

A person is standing in the center of a long, brightly lit server aisle. The aisle is flanked by tall server racks on both sides, which are filled with various components and glowing lights. The floor is made of metal grates, and the ceiling is also visible, showing the complex infrastructure of the data center. The overall atmosphere is one of a modern, high-tech environment.

Storage Scale AFM Deep Dive – San Jose UG 2025

Karrthik K G (kgkarrthik@in.ibm.com)

AFM Development

Disclaimer

- IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.
- IBM reserves the right to change product specifications and offerings at any time without notice. This publication could include technical inaccuracies or typographical errors. References herein to IBM products and services do not imply that IBM intends to make them available in all countries.

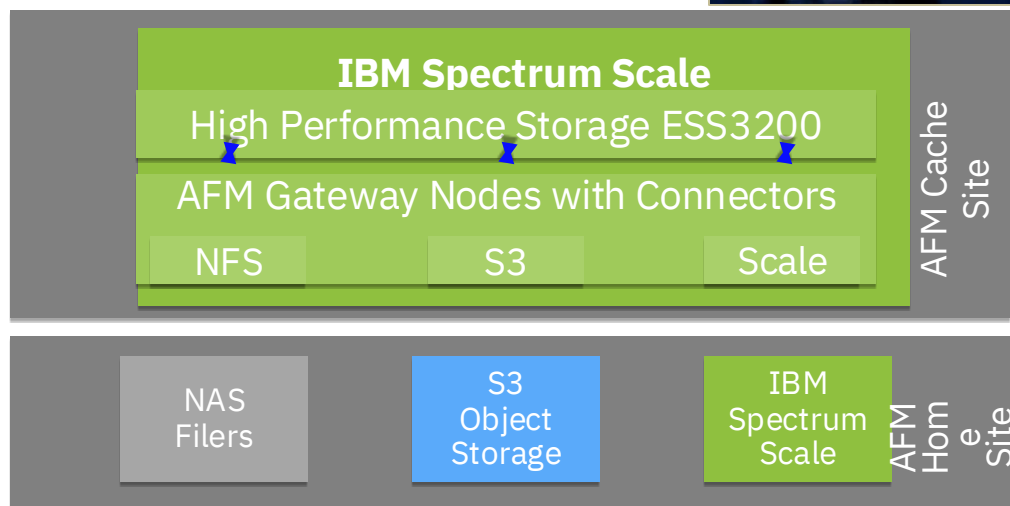


AFM Deep Dive: Agenda

- AFM overview
- AFM Modes
- AFM Migration/Tiering/Archive/AI
- Success Story

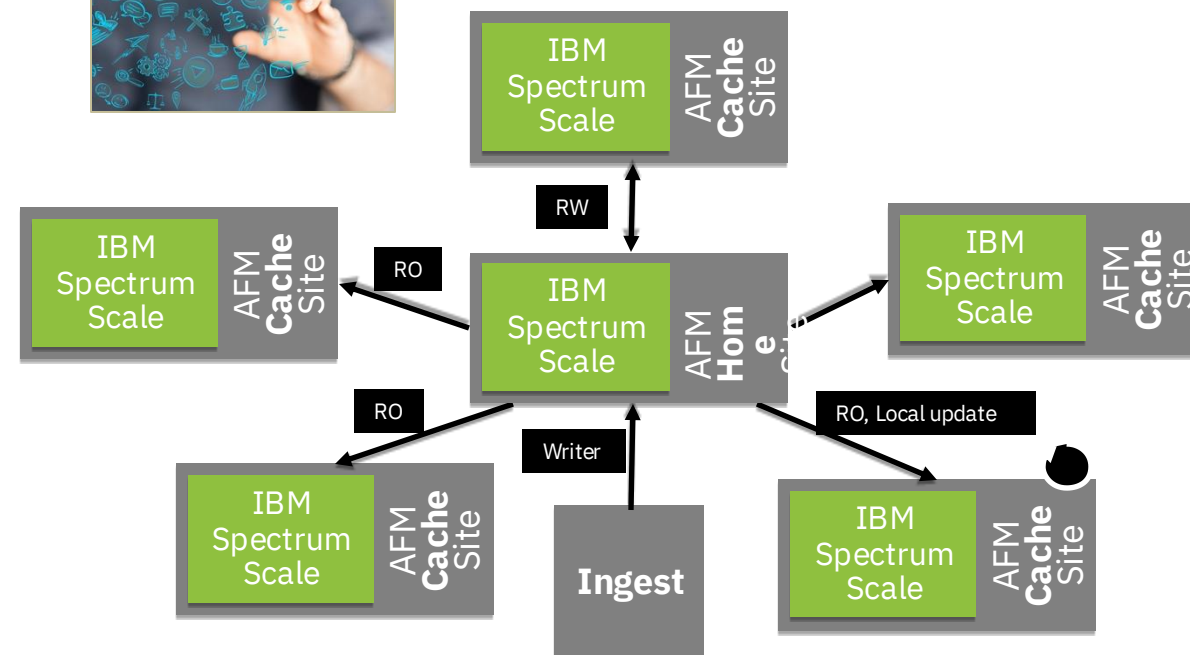
Data Caching Services (AFM) Use case details

Data Virtualization



- Vertical caching
- Common namespace across isolated data silos in legacy 3rd party data stores
- Transparent access to all data regardless of silos
- Scale-out Posix performance
- Data export via NFS, SMB, HDFS, Object
- Can be used to seamlessly migrate data to new storage

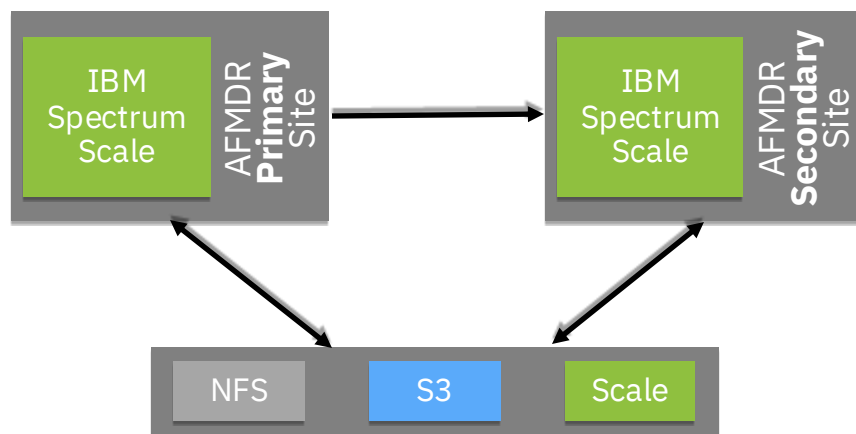
Data Collaboration



- Consistent cache provides a single source of truth with no stale data copies
- Horizontal caching
- Bi-direction traffic from Edge to Center
- Eventually Consistent data cache
- Transparent on-demand data access and transfer
- Policy driven data prefetch and eviction

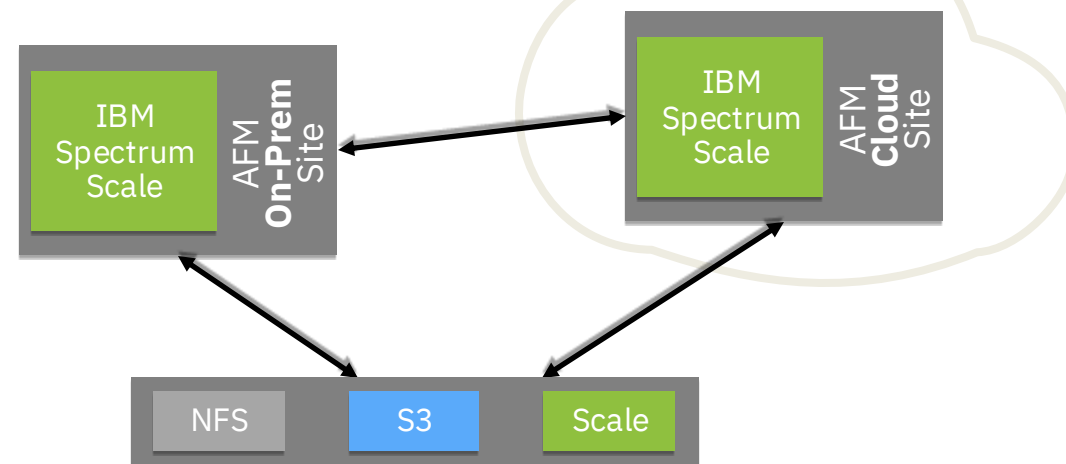
Data Caching Services (AFM) Use case details

Data Resilience



- Active-Passive DR over WAN or Cloud
- Designed for high latency and asynchronous DR
- Hot standby failover to DR site
- Automatic fallback data reconciliation
- Read-only access / analytics to all data at passive site

Hybrid cloud / Bursting

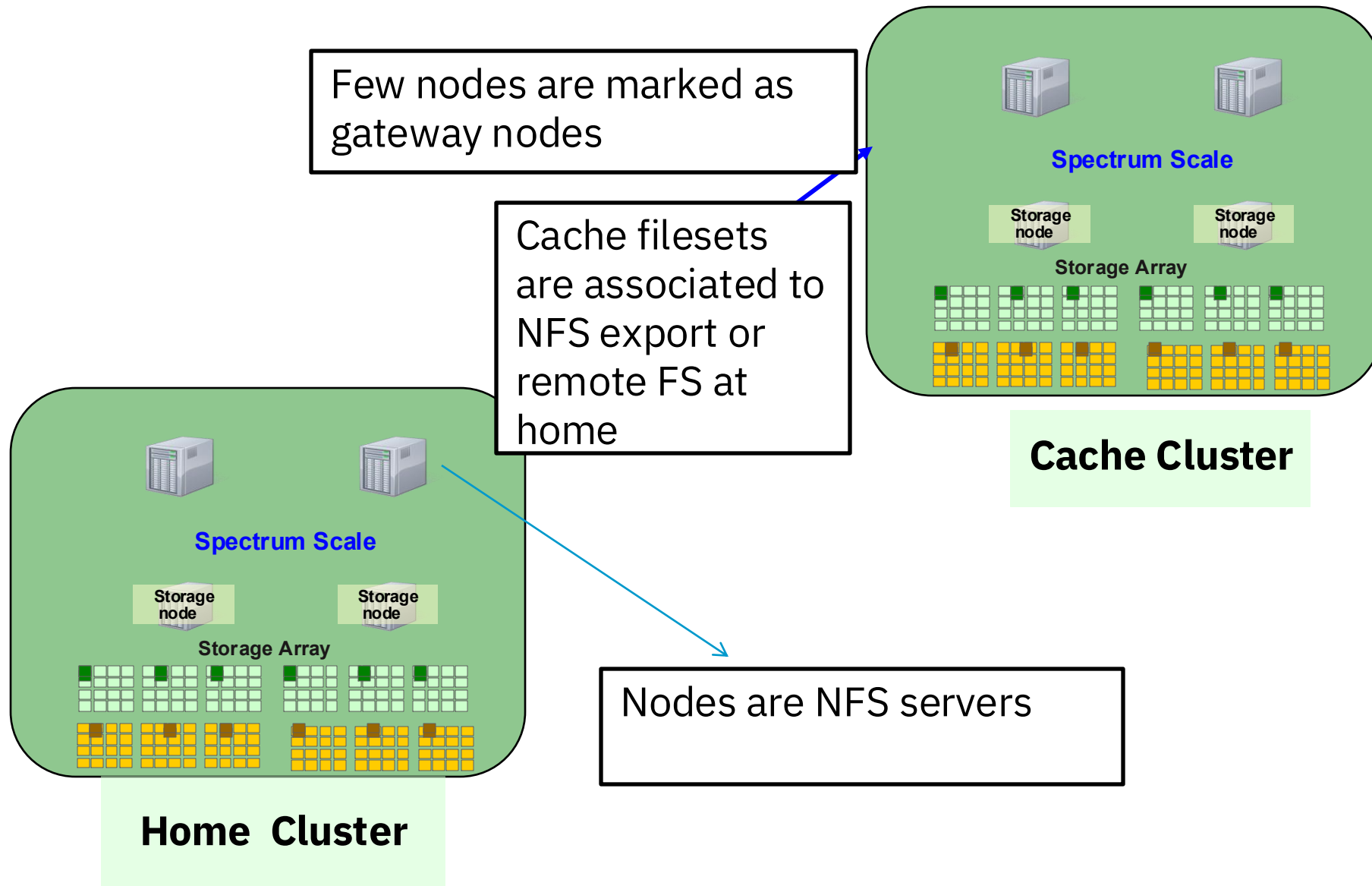


- Rapidly expand compute resources to cloud or data centers
- Common file system creates a single namespace across all locations
- Transparent access to data
- Cost effective way to increase compute on existing data
- Analytic results automatically pushed to home site

AFM Concepts

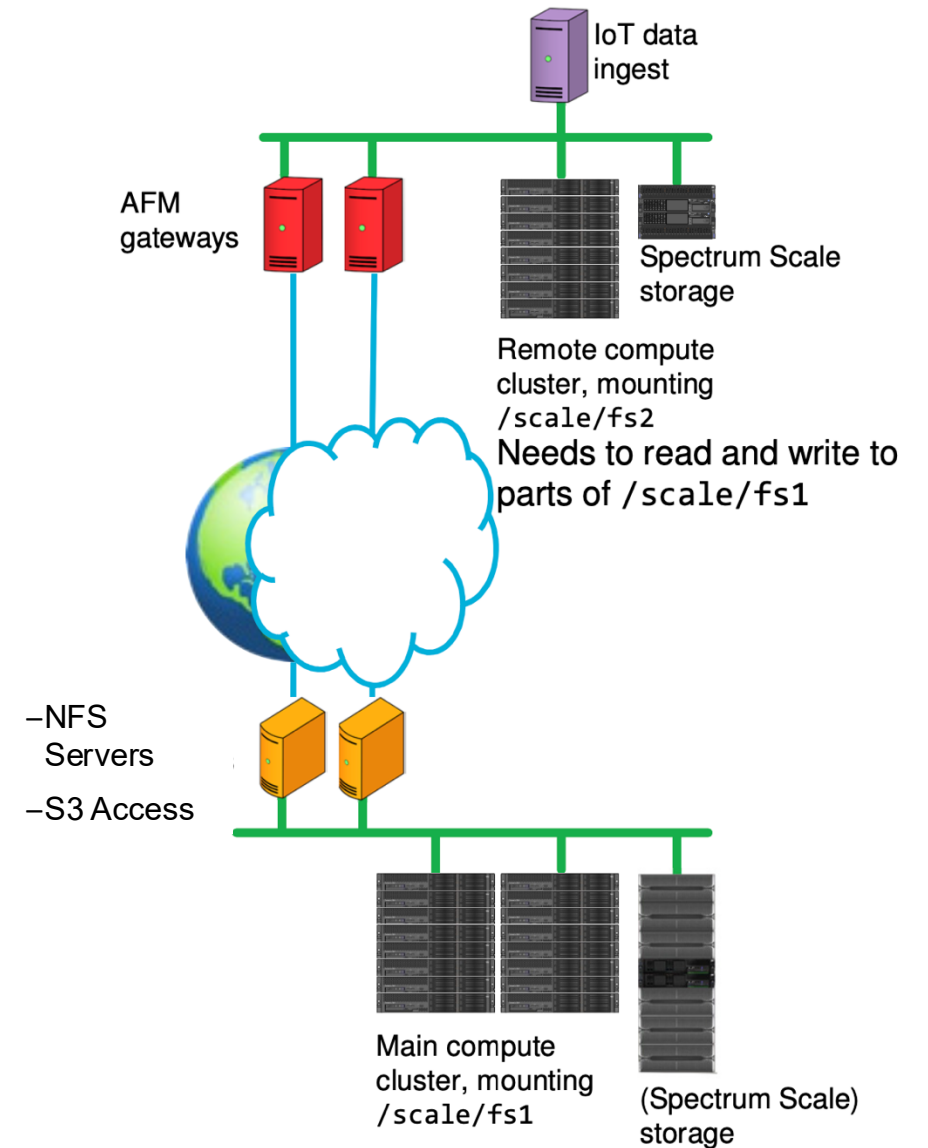
- Node types
 - **Application node**
 - Writes/reads data based on application request to the GPFS filesystem at cache cluster
 - Can be Linux/AIX/Windows – NFS/SMB mounts
 - **Gateway node(s)** is the node that connects to the home cluster
 - Queue of pending operations is in memory at the gw node
 - Selected based on hash of filesetid or user-defined.
 - Reads/writes data from the home cluster to the cache cluster
 - Checks connectivity with the home cluster and changes to disconnected mode on connection outage
 - Triggers recovery on failure
 - Only Linux supported
- Sites
 - Home cluster (Object Store)
 - Exports a fileset that can be cached
 - Cache cluster
 - Runs AFM and “connects” a local fileset with the home fileset.
- Transport Protocol
 - NFS, NSD, S3 and Azure Blob, Glacier/Deep Archive

AFM WAN Caching



Data Caching Services: Active File management (AFM)

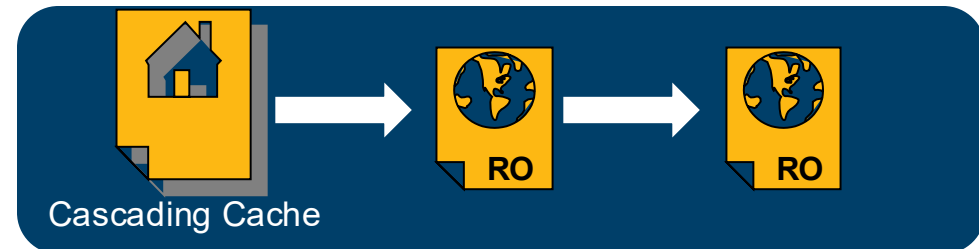
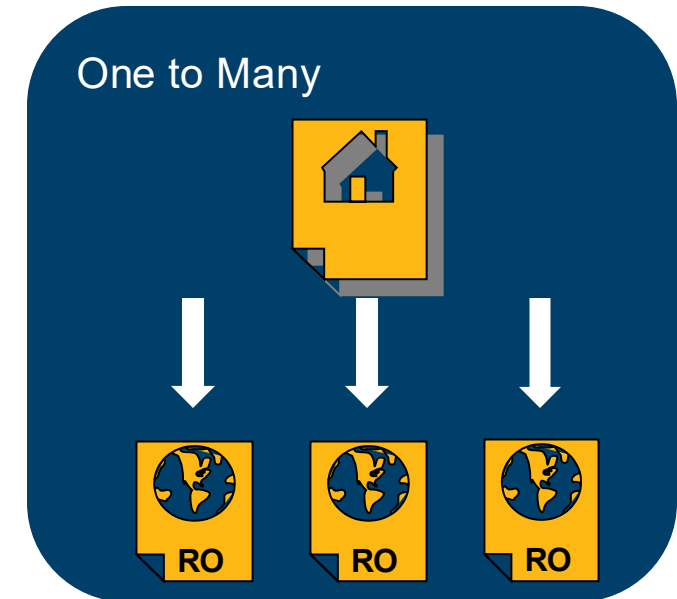
- Supported Storage Backends
 - Spectrum Scale/ ESS
 - IBM COS
 - 3rd Party NFS Storages
 - S3 compatible Object Storages
 - Azure Blob Storage
- Protocols to communicate with Storage backends
 - File: NFS v3, NFS v4.1/4.2 (KNFS), NFS 4.1 (Ganesha)
 - File: NSD
 - Object: S3 and Azure – HTTP, HTTPS
- Type of Caches
 - Read write (commonly used / default)
 - Read-only, Local Update, Manual update (Object only)
- Security considerations
 - Setup NFS to use sys, krb5, krb5i, or krb5p
 - Setup S3 Object access with https



AFM Modes

AFM mode: Read-only caching

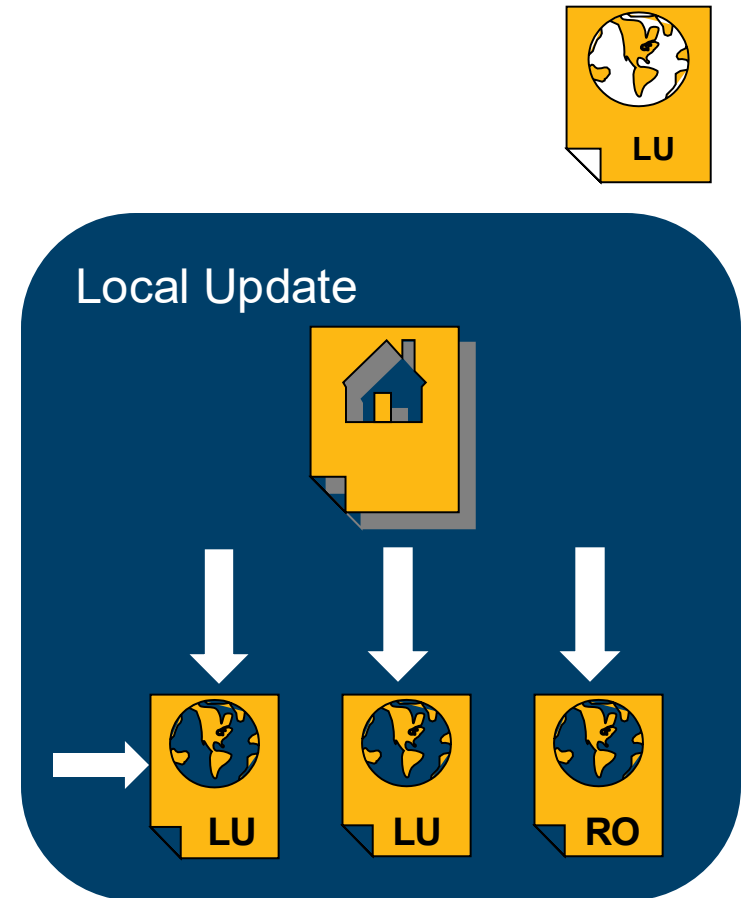
- Read caching mode
 - Data exists on the home fileset and one or more cache sites
- Data is moved to the cache on-demand.
 - File Metadata caching: Listing the contents of a directory moves the file metadata information into the cache
 - Data – Opening a file copies the data in the cache
 - Getting data to the cache
 - On-demand when opened
 - Pre-fetch using a GPFS policy
 - Pre-fetch using a list of files
- Caching behavior
 - Many to one
 - Optional LRU cleaning of cache
 - Cascading caches



AFM mode: Local-Update

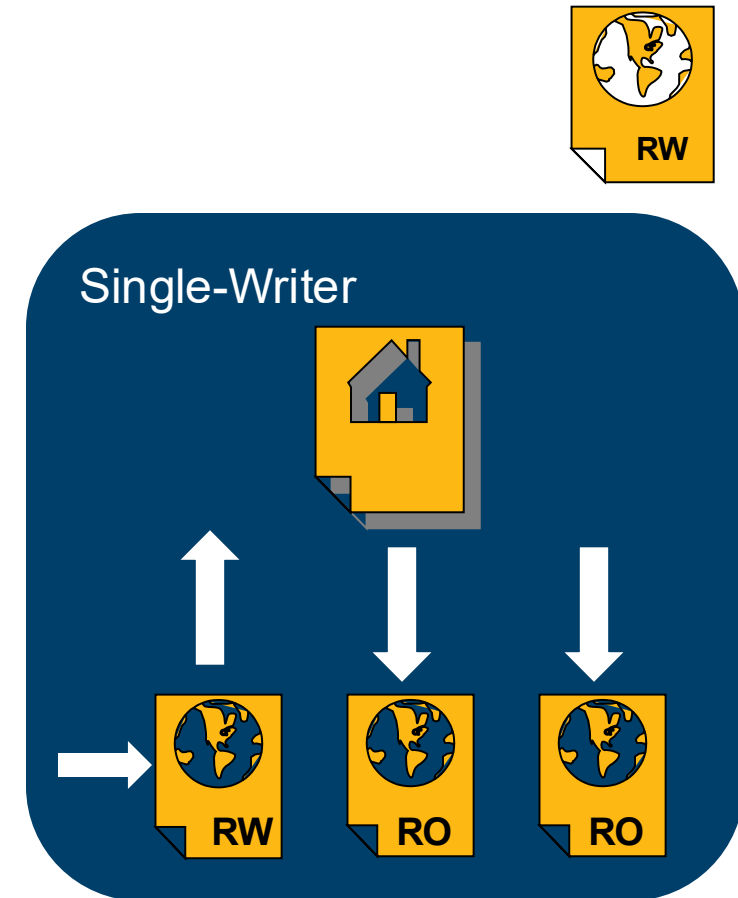
Similar to Read caching mode

- Data exists on the home fileset and one or more cache sites
- Cached file set data can be modified but are considered local updates
- Local updates are never pushed to home
- Once a file is modified in the cache, file is no longer compared to home file
- Appending, truncating or writing to an uncached file causes the file to be fetched before making the change locally
- NOTE: any small change done in the LU fileset directory could cause the fileset to be marked local and lose context with the home.
- For example, running chown or chmod on root directory in LU fileset causes the entire fileset to lose context with home



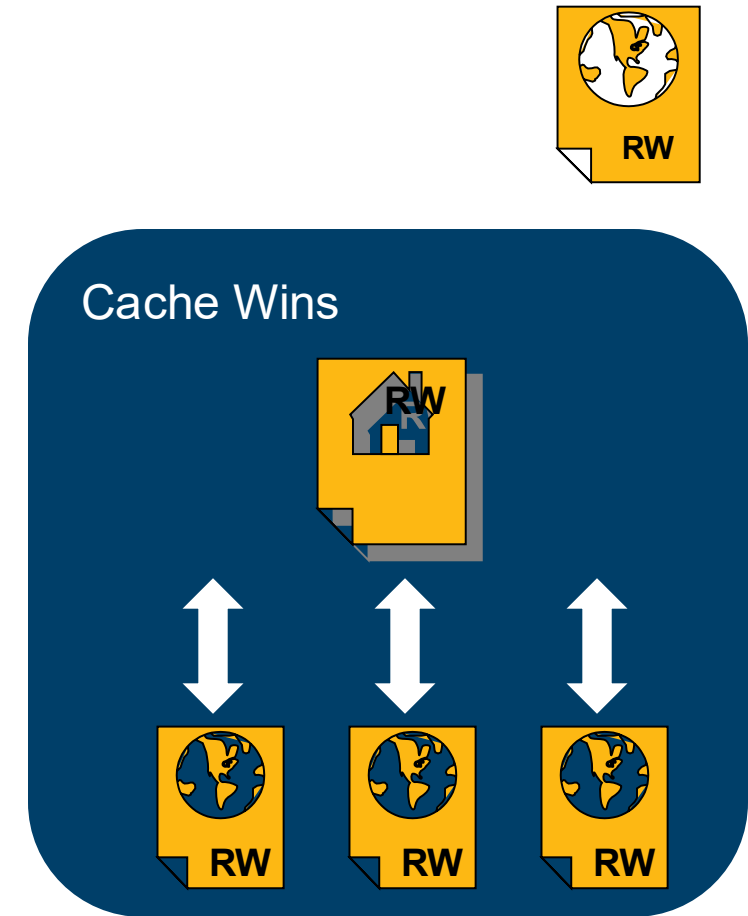
AFM mode: Single-writer

- Data written to a cache
- Asynchronous replication back to home
- Can have multiple read-only caches
- While one cache does all writing, all others are in RO or LU mode (not enforced).
- In SW mode, home is not allowed to write.
- Asynchronous replication back to home
- Appending to or truncating does not fetch file, but queues it back home.
- The usage of peer snapshots via mmpsnap command to provide snapshot of home and cache



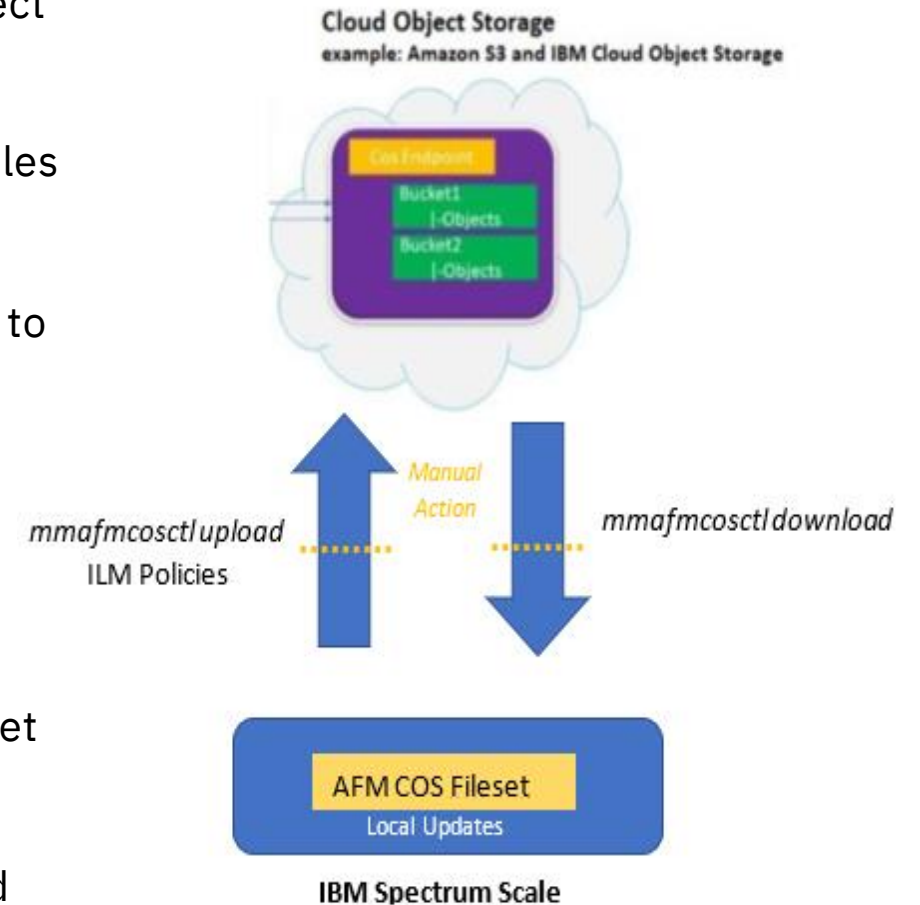
AFM mode: Independent Writer

- Multiple cache nodes
- All nodes can read/write data
- Conflict resolution
- Default: The last writer wins (no locking enforcement)
- Partition data among writers to avoid conflicts
- Appending or truncating a file in IW mode does not fetch the data into cache, but queues it to home
- Factor in time-zones if home/caches are separated



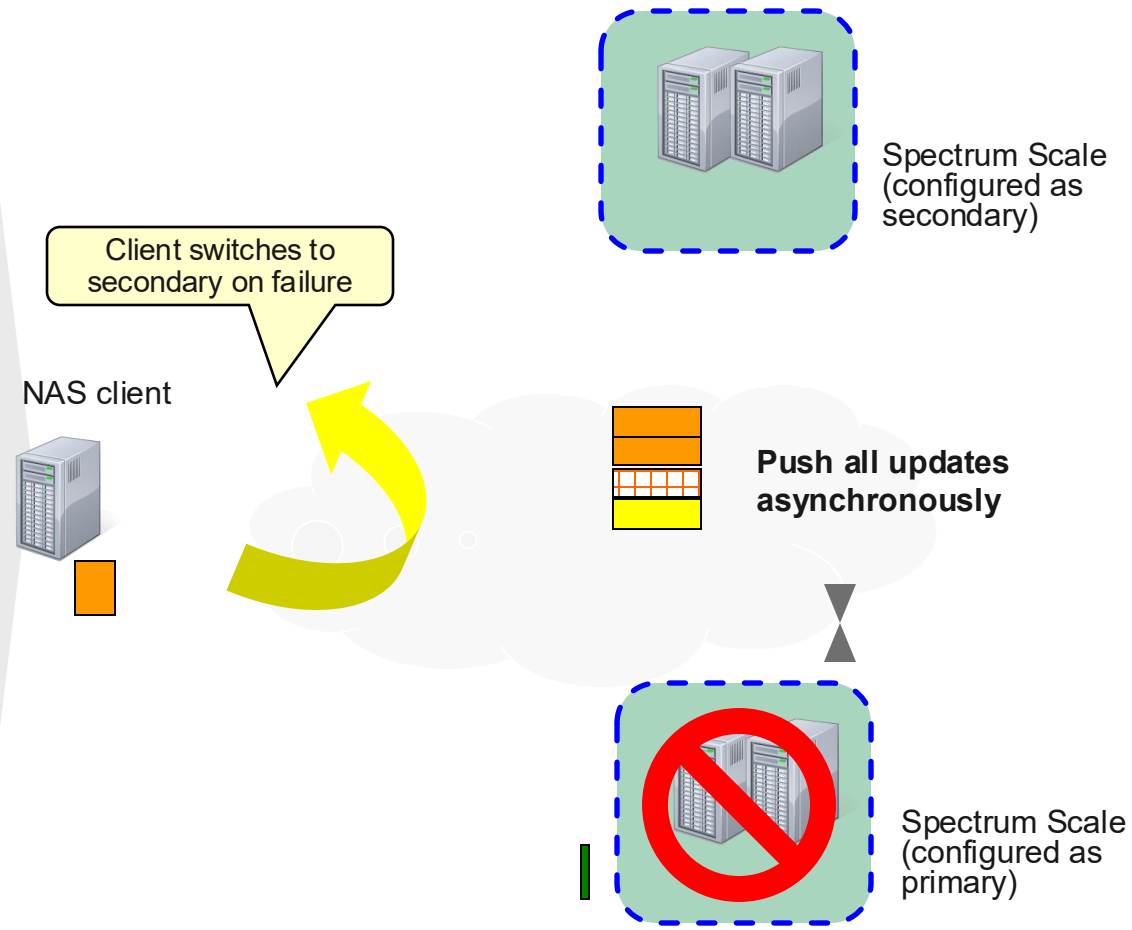
AFM Mode: Manual Update

- Supports manual upload/ download of objects using ILM policies or object list and avoids automatic upload/ download
- All changes are local. Manual action is required to upload or download files to cloud object storage
- Metadata is refreshed only once when the MU fileset is created pointing to non-empty bucket
- Manual control over deletion from cache
- Manual deletion from Cloud Object Storage
- An independent fileset can be converted to a MU mode AFM fileset
- Auto removal of files/ objects from Cache and Object Storage using fileset parameter 'afmMuAutoRemove'
- Specify policy to be used with `mmafmcosctl reconcile -policy` command



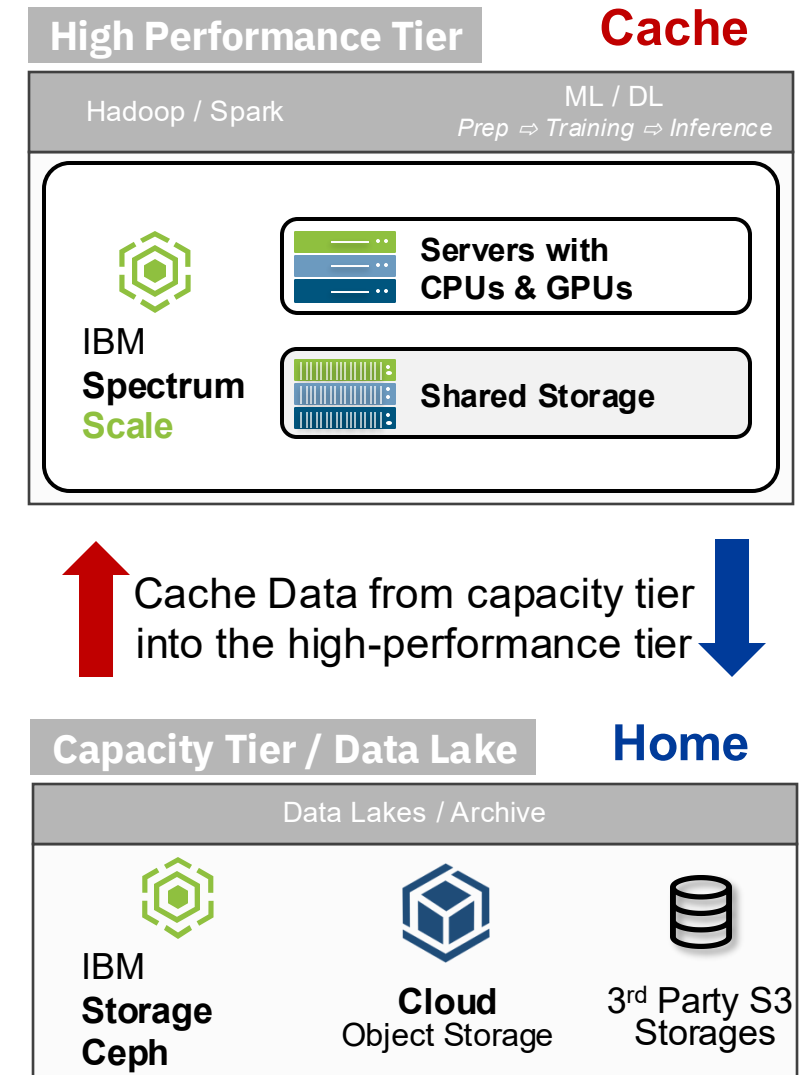
Disaster Recovery with AFM Async Replication

- Establish per-fileset replication relationship using AFM
 - Configure AFM in a primary-secondary relation
- Only deltas are pushed
 - AFM tracks exact filesystem operation
 - Only updated byte range will be pushed
- Multi-site Snapshots for consistent copy
 - Snapshots at both sides are in sync.



High Performance File cache for Cloud Object Storage

- Enables Active File management of IBM and 3rd party S3 Object Storages with Storage Scale multi-protocol access
- Supports multiple S3 Object Storages
 - IBM Cloud Object Storage, IBM Storage Ceph
 - Amazon S3, Google Cloud Storage
 - Seagate Lyve Cloud Object Storage
 - Microsoft Azure Blob Storage (Native Support) – in 5.1.9
- Partial or full Object caching
 - Controlled by afmPrefetchThreshold setting
 - Only fetches data blocks that are read by the application
- On demand Object caching
 - Cache content from offset
 - Start Accessing data without retrieving entire object
- Asynchronous updates of cached objects to S3 storage (writeback)
- Highly configurable ([link](#))
 - Cache eviction policy (automatic eviction)
 - Lookup refresh interval

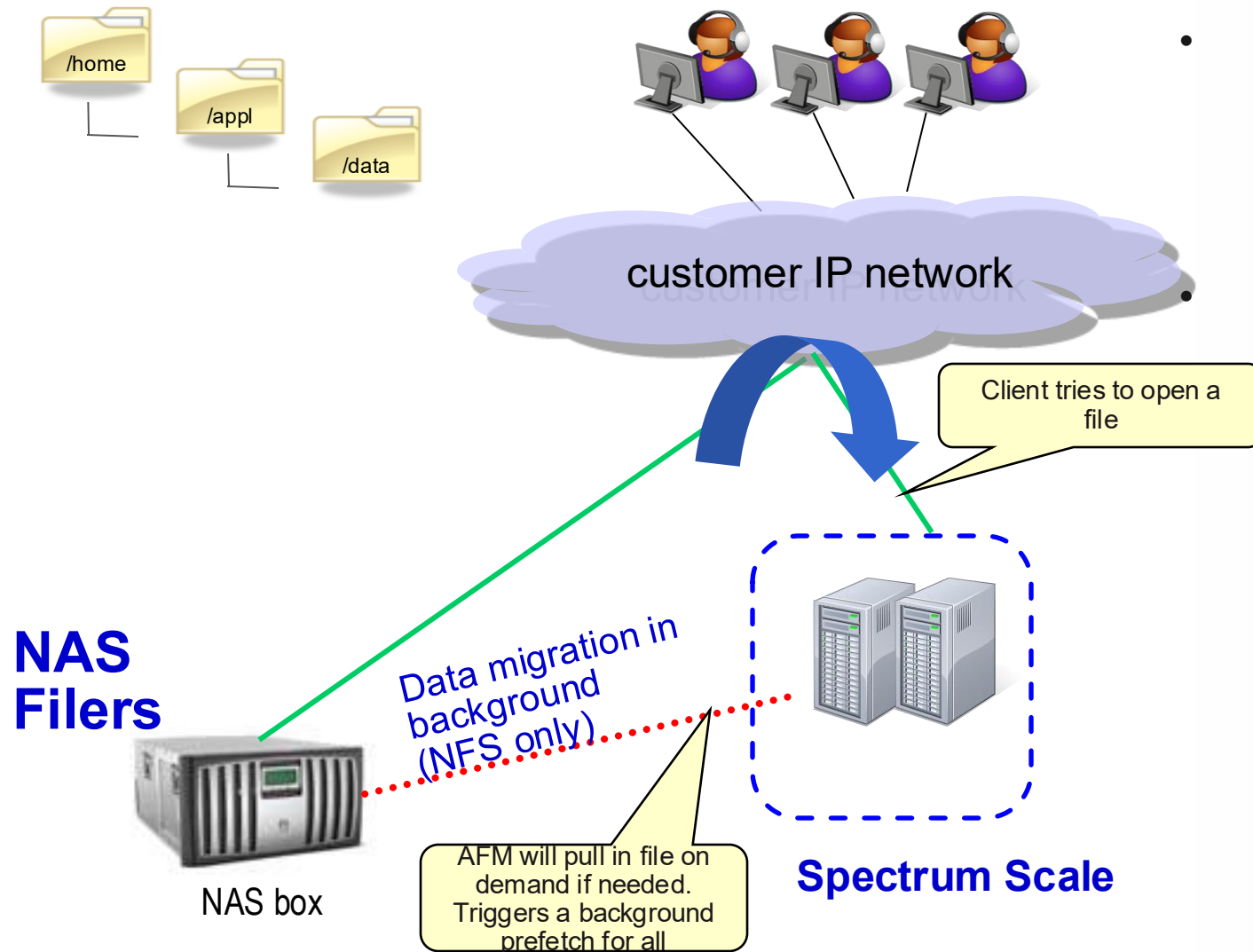


AFM Features

- Metadata download / population
- Manual and On demand prefetch
- Eviction - Manual and automated
- Fileset to Gateway mapping
- Refresh Intervals / Revalidation
- Partial File / Object Caching
- Parallel Data Transfers
- Out of band upload
- Out of band download
- Auto prefetch

AFM: Migration and Archive

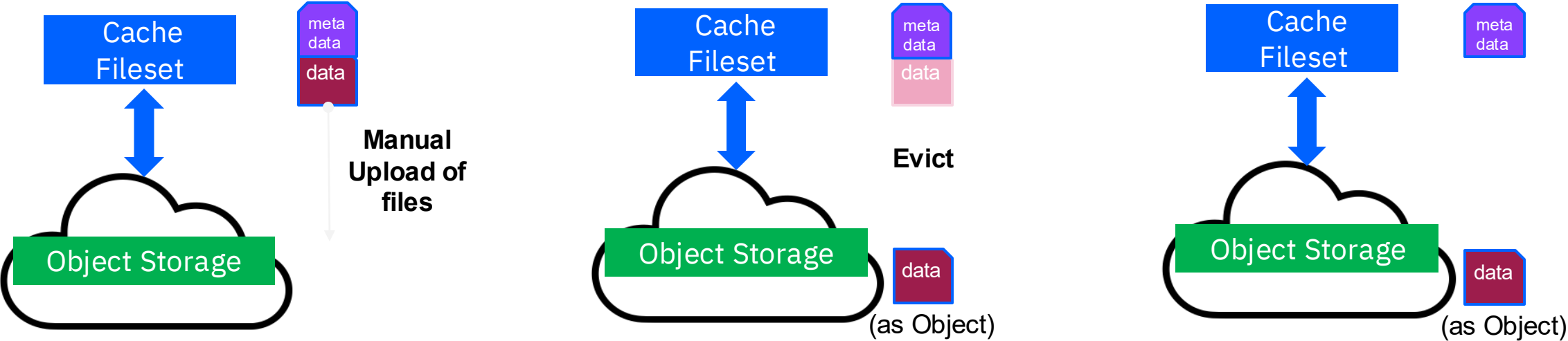
(Legacy) NAS Migration



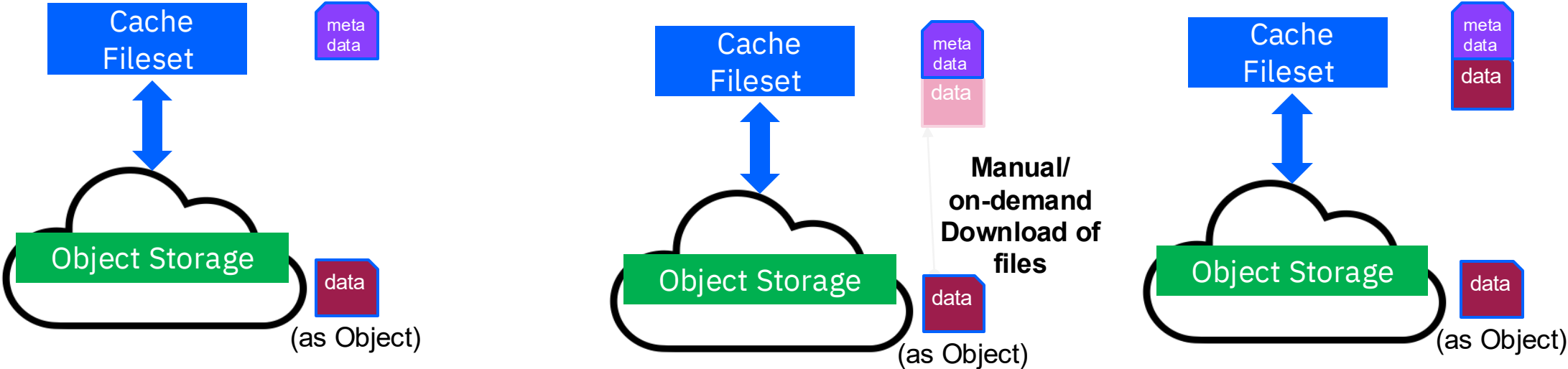
- **Clients switched to new Spectrum Scale box immediately**
- **Data movement in background**

- AFM will fetch data on read miss
- Trigger background prefetch
- All updates done to AFM will commit locally

AFM for archive use case with Manual Update (mu) mode



AFM for caching use case with Manual Update (mu) mode

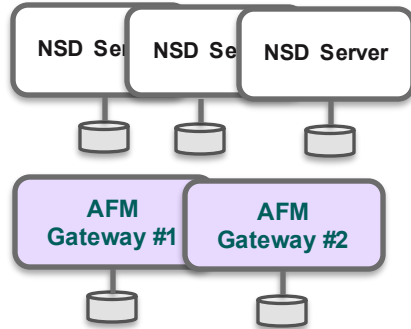


Multisite Replication with MU Mode

Virtual Private Cloud

Private Subnet

Storage Scale
AFM Cache
Cluster



Filesystem

Fileset1

Map1

Push/Pull

Push

Push

Fileset 1 to AWS S3

Fileset 1 to IBM COS

Fileset 1 to MINIO

Map1 → AWS/GW1,
ICOS/GW2,
MINIO/GW1

AWS S3
Storage

multitargetdemo

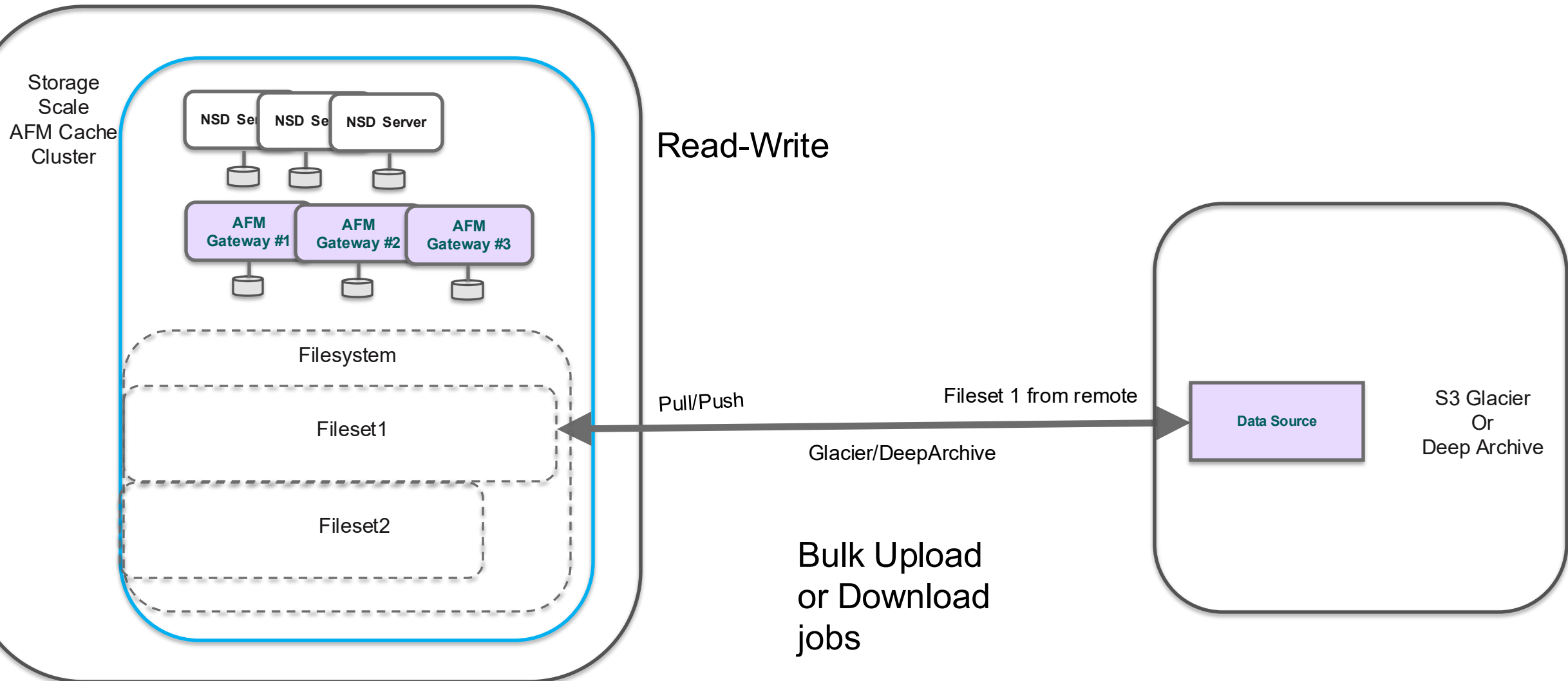
IBM COS

multitargetdemo

Minio

multitargetdemo

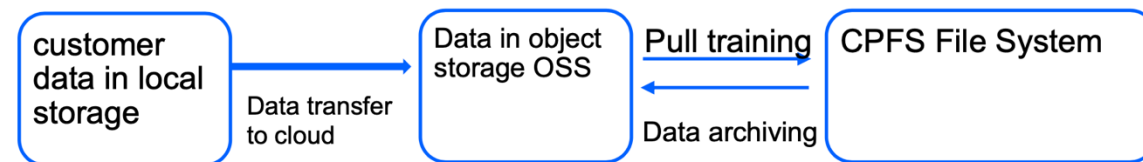
AFM $\leftrightarrow$ S3 Glacier/Deep Archive



AFM : Success Stories

Acceleration for S3 data on Cloud

===

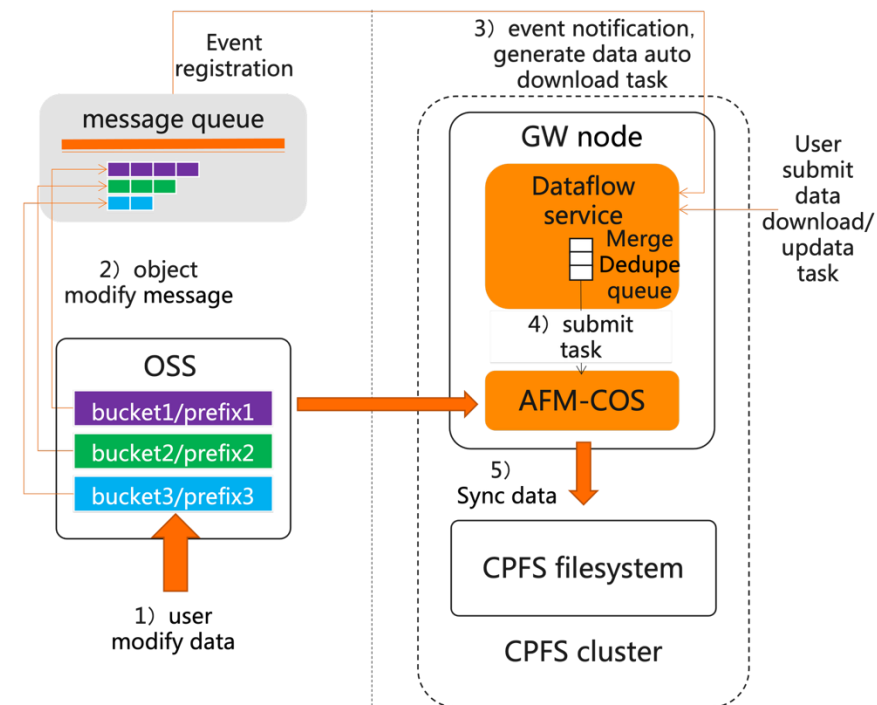


Solution for Data Analytics on Cloud with AFM + S3

- Data is ingested into Cloud S3 bucket by customer, end users
- AFM Caching / Acceleration is setup with S3 as backend storage
- Data is processed via Posix on CPFS (GPFS) filesystem

Use cases – ADAS, Media & Entertainment, Financials, HPC

Other customers considering similar solution: Other cloud providers in China, Financial customer on Azure (US), Top 10 Bank in USA on IBM Cloud



Solution offered by leading Cloud Provider in China

Active File management: Research example

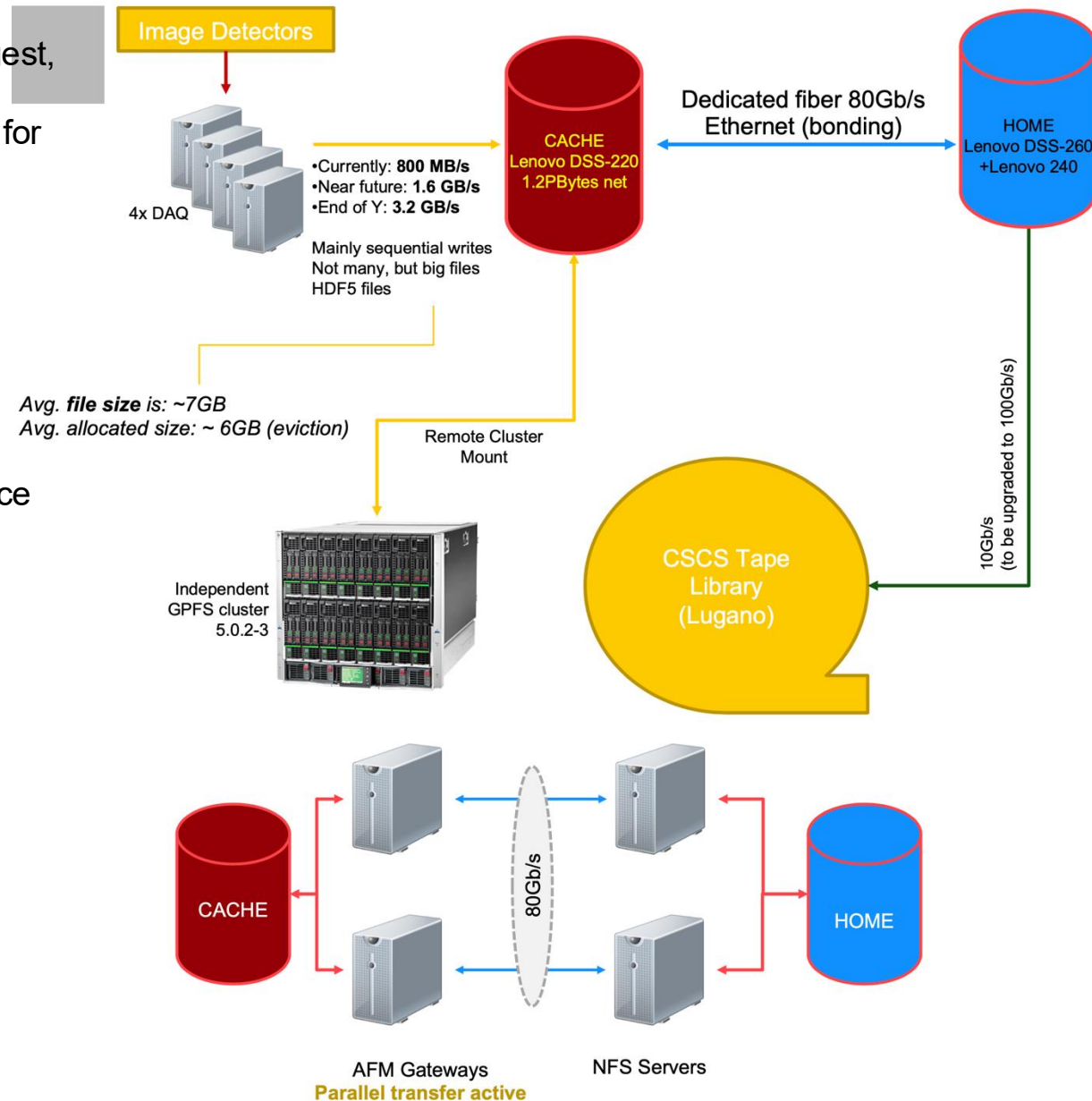
Research Institute in Switzerland

- Online Storage for large images processed by compute clusters for ingest, initial analysis & research
- Images once processed need to be stored offsite as long term storage for further analysis
- Needs data to be transferred promptly or real time from Online to Long term storage
- Produces ~1 PBytes of data (images) per year

Solution

- Online storage configured as AFM Cache; Long term storage as AFM Home
- Cache has eviction-enabled, to virtually extend its real fast-access space
- Eviction is automatic, based on filesets-level quota
- Spectrum Scale + GNR, GPFS 5.0.2.3, 2x AFM gateways
- NFS export

More: [Spectrum Scale user group presentation](#)



Research + HPC environment: University of Queensland, Australia

Overview

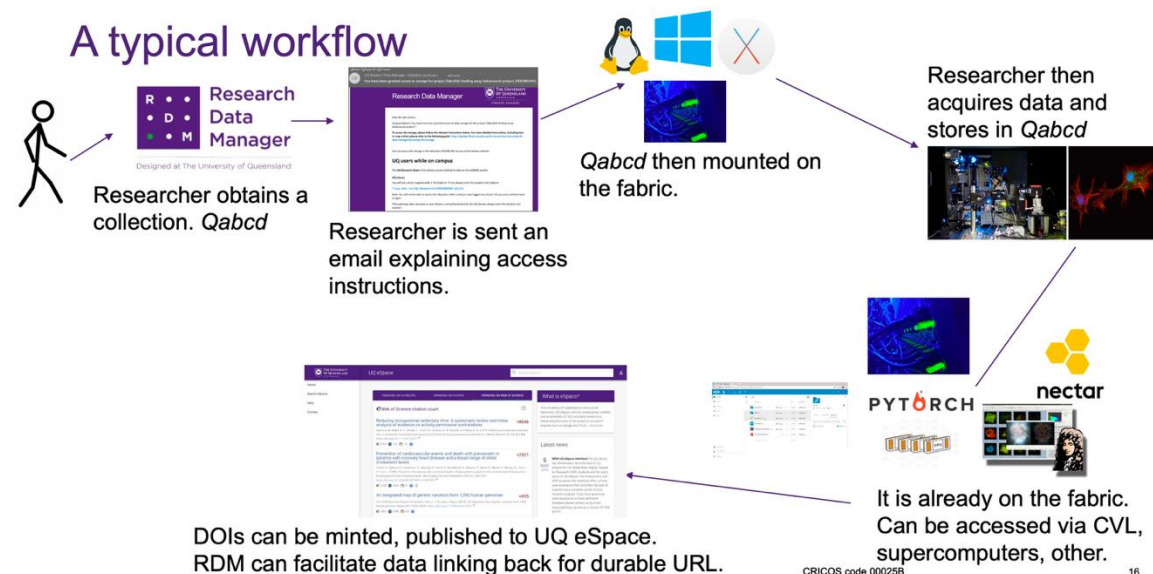
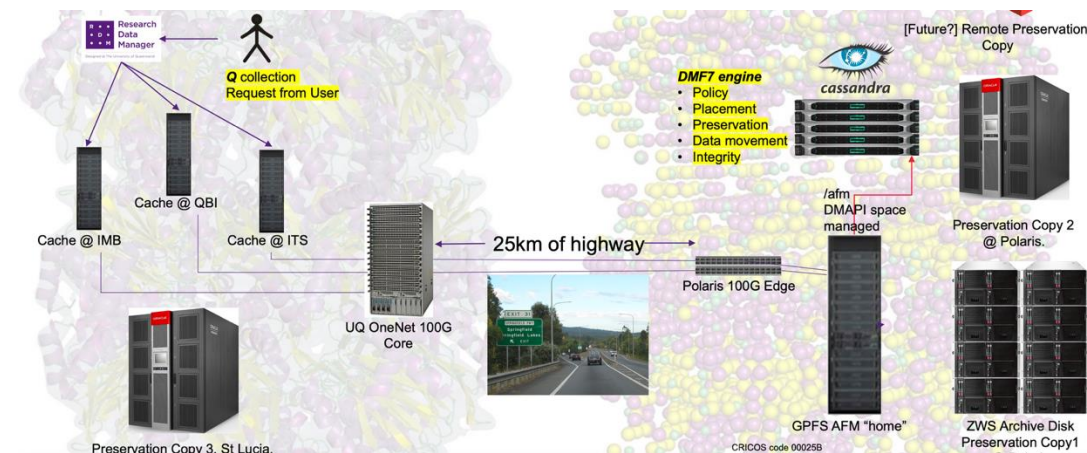
- Vocal reference customer for Spectrum Scale + AFM. Have delivered multiple user group presentations. Reference [Story](#).
- Signed joint agreement with IBM for Spectrum Scale/ESS in 2020

Use case – Single Data Fabric for research data and DR

- Medical, scientific instruments collect and ingest data via SMB, NFS at AFM cache sites
- Data is shared with home at Supercomputing Center
- Data needs to be accessed by Supercomputers, research stations, other departments spread across multiple campus locations for analysis and sharing results
- Certain data needs to be stored at DR location

Configuration Details

- Unique configuration that involves Data sharing + DR configuration (cascade)
- One of the largest sequential write performance over SMB for data ingest
- Customer uses 1500-2000 independent filesets (over published limits)
- Daily ingest of 100-290 TB data
- 2x ESS GH14S at Home connected to Cache over 100Gbps – 25Kms apart
- Use Scale Tiering to an external data pool at Home



DOIs can be minted, published to UQ eSpace.
RDM can facilitate data linking back for durable URL.

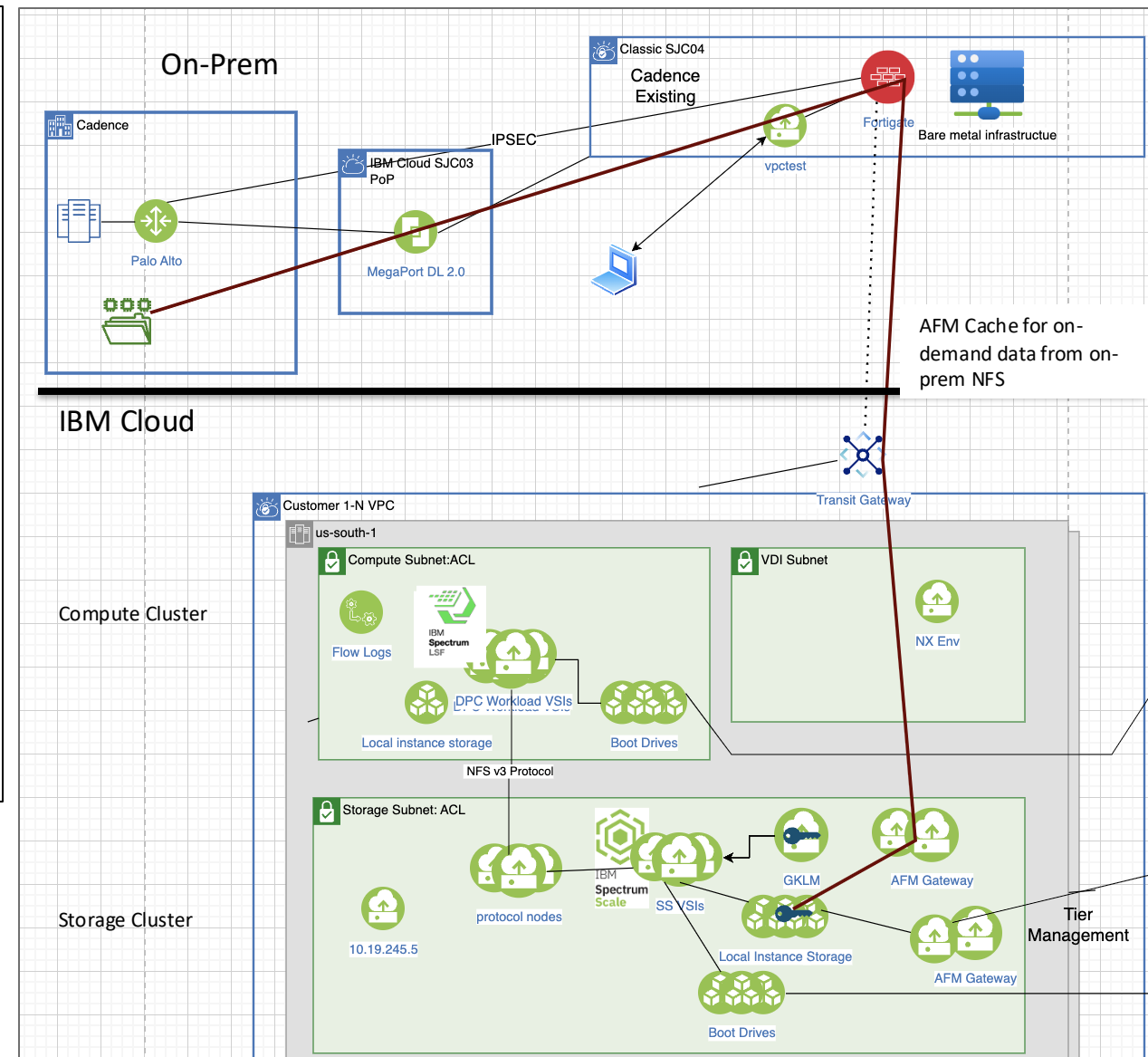
CRICOS code 000258

16

Cache Solution with NFS backend: Leading EDA provider

- Caching between multiple compute locations
 - High performance – throughput upto 10GB/s
 - Metadata and data read/write on multiple file in parallel
 - Compute and IO intensive, non-containerized
- How AFM is used in the environment
 - Data source (Home) is on-prem data centers on Netapp NFS
 - EDA jobs execution on another site (IBM cloud)
 - AFM Cache for on-demand data from Netapp
 - Read only Cache setup on IBM Cloud for jobs to access data
 - Job results are stored in AFM cache in independent writer mode to push results back to on-prem Netapp Filer

Reference: [Storage scale user group presentation](#)



AFM Cache solution: Medical research example

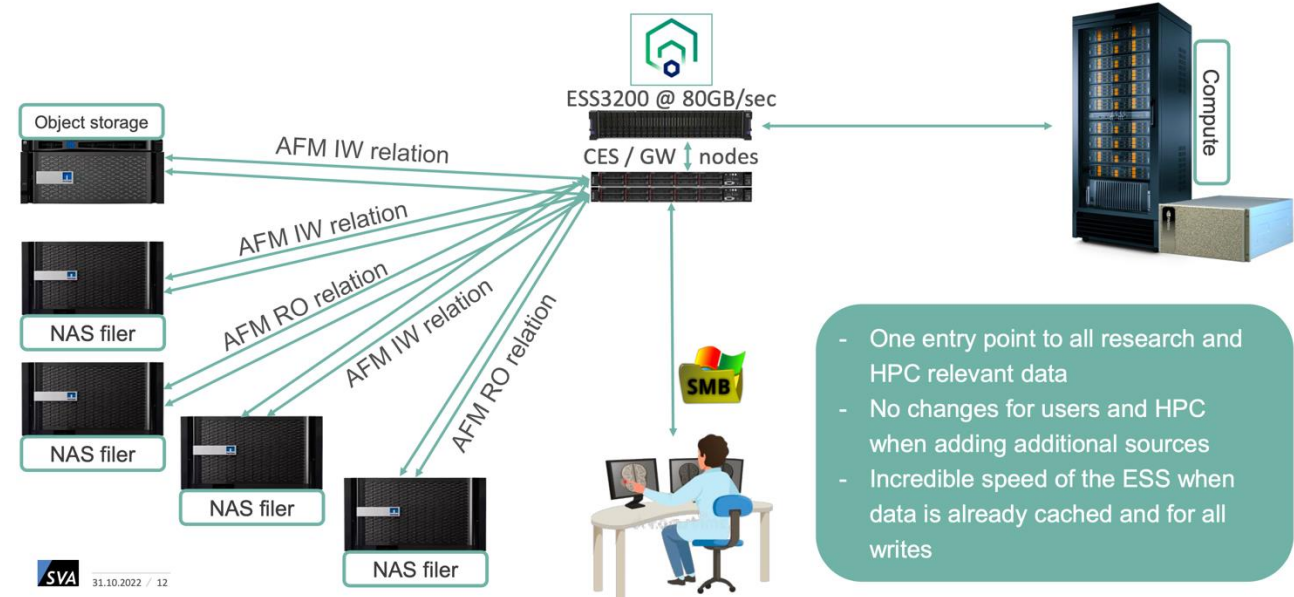
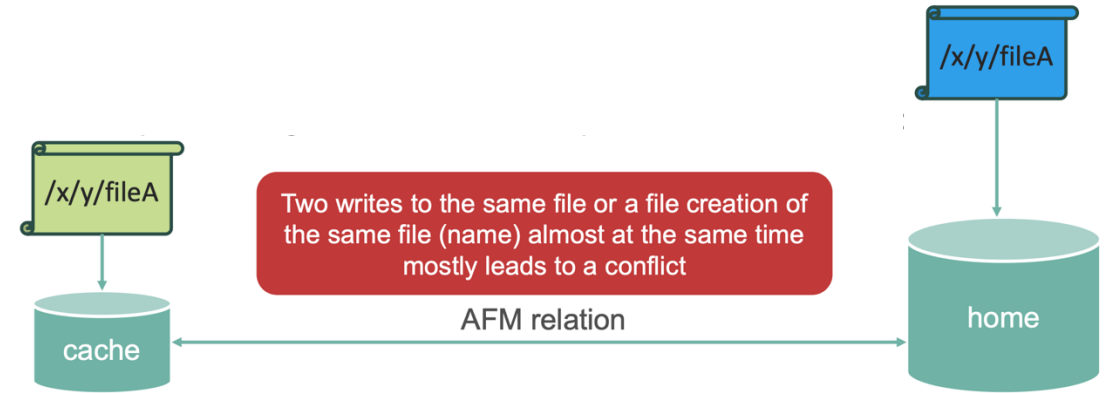
Before AFM

- Storages are distributed across org
- Medical researcher goes to all storages to get access to data – admin has to manage all access
- Legacy storages and hence slower access to data
- Data not available Centrally

After AFM

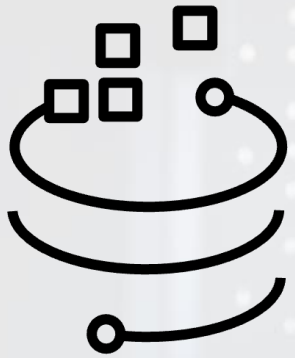
- Data available on fast storage at single, central location
- Investment in legacy storage is still protected making it much faster at additional costs
- ESS Storage capacity 10% of overall storage in the enterprise

Reference: [Scale user group reference](#)



Tiering data from
on-prem storage to
multiple Cloud Object Stores
using
Active File Management (AFM)

Thank you for using



Storage Scale



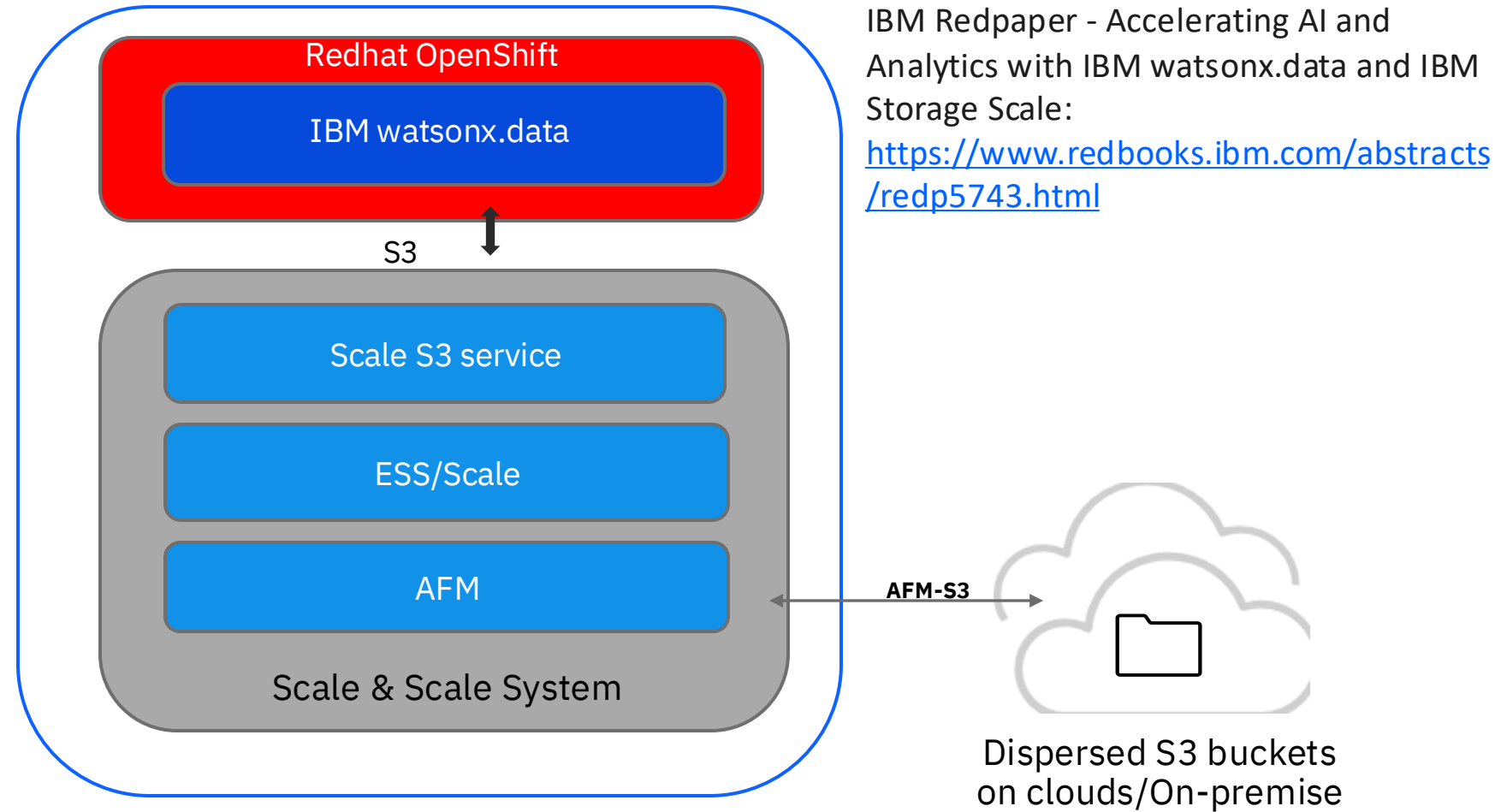
Storage Scale
System

Backup

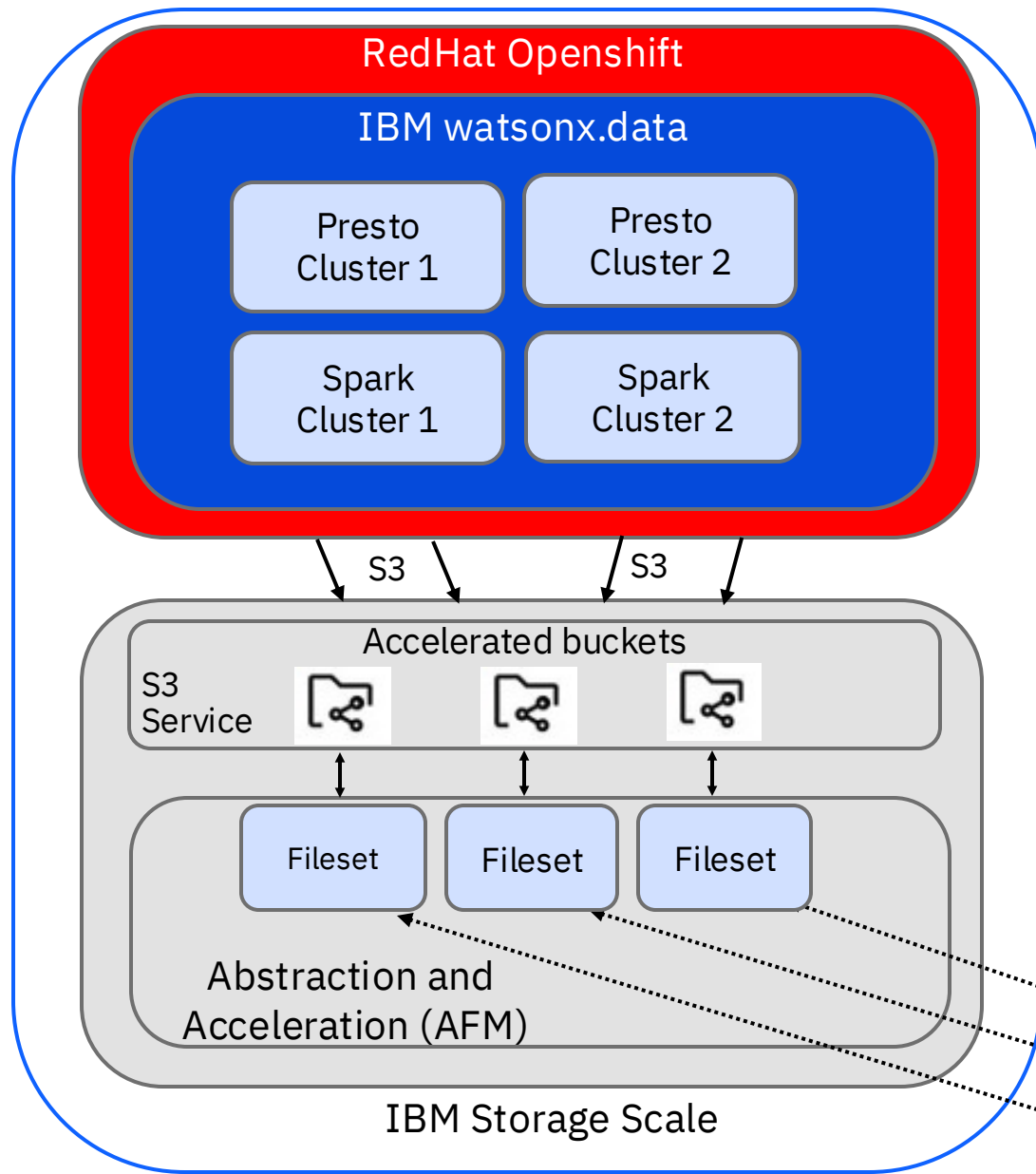
Analytics Use case : IBM watsonx.data + IBM Storage Scale (with AFM & CES S3)

Data buckets could reside locally on the IBM Storage Scale file system, Or could be resident on remote locations e.g. – on various public clouds.

Storage Scale AFM provides a persistent cache layer to deliver high performance access to remote/slower/legacy data buckets



****IBM watsonx.data – Hybrid, open data lakehouse to power AI and analytics with all your data, anywhere.**



Analytics Use case : IBM watsonx.data + IBM Storage Scale (with AFM & CES S3)

AFM can virtualize remote S3 buckets to make them look like local buckets and act like a High Performance Storage Tier for your “Hot Data”, significantly accelerating query performance

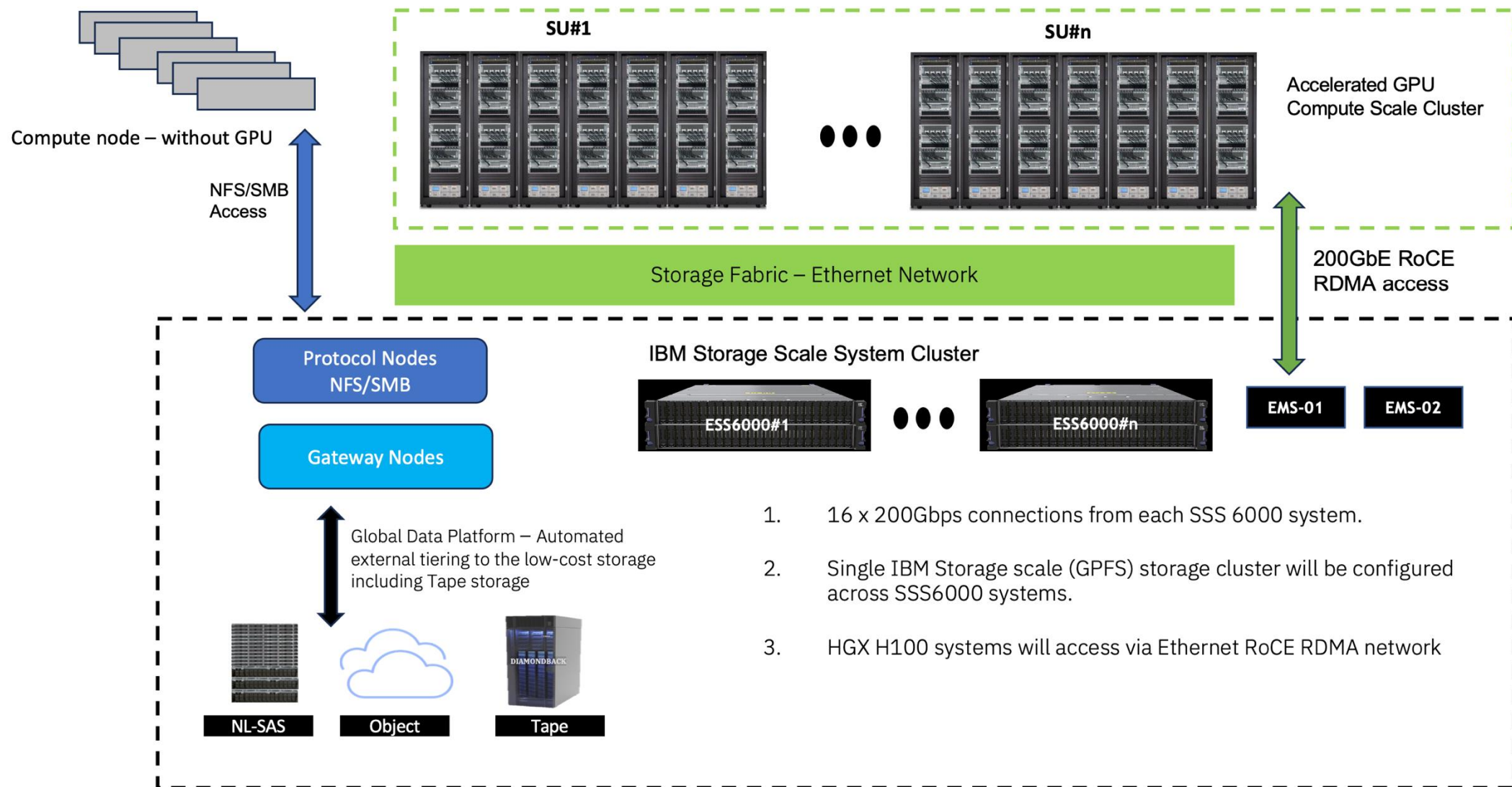
Cache available to all the engines and worker nodes in IBM watsonx.data. Cache survives engine restarts, available even for newly provisioned engines.

Cache is available to multiple protocols and not just to S3.

Huge reduction in public cloud data egress costs

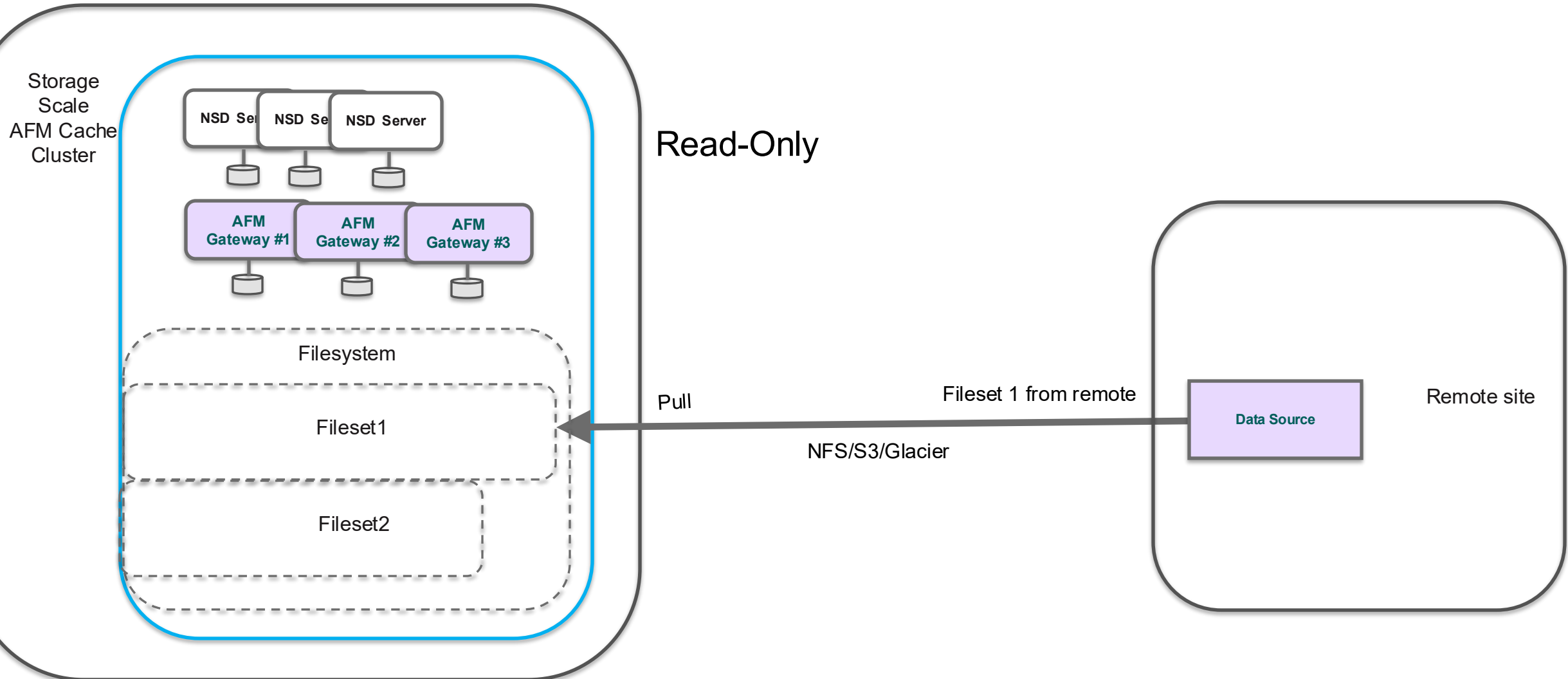
Dispersed S3 buckets
on clouds/On-premise

Storage Architecture for AI Learning with NVIDIA GPUs



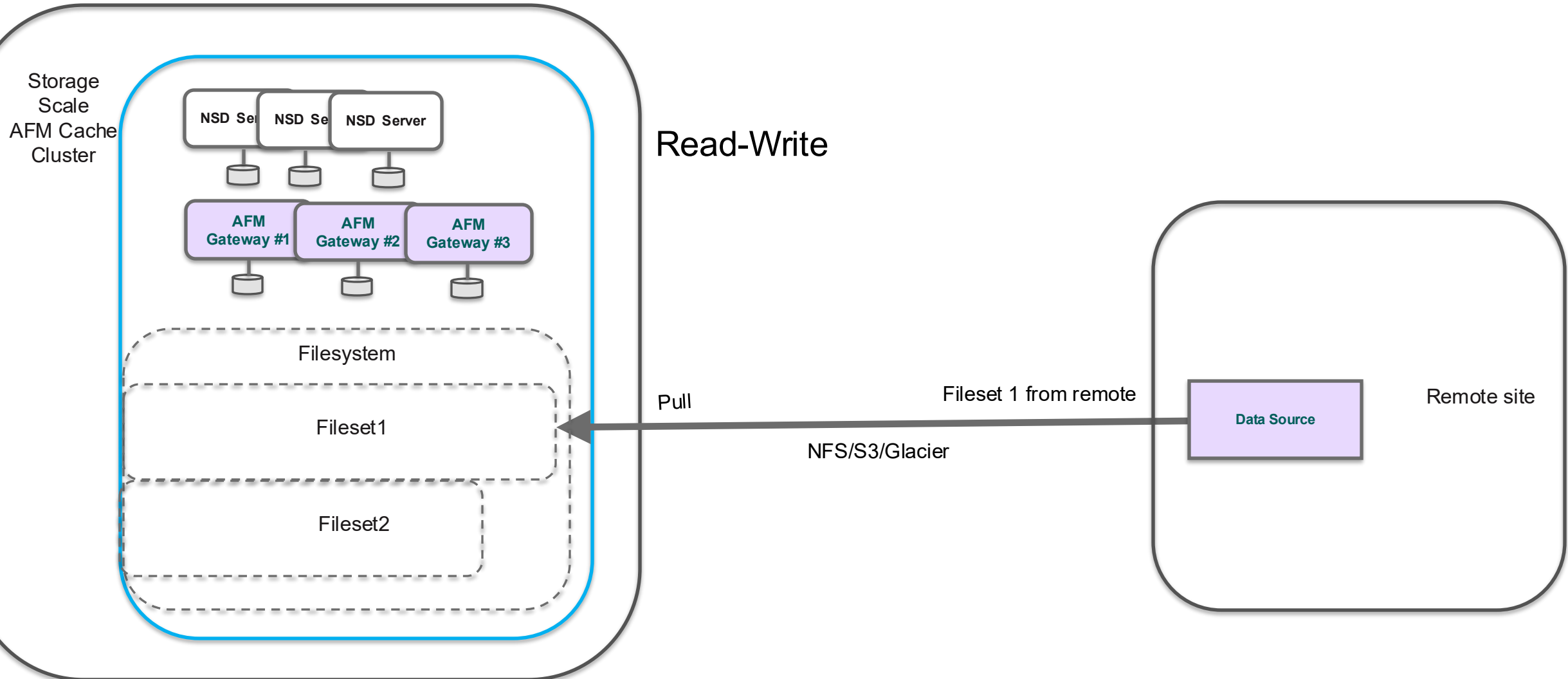
AFM Modes

Read-Only Mode



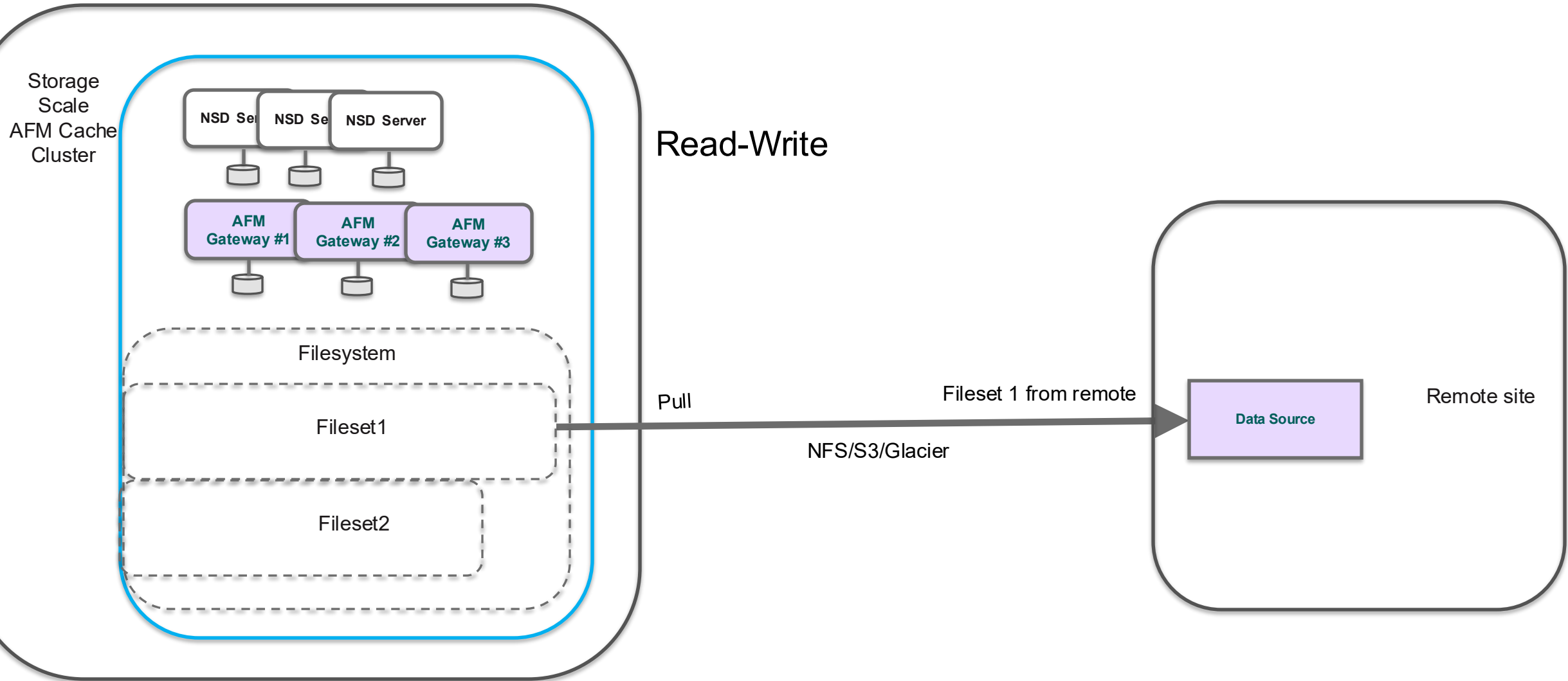
AFM Modes

Local-Updates Mode

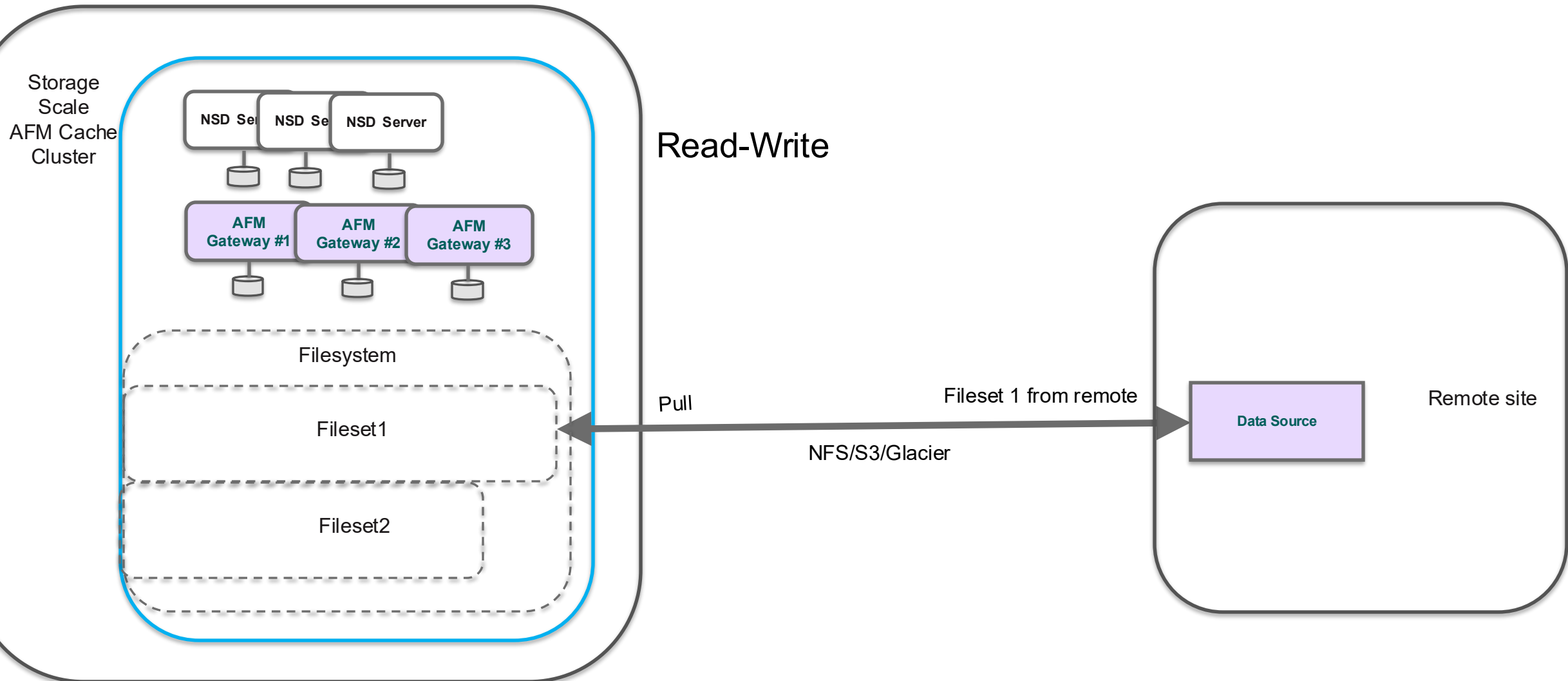


AFM Modes

Single-Writer Mode



AFM Modes Independent-Writer Mode



Storage Scale AFM Deep Dive – San Jose UG 2025



Karrthik K G (kgkarrthik@in.ibm.com)

AFM Development