Copyright

Information in this document, including URL and other website references, represents the current view of CommVault Systems, Inc. as of the date of publication and is subject to change without notice to you.

Descriptions or references to third party products, services or websites are provided only as a convenience to you and should not be considered an endorsement by CommVault. CommVault makes no representations or warranties, express or implied, as to any third party products, services or websites.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious.

Complying with all applicable copyright laws is the responsibility of the user. This document is intended for distribution to and use only by CommVault customers. Use or distribution of this document by any other persons is prohibited without the express written permission of CommVault. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of CommVault Systems, Inc.

CommVault may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from CommVault, this document does not give you any license to CommVault’s intellectual property.

COMMVAULT MAKES NO WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, AS TO THE INFORMATION CONTAINED IN THIS DOCUMENT.

©1999-2015 CommVault Systems, Inc. All rights reserved

CommVault, CommVault and logo, the “CV” logo, CommVault Systems, Solving Forward, SIM, Singular Information Management, Simpana, CommVault Galaxy, Unified Data Management, QiNetix, Quick Recovery, QR, CommNet, GridStor, Vault Tracker, InnerVault, QuickSnap, QSnap, Recovery Director, CommServe, CommCell, IntelliSnap, ROMS, Simpana OnePass, CommVault Edge and CommValue, are trademarks or registered trademarks of CommVault Systems, Inc. All other third party brands, products, service names, trademarks, or registered service marks are the property of and used to identify the products or services of their respective owners. All specifications are subject to change without notice.

All right, title and intellectual property rights in and to the Manual is owned by CommVault. No rights are granted to you other than a license to use the Manual for your personal use and information. You may not make a copy or derivative work of this Manual. You may not sell, resell, sublicense, rent, loan or lease the Manual to another party, transfer or assign your rights to use the Manual or otherwise exploit or use the Manual for any purpose other than for your personal use and reference. The Manual is provided "AS IS" without a warranty of any kind and the information provided herein is subject to change without notice.
# Table of Contents

Module 1: Course Introduction ........................................................................................................... 7
  Course Objectives .............................................................................................................................. 8
  Course Design Strategy ....................................................................................................................... 9
  Education Lifecycle Matrix .............................................................................................................. 10
  Master Certification ............................................................................................................................ 11
  Course Agenda .................................................................................................................................. 12

Module 2: Common Technology Engine .......................................................................................... 13
  Common Technology Engine Primer .................................................................................................. 14
  Physical and Logical Layers ............................................................................................................. 15
  Processes Overview .......................................................................................................................... 16
  Base Services ................................................................................................................................... 17
  CommServe® Server ........................................................................................................................... 19
    CommServe® Primer ........................................................................................................................ 20
    CommServe® Processes .................................................................................................................. 21
    CommServe® DR Backup Process ................................................................................................... 23
  MediaAgents and Indexing .................................................................................................................. 26
    MediaAgent Primer .......................................................................................................................... 27
    MediaAgent Processes ..................................................................................................................... 29
    Indexing Primer ............................................................................................................................... 31
    Indexing Processes .......................................................................................................................... 33
    Index Cache Structure ...................................................................................................................... 34
    Libraries .......................................................................................................................................... 35
    MediaAgent Placement and Scalability ............................................................................................ 37

Storage Policies .................................................................................................................................... 38
  Storage Policy Primer ........................................................................................................................ 39
  GridStor® Technology ....................................................................................................................... 42
  Storage Policy Data Path Configuration ............................................................................................ 44
  Advanced Storage Policy Configurations ........................................................................................... 46
  Storage Policy Stream and Performance Settings .............................................................................. 49
  Storage Policy Design Models .......................................................................................................... 51

Module 3: Deduplication ...................................................................................................................... 52
Deduplication Primer .................................................................................................................. 53
Deduplication Building Blocks ................................................................................................. 55
Partitioned Deduplication Databases ....................................................................................... 57
Deduplication Database Components and Processes ................................................................. 59
Deduplication Data Movement Processes .................................................................................. 61
  Deduplication Enabled Data Protection .................................................................................. 62
  DASH Full Process .................................................................................................................. 63
  DASH Copy Process ............................................................................................................... 64
Aging Deduplicated Data ........................................................................................................... 66
Deduplication Data Aging Primer .............................................................................................. 67
Data Aging Workflow ................................................................................................................ 68
Data Aging Log Files ................................................................................................................ 70
Deduplication Design Strategies ............................................................................................... 75
  Deduplication Storage Policies .............................................................................................. 76
  Standard Building Block Deployment ..................................................................................... 78
  Dedicated MediaAgent for DDB ............................................................................................ 79
  SILO Storage .......................................................................................................................... 80
Module 4: Virtualization ........................................................................................................... 82
  Virtual Data Protection Methods ............................................................................................ 83
  VSA Data Protection Process ................................................................................................. 84
    VMware Transport Modes .................................................................................................... 85
    Raw Device Mapping (RDM) ............................................................................................... 87
  VSA Design and Configuration .............................................................................................. 89
    VSA Design Considerations ............................................................................................... 90
    VSA Agent Configuration ................................................................................................. 91
Module 5: IntelliSnap® Technology ......................................................................................... 102
  Snapshot Technology ............................................................................................................ 103
    Copy on Write and Allocate on Write ................................................................................ 104
    Application and Crash Consistency .................................................................................... 106
  IntelliSnap® Technology Processes for VSA Part 1 ............................................................... 108
  IntelliSnap® Technology Processes for VSA Part 2 ............................................................... 109
  Configuring and Administering IntelliSnap® Technology ..................................................... 110
Module 1: Course Introduction
Course Objectives

• To provide the deepest level of customer based technical training available for the Simpana® Product Suite.
  • Provide deep level understanding of the Common Technology Engine (CTE) including: processes, configurations, log files and troubleshooting.
  • Advanced education on the most common CommVault® features including: deduplication, virtualization, snapshots, firewalls, Simpana OnePass™ backup/archive.
  • Provide advanced concepts for data and information management strategies.
• Facilitate the transfer of knowledge to adequately design, configure, administer and troubleshoot a CommCell® environment.
• Prepare learners for the CommVault Master Certification exam.
Course Design Strategy

- To conceptualize complex subjects and processes to maximize retention of information beyond the five day course.
- Repetitive reinforcement of key concepts throughout course to solidify knowledge.
- Organize subjects and topics to maximize conversation by relating to a specific aspect of: theory, design, configuration or troubleshooting.
- Encouragement of class discussions to engage all learners and allow knowledge transfer between all participants.
This chart illustrates the education lifecycle from the point an individual starts their career using Simpana® software to the point they achieve Master certification level. It provides basic guidance on topics required to achieve Professional certification status, current and future areas of specialization, and required knowledge to attain Master status.
Master Certification

• To become certified you must:
  • Currently be a CommVault Certified Professional and Specialist
  • Successfully pass the Master Certification exam
• Master Certification exam details:
  • Questions: 75
  • Time: 180 minutes
  • Open Book: YES
  • Passing Score: 80%
  • Is the test hard? YES! Even with the test being open book, the allotted time restricts the ability to research every question
• Benefits:
  • Gain more points in CommVault® Advantage
  • Distinguish yourself from others by attaining the highest certification level possible for a CommVault professional.
Course Agenda

Day 1: Common Technology Engine
Day 2: Storage and Deduplication
Day 3: Virtualization and Snapshots
Day 4: Data and Information Management
Day 5: Review and Exam
Module 2: Common Technology Engine
The CommServe® server coordinates all activity within a CommCell® environment. Data protection jobs (snapshots, backups, archive / OnePass) are initiated from the CommServe server by communicating with the client. For backup and archive operations a data pipe will be established from the client to the MediaAgent. For snapshot operations, MediaAgent processes will be used to communicate with the array and conduct and manage snapshot operations.

Deduplication processes will be used on the client to optionally compress data and then a signature will be generated on the data block. The block can also optionally be encrypted over the network or on media. Index data for each job will be managed in the MediaAgent’s index cache and will also be copied to protected storage when the job completes.
Physical and Logical Layers

The Simpana software suite is configured and managed at both a physical and logical level. Understanding this concept is important in understanding how Simpana software works.

The Physical View:

- The CommServe server centrally manages all communication and data movement within the CommCell environment.
- MediaAgents are responsible for moving data from source to destination.
- Libraries are storage devices that will hold protected data.
- Clients are any production servers managed by the Simpana software.

The Logical View:

- Agents are software modules installed to manage production data requiring protection.
- Data sets are defined to represent all data the Agent is responsible to protect.
- Subclients are used to define the actual content requiring protection.
- Storage Policies are used to manage protected data throughout its protected lifecycle.
- From a physical perspective clients and MediaAgents communicate with the CommServe server. Libraries are configured and connected to the MediaAgents. From a logical perspective Agents are installed to manage client data. Within the Agents, data sets and subclients are configured to granularly manage content. The subclients are associated with a storage policy which is used to direct subclient data through a MediaAgent and to a library.
Processes Overview

Processes are designed to serve specific purposes and may run on the CommServe server, MediaAgent, client or on all systems. Each process will correspond to one or more log files which log activity through the various phases of Simpana operations.

This is a high level example of the data movement process:

1. The JobMgr process will initiate a data protection operation.
2. The iFind process will begin scanning data based on the content definitions of the subclient.
3. The MediaManager process will communicate with the MediaAgent and the CVMountD process will access storage resources attached to the MediaAgent.
4. IndexingService process will either create a new index folder or gain access to the most recent index.
5. The CLBackup process will communicate with the CVD process to establish a data pipe between the client and the MediaAgent.
6. Events on the client will be sent using the EvMgrC to the EvMgrS process.
7. The EvMgrS process will feed update and configuration information to the CommCell console.
# Base Services

Base services will exist on all systems in which CommVault software is installed. These services provide the foundation in which the Common Technology is based on.

## Cvd (CVD.log cvfwd.log)

The CVD process provides the base communication which controls connectivity, firewall access, patch information, Pre/Post process execution and space checks.

For data protection and recovery jobs the CVD process will be used to assist in establishing data pipes from source to destination.

## EvMgrC (EvMgrC.log)

The EvMgrC (Event Manager Client) is used to forward events and conditions from the local machine to the CommServe server and is also used to assist in browsing application data on the local host.

## InstallUpdates (UpdateInfo.log)

The InstallUpdates process is used to install updates on the local machine and verify patch information with the local registry.

## Qlogin (qcommand.log on CommServe)

Qlogin is used to provide command line login access and execute scripts on the local machine.

---

<table>
<thead>
<tr>
<th>Process / Log</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cvd (CVD.log cvfwd.log)</td>
<td>Provides the base communication which controls connectivity, firewall access, patch information, Pre/Post process execution and space checks.</td>
</tr>
<tr>
<td>EvMgrC (EvMgrC.log)</td>
<td>Used to forward events and conditions from the local machine to the CommServe server and is also used to assist in browsing application data on the local host.</td>
</tr>
<tr>
<td>InstallUpdates (UpdateInfo.log)</td>
<td>Used to install updates on the local machine and verify patch information with the local registry.</td>
</tr>
<tr>
<td>Qlogin (qcommand.log on CommServe)</td>
<td>Provides command line login access and execute scripts on the local machine.</td>
</tr>
<tr>
<td>Qlogout (qcommand.log on CommServe)</td>
<td>Qlogout is used to terminate any script processes and log out.</td>
</tr>
</tbody>
</table>
Qlogout (qcommand.log on CommServe)

Qlogout is used to terminate any script processes and log out.
CommServe® Primer

The CommServe® server is the central management system within a CommCell environment. All activity is coordinated and managed by the CommServe server. The CommServe system runs on a Windows platform and maintains a Microsoft SQL metadata database. This database contains all critical configuration information. It is important to note that Simpana software does not use a centralized catalog system like most other backup products. This means the metadata database on the CommServe server will be considerably smaller than databases that contain catalog data. Due to the small size of the database, an automated backup of the database is executed by default every morning at 10:00 AM.

Key points regarding the CommServe server:

- For CommServe server high availability the following options are available:
  - The CommServe server can be clustered – This is recommended for larger environments where high availability is critical.
  - The CommServe can be virtualized – This is suitable for smaller environments.
- It is ABSOLUTELY CRITICAL that the CommServe database is properly protected. By default every day at 10 AM a CommServe DR backup job is conducted. This operation can be completely customized and set to run multiple times a day if required.
- All activity is conducted through the CommServe server; therefore it is important that communication between the CommServe server and all CommCell environment resources always be available.
CommServe® Processes

**AppMgrSvc (AppMgrService.log)**

AppMgrSvc provides access to client configuration information.

**ArchPrune (DataAging.log)**

The ArchPrune.exe process is initiated during data aging operations to clear out data that has exceeded retention. Job information for aged jobs is sent to the MediaAgent for pruning operations.

**AuxCopyMgr (AuxCopyMgr.log)**

The AuxCopyMgr process is responsible for communicating with the AuxCopy process on source and destination MediaAgents to control auxiliary copy operations. It controls auxiliary copy jobs and sends information on what chunk data is required to be copied.

**CommServeDR (CommserveDR.log)**

The CommServeDR process is responsible for coordinating both phases of the CommServe DR backup process.

**CopyToCache (CopyToCache.log)**

The CopyToCache process is responsible for copying updates to secondary cache locations.
DistributeUpdates (DistributeSoftware.log)

The DistributeUpdates process is responsible for pushing updates to client servers. It also coordinates activity on the client using the InstallUpdates and RemoveUpdates processes.

DownloadUpdates (DownloadSoftware.log)

The DownloadUpdates process is responsible for downloading service packs and packages from the central FTP location to the primary update cache location.

EvMgrS (EvMgrS.log)

The EvMgrS is responsible for receiving messages from the EvMgrC and feeding information to the CommCell console.

IndexingService (StartRestore.log StartSynthFull.log)

The IndexingService on the CommServe server is responsible for coordinating restore and synthetic full operations.

JobMgr (JobManager.log)

The JobMgr.exe process is responsible for initiating and controlling jobs, and communication with storage resources. It acts as the primary coordinator for all data movement operations and the JobManager.log is typically the first log to view when troubleshooting data movement problems. All starting and stopping of processes during a data movement operation will be logged in the JobManager.log.

JobMgr process example: Auxiliary Copy Job

The JobMgr will initiate the auxiliary copy job by communicating with the source MediaAgent to reserve storage resources for the source job. It will then communicate with the destination MediaAgent to reserve destination storage resources. It will then communicate with the AuxCopyMgr.exe to generate required data for the auxiliary copy job. Once the auxiliary copy job has completed the JobMgr.exe will then report the job as complete.

Note that in this example not only will the JobMgr communicate with the AuxCopyMgr but also communicate with both the source and destination MediaAgents to allocate storage resources.
CommServe® DR Backup Process

By default every day at 10:00 AM the CommServe DR backup process is executed. This process will first dump the CommServe SQL database and the registry hive to the:

<install path>\CommVault\Simpana\CommServeDR folder.

An Export process will then copy the folder contents to a user defined drive letter or UNC path. A Backup phase will then back up the DR Metadata and any user defined log files to a location based on the storage policy associated with the backup phase of the DR process. All processes, schedules and export/backup location are customizable in the DR Backup Settings applet in Control Panel.

Export
The Export process will copy the contents of the \CommServeDR folder to the user defined export location. A drive letter or UNC path can be defined. The export location should NOT be on the local CommServe server. If a standby CommServe server is available define the export location to a share on the standby server.

By default five metadata backups will be retained in the export location. It is recommended to have enough disk space to maintain one weeks’ worth of DR exports.

Backup
The Backup process is used to back up the DR metadata to protected storage. This is accomplished by associating the backup phase with a storage policy. A default DR storage policy is automatically created...
when the first library is configured in the CommCell environment. Although the backup phase can be associated with a regular storage policy it is recommended to use a dedicated DR storage policy to protect the DR metadata.

**DR Storage Policy**

When the first library in a CommCell environment is configured a CommServe Disaster Recovery storage policy will automatically be created. The Backup phase of the DR backup process will automatically be associated with this storage policy. If the first library configured is a disk library and a tape library is subsequently added, a storage policy secondary copy will be created and associated with the tape library.

There are several critical points regarding the DR storage policy and backup phase configurations:

- Although the backup phase can be associated with any storage policy in the CommCell environment, it is recommended to use a dedicated DR storage policy. Using a dedicated policy will isolate DR metadata on its own set of media making it potentially easier to locate in a disaster situation.
- The most common reason the backup phase is associated with regular data protection storage policies is to reduce the number of tapes being sent off-site. If the backup phase is associated with a regular storage policy consider the following key points:
  - Make sure the Erase Data feature is disabled in the storage policy. If this is not done the DR metadata will not be recoverable using the Media Explorer utility.
  - When secondary policies are created in the Associations tab of the copy, an option for the DR metadata will be available. Make sure every secondary copy contains the DR metadata.
  - Make sure you are properly running and storing media reports. This is especially important when sending large numbers of tapes off-site. If you don’t know which tape the metadata is on you will have to catalog every tape until you locate the correct media which is storing the DR metadata.

**Backup Frequency**

By default the DR backup will run once a day at 10:00 AM. The time the backup runs can be modified and the DR backup can be scheduled to run multiple times a day or saved as a script to be executed on demand. Consider the following key points regarding the scheduling time and frequency of DR backups:

- If tapes are being sent off-site daily prior to 10:00 AM then the default DR backup time is not adequate. Alter the default schedule so the backup can complete and DR tapes be exported from the library prior to media being sent off-site.
- The DR Metadata is essential to recover protected data. If backups are conducted at night and auxiliary copies are run during the day, consider setting up a second schedule after auxiliary copies complete.
- For mission critical jobs consider saving a DR backup job as a script. The script can then be executed by using an alert to execute the script upon successful completion of the job.
Locations
Multiple copies of the DR backup can be maintained in its raw (export) form using scripts. Multiple copies of the backup phase can be created within the DR storage policy by creating secondary copies or a data backup storage policy by including the metadata in the secondary copy Association tab. Follow these guidelines for locating the DR metadata backups.

- On-site and off-site standby CommServe servers should have a raw (export) copy of the metadata.
- Wherever protected data is located, a copy of the DR metadata should also be included.
- Whenever protected data is sent off-site a copy of the DR metadata should be included.
- Since DR metadata does not consume a lot of space copies should be kept as long as possible.

Retention
By default the export phase will maintain five copies of the metadata. A general recommendation is to maintain a weeks’ worth of metadata exports if disk space is available. This means if the DR backup is scheduled to run two times per day then 14 metadata backups should be maintained.

For the metadata backup phase, the default storage policy retention is 60 days and 60 cycles. A general best practice is that the metadata should be retained based on the longest data being retained. If data is being sent off site on tape for ten years, a copy of the DR database should be included with the data.

Metadata Security
Securing the location where DR Metadata is copied to is critical since all security and encryption keys are maintained in the CommServe database. If the metadata is copied to removable drives or network locations, best practices recommend using disk based encryption.
MediaAgents and Indexing
MediaAgent Primer

The MediaAgent is the high performance data mover which moves data from source to destination. It is a software module that can be installed on most operating systems. All of its tasks are coordinated by the CommServe server. The MediaAgent moves data from a client to a Library during a data protection operation or vice-versa during data recovery. MediaAgents are also used during auxiliary copy jobs when data is copied from a source library to a destination library.

There is a basic rule that all data must travel through a MediaAgent to reach its destination. One exception to this rule is when conducting NDMP dumps direct to tape media. In this case the MediaAgent is used to execute the NDMP dump and no data will travel through the MediaAgent. This rule is important to note as it will affect MediaAgent placement.

Simple Scalable Data Transport (SDT) Pipeline

The SDT pipeline is designed to optimize MediaAgent resource allocation when protecting many streams over a network connection. The MediaAgent setting ‘Optimize for concurrent LAN backup’ is used to enable or disable the SDT pipeline and is enabled by default.

Data Pipe

MediaAgents can be used to backup client data over a network or dedicated where a client and MediaAgent are installed on the same server using a LAN Free or preferred path to backup data directly to storage.
**Physical vs. Virtual MediaAgent**
CommVault recommends using physical MediaAgents to protect physical and virtual data. The advantages for using a physical MediaAgent are: better performance, more versatility as a multi-purposed data mover (protect VMs and physical data), and resiliency. A MediaAgent can be virtualized if all performance requirements including CPU, RAM, index cache location and deduplication database location are being met.
MediaAgent Processes

SynthFull (SynthFull.log)

The SynthFull process coordinates the restore and backup operations for Synthetic full backups.

ArchiveIndex (archiveindex.log)

The ArchiveIndex process is responsible for compacting the index and writing index to storage. Prior to version 10 it was also responsible for cleaning up index cache. In version 10 this process is handled by the IndexingService process.

AuxCopy (AuxCopy.log)

The AuxCopy process receives direction from the AuxCopyMgr (CommServe) and reads chunk data to be processed during auxiliary copy job. AuxCopy process on the destination MediaAgent receives chunk information and reports status updates on job back to the AuxCopyMgr process.

CVMountD (CVMA.log and SIDBPhysicalDelete.log)

The CVMountD process interacts with hardware storage devices attached to the MediaAgent.

IndexingService (CreateIndex.log and UpdateIndex.log)

The IndexingService process creates a new index or gains access to the most recent index.
Media Management Process Sample Workflow

1. JobManager initiates backup request
2. JobManager Initiates iFind process to perform scan phase
3. After scan completes JobManager reserves stream and storage resources
4. CVD starts a pipeline with the Media Agent
5. MediaManager receives mount request from CVD
6. CVMountD creates volume folder (revoke permissions if prevent accidental deletion of data from mount path is enabled)
7. When CVD log receives successful mount notification it prepares to write data to storage
8. MediaManager receives unmounts request
9. MediaManager receives mount request for archive index phase
10. JobManager receives notification on unmounts and job completion
Indexing Primer

- Dual Level Index
  - Job metadata maintained in CommServe® database.
  - Indexes in second tier contains information of individual protected objects.
    - Object – Media – Chunk – Offset
- Job restart vs. job chunk point restart
- Chunk size and index update
- Index Cache Configuration – Local Index, shared Index or Index Cache Server.
- Indexing Processes:
  - Simpana® IndexingService serves as both the CreateIndex and UpdateIndex.
    - CreateIndex – Create/Prepares the index.
    - UpdateIndex – Updates the objects during backup.
  - ArchiveIndex – Archives the index to media.

Simpana® Indexing Overview
CommVault uses a 2-tiered distributed indexing structure to index protected data. The top tier contains Job Summary data and is maintained in the SQL database on the CommServe® server. This data includes job information, chunks written to media, and logs media the job has been written to. This information is useful when locating media for restores. The summary index information for a job will be removed from the database once the data exceeds retention and has been pruned or overwritten.

The second tier contains the distributed indexes called the Index Cache. This cache will maintain index files for all jobs the media agent manages. Each subclient will have their own index files. Each time a full data protection operation is executed, a new index will be created. When dependent jobs are run (incremental or differential) the index files will be appended to, to include new indexing information. Each index cache will contain many small index files which can be individually managed, protected, and pruned.

How Indexing Works
Job summary data maintained in the CommServe database will keep track of all data chunks being written to media. As each chunk completes it is logged in the CommServe database. This information will also maintain media identities where the job was written to which can be used when recalling off site media back for restores. This data will be held in the database for as long as the job exists. This means even if the data has exceeded defined retention rules, the summary information will still remain...
in the database until the job has been overwritten. An option to browse aged data can be used to browse and recover data on media that has exceeded retention but has not been overwritten.

The detailed index information for jobs is maintained in the MediaAgent’s Index Cache. This information will contain each object protected, what chunk the data is in and the chunk offset defining the exact location of the data within the chunk. The index files are stored in the index cache and after the data is protected to media, an archive index operation is conducted to write the index to the media. This method automatically protects the index information eliminating the need to perform separate index backup operations. The archived index can also be used if the index cache is not available, when restoring the data at alternate locations, or if the indexes have been pruned from the index cache location.

One major distinction between the Simpana® software and other backup products is Simpana’s use of a distributed self-protecting index structure. The modular nature of the indexes allows the small index files to automatically be copied to media at the conclusion of data protection jobs. This means that separate backups of the index cache are not necessary.
Indexing Processes

Indexing Service
Creates a new index or gains access to the most recent index files in the index cache. A new directory is created:

- When creating a new index.
- Copying index data from old directory with a new time stamp.
- Restoring index files from the archive index.

Indexing service is also used to update index cache data at chunk boundaries. The DataPipe tail sends information to UpdateIndex logs about files and folders being protected (path, archive file and offset).

ArchiveIndex
After backup phase completes ArchiveIndex consolidates index information into an index archive file.

- Index information is held in index cache for future operations.
- Most recent information will be retained in cache and previous index files will be deleted to save space.
- ArchiveIndex also checks against index cache thresholds to prune index files from cache based on subclient job being executed. Based on the rules any index files for only the subclient being run will be pruned.
The index cache structure can be viewed in the Simpana\IndexCache folder. The IndexCacheView tool in the Simpana\base folder can be used to view contents of index files.
Libraries

• Library Connections
  • Direct Attached
  • Network Attached
  • SAN Attached
• Disk Library
  • Deduplication Enabled
  • 3rd Party Deduplication Devices
  • Replicated Libraries
• Tape Library
  • Dedicated
  • Shared
  • VTL
  • IP Based
• Cloud Library
• USB PnP

Direct Attached Storage (DAS)
Direct Attached Storage (DAS) means the production storage location is directly attached (not SAN) to the production server. This provides for a simple management of data protection, particularly when using Simpana’s building block guidelines. When multiple MediaAgents are used, the failure of a single component may not necessarily affect all data protection or recovery jobs. The primary disadvantages are higher administrative overhead and depending on budget limitations, lower quality storage being used instead of high quality enterprise class disks (typically found in SAN/NAS storage).

For some applications such as Exchange 2010 using DAG (Database Availability Groups), Direct Attached Storage may be a valid solution. The main point is that although the storage trend over the past several years has been to storage consolidation, DAS storage should still be considered for certain production applications.

One key disadvantage regarding DAS protection is that backup operations will likely require data to be moved over a network. This problem can be reduced by using dedicated backup networks. Another disadvantage is that DAS is not as efficient as SAN or NAS when moving large amounts of data.

Network Attached Storage (NAS)
Network Attached Storage (NAS) has made a strong comeback over the past few years by taking advantage of its versatility. Where NAS was once only used as file stores they are now considered good options for databases and virtual machines. NAS versatility includes the ability to attach Fibre or iSCSI

No unauthorized use, copy or distribution.
connections along with traditional NAS NFS/CIFS shares and has a primary advantage of device intelligence using specifically designed operating systems to control and manage disks and disk access. From a high availability and disaster recovery aspect, disk cloning or mirroring and replication provide sound solutions. Simpana's IntelliSnap® integration with supported hardware provides simple yet powerful snapshot management capabilities.

One key disadvantage of NAS is that it typically requires network protocols when performing data protection operations. This disadvantage can be greatly reduced through the use of snapshots and proxy based backup operations.

**Storage Area Network (SAN)**

Storage Area Networks (SAN) are very commonly implemented for the most mission critical systems within an environment. The ability to consolidate storage using efficient data movement protocols, Fibre channel and iSCSI provide flexibility and performance.

One key disadvantage of SAN is the complexity of configuring and managing SAN networks. Typically, specialized training is required and all hardware must be fully compatible for proper operation. Since SAN storage lacks the operating system that NAS storage has, it relies on a host system for data movement. Depending on the configuration, the load of data movement can be offloaded to a proxy and by adding Host Bus Adapters (HBA) connected to a dedicated backup SAN data can be more efficiently backed up.
MediaAgent Placement and Scalability

- Preferred Path vs. Network
- Data Path Override

Data Path Override
Storage policies allow multiple data paths to be defined within each copy. When configured in the round-robin or failover mode all the subclients associated with the policy copy will use all the paths defined within the policy based on the policy configuration. The data path override option, configured in the subclient properties, can be used to determine specific paths within the policy that the subclient will use. This method of specifying data paths can be useful when trying to consolidate the number of storage policies being used.
Storage Policies
The concept of Storage Policy Copies is that the data from the production environment only has to be moved to protected storage once. Once the data is in protected storage, the storage policy logically manages and maintains independent copies of the data. This allows for great flexibility when managing data based on the three key aspects of data protection: data recovery, disaster recovery, and data archiving.

There are five types of storage policy copies

- Primary snap copy
- Primary backup or classic Copy
- Secondary snap copies (NetApp On Command Unified Manager only)
- Secondary Synchronous Copy
- Secondary Selective Copy

Primary Snap and Backup (Classic) Copy
A storage policy Primary Copy sets the primary rules for protected data. Each storage policy can have two primary copies, a Primary Snap copy and Primary Classic copy. A Primary Snap is used to manage protected data using the Simpana IntelliSnap® feature and a Primary Classic which will manage traditional agent based data protection jobs. Most rules defined during the policy creation process can be modified after it has been created. Settings that cannot be changed after the creation is whether it is a deduplicated enabled policy and whether the primary copy is associated with a global deduplication policy.
During data protection operations, performance is a major issue and meeting operation windows is becoming more difficult as data requiring protection continues to grow. Understanding how CommVault software works and optimally configuring the primary copy is crucial. Using high performance storage as the primary copy target is the best method to ensure windows are met.

**Secondary Copies**

Once data is protected to the primary copy location, additional copies can be created. The advantage of this architecture is that additional copies can be generated within the protected storage environment without impacting production resources. You can configure as many secondary copies as you need to manage stored data.

**There are two types of secondary copies:**

- Secondary Synchronous
- Secondary Selective

**Secondary Synchronous Copy**

A *Synchronous Copy* defines a secondary copy to synchronize protected data with a source copy. All valid data (jobs that completed successfully) written to the source copy will be copied to the synchronous copy via an update process called an Auxiliary Copy operation. This means all full, incremental, differential, transaction log, or archive job from a source copy will also be managed by the synchronous copy. Synchronous copies are useful when you want a consistent point-in-time copy for any point within the cycle of all protected data available for restore.

When a synchronous copy is defined an effective start date can be set to determine the starting point for data to be copied. This is configured in the *Copy Policy* tab of the secondary copy. The default is *All Backups* which means all jobs currently in the source copy will be copied to the synchronous copy when an auxiliary job runs. You can customize this option and select an effective start date for when the synchronous copy will become active.

**Synchronous copies are used to meet the following requirements:**

- Consistent point-in-time copies of data required to restore data to a particular point in time within a cycle.
- Copies that are required to be sent off-site daily.
- To provide the ability to restore multiple versions of an object from a secondary copy within a cycle.

**Secondary Selective Copy**

A *Selective Copy* allows automatic selection of specific full backups or manual selection of any backup for additional protection. Selective copy options allow the time based automatic selection of: all, weekly, monthly, quarterly, half-year, and/or yearly full backups. Advanced options allow you to generate selective copies based on a frequency of number of cycles, days, weeks, or months. You can also choose the *Do Not Automatically Select Jobs* option which allows you to use auxiliary copy schedules to determine when copies of full backups will be made.
Selective copies are used to meet the following requirements:

- Data being sent off-site weekly, monthly, quarterly, or yearly.
- Archiving point-in-time copies of data for compliance and government regulations.
Data Path Configuration allows you to specify how multiple data paths will be used. This provides a simple method to use multiple MediaAgents attached to a shared library as pooled resource for load-balancing and failover.

**There are several methods to configure and customize data paths:**

- Preferred path (LAN-Free)
- Default path (LAN based)
- Round-Robin load balancing
- Failover
- Data path override

**Default Data Path**

When creating a storage policy, a default path is defined during the policy creation wizard. If additional paths are added, the default path can be changed. Given no other guidance, all LAN based subclients associated with the storage policy will use the default path.

**Preferred Data Path**

While multiple paths are defined, the preferred path is determined by the CommVault software. A preferred path is one that uses a locally hosted (non-LAN) Media Agent. When a preferred path is available, it will always be used to move data to storage. In a preferred path configuration where the
preferred path becomes unavailable, alternate paths will not be used. You can accomplish LAN-Free backups by co-locating Media Agent software on Client enabled servers.

**Data Path Load Balancing (Round-Robin)**
In a small backup environment using one Media Agent may be adequate for all backup needs. As an environment grows multiple Media Agents may be needed to distribute or route data movement for better performance. You can define multiple paths within a storage policy copy and configure those paths to Round-Robin load balance.

**Data Path Failover**
In some cases you may also want to define another path if the default path is not available. Data path failover can be used to failover to another path if the default path is unavailable or if a resource in the path is offline or unavailable.
Storage Policy Data Path Configuration

Data Path Properties
One of the many powerful features of GridStor technology is the ability to independently configure data path properties within the storage policy copy. This allows custom settings at the logical level and not bound to the physical path. For example, financial data backing up to an LTO5 tape drive might require encryption. This can be enabled at the data path level of the storage policy managing the financial data. Other data managed by a different storage policy may not require encryption so it would be disabled in the LTO 5 data path for that policy copy. Even though the same physical drive is being used, the encryption will be turned on or off based on the data path configuration for a storage policy copy.

The following settings can be customized for a data path:

- Hardware compression
- Hardware encryption
- Chunk size
- Block size

Hardware Compression
If a tape drive supports hardware compression, it can be enabled in the data path properties. If no other compression has been performed on the data, enabling this option will make more efficient use of tape media. For data paths defined to write to tape libraries this option will be enabled by default. Some applications will perform compression on data as it is being backed up. If this is the case, compression should be disabled for the data path.
When CommVault’s data deduplication features is enabled, by default the software will compress data on the client before it is backed up to media. This will make data movement more efficient and deduplication ratios better. If the data is being copied from deduplicated disk to tape, CommVault recommends disabling compression for the tape data path. If using other third party deduplication, check with the vendor to see where compression is taking place and whether the data is decompressed by their hardware. Set the data path properties based on whether the data remains compressed or not.

**Hardware Encryption**

For tape drives that support hardware encryption, CommVault can manage configure settings and manage keys. Keys will be stored in the CommServe database. Keys can optionally be placed on the media to allow recovery of data if the CommServe database is not available at time of recovery. The data path option Via Media Password will put the keys on the media. The option No Access will only store the keys in the CommServe database. Note: If you choose the Via Media Password option it is absolutely essential that a Media Password be configured or the encrypted data can be recovered without entering any password during the recovery process. A global Media Password can be set in the System Settings in the Control Panel applet. Optionally a storage policy level password can be set in the Advanced tab of the Storage Policy Properties.

**Chunk Size**

Chunk sizes define the size of data chunks that are written to media. The default size for disk is 2GB. The default size for tape is 4GB for indexed based operations or 16GB for non-indexed database backups. The data path Chunk Size setting can override the default settings. A higher chunk size will result in a more efficient data movement process. In highly reliable networks, increasing chunk size can improve performance. However for unreliable networks, any failed chunks will have to be rewritten, so a larger chunk size could have a negative effect on performance.

**Block Size**

The default block size CommVault uses to move and write data to media is 64KB. This setting can be set from 32KB – 2048KB. Like chunk size, a higher block size can increase performance. However, block size is hardware dependent. Before modifying this setting ensure all hardware being used at your production and DR sites support the higher block size. If you are not sure, don’t change this value.

When writing to tape media, changing the block size will only become affective when CommVault rewrites the OML header on the tape. This is done when new media is added to the library, or existing media is recycled into a scratch pool. Media with existing jobs will continue to use the block size established by its OML setting.

When writing to disk, it is important to match the block size data path setting to the formatted block size of the disk. Matching block sizes can greatly improve disk performance. The default block sizes operating systems use to format disks is usually much smaller than the default setting in CommVault. It is strongly recommended to format disks to the block size being used in CommVault. Consult with your hardware vendors documentation and operating system settings to properly format disks.
Advanced Storage Policy Configurations

- Incremental storage policies
- Legal Hold storage policies
- Erase Data
- Storage Policy management
  - Hide Storage Policy
  - Copy Precedence
  - Deleting a Storage Policy

Incremental Storage Policy
An Incremental Storage Policy links two policies together. The main policy will manage all Full backup jobs. The incremental policy will manage all dependent jobs (incremental, differential or logs). This is useful when the primary target for full backups needs to be different than dependent jobs. Traditionally this has been used with database backups where the full backup would go to tape and log backups would go to disk. When performing log backups multiple times each day, replaying logs from disk during restore operations is considerably faster than replaying the logs from tape.

Legal Hold Storage Policy
When using the Simpana Content Indexing and compliance search feature, auditors can perform content searches on end user data. The search results can be incorporated into a legal hold. By designating a storage policy as a Legal Hold policy, the auditor will have the ability to associate selected items required for legal hold with designated legal hold policies. It is recommended to use dedicated legal hold policies when using this feature.

Erase Data
Erase data is a powerful tool that allows end users or Simpana administrators to granularly mark objects as unrecoverable within the CommCell environment. For object level archiving such as files and Email messages, if an end user deleted a stub, the corresponding object in CommVault protected storage can be marked as unrecoverable. Administrators can also browse or search for data through the CommCell Console and mark the data as unrecoverable.
It is technically not possible to erase specific data from within a job. The way Erase data works is by logically marking the data unrecoverable. If a browse or find operation is conducted the data will not appear. In order for this feature to be effective, any media managed by a storage policy with Erase Data enabled will not be able to be recovered through Media Explorer, Restore by Job, or Cataloged.

It is important to note that enabling or disabling this feature cannot be applied retroactively to media already written to. If this option is enabled then all media managed by the policy cannot be recovered other than through the CommCell Console. If it is not enabled then all data managed by the policy can be recovered through Media Explorer, Restore by Job, or Cataloged.

If this feature is going to be used it is recommended to use dedicated storage policies for all data that may require the Erase Data option to be applied. For data that is known to not require this option disable this feature.

**Hide Storage Policy**

If a storage policy managing protected data is deleted then all of the data associated with the policy will be aged and subsequently deleted. If a storage policy is no longer going to be used to protect data, the option *Hide Storage Policy* in the *General* tab of the policy properties can be selected. This will hide the policy in the storage policy tree and also hide the policy in the subclient drop down box in the *Storage Device* tab. In order to hide a storage policy no subclients can be associated with it.

If hidden storage policies need to be visible in the storage policy tree set the *Show hidden storage policies* parameter to 1 in the *Service Configuration* tab in the *Media Management* applet.

**Copy Precedence**

Copy precedence determines the order in which restore operation will be conducted. By default, the precedence order specified is based on the order in which the policy copies are created. The default order can be modified by selecting the copy and moving it down or up. This changes the default order. Precedence can also be specified when performing browse and recovery operations in the Advanced options of the browse or restore section. When using the browse or restore precedence the selected copy becomes explicit. This means that if the data is not found in the location the browse or restore operation will fail.

Any storage policy with a primary snap copy will by default, set the primary snap to copy precedence one. This will be independent of when the primary snap copy was created. Using a primary snap copy allows a ‘live browse’ operation to be conducted. A live browse will mount the snapshot and generate and index on the fly to allow browse and recovery of snapshot data.

**Deleting Storage Policies**

If a storage policy is deleted, all protected data associated with the storage policy and all policy copies will be pruned during the next data aging operation. It is strongly recommended to hide the storage policy instead of deleting it. A storage policy can only be deleted if no subclients are associated with the policy.
To delete Storage policy, perform the following:

1. In the Storage Policy properties view the Associations tab to ensure no subclients are associated with the policy. A Storage Policy cannot be deleted if subclients are associated with the policy.
2. On the Storage Policy, right click | select View | Jobs. De-select the option to Specify Time Range then click OK. This step will display all jobs managed by all copies of the Storage Policy. Ensure that there are no jobs being managed by the policy and then exit from the job history.
3. Right click on the Storage Policy | Select All Tasks | Delete. Read the warning dialog box then click OK. Type erase and reuse media then click OK.
Storage Policy Stream and Performance Settings

Stream Randomization
Stream randomization can improve performance during multi-streamed auxiliary copy operations by randomizing access to source disk mount paths.

Distribute data evenly for offline read operations
This option can be used when performing multi-streamed offline read operations from the MediaAgent. Enabling this option can improve content indexing jobs.

Keep resource reservations cached
For database servers conducting frequent log backup operations when the application agent and MediaAgent are installed on the same server, the keep resource reservations cached option can be used for fast resource allocation during backups. Resources reservations will be kept in the CommServe database and when the job is initiated the reservation data is immediately sent to the MediaAgent to allocate library and stream resources. This setting is not recommended for MediaAgents protecting multiple clients since the resources will remain locked for a specific job and not made available for other backup operations.
**Tape Library Device Streams**
For tape libraries one sequential write operation can be performed to each drive. If there are eight drives in the library then no more than eight device streams will be used. By default each job stream will write to a device stream. To allow multiple job streams to be written to a single tape drive, multiplexing can be enabled. The multiplexing factor will determine how many job streams can be written to a single device stream. If a multiplexing factor of four is set and there are eight drives a total of thirty two job streams can be written to eight device streams.

**Combine to Streams**
A storage policy can be configured to allow the use of multiple streams for primary copy backup. Multi-streaming of backup data is done to improve backup performance. Normally, each stream used for the primary copy requires a corresponding stream on each secondary copy. In the case of tape media for a secondary copy, multi-stream storage policies will consume multiple media. The combine to streams option can be used to consolidate multiple streams from source data on to fewer media when secondary copies are run. This allows for better media management and the grouping of like data onto media for storage.
Storage Policy Design Models

When planning storage policy design strategies, there are several key points to consider:

A technical design strategy will approach storage policy design based on:

- Location of clients in relation to MediaAgents.
- Differentiating different data types.
- Simplified managed of policies and data by grouping like data (application type, retention requirements, or location) into specific policies.

A business design strategy will approach storage policy design based on:

- Value of data, number of copies required, retention requirements and security requirements.
- Compliance requirements such as legal hold and content indexing

Deduplication configurations will change how storage policies are designed based on:

- Like data types that deduplicate well against each other.
- Global deduplication policy requirements for mixed retention and remote office consolidation.
- Using different block sizes for the deduplication database
- Using dedicated or partitioned deduplication databases.
Deduplication Primer

Deduplication can be configured for Storage Side Deduplication or Client (source) Side Deduplication. Depending on how deduplication is configured, the process will work as follows:

**Storage Side Deduplication**
Once the signature hash is generated on the block, the block and the hash are both sent to the Media Agent. The Media Agent with a local or remotely hosted deduplication database (DDB) will compare the hash within the database. If the hash does not exist that means the block is unique. The block will be written to disk storage and the hash will be logged in the database. If the hash already exists in the database that means the block already exists on disk. The block and hash will be discarded but the metadata of the data being protected will be written to the disk library.

**Client Side Deduplication**
Once the signature is generated on the block, only the hash will be sent to the Media Agent. The Media Agent with a local or remotely hosted deduplication database will compare the hash within the database. If the hash does not exist that means the block is unique. The Media Agent will request the block to be sent from the Client to the Media Agent which will then write the data to disk. If the hash already exists in the database that means the block already exists on disk. The Media Agent will inform the Client to discard the block and only metadata will be written to the disk library.
**Client Side Disk Cache**
An optional configuration for low bandwidth environments is the client side disk cache. This will maintain a local cache for deduplicated data. Each subclient will maintain its own cache. The signature is first compared in the local cache. If the hash exists the block is discarded. If the hash does not exist in the local cache, it is sent to the Media Agent. If the hash does not exist in the DDB, the Media Agent will request the block to be sent to the Media Agent. Both the local cache and the deduplication database will be updated with the new hash. If the block does exist the Media Agent will request the block to be discarded.

**Deduplication Block Size**
As application or file data is read into memory it is optionally compressed and then it will be hashed. This hash is compared in the deduplication database to determine if the block already exists. If the hash exists then the block is a duplicate and if not it is unique. It is important to understand how we address data blocks within files and applications to best configure deduplication.

**Content Aware Block Deduplication**
Simpana’s ability to be aware of the content that is being deduplicated allows blocks to be better aligned when deduplication takes place. When a file is read into memory it is compressed into 128KB blocks by default. A hash is generated on that compressed block which is used for deduplication. But not all files are 128 KB in size and not all files are evenly divided by 128 KB. If a compressed file is smaller than 128 KB it will be hashed in its entirety down to a minimum size of 4KB. For larger files that have a trailing segment that is smaller than 128 KB, that segment will also be hashed in its entirety down to 4 KB.

It is important to note that as each file is read into memory the 128 KB buffer is reset. Files will not be combined to meet the 128 KB buffer size requirement. This is a big advantage in achieving dedupe efficiency. Consider the same exact file on 10 different servers. If we always tried to fill the 128 KB buffer each machine would use different data and the hashes would always be different. By resetting the buffer with each file, each of the 10 machines would generate the same hash for the file.
Deduplication Building Blocks

CommVault recommends using building block guidelines for scalability in large environments. There are two layers to a building block, the physical layer and the logical layer.

For the physical layer, each building block will consist of one or more MediaAgents, one disk library and one deduplication database.

For the logical layer, each building block will contain one or more storage policies. If multiple storage policies are going to be used they should all be linked to a single global deduplication policy for the building block.

A building block using a deduplication block size of 128 KB can scale to retain up to 120 TB of deduplicated data. This could retain approximately 40 – 60 TB of production data with a retention of 30 – 90 days. The actual size of data will vary depending on the uniqueness of production data and the incremental block rate of change.

It is critical to provide adequate hardware to achieve maximum performance for a building block.

Performance starts with properly scaling the MediaAgent. There should be a minimum of 32 GB of RAM on each MediaAgent hosting the deduplication database.

The disk location of the deduplication database should be direct attached solid state disks or Fusion IO cards to the MediaAgent and must meet IOPs requirements. The disks can optionally be SAN Fibre attached using dedicated physical disks but should never be on NAS or iSCSI disks.
Building block guide Link:

Partitioned Deduplication Databases

Partitioned Deduplication Databases

- How it works
- Storage configuration
- Use cases
  - Resiliency
  - Scalability
  - Storage policy consolidation
- Where partitioned DDB fits and where it doesn’t

Partition deduplication is a highly scalable and resilient solution that allows the deduplication database to be partitioned. It works by dividing signatures between multiple databases to increase the capacity of a single building block. If two dedupe partitions are used, it effectively doubles the size of the deduplication store.

Since deduplicated data can exist on either of the partitions, the disk library should be configured using NAS storage. UNC paths should be used for the NAS disk library so either MediaAgent will be able to access data even if the other MediaAgent is unavailable.

How Partitioned Databases Work

During data protection jobs, partitioned deduplication databases and the data protection operation will work using the following logic:

1. **Signature is generated at the source** - For primary data protection jobs using client side deduplication, the source location will be the client. For auxiliary DASH copy jobs, the source MediaAgent will generate signatures.

2. **Based on the signature it will be sent to its respective database** – Which database the signature is sent to will be based on the first couple of digits of the signature. The respective database will compare the signature to determine if the block is duplicate or unique.

3. **The defined storage policy data path will be used to protect data** – Regardless of which database the signature is compared in, the data path will remain consistent throughout the job. If GridStor® Round-Robin has been enabled for the storage policy primary copy, jobs will load...
balance across any MediaAgents defined within the data path tab of the primary copy properties.

It is important to note that the data path used to protect data is independent of the database managing a block’s signature. If one MediaAgent is being used as the data path for a job and a signature is sent to a second MediaAgent, the signature record will be maintained in the database on the second MediaAgent while the deduplication block will be written to storage by the first MediaAgent. If partitioned deduplication is going to be implemented using two MediaAgents, it is strongly recommended to use a shared disk library using NAS storage as this will allow either MediaAgent to recover data even if the other MediaAgent is not available.

**Connection Requirements for MediaAgents**
It is required that both MediaAgents are connected with a 10 GigE direct connection. This connection is required for best performance during DASH full and data aging operations. To ensure the direct connection is used, configure a Data Interface Pair (DIP) between the two MediaAgents.

**Partitioned Database for Scalability**
The primary purpose for partitioned deduplication databases is to provide higher scalability of a single deduplication engine. By splitting signatures between two databases, a single deduplication engine can scale up to twice the size of a single database engine. This will provide more efficient deduplication ratios than if two dedicated engines were used since duplicate signatures could exist within each engine.

**Partitioned Database for Resiliency**
Another possible use case for partitioned databases is for resiliency. In the event that one MediaAgent hosting a deduplication database goes offline, the other MediaAgent would be able to continue data protection jobs and the available deduplication database would continue signature lookups. However, with the loss of one of the databases, all signatures previously managed by the offline database would now be looked up in the remaining online database. This would cause existing signatures managed in the offline database being compared in the online database resulting in the signatures being treated as unique, and additional data being written to the library.
Deduplication Database Components and Processes

The deduplication database currently can scale to approximately 120 Terabytes of data stored within the disk library. This roughly equates to about 40 – 60 TB of production data being retained for 30 – 90 days using a 128 KB deduplication block size. If a smaller block size of 64 KB is used, then approximately 20 - 30 TB of production data can be stored and if a larger block size of 256 KB is used then approximately 80 - 120 TB of data can be stored.

The deduplication block size can range from 32 KB – 512 KB. Through extensive testing, it has been determined that 128 KB block size provides the most efficient deduplication ratio, scalability and performance. Using a smaller block size may marginally improve deduplication ratios, it will limit how much deduplicated data can be stored and will lead to more block fragmentation in protected storage.

**Deduplication Database Backup Process**

When a deduplication enabled Storage Policy is created, a DDBbackup subclient is automatically created on the MediaAgent hosting the dedupe database. It will automatically configured to backup every eight hours.

When a DDB backup runs, the database will be placed in a quiesced state to ensure database consistency during the backup. For Windows MediaAgents, VSS will be enabled on the volume hosting the DDB. It is recommended that the Copy on Write Cache (COW) is configured to be at least 10% of the size of the volume hosting the DDB.
For Linux MediaAgents, Logical Volume Manager (LVM) will be used to create software snapshots of the DDB. It is recommended that the LVM volume have at least 15% of unallocated space for the snapshots.
Deduplication Data Movement Processes
Deduplication Enabled Data Protection

Deduplication processes during a data protection job:

2. CLBackup process uses CVD process to initiate communication with CVD process on MediaAgent.
3. CVD process on MediaAgent launches the SIDB2 process to access the deduplication database.
4. SIDB2 process communicates with CommServe to retrieve deduplication parameters.
5. CLBackup process begins processing by buffering data based on deduplication block factor and generates signatures on each deduplication block.
6. Signature is checked in deduplication database:
   a. If the signature exists, the primary record counter is increased. Secondary tables will update with detailed job information for the block. The block metadata is sent to the MediaAgent but the data block is discarded.
   b. If the signature does not exist, it is added to the primary table and detailed job information related to the block is added to the secondary table. Block data and metadata are sent to the MediaAgent.
DASH Full Process

DASH Full Process

- No block data is physically moved
- Uses latest image file to determine what to carry forward
- Used to carry forward items based on subclient retention settings
- Reads block signatures from chunks
- Updates deduplication database counters
- Creates new index files

A DASH Full backup is a read optimized synthetic full backup job. A traditional synthetic full backup is designed to synthesize a full backup by using data from prior backup jobs to generate a new full backup. This method will not move any data from the production server. Traditionally the synthetic full would read the data back to the Media Agent and then write the data to new locations on the disk library. With deduplication when the data is read to the Media Agent during a synthetic full, signatures will be generated and compared in the deduplication database. Being that the block was just read from the library, there would always being a signature match in the DDB and the data blocks would be discarded. To avoid the read operation all together a DASH Full can be used in place of a traditional synthetic full.

A DASH Full operation will simply update the index files and deduplication database to signify that a full backup has been performed. No data blocks are actually read from the disk library back to the Media Agent. Once the DASH Full is complete a new cycle will begin. The DASH Full is considered a valid full and any older cycles eligible for pruning can be deleted during the next data aging operation.
A DASH Copy is an optimized auxiliary copy operation which only transmits unique blocks from the source library to the destination library. It can be thought of as an intelligent replication which is ideal for consolidating data from remote sites to a central data center and backups to DR sites. It has several advantages over traditional replication methods:

- **DASH Copies** are auxiliary copy operations so they can be scheduled to run at optimal time periods when network bandwidth is readily available. Traditional replication would replicate data blocks as it arrives at the source.
- **Not all data on the source disk needs to be copied to the target disk.** Using the subclient associations of the secondary copy, only the data required to be copied would be selected. Traditional replication would require all data on the source to be replicated to the destination.
- **Different retention values can be set to each copy.** Traditional replication would use the same retention settings for both the source and target.
- **DASH Copy is more resilient in that if the source disk data becomes corrupt the target is still aware of all data blocks existing on the disk.** This means after the source disk is repopulated with data blocks, duplicate blocks will not be sent to the target, only changed blocks. Traditional replication would require the entire replication process to start over if the source data became corrupt.
DASH Copy is similar to Client Side Deduplication but with DASH, the source is a Media Agent and the destination is a Media Agent. This is why Client Side Deduplication and DASH Copy operations are sometimes referred to as **Source Side Deduplication**. Once the initial full auxiliary copy is performed, only change blocks will be transmitted from that point forward.

**DASH Copy has two additional options; Disk Read Optimized Copy, and Network Optimized Copy.**

**Network Optimized** – source Media Agent generates a signature and query destination Media Agent DDB. If signature exists, the signature and any metadata will be sent to destination Media Agent. If unique the signature is sent to destination Media Agent and CVD will transmit data and metadata to destination Media Agent. Once block is written CVD will commit signature record to DDB.

**Disk Optimized** – Source Media Agent will read signatures from chunk metadata and send the signature to the destination Media Agent. If the signature exists CVD will write only metadata to destination Media Agent. If the signature is unique a new record is inserted in destination Media Agent and CVD will send block and metadata to the destination Media Agent. Once the block is written to disk, CVD will commit the record in the DDB.

**UseCacheDB** – This is an optional registry key which will create a local signature cache (similar to client side cache). Signatures will first be checked in the local signature cache before sending signature to destination Media Agent.

**Notes on using Network and Disk optimized Dash Copy**
- Disk optimized provides best performance.
- Network optimized provides better data integrity since the chunk is being read from source disk and data verification is being performed.
- Network optimized can work on deduplicated and non-deduplicated sources.
- Disk optimized required the source data to be deduplicated.

**Seeding Deduplicated Disk Libraries**
For low bandwidth networks, seeding a disk library can be performed to greatly reduce the data required to be sent over the network. This is done by temporarily placing a disk library at the source location. This library can be an external USB drive or regular disk storage. The data can be copied to the temporary disk library and then relocated to the destination location. These procedures require several detailed steps and it is recommended to consult with CommVault Professional Services for assistance. Seeding disk libraries can be used for Client Side Deduplication, DASH Full and DASH Copy operations.
Aging Deduplicated Data
Deduplication Data Aging Primer

- Micro and Macro Pruning
- Store Pruning
- Truncation Pruning (Linux)
- Logical Aging and Physical Pruning
- Pruning data from a sealed store
  - DDB available
  - DDB not available

Data aging is a logical operation that compares what is in protected storage against defined retention settings. Jobs that have exceeded retention are logically marked as aged. During normal data aging operations all chunks related to an aged job are marked as aged. With Simpana® deduplication data blocks within chunks can be referenced by multiple jobs. If the entire chunk was aged then jobs referencing blocks within the chunk would not be recoverable. The Simpana software uses a different mechanism when performing data aging operations for deduplicated storage.

The pruning process, which physically deletes data from deduplicated disk storage, works by checking with the deduplication database to determine if the block is being referenced by any jobs. If the block is being referenced then it will be maintained in storage. If the block is not referenced then the block will be pruned from the disk. This means that when using Simpana deduplication, data is not deleted from disk at the job level, instead data is pruned at the chunk or block level.

To prune chunks or blocks from storage, a counter system is used in the deduplication database to determine the number of times a deduplication block is being referenced. Each time a duplicate block is written to disk during a data protection job, a reference counter in the deduplication database is incremented. When the data aging operation runs, each time a deduplication block is no longer being referenced by an aged job, the counter is decremented. When the counter for the block reaches zero, it indicates that no jobs are referencing the block. At this point the block can be physically deleted from the disk library.
The aging and pruning process for deduplicated data is made up of several steps. When the data aging operation runs, it will appear in the job controller and may run for several minutes. This aging process logically marks data as aged. Behind the scenes on the MediaAgent, the pruning process will run, which can take considerably more time depending on the performance characteristics of the MediaAgent and deduplication database, as well as how many records need to be deleted.

**Aging and pruning process steps:**

1. Jobs are logically aged which will result in job metadata stored in the CommServe® database as archive files being moved into the MMDeleteAF table. This will occur based on one of two conditions:
   a. Data aging operation runs and jobs which have exceeded retention are logically aged.
   b. Jobs are manually deleted which will logically mark the job as aged.
2. Job metadata is sent to the MediaAgent to start the pruning process.
3. Metadata chunks will be pruned from disk. Metadata chunks contain metadata associated with each job so once the job is aged the metadata is no longer needed.
4. Signature references in the primary and secondary tables will be adjusted based on:
   a. Primary table – records for each signature will be decremented for each occurrence of the block.
   b. Secondary table – records for each signature related to the job will be deleted from the secondary table files.
5. Signatures no longer referenced will be moved into the zero reference table.
6. Signatures for blocks no longer being referenced will be updated in the chunk metadata information. Blocks will then be deleted using the drill holes, truncation or chunk file deletion method.
Data Aging Log Files

<table>
<thead>
<tr>
<th>Log File Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MediaManager.log</td>
<td>Shows prune request sent to MediaAgent</td>
</tr>
<tr>
<td>MediaManagerPrune.log</td>
<td>Shows MMDeletedAF entries being sent to MediaAgent</td>
</tr>
<tr>
<td>SIDBPrune.log</td>
<td>Show logical deduplication database pruning</td>
</tr>
<tr>
<td>SIDBPhysicalDeletes.log</td>
<td>Shows information on physical deletion processes</td>
</tr>
<tr>
<td>SIDBEngine.log</td>
<td>Show SIDB2 engine statistics</td>
</tr>
</tbody>
</table>
Phase 1: Metadata Deletes

<table>
<thead>
<tr>
<th>Log File</th>
<th>Log Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>MediaManager.log</td>
<td>3188 fdc 10/21 16:04:59 —— SERVICE [ ] PRUNE SIDB DATA for SIDB [57] on host [7, SVR10VMAN2general.local], RID [15481/0/0]</td>
</tr>
<tr>
<td>SIDBPrune.log</td>
<td>3196 73c 10/21 16:05:00 ### SIDBPruneRequest:3311 PHASE 1: PruneInfoList items [2]</td>
</tr>
<tr>
<td></td>
<td>3196 73c 10/21 16:05:02 ### SIDBPruneRequest:3409 Chunk [28772]</td>
</tr>
<tr>
<td></td>
<td>3196 73c 10/21 16:05:02 ### PruneChunk:602 Removed</td>
</tr>
<tr>
<td></td>
<td>[G:/m3/m33/CV_MAGNETIC/V_515279/CHUNK_28772/CHUNK_META_DATA_28772]</td>
</tr>
<tr>
<td></td>
<td>3196 73c 10/21 16:05:02 ### PruneChunk:602 Removed</td>
</tr>
<tr>
<td></td>
<td>[G:/m3/m33/CV_MAGNETIC/V_515279/CHUNK_28772/CHUNK_META_DATA_28772.idx]</td>
</tr>
<tr>
<td>SIDBPhysicalDeletes.log</td>
<td>1852 10/21 16:05:02 Deletion MOUNTPATHID=10 V_515279(chunk_28772 chunk-metadata-file</td>
</tr>
<tr>
<td></td>
<td>1852 10/21 16:05:02 Deletion MOUNTPATHID=10 V_515279(chunk_28772 chunk-metadata-index-file</td>
</tr>
<tr>
<td>MediaManagerPrune.log</td>
<td>3188 ff4 10/21 16:05:10 —— MNTPATH [ ] SIDB Prune Response: AF = 42233, Volume = 515279, CHUNK 28772, subStoreBitField 0, sidbPruningFlag 1, ErrorCode = 0 IsReconPruning[false]</td>
</tr>
<tr>
<td></td>
<td>3188 ff4 10/21 16:05:10 —— MNTPATH [ ] Removed AF[42233] Vol[515279] Chunk[28772] from mmdeletedaf...</td>
</tr>
</tbody>
</table>
Phase 2: Decrementing DDB Counters

- MMDeleteAF entries are sent to SIDB2 for logical pruning.
- DDB determines if references are tied to block.
  - If exists, decrements counter for each occurrence of block signature.
  - If does not exist block signature sent to Zero Ref table for physical pruning.

### Log File Entries

<table>
<thead>
<tr>
<th>Log File</th>
<th>Log Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3196 cac 10/21 16:10:51 ### SubstoreRoutingInfo: StoreId [57] (GroupNum-SubStoreId-SplitNum) 0-65-0 routed to 0-65-0 Client=[svr10v-ma2] Path=[F:\MA2_Windows_GDDB_2]</td>
</tr>
<tr>
<td></td>
<td>3196 cac 10/21 16:10:51 ### Connect: Connecting to Engine [57], Group [0], Client [svr10v-ma2], Host [SVR10V-MA2.general.local<em>svr10v-ma2</em>8400*8402], Recovery Mode [false]</td>
</tr>
<tr>
<td></td>
<td>3196 cac 10/21 16:10:57 ### Connected to Engine [57], Group [0]. Socket [1056]. Remote Process [SIDB2.exe:1424:3024], Address Family [IPv4], Local IP [127.0.0.1], Local Port [52067], Peer IP [127.0.0.1], Peer Port [61003]</td>
</tr>
<tr>
<td></td>
<td>3196 cac 10/21 16:10:57 ### SIDBPruneRequest:3566 Pruned AfId [42233] completely. Split summary [00000001]</td>
</tr>
</tbody>
</table>
Phase 3: Physical Deletion

Phase 3: Physical Deletion

<table>
<thead>
<tr>
<th>Log File</th>
<th>Log Entries</th>
<th>Date/Time</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIDBEngine.log</td>
<td>1424 874</td>
<td>10/21 16:10:53</td>
<td>### 57-0-65-0 LogCtrs:6314 [0][0]-[0]-[0]-[0].</td>
</tr>
<tr>
<td>SIDBPrune.log</td>
<td>3196 cac</td>
<td>10/21 16:10:57</td>
<td>PHASE 3: Pruning unreferenced primary records</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3196 cac</td>
<td>Open:1506 Initialized pruner object. Path</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3196 cac</td>
<td>[G:\m3\CV_MAGNETIC\V_515279\CHUNK_28772\SFILE_CONTAINER.idx], Drill Holes [true], Min Hole Size [131072], Enable Counters [false]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3196 cac</td>
<td>Finalize:2032 Finalizing SI entries in chunk [28772].</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3196 cac</td>
<td>Finalize:2522 Removed index file</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3196 cac</td>
<td>Finalize:2234 Going to remove the idx file</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3196 cac</td>
<td>Finalize:2252 Removed index file</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3196 cac</td>
<td>Finalize:234 Going to remove the idx file</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3196 cac</td>
<td>Finalize:2414 Removed vol</td>
</tr>
</tbody>
</table>
Interpreting the SIDB Engine Physical Delete Log Entries

<table>
<thead>
<tr>
<th>Primary [47817] - [7374636] - [0] - [0] - [0] - [0]</th>
</tr>
</thead>
<tbody>
<tr>
<td>The number of records</td>
</tr>
<tr>
<td>The size of those records</td>
</tr>
<tr>
<td>Number and size of blocks added since last time entry was made</td>
</tr>
<tr>
<td>Number and size of blocks removed since last time entry was made</td>
</tr>
</tbody>
</table>

- **Primary** - amount of unique blocks stored in the DDB.
- **Secondary** - amount of references stored in the DDB.
- **Pending Deletes** - amount in the Zero Ref DDB table.
- **Lonely** - Primary records with only one Secondary reference (newer blocks).
- **Uncommitted** - blocks had issues during the backup and have not been committed to the database.
- **Bad** - blocks are corrupt blocks.
Deduplication Design Strategies
Deduplication Storage Policies

- **Block Size**
  - Use 128 KB
  - For large datasets consider increasing block size
    - Large databases or static repositories e.g. A/V files
- **Store Configuration**
  - Use defaults
  - Configure properly to avoid sealing store
- **Compression**
  - Use compression for file and VM data
  - For database use either application or Simpana compression but never both
- **Global Deduplication**

Deduplication is centrally managed through storage policies. Each policy can maintain its own deduplication settings or can be linked to a global deduplication storage policy. Which method is used for configuring storage policies will depend on the type of data and your environment. This section will explain the elements of a Deduplication storage policy and when dedicated policies should be used and when global policies should be used.

**Dedicated Deduplication Storage Policy**

A dedicated deduplication storage policy will consist of one library, one deduplication database, and one or more Media Agents. For scalability purposes, using a dedicated deduplication policy allows for the efficient movement of very large amounts of data. Dedicated policies are also recommended to separate data types that do not deduplicate well against each other such as database and file system data.

**Global Deduplication Storage Policy**

*Global Deduplication* storage policies work by linking storage policy copies to a single deduplication database and store. This allows data to be managed independently by a specific storage policy while maintaining a more efficient deduplication ratio. Each policy can manage specific content and independently manage retention and additional copies. This provides for efficient deduplication ratios while providing scalability and flexibility for different data protection requirements.
Global Deduplication for Base Storage Policy Design
If you are planning a new storage policy architecture and you are unsure of how many policies will be needed, using a global deduplication policy as your base store could provide better deduplication ratios as your environment changes and grows. Even if one storage policy initially will be used, consider linking the primary copy to a global deduplication policy. This is best used when protecting object data or virtual machines. This use of global dedupe policy would not apply to databases even if the same DB application is being used as deduplication efficiency will not be realized and the result would just be a bigger deduplication database.

It is important to note that associating or not associating a storage policy copy with a global deduplication policy can only be done at the creation of the policy copy. Once the copy is created it will either be part of a global policy or it won’t. By using the global dedupe policy for the initial storage policy primary copy that will protect data, if additional policies are required, they can also be linked to the global dedupe policy. Using this method will result in better deduplication ratios and provide more flexibility for defining retention policies or consolidating remote location data to a central policy (which will be discussed next). The main caveat when using this method is to ensure that your deduplication infrastructure will be able to scale as your protection needs grow.

Global Deduplication for consolidating multiple remote sites
Global Deduplication storage policies were designed specifically to address remote site backups where backups were being performed locally at each site. Then using DASH Copy operations, the data is copied to a central data center location. Since duplicate blocks may exist at each of the sites, using a global deduplication storage policy associated with a secondary copy will use a single deduplication database and a single store to consolidate data blocks from all remote locations.

Global Deduplication for small data size with different retention needs
For small environments that do not contain a large amount of data but different retention settings are required, multiple storage policy Primary Copies can be associated with a global deduplication storage policy. This should be used for small environments with the data path defined through a single MediaAgent.
Standard Building Block Deployment

- MediaAgent, DDB and dedupe store
- DDB on solid state disks – must meet IOPS requirements
- Two DDBs maximum for a building block
  - Only one should be active at any time
- Estimate current size and future growth

[Diagram showing standard building block deployment with details on storage policies and data retention]

No unauthorized use, copy or distribution.
Dedicated MediaAgent for DDB

- Not a standard use case
- Clustered Applications
- UNIX Clients

Using a dedicated MediaAgent to host the deduplication database is not a very common design strategy. This specific example is used to point out that in certain situations it may be necessary to deviate from the standard building block recommendations.
SILO Storage

- SILO is NOT a DR solution
- For long term data preservation
- Copies closed folders to SILO copy
- May require periodic store sealing to control DDB growth

SILO storage allows deduplicated data to be copied to tape without rehydrating the data. This means the same deduplication ratio that is achieved on disk can also be achieved to tape. As data on disk storage gets older the data can be pruned to make space available for new data. This allows disk retention to be extended out for very long periods of time by moving older data to tape.

How SILO works
Data blocks are written to volume folders in disk storage. These folders make up the deduplication store. The folders have a maximum size which once reached the folder is marked closed. New folders will then be created for new blocks being written. The default volume folder size for a SILO enabled copy is 512 MB. This value can be set in the Control Panel, in the Media Management Applet. The SILO Archive Configuration setting Approximate Dedup disk volume size in MB for SILO enabled copy is used to specify the volume folder size. It is strongly recommended to use the default 512 MB value. For a SILO enabled storage policy, when the folder is marked full it can then be copied to tape. What this really is doing is backing up the backup.

How volume folders are moved to SILO Storage
When a storage policy is enabled for SILO storage an On Demand Backup Set is created in the File System Agent on the CommServe server. The On Demand Backup Set will determine which volume folders have been marked full and back them up to tape each time a SILO operation runs. Within the backup set a Default Subclient is used to schedule the SILO operations to run. Just like an ordinary data protection operation, right click the subclient and select Backup. The SILO backup will always be a full
backup operation and use the On Demand Backup to determine which folders will be copied to SILO storage.

**Silo Recovery Process**

1. The CommVault administrator performs a browse operation to restore a folder from eight months ago.
2. If the volume folders are still on disk the recovery operation will proceed normally.
3. If the volume folders are not on disk the recovery operation will go into a waiting state.
4. A SILO recovery operation will start and all volume folders required for the restore will be staged back to the disk library.
5. Once all volume folders have been staged, the recovery operation will run.
6. To ensure adequate space for SILO staging operations a disk library mount path can optionally be dedicated to SILO restore operations. To do this, in the Mount Path Properties **General** tab select the option **Reserve space for SILO restores**.
7. The procedure is straightforward and as long as SILO tapes are available the recovery operation is fully automated and requires no special intervention by the CommVault administrator.
Module 4: Virtualization
Virtual Data Protection Methods

- Virtual Server Agent (VSA)
- Agents within virtual machines
  - Databases
    - Multi-streaming
    - Log backup
  - Granular backup / restore
  - Content indexing
- IntelliSnap® Technology
  - Using VSA or application agent
  - High I/O VMs
  - Application consistent
- Use scripts to protect application data

There are three primary methods Simpana software can use to protect virtual environments:

- Virtual Server Agent (VSA)
- Agents installed within virtual machines
- IntelliSnap® Technology

Which method is best to use depends on the virtual infrastructure, type of virtual machines being protected and the data contained within the virtual machines. In most cases using the Virtual Server Agent will be the preferred protection method. For specific virtual machines using an agent inside the VMs will be the preferred method. For mission critical virtual machines, large virtual machines or virtual machines with high I/O processes, IntelliSnap feature can be used to coordinate hypervisor software snapshots with array hardware snapshots to protect virtual machines.
VSA Data Protection Process

VSA works by communicating with the hosting hypervisor to initiate software snapshots of virtual machines. Once the VMs are snapped, VSA will back them up to protected storage.

The following steps illustrate the process of backing up VMware virtual machines:

1. Virtual Server Agent communicates with the hypervisor instance to locate virtual machines defined in the subclient that require protection.
2. Once the virtual machines are located the hypervisor will prepare the virtual machine for the snapshot process.
3. The virtual machine will be placed in a quiescent state. For Windows VMs, VSS will be engaged to quiesce disks.
4. The hypervisor will then conduct a software snapshot of the virtual machine.
5. The virtual machine metadata will be extracted.
6. The backup process will then back up all virtual disk files.
7. Once the disks are backed up, indexes will be generated for granular recovery (if enabled).
8. The hypervisor will then delete the snapshots.
VMware Transport Modes

The VMware VADP framework provides three transport modes to protect virtual machines:

- SAN transport mode
- HotAdd mode
- NBD and NBD SSL mode

Each of these modes has their advantages and disadvantages. Variables such as physical architecture, source data location, ESXi resources, network resources and VSA proximity to MediaAgents and storage will all have an effect on determining which mode is best to use. It is also recommended to consult with CommVault for design guidance when deploying Simpana software in a VMware environment.

SAN Transport Mode
SAN Transport Mode can be used on a VSA proxy with direct Fibre channel or iSCSI access to snapshot VMs in the source storage location. This mode provides the advantage of avoiding network movement of VM data and eliminates load on production ESXi servers. Virtual machines can be backed up through the VSA and to the MediaAgent. If the VSA is installed on a proxy server configured as a MediaAgent with direct access to storage, LAN-Free backups can be performed.

HotAdd Mode
HotAdd mode uses a virtual VSA in the VMware environment. This will require all data to be processed and moved through the VSA proxy on the ESXi server. HotAdd mode has the advantage of not requiring...
a physical VSA proxy and does not require direct SAN access to storage. It works by ‘hot adding’ virtual disks to the VSA proxy and backing up the disks and configuration files to protected storage.

A common method of using HotAdd mode is to use Simpana deduplication with client side deduplication, DASH Full and incremental forever protection strategy. Using Change Block Tracking (CBT) only changed blocks within the virtual disk will have signatures generated and only unique block data will be protected.

**NBD Mode**

NBD mode will use a VSA proxy installed on a physical host. VSA will connect to VMware and snapshots will be moved from the VMware environment over the network and to the VSA proxy. This method will require adequate network resources and it is recommended to use a dedicated backup network when using the NBD mode.
Raw Device Mapping (RDM)

Raw Device Mapping is a mapping file that acts as a proxy for raw disk storage allowing a virtual machine to transparently access raw disk storage. The RDM, which will have a .vmdk extension contains metadata for managing and redirecting disk access to the physical device.

RDM Physical compatibility mode:
- Low level direct access to SCSI devices.
- VSA agent cannot backup RDM devices in physical compatibility mode. Agents must be installed within the VM to access and protect the data.
- Data on RDM storage can be protected using the IntelliSnap feature for supported hardware arrays.
- For large volumes greater than 2 TB using RDM volumes can provide a performance advantage over using regular .vmdk files.

RDM Virtual compatibility mode:
- Acts like a virtual disk file which allows virtual disk snapshots to be conducted and protected using VSA.
- Only passes read / write operations to the RDM device
- Appears to guest OS the same as vmdk disks – hardware characteristics are masked from OS.

When the VSA agent protects VMware virtual machines it will backup all data in VMDK files and virtual RDM volumes. It will not protect any data on volumes using physical RDM. For data that is located on physical RDM volumes it is recommended to either convert the volume to a standard VMDK file or install agents in the VM to protect the data.
In certain cases physical RDM volumes can be used as an advantage when designing solutions for protecting large databases. A VSA agent will be used to snap and backup the virtual disks as VMDK files but physical RDM volumes can be filtered from the backup. An application agent can then be installed in the VM and subclients can be configured to protect data on RDM volumes. The application agent will provide communication to provide consistent point-in-time backups of application data. If the RDM volume is on a dedicated LUN, the Simpana IntelliSnap feature can be used to conduct hardware snapshots of the volume for point-in-time restores and for mounting the volume for proxy backup.
VSA Design and Configuration
VSA Design Considerations

VSA Design Considerations

- General Considerations
  - Transport mode
  - Proxy resource allocation
- Protecting Application in Virtual Machines
  - Ensuring Application Consistency
    - Agent installed in machine
    - VSA and IntelliSnap
    - Freeze / Thaw scripts
- IntelliSnap® Technology
  - High I/O VMs
  - Live browse
  - Revert operations (NetApp)
VSA Agent Configuration

- Instance
  - Defining Proxies
- Subclient
  - Transport Modes
  - Defining Content
  - Filtering
    - Backup Set Level (SP7)
    - Disk level use cases
      - System drive vs. data drive
    - Datastore
      - VMDK split across datastores
    - Quiesce options

Definitions:
- Instance properties and VSA proxies for the instance
- Defines VMs to be protected, transport mode, proxy override, filtering, DataStore check and Quiesce options
- Defines collection group of all VMs for instance
VSA Client Overview

VSA – Client Overview

- VSA client added from Client Computer level
- VMware
- Hyper-V
- Upgrade considerations

VSA instances are created by right clicking on Client Computers, selecting New Client, Virtualization and then selecting either VMware or Hyper-V. This will create a new VSA client at the root level of Client Computers.
VSA Client and Instance Configuration

1. Enter vCenter Host Name and user access credentials.

2. Add VSA Proxies and set proxy priority. VSA proxies can be added based on client (not bold) and / or client computer group (bold).

Instance Property Configuration

The instance properties are configured with login credentials to access the hypervisor management system. VSA proxies can be assigned to the instance using the Proxies tab. Client or client computer groups can be added.
VSA Proxies
Simpana software uses VSA proxies to facilitate the movement of virtual machine data during backup and recovery operations. The VSA proxies are identified in the instance properties. For Microsoft Hyper-V, each VSA proxy will be designated to protect virtual machines hosted on the physical Hyper-V server. For VMware, the VSA proxies will be used as a pooled resource. This means that depending on resource availability different proxies may be used to backup VSA subclients each time a job runs. This method of backing up virtual machines provides for higher scalability and resiliency.
VSA Subclient Content Configuration

Configuring VSA Subclient
To configure a VSA subclient, right click on the subclient and select Properties.

To add a VSA subclient, right click on the backup set | All Tasks | New Subclient.

Subclients are configured to define specific VM content that will be protected and define specific methods for how each VM within the subclient will be protected.

In addition to standard subclient settings, the VSA subclient provide the following configuration settings:

- VM content and filters (for VMware - filter at VM and disk levels).
- VMware specific settings:
  - Transport mode
  - VSS quiesce options
  - DataStore free space check
**Default Subclient**

The default subclient content tab contains a backslash entry, similar to Windows File System agents to signify the subclient as a catch all. Any VMs not protected in other subclients will automatically be protected by the default subclient. It is recommended that the contents is not changed, activity is not disabled and the default subclient is regularly scheduled to back up, even if there are no VMs in the subclient. To avoid protecting VMs that do not need to be backed up, use the backup set level filters and add all VMs that don’t require protection.

![Default Subclient](image)

*Default Subclient will protect all discovered virtual machines not defined in a user defined subclient. It is STRONGLY recommended to regularly schedule the default subclient to back up even if no content exists.*

**Transport Modes (VMware)**

The VMware transport mode is configured in the General tab of the subclient. The default setting is Auto which will attempt to use SAN or HotAdd mode and fall back to NBD mode if other modes are not available. To configure a specific transport mode with no fall back, select the desired mode from the drop down box.
VM Content tab

VSA subclient contents can be defined using the Browse or Add buttons. Browse provides a vCenter like tree structure where resources can be selected at different levels including Cluster or DataStore. For most environments, it is recommended to select subclient contents at the cluster level. For smaller environments, or for optimal performance, defining subclient contents at the DataStore level can be used to distribute backup load across multiple DataStores.

The Add option can be used to define rules for VM content definition. Multiple rules can be nested such as all Windows VMs in a specific DataStore.
Content Best Practices

- When browsing for content, as a best practice, select content at the cluster or DataStore level.
- Ensure VSA proxies can access VMs defined within the subclient content.

VSA Subclient Settings

Data Readers
The data readers setting in the advanced tab of the subclient properties is used to determine the number of simultaneous virtual machine backups that will be conducted. The default setting is two
which means two VMs will be quiesced, snapped and backed up for the subclient through the VSA at any given time. The data readers option can be increased to provide better concurrency of VM backups. Increasing this setting could have a negative effect on backup performance if the DataStore holding the VMs cannot handle the additional load. It is recommended to only increase this setting if backup windows are not being met.
Subclient Proxies
Proxies are defined in the VSA instance but can be overridden at the subclient level. This is useful when specific subclient VM contents are not accessible from all VSA proxies. Proxies can be added, removed, and moved up or down to set proxy priority.

Subclient Filters
Subclient filters can be used to filter virtual machines for both Hyper-V and VMware. VSA for VMware also provides filtering capabilities at the disk level.
Virtual machine filtering

Virtual machines can be filtered by browsing for VMs or adding specific criteria for VM filtering. This can be useful when content is being defined at a parent level but specific virtual machines are to be excluded from backup.

Virtual disk filtering (VMware)

For VMware, disk level filtering can also be applied. This provides the ability to filter disks based on host, DataStore, VMDK name pattern or hard disk number. This can be useful when certain disks do not require protection or if Simpana agents installed within the VM will be used to protect data.

Example: A database server requires protection. For shorter recovery points and more granular backup and recovery functionality, a database agent will be used to protect application database and log files. For system drives, the virtual server agent will be used for quick backup and recovery. Disks containing the database and logs will be filtered from the subclient. The VSA will protect system drives and the application database agent will be used to protect database daily and log files every 15 minutes. This solution provides shorter recovery points by conducting frequent log backups, application aware backup and restores, and protects system drives using the virtual server agent.

Backup Options (VMware)

There are several subclient options that are specific to the VMware VSA subclient.

- Quiesce guest file system and applications – Configured in the Quiesce Options tab, this is used to enable (default) or disable the use of VSS to quiesce disks and VSS aware application for Windows virtual machines.
- Application aware backup for item based recovery – Configured in the Quiesce Options tab, this is available only when using the IntelliSnap feature and is used to conduct application aware snapshots of virtualized Microsoft SQL and Exchange servers.
- Perform DataStore free space check – Configured in the Quiesce Options tab, this sets a minimum free space (default 10%) for the DataStore to ensure there is enough free space to conduct and manage software snapshots during the VM data protection process.
Module 5: IntelliSnap® Technology
Snapshot Technology

- Snapshot methods
  - Copy on write
  - Allocate on write
  - Mirroring
- NetApp
  - Snap Mirror
  - Snap Vault
Snapshots are point in time logical views of a volume. The volume block mapping is snapped which represents a point-in-time view of the block structure when the snap occurred. When existing blocks need to be overwritten with new blocks the old blocks are preserved. References to these blocks are recorded to provide a frozen point-in-time snapshot view of the volume. This allows the volume to be reverted back to any point in which a snapshot was taken. The snapshot can also be mounted off line on a separate host for mining, testing, backing up or restoring data. Although vendors may use their own specific snap methods and different terminology, there are two primary methods for conducting snapshots:

- Copy on Write
- Allocated on Write (Write Optimized)

**Copy on Write**

The copy on write method uses snapshots to gather reference markers for blocks on the snapped volume. A copy on write cache will be created that will cache the original blocks when the blocks need to be overwritten. This requires a read-write-write operation to complete. When a block update of a snapped volume is required, the original block is read from the source volume. Next the original block is written to the cache location. Once the original block has been cached, the new block is committed to the production volume overwriting the original block. This method has the advantage of keeping production blocks contingent in the volume which provides faster read access. The disadvantage is the read-write-write processes increases I/O load on the disks.
Allocate on Write (Write Optimized)
Allocate on write uses additional space on a volume to write update blocks when the original block is modified. In this case the original block will remain in place and the new block is written to another section of the volume. Markers will be used to reference the new block for read requests of the production data. This has an advantage over copy on write in that there is only a single write operation decreasing I/O load on the disks. The disadvantage is that over time higher fragmentation may exist on the volume.
Application and Crash Consistency

With Application Consistent protection, the application itself is aware that it is being snapped. This awareness allows for the data to be protected and restored in a consistent and usable state. Application aware protection works by communicating with the application to quiesce data or by using scripts to properly quiesce the data. Application consistent protection is not critical for file data but is absolutely critical for application databases.

There are several methods to provide application consistent protection:

- Simpana application agents – An agent installed in the VM will directly communicate with application running in the VM. Prior to the snap operation the agent will communicate with the application to properly quiesce databases. For large databases this is the preferred method for providing application consistent point in time snap and backup operations. Using application agents in the VM also provide database and log backup operations and a simplified restore method using the standard browse and recovery options in the CommCell GUI.

- Scripting database shutdowns – Using external scripts which can be inserted in the Pre/Post processes of a subclient, application data can be placed in an offline state to allow for a consistent point-in-time snap and backup operation. This will require the application to remain in the offline state for the entire time of the snapshot operation. When the VM is recovered the application will have to be restarted after the restore operation completes. This method is only recommended when Simpana agents are not available for the application.

- IntelliSnap for VSA – For Microsoft SQL and Exchange virtual machines, application aware protection can be performed using the VSA agent and Simpana IntelliSnap feature.
• Application Consistent backup performs a snapshot and backup of the application data at a specified point in time. The application is aware that this is being performed and will quiesce data.

**Crash Consistent**

Crash Consistent backups are based on point-in-time snapshot and backup operations of a virtual machine that allows the VM to be restored to the point in which it was snapped. When the snapshot occurs all blocks on the virtual disks are frozen for a consistent point-in-time view.

There are several issues when performing crash consistent snapshot and backup operations. The first issue is that if an application is running on the virtual machine it is not aware the snapshot is being taken. VSA communicates with the hosting hypervisor to initiate snapshots at the VM level and there is no communication with the application. Any I/O processes being conducted by the application will continue without any knowledge that the snap has been performed. This may cause issues if a VM hosting an application has high disk I/O activity at the time the snap occurred.

The other issue is data integrity. Crash consistent means when a snap occurs, a logical view of the virtual disk block structure is preserved for the backup operation. The crash consistent view would be the same as if you turned the power off on an application server without properly shutting down the application. In this case, maintenance may need to be performed on the application databases before they would be usable and there is the possibility of data corruption. Crash consistent backups can work well for disk volumes containing file data but this is not recommended for protecting application databases.
IntelliSnap® Technology Processes for VSA Part 1

- JobManager initiates job

1. CVD takes job request from JobManager and launches vsbkp to coordinate both the software and hardware snapshots

2. Query vCenter for datastore information (LUN information)

VSBKP

3. CVD

4. VSBKP

1. CVD

No unauthorized use, copy or distribution.
CVD
VSBKP
VSBKP communicates to vCenter to conduct software snapshots through ESXi server of all virtual machines listed in the subclient contents.

5
7
CVMOUNT
CVMOUNT process on the MediaAgent will be used to verify communication and credentials to the array.

6

Once the hardware snapshot is complete, vCenter is contacted to delete the software snapshots.

8
9
vCenter
Software Snapshots

Once software snaps are removed, vsbkp contacts MediaAgent to initiate createindex.exe. The createindex takes information from vsbkp on all the files that were part of the hardware snapshot.
Configuring and Administering IntelliSnap® Technology

- Array Configuration
- Storage Policy Configuration
- Subclient Configuration
- Running Snapshot Operations
- Managing Snapshots
Array Configuration

Hardware arrays are configured from the Array Management applet which can be accessed from Control Panel or from the Manage Array button in the subclient. All configured arrays will be displayed in the Array Management window. Multiple arrays can be configured, each with their specific credentials. For some arrays, a Snap Configuration tab will be available to further customize the array options.
Subclient Configuration

In order to protect production data using IntelliSnap technology, the client must be enabled for the IntelliSnap feature and a subclient must be configured defining the content to be snapped and the IntelliSnap feature must be enabled for the subclient.

To enable the IntelliSnap feature for the client: select the client properties, click the Advanced button and check the Enable IntelliSnap option.

Once the IntelliSnap feature has been enabled for the client the IntelliSnap tab will be used to enable snapshot operations. Enabling the IntelliSnap check box will designate the contents of the subclient to be snapped when schedules for the subclient are executed. The snap engine must be selected from the drop down box. Use the Manage Array button to configure a new array, if one has not already been configured. A specific proxy can be designated for backup copy operations. This proxy must have the appropriate software and hardware configurations to conduct the backup copies. Refer to CommVault’s documentation for specific hardware and software requirements for the array and application data that is being snapped.

Once IntelliSnap operations have been configured for the subclient, ensure the subclient is associated with a snap enabled Storage Policy.
Storage Policy Design

- **Primary snap copy** - precedence 1
- **Primary classic** (backup) copy – precedence 2
- Snapshot scheduled at client (subclient) level
- Backup copy scheduled at storage policy
- Additional secondary copies are sourced from backup copies

Storage Policies can be used to manage both traditional data protection operations and snapshot operations. A Storage Policy can have a primary (classic) copy and a snap primary copy.

A primary snap copy can be added for any Storage Policy by right clicking the policy. Selecting All Tasks and then Create New Snapshot Copy. The copy can be given a name, define a data path location to maintain indexing data, and retention settings can be configured.

Retention can configured to maintain a specific number of snapshots, retain by days or retain by cycles. Note that if the days or cycles criteria is going to be used, it is critical have a complete understanding of how days and cycles criteria operate.
IntelliSnap® Backup Copy Operations

Backup copy jobs are when snapshot data is backed up to protected storage. The Storage Policy snap copy is used to manage snapshots and the primary (classic) copy is used to manage backup data. Typically data is protected to the primary (classic) copy by scheduling backups on the production host. Use the Create Backup Copy option in the storage policy drop down menu to generate backup copies of snapshot data.

Backup copy options includes:

- Number of simultaneous jobs.
- Start new media and mark media full which is used when isolating backup copy jobs to tape media.
- Job initiation options which include Run Immediately, Schedule, Automatic and Save as Script. The automatic copy option will execute automatically at predefined intervals (default 30 minutes) and detect if any snap copies are eligible to be copied to protected storage. If there are eligible copies they will be backed up and if not the job will terminate and execute again at the next check interval.
By default a backup copy will copy all available snapshots to protected storage. This can be customized in the Storage Policy properties, Snapshot tab. In the Job Selection rules section, select the Advanced button to specify which snapshots will be selected for backup copy operations. This is useful when you periodically conduct snapshots of production data but just want to backup one of the snaps, such as creating a daily full backup from the last snapshot of the day.
IntelliSnap® Design Strategies
Virtualization Snapshot Solutions for VSA

The Simpana IntelliSnap® feature provides integration with supported hardware vendors to conduct, manage, and backup snapshots. This technology can be used to snap VMs at the volume level and back them up to protected storage.

The IntelliSnap for VSA features provide the following benefits:

- Fast hardware snapshots result in shorted VM quiesce times and faster software snapshot deletes. This is ideal for high transaction virtual machines.
- Live browse feature allows administrators to seamlessly mount and browse contents of virtual machines for file and folder based recovery.
- Revert operations can be conducted in the event of DataStore corruption. For NetApp arrays, individual virtual machine reverts can also be conducted.
- Hardware snapshots can be mounted to an ESXi proxy server for streaming backup operations eliminating the data movement load on production ESXi hosts.
Module 6: Data Management
Client Processes

CommServe

- AppMgrSvc
- EvMgrS

MediaAgent

- CLBackup: Reads Collect Files and Sends Data
- CLIDRestore: Indexed Based Restores
- CLIFRestore: Index Free Job Based Restores

Client Side Dedupe
Signature Caching

Status Updates

Client Configuration

Reads Subclient Content for Scan / Generate Collect Files

CommVault® Master Class Updated September 15, 2014
Agent and Subclient Customization

• On Demand Backup Set
• Custom Subclients
  • Performance
  • Custom content
  • Retention
  • Additional settings
• Agent specific options – Discussion
Data Protection Process

1. Synthetic Full verify
2. Pre Scan
   • Script insert
   • Resource reservation
     • Advanced backup
     • Storage policy
3. Index Failures
   • Convert to full
4. Scan
   • Collect files
5. Backup
   • Index vs. No Index
   • Chunk size
   • Block size
6. Archive index
Simpana® OnePass™

Simpana OnePass™ feature is a comprehensive solution incorporating traditional back up and archiving processes in a single operation. Data is backed up only once as part of the backup operation and objects that meet archiving rules are deleted or optionally stubbed in place. Stubs are application and user access points to facilitate the recall of the data that was moved. Simpana OnePass is able to selectively age archived objects separately from backed up data allowing longer retention before pruning. This allows you to reclaim space in your secondary storage.

Predicting Archive Benefits

The benefits of OnePass can be predicted for Windows File Systems and Microsoft Exchange Server. System Discovery and Archive Analyzer tools are non-intrusive and highly secure tools that externally collect file system and email details from a selected list of servers. Data collected by the tools can be uploaded to a virtual CommCell from which archive analysis and other reports can be generated.

Role of Synthetic Full

For OnePass archiving, a Synthetic Full job is used to facilitate retention of objects in protected storage by including deleted objects when synthesizing the new full backup. Protected storage is library or cloud storage managed by Simpana® Software.

A Synthetic Full job uses the previous incremental backup job’s inventory of all objects scanned (image file) to create the new full subclient content in protected storage. For a regular Synthetic Full job, the inventory list is used to read objects from the protected storage then immediately write them back to
the protected storage as the newly synthesized full backup. With deduplicated storage, a DASH Synthetic Full mimics this process by just updating the object index and deduplication database (DDB). OnePass archiving uses Synthetic Full to carry forward archived objects by appending a list of deleted objects to the Synthetic Full

**Role of Stubs**

A *stub* is a small placeholder similar to a shortcut file for an object that has been archived. A stub contains necessary information to recall the original object should the stub be opened. Stubs are optional with archiving. If a stub is not used or the object is deleted, the archived object can still be restored in the same manner as a backed up object. Stubs are also backed up and can be restored in the same manner as a backed up object without recalling the archived object.

**Delayed Stubbing**

Objects (files and messages) that are archived will not be immediately stubbed. Stubbing (if enabled) occurs with the next job after the next Disaster Recovery Backup of the CommServe databases and a configurable (default 24 hours) time.

For example, a backup job is started at 8pm with objects meeting archive rules. A DR Backup is run the next day at 10am. Another backup job run at 1pm will not stub the object since the time difference has not been met. Another backup job is run at 8pm. This job will stub the qualified objects from the previous 8pm backup job since both the DR backup and time difference has been met. This ensures that in a disaster recovery scenario you can roll back to a previous CommServe DR version without any data loss.

Without delayed stubbing, should you perform a DR restore that doesn’t include the most recent jobs, recalls might fail for objects that were backed up after a DR backup and before the DR restore.

**OnePass Archiving Only**

There may be circumstances where you want to manage Backup and Archiving operations separately from each other. For example, you may want to send archived data to a separate storage policy or run archiving operations on a different schedule. For File system agent, the best way to handle this is to create a new backup set. In the subclients for the backup set, enable the subclient content option to *Only backup files that qualify for archiving*.
This section is being provided as a detailed example of a job process within a CommCell environment. In this example, the auxiliary copy process is being expanded to include detailed process steps and corresponding log entries. It is not that more detail is required for auxiliary copy operations, but rather this is being used simply as an example of how jobs communicate with multiple processes and log entries in various log files.

**Step 1: Job Manager Initiates Auxiliary Copy Operation (JobManager.log).**
Job manager will initiate auxiliary copy. Log will indicate if job was schedule or immediately executed.

```
3916 11c8 07/07 16:25:27 1085 Servant [---- IMMEDIATE AUXILIARY COPY REQUEST ----]. Task Id [263]
```

**Step 2: Resources on the source MediaAgent and library will be reserved.**
Log will include storage policy and library information.

```
3916 d34 07/07 16:25:33 1085 Resource Reserved Resource: Reservation [1485], ResourceUser [390], SP [Storage Policy], Copy [Disk Copy[ID:59]], MediaGroup [325], Volume [799], Media [CV_MAGNETIC[ID:15]], Drive [E[ID:9]], DrivePool [DrivePool(CVMA1)11(ID:11)], Library [Magnetic Library], MediaAgent[CVMA1], PRChckFailCount = 0, RMChckFailCount = 0
```
Step 3: Resources on the destination MediaAgent and library are reserved.
Log will include storage policy and library information.

3916 d34 07/07 16:25:33 1085 Resource Reserved Resource: Reservation [1486], ResourceUser [391], SP [Storage Policy], Copy [Tape Copy(ID:60)], MediaGroup [326], Volume [611], Media [004015L2(ID:16)], Drive [IBM ULTRIUM-TD2_1(ID:10)], DrivePool [DrivePool(CVMA2)14(ID:14)], Library [Tape Library], MediaAgent[CVMA2], PRChckFailCount = 0, RMChckFailCount = 0

Step 4: AuxCopyMgr process starts on CommServe server (AuxCopyMgr.log)
3916 12a0 07/07 16:25:33 1085 Scheduler Phase [1-Auxiliary Copy] (0,0) started on [sol.cemm.lab] - auxCopyMgr.exe -j 1085 -t 263 -a 43 -jt 1085:1

Step 5: AuxCopyMgr Process on CommServe server starts reading chunk data from CommServe database (AuxCopyMgr.log)
AuxCopyMgr process reads AuxCopyMgr.log information which contains information for auxiliary copy job

1800 12e8 07/07 16:25:36 1085 AuxCopyManager::getConfigParams Job option: Continue with next chunk on read errors.
1800 12e8 07/07 16:25:37 1085 AuxCopyManager::getConfigParams Job option: Max number of chunks per message [10].
1800 12e8 07/07 16:25:37 1085 AuxCopyManager::getConfigParams Job option: Max number of jobs per message [20].
1800 12e8 07/07 16:25:37 1085 AuxCopyManager::getConfigParams Job option: Report progress every [512] MB.

Step 6: AuxCopy process starts on source MediaAgent (AuxCopy.log) and begins receiving chunk data from AuxCopyMgr
1. AuxCopy.log information will include storage policy, library, stream, chunk size, encryption settings, etc...
2. AuxCopy starts pipeline with CVD to the destination MediaAgent.
3. AuxCopy starts reading chunk data and sends it through CVD to destination MediaAgent. Log will report each chunk as sent and completed.

5296 1590 07/07 16:25:40 1085 +++ AuxCopy Thread Params +++
5296 1590 07/07 16:25:40 1085 Source Copy [Primary] ID [59] Is Dedup Copy [0]
5296 1590 07/07 16:25:40 1085 Target Copy [Secondary] ID [60] Is Dedup Copy [0]
5296 1590 07/07 16:25:40 1085 Target RC ID [391] Source RC ID [390]
5296 1590 07/07 16:25:40 1085 +++ Source Chunk Info +++
5296 1590 07/07 16:25:40 1085 Source ChunkId [1216]
5296 1590 07/07 16:25:40 1085 CommCellId [2]
Step 7: Source MediaAgent uses CVD process to establish data pipe to destination MediaAgent

AuxCopy on source MediaAgent uses CVD to establish pipeline with destination MediaAgent

1. CVD receives incoming connection from AuxCopy.
2. Receives mount request from CVMA. Log will show media type, block size drive serial number, SCSI reservation, chunk size, compression options and OML information.

CVD process on destination MediaAgent receives connection request from AuxCopy

CVD receives mount information from CVMA

CommVault® Education Services
Path to (tape) library and block size being used
1252 dcc 07/07 16:25:53 1085 391-1486 [MEDIAFS ] mediatype [142], mountpath [\\\Tape0]
1252 dcc 07/07 16:25:53 1085 391-1486 [MEDIAFS ] The volume will be recorded with the block size [64] KB

Serial number match for destination drive
1252 dcc 07/07 16:25:53 1085 391-1486 [MEDIAFS ] Serial number 1110310836 for drive, access path- \\Tape0 Matched succesfully.
SCSI-2 reservation completes successfully
1252 dcc 07/07 16:25:53 1085 391-1486 [MEDIAFS ] Successfully completed SCSI2 reservations for drive accespath \\Tape0.

Chunk size set for auxiliary copy
1252 dcc 07/07 16:25:56 1085 391-391 [DM_BASE ] The size of the chunk will be around 4096 MB

CommServe database updates information for new chunk ID to be written to destination MediaAgent
after setting the volume id for the chunk in the database

Entry shows hardware compression has been enabled

Media Label information for tape media being used is read and logged
1252 dcc 07/07 16:25:57 1085 OML [] VERIFY OML returned OML on the media
MagicNumber =CVMEDIALABEL
LabelVersion =9.0.0(BUILD84)
MMSCommCellId =3339
LabelType =CommVault Media Label
Vendor =CommVault Systems
MediaCreationTime =1302807101
Application =Galaxy
MediaName =
MediaID =3339_BC_004015L2_16
LabelGUID =0
BarCode =004015L2
SideName =A_16
FriendlyName =
CheckSum =

Step 8: Chunk data movement from source to destination MediaAgent
Chunk data will be written to destination MediaAgent. Log will show each chunk successfully written. Size of data, time to write and write speed.

AuxCopy Process – Destination MediaAgent (AuxCopy.log)

1. Upon write of chunks, AuxCopy will request additional chunks. Once all chunks written, process will begin quit routine.
2. Log file will also include performance values including size, time, speed and next chunk receive time.
**AuxCopyMgr Process (AuxCopyMgr.log)**

1. AuxCopy reports back to AuxCopyMgr as each chunk is successfully written.
2. Once all chunks copied, AuxCopyMgr reports to AuxCopy processes to quit.
3. AuxCopyMgr begins quit routine and reports to JobMgr that the job has completed successfully.

**Starts writing chunk information to destination location**

```
1252 dcc 07/07 16:25:57 1085 391-391 [MEDIAFS ] Ready to write to the filemarker=1 on the Tape Volld=611
```

**Chunk is successfully written to destination**

```
1252 dcc 07/07 16:26:00 1085 391-391 [MEDIAFS ] Writing TapeMark 2 on the tape for volume Volld=611
1252 dcc 07/07 16:26:01 1085 391-391 [MEDIAFS ] Size of galaxy data on disk is 31129600 Total write time to media in seconds =4
1252 dcc 07/07 16:26:01 1085 391-391 [DM_BASE ] Successfully closed chunk on Media for archive file id =-1, VolumeId = 611
```

**CVD tracks performance information for write operation**

```
1252 dcc 07/07 16:26:04 1085 ID [DSBackup Media Write Speed], Job Id [1085], Bytes [31142049], Time [3.534008] Sec(s), Average Speed [8.403878] MB/Sec
```

**AuxCopy process on destination MediaAgent requests additional chunk information**

```
```

**AuxCopy logs performance information for write operation**

```
5296 1590 07/07 16:26:04 1085 ID [Media Read Speed], Job Id [1085], Bytes [31096403], Time [0.097533] Sec(s), Average Speed [304.058698] MB/Sec
5296 1590 07/07 16:26:04 1085 ID [Next chunk recv times], Job Id [1085], Samples [2], Time [0.167610] Sec(s), Average [0.083805] Sec/Sample
5296 1590 07/07 16:26:04 1085 ID [Media Open Times], Job Id [1085], Samples [2], Time [1.139735] Sec(s), Average [0.569867] Sec/Sample
5296 124c 07/07 16:26:07 1085 Reader [1] <Copy/Stream> Source <59/1> Target <60/1>: Reporting FREE_STREAM to the auxcopy manager. ChunkId [1218] Bytes copied [0]
```

**Step 9: Chunk Copies Logged in Database and Job Completion**

**AuxCopyMgr logs chunks in CommServe database – once all chunks copied it communicates with AuxCopy to quit**

```
1800 12e8 07/07 16:25:59 1085 AuxCopyManager::handleSuccessReport <Copy/Stream> Source <59/1> Target <60/1>: Chunk [1216] has been read successfully. [28917221] bytes
1800 12e8 07/07 16:26:00 1085 AuxCopyManager::handleSuccessReport <Copy/Stream> Source <59/1> Target <60/1>: Chunk [1218] has been read successfully. [2179182] bytes
```

**AuxCopyMgr starts exit routine**

```
1800 12e8 07/07 16:26:08 1085 AuxCopyManager::finish Set job status as SUCCESS after checking completion
1800 12e8 07/07 16:26:08 1085 AuxCopyManager::finish *** Job [1085] completed successfully ***
```

**AuxCopyMgr reports to JobManager that copy successfully completed**

```
1800 12e8 07/07 16:26:08 1085 COMPLETE CALLED (PHASE Status::SUCCESS), Job ID = 1085
JobManager process shows 100% complete in Job Controller and updates log for job as complete
3916 11cc 07/07 16:26:08 1085 JobSvr Obj Phase [Auxiliary Copy] for Job Completed.
```
Understanding Log Files for Data Movement Process
Anatomy of a Log File

Thread ID of the Job Manager Process
Process ID of the Job Manager process
Date and Time of the Event
Job ID number
Process Subroutine
Descriptions
Navigating Log Files - Job Phases

Scan Phase

Backup Phase

Archive Index Phase

Job Complete

66 JobSvr Obj Phase [4-Scan] for Backup Job Completed. Backup will continue with phase [Backup].

66 Scheduler Phase [7-Backup] (0,0) started on [commserve.company.com] - cBackup.exe

66 JobSvr Obj Phase [7-Backup] for Backup Job Completed. Backup will continue with phase [Archive Index].
## Job Phases and Log Files

<table>
<thead>
<tr>
<th>Job Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Finish Synthetic Full</td>
</tr>
<tr>
<td>2. Index Restore</td>
</tr>
<tr>
<td>3. PreScan Process</td>
</tr>
<tr>
<td>4. Scan</td>
</tr>
<tr>
<td>5. PostScan Process</td>
</tr>
<tr>
<td>6. PreBackup Process</td>
</tr>
<tr>
<td>7. Backup</td>
</tr>
<tr>
<td>8. System State Backup</td>
</tr>
<tr>
<td>9. PostBackup Process</td>
</tr>
<tr>
<td>10. Archive Index</td>
</tr>
<tr>
<td>11. Stubbing</td>
</tr>
</tbody>
</table>
Retention
Storage Policy Retention

With Simpana® features such as deduplication, DASH-Full, DASH-Copy and SILO tape storage, the philosophy and approach to configuring retention has changed significantly. Where organizations would traditionally conduct full backups on weekends when resources were not being used, Client Side Deduplication and DASH-Full now allows Full backups to run incredibly fast and use less network bandwidth. DASH-Copy makes copying data to secondary disk locations on or off site significantly faster using minimal bandwidth. The SILO to tape feature makes it possible to not even bother with retention and keep everything forever. These features are changing the way CommVault promotes configuring retention policies. In this section the focus will be on understanding retention and how these new features can allow Simpana administrators to think outside the box when implementing retention strategies.

Retention Rules
Policy based retention settings are configured in the storage policy copy Retention tab. The settings for backup data are Days and Cycles. For archive data the retention is configured in Days. Retention can also be set through schedules or applied retroactively to a job in a storage policy copy.

Cycles
A cycle is traditionally defined as a complete full backup, all dependent incremental, differential, or log backups; up to, but not including the subsequent full. In real world terms a cycle is all backup jobs required to restore a system to a specific point in time. To better understand what a cycle is we will
reference a cycle as **Active** or **Complete**. As soon as a full backup completes successfully it starts a new cycle which will be the active cycle. The previous active cycle will be marked as a complete cycle.

An active cycle will only be marked complete if a new full backup finishes successfully. If a scheduled full backup does not complete successfully, the active cycle will remain active until such time that a full backup does complete. On the other hand a new active cycle will begin and the previous active cycle will be marked complete when a full backup completes successfully regardless of scheduling.

In this way a cycle can be thought of as a variable value based on the successful completion or failure of a full backup. This also helps to break away from the traditional thought of a cycle being a week long, or even a specified period of time.

**Days**

A day is a 24 hour time period defined by the start time of the job. Each 24 hour time period is complete whether a backup runs or not. In this way a day is considered a constant.

**Days and Cycles relation**

A rule of thumb that has been followed for years was that cycles and days should directly or indirectly equal each other. 2 cycles and 14 days with weekly full backups. 4 cycles and 30 days being approximately 1 month. 12 cycles and 365 days for month end fulls being retained for a year. But what about 52 cycles and 365 days? In situations like this it is rather irrelevant how many cycles are set. The truth is, 2 cycles and 365 days is good enough. You will meet your retention requirements since you will be keeping data for one year and if backups don’t run for over a year you are still guaranteed to have at least 2 cycles of data in storage based on the aging entire cycles rule.

When setting retention in the policy copy, base it on the primary reason data is being protected. If it is for DR ensure the proper number of cycles are set to guarantee a minimum number backup sets for full restore. If you are retaining data for data recovery then set the days to the required length of time determined by retention policies. If the data recovery policy is for three months, 12 cycles and 90 days or 1 cycle and 90 days will still meet the retention requirements.

**Data Aging for Non-Deduplicated Data**

There are two processes that will be performed during a data aging operation. **Aging** simply marks jobs that have exceeded retention as aged. **Pruning** will physically delete eligible disk jobs or recycle a tape when all jobs on it have been marked aged.

The **Data Aging** process will compare the current retention settings of the storage policy copy to jobs in protected storage. Any jobs that are eligible to be aged will be marked aged. By default the data aging process runs every day at 12PM. This can be modified and multiple data aging operations can be scheduled if desired.

**Pruning** is also part of the data aging process. How Pruning occurs depends on whether jobs are on disk or tape. For disk jobs if Managed Disk Space is disabled and no auxiliary copies are dependent on the jobs, they will be pruned. This will physically delete the data from the disk. If deduplication is being
used, job blocks that are not being referenced by other jobs will be deleted. If Managed Disk Space is enabled, the jobs will remain until the Disk library reaches the upper watermark threshold defined in the Library Properties.

For tape media, when all jobs on the tape have been marked as aged, and there are no auxiliary copies dependent on the jobs, the tape will be moved into a scratch pool and data will be overwritten when the tape is picked for new data protection operations. In this case the data is not deleted and can still be recovered by browsing for aged data, until the tape label is overwritten. If the storage policy copy option ‘mark media to be erased after recycling’ has been selected or if the tape is manually picked to be erased, the data will physically be destroyed. This is done by overwriting the OML header of the tape making the data unrecoverable through the CommCell environment or using Media Explorer.

Rules for Aging Data
There are several rules that are applied during the data aging process

1. Both days and cycles criteria must be met for aging to occur.
2. Data is aged in complete cycles.
3. Days criteria is not dependent on jobs running on a given day.

Rule 1: Both CYCLES and DAYS criteria must be met before Data will age
Simpana software uses AND logic to ensure that both retention parameters are satisfied. Another way of looking at this is the longer of the two values of cycles and days within a policy copy will always determine the time data will be retained for.

Rule 2: Data is aged in complete cycles
Backup data is managed within a storage policy copy as a cycle or a set of backups. This will include the full which designates the beginning of a cycle and all incrementals or differentials. When data aging is performed and retention criteria allow for data to be aged, the entire cycle is marked as aged. This process ensures that jobs will not become orphaned resulting in dependent jobs (incremental or differential) existing without the associated full.

Rule 3: Day is based on a 24 hour time period
A day will be measured as a 24 hour time period from the start time of a data protection operation. Days are considered constants since regardless of a backup being performed or completed successfully the time period will always be counted. If a backup fails, backups are not scheduled or if power goes out a day will still count towards retention. This is why it is so critical to measure retention in cycles and days. If retention was just managed by days and no backups were run for a few weeks all backup data may age off leaving no backups.
Job Based Retention

Job Based Retention

- Set retention on job schedule
- Change retention on completed job(s)
  - From storage policy
  - From media groups
- Enable / disable data aging
  - CommCell
  - Policy Copy
  - Client

Job Based Retention

Typically retention is based on company policy and therefore managed through storage policy retention settings that affect all data being managed by the policy. There may be situations where jobs retention would need to be individually set. There are two methods to apply job based retention: through schedules or through storage policy copy job history.

Retention Set Through Schedules

Retention can be extended beyond the defined storage policy primary copy retention through a schedule or schedule policy. This is done by setting the Extend Job Retention options in the Media tab of Advanced Options. The default setting is to use storage policy primary copy retention settings. You can set schedule based retention for a specified number of days or infinitely retain the data. Retention settings at the schedule level cannot be shorter than the retention defined in the storage policy primary copy.

Retention Applied to Job in Policy Copy

Retention for a job in a primary or secondary storage policy copy can be retroactively modified by going to the job history for the copy. Do this by selecting the storage policy copy where the job is located, right click the copy and select View | Jobs. Specify the time range of the job then click OK. Right click on the job and select Retain Job. The job can be retained infinitely or until a specific date. The job icon will change to reflect that the job has been pegged down.
Object Based Subclient Retention

Simpana® Version 10 software offers two primary retention methods:

- **Job based retention** – Configured at the Storage Policy copy level, job schedule level or manually by selecting jobs or media to retain, and apply different retention.
- **Object based retention** – Configured at the subclient level, it applies retention based on the deletion point of an object. Object based retention is based on the retention setting in the subclient properties plus the Storage Policy copy retention settings.

How Object Based Retention Works

In order to understand how object based retention works, an explanation of synthetic full backups, a key component of its functionality is needed.

A synthetic full backup synthesizes a full backup by using previous data protection jobs to generate a new full backup. Objects required for the synthetic full backup will be pulled from previous incremental or differential backups and the most recent full. To determine which objects are required for the synthetic full, an image file is used. An image file is a logical view of the folder structure including all objects within the folders and is generated every time a traditional backup is executed. The synthetic full backup will use the image file from the most recent traditional backup that was conducted on the production data to determine which objects are required for the new synthetic full.

When an image file is generated, all objects that exist at the time of the scan phase of the backup job are logged in the image file. This information will include date/time stamp and journal counter.
information which is used to select the proper version of the object when the synthetic full runs. If an object is deleted prior to the image file being generated, it is not included in the image file and will not be backed up in the next synthetic full operation. The concept of synthetic full backups and deleted objects not being carried over to the synthetic full is the key aspect of how object based retention works.

THE FOLLOWING DIAGRAM ILLUSTRATES HOW SYNTHETIC FULL BACKUPS WORK. IMAGE FILES ARE GENERATED EACH TIME A BACKUP JOBS RUNS. THE LATEST IMAGE FILE IS USED TO DETERMINE WHICH OBJECTS ARE USED FOR THE SYNTHETIC FULL, SELECTING THE PROPER VERSION OF THE OBJECT. IF AN OBJECT IS DELETED PRIOR TO THE MOST RECENT IMAGE FILE BEING GENERATED, IT WILL NOT BE CARRIED OVER TO THE NEXT SYNTHETIC FULL.

Object based retention uses the principals of synthetic full backups to create, in a way, a carry forward image file. When an object is deleted from the production environment, the object is logged with a countdown timer which is based on the subclient retention setting. The object will be carried forward with each subsequent synthetic full backup until the timer reaches zero. When the time has expired, the object will no longer be carried forward and once the synthetic full exceeds Storage Policy copy retention it is pruned from protected storage. So if the subclient retention is set to 90 days, once the item is deleted it will be carried forward with each synthetic full backup for a period of 90 days.

Requirements for Synthetic Full Backups
In order for subclient retention to function properly, only synthetic full backups can be used. If a full backup is conducted, it will break the chain of carrying forward deleted items. When a subclient has been enabled for OnePass archiving, the option to run traditional full backups will no longer be available. However, if OnePass is not enabled, subclient retention can still be used but the option to run traditional full backups will still be available. It is critical that if subclient retention is going to be used to manage data retention that traditional full backups are no longer conducted for the subclient.
Subclient and Storage Policy Retention Combination

It is important to note that subclient retention is not used in place of Storage Policy based retention, rather the two retentions are added to determine when an object is pruned from protected storage. If an object is carried forward for 90 days upon deletion, each time a synthetic full runs it will be carried forward until the 90 days elapses. The synthetic full backups themselves are retained based on the Storage Policy copy retention rules. So if the Storage Policy copy has a retention of 30 days and 4 cycles, then a synthetic full will remain in storage until the job exceeds retention. In this instance, the object is carried forward for 90 days and the last synthetic full that copies the object over will be retained for 30 days, then the object will remain in storage from time of deletion for 120 days – 90 day subclient retention and 30 day Storage Policy copy retention.

Subclient Retention Settings

A new tab in the subclient properties page ‘Retention’ is used to configure object based retention. The setting determines how long to keep an object from the point in which is deleted. There are three options available:

- **Delete immediately** – This does NOT mean to delete immediately. What this means is to ignore any subclient retention settings and follow Storage Policy retention. Once an object is deleted, it will not be carried forward to any synthetic full backups.
- **Keep for nnn days** – From the point in which an object is deleted, the keep for setting determines how many days the deleted object will be continued to be carried forward to new synthetic full backups.
- **Keep forever** – When the object is deleted it will be carried forward to new synthetic full backups indefinitely.

1 Cycle and 0 Days Storage Policy Retention

One strategy for using subclient retention is to set the storage policy primary copy retention to one cycle and zero days. This method can be used to bypass Storage Policy based retention for the primary copy. If this method is used then the retention for a deleted objects would be based on the subclient retention setting and the frequency in which synthetic full backups are run. If the subclient retention is 90 days and a synthetic full is run once a week, a deleted object will remain for up to 97 days depending on which point in time the object was deleted. If it was deleted a day prior to the synthetic full, then it will be retained for 91 days – right after the synthetic full finishes it will be retained for 97 days.

To configure object based retention to a definitive number of days, which may be required for compliance purposes, the Storage Policy copy retention can be set for 1 cycle and 0 days and synthetic full backups can be run every day. For best performance, this method should only be used with Simpana deduplication and DASH Full backup operations.

Storage Policy Secondary Copies

Object based retention applies to how long an item will be carried forward when synthetic full backups are run. This applies to backup jobs managed by the Storage Policy primary copy. Secondary copies will always have retention applied to the copy in the traditional manner. If subclient retention is set to 90 days, Storage Policy primary copy retention is 1 cycle and 0 days, and synthetic full backups are being
run daily; a deleted item will be retained for 91 days. If a secondary copy has been configured with a retention of 8 cycles and 90 days, the object may be retained for up to an additional 90 days.

How long a deleted object is potentially retained in a secondary copy depends on the copy type. If the secondary copy is a synchronous copy then the deleted object will always be retained for the retention defined in the secondary copy since all synthetic full backups will be copied to the secondary copy. Selective copies however, allow the selection of full backups at a time interval. If synthetic full backups are run daily and a selective copy is set to select the month end full, then any items that are not present in the month end synthetic full will not be copied to the selective copy. To ensure all items are preserved in a secondary copy, it is recommended to use synchronous copies and not selective copies.

**Object Based Retention Benefits**

The primary benefit of subclient based retention is the efficient use of storage. Traditionally, specific backups such as month end or quarter end were retained for long periods of time or kept indefinitely. There are three major drawbacks to this approach:

1. Cost for storing data long term can be expensive.
2. Each time a long term job is retained there are numerous redundant objects that are within the job – the same objects that were in the previous job. These redundantly stored objects add up over time.
3. Selecting periodic jobs for long term retention does not guarantee all data is protected. If a month end job is retained for 10 years and an object was deleted several days prior to month end, the object will not be present in the month end job.
Variants on Retention

Managed Disk Space
Managed Disk Space is a feature used with disk libraries which allows data to reside on the disk beyond its retention settings. This allows you to increase the chances of recovering data faster from primary storage on disk without changing retention settings. Managed disk space cannot be used when using Simpana deduplication.

Managed data will be held on the disk beyond the standard retention settings until an upper threshold is reached. A monitoring process will detect data exceeding the upper threshold and then delete aged jobs from the media until a lower threshold is reached. It is important to note that only aged jobs will be pruned. If all aged jobs are pruned and the lower threshold is not met no more pruning will occur.

Managed disk thresholds are configured in the disk library properties and can be enabled in each storage policy copy.

As a general rule of thumb the upper threshold should be set to allow one hour of backups to run after the threshold is reached. The lower threshold should be set so that the managed disk space pruning operation will not run more than once in a backup time period as the pruning operation will have a negative effect on the performance of backups.
Custom Calendars

Custom business calendars allow custom calendars to be defined based on fiscal time periods. The standard calendar used by Simpana software runs from January 1st to December 31st. This can result in period based jobs with selective copies or extended retention rules to protect the wrong jobs. Setting a custom calendar allows for selective copies, extended retention rules, and job schedules to correspond to user-defined calendars.

Calendars are defined in the Custom Calendars applet in Control Panel. A calendar can be defined and set as the default calendar for all operations. Multiple calendars can also be created and then associated with specific policy copies or schedules.

Another use of custom calendars is the ability to define custom months. You can set every month to start on a Friday or Saturday. You can set all months in a fiscal year to have 28 or 35 days. The use of custom months adds a level of complexity into the environment but it provides a powerful method to customize time periods to meet different protection requirements.
Structured and Unstructured Data

Using a database to seek out sales records and information is an example of structured data. Within the database system information is organized and indexed in a manner that allows for fast access to relevant information. The database will contain multiple tables that are linked together that contain different records. Each record within a table will contain different information. One table may contain address and contact information for a customer. Another table will contain shipping information and another will contain sales entries. When the user accesses the record for the sales order the database system will quickly access all of the relevant information and present it to the user as the sales record. This record will contain the customer contact information, shipping information and the sales information. This represents structured data.

Now let’s say the same information was stored using a different method. The sales person keeps contact information for all customers in a spreadsheet which he keeps on his computer. Someone in finance logs all sales orders in their own spreadsheet. The final sales order is drafted in a word document. The shipping department logs all shipments in a desktop database application running on a standalone workstation. Accessing the required information would be considerably more difficult. This represents unstructured data.

The concepts of structured and unstructured data are the essence of what information management is all about. If everything in a datacenter was maintained in database systems that could be linked...
together and accessed through a single interface, information management would be simple. In modern business environments information exists in so many locations it may seem impractical to successfully manage, preserve and access it. Although several different models have been developed to attempt to organize information, these models are more conceptual and ideological rather than practical. Some software and hardware applications have attempted to meet the complex requirements of these models but the capabilities of these systems have traditionally been limited. They may provide powerful capabilities that meet the requirements of one aspect of information management but they fall far short of providing a comprehensive information management strategy.

Data Management

*Data Management* is the idea of treating large amounts of data in bulk and simply identifying the data based on what it is and where it is stored. User files for example are treated as data based on the name of the file and where it is located. Email data is addresses based on the database in which it resides.

**Data management policies are based on the three primary reasons for protecting data:**

- **Disaster recovery** – is the primary reason data is protected. It provides the ability to recover business systems, servers, disks or entire sites in the event of a limited or complete data loss.
- **Data recovery** - provides the ability to recover specific data. This is typically applied to end user files and Emails where a specific request is made to recover the data.
- **Compliance archiving** – Is the concept of taking point-in-time views of data and preserving the data for compliance reasons. Data such as financial databases, legal files, or mailboxes are examples of data that may require compliance copies to be created and preserved for long period of time.

Information Management

The concept of *Information Management* is addressing the data based on its content and value to an organization. When a user creates files and Emails they are considered information to the individual who created them and others who view them. The user accesses this information through front end applications and operating systems which are capable of presenting this information in a way they can understand. Managing information is the concept of indexing the contents of data and applying specific management policies based on the contents of the data.
Data Security
When CommCell® components need to communicate or move data through a firewall, firewall settings must be configured for each component. This can be done by configuring individual firewall settings for a specific client or firewall settings can be applied to a client computer group. For example, if a client needs to communicate with a CommServe server through a firewall and backup data to a MediaAgent through a firewall, all three components would require firewall configuration.

There are three primary methods for connecting through a firewall:

- **Direct** – where the CommCell components communicate directly with each other through a firewall.
- **Through a proxy** – where CommCell components use a proxy in a demilitarized Zone or DMZ to communicate with each other.
- **Gateway** – where CommCell components communicate through a gateway resource.

### Defining Firewall Rules for Client and Client Groups

To configure firewall settings for a client or client group, right-click on the entity in the CommCell console, select properties and then click the advanced button. Select the firewall tab and then click configure firewall settings. Click the advanced radio button to enable full firewall configuration.

- There are four configuration tabs available
- Incoming connections
- Incoming ports
- Outgoing routes
• **Options**

A fifth tab will show a summary of all options configured for the firewall settings. This summary will be in the format that will be used to populate the FWConfig.txt file that will be located in the base folder of all CommCell components using firewall configurations.

**Configuring Firewall Settings**

![Image of firewall configuration process]

**Configuring Incoming Connections**

The incoming connections tab is used to determine if other CommCell components can connect the client or client group where the firewall settings are being configured. There are three connection options:

- **Open connection** – there are no firewall restrictions. In this case, no incoming connections need to be configured.
- **Restricted** – there are firewall port restrictions in place and a component on the other side of the firewall can reach the component that is currently being configured.
- **Blocked** – there are firewall port restrictions in place and a component on the other side of the firewall can NOT reach the component that is currently being configured.

Simpana software uses port 8400 as the default communication port for all CommCell traffic. When firewall settings are enabled for a CommCell component, by default, port 8403 will be used as a listening port for any inbound connection attempts. Additionally, a dynamic port range can be configured to provide additional data traffic ports for backup and recovery operations. How these ports will be used is dependent on a number of factors:

1. Communication will be based on the “listen for tunnel connections on port” setting.
2. If port 8400 is available on the firewall, once initial communication is made using the listen port, by default, data transmission will use port 8400 and metadata and communication will use port 8403.

3. By default, a dynamic port range will not be used for data traffic. This is by design of the network model Simpana® software uses to transmit data to a MediaAgent. When the MediaAgent setting in the control tab, “optimize for concurrent LAN backups” is enabled, all data will be tunneled through a single data port. This means dynamic port ranges are not needed by Simpana software to backup and restore data through a firewall. In certain situations, performance may be improved by disabling the “enable for concurrent LAN backup” option and defining a dynamic port range. Keep in mind, that when the LAN optimization option is disabled, the maximum number of streams a MediaAgent can process will be limited to 25.

**Configuring Outgoing Routes**
The outgoing routes tab determines how CommCell components will communicate with each other.

**There are three route types:**
- Direct
- Via gateway
- Via proxy

For each route type, encryption options can be set by determining the connection protocol that will be used.

**There are three connection protocol options:**
- **Regular** – authentication and data will NOT be encrypted.
- **Authenticated** – authentication will be encrypted but data transfer will not be encrypted.
- **Encrypted** – Authentication and data will both be encrypted.

The default option ‘Authenticated’ is the recommended option. If data transfer requires encryption, consider using client ‘inline’ encryption instead of using the ‘encrypted’ option in the firewall settings.

**Configuring Options**
When the CommServe® server can reach clients to initiate data protection and recover jobs, it will be configured as restricted on the clients. If the CommServe server cannot communicate to the client, it will be configured as blocked and the client will be responsible for establishing connections with the CommServe server. The keep-alive interval and tunnel Init interval are used to determine how connections are made and maintained when the CommServe server is blocked from communicating with clients.

The ‘Tunnel Init Interval, seconds” option determines the frequency in which the client will attempt to establish a connection with the CommServe server. The “keep-alive interval, seconds” determines how long the connection will be kept alive. At the end of the keep alive interval which defaults to five minutes, the client will attempt to renew the connection.
Pushing Firewall Settings

Once all firewall settings have been configured, the summary tab will show the firewall output information which will be pushed to the CommCell components.

The configuration will need to be pushed using one of the three following methods:

1. Client services started – the client will communicate with the CommServe server which will push out firewall settings.

2. When Data Interface Pairs are configured it will automatically push firewall configuration settings.

3. Firewall configurations can manually be pushed to client groups or clients by right-clicking on the component, selecting all tasks, and then push firewall configuration.
One Way Firewall Configuration

Connection from client added to CommServe and MediaAgent set as Restricted

Optional incoming port range can be defined

Connection to CommServe and MediaAgent added as Blocked

Route setting from client to CommServe and MediaAgent set to Direct

Two Way Firewall Configuration

Connection from client added to CommServe and MediaAgent set as Restricted

Optional incoming port range can be defined for client, CommServe and MediaAgent

Connection to CommServe and MediaAgent added as Restricted

Route setting from client to CommServe and MediaAgent set to Direct
CommVault software offers three methods to encrypt data.

- Inline encryption will encrypt data during the backup job,
- Offline encryption which encrypts backup data while being copied to secondary copies, or
- Hardware encryption using LTO4, 5 or 6 drives.

Inline and Offline encryption is software based. Inline encryption can be performed on the Client or MediaAgent. Offline encryption will be performed on the MediaAgent. LTO 4, 5 and 6 drives support hardware encryption which is performed on the drive itself.
The following chart illustrates how encryption can be used with CommVault software and advantages / disadvantages of each method.

<table>
<thead>
<tr>
<th>Type</th>
<th>Where Encryption is Performed</th>
<th>How it is Enabled / Disabled</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>In-Line</td>
<td>Client or Media Agent</td>
<td>Turned on/off at subclient level</td>
<td>Allows encryption over network</td>
<td>Software based hits CPU &amp; memory of client or Media Agent</td>
</tr>
<tr>
<td>Off-line</td>
<td>Media Agent</td>
<td>Turned on/off at storage policy secondary copy.</td>
<td>Does not affect primary backup windows</td>
<td>Software based hits CPU &amp; memory of client or Media Agent</td>
</tr>
<tr>
<td>Hardware</td>
<td>LTO4, 5 or 6 drive with encryption support</td>
<td>Turned on/off at storage policy secondary copy.</td>
<td>Hardware based faster encryption &amp; no load on client or Media Agent</td>
<td>Requires dedicated hardware for backups and restores</td>
</tr>
</tbody>
</table>

With any of these encryption solutions, keys will always be stored in the CommServe® database. Optionally keys can be stored on the media as well. This can be useful when using the Media Explorer tool to recover data from media.
Additional Security Settings

Agent Installation
When installing a Simpana agent within a CommCell environment, the only required information to authenticate the install process is the host name or IP address of the CommServe server. To require an administrator username and password to be entered during the installation process, in the CommServe properties | security tab | select the option ‘require authentication for agent installation’.

CommVault Edge® Settings
Data Loss Prevention (DLP) is a file-level security solution that prevents unauthorized access to important data on laptop devices. DLP is comprised of two components, Periodic Document Encryption and Secure Erase.

Periodic Document Encryption enables the administrator to configure certain files to be locked according to settings in the CommCell console. End-users can also configure Periodic Document Encryption from the Web Console to protect documents on their own laptops.

The second component, Secure Erase, allows the administrator to configure certain files to be erased from a laptop when the laptop is offline for more than a set number of days. Secure Erase can be configured from the CommCell console and is only available to administrators.

Administrators can enable Periodic Document Encryption on a laptop from the CommCell console. If necessary, Secure Erase can also be configured to delete sensitive files on a client or client group. End
users have the ability to create their own passwords, called pass-keys, for authorizing access to files locked with Periodic Document Encryption.

These two features, when enabled ensure that the data remains secure. If the laptop goes missing, the end-user or the administrator can mark the device as lost or stolen within the CommCell which will render all “locked” data on the device essentially useless without the user created pass-key. If the lost or stolen laptop is recovered, the data can be recovered by an authorized user.
Performance
Stream Management

**Stream Management**

- Subclient Stream Settings
  - Data Readers
  - Multiple readers in drive or mount path
- Multiple subclients
- Application specific
  - SQL and Oracle database and log stream configuration
  - VSA and data readers
  - Agents that don’t support multiple streams

**Data Readers**

Data Readers determine the number of concurrent read operations that will be performed when protecting a subclient. For file system agents, by default, the number of readers permitted for concurrent read operations is based on the number of physical disks available. The limit is one reader per physical disk. If there is one physical disk with two logical partitions, setting the readers to 2 will have no effect. Having too many simultaneous read operations on a single disk could potentially cause the disk heads to thrash slowing down read operations and potentially decreasing the life of the disk. The Data Readers setting is configured in the General tab of the subclient and defaults to two readers.

**Allow multiple readers within a drive or mount point**

When a disk array containing several physical disks is addressed logically by the OS as a single drive letter, the Allow multiple readers within a drive or mount point can be used as an override. This will allow a backup job to take advantage of the fast read access of a RAID array. If this option is not selected the CommVault software will use only use one read operation during data protection jobs.

**Data Readers for virtual machine backups**

Virtual machines are backed up using a single stream or reader. This means the number of concurrent virtual machines that can be protected will always correspond to the number of data readers defined in the subclient.
Data Streams for SQL agents

Multiple streams can be used for a subclient to improve the backup performance of larger SQL databases. Traditionally, there has been a limitation in the restorability of multi-streamed SQL backups to tape media. If multiple subclient streams were combined to a single tape, they would need to be first staged to a disk target by aux copying the streams before the data could be restored. As of Simpana version 10 – SP7, when restoring multiple SQL subclient streams from a single tape, the restore operation will use the job results folder location on the client to cache the streams during the restore eliminating the need to stage the restore to disk.

Multiple Subclients

There are many advantages to use multiple subclients in a CommCell environment. These advantages are discussed throughout this book. This section will focus only on the performance aspects of using multiple subclients.

Running multiple subclients concurrently allows multi-stream read and data movement during protection operations. This can be used to improve data protection performance and when using multi-stream restore methods, it can also improve recovery times. Using multiple subclients to define content is useful in the following situations:

- **Using multiple subclients to define data on different physical drives** – This method can be used to optimize read performance by isolating subclient content to specific physical drives. By running multiple subclients concurrently each will read content from a specific drive which can improve read performance.

- **Using multiple subclients for iDataAgents that don’t support multi-stream operations** – This method can be used for agents such as the Exchange mailbox agent to improve performance by running data protection jobs on multiple subclients concurrently.

- **Using multiple subclients to define different backup patterns** – This method can be used when the amount of data requiring protection is too large to fit into a single operation window. Different subclients can be scheduled to run during different protection periods making use of multiple operation windows to meet protection needs.
Data Movement Parameters

<table>
<thead>
<tr>
<th>Data Movement Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Chunk Size</td>
</tr>
<tr>
<td>• Block Size</td>
</tr>
<tr>
<td>• Pipeline Buffers</td>
</tr>
</tbody>
</table>

**Chunk Size**

Chunk size for different agents can be configured in the media management applet in control panel for tape media. Chunk size can also be configured in the storage policy copy’s data path properties for disk, cloud and tape. Depending on the storage media defined in the data path, different chunk sizes may be recommended.

**Block Size**

Block size can be configured in the storage policy copy’s data path properties. A higher block size can result in better performance but all hardware including NIC, HBA, switches and drives must support the higher block setting.

**Pipeline Buffers**

The Data pipe buffers determine the amount of shared memory allocated on each computer for data pipes. The size of each buffer is 64K. By default, 30 data pipe buffers are established on each server for data movement operations. You can increase the data transfer throughput from the client by increasing the number of data pipe buffers.

When you increase the number of data pipe buffers, more shared memory is consumed by the client or MediaAgent. This may degrade the server performance. Therefore, before increasing the number of data pipe buffers, ensure there is adequate shared memory is available. You can optimize the number of data pipe buffers by monitoring the number of concurrent backups completed on the server.
Pipeline buffers is configured on a client or MediaAgent by adding the additional setting registry key: nNumPipelineBuffers. If the key is set on both the client and the MediaAgent, the client setting will take precedence. For detailed steps on configuring pipeline buffers, refer to: http://documentation.commvault.com/commvault/v10/article?p=features/network/data_pipe_buffers.htm