GRAVIC®
Shadowbase

“Breaking the Four 9s Barrier”
An Informational Series on Enterprise Computing

As Seen in *The Connection*, An ITUG Publication
September 2002 – December 2003

About the Authors:

Dr. Bill Highleyman, Paul J. Holenstein, and Dr. Bruce Holenstein, have a combined experience of over 90 years in the implementation of fault-tolerant, highly available computing systems. This experience ranges from the early days of custom redundant systems to today’s fault-tolerant offerings from HP (NonStop) and Stratus.

Series Topics:

***Gravic, Inc*.**
Shadowbase Products Group
17 General Warren Blvd.
Malvern, PA 19355
610-647-6250
www.ShadowbaseSoftware.com

ITI ITUG Availability Series
Breaking the Four 9s Barrier

Dr. Bill Highleyman
Paul J. Holenstein
Dr. Bruce D. Holenstein

Introduction

There is an old saying in business. You can optimize schedule, cost, and quality. Pick any two.

When it comes to configuring your data processing system, there is an equivalent saying. You can optimize performance, cost, and availability. Pick any two.

We typically configure our systems for performance and cost and let availability fall where it may. Even so, we are achieving impressive availabilities by industry standards. Typically, highly available systems such as HP NonStop® Servers will fail only about once every five to ten years and will then be down for an average of about four hours. This means that our systems are up 99.99% of the time, delivering four 9s of availability to our businesses.

When we talk about availability, we mean that all services upon which our business depends are available. "Available" means not only just working but also working at a performance level that makes our services useful. When sub-second response is expected, a multi-second response may be no better than no response at all.

Are four 9s good enough for you? You certainly have faced what an outage of several hours means to your business. If you are running a Web service, this could mean irate customers, lost sales, and perhaps lost customers. If you are providing banking services, this may result in large fines by governmental authorities. A stock market trading system outage would make international headlines. The failure of a critical emergency service such as a "911" system in the US could be party to a cardiac arrest or a building burned to the ground.

How would you like to improve the availability of your systems so that the loss of any significant capacity is measured in terms of centuries rather than years – *at little or no additional cost*? As an added plus, your systems could tolerate major disasters such as floods, fires, earthquakes, and terrorism as well as environmental failures which may take out your power or air conditioning. Getting more availability for your system expenditures is what this monograph is all about.

The availability of a system is directly related to the number of ways in which it may fail – its failure modes. We show in "Availability Part 1 – The 9s Game" that the number of failure modes can be reduced significantly by paying attention to how we allocate critical processes to processors.

We show in “Availability Part 2 – System Splitting” that failure modes can be further reduced by splitting a system into several smaller, independent nodes. This strategy not only dramatically improves availability but also, when a node outage does occur, only the capacity provided by that node is lost. Furthermore, the chance of losing the capacity provided by two or more nodes is virtually never.

As an added advantage, splitting a system into several nodes allows you to do upgrades and maintenance a node at a time, virtually eliminating planned downtime.

However, nothing comes for free. If we split a system into several cooperating nodes, the system data base must also be distributed across these nodes. Providing many duplicate copies of the data base can be very expensive as often the cost of the database subsystem represents a majority of the system cost. Part 2 also shows how we can distribute a system across geographically dispersed nodes without suffering additional database costs.

Another severe problem with distributing a data base is data collisions that lead to database contamination. This occurs when two users at different locations simultaneously update the same data item on different database copies. The result is inconsistent data propagated across the network. Data collisions can happen with surprising frequency and usually require manual intervention to resolve. The only general solution to this predicament is to avoid data collisions.

“Availability Part 3 – Synchronous Replication” describes how to avoid data collisions by ensuring that distributed copies of a data base are kept in exact synchronism by the replication of updates across the network as atomic transactions. However, doing so can increase transaction response time because of communication network delays. The performance impact of different methods for synchronous replication is evaluated, and we show that the method of choice depends upon whether the system nodes are collocated or are geographically dispersed. Distributed transactions (such as NonStop’s Network TMF) are generally appropriate for short transactions within a collocated distributed system. For geographically distributed systems or for large transactions, the use of asynchronous data replication with coordinated commits of distributed transactions is more efficient. In either event, the performance impact is more than overcome by the increased speed of new systems being introduced today.

Up until now, we have considered redundant systems whose outages are caused by multiple hardware failures. It turns out that more important factors are software faults and operator errors. “Availability Part 4 – The Facts of Life” extends the availability concepts to include these sources of outages. Here we stress the importance of fast recovery from an outage and point out that this is not only a technical issue but also more importantly a serious business process issue.

As in every facet of life, there are trade-offs and compromises. When it comes to availability, two important considerations are the time that it will take for a system to recover from a failure and the amount of data that may be lost due to a failure. Every organization must grapple with their own tolerance to recovery time and data loss and set objectives for what can be tolerated. The corporate objective for recovery time is known at the Recovery Time Objective, or RTO. The corporate objective for acceptable data loss is called the Recovery Point Objective, or RPO. There are a variety of technologies today that allow one to replicate a system, and each has its own RTO and RPO characteristics. These various technologies are reviewed in "Availability Part 6 – RTO and RPO.

In "Availability Part 5 – The Ultimate Architecture," we put all we have learned into configuring systems that can meet the following objectives:

- The frequency of losing more than a tolerable amount of capacity is measured in centuries.
- The system can be distributed geographically for disaster tolerance.
- The reconfigured system will incur little if any additional cost.
- Reconfiguring for availability is non-intrusive. It does not require application rewrites.
- By and large, the facilities for achieving these objectives are available today.

These are tough objectives indeed, and certainly there will be some cost and performance impacts. But are these impacts worth the significantly enhanced availability that can be achieved? Only you can answer that question.

Remember – a system that is down has zero performance
and perhaps an incalculable cost.

Authors' Note: In many places throughout this monograph, reference is made to HP NonStop systems. NonStop systems were originally developed by Tandem Computers to provide very high availability. The authors have considerable experience with these systems. However, concepts and recommendations presented in this monograph are extendable to all system types, including NT, Unix, and IBM Sysplex systems.

This monograph is a compilation of a six-part series of papers published in The Connection starting with the November/December, 2002, issue. The Connection is the official publication of ITUG, The International HP NonStop Users' Group.

Author's Biographies

Dr. Wilbur H. (Bill) Highleyman brings more than forty years experience in the design and implementation of computer systems to his position as Chairman of The Sombers Group, Inc. SGI is a turnkey custom software house specializing in the development of real-time, on-line data processing systems, with particular emphasis on fault-tolerant systems and large communications-oriented systems. He is also Chairman of NetWeave Corporation, which developed the middleware product NetWeave. NetWeave is used to integrate heterogeneous computing systems at both the messaging and the database levels. Dr. Highleyman, a graduate of Rensselaer Polytechnic Institute and MIT, earned his doctorate in electrical engineering from Polytechnic Institute of Brooklyn. He has published extensively on availability, performance, middleware, and testing and is the author of "Performance Analysis of Transaction Processing Systems," published by Prentice-Hall. He holds four patents and can be reached at billh@sombers.com.

Paul J. Holenstein is Executive Vice President of Gravic, Inc., the makers of the ITI Shadowbase line of data replication products. Mr. Holenstein has more than twenty-two years of experience providing architectural designs, implementations, and turnkey application development solutions on a variety of UNIX, Windows, and VMS platforms, with his HP NSK experience dating back to the NonStop I days. He was previously President of Compucon Services Corporation, a turnkey software consultancy. Mr. Holenstein's areas of expertise include high-availability designs, data replication technologies, disaster recovery planning, heterogeneous application and data integration, communications, and performance analysis. Mr. Holenstein, an HP-certified Accredited Systems Engineer (ASE), earned his undergraduate degree in computer engineering from Bucknell University and a master's degree in computer science from Villanova University. He has co-founded two successful companies and holds patents in the field of data replication. He can be reached at info@iticsc.com.

Dr. Bruce D. Holenstein is President and CEO of Gravic, Inc. Gravic's ITI Shadowbase software supports many of the architectures described in this book and operates on systems such as UNIX, Windows, NonStop and other platforms running databases including Oracle, Sybase, DB/2, and SQL/MP. Dr. Holenstein began his career in software development in 1980 on a Tandem NonStop I. His fields of expertise include algorithms, mathematical modeling, availability architectures, data replication, pattern recognition systems, process control, and turnkey software. Dr. Holenstein earned his undergraduate degree in Electrical Engineering from Bucknell University and his doctorate from the University of Pennsylvania. Dr. Holenstein has co-founded and run three successful companies and holds patents in the field of data replication. He can be reached at info@iticsc.com.