

Quantum®

WHITE PAPER

DEPLOYING STORNEXT WITH OBJECT STORAGE AND CLOUD STORAGE

CONTENTS

Introduction	3
StorNext and the Storage Continuum	4
Compatibility	5
Reference Architecture Diagrams	7
StorNext S3 Front-End	8
FlexTier for Private Cloud	9
FlexTier for Public Cloud	10
Configuring an Object Storage or Public Cloud Target	11
Configuring the Storage Destination	11
Configuring the Storage Policy	11
Related Documents & Links	11
StorNext® 6	11
Xcellis®	11

INTRODUCTION

Now that object storage and cloud storage technology have moved from mysterious to mature, the question for most organizations is not *whether* to use them, but *when* to use them to their best advantage. When running a “digital factory” with data as the product, data storage decisions directly affect the bottom line. Hybrid approaches are common—data may originate in the cloud, or some data processing may occur in the cloud, with more on premise. Multi-site sharing and collaboration are frequently required. There are always security concerns and the need to plan for disaster resilience and recovery.

Block vs. File, Cloud vs. On Premise, and SSD vs. HDD are false dichotomies too often propped up to sell the product of the day. This way of thinking about data storage is too limiting and doesn't serve the needs of those who manage the data production workflow. Quantum StorNext makes it possible to take a more nuanced approach that enables data to live on the right storage at the right time, managing performance and cost across the data lifecycle.

This document outlines reference architectures for using StorNext as an object store, *with* third-party object stores, and *with* cloud object storage. A step-by-step example illustrates how quickly and easily StorNext may be integrated with an object storage target.

STORNEXT AND THE STORAGE CONTINUUM

Today StorNext supports multiple types and tiers of storage, from blazing-fast NVMe flash to cloud cold archives, and everything in between. Policies enable tiering of data between types of storage. This includes tiering between primary tiers, for example flash and HDD, and tiering from primary to secondary such as from HDD to cloud, object, or tape.

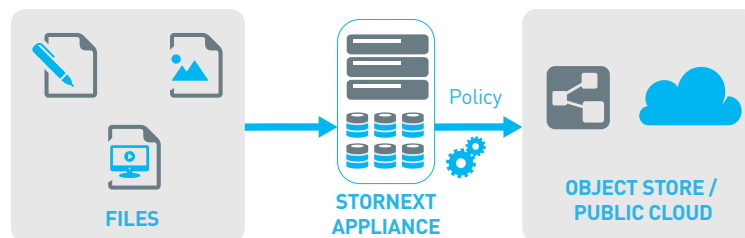


Figure 1 – StorNext as File Repository Tiering to Object Storage or Public Cloud

In addition to the ability to tier data to local or cloud-based object stores, StorNext may also, itself, be configured to appear in whole or in part as an S3-compatible object store. Data stored via the S3 object interface lands on primary storage and may in turn be tiered to less-expensive storage such as tape for long-term safekeeping.

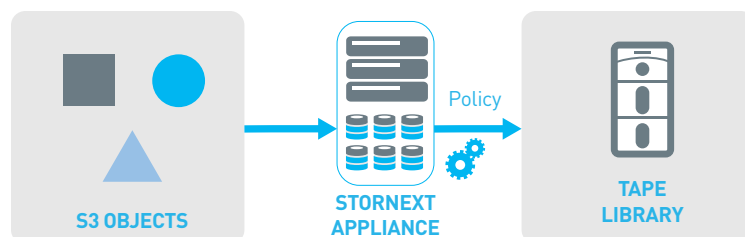


Figure 2 – StorNext as an Object Store Tiering to Tape

Table 1 below summarizes factors to consider when choosing between the StorNext S3 front end, a third-party on-premise object store, and a cloud object store.

Tiering Architecture	Archive On Prem?	Multi-Site Archive?	Retrieve Latency	Cost	Delivery Model
StorNext S3 → Tape	Yes	Yes	Minutes	\$	Purchase or AAS
StorNext FS → Object Store	Yes	Yes	Seconds	\$\$\$	Purchase
StorNext FS → Public Cloud	No	Yes	Seconds to Days	\$\$	AAS

Table 1 – Reference Architecture Characteristics

In the table, “Archive On Prem” indicates whether the archive is resident at the customer premises or lives only in the cloud. One is not better than the other in all cases; it depends on the situation. For example, for smaller archives that are accessed rarely, cloud-only storage may be appropriate.

The “Multi-Site Archive” column shows that all three architectures can provide DR protection by spreading data across multiple sites, though each one accomplishes this goal in a different way. For StorNext S3 front end and Tape, data can be replicated from the primary site to one or more secondary sites and stored on tape at any or all those locations. When StorNext tiers data to a disk-based object store, the geospreading capability of the object storage system can be used to protect the data across sites. Public cloud providers all offer the option to protect data across multiple sites, which may be leveraged when StorNext tiers to the cloud.

Key to the design of a tiered storage system is the retrieve latency—the time elapsed between the submission of a request to retrieve data from a slower tier and when the data becomes available. This factor is often tied to the cost of the storage system, but not always. For a secondary storage tier, disk-based on-prem object stores are expensive, but for large quantities of “warm” data that must remain immediately available, they are the best answer. Public cloud offers a range of price points and retrieve latencies. If you pay a lot, you can get your data back very quickly. If you must move large quantities of data but can wait days for it, you can pay a lot less. In comparison, tiering to tape can be extremely attractive. Data is accessible within a minute or two of a request, and tape has the absolute lowest cost per TB of any storage medium.

Finally, the purchasing model can be relevant. Disk-based on-prem object stores are usually just like traditional disk arrays, in that they live on the corporate data center floor and are managed by corporate staff. Cloud storage is purchased “as-a-service” (AAS), putting no burden on local resources beyond paying the monthly bill. A StorNext-based object storage solution with tiering to tape offers the security and availability of on-prem storage and can be acquired as a customer-managed capital purchase, or a Quantum-managed AAS OPEX offering.

COMPATIBILITY

StorNext's capability to tier to object storage systems and clouds has been tested with a wide range of disk-based object stores and cloud storage providers. The current list is available on Quantum.com in the StorNext 6 Compatibility Guide found [here](#). Object stores and cloud services not specifically listed will often work using the *S3 Compatible* selection in the *Provider* list in the StorNext GUI. Formal compatibility testing for devices and providers not listed in the StorNext 6 Compatibility Guide may be requested through [Quantum Sales](#).

REFERENCE ARCHITECTURES

Several high-level reference architecture diagrams and descriptions follow, designed to illustrate how to deploy StorNext as an S3 object store with tape, and how to tier data from StorNext primary storage to both on-premise object stores and public cloud destinations. The characteristics in Table 1 may be used to help select the most appropriate architecture for a particular use case, but for detailed sizing and configuration guidance or a detailed demonstration of these features, a [Quantum Solution Architect](#) should be consulted.

StorNext S3 Front End

Notable characteristics of this configuration include:

- Direct S3 interface for cloud-aware applications
- Tiering to tape for data protection & cost control
- Optional multi-site DR capability via FlexSync™ replication
- Best cost profile for warm and cold data

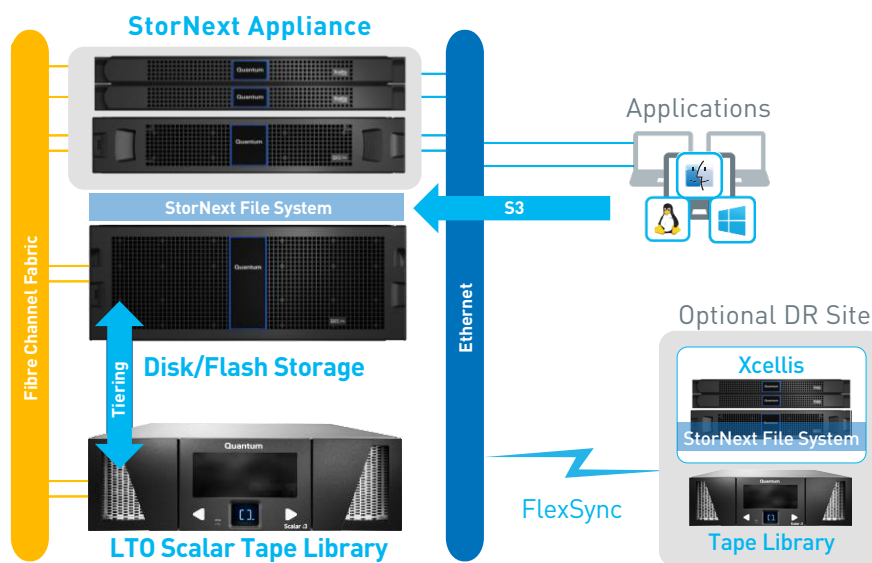


Figure 3 – StorNext S3 Front End

StorNext Tiering to Object Storage aka Private Cloud

This architecture features:

- Familiar NAS interface for ease of use
- Choice of Fibre Channel or Ethernet storage network
- Optional direct clients offer highest performance
- On-premise object tier with optional geospread for DR or collaboration
- Wide variety of object storage systems supported
- Best for active archives where large quantities of “warm” data must be immediately available

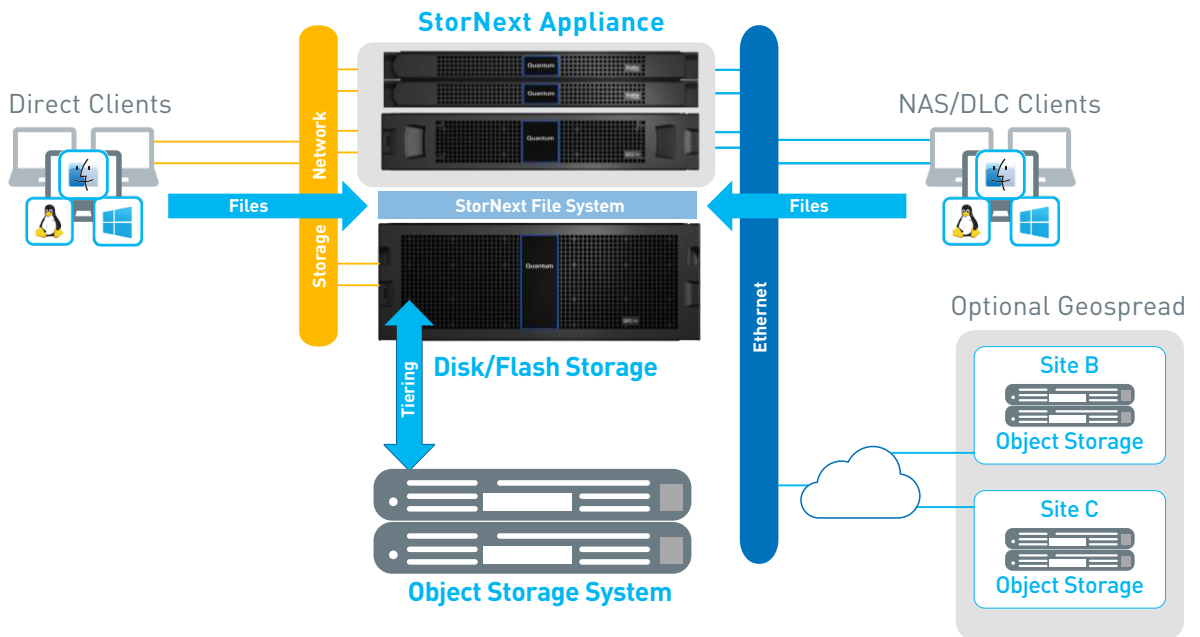


Figure 4 – StorNext Tiering to Object Storage

FlexTier for Public Cloud

Features of this architecture include:

- Familiar NAS interface for ease of use
- Choice of Fibre Channel or Ethernet storage network
- Optional direct clients offer highest performance
- Tiering to a wide selection of public cloud providers
- Best for cold archives with infrequent retrieval

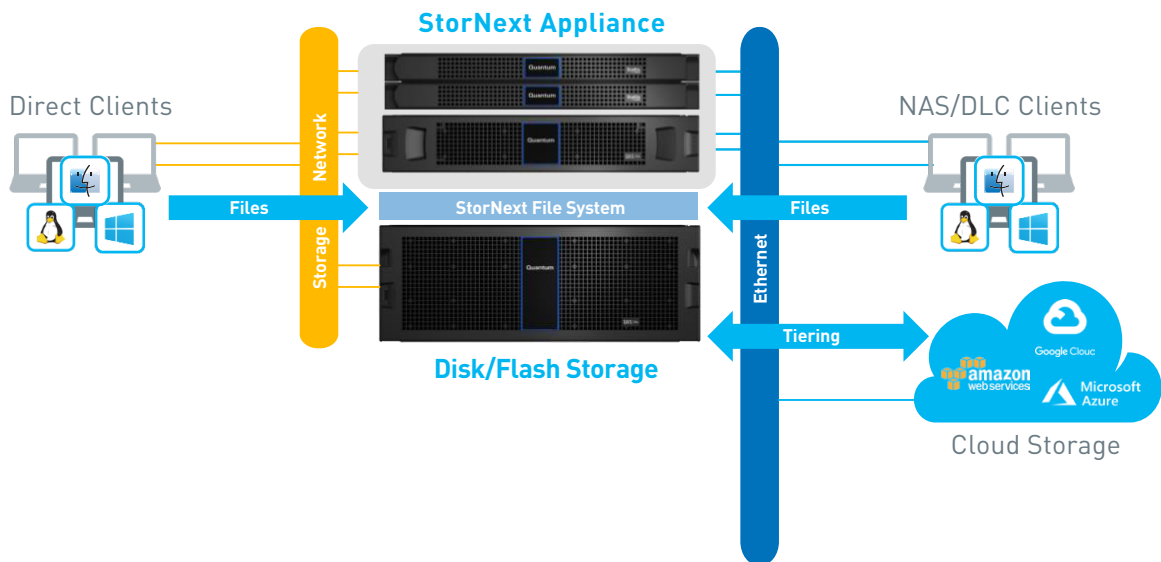


Figure 5 – FlexTier for Public Cloud

CONFIGURING TIERING TO AN OBJECT STORAGE OR PUBLIC CLOUD TARGET

The following example illustrates how to configure StorNext to tier data to an AWS S3 target. Configuration with disk-based object stores and other public cloud providers is very similar, but not identical. The StorNext GUI was used for this example, alternatively the CLI may be used, or the steps may be scripted.

Detailed instructions for configuring StorNext tiering with all supported object stores and public clouds are contained in the [online documentation at Quantum.com](#).

Once the steps below are complete, the system will behave in the following way:

- When a new file lands in the archive directory, and has not been accessed for five minutes, a copy of the file will be made to AWS. This provides immediate data protection and DR resilience.
- If the primary disk file system fills up, the data blocks of the least-recently used files will be removed from the primary disk and reside only on AWS. To applications and users all files still appear in the file system as normal, but when one of the older files is accessed, there is a short delay while the data blocks are copied back to primary disk from AWS.

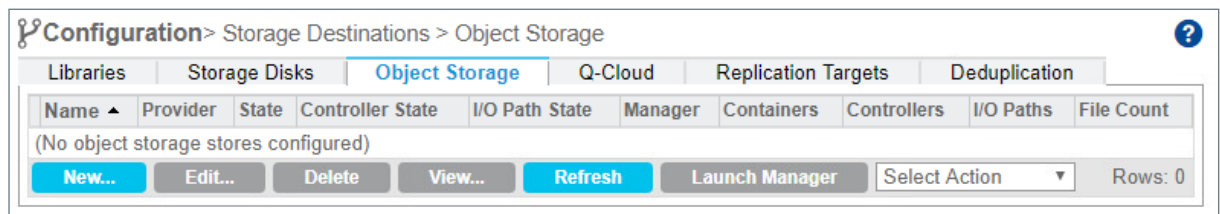
This configuration allows a relatively small-capacity local disk array to essentially act as a cache for AWS. Users and applications never need to be aware of where the data lives or how many copies exist.

There are two major steps in the process to achieve this configuration:

- Configure the object or cloud storage destination.
- Configure a storage policy to use the new storage destination.

Configuring the Storage Destination

1. A web browser is used to open the StorNext GUI, log in, and proceed to the *Configuration > Storage Destinations* screen. The *Object Storage* tab is selected to configure object storage and cloud destinations.



2. When the *New...* button is clicked, a screen appears with the fields required for configuring a new object storage destination.

- The top section of the page is used to enter the *Name* of the object storage destination used within StorNext, the *Provider* (AWS in this case), and the management endpoint, port, and protocol to be used.

Configuration > Storage Destinations > Object Storage > New

Libraries | Storage Disks | **Object Storage** | Q-Cloud

* Name:

* Provider:

* Manager Host:

Manager Port:

Manager Protocol:

Authentication:

User Name:

Password:

The *Authentication*, *User Name*, and *Password* fields are not used with AWS, and are left blank. When configuring on-premise object storage, these fields will contain the user name and password used to access the device’s administrative GUI. Entering them here enables the object store’s GUI to be opened directly from the StorNext GUI via the link on the object storage destinations page.

- Next, a controller is added by clicking the *Add* button in the *Controllers* section and entering a name. There is usually no reason to alter the Max Streams. On-premise object stores usually have multiple controllers. Defining all controllers here enables StorNext to use multiple controllers in parallel.

Controllers

Name	Max Streams
<input type="text" value="AWSC1"/>	<input type="text" value="48"/>

- An I/O path is defined and mapped to the controller by clicking *Add* in the *I/O Paths* section and providing a name for the path. The AWS media type is selected, desired URL Style (*PATH* or *VHOST*), *Object Access Protocol* (https recommended), and the *Host*. *Host* corresponds to the AWS region, such as US East, US West, GovCloud, C2S, etc.

On-premise object stores usually have multiple paths per controller. Defining all paths here enables StorNext to use multiple paths in parallel.

I/O Paths

Name	Controller Name	Media Type	URL Style	Object Access Protocol	Host[:Port] (Port Optional)
<input type="text" value="AWSP1"/>	<input type="text" value="AWSC1"/>	<input type="text" value="AWS"/>	<input type="text" value="VHOST"/>	<input type="text" value="https"/>	<input type="text" value="s3-us-west-2.amazonaws.com"/>

- Finally, in the Containers section, click *Add* and complete the following fields:
 - Container*: The name of the AWS bucket where data will be stored
 - Media ID*: A unique name for this bucket that will be used within StorNext. Each bucket appears as a separate ‘Media’, much like a tape cartridge but with unlimited capacity.

- c. *Media Type*: AWS
- d. *Signing Type*: V2 or V4 (V4 recommended)
- e. *Storage Class*: The AWS Storage Class chosen when the bucket was created. Standard, Infrequent Access, Glacier, etc.
- f. *Authentication Type*: AWS Authentication type – Standard, GovCloud, etc.
- g. *User Name*: The AWS Access Key ID for this bucket
- h. *Password*: The AWS Secret Access Key for this bucket
- i. *Copy Number*: This bucket will be used by StorNext to store file copies with this copy number. Normally left at the default of one.
- j. *Policy Class*: Normally left at the default of *System Blank*, so that any StorNext Policy Class may use this bucket.
- k. *Batch Delete*: Leave checked (default) to enable more efficient deletions of multiple objects.
- l. *V4 Full-Payload Signing*: Check if this feature is desired.

Some object stores and cloud storage providers enable scanning for containers. When using container scanning, the container name is simply selected from a drop-down menu, and most of the container parameters are detected and entered automatically.

7. Additional buckets / containers may be configured by clicking *Add* and entering the details for each bucket as in the previous step.
8. Click the *Apply* button at the bottom of the screen to commit the changes. Click *Yes* and *OK* at the confirmation prompts.



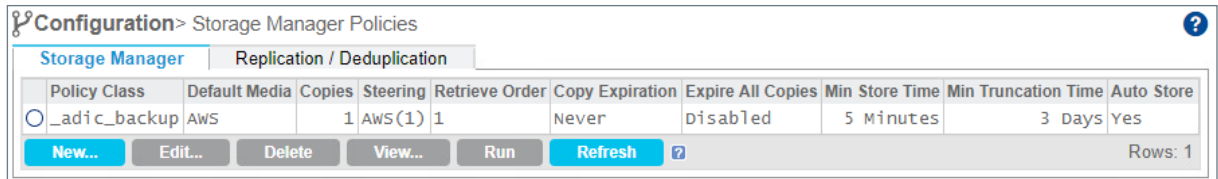
9. The Object Storage Destination will appear with a status of Online.

Name	Provider	State	Controller State	I/O Path State	Manager
AWS	Amazon web services (AWS)	online	online	online	https://s3.amazonaws.com:443

Configuring the Storage Policy

Now that the AWS Storage Destination has been defined and StorNext knows how to connect to it, a Storage Policy must be configured to enable tiering of files to this destination.

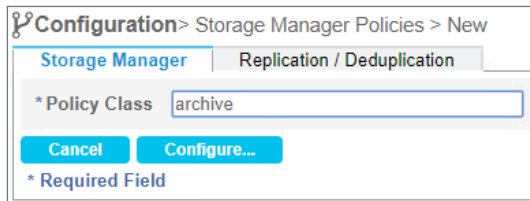
1. Storage Policies are configured in the StorNext GUI *Configuration > Storage Manager Policies* page. Note that a backup policy has been automatically created to protect the StorNext configuration and metadata in AWS. This information is needed in order to quickly recover from a site disaster.



The screenshot shows a table titled "Configuration > Storage Manager Policies" with a sub-tab "Replication / Deduplication". The table has the following columns: Policy Class, Default Media, Copies, Steering, Retrieve Order, Copy Expiration, Expire All Copies, Min Store Time, Min Truncation Time, and Auto Store. There is one row with the following values: Policy Class: _adic_backup, Default Media: AWS, Copies: 1, Steering: AWS(1), Retrieve Order: 1, Copy Expiration: Never, Expire All Copies: Disabled, Min Store Time: 5 Minutes, Min Truncation Time: 3 Days, Auto Store: Yes. Below the table are buttons for "New...", "Edit...", "Delete", "View...", "Run", and "Refresh". The bottom right corner indicates "Rows: 1".

Policy Class	Default Media	Copies	Steering	Retrieve Order	Copy Expiration	Expire All Copies	Min Store Time	Min Truncation Time	Auto Store
_adic_backup	AWS	1	AWS(1)	1	Never	Disabled	5 Minutes	3 Days	Yes

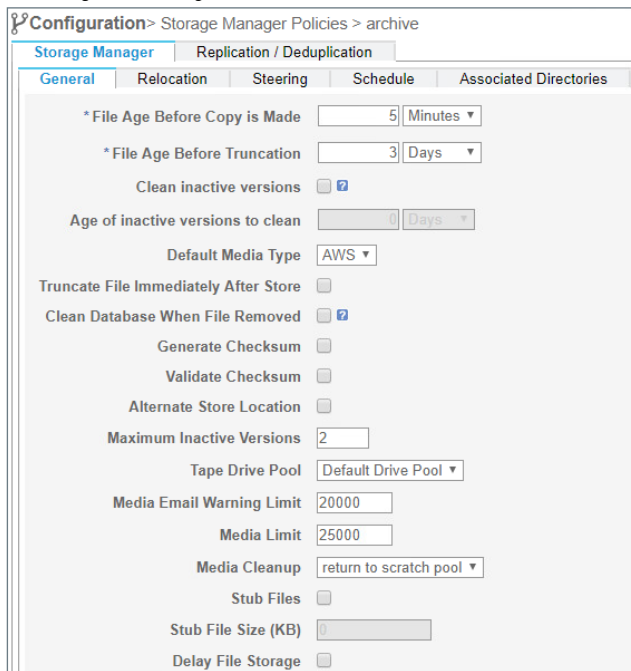
2. To create a storage policy the *New...* button is clicked. A screen appears requesting a name for the new Storage Policy, also known as a *Policy Class*.



The screenshot shows a dialog box titled "Configuration > Storage Manager Policies > New". It has a sub-tab "Replication / Deduplication". There is a text input field labeled "* Policy Class" with the value "archive". Below the input field are two buttons: "Cancel" and "Configure...". At the bottom left, there is a note "* Required Field".

For this example, the policy is named *archive*.

3. Clicking the *Configure...* button commits the name and moves to the configuration screen.



The screenshot shows the configuration screen for the "archive" Storage Policy. It has a sub-tab "Replication / Deduplication" and a "General" sub-tab. The configuration options are as follows:

- * File Age Before Copy is Made: 5 Minutes
- * File Age Before Truncation: 3 Days
- Clean inactive versions:
- Age of inactive versions to clean: 0 Days
- Default Media Type: AWS
- Truncate File Immediately After Store:
- Clean Database When File Removed:
- Generate Checksum:
- Validate Checksum:
- Alternate Store Location:
- Maximum Inactive Versions: 2
- Tape Drive Pool: Default Drive Pool
- Media Email Warning Limit: 20000
- Media Limit: 25000
- Media Cleanup: return to scratch pool
- Stub Files:
- Stub File Size (KB): 0
- Delay File Storage:

The policy configuration screen contains many options across multiple tabs. Policies are extremely flexible to fit any need. Don't be intimidated! Most policies are simple and require only a few changes to the defaults. A detailed description of all the options is beyond the scope of this document, but details may be found in the [online product documentation](#).

4. On the *General* tab, no changes are necessary. Because only one Storage Destination is defined, the *Default Media Type* is already set to *AWS*, which is correct.

Default Media Type AWS ▾

The *Relocation* tab does not apply for object storage or cloud destinations and may be ignored.

5. The *Steering* tab is already configured appropriately since there are no other storage destinations defined. If, for example, there was also a local tape library, this screen would enable configuring the copy and retrieve order for each destination.

Configuration > Storage Manager Policies > archive

Storage Manager | Replication / Deduplication

General | Relocation | **Steering** | Schedule | Associated Directories

Copy	Media Type	Media Format	Tape Drive Pool	Drive Limit	Retrieve Order	Copy Expiration
1	AWS ▾	---Select---	<Not configured>	None ▾	1 ▾	0 Never ▾
2	<Not configured>	---Select---	<Not configured>	None ▾	---Select-- ▾	0 Never ▾

6. The *Schedule* tab is set by default to store file copies automatically, and there is rarely any reason to change this.

Configuration > Storage Manager Policies > archive

Storage Manager | Replication / Deduplication

General | Relocation | Steering | **Schedule** | Associated Directories

Store Files Automatically

Cancel Apply

* Required Field

7. The only settings that need to be manually configured are on the *Associated Directories* tab, where it is necessary to specify the file system and directory paths to which this policy will apply. In this case there is a single file system *snfs1* which has been selected automatically. To apply the policy to the directory named *archive*, it is only necessary to choose the directory from the drop-down.

Configuration > Storage Manager Policies > archive

Storage Manager | Replication / Deduplication

General | Relocation | Steering | Schedule | **Associated Directories**

File System snfs1 ▾

Directory Filter Apply Filter

Directory -- Select Directory -- ▾ Create Directory

Directory Path -- Select Directory --

archive
lost+found

Associated Directories

(No associated directories configured.) ▾

If the directory did not already exist, it could be created here by filling in the name and clicking the *Create Directory* button. Once the directory exists, is selected, and the *Add* button clicked, the directory appears in the list of *Associated Directories*.

Associated Directories

/stornext/snfs1/archive ▲

8. Finally, clicking *Apply* creates the storage policy named *archive*, which then appears in the list.

Configuration> Storage Manager Policies ?

Storage Manager **Replication / Deduplication**

	Policy Class	Default Media	Copies	Steering	Retrieve Order	Copy Expiration	Expire All Copies	Min Store Time	Min Truncation Time	Auto Store
<input type="radio"/>	_adic_backup	AWS	1	AWS(1)	1	Never	Disabled	5 Minutes	3 Days	Yes
<input type="radio"/>	archive	AWS	1	AWS(1)	1	Never	Disabled	5 Minutes	3 Days	Yes

New... **Edit...** **Delete** **View...** **Run** **Refresh** Rows: 2

That's it! The system will now exhibit the behavior described at the beginning of this section, automatically protecting data to AWS as it lands on disk, and using AWS as a tier once the local disk fills up.

CONCLUSION

Quantum StorNext offers wide compatibility with third-party object storage systems and cloud storage providers. Configuring StorNext to tier to object and cloud storage is very straightforward. As a low-cost alternative, StorNext may be configured as an S3 object store itself, using tape as a tier. Choosing the right architecture for a specific situation depends on factors such as acceptable retrieve latency, requirements to keep data on premise, budget, and desire for a CAPEX or OPEX purchasing model. A wide variety of options are available to meet virtually any need.

RELATED DOCUMENTS & LINKS

StorNext 6

Product Page: www.quantum.com/stornext

Documentation Center: www.quantum.com/snsdocs

Compatibility Guide: http://qsupport.quantum.com/kb/flare/Content/stornext/SN6_PDFs/Compatibility_Guide.pdf

Xcellis

Product Page: www.quantum.com/xcellis

Documentation Center: www.quantum.com/documentation

Quantum®

ABOUT QUANTUM

Quantum technology and services help customers capture, create, and share digital content—and preserve and protect it for decades at the lowest cost. Quantum’s platforms provide the fastest performance for high-resolution video, images, and industrial IoT, with solutions built for every stage of the data lifecycle, from high-performance ingest to real-time collaboration and analysis and low-cost archiving. Every day the world’s leading entertainment companies, sports franchises, research scientists, government agencies, enterprises, and cloud providers are making the world happier, safer, and smarter on Quantum. See how at www.quantum.com.

www.quantum.com • 800-677-6268