTEMS IN PARALLEL



From the previous example with a single processor and a single power supply, we calculated the system availability to be 0.9999 but, in this example, we will assume each is a very low-cost system with an availability of 0.99 or 3+ days of outage per year. While the systems in Figure 8 show sub-components, these should be considered to be generic systems. In the top example in Figure 8, we can calculate the availability for two of these systems in parallel:

Availability = 1- (1-0.99)x(1-0.99) = 0.9999

By utilizing two low availability systems in parallel we were able to decrease our expected downtime from 3+ days per year to under an hour per year. We can also calculate the increased availability of an additional redundant system as shown in the bottom example in Figure 8 as follows:

Availability = 1- (1-0.99)x(1-0.99) x(1-0.99) = 0.999999

We now have an availability of six nines which brings our statistical downtime to 32 sec per year. This example of significantly reducing downtime with additional redundancy can be

summarized in Table 5 below. Keep in mind that this assumes an instant failover to a redundant system.

TABLE 5 - PARALLEL SYSTEM AVAILABILITY COMPARISON

CONFIGURATION	AVAILABILITY	DOWNTIME (PER YEAR)
Single Generic System	99%	4 days
Two Parallel Generic Systems	99.99%	53 minutes
Three Parallel Generic Systems	99.9999%	32 seconds

SUMMARY

We have seen how adding redundancies into a system will increase its statistical availability and therefore decrease its potential downtime. Increasing availability becomes a business decision as there are costs associated with higher availabilities. For example, in the long-term, is the cost of implementing and maintaining a highly available hot standby system worth the cost of an outage? Would a much less expensive warm or cold standby be a more reasonable alternative? Is a four-hour support contract worth the price or should all spare parts be kept locally? What is the cost to implement and maintain redundant power systems, each backed up by a generator and a UPS? If you do implement a hot standby system at another site, do you also need to add the redundant power at each site or would this be overkill for your business model?

Combining traditional and modern methods of redundancy along with sufficient monitoring and maintenance can provide the availability required for a business. Many of the available tools have been mentioned in this paper. We saw how availability can be calculated but this should be used as a general tool to provide an insight into the cost-benefit calculations for the business as every business has a different model.