the Availability Digest

achieving 100% uptime

Volume 2
Issue 4

April 2007

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.

Complete articles may be found at
   www.availabilitydigest.com.

In my talks and presentations to companies considering a transition to active/active, I hear a common theme of apprehension. *"This is going to be a big step, and we don't know where to start."* My response to such concerns is to remind people that if they already are operating transaction processing systems, an active/active solution is within relatively easy reach. After all, it is the transaction log created by such transaction processing systems that often serves as the source of data replication engine changes required to keep in synchronism the database copies in the active/active application network.

If you would like help in dipping your toe into the active/active waters, be sure to contact us at editor@availabilitydigest.com.

To many of us, transaction processing is well-known as a fundamental technology. For others not involved in modern-day business applications, transaction processing remains a mysterious technology viewed as difficult with which to work. For that audience, many articles in this issue of the Availability Digest focus on transaction processing systems, their benefits, and their challenges.

Access to many of our articles is by paid subscription. To gain the most benefit from our articles, be sure to subscribe to the Availability Digest at http://availabilitydigest.com/subscribe.htm.

Dr. Bill Highleyman, Managing Editor

Case Studies

BANKSERV Goes Active/Active

BANKSERV provides interbank electronic transaction switching services to the South African banking industry. It routes transactions from ATMs and point-of-sale devices to the issuing banks for approval. Should the BANKSERV services go down for any reason, customers will not be able to get their cash or complete their purchases.

To provide guaranteed uptime, BANKSERV is moving to an active/active system with NonStop nodes in Johannesburg and Midrand, 35 kilometers away. Each bank is assigned one of the nodes as its primary node. It routes all of its transactions to its primary node, and that node replicates the database changes caused by that transaction to the other node. Should a bank's primary node fail, it will route all transactions to the surviving node.

Anton Lessing, Technical Manager for BANKSERV, says that BANKSERV no longer talks about disaster recovery plans. The term is now Continuous Processing.

--more for subscribers--

Never Again

The Alaska Permanent Fund and the $38 Billion Keystroke

Do you ever have that sinking feeling just before you depress the delete key? Am I deleting the correct file? Can I recover it if I'm wrong?

An employee of the Alaska Department of Revenue perhaps should have thought twice before pushing that key. While maintaining a system that distributed oil revenues to Alaskans, he made that one fateful keystroke which totally wiped out an account worth $38 billion – and its backup!

Two months, $200,000, and 800,000 scanned documents later, the system finally got back online. Management learned from this disaster, and new procedures are now in place that should prove to be much more robust.

--more for subscribers--

Best Practices

Transaction–Oriented Computing

Building your systems utilizing transaction processing technology can bring a wide range of benefits. These include a guarantee that the application database is always consistent and that it is durable. Performance can be increased, lost data can be recovered, and data changes can be easily replicated to a remote site for disaster tolerance.

The database is consistent because it always holds the "truth." No single change can be made to it without the simultaneous application of all related changes – the atomic characteristic of a transaction.

The database is durable because once a change is made, it will always be there, even after the failure of the system or one of its components. The transaction log maintained by these systems provides the record of database changes that can be used to recover lost data. This log of database changes can be replicated to remote systems for disaster recovery and for distributed computing such as active/active systems.

Transaction processing technology is well known to and understood by those who develop business systems. However, it may not be so well understood by those developing other sorts of applications nor by those who are dealing with old, nontransactional applications. This article is aimed at the latter group.

--more for subscribers--

Active/Active Topics

Collision Detection and Resolution

Since each node in an active/active system is processing transactions independently of other nodes, there is the possibility that nodes may process transactions in different orders. Depending upon the method used to synchronize the database copies, this can lead to various problems:

- If synchronous replication is used, deadlocks may occur. Deadlocks are immediately observable to the applications and can be resolved with well-understood techniques.

- If asynchronous replication is used, data collisions may occur. Data collisions can sometimes be avoided by proper application structure. Otherwise, they can be detected and in some cases resolved automatically.

- If transaction replication is used, the results of the transaction may lead to diverging database copies. Diverging database copies can only be detected by periodic database comparisons and require manual correction.

In the design of active/active systems, it is important that these problems be understood and that procedures be in place to resolve them should they occur.

--more for subscribers--

Recommended Reading

Transaction Processing: Concepts and Techniques

 Transaction Processing: Concepts and Techniques is the classic book on transaction processing by Jim Gray and Andreas Reuter. The book describes transaction processing from the viewpoint of the users, the system administrators and operators, and the application designers by discussing in depth the various features of a transaction processing system.

The authors explore the basics of fault tolerance, summarizing the recovery from hardware faults and, more importantly, software faults. Transaction models are explored and introduce the ACID properties of a transaction – atomicity, consistency, isolation, and durability.

Transaction processing monitors, which provide the core services of a transaction processing system, and transaction managers, which manage the ACID properties of transactions, are described, first in overview and then in some depth, especially with respect to concurrency and recovery. The book ends with a description of many of the transaction processing systems available today.

--more for subscribers--

Update - Backup and Recovery

In the February issue of the Availability Digest, we reviewed the book, *Unix Backup and Recovery*, by W. Curtis Preston. The author has recently published an update, *Backup and Recovery*, published by O'Reilly Media, Inc; 2007. In addition to Linux and the various Unix flavors, his new book is extended to include Windows, Mac OS, and additional databases. The roster of products described include Amanda; Backup PC; open source near-CDP (continuous data protection) systems, including rsync, rsnapshot, and rdiff-backup; Solaris; Windows; Linux; HP-UX; AIX; Mac OS X; Oracle; Sybase; DB2; Sql Server; Exchange; PostgresSQL; and MySQL.

Product Reviews

Virtual Transactions with NonStop AutoTMF

Transaction-oriented computing can bring with it many benefits, such as guaranteed database consistency, recovery of lost data, higher performance, and disaster tolerance. Yet many older business applications running today are not transaction-oriented since they were written before this technology became mainstream.

AutoTMF, developed by Carr Scott Software and marketed by HP's NonStop Division, solves this problem. AutoTMF noninvasively binds to a nontransactional application and intercepts all I/O calls before passing them on to the operating system for execution. Based on the data access patterns, AutoTMF starts and ends transactions on the application's behalf and allows it to share in all of the benefits of transaction-oriented computing.

--more for free--

The Geek Corner

Calculating Availability – Environmental Faults

In our previous six issues, we have looked at the impact of various strategies and failure mechanisms on the availability of a redundant system. These have included:

- the impact of redundancy
- the impact of repair strategies
- the impact of node recovery and system restore times
- the impact of continuing improvements in hardware reliability
- the impact of failover time

- the impact of failover faults

System outages are caused by many factors, including hardware failure, software bugs, operator errors, and environmental problems. We have considered all of these factors in our previous articles except for the impact of environmental faults. In this article, we complete our analysis of availability by considering faults due to such events as power, fire, flood, earthquakes, hurricanes, social unrest, and malicious acts.

--more for subscribers--

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

The free Digest, published monthly, provides abbreviated articles for your review.
Access to full article content is by paid subscription only at
www.availabilitydigest.com/subscribe.

Name: _____

Email Address: _____

Company: _____

Title: _____

Telephone No. _____

Address: _____

_____

_____

_____