Storage Replication for Disaster Recovery
<table>
<thead>
<tr>
<th>Authors</th>
<th>Sony Pandi, Elva Chen</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document Title</td>
<td>Storage Replication Proposal.docx</td>
</tr>
<tr>
<td>Date</td>
<td>08-13-2013</td>
</tr>
<tr>
<td>Version</td>
<td>1.0</td>
</tr>
<tr>
<td>Status</td>
<td>For review</td>
</tr>
</tbody>
</table>
# Table of Contents

1 PROJECT .................................................................................................................. 4

2 INTRODUCTION ...................................................................................................... 4
2.1 Objective ................................................................................................................. 4
2.2 Problem Description .............................................................................................. 4
2.3 Project Relation with Operating Systems ................................................................. 5
2.4 Other Approaches .................................................................................................. 5
2.5 Our Approach ........................................................................................................ 5
2.6 Problem Statement ................................................................................................. 5
2.7 Area or Scope of Investigation .............................................................................. 6

3 THEORETICAL BASES AND LITERATURE REVIEW ........................................... 6
3.1 Definition of the Problem ...................................................................................... 6
3.2 Theoretical Background of the Problem ................................................................. 6
3.3 Related Research to Solve the Problem ................................................................. 6
3.4 Advantage and Disadvantages of Research ............................................................ 7
3.5 Proposed to Solution to Solve the Problem ............................................................ 7
3.6 Solution Different from Others ............................................................................. 7
3.7 Why This is Better ................................................................................................ 8

4 HYPOTHESIS .......................................................................................................... 8
4.1 Positive/Negative Hypothesis ................................................................................. 8

5 METHODOLOGY ..................................................................................................... 9
5.1 Input Data Collection ............................................................................................. 9
5.2 Approach – Problem Solution .............................................................................. 9
  5.2.1 Algorithm Design ............................................................................................ 10
  5.2.2 Language Used ............................................................................................... 11
5.3 Output Generation ................................................................................................. 11
5.4 Test Against Hypothesis ....................................................................................... 11

6 IMPLEMENTATION ................................................................................................. 11
6.1 Code ..................................................................................................................... 11
6.2 Design Document ................................................................................................... 11

7 DATA ANALYSIS AND DISCUSSION .................................................................. 15
7.1 Output Analysis .................................................................................................... 15

8 CONCLUSIONS AND RECOMMENDATIONS ..................................................... 15

9 BIBLIOGRAPHY ..................................................................................................... 15

10 APPENDICES ....................................................................................................... 16
1 Project

2 Introduction

2.1 Objective

The main objective of this project is to establish a connection and set up replication between the two newly created Virtual Machines. This is required if any disaster occur in the primary site. In order to recover the issues of the disasters, the Operations will be pointed to secondary site, as this is exact replica of Primary site.

2.2 Problem Description

Businesses are faced with increasing volumes of data to protect while maintaining stringent business continuity plans. Logical corruption, or loss of data due to threats such as viruses and software bugs, can be protected by ensuring that there is a viable copy of data available at all times. Performing regularly scheduled backups of the organizations’ data typically protects against logical types of data loss.

Another threat that may result in data loss is component or hardware failure. While most devices have begun to build in redundancy, there are other technologies such as application clustering technologies that can protect against a failure of a component while continuing to enable applications to be available. Just as the levels of protection for logical and component failures have grown, so has the reliance on the information systems being protected.

But, many organizations now realize that logical and local protection is no longer enough to guarantee that the organization will continue to be accessible. This loss can stem from planned downtime such as complete site maintenance to unplanned downtime such as power or cooling loss to natural disasters such as fire and flooding to acts of terrorism or war. The loss of a complete data center facility would so greatly affect an organization’s capability to function, that protection must be established at the data center level.

The challenge facing most organizations is how to minimize the lost revenue and operating expenses disasters cause. Even without losing buildings and equipment to hurricane or floods, organizations suffer in many ways including:

- Loss of employee productivity
- Loss of customer business during downtime
- Financial penalties for violation of legal requirements.
Failure to recover from disaster has lead to the demise of many organizations and threatened the existence of others.

2.3 Project Relation with Operating Systems

An Operating System is a set of programs that controls and coordinates the use of computer hardware among various application programs. It provides an environment within which user can execute programs. The one of the main functioning of the Operating system is Memory and File Storage management. It is very essential for the System to function properly. Also in real time scenario, Storage Replication plays major role when any disaster occurs.

2.4 Other approaches

The simplest way to protect system failure or file corruption is to copy the entire contents of the system to a backup device. However, full backup reading and writing the entire file system is slow, and storing a copy of the file system consumes significant capacity on the backup medium.

Faster and smaller backups can be achieved using an incremental backup scheme, Which copies only those files that have been created or modified since a previous backup. Incremental schemes reduce the size of backups, since only a small percentage of files change on a given day. A typical incremental scheme performs occasional full backups supplemented by frequent incremental backups.

2.5 Our approach

Our technique is to implement the Synchronous Replication. This type of replication writes data to primary storage and secondary storage simultaneously. On completion of replicating the data in the secondary site, the acknowledgement will be received.

Here we are choosing to implement continuous Synchronous Replication, which copies the data to secondary site continuously.

2.6 Problem Statement

Several issues could occur between systems that copy data to the backup medium as contiguous files vs. those that copy physical blocks regardless of file structure. File-based backups facilitate restores and are more portable, while device-based backups access the disk more efficiently. The file system can disallow writes during backup, or it can create instantaneous snapshots read-only copies of the file system, which may then be copied to the backup medium. Many systems that create snapshots provide a copy-on-write scheme that replicates data only when it must be modified, thus saving substantial disk space.
2.7 Area or Scope of Investigation

In the scope of IT, Disaster Recovery refers specifically to the implementation of some mechanism to protect the company's data, usually transporting it or replicating it offsite where it can then be used to get critical business systems back up and running. This alternate site can be a cold site, or even a production data center at another of your organization's sites.

When replication is invariably implemented at a raw disk or block level, only the low-level writes or disk blocks that have been modified are copied to the destination volume. The scope of this article is limited to disk- or block-level replication.

3 Theoretical bases and literature review

3.1 Definition of the Problem

In sum, disaster backup system can be categorized in full, incremental, full plus incremental and continuous backup system. A full plus incremental scheme supplements occasional full backups with frequent incremental backups. An incremental-only scheme avoids full backups of a file system and only writes incremental changes. A continuous incremental system copies new data within a few minutes of its being written.

3.2 Theoretical background of the problem

Techniques associated with disaster recovery should be able to ensure that recovery will happen as soon as possible to avoid time lag after any disasters. Dump tapes that contain full backups should be stored off-site to prevent data loss in the event of natural disaster or fire. Backup tapes should be secured against access by unauthorized persons, since they contain all of an installation's data. IBM's system automatically generates a disaster recovery plan that contains recovery instructions and scripts; the plan includes a list of necessary backup tapes and tape drives and their physical locations.

3.3 Related research to solve the problem

Several different schemes have been investigated for providing concurrent backup of multiple file systems over a network, including the ability of file systems to compress files before they are written to the backup device. Many systems also provide support for managing tapes and tape drives, and some provide automatic protection from site disasters by creating and managing remote copies of data.
3.4 Advantage and disadvantages of research

As data and file systems grow dramatically in today’s internet environment, it is likely that new backup strategies will be required to protect them. Our research and technique utilize the most promising techniques for handling very large file systems appear to be incremental-only backup schemes, device-based backup to use disk bandwidth efficiently and to avoid writing entire files based on small file changes, snapshots and copy-on-write for on-line backup, compression of data before backup, and automated creation of off-site backup files.

3.5 Proposed to Solution to solve the problem

Most organizations need a business continuity solution that delivers better, faster, more storage- and network-efficient backup and disaster recovery, regardless of the kind of disaster.

Synchronous Block Level Replication:

Synchronous Replication writes data to primary storage and Secondary Storage simultaneously. Here we are choosing to implement continuous Synchronous Replication, which copies the data to secondary site continuously.

Synchronous replication eliminates the potential performance problems of synchronous methods. The secondary site may lag behind the primary site, typically only by less than one minute, offering essentially real-time replication without the application performance impact. During Synchronous replication, application updates are written at the primary and queued for forwarding to each secondary location as network bandwidth allows.

With synchronous replication, if the network link becomes congested or broken, replication stops until the network link is reestablished. Any queued writes beyond the first couple of minutes of the disruption aren’t transmitted. Once the network connection returns to normal, the Synchronous replication program needs to synch the volumes on the source and target storage systems, which increases recovery times and the RPO.

3.6 Solution different from others

We are choosing to implement Continuous Synchronous Replication, which imports the qualities of both synchronous as well as Continuous replications.

In Synchronous replication, first the data will be copied on to the secondary site and then it will be written to the primary site, in which we may find some performance
issues. Where as in regular Asynchronous replication, data will be copied in periodic intervals.

In Synchronous replication, there will be little performance issues where as there would not be any stale data on Primary Site.

But as we are performing the replication at block level, this replication will facilitate both the features i.e. no performance issues and no stale data.

3.7 Why this is better

Storage array-based replication has the advantage of being operating system independent, since replication is done from array to array, and since it does not run on the host server, it adds no processing overhead.

**Synchronous Remote Replication allows us to:**

- Inexpensively synchronize business data between local and long distance remote volumes via IP, allowing immediate access to remote volumes
- Simplify moving and using data over extremely long distances from external sources, making the entire process straightforward, efficient and affordable
- Enable companies to restart mission-critical applications immediately after a primary site disaster, bringing critical activities back online.

4 Hypothesis

4.1 Positive/negative hypothesis

We would like to verify, first, with our method, the response time increases slower as the size of data increases comparing to other traditional backup system; second, the response time increases also slower as the fraction of write references increases.

In addition, we would like to demonstrate that the size of the allocated memory has little direct effect on performance. Instead, the execution time is most influenced by the amount of the memory that must be copied, which can be determined from the product of memory allocated and the fraction of memory written. We also hypothesize that, the worst case occurs when large address space programs update much of their memory; the biggest saving will come from programs with large address spaces, which update a small fraction.
5 Methodology

5.1 Input Data Collection

The desired goal of this replication is to transmit the contents of the primary site to secondary site in real time.

Initially, the input data would be the entire data presented in the disks of Primary site. Later on once the secondary site is in sync with the Primary Site, the input data, which has to be written to the secondary site and primary site will be same and generated at the same time.

This input data will be the business data that will be provided by business users in real time.

5.2 Approach – Problem Solution:

In this solution, Synchronous Replication easily replicates data to secondary sites in a cost-effective and highly efficient manner. Storage infrastructures will be easily extended via iSCSI to set up secondary remote storage locations anywhere in the world, without distance limitations. Synchronous Remote Replication provides quick and efficient transfer of data to off-site locations, whether the location is around the corner, across the country or around the globe.

What is iSCSI:

In a world where Internet Protocol (IP) dominates local and wide area networks, and data storage requirements grow unabated, it seems inevitable that these two forces converge. The Internet Small Computer Systems Interface (iSCSI) protocol unites storage and IP networking. iSCSI enables the transport of block-level storage traffic over IP networks. It builds on two widely used technologies — SCSI commands for storage and IP protocols for networking. iSCSI is an end-to-end protocol for transporting storage I/O block data over an IP network. The protocol is used on servers (initiators), storage devices (targets), and protocol transfer gateway devices. iSCSI uses standard Ethernet switches and routers to move the data from server to storage. It also enables IP and Ethernet infrastructure to be used for expanding access to SAN storage and extending SAN connectivity across any distance.

Synchronous Replication is block based and supports iSCSI logical volumes. It provides data availability in case of source system disaster.

Actual Procedure of Disaster recovery:
Data is written and read to Primary Site.
Data is continually replicated to Secondary Site via Internet connection (iSCSI)
In case of raid array error or disk drive error in the Primary Site, the server will send an e-mail notification to the administrator,
In the case of a failure of Primary Site, users will be notified,
Administrator then switches users to the Secondary Site over the WAN
After switching, replicated volume will be available on Secondary Site.

5.2.1 Algorithm Design

**Algorithm or Procedure to implement Synchronous Replication:**

- Configure Hardware
- Configure the Secondary Server i.e. destination node
  Physical storage space is needed at the secondary site to hold replicated data from the primary site.
- Configure the Primary Site i.e. source node
- Configure iSCSI.
- Implement Schedule Replication.

With Synchronous data replication, data from one disk is passed only to a local server before an acknowledgement is received.

- The local disk will then pass data across the WAN to the remote disk as time and bandwidth allow.
- Hard-drive data is organized on the disk in a large circular grid with each grid point being called a "block"
- The data will be read from Primary Site in Block Granularity. All writes on the Primary Site, undergoing replication are tracked in the Log File.
- Block-level cloning copies data by bypassing any interpretation of the logical file system structure and copying the drive’s internal block-level organization
- By looking at the which blocks contain file or application data, copy the content of the blocks.
While copying Write order fidelity has to be maintained.

Once the Initial Sync is performed, Incremental replication will be set up between Primary and Secondary Servers.

In many cases, the replicated disk content will lag behind the local data, by as much as several hours. However, asynchronous behavior works well over long distances (because latency isn’t a factor) and inexpensive low-bandwidth links.

Check the status of Disk Replication

5.2.2 Language Used

We will be using C language, in order to implement the operations of copying the newly added data from Primary Site to Secondary Site in the remote location. These operations will be performed at kernel Level.

5.3 Output Generation

The desired Output of this replication is that secondary site should be replica of Primary site. In case any disaster occurs Secondary site should come up and operations should be carried out without any impact on Business.

5.4 Test Against Hypothesis

As discussed in Hypothesis, tests will be carried based on response time. Also if huge volumes of data are to be replicated over the network, data consistency and response time parameters will be checked.

6 Implementation

6.1 Design Document

In order to implement the Data Replication task for Disaster Recovery (as part of this project), following configurations have been performed before actually implementing the code.

As we are planning to implement the Data Replication on two virtual machines, implementation of required prior tasks are as follows:
Installation of Virtual Manager: In order to manage the virtual boxes, Virtual manager was installed; In this we have installed Oracle Virtual Manager. It provides a user Interface, which is standard Application development Framework.

Creation of Virtual Machines: As we are planning to implement the replication task on two virtual machines (Source and Destination), two virtual machines should be installed.

OS Installation (CentOS): On creation of two new virtual machines, Operating System has be to installed on virtual machines. As part of this, CentOS Operating System was installed on both the virtual machines.

Assign the Static IP: Once the Virtual machines are created, they will be created with a default, dynamic IP Address. Therefore, IP address was changed to Static mode and assigned the IP addresses for the VMs.

For Virtual Machine 1 (VM1) – 192.168.1.32
For Virtual Machine 2 (VM2) – 192.168.1.30

Establish the connection between two Virtual machines – In order to copy the data from Source machine to Destination machines, there should be a connection established between the two virtual machines. On establishing the connection make sure that two virtual machines are able to ping each other.

Configure iSCSI initiator on source machine:

iSCSI is a network protocol defined to allow scsi commands over TCP/IP stack, allowing to hosts I/O block operations like a device storage attached locally. With iSCSI we’ve to difference to basic concepts:

iSCSI initiator: is called the scsi client, and can be connected to the server of two ways:
Software initiator: Normally is implemented as a module that will used by the network interface and emulate scsi devices. Is the most implementation.
Hardware initiator: It use a dedicated hardware to implement iSCSI. The work load of the iSCSI process is handled by this hardware.
This can be done as:

- Install the software package:
  - Yum `–y install iscsi-initiator-utils`
- Configure the iqn name for the initiator:
  - vi /etc/iscsi/initiatorname.iscsi and change the iqn name.
- Start iSCSI initiator daemon:
  - /etc/init.d/iscsid start
  - Discovering targets in our iSCSI server:
    - `iscsiadm --mode discovery -t sendtargets --portal 172.16.201.200`

✓ Install SCST and configure iSCSI target on Destination server

Download and Install the components scst, scstadmin, iscsi-scst and configure the scst target on destination server. Upon configuring the SCST target on destination server, Source machine should be able to identify and storage device of secondary server.
On completion of all the configurations replication task will be performed. In order to setup the block level replication between two virtual machines, we have implemented a framework called FUSE i.e. Files system in User Space.

FUSE:

File system in User space (FUSE) is an operating system mechanism for Unix-like computer operating systems that lets non-privileged users create their own file systems without editing kernel code. This is achieved by running file system code in user space while the FUSE module provides only a "bridge" to the actual kernel interfaces.

The FUSE kernel module and the FUSE library communicate via a special file descriptor which is obtained by opening /dev/fuse. This file can be opened multiple times, and the obtained file descriptor is passed to the mount syscall, to match up the descriptor with the mounted file system.

On Installing the FUSE Framework n Source Virtual machines, block device will be implemented by looping over.
7 Data Analysis and Discussion

7.1 Output analysis

Given the discussion above, we would like to analyze the data so we can understand quantitatively the effects of our Replication on response time. There are a total of four parameters to be considered:

✔ The frequencies of a block of data are copied.
✔ The number of memory that a process allocates
✔ A fraction of memory, which is to be written.
✔ The response time.

Based upon the above four parameters, analyses can focus on comparing the relationship between memory and response time, and between fraction of memory written and response time.

8 Conclusions and Recommendations

With the Synchronous replication performed on two virtual machines, data will be transmitted from primary site to secondary site. In future improvements, we could extend the scope of the replication over WAN. Also we could track

9 Bibliography


10 Appendices