Off Site Database Management

Jesús J. Venegas Ríos
Computer Science
Juan M. Ramírez, PhD.
Department of Electric Engineering and
Computer Engineering and Computer Science
Polytechnic University of Puerto Rico

Abstract**—** *In today's database systems there is a rise in security needs and Database redundancy procedures in order to keep information readily available and secure in the event of a catastrophe. There are different ways to approach this with different types of Database Redundancy Protocols the most known of these are: Database Shadowing, Electronic Vaulting, Remote Journaling, Network attached storage and storage area networks. We are going to take a look at some of these into detail and see how they work, the advantages and the disadvantages they possess and their requirements*

Key terms**—** *Database Shadowing, Electronic Vaulting, Network Attached Storage, Remote Journaling*

INTRODUCTION

In the event of a disaster there is a need to bring systems back up as soon as possible, and while bringing the systems up may impose a challenge, when they are up again if the databases are empty the systems will not have any information to work on for the daily use of them. For this there are a few different Database Redundancy procedures. In this paper we will be mentioning the most common iterations of Database Redundancy protocols. But the deployment of servers and storage systems at numerous sites is both and administrative and a security nightmare for many organizations. The key challenges faced by these distributed systems include: [1]

- Deploying and managing backup software in a widely dispersed location.
- Implementing media management policies for both onsite and offsite storage of backup tapes that often require the use of third-party transportation and vaulting companies, increasing the risk of lost or misused data.
- Monitoring success/failure rates on remote backup processes and undertaking complex data/application recovery procedures, where local IT expertise is limited or nonexistent.
- Implementing an effective disaster recovery plan for large numbers of remote sites that allows a company to quickly recover applications to a separate facility. [1]

In order to approach these and many other challenges different types of procedures have been designed over the years by various companies including: Database Shadowing, Electronic Vaulting and, Remote Journaling. [2]

DATABASE SHADOWING

Shadow journaling, or database shadowing, enables secondary computers to maintain a "shadow" copy of selected databases as they are updated on a primary machine. By continually transferring journal information from the primary machine to the secondary machines, shadowing enables recovery to a system which is typically within only a few transactions of the source database. [3]

You can use shadowing for many purposes, each with its own set of important considerations depending on your system environment. Some of the most common objectives satisfied by shadowing include the following: [3]

- Disaster recovery, the most common use; it is simple and inexpensive.

- Read-only report server where ad hoc reporting tasks can operate on current data without affecting production.
- Low-budget replication where the databases are replicated on the shadow instance using journaling.
- Failover in some specific circumstances.

Database Shadowing Overview

A shadow Journal program instance may have one or more shadows. Shadow journaling monitors database activity on a primary system, the source, and causes the same activity to occur on a secondary system, the destination. It does this through a shadow client service running on the destination that continually requests journal file details from a shadow service running on the source. The shadow service responds by sending the details of the actual Set, Kill, and Bit journal record entries to the destination shadow over a TCP connection. The source and destination servers can be of different hardware, operating system, or CPU chipset.

All shadowing uses a fast transmission method which allows more efficient performance by sending the compacted journal file block by block. The shadow applies all transactions to the local databases. The transmission mode requires the data to be written to the journal file, which may introduce a delay of a few seconds. The shadow establishes a TCP connection to the server and receives the journal file. As the journal file downloads, another shadow process applies the journal entries to the local destination copy of the database.

Upon connecting to the data source server, the destination shadow sends the server the name of the journal file and the starting point. The shadow checks for new records periodically. If it does not have the latest records, the shadow downloads them and updates the databases. During these processes, the program continually stores checkpoints in a shadow global to facilitate rollback and restart capabilities.

The program purges the destination shadow copies of source journal files automatically. You can configure how long to keep files that are eligible for purging, that is, ones that have been dejournaled and do not contain any open transactions. For a visual example see Figure 1.

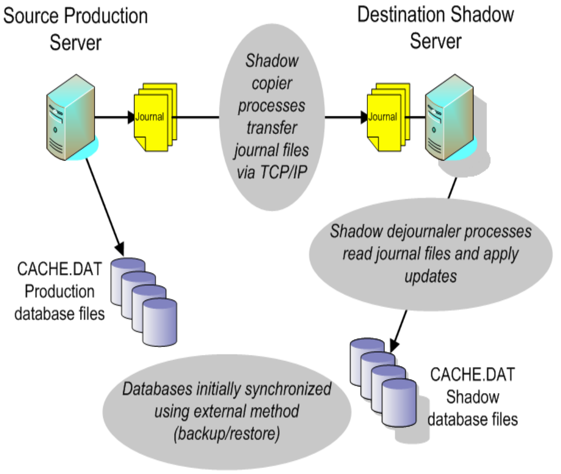
Source Production Server
Destination Shadow Server
Journal
Shadow copier processes transfer journal files via TCP/IP
Journal
Shadow dejournaler processes read journal files and apply updates
CACHE.DAT Production database files
Databases initially synchronized using external method (backup/restore)
CACHE.DAT Shadow database files

Figure 1
Database Shadowing

Shadowing Advantages

Shadowing offers several advantages:

- Recovery is quick: Activating a shadow makes it available immediately.
- Creating a shadow does not require exclusive access to the database.
- You can control the allocation of disk space. A shadow can span multiple files on multiple disks.
- Shadowing does not use a separate process. The database process handles writing to the shadow.
- Shadowing runs behind the scenes and needs little or no maintenance.

Shadowing Limitations

Shadowing has the following limitations:

- Shadowing is useful only for recovery from hardware failures or accidental deletion of the database. User errors or software failures that corrupt the database are duplicated in the shadow.

- Recovery to a specific point in time is not possible. When a shadow is activated, it takes over as a duplicate of the database. Shadowing is an "all or nothing" recovery method.

ELECTRONIC VAULTING

Among the many paradigm shifts sweeping across corporate America, one of the most dramatic is unfolding in the business continuity industry. It is a recovery option few companies would have considered even three to five years ago – electronic vaulting. [4]

Traditionally employed for only the highest-end solutions, electronic vaulting is emerging as a viable option for all environments. [4]

The concept of electronic vaulting in the disaster recovery industry is becoming an increasingly popular subject for several reasons:

- **Massive Amounts of Data to Manage-**As Enterprise Systems continue to grow and proliferate, the cost per megabyte of storage drops. IS departments are finding themselves with massive amounts of data to manage and back up. The larger the storage complex, the longer it takes to restore. [5]
- **New Regulatory Demands-** These demands have been placed on businesses regarding the amount of time by law they need to be back "on-line." For example, the SEC places severe time frame limitations on firms and businesses that are trading in the market. Other governmental restrictions may fall on hospitals or utility companies. Because of these new restrictions, all companies – those who have planned for unplanned outages and those who have not, are finding themselves behind the timeline of traditional disaster recovery. They are looking for some form of mirrored environment where recovery may be only minutes away.
- **Competitive Pressures-** These pressures are causing organizations to look deeper into their business continuity programs. Critical systems are being examined not only from a pure recovery perspective, but also from what it means to the organization as they relate to their stakeholders and competition. [6]

Before the arrival of electronic vaulting, all other changes in recovery methodology had been incremental – small but nonetheless significant improvements in slowly reducing the recovery window. The shift to electronic vaulting signifies a leapfrog in the way companies handle, protect and recover information. In order to better understand electronic vaulting and assess its relevance for a particular company, it is helpful to understand the factors causing this new view of recovery options. [5]

Electronic vaulting is especially valuable for small businesses and branch offices that do not have IT people on site. With electronic vaulting, backup to a secure site is automatic and the storage professionals at the vaulting company's site handle the details. Even with sites that have their own IT staff, electronic vaulting can make restores faster and more complete by moving set points closer to real time. [6]

Electronic Vaulting

Electronic vaulting (e-vaulting), or tape vaulting has been used by companies to mean different things. It typically fits between tape backup and disk mirroring as part of an overall data protection plan. It can be a service offering, a product, a feature of a product, or some combination of these. However, in all cases, electronic vaulting involves moving some amount of data from a primary site to another (secure) location via a network. [7]

Electronic vaulting can be valuable to all types and sizes of companies, although it's growing increasingly popular for small- to medium-sized businesses (SMBs) with limited IT staff, and for larger companies looking to protect a specific segment of users and data cost effectively, such as mobile laptop users.

It's typically implemented as part of a disaster recovery (DR) or business continuance plan. Many legal regulations such as HIPAA and Sarbanes-Oxley require that information be preserved, regardless of disasters. [7] You can see a visual example in Figure 2

Electronic Vaulting Advantages

Electronic vaulting has the following advantages:

- Faster recovery, as the data does not have to be retrieved from off-site and down-loaded.
- No need to ship the backups manually to a warehouse and store them.

Electronic Vaulting Limitations

Electronic vaulting has the following limitations:

- Cost of reserving the DASD at the hot site.
- The cost of the link required to the hot site.
- The cost of the software to transfer the data to the hot site.

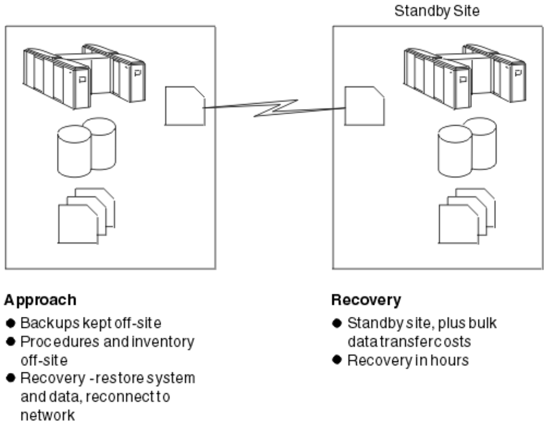
Standby Site
Approach
● Backups kept off-site
● Procedures and inventory off-site
● Recovery - restore system and data, reconnect to network
Recovery
● Standby site, plus bulk data transfer costs
● Recovery in hours

Figure 2
Electronic Vaulting

REMOTE JOURNALING

Remote journaling is a feature of IBM System I that allows you to automatically make copies of local journal receivers on one or more backup systems. The journal receivers could be broadcast to multiple systems from one system, or cascaded from one system to the next in a chain. These remote journal receivers can be used for a number of operations, such as saving them to tape on the backup system instead of tying up resources on the local system, or replaying changes to Journaled objects on a backup system so as to have a remote copy of your data. [8]

Traditionally, remote journal receivers have always been exact duplicates of the original local journal receivers. Remote journal filtering breaks this tradition and allows remote journal receivers to only have a subset of the journal entries existing in the local journal receivers. By setting filtering criteria, journal entries that match that criteria are filtered out on the source system and are never sent across the communication line. The remaining journal entries are completely unchanged and are sent to the remote system as normal. The resulting remote journal receiver will appear to have "holes" where journal entries are missing. Since the remote receiver may have less journal entries than the original local receiver, it may also consume less space. [8]

There are three flavors of remote journal filtering: before images filtering, filtering by object, and filtering by program. These three flavors are not mutually exclusive and may be used in conjunction with each other. A journal entry is filtered if it matches any of the filtering criteria for a remote journal connection. Each remote journal connection may have a different set of criteria, even if it is broadcast from the same local journal. In a cascade environment, only the original source system may specify filtering criteria; remote journal connections farther down the chain may not specify filtering criteria. Filtering criteria is specified when activating a remote journal environment, whether by the Change Remote Journal (CHGRMTJRN) command or the QjoChangeJournalState API.

Before images filtering

Database files and data areas have the option of journaling before images (as opposed to only after images). These are images of the objects written as journal entries before each update occurs. Before images were necessary in some environments, particularly for commitment control. They are, however, rarely needed on a remote system. All before images can be filtered from a remote journal by using the FTRIMAGES parameter on the Change Remote Journal (CHGRMTJRN) command or the Filter images field on format CJST0500 of the QjoChangeJournalState API. [9]

While remote journal receivers that may have had journal entries filtered cannot normally be used for replaying journal entries with the Apply Journaled Changes (APYJRNCHG) command, journal receivers that have only had before images filtered may be used with APYJRNCHG. They cannot be used with the Remove Journaled Changes (RMVJRNCHG) command.

Filtering by object

Not all objects that are Journaled need to be replicated to a backup system. Temporary files, for example, may come and go on the source system and are never needed on a target system. For such objects, remote journal filtering by object could be very useful.

Using remote journal filtering by object is a two-step process. First, you need to indicate which objects are to be filtered. This indication is a new journaling attribute that can be set for database files, data areas, and data queues. To set this attribute for existing journaled objects, use the RMTJRNFTR parameter of the Change Journaled Object (CHGJRNOBJ) command. Objects that automatically start journaling when they are created, moved, or restored into a journaled library can have this attribute set automatically. Check the inheritance rules for the journaled library using the Display Library Description (DSPLIBD) command, and change them using the CHGJRNOBJ command. The inheritance rules can also be set when the library is initially journaled using the Start Journal Library (STRJRNLIB) command. [8]

The second step of the process is turning on remote journal filtering for the remote journal connection. This is done with the FTROBJ parameter on the Change Remote Journal (CHGRMTJRN) command or the "Filter by object field" on format CJST0500 of the QjoChangeJournalState API. There are only two values, *YES and *NO. What this means is that either all journal entries for all objects with RMTJRNFTR(*YES) are filtered from the remote journal or none of them.

Only journal entries deposited while an object is set to RMTJRNFTR(*YES) are filtered. Journal entries deposited before the attribute is changed for the object are not filtered. The converse is true as well: journal entries deposited while RMTJRNFTR is set to *YES will continue to be filtered even after the object is changed to RMTJRNFTR(*NO). That is, the attribute that the object had at the time a journal entry was deposited determines whether or not that journal entry is eligible to be filtered.

It is worth noting that multiple remote journal connections broadcast from the same source journal cannot filter different sets of objects. Each connection must decide to filter all journal entries by objects marked as such, or filter none of them.

Filtering by program

In some cases it is useful to filter out journal entries sent on behalf of changes made by a specific program. Format CJST0500 of the QjoChangeJournalState API allows you to specify up to 20 different programs for filtering. All journal entries sent on behalf of any of these programs will not be sent to the remote journal. The programs are specified as 10-character name and library name fields and are case sensitive.

Each remote journal connection broadcast from the same source journal may have a different set of programs to filter.

Remote Journal Filtering Restrictions

There are a few restrictions associated with remote journal filtering that you need to be aware of before using this support. [10]

If you end your remote journal environment and then simply try to restart it with different filtering criteria, the remote journal connection will likely fail to restart. This is because a journal receiver can only have one set of remote journal filtering criteria associated with it. The receiver that is still attached to the remote journal has the old filtering criteria associated with it and cannot have its criteria changed. To restart this remote connection with different filtering criteria, you need to first delete the journal receiver attached to the remote journal.

As mentioned before, only the original local journal in a cascaded remote journal environment can set filtering criteria. Remote journal connections farther down the chain cannot modify or add to these criteria. [10]

Remote journal filtering is not allowed to releases prior to 7.1. Attempts to activate remote journal environments to previous releases will fail. Remote journal receivers that may have had journal entries filtered will not be allowed to be saved to prior releases. [10]

In most cases, you may not use the Apply Journaled Changes (APYJRNCHG), Apply Journaled Changes Extend (APYJRNCHGX), or the Remove Journaled Changes (RMVJRNCHG) commands with a remote journal receiver that has filtering criteria associated with it. This is true even if no journal entries were actually filtered. So do not use remote journal filtering if you plan to use any of these commands on the remote journal receivers. The one exception to this is that you may still use journal receivers that filtered before images on the APYJRNCHG and APYJRNCHGX commands. You may not use them with RMVJRNCHG.

Finally, journal entries may be missing from the remote journal receiver, obviously. Be careful not to filter out any journal entries needed for replication, backup, or auditing on your remote system. A visual representation for a hot back up with and without remote journaling can be seen in Figure 3 and 4.

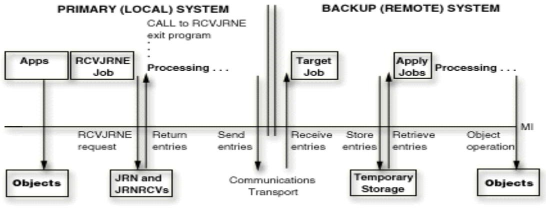

Figure 3
Hot-Backup Environment without Remote Journal Function, and Application-Code Based Apply

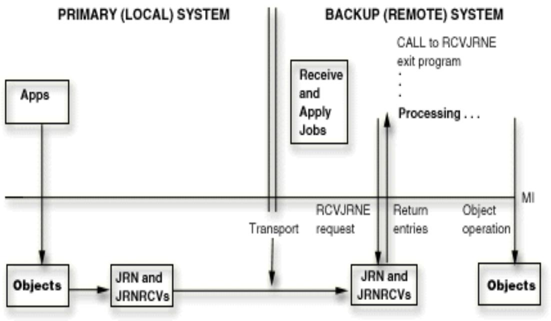

Figure 4
Hot-Backup Environment with Remote Journal Function, and Application-Code Based Apply

STORAGE AREA NETWORKS

The Storage Network Industry Association (SNIA) defines the SAN as a network whose primary purpose is the transfer of data between computer systems and storage elements. A SAN consists of a communication infrastructure, which provides physical connections; and a management layer, which organizes the connections, storage elements, and computer systems so that data transfer is secure and robust. The term SAN is usually (but not necessarily) identified with block I/O services rather than file access services. A SAN can also be a storage system consisting of storage elements, storage devices, computer systems, and/or appliances, plus all control software, communicating over a network. [11]

Put in simple terms, a SAN is a specialized, high-speed network attaching servers and storage devices and, for this reason, It is sometimes referred to as "the network behind the servers." A SAN allows "any-to-any" connection across the network, using interconnects elements such as routers, gateways, hubs, switches and directors. It eliminates the traditional dedicated connection between a server and storage, and the concept that the server effectively "owns and manages" the storage devices. [11] It also eliminates any restriction to the amount of data that a server can access, currently limited by the number of storage devices attached to the individual server. Instead, a SAN introduces the flexibility of networking to enable one server or many heterogeneous servers to share a common storage utility, which may comprise many storage devices, including disk, tape, and optical storage. Additionally, the storage utility may be located far from the servers that use it.

The SAN can be viewed as an extension to the storage bus concept, which enables storage devices and servers to be interconnected using similar elements as in local area networks (LANs) and wide area networks (WANs): Routers, hubs, switches, directors, and gateways. A SAN can be shared between servers and/or dedicated to one server. It can be local, or can be extended over geographical distances.

SANs create new methods of attaching storage to servers. These new methods can enable great improvements in both availability and performance. Today’s SANs are used to connect shared storage arrays and tape libraries to multiple servers, and are used by clustered servers for failover.

A SAN can be used to bypass traditional network bottlenecks. It facilitates direct, high-speed data transfers between servers and storage devices, potentially in any of the following three ways:

- Server to storage: This is the traditional model of interaction with storage devices. The advantage is that the same storage device may be accessed serially or concurrently by multiple servers.
- Server to server: A SAN may be used or high-speed, high-volume communications between servers.
- Storage to storage: This outboard data movement capability enables data to be moved without server intervention, thereby freeing up server processor cycles for other activities like application processing. Examples include a disk device backing up its data to a tape device without server intervention, or remote device mirroring across the SAN.

SANs allow applications that move data to perform better, for example, by having the data sent directly from the source to the target device with minimal server intervention. SANs also enable new network architectures where multiple hosts access multiple storage devices connected to the same network. A visual representation can be seen in Figure 5.

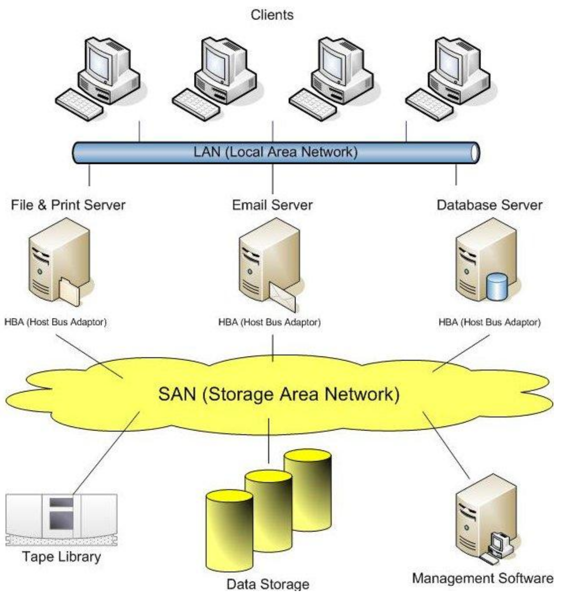

Figure 5
Storage Area Network

Using a SAN can potentially offer the following benefits:

- Improvements to application availability: Storage is independent of applications and accessible through multiple data paths for better reliability, availability, and serviceability.
- Higher application performance: Storage processing is off-loaded from servers and moved onto a separate network.
- Centralized and consolidated storage: Simpler management, scalability, flexibility, and availability.
- Data transfer and vaulting to remote sites: Remote copy of data enabled for disaster protection and against malicious attacks.
- Simplified centralized management: Single image of storage media simplifies management.

CONCLUSION

In conclusion all of these different forms of database Redundancy procedures are good by themselves but when you combine them they are an excellent form of database security and can maintain a database up and running and error free. One of such combinations is Database shadowing combined with Electronic vaulting. With this combination of procedures you get the advantages of database shadowing with none of the weaknesses, since you can have a backup of several days, weeks or even months in storage in case of an accidental user input.

REFERENCES

[1] "Data Definition Guide" ,InterBase, November 2011 retrieved from http://docs.embarcadero.com/products/interbase/IB2009/IB2009_DataDef.pdf

[2] "Six tiers of solutions for off-site recovery", IBM, November 2011 retrieved from https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.doc/dfht2/topics/dfht2ln.html

[3] "Caché Data Integrity Guide Shadow journaling", InterSystems, November 2011 Retrieved from http://docs.intersystems.com/ens20102/csp/docbook/DocBook.UI.Page.cls?KEY=GCDI_shadow

[4] "Electronic Vaulting systems", Business Continuity Expo November 2011, retrieved from http://www.businesscontinuityexpo.co.uk/ExhibitorLibrary/56/EVS_White_Paper_4.pdf

[5] Fellows, R., "What is electronic Vaulting?", SearchDataBackup, November 2011retrieved from http://searchdatabackup.techtarget.com/tip/What-is-electronic-vaulting

[6] "The electronic Vault Advantage", Pitney Bowes, November 2011 recovered from http://www.pbinsight.com/files/resource-library/resource-files/ElectronicVaultAdvantage.pdf

[7] Lindeman, J., "Electronic Vaulting Facilitating a New Era of Rapid Recovery", SunGuard Recovery Services inc. November 2011 retrieved from http://www.disaster-resource.com/articles/electric_vault_rapid_lindeman.shtml

[8] "Journaling: Remote journal filtering in IBM System I 7.1", IBM, November 2011 retrieved from http://www.redbooks.ibm.com/abstracts/tips0795.html?Open

[9] "Remote Journal Concepts", IBM, November 2011 retrieved from http://publib.boulder.ibm.com/iseries/v5r2/ic2924/index.htm?info/rzaki/rjournals/rzakirconcepts.htm

[10] Miszcyk, J., & et al, "AS/400 Remote Journal Function for High Availability and Data Replication", IBM, November 2011 retrieved from http://www.redbooks.ibm.com/redbooks/pdfs/sg245189.pdf

[11] Tate, J., & et al, "Introduction to Storage Area Networks", IBM, November 2011, retrieved from http://www.redbooks.ibm.com/redbooks/pdfs/sg245470.pdf