

EMC DATA DOMAIN REPLICATOR

A Detailed Review

Abstract

Replication of deduplicated, compressed data offers the most economical approach to the automated movement of data copies to a safe site using minimum WAN bandwidth. This ensures fast recovery in case of loss of the primary data, the primary site, or the secondary store. EMC® Data Domain® Replicator software provides simple, fast, robust WAN-based disaster recovery for the enterprise. This software offers comprehensive flexibility for a variety of topologies in the distributed enterprise, from remote offices at the edge to large core data centers. Unlike other deduplication methods, Data Domain deduplication is inline, so replication completes as fast as possible to minimize risk and maximize currency of the restore point.

May 2012

Copyright © 2012 EMC Corporation. All Rights Reserved.

EMC believes the information in this publication is accurate of its publication date. The information is subject to change without notice.

The information in this publication is provided “as is”. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com.

All other trademarks used herein are the property of their respective owners.

Part Number h7082.4

Table of Contents

Executive summary	4
Introduction	5
Audience.....	5
Data Domain Replicator overview	6
Leveraging logical storage layers to meet different replication requirements	7
Directory replication	8
Managed file replication.....	11
MTree replication	11
Collection replication	12
General Considerations when using Data Domain Replicator	13
Only deduplicated data	14
Compression	14
Encryption	14
Independent data retention at source and destination	14
Retention Lock	14
Flexible Replication Topologies	15
Network management	15
Choosing between directory, managed file, MTree and collection replication approaches	17
Comparing deduplication storage: RPO, RTO, and time-to-DR	19
Recovery point.....	19
Recovery time.....	20
Time-to-DR summary	21
Conclusion	21

Executive summary

Replication has been the preferred method for automating protection of storage against site failure for a long time, but it is deployed only in specific market situations because of associated bandwidth cost. Some IT budgets can afford it—most cannot—and therefore organizations have used the manual, insecure, and error-prone fallback approach of tapes and trucks. If the reader is in that situation, just ask yourself: How confident are you in the outcome of a test of your disaster recovery (DR) readiness?

Normal replication methods just send new delta writes. Let's consider the impact of backup and archive data sets. Full backups write the same data over and over and normal replication methods send the same data every time consuming significant bandwidth. This makes full backups the worst case for replication. Archiving processes for unstructured data like files, emails, etc. further increases the amount of data that needs to be replicated.

Deduplication replication offers a way out. By efficiently finding the differences across different versions of the data, it enables replication for backup and archive data without taxing the network infrastructure. If it is done well, it can save more than 80 percent of the bandwidth used by block delta replication.

EMC® Data Domain® leverages dynamically variable-length deduplication coupled with local compression and therefore can eliminate up to 99 percent of the bandwidth used by normal replication methods. By lowering the cost floor for replication deployments, it encourages much broader use of this simplifying technology. Feedback from many customers applying deduplication to enable wide area network (WAN) replication (including, for example, the State of Louisiana, post-Katrina) has led Data Domain to three critical conclusions about what matters in its deployment.

- **Speed:** Time-to-DR readiness is critical. While backup and archive applications do not require the synchronous behavior of transactional replication, they still have to be designed to meet or exceed the recovery requirements of the tape-centric solutions they replace. Data Domain design elements, such as true inline deduplication, continuous data consistency at the replica, and fast restore streams from replicas, all contribute to easy, fast recovery at the replica site and as quickly as possible after data is initially stored on the source system.
- **Flexibility:** Network characteristics like latency, bandwidth, and packet loss differ from one deployment to the other. It is important to offer choices in setting policies to achieve a balance in speed versus efficiency over these different network types. Data Domain Replicator offers numerous replication types and policies and also supports a wide variety of topologies to meet the needs of various deployments.
- **Simplicity:** In a production-class product, deduplication is not a simple feature. It requires a sophisticated system design for consistent high performance and resilience. The same is true for dedupe replication. Data Domain leverages deduplication deeply in everything from DR system mirroring to system migration

and redeployment. The EMC Data Domain Operating System (DD OS) integrates replication with all supported protocols and software options such as DD Encryption, DD Extended Retention and DD Retention Lock software.

Overall, the DD Replicator software option enables policy-driven approaches to dedupe replication that integrate tightly with the underlying DD OS data storage framework. This allows Data Domain customers to gain the full benefits of a dedupe-centric storage network for an end-to-end data protection strategy.

Introduction

This white paper introduces EMC Data Domain Replicator software and explains how it delivers flexible replication topologies for enhanced disaster recovery in various enterprise environments. Read this white paper to find out how DD Replicator addresses DR needs for backup and archive data in centralized and distributed enterprises.

In the following sections, we will describe the unique characteristics of DD Replicator, including cross-site deduplication, as well as deployment scenarios and best practices to meet recovery objectives.

Audience

This white paper is intended for EMC customers, technical consultants, partners, and members of the EMC and partner professional services community who are interested in learning more about Data Domain Replicator software.

Data Domain Replicator overview

Within a Data Domain system, there are several levels of logical data abstraction above the physical disk storage, as illustrated in Figure 1.

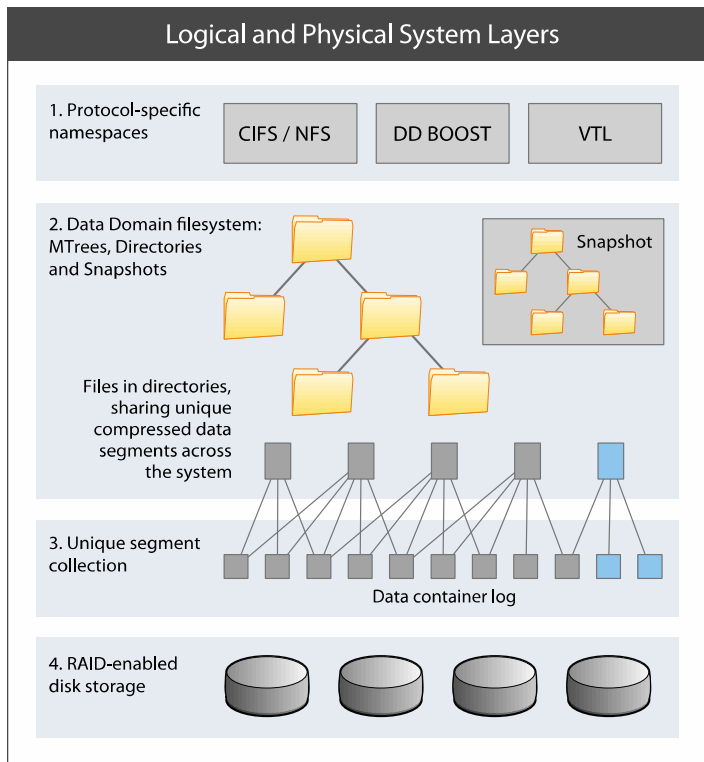


Figure 1. In a DD OS filesystem, protocol-specific namespaces are presented to clients/applications for accessing the logical filesystem layer. The files and directories within MTrees as well as MTree snapshots, all reference the same pool of unique segments, called a collection, which is made up of log-structured containers that organize the segments on disk to optimize throughput and deduplication effectiveness.

These layers are described below:

- 1. Protocol-specific namespaces:** As an external interface to applications, there are protocol namespaces, such as virtual tape libraries (VTL, over Fibre Channel), Data Domain Boost storage units (for use with EMC NetWorker, EMC Avamar, EMC Greenplum, Symantec OpenStorage, Quest vRanger and Oracle RMAN), and CIFS/NFS fileshares (over Ethernet). A Data Domain deployment may use any combination of these simultaneously to store and access data.
- 2. Data Domain filesystem: MTrees, Directories and snapshots:** Files and directories for each namespace are stored in an MTree within the Data Domain filesystem. With VTL, the virtual tape cartridges are stored as files under special directories. MTree snapshots in DD OS are logical; they share the same underlying data segments in the collection, and are very space-efficient.

3. **Unique segment collection:** A ‘collection’ is the set of files (or virtual tapes) and logical MTree snapshots. The system identifies and eliminates duplicates within each container and then writes compressed deduplicated segments to physical disk. Segments are unique within the collection (not including specific duplicates maintained in DD OS to enable self-correction or recovery). Each Data Domain system has a single collection that is stored in a log of segment locality containers. For more about segment localities, see the white paper [EMC Data Domain SISL Scaling Architecture](#).
4. **RAID-enabled disk storage:** These collection containers layer over RAID-enabled disk drive blocks. Data Domain deduplication storage systems use Data Domain RAID 6 internal disk and storage expansion shelves to protect against dual disk failures.

Leveraging logical storage layers to meet different replication requirements

Data Domain Replicator software offers four replication types that leverage these different logical levels of the system for different effects. All four replication types are designed to deal with network interruptions that are common in the WAN and recover gracefully with very high data integrity and resilience. This ensures that the data on the replica is in an application usable state. This is critically important for optimizing utility of the replica for DR purposes.

At a high level, the four replication types are:

- **Directory replication** transfers deduplicated changes of any file or subdirectory within a Data Domain filesystem directory that has been configured as a replication source to a directory configured as a replication target on a different system. Directory replication offers flexible replication topologies including system mirroring, bi-directional, many-to-one, one-to-many, and cascaded, enabling efficient cross-site deduplication.
- **Managed file replication** is used by the DD Boost software option, for optimized levels of performance and integration with EMC NetWorker, EMC Avamar, Symantec OpenStorage and Oracle RMAN. Managed file replication directly transfers a backup image from one Data Domain system to another, one at a time on request from the backup software. The backup software keeps track of all copies, allowing easy monitoring of replication status and recovery from multiple copies. This form of replication provides the same cross-site deduplication effects and flexible network deployment topologies as directory replication.
- **MTree replication** is used to replicate MTrees between Data Domain systems. Periodic snapshots are created on the source and the differences between them are transferred to the destination by leveraging the same cross-site deduplication mechanism used for directory replication. This ensures that the data on the destination is always a point-in-time copy of the source with file-consistency. This also reduces replication of churn in the data leading to more efficient utilization of the WAN. MTree replication supports all the replication topologies supported by directory replication.

- **Collection replication** performs whole-system mirroring in a one-to-one topology, continuously transferring changes in the underlying collection, including all of the logical directories and files of the Data Domain filesystem. While collection replication does not support the flexibility of the other three types, it is very simple and lightweight, so it can provide higher throughput and support more objects with less overhead, which is ideal in high-scale enterprise cases.

A detailed examination of each replication type follows.

Directory replication

With *directory replication*, a replication context pairs a directory (and all files and directories below it) on a source system with a destination directory on a different system, as seen in Figure 2. During replication, deduplication is preserved since data segments that already reside on the destination system will not be resent across the WAN. The destination directory will be read-only as long as the replication context is configured.

The replication destination can contain other replication destination directories, replication source directories, and other local directories, all of which will share deduplication in that system's collection. As a result, directory replication offers a wide variety of topologies: simple system mirroring, bi-directional, many-to-one, one-to-many, and cascading.

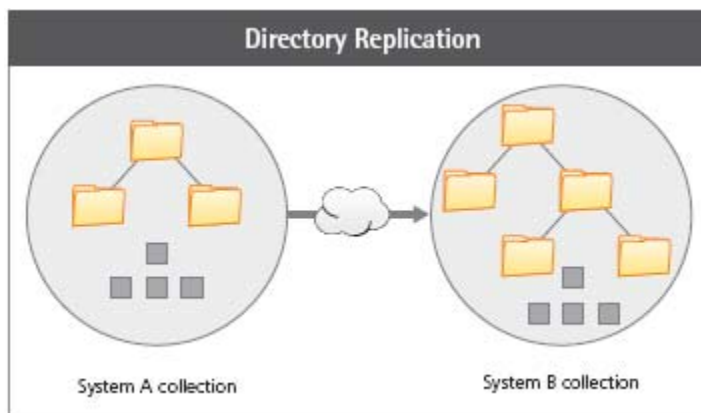


Figure 2. With directory replication, the source and destination can have independent collections.

In directory replication, the file transfer is triggered by a file closing, and the order of the closes is preserved. In cases where closes are infrequent, DD Replicator will force the data transfer periodically. As metadata and corresponding unique data segments are transferred, the files are separately created and maintained on the remote system—that is, the collection of the destination is independent of the source. Figure 2 shows that System A replicates to System B and each has its own separate collection. Once the destination system receives the complete file, it is immediately made visible to the namespace (CIFS, NFS, or VTL) at the destination and can be used for recovery purposes, writing to tape, etc.

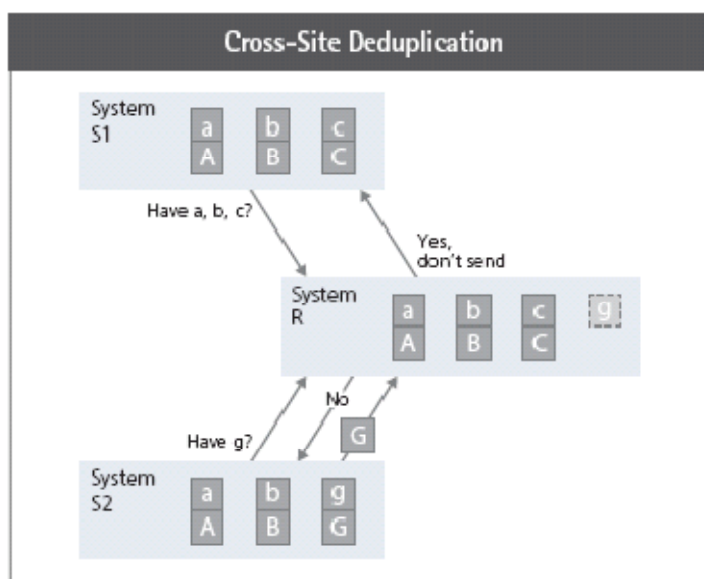


Figure 3. Metadata exchange between the source and destination ensures that a data segment only needs to be sent to a destination once, irrespective of where the data comes from. This provides significant efficiencies over the WAN in many-to-one deployments since common segments on different sources only need to be sent once.

The effect of cross-site deduplication provides WAN replication efficiencies comparable to the deduplication effect on storage, and the benefits aggregate in a multi-site topology. As an illustration, imagine that there are three sites, S1, S2, and R, in a two-to-one topology as seen in Figure 3. R is the destination for source directories replicating from sources S1 and S2. Assume that the replicating directories in S1 and S2 have identical data, but only S1 has replicated already, and S2 is just getting started replicating to R. The time and bandwidth required for S2's data to replicate to R are very small; the data is already there so just the metadata needs to transfer. The effect on bandwidth between S1's initial data being sent versus S2's redundant data is similar to the difference between the first full and the second full in local deduplication storage capacity used. S1 would typically have had to send data about a third the size of a full backup to synchronize to the replica system R; S2, with the same data coming later, would send about 1/60th the size of that same full backup.

While a secondary copy of data is sufficient for many organizations, some require a tertiary (or even greater number) copy, particularly in highly distributed enterprises. Creating an additional copy provides increased data protection and the ability to distribute data for multi-site usage. For example, QA/ testing content or training material can be reliably and efficiently replicated to different remote sites. DD Replicator supports two powerful replication topologies, one-to-many and cascaded, that enable the creation of multiple copies of data. One-to-many replication creates multiple copies from the source system, and cascaded replication creates copies from successive replication of the source system data. Combining these two provides the greatest flexibility in leveraging existing networks with complex topologies and varying bandwidths.

As shown in Figure 4, one-to-many replication allows the same source directory to be replicated in parallel to multiple remote sites. Setting up one-to-many replication is similar to creating multiple independent replication contexts, one at a time, all with the same source directory.

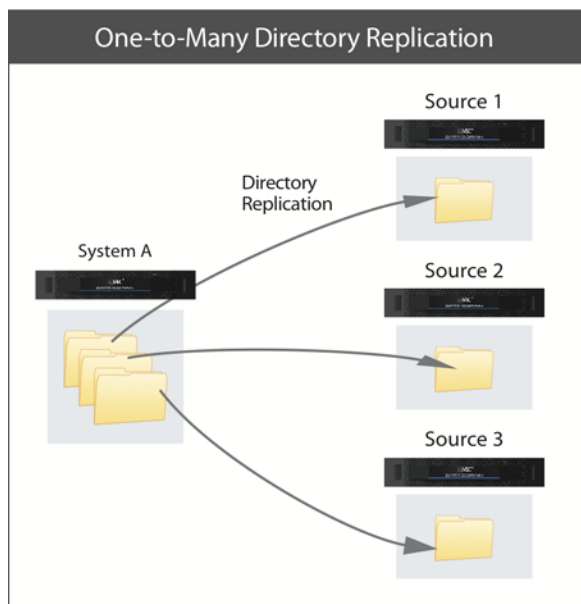


Figure 4. One-to-many replication allows a directory to be replicated concurrently to multiple remote systems. A replication context is created for each destination from the same source directory.

With cascaded replication, a directory on a Data Domain system can be configured to be both the destination of one replication context and the source of another. This

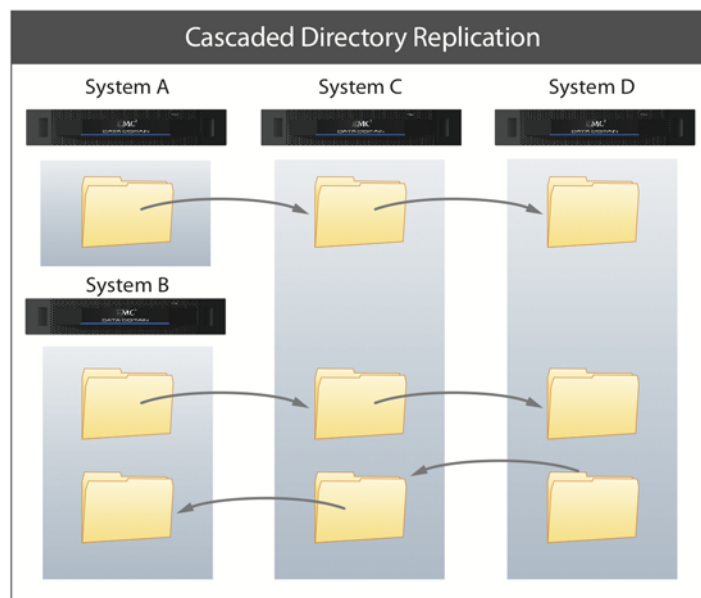


Figure 5. Cascaded replication allows directory replication contexts to be configured in a chain of sequential replication hops across multiple systems. Bi-directional replication is supported.

allows the replication of a directory (and all files and sub-directories) from Data Domain system A or Data Domain system B to Data Domain system C, and a subsequent replication of the same directory to Data Domain system D, as shown in Figure 4. This enables datasets to be replicated to as many sites as required in multiple sequential hops.

The recovery or resynchronization of a cascaded chain of replication contexts can be done in the opposite direction of the replication flow for example, from System D to System C and then to System B.

Depending on requirements either one-to-many or cascaded replication may be preferable. For example, since one-to-many creates copies from the same source directory, data arrives at the destination sooner than cascaded replication, which requires the data to arrive at the intermediate system first before it is replicated to its final destination. Therefore, if the fastest speed to multiple copies is a priority, one-to-many may be preferable. However, if network bandwidth is limited at the source and/or because getting one copy offsite first is more critical for DR readiness than any additional copies, then cascaded replication may be preferable.

Managed file replication

Managed file replication using DD Boost allows the backup software to control the replication on a per-file basis. When integrated with DD Boost, the backup software's users can configure policies to selectively replicate the individual backup image or dataset to another system after completion of the backup. Unlike traditional vaulting or cloning to tape, the data is not read by the backup server to be written elsewhere. Instead, the backup software delegates the data movement to the DD system; thereby leveraging the most efficient method available to create a DR copy of the data.

The backup software decides when to get started, and knows when it is finished, based on interactive signaling between DD Boost and the Data Domain system. Using this approach, the backup software knows that the destination holds a copy of the file that is separate and different from the source's file, and retention periods for the two can be managed independently, for example, to keep full backups longer on the DR site.

MTree replication

MTrees are user-defined logical partitions of the Data Domain file system that enable more granular management of the Data Domain filesystem. MTree replication enables the creation of disaster recovery copies of MTrees at a secondary location. In addition, one can also enable DD Retention Lock on an MTree-level at the source, which will get replicated to the destination.

MTree replication creates periodic snapshots at the source and transmits the differences between two consecutive snapshots to the destination. At the destination Data Domain system, the latest snapshot is not exposed until all the data for that snapshot is received. This ensures the destination will always be a point-in-time image of the source Data Domain system. In addition, files will not up show out-of-order at the destination and provides file-level consistency that simplifies recovery

procedures and reduces RTOs. Users are also able to create a snapshot at the source Data Domain system to capture a consistent point-in-time image (for example, after archiving a user's emails), which gets replicated to the destination where the data can be used for recovery in a disaster scenario.

MTree replication groups all changes between snapshots and replicates them. Consequently, any churn in the data between snapshots will not be replicated. This makes MTree replication suitable for applications that make frequent updates to the filesystem, for example filesharing and archiving workloads or certain backup applications that create, modify and delete temporary lock files within a short interval.

MTree replication has a lot of commonality with directory replication. It uses the same WAN deduplication mechanism used by directory replication to avoid sending redundant data over the WAN. It also supports all the topologies supported by directory replication (one-to-one, bi-directional, one-to-many, many-to-one, cascaded). In addition, one can configure MTree replication to replicate MTree data on a system that already leverages directory replication and/or managed file replication.

Collection replication

The fastest and lightest impact replication type is at the collection level. Unlike the prior three, there is no on-going negotiation between the systems regarding what to send. Collection replication is mostly unaware of the boundaries between files. Replication operates on segment locality containers that are sent once they are closed.

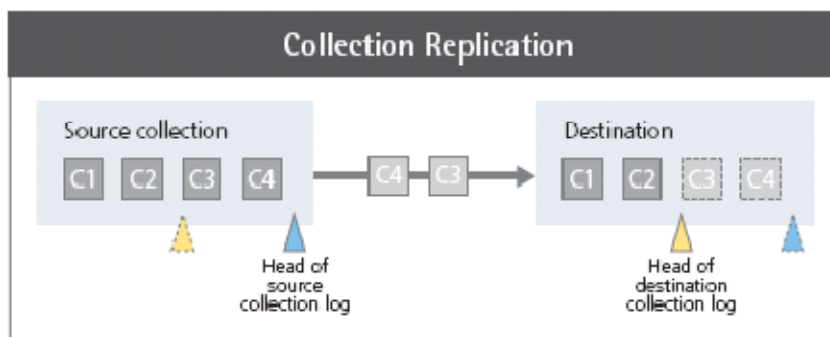


Figure 6. By leveraging the log structure of the collection, collection replication tracks the delta between the head of the source and destination collections, and transfers each container containing unique segments, in order, until it catches up.

Because there is only one collection per Data Domain system, this is specifically an approach to system mirroring. The destination system cannot be shared for other roles. It is read-only and shows only data from one source. Once the data is on the destination, files (and virtual cartridges) become immediately visible for recovery.

As previously described, the collection's container set is a log structure. Therefore, transferring data in this way means simply comparing the heads of the source and destination logs, and catching up one container at a time, as shown in Figure 6. If it gets behind, it will catch up later. This approach is very well adapted to enterprise deployments wishing to minimize the resource overhead of the selectivity and cross-site filtering overheads of directory or MTree replication (for example for very large DR deployments using high-bandwidth WANs), or systems containing millions of files in an archiving deployment. Due to this light weight approach, collection replication can provide a logical throughput of up to 52 TB/hr on a 10 Gb network.

As shown in Figure 7, the final hop in a cascaded chain of replication contexts can also be configured to use collection replication when the entire contents of the intermediate system need to be replicated to a secondary DR site.

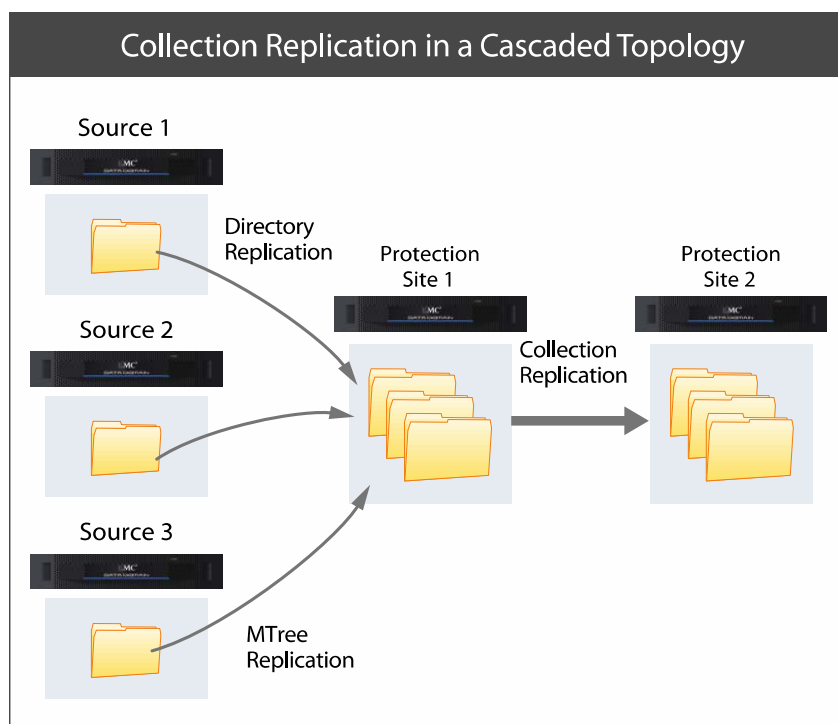


Figure 7. Collection replication can be used in the final hop of a cascaded replication topology, mirroring the contents of the Data Domain system in protection site 1 to the system in protection site 2.

General Considerations when using Data Domain Replicator

The following considerations are important to be aware of when designing the system.

Only deduplicated data

In DD OS, data is deduplicated as it is written to the source system and replication preserves deduplication. This ensures that the network is efficiently utilized for creating a DR copy of backup and archive data.

Compression

DD OS offers a choice of local compression types: LZ-style (default), GZ-fast, GZ, or no compression. Data transferred over the network using DD replicator is always compressed using the same algorithm as that of the destination. If source and destination have different compression types, then the data is first uncompressed at the source and then recompressed using the destination's algorithm before being sent across the network.

Encryption

Data Domain Encryption software allows the user to encrypt data at rest by using RSA BSAFE FIPS 140-2 compliant libraries with standard 128-bit or 256-bit Advanced Encryption Standard (AES) algorithms. Depending on IT security policies, the block cipher modes for the AES algorithm can be selected either as Cipher Block Chaining (CBC) or Galois Counter Mode (GCM). DD Replicator is compatible with DD Encryption and any data transferred over the network is always encrypted using the encryption key of the destination. If source and destination of a context have different encryption keys, then data at the source is first decrypted, the source system will obtain the encryption key of the destination and data is re-encrypted using the destination's encryption key before sending the data across the network.

In addition to supporting data-at-rest encryption, DD Replicator also supports encryption of data-in-flight by using standard Secure Socket Layer (SSL) protocol version 3, which uses the ADH-AES256-SHA cipher suite to establish secure replication connections.

Independent data retention at source and destination

Data Domain systems offer two options for retaining data for different periods of time at the source and destination. The customer can retain data for 30 days on the source Data Domain system at the remote office and keep it for 6 months on the destination Data Domain system at the central data center. When managed file replication is used with Boost, the user can configure separate retention periods for backup files on the source and destination Data Domain systems. With directory replication, user can create independent snapshots on the source and destination and retain these snapshots for the desired durations.

Retention Lock

EMC Data Domain Retention Lock[®] (DD Retention Lock) software provides immutable file locking and secure data retention capabilities for customers to meet both corporate governance and compliance standards (such as SEC 17a-4(f)). DD Retention Lock comes in two editions – EMC Data Domain Retention Lock Governance

edition and EMC Data Domain Retention Lock Compliance edition. Both editions provide the capability for IT administrators to configure minimum and maximum retention periods at the MTree level and apply retention policies at an individual file level. Collection replication can be used to replicate the locked data and the MTree-level retention periods for both the Retention Lock editions. For DD Retention Lock Governance edition, directory and MTree replication can be used to replicate the locked data, while MTree replication can also replicate the MTree-level retention periods.

Flexible Replication Topologies

To enable enterprise-wide data protection, DD Replicator provides multiple replication topologies - system mirroring, selective data replication, bi-directional replication, many-to-one replication, one-to-many replication and cascaded replication. With many-to-one replication, up to 270 data domain systems in geographically distributed locations can replicate into a single system at the central data center.

Network management

There are many network management capabilities that benefit all of the Data Domain replication approaches.

- **Identifying status.** Data Domain Enterprise Manager includes a GUI for setting up replication and managing all replication choices. Figure 8 is an example pane that highlights current status and current replication completeness on a many-to-one directory replication destination node.

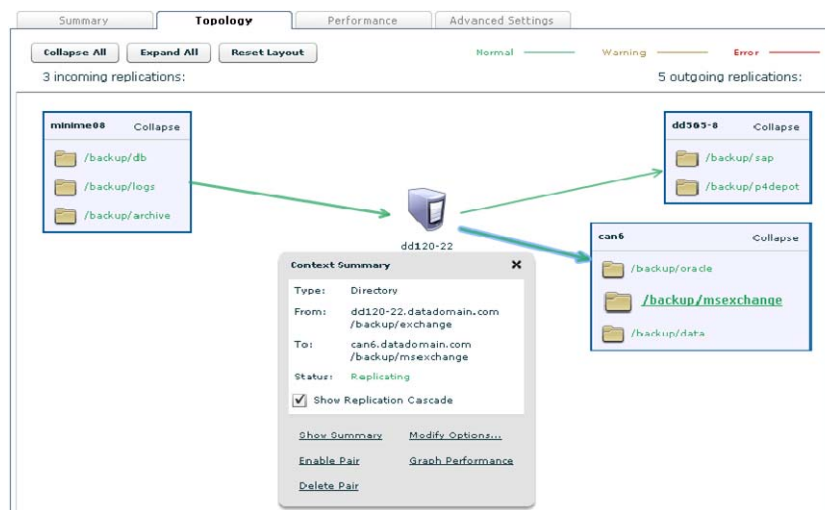


Figure 8. EMC Data Domain Enterprise Manager enables management of replication across all systems in the distributed enterprise.

To further monitor replication, additional views are available on a per-context basis. In these views, shown in Figure 9, additional focus is provided for the most common

administrative questions regarding DR-readiness, such as synchronization status, along with a graph to quickly view the ongoing delay curves between data storage and data transmission rates.

The screenshot displays the 'Summary' tab of the Data Domain Enterprise Manager. It features a table of replication contexts with columns for Source, Destination, Type, State, Synced As Of Time, Pre-Comp Remaining, and Time To Completion. The selected context is 'dd690-54...repl_many2zone/dd690-54/dir2' to 'qa-bw-14...any2zone/dd690-54/dir2', which is in a 'Normal' state and has completed. Below the table, the 'Detailed Information' section shows the state as 'Replicating' and provides details on completion statistics, including 'Synced As Of Time' (2010/11/22 5:00:01 PM) and 'Time To Completion' (Completed). A 'Completion Predictor' section shows the source time as 'Today' and the completion time as 'Completed'.

Figure 9. Detailed views of replication configuration and status for each replication context are managed by Data Domain Enterprise Manager.

- **Throttling.** As a basic form of quality of service (QoS), administrators may establish times of day during which data may or may not be sent, along with limits to the amount of bandwidth that can be used for replication. For more advanced QoS functionality, use of a more sophisticated system in the network itself is recommended.
- **Multi-streaming of a replication context for high-bandwidth networks.** To maximize the use of high-bandwidth WAN links, Data Domain systems can allocate multiple streams for each replication context. This improves the replication throughput and subsequently the time-to-DR readiness.
- **Optimization of a replication context for low-bandwidth networks.** For enterprises with small datasets and 6 Mb/s or less bandwidth networks, DD Replicator can further reduce the amount of data to be sent using the low-bandwidth optimization mode. This enables remote sites with limited bandwidth to use less bandwidth or to replicate and protect more of their data over existing networks.
- **Secure connection authentication.** All connections use Diffie-Hellman key exchange between source and destination systems.

- **Replication-level data verification.** Data Domain Replicator uses its own large checksum to verify correctness of all sent data, above and beyond what TCP provides. The TCP embedded checksum is not strong enough for dedupe, and can have thousands of errors per week on many WANs. Additional verification is also provided at the storage level, as discussed in the [EMC Data Domain Data Invulnerability Architecture](#) white paper.
- **Use of alternate ports.** Replication may use any available Ethernet/IP port, allowing administrators to configure dedicated replication networks to avoid interface contention with backups writing to a CIFS/NFS fileshare.
- **Scripting and reporting tool integration.** Data Domain systems have a rich command line interface that includes full support for replication management. All event and status information is stored in ASCII logs for easy adoption by third-party reporting tools. Warnings and summary status are provided in email reports distributed by the Data Domain autosupport process. These reports are also sent to EMC Support to provide history for optimal customer service on demand.

Choosing between directory, managed file, MTree and collection replication approaches

To determine the best replication approach for a given Data Domain deployment refer to Table 1 to figure out the supported replication types based on the protocol being used.

Protocol	Directory	Managed File	MTree	Collection
NFS/CIFS	✓		✓	✓
DD Boost		✓		✓
VTL/NDMP	✓		✓	✓

Table 1. Replication protocol support

Then use Table 2 and Table 3 to select the replication type depending on the replication topology to be used.

- If all that is needed is one-way system mirroring between two sites, then use collection replication. It is the fastest and lowest impact method for DR readiness. In some large enterprise data centers, based on the size of the systems and link speeds involved, collection replication may be the most appropriate means of creating a DR copy of the data.
- If systems are used for storing archive data sets, and there are millions of small files to be replicated, then use collection replication for optimal performance.
- If the system contains compliant archive data (whether as the only data set or in addition to backup data) that is locked using DD Retention Lock Compliance edition, then collection replication must be used for creating a DR copy of this data.

- If you are using applications with DD Boost support (e.g. EMC NetWorker, EMC Avamar, Symantec OpenStorage and Oracle RMAN), then use managed file replication. It is backup operator-centric, and offers simple methods for more advanced policies, such as maintaining separate retention periods for original and replica image storage.
- If you are creating MTrees for logically partitioning the Data Domain system, then use MTree replication to get the benefits of flexibility in topologies and WAN efficiency.
- If you want to protect data with frequent churn or want the capability to define application-consistent snapshots to simplify disaster recovery, then use MTree replication.
- If a system with DD Extended Retention license contains both short-term and long-term retention data, then use MTree replication for creating DR copy of the short-term data and managed file replication for the long-term retention data.
- For everything else, use directory replication.

Topology	Directory	Managed File	MTree	Collection
Single-system mirroring	✓	✓	✓	✓
Selective data replication or different data to different locations	✓	✓	✓	
Bi-directional	✓	✓	✓	
Many-to-one	✓	✓	✓	
One-to-one	✓	✓	✓	
Cascaded	✓	✓	✓	Final Hop

Table 2. Replication topologies

Source	Destination		
	DD	DD with Extended Retention software	GDA
DD	Directory, managed file, MTree, collection	Directory (no file migration), MTree (no file migration), collection, managed file	Managed file
DD with Extended Retention software	NA	Collection, MTree (no file migration), managed file	NA
GDA	NA	NA	Collection, managed file

Table 3. Supported replication types

Comparing deduplication storage: RPO, RTO, and time-to-DR

Two well-known metrics, recovery point objective (RPO, how old is the recoverable data at the replica?) and recovery time objective (RTO, how long does it take to recover the data to usability at the replica site?), are useful when considering replication techniques. To compare deduplication storage systems, consider one additional, composite metric. Starting with a suitably large full backup definition such as 20 TB, time how long it will take to:

1. Back up and deduplicate at the originating system
2. Replicate across an IP network
3. Restore the data from the replica to a different set of servers at the DR site

For the sake of brevity, call this composite metric time-to-DR. This is the end-to-end time from the beginning of a backup to completion of data recovery at the replica site. In proof-of-concept tests comparing new deduplication storage systems intended for DR use, this is the most telling indicator.

Recovery point

The recovery point of the data at the replica will be older if it takes longer to replicate. Whatever most recent complete replica exists already, it will still be the best restorable copy until the new one gets there. For example, assume a new backup is starting, and the recovery point at that time is from yesterday's backup. In dedupe replication, users typically only want to replicate the deduped (smaller bandwidth) data. In a Data Domain system, dedupe is fast and inline, and replication can be simultaneous with backup, so it can finish shortly thereafter. The restore image is available immediately on arrival at the replica.

In a slower dedupe-rate system, especially in one that delays the beginning of dedupe through being a "post-process" dedupe system, replication takes much longer. In a post-process system, the dedupe rate is typically less than half the ingest rate to non-dedupe storage (otherwise, no one would bother with the two-step process and its complexity, boundary conditions, and extra disk provisioning requirements). Since replication can only complete when dedupe is finished, this typically compromises the real arrival rate at the replica site. If half the dedupe rate of a Data Domain system, that means data can get to the replica site at no more than half the speed. So the restore point would be from yesterday plus two times the backup window, or worse.

Some systems compromise even further:

- Some dedupe systems do not compress on the WAN, resulting in either twice the bandwidth cost or twice the time for data to arrive at the replica.
- Some dedupe systems are not continuously consistent at the replica. They have to do a batch or manual process to enable the newly replicated data to be readable. The timing and effort required for this need to be taken into account in planning for recovery. For example, in a system that synchronizes in batch on a daily

schedule, after dedupe is presumed to have crossed the wire, there could be an additional window where even though data has arrived, it is not restorable. Therefore, yesterday's backup remains the recovery point.

Recovery time

On the replica, there is only deduplicated data. The recovery time is the same as the restore rate from the dedupe pool in the replica. This should be measured carefully with a large dataset to ensure sustained performance characteristics. Because of the SISL architecture, Data Domain deduplication storage provides fast restores at both the originator and replica. The *only* performance rates published by Data Domain are from *deduplicated* storage.

Post-process systems, at the time of publication, do not provide their restore rates from their dedupe pools, especially on replicas. There seems to be a significant drop-off in performance from their rated specifications (which are all benchmarked against their non-deduplicated cache storage). Recovery time is somewhere between slower and infinite.

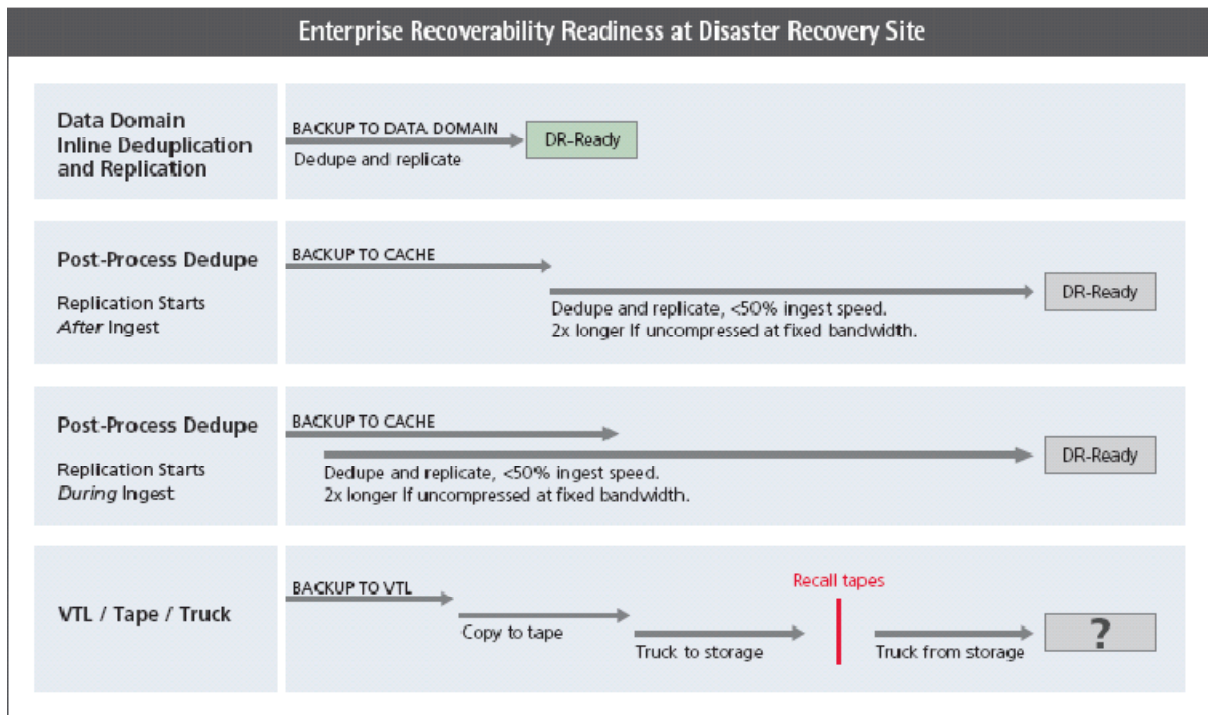


Figure 10. Today, enterprises back up data to enable both onsite and offsite recovery in the event of a disaster. Data Domain deduplicates this data inline and begins replication immediately, enabling the remote system to be DR-ready faster. Post-process approaches first ingest backup data to a disk cache and then perform deduplication, typically at < 50% of ingest speed. Regardless of whether the deduplication and replication begin during or after ingest, the point at which these systems are DR-ready is delayed.

Time-to-DR summary

Once the recovery point is established — faster transfer is better to make this current sooner — recovery from it must also be timed to systems in the remote site to complete a DR test. Deduplication storage systems offer a range of alternative approaches, but Data Domain has chosen to optimize aggressively in favor of recent recovery points and faster recovery times at the replica. As shown in Figure 10, others clearly have not.

Conclusion

Most large enterprise users require a global disaster recovery (DR) strategy that protects the entire organization by having one or more copies of their data at offsite locations. EMC Data Domain Replicator software asynchronously transfers only the compressed, deduplicated data over the WAN, making network-based replication cost-effective, fast and reliable without requiring manual intervention. The key features of DD Replicator are as follows:

- Safe and network-efficient replication
 - Cross-site deduplication
 - Low-bandwidth optimization
 - Encrypted replication
 - 99 percent bandwidth reduction
- Scalable replication throughput
 - Up to 52 TB/hr logical throughput
 - Multi-stream optimization
- Enterprise deployment flexibility
 - Flexible replication topologies
 - Consolidate data from up to 270 remote Data Domain systems
 - Policy-based data management
 - Significant advantages in RPO, RTO and fastest “time-to-DR”
- Easy integration
 - Supports leading backup and archive enterprise applications
 - Compatible with all DD OS functionality - Compression, Encryption, Extended Retention, Retention Lock
 - Replicates VTL, CIFS, NFS, NDMP and EMC Data Domain Boost data