Dell PowerScale: CloudPools and Microsoft Azure

Architectural Overview, Considerations, and Best Practices

May 2023

H17746.7

White Paper

Abstract

This whie paper provides an overview of Dell PowerScale CloudPools software in OneFS 9.4.0.0. It describes policy-based capabilities that can reduce storage costs and optimize storage by automatically moving infrequently accessed data to Microsoft Azure.

DELL Technologies

Copyright

Contents

Executive summary 4

CloudPools solution architectural overview 6

CloudPools 2.0 12

Best practices for PowerScale storage and Microsoft Azure 20

Reporting 27

Commands and troubleshooting 28

Technical support and resources 31

Appendix A: Step-by-step configuration example 32

Executive summary

Overview

This white paper describes about how Dell PowerScale CloudPools in OneFS 9.0 integrates with Microsoft Azure and it covers the following topics:

- CloudPools solution architectural overview
- CloudPools 2.0 introduction with a focus on the following improvements:
  - Dell PowerScale NDMP and Dell PowerScale SyncIQ support
  - Non-disruptive upgrade (NDU) support
  - Snapshot efficiency
  - Sparse files handling
  - Quota management
  - Anti-virus integration
  - WORM integration
- General considerations and best practices for a CloudPools implementation
- CloudPools reporting, commands, and troubleshooting

Audience

This white paper is intended for experienced system administrators, storage administrators, and solution architects interested in learning how CloudPools works and understanding the CloudPools solution architecture, considerations, and best practices.

This guide assumes the reader has a working knowledge of the following topics:

- Network-attached storage (NAS) systems
- Dell PowerScale scale-out storage architecture and Dell PowerScale OneFS operating system
- Microsoft Azure

The reader should also be familiar with PowerScale and Azure documentation resources including:

- Dell OneFS release notes, available on Dell Support, containing important information about resolved and known issues
- Dell PowerScale OneFS Best Practices
- Microsoft Azure

Revisions

Date	Part number/ revision	Description
April 2019	H17746	Initial release
October 2019	H17746.1	Updated snapshot efficiency
June 2020	H17746.2	Updated best practices
October 2020	H17746.3	Updated CloudPools operations

Date	Part number/ revision	Description
April 2021	H17746.4	Updated best practices
October 2021	H17746.5	Updated performance
April 2022	H17746.6	Updated reporting
May 2023	H17746.7	Updated best practices

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: Jason He

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

CloudPools solution architectural overview

Introduction to CloudPools

The CloudPools feature of OneFS allows tiering cold or infrequently accessed data to lower-cost cloud storage. It is built on the Dell PowerScale SmartPools file pool policy framework, which provides granular control of file placement on a PowerScale cluster.

CloudPools extends the PowerScale namespace to the public cloud, Microsoft Azure, as illustrated in Figure 1. It allows applications and users to seamlessly retain access to data through the same network path and protocols regardless of where the file data physically resides.

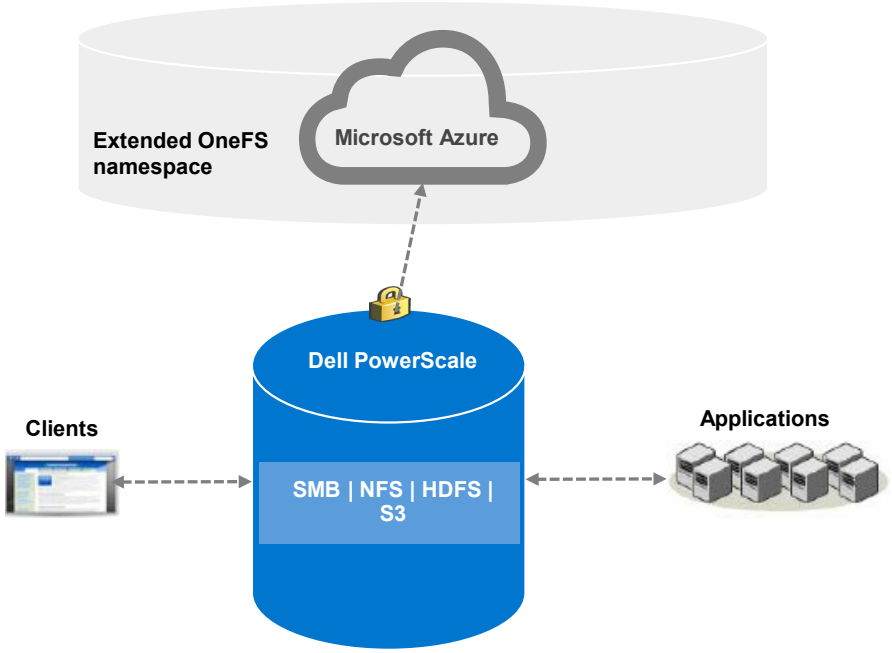

Figure 1. CloudPools solution overview

Note: A SmartPools license and a CloudPools license are required on each node of the PowerScale cluster. A minimum of Dell Isilon OneFS version 8.0.0 is required for CloudPools 1.0, and Dell Isilon OneFS version 8.2.0 for CloudPools 2.0.

Policies are defined on the PowerScale cluster and drive the tiering of data. Clients can access the archived data through various protocols including SMB, NFS, HDFS, and S3.

CloudPools concepts

This section describes key CloudPools concepts including:

- SmartPools
- SmartLink files
- File pool policies

SmartPools

SmartPools is the OneFS data-tiering framework of which CloudPools is an extension. SmartPools alone tiers data between different node types within a PowerScale cluster. CloudPools also adds to tier data outside of a PowerScale cluster.

SmartLink files

Although file data is moved to cloud storage, the files remain visible in OneFS. After file data has been archived to the cloud storage, the file is truncated to an 8 KB file. The 8 KB file is called a SmartLink file or stub file. Each SmartLink file contains a data cache and a map. The data cache is used to retain a portion of the file data locally, and the map points to all cloud objects.

Figure 2 shows the contents of a SmartLink file and the mapping to cloud objects.

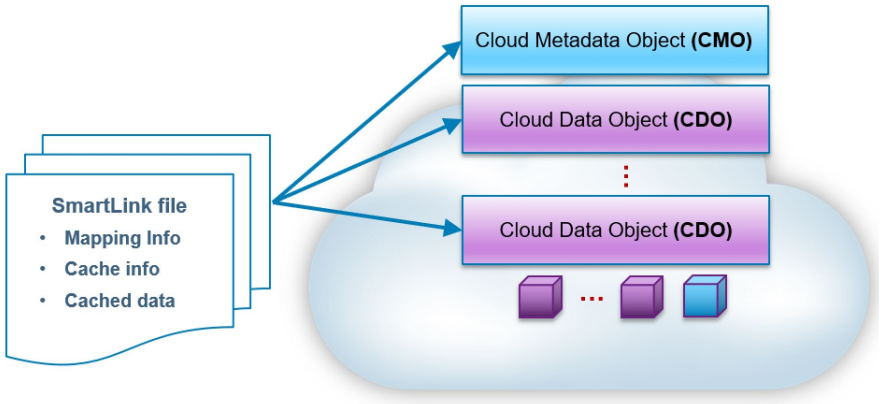

Figure 2. SmartLink file

File pool policies

Both CloudPools and SmartPools use the file pool policy engine to define which data on a cluster should live on which tier or be archived to a cloud storage target. The SmartPools and CloudPools job has a customizable schedule that runs once a day by default. If files match the criteria specified in a file pool policy, the content of those files is moved to cloud storage during the job execution. A SmartLink file is left behind on the PowerScale cluster that contains information about where to retrieve the data. In CloudPools 1.0, the SmartLink file is sometimes referred to as a stub, which is a unique construct that does not behave like a normal file. In CloudPools 2.0, the SmartLink file is an actual file that contains pointers to the CloudPools target where the data resides.

This section describes the key options when configuring a file pool policy, which includes:

- Encryption
- Compression
- File matching criteria
- Local data cache
- Data retention

Encryption

CloudPools provides an option to encrypt data before it is sent to the cloud storage. It leverages the PowerScale key management module for data encryption and uses AES-256 as the encryption algorithm. The benefit of encryption is that only encrypted data is being sent over the network.

Compression

CloudPools provides an option to compress data before it is sent to the cloud storage. It implements block level compression using the zlib compression library. CloudPools does not compress data that is already compressed.

File matching criteria

When files match a file pool policy, CloudPools moves the file data to the cloud storage. File matching criteria enable defining a logical group of files as a file pool for CloudPools. It defines which data should be archived to cloud storage.

File matching criteria include:

- File name
- Path
- File type
- File attribute
- Modified
- Accessed
- Metadata changed
- Created
- Size

Any number of file matching criteria can be added to refine a file pool policy for CloudPools.

Local data cache

Caching is used to support local reading and writing of SmartLink files. It reduces bandwidth costs by eliminating repeated fetching of file data for repeated reads and writes to optimize performance.

Note: The data cache is used for temporarily caching file data from the cloud storage on PowerScale disk storage for files that have been moved off cluster by CloudPools.

The local data cache is always the authoritative source for data. CloudPools looks for data in the local data cache first. If the file being accessed is not in the local data cache, CloudPools fetches the data from the cloud. CloudPools writes the updated file data in the local cache first and periodically sends the updated file data to the cloud.

CloudPools provides the following configurable data cache settings:

- **Cache expiration:** This option is used to specify the number of days until OneFS purges expired cache information in SmartLink files. The default value is one day.

- **Writeback frequency:** This option is used to specify the interval at which OneFS writes the data stored in the cache of SmartLink files to the cloud. The default value is nine hours.
- **Cache read ahead:** This option is used to specify the cache read ahead strategy for cloud objects (partial or full). The default value is partial.
- **Accessibility:** This option is used to specify how data is cached in SmartLink files when a user or application accesses a SmartLink file on the PowerScale cluster. Values are **cached** (default) and **no cache**.

Data retention

Data retention is a concept used to determine how long to keep cloud objects on the cloud storage. There are three different retention periods:

- **Cloud data retention period:** This option is used to specify the length of time cloud objects are retained after the files have been fully recalled or deleted. The default value is one week.
- **Incremental backup retention period for NDMP incremental backup and SyncIQ:** This option is used to specify the length of time that CloudPools retains cloud objects referenced by a SmartLink file. And SyncIQ replicates the SmartLink file or NDMP backs up the SmartLink file using an incremental NDMP backup. The default value is five years.
- **Full backup retention period for NDMP only:** This option is used to specify the length of time that OneFS retains cloud data referenced by a SmartLink file. And NDMP backs up the SmartLink file using a full NDMP backup. The default value is five years.

Note: If more than one period applies to a file, the longest period is applied.

Microsoft Azure

This section describes the following cloud objects in Microsoft Azure:

- Cloud metadata object
- Cloud data object

Cloud metadata object

A cloud metadata object (CMO) is a CloudPools object in Microsoft Azure that is used for supportability purposes.

Cloud data object

A cloud data object (CDO) is a CloudPools object that stores file data in Microsoft Azure. File data is split into 2 MB chunks to optimize performance before sending it to Microsoft Azure. The chunk is called a CDO. If file data is less than the chunk size, the CDO size is equal to the size of the file data.

Note: The chunk size is 1 MB in CloudPools 1.0 and versions earlier than OneFS 8.2.0.

CloudPools operations

This section describes the workflow of CloudPools operations:

- Archive
- Recall

- Read
- Update

Archive

The archive operation is the CloudPools process of moving file data from the local PowerScale cluster to cloud storage. Files are archived either using the SmartPools Job or from the command line. The CloudPools archive process can be paused or resumed. For details, see Commands.

Figure 3 shows the workflow of the CloudPools archive.

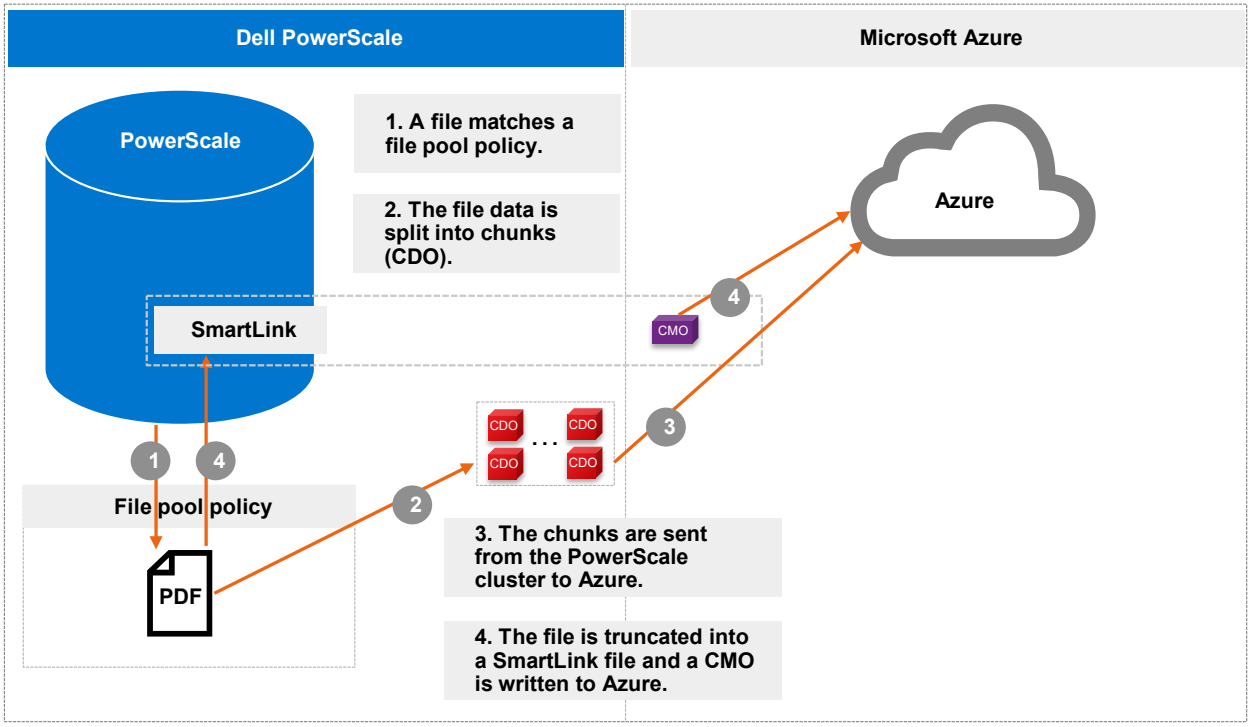

Figure 3. Archive workflow

The file pool policy (see File pool policies) in step 1 in Figure 3 specifies a cloud target and cloud-specific parameters.

Policy examples include:

- Encryption
- Compression
- Local data cache
- Data retention

When chunks are sent from the PowerScale cluster to Azure (step 3 in Figure 3), a checksum is applied for each chunk to ensure data integrity.

Recall

The recall operation is the CloudPools process of reversing the archive process. It replaces the SmartLink file by restoring the original file data on the PowerScale cluster

and removing the cloud objects in Azure. The recall process can only be performed using the command line. The CloudPools recall process can be paused or resumed. For detailed instructions, see Commands.

Figure 4 shows the workflow of CloudPools recall.

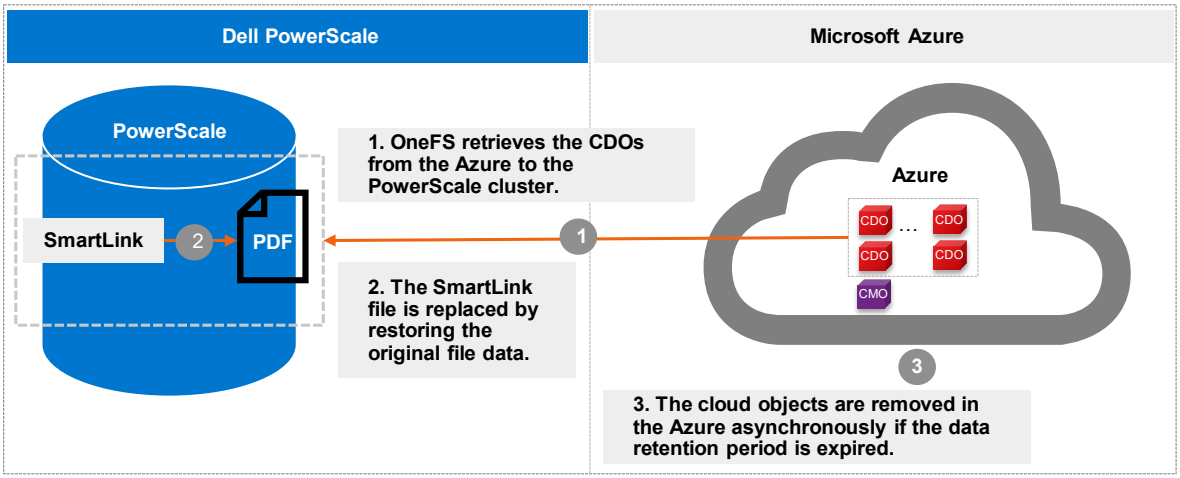

Figure 4. Recall workflow

Read

The read operation is the CloudPools process of client data access, known as inline access. When a client opens a file for read, the blocks are added to the cache in the associated SmartLink file by default. The cache can be disabled by setting the accessibility. For more information, see File pool policies.

Figure 5 shows the workflow of CloudPools read by default.

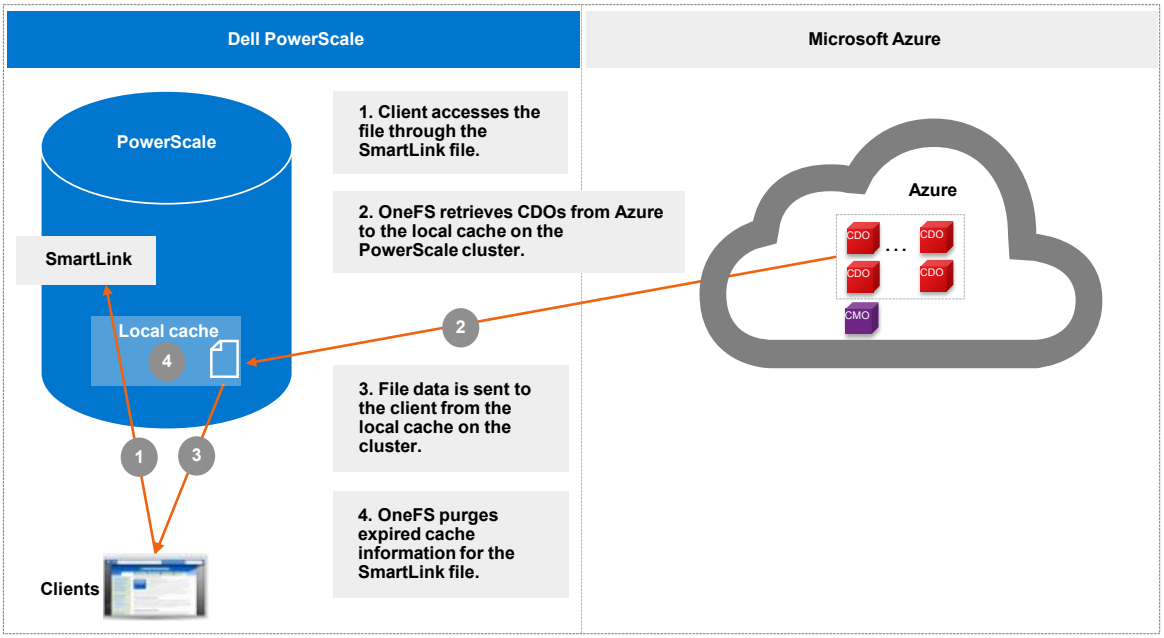

Figure 5. Read workflow

Starting from OneFS 9.1.0.0, cloud object cache is introduced to enhance CloudPools functions for communicating with cloud. In step 1 in Figure 5, OneFS looks for data in the object cache first and OneFS retrieves data from the object cache if the data is already in the object cache. Cloud object cache reduces the number of requests to Azure when reading a file.

Before OneFS 9.1.0.0, OneFS looks for data in the local data cache first in step 1. It moves to step 3 if the data is already in the local data cache.

Note: Cloud object cache is per node. Each node maintains its own object cache on the cluster.

Update

The update operation is the CloudPools process that occurs when clients update data.

When clients change to a SmartLink file, CloudPools first writes the changes in the data local cache and then periodically sends the updated file data to Azure. The space used by the cache is temporary and configurable. For more information, see File pool policies.

Figure 6 shows the workflow of the CloudPools update.

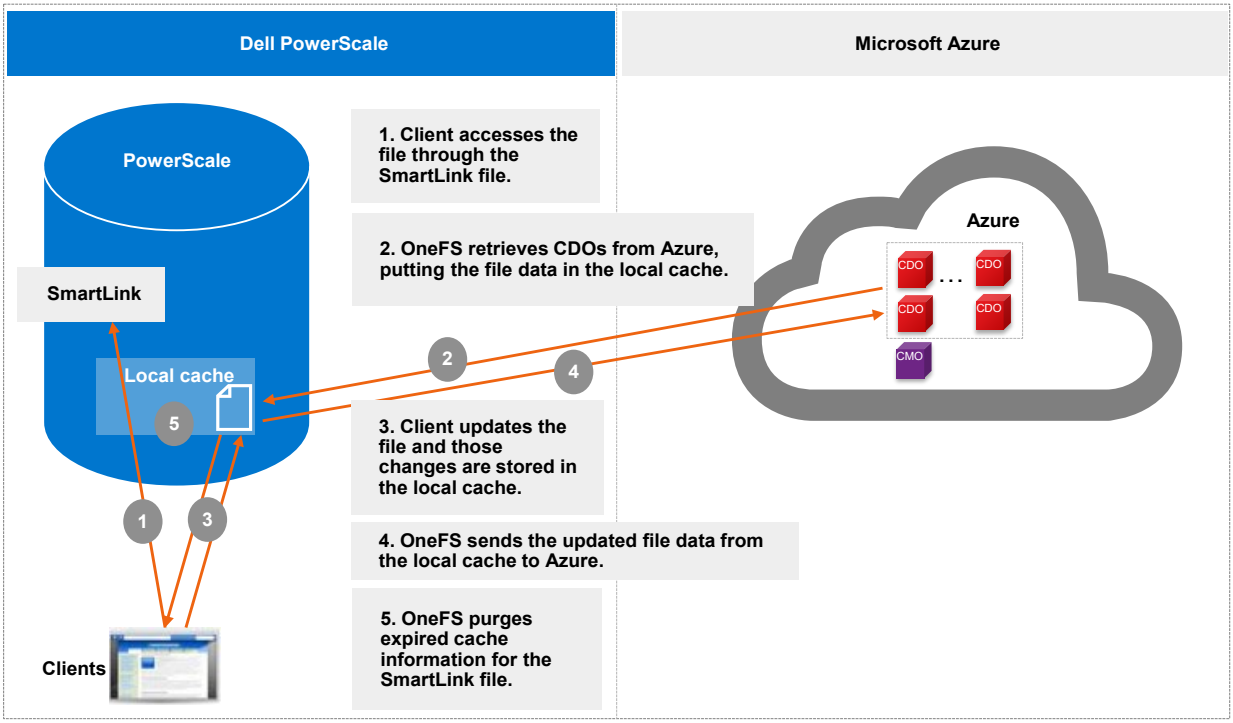

Figure 6. Update workflow

CloudPools 2.0

Introduction to CloudPools 2.0

CloudPools 2.0 is the next generation of CloudPools, released in OneFS 8.2.0. This chapter describes the following improvements in CloudPools 2.0:

- NDMP and SyncIQ support

- Non-disruptive upgrade (NDU) support
- Snapshot efficiency
- Sparse files handling
- Quota management
- Anti-virus integration
- WORM integration

NDMP and SyncIQ support

When the CloudPools version is different between the source cluster and the target PowerScale cluster, the CloudPools cross-version compatibility is handled.

NDMP and SyncIQ provide two types of copy or backup: shallow copy and deep copy. For more information about NDMP and SyncIQ protection, see the *High Availability and Data Protection with Dell PowerScale Scale-out NAS* white paper.

- **Shallow copy (SC)/backup:** Replicates or backs up SmartLink files to the target PowerScale cluster or tape as SmartLink files without file data.
- **Deep copy (DC)/backup:** Replicates or backs up SmartLink files to the target PowerScale cluster or tape as regular files or unarchived files. The backup or replication is slower than normal. Disk space is consumed on the target cluster for replicating data.

Table 1 shows the CloudPools and OneFS mapping information. CloudPools 2.0 is released along with OneFS 8.2.0. CloudPools 1.0 is running in OneFS 8.0.x or 8.1.x.

Table 1. CloudPools and OneFS mapping information

OneFS version	CloudPools version
OneFS 8.0.x/OneFS 8.1.x	CloudPools 1.0
OneFS 8.2.0 or higher	CloudPools 2.0

Table 2 shows the NDMP and SyncIQ supported use cases when running a different version of CloudPools on the source and target clusters. As noted, if CloudPools 2.0 is running on the source PowerScale cluster and CloudPools 1.0 is running on the target PowerScale cluster, shallow copies are not allowed.

Table 2. NDMP and SyncIQ Supported use cases with CloudPools 2.0

Source	Target	SC NDMP	DC NDMP	SC SyncIQ replication	DC SyncIQ replication
CloudPools 1.0	CloudPools 2.0	Support	Support	Support	Support
CloudPools 2.0	CloudPools 1.0	No Support	Support	No Support	Support

Non-disruptive upgrade support

When a cluster that has been using CloudPools 1.0 is upgraded to OneFS 8.2.0 or higher, a new CHANGEOVER process is initiated automatically after the upgrade commit. The process ensures a smooth transition from CloudPools 1.0 to CloudPools 2.0. CloudPools 2.0 is ready to use once the upgrade state is committed. For more information about upgrade states, see the *PowerScale Non-Disruptive Upgrade (NDU) Best Practices* white paper.

Snapshot efficiency

Before OneFS 8.2.0, CloudPools 1.0 supported archiving files with existing snapshots. However, CloudPools 1.0 had a limitation when archiving files that have existing snapshots: the copy-on-writes (CoW) process copied the entire contents of the file into the snapshot. Archiving files with existing snapshots therefore did not save space on the PowerScale cluster until the previously CoW-created snapshots expired. CloudPools 1.0 offered an option (clear the **Archive files with snapshots** checkbox in the WebUI) to skip such files with snapshots. A user might have not chosen to archive files with snapshots if the previously CoW-created snapshots had long retentions. This case is to avoid creating another copy on cloud storage where the retention period meant it would persist on PowerScale storage anyway.

CloudPools 2.0 eliminates CoW on the primary data source PowerScale cluster when archiving files with snapshots to the cloud. The file data is only stored in the cloud storage, which saves space on the PowerScale cluster. For more information about data CoW for snapshots, see the *Data Protection with Dell PowerScale SnapshotIQ* white paper.

However, CloudPools 2.0 does not archive files on the target cluster in a SyncIQ relationship. In an environment with long snapshot retentions and an expectation that the same snapshots are maintained in both clusters. It is possible for storage usage on a target cluster to grow larger than the storage on the primary cluster that has CloudPools enabled. For space efficiency, a user with requirements for long snapshot retentions on two clusters in a SyncIQ relationship might choose to use natively tiered PowerScale archive storage, rather than CloudPools.

SnapshotIQ can take read-only, point-in-time copies of any directory or subdirectory within OneFS. A file in one directory can be either a regular file or a SmartLink file before creating a snapshot. A regular file can be truncated to a SmartLink file after archiving its file data to the cloud. A SmartLink file can be converted to a regular file after recalling its file data to the PowerScale cluster. When a snapshot is taken, it preserves the exact state of a file system at that instant. A file in the snapshot directory (/ifs/.snapshot) is a SmartLink file if the same file in the source directory is a SmartLink file. A file in the snapshot directory is a regular file if the same file in the source directory is a regular file. The earlier version of data can be accessed later in the snapshot directory.

The following scenarios address CloudPools 2.0 and snapshots. HEAD is the current version of a SmartLink file in the source directory.

- The file is already a SmartLink file in the source directory before creating a snapshot.
  - Scenario 1: Update HEAD
  - Scenario 2: Update HEAD multiple times and a new snapshot is created between multiple updates
  - Scenario 3: Read file data from a snapshot
- The file is still a regular file in the source directory before creating a snapshot. Then, the regular file is archived to the cloud after a snapshot creation.
  - Scenario 4: Update HEAD
  - Scenario 5: Read file data from a snapshot

Scenario 1

When updating HEAD (SmartLink files in snapshot), a new SmartLink is generated for HEAD when updating HEAD and write-back to the cloud. Cache for HEAD will be empty once its own cache expires. For the workflow of updating a SmartLink file, see Update. The original version SmartLink file is still used for the next snapshot of HEAD. This scenario does not cause the snapshot space to grow. Figure 7 shows the process of scenario 1 to update HEAD when SmartLink files are in the snapshot directory.

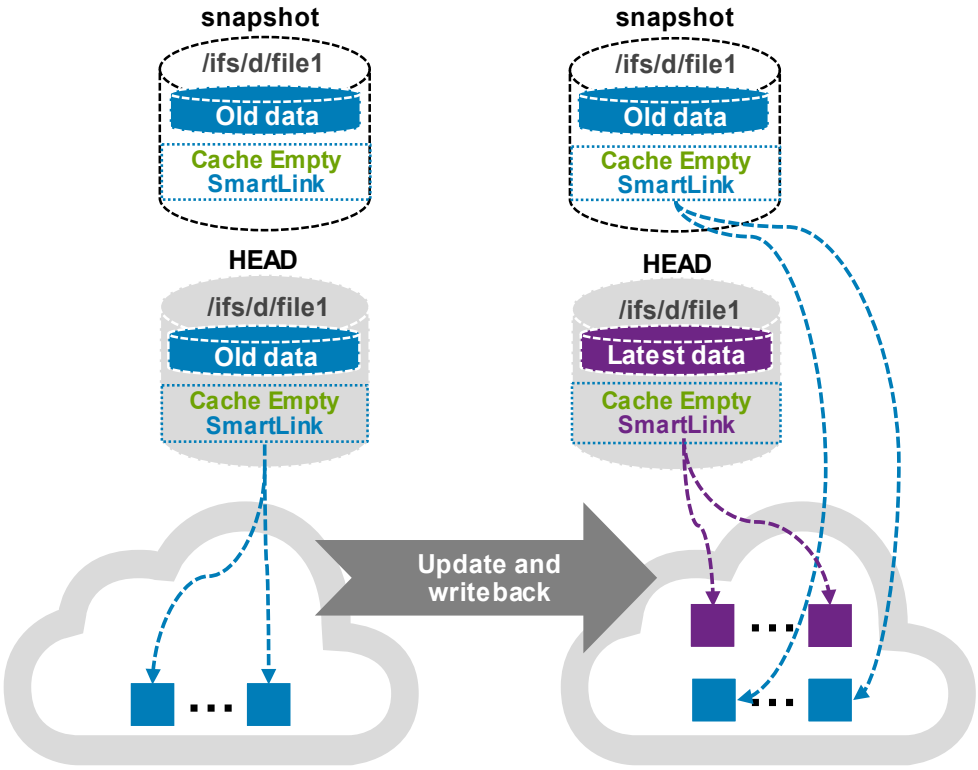

Figure 7. Scenario 1: Update HEAD when SmartLink files are in the snapshot directory

Scenario 2

This scenario describes updating HEAD multiple times, and a new snapshot is created between multiple updates (SmartLink files in snapshot). For example, a user updates HEAD (the first update) while a new (most recent) snapshot is created before the first update write-back is made to the cloud. Then, another user updates (the second update) HEAD again after the new (most recent) snapshot is created. Now there are two snapshots: one snapshot is the next snapshot of HEAD, the other is the most recent snapshot of HEAD.

When a snapshot is taken, it preserves the exact state of a file system at that instant. Data for the next snapshot of HEAD is the old data that is already archived to the cloud and its cache is empty. Data for the most recent snapshot is the new data and its cache is dirty before the new data write-back is made to the cloud. The new data contains old data with the first update. Data for HEAD is the latest data and its cache is dirty before the latest data write-back is made to the cloud. The latest data contains old data with the first update and the second update. A new version SmartLink is generated for the most recent snapshot after the new data write-back is made to the cloud (write-back in the snapshot).

The new data contains old data with the first update. Also, a new version SmartLink is generated for HEAD after the latest data write-back is made to the cloud (write-back in HEAD). Cache for the most recent snapshot or HEAD becomes empty once its own cache expires. Now, all file data is only stored on the cloud and saves space on the PowerScale cluster. Users can read file data from its own SmartLink file at any time.

Figure 8 shows the process of scenario 2.

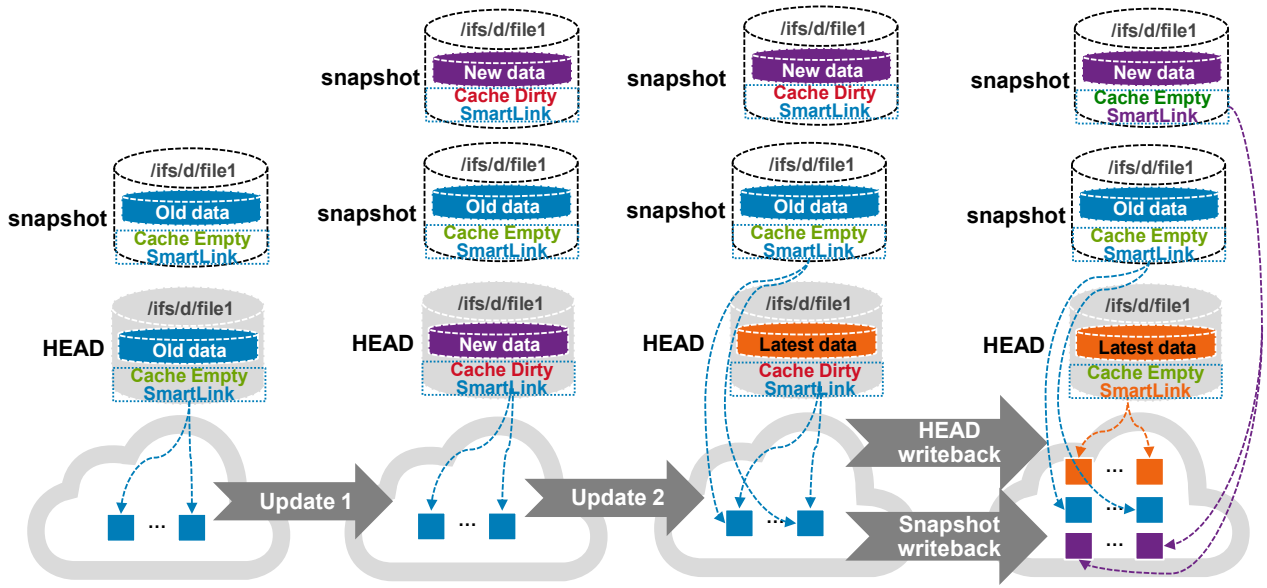

Figure 8. Scenario 2: Update HEAD multiple times and perform a write-back in the snapshot

Scenario 3

This scenario describes reading file data from a snapshot (SmartLink files in snapshot). The files in the next snapshot and HEAD use the same version of SmartLink file when not updating HEAD after the snapshot is created. This scenario is no different than reading the same file from HEAD or the next snapshot of HEAD. For the workflow of reading a SmartLink file, see Read. The same local data cache is used when reading the same file from HEAD and the next snapshot of HEAD simultaneously. This scenario does not cause the snapshot space to grow. The file in the snapshot directory uses its version of SmartLink file when updating HEAD and performing a write-back to the cloud like in scenario 1 or scenario 2. Users can read earlier versions of file data in the snapshot directory. The snapshot space could grow temporarily for cache data, and the grown space is released once its own cache expires.

Scenario 4

In this scenario, when updating HEAD (regular files in snapshot). A SmartLink file is used for HEAD, and a regular file is used for the same file in the next snapshot of HEAD. A new SmartLink file is generated for HEAD when updating HEAD and performing a write-back to the cloud. The cache for HEAD is empty once its own cache expires. Meanwhile, OneFS enables the Block Allocation Manager Cache Manager (BCM) on the regular file in the next snapshot of HEAD. BCM contains the metadata of mapping to cloud objects for the regular file in the next snapshot of HEAD. This scenario does not cause the snapshot space to grow.

Figure 9 shows scenario 4.

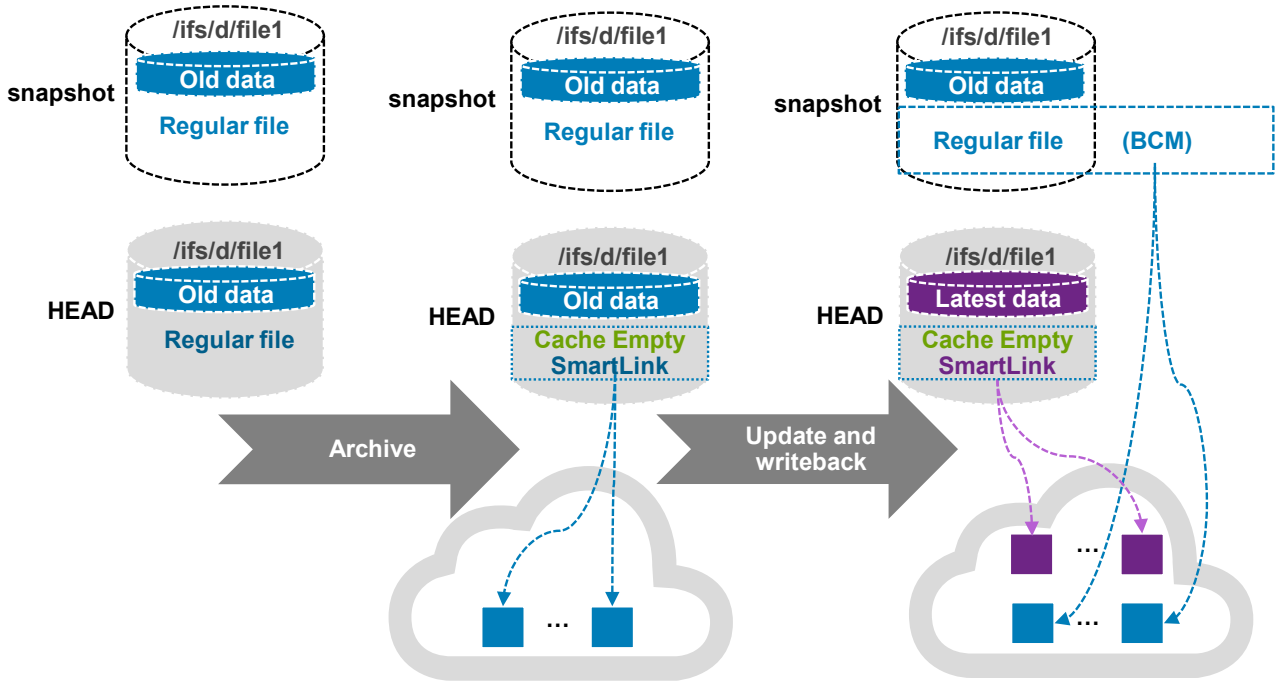

Figure 9. Scenario 4: Update HEAD when regular files are in the snapshot directory

Scenario 5

In this scenario, when reading file data from a snapshot (regular files in snapshot). File data is the same for HEAD (SmartLink file) and the same file (regular file) in the next snapshot of HEAD when not updating HEAD after the snapshot creation. File data is read from HEAD when reading the same file in the next snapshot of HEAD. This scenario does not cause the snapshot space to grow. The file in the next snapshot of HEAD is a regular file (enabled BCM). And the file has the earlier version of data when updating HEAD and performing a write-back to the cloud like in scenario 4. The earlier version of data is retrieved from the cloud by BCM. File data is stored on the PowerScale cluster when reading the earlier version of data from the regular file in the next snapshot of HEAD. The snapshot space grows, and the grown space is not released unless the snapshot is deleted.

Note: In OneFS 8.2.0, CloudPools 2.0 supports write-back in a snapshot. For details, see Scenario 3. However, CloudPools 2.0 does not support archiving and recalling files in the snapshot directory. Consider the case when there is already file data in a snapshot on a cluster running a OneFS release earlier than OneFS 8.2.0. That data takes up storage space on the PowerScale cluster, and then the cluster is upgraded to OneFS 8.2.0. Because CloudPools 2.0 does not support archiving files in snapshots to the cloud, the storage space for this snapshot cannot be released when the cluster is upgraded.

If SyncIQ or NDMP backs up the SmartLink files, the mapping file data should be retrieved from the cloud using the backup copy of the SmartLink file. If the backup retention has not expired, the CDOs of the mapping file data cannot be deleted even though the snapshot has been deleted. The reason is that the SmartLink file backup still references the CDOs of the mapping file data. When the backup retention period has

expired and the CDOs of the mapping file data are no longer used, the CDOs of the mapping file data are deleted. For more information about data retention, see Data retention. If SyncIQ or NDMP does not back up SmartLink files, the CDOs of the mapping file data are deleted after the snapshot is deleted.

Users can revert a snapshot or access snapshot data through the snapshots directory (/ifs/.snapshot). The main methods for restoring data from a snapshot are as follows:

- Revert a snapshot through the SnapRevert job.
- Restore a file or directory by using Microsoft Shadow Copy Client on Windows or running the `cp` command on Linux.
- Clone a file from a snapshot (CloudPools does not support cloning a file from a snapshot).

For details on restoring snapshot data, see the *OneFS 8.2.0 Web Administration Guide*. CloudPools does not support cloning a file from a snapshot. The other two methods for restoring data from a snapshot in a CloudPools environment are described as follows.

When using the SnapRevert job to restore data from a snapshot, it reverts a directory back to the state it was in when a snapshot was taken. For example, there is a `/ifs/test` directory including a regular.txt regular file, and a `smartlink.txt` SmartLink file that has its file data archived to the cloud. A snap01 snapshot is created on the `/ifs/test` directory, and updates are made on the two files. The `regular.txt` file is then archived to the cloud, and it is truncated to a SmartLink file. Then, the SmartLink file `smartlink.txt` is recalled and it is converted to a regular file. If the snapshot snap01 is restored, it overwrites the files in directory `/ifs/test`. The `regular.txt` file reverts to a regular file, and `smartlink.txt` reverts to a SmartLink file. The directory `/ifs/test` is reverted to the state it was in when snap01 was taken.

When you use Microsoft Shadow Copy Client on Windows or run the `cp` command on Linux, the file data is retrieved from the cloud through SmartLink files in a snapshot. This copy operation creates new regular files. That means extra space is required for the new regular files restored from a snapshot.

Sparse files handling

CloudPools 2.0 provides a new sparse file format to improve handling of empty blocks. With this improvement, sparse zeros are not in CloudPools operations, which reduce network utilization and saves space on the cloud target.

Note: No cloud objects are written when archiving full sparse files (fully empty blocks).

Quota management

In OneFS 8.2.0, quotas present actual space consumed on the PowerScale cluster.

For example, there is a directory or user quota of 500 GB and it is reporting 400 GB used. 200 GB of files are archived from the PowerScale cluster to cloud. Moving data to the cloud reduces the quota's measured node space consumption. In OneFS releases earlier than 8.2.0, the amount of data that has been archived to the cloud frees the quota. And the quota shows 200 GB (400 GB to 200 GB) used out of 500 GB. That means the user or directory quota can exceed the set limit (500 GB). In OneFS 8.2.0, the application logical size integrated with CloudPools 2.0 measures the true capacity consumption even if data is archived from the PowerScale cluster to the cloud. And the quota shows 400 GB used

out of 500 GB through the application logical size. That means the user or directory quota cannot exceed the set limit of 500 GB.

For more information about the new SmartQuotas reporting capabilities in OneFS 8.2.0, see the *Storage Quota Management and Provisioning with Dell PowerScale SmartQuotas* white paper.

Anti-virus integration

In OneFS releases before OneFS 8.2.0, SmartLink files were skipped for anti-virus scanning.

In OneFS 8.2.0, CloudPools 2.0 provides a configurable option for anti-virus scanning of SmartLink files. The file data is retrieved from the cloud and cached on the cluster for the scan only if the option is enabled. The scan is slower than normal. As shown in Figure 10, the **Scan Cloudpool Files** option is configured and verified using the command line.

```
hop-isi-n-1# isi antivirus settings modify --scan-cloudpool-files=1
hop-isi-n-1# isi antivirus settings view
           Fail Open: Yes
         Glob Filters: -
Glob Filters Enabled: No
Glob Filters Include: No
        Path Prefixes: -
               Repair: Yes
       Report Expiry: 1Y
       Scan On Close: No
        Scan On Open: No
Scan Cloudpool Files: Yes
   Scan Size Maximum: 2.00G
             Service: No
          Quarantine: Yes
            Truncate: No
```

Figure 10. Enable Scan Cloudpool Files

Note: The **Scan Cloudpool Files** option is disabled by default, which means SmartLink files are skipped when scanning a directory that includes SmartLink files.

WORM integration

Dell PowerScale SmartLock is an optional software feature of OneFS that enables SEC 17-a4 data compliance. In enterprise mode, individual directories can be set up as Write Once, Read Many (WORM) directories. And the data is immutable by everyone except the root account on the cluster once the files have been committed. A PowerScale cluster can also be set up in compliance mode where the root account on the cluster is removed. And no user can change or delete data in WORM-locked folders.

In OneFS versions before OneFS 8.2.0, SmartLink files are not allowed in both enterprise and compliance modes. In OneFS 8.2.0, CloudPools 2.0 and SmartLock integration is as follows:

- **Compliance mode**: SmartLink files are not allowed in compliance mode.
- **Enterprise mode**: SmartLink files are allowed in enterprise mode.
  - Enterprise mode can be enabled on a directory with SmartLink files.

- SmartLink files can be moved into an Enterprise mode directory which prevents modifying or deleting the SmartLink files.
- SmartLink files can be recalled from the cloud to the PowerScale cluster once they are committed.

Best practices for PowerScale storage and Microsoft Azure

PowerScale configuration

This section includes considerations and best practices for configuring PowerScale CloudPools.

CloudPools settings

CloudPools settings can be changed either on the CloudPools setting tab or on a per file pool policy from the OneFS WebUI. It is highly recommended to change these settings on a per file pool policy. The following list includes general considerations and best practices for CloudPools settings.

- **Encryption:** Encryption is an option that can be enabled either on the PowerScale cluster or on Microsoft Azure. The recommendation is to enable encryption on the PowerScale cluster instead of on the Microsoft Azure. If the average CPU is high (greater than 70%) on the PowerScale cluster, the encryption can be enabled on Microsoft Azure instead of on the PowerScale cluster. Encryption adds an additional load on the PowerScale cluster. Encryption can also impact the CloudPools archive and recall performance. For more information about Azure Encryption, see the Microsoft Azure documentation.
- **Compression:** Compression is an option that can be enabled on the PowerScale cluster, in which file data is compressed before sending it to Microsoft Azure. If network bandwidth is a concern, the recommendation is to enable compression on the PowerScale cluster to save network resources. Compression adds an additional load on the PowerScale cluster, which means it might take more time to archive files from PowerScale storage to Microsoft Azure.
- **Data retention:** The recommendation is to explicitly set the data retention for the file data being archived from the PowerScale cluster to Microsoft Azure. If the SmartLink files are backed up with SyncIQ or NDMP, the data retention defines how long the cloud objects remain on Microsoft Azure. Once the retention period has passed, the PowerScale cluster sends a delete command to Microsoft Azure. Microsoft Azure marks the associated cloud objects for deletion. The delete process is asynchronous and the space is not reclaimed until garbage collection completes. This process is a low-priority background process, which may take days to fully reclaim the space depending on how busy the system is.
- **Local data cache:** If the storage space is limited on the PowerScale cluster, the recommendation is to set lower values for the Writeback Frequency and Cache Expiration. This option reduces the time to keep file data in the local data cache and frees up storage space sooner on the PowerScale cluster.

File pool policy

File pool policies define what data is archived from the PowerScale cluster to Microsoft Azure. The considerations include:

- Ensure that the priority of file pool policies is set appropriately. Multiple file pool policies can be created for the same cloud storage account. When the SmartPools job runs, it processes file pool policies in priority order.
- In terms of freeing up storage space on the PowerScale cluster, the recommendation is not to archive small files that are less than 32 KB in size.
- If the files need to be updated frequently, the recommendation is not to archive those files.
- OneFS supports a maximum of 128 file pool policies (SmartPools and CloudPools combined). The recommendation is not to exceed 30 file pool policies per PowerScale cluster.
- If the file pool policy is updated, it has no impact on the files already archived. It affects only the files to be archived when the SmartPools job next runs.
- Archiving based on *accessed time*, rather than *modified* or *created* times, results in files that are used often, including applications, libraries and scripts. Take care to exclude these types of files from being archived to the cloud, which would result in delays for clients or users loading these applications. One example is when you are archiving user home directories that contain files that are created once but accessed often.

Other considerations

Additional considerations include:

- **Deduplication:** CloudPools can archive deduped files from a PowerScale cluster to cloud storage. However, undeduped files are created when those files are recalled from the cloud to the PowerScale cluster. For more information about deduplication within OneFS, see the *Next-Generation Storage Efficiency with Dell PowerScale SmartDedupe* white paper.
- **Small file storage efficiency (SFSE):** CloudPools and SFSE cannot work together. For PowerScale clusters using CloudPools, any SmartLink files cannot be containerized or packed. It is best practice to not archive small files that will be optimized using SFSE. The efficiencies gained from implementing SFSE for small files, outweigh the storage advantages gained from archiving them to the cloud using CloudPools. For more information about the Small File Storage Efficiency feature of OneFS, see the *Dell PowerScale OneFS Storage Efficiency* white paper.
- **Network proxy:** When a PowerScale cluster cannot connect to the CloudPools storage target directly, network proxy servers can be configured for an alternate path to connect to the cloud storage.
- **SmartConnect:** If users access SmartLink files regularly through a specific node, clogging the inline access path might affect client performance. You can configure PowerScale SmartConnect for load-balancing connections for the cluster. For more information about SmartConnect, see the *Dell PowerScale Network Design Considerations* white paper.

- **Cloud storage account:** Do not delete a cloud storage account that is in use by archived files. Any attempt to open a SmartLink file associated with a deleted account will fail. In addition, NDMP backup and restore and SyncIQ failover and failback will fail when a cloud storage account has been deleted.
- **Cloud objects and data retention:** Cloud objects are crucial for SmartLink files. Any attempt to open a SmartLink file associated with deleted cloud objects will fail. OneFS checks data retention and the reference count for cloud objects before garbage collection. When data retention has expired and there is no reference count for cloud objects, cloud objects are deleted through garbage collection. Data retention is a concept used to determine the Date of Death (DoD) setting for objects that support a SmartLink file. DoD is used to trigger garbage collection only if the reference count is zero for a file on the cluster only. The reference count is a concept used to determine whether cloud objects are associated with SmartLink files, including SmartLink files in the snapshots, SyncIQ backup, and NDMP backup. The considerations include:
  - Data retention periods include cloud data retention period, incremental backup retention period for NDMP incremental backup and SyncIQ, and full backup retention period for NDMP only. If more than one period applies to a SmartLink file, the longest period is applied.
  - If a SmartLink file is unchanged through multiple SyncIQ backups or NDMP backups, its data retention remains unchanged.
  - Data retention is set or updated on any event that changes the backed-up version of a file or the state of the SmartLink file.
  - If a SmartLink file is changed and incrementally backed up, its data retention is set by calculating the current time plus incremental backup retention period.
  - If a SmartLink file is recalled, the reference count is removed, and its data retention are set by calculating the current time plus cloud data retention period. Its cloud objects are deleted through garbage collection after its data retention has expired.
  - If a SmartLink file is deleted, its data retention is set by calculating the current time plus cloud data retention period. If cloud objects are still associated with snapshots, SyncIQ backup, or NDMP backup, its cloud objects are not deleted through garbage collection after its data retention has expired.
- **OneFS upgrade (CloudPools 1.0 to CloudPools 2.0):** Before beginning the upgrade, check the OneFS CloudPools upgrade path shown in Table 3.

Table 3. OneFS CloudPools upgrade path

Installed OneFS version (CloudPools 1.0)	Upgrade to OneFS version (CloudPools 2.0)			
	8.2.0	**8.2.1 with May 2020 RUPs**	**8.2.2 with May 2020 RUPs**	**9.x**
8.0.x or 8.1.x	Strongly discouraged	OK if needed but 8.2.2 recommended	Strongly recommended	Strongly recommended

Note: Contact your Dell representative if you plan to upgrade OneFS to 8.2.0.

In a SyncIQ environment with unidirectional replication, the SyncIQ target cluster should be upgraded before the source cluster. The reason is that OneFS allows the CloudPools-1.0-formatted SmartLink files to be converted into CloudPools-2.0-formatted SmartLink files through a post-upgrade SmartLink conversion process. Otherwise, SyncIQ policy needs to be reconfigured to **deep copy**, but deep copy causes archived file content to be read from the cloud and replicated. In a SyncIQ environment with bi-directional replication, recommended actions are to disable SyncIQ on both source and target clusters and upgrade both source and target clusters simultaneously. You can then reenable SyncIQ on both source and target clusters once the OneFS upgrades have been committed on both source and target clusters. Depending on the number of SmartLink files on the target DR cluster and the processing power of that cluster, the SmartLink conversion process can take considerable time.

Note: There is no need to stop SyncIQ and Snapshot during the upgrade in a SyncIQ environment with unidirectional replication. SyncIQ must resynchronize all converted stub files, it might take SyncIQ some time to catch up with all the changes.

To check the status of the SmartLink upgrade process, run the following command, substituting the appropriate job number:

```
# isi cloud job view 6
              ID: 6
              Description: Update SmartLink file formats
              Effective State: running
             Type: smartlink-upgrade
             Operation State: running
             Job State: running
             Create Time: 2019-08-23T14:20:26
          State Change Time: 2019-09-17T09:56:08
          Completion Time: -
          Job Engine Job: -
          Job Engine State: -
          Total Files: 21907433
          Total Canceled: 0
          Total Failed: 61
          Total Pending: 318672
          Total Staged: 0
          Total Processing: 48
          Total Succeeded: 21588652
```

Note: CloudPools recall jobs will not run while SmartLink upgrade or conversion is in progress.

For Not All Nodes on Network (NANON) cluster, getting the unconnected nodes connected to the network before starting the SmartLink conversion is recommended. Also, you need disable SnapDelete until the SmartLink conversion is completed.

Microsoft Azure configuration

Before PowerScale CloudPools is configured on the PowerScale cluster, Microsoft Azure must be configured properly. General considerations and best practices for configuring Microsoft Azure for CloudPools include:

- **URI for CloudPools:** The blob endpoint is used as the URI for CloudPools. For example, if your general-purpose storage account is named `mystorageaccount`, the default blob endpoint for CloudPools is https://mystorageaccount.blob.core.windows.net. The endpoint is the URI for CloudPools.
- **Azure storage account:** Azure storage supports several types of storage accounts. For more details, see Azure storage account overview on the Microsoft website. CloudPools supports blob storage, hot access tiers. CloudPools does not support cold blobs.

Protecting SmartLink files

SmartLink files are the sole means to access file data stored in Microsoft Azure, so protecting them from accidental deletion is important.

This section discusses using PowerScale SyncIQ and NDMP to back up SmartLink files.

Note: SmartLink files cannot be backed up using a copy command, such as secure copy (scp).

SyncIQ

SyncIQ is CloudPools-aware, but consider the snapshot-efficiency guidance in Sparse files handling, especially where snapshot retention periods on the target cluster will be long.

SyncIQ policies support two types of data replication for CloudPools:

- **Shallow copy:** This option is used to replicate files as SmartLink files without file data from source PowerScale cluster to target PowerScale cluster.
- **Deep copy:** This option is used to replicate files as regular files or unarchived files from source PowerScale cluster to target PowerScale cluster.

For information about cross-version compatibility of CloudPools, see NDMP and SyncIQ support.

SyncIQ, SmartPools, and CloudPools licenses are required on both the source and target PowerScale cluster. It is highly recommended to setup a scheduled SyncIQ backup of the SmartLink files. For more information about PowerScale SyncIQ, see the *Dell PowerScale SyncIQ: Architecture, Configuration, and Considerations* white paper.

When SyncIQ replicates SmartLink files, it also replicates the local cache state and unsynchronized cache data from the source PowerScale cluster to the target PowerScale cluster. Figure 11 shows the SyncIQ replication when replicating directories including SmartLink files and unarchived normal files. Both unidirectional and bi-directional replication are supported. Appendix A: Step-by-step configuration example provides steps for failing over to a secondary PowerScale cluster and failing back to a primary PowerScale cluster.

Note: OneFS manages cloud access at the cluster level and does not support managing cloud access at the directory level. You need to remove cloud access on the source cluster and add cloud access on the target cluster when failing over a SyncIQ directory containing SmartLink files

to a target cluster. If there are multiple CloudPools storage accounts, removing or adding cloud access affects all CloudPools storage accounts on the source or target cluster.

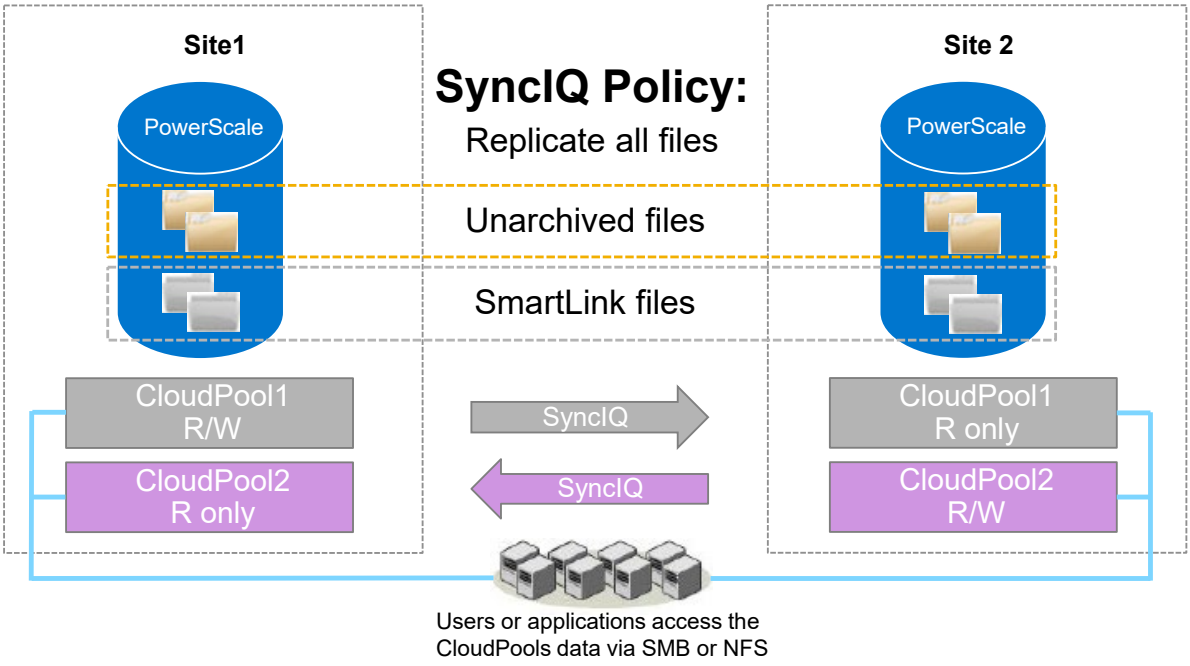

Figure 11. SyncIQ replication

Note: If encryption is enabled in a file pool policy for CloudPools, SyncIQ also replicates all the relevant encryption keys to the secondary PowerScale cluster along with the SmartLink files.

NDMP

NDMP is also CloudPools-aware and supports three backup and restore methods for CloudPools:

- **DeepCopy:** This option is used to back up files as regular files or unarchived files. Files can only be restored as regular files.
- **ShallowCopy:** This option is used to back up files as SmartLink files without file data. Files can only be restored as SmartLink files.
- **ComboCopy:** This option is used to back up files as SmartLink files with file data. Files can be restored as regular files or SmartLink files.

For information about cross-version compatibility of CloudPools, see NDMP and SyncIQ support.

It is possible to update the file data and send the updated data to the cloud storage. Multiple versions of SmartLink files can be backed up to tapes using NDMP, and multiple versions of CDOs are protected on Microsoft Azure under the data retention setting. You can restore a specific version of a SmartLink file from tapes to a PowerScale cluster and continue to access (read or update) the file like before.

Note: If encryption is enabled in the file pool policy for CloudPools, NDMP also backs up all the relevant encryption keys to tapes along with the SmartLink files.

Performance

CloudPools is designed to move cold data from primary storage to the cloud. It is deliberately slow to ensure that it does not compete with things that are performance sensitive like SMB and NFS user activity. By default, CloudPools is using 10 threads per node, which balances CloudPools CPU usage with other cluster functions. It is recommended to use the default number of threads for typical workloads. CloudPools does provide an option to modify the number of archive and recall threads. However, modifying the number of archive and recall threads can improve archive and recall performance but can also have significant impact on the CPU load of your system.

Note: Contact your Dell representative if you want to configure higher number of threads.

CloudPools archive and recall performance are highly dependent upon many factors, such as the network bandwidth between the PowerScale cluster and the cloud, available system resources, and file size. These performance considerations include:

- As the file size increases, the CloudPools archive and recall performance also increases. However, the effect on archive performance is minor when file size is greater than or equal to 10 MB. The effect on recall performance is negligible when file size is greater than or equal to 10 MB.
- As the thread counts increase, the CloudPools archive and recall performance also increases. However, the effect on archive and recall performance is negligible when the number of threads per PowerScale node is greater than or equal to 40.
- For a single large file, the effect on archive and recall performance is negligible regardless of the file size or number of threads per node. A single thread manages a single file transfer on a single node.
- Starting from OneFS 9.3.0.0, CloudPools creates Likewise sparks to drive the read of each CDO from the cloud. This enhancement can cache multiple chunks or CDOs of a stub file concurrently to improve CloudPools read and recall performance.
- The effect on archive and recall performance is negligible regardless of number of file pool policies or jobs.
- With the expansion of PowerScale nodes, CloudPools archive and recall performance increases, but not linearly.
- For a single, heterogeneous cluster, tier 1 (fast) node pool has a minor impact on CloudPools archive performance and a large impact on CloudPools recall performance. The archive and recall performance are better when data is stored in the tier 1 node pool. The setting **Data Storage Target** of a file pool policy can determine the node pool for recall. However, the node pool cannot be changed for inline read. The node pool for a stub is used for inline read for this stub.
- Not All Nodes on Network (NANON) cluster has a large impact on CloudPools archive and recall performance.

Reporting

Introduction

This section describes reporting for CloudPools network stats and includes the following topics:

- CloudPools network stats
- Query network stats by CloudPools account
- Query network stats by file pool policy
- Query history network stats

CloudPools network stats

CloudPools network stats collect every network transaction and provide network activity statistics from connections to the cloud storage. The network activity statistics include bytes in, bytes out, and the number of GET, PUT, and DELETE operations. CloudPools network stats are available in two categories:

- Per CloudPools account
- Per file pool policy

Note: CloudPools network stats do not provide file statistics, such as the file list being archived or recalled.

Query network stats by CloudPools account

Run the following command to check the CloudPools network stats by CloudPools account:

```
isi_test_cpool_stats -Q --accounts <account_name>
```

Figure 12 shows an example of current CloudPools network stats by CloudPools account.

```
hop-isi-p-1# isi_test_cpool_stats -Q --accounts testaccount
Account Name    Bytes In        Bytes Out       Num Reads       Num Writes      Num Deletes
------------------------------------------------
testaccount     4194896000      4194905034      4000    2001    8001
------------------------------------------------
```

Figure 12. Network stats by CloudPools account

Query network stats by file pool policy

Run the following command to check the CloudPools network stats by file pool policy:

```
isi_test_cpool_stats -Q --policies <policy_name>
```

Figure 13 shows an example of current CloudPools network stats by file pool policy.

```
hop-isi-p-1# isi_test_cpool_stats -Q --policies testpolicy
Policy Name     Bytes In        Bytes Out       Num Reads       Num Writes
------------------------------------------------
ecspolicy       4194896000      4194905034      4000    2001
------------------------------------------------
```

Figure 13. Network stats by file pool policy

Note: The command output does not include the number of deletes by file pool policy.

Query history network stats

Run the following command to check the history CloudPools network stats:

```
isi_test_cpool_stats -q -s <number of seconds in the past to start stat query>
```

Use the `s` parameter to define the number of seconds in the past. For example, set it as 86,400 to query CloudPools network stats over the last day.

Figure 14 shows an example of CloudPools network stats over the last day.

```
hop-isi-p-1# isi_test_cpool_stats -q -s 86400
Account         bytes-in        bytes-out       gets    puts    deletes
testaccount             | 4194896000            | 4194905034            | 4000  | 2001  | 8001
```

Figure 14. Network stats last day

Run the following command to flush stats from memory to database and get the latest CloudPools history network stats:

```
isi_test_cpool_stats -f
```

Cloud statistics namespace with CloudPools

The cloud statistics namespace with CloudPools is added in OneFS 9.4.0.0. This feature leverages existing OneFS daemons and systems to track statistics about CloudPools activities. The statistics include bytes In, bytes Out, and the number of Reads, Writes, and Deletions. CloudPools statistics are available in two categories:

- Per CloudPools account
- Per file pool policy

Note: The cloud statistics namespace with CloudPools does not provide file statistics, such as the file list being archived or recalled.

You can run the `isi statistics cloud` command to view statistics about CloudPools activities. For more information about the `isi statistics cloud` command, see the *PowerScale OneFS 9.4.0.0 CLI Command Reference*.

Commands and troubleshooting

Commands

This CloudPools operations and job monitoring commands discussed in this section include:

- CloudPools archive
- CloudPools recall
- CloudPools monitoring

CloudPools archive

Run the following command to archive files from a PowerScale cluster to the cloud on demand:

```
isi cloud archive <file name> --recursive [true | false] --policy <policy name>
```

Parameters:

- **<file name>:** File name to be archived
- **--recursive:** Whether the archive should apply recursively to nested directories
- **--policy:** Policy name to be used with archiving

Run either of the following two commands to check whether the file is a SmartLink file, as shown in Figure 15.

```
ls -loh <file name>

isi get -DD <file name> | grep -i smartlink
```

```
hop-isi-n-1# ls -loh test01.mp3
-rwx------ +   1 root  wheel  uarch,inherit,writecache,wcinherit,ssmartlinked,shasntfsacl  339K Mar 14 0
5:20 test01.mp3
hop-isi-n-1# isi get -DD test01.mp3 | grep -i smartlink
*  SmartLinked:        True
hop-isi-n-1#
```

Figure 15. SmartLink file

CloudPools recall

Run the following command to recall files from the cloud to a PowerScale cluster on demand:

```
isi cloud recall <files> --recursive [true | false]
```

Parameters:

- **<file name>:** File name to be archived
- **--recursive:** Whether the archive should apply recursively to nested directories

CloudPools job monitoring

To check the CloudPools job status, run the following command:

```
isi cloud jobs list
```

To check the archive or recall file list status for a specific CloudPools job, run the following command. As shown in Figure 16, the job ID can be found by using the command `isi cloud jobs list`.

```
isi cloud jobs files list <job id>
```

```
hop-isi-n-2# isi cloud jobs files list 219
Name                                State
-----------------------------------------
/ifs/ecs/InsightIQ_b.4.1.2.7.zip completed
/ifs/ecs/a.pptx                  completed
-----------------------------------------
Total: 2
```

Figure 16. File list of specific CloudPools job

Note: The output of the prior command only shows the file name and state for specific CloudPools job.

To perform additional actions, run the following commands:

- Pause a CloudPools job:

  ```
  isi cloud jobs pause <job id>
  ```

- Resume a paused CloudPools job:

  ```
  isi cloud jobs resume <job id>
  ```

- Cancel a CloudPools job:

  ```
  isi cloud jobs cancel <job id>
  ```

- Check the file list state of writing updated data to the cloud (job id is 1), which is an internal CloudPools job and always running:

  ```
  isi cloud jobs files list 1
  ```

Note: The CloudPools system jobs should not be paused except temporarily for troubleshooting. No jobs should be left paused for an indefinite time.

Troubleshooting

This section describes various CloudPools troubleshooting methodologies, which include:

- CloudPools state
- CloudPools logs

CloudPools state

To check the CloudPools storage account state, run the following command:

```
isi cloud accounts view <cloudpools storage account name>
```

To check the CloudPool state, run the following command:

```
isi cloud pools view <cloud pool name>
```

To check the file pool policy state, run the following command:

```
isi filepool policies view <filepool policy name>
```

CloudPools logs

Check the CloudPools log if needed. The location of CloudPools log is as follows:

- Most normal daemon log is at `/var/log/isi_cpool_d.log`
- The log of IO to the cloud is at `/var/log/isi_cpool_io_d.log`
- Key management log is at `/var/log/isi_km_d.log`
- CloudPools job (Job Engine) log is at `/var/log/isi_job_d.log`

Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

The Dell Technologies Info Hub provides expertise that helps to ensure customer success on Dell storage platforms.

The following list provides links to documents and other assets that are referenced in this paper along with other resources that might be helpful in deployment of CloudPools on PowerScale:

- OneFS CloudPools Administration Guide
- OneFS Technical Overview
- Next-Generation Storage Efficiency with Dell PowerScale SmartDedupe
- Dell PowerScale OneFS Storage Efficiency
- Dell PowerScale SyncIQ: Architecture, Configuration, and Considerations
- High Availability and Data Protection with Dell PowerScale Scale-out NAS
- Storage Quota Management and Provisioning with Dell PowerScale SmartQuotas
- PowerScale Non-Disruptive Upgrade (NDU) Best Practices
- Data Protection with Dell PowerScale SnapshotIQ
- Dell PowerScale: Network Design Considerations
- Microsoft Azure

Appendix A: Step-by-step configuration example

Introduction

This section describes a step-by-step configuration example for CloudPools and Microsoft Azure and includes the following topics:

- Microsoft Azure
- PowerScale configuration
- SmartLink files and cloud data protection

Microsoft Azure configuration

This section describes the Microsoft Azure configuration for CloudPools.

The Microsoft Azure configuration example is a general guide for when Microsoft Azure is used for CloudPools. It does not cover all details of Microsoft Azure configuration for other use cases. For more information about Azure configuration, see the Microsoft Azure documentation.

1. Ensure your Azure account is working properly.
2. Log in to the Azure portal using your own username and password.
3. In the Azure portal, follow the document Create a storage account on the Azure website to create a storage account.
4. Before configuring CloudPools, use the tool Azure Storage Explorer from Microsoft to get the URI (Blob Endpoint), username (Account Name), and passkey (Primary Key) for CloudPools, shown in Figure 17.

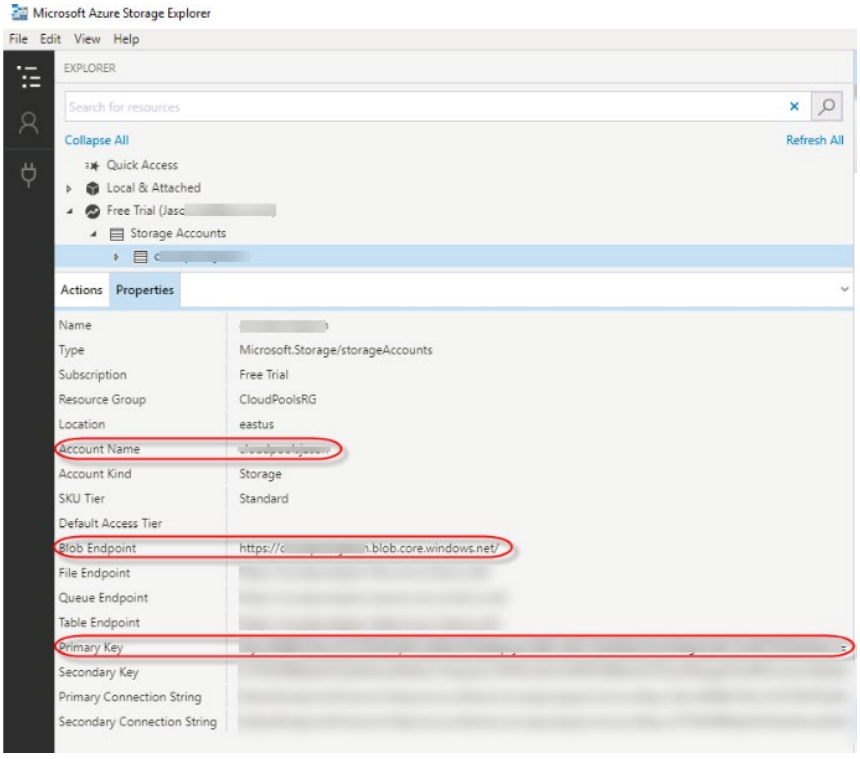

Figure 17. Azure Storage Account Information

Now all Microsoft Azure information is gathered for CloudPools.

CloudPools configuration

This section describes the CloudPools configuration on a PowerScale cluster, which includes:

- Verifying license
- Creating cloud storage account
- Creating CloudPool
- Creating file pool policy
- Running SmartPools job for CloudPools
- Creating SyncIQ policy

Verifying license

This section describes how to verify licensing on the PowerScale system.

1. Log in to the OneFS WebUI and go to **Cluster management** > **Licensing**, as shown in Figure 18.

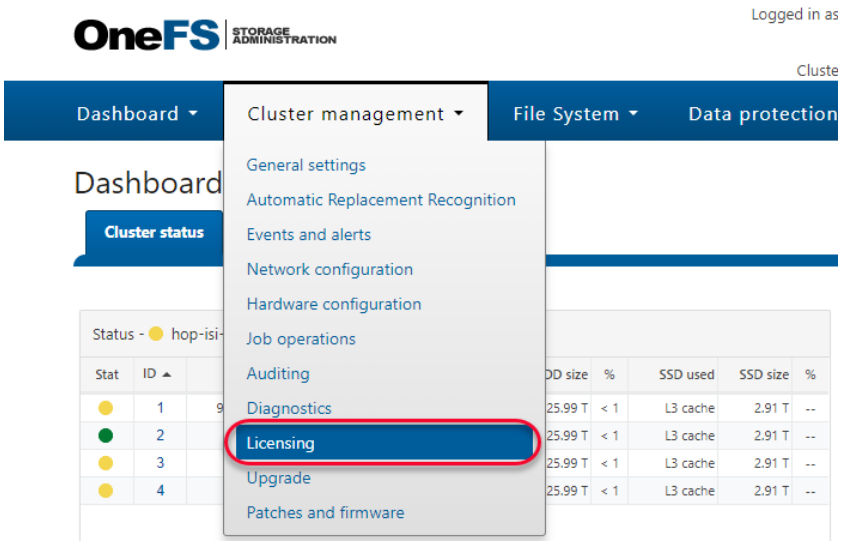

Figure 18. Verifying licenses

2. Verify that the CloudPools and SmartPools license status is **Activated**.

Creating cloud storage account

This section describes how to create a cloud storage account on the PowerScale cluster.

1. Log in to the OneFS WebUI and go to **File System** > **Storage Pools**. Click **CloudPools**, as shown in Figure 19.

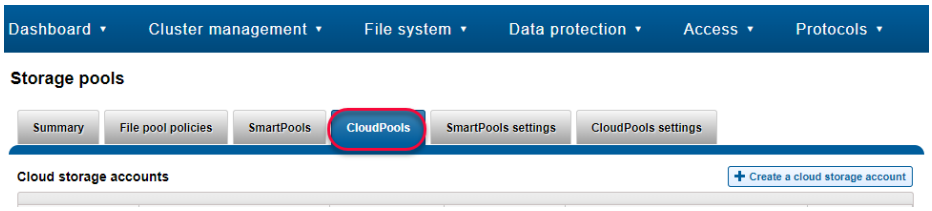

Figure 19. CloudPools

2. On the **Create a Cloud Storage Account** page, click **+ Create a Cloud Storage Account**, as shown in Figure 20. The minimum information for CloudPools and Microsoft Azure is as follows:
   - **Name or alias**: Type a name to identify the cloud storage account.
   - **Type**: Select Microsoft Azure.
   - **URI**: Type the URI to connect Microsoft Azure.
   - **User name (account name)**: Type the account name gathered on the Microsoft Azure portal.
   - **Key (access key)**: Type the access key gathered on the Microsoft Azure portal.

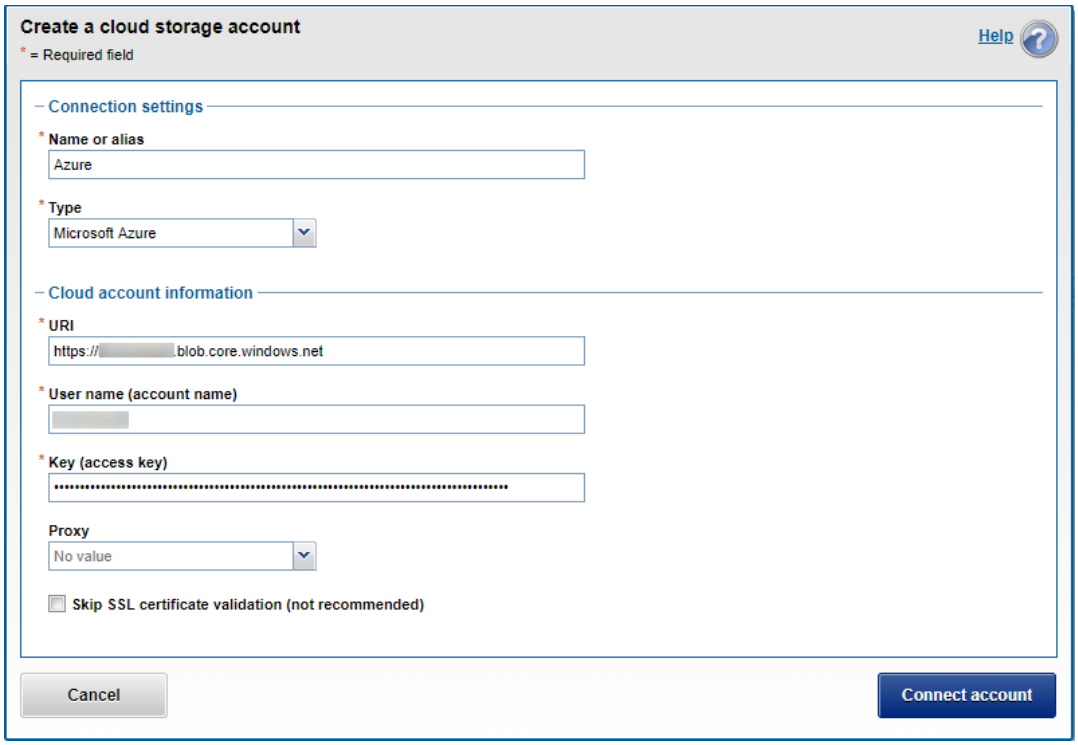

Figure 20. Create a cloud storage account

3. Click **Connect account** to create a cloud storage account. This operation results in two buckets being created in Microsoft Azure. One bucket will start with a **d** as a container to store the CDOs, and the other will start with an **m** as a container to store the associated metadata.

Creating CloudPool

This section describes how to create a CloudPool for Microsoft Azure on the PowerScale cluster.

1. Log in to the OneFS WebUI and go to **File System** > **Storage Pools.** Click **CloudPools**, as shown in Figure 19.
2. On the **Create a CloudPool** page, click **+ Create a CloudPool**, as shown in Figure 21. The minimum information is as follows:
   - **Name**: Type a name to identify the CloudPool.

- **Type**: Select Microsoft Azure.
- **Account in CloudPool**: Select the cloud storage account.

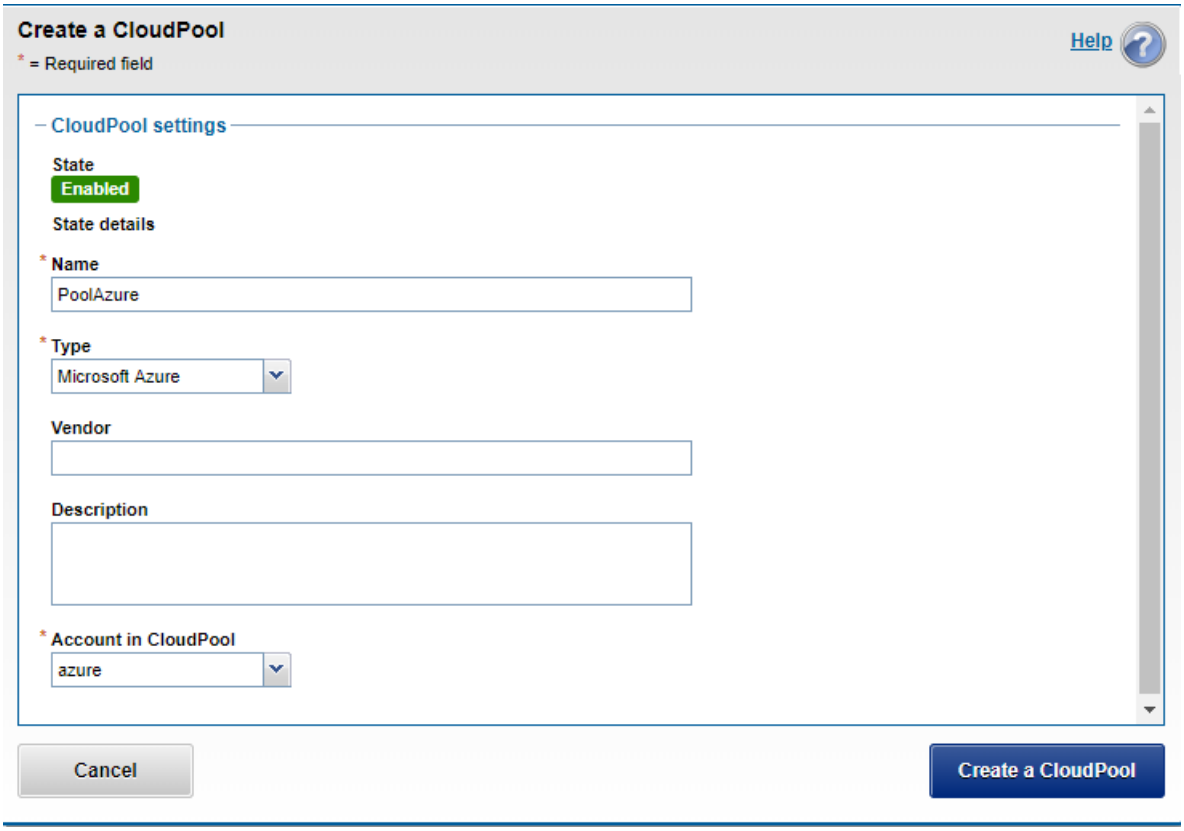

Figure 21. Create a CloudPool

3. Click **Create a CloudPool** to create a CloudPool.

Creating file pool policy

Create a file pool policy on the PowerScale cluster as follows:

1. Log in to the OneFS WebUI and go to **File System** > **Storage Pools**. Click **File Pool Policies**, as shown in Figure 22.

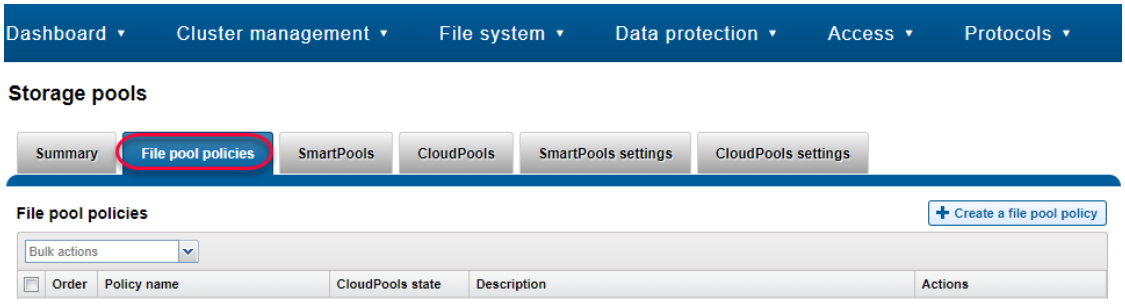

Figure 22. Create a file pool policy

2. On the **Create a file pool policy** page, click **+ Create a File Pool Policy**, as shown in Figure 23 and Figure 24. Provide the following information, at a minimum:
   - **Policy Name**: Type a name to identify the file pool policy.
   - **File Matching Criteria**: Define a logical group of files for CloudPools. See file matching criteria.
   - **Move to cloud storage**: Select the specific CloudPool as the CloudPool storage target.
   - **Data retention settings**: Set the data retention as your own. See Data retention.

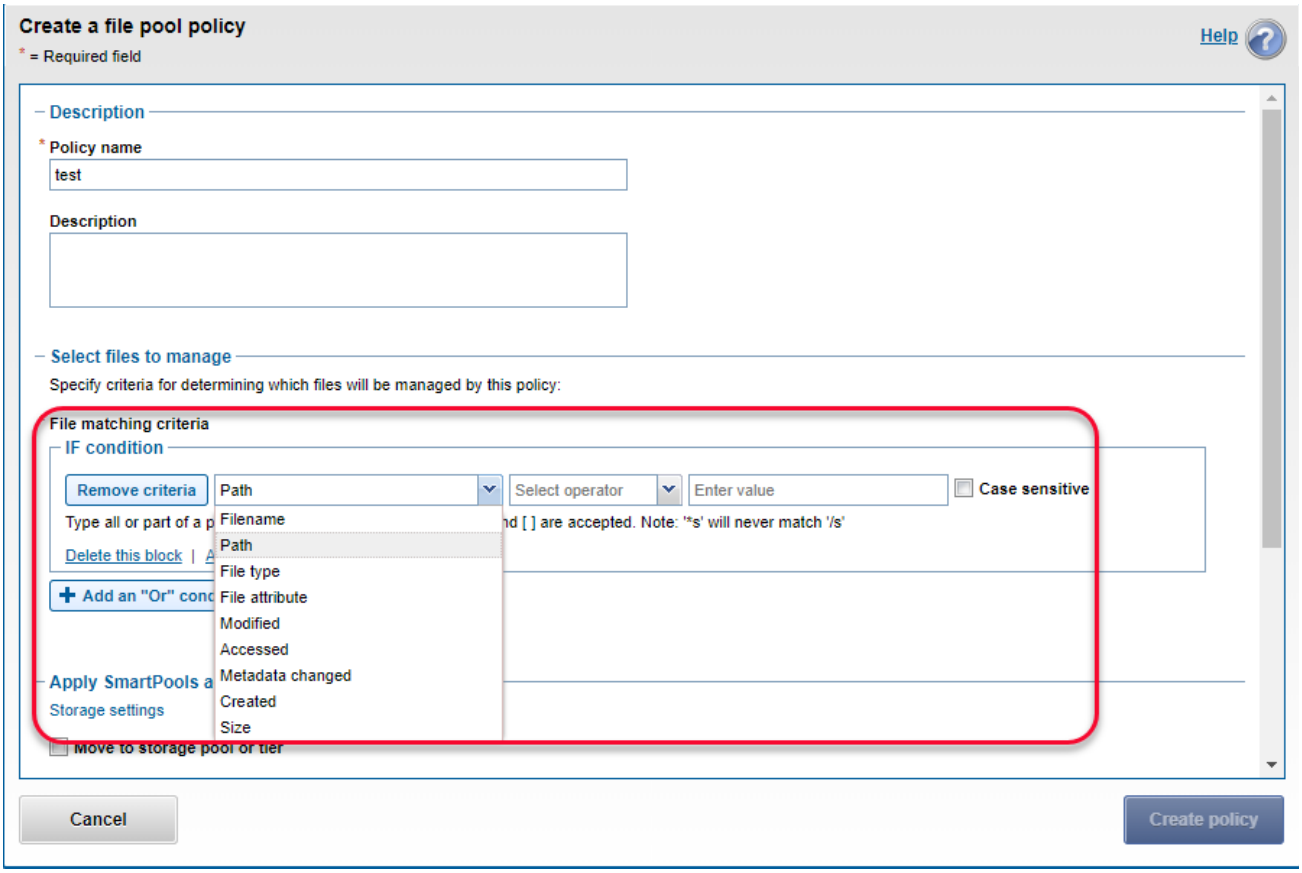

Figure 23. Create a file pool policy

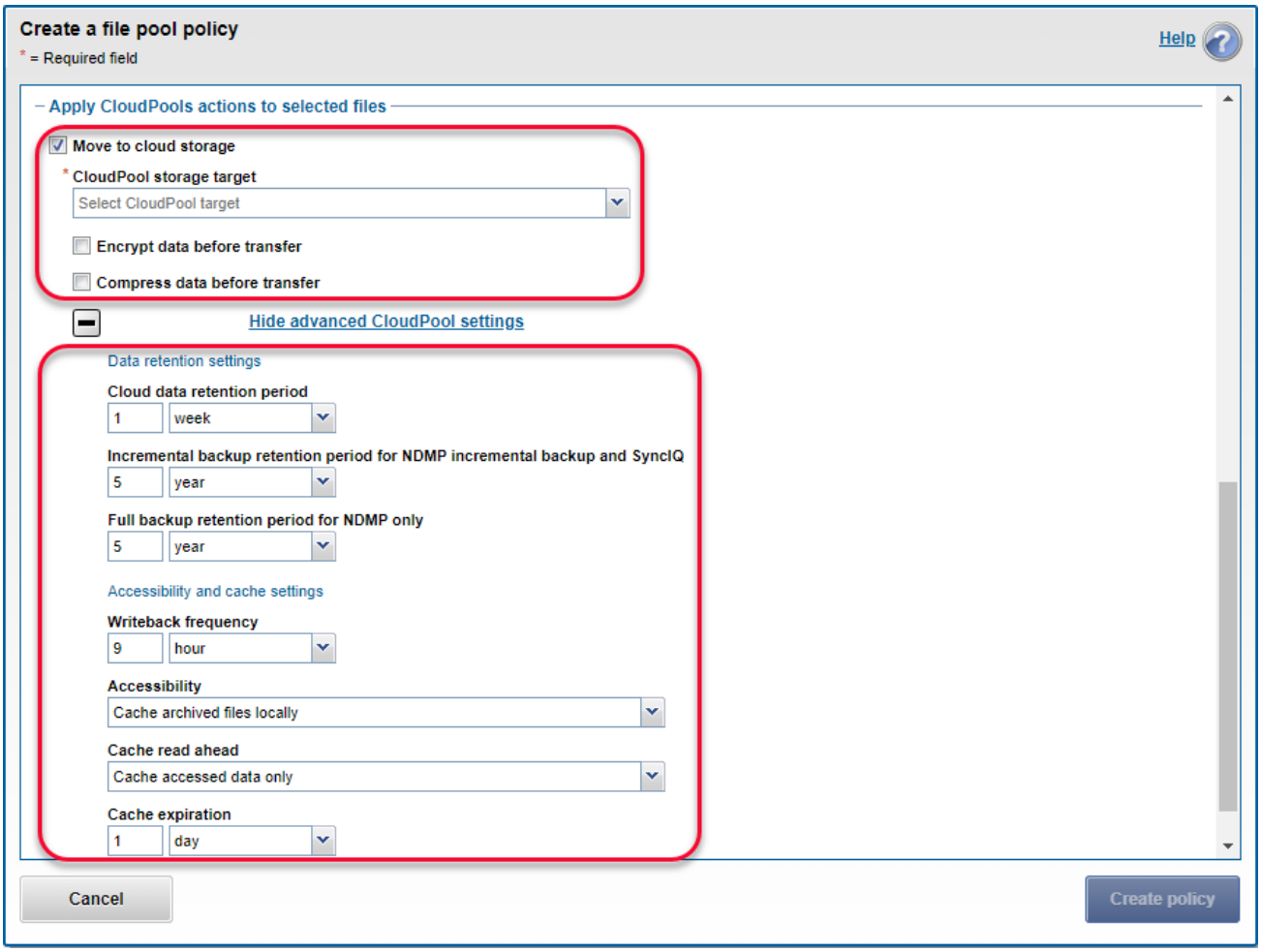

Figure 24. Create a file pool policy (continued)

3. Click **Create policy** to create a file pool policy.

Running SmartPools job for CloudPools

Run a SmartPools job for CloudPools on the PowerScale cluster as follows:

1. Log in to the OneFS WebUI and go to **Cluster management** > **Job operations**. Click **Job types**, as shown in Figure 25.

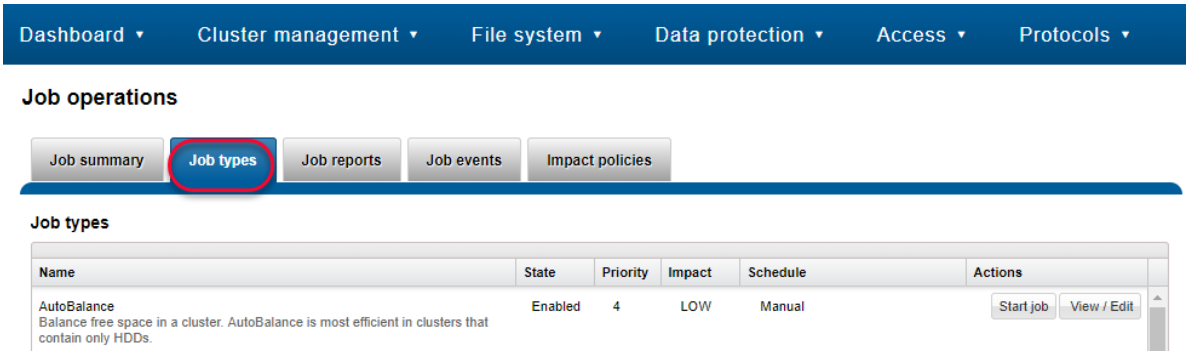

Figure 25. Job types

2. Select the **SmartPools** item and click **Edit**, as shown in Figure 26.

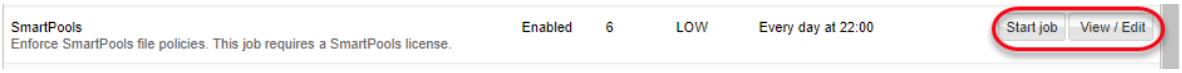

Figure 26. SmartPools job

3. From the **Edit job type details** page, as shown in Figure 27, you can:
   - Enable or disable the job
   - Set the priority of the job
   - Set the impact policy
   - Set the job schedule as manual or scheduled

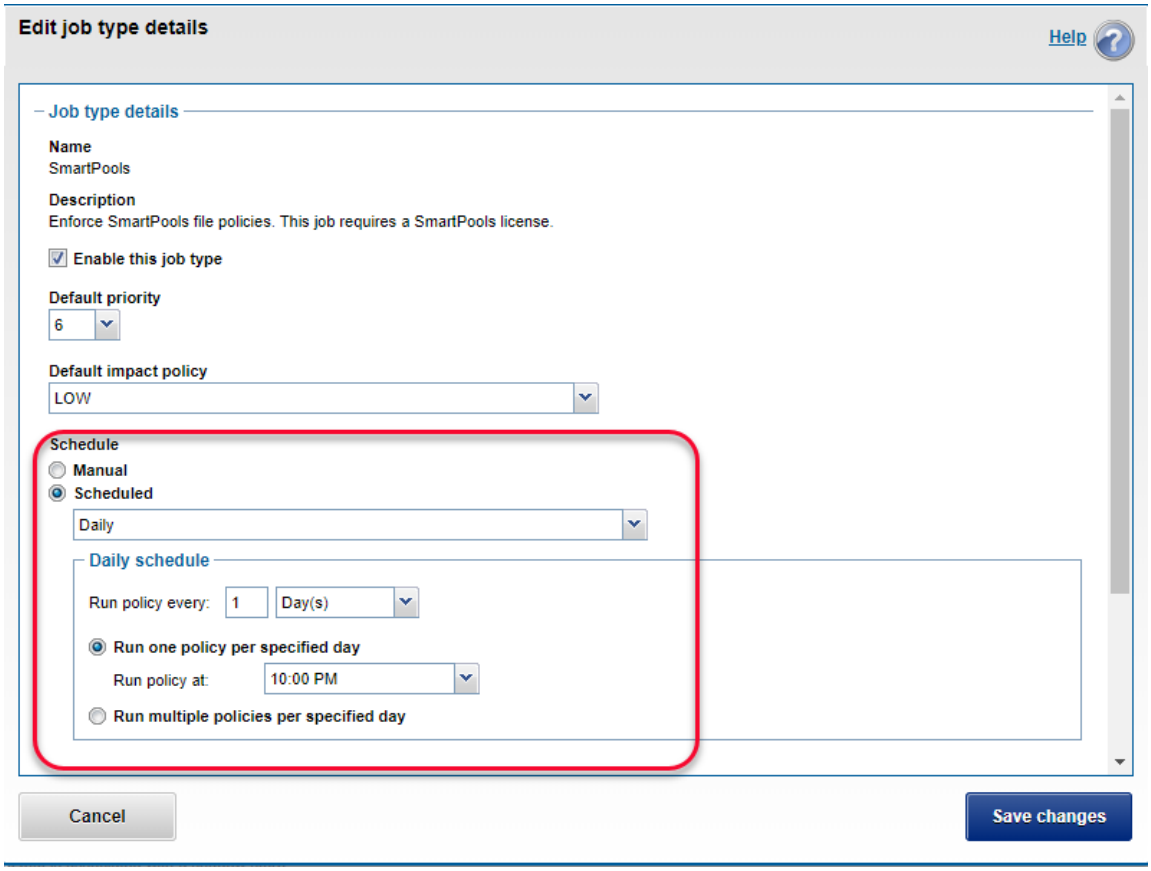

Figure 27. Edit job type details

4. Click **Start job**, as shown in Figure 26, to run the file pool policy to archive files from the PowerScale cluster to Microsoft Azure. If you want to start a specific file pool policy job manually, see Commands and troubleshooting.

Creating SyncIQ policy

Create a SyncIQ policy on the PowerScale cluster as follows:

1. Log in to OneFS WebUI and go to **Cluster Management** > **Licensing**, as shown in Figure 18. Verify that the CloudPools, SmartPools, and SyncIQ license status are **Activated**.

2. Go to **Data Protection** > **SyncIQ** > **Policies** and click **+ Create a SyncIQ policy**, as shown in Figure 28 and Figure 29. Provide the following information, at a minimum:
   - **Policy name**: Type a name to identify the policy name.
   - **Source root directory**: Type the directory name from source PowerScale cluster you want to replicate to the target PowerScale cluster.
   - **Target host**: Type the IP or name of the target PowerScale cluster.
   - **Target directory**: Type the directory name from the target PowerScale cluster you want to store the data replicated from the source PowerScale cluster.
   - **Deep copy for CloudPools**: Select the type you want to use.

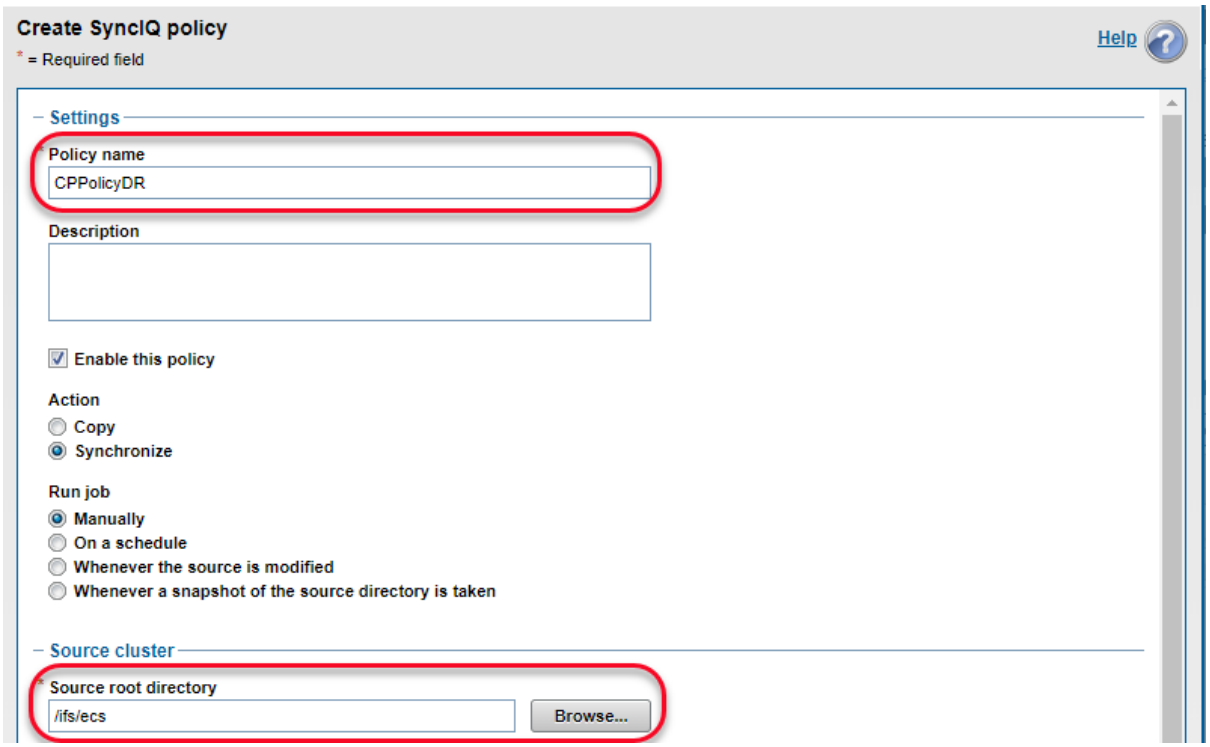

Figure 28. Create SyncIQ policy

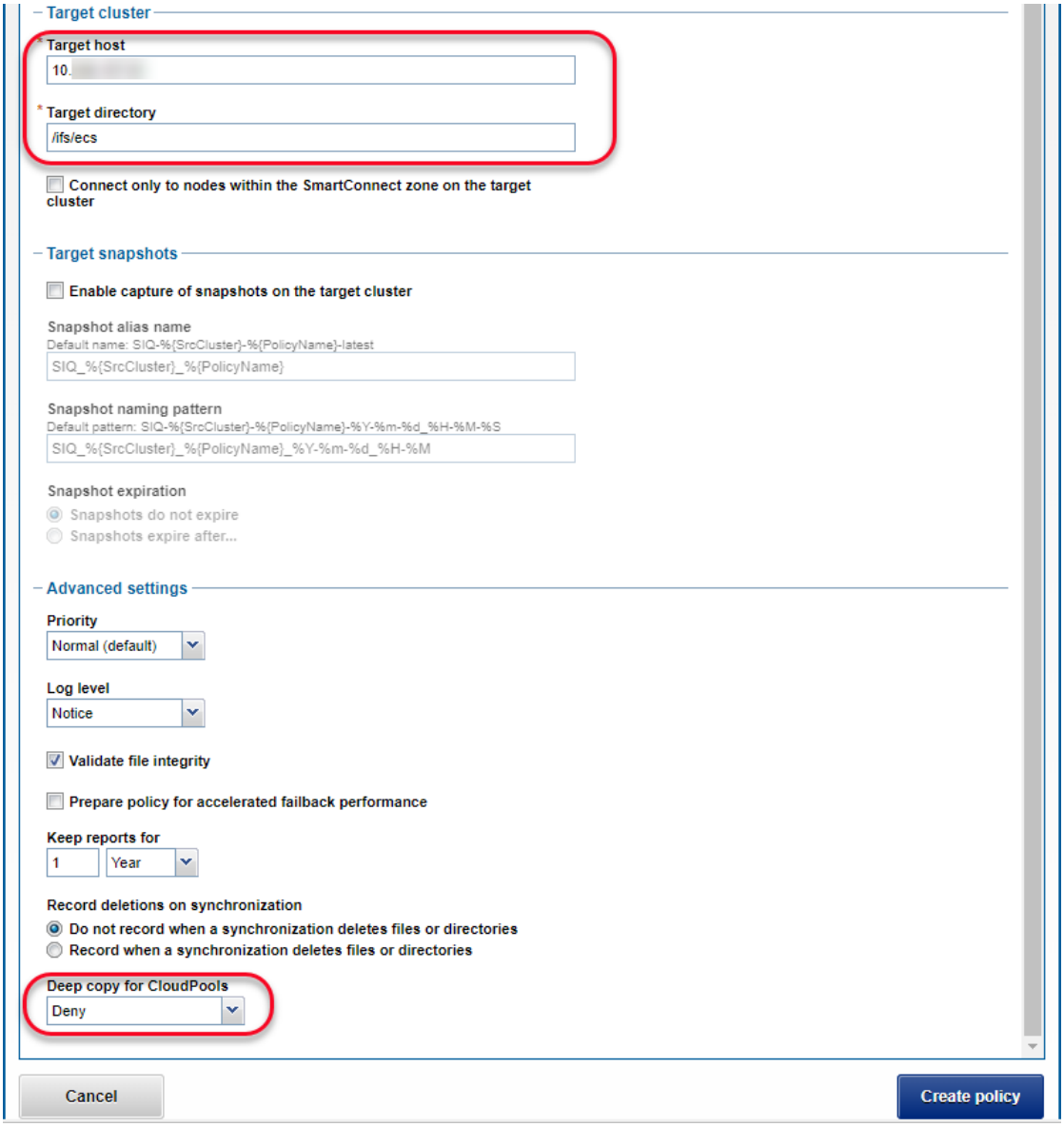

Figure 29. Create SyncIQ policy (continued)

3. Click **Create policy** to create a SyncIQ policy.

SmartLink files protection

This section provides an example of how to protect SmartLink files and cloud data. Ensure that you have already configured SyncIQ on the PowerScale clusters, including:

- Fail over to the secondary PowerScale cluster
- Fail back to the primary PowerScale cluster

Fail over to the secondary PowerScale cluster

The following steps are required to fail over to the secondary PowerScale cluster:

1. Log in to the *secondary* OneFS WebUI and go to **Data Protection** > **SyncIQ.** Click **Local Targets** on the policy that you want to fail over and select **More** > **Allow Writes**, as shown in Figure 30. This operation grants read/write access to the data

on the primary PowerScale cluster being replicated to the secondary PowerScale cluster.

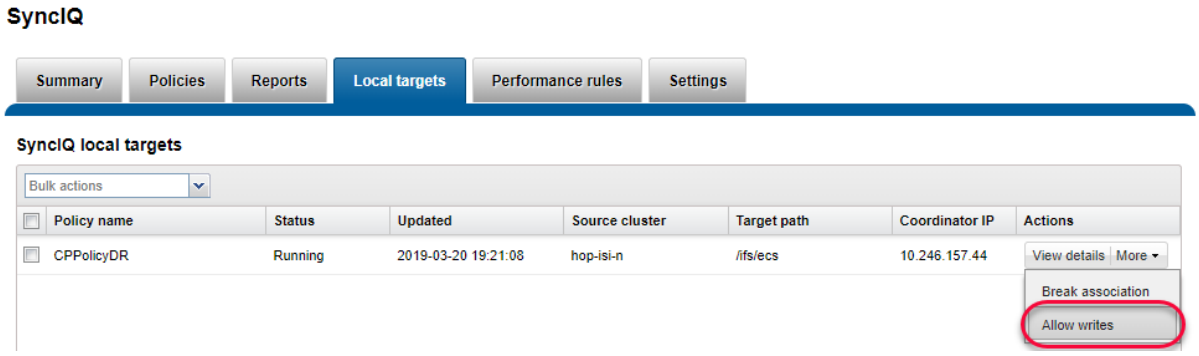

Figure 30. Allow writes on secondary cluster

Note: If the primary PowerScale cluster is still online, stop all writes to the replication policy's directory.

2. Check and change cloud access. Log in to the PowerScale clusters using SSH. To identify the CloudPools GUID, run the commands `isi cloud access list` and `isi cloud access view <GUID>`. Figure 31 shows the cloud access status on the secondary PowerScale cluster.

```
hop-isi-p-1# isi cloud access list
Name      GUID                                                Synced From  State
-------------------------------------------------------------------------------
hop-isi-n 006016894ae21826755c5a15e4a547aba6bb                hop-isi-n    not permitted
hop-isi-p 006048509d1c6325875cd003f35f88a983de (current)                   permitted
-------------------------------------------------------------------------------
Total: 2
hop-isi-p-1# isi cloud access view 006016894ae21826755c5a15e4a547aba6bb
       Name: hop-isi-n
       GUID: 006016894ae21826755c5a15e4a547aba6bb
Synced From: hop-isi-n
      State: not permitted
   Accounts: testaccount
   Policies: ecspolicy
```

Figure 31. Identify CloudPools GUID to be transferred

3. On the *primary* PowerScale cluster, remove the cloud write permission using the command `isi cloud access remove <GUID>` as shown in Figure 32. This operation disables the file pool policy, CloudPool, and cloud storage account on the primary PowerScale cluster.

```
hop-isi-n-1# isi cloud access remove 006016894ae21826755c5a15e4a547aba6bb
Removing access to 006016894ae21826755c5a15e4a547aba6bb will disable the following CloudPool accounts an
d FilePool policies:
       testaccount (CloudPool Account)
       ecspolicy (FilePool Policy)
Are you sure?? (yes/[no]): yes
hop-isi-n-1#
```

Figure 32. Remove Cloud write access on the primary PowerScale cluster

4. On the *secondary* PowerScale cluster, add the cloud write permission using the command `isi cloud access add <GUID>, as` shown in Figure 33. This

operation enables the file pool policy, CloudPool, and cloud storage account on the secondary PowerScale cluster.

```
hop-isi-p-1# isi cloud access add 006016894ae21826755c5a15e4a547aba6bb
Giving access to 006016894ae21826755c5a15e4a547aba6bb will enable the following CloudPool accounts and F
ilePool policies:
        testaccount (CloudPool Account)
        ecspolicy (FilePool Policy)
Are you sure?? (yes/[no]): yes
To ensure proper cleanup, a job must be run for each S3 enabled account to set an expiration date for al
l stale cloud files.
Failure to set an expiration date will cause leaked data in the cloud resulting in additional costs from
 cloud service providers.
Note that after the expiration date has passed, backups may no longer be able to restore deleted files.
Expiration dates can be set later using the 'isi cloud restore-coi' command.
To start expiration date jobs for applicable accounts, enter an expiration date now or 'default' to acce
pt the default expiration date (2029-03-20): (<date>/[default]/cancel):
hop-isi-p-1#
```

Figure 33. Add cloud write access on the secondary PowerScale cluster

Note: Do not allow write access to the CloudPools from more than one PowerScale cluster.

The SyncIQ failover is complete.

Fail back to primary PowerScale cluster

The following steps are required to fail back to the primary PowerScale cluster:

1. Log in to the *primary* OneFS WebUI and go to **Data Protection** > **SyncIQ**. Click **Policies** on the policy that you want to failback and select **More** > **Resync-prep**, as shown in Figure 34. This operation creates a SyncIQ replication mirror policy on the secondary PowerScale cluster.

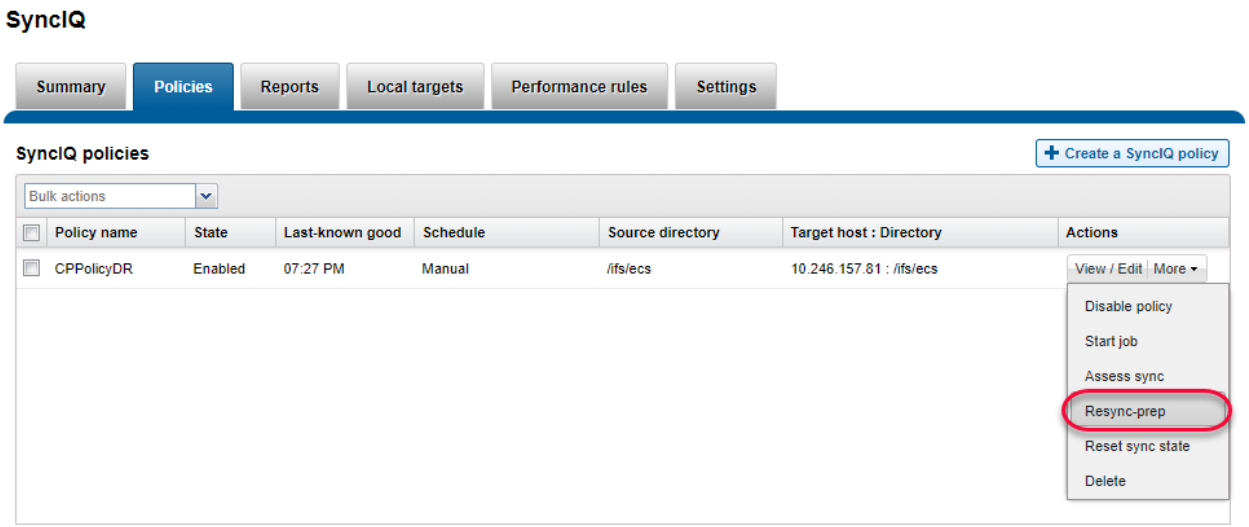

Figure 34. Resync prep SyncIQ policy on primary PowerScale cluster

2. Log in to the secondary OneFS WebUI and go to **Data Protection** > **SyncIQ**. Click **Policies** on the replication mirror policy that you want to failover and select **More** > **Start Job**, as shown in Figure 35. This operation synchronizes any changes that have been written to the secondary PowerScale cluster back to the primary PowerScale cluster.

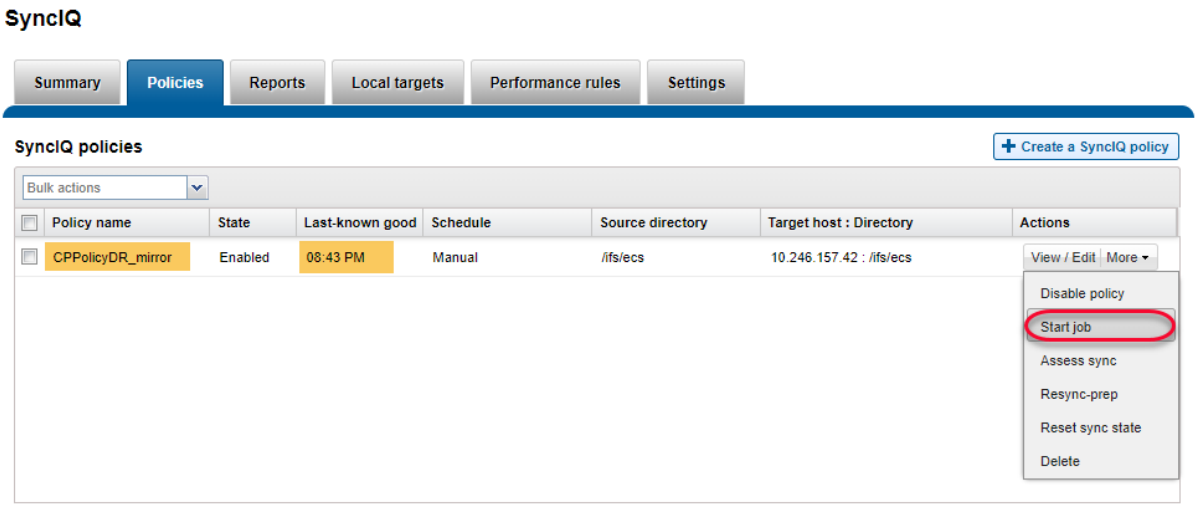

Figure 35. Sync data from secondary PowerScale cluster to primary PowerScale cluster

3. Log in to the *primary* OneFS WebUI and go to **Data Protection** > **SyncIQ**. Click **Local Targets** on the policy that you want to failover and select **More** > **Allow Writes**, as shown in Figure 36. This operation grants read/write access to the replication directory back to the primary PowerScale cluster and changes the secondary PowerScale cluster's access to this directory as read-only.

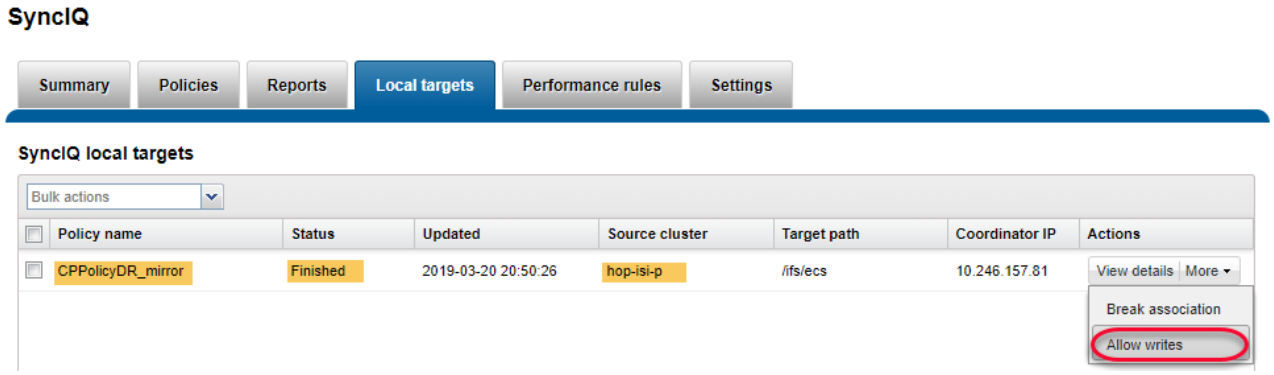

Figure 36. Allow writes on primary PowerScale cluster to SyncIQ replication directory

Note: If the secondary PowerScale cluster is still online, stop all writes to the replication policy's directory. Perform a final replication from the secondary PowerScale cluster to the primary PowerScale cluster to ensure both sites are synchronized.

4. Check and change cloud access. Log in to the PowerScale clusters using SSH. To identify the CloudPools GUID, run the commands `isi cloud access list` and `isi cloud access view <GUID>`. Figure 37 shows the cloud access status on the secondary PowerScale cluster.

```
hop-isi-p-1# isi cloud access list
Name      GUID                                                Synced From  State
-------------------------------------------------------------------------------
hop-isi-n 006016894ae21826755c5a15e4a547aba6bb                hop-isi-n    permitted
hop-isi-p 006048509d1c6325875cd003f35f88a983de (current)                   permitted
-------------------------------------------------------------------------------
Total: 2
hop-isi-p-1# isi cloud access view 006016894ae21826755c5a15e4a547aba6bb
       Name: hop-isi-n
       GUID: 006016894ae21826755c5a15e4a547aba6bb
Synced From: hop-isi-n
      State: permitted
   Accounts: testaccount
   Policies: ecspolicy
hop-isi-p-1#
```

Figure 37. Identify GUID for CloudPools account and file pool policy

5. On the *secondary* PowerScale cluster, remove the cloud write permission by running the command `isi cloud access remove <GUID>`, as shown in Figure 38. This operation disables the file pool policy, CloudPool, and a cloud storage account on the secondary PowerScale cluster.

```
hop-isi-p-1# isi cloud access remove 006016894ae21826755c5a15e4a547aba6bb
Removing access to 006016894ae21826755c5a15e4a547aba6bb will disable the following CloudPool accounts an
d FilePool policies:
       testaccount (CloudPool Account)
       ecspolicy (FilePool Policy)
Are you sure?? (yes/[no]): yes
```

Figure 38. Remove cloud write access on the secondary PowerScale cluster

6. On the *primary* PowerScale cluster, add the cloud write permission using the command `isi cloud access add <GUID>`, as shown in Figure 39. This operation enables the file pool policy, CloudPool, and cloud storage account on the primary PowerScale cluster.

```
hop-isi-n-1# isi cloud access add 006016894ae21826755c5a15e4a547aba6bb
Giving access to 006016894ae21826755c5a15e4a547aba6bb will enable the following CloudPool accounts and F
ilePool policies:
       testaccount (CloudPool Account)
       ecspolicy (FilePool Policy)
Are you sure?? (yes/[no]): yes
To ensure proper cleanup, a job must be run for each S3 enabled account to set an expiration date for al
l stale cloud files.
Failure to set an expiration date will cause leaked data in the cloud resulting in additional costs from
 cloud service providers.
Note that after the expiration date has passed, backups may no longer be able to restore deleted files.
Expiration dates can be set later using the 'isi cloud restore-coi' command.
To start expiration date jobs for applicable accounts, enter an expiration date now or 'default' to acce
pt the default expiration date (2029-03-20): (<date>/[default]/cancel):
hop-isi-n-1#
```

Figure 39. Give the primary PowerScale cluster cloud write access

Note: Do not allow write access to the CloudPools from more than one PowerScale cluster.

7. Log in to the *secondary* OneFS WebUI and go to **Data Protection** > **SyncIQ**. Click **Policies** on the policy that you want to failback and select **More** > **Resync-prep**. This operation disables the SyncIQ replication mirror policy on the secondary PowerScale cluster and places the secondary PowerScale cluster back into read-only mode. In addition, this operation enables the SyncIQ replication policy on the primary PowerScale cluster.

The SyncIQ failback is complete.