

High Availability Technical Primer

William J. Bender and Abhinav Joshi
Project Performance Corporation
2121 Crystal Drive, Suite 701
Arlington, VA 22202
703-920-0033

Overview

The use of computers in our daily business and personal lives is pervasive. We literally cannot drive a car, travel, shop at any store, or perform our jobs without the use of computers. In many cases, there are no manual procedures available to replace what computers do for us. Email, Internet Web, and instant messaging have replaced US Postal Mail as our preferred method to communicate.

When these systems and the applications are not available, due to technical problems, we suffer loss of productivity at work and inconvenience in our personal lives. It is not difficult to calculate the real costs incurred when critical business systems are down. More serious consequences occur when critical systems such as traffic control, medical life support, or Health services systems are not functioning.

Whatever the situation, when an application or computer system is supposed to be running and isn't, it impacts productivity and it costs money. When an application becomes unavailable, the work that it was doing simply stops. At best, such an outage simply results in lost productivity - the application will be up and running some time later, and the work will be completed later. More serious consequences can occur through safety, legal actions, fines or simply negative publicity.

The impact of downtime will vary from business to business and within a business from application to application. Global businesses or Internet-enabled businesses may need order-entry applications to be continuously available. Customer support staff may need access to data 24 hours a day with a response time that is measurable in seconds. On the other hand, some departments may need access during normal business hours only and can tolerate somewhat sluggish responses. The cost of the measures taken to ensure that systems keep running must be aligned with the risk (both human safety and financial) and the associated benefits.

Availability, High Availability, and Fault Tolerance: What do these terms mean?

Although incorrect, these terms are sometimes used interchangeably. We need to gain a clear understanding of each before we can decide which is appropriate for our various computer systems. Each carries a specific price tag with increasing cost, and possibly decreasing benefit when misapplied.

Availability is the percentage of time that a system operates during its intended duty cycle. For example, if a given system is expected to be functional for 8 hours per day, then availability is measured as a percentage of those eight hours. If a system is non-functional outside this period, it is not counted against the “Availability Metric.”

High Availability attempts to specify an amount of time as a percentage of the intended duty cycle that a system must be functional. For example, if we specify an availability metric as “Five Nines,” it is understood to mean that the system should be functional for 99.999% of the desired duty cycle. Refer to the following table for examples of various levels of availability and associated allowable downtime per week/year assuming a 24 hour per day duty cycle.

Percentage Uptime	Percentage Downtime	Downtime Per Year	Downtime Per Week
98%	2%	7.3 days	3.37 hours
99%	1%	3.65 days	1.68 hours
99.8%	0.2%	17.5 hours	20.2 minutes
99.9%	0.1%	8.75 hours	10.1 minutes
99.99%	0.01%	52.5 minutes	1 minute
99.999%	0.001%	5.25 minutes	6 seconds

Table 1: Measuring Availability

There are two factors that determine system availability:

1. The first factor is the reliability of the individual components that comprise the system. These components include server hardware, server operating system and the application itself. Other components may include data storage devices, network access devices, databases, file systems and the data center infrastructure.
2. The second factor is the time it takes for the application to be restored once a failure has occurred. The amount of time it takes to bring an application online again is dependent on the component that failed. If the application itself has failed, all that may be required for recovery is to simply restart the application.

If, on the other hand, the application has failed due to a hardware failure, recovery could involve:

- Notifying the service provider of the failure;
- Waiting for the arrival of the service technician;
- Determining the failed component;
- Replacing of the failed component;
- Rebooting the operating system;
- Recovering the file system;
- Recovering the database;
- Restarting the networking software;
- Restarting the application.

In most cases, when system vendors make availability guarantees, all they are warranting is the capability of the server to deliver an operating system prompt. Availability, as the application user perceives it, will always be lower. High availability must encompass all components required to provide a service. Certain components of a service may be highly available, however, other system components may not support the same level of high availability, therefore, overall availability is reduced.

High Availability vs. Fault Tolerance

Now that we have a clear understanding of high availability, we can contrast this concept with fault tolerance. Fault tolerance differs from high availability by providing additional resources that allow an application to continue functioning after a component failure without interruption. Many of the high-availability solutions on the market today actually provide fault tolerance for a particular application component. Disk mirroring, where there are two disk drives with identical copies of the data, is an example of a fault-tolerant component. If one of the disk drives fail, there is another copy of the data that is instantly available so the application can continue execution.

A fully fault-tolerant solution requires that all the resources the application is dependent on be replicated including the application process itself. This requires an independent processor (not part of the same computer) and a copy of the memory that the application uses. In the worst case failure scenario, one in which the processor or memory fails, the replicated version of the application continues to execute. Other failures simply require the application to use the alternate resources (disks, communications devices). As a result of this complete hardware and process replication, fault-tolerant systems are significantly more expensive than highly available systems.

A fault-tolerant system would be used in a situation where no downtime can be tolerated at all, such as an air-traffic-control system, an emergency-response system (9-1-1) or financial-trading systems. It is important to note that in order for a

system to be fault tolerant, all components and services of that system must be fault tolerant.

Justifying High Availability Solutions

Now that we understand the difference between high availability and fault tolerance, we are able to make informed decisions for a given business need. The first step in this process is to decide if the system is in the category of mission critical where no downtime can be tolerated. Good examples are:

- air-traffic control;
- life support;
- life safety; or
- financial-trading system.

If it is a mission critical system, then all the components for that system must be configured for fault tolerance; there must be a duplicate of each component. If the systems are not in the “mission critical” category, maximum return on investment (ROI) can be achieved through engineering a lower level of availability that will satisfy business needs.

Because high availability systems are expensive, each candidate system must be examined to determine the business impact and loss of revenue when it is not available. This examination process will help determine the “level” of availability to design for a given system. For example, if a system supports 6000 users, and renders all users idle if it is not available, and results in a \$1 million/hour revenue loss, it would be a good candidate for a high availability solution.

Conversely, if a system supports 25 users and the business process provided by this system can be continued via alternative automated or manual systems, it would be difficult to justify the cost of building and supporting a high availability configuration.

Designing For Downtime

There are two ways to categorize system downtime. Some downtime are results from a system failure and others from scheduled outages. Scheduled outages, such as those for repair and upgrades that have minimal impact on the business, are considered maintenance. For many applications, availability during business hours is required, but some downtime during non-business hours is acceptable.

Applications deployed by global businesses, as well as Internet-enabled applications, must be up and running 24 hours a day, 7 days a week. These applications are usually “public facing” and although an outage would not create a life threatening or catastrophic financial impact, they are critical for public access to the business resources and therefore justify a higher level of availability.

All systems will require maintenance at some point. If management does not plan for system maintenance, the system will pick the time and duration for an outage! It is up to the system designer to understand the business need and design the system to allow for planned downtime, therefore minimizing the risk of a system failure.

Causes of Downtime

To reduce downtime, we need to develop a detailed understanding of the causes. Figure 1.0 illustrates that there are three major causes of unplanned downtime:

Most Common Causes Of Unplanned Downtime¹

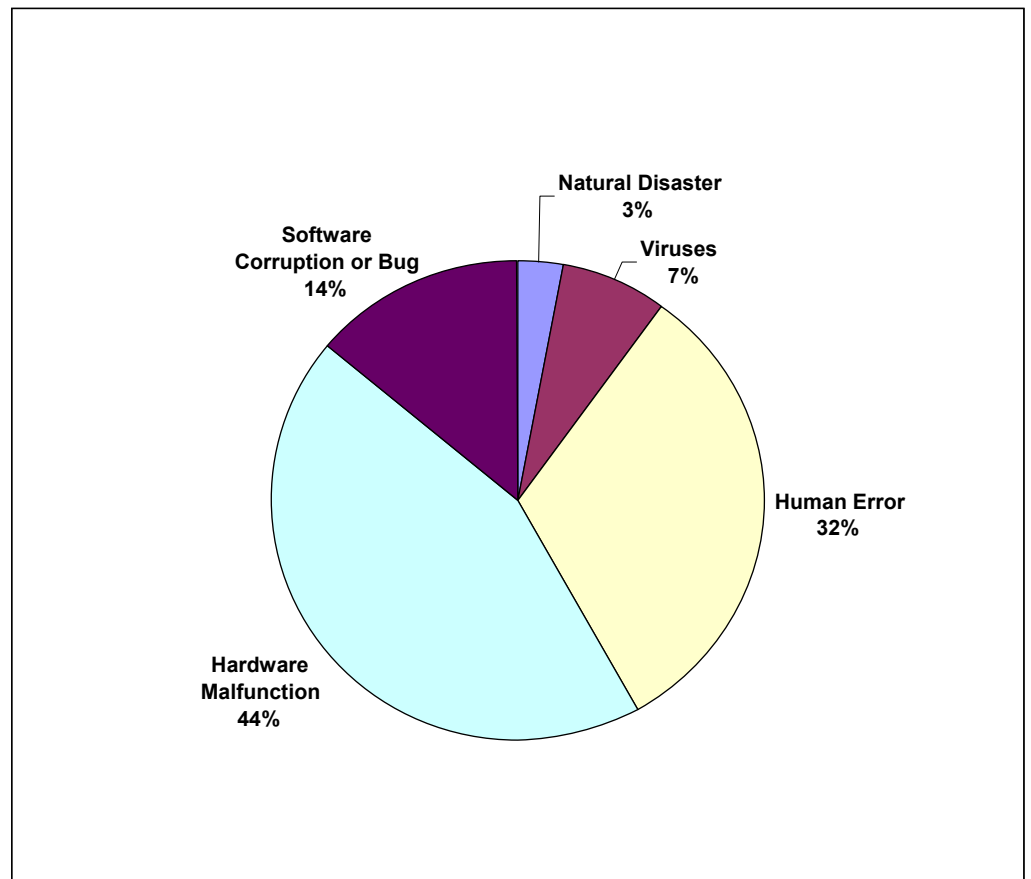


Figure 1.0

Most benefit can be derived by working to mitigate the failures caused by the three largest areas. Let us now take a closer look at each of these causes and then discuss how they can be addressed for any large enterprise.

¹ Source: CNT: Storage Networking Solutions, "Achieving High Availability Objectives," www.cnt.com, 2003.

Downtime Cause 1: Hardware Failure

Hardware failure can be addressed at three levels: Component, System, and IT infrastructure. At the component level we have devices like Network Interface Card (NIC), Host Bus Adapter (HBA), Disk, Central Processing Unit (CPU), Memory, Power Supplies, and all other parts that go into making up a server, switch, router, tape library or disk storage system. All individual components (including driver and OS software) need to be carefully selected to make sure that they will work in a given system and interact properly with other external systems. A good example of this is an HBA installed in a server where the HBA is certified by the server vendor and the Storage Area Network (SAN) switch vendor to operate properly with their respective hardware. These components should be selected by the engineering staff for the systems used to build an enterprise IT architecture. Reducing downtime requires careful selection of each component and in many cases using redundant components and systems. Each approach adds complexity and cost.

Server Hardware

Most of the application and database servers incorporate high availability features into the basic box commonly referred to as N+1 components. Multiple power supplies are installed in the server so that if a single power supply fails, the remaining power supplies can continue to carry the load. In high-end UNIX servers, a memory or CPU failure may cause the server to crash, but it will reboot and isolate the failed component. Application software will need to be restarted, but the outage will normally be no longer than the time it takes to reboot and start the applications. If more reliability is required, servers can be configured in redundant clusters of two or more nodes. Cluster technology supports two main architectures that provide availability. In an active-active configuration, two or more nodes in a cluster are actively supporting processing. Loss of any node in the cluster typically results in less than one minute of downtime as processes and users are migrated to the surviving nodes. Alternatively, an active-passive configuration provides a standby server that is available to assume the role of the primary server in the event of a failure. Downtime in active-passive configurations is typically in the 15 to 30 minute timeframe. Clusters are usually deployed for critical applications such as exchange mail, and database servers.

Storage Hardware

Consolidated SAN based disk storage systems must be deployed for all the mission critical applications and systems. The important features for any storage hardware that make it the best choice for designing reliable, highly available systems is as follows:

- **RAID** - individual disks are configured to duplicate data, this prevents loss of data should any one disk fail.
- **N+1 Power** - all storage units are configured with redundant power supplies.
- **UPS** - all SAN storage devices incorporate an internal backup power supply to allow flushing of cache in the event of an external power failure.
- **Redundant Data Paths** - all internal components required to manage data flow are redundant. Should a fault occur in any component, the system software will “offline” that component and use its alternate.
- **Phone Home** - all SAN based disk storage systems incorporate internal self-diagnosis software that has the ability to proactively predict failure in any of the critical subsystems. Upon detection, the unit will phone the support center. The support center can diagnose problems remotely or dispatch a service engineer to perform a repair. This often occurs before the onsite personnel even notice a problem.

Backup Hardware

The fundamentals of tiered storage and backup system are to establish multiple classes of storage with different tradeoffs in terms of costs, performance, and availability in order to optimize the placement of data. Critical data should be backed up to SAN based consolidated disk storage first, then copied to a tape library. Tapes are then shipped offsite for safe keeping and archival. Data that is backed up to disk can be quickly recovered when necessary. All disk storage units and tape libraries are connected via a SAN to allow moving data quickly between servers or storage.

Major storage and tape library vendors are addressing this need with new products. Future enhancement shall include virtualization of the disk and tape drives in the library. Although a server may think it is backing up to tape, it may actually be backing up to a disk. The production server or backup server need not be involved in the operation.

Database High Availability

Many of today’s enterprise applications rely on commercial databases, therefore it is appropriate to review the current “best practice” regarding their design and configuration. Although there are many similarities between database products, we base our discussion on Oracle.

We discussed earlier that hardware failure is the number one cause of downtime. There are two main hardware concerns with respect to maintaining a highly available database environment: server high availability and storage availability. For Oracle databases, two complementary technologies support high availability:

- Oracle Real Application Clusters (RAC)
- Oracle Data Guard.

Oracle RAC technology is based on an active-active server clustering model. RAC supports high availability in the event of a failure of a server, operating system, or database software on any node of the cluster. Users on surviving nodes see little or no downtime. Users on the failed node can be automatically connected to any of the remaining nodes to resume processing, typically in under a minute. RAC can also be used to avoid downtime in the event of planned or unplanned server maintenance.

Oracle Data Guard is used to provide high availability in the event of a catastrophic loss or extended unavailability of a system. Data Guard maintains one or more transactionally consistent copy of a database on a separate storage system, possibly at a remote location. Data Guard copies are known as standby databases and can be configured for little or no data loss. Standby databases provide additional protection over conventional disk mirroring based solutions since they do not propagate corruptions that can occur at the operating system or disk subsystem level. Standby databases can also be used to off-load backups from production systems, support read only reporting applications, and to avoid downtime in the event of planned maintenance.

An additional Oracle feature is the Flashback Database technology. Recovery from corruption introduced by human error currently involves time-consuming database operations and often additional manual steps to restore the state of all or a subset of a database to an earlier point in time. Recovery from human error is usually measured as a function of the time to create the error plus the time to restore a database from backup. Oracle's Flashback Database technology supports "rollback" to the state of a database, database table, or individual database row to a prior point in time. Flashback will reduce the time required to recover from human error to a function of the time to create the error. This can be thought of as a sophisticated "undo" feature.

Data Center Infrastructure

Although not traditionally thought of as part of the hardware failure cause of downtime, the data center infrastructure is a key design element for high availability. The data center houses all of the various support elements of an organization's IT infrastructure and therefore must be designed accordingly to meet the high availability goals. Let's briefly examine the major components:

- **Power**
The data center should provide a robust and redundant power system for each server, network switch, router, and all other equipment required to keep the IT infrastructure running. A high availability solution requires the power to be dual fed from two separate substations. The power system

should incorporate a central UPS and a diesel generator. Best practice design includes multiple Power Distribution Units (PDUs) to supply power to the equipment racks, so that each rack is dual fed. In the event of an external power failure, the generator can be online within 20 seconds and carry the entire data center load.

- **Networking**

Communications between servers, consolidated disk storage, tape libraries, LAN and WAN links should be completely redundant within the data center. The Storage Area Network should incorporate a “dual fabric” design where dual HBAs are installed in each server, and these are connected to two separate SAN switches. A third HBA should be installed if SAN based tape libraries will be connected. The servers should incorporate dual network interface cards (NIC) and each connection should be attached to separate IP switches. The switches in turn should connect to primary and secondary routers. Connection to the external internet should also be via two separate WAN links.

- **Security**

Access to the data center should be controlled via TV monitor, locked door, and full time guard. Visitors must be signed in and escorted. Employees must have a special data center badge to present to the guard who will grant access. A logbook should be kept and all persons entering must sign in and out.

- **Fire and Smoke Detection**

All data centers are required by National Fire Protection Association Code (NFPA) to employ a fire and smoke detection system. These systems usually use both ionization and smoke detectors in a cross zoned configuration. Two detectors from different zones must alert in order for an alarm condition to exist. Where a raised floor is present, detectors must be placed below the floor as well as above. When the alarm system detects smoke or fire, it will automatically turn off power to all equipment in the data center. For fire suppression, most data centers employ overhead sprinklers. Normally the detection is far more sensitive than the heat activated sprinkler heads, therefore power will be off before any water is released. The design of these systems is usually handled by a firm specializing in data center protection.²

- **Cooling and Humidity Control**

Best practice design dictates that all data center cooling be redundant. No data center can operate for very long without adequate cooling. The

² For additional information, refer to National Fire Protection Association publication NFPA 75: Standard for the Protection of Electronic Computer/Data Processing Equipment, 1999 Edition.

temperature should be monitored and alarmed when a predetermined limit is reached. Modern cooling units also have the ability to control humidity. Cooling removes a tremendous amount of moisture from the air, and abnormally low humidity levels can lead to the generation of static electricity. These static charges are reduced by the use of conductive and grounded raised flooring, but it is best to control the source as much as possible. This means keeping the relative humidity (RH) within acceptable limits of 45 to 50% RH.

- **Monitoring**

Critical data centers are monitored by a NOC (Network Operations Center) which may be in-house or outsourced to a third party. The NOC is the first place outages are realized and the starting point for corrective action. NOCs are generally staffed during the data center's hours of operations. In 24 hour a day, 7 days a week data centers, the NOC is an around the clock department. Equipment monitoring devices will advise the NOC of problems such as overheating, equipment outages, and component failure via a set of triggers that can be configured on the equipment or via a third party monitoring software. This software can be configured to notify responsible personnel when an abnormal condition exists.

Downtime Cause 2: Human Error

System Management Tools

Recently, there has been an accelerated effort to address the underlying causes of human error and to mitigate its effect by reducing or eliminating the complexity of the information system. The use of integrated management tool suites with consistent operator interfaces reduce the amount of information that an operator needs to master and thus help to eliminate operator errors that can cause downtime. Web-based management tools can provide professional administrative expertise at remote locations where such expertise was previously unaffordable and unavailable. High-end products use real-time performance and usage data to adjust system parameters to ensure that performance bottlenecks or resource shortages do not cause outages.

Staff Training

Organizations should continuously train their staff on the latest versions of systems and software used in the IT infrastructure. Vendors usually release new versions of their operating systems, database, or applications software about every 18 months. System operators or administrators need to keep their skills current by attending training classes. Technical training tends to be expensive, therefore it is sometimes given a low priority, yet it is one of the most effective means of reducing operator error.

Process-Oriented IT Operations

Organizations need carefully documented procedures and processes for each system. This includes keeping accurate change records and a run book of each system. The run book should provide the system's standard configuration and a troubleshooting guide. A system administrator should be able to quickly find the appropriate procedure for trouble shooting a software application or hardware server. Standard configurations and recovery procedures need to be up to date and available. Each time a system failure occurs, it should be analyzed to determine if staff followed documented procedures or if the operation procedures need to be modified. This is usually referred to as "root cause analyses" and if approached as a learning tool, they can be extremely useful in reducing downtime.

Downtime Cause 3: Software Corruption or Bug

Software problems are the third largest cause of downtime. Most companies use a combination of Commercial Off The Shelf (COTS) software and internally developed applications. All of these in turn depend on hundreds of other software applications such as drivers, operating systems, middleware, and virus protection. A failure in any one of these can become the cause of an outage or downtime. Because of this complexity, the interaction of all the various software products cannot be completely understood by any one person. The software development and infrastructure engineering staffs should work closely early in the design process when any major upgrade or new application is contemplated.

This teamwork will provide an opportunity for the infrastructure engineers and the application developers to take full advantage of features that can minimize the impact on users in the event of an unplanned outage. For example, in active-active cluster configurations, such as Oracle RAC, applications can take advantage of automatic failover of database connections to surviving nodes and can automatically retry failed transactions without impact on the user experience.

Cost of High Availability

There is a significant cost associated with achieving high levels of availability. For unplanned downtime alone, it costs approximately 2.5 times the cost of a standard application to achieve a minimum availability of 99.5 % of scheduled uptime. A large proportion of these additional costs can be attributed to building a redundant infrastructure, but there is increased costs related to operations, management, high-availability, support, maintenance, and high-availability design, development and testing.

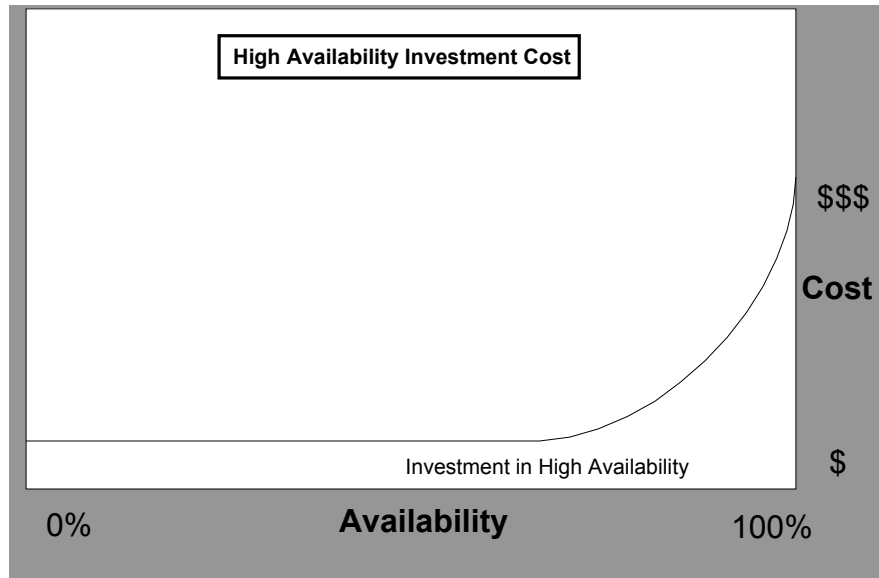


Figure 2.0

Decision Criteria for High Availability

An enterprise must determine the optimum balance between the costs of high availability solutions and the cost of unavailability. Figure 2.0 suggests how to analyze the trade-offs between the cost and the consequences of an outage. The figure suggests that high availability becomes increasingly expensive as we approach 100% availability. Moreover, the loss and consequences due to unavailability goes down with the increase in the availability.

Some of the key decision criteria to consider when deciding on a level of availability for a given information system are the following:

- Is the information system a revenue generator?
- Are there alternate methods to conduct business while the system is repaired?
- What is the value of the lost revenue if the information system is not available?
- Does the information system downtime affect the employee productivity?
- Is the information system mission critical?
- Is the loss of reputation and trust due to poor availability critical to the enterprise?
- Does the non-functioning of the information system result in lost customers?

Here are some general recommendations to help achieve a higher level of availability.

- | | |
|-----------------------------------|---|
| • Spend money...not blindly | • Examine the system history for failure patterns |
| • Assume nothing | • Build for growth |
| • Remove single points of failure | • Choose mature software |
| • Consolidate the servers | • Select reliable and serviceable hardware |

- Automate common tasks
- Document everything
- Establish Service Level Agreements
- Plan ahead for outages and disasters
- Re-use configurations
- Exploit external resources
- Keep it simple
- Conduct planning meetings for maintenance
- Cross train staff to degree possible
- Set reasonable customer expectations
- Plan for system maintenance
- Maintain tight security
- Test everything in lab first
- Maintain separate environment for test and lab
- Invest in failure isolation

Summary

Designing a cost-effective, high-availability environment for an information system(s) requires understanding the causes of outages, the critical elements for application execution, and the impacts of an application outage on the business. With today's technology, there is a range of solutions to support business-critical applications. For many businesses, UPS, RAID disk array, a journal file system, redundant power and cooling will provide adequate protection.

Although outages may occur, recovery is likely to be quick. If an application outage of more than a few minutes will severely impact business, a clustering solution may be necessary. For the really demanding, constant 24 hour, applications where outages either are life-threatening or will directly affect the survival of the business, high-end, fault-tolerant solutions may be required. Finally, be aware that good operational procedures can make an enormous difference between theoretical availability and the actual availability of a solution.