ZERO NINES
ALWAYS AVAILABLE™

Architecture Overview

ZERONINES: WHITEPAPER

www.zeronines.com

This document ("Brief") was prepared by the management of ZeroNines Technology Incorporated ("ZeroNines"), and is being furnished by ZeroNines, subject to the prior execution of the Confidentiality Agreement, solely for use by a limited number of third parties potentially interested in exploring business continuity solutions. ZeroNines does not make any representations as to the future performance of ZeroNines. Additionally, ZeroNines believes that the sources of the information presented herein are reliable, but there can be no assurance that such information is accurate and ZeroNines expressly disclaims any and all liability for representations or warranties, expressed or implied, contained in, or for omissions from, this Guide or any other written or oral communication transmitted or made available, except such representations and warranties as may be specifically provided in definitive contracts to be executed and delivered. Except as otherwise indicated, this Guide speaks as of the date hereof. Neither the delivery of this Guide nor any ensuing discussions conducted hereunder shall, under any circumstances, create any implication that there has been no change in the affairs of ZeroNines after the date hereof, or other specified date. This Guide is being furnished for information purposes only with the understanding that recipients will use it only to decide whether to proceed with discussions with ZeroNines management involving ZeroNines solutions. The information contained in this Guide is confidential and proprietary to ZeroNines and is being submitted solely for recipients' confidential use with the express understanding that, without the prior express permission of ZeroNines, such persons will not release this document or discuss the information contained herein or make reproductions or use it for any purpose other than potential discussions with ZeroNines management. By accepting this Guide, the recipient reaffirms its obligations set forth in the Confidentiality Agreement entered into in connection with the receipt of the Guide and agrees: (a) to maintain in strict confidence the contents of the Guide in accordance with such Confidentiality Agreement; (b) not to copy any portion of this Guide, and (c) if the recipient of the Guide does not enter into a transaction with ZeroNines to promptly return this Guide to ZeroNines at the address below. Inquiries regarding ZeroNines should be directed as follows:

For financial matters

Mr. Sean Myers, COO
ZeroNines Technology, Inc.
Corporate Headquarters
5445 DTC Parkway, Penthouse
Four Greenwood Village, CO 80111
+1.844.976.3696
Sean.Myers@ZeroNines.com

For all other matters

Mr. Alan Gin, President and CEO
ZeroNines Technology, Inc.
Corporate Headquarters
5445 DTC Parkway, Penthouse
Four Greenwood Village, CO 80111
+1.844.976.3696
Alan.Gin@ZeroNines.com

Contents

Introduction

ZeroNines Technology Architecture

ZeroNines' belief in the value of business continuity exceeds our faith in disaster recovery strategy and commercially available products and services. Our founders have seen so many organizations go down because of the limitations of widely used single-vendor DR implementations. ZeroNines has developed the patented AlwaysAvailable method and architecture to enable real multivendor business continuity.

ZeroNines does not use a disaster recovery strategy, and does not advocate it for our customers, for strategic and practical reasons. Our strategic reason is this: *recovery* is reactive, what happens *after* a disaster has *already harmed* your business. On its face, this is unsound strategy. Even if DR were strategically tenable, however, we would not rely on it because the methods available today for its implementation are riddled with failure points.

ZeroNines' AlwaysAvailable architecture disaster-proofs an application without a wholesale application rewrite.

Application availability on a ZeroNines AlwaysAvailable configuration exceeds commercial alternatives at the same or lower cost for the same or greater uptime.

Our architecture overcomes the limitations of disaster recovery architecture with novel topology and protocols.

The design principles of a ZeroNines AlwaysAvailable architecture are:

- A one-to-many (1:m) session type is supported
- Server hierarchy is eliminated
- Server sites are diverse
- Heterogeneous product sets are accommodated
- Load balancing is a side effect.

ZeroNines Technology, Inc., invented MultiSynch technology and has been using it for years in our own business for our own operational continuity. We rely on it.

Example Since activation on July 15, 2004, MyFailSafe.com has furnished continuous service to email clients. There has never been an interruption of service to email clients for any cause: scheduled or unscheduled maintenance, server ugrades, virus attack, distributed denial of service attack, or natural disasters. Never, for any cause.

The remainder of this Technical Brief furnishes details about these themes for senior IT architects and other readers who prefer a technical treatment. Readers who prefer a compliance or business perspective without technical detail are encouraged to consult a related Executive Brief, *Compliance, Continuity and Security*.

Why we developed this

ZeroNines' belief in the value of business continuity exceeds our faith in disaster recovery strategy and commercially available products and services. Our founders have seen so many organizations go down because of the limitations of widely used single-vendor DR implementations. ZeroNines has developed the patented AlwaysAvailable method and architecture to enable real multivendor business continuity.

Continuity is valuable

How valuable is business continuity? And why?

Data security and business continuity are valuable because operational failures are expensive in their direct and indirect costs. A vivid example of direct cost is lost revenue. An indirect cost is a drop in the company's stock price after an operational crisis.

A recent study of 350 operational crises at North American and European financial institutions, in which the direct financial loss exceeded $1 million per crisis, shows shareholder loss metastasizes to 12x the direct loss over 120 working days, cutting total shareholder returns by an average of 2 percent. The average direct loss in the sample is $65 million. Less than half of the risk events in the sample are from betrayals such as embezzlement, loan fraud, deceptive sales practices, antitrust violations and noncompliance with industry regulations, leaving more than half to other categories such as natural disasters and computer system failures.[1]

Additional quantitative studies of operational failures include:

- Since 1982, "failover" software recovery attempts using traditional disaster recovery approaches have averaged 40 per year, primarily due to loss of electricity, hardware and fires.[2]
- Large companies forego *3.6 percent of revenue* annually due to downtime, and the leading cause of those failures is application software faults, 36 percent of the total.[3]
- Of the 350 companies in the World Trade Center before the 1993 bombing, 150 were out of business a year later because of the disruption.[4]

These are examples of *private* value of business continuity, when the paychecks of one set of employees, or the wealth of one set of shareholders, is at risk.

New expectations for resilience

Systemic risk is the value lost when the interaction of different companies or parts of the economy is disrupted. This is the conceptual space where economic damage of a disaster grows exponentially and the complexity of recovery stupefies the imagination. It is the place where more and more companies now greet regulators who are interested in uptime. We believe regulators are beginning to view firms that cannot recover quickly as imposers of economic externalities, like polluters. Appropriately or not, what has long been a private matter of competition is becoming a public matter of regulation.

As part of the Federal regulatory response to 9/11, three Federal agencies solicited financial services industry comments on draft resilience practices for the US financial system. The thrust and intent of the draft was retained in the Interagency Paper issued in April 2003. The Paper now has Final Rule status.[5]

In interpreting the Interagency Paper, ZeroNines concurs with the Evaluator Group, a consultancy, to wit:

> Every CIO and Chief Legal Officer needs to read these documents. While they apply only to their industries in the short run…, they…. will define security standards for much of the IT industry by the end of this decade.[6]

Regulators now expect essential industry participants to affirm reasonable disaster recovery objectives, to implement sound practices for fulfilling those objectives, and we believe the regulators have asked firms—this is crucial—to exceed the capability of all commercial disaster recovery technologies known to exist at the time the rules were disseminated.

The disaster recovery objectives that the regulators sought to affirm are:

- Rapid recovery and timely resumption of critical (i.e. essential) operations following a wide-scale disruption
- Rapid recovery and timely resumption of critical operations following the loss or inaccessibility of staff in at least one major operating location
- A high level of confidence, through ongoing use or robust testing, that critical internal and external continuity arrangement are effective and compatible.[7]

In general terms, the sound practices that the regulators require are:

- identify essential activities in support of the firm's stakeholders, especially its transaction counterparties
- determine appropriate recovery and resumption objectives for these activities
- maintain sufficient geographically dispersed resources to meet recovery and resumption objectives
- routinely use or test recovery and resumption arrangements.

Regulators expect essential firms to recover and resume with zero data loss within two hours of a disaster (the two-hour rule) using a distant secondary site (the dispersal rule). They state that "back-up sites should not rely on the same infrastructure components (e.g. transportation, telecommunications, water supply and electrical power) used by the primary site." Regulators clearly want a failover site hundreds of miles away from the primary site so the secondary site is not disrupted by the same weapon of mass destruction, earthquake or hurricane that disrupts or destroys the primary site. When the Interagency draft was circulated for comment in August 2002, all three of these trauma scenarios were plausible.

Information security and business continuity standards are changing and the trend is clear. Customers are beginning to judge by the new standard of *business continuity*, virtually 100 percent accessibility. The more important your firm is to the economy—the more successful it is or the more central its role in commerce—then the more likely you face the security and continuity requirements of regulated industries. We are not saying that this degree of government involvement is appropriate or not. We state that it is expanding.

Why disaster recovery is not sound

ZeroNines does not use a disaster recovery strategy, and does not advocate it for our customers, for strategic and practical reasons. Our strategic reason is this: *recovery* is reactive, what happens *after* a disaster has *already harmed* your business. On its face, this is unsound strategy. Even if DR were strategically tenable, however, we would not rely on it because the methods available today for its implementation are riddled with failure points.

An executive from EMC Corporation, the leading computer storage equipment firm, puts it this way: "failover infrastructures are failures waiting to happen."[8]

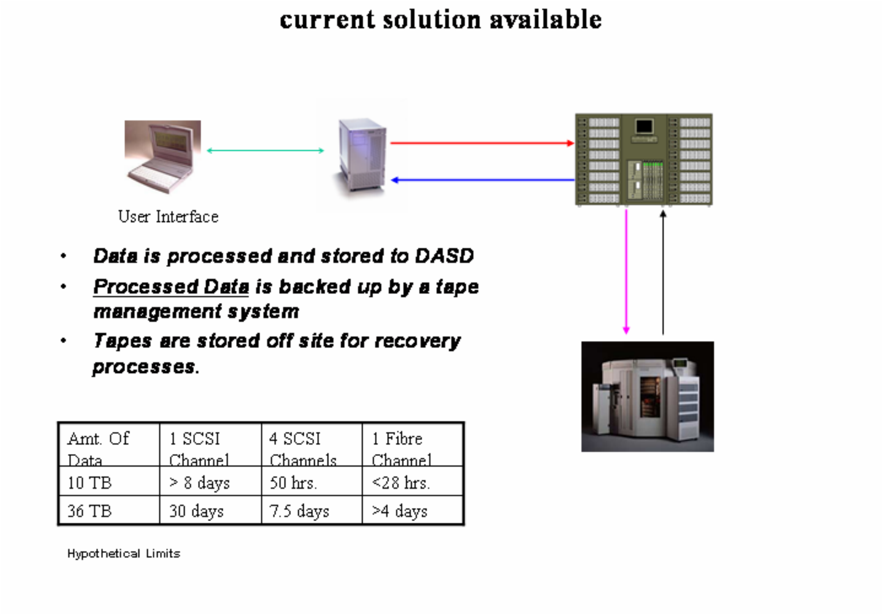
> If the boards of several publicly traded companies had any idea how much they are spending on today's disaster recovery architectures, they would realize they are paying

> for a fire sprinkler system that probably won't work if they have a fire.[9]

ZeroNines believes that existing disaster recovery designs are weak. These weaknesses aren't the fault of IT departments, but flaws propagated by vendor designs that have been present for years.

Before 9/11 these designs were usually deemed "good enough." They are:

- tape-based disaster recovery
- remote vaulting
- server failover and clustering.



current solution available

- ***Data is processed and stored to DASD***
- ***Processed Data is backed up by a tape management system***
- ***Tapes are stored off site for recovery processes.***

Amt. Of Data	1 SCSI Channel	4 SCSI Channels	1 Fibre Channel
10 TB	> 8 days	50 hrs.	<28 hrs.
36 TB	30 days	7.5 days	>4 days

Hypothetical Limits

Figure 1
Tape-based disaster recovery

Tape-based disaster recovery

Most companies use a tape-based disaster recovery strategy that was developed in the 1970s, before IT moved from the back office to become central in business. Tape-based disaster recovery uses a failover approach as depicted in Figure 1 and described as follows:

1 Periodically, backup copies of essential business data are produced at the primary site and transported to an offsite storage facility. For 90% of Global 1000 firms that use failover services,[10] each backup copy utilizes myriad magnetic tape cartridges, each about the size of a paperback book.

2 The primary site fails.

3 Seeking access to a contracted secondary site run by a disaster recovery service provider (DRSP), such as IBM, Sungard or HP, the CIO meets the contractual access requirement by declaring a disaster. If the CIO is not the first to declare a disaster in a shared-resource contract, access to the secondary site is not assured.[11]

4 The most recent backup copy from Step 1 is ordered transported to the secondary site. All tapes might be included in the shipment, but perhaps one is omitted accidentally. Subsequent transit time depends on weather conditions and the means of transit.

5 Tapes are used to "restore" the data and application software to the computers at the secondary site. If a single tape is damaged, used out of sequence, or is missing, the restore operation fails and must be restarted—assuming all tapes are present.

6 Operations resume at the secondary site.

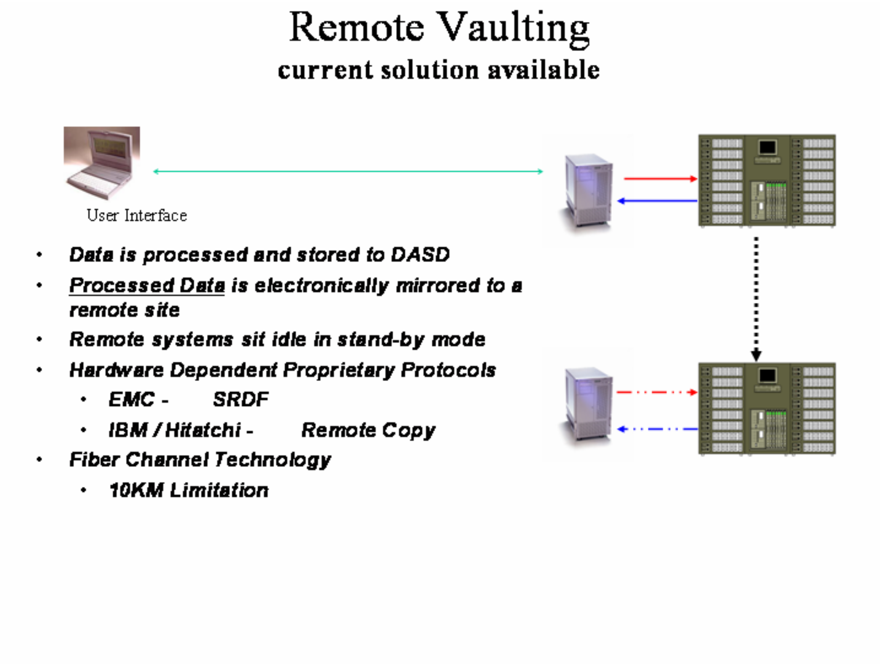
With tape-based disaster recovery, a delay is inevitable and of unknown duration: because of weather at the storage site, the secondary site and in between. A jet cannot deliver tapes if it cannot land due to poor visibility. A truck cannot deliver tapes if the road is coated with ice or diced by a hurricane or earthquake.

Examples On August 29, 2005, the surface course of five miles of Interstate 10, the principal road access to New Orleans across the eastern edge of Lake Pontchartrain, was chopped to pieces by Hurricane Katrina and did not reopen until October 14. Both other routes across the lake, US 11 and US 90, were restricted to emergency personnel for three days. The freeway system of Los Angeles was heavily damaged by the Northridge quake.

Remote vaulting

Attempting to address the weaknesses with tape-based recovery, vendors now support remove vaulting with split-mirror imaging. Vaulting has the advantage of reducing data transportation risk to practically zero by utilizing highly reliable telecommunications networks. But vaulting is inadequate for continuous application availability. The remote system is not fully replicated during the operations to split the mirror, image remotely and then resync the mirror. Transactions that occur between the last mirror (replication) and the trauma event cannot be assured. Even commit loggers cannot protect in-flight transactions. As with every other commercially available alternative, supported distances are inadequate to meet geodispersal requirements. Remote vaulting also requires matched hardware from the same vendor, limiting your flexibility.

Figure 2
Remote vaulting



Remote Vaulting

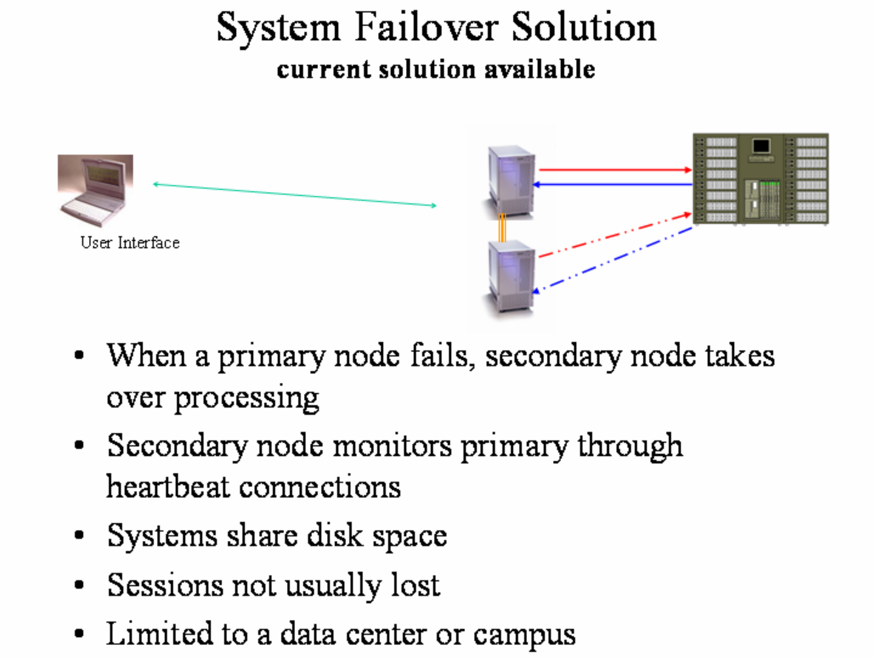
current solution available

- *Data is processed and stored to DASD*
- *Processed Data is electronically mirrored to a remote site*
- *Remote systems sit idle in stand-by mode*
- *Hardware Dependent Proprietary Protocols*
    - *EMC - SRDF*
    - *IBM / Hitatchi - Remote Copy*
- *Fiber Channel Technology*
    - *10KM Limitation*

Server failover and clustering

To address the glaring exposure to data loss of remote vaulting, computer systems vendors propose server failover and clustering. In these architectures, a heartbeat system tracks the health of physically close and connected servers. If one fails, another maintains client session processing without loss of service.

Figure 3
Server failover



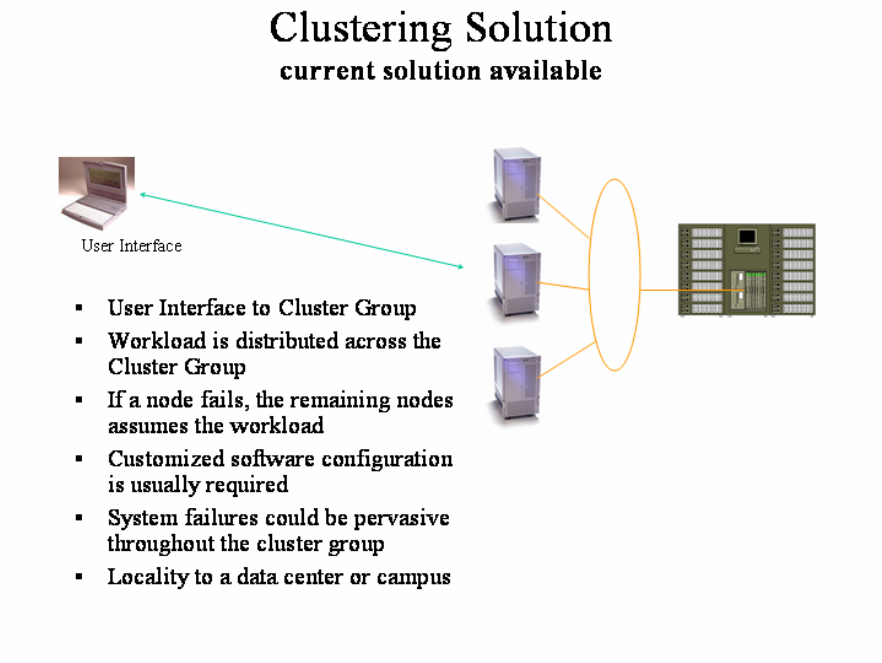
System Failover Solution

current solution available

- When a primary node fails, secondary node takes over processing
- Secondary node monitors primary through heartbeat connections
- Systems share disk space
- Sessions not usually lost
- Limited to a data center or campus

Server failover and clustering are the best of the conventional attempts at application continuity, but they have their own problems. Duplicate hardware and operating system configurations from the same vendor are required. Shared storage must be replicated carefully or it becomes a single point of failure; even then, cutover and block rewrite issues must be addressed. Server failover and clustering are so expensive and give the vendor so much negotiating leverage that they are rarely used. Supported distances are vulnerable to the same physical trauma.

Server clustering



Clustering Solution

current solution available

- User Interface to Cluster Group
- Workload is distributed across the Cluster Group
- If a node fails, the remaining nodes assumes the workload
- Customized software configuration is usually required
- System failures could be pervasive throughout the cluster group
- Locality to a data center or campus

Conclusion

Disaster recovery is not strategically tenable. Extensively used disaster recovery architectures have fundamental design exposures that cannot be worked around. IT organizations cannot circumvent the weaknesses with clever and diligent implementation. Disaster recovery designs are indadequate to support continuous application availability.

The two-hour rule and the dispersal rule cannot be satisified jointly by any commercial disaster recovery technology from any leading service provider or vendor today.

We are not the only firm to notice this. Sungard, a disaster recovery service market share leader that was taken private in August 2005, issued a press release in response to the draft Interagency guidance along these same lines. Sungard wrote:

> [A]ccelerated intra-day recovery/resumption with zero data loss, and a separation of 200-miles [sic] between primary and secondary sites, are technologically incompatible at this time….[C]yber-attacks, which represent a clear and present danger … are not sufficiently addressed by the Draft Interagency White Paper.[12]

How our architecture works

Introduction

ZeroNines' AlwaysAvailable architecture disaster-proofs an application without a wholesale application rewrite.

The "application" in this sense is the user of the AlwaysAvailable architecture. Examples include but are not limited to:

- storage configurations
- databases
- transactions monitors
- email systems
- other business application software.

Conventional disaster-proofing of application software is like fixing the Year 2000 bug, but managerially worse. The problem, albeit certain—a disaster *will* strike—has an unknown deadline As with other initiatives, this uncertainty affects the budgeting process in many large firms: something that is important never seems urgent, but when it becomes urgent, it's too late.

The conventional approach to disaster-proofing application software includes attention to its input/ouput (I/O) controls, the means by which programs read data to or write data from hard disks and other "peripherals" that serve the processors. *Most I/O controls have not been written to tolerate failures*, either by waiting a specified length of time or attempting the operation again. (Even those that do must limit retries in some determinative way.) As with the year references in the Y2K bug, each disaster-dumb I/O control must be identified, assessed, fixed and tested.

ZeroNines patented architecture includes interfaces that handle the input/output operations. The AlwaysAvailable architecture enables the application software to keep going.

Application availability on a ZeroNines AlwaysAvailable configuration exceeds commercial alternatives at the same or lower cost for the same or greater uptime.

Our architecture overcomes the limitations of disaster recovery architecture with novel topology and protocols.

Design principles

The design principles of a ZeroNines AlwaysAvailable architecture are:

- A one-to-many (1:m) session type is supported
- Server hierarchy is eliminated
- Server sites are diverse
- Heterogeneous product sets are accommodated
- Load balancing is a side effect.

A one-to-many (1:m) session type is supported

An AlwaysAvailable configuration maintains application sessions that are one-to-many (1:m) in nature. Each session from a client (service requestor) is maintained with multiple application servers (service responders). Duplicate replies from servers are eliminated during return to the client, ensuring integrity of the application image.

The application need not be session-oriented from the application's point of view. ZeroNines AlwaysAvailable supports sessionless and session-oriented applications.

Server hierarchy is eliminated

Each application server in an AlwaysAvailable configuration is always logically primary. In contrast with human relationships, server hierarchy does not exist in a ZeroNines AlwaysAvailable configuration. There are no secondary servers—not even the concept of "first among equals." Server primacy is perfectly shared without loss of effectiveness. At least two servers process every client request. Because there are no secondary servers, logical failover at the application layer does not occur, nor does it need to occur. Processing by one server might cease within the AlwaysAvailable configuration for typical reasons such as scheduled maintenance or physical trauma, but the other servers in that configuration continue processing in a zero-loss manner that is transparent to the application.

Server sites are diverse

ZeroNines uses the "site diversity" concept to indicate a number of server sites that share no physical exposures, such as infrastructure failure, natural disaster, fire or explosion. When server sites are diverse, dispersed by hundreds or thousands of miles and not dependent on the same infrastructure, AlwaysAvailable application availability is feasible.

Example Sites in New York and Singapore are diverse. They share neither natural disasters nor essential infrastructure such as electricity, water, or local exchange carriers. In this example, site diversity is two: two sites with no shared exposure.

Application availability is augmented as diverse sites are added to a configuration: five nines, seven nines or, with larger numbers of servers, effectively zero nines—100% application uptime to client requests, even with unscheduled server maintenance.[13]

The combination of shared server primacy and site diversity obviates application-wide recovery because application-wide failure does not occur.

Heterogeneous product sets are accommodated

Heterogeneity as a design principle produces more robust systems by minimizing system-wide effects of:

- attacks that are specific to a particular operating system
- vulnerabilities to model-specific defects of vendor hardware or software.

Example Every IT professional knows of situations in which Linux servers kept running when NT servers were under attack. Any operating system can be attacked. That said, we have never heard of a successful *all-OS* attack in a commercial setting.

ZeroNines AlwaysAvailable capability can be achieved with heterogeneous product sets. You can mix and match old and new hardware and operating systems, even from different vendors, without compromising AlwaysAvailable integrity. ZeroNines' protocols prevent race conditions and operate asynchronously across thousands of miles.

Removing matched-speed and matched-capacity constraints eases the burden of prototype projects and enables maintenance and upgrade of production servers and networks. You don't have to do everything at once to develop a prototype, deploy or maintain production.

The benefits of heterogeneity can be considered in the context of increased complexity. Some IT organizations prefer to standardize on one server operating system to achieve economies of scope and scale. Other organizations have long ceased attempting such an approach in favor of accommodating top-down decisions driven by user software requirments. Being application- and platform-agnostic, ZeroNines' architecture does not constrain the choice of server operating system, hardware or network protocols, enabling heterogeneity as a design strategy for those who choose it without excluding those who do not.

Load balancing is a side effect

The combination of shared server primacy and heterogeneity produces, as a side effect, a survival-of-the-fittest load balancing to support your application layer. AlwaysAvailable servers effectively compete to return results to requesting clients. A server that is closer to the requesting client or that temporarily has less workload might return a result more quickly than a faster processor that is more geographically distant or temporarily under heavier workload.

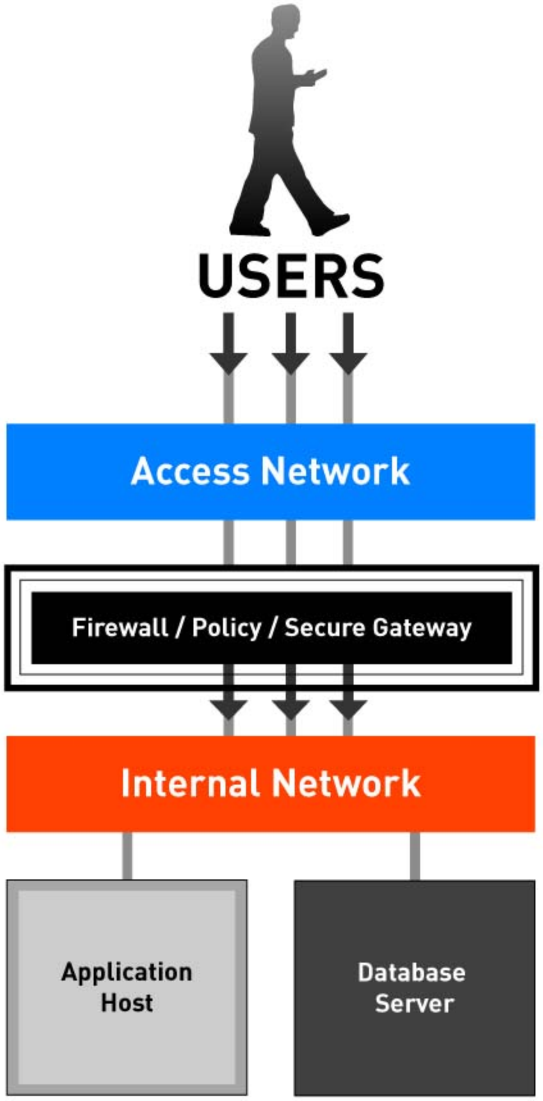
Designers remain free to match speeds and capacities of servers or networks for proprietary application-layer load balancing algorithms without disrupting AlwaysAvailable capability.

A working configuration before and after

To understand how a ZeroNines AlwaysAvailable configuration differs in a general sense from typical application access, consider the following exhibits.

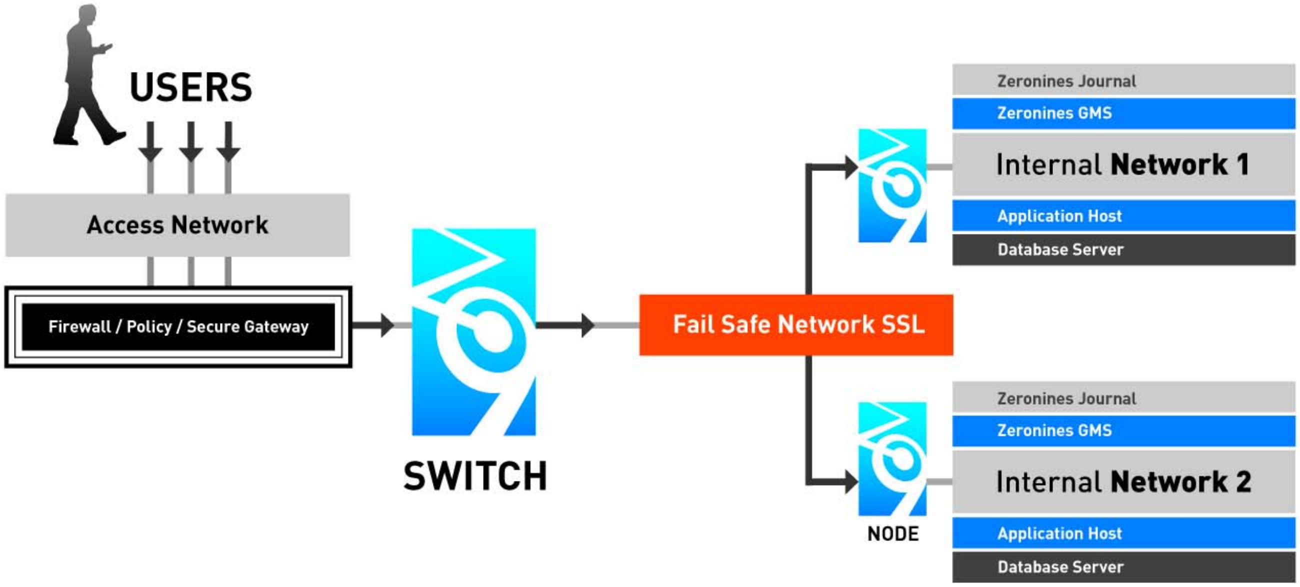
Figure 4 depicts a typical application access topology, before AlwaysAvailable. An access network links users' application clients to a datacenter's internal network via firewall, router and secure gateway. The application host responds to application requests, perhaps utilizing a separate database server for ease of reconfiguration or performance.

Figure 4
Typical application access (not AlwaysAvailable)



In a ZeroNines AlwaysAvailable topology as shown in Figure 5, two (or more) ZeroNines AlwaysAvailable switches are present between the application user network and the application server network. Each AlwaysAvailable switch may have one or more state-accurate shadowing switches that continue service to the application clents in case a switch discontinues service for any reason, such as scheduled maintenance. AlwaysAvailable Switches may be clustered for load balancing as desired.

Figure 5
AlwaysAvailable application access



The mere fact that an AlwaysAvailable configuration contains fewer single points of failure from a hardware perspective does not fully explain why continuous application availability is assured. Simply buying more servers and configuring them for traditional DR failover is insufficient to enable 100% uptime. Failover is insufficient for continous availability. An AlwaysAvailable architecture requires the AlwaysAvailable design principles to be implemented.

In a ZeroNines AlwaysAvailable configuration, each application server is associated with a ZeroNines AlwaysAvailable *node*, a listener function. When a client requests application service, at least two AlwaysAvailable switches pass the request to at least two AlwaysAvailable node listeners, each of which completely and independently processes the request using the respective servers associated with those listeners. The results generated by the servers are returned by the respective listeners to the switches, which cooperatively return *one* copy of the result to the requesting client. Thus a 1:m session is implemented. Duplication of data is prevented, and integrity of results is ensured, by the ZeroNines protcols and formats that are completely transparent to the application. Listener functions may be implemented as hardware integrated with the server or one or more software modules running on the associated server.

A ZeroNines configuration can utilize gateway, unicast or multicast protocols, depending upon your requirements. This network protocol flexibility is captured in our Transaction MultiSynch marque.

The intellectual property, the architecture and the way the pieces relate to each other, is the subject of U.S. Patent 6,760,861, issued July 6, 2004, and other filings.

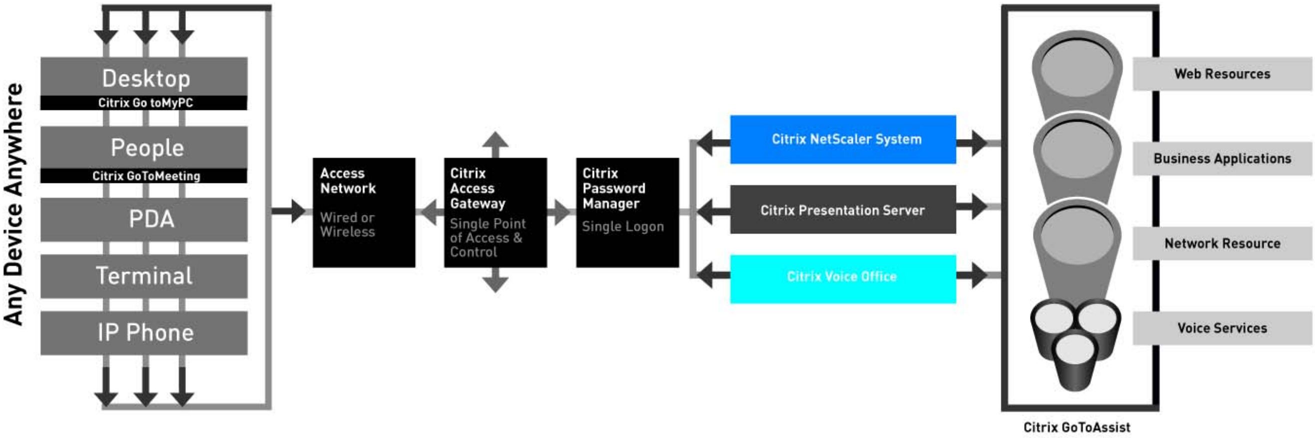
> A system, method, and apparatus for providing continuous operations of a user application at a user computing device. At least two application servers are provided with each application server running the user application concurrently and independently. Each application server may have a persistent storage device assciated with it for storing data. In response to a user request for data processing within the user application, the user request is transmitted to the at least two application servers for processing therein. A return result—responsive to the user request as processed by the one of the at least two application servers—is passed to the user computig device from one of the at least two application servers. In this manner, if one of the application servers fails or becomes unavailable due to a disaster or otherwise, the user requests can be continuously processed by at least the other application server without any delays.

Citrix example

As a supplement to the general before/after case, consider th following Citrix example, which show how a typical Citrix topology can be enabled with the AlwaysAvailable architecture.

The typical Citrix architecture in Figure 6 depicts a user “demand” side to the left and the server “supply” side to the right. Citrix components include the following.

Figure 6
Typical Citrix configuration before AlwaysAvailable



Server “supply” side

Access gateway A universal SSL VPN appliance that provides a secure, always-on, single point-of-access to any information resource, such as published applications, web applications, network files shares, etc.

Pasword manager Enables single sign-on.

Presentation server Ties the user to the back-end enterprise resources. The user may be remote across the WAN or, as is usually the case with the majority of our customers, local on the LAN. Citrix covers Windows, Web, and UNIX applications to any device form factor, OS and browser version.

NetScaler appliance Network appliance that combines application acceleration, layer 4-7 traffic management, SSL acceleration and application security.

Application gateway Extends access to include IP telephones by delivering voice and data applications to the screens and speakers of IP phones and wireless devices., and also can transform existing Web-based applications for such devices and phones.

GoToAssist server Distributes user access to help desks and call centers.

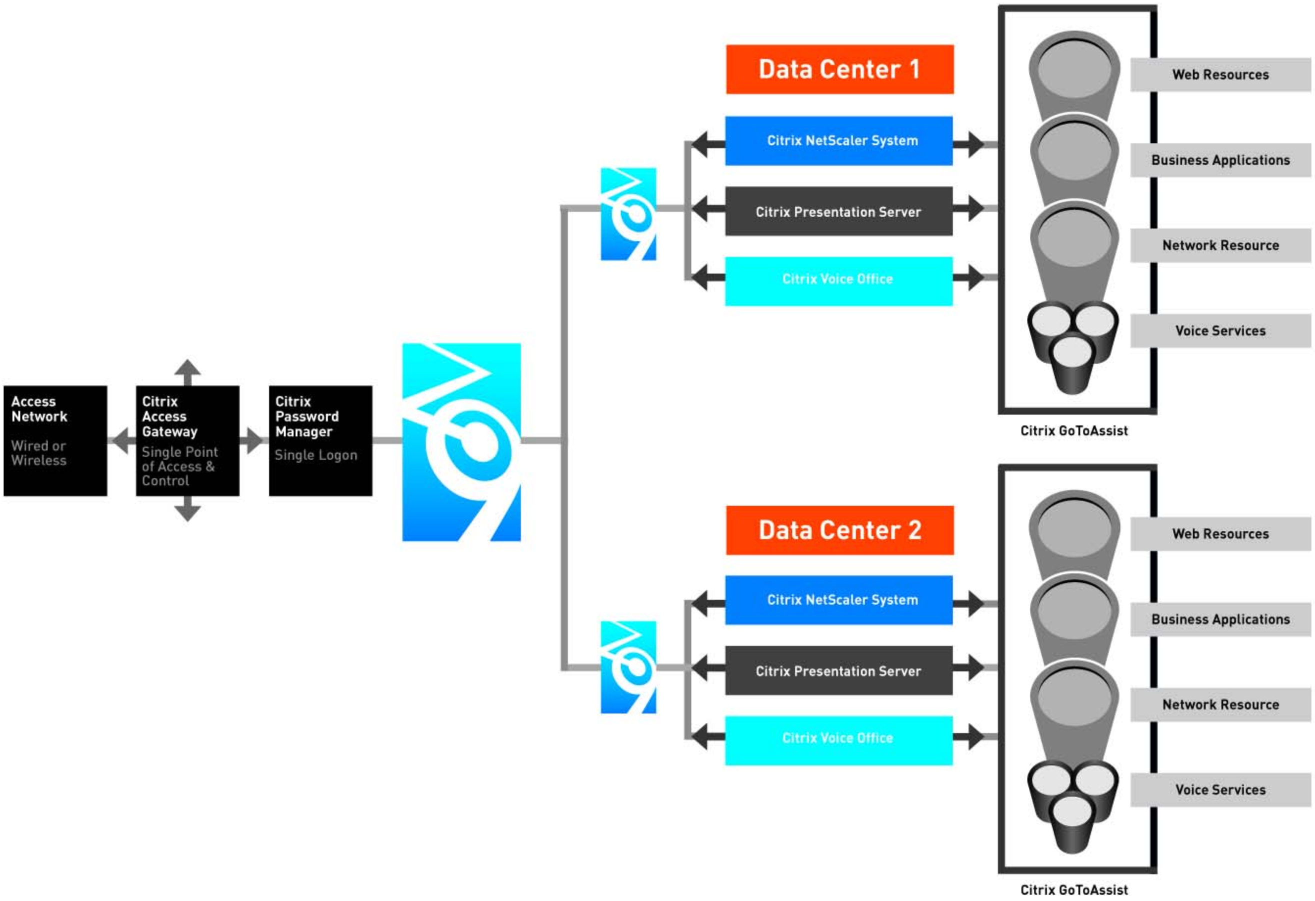
User “demand” side

GoToMyPc Remote desktop access for traveling users.

GoToMeeting Web conferencing for local and published applications.

With the improvement shown in Figure 7, Citrix service is continuously assured by the insertion of ZeroNines AlwaysAvailable technology between the supply and demand sides of the Citrix architecture. The ZeroNines switch and node depiction is simply represented by the ZeroNines logo.

Figure 7
AlwaysAvailable Citrix configuration

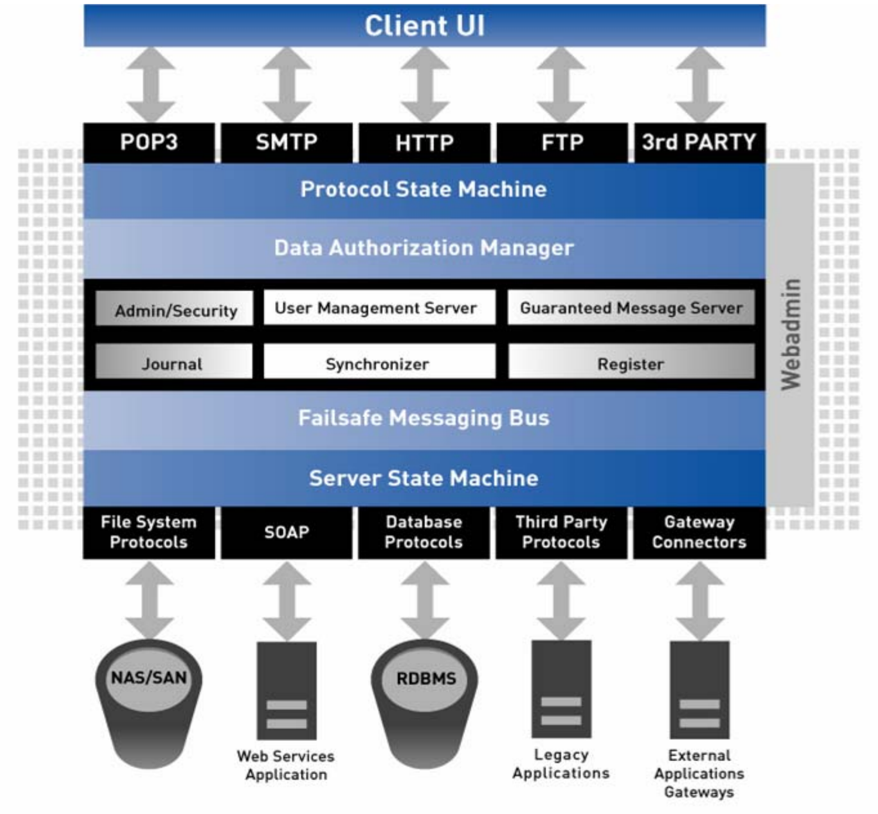


Overview of components

ZeroNines has developed a logical design architecture of functional layers. The enabled result is continuous application availability across heterogeneous platforms that can be separated by hundreds or thousands of miles. Functions can be deployed across software, firmware and hardware.

Figure 8 depicts AlwaysAvailable componentry with the application user layer at the top, core functional services in the center, and infrastructure and related interface support at the base.

Figure 8
AlwaysAvailable componentry



AlwaysAvailable client interfaces

The ZeroNines AlwaysAvailable architecture natively supports client requests via the following protocols:

- FTP
- HTTP
- POP3
- SMTP.

Applications that do not use a natively supported interface issue requests via a third-party interface developed with the AlwaysAvailable protocol interface development kit.

Protocol state machine

The Protocol State Machine (PSM) component near the top acts as a joining point for the supported client interfaces, manages protocol-specific wrappers that surround the transaction payload, and passes that payload to and from the Data Authorization Manager. The PSM also maintains session state information across AlwaysAvailable nodes so that stateless protocols can be utilized for business continuity.

Data authorization manager

Data Authorization Manager (DAM) implements secure access policies using information from the Admin/Security core function. The DAM decides whether an inbound message, connection or session attempt is permitted to be processed or rejected.

In the MyFailSafe.com case study that starts on page 24, DAM is implemented as a powerful spam filter.

AlwaysAvailable core functions

AlwaysAvailable core functions are as follows.

Admin / Security Administrative and security functions within the MyFailSafe core govern the privileges of a process to upload, download information and manage information. These functions also serve monitoring and auditing purposes.

User Management Server User Management server enables a network-wide management of the way that application servers communicate with the ZeroNines AlwaysAvailable services, managing the interface across all ZeroNines nodes. This capability is essential to make the hundreds or thousands of miles between sites to be manageable. A typical example of a user management request is the insertion of a protocol into the running system.

Guaranteed Message Server Guaranteed Message Server ensures point-to-point delivery of a message between AlwaysAvailable switches and listening devices. GMS monitors message acknowledgements from the listening devices and orders resends as required. AlwaysAvailable messaging is stateless. Protocol interface components maintain session integrity for the application, including sequence. GMS provides delivery commitment.

Journal The Journal provides a robust journaling service that utilizes a structured format and a cache that can be implemented with disk, tape or even flash memory. Journal keeps a copy of all transactions, enabling selective or comprehensive rebuild at any time for:

- sychronization of a server that has been offline for maintenance
- addition of a node for enhanced performance
- restoration of a site lost to a disaster, and
- forensic research for policy or regulatory audits.

Synchronizer Synchronizer enables sites to join running configurations on-the-fly and, commensurately, enables sites to leave a running configuration for maintenance. Synchronizer functions replicate the application image to a candidate site using predefined synchronization points for control and the Journal for content. When the candidate site "catches up" to the application image, the node is reactivated by the Register and joins processing.

Register Register functions track basic topology configuration data for management purposes.

Register manages information such as:

- How many sites are serving a particular application?
- Which sites are currently operational and which not?
- Where should messages be sent?

AlwaysAvailable messaging bus

AlwaysAvailable components communicate via the Messaging Bus. The Bus is highly secure, takes in transaction information and encapsulates it in our message formats. The Bus also maintains sequence of transmission for the application.

Server state machine

Server State Machine is the server side of the same function set as the Protocol State Machine. It tracks the discrete states of operation that a server may assume and the proper transitions between those states.

Other interfaces

The ZeroNines AlwaysAvailable protocol interface also supports file system protocols, database protcools, soap, etc. These are typically depicted next to the Server State Machine as "back-end" or "infrastructure" functions in large organizations.

Webadmin

Webadmin serves a browser-based administrative interface for control and monitoring of an AlwaysAvailable nework. Its requests are served by the User Management Server.

Relating nodes to 9's

ZeroNines has developed configuration guidelines for estimating the number of servers and other elements necessary to achieve desired application availability. We have tested these guidelines in our own business with our own mission-critical application.

Your AlwaysAvailable configuration must reflect the imperatives of your organization's Business Impact Analysis, business plan and regulatory requirements. ZeroNines believes that clients appreciate sizing approximations as a starting point for proof-of-concept and prototyping projects. Consultative services are available for the sizing of an AlwaysAvailable production configuration.

The site diversity indicated in supports the indicated application availability during trauma to *one* of the sites.

Availability in prototype (%)	Sites diversity required
99.999	2
99.99999	3
100	> 3

Scheduled maintenance ignored. Minima shown are adequate for prototyping projects. ZeroNines offers services for designing production configurations.

Augmenting the minima shown by adding incrementally diverse sites supports greater availability, such as during routine maintenance, upgrades, or additional trauma that causes simultaneous service interruptions at two or more sites.

Case study: MyFailSafe.com

Design

ZeroNines Technology, Inc., invented MultiSynch technology and has been using it for years in our own business for our own operational continuity. We rely on it.

For us, email is a mission-critical business application, so we commenced a MultiSynch implementation with the MyFailSafe.com email service (Figure 9).

- We standardized on one operating system for all three server nodes, but CPU, RAM and disk are neither speed- nor capacity-matched.
- Each server node is scheduled for 15 minutes of downtime per month for log resets, staggerred to ensure that no two nodes are ever scheduled for simultaneous maintenance.
- Telecommunication links are described in Table 1 on page 25.

Figure 9
MyFailSafe.com topology, on continuously since 2Q2004

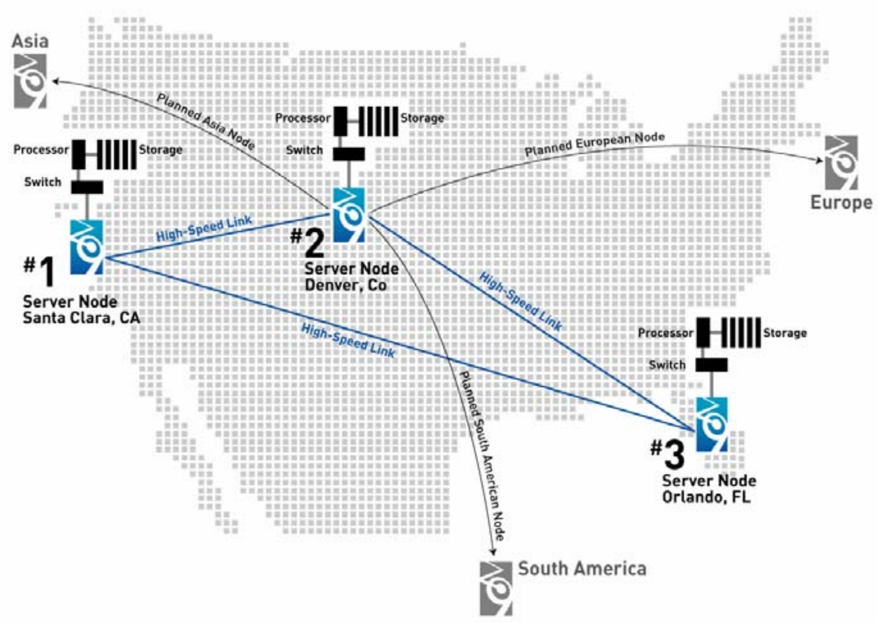


Table 1
MyFailSafe.com
telecommunication links

City	Carrier	Link characteristics
Santa Clara, California	MCI	1MB, burstable
Denver, Colorado	Level 3	1MB, burstable
Orlando, Florida	Time Warner Telecom	1MB–10MB

Table 2

Results

Since activation on July 15, 2004, MyFailSafe.com has furnished continuous service to email clients. There has never been an interruption of service to email clients for any cause: scheduled or unscheduled maintenance, server ugrades, virus attack, distributed denial of service attack, or natural disasters. Never, for any cause.

Example On August 12, 2004, Hurricane Charley caused electrical grid fluctuations that drained the Orlando local exchange carrier battery backup systems, isolating our node. Our own battery system prevailed and still had 75% of required charge when commercial power was reliabliy restored, but the site could not communicate for 16 hours because of LEC downtime.

Example During the late-December 2004 Santy worm attack on phpBB code, AOL email to two of our board members was disrupted as AOL battled the worm. Email service by our system was not disrupted.

Example In December 2004, a 3-day data center move disrupted service from the Florida node. As before, email clients received uninterrupted service.

Conclusion

ZeroNines Technology Architecture

We have seen that business continuity is valuable, both privately and publicly—so much so that Federal regulators are now setting security standards that increasingly influence IT outside of the traditionally regulated industries. Information security and business continuity standards are changing and the trend is clear. Customers are beginning to judge by the new standard of *business continuity,* virtually 100 percent accessibility. The more important your firm is to the economy—the more successful it is or the more central its role in commerce—then the more likely you face the security and continuity requirements of regulated industries. We are not saying that this degree of government involvement is appropriate or not. We state that it is expanding.

Disaster recovery is not strategically tenable. Extensively used disaster recovery architectures have fundamental design exposures that cannot be worked around. IT organizations cannot circumvent the weaknesses with clever and diligent implementation. Disaster recovery designs are indadequate to support continuous application availability.

The two-hour rule and the dispersal rule cannot be satisified jointly by any commercial disaster recovery technology from any leading service provider or vendor today.

ZeroNines' AlwaysAvailable architecture disaster-proofs an application without a wholesale application rewrite.

Application availability on a ZeroNines AlwaysAvailable configuration exceeds commercial alternatives at the same or lower cost for the same or greater uptime.

Our architecture overcomes the limitations of disaster recovery architecture with novel topology and protocols.

The design principles of a ZeroNines AlwaysAvailable architecture are:

- A one-to-many (1:m) session type is supported
- Server hierarchy is eliminated
- Server sites are diverse
- Heterogeneous product sets are accommodated
- Load balancing is a side effect.

ZeroNines Technology, Inc., invented MultiSynch technology and has been using it for years in our own business for our own operational continuity. We rely on it.

So can you. Contact us to start your prototype for testing.

Notes

[1] The McKinsey study assessed 350 events since 1990 from Fitch Risk Management's OpVar Loss database. Events were classified with guidance from the Bank for International Settlements. Early stage work was performed by Professor Ron Anderson of the London School of Economics, and the study was completed by Dunnett, Simoes and Levy of McKinsey & Company's London office. "Managing Operational Risk in Banking," *McKinsey Quarterly* 2005, 1.

[2] CPR Research, 2005.

[3] "The Costs of Enterprise Downtime", Infonectics Research, 2/11/2004.

[4] Gartner/RagingWire report cited in "Without the wires," Fabio Campagna, *Disaster Recovery Journal*, Winter 2002.

[5] Unless otherwise noted, what follows is based on ZeroNines analysis and "Interagency Paper on Sound Practices to Strengthen the Resilience of the U.S. Financial System." Board of Governors of the Federal Reserve System, Office of the Comptroller of the Currency, Securities and Exchange Commission. April 2003.

[6] "All aboard the new federal security rules super train," Jack Scott, TechTarget.com, 6/11/2003.

[7] The new focus on "external continuity arrangements are effective and compatible" addresses a concern that ZeroNines articulated in a *Brief* to our clients in 2002. Prior to the Resilience rules, the challenge of interpreting SAS audit requirements led us to conclude that interaction between a DRSP and its client was not an auditable matter. We believed that audit committees and boards were, therefore, relatively uniformed about the systemic risk of oversubscribed assets. The Resilience rules suggest that DRSP–client interaction is now auditable. This is a step in the right direction. Oversubscription of DRSP assets, however, appears to pose systemic risk.

[8] Dorian Naveh, Director, Product Marketing, 2005.

9 Conversation with ZeroNines, Benjamin Taylor, Chairman Emeritus, Disaster Recovery Institute, January 2002.

10 GartnerGroup.

11 Contracts for dedicated resources average 7x the cost of the shared-resource alternative. “Things to consider before choosing a primary site recovery approach or telecommunications vendor,” Randolph Fisher, CBCP. Disaster-Resource.com.

12 “SunGard Offers Comments on Draft Interagency White Paper on Sound Practices to Strengthen the Resilience of the U.S. Financial System.” Press release, 12/18/2002. http://www.sungard.com.

13 The name ZeroNines was coined in 2000 when one of our founders briefed an analyst from GartnerGroup, a consultancy with its heritage in information technology. As we concluded our presentation the analyst noted, “What you’ve developed in this architecture offers customers more than five nines availability…. You essentially take information technology availability to zero nines.”

Intentionally blank

ZERO NINES
ALWAYS AVAILABLE™

www.zeronines.com

For more information: **info@zeronines.com**

ZERO NINES
ALWAYS AVAILABLE

Corporate HQ
5445 DTC Parkway
Penthouse Four
Greenwood Village, CO 80111

T: (844) 976-3696

ZeroNines® Technology, Inc. provides a new standard in network disaster recovery, shifting the paradigm from reactive recovery to proactive business continuity. Our Always Available™ information security and availability technology pushes application uptime beyond five nines (99.999%) to virtually 100% anytime, all the time – zero nines. This enables uninterrupted access to business data, applications, and transactions despite disasters or network disruptions that would otherwise cripple the enterprise. Always Available™ processes all transactions in parallel on geographically dispersed servers that are all hot and all active, eliminating single points of failure. It operates agnostically across multiple platforms, leveraging existing processing and storage infrastructure. We also offer enterprise infrastructure assessment, program management and project implementations. Founded in 2000 and based in Denver, Colorado, ZeroNines' primary target customer base includes Global 2000 companies.

Contact ZeroNines today to find out how your business can be **Always Available**!

info@zeronines.com
www.zeronines.com