White Paper

IBM TSM Backup with EMC Data Domain Deduplication Storage

Best Practices Planning

Abstract

This white paper provides configuration and best practices recommendations for EMC® Data Domain® deduplication storage systems when used for backup with IBM Tivoli Storage Manager (TSM) in NAS and SAN environments.

October 2010

Copyright © 2010 EMC Corporation. All rights reserved.

EMC believes the information in this publication is accurate as 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 h8065

Table of Contents

Executive summary

EMC® Data Domain® deduplication storage offers simple and reliable disk-based backup and recovery, deduplicating data inline, before it ever hits disk. It is designed for optimal performance and ease of use and effortlessly scales to meet customer storage needs. Data Domain systems integrate seamlessly into any backup environment and provide leading-edge backup and restore operations.

This paper will focus on backup best practices and tuning, as well as configurations to avoid, with the goal of effective integration of a Data Domain system into a TSM environment.

Using a Data Domain system for IBM Tivoli Storage Manager (TSM) progressive backups reduces the TSM 24-hour maintenance duty cycle by shortening the time TSM spends moving data. Some options for accomplishing this include:

- Depending on the environment, the TSM administrator can save time by eliminating migration to tape if the Data Domain system is used in lieu of the primary disk pool.
- Data Domain Replicator software saves time by using replication to replace TSM copying data to a backup storage pool.

Other notable benefits, discussed later in the paper, gained by using TSM with Data Domain deduplication storage include:

- Adding a Data Domain system to a TSM environment gives the TSM administrator greater flexibility in the TSM database backup and recovery scheme implementation.
- By eliminating redundant data segments inline, Data Domain systems allow many more TSM backups to be retained with minimal storage footprint.
- Data Domain deduplication storage integrates seamlessly into a TSM environment by presenting itself either as a NFS/CIFS or VTL storage server.
- Compared to having the TSM server manage deduplication, Data Domain systems offload the deduplication engine from the TSM server, thus reducing the server's CPU load.
- Integrating Data Domain deduplication storage and DD Replicator into TSM environments minimizes the time to disaster recovery (DR).
- Power, cooling, and space savings in the data center are realized with Data Domain storage in a TSM environment by minimizing the storage footprint required to hold full/incremental TSM backups.

Introduction

This white paper provides configuration and best practices recommendations for EMC Data Domain deduplication storage systems when used for backup with IBM TSM in network-attached storage (NAS) and storage area network (SAN) environments.

This guide reviews TSM configuration and best practices to assist in eliminating the bottlenecks associated with functional testing and deployment of this combined solution.

In addition to backup and archiving, Data Domain systems can be used to enable fast, network-efficient, offsite DR as an alternative to tape. For a detailed review of best practices for DR with Data Domain systems, see *IBM TSM Disaster Recovery with EMC Data Domain Deduplication Storage*.

Audience

EMC customers, partners, and professional services engineers who are interested in configuration and backup best practices information when using EMC Data Domain systems with TSM are encouraged to use this paper.

Basic concept

Data Domain deduplication storage systems are designed and optimized specifically for backup and archive data, with key product attributes, including:

- High-speed, inline deduplication using small, variable-length sequences to identify and eliminate redundant data segments before storing to disk
- Integrated data protection technologies such as RAID 6 and the Data Domain Data Invulnerability Architecture, providing post-backup data verification and periodic validation checks of existing data sets
- Automated replication of backup data for DR using cost-effective, low-bandwidth WAN links
- Backup and archive storage in one appliance through generalized support for multiple protocols, such as NAS interfaces over Ethernet, a virtual tape library (VTL) interface option over Fibre Channel (FC), and product-specific interfaces such as TSM Storage Agent

Data Domain deduplication storage is tuned for applications that perform sequential I/O such as backups. While Data Domain systems support multiple interfaces, only NAS and VTL interfaces are supported by both TSM and Data Domain deduplication storage.

On a Windows network, the Data Domain storage system presents shares via a Microsoft Common Internet File System (CIFS) protocol. On a UNIX or Linux network, it presents shares accessible via a Network File System (NFS) protocol. On a SAN environment, it is accessible via a VTL protocol. A single Data Domain system can present all protocols simultaneously.

TSM is an enterprise-wide storage management backup application that delivers automated storage management services to workstations, file servers, databases, and mail server applications. Tivoli Storage Manager supports performing backups to local tape drives, local disk, or a NAS device, as shown in Figure 1.

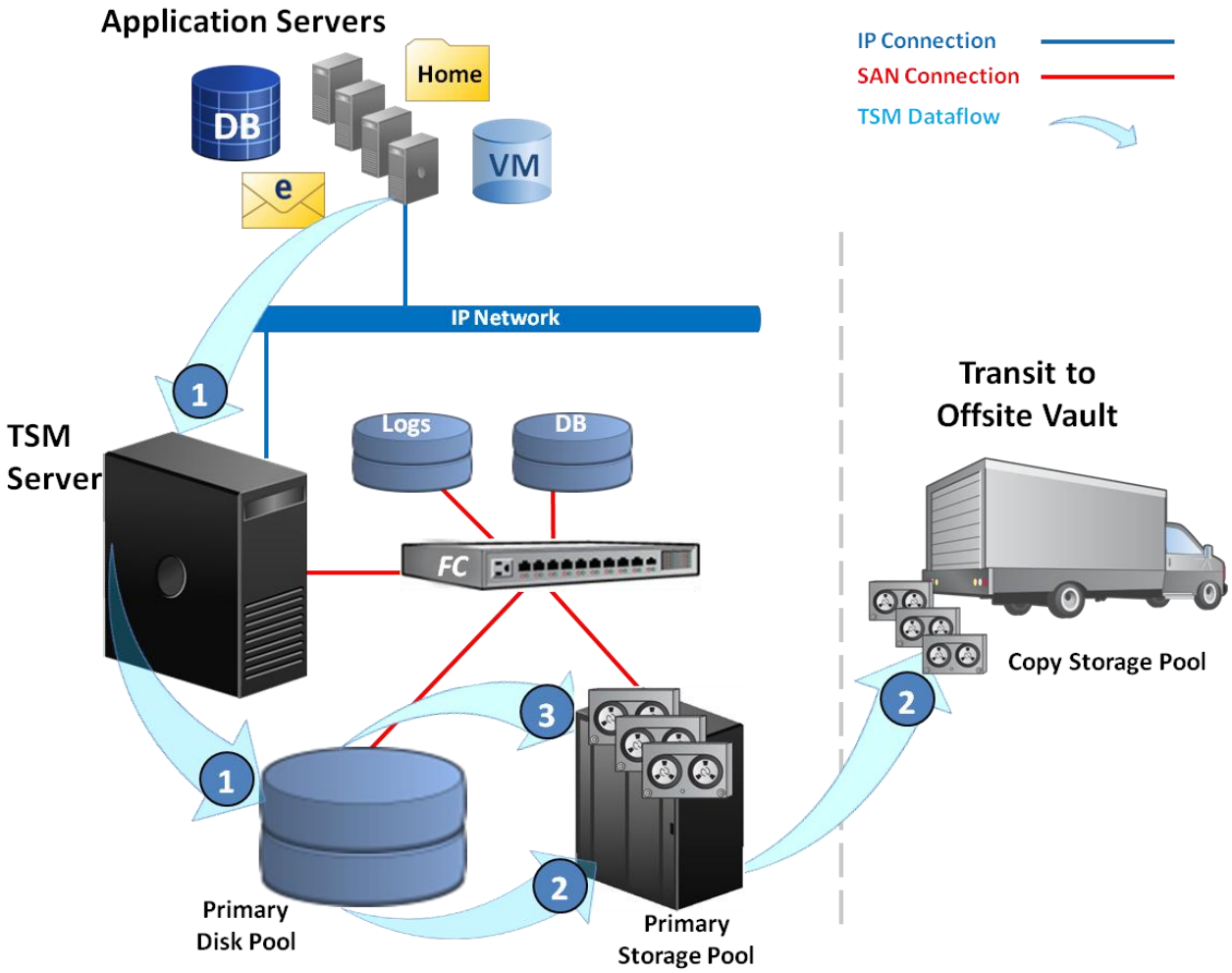

Figure 1. Example of a traditional backup environment

The traditional TSM workflow is as follows:

1. Client application server data is backed up to the Primary Disk Pool staging area.
2. Backup objects are copied from the Primary Disk Pool to the Copy Storage Pool for offsite vaulting.
3. Data is migrated off of the Primary Disk Pool to the Primary Storage Pool to make room for staging the next night's backups.

Figure 2 shows a Data Domain deduplication storage system used as a backup target for TSM.

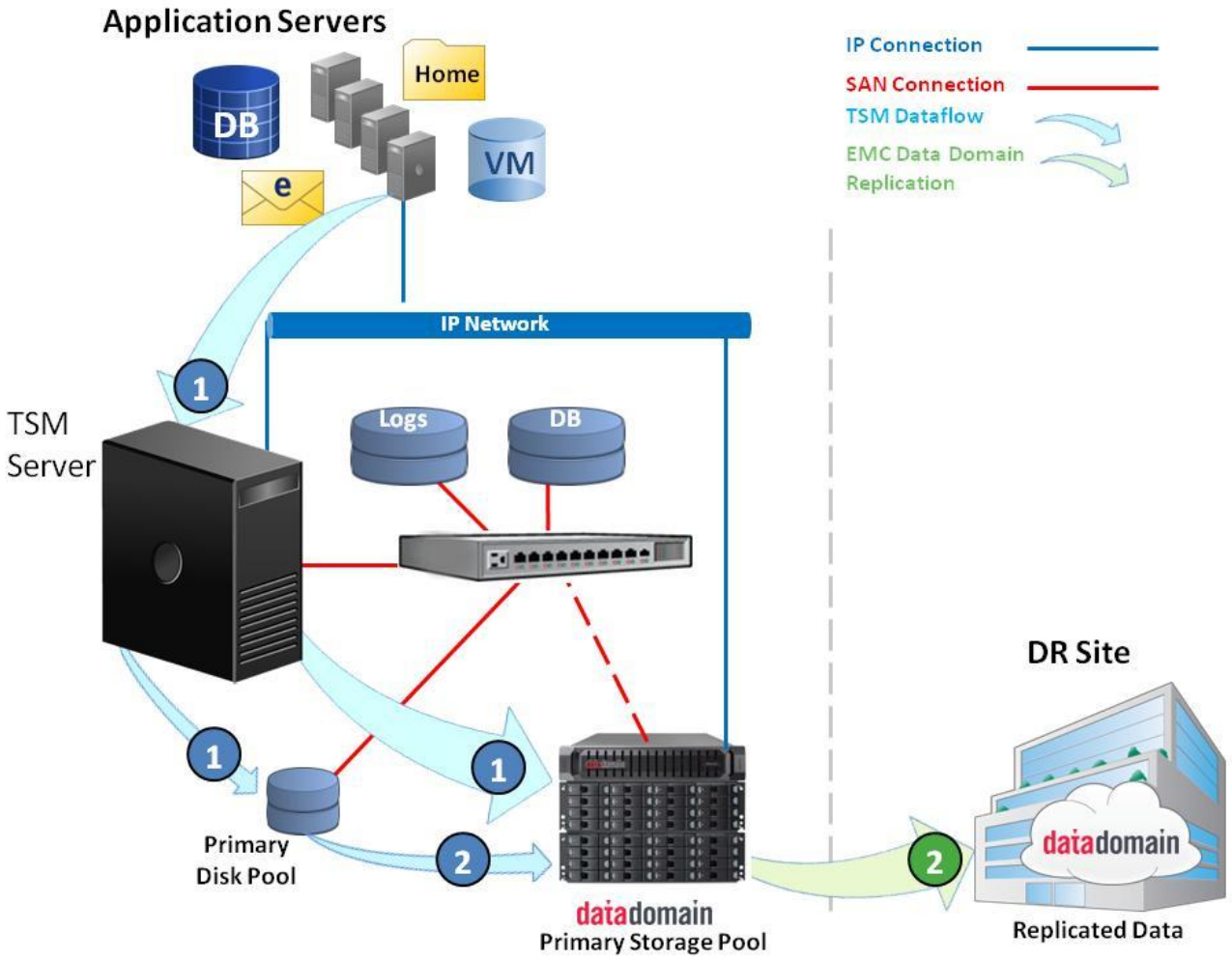

Figure 2. Tivoli Storage Manager environment with Data Domain deduplication storage

Tuning parameter best practices for Tivoli Storage Manager

For backup administrators already well briefed on both Tivoli Storage Manager and Data Domain systems, a summary of the suggested best practice parameter values is presented in Table 1. Details for each listed item are included later in this paper.

Table 1. Summary of best practice settings

Parameter or Option	Setting
Client or Server compression	No
Encryption	No
Reclamation setting	90%
Deduplication	No
Dynamic Tracking*	Disabled

REUsedelay	0
TDP Multiplexing	No

NFS Mount Options	Setting**
Linux	intr,hard,rsize=32768,wsize=32768,proto=tcp,vers=3,nolock
AIX	intr,hard,combehind,rsize=32768,wsize=32768,llock,vers=3,proto=tcp
Other UNIX	intr,hard,rsize=32768,wsize=32768,llock,proto=tcp
Replication	Yes, use Data Domain systems to replicate backup sets to remote DR sites. Replicated backup sets allow for the elimination of Copy Storage Pools.
Access to mount point	Restrict access to a mount point to only the UNIX or Windows systems actually running the TSM server.

Miscellaneous Options	Settings
STK L180 changer	TSM changer driver
STK L180 LTO Tape drives	IBM Atape drivers***
TSM Server version level	Minimum 5.5.3.x****

*Configured on the AIX server only via the HBA level

**The recommended rsize/wsize is a minimum. Larger values may be used and have been seen to have beneficial results

***Refer to the latest VTL Data Domain systems compatibility matrix[1] for a listing of supported drivers

****This version allows the advantage of the RELABELSCRATCH parameter

Tivoli Storage Manager concepts and terminology

Fully utilizing Tivoli Storage Manager requires an understanding of a set of important definitions and parameters.

TSM device types

A device type is a fundamental requirement to build a storage pool where TSM stores data. TSM supports two major device classes.

- Random access storage device – DISK in TSM, usually on primary (tier 1) or secondary (tier 2) type storage.
- Sequential access storage device – Typically a FILE or TAPE, on formatted file systems.

[1] https://my.datadomain.com/US/en/compatibility.jsp

TSM deduplication

In TSM 6.0, IBM introduced post-process deduplication functionality on the server, and then in TSM 6.2, IBM introduced inline deduplication functionality on the client.

EMC recommends turning off native TSM deduplication when using Data Domain deduplication storage. For a detailed discussion, see the "TSM deduplication software" section on page 22.

TSM storage pools

A storage pool is a group of storage media of the same type on which data can be stored. Storage pools are defined based on a *device* class and each of the types of device class comes with a particular set of restrictions/requirements. Multiple storage pools can be created, each with a specific device type and associated policies. TSM controls the repositories and the flow of data among them.

Table 2 defines several types of storage pools.

Table 2. Types of storage pools

TSM Storage Pool Type	Description
Primary DISK (random access)	Primary storage pools receive backup and archive data written to them by the TSM server or client. These pools are typically configured to migrate data at certain capacity thresholds to *Primary* Sequential pools. These pools typically represent a "landing zone" or buffer of fast primary (tier 1) or secondary (tier 2) storage. Random access storage pool classifications are not recommended for Data Domain systems.
Primary Sequential (sequential media)	Primary storage pools receive backup and archive data written to them by the TSM server or client. These pools are typically configured to receive migrated data at certain thresholds from *Primary DISK* pools, but can be bypassed for larger backups or when the primary disk pool fills. These pools are typically sequential storage such as LTO-type tape media and file devices.
Active Data Pools (typically FILE device class or random-access disk)	Active data pools are storage pools that contain only active versions of client backup data. Data migrated by hierarchical storage management (HSM) clients and archive data is not permitted in active data pools. As updated versions of backup data are stored in active data pools, older versions are deactivated and removed during reclamation.
Copy Storage Pools	Copy storage pools are created by copying data from a primary storage pool. Two typical reasons exist for creating copy pools: • Protect against data loss due to media failure of a single piece of media • Maintain offsite copies of data for DR purposes

TSM policy domains

TSM has certain logical entities that group and organize the storage resources and define relationships between them. Client systems, or "nodes," are grouped together with other nodes having common storage management requirements into a policy domain.

The policy domain links the nodes to a policy set, which is a collection of storage management rules for different storage management activities. A policy set consists of one or more management classes. A

management class contains the rule descriptions, called copy groups, and links these to the data objects to be managed. A copy group is the place where all the storage management parameters, such as the number of stored copies, retention period, and target storage pools, are defined. When the data is linked to particular rules, it is said to be bound to the management class that contains those rules.

TSM progressive incremental backup

TSM backups, as compared to traditional backup software, typically use an incremental forever backup – or what TSM refers to as progressive incremental. In this scheme, the first time a given client or file system is backed up, all data is backed up, resulting in a traditional full backup. Each subsequent backup of that client or file system is incremental, meaning that TSM does not back up those files that have not changed. Although this backup method reduces network traffic and enables faster backups and more cost-effective media utilization, it requires proper configuration of the TSM environment within the available hardware resources in order to reduce the number of disparate volumes required during recovery.

TSM maintenance cycle

In a TSM environment, there are several administrative operations that should occur on a regular basis, usually in a specific sequence (see Figure 3). These administrative operations typically occur outside of the backup window and include database backup, expirations, reclamation, and migration. Because these administrative tasks require the use of the tape library and drives, scheduling can be problematic if not planned properly. If a client backup exceeds its window, some tasks in the administrative schedule may be compromised. Optimally, overlapping of the backup window and the daily administrative tasks should be avoided.

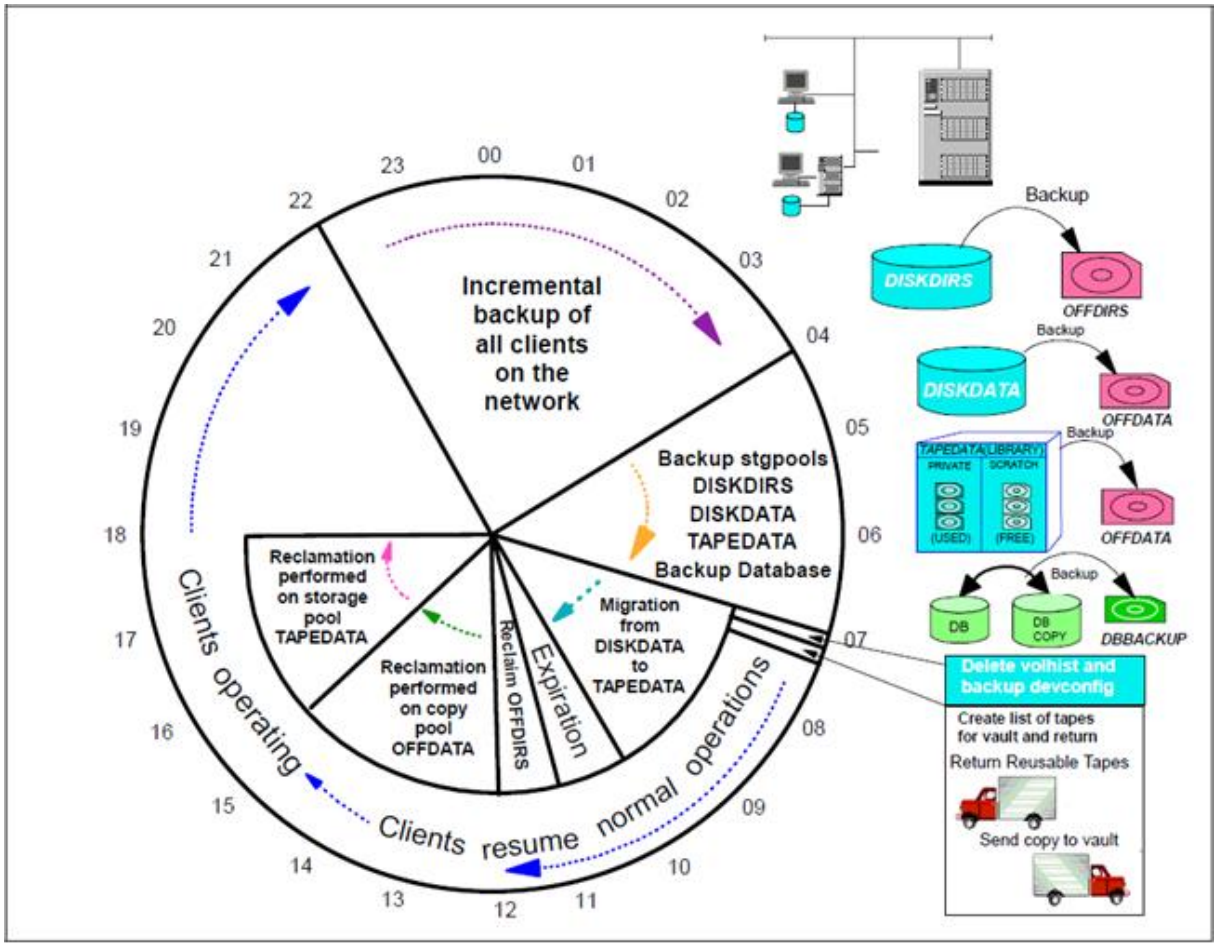

Figure 3. Scheduling of operations

Elements of the maintenance cycle

TSM database backup
TSM maintains a database of details about the backup objects (files, images, directories, volumes, and so on) and their associated management policies from each of its clients, as well as a mapping to the volumes containing the backup data. The database facilitates the ability to locate and recover data rapidly. TSM database backup is critical to the recovery of a TSM server.

TSM expiration process
The expiration process will remove backup and archive data entries in the database. It is important to note that the actual data is not removed from the storage pools; only the pointer in the database is deleted. The inventory expiration process can be run manually, automatically, or by schedule. By default, a TSM server will run this process daily but that can be controlled using the EXPINTERVAL parameter in the server options file, which specifies the number of hours between automatic expiration processing.

The inventory expiration process can be quite CPU-intensive, so the preferred method for running it is to use the TSM scheduler to define this operation to run at a pre-determined, convenient time.

TSM reclamation
Client data is retained based on defined policies. Because TSM is policy-based, backup objects, rather than the entire backup or individual piece of media, are expired. As backup objects on a volume expire, the volume contains less and less data actively tracked by the TSM database. Eventually the amount of data remaining on a volume drops below a predefined reclamation percentage threshold and needs to be moved to another volume within the same storage pool. The TSM reclamation process mounts the volume to be reclaimed, mounts another tape with free space, and copies all the remaining valid data from one volume to another. The volume that was the source for the reclamation is now empty and can be reused. This TSM reclamation process is usually scheduled to run once per day and executes according to an internal server algorithm that determines the appropriate list and order of volumes whose content is below the threshold. TSM reclamation ensures that data is stored efficiently to improve storage utilization and facilitate recovery.

TSM migration
Migration is a daily administrative process where backup objects are migrated (moved) automatically from one storage pool to another based on the pool's utilization thresholds. Typically, TSM administrators will design the storage hierarchy to back up data to a primary DISK device class pool (for performance) and then migrate the data to a primary (typically tape) storage pool during off-peak hours.

Data Domain deduplication storage systems background

Data Domain systems have a number of unique capabilities that are designed to directly address the challenges of using disk for data protection and DR. Data Domain inline deduplication breaks the incoming data stream into variable-length segments and uniquely identifies each one, then compares the segments to previously stored data. If the segment is unique, it is compressed and stored on disk along with associated metadata. If an incoming data segment is a duplicate of what has already been stored, only the metadata reference to the already-stored segment is kept. The Data Domain Data Invulnerability Architecture provides advanced data verification processes, including RAID 6 protection, continuous fault detection, healing, and write verification, to ensure maximum data integrity, availability, and recoverability. Finally, Data Domain Replicator (DD Replicator) software transfers only the deduplicated and compressed data across any IP network, requiring a tiny fraction of the bandwidth, time, and cost compared to traditional replication methods, enabling cost-effective DR. (For a detailed review of best practices for DR with Data Domain systems, see *IBM TSM Disaster Recovery with EMC Data Domain Deduplication Storage*.)

For more detailed information on Data Domain technology, please refer to the following technical white papers at www.datadomain.com/resources/whitepapers.html:

- *Data Domain SISL™ Scalability Architecture*
- *Data Domain Replicator Software*
- *Data Invulnerability Architecture: Ensuring Data Integrity and Storage System Recoverability*

Benefits of using Data Domain systems as a target for TSM

By eliminating redundant data segments inline, Data Domain systems allow many more backups to be retained for longer than would be possible using traditional storage or other deduplication techniques. The ability of the Data Domain system to store several weeks or months of full/incremental TSM backups and a configurable number of backup versions enables TSM backup administrators to implement a backup and recovery scheme with greater flexibility and protection while consuming a minimal amount of physical storage. Moreover, if the primary disk pool can be minimized or replaced by a Data Domain system, daily migration of data on *Primary DISK* device class pools to *Primary* sequential pools can be reduced or eliminated. The integration of Data Domain deduplication storage into a TSM environment is seamless since the Data Domain system presents itself either as a NFS/CIFS or a VTL storage server.

A best practice for enterprise environments is replication of Primary Data and TSM database backups to a secondary location. DD Replicator offers extremely bandwidth-efficient replication that is also easy to deploy, providing TSM backup administrators with excellent DR capabilities.

The primary benefit of DD Replicator is the fact that only deduplicated and compressed data is transferred across the network. Because deduplication is inline, replication takes place while the TSM backup process is still active. As the TSM backup process proceeds, the unique segments and metadata representing each file in the backup set are replicated to a remote site, allowing the overall "time-to-DR" to be minimized. In many cases, replication is completed very soon after the initial backup completes.

Allowing the Data Domain system to perform replication reduces the CPU and network requirements compared to what would be required if the TSM server itself was used. The replicated image occupies the same minimal footprint as the primary backup image, keeping overall infrastructure to a minimum. With the existence of the second replicated image, the TSM administrator no longer has the need for TSM to copy backup images to the *Copy Storage Pool*. This allows the administrator to reduce TSM operational maintenance and helps to accelerate operational tasks (see Figure 4 and Figure 5).

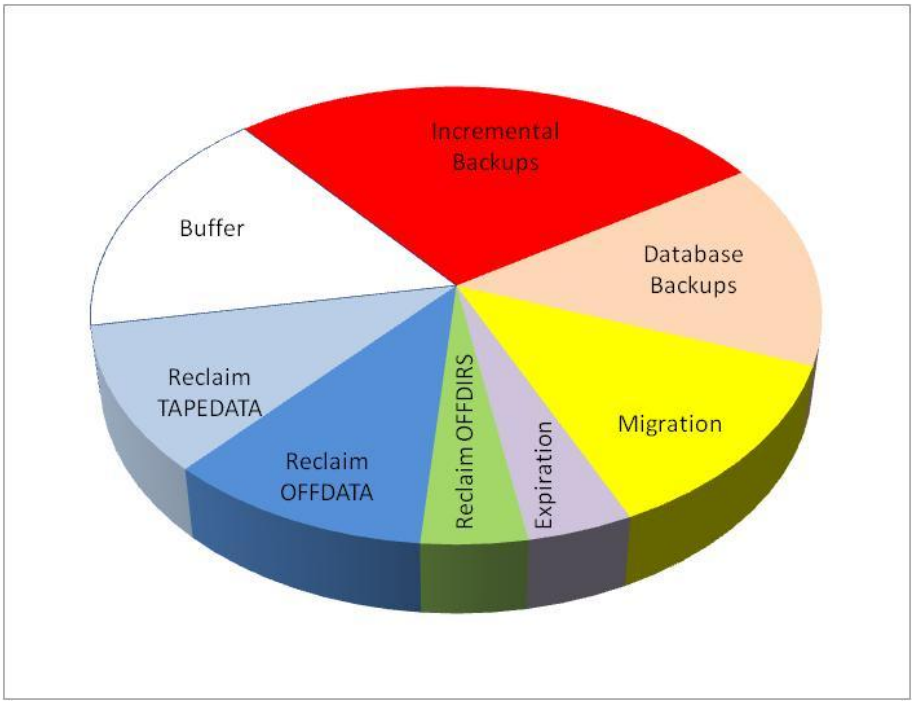

Figure 4. TSM 24-hour maintenance duty cycle

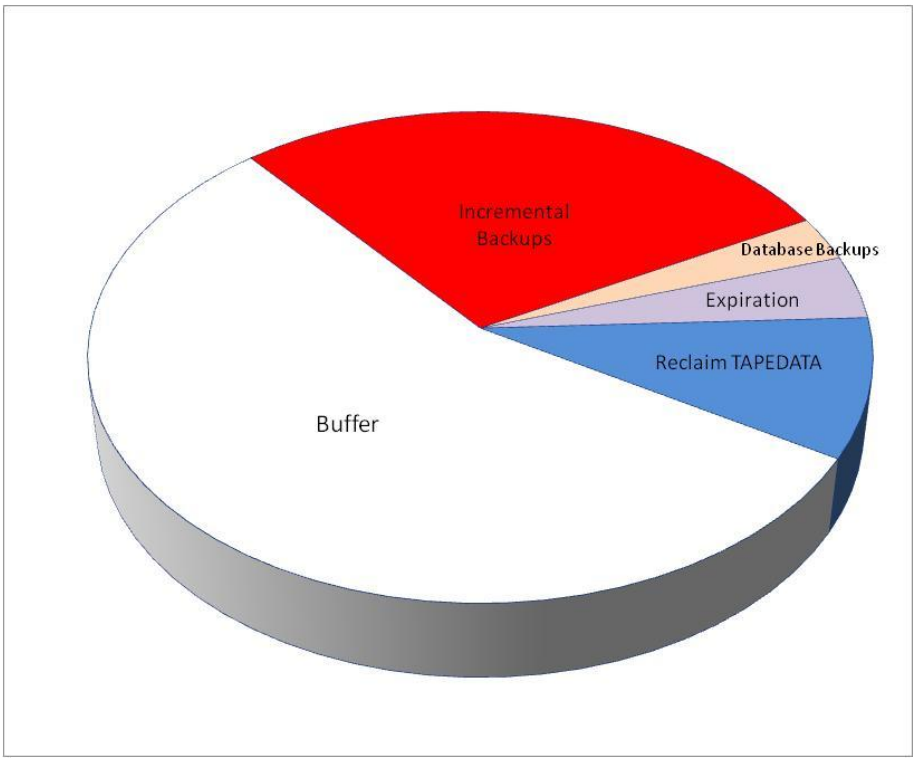

Figure 5. Accelerated TSM duty cycle with Data Domain systems

Deployment options

A Data Domain system can be deployed as a VTL or as a NAS device. Currently Data Domain Boost integration is not available with TSM.

The following section focuses on integrating Data Domain deduplication storage with TSM as a VTL and NAS device. Both of these options involve attaching the Data Domain system to an Ethernet network for NAS and FC SAN network for VTL.

VTL or NAS

When deciding whether to use VTL or NAS with TSM, consider the following points:

- Best practice is to deploy Data Domain systems as NAS devices; however, users can deploy as VTL if they need to leverage an existing FC infrastructure.
- A *FILE device* class via NFS exports from the Data Domain system avoids having to use primary storage as a backup target.
- When using NFS, reconfiguring TSM for different tape hardware at the DR site is not needed.

  Recovery is faster due to recovery from a FILE. The database restore process takes the physical path to the database backup as a parameter.
- When using NFS, there is no "virtual" cartridge handling.
  - No tape labeling and check-in
  - No defining virtual drives and virtual paths
  - No cartridges stuck in drives
  - No device driver issues
- Using NFS makes it easier to segregate your data and assign volume size by type. Reclamation time is reduced by sizing the volumes for each data type and/or retention
- Consistent retention periods for more rapid recycling of volumes

- Mount time is not needed for NFS. TSM requires mount time even for VTL cartridges — while each individual mount may not take long, substantial cumulative time is lost over thousands of mounts.
- NFS volumes allow for concurrent read and writes.

Best practices for TSM with Data Domain deduplication storage

Using a Data Domain system as a target for TSM backup is relatively straightforward since the system appears as normal disk storage or a tape library. However, planning for details such as network throughput, tape sizing, data segregation, SAN device discovery, replication bandwidth, and recovery operations is necessary to ensure that the entire system fulfills the requirements for backup windows, recovery time objectives, desired retention, ease of administration, and DR.

Networking

In theory, the faster the network and the greater the number of network paths, the faster data can move; however, there are other potential bottlenecks in the communication path that must be considered. For example, due to internal constraints, many clients and backup servers cannot put data onto a network at full line speed. Additionally, each Data Domain system has a rated limit for number of streams and throughput (see Table 3). Depending on the environment, TSM administrators can eliminate the *disk pool* as the primary target and just send the backups to the Data Domain system directly. Again this depends on the stream requirements in the existing environment.

✍ NOTE: Before making any architecture changes to the environment, it is always wise to check the Data Domain website[2] for your specific system model and the version of Data Domain Operating System (DD OS) that you are running to discern if the number of write streams is sufficient to meet your specific needs.

Table 3. Data Domain deduplication storage rated streams and throughput

Platform	Total	Max Write	Max Read	Mixed
DD880	180	180	50	<= 180 writes and <= 50 reads
DD690	90	90	50	<= 90 writes and <= 60 reads
DD660 DD670 DD690	90	90	30	<=90 writes and <= 30 reads
DD580	45	45	30	<= 45 writes and <= 30 reads
DD565 DD560	45	45	20	<= 45 writes and <= 20 reads
DD630	20	20	16	<= 20 writes and

[2] Data Domain product documentation website: https://my.datadomain.com/US/en/platform.jsp

				<= 16 reads
DD120 DD140 DD4xx DD510 DD530 DD610	16	16	4	<= 16 writes and <= 4reads

As a useful data point, a single gigabit Ethernet connection can move data at a maximum rate of 125 MB/s, so to achieve higher aggregate performance on a Data Domain system, more than one link would need to be configured and multiple write streams used. Multiple network connections can also be aggregated to improve throughput and/or to provide additional protection against link failures. Another approach is to use 10 Gb Ethernet in order to increase the throughput on a single connection.

Configure multiple independent network connections between TSM and the Data Domain system to maximize throughput. The use of multiple independent network connections avoids the bandwidth overhead associated with link aggregation protocols. The backup administrator must create a host-based route between the TSM server and the Data Domain system for each interface that will be utilized. Once that configuration is complete, multiple CIFS or NFS mounts are created, one for each network link. When defining the TSM device class, each mount point should be listed in the DIRECTORY= parameter.

Each directory is comma delimited.

"directory=/tsm/ddr/tsm_dir1,/tsm/ddr/tsm_dir2, /tsm/ddr/tsm_dir3"

This will allow the TSM server to round robin incoming data streams over each network link. There is no configuration control over path preferences, and in the event the link of a given path is not available, TSM will skip that path and open the next data stream on the next available path.

It is also recommended that these network interfaces be dedicated to TSM, in order to segregate the backup traffic for administrative or security reasons – this can be as simple as using directly connected network segments between TSM server hosts and the Data Domain system, or as complex as configuring a dedicated VLAN within a multi-tier network infrastructure. Best practices for tuning performance in a NAS environment can be found on the Data Domain support portal and should be followed for the specific environment being configured.

Security

Security with a Data Domain system is usually maintained by controlling access to the share points. By limiting hosts to specific directories on the Data Domain system, the TSM administrator can control access to the backup files. By setting UID/GID credentials (in a UNIX environment) or SID credentials (in a Windows Active Directory environment), TSM administrators can add additional controls. In either case, the correct setup is needed to ensure that TSM has access to the necessary path.

Administrators should restrict the shares (both NFS and CIFS) accessed directly by TSM to the servers actually running TSM server instances. In Microsoft environments, Active Directory integration will allow consolidated service credentials to be used across the entire deployment, simplifying security management.

Data segregation considerations

In general, segregate data stored in TSM by retention and data type. While TSM will allow mingling data of almost any type, in terms of the management of deduplication appliances, it is better to segregate data by

type in order to ease troubleshooting of deduplication/compression issues. Decisions on data segregation may vary by environments but some general segregation examples may be helpful:

- Oracle database-type backups are stored in TSM using the appropriate TSM Tivoli Data Protection (TDP) application.
- Backups of user space file systems are stored in the same TSM server using the TSM Backup/Archive (BA) client.

Typically in this segregation scenario, compression characteristics of the data will be different due to the type of data and the differences in the default TDP configuration and BA client backups.

If data is stored in separate structures on the Data Domain appliance, commands are available on the Data Domain system to give indicators of deduplication and compression performance for these data sets.

✍NOTE: Segregating data types will not change the overall deduplication/compression ratios, but it will provide the administrator with a high-level view to compare deduplication/compression performance among data types. Thus if a sudden drop in deduplication is encountered, the administrator can determine which data type is drawing down the average.

Optimizing deduplication

When dealing with any deduplication technology, a few key factors need to be kept in mind. Most important is the fact that any process that modifies a backup data stream on the fly is likely to be detrimental to the deduplication effect. This is because all deduplication techniques rely on being able to identify repeated data patterns. When backup streams of otherwise unchanged data are multiplexed, compressed, or encrypted before they get to the Data Domain system, the resultant data streams are different each time. This will cause less redundancy to be detected, resulting in lower deduplication rates, more physical storage used, and greater bandwidth needs for replication of each backup.

Multiplexing, compression, and encryption of backup data address the performance and security issues affecting backup to tape. Backing up to a Data Domain system is inherently more secure than tape (avoids chain-of-custody and handling issues in transporting tape media offsite) and does not suffer performance degradation due to slow backup streams. Thus, just as with native TSM deduplication, the recommendation is to disable all of these features when using a Data Domain system as the target for TSM. Encryption of data at rest is an option on Data Domain systems. Replication traffic can also be encrypted externally to provide security over a WAN, as deduplication occurs before transmission.

Volume size considerations

Selecting an appropriate volume size for a given data type in TSM can provide substantial performance and capacity benefits on deduplication storage.

As in the previous example of different data types, BA client data is typically in much smaller and disparately sized objects within TSM. As expiration occurs, the reclaimable space for a given volume will drop slower for typical file system data due to the variations of change and file sizes. Oracle-type data will be of a more consistent file size as the "chunk" size delivered to TSM is controlled by Oracle Recovery Manager (RMAN) settings.

A volume size can be selected for the storage pool containing the Oracle data such that when expiration occurs, full TSM volumes are emptied and are more likely to return immediately to scratch. This can avoid much of the expensive read/write process required of TSM reclamation.

Note that in a typical TSM environment you may never see a volume empty completely upon expiration and require no reclamation, but from a performance standpoint of both TSM and deduplication storage appliances, you will see better overall performance of both TSM and the storage if you size storage volumes appropriately.

SAN considerations

A Data Domain system has an option for a dual-port 4 Gb HBA FC card as a target mode for a SAN-attached configuration. All connections to these ports are made via a FC switch.

Dedicate a FC port on the initiator host (TSM server) to the Data Domain system VTL. Sharing a FC port with other devices other than a Data Domain system should be avoided whenever possible. Segregating the Data Domain storage system on a dedicated HBA on the TSM server simplifies troubleshooting of Data Domain VTL issues. It also reduces any impact on other devices that share that same HBA on the TSM server during the troubleshooting session.

VTL considerations

In a typical TSM environment, a single virtual tape library will be defined. TSM typically handles any drive sharing arrangement with better efficiency than configuring multiple VTLs with independent drive sets. In this type of configuration, TSM can segregate data by storage pool, but all storage pools will be in the same VTL pool.

Careful consideration of all aspects of the environment should come into play in making the decision to create multiple VTLs in order to segregate data by type.

EMC recommends the use of the L180 VTL emulation. The L180 library presents itself as a STK L180, which is a widely deployed library that is easily recognizable by TSM. TSM does not support the RESTORER-L180 VTL emulation.

Library manager

When TSM servers share a library, one server, the library manager, controls the device operations and handles client requests. These operations include mount, dismount, volume ownership, and library inventory. Other TSM servers, or library clients, use server-to-server communications to contact the library manager and request device service. Data moves over the SAN between each server and the storage device.

Library sharing in TSM

TSM library sharing requires a library manager, which:

- Controls all access to the library
- Is the only TSM server with robot control
- Manages all scratch pools for the other TSM library clients
- Handles all mount and dismount requests
- Defines all drive paths
  - The administrator is required to match server and client devices
  - The SAN DISCOVERY option is generally not recommended

EMC recommends sharing library and scratch pools, but not virtual drives, when using library manager.

Device discovery considerations

When an operating system scans its storage buses (FC or SCSI) for devices, it may assign a particular device to a different device name from the one it used during a previous scan. Since backup software saves information about the device names of the various tape drives it sees in its database during the configuration process, device reassignment during a scan will impact the availability of backup devices to the application.

Persistent binding is the ability to fix a particular device to a particular device name. In general, this is done by associating a unique device characteristic (such as a World Wide Port Name or serial number) with a

device name to maintain persistence of a particular device to a specific device name. The method for doing this varies for different operating systems and FC disk and tape device drivers.

If persistent binding is improperly configured, the operating system may provide a device with multiple names — giving a device one device name after a boot, but giving it a different name on a subsequent boot.

Virtual tape cartridge considerations

Data Domain VTL does not pre-allocate disk space when it creates virtual tape cartridges. As such, there are several factors to consider:

- The virtual tape cartridge size represents the amount of native format data that can be stored uncompressed by that tape.
- Although a large number of virtual tapes can be created, it is very important to take into consideration cartridge reuse. When too many tapes are available, TSM will continue to use scratch volumes rather than those that have been written to, but are not full yet. The Data Domain VTL file space cleaning process will not be able to recover space occupied by TSM reclaimed tapes until they are relabeled by TSM using the RELABELSCRATCH command from TSM.

As a best practice, it is recommended that the virtual cartridge be small in capacity. This allows a virtual tape to be filled, even when backing up smaller data sets. By having smaller cartridges and constantly filling them to their capacity, reclamation by TSM becomes more efficient.

Consider creating only enough media to accommodate the scratch requirements for the duration of the next operational period. Another best practice to consider is to configure TSM to use expired virtual tapes before using scratch tapes since expired tapes still consume space on the Data Domain system. Once a virtual tape has been reclaimed, it enters into a PENDING state. The virtual tape is not eligible for reuse (or deletion) until the number of days specified in the REUSEDELAY parameter of the parent storage pool has passed. By setting the parameter REUSEDELAY=0, the virtual tapes are returned quickly to the scratch pool for reuse. Spaces occupied by data in the expired tapes are not reclaimed by the Data Domain system until the expired tapes are reused (overwritten with new data or truncated by relabeling) and file system cleaning on Data Domain deduplication storage is performed. Expired space on virtual media does not allow consumed space on the Data Domain system to become eligible for cleaning until the virtual tape is also relabeled or the FILE volume is released at the OS level. Relabeling an expired tape truncates the volume and allows the Data Domain file system cleaning process to de-reference and subsequently delete the unique blocks associated with the backups on the virtual media.

Re-label scratch

When TSM deletes a volume, there is no direct communication of the event to the Data Domain system. As a result, the virtual tape representing that volume may display as empty or scratch in TSM, but this same tape will appear in the Data Domain system to contain data. This indicates the data on the scratch volume is still using space in the Data Domain system. To reclaim this space, TSM must re-label the volume by writing a data block to the beginning of the scratch volume. TSM can re-label volumes using the RELABELSCRATCH parameter. Support for this parameter is available in Fix Pack 5.5.2 and later levels. (Do not use this parameter in levels prior to 5.5.2.) For more information, see the article "RELABELSCRATCH parameter supported for DEFINE and UPDATE library commands (for Virtual Tape Libraries)" on the IBM website[3].

✍Note that this re-label command is only valid on VTL devices.

[3] www.ibm.com/support/docview.wss?uid=swg27012510

Reclamation

TSM reclamation occurs significantly faster with the Data Domain system than can be achieved with conventional tape drives. The time involved in mounting and positioning tapes to locate backup objects is streamlined because these activities become unnecessary with disk-based virtual media.

Make sure that volumes expired by TSM reclamation are also re-labeled as described above. This will allow the Data Domain system to recover the space previously occupied by those volumes and allow the system's filesystem cleaning process to remove unique blocks associated with those volumes that are no longer referenced by other volumes in the system. For file volumes, there is no need to re-label to reclaim space.

In addition to TSM reclamation, TSM administrators need to schedule the Data Domain system cleaning process to occur during hours outside the backup window. By default, the Data Domain system filesys clean process runs at 6 A.M. every Tuesday.

Virtual tape media barcode and tape drive

Create virtual cartridges only with unique upper-case barcodes. An error appears during an attempt to create a duplicate barcode in the same pool, but no error appears when creating duplicate barcodes in different tape pools; however, duplicate barcodes in different pools can result in operator confusion and anomalous backup application behavior.

One other observed TSM behavior is the failure to label a tape medium if an incorrect barcode suffix is used for the tape medium (see Table 4 for the correct barcode suffix to tape drive mapping).

When emulating tape drives, use the last two digit barcode suffixes mapped in Table 4.

Table 4. Barcode type to LTO drive mapping

Barcode Suffix (Tape Media Generations)	NEEDED
xxxxxxLA, xxxxxxLB, xxxxxxLC, and xxxxxxL1	LTO-1
xxxxxxLA, xxxxxxLB, xxxxxxLC xxxxxxL1, and xxxxxxL2	LTO-2
xxxxxxL2, xxxxxxL3	LTO-3

As an example of why barcode mapping of suffixes is important to LTO tapes, a LTO-3 drive will read and write data on a LTO-2 medium and read data on a LTO-1 drive. LTO tape technology is backward write compatible only one generation and backward read compatible only two generations. TSM manages these limitations by using only the barcode suffix on the tape.

TSM data migration considerations

There are many methods and timing criteria for introducing Data Domain systems into a TSM environment. The methodology required is usually dictated by outside concerns (that is, a tape library is going off lease and must be returned by a certain date).

Since TSM does incremental forever backups, TSM will rarely expire backup files. Sometimes, there are files that get backed up that will never change, such as system files. These system files will never expire because TSM will always maintain one copy of the active version, unless of course some kind of regular full backup is performed for all the clients. EMC recommends moving all the data from the old system to the Data Domain system. This will allow the introduction of new storage to TSM and convert normal daily backup jobs over to the Data Domain system.

If retirement of the old storage media is mandated before expiration of the data completes, there are several methods to transfer data from existing storage pools and volumes to storage pools and volumes on Data

Domain systems. All of these methodologies are internal to TSM and should be discussed with an EMC system architect or technical consultant.

Best practices for configuring TSM options

Compression

TSM has two methods of compression: client node compression and device compression. The TSM administrator can choose either one but not both.

Compression does not affect deduplication of files that do not change; however, TSM client-side compression does affect deduplication because the compression occurs on the fly during backup. Compression on the fly can cause a small change to ripple throughout the compressed backup streams, thus greatly reducing the commonality that can be found between current and prior versions, or similar but not identical documents on different clients. As a result, software client-side compression prevents optimal deduplication from taking place since the data that is already compressed does not deduplicate as efficiently.

As recommended above, disable any compression on backup jobs, whether it is client node compression or device compression. Backup data should not be compressed when transmitted to the Data Domain deduplication storage system. Compressing the data prior to sending it to the Data Domain system will most likely result in negligible redundancy and poor deduplication ratios. Sometimes client-side compression can actually cause compressed data to grow as a new compression header is added to the compressed data. Uncompressed data allows Data Domain deduplication storage to increase the matches the deduplication engine can find in the data stream. Data Domain systems compress the deduplicated data when writing to disk.

Other disadvantages of using client-side or server-side compression are as follows:

- Consumes CPU cycles on the file/application/backup server
- Increases backup window time
- Requires possible hardware upgrades on the client or backup server to handle additional load

Encryption

TSM can perform encryption of file data on the fly; however, like multiplexing and compression, encryption patterns are presented to the storage pool and adversely affect deduplication ratios on Data Domain systems.

It is therefore recommended that TSM client-side encryption not be used.

If encryption is necessary, the TSM administrator can utilize the optional encryption of the data-at-rest feature built into the Data Domain system. This Data Domain feature encrypts all incoming data to Data Domain deduplication storage before it is written to the physical storage media. The data is physically stored encrypted and cannot be accessed on the existing Data Domain system or any other environment without first decrypting it. This feature is independent of the TSM built-in encryption and is not aware of the encryption from the Data Domain system. For more information, see chapter 5 of the *Data Domain Administration Guide* at https://my.datadomain.com/.

Multiplexing

There are very few TSM clients that support multiplexing – typically, these are Oracle- and SAP-type data protection agents. Multiplexing is used to:

- Avoid shoe-shining on a physical tape drive
- Shorten the backup window by writing multiple backup streams from a client to a single drive

- Increase backup stream throughput by allowing the client to present the data to the TSM server from multiple client LUNs simultaneously

All but the first item (shoe-shining) apply to Data Domain systems.

Multiplexing interleaves backup streams, writing a little of data stream 1, then a little of datastream 2, and so forth, so that the client data is writing data from different files or tables from the database in, typically, a different order in every backup. Unfortunately, this interleaving of the data streams has a significant impact on deduplication efficiency. Multiplexed streams hinder the deduplication process from efficiently identifying blocks of common data because the data order is shuffled.

In order to realize the full benefit of deduplication, turn off multiplexing when using Data Domain deduplication storage. The performance lost by reducing or eliminating multiplexing can be easily regained through the additional mount points gained when using a Data Domain system.

AIX Dynamic Tracking

Currently, Data Domain VTL does not support this feature and thus this feature should be disabled. "Appendix B: How to verify Dynamic Tracking is enabled or disabled" shows how to check and disable the functionality.

TSM deduplication software

There are two types of backup software deduplication: client-side and server-side. Backup application software deduplication strips the redundant data before transferring it to the Data Domain deduplication engine. As a result, unless the software is compatible with DD Boost, client/server-side deduplication renders the Data Domain deduplication engine ineffective.

Since TSM does not yet support DD Boost, disable the client-side and server-side deduplication option when using Data Domain deduplication storage. Backup data should be transferred without any deduplication done by the TSM software before it reaches the Data Domain system.

TSM database backup

The TSM database is a critical component in the recovery of a TSM server. Currently, there is no way to rebuild a TSM database from just the TSM backup set. Therefore, it is necessary that TSM administrators back up their TSM database regularly. Best practice is to perform multiple full backups of TSM database daily and maintain copies of it onsite and snapshot copies offsite. For offsite replication, we recommend the use of Data Domain Replicator software. EMC strongly discourages putting the TSM database and logs directly on the Data Domain system because the TSM database presents data as non-sequential writes. Data Domain systems are optimized for sequential writes, thus the backup administrator needs to configure the database on primary disk storage and use the Data Domain system for the TSM database backup.

The recommended methodology for protecting the TSM database, even when using a CIFS or NFS, involves using a FILE device class.

- Create and use a FILE device class for the database snapshots and allow them to replicate to the remote Data Domain system. Ensure that all necessary files (PREPARE plan, and so on) are also replicated along with the database snapshot.
- An alternate method involves conducting database backups to the NFS or CIFS shares and allowing the volumes to be replicated to the remote Data Domain systems.

A secondary recommended method using VTL is as follows:

- Back up the TSM database to a VTL volume and recover the database using DD Replicator software as part of the VTL disaster recovery. The FILE device class (NFS or CIFS) method described above is the preferred method.

Conclusion

Planning, sizing, and integration strategies described in this paper make Data Domain systems a compelling solution for backup and DR optimization. Data Domain deduplication storage perfects the Tivioli Storage Manager data protection solution by optimizing TSM administrative tasks and containing the high operational costs that are associated with progressive backups and replication. This combination is extremely powerful and provides a best-of-breed data protection solution for enterprise-class data centers.

Appendix A: Useful TSM commands

The following is a list of commands that can be used quickly to configure Data Domain systems for data storage in TSM and a minimal configuration with at least one node to write data on the new storage pool.

Define a new FILE device class
tsm: SERVER1>define devclass DDR_DEV devtype=FILE directory=/<mount_point>/stgpool1 shared=no mountlimit=10 maxcap=2G

Define a Primary Pool on the Data Domain system
tsm: SERVER1>define stgp DDR_PRIMSTG1 DDR_DEV POOLTYPE=PRIM MAXSCRATCH=200

Define a new domain
Tsm: SERVER1>define domain DDR_TEST descry=”testdomain” bacjret=30 archret=30

Define a new policy in the new domain
Tsm: SERVER1> define policyset DDR_TEST DDR_TEST descry=”DDR_TEST Policy”

Define a new mgmtclass in the new policy
Tsm: SERVER1> define mgmtclass DDR_TEST DDR_TEST DDR_TEST Descr="DDR_TEST MGMTCLASS"

Define a BACKUP copygroup in the new mgmtclass
Tsm: SERVER1> define copygroup DDR_TEST DDR_TEST DDR_TEST STANDARD Type=Backup Destination=DDR_PRIMSTGP1 VERE=2 VERD=1 RETE=30 RETO=30

Define an ARCHIVE copygroup in the new mgmtclass
tsm: SERVER1>define copygroup DDR_TEST DDR_TEST DDR_TEST STANDARD Type=Archive Destination=DDR_PRIMSTGP1 RETVER=30

Assign the default mgmtclass in the policy
SERVER1>ASsign DEFMGmtclass DDR_TEST DDR_TEST DDR_TEST

Validate the policy
tsm: SERVER1>validate policy DDR_TEST DDR_TEST act pol DDR_TEST DDR_TEST

Activate the policy
tsm: SERVER1>activate policy DDR_TEST DDR_TEST
Do you wish to proceed? (Yes (Y)/No (N)) y
ANR1514I Policy set DDR_TEST activated in policy domain DDR_TEST.

Register a new node in the new domain
tsm: SERVER1>register node DDR_TEST 0000 domain=DDR_TEST url=" user=none

Query within TSM

Device class

```
tsm: SERVER1>query devcl ddr_dev f=d
          Device Class Name: DDR_DEV
      Device Access Strategy: Sequential
         Storage Pool Count: 0
                Device Type: FILE
                     Format: DRIVE
      Est/Max Capacity (MB): 2,048.0
                Mount Limit: 10
           Mount Wait (min):
      Mount Retention (min):
             Label Prefix:
                  Library:
                Directory: /nfs/…/stgp1
              Server Name:
             Retry Period:
            Retry Interval:
                   Shared:
         High-level Address:
          Minimum Capacity:
                    WORM: No
          Drive Encryption:
           Scaled Capacity:
Last Update by (administrator): ADMIN
      Last Update Date/Time: 11/10/2008 11:18:29
```

Storage Pool

```
tsm: SERVER1>query stgp DDR_PRIMSTGP1 format=detail
            Storage Pool Name: DDR_PRIMSTGP1
            Storage Pool Type: Primary
            Device Class Name: DDR_DEV
           Estimated Capacity: 0.0 M
           Space Trigger Util: 0.0
                     Pct Util: 0.0
                     Pct Migr: 100.0
                  Pct Logical: 0.0
                 High Mig Pct: 90
                  Low Mig Pct: 70
               Migration Delay: 0
            Migration Continue: Yes
           Migration Processes: 1
         Reclamation Processes: 1
             Next Storage Pool:
          Reclaim Storage Pool:
       Maximum Size Threshold: No Limit
                       Access: Read/Write
                  Description:
            Overflow Location:
        Cache Migrated Files?:
                  Collocate?: Group
        Reclamation Threshold: 60
      Offsite Reclamation Limit:
Maximum Scratch Volumes Allowed: 200
 Number of Scratch Volumes Used: 0
  Delay Period for Volume Reuse: 0 Day(s)
```

Migration in Progress?: No
Amount Migrated (MB): 0.00
Elapsed Migration Time (seconds): 0
Reclamation in Progress?: No
Last Update by (administrator): ADMIN
Last Update Date/Time: 11/10/2008 11:20:06
Storage Pool Data Format: Native
Copy Storage Pool(s):
Active Data Pool(s):
Continue Copy on Error?: Yes
CRC Data: No
Reclamation Type: Threshold
Overwrite Data when Deleted:

Appendix B: How to verify Dynamic Tracking is enabled or disabled

How to check if Dynamic Tracking is enabled

-bash-3.2# lsattr -El fscsi0
(Assuming the fscsi device instance is fscsi0.)

attach	switch	How this adapter is CONNECTED	True / False
dyntrk	yes	Dynamic Tracking of FC Devices	True
fc_err_recov	delayed_fail	FC Fabric Event Error RECOVERY Policy	True
scsi_id	0x21800	Adapter SCSI ID	False
sw_fc_class	3	FC Class for Fabric	True

From the example above, “Dynamic Tracking” is enabled.

How to disable Dynamic Tracking

-bash-3.2# chdev –l fscsi0 –a dyntrk=no
(Assuming the fscsi device instance is fscsi0.)