Remote Replication Technical Report – Dell PowerVault MD3 Storage Arrays

A White Paper

Dell Storage
May 2015

A Dell Technical White Paper

Table of contents

Overview

The remote replication feature for PowerVault MD3 storage systems efficiently replicates data between locations to make sure that information is protected from both system failures and site failures. This feature duplicates data between PowerVault MD3 storage systems to make sure data is available in the event of a disaster and for other business data migration needs. This document describes how the remote replication feature provides recovery points and group-oriented consistency across multiple virtual disks and supports efficient data transfers over FC and iSCSI (TCP/IP) networks.

Note: Remote Replication is supported only on Fibre Channel and iSCSI versions of PowerVault MD3 Storage Arrays.

The remote replication feature offers an option for customers who need disaster protection, but at a much lower cost than what is possible with synchronous remote replication models. The assumption is that such customers can relax their demand for an absolute up-to-the-instant-of-failure coherency plan as a tradeoff for the lower-cost option. In other words, remote replication allows customers to use slower and much less expensive communication links between the primary and secondary arrays, with the proviso that the primary and secondary arrays might have coherency delays that are not experienced when using the synchronous replication model. The extent of the coherency delays can be managed and tuned within the remote replication feature model to suit customer needs and to align with a wide variety of link latency levels and cost characteristics. One of the biggest benefits to the remote replication user is that application performance on the primary array is decoupled from the performance of the communication link. Link performance definitely affects the coherency delay, but it does not have any direct, significant impact on the performance level of the application.

Introduction

Customer needs addressed by Remote Replication

To understand the customer needs addressed by remote replication solutions, it is helpful to first consider the simpler approach called synchronous remote replication. In the synchronous model, one array holds the primary-side virtual disk of a replicated pair, and a second, remote array holds the secondary-side virtual disk of that pair. After the relationship between the primary and secondary virtual disks is established, the primary-side array processes host-initiated write operations on the primary virtual disk by propagating the write request to the remote array while performing the write operation on the local array as well. The primary array returns a successful completion indication to the host system only after both arrays finish processing the write operation.

The synchronous model has some very desirable benefits, the most important of which is the ability to recover application data content, including the most-recently updated content, from the secondary array. If the primary array suffers a catastrophic failure or disaster, application data on the secondary array can be recovered, and only the write requests that were outstanding during the failure cause content discrepancies. Because the application on the primary side would have still been waiting for positive acknowledgement of completion of the outstanding requests, the loss of those write requests is no different than if the primary-side host suffered a failure or a reboot with write requests in flight. Applications with even the most basic resiliency capabilities are able to recover their data to a consistent state after such events. Consequently, use of a synchronous remote replication model allows such applications to be made disaster-tolerant through remote replication.

In contrast to the benefits of its solid coherency characteristics, the synchronous model also has some substantial drawbacks. The most significant of those drawbacks are related to cost and performance. Because the primary array's response to any host-initiated write request is delayed until that request has been processed on the primary and secondary arrays, the bandwidth and latency characteristics of the inter-array communication link directly affect application performance. Use of a low-bandwidth or a high-latency link causes the primary-side application's performance to suffer. In other words, the performance of the primary-side application is gated by communication link performance. This consideration affects the cost model for the overall system. To obtain acceptable primary-side application performance, the customer must establish a high-speed communication link. This generally means a high solution cost because high-speed communication links are expensive to build or lease. For customers, there really are only two choices when synchronous remote replication is used:

1. Incur the expense of a high-performance, low-latency link to achieve acceptable primary-side application performance.
2. Use a lower-cost link that allows acceptable primary-side application performance. This typically means constraining the distance between primary and secondary sites to campus-wide levels. However, campus-wide distances imply a reduction in disaster tolerance because of the proximity of the primary and secondary arrays. A single disaster, for example, could be widespread enough to destroy both the primary and the secondary sites in such cases.

The primary objective of the remote replication feature is to enable remote replication over wide-area TCP/IP networks. Such networks offer the benefit of relatively low-cost, but are also extremely attractive from an administrative perspective, given their use of established, well-understood network infrastructure and switching equipment. By doing this in a way that allows primary-side applications to run without

network-induced performance limitations, the remote replication feature provides immense value to customers who require a disaster recovery solution.

Comparison to other PowerVault MD3 Remote Replication offerings

PowerVault MD3 storage offers two remote replication modes: remote replication and synchronous remote replication. The most significant difference between the two modes is that synchronous remote replication uses a synchronous coherency model as its fundamental mode of operation, but remote replication uses a fully asynchronous approach.

There is an option within the synchronous remote replication feature option called asynchronous mode, which is better described as being pseudo synchronous. When used, it simply builds up a persistent queue of outbound write requests from the primary to the secondary system and responds with a completion indication to the initiating host as soon as a write request is added to the queue. Asynchronous mode offers relief in some configurations in which link response time is uneven, but it does not address the core issue that is targeted by the new remote replication feature. Specifically, asynchronous mode still requires a relatively high-performance (and therefore a high-cost) link, but that link's latency characteristics can fluctuate over time. The buffering mechanism provided by the queuing of outbound requests allows such fluctuations to be tolerated. However, if the fluctuations are too severe, or if the overall link speed is insufficient for the application's write workload, the queue fills up. When that happens, the primary-side virtual disk's performance is completely gated by the link speed.

By comparison, the remote replication feature decouples primary-side virtual disk performance from link performance, and therefore enables the use of lower-cost links. Because the baseline assumption with remote replication is that coherency delays are expected and are likely to be fairly large, the remote replication feature provides a sophisticated method for restoring coherency at widely separated time intervals. Consequently, remote replication is a significantly different feature than the asynchronous mode within synchronous remote replication.

Both remote replication and synchronous remote replication use the host-side I/O ports on the storage array to convey replicated data from the primary side to the secondary side. With the synchronous remote replication feature, the only supported interface type is Fibre Channel (FC). The new remote replication feature operates over either FC or iSCSI front-end connections. Because remote replication is intended for higher-latency, lower-cost networks, iSCSI and connections based on TCP/IP are a good fit. As with the synchronous remote replication feature, remote replication over FC requires that one front-end FC port on each controller be dedicated to remote replication traffic; no host I/O requests are allowed on those ports. When remote replication is used in iSCSI environments, it is not necessary to dedicate any of the array's front-end iSCSI ports to remote replication; those ports are shared between remote replication traffic and host-to-array I/O connections.

General aspects of Remote Replication feature operation

The remote replication feature is intended to operate in configurations with relatively high-latency and/or low-bandwidth communication links (TCP/IP links for iSCSI, for example). Its fundamental mode of operation involves configurable periods of noncoherency during which no attempt is made to propagate write data from the primary array to the secondary array. Instead, by using a *dirty map*, the array tracks which portions of the primary virtual disk have been modified so that they can be selectively transmitted to the secondary array later. Eventually, when the configured time period has elapsed, the primary array begins a background process that carries out the relevant data transfers to the secondary array.

This approach offers the following benefits, each of which is especially important in cases in which a higher latency, WAN-oriented communication link is used:

Performance. During a noncoherency interval, no attempt is made to propagate changes across the link. Consequently, the performance on the primary array is not affected by the communication link characteristics. Rather, the only performance impact is due to the relatively small additional latency associated with tracking which blocks have changed. This is done by updating the dirty map on the primary array. During resynchronization operations, using a point-in-time image allows data replication to occur without introducing any significant dependencies of primary virtual disk I/O rates on replication link speed or latency.

Efficiency. Because many updates to the same data blocks can occur while changes are accumulated during a noncoherency interval, the remote replication feature can avoid transfer of the intermediate contents. Only the block content that exists when the resynchronization action begins is actually sent over the link.

Reliability. Careful consideration is given to make sure that the overall block content of a primary virtual disk is in a valid, recoverable state prior to initiating a resynchronization operation with the secondary array. Specifically, the remote replication feature verifies that the block content selected for a primary virtual disk's resynchronization cycle reflects a write ordering that was actually observed on the primary array. As a result, the user can be certain that the data images that are captured on the secondary array are usable and can be recovered in situations in which the secondary virtual disk must be elevated into active use.

Simplicity. In a scenario in which the secondary virtual disk is elevated into active use because of a problem or disaster at the primary side, it is possible that the primary array might later be restored to an operational status. This could occur if the problem or disaster caused the primary side to be cut off from its communication links, but the outage was later corrected. In such cases, the remote replication feature supports an optimized, deltas-only resynchronization back to the original primary side. In other words, only the blocks that changed since the activation of the secondary virtual disk are transferred back to the original primary virtual disk. Therefore, it is possible to restore the original primary to its preferred role without an expensive complete resynchronization.

Distance. Remote replication supports both FC and iSCSI networks. iSCSI leverages standard IP networks to replicate data over much greater distances than typical FC configurations. In addition, remote replication does not require a dedicated host port, which allows general host-initiated I/O operations replication activities to be mixed on the same iSCSI ports.

Operational model

Virtual disk-level Remote Replication configurations

The remote replication feature allows individual virtual disks on one array to be bound into paired relationships with virtual disks on other arrays. A given virtual disk can only participate in one remote replication (or synchronous remote replication) relationship. However, multiple virtual disks on an array can be configured for remote replication, with each virtual disk having a designated partner virtual disk on a remote array. The maximum number of virtual disks that can be configured for remote replication is 32.

Every paired remote replication relationship involves one virtual disk in a primary role and a second virtual disk (on a different array) in a secondary role. A given array can have some of its virtual disks assigned to primary roles and others assigned to secondary roles, each within its own respective pair.

Furthermore, a given array can have some of its remote replication virtual disks paired up with partners on a second array, while others are paired with partners on a third array, and so forth. It is not necessary for all of a given array's remote replication-enabled virtual disks to be paired with partners from a single remote array.

General aspects of resynchronization

Because remote replication is a replication feature, the heart of its implementation is focused on maintaining a copy of a primary-side virtual disk on a remote, secondary-side array. Such copies are maintained by periodically propagating changed blocks from the primary array to the secondary one in a process called resynchronization.

Some of the most important points regarding this recurring resynchronization activity include:

The time interval between the start of successive resynchronization actions is referred to as the synchronization interval, which can be specified in minutes. The minimum value is 10 minutes. This attribute is independently set for each replicated virtual disk pair.

An administrator can also configure any replicated pair so that automatic (interval-based) resynchronization does not occur, in which case all background resynchronizations must be initiated manually through explicit administrative action. Automatic synchronization is the preferred mode of operation in most cases.

When it is time for a resynchronization, the primary-side array uses an embedded snapshot process to create a snapshot of the primary virtual disk. This snapshot is used as the source that feeds the background data transfer process. Creation of a snapshot serves two purposes:

1. It forces all in-flight, host-initiated write requests for the primary-side virtual disk to be completed immediately, and any newly arriving write requests to be queued until after snapshot creation is complete. This process establishes a recoverable image that is captured in the snapshot. The data content of the snapshot is considered recoverable because any outstanding write requests, such as those queued during snapshot creation, were submitted to the array by the primary-side application with full knowledge that they could all be in flight concurrently. This implies that the

application is capable of recovering from a scenario in which any or all of the writes were prevented from completing for any reason (for example, a host reboot).

2. Because the data to be transmitted is captured in a snapshot, it is possible for ongoing host-initiated write requests to begin flowing to the primary-side virtual disk during the resynchronization process. Those updates do not affect the content of the data being propagated to the secondary virtual disk because the propagation is based on the snapshot content.

When a background resynchronization process begins, the primary-side array starts using a new (and initially empty) dirty map, which then tracks updates made to the primary virtual disk after snapshot creation. This new dirty map drives the update process in the next coherency interval, that is, after the current interval's resynchronization activity is complete.

A significant benefit of using this interval-based approach is that overwrites of data blocks during the noncoherency interval do not result in increased resynchronization activity later. Regardless of the number of overwrites that occur during such an interval, it is only the content of the affected blocks at the end of the interval that is transmitted across the link.

To keep the data transfer sizes small during resynchronization processing, the remote replication feature uses relatively small granularity for tracking dirty blocks in its dirty map. Each dirty indicator in the map covers a 4KB chunk of the primary virtual disk's logical block addressing space.

The additional latency incurred on the primary-side virtual disk in a remote replication pair is due solely to primary-side activities. In particular, the ever-present management of the dirty map and the management of the snapshot that exists during a synchronization interval add some latency to I/O operations (specifically to writes) on the primary virtual disk. However, the magnitude of these impacts is independent of the performance of the communication link.

A typical operational sequence for the remote replication feature is illustrated in Figure 1 Sample remote replication timing diagram. At the beginning of a synchronization interval, a snapshot is established to capture the content of the primary-side virtual disk at that instant so that relevant (changed) blocks can be transferred to the secondary array. Such snapshots exist only until the resynchronization process completes for that interval. A new snapshot is created in the ensuing interval as the cycle repeats.

Figure 1 also illustrates that one dirty map exists at all times and is actively updated to track the blocks that are being modified on the primary side. At the end of a synchronization interval, that dirty map is frozen and is then used to determine which primary-side data blocks must be transferred to the secondary array. After the data transfer (the resynchronization) completes, the dirty map is discarded along with the snapshot that was created when that dirty map was frozen. Furthermore, when the aforementioned dirty map is frozen, a new dirty map is established to track updates that continue to occur on the primary side that must be propagated to the secondary array in the next synchronization interval.

In the portion of Figure 1 that represents elements on the secondary array, the secondary virtual disk alternates between periods of inconsistency (not recoverable) and periods of consistency (recoverable).

The inconsistent, unrecoverable periods are colored black, and the consistent, recoverable periods are colored white.

The reason the secondary virtual disk might temporarily become inconsistent is because updates from the primary array to the secondary, which occur during a resynchronization process, do not preserve the original host-initiated write ordering that took place on the primary side. In fact, overwrites of a given

block are all collapsed into a single primary-to-secondary update when resynchronization takes place. As a result, there is no assurance of consistency or recoverability for the secondary virtual disk until the resynchronization process completes. At that point, the secondary virtual disk is brought to the same consistent point that was established with creation of the snapshot on the primary array at the start of the synchronization interval.

The obvious question, given this period of inconsistency, is: How do we protect against primary-side disasters that might occur during the resynchronization process that would leave the secondary array in an unusable state? The answer is the remote replication feature creates a snapshot of the secondary virtual disk every time a resynchronization process completes successfully. Therefore, that snapshot is available for recovery in the event that the next resynchronization process fails to complete and leaves the secondary virtual disk in an inconsistent state. If the secondary virtual disk is required for recovery purposes, its content can be rolled back by using the previously established snapshot, effectively discarding the (partial or incomplete) updates that took place during the aborted resynchronization process. In Figure 1 these protective snapshots on the secondary array are shown in green, and they are labeled with the timestamp associated with the primary-side content that was captured in the snapshot.

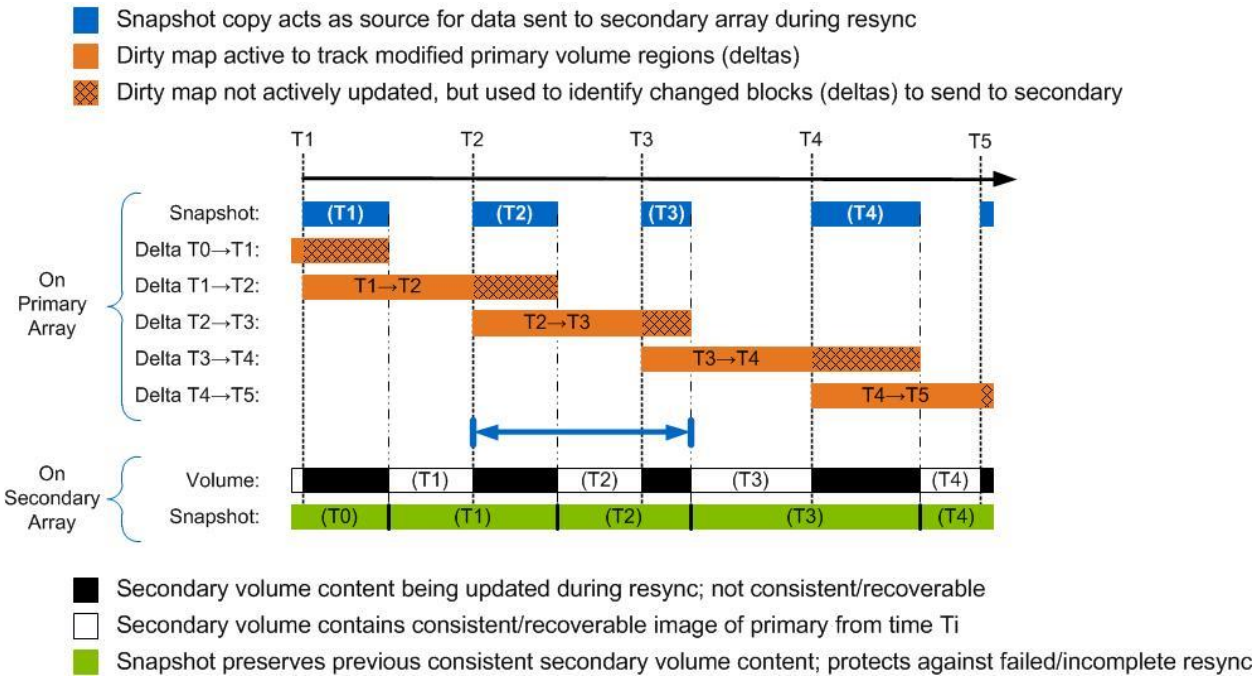

Figure 1 Sample remote replication timing diagram

The time between the start of successive synchronization intervals (for example, from T1 to T2, T2 to T3, and so forth) is clearly a key parameter in this model. However, from a recoverability standpoint, the critically important time interval is the amount of time that elapses from the beginning of a synchronization interval to the end of that interval until all relevant updates have been propagated to the secondary site. A sample interval of this type is shown in Figure 1 by the blue line, which originates at T2 and continues until a point beyond T3, that is, until the update propagation (or resynchronization)

operation completes for that interval.

Why is this time interval so important? The time interval is important because it represents the amount of time during which updates made on the primary side are susceptible to loss in the event of a primary-side disaster. Consider a write operation that takes place immediately after T2. No attempt is even made to propagate that data to the secondary site until T3 arrives. Furthermore, starting at T3, some additional time is required to transmit all of the affected data blocks to the secondary array. The new data is not truly usable at the secondary side until the entire resynchronization process completes, at which time the secondary array possesses that modified data as part of its newly established image of the T3 version of the data (as shown in the secondary-side snapshot labeled T3).

The blue line shown for the example interval (beginning at T2) covers both the solid and the cross-hatched portions of the orange bar for that interval. In fact, for every resynchronization cycle shown in the diagram, it is the combination of the solid and cross-hatched portions of the respective orange bars that is of interest to us. It is also important to note that the resynchronization times, as illustrated by the cross-hatched portion of each bar, can vary substantially from one interval to another.

From an administrative standpoint, the storage system user typically wants to specify an expectation regarding the worst-case time that updates made on the primary array remain susceptible to loss in the event of a disaster. In industry parlance, this expectation is called the recovery point objective (RPO). Based on the remote replication design described here, the RPO specification and the synchronization interval have clear dependencies on each other. The following observations are relevant:

1. The RPO is the administrator's objective for the upper bound of the amount of time during which changes made on the primary side are subject to loss in the event of a primary-side disaster.
2. The synchronization interval is the amount of time that primary-side changes are accumulated without any attempt at propagation to the secondary side.
3. The actual resynchronization time (ART), indicated by the cross-hatched bars in Figure 1 can vary from one interval to another. That time depends on network conditions (such as latency) and also on the amount of changed data that has accumulated during the synchronization interval. As long as the (fixed) synchronization interval plus the (variable) ART for a given cycle is less than the (fixed) RPO, then the system is working as desired. However, if that sum is greater than the RPO, presumably due to an excessively long ART, then the administrator's objective is not met for the interval in question.

Actions taken when a replicated pair is established

When a replicated pair is established, several actions take place to enable ongoing management of the resynchronization actions on both the primary and the secondary array. Most importantly, a replication repository virtual disk is created on each array and is thereafter used to hold the following important information for remote replication operations.

For the virtual disk that is in the primary role, the replication repository virtual disk contains:

1. The dirty map that is constantly updated to track which blocks have been written in the current interval, that is, the blocks that must be propagated to the secondary array in the upcoming resynchronization operation.
2. The dirty map that is no longer being updated, but which contains information about the blocks that are currently being transferred to the secondary array as part of an active resynchronization operation. After the resynchronization completes, this dirty map is discarded.

Capacity for saving copy-on-write data associated with the snapshot that captures the data content that is currently being transferred to the secondary array as part of an active resynchronization operation. For example, at T2 in Figure 1, the resynchronization process establishes a snapshot that captures a frozen image of the primary virtual disk. This image is used in conjunction with the dirty map that is no longer being updated to transfer only the blocks that were modified between T1 and T2 to the secondary side. The Snapshot is maintained using a copy-on-write algorithm, so blocks that are updated on the primary array after T2 have their old (T2) content saved to the replication repository virtual disk to make sure that the blocks are available when they are required by the copy engine.

For the virtual disk that is in the secondary role, the replication repository virtual disk contains capacity for saving copy-on-write data associated with the Snapshot that captures the most recently completed resynchronization from the primary side. This Snapshot is required because the next active resynchronization operation will begin to modify the secondary virtual disk, but in ways that do not necessarily result in a recoverable (consistent) image until the entire resynchronization operation completes. If a disaster occurs that interrupts the resynchronization, and the resynchronization cannot complete, it must be possible to revert back to the preresynchronization image of the secondary virtual disk to restore a recoverable and usable image. The Snapshot is the recoverable and usable image. It is maintained using copy-on-write algorithms, therefore a repository is required to save blocks before they are overwritten during the resynchronization operation. If a resynchronization operation completes successfully, a new Snapshot is established and old copy-on-write data is discarded because the secondary virtual disk now has a new, recoverable image that must be preserved for the resynchronization operation.

Resynchronizing during steady state

The steady-state mode of operation for the remote replication feature is depicted in Figure 1**.** It illustrates how synchronization times can vary from one interval to another. However, as long as the times are acceptably short, there is a very regular pattern of update completion, and recoverable images at the secondary side are established.

Special handling for initial synchronization

During the initial synchronization that occurs when replicated pairs are established, a special form of processing takes place to avoid potential problems with Snapshot management on the primary array.

In a perfect world, the initial synchronization operation would operate just like any other resynchronization, except that the primary array would consider all data blocks to be dirty or in need of transfer to the secondary array. In other words, a Snapshot is established on the primary side and used to capture the stable image that is conveyed to the secondary side. However, because the initial resynchronization must transfer all of the primary virtual disk's blocks, and the remote replication communication links are expected to be relatively slow, it is likely that the initial synchronization will require significantly more time than a typical future resynchronization.

Because maintaining a Snapshot on the primary side for a long time can result in significant capacity demands for copy-on-write data, the initial synchronization interval is susceptible to repository overflow conditions for that snapshot unless an extremely large repository is created. Rather than imposing such a capacity requirement, the remote replication feature instead treats the initial synchronization of a replicated pair in a special way. Specifically, no Snapshot is created on the primary side. Instead, the active (and presumably nonrecoverable) content of the primary virtual disk is used to drive the initial resynchronization process. However, the standard dirty map mechanism is used to track the changes that

occur on the primary side during the initial synchronization process. When this process eventually completes, the content of the secondary virtual disk is presumably close to the content of the primary virtual disk, and it clearly has the desired recovery outcome. The remote replication implementation marks the secondary virtual disk with a special state indicator to track this temporary condition. The ensuing resynchronization (after the initial one completes) then operates in the usual way. Specifically, a Snapshot is created to capture the content at the start of that synchronization interval (T2). Because the dirty map was maintained in the usual way on the primary side, all primary blocks that were modified either during or after the initial resynchronization (but before T2) are transferred to the secondary array. The resulting secondary virtual disk image on the secondary array is the recoverable, Snapshot-preserved T2 image of the primary virtual disk. At that point, the special state indicator is removed, and the replicated pair moves into a steady-state operational mode.

The one relevant side effect of using this approach is that the initial resynchronization operation does not produce a recoverable image at the secondary array. However, that condition is quickly remedied by the second resynchronization operation. This is a small price to pay for the benefit of not requiring an excessively large repository to support the initial resynchronization operation.

Groups of related virtual disks

In many cases, the storage application running on host systems uses multiple virtual disks within a given array. For example, some common database-oriented applications might have some virtual disks that are used for write-ahead logs and others that are used for table storage. In such applications, it is very important that all virtual disks related to that application be replicated to the secondary site in a way that makes group-level recoverability possible. Specifically, the propagated images of all primary-side virtual disks must represent a valid write ordering across the entire set.

To achieve this objective, the remote replication feature treats all replication configurations using a grouping concept; such groups are called remote replication groups (RRGs). Even if only a single virtual disk is initially replicated, the administrator sets up an RRG for that virtual disk. Later, if more virtual disks must be added to the RRG, it is easy to do so.

The RRG is the focal point for establishing many of the properties that govern synchronization activity for the remote replication feature. From a multi-virtual disk write-ordering perspective, the key aspect of an RRG's operation is that all virtual disks within the RRG are drained of I/O activity prior to creating their Snapshot images that feed the resynchronization activity for each interval. This process makes sure that all virtual disks in the RRG can proceed to a precise write-order point that is then captured and propagated to the secondary array.

In some environments, the concept represented by an RRG is referred to as a consistency group. However, in the PowerVault products, the term consistency group is used for snapshot and snapshot (legacy) concepts that are unrelated to the remote replication feature.

Establishing an appropriate RPO value

The RPO value for a replicated pair is an indication of the storage administrator's tolerance for lost updates that might occur during a primary-side outage. For example, an RPO of 15 minutes essentially indicates that it is acceptable for a propagation delay of up to 15 minutes from the time a block is written on the primary virtual disk until that block is reflected in a recoverable image of the virtual disk on the secondary array. In a perfect world, an administrator would be free to select an RPO that is extremely small to make sure that only a minimal number of updates are lost during an outage or disaster. However, extremely

small RPOs are generally associated with very fast (and therefore very expensive) primary-to-secondary links. In fact, a synchronous RVM model is generally the only way to achieve an RPO of zero, though it clearly has performance implications even in cases where the communication link is very fast. The remote replication feature's primary value is that it allows the administrator to choose the desired RPO, based on the cost metrics of the available communication links, and configure the storage array to operate within the selected boundaries. Ultimately, the administrator must select a synchronization interval that is appropriate for the chosen link configuration.

Another critical factor to consider is the block update behavior of the application that is running on the primary array. Both of these attributes impact the amount of time that is required to complete a resynchronization operation. Obviously, an application that changes a large number of data blocks during each synchronization interval induces a fairly large load on the resynchronization process, and it drives the resynchronization time higher.

Although there are a large number of factors to consider when establishing an RPO value, the following process can be used as a starting point for an administrator who is configuring this attribute for a replicated pair in remote replication:

1. Start by selecting a value for the RPO that is at the high end of the comfort zone for the acceptable amount of time between a primary side update and its propagation to the secondary side. For example, if the administrator can tolerate 60 minutes of lost updates in the event of a disaster, even though that might be really uncomfortable, use 60 minutes as a starting point. For the following steps, assume that 60 minutes is the selected value.

2. Establish an initial synchronization interval value for the replicated pair that is approximately 0.7 times the RPO. In this example, set the synchronization interval to 42 minutes. After this is done, the primary array accumulates changes for 42 minutes before attempting to propagate them to the secondary side. After 42 minutes of accumulation, it engages its resynchronization process. That process takes some time to complete, and until it does, the accumulated changes (even those made at the very beginning of the accumulation interval) are unavailable at the secondary side. For example, if the resynchronization process takes 10 minutes to complete, the overall window of vulnerability for that particular cycle would be 52 minutes. Each resynchronization process varies in its time requirements because there is a dependency on the application update intensity, the link conditions, and even on the ensuing I/O load on the primary and secondary arrays, which can impact the resynchronization processing.

3. Allow several synchronization intervals to elapse with this initial setting, and then invoke the remote replication feature's mechanism for displaying resynchronization history and statistics by using the PowerVault Modular Disk Storage Manager (MDSM) application.
    a. If the historical view indicates that resynchronization operations are completing fairly quickly (relative to the initial synchronization interval), it is an indication that a smaller RPO might be achievable. Suppose the maximum observed resynchronization time is 10 minutes. In this case, the administrator might add a 50% contingency and expect that all future resynchronizations should complete in 15 minutes at most. That establishes an upper bound for the resynchronization time and a lower bound for the achievable RPO. In general, picking an RPO that is twice the upper bound resynchronization time is a reasonable next step. Therefore, the refined RPO in this case would be 30 minutes. The synchronization interval can then be adjusted to the RPO value minus the upper bound resynchronization time, or 30 - 15 = 15 minutes. Keep in mind that the minimum allowed value for the synchronization interval is 10 minutes, regardless of the observed performance of resynchronization operations.
    b. If the historical view indicates that resynchronization operations are completing fairly slowly

(relative to the initial synchronization interval), it is an indication that a larger RPO must be chosen. Suppose the maximum observed resynchronization time is 50 minutes. In this case, the administrator might add a 50% contingency and expect that all future resynchronizations would complete in 75 minutes at most. That establishes an upper bound for the resynchronization time and a lower bound for the achievable RPO. Picking an RPO that is twice the upper bound resynchronization time is a reasonable next step. Therefore, the RPO in this case would be 150 minutes. The synchronization interval can then be adjusted to the RPO value minus the upper bound resynchronization time, or 150 - 75 = 75 minutes.

4. Allow several more synchronization intervals to elapse, and then repeat the analysis from step 3. With more observations available, it might be possible to reduce the 50% contingency to a smaller value.

5. If the refined RPO value is larger than desired, it is possible to select an RPO that is less than twice the upper bound resynchronization time. However, there are some important points to keep in mind about configurations in which the RPO value is fairly close to the upper bound resynchronization time:
   a. In such configurations, the resynchronization process runs almost continuously, and this means that a snapshot exists on the primary array almost constantly because it is required to drive the resynchronization process. The presence of a Snapshot imposes a certain amount of overhead on the primary virtual disk. This is generally tolerable, but in cases in which maximum performance is desired, it has an impact that must be considered.
   b. Any unexpected additional latency in a given resynchronization activity triggers a *needs attention* condition on the array, assuming the recommendations provided in step 7 are accepted and a threshold is established for alert generation. This requires administrative attention each time the threshold is exceeded.

If the refined RPO value is well within the desired range, consider increasing it somewhat by selecting a slightly larger synchronization interval. Doing so increases the amount of time during which changes accumulate on the primary side and during which no primary-side snapshot must be in place. This can be performance benefit for the primary virtual disk.

Note: Increasing the synchronization interval can also cause the average resynchronization time to increase because it might result in more updated blocks that must be transferred.

7. After a reasonable configuration is established, configure the replicated pair's synchronization warning threshold value to be RPO - synchronization interval. This makes certain that an alert is raised any time the replicated pair fails to achieve the RPO for a given synchronization interval.

Note: The initial synchronization that occurs immediately after establishing a paired relationship between virtual disks operates much differently than the successive normal resynchronizations, therefore its behavior is not relevant for this process. Be sure to allow that initial synchronization to complete before you begin this RPO determination process.

Resolving resynchronization issues

The remote replication feature is designed to withstand a wide variety of communication link problems and other resynchronization challenges. However, it is possible for problems to arise due to a number of different causes, including physical system failures, long-term link outages, and usage changes that drive new resource demands for remote replication. Some potential problem scenarios are described in the following sections, along with recommendations for how to resolve them.

Synchronization warning threshold exceeded

When the synchronization warning threshold is exceeded, the primary-side array generates a critical Major Event Log (MEL) entry that triggers an alert through any configured delivery mechanism, such as SNMP traps, e-mail, and so on. It also triggers a persistent needs attention condition for the array, and this means that administrative intervention is required to clear the condition. The reason for the alert is that the replicated pair was unable to achieve its desired RPO for the interval in question, and this is an important condition that the administrator must know about. The administrator can clear the needs attention state for the replicated pair by using the MDSM application.

When such conditions arise, Dell recommends that the administrator view the resynchronization history and statistics for the replicated pairs in question to obtain insight into the nature of the conditions that caused the threshold to be exceeded. Some common causes are an overly aggressive RPO value, an unreliable or intermittent remote communication link, an unexpectedly large amount of change activity on the primary-side virtual disk, excessive performance load on the remote virtual disk, and so forth. In some cases, the cause of the problem can be deemed intermittent or rare, in which case the needs attention condition can simply be cleared without further action. Otherwise, the synchronization warning threshold (and therefore, the RPO) should be increased.

Synchronization process extends into next synchronization interval

If a synchronization process extends into the next synchronization interval, it is clearly an indication that the desired behavior is not being achieved. The remote replication feature does not generate an explicit alert for such conditions because a properly configured synchronization warning threshold causes an appropriate alert. In such cases, the synchronization for the interval following the excessively long one is simply skipped because it could not be started at the desired point in time. The remote replication implementation looks for its next regularly spaced synchronization interval after fully processing the previous interval, regardless of how much time that previous interval required.

Primary-side snapshot repository overflow

A snapshot is created for each primary virtual disk at the beginning of a synchronization interval, and it is used to feed the data transfer process that propagates changed data to the secondary array. If the snapshot repository for a primary virtual disk is too small to accommodate copy-on-write data that accumulates during the synchronization process, the propagation must be terminated for that interval. In such cases, all progress that had been made in sending updates is retained, but the secondary virtual disk does not achieve its desired consistent image. The secondary side snapshot that was created at the end of the preceding synchronization interval is still available, so there is a recoverable image (albeit one that is older than desired) on the secondary array even after this problem occurs.

An administrative alert is delivered if this problem occurs, and a needs attention condition is raised for the array. The remote replication implementation keeps track of the portions of the primary virtual disk that

could not be propagated due to loss of the snapshot image, and those blocks simply add to the workload that must be completed in the ensuing synchronization interval. It is possible that the resynchronization in the ensuing interval will work and that the secondary virtual disk will achieve its next desired recovery point, but the fact that a snapshot overflow occurred is still a concern. In such cases, the administrator should typically increase the capacity of the primary-side snapshot repository for the virtual disk that encountered the problem because this helps reduce the likelihood of future recurrences. If some unusual activity occurred on the primary that triggered excessive updates during the problematic time window, and that activity is not expected again, the administrator can simple dismiss the needs attention condition without increasing the capacity of the snapshot repository.

Secondary-side snapshot repository overflow

Every time a synchronization interval completes successfully, a new snapshot is created on the secondary array. This snapshot provides a recovery point in cases in which the ensuing synchronization process encounters a problem and is only partially completed. Consequently, when updates are propagated from primary array to secondary array during a given synchronization operation, the secondary array's snapshot accumulates copy-on-write data that protects the snapshot. That snapshot is treated with great care because it represents the most recent recoverable image of the primary virtual disk on the secondary array. If the updates associated with the synchronization operation cause an overflow of the secondary array's snapshot repository, then those updates are rejected. This causes the synchronization process for the virtual disk in question to fail for the given synchronization operation. Furthermore, all ensuing resynchronization operations immediately encounter the same problem, so until the condition is resolved, there is no reason to try additional resynchronizations.

This scenario triggers a repository full condition on the secondary array, which includes an administrative alert and a needs attention state there. It also causes other replication operations for the primary virtual disk to be suspended, which drives an alert and a needs attention state on the primary array. The administrator must respond by increasing the capacity of the snapshot repository on the secondary array because this allows other replication operations to resume. When they do resume, they pick up where they left off by propagating only the necessary (modified) blocks from the primary to the secondary array.

Synchronization interrupted by catastrophic event on primary

It is possible for a synchronization operation to be active when a catastrophic event occurs on the primary side (for example, a natural disaster that destroys the primary array). In such cases, the secondary virtual disks will likely be in an intermediate state that reflects some, but not all, of the necessary updates from the current synchronization interval. However, the secondary-side snapshot images established at the end of the previous synchronization interval still contain fully recoverable data that can be used when activating the secondary side as a recovery site. Refer to section 4.4 for details.

Administrative management tools

Synchronization warning threshold

The synchronization warning threshold is a valuable tool that allows the administrator to specify the expected RPO behavior for remote replication configurations. When this time threshold is exceeded, it triggers an alert notification (through e-mail, for example) and drives a needs attention state for the array. By setting an appropriate value, the administrator can receive notifications when anything unexpected occurs that results in the failure to achieve the desired RPO.

Synchronization history and statistics

This tool allows the administrator to view key characteristics of the synchronization history for each configured remote replication virtual disk.

Statistics are captured for replicated virtual disks in the primary role. The synchronization process data that is stored for administrative viewing includes the following information:

1. Sync start time
2. Number of bytes sent
3. Maximum write time (for a single write)
4. Minimum write time (for a single write)
5. Minimum synchronization data rate
6. Maximum synchronization data rate
7. Total write time
8. Primary snapshot repository utilization (%)
9. Recovery point age
10. Configuration settings:
    a. Role
    b. Synchronization interval ordinal
    c. Virtual disk ownership
    d. Synchronization interval time
    e. Synchronization warning threshold
    f. Repository utilization threshold
11. Synchronization duration

In addition, several different batches of these statistics are maintained that correspond to conditions that are of particular interest to administrators including:

12. Statistics for the most recent 50 resynchronization samples.
13. Statistics for the 20 longest resynchronization samples. This set of statistics can be cleared by the administrator to trigger the collection of a new batch, starting at the next synchronization interval.
14. Statistics for the most recent 20 failed resynchronization samples. These samples include a failure code, such as, primary snapshot full, secondary snapshot full, synchronization extended beyond the synchronization interval, and so on.

In cases in which timing problems or other synchronization errors occur, it can be extremely helpful to the administrator to view this information by using the MDSM application. Doing so can give solid insights into the nature of any problems and the actions required to address them.

Orderly role reversal

The remote replication feature provides a mechanism by which the roles (primary and secondary) in replication relationships for RRGs can be reversed in an orderly fashion. In this operation, the primary and secondary arrays carefully sequence their synchronization and role-change activities so that each secondary virtual disk can be promoted to a primary role with full confidence that it possesses the most up-to-date images of the original primary virtual disk. Such actions typically take place if the administrator is conducting some form of scheduled maintenance on the primary site and wants to activate applications and storage at the secondary site for continuity of service. Later, when that maintenance completes, the roles can be reversed again to restore the original primary site to normal operations. Again, there is a carefully orchestrated transition process that makes sure that the restored primary site's virtual disks contain the latest updates that occurred while roles were reversed during the maintenance interval.

Forced role reversal

In cases in which some form of catastrophic event occurs at the primary site, it becomes critically important to activate the secondary site for continuity of service. Because communication between primary and secondary sites is typically not possible under such conditions, the administrator must have the ability to force a promotion of the secondary virtual disks into primary status. Remote replication provides this ability. The key difference between orderly and forced role reversal is that the latter can occur without any communication or coordination between the primary and secondary sites.

When a forced role reversal occurs, the remote replication implementation on the secondary array first determines if a synchronization operation is currently in flight. If so, it is terminated, and the preserved snapshot images that represent the last known consistent state are used in a snapshot rollback operation. This causes all partial updates made to the secondary virtual disk during the aborted synchronization operation to be discarded.

Note: The rollback data transfers occur as a background operation. The secondary virtual disk is immediately available for host I/O while those transfers are in progress, and any host system that accesses data through the secondary array can see the desired image, even though the necessary data transfers are still taking place in the background.

Upon committing the rollback operation for the secondary virtual disks, the remote replication feature promotes those virtual disks into primary status, at which point they begin operating just like a normal primary remote replication virtual disk. However, the original primary array is unaware of this change. If the original primary array was destroyed, this lack of awareness is not a concern. However, if it was isolated or partitioned and later becomes accessible for communication with the newly established primary array, then both sides recognize that a dual-primary conflict exists.

Presumably, all customer services have transitioned to the secondary site, so the administrator knows that the promoted secondary (now the primary) is the true master. In cases in which the original primary array is actually intact, the administrator might eventually want to transfer responsibility back to it as the primary site.

The remote replication feature allows this to occur and also makes sure that only the necessary updated blocks from the promoted secondary (since the time of its role change) must be propagated back to the original primary. That sequence is initiated by a special operation that first demotes the original primary to

a secondary role, which allows the promoted site to act as a standard primary array and propagate its updates in order to restore coherency. When that process completes, the orderly role reversal mechanism can be used to restore the original primary to its desired role.

Interactions with other features

Table 1 Remote replication feature interaction considerations.

Feature	Remote Replication interaction considerations
Snapshot groups, snapshots, snapshot virtual disks, consistency groups	All of these features are fully enabled for mutual operations. However, there are special provisions for creating snapshots on virtual disks that are acting in a secondary role in a remote replication relationship. In particular, if a secondary virtual disk is in the midst of a synchronization interval, any attempt to create a snapshot for it is placed in a pending state. When the synchronization completes successfully, the snapshot creation occurs automatically.
Virtual disk copy	A virtual disk participating in a virtual disk copy request cannot be a replication of a secondary virtual disk. The source of a virtual disk copy request can be a replication of a primary virtual disk. The target of a virtual disk copy request cannot be a replica of a primary virtual disk.
Synchronous (legacy) Remote Replication	Remote Replication operations are supported in conjunction with the remote replication feature, but a given virtual disk cannot participate in both (legacy) Remote Replication and remote replication pairs.

Conclusion

The remote replication feature for Dell PowerVault MD3 storage provides a valuable solution for customers who require disaster protection for their critical business data, but at a much lower cost than what is possible with synchronous remote replication mechanisms. Remote replication can tolerate relatively low-bandwidth and high-latency communication links by allowing the user to make meaningful compromises between RPO and link cost. Consequently, the remote replication feature enables long-distance replication over TCP/IP networks for disaster protection, but does so without imposing significant performance penalties on user applications running at the primary site.