GPFS AFM to Cloud

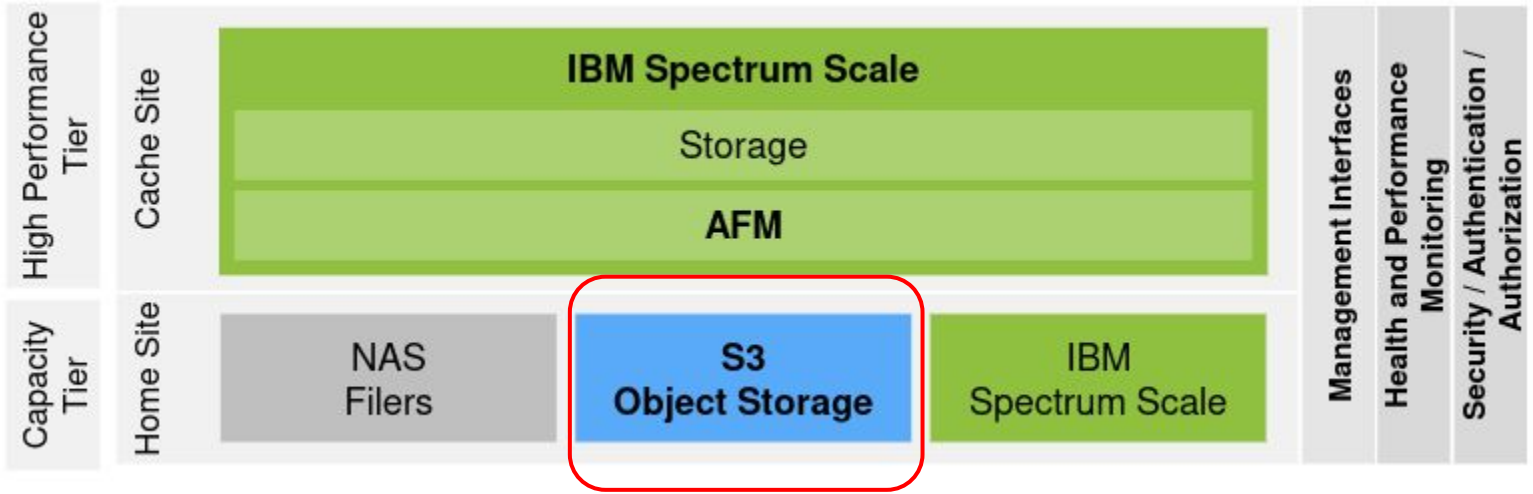
Enabling transparent access to Object Storage

Vladimir Sapunenko (vladimir.sapunenko@cnaf.infn.it)
Federico Fornari (federico.fornari@cnaf.infn.it)

Agenda

- New features & improvements in Spectrum Scale (GPFS) v.5
- Accessing Cloud Object Storage from GPFS
- Setup
- Pros and cons
- Demo

AFM: in GPFS 5.1 added S3 object protocol as an additional target



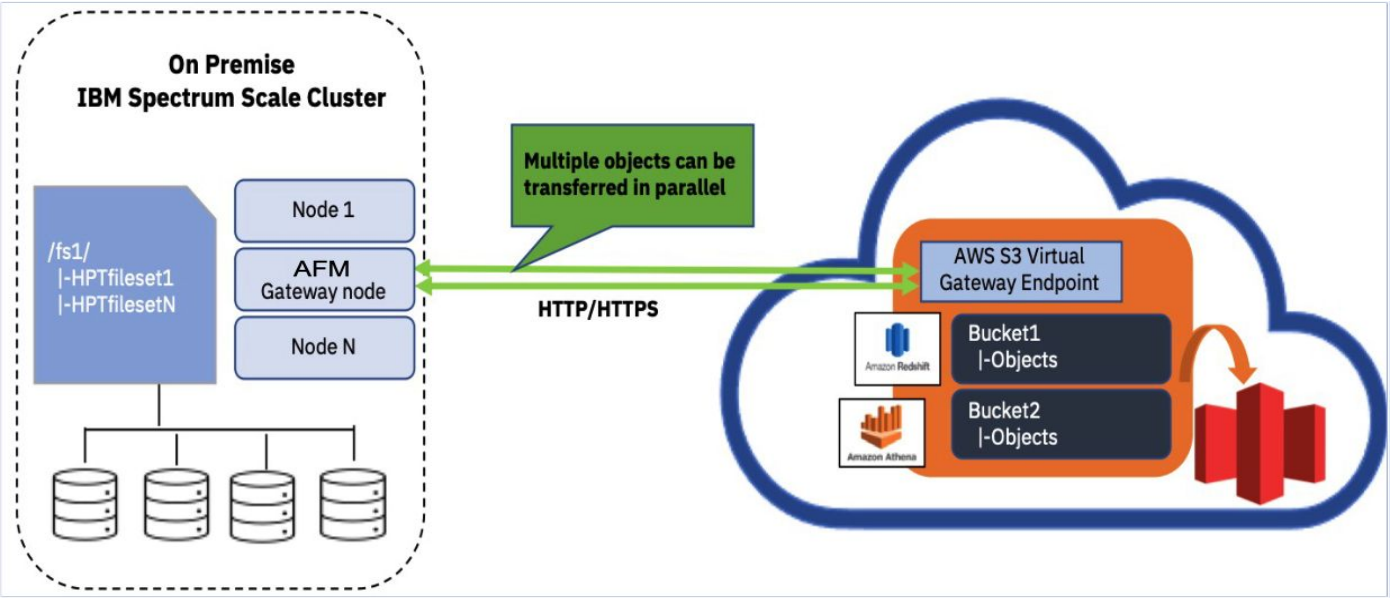
AFM Performance Updates

- AFM resync version 2 improves replication on heavily stressed systems
  - Updates to message queuing to improve AFM resync and recovery
  - Lower memory usage on gateway nodes with faster replication
  - Faster recovery and resync after gateway node failures
  - Improved role reversal in AFM-DR
  - See ‘AFM resync version 2’ in the knowledge center, and the afmResyncVer2 parameter in mmchfileset to activate this feature
- NFSv4 support for AFM replication
  - Allows NFSv4 to be used as the underlying AFM protocol for data transfer
- New queries and statistics available in mmafmctl
  - Query uncached and dirty files, and new read and write statistics
  - See mmafmctl man page for more details

Key features

- The AFM to cloud object storage (COS) enables placement of files or objects in a GPFS cluster to an object storage.
- The AFM to COS allows associating a GPFS fileset with a cloud object storage.
- The data from cloud or local object storage can be cached on AFM to COS filesets for faster computation and synchronize back to the cloud object storage server.
- The front-end for data access is an AFM to COS fileset connected to the object storage buckets in the background providing high performance.
- Cloud object storage can be used as a backup of important data.

As it’s been originally designed



The AFM to cloud object storage on an IBM Spectrum Scale fileset becomes an extension of cloud object storage buckets for high-performance or used objects.

AFM as extension of cloud storage buckets

- The objects that are created by applications in cache can be synchronized to the objects on a cloud object storage asynchronously.
- An AFM to cloud object storage fileset can cache only metadata or both metadata and data.
- The AFM to cloud object storage also allows data center administrators to free the IBM Spectrum Scale storage capacity by moving less useful data to the cloud storage.
- This feature reduces capital and operational expenditures.
- The AFM-based cache eviction feature can be used to improve the storage capacity manually and by using policies.
- The AFM to cloud object storage uses the same underlying infrastructure as AFM.
- The AFM to cloud object storage is available on all IBM Spectrum Scale editions.

AFM to cloud object storage limitations

- Hard links are supported only in the Local-Update (LU) mode. The creation of hard links fails with permission denied or `E_PERM` error on other modes.
- The primary resources of object servers are buckets and objects.
- Directory names are prefix to the objects. Empty directories are not replicated to the target object server.
- The ChangeTimes operation is supported on the cache filesets. However, this operation does not replicate to the target cloud object server.
- To replicate the chmod and chown metadata operations to a cloud object storage, you need to use the mmafmcosconfig command with the --xattr option.
- The rename operation is not supported on a non-empty directory. When a non-empty directory is renamed, the rename operation fails with the `E_NOTEMPTY` error. However, empty directories and local directories can be renamed.

Limitations (cont.)

- Parallel data transfer for write operations is not supported on an AFM to COS fileset. Objects are not synchronized by splitting chunks across gateway nodes but objects are queued on different gateway nodes based on the mapping.
- Deletion of objects on the COS synchronizes the deletion of files on the cache. Empty directories might remain on the cache.
- Support of file and directory naming convention is in accordance with the COS server supported guidelines. Some cloud object storage servers do not support special characters. The file or directory name must not contain special characters.
- File paths that have more characters than maximum characters limitation are not supported.
- AFM to COS commands (mmafmcoskeys, mmafmcosconfig, mmafmcosctl, and mmafmcosaccess) only supported on Linux.
- Immutability and appendOnly features are not supported on AFM to COS filesets.
- The --iam-mode option is not supported on AFM to COS filesets.
- If a file is evicted from AFM to COS fileset, the snapshot file of the evicted file contains zeros.
- AFM to COS supports Amazon S3 and IBM Cloud® Object Storage. - vero ma incompleto
- AFM to COS fileset supports setting an access control list (ACL) on a file of up to 2 KB size. When an ACL is assigned on a file of more than 2 KB size, the file is discarded and only file data is synchronized with the bucket.
- The symlinks that are created on an AFM to COS local update mode (LU) fileset cannot be uploaded manually to the bucket.
- A dependent fileset linking is not supported in an AFM to COS fileset.

Steps to setup

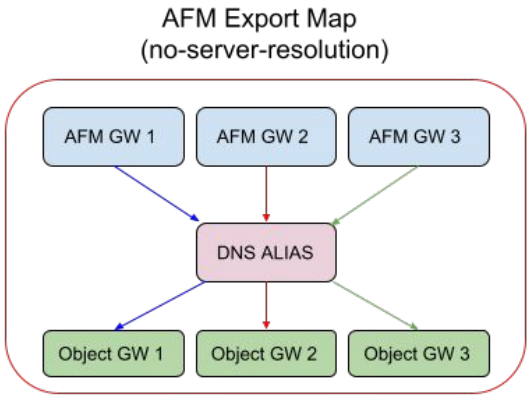
- Ensure that the **gpfs.afm.cos** package is installed on the gateway nodes.
- Create an export map by using multiple gateway nodes (use hostnames and --no-server-resolution option if an endpoint resolves to multiple IP addresses)
  - `mmafmconfig add ceph_map --export-map 131.154.129.89/ds-011,131.154.129.90/ds-010`
- Set up an access key and a secret key for the bucket by using the export map.
  - `mmafmcoskeys objectstoreexpone:ceph_map set pFcT4AWGCPcIkql9 8b2HxCYsgCd3Vx2P`
  - `mmafmcoskeys all get --report`
  - `mmafmcosconfig gpfs ObjectStorExp1 --endpoint http://ceph_map --uid 0 --gid 0 --bucket objectstoreexpone --mode sw --object-fs --xattr --cleanup –acls`
- Tune the parallel transfer thresholds for parallel reads. The afmParallelReadThreshold parameter value is 1 GB and the afmParallelReadChunkSize parameter value is 512 MB.
  - `mmchfileset gpfs ObjectStorExp1 -p afmParallelReadThreshold=1024 -p afmParallelReadChunkSize=536870912`

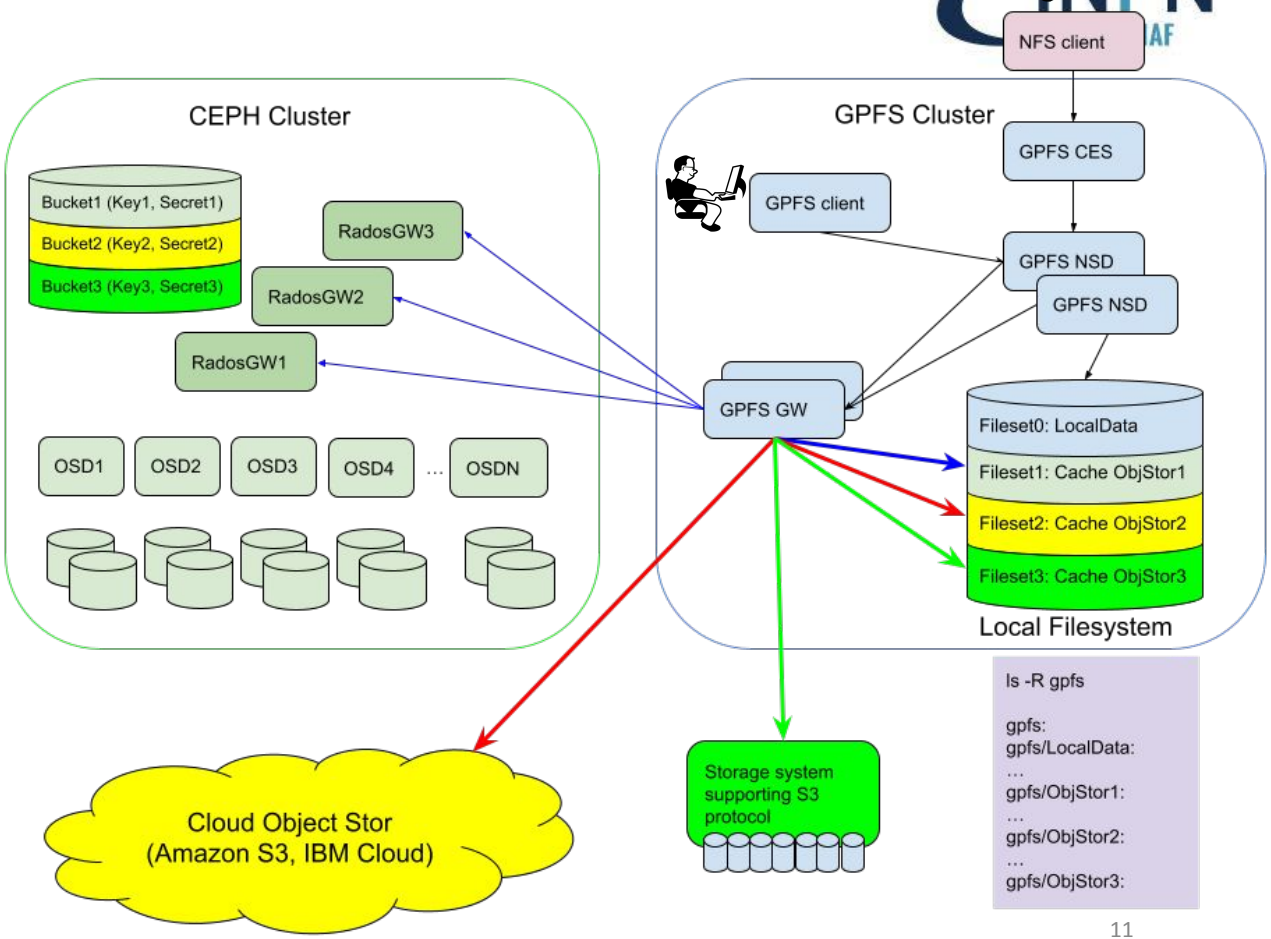
```
# mmafmctl gpfs getstate

Fileset Name    Fileset Target                          Cache State          Gateway Node    Queue Length    Queue numExec
------------    --------------                          -------------        ------------    ------------    -------------
ObjectStorExp1  http://ceph_map:80/objectstoreexpone     Active               ds-011          0               81
ObjectStorExp2  http://ceph_map:80/objectstoreexptwo     Active               ds-010          0               6
```

https://www.ibm.com/docs/en/spectrum-scale/5.1.2?topic=iacos-afm-cloud-object-storage-parallel-read-data-transfer

GPFS Cache to CEPH, COS and S3 enabled storage system

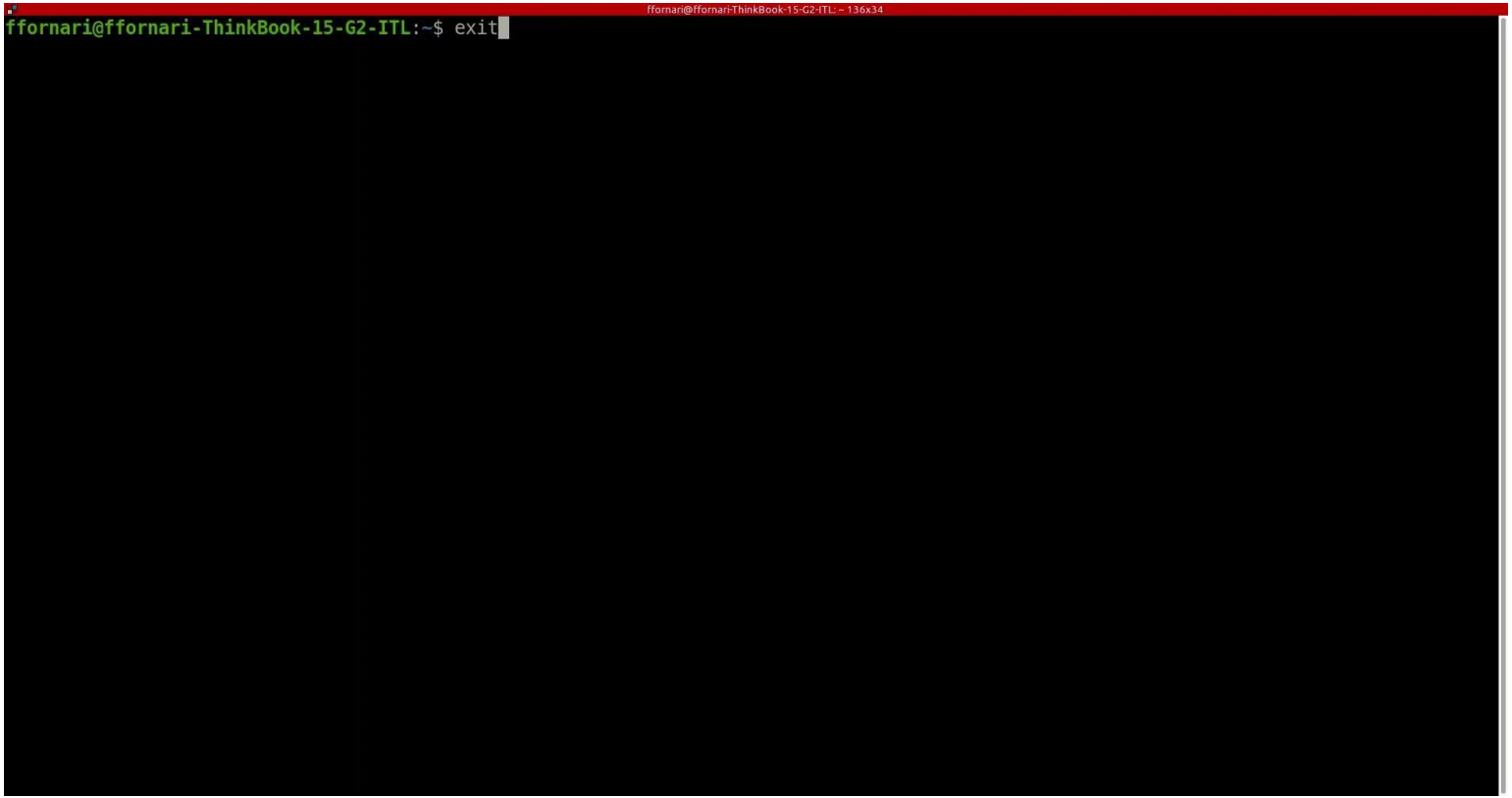




Pros and Cons

- Pros
  - High performance in cache and high capacity in back-end
  - POSIX access mode (quasi)
  - Unified view of different COS and a local filesystem via “POSIX” interface
  - Authorization/authentication machinery hidden from end user
- Cons
  - Access bandwidth to BE limited by GW nodes
  - Writes to COS (gpfs cache -> object stor) is done by one RGW node only while reads (prefetch) are going in parallel over all RGW

Demo


ffornari@ffornari-ThinkBook-15-G2-ITL: ~ 136x34
ffornari@ffornari-ThinkBook-15-G2-ITL:~$ exit