IBM Spectrum Protect

# *Configuring IBM Spectrum Protect for IBM Spectrum Scale Active File Management*

**Document version 1.5**

*Dominic Müller-Wicke*
*IBM Spectrum Protect Development*

*Nils Haustein*
*EMEA Storage Competence Center*

IBM

# CONTENTS

# 1   Introduction

This paper describes scenarios and provides configuration guidance for IBM® Spectrum® Scale (previously known as IBM GPFS[1]) Active File Management (AFM) in conjunction with data protection such as backup and hierarchical storage management (HSM).
Because AFM can be configured in a variety of ways with data protection, each scenario is described.

HSM is implemented with IBM Spectrum Protect for Space Management (also known as UNIX HSM[2]). Backup is implemented with the `mmbackup` tool, which is part of IBM Spectrum Scale. The `mmbackup` tool uses and requires the IBM Spectrum Protect backup-archive client to be installed on the nodes used.

**Who should read this paper?** The documentation was written for IBM Spectrum Scale and IBM Spectrum Protect administrators who plan to implement data protection in the AFM environment. This paper is also intended for administrators who have data protection for AFM already implemented, want to understand the challenges of this combination of technologies, and want guidelines for dealing with these challenges. In addition, the paper can help technical sales personnel to integrate the technologies into the customers' business processes.

**How should this paper be read?** You should identify and read the scenario that is closest to your implementation plans.

## 1.1   IBM Spectrum Scale Active File Management

Active file management (AFM) is a scalable, high-performance file-system caching layer that is integrated with the IBM Spectrum Scale cluster file system. AFM is based on a home-cache model. A single home provides the primary file storage that is exported. One or more caches provide a view into the exported home file system without storing the file data locally. Upon file access in the cache, the data is fetched from home and stored in cache. Another way to get files transferred from home to cache is through prefetching. The prefetching process uses the IBM Spectrum Scale policy engine to quickly identify files that meet certain criteria.

---

[1] IBM General Parallel File System (GPFS)
[2] Note: Throughout this document, the term UNIX HSM is used to refer to IBM Spectrum Protect for Space Management.

When files are created or changed in cache, they can be replicated back to home. A file that was replicated back to home can be evicted in cache, whereby the user still sees the file in cache (the file is uncached), but the actual file content is stored in home. Eviction is triggered by the quota that is set on the AFM fileset. Files can be evicted based on size or the most recently applied criteria.

Cache must be an IBM Spectrum Scale independent fileset. Home can be an IBM Spectrum Scale file system, a Network File System (NFS) export from any other file system, or a file server (except for the disaster-recovery use case). The caching relationship between home and cache can be based on the NFS or native IBM Spectrum Scale protocol. In the latter case, home must be an IBM Spectrum Scale file system in a different IBM Spectrum Scale cluster. In this document, we assume the cache relation is based on the NFS protocol.

AFM masks wide area network latencies and outages by using IBM Spectrum Scale to cache massive data sets, allowing data access and modifications in cache even when a remote home storage cluster is unavailable. The AFM implementation leverages the inherent scalability of IBM Spectrum Scale to provide a multinode, consistent cache of data located at a home cluster. By integrating with the file system, AFM provides a POSIX-compliant interface, making the cache completely transparent to applications. For more information about AFM, see the IBM Spectrum Scale documentation.

The AFM relation is typically configured on the cache fileset in one specific mode (see AFM Modes). The AFM mode determines where files can be processed (created, updated, and deleted) and how files are managed by AFM according to the file state (see Figure 1).

### 1.1.1 AFM Modes

The relation between cache and home can be based on different modes. The mode determines how AFM behaves. One of the following modes must be set in cache when creating an AFM relation:

**SW (Single Writer):** The home export is the AFM target for one cache. Files in uncached state are fetched from home upon access or by using the prefetch command.  Files can be modified in cache and are replicated to home asynchronously. Files that are deleted in cache are deleted in home. Files can be evicted from cache when present in home. Evicted files that are in uncached state can be seen in cache and are fetched upon access

(or prefetching). Home is read-only in this mode from a user perspective (not from the AFM perspective).

**IW (Independent Writer):** Similar to SW mode, but home is the AFM target for one or more caches. All changes in the caches will be replicated to home asynchronously. Changes to the same data will be applied to the home fileset so that the changes are replicated from the caches; there is no cross-cache locking. Potential conflicts must be resolved at the respective cache site.

**LU (Local Update):** The home export is the AFM source for one cache. Files in uncached state are fetched from home upon access or by using the prefetch command. Files can be changed in cache but are not replicated back to home as they become local. Also, files that are created in cache are not replicated to home. Eviction should be disabled for AFM LU mode. This AFM mode is typically used for data migration, whereby home is the legacy file system that is migrated to cache via fetching and prefetching files. File changes on cache are not applied to home.

**RO (Read Only):** The home export is the AFM source for one cache. Uncached files in cache are fetched from home upon access or with prefetching. Cache is read-only; files cannot be changed or deleted in cache. Files can be evicted from cache. In that case, the file state is changed to uncached. Uncached files can be seen in cache and are fetched upon access or prefetching.

**DR (Disaster Recovery):** AFM DR is a disaster recovery function to augment the overall business recovery solution. The function asynchronously copies files from one fileset in one IBM Spectrum Scale cluster to another fileset in another IBM Spectrum Scale cluster and provides failover and failback capabilities. The AFM DR architecture is represented by two sites: primary and secondary. The primary site is a read-writeable fileset where the applications have read/write access to the files. The secondary site is another fileset that should be operated in read-only mode if the primary site is online. All data created, modified, and deleted on the primary fileset is asynchronously replicated to the secondary fileset in the same order in which the files changed. The primary site is continuously available to the applications even in the event of communication failures with the secondary site. AFM DR also provides a Recovery Point Objective feature (RPO), which is used to recover data from a specified time from the secondary site when the primary site has failed; this is failover functionality. The RPO feature is based on *peer snapshots*. Peer snapshots are created in both primary and secondary filesets at the same recovery point.

AFM DR also provides a failback procedure. Assume that the original primary system failed, and a failover to the secondary system was completed. The secondary system is now the new primary system. The failback procedure copies all data that was added or changed on the new primary system to the original primary system. When this copy process is completed, the original primary system resumes its role as the primary system and operations continue as usual.

---

**Restriction:** The usage of UNIX HSM in combination with AFM DR is not supported because snapshots created by AFM DR interfere with HSM operations and can cause massive recalls.

---

AFM modes of cache filesets running in SW, IW, or RO can be changed to all other AFM modes. Filesets running in LU mode can't be changed to other modes. In all cases, AFM can be disabled on filesets, which then become pure independent filesets.

### 1.1.2 AFM File States

A file in an AFM cache can have different states, as shown in Figure 1. The file states can differ, depending on the AFM modes.
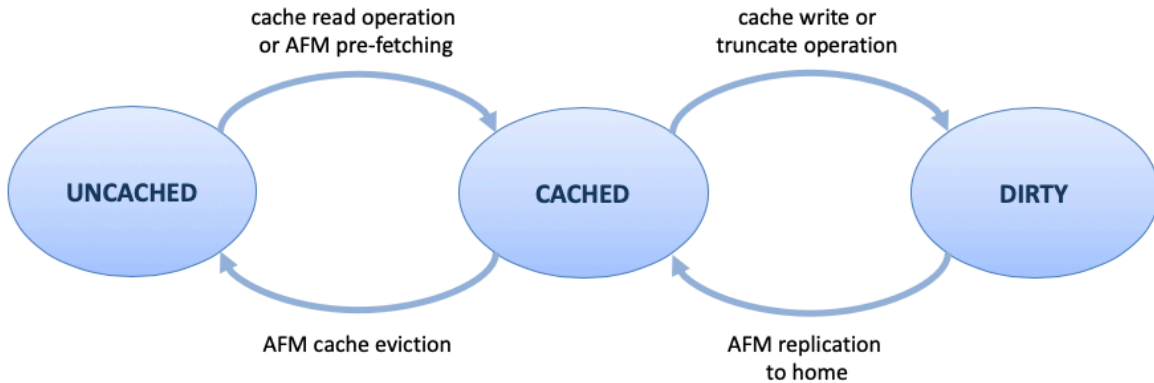


*Figure 1: AFM file states at cache*

**UNCACHED:** When an AFM relation is created between cache and home, and files are available at home, these files can be seen in cache without being present. This means that the file metadata is present in the cache, but the file content is still at home. Such files are in uncached state. In addition, the uncached state is achieved by evicting files from the cache. If AFM DR mode is enabled, files are never in uncached state on the primary system.

**CACHED:** When an uncached file is accessed in cache for read or write, the file is fetched from home. *Fetching* is the process of copying a file from home to cache. Files fetched from home to cache are in cached state. Another way to fetch files from home to cache is by using the AFM prefetch command (`mmafmctl prefetch`). This command can leverage the policy engine to identify files quickly, according to certain criteria. In AFM DR mode, files in the primary system are cached. The files might be in a dirty state, as well, if the files have not been replicated.

**DIRTY:** When a cached file in the AFM cache is modified, that file is marked dirty, indicating that it is a candidate for replication back to home. If a file has been replicated to home, the dirty state of the file is reset to cached. When a file is deleted in cache, the file is also deleted at home. Modifying or deleting files in cache is not possible in AFM read-only mode. In AFM LU mode, dirty files are not replicated back to home; the files become local. For this reason, the dirty state is also called local in AFM LU mode.

A cached file can be evicted from cache. The eviction changes the file state to uncached and the file content is no longer stored in cache. An eviction can occur only if a file is at home and cached in cache. If a file was migrated to cache, the file will not be evicted. Eviction is automatically controlled by AFM via the quota setting of the AFM cache fileset and is not available in DR mode. You can evict a cached file by using the AFM evict command:

```
mmafmctl evict
```

## 1.2    IBM Spectrum Protect for Space Management

UNIX HSM supports the tiering of files in an IBM Spectrum Scale file system from disk to an external pool. The external pool is represented by the IBM Spectrum Protect server, which can store files on tape.

In terms of UNIX HSM, a file can have three states:
- A new or modified file is in resident state, which means that the file exists only in the file system.
- After a file is premigrated, the file exists in a dual mode. One copy of the file resides in the file system, while the other copy of the file is stored in IBM Spectrum Protect. This file state is called premigrated.
- The third file state is called migrated. This file state indicates that the file exists in the IBM Spectrum Protect server file space only. The copy of the file in the file

system is replaced with a stub file, which typically doesn't require any space in the data area of the file system.

HSM commands can be used to "premigrate" and "migrate" files. The environment might be configured to perform automatic threshold migration.

The following Portable Operating System Interface (POSIX) operations on files will result in UNIX HSM operations:

READ:

- Read access to a file requires that the file data is present, but the data cannot be changed. UNIX HSM provides different recall modes. All modes can be applied to the recall in terms of read access.

- Read access does not affect files in resident state.

- Read access does not affect files in premigrated state.

- With read access, the migrated file state is changed to premigrated. A recall is initiated because UNIX HSM allows multiple configurations for migrated files. To learn about exceptions to this rule, see the *IBM Spectrum Protect for Space Management User's Guide*.

WRITE:

- A write operation changes the content of a file. Therefore, the file content must be available during the operation. All write access to files requires a full recall of the file data. The recall occurs in normal recall mode. Different recall mode settings for the file will be disabled.

- A write operation has no influence on resident files from the UNIX HSM point of view.

- A write operation changes the premigrated file state to resident. No recall occurs.

- A write operation changes the migrated file state to resident. A recall is initiated. (Because UNIX HSM allows multiple configurations for migrated files,  exceptions to this rule exist. For more information, see the *IBM Spectrum Protect for Space Management User's Guide*.)

TRUNCATE:

- A truncate operation is comparable to a write operation that reduces file size. A truncate operation removes data from the end of the file. You can truncate a file

to size null. Truncate access to files requires a full recall of the file data, except in cases where the file is truncated to null. The recall occurs in normal recall mode. Different recall mode settings for the file will be disabled.

- A truncate operation has the same effect as a write operation. The only exception is the case where the file is truncated to size null. In this case, the file state is changed to resident and no recall is initiated.

If metadata is changed or the file is renamed, the file migration state does not change. File movement has no effect on the file migration state, assuming that the move takes place within an IBM Spectrum Scale fileset or within an IBM Spectrum Scale file system. If the file is moved outside these borders, a recall is initiated, and the file is set to resident state.

By default, AFM caches entire files. In general, reading more than three blocks of a file will cause AFM to cache the entire file in the background to optimize performance. To avoid reading large files when an application might be "peeking" into the file (for example, GUI viewers can read a few bytes to detect the file MIME type), certain file reads at the beginning of the file are handled individually. UNIX HSM provides the function to define different stub sizes for files. This means that a defined number of blocks will reside on disk while the rest of the data is migrated to the IBM Spectrum Protect server. To synchronize the AFM function to initiate the caching after reading three blocks of the file, the stub size for UNIX HSM migration should be set to three blocks as well. This setting will significantly reduce the number of recalls on home.

Sequential read requests to an uncached file that are initiated from an application running on the cache cluster are pipelined. This means that data blocks read sequentially from the application are directly passed from the home cluster to the application (by using the AFM gateway node). This process interlocks well with the UNIX HSM streaming recall function, which is similar on an HSM file system level. In cases where applications read the data in streams on the cache, the UNIX HSM function streaming recall should be enabled for these files in the home cluster. Because a file in streaming recall mode can be read immediately after the recall has started, the combination of AFM streaming and UNIX HSM streaming recall will increase the overall read performance for uncached files in the cache.

**Tip:** UNIX HSM operates on file system level. In an AFM environment where a file system contains both cache and a home fileset, consider using IBM Spectrum Protect EXCLUDE

rules to exclude filesets from UNIX HSM processing. Follow this guideline especially for filesets where HSM is not applicable or required. In this way, you can help prevent unexpected recall or migration behavior.

### 1.2.1  Important Environment Settings

To enable AFM support, the IBM Spectrum Protect processing option `AFMSKIPUNCACHEDFILES YES` must be set in the `dsm.sys` configuration file above all server stanza entries.

### 1.3  Guidelines

Follow these general guidelines when using IBM Spectrum Scale AFM and UNIX HSM:

- Prevent cache eviction in combination with UNIX HSM space management operations on the same fileset. Both techniques have the same goal of reducing the space required in the fileset. The combination of both techniques unnecessarily increases the complexity of the environment.

- IBM Spectrum Scale snapshots and UNIX HSM have limited compatibility. The deletion of a stub file that is reflected in a snapshot (that was generated before or after the file was migrated) will cause the recall of the file data. The file data will be stored in the snapshot so that the data can be accessed later. Therefore, avoid using snapshots for an AFM fileset (in home or cache) and in the file system that is hosting the AFM fileset.
  Because AFM DR uses peer snapshots internally (creating snapshots in the primary and the secondary AFM fileset to accommodate RPO) the usage of UNIX HSM in combination with AFM DR is not supported. If files are not deleted, a solution with AFM DR and UNIX HSM is conceivable, but in most cases, this solution is not realistic.

- If you use UNIX HSM at home or in cache, be aware that access (read or write) to multiple migrated files at the same time causes bulk recalls. Access to multiple files can be caused by users (when users copy an entire directory) or by AFM, when changed files are replicated to home, where the previous versions are migrated. You can avoid these issues by using the tape-optimized recall process, which requires a list of files to be recalled before processing.

- If you use the UNIX HSM "multiserver" feature, which enables migration of files from a single source file system to multiple target IBM Spectrum Protect servers,

the preferred method is to set up only one IBM Spectrum Protect server per AFM file system. Using multiple IBM Spectrum Protect servers per AFM fileset would significantly increase the complexity.

When running IBM Spectrum Scale AFM and IBM Spectrum Protect backup operations, prevent cache eviction in combination with IBM Spectrum Protect backup operations on the same fileset, if possible. Evicted (uncached) files will be skipped during backup processing. This can lead to file-versioning errors on the IBM Spectrum Protect server.

## 1.4    General Processing Examples

Beginning with IBM Spectrum Protect Version 7.1.3, the IBM Spectrum Protect for Space Management client and the backup-archive client skip files from processing that are in AFM dirty state or AFM uncached state. IBM Spectrum Scale eviction will skip a file migrated with UNIX HSM. For these cases, you can use the command-line interfaces of IBM Spectrum Protect and IBM Spectrum Scale. You might find the examples helpful.

The following example shows the messages that are issued if IBM Spectrum Protect for Space Management migration is performed on an uncached or dirty file:

```
> dsmmigrate file1
IBM Tivoli Storage Manager
Command Line Space Management Client Interface
  Client Version 7, Release 1, Level 3.0
  Client date/time: 07/20/15   10:56:47
(c) Copyright by IBM Corporation and other(s) 1990, 2015. All Rights
Reserved.

ANS9828W File: /gpfs1/file1 is skipped for migration: the file is either
AFM uncached or dirty.
```

The following example shows the messages that are issued if an IBM Spectrum Protect backup operation is performed on an uncached or dirty file:

```
> dsmc incremental /gpfs1/file1
IBM Tivoli Storage Manager
Command Line Backup-Archive Client Interface
  Client Version 7, Release 1, Level 3.0
  Client date/time: 07/20/15   10:58:44
(c) Copyright by IBM Corporation and other(s) 1990, 2015. All Rights
Reserved.
. . .
Incremental backup of volume '/gpfs1/file1'
Normal File--> 15,539 /gpfs1/file1 [Skipped]
```

```
Total number of objects inspected:          1
Total number of objects backed up:          0
. . .
Total number of bytes transferred:          0  B
. . .
```

The following example shows the messages that are issued if IBM Spectrum Scale eviction is performed on a migrated file:

```
> mmafmctl gpfs1 evict -j sw2 --filter FILENAME=file1 --safe-limit=1
mmafmctl: [N] Unable to find any file to evict.
mmafmctl: Could not evict any of selected files.
```

Avoid evicting migrated files. While you can evict a file that is migrated but has resident blocks (this type of file might have a defined stub size or might be a file that is in "partial recall" mode and was partially accessed), in terms of access to a migrated and evicted file, the IBM Spectrum Protect server recalls the file before the file data is fetched from AFM home.

# 2    User Scenario: Disaster Recovery

In this scenario, the cache fileset represents the primary site and is called primary fileset. While the primary fileset is available, it allows read-write access to all files. The home fileset is the secondary site and it is called the secondary fileset. The secondary fileset obtains copies of all files asynchronously (as shown in <u>Figure 2</u>). In the case of a disaster at the primary site, the secondary site can be activated and become the new primary site using the failover process.

One or more primary filesets can be connected to a single secondary cluster, whereby each primary fileset replicates to a distinct secondary fileset. For simplicity, we will discuss a configuration in which one primary fileset is connected to one secondary fileset.

The primary fileset is configured in PRIMARY mode; the secondary fileset is configured in SECONDARY mode and should be treated as read-only. Read-only mode on the secondary fileset is technically not enforced; the mode must be assured from an operational perspective because files created on the secondary fileset are not replicated to the primary fileset during normal operations. Files are managed (created, modified, read, and deleted) on the primary fileset and asynchronously replicated to the secondary fileset by using the AFM function. If the primary fileset fails, the secondary fileset can be made the new primary fileset, and read-only mode is revoked (failover). When the old primary fileset is back online, it can be made the primary fileset again (failback). The normal failback process copies all changed files from the new primary fileset to the old primary fileset before the old primary fileset becomes primary again.

In AFM DR mode, there is no concept of uncached files in the primary fileset. This means that files created on the secondary fileset are not shown in the primary fileset during normal operation. If a file on the primary fileset has been changed, the fileset becomes dirty and is replicated to the secondary fileset during the next replication cycle. This is also true during failback: All files from the new primary fileset are copied to the old primary fileset and appear there in cached state.

AFM DR has a feature to assure a defined Recovery Point Objective (RPO) during failover and failback. The RPO feature is based on peer snapshots. A peer snapshot is initiated on the primary fileset, takes a snapshot of the primary fileset and adds a snapshot marker to the replication queue. Because the replication of the queue entries is in sequential order, the peer snapshot marker arrives at the secondary fileset in the same order. When the

snapshot marker arrives in the secondary fileset, another snapshot is taken. Because the replication is in order, both snapshots (in the primary and secondary filesets) have the same content, even though they were taken at distinct points in time. The existence of peer snapshots means that file inventory can be recovered from the secondary fileset for a defined point in time, when the peer snapshot was taken.

---

**Restriction: The usage of UNIX HSM in combination with AFM DR is not supported.**

Background: If a file that is captured in an IBM Spectrum Scale snapshot is in migrated state, the deletion of this file causes a transparent recall into the snapshot. When many files are deleted, this may cause a recall storm in the primary fileset. (In a *recall storm*, resources are overburdened by too many recall requests.) In addition, because the deletion is propagated to the secondary fileset, a recall storm for migrated files can occur in the secondary fileset, as well.

If the solution is configured to ensure that no files are deleted, a solution with AFM DR and UNIX HSM is conceivable. However, in most cases, this is not realistic because the solution is used by many users and applications. For this reason, UNIX HSM in combination with AFM DR is not supported.

To understand the limitations on UNIX HSM and IBM Spectrum Scale snapshots, see [Guidelines](#).

---

The subsequent sections elaborate on this user scenario and consider different cases with backup operations on secondary and primary filesets.

## 2.1    Backups on Secondary and Primary Sites

In this section, we elaborate on the AFM disaster-recovery use case with backups on primary and secondary sites, as shown in [Figure 2](#).
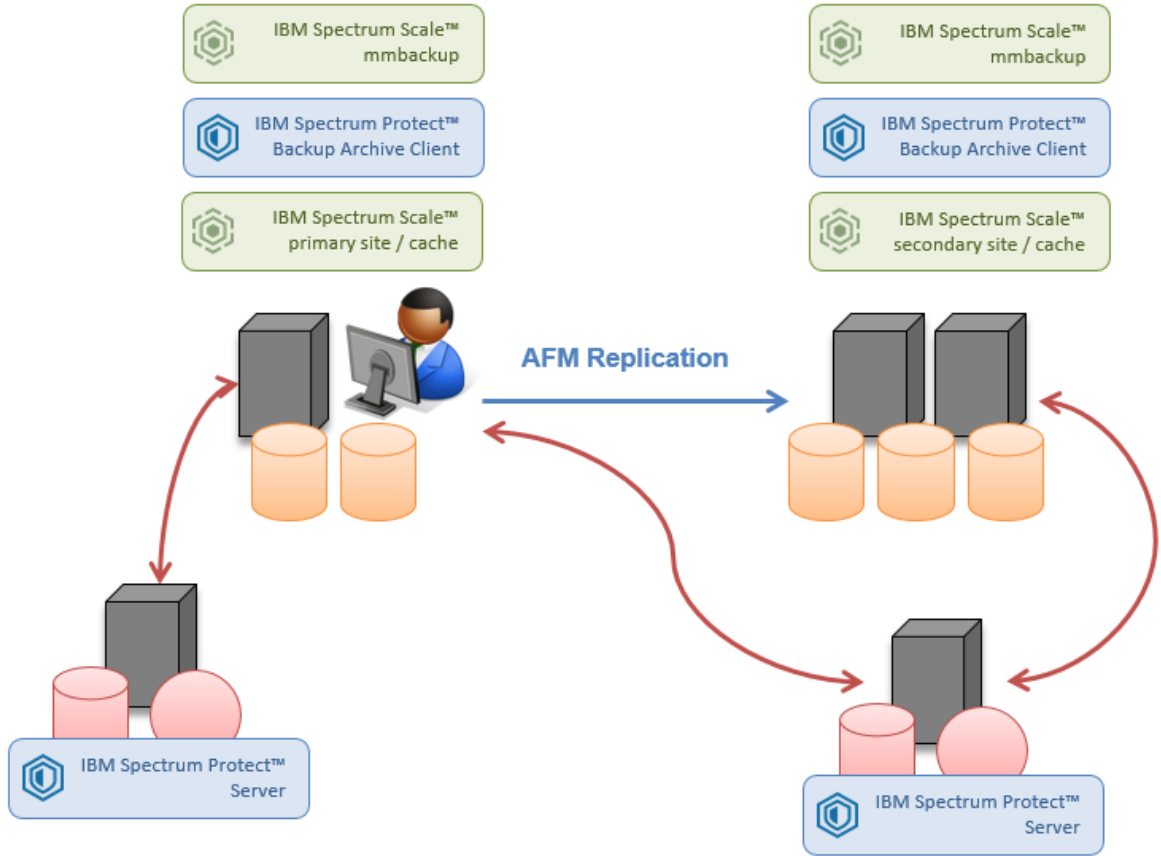
*Figure 2: Disaster recovery use case with backups on secondary and primary sites*

The backup in IBM Spectrum Scale is accomplished with the `mmbackup` utility, which uses the IBM Spectrum Protect backup-archive client. Backup of secondary and primary sites can be configured with the same or different IBM Spectrum Protect servers. If you use the same IBM Spectrum Protect server to back up data from secondary and primary sites, the file system names in the secondary and primary sites must be distinct. In this way, you prevent concurrent access to the same data from both AFM sites (primary and secondary). Concurrent access from both sites would lead to inconsistent backups.

The backup utility will not back up files that are in the dirty state from the primary site[3].

### 2.1.1 Primary Site Operations

The following table summarizes file operations on primary sites and the consequences of file operations on secondary and primary sites. From an HSM perspective, the behavior is

---

[3] The IBM Spectrum Protect V7.1.3 and later backup-archive client skips uncached and dirty files during backup operations.

identical to the use case with HSM on secondary and primary sites (see Backups on Secondary and Primary Sites); we will not repeat this information. Instead, we will focus on the backup.

| Operation | Primary site | Secondary site |
|---|---|---|
| CREATE file | The next backup operation copies the file to the IBM Spectrum Protect server on the primary site. If the next backup cycle happens before the replication, the file is skipped because it is dirty. | After replication, the next backup operation copies the file to the IBM Spectrum Protect server on the secondary site. |
| READ file | There is no impact to backup processing. | There is no impact to backup processing. |
| WRITE file | After the write operation, the modified file is replicated back to the secondary site asynchronously. The next backup generates a new version of the file in the IBM Spectrum Protect server at the primary site. If the next backup cycle happens before the replication, the file is skipped because it is dirty. | After the write operation, the file is replicated to the secondary site. The next backup operation copies the file to the IBM Spectrum Protect server at the secondary site. |
| TRUNCATE file | The next backup operation generates a new version of the file in the IBM Spectrum Protect server at the primary site. If the next backup cycle happens before the replication, the file is skipped because it is dirty. | A truncate operation is promoted to the secondary site without transferring the file during the next replication cycle. The next backup operation generates a new version of the file in the IBM Spectrum Protect server at the secondary site. |
| RENAME or MOVE file within fileset | The next backup operation creates a file in the IBM Spectrum Protect server and expires the old file (name). | A move operation is promoted to the secondary site without transferring the file during the next replication cycle. The next backup operation generates a file in the IBM |

| Operation | Primary site | Secondary site |
|---|---|---|
| | | Spectrum Protect server and expires the old file (name) |
| MOVE file to other fileset | The next backup operation expires the moved file in the IBM Spectrum Protect server at the primary site. | The next replication cycle removes the file from the secondary fileset. The next backup operation expires the file in the IBM Spectrum Protect server at the secondary site. |
| EVICTION | N/A. Eviction is disabled with AFM DR. | There is no impact. |
| DELETE file | The next backup operation expires the file in the IBM Spectrum Protect server at the primary site. | The next replication cycle deletes the file from the secondary fileset. The next backup operation expires the file in the IBM Spectrum Protect server at the secondary site. |
| PEER SNAPSHOTS | There is no impact. | There is no impact. |
| FAILOVER | Because the primary fileset might not be available, no files are backed up. | There is no impact. |
| FAILBACK | If a backup operation is performed during failover, this situation might cause files to be skipped because they are dirty. The next backup cycle will back up these files. | There is no impact. |

### 2.1.2   Challenges

You might encounter the following challenges in the disaster-recovery user scenario:

- Backing up files at the primary site before replication will cause these files to be skipped because changed files that are not replicated are in a dirty state[4]. This is particularly important to consider if the primary site is disconnected from the secondary site for a longer period.

- The backup operation at the primary site might interfere with the asynchronous replication, causing race conditions on files.

- A backup operation at the primary and secondary sites might cause recalls of migrated files that have not been backed up. This might result in massive recalls delaying the backup operation. In addition, this might cause unwanted growth of data in the secondary and primary file systems.

- If the primary fileset is down (failover condition), the backup operation on the primary file system skips the files in the primary fileset.

- During failback, files from the secondary fileset are copied to the primary fileset, either in-band using the apply-updates process or out-of-band using copy utilities. The backup operation on the primary fileset can interfere with these processes, causing race conditions and skipped files.

- All operations that lead to new backup versions of files in the primary fileset will be asynchronously promoted to the secondary fileset and will cause the same backup operation at the secondary fileset. Because the secondary file system can contain several secondary filesets exported to several primaries, the amount of changed and newly created files can be large. The IBM Spectrum Protect backup environment that is used to protect the files in the secondary file system must be configured to achieve the expected backup performance and scalability.

### 2.1.3   Guidelines

The following guidelines apply to backup operations in the disaster-recovery user scenario:

- Configure the backup-archive client option `AFMSKIPUNCACHEDFILES YES`.

---

[4] The IBM Spectrum Protect V7.1.3 and later backup-archive client skips uncached and dirty files during backup operations.

- Plan the backup schedules at the primary and secondary sites to have the least interference with the replication operation. AFM replication is done automatically in the background from primary to secondary as specified by the asynchronous delay configuration parameter (`afmAsyncDelay`). If files change at the primary site, the files will be replicated to the secondary site. The following considerations apply when you schedule backup operations on the primary and secondary sites:
    - o Run backup operations when files at the primary site are rarely changed or created, for example, at night.
    - o Delay asynchronous replication before starting the backup operation by adjusting the AFM parameter `afmAsyncDelay` on the primary fileset before and after the backup operation.

- If the primary site is disconnected from the secondary site, suspend the backup operation at the primary site to prevent skipping many dirty files. If the primary site is reconnected to the secondary site, let the replication finish before resuming the backup operation.

- When using the same IBM Spectrum Protect server as the destination for backup operations from the primary and secondary sites, ensure that the file system names at the primary and secondary sites are distinct. Otherwise, all files from the primary and secondary sites are stored in the same IBM Spectrum Protect server file space. This situation can cause confusion. Furthermore, when using the same IBM Spectrum Protect server, provision different node names for the primary and secondary systems (proxy and target nodes).

- During failback, suspend the backup operation on the primary site until files are copied from the secondary to the primary site to avoid race conditions.

- To cope with the backup loads at the secondary site, certain filesets on the secondary file system can be excluded from backup operations by using exclude lists or by running fileset-level backup operations.[5]

---

[5] In IBM Spectrum Scale V4.1.1 and later, the `mmbackup` command supports backup operations to single filesets. You can use this command for AFM home filesets to apply the same backup policy to both home and cache.

# 3    User Scenario: Branch Office Caching

In this scenario, one or more caches in branch offices work with files centrally stored in a home site (see Figure 3). Home is the source of files and acts as a central location where files are stored, managed, and protected. Files in home can be seen in the caches without being present. Upon access, the file will be transferred from home to cache. A controllable alternative way of transferring files is by prefetching.

In this scenario, a cache is typically configured in SW or IW mode, rarely in RO mode. In SW or IW mode, the users at the cache site can create, modify, read, and delete files that are asynchronously replicated back to home. In IW mode, multiple caches can see and work with the same files exported by a single home. Care must be taken because there is no cross-cache locking. In RO mode, the cache can see and read all files stored at home, but files cannot be added, modified, or deleted at the cache site.

In the branch office scenario, caches are often represented by a smaller system providing less storage capacity. To cope with the data growth at the cache site, eviction might be used. Eviction files that are synced with home are stubbed in the cache, consuming almost no storage capacity. Eviction is triggered by the cache fileset quota and can be changed to evict files based on size or LRU.

The subsequent sections elaborate on this user scenario and consider different cases with HSM at home, HSM at home and in cache, and backup at home and in cache.

## 3.1    HSM at Home

In this section, we elaborate on the branch-office use case with HSM installed and configured on home, as shown in Figure 3.
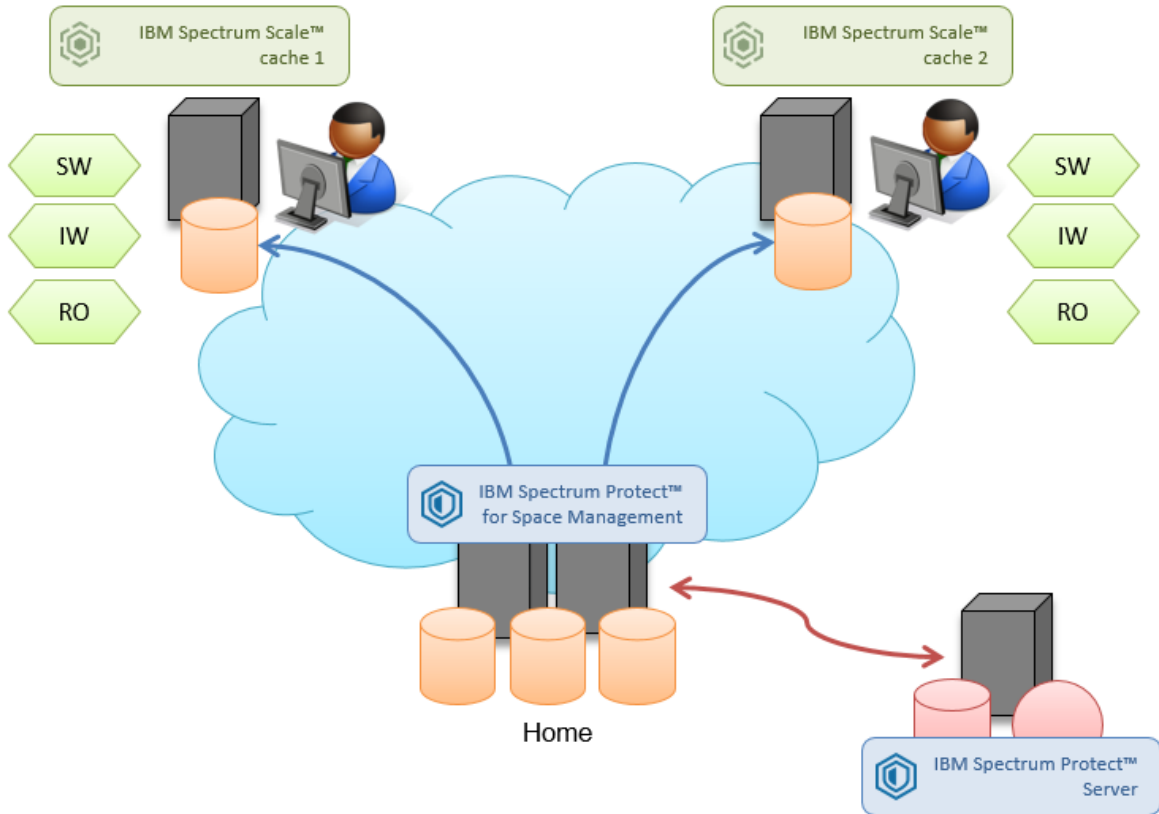
*Figure 3: Branch office setup with HSM at home*

AFM home is an IBM Spectrum Scale fileset contained in an IBM Spectrum Scale file system managed from UNIX HSM. The AFM cache fileset is configured in SW, IW, or RO mode and connects to the home fileset by using NFS. In SW and RO mode, home exports a fileset for each cache, so that multiple filesets might exist at home. In IW mode, there might be one fileset at home that is used by all caches.

Data that is stored in the home export can be seen in the cache fileset. Data created or modified in the cache fileset is replicated back to home asynchronously (SW or IW mode only). Prefetching might be required only if home has data before establishing the AFM relation to make this data available to the caches.

UNIX HSM at home is configured to manage the file system that contains one or more home filesets for one or more caches. Files might be created on home; however, care must be taken when modifying and deleting existing files at home because these operations might conflict with modifications of the same files on cache. In addition, files will be replicated from cache to the corresponding home fileset.

### 3.1.1 Cache Operations

The following table summarizes the file operations on cache and consequences to home and cache regarding UNIX HSM.

| Operation | Cache | Home |
|---|---|---|
| CREATE file | After creation, the file is replicated to the home fileset asynchronously. | After replication, the file becomes a candidate for migration in the home fileset. |
| READ file | If the file is uncached, it must be fetched from home. This might also occur after eviction. | If file must be fetched and is migrated in home, the file is recalled during the fetching operation. When multiple files are read at the same time (entire directory), the situation can result in massive recalls on home. |
| WRITE file | If the file is uncached, it must be fetched from home. This might also occur after eviction. After a write operation, the file is replicated to the home fileset asynchronously. If the replication causes recalls on home, the asynchronous replication might stall. | If file must be fetched and is migrated in home, the file is recalled during the fetching operation. When multiple files are written at the same time, the situation can result in massive recalls on home. After the write operation, during replication, if the file is migrated on home, this operation recalls the file. The subsequent write operation on home changes the file state to resident. When multiple files replicated from cache are in migrated state at home, this situation might cause massive recalls. |
| TRUNCATE file | There is no impact. A truncate operation on an uncached file | After a truncate operation, during replication, if the file is |

| Operation | Cache | Home |
|---|---|---|
| | does not cause fetching. After the truncate operation, file size is adjusted at home during asynchronous replication (no data is replicated). | not truncated to a size of 0 and it is migrated at home, this will cause a recall on home. The subsequent truncate operation at home changes the file state to resident. When multiple truncated files replicated from cache are in migrated state at home, massive recalls might result. |
| RENAME or MOVE file within fileset | There is no impact. Uncached files are not fetched. | There is no impact. Subsequent replication applies a move operation at home. UNIX HSM reconciliation assigns an updated name to migrated files. |
| MOVE file to other fileset | If the file is in uncached state, it will be fetched from home. Subsequent replication will remove the file from home. | If file must be fetched from home and is migrated at home, the file is recalled during the fetching operation. When multiple files are moved on cache (for example, a directory) and are migrated at home, massive recalls result. Subsequent replication removes the files from the original location at home. |
| EVICTION | The file will be set to an AFM uncached state in the cache fileset. | There is no impact. |
| DELETE file | There is no impact. Subsequent asynchronous replication deletes the file at home, and no data is transferred. | There is no impact. The file is deleted from home during the next replication operation. |

### 3.1.2 Challenges

- File modification in cache that requires replication to home can lead to the recall of the file on home if the file was migrated. Because many files may be replicated to home in each asynchronous replication cycle, massive recalls can occur on home. This situation can cause unwanted delays of the asynchronous replication. It might also cause the occupancy of the home fileset to grow unexpectedly.

- Evicted files on cache are fetched from home on access. If the file was migrated at home, this situation can lead to the recall of the file on home if the file was migrated. If multiple evicted files are accessed at the same time, massive recalls can occur on home. This situation can cause unwanted delays of the file operation on cache. It might also cause the occupancy of the home fileset to grow unexpectedly.

- The recall behavior of UNIX HSM is significantly influenced by the storage medium where the migrated data is stored in the IBM Spectrum Protect server. If the data is stored in tape pools, the recall of each file might take longer depending on the used tape technology (2-5 minutes).

### 3.1.3 Guidelines

- Files that will change frequently in the cache fileset and require frequent replications to the home fileset must be excluded from the UNIX HSM migration to the IBM Spectrum Protect server to prevent unwanted recalls. The access time stamp of files can be used to identify and exclude those files from the migration policy.

- The policy-driven threshold migration that is used to reduce the amount of space that is needed for data in the file system by applying UNIX HSM migration functions must be configured appropriately. You must ensure that enough candidates can be found to release space for the recalls required for the replication of changed data.

- When you size the home storage capacity, consider the recalls that might happen for different reasons, as described previously. Recalls cause the storage occupancy in the home fileset to grow and can cause race conditions with threshold migrations. If several cache filesets replicate data to home, additional storage capacity is required.

- When using eviction on cache, remember that access to files is promoted to home, where files might be migrated. This situation might cause more recalls and overload the home system. Therefore, eviction should either be disabled or coordinated with migration on home, for example, by using the age of files.

- If after a catastrophic failure, the cache must be rebuilt, the policy engine can be used to identify all files migrated on home and the UNIX HSM tape optimized recall function can be used to recall all these files prior to prepopulating or prefetching them to cache.

## 3.2    HSM at Home and Cache

In this section, we elaborate on the branch-office use case with HSM installed and configured at home and in cache, as shown in [Figure 4](#).
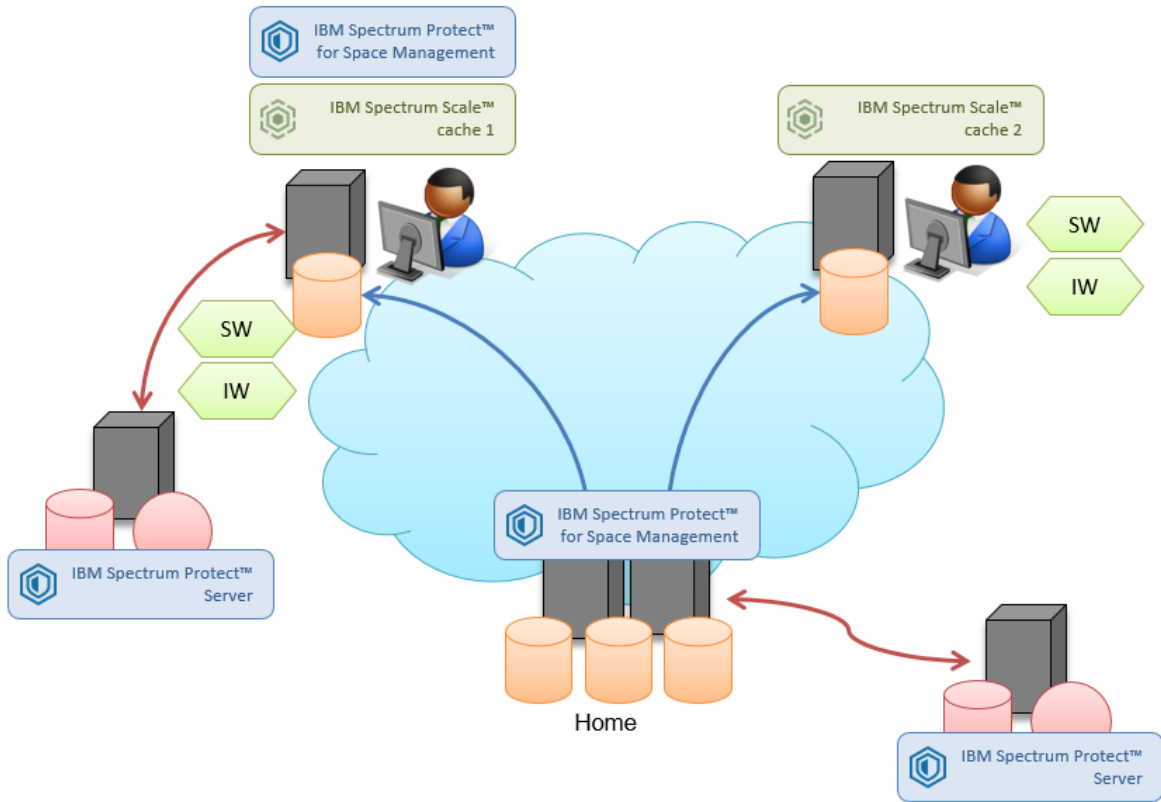


*Figure 4: Branch office use case with HSM at home and in cache*

AFM home is an IBM Spectrum Scale fileset contained in an IBM Spectrum Scale file system managed from UNIX HSM. The AFM cache fileset is managed by another instance of UNIX HSM that works independently from the instance configured on the home fileset (and other cache filesets). The cache fileset can be configured in SW or IW mode. Configuring the cache in RO mode does not make sense in combination with HSM on the cache site. The AFM cache fileset connects to the home fileset via NFS. In SW mode, home exports a fileset for each cache, so there might be multiple filesets at home. In IW mode, there might be one fileset at home that is used by all caches.

Data stored in the home export can be seen in the cache fileset. Data created or modified in the cache fileset is replicated back to home asynchronously. Prefetching might be required only if home has data before establishing the AFM relation, to make this data available to the caches.

One instance of UNIX HSM is configured to manage one or more home filesets for one or more caches. Files might be created at home; however, care must be taken when modifying and deleting existing files at home because these operations might conflict with modifications of the same files on cache. In addition, files will be replicated from cache to the corresponding home fileset.

### 3.2.1 Cache Operations

The following table summarizes the file operations on cache and consequences to home and cache in regard to UNIX HSM.

| Operation | Cache | Home |
|---|---|---|
| CREATE file | After creation, the file is replicated to the home fileset asynchronously. | After replication, the file becomes a candidate for migration in the home fileset. |
| READ file | If the file is migrated on cache, it will be recalled. When multiple files (entire directory) are read, this might cause massive recalls. If the file is uncached in cache (after eviction), it must be fetched from home. | If a file must be fetched and is migrated at home, the file recalled during the fetching operation. When multiple files are read at the same time (entire directory), this situation might cause massive recalls on home. |

| Operation | Cache | Home |
|---|---|---|
| WRITE file | If the file is migrated on cache, the file is recalled.<br>If the file is uncached in cache (after eviction), the file must be fetched from home.<br>After the write operation, the file is marked as dirty. The subsequent replication to home clears this state. Eventually, the file becomes a candidate for migration in cache[6]. If the replication causes recalls on home, the asynchronous replication may stall. | If file must be fetched and is migrated in home, the file is recalled during the fetching operation.<br>After the write operation, during replication, if the file is migrated at home, this operation recalls the file. The subsequent write operation on home changes the file state to resident. When multiple files replicated from cache are in migrated state at home, this situation might cause massive recalls. |
| TRUNCATE file | If the file is migrated on cache and the file is *not* truncated to a size of 0, it is recalled.<br>If the file is uncached, the truncate operation will *not* cause fetching.<br>After a truncate operation, the file size is adjusted at home during asynchronous replication (no data is replicated). | After a truncate operation, during replication, if the file is not truncated to a size of 0 and it is migrated at home, this situation will cause a recall on home. The subsequent truncate operation on home changes the file state to resident. When multiple truncated files replicated from cache are in migrated state at home, this situation might cause massive recalls. |
| RENAME or MOVE file within fileset | There is no impact. UNIX HSM reconciliation applies a new name to the migrated files. | After replication from cache, UNIX HSM reconciliation applies a new name to migrated files. |
| MOVE file to other fileset | If the file is migrated, it is recalled. When moving | If file must be fetched from home and is migrated at home, |

---

[6] IBM Spectrum Protect for Space Management V7.1.3 and later skips dirty files during the migration process. Nevertheless, the used migration policy should filter out uncached files to reduce the processing effort.

| Operation | Cache | Home |
|---|---|---|
| | multiple files (entire directory), this will cause massive recalls. If the file is in uncached state, it is fetched from home. Subsequent replication operations will remove the file from home. | the file is recalled during the fetching operation. When multiple files are moved on cache (entire directory) and are migrated at home, this situation causes massive recalls. |
| EVICTION | If the file is migrated on cache, the file will not be affected. After eviction, the AFM file state is set to uncached. There is no impact to HSM processing[7]. | There is no impact. |
| DELETE file | There is no impact. Subsequent reconciliation expires the file at the IBM Spectrum Protect server. Asynchronous replication will delete the file at home; no data is transferred. | There is no impact. The file is deleted at home during the next replication operation. UNIX HSM reconciliation expires the file at the IBM Spectrum Protect server. |

### 3.2.2  Challenges

Generally, the same challenges exist as with UNIX HSM at home. Beyond that, there are more challenges associated with UNIX HSM configured on cache:

- Migration of uncached and dirty files will generate warning messages. Uncached and dirty files won't be migrated[8].

- In general, eviction of files from the cache is not supported in combination with UNIX HSM. The usage of eviction might lead to unexpected behavior in terms of file operations like read, write, truncate, or move. Eviction of migrated files from cache will generate warning messages and those files are not evicted. Only whole

---

[7] IBM Spectrum Protect for Space Management V7.1.3 and later skips uncached files during the migration process. Nevertheless, the used migration policy should filter out uncached files to reduce the processing effort.

[8] IBM Spectrum Protect for Space Management V7.1.3 and later skips uncached files during the migration process. Nevertheless, the used migration policy should filter out uncached files to reduce the processing effort.

files are evicted.

- File operations on cache, such as read, write, truncate or move, will cause recalls on cache. If file operations that cause recalls are done for multiple files, massive recalls can occur on cache. This situation might cause file operations to be delayed and the occupancy of the cache fileset to grow unexpectedly.

- File write operations on cache might lead to recalls on both sites' cache and home. If the file is migrated on cache, it will be recalled on cache. If the file is migrated at home, the next replication will cause a recall on home. This might cause file operations such as replication to be delayed, and the occupancy of the file systems on cache and home to grow unexpectedly.

- The recall behavior of UNIX HSM is significantly influenced by the storage medium where the migrated data is stored in the IBM Spectrum Protect server. If the data is stored in tape pools, the recall of each file can take longer depending on the used tape technology (2-5 minutes).

### 3.2.3  Guidelines

Generally, the same guidelines apply as were previously described for UNIX HSM activity on home. In addition to that, there are guidelines associated with UNIX HSM configured on cache:

- Cache eviction must be disabled on the cache fileset for this scenario, because it can conflict with UNIX HSM operations.

- To prevent migration of uncached files,[9] the migration policy should omit uncached files. The following example defines uncached files that could be used to prevent the migration of uncached files.

```
/* define macro to identify uncached files */

define( is_uncached,
( NOT RegEx(misc_attributes,['[DXazu]'])
  AND NOT RegEx(mode,['[bcps]']) ) )
```

---

[9] IBM Spectrum Protect for Space Management V7.1.3 and later skips uncached files during migration processes. Nevertheless, the used migration policy should filter out uncached files to reduce the processing effort.

- A file is marked dirty if it has content changes on the cache fileset site that were not yet replicated to home. To prevent migration of files on cache that have not been replicated, the migration policy applied to the cache fileset must omit dirty files[10]. In the following example, a macro is defined to identify dirty files:

```
/* define macro to identify dirty files */

define( is_dirty,
( NOT RegEx(misc_attributes,['[DXaz]'])
  AND RegEx(misc_attributes,['[w]'])
  AND NOT RegEx(mode,['[bcps]']) ) )
```

- Files that will change frequently in the cache fileset should not be migrated. The policy engine can be configured to identify files that will not change, for example, by using the access time stamp. The following example shows a command that is used to migrate files accessed 5 days ago or earlier:

```
/* define macro for modification time */

define( acc_age,
( DAYS(CURRENT_TIMESTAMP) - DAYS(ACCESS_TIME) ) )

/* migration based on modification age*/

RULE 'modAge' MIGRATE
 FROM POOL 'system'
 TO POOL 'silver'
  WHERE (acc_age > 5)
```

- Files that will change frequently in the cache fileset and require frequent replications to the home fileset should not be migrated at home to prevent unwanted recalls. The access time stamp of files can be used to identify and exclude those files from the migration policy, as shown in the previous example.

- The threshold migration must be configured at home and in cache to ensure that sufficient candidates can be found to release space for the recalls that are required for the replication of changed data.

---

[10] IBM Spectrum Protect for Space Management V7.1.3 and later skips dirty files during the migration process. Nevertheless, the used migration policy should filter out dirty files to reduce the processing effort.

- The home and cache storage capacity must be appropriately sized to accommodate recalls that might happen for different reasons, as described previously. Recalls cause the storage occupancy in the home fileset to grow and can cause race conditions with threshold migrations.

- If after a catastrophic failure, the cache must be rebuilt from home, the policy engine can be used to identify all files migrated at home that belong to the failed cache. The UNIX HSM tape-optimized recall function can be used to recall all these files before prefetching them to cache. The home fileset must be excluded from threshold migration until all files are prefetched.

## 3.3    Backup at Home and Cache

In this section, we elaborate on the branch-office use case with HSM and the backup-archive client installed and configured at home and in cache, as shown in Figure 5.
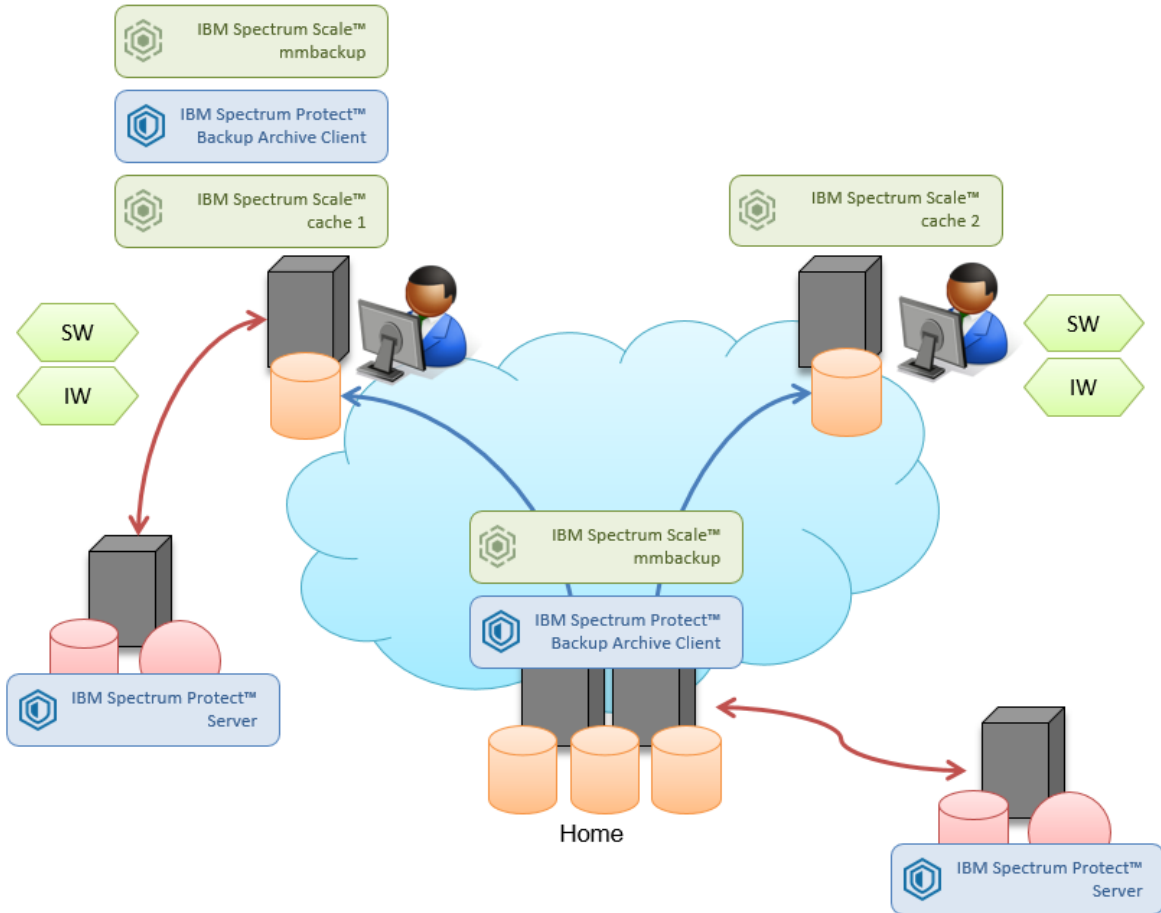
*Figure 5: Branch office use case with backup and HSM at home and cache*

The setup of the AFM solution corresponds to the setup described in HSM at Home. In addition to UNIX HSM being installed and configured on the file systems hosting the cache filesets and on the home filesets, managing file migration and recall independently between cache and home, the backup of these file systems is enabled.

The backup in IBM Spectrum Scale is accomplished with the `mmbackup` command, which uses the IBM Spectrum Protect backup-archive client. Backup operations for home and cache can be configured independently. The IBM Spectrum Protect server file space must be different for the home fileset and the cache fileset. In this way, you prevent concurrent access to the same data from both AFM sites: cache and home. Concurrent access would lead to inconsistent backups.

### 3.3.1    Cache Operations

The following table summarizes the file operations on cache and consequences to home and cache. From an HSM perspective, the behavior is identical to the use case with HSM at home and in cache; we will not repeat that information. Instead, we will focus on the backup operations.

| Operation | Cache | Home |
|---|---|---|
| CREATE file | The next backup generates a backup copy in the IBM Spectrum Protect server. | After replication, the next backup copies the file to the IBM Spectrum Protect server. |
| READ file | There is no impact to backup processing. | There is no impact to backup processing. |
| WRITE file | After the write operation, the modified file is replicated back to home asynchronously.<br>The next backup operation generates a new version of the file in the IBM Spectrum Protect server on cache. | After the write operation, the file is replicated back to home. The next backup operation generates a new version of the file in the IBM Spectrum Protect server at home. |
| TRUNCATE file | The next backup operation generates a new version of the file in the IBM Spectrum Protect server. | The truncate operation is promoted to home without transferring the file during the next replication cycle.<br>The next backup operation generates a new version of the file in the IBM Spectrum Protect server. |
| RENAME or MOVE file within fileset | The next backup operation generates a new file in the IBM Spectrum Protect server and expires the old file (name). | A move operation is promoted to home without transferring the file during the next replication cycle. The next backup operation generates a new file in the IBM Spectrum Protect server and expires the old file (name) |
| MOVE file to other fileset | The next backup operation expires the moved file. | The next backup operation expires the moved file. |
| EVICTION | Eviction should be disabled. | There is no impact. |

| Operation | Cache | Home |
|---|---|---|
| DELETE file | During the next backup operation, the file is expired in the IBM Spectrum Protect server. | During the replication after the delete operation, the file is deleted from home. The next backup operation expires the file in the IBM Spectrum Protect server. |

### 3.3.2   Challenges

Generally, the same challenges as with HSM at home and in cache apply here. However, there are more challenges associated with backup configured on cache:

- Uncached and dirty files are not backed up. This might cause inconsistency of the backed-up data in the cache if files that are uncached have not been backed up.

- The backup operation on cache and home might cause recalls of migrated files that have not been backed up. This might result in massive recalls delaying the backup operation. In addition, this might cause unwanted growth of data in the home and cache file systems.

- The backup operation on cache might interfere with the asynchronous replication to home, causing missed files and race conditions.

- The replication operation to home might interfere with the backup operation on home, causing missed files and race conditions.

- All operations that lead to new backup versions of files in the cache fileset will be asynchronously promoted to home and will cause the same backup operation at home. Because the home file system can contain several home filesets that are exported to several caches, the number of changed and newly created files can be high. Ensure that the IBM Spectrum Protect backup environment that is used to protect the files in the home file system is configured to achieve the expected backup performance and scalability.

- The backup of a file that is evicted from cache causes the file to be replicated from home to cache. If the files are migrated at home, this situation might cause massive recalls on home.

### 3.3.3 Guidelines

Generally, the same guidelines as with HSM at home and in cache apply. In addition, there are challenges associated with backup operations configured on cache and home:

- Do not use eviction and backup for the same fileset. If eviction is used on the cache site, ensure that the backup operation happens before eviction. Because evicted (uncached) files are skipped during the backup operation, this situation can lead to inconsistencies in terms of backup versions in the IBM Spectrum Protect server.

- When using HSM and backup for the same IBM Spectrum Scale file system and fileset, use the same IBM Spectrum Protect server. In this way, you prevent files from being migrated before they have been backed up by using the management class parameter `MIGREQUIRESBkup=yes`.

- Do not use the same instance of IBM Spectrum Protect server as the destination for UNIX HSM on cache and home. This practice defeats the idea of disaster protection because if this single IBM Spectrum Protect server instance fails, both copies of migrated files might be lost. However, if you are using only backup operations on cache and home, the same IBM Spectrum Protect server instance can be used. Ensure that the name of the file system mount point is different for the AFM cache fileset and the home fileset.

- The backup schedules on home and cache should take into consideration the replication cycles. AFM replication is done automatically in the background from cache to home. If files change in cache, the files are replicated to home. You can use two options to accommodate this:
    o Run backup operations when files on cache are rarely changed or created, for example, at night.
    o Delay asynchronous replication before starting the backup operation by adjusting the AFM parameter `afmAsyncDelay` on the cache fileset before and after the backup operation.

- When the cache must be re-created (after a failure), prefetch all files back from home to cache before starting the backup operation. In this way, you can ensure

that subsequent backups do not cause fetching of files from home to cache and perhaps recalls on home.

- When prefetching files from home, the policy engine can be used on home to identify all files that have been migrated. The UNIX HSM tape-optimized recall method should be used to recall all these files to home before prefetching them to cache.

- To help process the backup loads at the home site, certain filesets at the home file system can be excluded from backup operations by using exclude lists or performing fileset-level backups.

- Eviction is disabled for all cache filesets, even if UNIX HSM is not configured because backup of files on cache can cause uncached (evicted) files to be fetched from home. In addition, if these files are migrated at home, that situation can cause massive recalls and the backup operation to be delayed. In addition, this can cause unwanted growth of occupancy in the home file system.

# 4    User Scenario: System Migration

In this scenario, AFM replication is used to perform a system migration from a legacy file system to an IBM Spectrum Scale file system. The cache fileset is in an IBM Spectrum Scale file system. The corresponding home can be an IBM Spectrum Scale fileset or a different storage device that is accessible via NFS mount. Each cache fileset corresponds to one home fileset or one NFS-mounted storage device. The cache does all the work.

The cache fileset is typically configured in LU mode. In LU mode, files from home are visible in cache but not yet present (uncached state). Files are fetched from home on access or by prefetching. Any changes done to files on cache are not replicated back to home. Therefore, the AFM relation for each file is broken when the file is modified. Newly created files on cache are also not replicated from home.

System migration is done by connecting a cache in LU mode to an NFS export provided by home. Files can be immediately seen in cache without being present there. Thus, users could immediately start working on cache, whereby uncached files are fetched from home on access. In most cases, a subset of files, such as the most active ones, are copied from home to cache by using AFM prefetching, before users and applications use the new system (cache). After the required files are available on cache, the users and application can use the cache while more files are fetched from home. If all files are fetched from home to cache, the legacy filer (home) can be discontinued.

In many cases, a cache is represented by a larger system that provides more storage capacity. One or more cache filesets might be used to consolidate multiple legacy storage devices (each represented by NFS export) or legacy IBM Spectrum Scale file systems.

---

**Restriction:** Eviction must not be used with the system-migration use case because eviction defeats the purpose of data migration.

---

## 4.1    HSM at Home

In this section, we elaborate on the system-migration use case with HSM installed and configured at home, as shown in Figure 6.
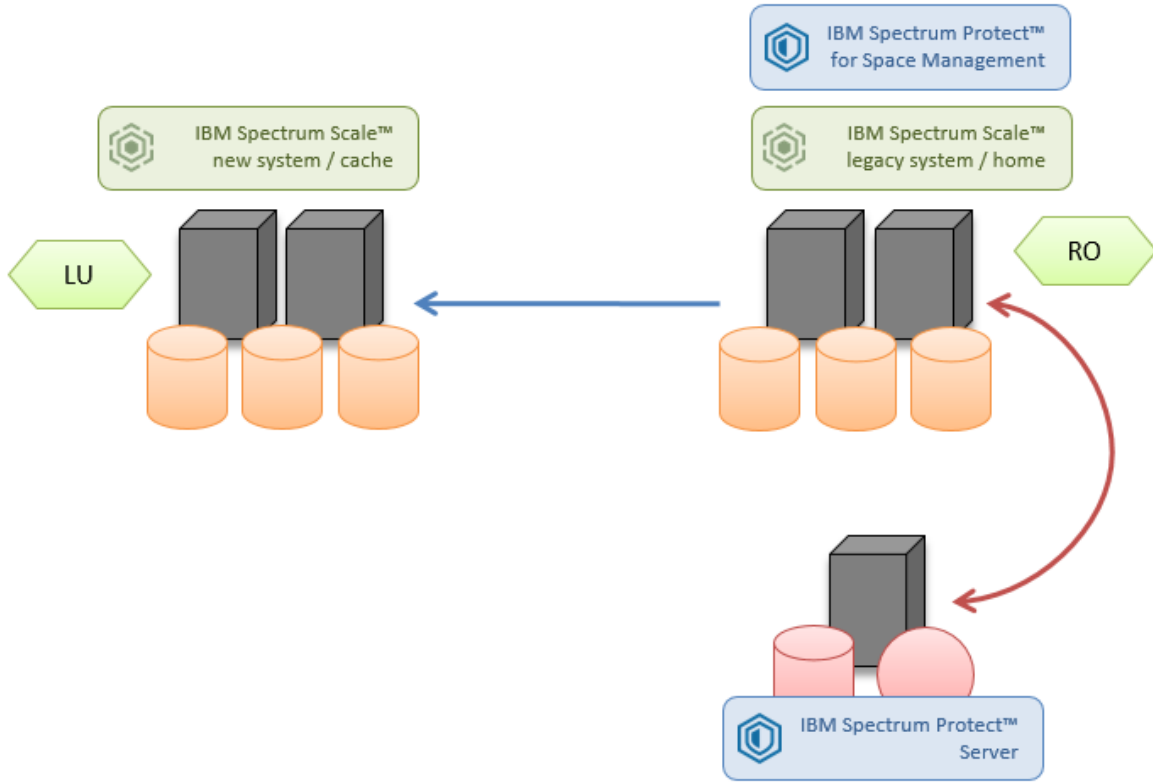
---

*Figure 6: System migration setup with HSM at home*

Home is an IBM Spectrum Scale fileset contained in an IBM Spectrum Scale file system that is managed from UNIX HSM. The AFM cache fileset is running in LU mode. File content changes in the cache fileset are not replicated to the home fileset.

Because UNIX HSM was used on the legacy system in the past, you can assume that many files that were in migration state were migrated and must be recalled for the system migration. Threshold migration should be enabled at home to prevent an out-of-space condition in terms of the expected file recalls for the system migration.

### 4.1.1   Cache Operations

| Operation | Cache | Home |
|-----------|-------|------|
| PREFETCHING | File data is created in cache. | Prefetching of a file will lead to a recall of the file to the home fileset if the file was in migrated state. This situation increases the threshold of the file system. |

| Operation | Cache | Home |
|---|---|---|
| | | There is no impact if the file is in migration state premigrated. |
| CREATE file | The file is created in cache, but no replication to home will occur in LU mode. | There is no impact to HSM processing. |
| READ file | If the file is in uncached state, the file must be fetched from home. | If file must be fetched and is migrated in home, the file will be recalled during the fetching operation. When multiple files are read at the same time (entire directory), massive recalls on home might result. |
| WRITE file | If the file is uncached, it must be fetched from home. | If the file must be fetched and is migrated at home, the file will be recalled during the fetching operation. When multiple files are written at the same time, massive recalls on home might result. |
| TRUNCATE file | The file is truncated to the new size. | There is impact to HSM processing. |
| RENAME or MOVE file within fileset | There is no impact. Uncached files won't be fetched. | There is no impact to HSM processing. |
| MOVE file to other fileset | If the file is in uncached state, the file will be fetched from home. | If the file must be fetched from home and is migrated at home, the file is recalled during the fetching operation. When multiple files have been moved on cache (directory) and are migrated at home, massive recalls will occur. |
| EVICTION | N/A | Eviction must be disabled. |
| DELETE file | The file will be deleted from cache. | There is impact to HSM processing. |

### 4.1.2   Challenges

- After the AFM relation between cache and home has been established, files can be accessed in the cache fileset. Any access will lead to a fetching of the file data to the cache fileset. This will lead to random recalls of files in the home fileset, if files are migrated at home. Because no access optimization (ordering of recalls) will be applied by default for the recall of the files, a significant performance impact can be assumed.

- The recall behavior of UNIX HSM is significantly influenced by the storage medium where the migrated data is stored in the IBM Spectrum Protect server. If the data is stored in tape pools, the recall of each file may take longer depending on the used tape technology (2 - 5 minutes).

- Because the home fileset or legacy system uses the UNIX HSM function for a long time, it can be assumed that a huge number of files are in migration state migrated at the home fileset. Therefore, prefetching can lead to a mass recall of files to the home fileset.

### 4.1.3   Guidelines

- To prevent random mass recalls of migrated files caused by the AFM pre-fetching or user access, take the following actions:

   o   Generate the list of all migrated files on the legacy system (preferred processing on IBM Spectrum Scale: usage of the policy engine). Files in premigrated state must not be listed. Those files can be read without recall.

   o   Initiate tape-optimized recall processing in prepare mode to obtain information about the location of files and their sizes. This information can be used to calculate the space required for the recall of the files.

   o   Initiate a tape-optimized recall of a subset of all migrated files to file state premigrated. Ensure that the defined subset of files will fit into the free space available in the home fileset.

   o   Use the same file list to initiate the prefetching of files to the cache fileset.

- o After data replication is finished, re-migrate the files on the legacy system. (Because the files were recalled to premigrated state, the re-migration is a local deletion process, which doesn't require data transfer.)

- The home storage capacity must be appropriately sized to handle the massive number of files that will be recalled. You might have to temporarily add storage capacity to the home system. Alternatively, the recall and prefetch process must be executed in a way to prevent home from becoming full.

- The threshold migration must be configured to ensure that enough candidates can be found to release space for the recalls that are required for the prefetching of files to the cache.

## 4.2 HSM at Home and Cache

In this section, we elaborate on the system-migration use case with HSM installed and configured on home and cache, as shown in Figure 7.
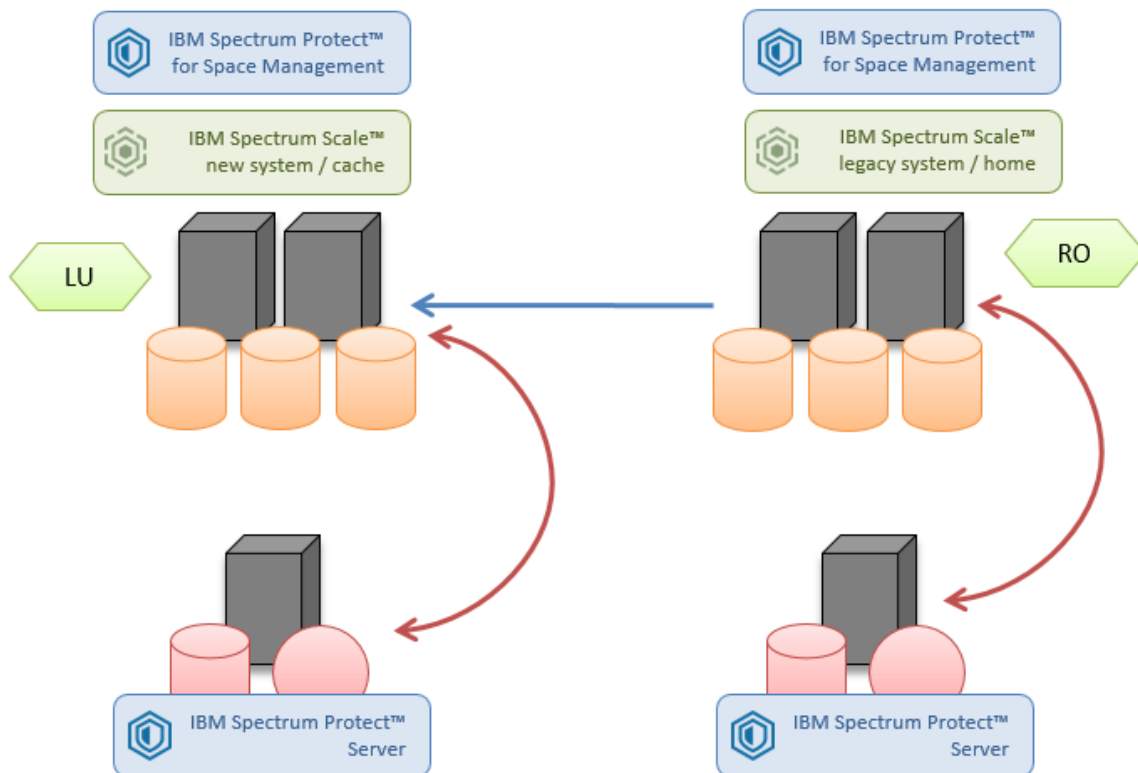


*Figure 7: System migration setup with HSM at home*

Home is an IBM Spectrum Scale fileset contained in an IBM Spectrum Scale file system that is managed from UNIX HSM. The AFM cache fileset is running in LU mode. File content changes in the cache fileset are not replicated to the home fileset.

Because UNIX HSM was used on the legacy system in the past, it can be assumed that many files that are in migration state were migrated and must be recalled for the system migration. Threshold migration will be enabled on home to prevent an out-of-space condition in terms of the expected file recalls for the system migration.

### 4.2.1   Cache Operations

| Operation | Cache | Home |
|---|---|---|
| PREFETCHING | File data is created in cache. | Prefetching of a file will lead to a recall of that file in the home fileset if the file was in migrated state. This situation will increase the threshold of the file system. There is no impact if the file is in premigrated state. |
| CREATE file | The file is created in cache, but no replication to home will occur in LU mode. The file becomes a candidate for migration. | There is no impact to HSM processing. |
| READ file | If the file is uncached in cache state, the file must be fetched from home. A read operation might take a while if the file must be replicated from home and is migrated at home. When multiple files are read at the same time, this might cause recall storms on home and timeouts of the read operations. If the file is migrated on cache, it will be recalled. When multiple | If a file must be fetched and is migrated at home, the file is recalled during the fetching operation. When multiple files are read at the same time (entire directory), massive recalls on home can occur. |

| Operation | Cache | Home |
|---|---|---|
| | files (entire directory) are read, massive recalls might result. | |
| WRITE file | If the file is uncached in cache, the file must be fetched from home.<br>After a write operation, the file will become candidate for migration in cache.<br>If the file is migrated on cache, the file will be recalled. | If the file is in uncached state in the cache and must be fetched and is migrated at home, the file will be recalled during the fetching operation. When multiple files are written at the same time, massive recalls on home can occur. |
| TRUNCATE file | If the file is migrated on cache and the file is *not* truncated to a size of 0, the file will be recalled.<br>If the file is uncached, the truncate operation will *not* cause fetching. | There is no impact to HSM processing. |
| RENAME or MOVE file within fileset | There is no impact. Uncached files won't be fetched.<br>UNIX HSM reconciliation will assign a new name to migrated files. | There is no impact to HSM processing. |
| MOVE file to other fileset | If the file is migrated, it is recalled from the IBM Spectrum Protect server. When moving multiple files (entire directory), massive recalls occur.<br>If the file is in uncached state, the file is fetched from home. | If a file must be fetched from home and is migrated at home, the file is recalled during the fetching operation. When multiple files have been moved on cache (an entire directory) and are migrated at home, massive recalls will occur. |
| EVICTION | Eviction must be disabled. | Eviction must be disabled. |
| DELETE file | The file is deleted from cache. Subsequent UNIX HSM reconciliation will remove the file from the IBM Spectrum Protect server. | There is no impact to HSM processing. |

| Operation | Cache | Home |
|-----------|-------|------|
|  | There is no replication to home in LU mode. |  |

### 4.2.2 Challenges

Generally, the same challenges apply as with UNIX HSM at home (see HSM at Home). Additional challenges are associated with UNIX HSM configured on cache:

- Migration of uncached files will generate warning messages. Uncached files won't be migrated[11].

- For migrated files in cache, file operations such as read, write, truncate, or move will cause recalls on cache. If file operations causing recalls are done for multiple files, massive recalls can occur on cache. This might cause file operations to be delayed and the occupancy of the cache fileset to grow unexpectedly.

- The recall behavior of UNIX HSM is significantly influenced by the storage medium where the migrated data is stored in the IBM Spectrum Protect server. If the data is stored in tape pools, the recall of each file might take longer depending on the used tape technology (2-5 minutes).

### 4.2.3 Guidelines

Generally, the same guidelines apply as described previously for HSM activity on home (see HSM at Home). In addition, there are guidelines associated with HSM configured on cache:

- To prevent migrating uncached files[12] (which should not occur in this scenario), the migration policy should omit uncached files. The following example shows how to define a macro to identify uncached files:
  ```
  /* define macro to identify uncached files */
  ```

---

[11] IBM Spectrum Protect for Space Management V7.1.3 and later skips uncached files during the migration process. Nevertheless, the used migration policy should filter out uncached files to reduce the processing effort.

[12] IBM Spectrum Protect for Space Management V7.1.3 and later skips uncached files during the migration process. Nevertheless, the used migration policy should filter out uncached files to reduce the processing effort.

```
define( is_uncached,
( NOT RegEx(misc_attributes,['[DXazu]'])
   AND NOT RegEx(mode,['[bcps]']) ) )
```

- Files that will change frequently in the cache fileset should not be migrated. The policy engine can be configured to identify files that do not change anymore, for example, by using the access time stamp. In the following example, a macro is defined to migrate files that were accessed 5 days ago or earlier:

```
/* define macro for modification time */

define( acc_age,
( DAYS(CURRENT_TIMESTAMP) - DAYS(ACCESS_TIME) ) )

/* migration based on modification age*/

RULE 'modAge' MIGRATE
 FROM POOL 'system'
 TO POOL 'silver'
  WHERE (acc_age > 5)
```

## 4.3    Backup at Home and Cache

In this section, we elaborate on the system-migration use case with the backup-archive client installed and configured on home and on cache, as shown in Figure 8.
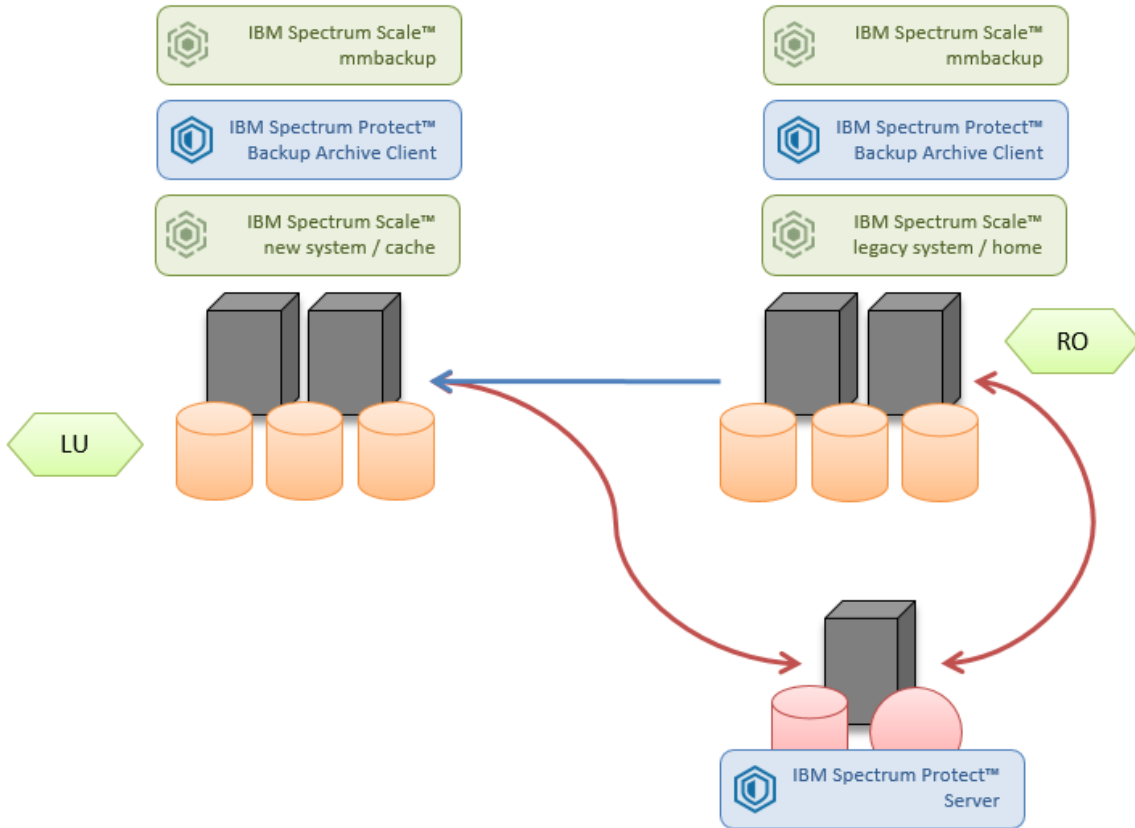
*Figure 8: Backup at home and in cache*

The backup in IBM Spectrum Scale is accomplished with the `mmbackup` command, which uses the IBM Spectrum Protect backup-archive client. Backup of home and cache can be configured independently. The IBM Spectrum Protect server file space must be different for the home fileset and the cache fileset. In this way, you prevent concurrent access to the same data from both AFM sites: cache and home. Concurrent access would lead to inconsistent backups.

### 4.3.1 Cache Operations

The following table summarizes the file operations on cache and consequences to home and cache with a focus on backup operations:

| Operation | Cache | Home |
|---|---|---|
| CREATE file | The next backup operation generates a backup copy in the IBM Spectrum Protect server. | There is no impact to backup processing. |
| READ file | There is no impact to backup | There is no impact to backup |

| Operation | Cache | Home |
|---|---|---|
| | processing. | processing. |
| WRITE file | After the write operation, the modified file is not replicated to home.<br>The next backup operation generates a new version of the file in the IBM Spectrum Protect server on cache. | There is no impact to backup processing. |
| TRUNCATE file | The next backup operation generates a new version of the file in the IBM Spectrum Protect server. | There is no impact to backup processing. |
| RENAME or MOVE file within fileset | The next backup operation generates a new file in the IBM Spectrum Protect server and expires the old file (name). | There is no impact to backup processing. |
| MOVE file to other fileset | The next backup operation expires the moved file. | There is no impact to backup processing. |
| EVICTION | Eviction should be disabled. | There is no impact to backup processing. |
| DELETE file | The next backup operation expires the deleted file. | There is no impact to backup processing. |

### 4.3.2 Challenges

The main challenge for backup operations on home and cache is that uncached files won't be backed up. This situation can cause inconsistency of the backed-up data in the cache if files that are uncached have not been backed up.

### 4.3.3 Guidelines

Guidelines for the usage of backup operations in the system-migration user scenario:

- Ensure that files on cache that are in uncached state are previously backed up. Fetch all files from home to cache before activating backup (and HSM). In general,

do not have uncached files in cache. Fetch all files from home to cache before activating backup (and HSM).

- To help process the backup loads, certain filesets can be excluded from backup operations by using exclude lists or performing fileset-level backups.

- When using HSM and backup operations for the same IBM Spectrum Scale file system and fileset, use the same IBM Spectrum Protect server. In this way, you can prevent files from being migrated before they are backed up by using the management class parameter: `MIGREQUIRESBkup=yes`.

- Do not use the same instance of IBM Spectrum Protect server as destination for UNIX HSM on cache and home. This defeats the idea of disaster protection because if this single IBM Spectrum Protect server instance fails, both copies of migrated files might get lost. However, when using only backup on cache and home, the same IBM Spectrum Protect server instance might be used. In this case, however, it is important that the file space of the AFM cache fileset is different than the home fileset. This can be achieved by selecting a different name for the root directory of the fileset on cache and home.

- The backup schedules on home and cache should take into consideration the replication cycles. AFM replication is done automatically in the background from cache to home. If files change in cache, the files are replicated to home. You have two options to accommodate this:
  - Run backup operations when files on cache are rarely changed or created, for example, at night.
  - Delay asynchronous replication before starting the backup operation by adjusting the AFM parameter `afmAsyncDelay` on the cache fileset before and after the backup operation.

- When the cache must be created (after a failure), prefetch all files back from home to cache before starting the backup operation. In this way, you help to ensure that subsequent backups do not cause fetching of files from home to cache and perhaps recalls on home.

- When prefetching files from home, the policy engine can be used at home to identify all files that were migrated. The UNIX HSM tape-optimized recall method should be used to recall all these files to home before prefetching them to cache.

# APPENDIX

## References

IBM Spectrum Protect on IBM Knowledge Center:

https://www.ibm.com/support/knowledgecenter/SSEQVQ/landing/welcome_sseqvq.html

IBM Elastic Storage Server on IBM Knowledge Center:

https://www.ibm.com/support/knowledgecenter/P8ESS/p8ehc/p8ehc_storage_landing.htm

Tivoli Field Guide - IBM Spectrum Protect for Space Management for UNIX-IBM Spectrum Scale
Integration:
https://www.ibm.com/support/pages/node/324677

Tivoli Field Guide - UNIX HSM in a mixed IBM Spectrum Scale cluster environment:

http://www.ibm.com/support/docview.wss?uid=swg27028178

Tivoli Field Guide - Performance of IBM Spectrum Scale/HSM - IBM Spectrum Protect for Space
Management: Part 1: Automigration:

http://www.ibm.com/support/docview.wss?uid=swg27020987

Data Storage Management (XDSM) API:

http://pubs.opengroup.org/onlinepubs/9657099/

# Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing*
*IBM Corporation*
*North Castle Drive, MD-NC119*
*Armonk, NY 10504-1785*
*US*

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

icensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Director of Licensing*
*IBM Corporation*
*North Castle Drive, MD-NC119*
*Armonk, NY 10504-1785*
*US*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBMproducts. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

UNIX is a registered trademark of The Open Group in the United States and other countries.