High Availability and Data Protection with Dell PowerScale Scale-Out NAS

April 2024

H10588.17

White Paper

Abstract

This white paper details how Dell PowerScale scale-out NAS and the OneFS operating system architecture provide high availability and data protection. These features address the challenges that organizations face as they deal with the deluge of digital content and unstructured data and the growing importance of data protection.

DELL Technologies

Copyright

Contents

Executive summary

Introduction

Today, organizations of all sizes across the full spectrum of the business arena are facing a similar problem: An explosion in the sheer quantity of file-based data they are generating and, by virtue, are forced to manage. This proliferation of unstructured data, often called “big data,” has left traditional storage architectures unable to satisfy the demands of this growth and has necessitated the development of a new generation of storage technologies. Also, broader data retention requirements, regulatory compliance, tighter availability service level agreements (SLAs) with internal or external customers, and cloud and virtualization initiatives are only serving to compound this issue.

Document purpose

This paper presents information for deploying, managing, and protecting Dell PowerScale clusters. This paper does not intend to provide a comprehensive background to the OneFS architecture.

For more details about the OneFS architecture, see the *PowerScale OneFS Technical Overview white paper*.

Audience

The target audience for this white paper is anyone configuring and managing a PowerScale clustered storage environment. It is assumed that the reader has an understanding and working knowledge of the OneFS components, architecture, commands, and features.

For more information about OneFS commands and feature configuration, see the *PowerScale OneFS CLI Administration Guide*.

Revisions

Date	Part number/ Revision	Description
November 2013		Initial release for OneFS 7.1
June 2014		Updated for OneFS 7.1.1
November 2014		Updated for OneFS 7.2
June 2015		Updated for OneFS 7.2.1
November 2015		Updated for OneFS 8.0
September 2016		Updated for OneFS 8.0.1
April 2017		Updated for OneFS 8.1
November 2017		Updated for OneFS 8.1.1
February 2019		Updated for OneFS 8.1.3
April 2019		Updated for OneFS 8.2
August 2019		Updated for OneFS 8.2.1
December 2019		Updated for OneFS 8.2.2
June 2020		Updated for OneFS 9.0

Date	Part number/ Revision	Description
September 2020		Updated for OneFS 9.1
April 2021		Updated for OneFS 9.2
September 2021		Updated for OneFS 9.3
March 2022		Updated for OneFS 9.4
January 2023		Updated for OneFS 9.5
March 2023	H10588.17	In architectural overview, expanded failure domains and resource pools to include discussion of partner node protection and chassis protection
January 2024		Updated for OneFS 9.7
April 2024		Updated for OneFS 9.8

We value your feedback

Dell Technologies and the authors of this document welcome your feedback on this document. Contact the Dell Technologies team by email.

Author: Nick Trimbee

Note: For links to other PowerScale documentation, see the PowerScale Info Hub.

High availability and reliability at the petabyte scale

Introduction

Once datasets grow into the petabyte realm, a whole new level of availability, management, and protection challenges arise. At this magnitude, given the law of large numbers with the sheer quantity of components involved, one or more components will almost always be in a degraded state at any one time within the storage infrastructure. As such, guarding against single points of failure and bottlenecks becomes a critical and highly complex issue. Other challenges that quickly become apparent at the petabyte scale include the following:

- **File System Limitations:** How much capacity and how many files can a file system accommodate?
- **Disaster Recovery:** How do you duplicate the data off site and then how do you retrieve it?
- **Scalability of Tools:** How do you take snapshots of massive datasets?
- **Software Upgrades and Hardware Refresh:** How do you upgrade software and replace outdated hardware with new?
- **Performance Issues:** How long will searches and tree-walks take with large, complex datasets?
- **Backup and Restore:** How do you back up a large dataset and how long will it take to restore?

Given these challenges, the requirement for a new approach to file storage is clear. Fortunately, when done correctly, scale-out NAS can fulfill this need.

Areal density and drive rebuild times

In today's world of large capacity disk drives, the probability that secondary device failures will occur has increased dramatically. Areal density, the amount of written information on the disk's surface in bits per square inch, continues to outstrip Moore's law. However, the reliability and performance of disk drives are not increasing at the same pace, and this is compounded by the growing amount of time it takes to rebuild drives.

Large capacity disks such as 16 TB and 20 TB SATA drives require much longer drive reconstruction times because each subsequent generation of disk still has the same number of heads and actuators servicing increased density platters. This significantly raises the probability of a multiple drive failure scenario.

Silent data corruption

Another threat that needs to be addressed, particularly at scale, is the looming specter of hardware-induced corruption. For example, when CERN tested the data integrity of standard disk drives, they discovered some alarming findings. They built a simple write and verify application that they ran across a pool of 3,000 servers, each with a hardware RAID controller. After five weeks of testing, they found more than 500 instances of silent data corruption spread across 17 percent of the nodes–after having previously thought everything was fine. Under the hood, the hardware RAID controller only detected a handful of the most blatant data errors, and the rest passed unnoticed.

Suffice to say, this illustrates two inherent data protection requirements: First, the need for an effective, end-to-end data verification process to be integral to a storage device to detect and mitigate such instances of silent data corruption. Second, the requirement for regular and reliable backups is the linchpin of a well-founded data protection plan.

Data protection continuum

The availability and protection of data can be usefully illustrated in terms of a continuum:

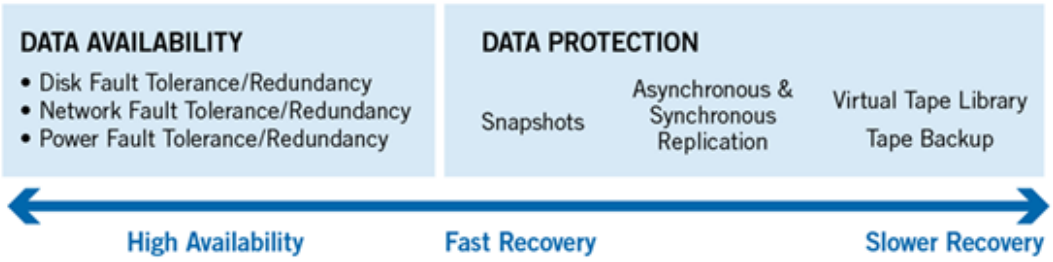

Figure 1. Data protection continuum

At the beginning of the continuum sits high availability. This requirement is usually satisfied by redundancy and fault tolerant designs. The goal here is continuous availability and the avoidance of downtime by the use of redundant components and services.

Further along the continuum lie the data recovery approaches in order of decreasing timeliness. These solutions typically include a form of point-in-time snapshots for fast recovery, followed by synchronous and asynchronous replication. Finally, backup to tape or a virtual tape library sits at the end of the continuum, providing insurance against large-scale data loss, natural disasters, and other catastrophic events.

High availability with OneFS

Introduction

Dell PowerScale NAS solutions, powered by the OneFS operating system, use a holistic approach to ensure that data is consistent and intact—both within the file system and when exiting the cluster over a network interface. Furthermore, the OneFS clustering technology is uncompromisingly designed to simplify the management and protection of multi-petabyte datasets.

Scale-out architecture

A Dell PowerScale cluster is built on a highly redundant and scalable architecture based on the hardware premise of shared nothing. The fundamental building blocks are platform nodes of which there are anywhere from three to two hundred and fifty-two nodes in a cluster. Each of these platform nodes contains CPU, memory, disk, and I/O controllers in an efficient 1U or 4U rack-mountable chassis. Redundant Ethernet or InfiniBand (IB) adapters provide a high-speed back-end cluster interconnect—essentially a distributed system bus—and each node houses mirrored and battery-backed file system journals to protect any uncommitted writes. Except for the LCD control front panel, all node components are standard enterprise commodity hardware.

These platform nodes contain various storage media types and densities, including SAS and SATA hard drives, solid state drives (SSDs), and a configurable quantity of memory. This allows customers to granularly select an appropriate price, performance, and protection point to accommodate the requirements of specific workflows or storage tiers.

Highly available storage client access is provided over multiple 1, 10, 25, or 40 Gb/s Ethernet interface controllers within each node, and across various file and object protocols, including NFS, SMB, S3, and HDFS. OneFS also provides full support for both IPv4 and IPv6 environments across the front-end Ethernet networks.

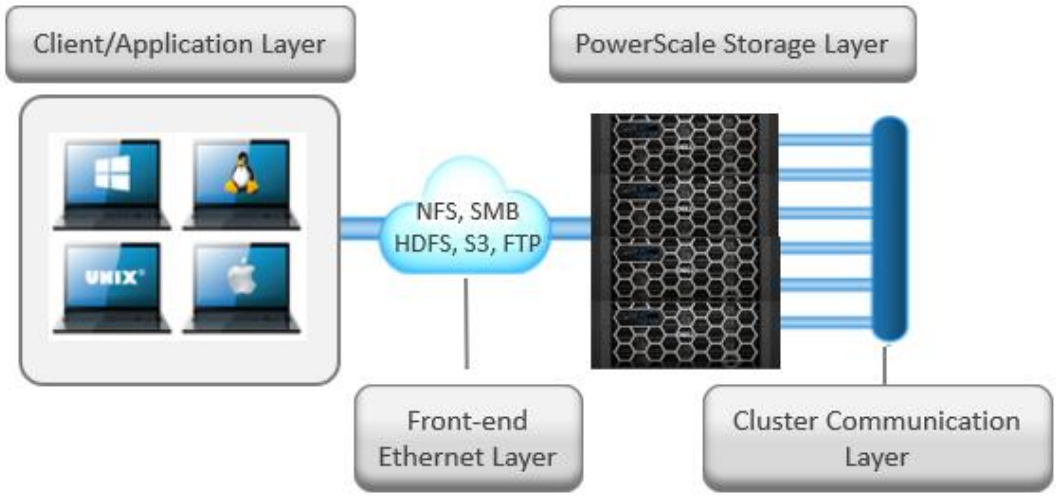

Figure 2. OneFS NAS architecture

Heterogeneous clusters can be architected with a wide variety of node styles and capacities, in order to meet the needs of a varied dataset and wide spectrum of workloads. These node styles fall loosely into four main categories or tiers. The following figure illustrates these tiers, and the associated node models:

Tier	I/O Profile	Drive Media	Nodes
Performance	High Perf, Low Latency	Flash NVMe/SAS	F900 F810 F710 F800 F210 F600 F200
Hybrid / Utility	Concurrency & Streaming Throughput	SATA/SAS & SSD	H700 H600 H7000 H5600 H500 H400
Archive	Nearline & Deep Archive	SATA	A300 A200 A3000 A2000

Figure 3. Hardware tiers and node types

OneFS architectural overview

OneFS collapses the traditional elements of the storage stack—data protection, volume manager, file system—into a single, unified software layer (see Figure 4). This allows for a highly extensible file system that affords unparalleled levels of protection and availability.

Built atop FreeBSD's UNIX implementation, availability and resilience are integral to OneFS from the lowest level on up. For example, unlike BSD, OneFS provides mirrored volumes for the root and /var file systems using the Mirrored Device Driver (IMDD), stored on flash drives. OneFS also automatically saves last known good boot partitions for further resilience.

On the network side, the logical network interface (LNI) framework provides a robust, dynamic abstraction for easily combining and managing differing interfaces, enabling network resilience. Multiple network interfaces can be trunked together with Link Aggregation Control Protocol (LACP) and Link Aggregation and Link Failover (LAGG) to provide bandwidth aggregation in addition to client session failover and general network resilience.

Within the cluster, every disk within each node is assigned both a Globally Unique Identifier (GUID) and logical drive number and is subdivided into 32 MB cylinder groups consisting of 8 KB blocks. Each cylinder group is responsible for tracking, using a bitmap, whether its blocks are used for data, inodes or other metadata constructs. The combination of node number, logical drive number, and block offset consist of a block or inode address and fall under the control of the aptly named Block Allocation Manager (BAM).

In addition to block and inode allocation, the BAM also handles file layout and locking and abstracts the details of OneFS distributed file system from the kernel and user space. The BAM never touches the disk itself, instead delegating tasks to the local and remote block manager elements respectively on the appropriate nodes. The Remote Block Manager (RBM) is essentially a Remote Procedure Call (RPC) protocol that uses either IP over Ethernet or the Socket Direct Protocol (SDP) over redundant Ethernet or InfiniBand for reliable, low-latency back-end cluster communication. These RBM messages—everything from cluster heartbeat pings to distributed locking control—are processed by a node's Local Block Manager using the Device Worker Thread (DWT) framework code.

Figure 4. OneFS collapsed stack storage architecture

For more details about the OneFS architecture, see the *PowerScale OneFS Technical Overview white paper*.

Safe writes

For write operations, where coherency is vital, the BAM first sets up a transaction. Next it uses a 2-phase commit protocol (2PC) over the RBM to guarantee the success of an atomic write operation across all participant nodes. This is managed using the BAM Safe Write (BSW) code path. The 2PC atomically updates multiple disks across the 2PC participant nodes, using their journals for transaction logging. The write path operates as follows:

1. **Client performs a transactional write.**
   Block is written to journal; memory buffer is pinned.
   Rollback data is maintained.
2. **Transaction commits.**
   Journal data is pinned; memory buffer is dirty.
   Rollback data can now be discarded.
   Top-level operation is complete.
3. **OneFS asynchronously flushes dirty buffers to disk at some point.**
   Placed into the writeback cache.
   Journal data is still required, and memory buffer is discarded.
4. Journal approaches full or timeout and issues disk writeback cache flush.
   This occurs relatively infrequently.
5. **Cache flush complete.**
   Journal data is discarded for writes that were returned prior to flush.

Cluster group management

The OneFS Group Management Protocol (GMP) handles cluster coherence and quorum. The challenge is combining the various elements—performance, coherency, client access protocols—across multiple heads. The GMP is built on several distributed algorithms and strictly adheres to Brewer's Theorem, which states that it is impossible for a distributed computer system to simultaneously guarantee consistency, availability, and partition tolerance. OneFS does not compromise on either consistency or availability.

Given this, a quorum group consisting of more than half of a cluster's nodes must be active and responding at any given time. If a node is up and responsive but not a member of the quorum group, it is forced into a read-only state.

OneFS employs this notion of a quorum to prevent “split-brain” conditions that might possibly result from a temporary cluster division. The quorum also dictates the minimum number of nodes required to support a given data protection level. For example, seven or more nodes are needed for a cluster to support a +3n configuration. This allows for a simultaneous loss of three nodes while still maintaining a quorum of four nodes, allowing the cluster to remain operational.

The GMP tracks the state of all the nodes and drives that are considered part of the cluster. Whenever devices are added or removed from the cluster, either proactively or reactively, a group change is broadcast, the group ID is incremented, and any uncommitted journal write transactions are resolved.

For more information about cluster group management, see the *PowerScale OneFS: Cluster Composition, Quorum, and Group State white paper*.

Concurrency and locking

OneFS employs a distributed lock manager that uses a proprietary hashing algorithm to orchestrate coherent locking on data across all nodes in a storage cluster. The design is such that a lock coordinator invariably ends up on a different node than the initiator and either shared or exclusive locks are granted as required. The same distributed lock manager mechanism is used to orchestrate file system structure locks and protocol and advisory locks across the entire cluster. OneFS also supports delegated locks (SMB opportunistic locks and NFSv4 delegations) and also byte-range locks.

File layout

OneFS is a single file system providing one vast, scalable namespace—free from multiple volume concatenations or single points of failure. As such, all nodes access the same structures across the cluster using the same block addresses and all directories are inode number links emanating from the root inode.

The way data is laid out across the nodes and their respective disks in a cluster is fundamental to OneFS functionality. OneFS uses an 8 KB block size, and 16 blocks are combined to create a 128 KB stripe unit. Files are striped across nodes, allowing files to use the resources (spindles and cache) of up to 20 nodes, based on per-file policies.

The layout decisions are made by the BAM on the node that initiated a particular write operation using the 2PC described in Safe writes. The BAM Safe Write (BSW) code takes the cluster group information from GMP and the chosen protection policy for the file. It then uses that information to make an informed decision about where best to write the data blocks to ensure that the file is properly protected. The BSW generates a write plan, which consists of all the steps required to safely write the new data blocks across the protection group. Once the plan is complete, the BSW performs this write plan and guarantees its successful completion.

All files, inodes, and other metadata structures (B-trees) within OneFS are either mirrored up to eight times or erasure code protected, with the data spread across the various disk cylinder groups of multiple nodes. Erasure code protection uses an N+M scheme with N representing the number of nodes—the stripe width—and M the number of error correcting code (ECC) blocks. Flexible protection provides more details.

OneFS will not write files at less than the desired protection level. However, the BAM will attempt to use an equivalent mirrored layout if there is an insufficient stripe width to support a particular forward error correction (FEC) protection level.

Flexible protection

OneFS is designed to withstand multiple simultaneous component failures (currently four) while still affording unfettered access to the entire file system and dataset. Data protection is implemented at the file system level and, as such, is not dependent on any hardware RAID controllers. This provides many benefits, including the ability to add new data protection schemes as market conditions or hardware attributes and characteristics evolve. Since protection is applied at the file-level, a OneFS software upgrade is all that is required in order to make new protection and performance schemes available.

OneFS employs the popular Reed-Solomon erasure coding algorithm for its protection calculations. Protection is applied at the file-level, enabling the cluster to recover data quickly and efficiently. Inodes, directories, and other metadata are protected at the same or higher level as the data blocks they reference. Since all data, metadata and FEC blocks are striped across multiple nodes, there is no requirement for dedicated parity drives. This both guards against single points of failure and bottlenecks and allows file reconstruction to be a highly parallelized process. Today, OneFS provides +1n through +4n protection levels, providing protection against up to four simultaneous component failures respectively. A single failure can be as little as an individual disk or, at the other end of the spectrum, an entire node.

OneFS supports several protection schemes. These include the ubiquitous +2d:1n, which protects against two drive failures or one node failure.

Note: The best practice is to use the recommended protection level for a particular cluster configuration. This recommended level of protection is clearly marked as **suggested** in the OneFS WebUI storage pools configuration pages and is typically configured by default.

The hybrid protection schemes are particularly useful for PowerScale chassis-based nodes and other high-density node configurations, where the probability of multiple drives failing far surpasses that of an entire node failure. In the unlikely event that multiple devices have simultaneously failed, such that the file is “beyond its protection level,” OneFS will reprotect everything possible and report errors on the individual files affected to the cluster’s logs.

OneFS also provides various mirroring options ranging from 2x to 8x, allowing from two to eight mirrors of the specified content. Metadata, for example, is mirrored at one level above FEC by default. For example, if a file is protected at +1n, its associated metadata object will be 3x mirrored.

The following table summarizes the full range of OneFS protection levels:

Table 1. OneFS FEC protection levels

Protection level	Description
+1n	Tolerate failure of 1 drive OR 1 node
+2d:1n	Tolerate failure of 2 drives OR 1 node

Protection level	Description
+2n	Tolerate failure of 2 drives OR 2 nodes
+3d:1n	Tolerate failure of 3 drives OR 1 node
+3d:1n1d	Tolerate failure of 3 drives OR 1 node AND 1 drive
+3n	Tolerate failure of 3 drives or 3 nodes
+4d:1n	Tolerate failure of 4 drives or 1 node
+4d:2n	Tolerate failure of 4 drives or 2 nodes
+4n	Tolerate failure of 4 nodes
2x to 8x	Mirrored over 2 to 8 nodes, depending on configuration

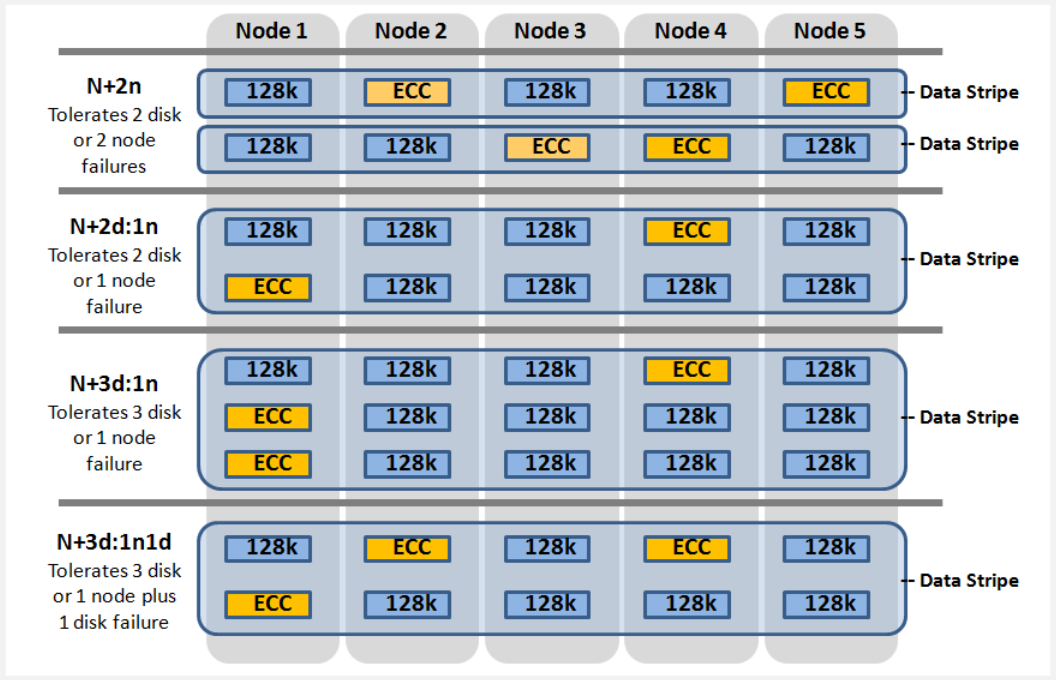

Figure 5. OneFS hybrid protection schemes

Striped, distributed metadata coupled with continuous auto-balancing affords OneFS truly linear performance characteristics, regardless of fullness of file system. Both metadata and file data are spread across the entire cluster, keeping the cluster balanced at all times.

For more details about OneFS data protection levels, see the *PowerScale OneFS Technical Overview white paper*.

Failure domains and resource pools

Data tiering and management in OneFS are handled by SmartPools software. From a data protection point of view, SmartPools facilitates the subdivision of large numbers of high-capacity, homogeneous nodes into smaller, more Mean Time to Data Loss (MTTDL)-friendly disk pools. For example, an 80-node H500 cluster typically runs at +3d:1n1d protection level. However, partitioning it into four, 20-node pools would allow each pool to

run at +2d:1n, thereby lowering the protection overhead and improving data utilization without any net increase in management overhead.

Automatic partitioning

In keeping with the goal of storage management simplicity, OneFS automatically calculates and partitions the cluster into pools of disks or “node pools,” which are optimized for both MTTDL and efficient space utilization. This means that protection-level decisions, such as with the preceding 80-node cluster example, are not left to the customer—unless the customer wants to make those decisions.

With automatic provisioning, every set of equivalent node hardware is automatically divided into node pools consisting of up to forty nodes and six drives per node. These node pools are protected by default at +2d:1n, and multiple pools can then be combined into logical tiers and managed using SmartPools file pool policies. By subdividing a node’s disks into multiple, separately protected pools, nodes are significantly more resilient to multiple disk failures than previously possible.

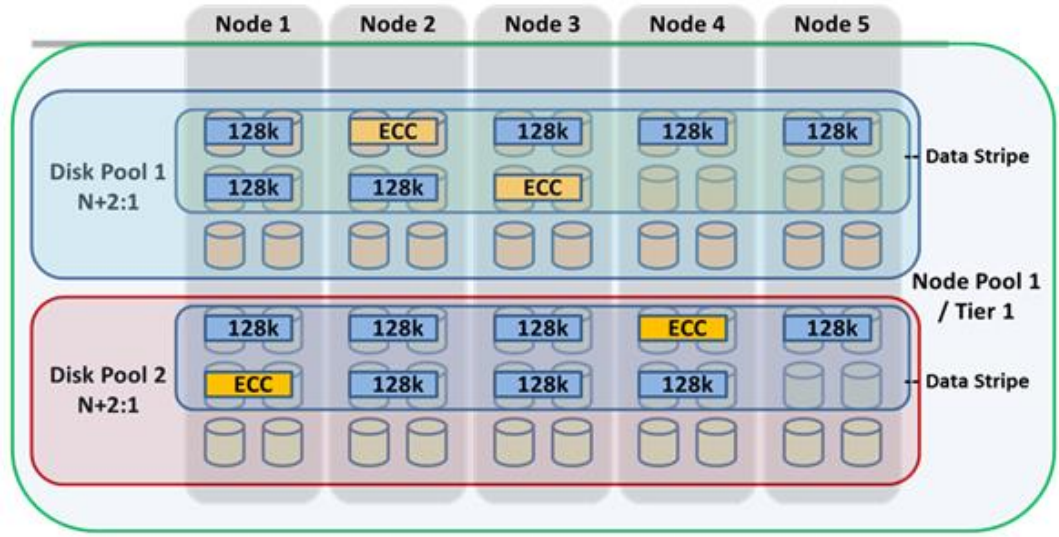

Figure 6. SmartPools automatic provisioning

For more information, see the *Storage Tiering with Dell PowerScale SmartPools white paper*.

The PowerScale modular hardware platforms such as the H700 feature a dense, modular design in which four nodes are contained in a single 4RU chassis. This approach enhances the concept of disk pools, node pools, and “neighborhoods,” which adds another level of resilience into the OneFS failure domain concept. Each PowerScale chassis contains four compute modules (one per node), and five drive containers, or sleds, per node.

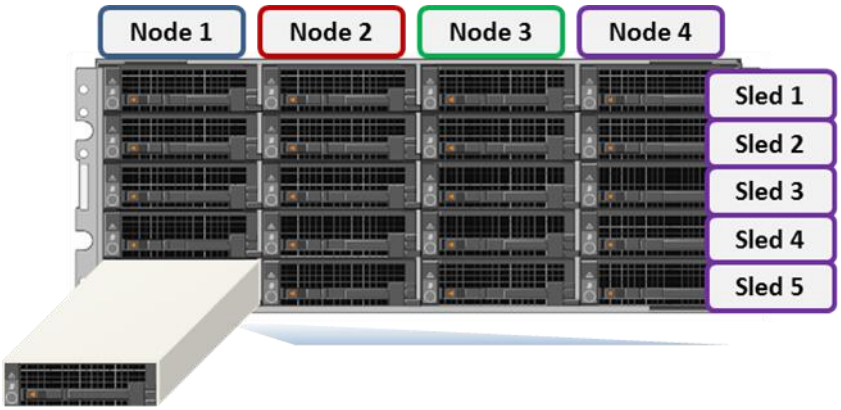

Figure 7. PowerScale chassis front view showing drive sleds

Each sled is a tray which slides into the front of the chassis and contains between three and six drives, depending on the configuration of a particular chassis. Disk Pools are the smallest unit within the Storage Pools hierarchy. OneFS provisioning works on the premise of dividing similar nodes' drives into sets, or disk pools, with each pool representing a separate failure domain. These disk pools are protected by default at +2d:1n (or the ability to withstand two disks or one entire node failure).

Disk pools are laid out across all five sleds in each PowerScale chassis-based node. For example, a node with three drives per sled will have the following disk pool configuration:

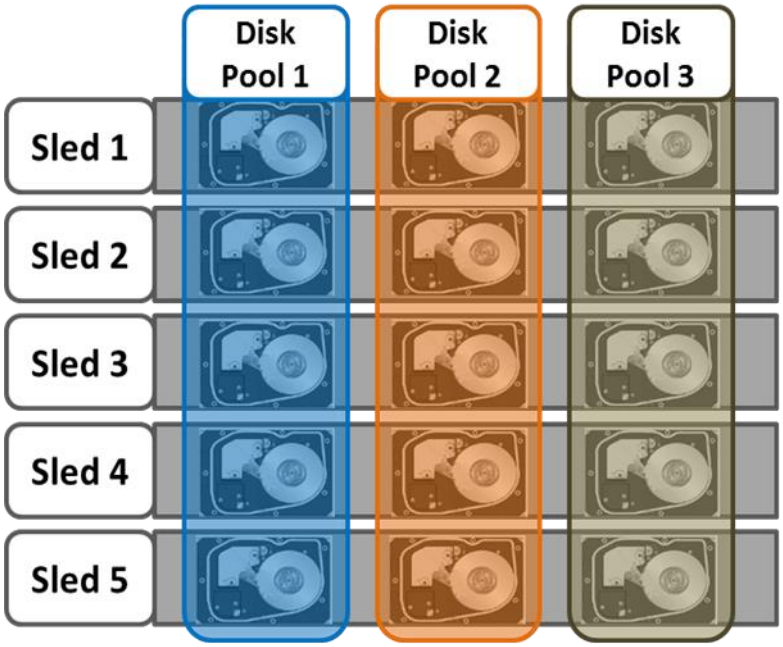

Figure 8. PowerScale chassis-based node disk pools

Node Pools are groups of Disk Pools, spread across similar storage nodes (compatibility classes). Multiple groups of different node types can work together in a single, heterogeneous cluster. For example: one Node Pool of F-Series nodes for IOPS-intensive applications, one Node Pool of H-Series nodes, primarily used for high-concurrent and sequential workloads, and one Node Pool of A-series nodes, primarily used for nearline or deep-archive workloads.

This allows OneFS to present a single storage resource pool consisting of multiple drive media types—SSD, high-speed SAS, large capacity SATA—providing a range of different performance, protection, and capacity characteristics. This heterogeneous storage pool in turn can support a diverse range of applications and workload requirements with a single, unified point of management. It also facilitates the mixing of older and newer hardware, allowing for simple investment protection even across product generations, and seamless hardware refreshes.

Each Node Pool contains only disk pools from the same type of storage nodes, and a disk pool may belong to exactly one node pool. For example, F-Series nodes with 1.6 TB SSD drives would be in one node pool, whereas A-Series nodes with 10 TB SATA drives would be in another. Today, a minimum of four nodes (one chassis) is required per node pool for chassis-based platforms, such as the PowerScale H700, or three nodes per pool for self-contained nodes such as the PowerScale F900.

OneFS “neighborhoods” are fault domains within a node pool, and their purpose is to improve reliability in general and guard against data unavailability from the accidental removal of drive sleds. For self-contained nodes such as the PowerScale F710, OneFS has an ideal size of 20 nodes per node pool and a maximum size of 39 nodes. On the addition of the 40th node, the nodes split into two neighborhoods of 20 nodes.

Neighborhood	F-series Nodes	H-series and A-series Nodes
Smallest Size	3	4
Ideal Size	20	10
Maximum Size	39	19

With the PowerScale chassis-based platforms, such as the H700 and A300, the ideal size of a neighborhood changes from 20 to 10 nodes. This protects against simultaneous node-pair journal failures and full chassis failures.

Partner nodes are nodes whose journals are mirrored. With the chassis-based platform, rather than each node storing its journal in NVRAM, the nodes’ journals are stored on SSDs, and every journal has a mirror copy on another node. The node that contains the mirrored journal is referred to as the partner node. A mirrored journal provides several reliability benefits. For example, SSDs are more persistent and reliable than NVRAM, which requires a charged battery to retain state. Also, with the mirrored journal, both journal drives have to die before a journal is considered lost. As such, unless both of the mirrored journal drives fail, both of the partner nodes can function as normal.

With partner node protection, where possible, nodes are placed in different neighborhoods—and hence different failure domains. Partner node protection is possible once the cluster reaches five full chassis (20 nodes), when, after the first neighborhood split, OneFS places partner nodes in different neighborhoods:

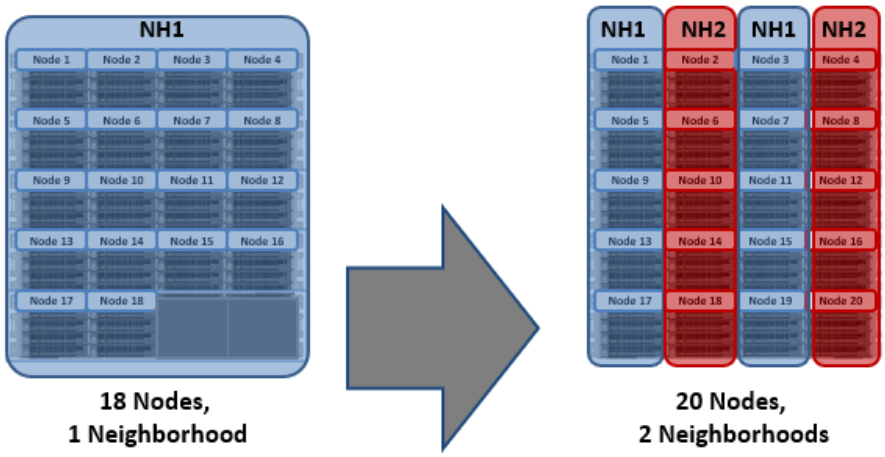

Figure 9. Split into two neighborhoods at 20 nodes

Partner node protection increases reliability. If both nodes go down, they are in different failure domains, so their failure domains only suffer the loss of a single node.

With chassis protection, when possible, each of the four nodes within a chassis is placed in a separate neighborhood. Chassis protection becomes possible at 40 nodes because the neighborhood split at 40 nodes enables every node in a chassis to be placed in a different neighborhood. As such, when a 38-node Gen6 cluster is expanded to 40 nodes, the two existing neighborhoods are split into four 10-node neighborhoods.

Chassis protection ensures that if an entire chassis was to fail, each failure domain would only lose one node.

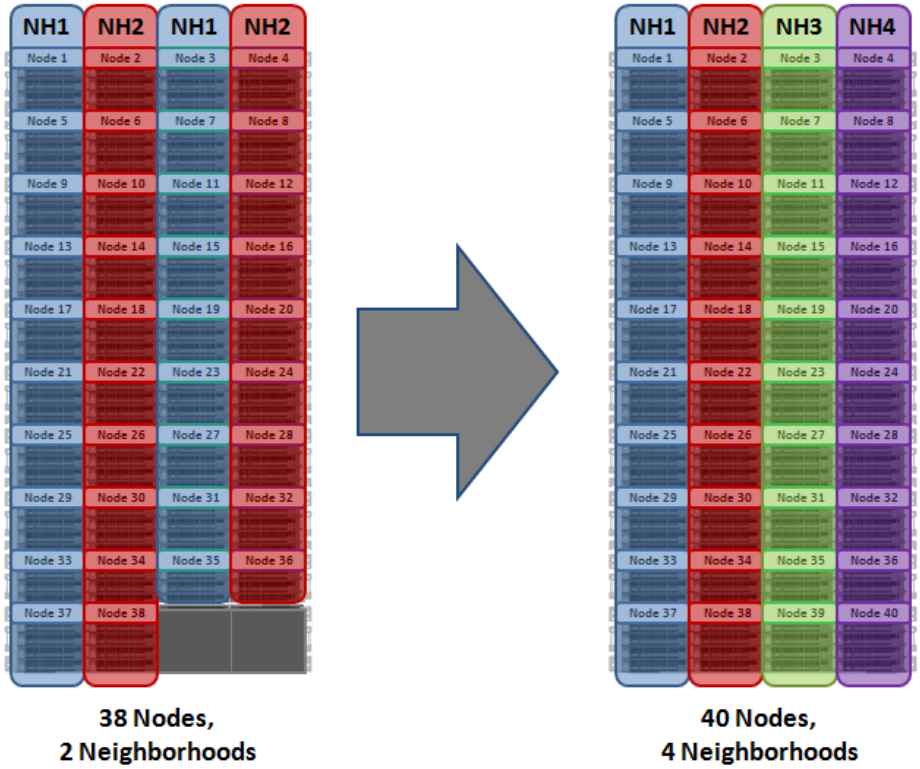

Figure 10. OneFS neighborhoods—four-neighborhood split

A 40-node or larger cluster with four neighborhoods and protected at the default level of +2d:1n can sustain a single node failure per neighborhood. This protects the cluster against a single full-chassis failure.

Overall, a cluster consisting of chassis-based nodes has reliability at least one order of magnitude greater than previous-generation clusters of a similar capacity as a direct result of the following enhancements:

- Mirrored journals
- Smaller neighborhoods
- Mirrored boot drives

Manual node pool management

Once a node pool has been automatically provisioned, additional manual node pools can be created. When complete, the constituent nodes can then be manually reassigned across these node pools, as wanted. Manual node pools require a minimum of three nodes in each pool (four for chassis-based hardware, such as the A300). They are considered an advanced cluster configuration because they can have a significant impact on cluster performance.

Virtual hot spares

SmartPools also provides a virtual hot spare option, if wanted. This functionality allows space to be reserved in a disk pool, equivalent to up to four full drives. This virtual hot spare pool can be immediately used for data reprotection in the event of a drive failure.

From a data availability and management point of view, SmartPools also applies storage tiering concepts to disk pools, allowing the storage and movement of data according to rich file policies or attributes. As such, SmartPools facilitates the automated alignment of data with the appropriate class of storage according to its business value, performance profile, and availability requirements. A cluster can thereby provide multiple storage pools, each supporting a range of availability SLAs within a single, highly scalable, and easily managed file system. This resource pool model aligns well with the current IT trend of private and hybrid cloud initiatives.

For more information, see the *Storage Tiering with Dell PowerScale SmartPools white paper*.

OneFS fault tolerance

File system journal

Each node has a battery backed and mirrored file system journal. OneFS uses each journal as stable storage and guards write transactions against sudden power loss or other catastrophic events. The journal protects the consistency of the file system, and, as such, OneFS is fully journaled.

Protective device failure

OneFS proactively removes, or SmartFails, any drive that reaches a particular threshold of detected Error Correction Code (ECC) errors, and automatically reconstructs the data from that drive and locates it elsewhere on the cluster. Both SmartFail and the

subsequent repair process are fully automated and hence require no administrator intervention.

Data integrity

Isi Data Integrity (IDI) is the OneFS process that protects file system structures against corruption using 32-bit CRC checksums. All OneFS file system blocks, both for file and metadata, use checksum verification. Metadata checksums are housed in the metadata blocks themselves, whereas file data checksums are stored as metadata, providing referential integrity. All checksums are recomputed by the initiator, the node servicing a particular read, on every request.

If the recomputed checksum does not match the stored checksum, OneFS will generate a system alert, log the event, and retrieve and return the corresponding error correcting code (ECC) block to the client and attempt to repair the suspect data block.

Protocol checksums

In addition to blocks and metadata, OneFS also provides checksum verification for Remote Block Management (RBM) protocol data. The RBM is a unicast, RPC-based protocol developed by Dell for use over the back-end cluster interconnect. Checksums on the RBM protocol are in addition to the hardware checksums provided at the network layer. They are used to detect and isolate machines that have certain faulty hardware components and exhibit other failure states.

Dynamic Sector Repair

OneFS includes a Dynamic Sector Repair (DSR) feature whereby the file system can force bad disk sectors to be rewritten elsewhere. When OneFS fails to read a block during normal operation, DSR is invoked to reconstruct the missing data and write it to either a different location on the drive or to another drive on the node. This is done to ensure that subsequent reads of the block do not fail. DSR is fully automated and transparent to the end user. Disk sector errors and Cyclic Redundancy Check (CRC) mismatches use almost the same mechanism as the drive rebuild process.

MediaScan

MediaScan's role within OneFS is to check disk sectors and deploy the DSR mechanism described in Dynamic Sector Repair to force disk drives to fix any sector ECC errors they might encounter. Implemented as one of the phases of the OneFS Job Engine, MediaScan runs automatically based on a predefined schedule. Designed as a low-impact, background process, MediaScan is fully distributed and so can leverage the benefits of the OneFS unique parallel architecture.

IntegrityScan

IntegrityScan, another component of the OneFS Job Engine, is responsible for examining the entire file system for inconsistencies. It does this by systematically reading every block and verifying its associated checksum. Unlike traditional fsck-style file system integrity checking tools, IntegrityScan is designed to run while the cluster is fully operational, removing the need for any downtime. If IntegrityScan detects a checksum mismatch, a system alert is generated and written to the syslog and OneFS automatically attempts to repair the suspect block.

The IntegrityScan phase is run manually if the integrity of the file system is ever in doubt. Although this process may take several days to complete, the file system is online and completely available during this time. Additionally, like all phases of the OneFS Job

Engine, IntegrityScan can be prioritized, paused, or stopped, depending on the impact to cluster operations.

Additional information is available in the *PowerScale OneFS Job Engine white paper*.

Fault isolation

Because OneFS protects its data at the file-level, any inconsistencies or data loss is isolated to the unavailable or failing device—the rest of the file system remains intact and available.

For example, an eight-node PowerScale H7000 cluster protected at +2n:1d sustains three simultaneous drive failures—one in each of three nodes. Even in this degraded state, I/O errors would only occur on the very small subset of data housed on all three of these drives. The remainder of the data striped across the other 157 drives would be unaffected. Contrast this behavior with a traditional RAID 6 system, where losing more than two drives in a RAID set renders it unusable and necessitates a full restore from backups.

Similarly, in the unlikely event that a portion of the file system does become corrupt (whether as a result of a software or firmware issue) or a media error occurs where a section of the disk has failed, only the portion of the file system associated with this area on disk would be affected. All healthy areas would still be available and protected.

Referential checksums of both data and metadata are used to catch silent data corruption (data corruption not associated with hardware failures). The checksums for file data blocks are stored as metadata outside the blocks that they reference, and thus provide referential integrity.

Accelerated drive rebuilds

The time that it takes a storage system to rebuild data from a failed disk drive is crucial to the data reliability of that system. With the advent of multi-terabyte drives, and the creation of increasingly larger single volumes and file systems, typical recovery times for large, multi-terabyte SATA drive failures are becoming multiple days or even weeks. During this Mean Time to Data Loss (MTTDL) period, storage systems are vulnerable to additional drive failures and the resulting data loss and downtime.

Because OneFS is built upon a highly distributed architecture, it can use the CPU, memory, and spindles from multiple nodes to reconstruct data from failed drives in a highly parallel and efficient manner. Because a cluster is not bound by the speed of any particular drive, OneFS can recover from drive failures extremely quickly, and this efficiency grows relative to cluster size. As such, a failed drive within a cluster is rebuilt an order of magnitude faster than hardware RAID-based storage devices. Also, OneFS has no requirement for dedicated hot spare drives.

Because OneFS protects each file individually through FEC erasure encoding, the wall clock time required to reprotect data when a drive fails depends upon:

- The number of and size of files that have some footprint on the drive
- The number of files in the cluster (because OneFS scans each file to determine whether any part of it is on the failed drive)
- The number of nodes in the cluster
- How full the drives are within the disk pool in which the drive failed
- The load on the cluster

While difficult to predict with high accuracy, OneFS strives to ensure that the drive rebuild rates remain above an MTTDL of at least 5,000 years or higher, depending on the configured protection level.

Most of the reprotection job's total run time is usually spent scanning all files on the cluster to determine which require repairing. On most clusters, because only a small fraction of the files will need attention, the size of those files and the disk pool's I/O performance become significant.

For flash drives there is a negligible difference between TLC and QLC media, or between drives sizes, because the drives themselves are not the bottleneck. This means that the reprotection time for flash drives (SSDs) of any type or capacity is fairly uniform. It can be expressed as:

Data size on the drive / 1,500 GiB/hr (1.46 TiB/hr)

where 1,500 GiB/hr is the lower bound (as seen in internal testing).

Automatic drive firmware updates

A cluster supports automatic drive firmware updates for new and replacement drives, as part of the nondisruptive firmware update process. Firmware updates are delivered using drive support packages, which both simplify and streamline the management of existing and new drives across the cluster. This ensures that drive firmware is up to date and mitigates the likelihood of failures due to known drive issues. As such, automatic drive firmware updates are an important component of OneFS high availability and nondisruptive operations strategy. Drive and node firmware can be applied as either a rolling upgrade or by a full cluster reboot.

Before OneFS 8.2, node firmware updates had to be installed one node at a time, which was a time-consuming operation especially in large clusters. Node firmware updates can now be choreographed across a cluster by providing a list of nodes to be simultaneously updated. The upgrade helper tool can be used to select a combination of nodes that can be updated simultaneously and an explicit list of nodes that should not be updated together (for example, nodes in a node-pair).

Rolling upgrade

A rolling upgrade individually upgrades and restarts each node in the cluster sequentially. During a rolling upgrade, the cluster remains online and continues serving data to clients with no interruption in service. Before OneFS 8.0, a rolling upgrade could only be performed within a OneFS code version family and not between OneFS major code version revisions. Beginning with OneFS 8.0, every new release is rolling-upgradable from the previous version. The OneFS upgrade image is cryptographically signed, allowing independent verification that software being installed has not been tampered with.

Nondisruptive upgrades

Nondisruptive upgrades (NDUs) allow a cluster administrator to upgrade the storage operating system while their end users continue to access data without error or interruption. Updating the operating system on a cluster is a simple matter of a rolling upgrade. During this process, one node at a time is upgraded to the new code, and the active NFS and SMB3 clients attached to it are automatically migrated to other nodes in the cluster. Partial upgrade is also permitted, whereby a subset of cluster nodes can be upgraded. The subset of nodes may also be grown during the upgrade. Beginning with

OneFS 8.2, an upgrade can be paused and resumed, allowing customers to span upgrades over multiple smaller maintenance windows.

Parallel upgrades

OneFS 8.2.2 and later versions offer parallel upgrades, whereby clusters can upgrade an entire neighborhood, or fault domain, at a time, substantially reducing the duration of large cluster upgrades. Also, OneFS 9.2 and later versions combine operating system and firmware upgrades, substantially reducing the impact and duration of upgrades by allowing them to occur in tandem. Version 9.2 and later also include drain-based upgrades, whereby nodes are prevented from rebooting or restarting protocol services until all SMB clients have disconnected from the node.

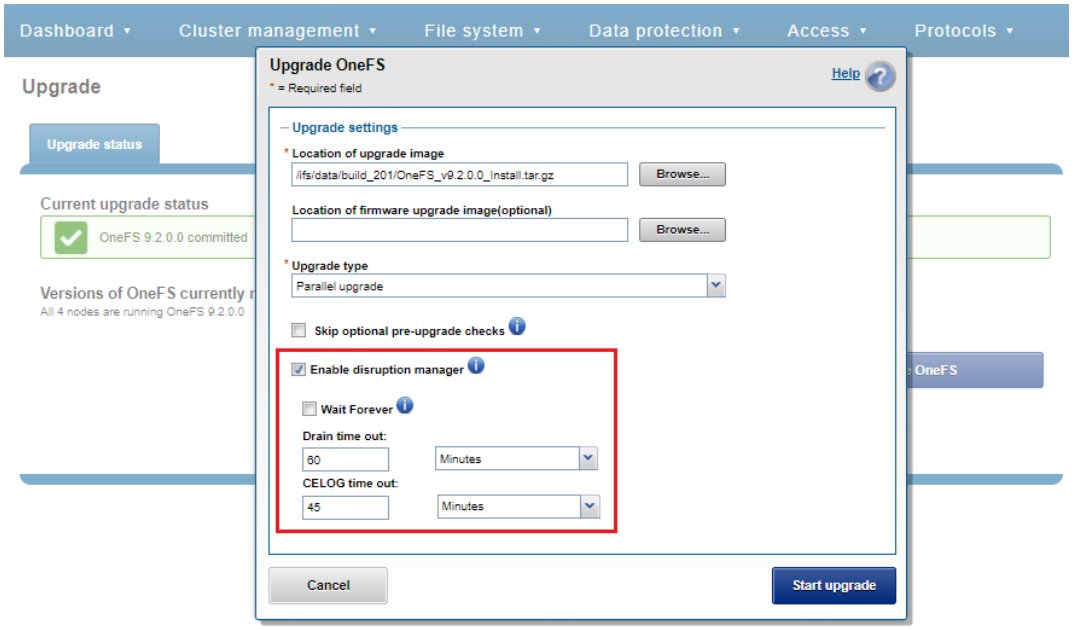

Figure 11. OneFS parallel upgrade disruption manager configuration

SMB client connection draining can also be managed at the node level during an upgrade through the WebUI **Upgrade status** page.

Rollback capable

Upgrade rollback provides the ability to return a cluster with an uncommitted upgrade to its previous version of OneFS.

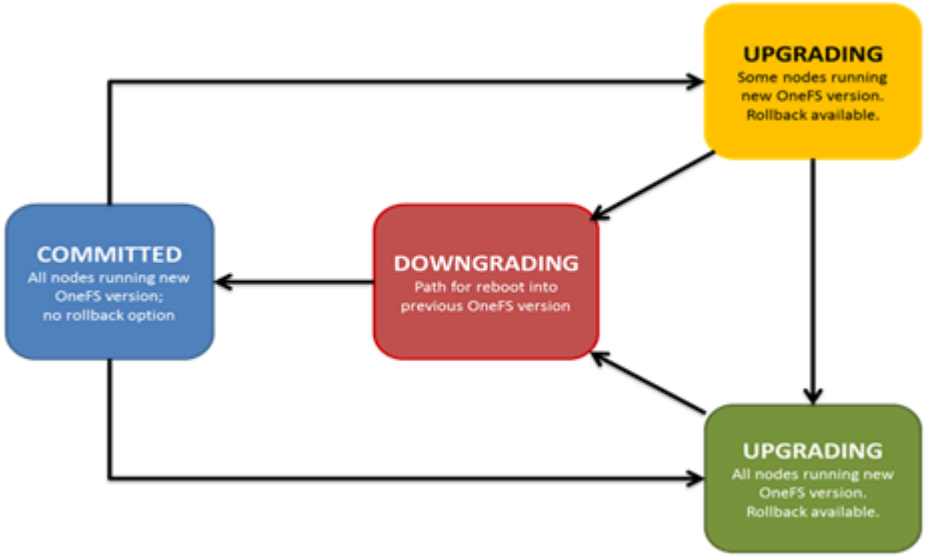

Figure 12. OneFS nondisruptive upgrade states

Performing the upgrade

As part of an upgrade, OneFS automatically runs a pre-installation verification check. This process verifies that the configuration in the currently installed version of OneFS is compatible with the OneFS version that is intended for the upgrade. When an unsupported configuration is found, the upgrade is stopped and instructions on troubleshooting the issue are displayed. Proactively running the pre-installation upgrade check before starting an upgrade helps to avoid any interruption due to incompatible configuration.

Data protection

To effectively protect a file system that is hundreds of terabytes or petabytes in size requires an extensive use of multiple data availability and data protection technologies. The demand for storage is continuing to grow exponentially, and all predictions suggest that it will continue to expand at an aggressive rate for the foreseeable future.

In tandem with this trend, the demand for ways to protect and manage that storage also increases. Today, several strategies for data protection are available and in use. As previously discussed, if data protection is perceived as a continuum, at the beginning lies high availability. Without high availability technologies such as drive, network and power redundancy, data loss and its subsequent recovery would be considerably more prevalent.

Historically, data protection was always synonymous with tape backup. However, over the past decade, several technologies, such as replication, synchronization, and snapshots, in addition to disk-based backup (such as nearline storage and VTL), have become mainstream and established their place within the data protection realm. Snapshots offer rapid, user-driven restores without the need for administrative assistance, while synchronization and replication provide valuable tools for business continuity and offsite disaster recovery.

High availability and data protection strategies

Overview

At the core of every effective data protection strategy lies a solid business continuity plan. All enterprises need an explicitly defined and routinely tested plan to minimize the potential impact to the workflow when a failure occurs or in the event of a natural disaster. There are several ways to address data protection and most enterprises adopt a combination of these methods, to varying degrees.

Among the primary approaches to data protection are fault tolerance, redundancy, snapshots, replication (local, geographically separate, or both), and backups to nearline storage, VTL, or tape.

Some of these methods are biased towards cost efficiency but have a higher risk associated with them, and others represent a higher cost but also offer an increased level of protection. Two ways to measure cost compared to risk from a data protection point of view are:

- **Recovery Time Objective (RTO):** RTO is the allotted amount of time within a Service Level Agreement (SLA) to recover data. For example, an RTO of four hours means data must be restored and made available within four hours of an outage.
- **Recovery Point Objective (RPO):** RPO is the acceptable amount of data loss that can be tolerated per an SLA. An RPO of 30 minutes means that 30 minutes is the maximum amount of time that can elapse since the last backup or snapshot was taken.

OneFS high availability and data protection suite

Data Protection: Described in detail earlier, at the heart of OneFS is FlexProtect. This unique, software-based data protection scheme allows differing levels of protection to be applied in real time down to a per-file granularity, for the entire file system, or at any level in between.

Redundancy: As we have seen, the Dell PowerScale clustered architecture is designed from the ground-up to support the following availability goals:

- No single point of failure
- Unparalleled levels of data protection in the industry
- Tolerance for multi-failure scenarios
- Fully distributed single file system
- Proactive failure detection and preemptive, fast drive rebuilds
- Flexible data protection
- Fully journaled file system
- High transient availability
- Nondisruptive upgrades and operations

The following diagram illustrates how the core components of the PowerScale data protection portfolio align with the notion of an availability and protection continuum and associated recovery objectives.

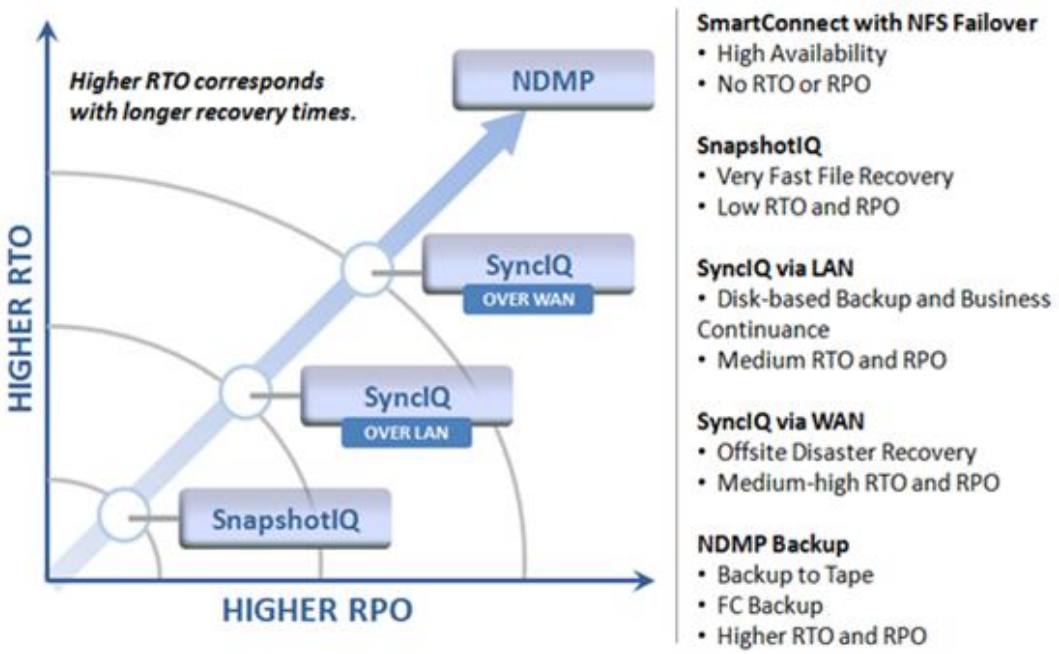

Figure 13. OneFS data protection technology alignment with protection continuum

Connection load balancing and failover—SmartConnect

High availability is at the leading edge of the data protection continuum. This not only includes disk, CPU, and power redundancy, but also network resilience. SmartConnect software contributes to data availability by supporting dynamic NFS failover and failback for Linux and UNIX clients and SMB3 continuous availability for Windows clients. This ensures that when a node failure occurs or preventative maintenance is performed, all in-flight reads and writes are handed off to another node in the cluster to finish its operation without any user or application interruption.

During failover, clients are evenly redistributed across all remaining nodes in the cluster, ensuring minimal performance impact. If a node is brought down for any reason, including a failure, the virtual IP addresses on that node is seamlessly migrated to another node in the cluster. When the offline node is brought back online, SmartConnect automatically rebalances the NFS and SMB3 clients across the entire cluster to ensure maximum storage and performance utilization. For periodic system maintenance and software updates, this functionality allows for per-node rolling upgrades affording full-availability throughout the duration of the maintenance window.

To further increase the protection and security of in-flight data, OneFS supports encryption for the SMBv3 protocol version. This can be configured on a per-share, zone, or cluster-wide basis. Only operating systems that support SMB3 encryption can work with encrypted shares. These operating systems can also work with unencrypted shares if the cluster is configured to allow nonencrypted connections. Other operating systems can access non-encrypted shares only if the cluster is configured to allow nonencrypted connections.

OneFS also supports HDFS transparent data encryption (TDE) for Apache, Cloudera, and Hortonworks Hadoop stacks. Encryption is performed on the client side and can use an external key management server. Data written to and read from these HDFS encryption zones can only be accessed over the HDFS protocol.

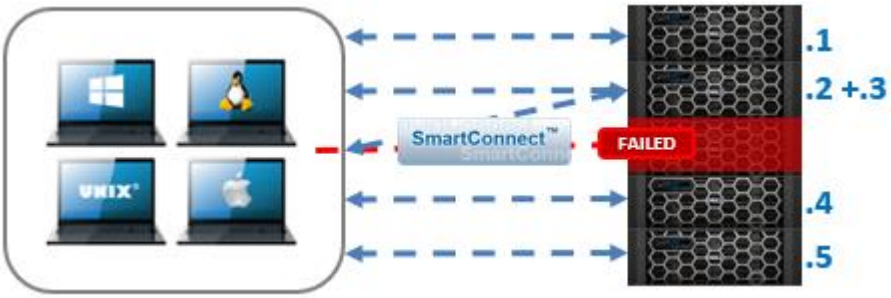

Figure 14. Seamless client failover with SmartConnect

Additional information is available in the OneFS SmartConnect white paper.

Snapshots

SnapshotIQ

Next along the high availability and data protection continuum are snapshots. The RTO of a snapshot can be small, and the RPO is also highly flexible with the use of rich policies and schedules. SnapshotIQ software can take read-only, point-in-time copies of any directory or subdirectory within OneFS.

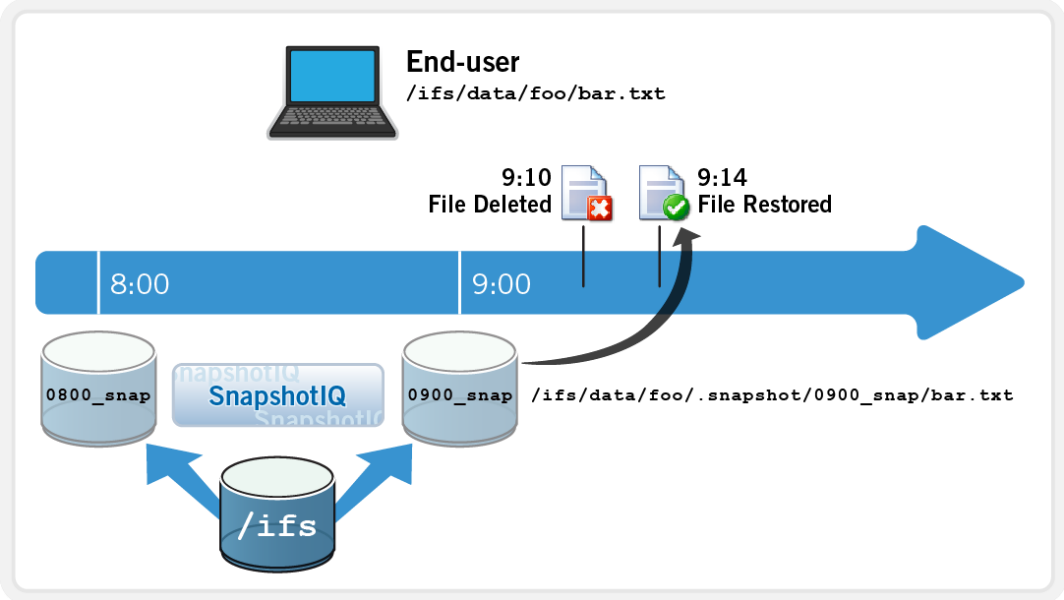

Figure 15. User-driven file recovery with SnapshotIQ

OneFS Snapshots are highly scalable and typically take less than one second to create. They create little performance overhead, regardless of the level of activity of the file system, the size of the file system, or the size of the directory being copied. Also, only the changed blocks of a file are stored when updating the snapshots, thereby ensuring highly efficient snapshot storage utilization. User access to the available snapshots is through a /.snapshot hidden directory under each file system directory.

SnapshotIQ can also create unlimited snapshots on a cluster. This provides a substantial benefit over most other snapshot implementations because the snapshot intervals can be far more granular and so offer improved RPOs.

Snapshot architecture

SnapshotIQ has several fundamental differences as compared to most snapshot implementations. The most significant of these are, first, that OneFS snapshots are per-directory based. This is in contrast to the traditional approach, where snapshots are taken at a file system or volume boundary. Second, since OneFS manages and protects data at the file-level, there is no inherent, block-level indirection layer for snapshots to use. Instead, OneFS takes copies of files, or pieces of files (logical blocks and inodes) in a logical snapshot process.

The process of taking a snapshot in OneFS is relatively instantaneous. However, there is a small amount of snapshot preparation work that has to occur. First, the coalescer is paused and any existing write caches flushed in order for the file system to be quiesced for a short time. Next, a marker is placed at the top-level directory inode for a particular snapshot and a unique snapshot ID is assigned. Once this has occurred, the coalescer resumes and writes continue as normal. Therefore, the moment a snapshot is taken, it essentially consumes zero space until file creates, delete, modifies, and truncates start occurring in the structure underneath the marked top-level directory.

Any changes to a dataset are then recorded in the pertinent snapshot inodes, which contain only referral (“ditto”) records, until any of the logical blocks they reference are altered, or another snapshot is taken. To reconstruct data from a particular snapshot, OneFS iterates though all the more recent versions snapshot tracking files (STFs) until it reaches HEAD (current version). In so doing, it will systematically find all the changes and “paint” the point-in-time view of that dataset.

OneFS uses both Copy on Write (CoW) and Redirect on Write (RoW) strategies for its differential snapshots and uses the most appropriate method for any given situation. Both have advantages and disadvantages, and OneFS dynamically picks the method that will maximize performance and keep overhead to a minimum. Typically, CoW is most prevalent and is primarily used for small changes, inodes, and directories. RoW is adopted for more substantial changes such as deletes and large sequential writes.

OneFS does not require reserved space for snapshots. Snapshots can use as much or little of the available file system space as desirable. A snapshot reserve can be configured if preferred, although this will be an accounting reservation rather than a hard limit. Also, when SmartPools is used, snapshots can be stored on a different disk tier than the one that the original data resides on. For example, the snapshots taken on a performance-aligned tier can be physically housed on a more cost-effective archive tier.

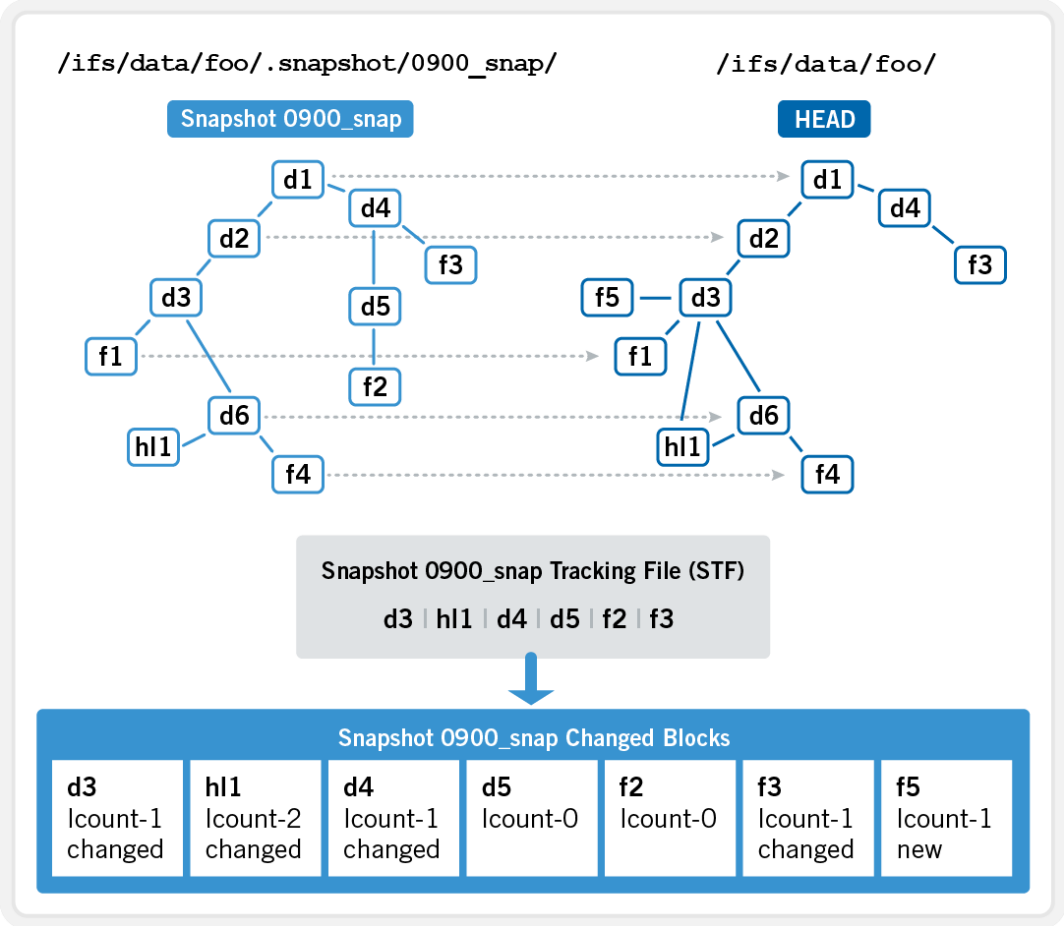

Figure 16. Snapshot change tracking

Snapshot scheduling

Snapshot schedules are configured at a daily, weekly, monthly, or yearly interval, with single or multiple job frequencies per schedule, down to a per-minute granularity. Automatic deletion can be configured per defined schedule at an hourly through yearly range.

Snapshot deletes

When snapshots are manually deleted, OneFS will mark the appropriate snapshot IDs and queue a Job Engine job to affect their removal. The SnapshotDelete job is queued immediately, but the Job Engine typically waits a minute or so to actually start running it. During this interval, the snapshot is marked as **delete pending**.

A similar procedure occurs with expired snapshots. Here, the snapshot daemon is responsible for checking expiration of snapshots and marking them for deletion. The daemon performs the check every 10 seconds. The job is then queued to delete a snapshot completely, and then it is up to the Job Engine to schedule it. The job might run immediately (after a minute or so) if the Job Engine determines that the job is runnable and there are no other jobs with higher priority running. For SnapshotDelete, the job is only run if the group is in a pristine state (no drives or nodes are down).

The most efficient method for deleting multiple snapshots simultaneously is to process older through newer, and SnapshotIQ will automatically attempt to orchestrate deletes in this manner. A SnapshotDelete Job Engine schedule can also be defined so snapshot deletes only occur during selected times.

In summary, SnapshotIQ affords the following benefits:

- Snapshots are created at the directory-level instead of the volume-level, providing improved granularity.
- There is no requirement for reserved space for snapshots in OneFS. Snapshots can use as much or little of the available file system space as desirable.
- Integration with Windows Volume Snapshot Manager allows Windows clients a method to restore from “Previous Versions.”
- Snapshots are easily managed using flexible policies and schedules.
- Using SmartPools, snapshots can physically reside on a different disk tier than the original data.
- Up to 1,024 snapshots can be created per directory.
- The default snapshot limit is 20,000 per cluster.

Snapshot restore

For simple, efficient snapshot restoration, SnapshotIQ provides SnapRevert functionality. Using the Job Engine for scheduling, a SnapRevert job automates the restoration of an entire snapshot to its top-level directory. This is invaluable for quickly and efficiently reverting to a previous, known-good recovery point, for example, if there is a virus or malware outbreak. Also, individual files, rather than entire snapshots, can also be restored in place using FileRevert functionality. This can help drastically simplify virtual machine management and recovery.

For more information, see the *Data Protection with Dell PowerScale Snapshot IQ white paper*.

File clones

OneFS File Clones provides a rapid, efficient method for provisioning multiple read/write copies of files. Common blocks are shared between the original file and clone, providing space efficiency and offering similar performance and protection levels across both. This mechanism is ideal for the rapid provisioning and protection of virtual machine files and is integrated with VMware's linked cloning and block and file storage APIs. This uses the OneFS shadow store metadata structure, which can reference physical blocks, references to physical blocks, and nested references to physical blocks.

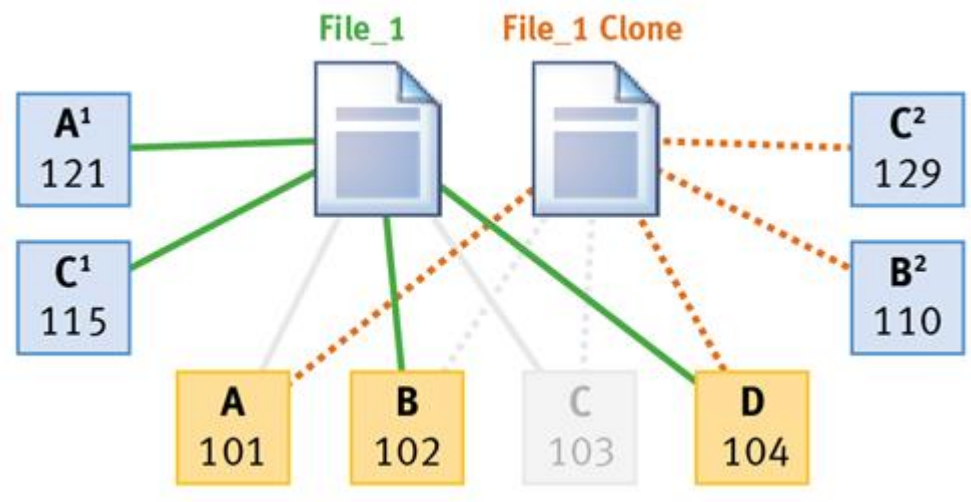

Figure 17. File clones

Writable snapshots

Introduced in OneFS 9.3, writable snapshots enable the creation and management of a space-efficient and time-efficient, modifiable copy of a regular OneFS snapshot. As such, they present a writable copy of a source snapshot, accessible at a directory path within the /ifs namespace. The snapshot can be accessed and edited through the use of any of the cluster's file and object protocols, including NFS, SMB, and S3.

The writable snapshot architecture provides an overlay to a read-only source snapshot. This architecture allows a cluster administrator to create a lightweight copy of a production dataset using a simple CLI command, and present and use it as a separate writable namespace.

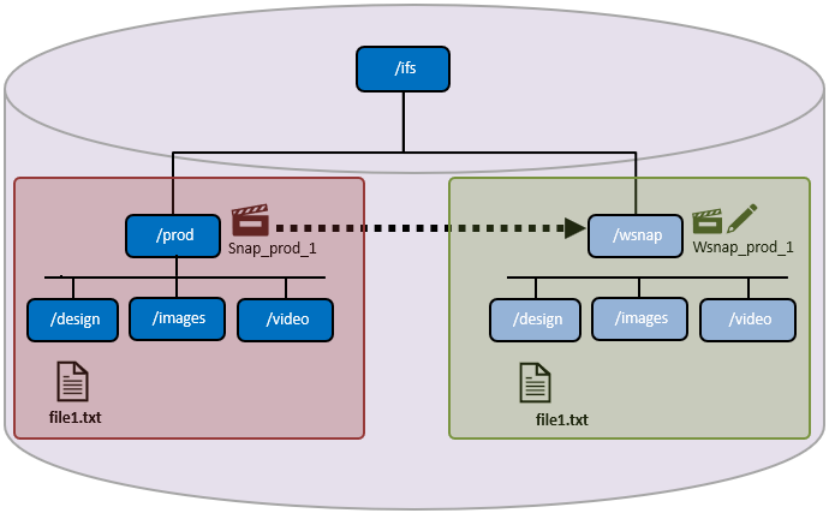

Figure 18. Writable snapshots

A writable snapshot contains the same subdirectory and file structure as the original `prod` directory, but without the added data capacity footprint.

OneFS 9.3 introduced a new protection group data structure, PG_WSNAP, which provides an overlay that allows unmodified file data to be read directly from the source

snapshot while storing only the changes in the writable snapshot tree. When files within a newly created writable snapshot are first accessed, data is read from the source snapshot, populating the files' metadata, in a process known as copy-on-read, or CoR. Unmodified data is read from the source snapshot and any changes are stored in the writable snapshot's namespace data structure (PG_WSNAP).

Since a new writable snapshot is not copy-on-read up front, its creation is extremely rapid. As files are later accessed, they are enumerated and begin to consume metadata space.

SmartDedupe

Shadow stores provide the basis for SmartDedupe, which maximizes the storage efficiency of a cluster by decreasing the amount of physical storage required to house an organization's data. Efficiency is achieved by scanning the on-disk data for identical blocks and then eliminating the duplicates. This means that initial file write or modify performance is not impacted, since no additional computation is required in the write path.

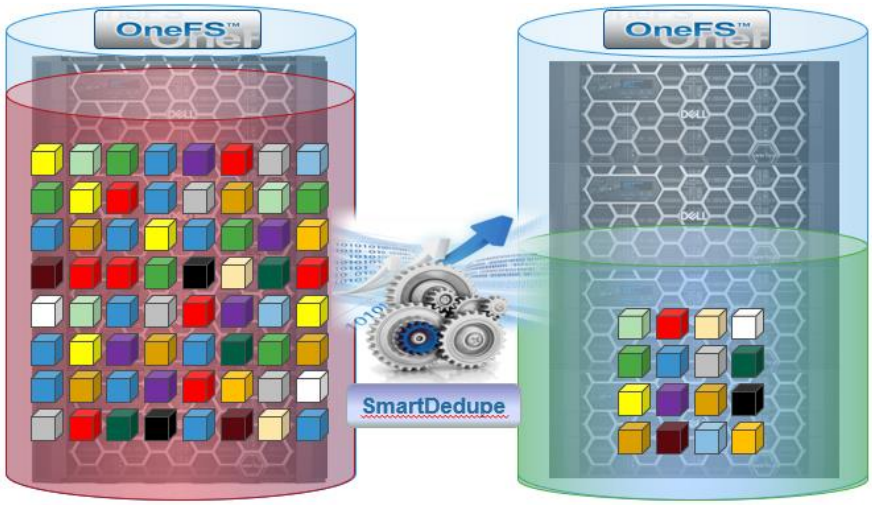

Figure 19. OneFS storage efficiency with SmartDedupe

When SmartDedupe runs for the first time, it scans the dataset and selectively samples data blocks from it, creating the fingerprint index. The index is scanned for duplicates. When a match is found, a byte-by-byte comparison of the blocks is performed to verify that they are identical and to ensure that there are no hash collisions. Then, if the blocks are determined to be identical, duplicate blocks are removed from the actual files and replaced with pointers to the shadow stores.

For more information, see the *Dell PowerScale OneFS: Data Reduction and Storage Efficiency white paper*.

Small File Storage Efficiency

Another principal consumer of OneFS shadow stores is Small File Storage Efficiency. This feature maximizes the space utilization of a cluster by decreasing the amount of physical storage required to house the small files that often consist of an archive dataset, such as found in healthcare PACS workflows.

Efficiency is achieved by scanning the on-disk data for small files, which are protected by full copy mirrors, and packing them in shadow stores. These shadow stores are then parity protected, rather than mirrored, and typically provide storage efficiency of 80 percent or greater.

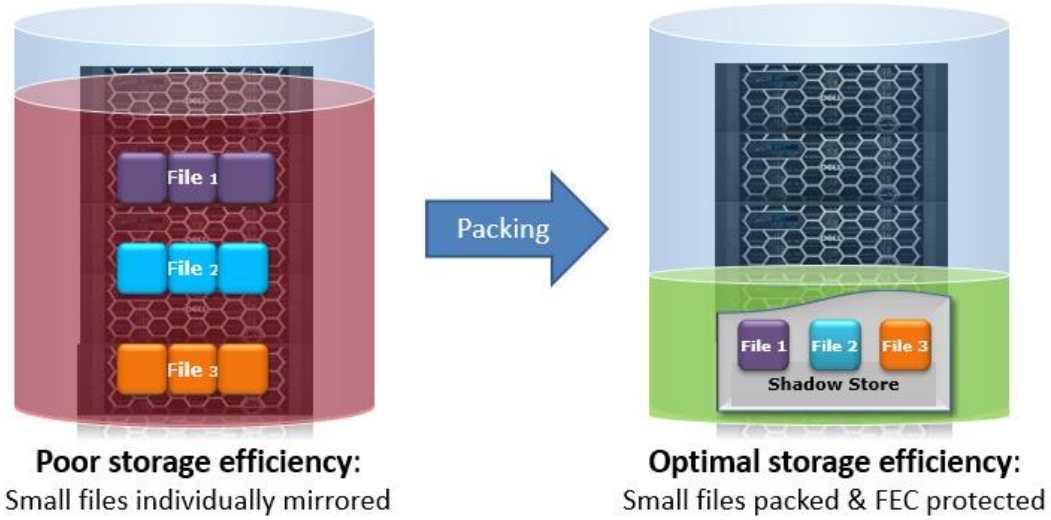

Figure 20. Small file containerization

Small File Storage Efficiency trades a small read latency performance penalty for improved storage utilization. The archived files obviously remain writable, but when containerized files with shadow references are deleted, truncated, or overwritten it can leave unreferenced blocks in shadow stores. These blocks are later freed and can result in holes, which reduces the storage efficiency.

The efficiency loss depends on the protection level layout used by the shadow store. Smaller protection group sizes are more susceptible, as are containerized files, since all the blocks in containers have at most one referring file and the packed sizes (file size) are small.

A shadow store defragmenter is integrated into the ShadowStoreDelete job to help reduce the fragmentation of files as a result of overwrites and deletes. The defragmentation process works by dividing each containerized file into logical chunks (~32 MB each) and assessing each chunk for fragmentation.

If the storage efficiency of a fragmented chunk is below target, that chunk is processed by evacuating the data to another location. The default target efficiency is 90 percent of the maximum storage efficiency available with the protection level used by the shadow store. Larger protection group sizes can tolerate a higher level of fragmentation before the storage efficiency drops below this threshold.

Inline data reduction

OneFS inline data reduction is available on the all-flash F900, F810, F710, F600, F210, F200 nodes, the hybrid H700/7000 and H5600 chassis, and the archive A300/3000 platforms. The architecture consists of the following principal components:

- Data Reduction Platform
- Compression Engine and Chunk Map
- Zero block removal phase
- Deduplication In-memory Index and Shadow Store Infrastructure
- Data Reduction Alerting and Reporting Framework
- Data Reduction Control Path

The inline data reduction write path consists of three main phases:

- Zero Block Removal
- In-line Deduplication
- In-line Compression

If both inline compression and deduplication are enabled on a cluster, zero block removal is performed first, followed by deduplication, and then compression. This order allows each phase to reduce the scope of work each subsequent phase has to perform.

Figure 21. Inline data reduction workflow

The F810 is different from the other inline data reduction supporting nodes in that it includes a hardware compression offload capability, with each node containing a Mellanox Innova-2 Flex Adapter. The Mellanox adapter transparently performs compression and decompression with minimal latency, avoiding the need for consuming a node's expensive CPU and memory resources.

The OneFS hardware compression engine uses zlib, with a software implementation of igzip as fallback if there is a compression hardware failure. OneFS employs a compression chunk size of 128 KB, with each chunk consisting of sixteen 8 KB data blocks. This is optimal because it is also the same size that OneFS uses for its data protection stripe units, providing simplicity and efficiency by avoiding the overhead of additional chunk packing.

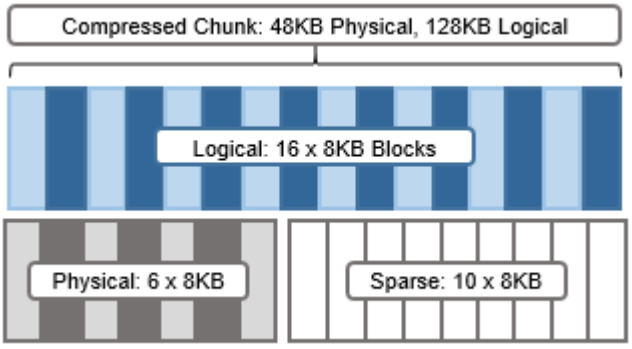

Figure 22. Compression chunks and OneFS transparent overlay

After compression, this chunk is reduced from sixteen to six 8 KB blocks in size. This means that this chunk is now physically 48 KB in size. OneFS provides a transparent logical overlay to the physical attributes. This overlay describes whether the backing data is compressed and which blocks in the chunk are physical or sparse, such that file system consumers are unaffected by compression. As such, the compressed chunk is logically represented as 128 KB in size, regardless of its actual physical size.

Efficiency savings must be at least 8 KB (one block) in order for compression to occur, otherwise that chunk or file will be passed over and remain in its original, uncompressed

state. For example, a file of 16 KB that yields 8 KB (one block) of savings would be compressed. Once a file has been compressed, it is then FEC protected.

Compression chunks will never cross node pools. This avoids the need to decompress or recompress data to change protection levels, perform recovered writes, or otherwise shift protection-group boundaries.

Replication

SyncIQ

While snapshots provide an ideal solution for infrequent or smaller-scale data loss occurrences, when it comes to catastrophic failures or natural disasters, a second, geographically separate copy of a dataset is clearly beneficial. Here, a solution is required that is significantly faster and less error-prone than a recovery from tape, yet still protects the data from localized failure.

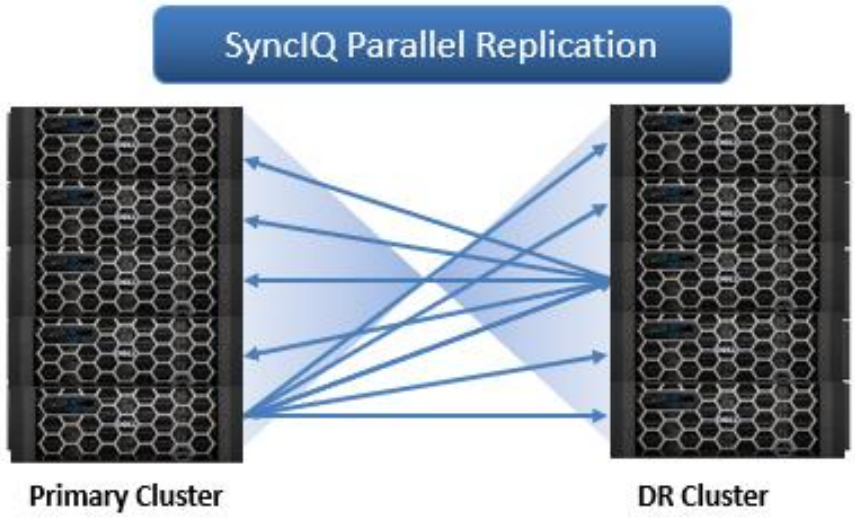

Figure 23. SyncIQ parallel replication across a source and target cluster.

OneFS SyncIQ software delivers high-performance, asynchronous replication of unstructured data to address a broad range of recovery point objectives (RPO) and recovery time objectives (RTO). This enables customers to make an optimal tradeoff between infrastructure cost and potential for data loss if a disaster occurs. SyncIQ does not impose a hard limit on the size of a replicated file system so will scale linearly with an organization's data growth up into the multiple petabyte ranges.

SyncIQ is easily optimized for either LAN or WAN connectivity and includes policy level bandwidth control and reservation. This allows replication over both short and long distances, while meeting consistent SLAs and providing protection from both site-specific and regional disasters. Also, SyncIQ uses a highly parallel, policy-based replication architecture designed to leverage the performance and efficiency of clustered storage. As such, aggregate throughput scales with capacity and allows a consistent RPO over expanding datasets.

There are two basic implementations of SyncIQ:

- The first uses SyncIQ to replicate to a local target cluster within a data center. The primary use case in this scenario is disk backup and business continuity.

- The second implementation uses SyncIQ to replicate to a remote target cluster, typically in a geographically separate data center across a WAN link. Here, replication is typically used for offsite disaster recovery purposes.

In either case, a secondary cluster synchronized with the primary production cluster can afford a substantially improved RTO and RPO than tape backup and both implementations have their distinct advantages. And SyncIQ performance is easily tuned to optimize either for network bandwidth efficiency across a WAN or for LAN speed synchronization. Synchronization policies may be configured at the file-, directory-, or entire-file-system-level and can either be scheduled to run at regular intervals or run manually.

SyncIQ supports up to one thousand defined policies, of which up to fifty may run concurrently. SyncIQ policies also have a priority setting to allow favored policies to preempt others. In addition to chronological scheduling, replication policies can also be configured to start whenever the source is modified (change based replication). If preferred, a delay period can be added to defer the start of a change-based policy.

SyncIQ encryption, which is available in OneFS 8.2 and later, ensures that the security of replicated data in-flight. X.509 Transport Layer Security (TLS) certificates used by the nodes in the replicating clusters are managed through the SyncIQ certificate store. SyncIQ encryption can be configured either per-policy or globally.

SyncIQ linear restore

Leveraging OneFS SnapshotIQ infrastructure, the Linear Restore functionality of SyncIQ can detect and restore (commit) consistent, point in time, block-level changes between cluster replication sets, with a minimal impact on operations and a granular RPO. This “change set” information is stored in a mirrored database on both source and target clusters and is updated during each incremental replication job, enabling rapid failover and failback RTOs.

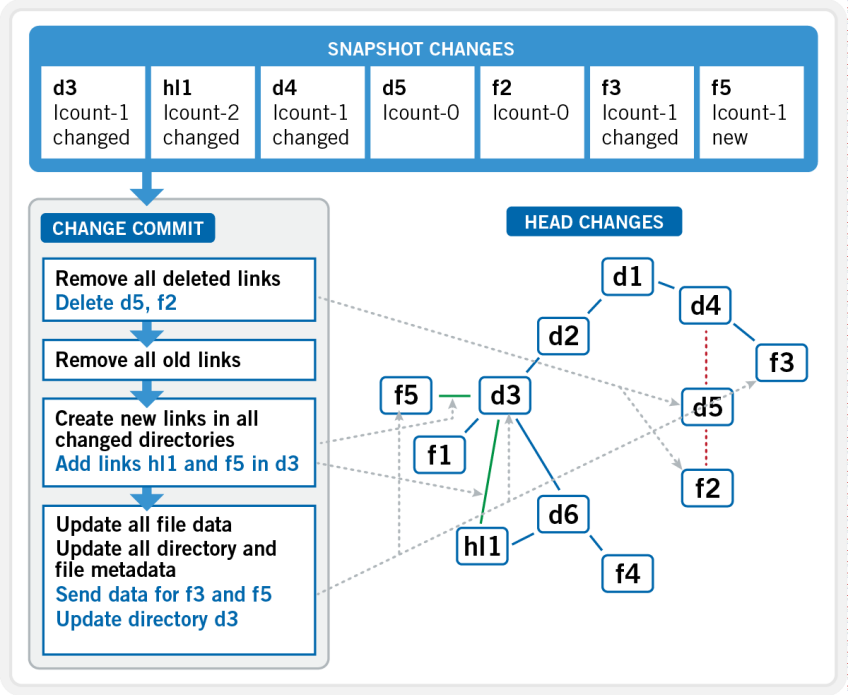

Figure 24. SyncIQ linear restore change commit mechanism

SyncIQ replica protection

All writes outside of the synchronization process itself are disabled on any directory that is a target for a specific SyncIQ job. However, if the association is broken between a target and a source, the target may then return to a writable state. Subsequent resolution of a broken association will force a full resynchronization to occur at the next job run. As such, restricted writes prevent modification, creation, deletion, linking, or movement of any files within the target path of a SyncIQ job. Replicated disaster recovery (DR) data is protected within and by its SyncIQ container or restricted-writer domain, until a conscious decision is made to bring it into a writable state.

SyncIQ failover and failback

If a primary cluster becomes unavailable, SyncIQ enables failover to a mirrored, DR cluster. During such a scenario, the administrator decides to redirect client I/O to the mirror and initiates SyncIQ failover on the DR cluster. Users continue to read and write to the DR cluster while the primary cluster is repaired.

Once the primary cluster becomes available again, the administrator may decide to revert client I/O back to it. To achieve this, the administrator initiates a SyncIQ failback prep process which synchronizes any incremental changes made to the DR cluster back to the primary.

Failback is divided into three distinct phases:

1. First, the prep phase readies the primary to receive changes from the DR cluster by setting up a restricted writer domain and then restoring the last known good snapshot.
2. Next, upon successful completion of failback prep, a final failback differential sync is performed.
3. Lastly, the administrator commits the failback, which restores the primary cluster back to its role as the source and relegates the DR cluster back to a target again.

In addition to the obvious unplanned failover and failback, SyncIQ also supports controlled, proactive cluster failover and failback. This provides two major benefits:

- The ability to validate and test DR procedures and requirements
- The ability to perform planned cluster maintenance

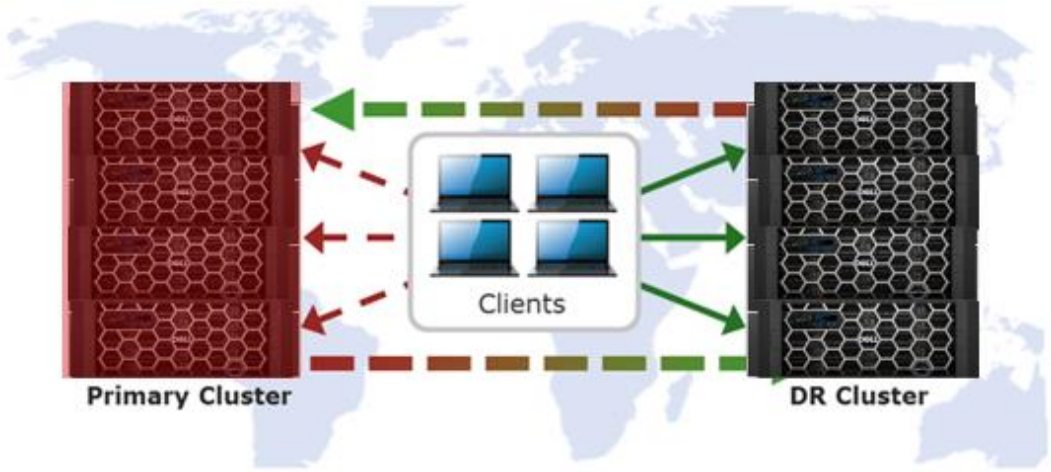

Figure 25. SyncIQ automated data failover and failback

Note: SyncIQ data failover and failback is fully supported for both enterprise and compliance SmartLock WORM datasets.

Continuous replication mode

Complementary to the manual and scheduled replication policies, SyncIQ also offers a continuous mode, or replicate on change, option. When the **Whenever the source is modified** policy configuration option is selected, SyncIQ will continuously monitor the replication dataset (sync domain) and automatically replicate and changes to the target cluster. Events that trigger replication include file additions, modifications and deletions, directory path, and metadata changes. Also, include and exclude rules can also be applied to the policy, providing a further level of administrative control.

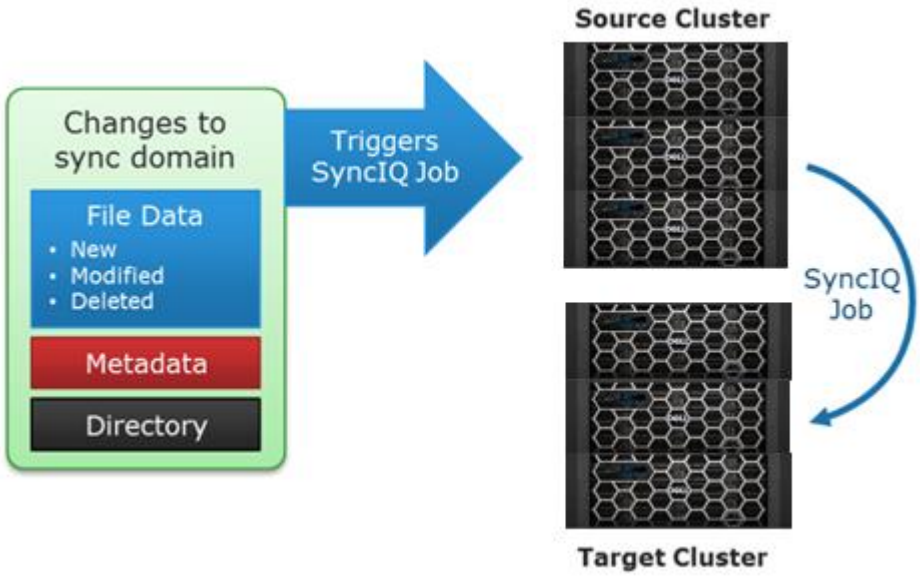

Figure 26. SyncIQ replicate on change mode

Archiving and data security

As we have seen, SyncIQ software enables the simple creation of a secure remote archive.

SmartPools

SmartPools software enables the separation of data according to its business value, aligning it with the appropriate class of storage and levels of performance and protection. Data movement is seamless. Also, with file-level granularity and control using automated policies, manual control, or the API interface, you can tune performance and layout, storage tier alignment, and protection settings—all with minimal impact to end users.

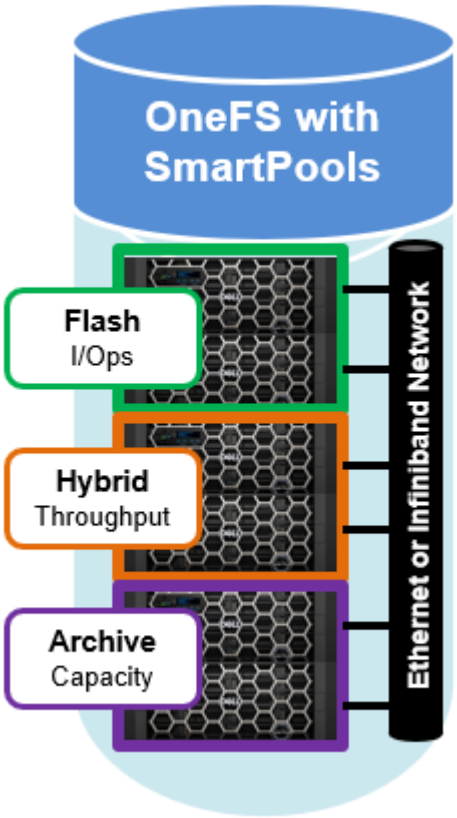

Figure 27. SmartPools tiering model

SmartPools facilitates the creation and management of a dedicated local archive pool within a cluster for data retention and high availability purposes.

CloudPools

CloudPools software provides a cloud storage tiering capability which is built on the SmartPools data management framework. CloudPools enables data to be stored in a cold or frozen data tier or archive, thereby taking advantage of lower-cost, off-premises storage.

File pool policies define which old or unused data on a cluster should be archived to the cloud. The supported cloud storage options include Amazon S3, Microsoft Azure, Google Cloud, Alibaba, Dell ECS, and OneFS to OneFS through the OneFS RAN (RESTful access to namespace API). CloudPools automatically optimizes and protects the transfer of data to cloud storage with the use of both encryption and compression.

Although file data is moved to remote storage, the files remain visible in the OneFS file system. CloudPools accomplishes this by retaining a local SmartLink (or stub) file, which is a pointer to the location of data in the cloud. CloudPools is also integrated with the

other OneFS storage management and protection services including SnapshotIQ, SyncIQ replication, SmartQuotas, and NDMP backup.

SmartLock

OneFS uses SmartLock software to provide immutable storage for data. Based on a write once, read many (WORM) locking capability, SmartLock ensures tamper-proof archiving of critical datasets for disaster recovery and regulatory compliance purposes. Configured at the directory-level, SmartLock delivers simple to manage secure data containers that remain locked for a configurable duration or indefinitely. Additionally, SmartLock satisfies the regulatory compliance demands of stringent data retention policies, including SEC 17a-4.

Data Encryption at Rest

OneFS also provides a solution for the security of data at rest. This involves dedicated storage nodes containing self-encrypting drives (SEDs), with an encryption key management system embedded within OneFS. Data is encrypted on disk using the AES-256 cipher, and each SED has a unique data encryption key (DEK) which is used to encrypt and decrypt data as it is read from and written to disk. OneFS automatically generates an authentication key (AK) that wraps and secures the DEK. This means that the data on any SED which is removed from its source node cannot be unlocked and read, thereby guarding against the data security risks of hard drive theft.

The Data Encryption at Rest solution also allows SED drives to be securely wiped before being repurposed or retired, using cryptographic erasure. Cryptographic erasure involves “shredding” the encryption keys to wipe data and can be done in a matter of seconds. To achieve this, OneFS irreversibly overwrites the vendor-provided password, or MSID, on each drive, resulting in all the on-disk data being scrambled.

OneFS encryption of data at rest satisfies several industries’ regulatory compliance requirements, including U.S. Federal FIPS 140-2 Level 2 and PCI-DSS v2.0 section 3.4.

Audit

OneFS Audit can detect potential sources of data loss, fraud, inappropriate entitlements, access attempts that should not occur, and a range of other anomalies that are indicators of risk. This can be especially useful when the audit process associates data access with specific user identities.

In the interests of data security, OneFS provides “chain of custody” auditing by logging specific activity on the cluster. This includes OneFS configuration changes plus NFS, SMB, and HDFS client protocol activity, which are required for organizational IT security compliance, as mandated by regulatory bodies like HIPAA, SOX, FISMA, and MPAA.

OneFS auditing uses Dell’s Common Event Enabler (CEE) to provide compatibility with external, third-party audit applications such as Varonis DatAdvantage. A cluster can write audit events across up to five CEE servers per node in a parallel, load-balanced configuration, allowing OneFS to deliver an end to end, enterprise grade audit solution.

OneFS hardening

OneFS provides a hardened profile that can be enabled for sites that are looking for additional security or that need to comply with the U.S. Department of Defense’s Security Technical Implementation Guide (STIG). This profile can be enabled or disabled on a cluster as required but does require a zero-cost license to activate.

OneFS 9.5 introduced several additional security enhancements that satisfy Defense Information Systems Agency (DISA) and other Federal and corporate security mandates and requirements. These include:

- Multifactor authentication (MFA) using access cards (CAC and PIV), single sign-on (SSO) through SAML for the WebUI, and PKI-based authentication
- FIPS 140-2 Data in Flight encryption for major protocols, FIPS 140-2 data at rest using SEDs, SEDs Master Key rekey, and TLS 1.2 support
- Introduction of a host-based firewall, permitting restriction of the management interface to a dedicated subnet and hosts to specified IP pools
- IPV6-only network support for the USGv6R1 standard

Secure Remote Services and SupportAssist

OneFS communicates with the Dell backend using Secure Remote Services (ESRS). This is used to send alerts, log gathers, usage intelligence, and managed device status to the backend. Clusters provisioned with this back end can download updates, patches, OneFS software packages, or any other file that has been designated for a cluster to download. Secure Remote Services provides data for technical support personnel to investigate and resolve cluster issues and support requests. In order to use this service, customers are required to host a Secure Remote Services Gateway, which securely proxies Secure Remote Services communication with the cluster.

OneFS 9.5 introduced integration with Dell SupportAssist, the next generation remote connectivity system for transmitting events, logs, and telemetry from a PowerScale cluster to Dell Support. SupportAssist provides a full replacement for ESRS and enables Dell Support to perform remote diagnosis and remediation of cluster issues. Intended for all customers who can send telemetry off-cluster to Dell over the Internet, SupportAssist integrates ESE into PowerScale OneFS along with a suite of daemons to allow its use on a distributed system.

For more information about deploying and configuring Secure Remote Services, see the *Site Planning Guide*.

File filtering

OneFS file filtering can be used across NFS and SMB clients to allow or disallow writes to an export, share, or access zone. This feature prevents certain types of file extensions to be blocked, for files which might cause security problems, productivity disruptions, throughput issues, or storage clutter. Configuration can be done either with an exclusion list, which blocks explicit file extensions, or with an inclusion list, which explicitly allows writes of only certain file types.

Nearline, VTL, and tape backup

At the trailing end of the protection continuum lies traditional backup and restore—whether to tape or disk. This is the bastion of any data protection strategy and usually forms the crux of a “data insurance policy.” With high RPO and RTOs—often involving a retrieval of tapes from secure, offsite storage—tape backup is typically the mechanism of last resort for data recovery in the face of a disaster.

FC backup accelerator card

OneFS 8.2 and later versions support a combination 10 Gb Ethernet and Fibre Channel PCI card, which allows PowerScale nodes with an Ethernet backend to perform two-way NDMP by directly connecting to SAN attached tape library or VTL.

Backup from snapshots

In addition to the benefits provided in terms of user recovery of lost or corrupted files, SnapshotIQ also offers a powerful way to perform backups while minimizing the impact on the file system.

Initiating backups from snapshots affords several substantial benefits. The most significant benefit is that the file system does not need to be quiesced since the backup is taken directly from the read-only snapshot. This eliminates lock contention issues around open files and allows users full access to data throughout the duration of the backup job.

SnapshotIQ also automatically creates an alias which points to the latest version of each snapshot on the cluster, which facilitates the backup process by allowing the backup to always refer to that alias. Since a snapshot is, by definition, a point-in-time (PIT) copy, by backing up from a snapshot, the consistency of the file system or subdirectory is maintained.

You can use the Network Data Management Protocol (NDMP) snapshot capability to further streamline this process. With this capability, you can create a snapshot as part of the backup job, and then delete the snapshot upon successful completion of the backup.

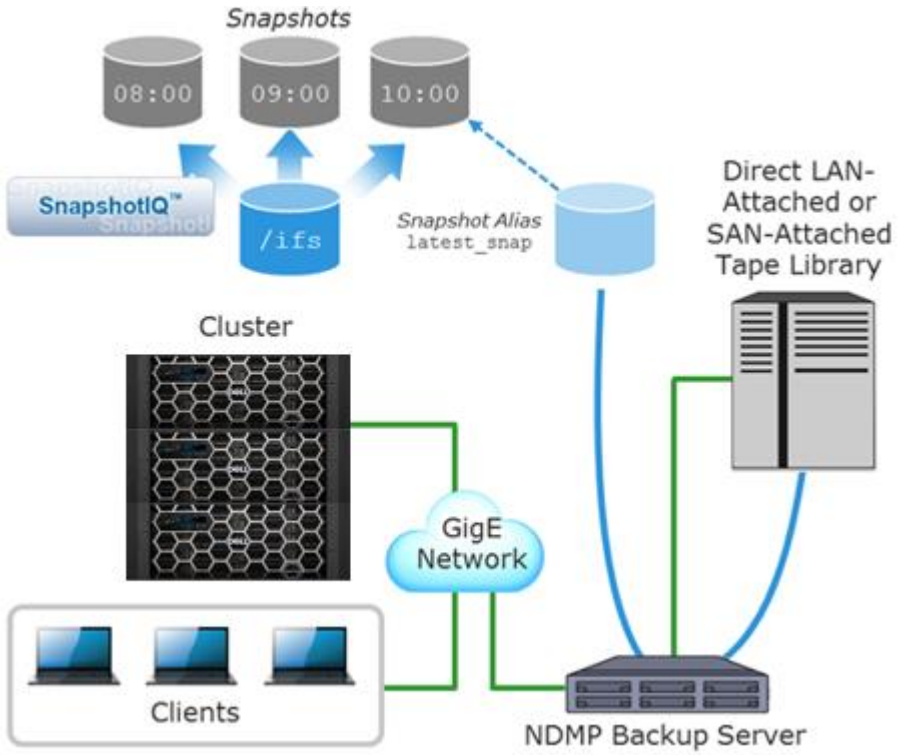

Figure 28. Backup using SnapshotIQ

Parallel streams

The OneFS distributed architecture allows backups to be spread across multiple network streams from the cluster, which can significantly improve performance. This is achieved by dividing the root file system into several paths based on the number of nodes in the cluster and the structure of subdirectories under the file system root. For example, if the file system on a four-node cluster can be separated logically among four subdirectories, each of these subdirectories can be backed up as a separate stream, with one subdirectory served from each node.

OneFS also includes multistream NDMP backup. In this case, a backup job can be configured against a top-level directory, and a separate NDMP stream will be used to back up each subdirectory in parallel. This drastically increases the throughput of backups and simplicity of configuration, thereby allowing fast job completion and the ability to define tighter recovery objectives. Parallel NDMP is supported by both Dell NetWorker and Commvault Simpana DMAs.

NDMP

OneFS facilitates performant backup and restore functionality through its support of the ubiquitous Network Data Management Protocol (NDMP). NDMP is an open-standard protocol that provides interoperability with leading data-backup products and OneFS supports both NDMP versions 3 and 4. OneFS also supports both direct NDMP (referred to as 2-way NDMP), and remote NDMP (referred to as 3-way NDMP) topologies.

The OneFS NDMP module includes the following functionality:

- Full and incremental backups and restores using NDMP
- Direct Access Restore/Directory Direct Access Restore (DAR/DDAR), single-file restores, and three-way backups
- Restore-to-arbitrary systems
- Seamless integration with access control lists (ACLs), alternate data streams, and resource forks
- Selective file recovery
- Replicate then backup
- Multistream NDMP backup
- Backup Restartable Extension (BRE) and multistream BRE

While some backup software vendors may support backing up OneFS over SMB and NFS, the advantages of using NDMP include:

- Increased performance
- Retention of file attributes and security and access controls
- Backups use automatically generated snapshots for point-in-time consistency.
- Redirector automatically distributes two-way NDMP local backup or restore operations to nodes with less load.
- Throttler manages CPU usage of backup, ensuring that NDMP operations do not overwhelm any nodes.
- Extensive support by backup software vendors
- Integration with CloudPools, allowing backup of Smartlinked files as regular files or stubs (with or without data).

Direct NDMP model

This is the most efficient model and results in the fastest transfer rates. Here, the data management application (DMA) uses NDMP over the Ethernet front-end network to communicate with the cluster. On instruction, the cluster, which is also the NDMP tape server, begins backing up data to one or more tape devices which are attached to it by Fibre Channel.

The DMA, a separate server, controls the tape library's media management. File History—the information about files and directories—is transferred from the cluster by NDMP to the DMA, where it is maintained in a catalog.

Direct NDMP is the fastest and most efficient model for backups with OneFS. Direct NDMP requires either a B100 backup accelerator (or a pool of Gen6 nodes with Fibre Channel connectors) to be present within a cluster.

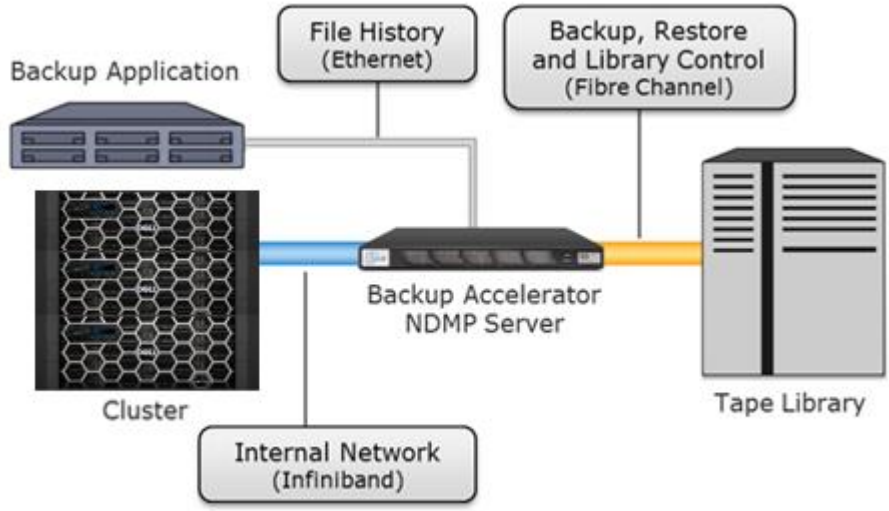

Figure 29. Recommended two-way NDMP over Fibre Channel

Remote NDMP model

In the remote NDMP scenario, there are no Fibre Channel connectors present in the cluster. Instead, the DMA uses NDMP over the LAN to instruct the cluster to start backing up data to the tape server—either connected by Ethernet or directly attached to the DMA host. In this model, the DMA also acts as the Backup/Media Server.

During the backup, file history is transferred from the cluster over NDMP over the LAN to the backup server, where it is maintained in a catalog. In some cases, the backup application and the tape server software both reside on the same physical machine.

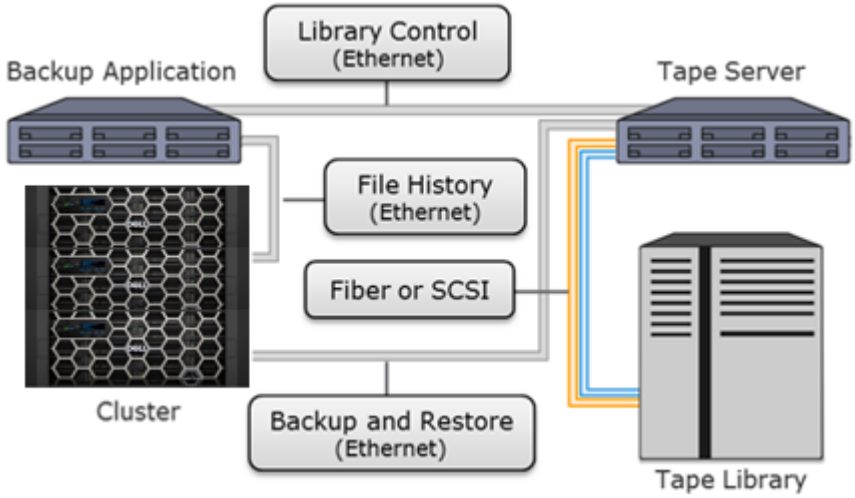

Figure 30. Remote three-way NDMP backup

OneFS remote NDMP backup also includes intelligent IP selection functionality. This enables backup applications such as Avamar and Commvault to select the most desirable interfaces and preferred network for NDMP traffic.

Incremental backup

OneFS accommodates the range of full, incremental, and token-based backups. In standard DR nomenclature, Level 0 indicates a full backup, and levels 1 to 9 are incrementals. Any level specified as 1 to 9 will back up all the files that have been modified since the previous lower-level backup.

Token-based incremental backups are also supported. These are achieved by configuring the data management application (DMA) to maintain a timestamp database and to pass the reference time token on to the cluster for use during each incremental backup. This method does not rely on level based incremental backups, as described above, at all.

Direct access recovery

OneFS provides full supports for Direct Access Recovery (DAR). Direct Access Recovery allows the NDMP server to go directly to the location of a file within an archive and quickly recover that file. As such, it eliminates the need to scan through vast quantities of data typically spread across multiple tapes in an archive set, in order to recover a single file. This capability uses the offset information that is contained in the file history data passed to the DMA at backup time.

Directory DAR

OneFS NDMP also supports Directory DAR (DDAR), an extension of DAR. DDAR allows the NDMP server to go directly to the location of a directory within an archive and quickly recover all files/directories contained within the directory tree hierarchy. Clearly, both DAR and DDAR provide an improved RTO for smaller scale data recovery from tape.

OneFS NDMP offers Selective File Recovery—the ability to recovering a subset of files within a backup archive. Also supported is the ability to restore to alternate path locations.

Certified backup applications

OneFS is certified with a wide range of leading enterprise backup applications, including:

- Symantec NetBackup and Backup Exec
- Dell Avamar and NetWorker
- IBM Tivoli Storage Manager
- CommVault Simpana
- Quest NetVault
- ASG Time Navigator

OneFS is also certified to work with the Dell Cloud Tiering Appliance to simplify data migration and with Dell PowerProtect appliance products for deduplicated backup and archiving.

OneFS 8.2 and later versions include sparse file support when using the cluster as a back target with Commvault Simpana V11 SP10 or later, providing considerable space efficiency improvements.

For more information, see the *Dell PowerScale: NDMP Technical Overview and Design Considerations white paper*.

Summary

Organizations of all sizes around the globe are dealing with a deluge of digital content and unstructured data that is driving massive increases in storage needs. As these enterprise datasets continue to expand to unprecedented sizes, data protection has never been more crucial. A new approach is needed to meet the availability, protection, and performance requirements of this era of big data.

Dell PowerScale enables organizations to linearly scale capacity and performance within a single file system—one which is both simple to manage and highly available and redundant, as we have seen. Built on commodity hardware and powered by the revolutionary OneFS distributed file system, PowerScale NAS solutions deliver the following key tenets:

- Unparalleled levels of data protection
- No single point of failure
- Fully distributed single file system
- Industry-leading tolerance for multi-failure scenarios
- Proactive failure detection and preemptive, fast drive rebuilds
- Flexible, file-level data protection
- Fully journaled file system
- Extreme transient availability

OneFS glossary

The following table lists OneFS related abbreviations and their definitions:

Table 2. OneFS abbreviations

Abbreviation	Definition
BAM	Block Allocation Manager
BAT	Block Allocation Type
BH	Block History
BSD	Berkeley Software Distribution UNIX
BSW	BAM Safe Write
CIFS	Common Internet File System
CoW	Copy on Write Snapshot
DFM	Directory Format Manager
DSR	Dynamic Sector Repair
DWT	Device Worker Thread
FEC	Forward Error Correction
IDI	Isi Data Integrity
IFM	Inode Format Manager
IMDD	Mirrored Device Driver
LACP	Link Aggregation Control Protocol
LAGG	Link Aggregation
LBM	Local Block Manager
LIN	Logical Inode
MDS	Mirrored Data Structure
NFS	Network File System
PIT	Point in Time Snapshot
POSIX	Portable Operating System Interface for UNIX
RM	Remote Block Manager
RoW	Redirect on Write Snapshot
SDP	Sockets Direct Protocol
SMB	Server Message Block
TXN	Transaction Code
VFS	Virtual File System
VOPs	Vnode Operations