StorTrends Whitepaper

"Journal-Assisted Replication"

American Megatrends International GmbH
12.12.2007

INHALTSVERZEICHNIS

1. Introduction

Continuous Asynchronous Replication solutions offer an affordable, highly reliable method of recovery and enterprise backup. Asynchronous replication provides the ability to migrate data between storage platforms, provides disaster recovery over any distance between any hosts and storage. Asynchronous Replication works by taking a copy of every transaction on a SAN and replicating that to a remote site. Unlike in Synchronous replication where every I/O has to wait for an acknowledgement from the remote location, in this form of replication, the incoming I/Os are collected and staged locally before replicating them. Replication is not done inline with the host I/O and that is why it is called asynchronous replication. The Primary site has a staging area to hold the I/Os before replicating. This staging area could be local RAM or a more permanent repository. Local RAM buffering requires larger memory footprints and entails frequent I/O throttling to guard against local buffer overflows. This solution is relatively easy to implement but breaks when there is a link disruption for extended period or if there is a wide disparity between the incoming and out going link rates. StorTrends stages data in a rolling log and is thereby is more tolerant to link speed disparities or communication outages. Every I/O has to be replicated out in the same sequence it arrives at the Production site in order to ensure application consistency. This write-order-fidelity is normally ensured by time-stamping incoming I/Os and transferring them out maintaining their chronological dependence.

Yet another popular form of Asynchronous replication is Snap Assisted Replication (SAR). This is the most Link efficient method for replication. In this method only the block level changes between two snapshots are replicated and not every I/O is sent out to the wire. This, however, suffers from a poorer RPO and does only guarantee discrete and not continuous recovery points.

2. StorTrends Journal Assisted Replication (JAR)

StorTrends implements a very I/O efficient Journaling Module that registers with an N-Way router to configure the flow In the I/O path. This allows Journaling module to Implement Zero-Memory-Copy transactions and coexist with other replication agencies (e.g. Synchronous Replication). Residing at the very core of the I/O stack, this module forms the nucleus of Continuous Data Technology (CDT). This granular and chronologically ordered I/O streams can be interfaced with different CDT agencies to provide Continuous Data Protection (CDP), Continuous Data Replication (CDR), Continuous Data Imaging (CDI) etc. JAR is one such agent of the Journal Module that implements CDR.

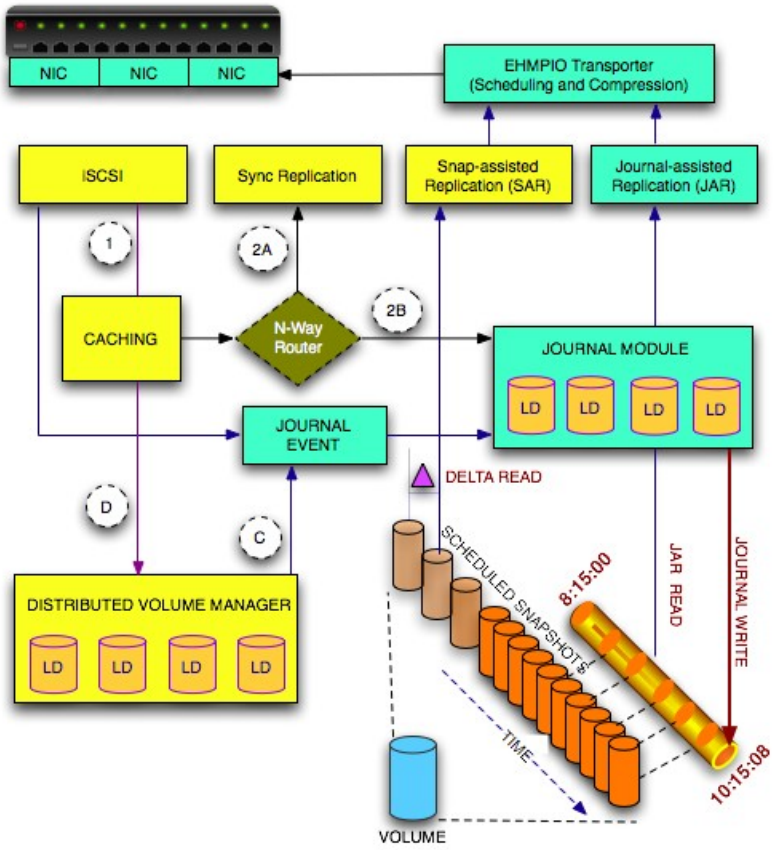

Figure 1: Journal-assisted replication architecture

StorTrends has a very intuitive wizard that guides the user through the steps of creating JAR with minimal mouse clicks. First, Logical Drives are configured to create 'journal-files' for the I/O staging repository. StorTrends allows online growth of 'journal-files' by either expanding the Logical Drive or by adding segments of different Logical Drives. Next, the replica group and replication schedules are configured.

Once the system is configured and enabled for journaling, incoming I/Os are routed through the N-Way router into the 'journal-file'. Each I/O is journaled densely by using a very space efficient meta-data. For performance enhancements, StorTrends journals different Consistency Groups in distinctive 'journal-files'.

The Journal Module exposes a journal event interface whereby external application server specific events (e.g. database checkpoints) and snapshot events can be also tagged in the log. In this way a true Time and Event Addressable Storage is implemented. From the figure it can be seen that the 'journal-file' is tightly coupled with the Snapshots to enable Continuous Data Protection (CDP) and Continuous Data Recovery (CDR). These 'journal-files' form the basis of various activities on the fine grained chronologically captured data. The JAR module is essentially an application layer that registers with the Journal Module as a client and replicates data out to the Recovery site according to an administrator defined schedule. Depending on the requirement, it could do continuous replication or a periodic replication. This module also has the intelligence to compress data (with varying depths of compression) and send them out for efficient link utilization. Data is sent out with meta-data information to the remote site so that the Recovery server may also maintain its own CDP log and/or create snapshots.

3. JAR: A deeper look

Journal Module is essentially an I/O staging layer, where it annotates the received I/Os with relevant meta-data information and stores the data in a repository. This module is central to the StorTrends core and offers various interfaces for other agents to communicate with. Innate to its nature, this module tags the I/Os with timestamps and thereby creates a "Time Addressable Storage". This, itself, would be sufficient to review and replay any historic data anywhere on the timeline. But recreating old data to any-point-in-time (APIT) may only result in Data recoverability and not necessarily application recoverability. As can be seen from the diagram, the Journal Module exposes a very simple event registration interface which can be used by other agents to tag the I/O stream with events holding relevance to the application. Snapshots, DBMS Checkpoints etc. are such events that get mingled with the stream maintaining the chronological dependence.

Apart from the Data staging or capture activity, the Journal Module also exposes data or Journal reference interfaces for other application specific tasks like CDP, CDR and CDI.

4. Seamless Asynchronous Replication

Once JAR is enabled and is in action, it faithfully replicates every incoming I/O to the recovery site allowing a continuous spectrum of recovery points. However, if there is a big disparity between the incoming and outgoing rates, then the Journal size increases and eventually a situation may arrive when the journal will overflow and the replication can not proceed. As seen from the diagram, simple flow control strategies are implemented by the stack to throttle the incoming pipe to mitigate this effect. If this first level of defense fails due to, say, extended periods of link failure then JAR agent seamlessly migrates to Snap Assisted Replication. There are other instances like Journal file failure, Volume Rollback etc. which also cause this transition. Eventually, over time, when the Journal catches up again, a reverse transition brings it back to JAR. Elaborate instrumentations are provided for monitoring and controlling the JAR operations. For example, a Journal can be manually stopped to push out the replication to SAR.

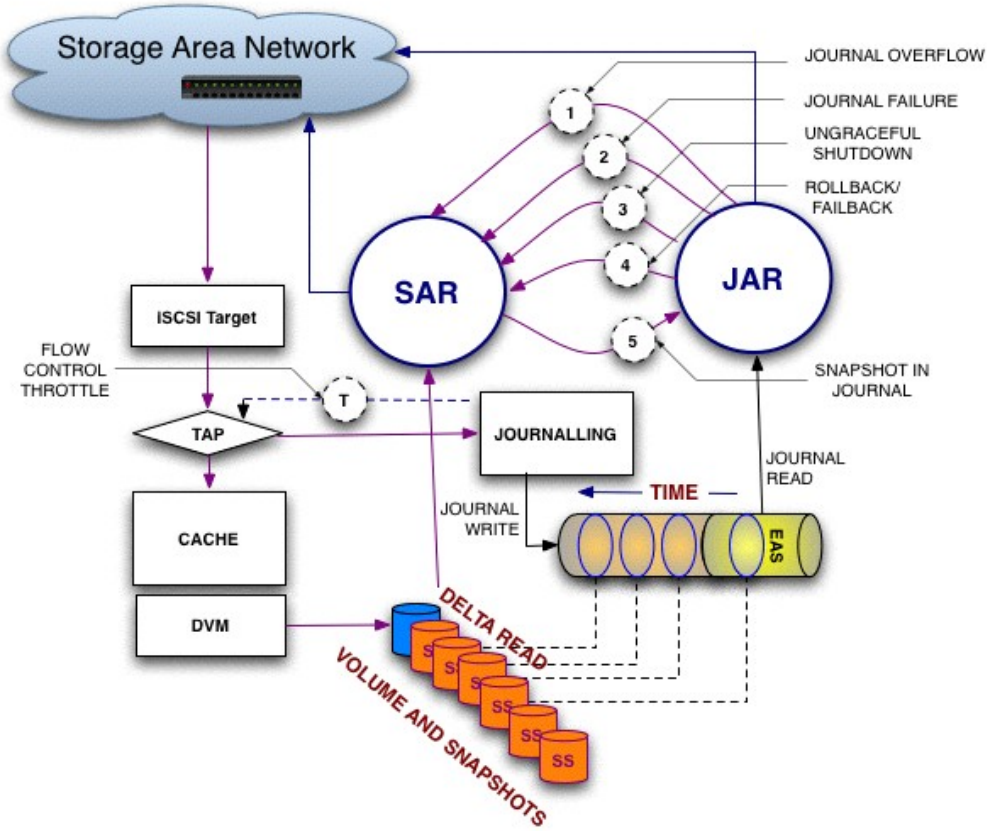

Figure 2: Seamless asynchronous replication transition between SAR and JAR

5. Flexible configuration of I/O path

StorTrends has a very powerful N-Way router module in the I/O path that can be configured to route the incoming I/Os through different paths.

For example on the recovery site, the I/O router can be configured to route the I/Os so that they are concurrently sent out to the recovery image volumes and the Journal repository. This, though, is a more traditional approach, suffers from I/O latencies. The reason is that the I/Os on the repository volumes are generally random in nature and thus take longer time to complete.

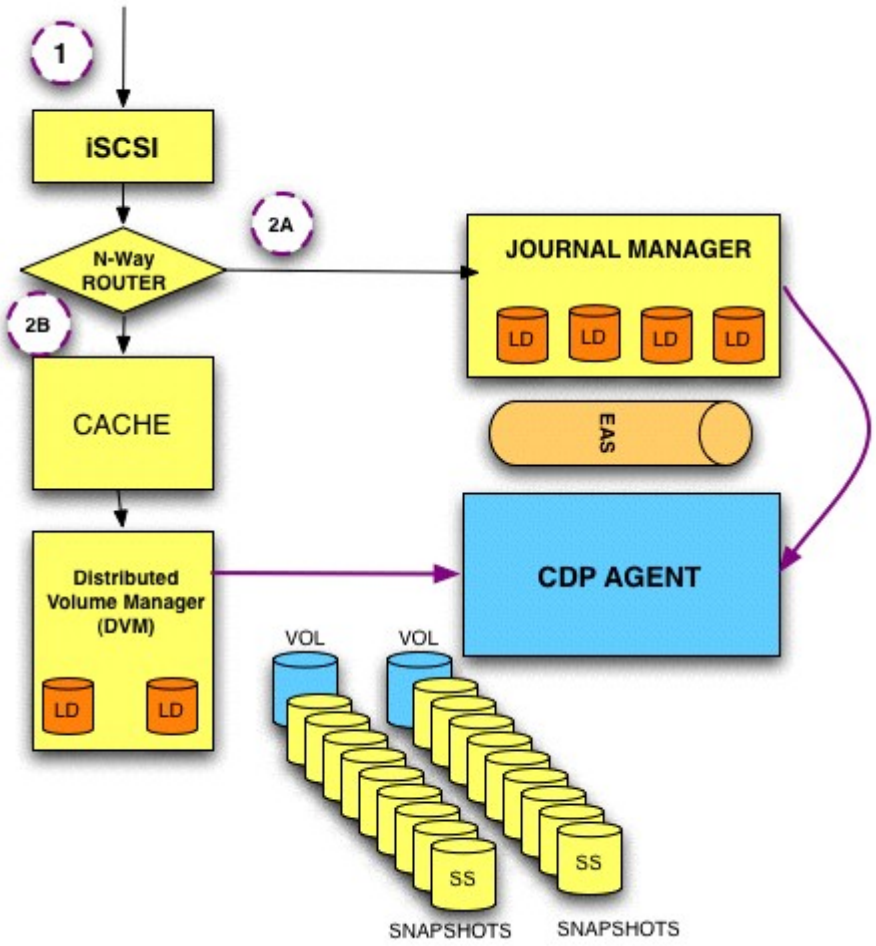

Figure 3:N-Way router used to hae a flexible I/O path

Alternatively, as depicted, The N-Way Router can be easily configured to route the I/Os only to the Journal repository. These complete very quickly because they are essentially sequential I/Os. The journal agent for Distributed Volume Manager (DVM) reads the journal in background and updates the recovery image volumes.

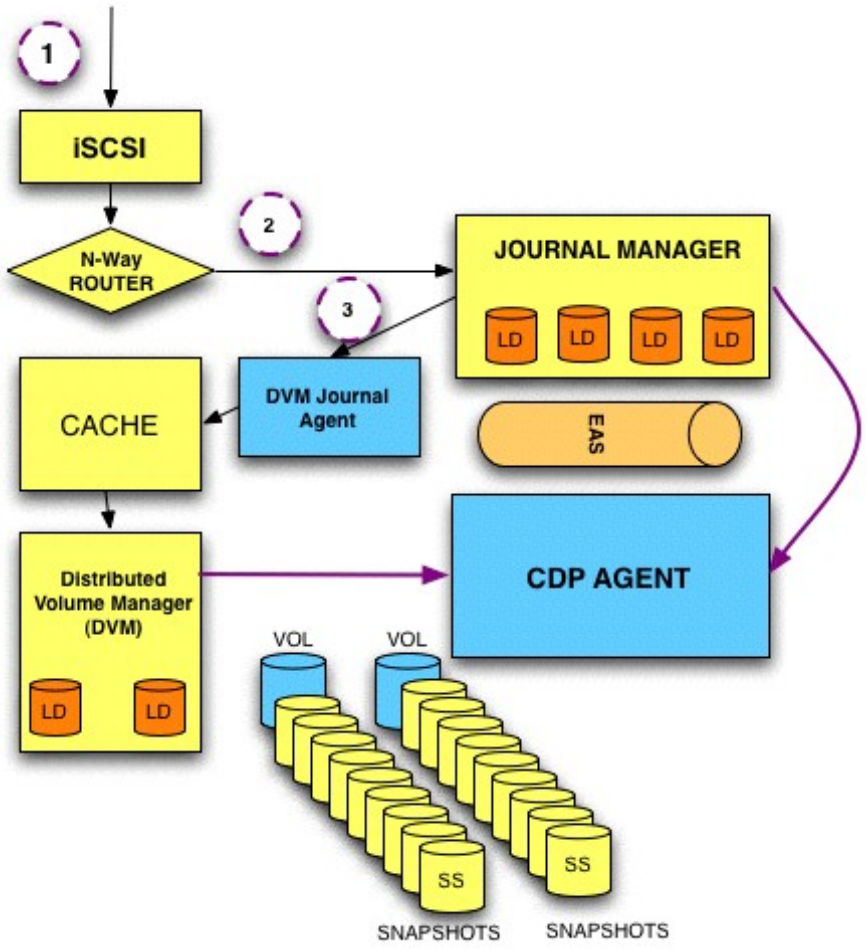

Figure 4: Performance improvement with N-Way Router

StorTrends JAR Advantages:

- Supports up to 32 consistency groups and Journal Volumes
- Journal volumes are extendible online and on the fly
- Extensive instrumentations for monitoring and control
- Supports both continuous and periodic replications
- Seamless migration between various modes of replications
- Optionally compresses data to offer link efficient replication
- Offers Link level scheduling
- Flexible path configuration for performance enhancements

TRADEMARK AND COPYRIGHT ACKNOWLEDGMENTS

FOR ADDITIONAL INFORMATION

Call American Megatrends at 1-800-246-8600 for additional information. You can also visit us online at ami.com.

Call American Megatrends International GmbH at +49 89 96 999 510 for additional information. You can also visit us online at ami.de.

LIMITATIONS OF LIABILITY

In no event shall American Megatrends be held liable for any loss, expenses, or damages of any kind whatsoever, whether direct, indirect, incidental, or consequential, arising from the design or use of this product or the support materials provided with the product.

LIMITED WARRANTY

No warranties are made, either express or implied, with regard to the contents of this work, its merchantability, or fitness for a particular use. American Megatrends assumes no responsibility for errors and omissions or for the uses made of the material contained herein or reader decisions based on such use.

REVISION HISTORY

15.3.2007 Preliminary release

12.12.2007 Revised version

More information: europe@ami.com - www.ami.de - www.ami.com