

The digest of current topics on Continuous Processing Architectures. More than Business Continuity Planning.

BCP tells you how to *recover* from the effects of downtime.
CPA tells you how to *avoid* the effects of downtime.

In this issue:

[Best Practices](#)

[Rules of Availability - Part 2](#)

[Recommended Reading](#)

[Breaking the Availability Barrier](#)

[Product Reviews](#)

[solidDB - a 5 9s Memory-Resident DB](#)

[The Geek Corner](#)

[Heterogeneous Systems - Part 2](#)

Complete articles may be found at
<http://www.availabilitydigest.com/articles.htm>.

Join Me in a "Sneak Peek" Webcast of my HPTF Pre-Conference Seminar on Active/Active

As I mentioned in the last issue of the Digest, I will be presenting a full-day seminar on active/active systems at the 2008 HP Technology Forum, to be held in Las Vegas, Nevada, USA, this June. The seminar will be held on Monday, June 16, starting at 8:30 AM and will cover the theory and implementation of active/active systems, with several case studies of successful implementations now in production.

I also will be giving a one-hour "sneak preview" of the seminar in a Webcast on Monday, May 19, at 1 PM EDT. This seminar will summarize in an hour what I will be talking about for a day at HPTF. Anyone can register for this free Webcast at <https://www1.gotomeeting.com/register/907760837>.

The full pre-conference seminar is available for Encompass or ITUG attendees. You can register for it at www.encompassus.org or www.itug.org.

I hope you can join me on the Webcast or at the full pre-conference seminar.

Dr. Bill Highleyman, Managing Editor

Best Practices

Rules of Availability – Part 2

There are many ways in use today to achieve high availabilities. Predominant among these techniques are lockstepped processors, checkpointed or persistent processes, clusters, and active/active systems. All use some form of redundancy to recover quickly from faults, and all are subject to a common set of principles.

We continue in this article a review of our sixty-four “Rules of Availability,” as published in our series of books entitled *Breaking the Availability Barrier*. We chose those rules that are particularly applicable as *best practices* to achieve continuous availability with redundant systems, with a focus on active/active systems.

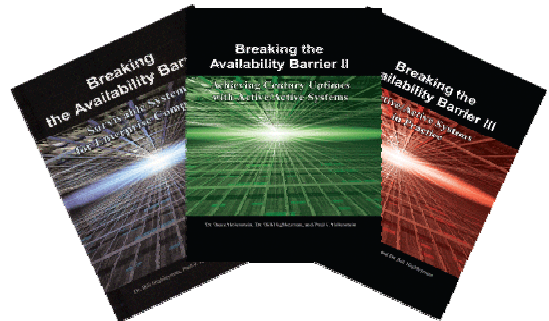
[--more--](#)

Recommended Reading

Breaking the Availability Barrier

Breaking the Availability Barrier is a three-volume set that focuses on active/active systems. According to the authors, “an active/active system is a network of independent processing nodes cooperating in a common application. Should a node fail, all that needs to be done is to switch over that node’s users to a surviving node. Recovery is in subseconds to seconds.”

Active/active systems don’t just provide high availability; they provide continuous availability. They can provide availabilities in excess of six 9s and uptimes measured in centuries. This three-volume series covers in detail the logical and mathematical theories behind active/active systems, additional benefits that these configurations can provide, a review of current product technologies required for the implementation of these systems, and several case studies of active/active systems in successful production.



[--more--](#)

Product Reviews

solidDB – a Five 9s Memory-Resident Database

Disk-based database systems are on a collision course with Moore’s Law. New multicore processors can support massive amounts of memory. The databases for many real-time applications can now comfortably fit in these large, high-speed memories. solidDB, from Solid Information Technology, is a memory-resident database that takes advantage of this new reality.

So why not simply configure a system with enough cache memory to hold the database? This will certainly eliminate most disk activity and will provide very fast database access. However, the database structures that are optimum for disk-resident databases are far from optimum for memory-resident databases. Therefore, a database specifically structured for memory residence can have a significant performance improvement over a disk-resident database.

There is one problem, however. If the intent is to eliminate disk activity, what happens to the “durable” in ACID? solidDB solves this dilemma through efficient disk logging or by providing a synchronized in-memory copy of the database in another server. In the latter case, should the primary database fail, the secondary database can take over with a complete, up-to-date copy of the database in tens of milliseconds.

[--more--](#)

The Geek Corner

Calculating Availability – Heterogeneous Systems Part 2

In all of our availability analyses to date, we have assumed that the nodes in a system are identical. But what if the nodal availabilities are not the same? What if one node is in a safe area, and the other is in Hurricane Alley in Florida? The Florida node will have an availability less than the other node because it stands to be impacted by a hurricane at some time. What, then, is the availability of the overall redundant system?

In Part 1 of this series, we reviewed the elementary probability concepts that apply to calculating availability. In this Part 2 of our series, we first briefly review these concepts; and we then apply them to redundant systems in which the nodes do not have the same availability (an assumption that we have made up to this point). We consider the impact on system availability of nonsymmetrical failover times, failover faults, and dual-node failures.

[--more--](#)

Would You Like to Sign Up for the Free Digest by Fax?

Simply print out the following form, fill it in, and fax it to:

Availability Digest
+1 908 459 5543

Name: _____

Email Address: _____

Company: _____

Title: _____

Telephone No. _____

Address: _____

The Availability Digest may be distributed freely. Please pass it on to an associate.
To be a reporter, visit www.availabilitydigest.com/reporter.
Managing Editor - Dr. Bill Highleyman editor@availabilitydigest.com.
© 2008 Sombers Associates, Inc., and W. H. Highleyman