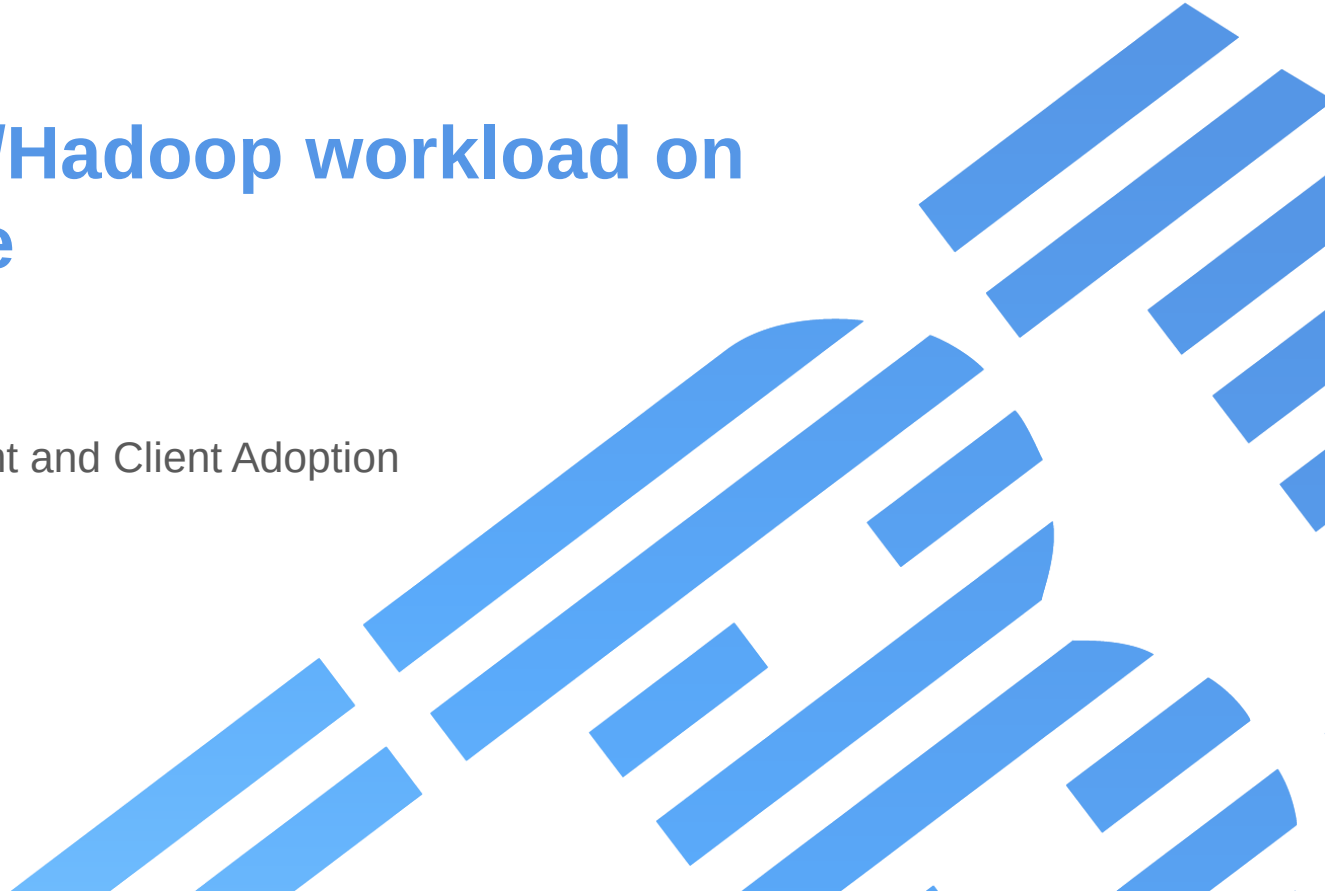
Running Spark/Hadoop workload on Spectrum Scale

Wei Gong

Spectrum Scale Development and Client Adoption



Typical Challenges in Big Data Analytics

Inefficient data movements and many copies

- Different analytics workloads might need different interface
- Siloed data lakes bring inefficient storage space utilization, further issues on cost, management and scaling

Inefficient data lifecycle management

- Different tiers for different data(hot/warm/cold data)
- Leverage different disks from NVMe to Tape according to performance/cost

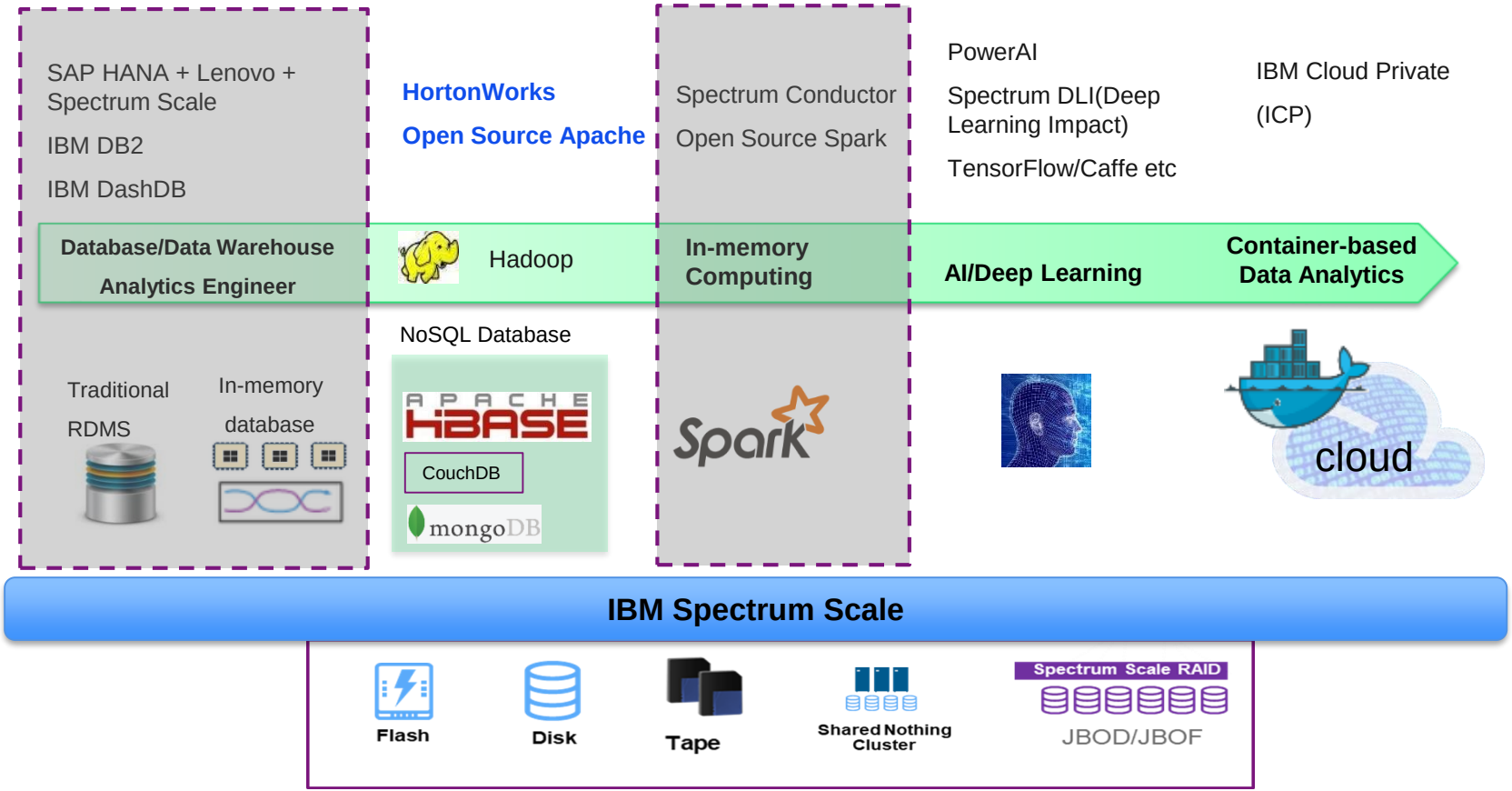
Cross-site HA or DR for business continuity

- Flexible options for different business continuity requirements
- Different options for low/medium/high cost for cross-site HA or DR

Inefficient Data Archive and Share to Cloud Storage

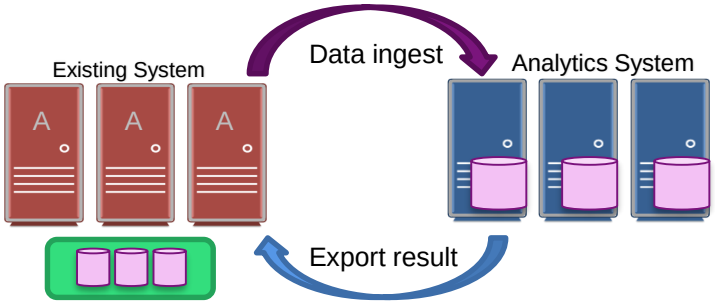
- Efficient data sharing cross different sites
- Efficient data sharing between off-premise and on-premise

Spectrum Scale for Big Data and Analytics

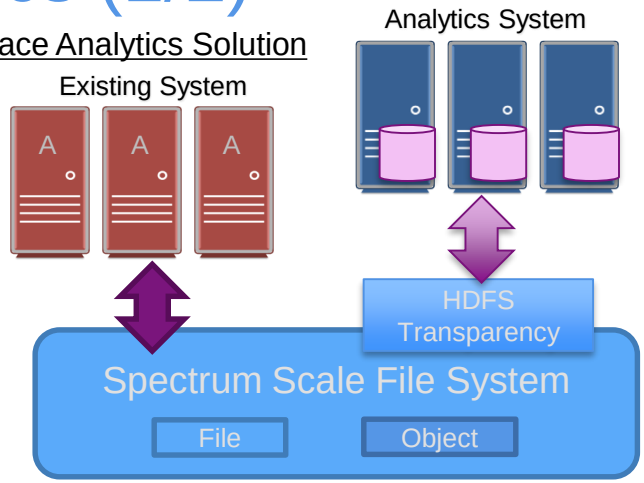


Spectrum Scale In-Place Analytics (1/2)

Traditional Analytics Solution



All-in-place Analytics Solution

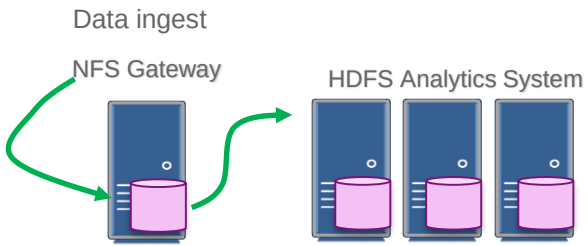


- Build analytics system from scratch, not only for compute but also for storage
- Add storage and compute resource at the same time no matter it's required
- Native HDFS doesn't support native POSIX
- Lacks of enterprise data management and protection capability

- ✓ Can leverage existing Spectrum Scale storage
- ✓ Unified interface for File and Object analytics
- ✓ POSIX compatibility
- ✓ Mature enterprise data management and protect solutions derived from Spectrum Storage family and 3rd part components

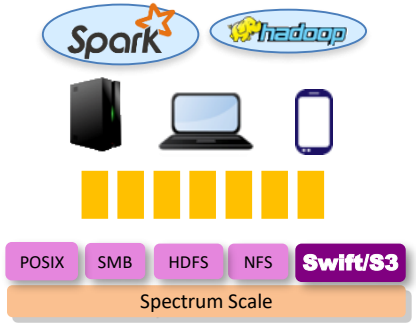
Spectrum Scale In-Place Analytics (2/2)

NFS interface from HDFS



- HDFS doesn't support random read/write, only append mode
- HDFS NFS Gateway has to write data from clients to the local disks first and then move it to HDFS to handle the out-of-order write requests
- No HA for NFS Gateway so far

Spectrum Scale In-place Analytics Solution



- ✓ Rich data ingest interface(SMB, NFS, HDFS, swift/S3 etc)
- ✓ Spectrum Scale Protocol HA
- ✓ Random read/write support
- ✓ Efficient data ingest because of no data movement from local disk to Spectrum Scale
- ✓ Only one data copy and all data are visible immediately from all interface

Spectrum Scale BD&A New Features

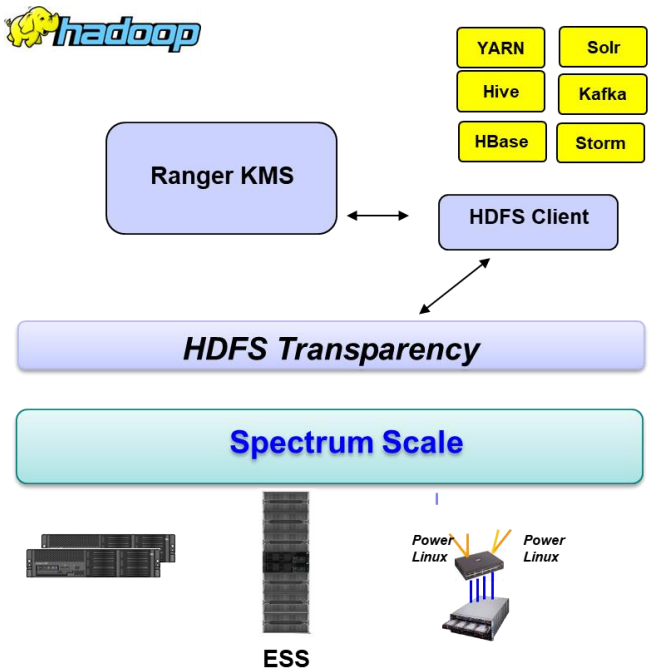
HDP 3.0 support with IBM Spectrum Scale

The latest certification covers HDP 3.0 testing with the most current mod update of IBM Spectrum Scale 5.0.

Recent highlights:

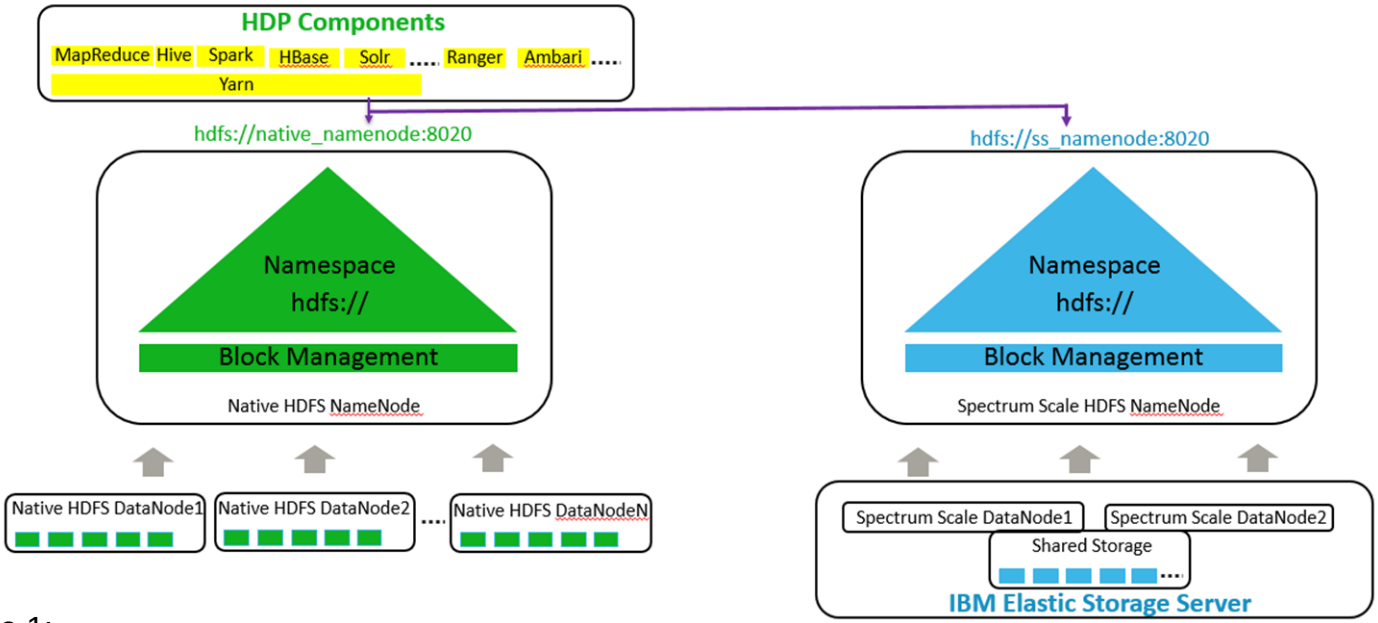
- Spectrum Scale Hadoop Storage Tiering with IBM Spectrum Scale to be able to access IBM Spectrum Scale namespace from existing Hadoop cluster setup with native HDFS.
- IBM ESS comes with its own erasure coding implementation which is not just meant for the cold or archive tier. But it allows you to run analytics directly on the erasure coded data avoiding 3-way replication completely
- IBM Spectrum Scale offers in-built encryption support already. We are now support for HDFS level encryption as well for IBM Spectrum Scale HDFS transparency connector.

Support Native HDFS Encryption



- Native HDFS encryption support since HDFS Transparency 2.7.3-4 and 3.0.0
- Ranger KMS will manage the key permission for different users
- Encrypted data can only be read from HDFS interface and you can not decrypt data from Spectrum Scale POSIX/NFS/SMB

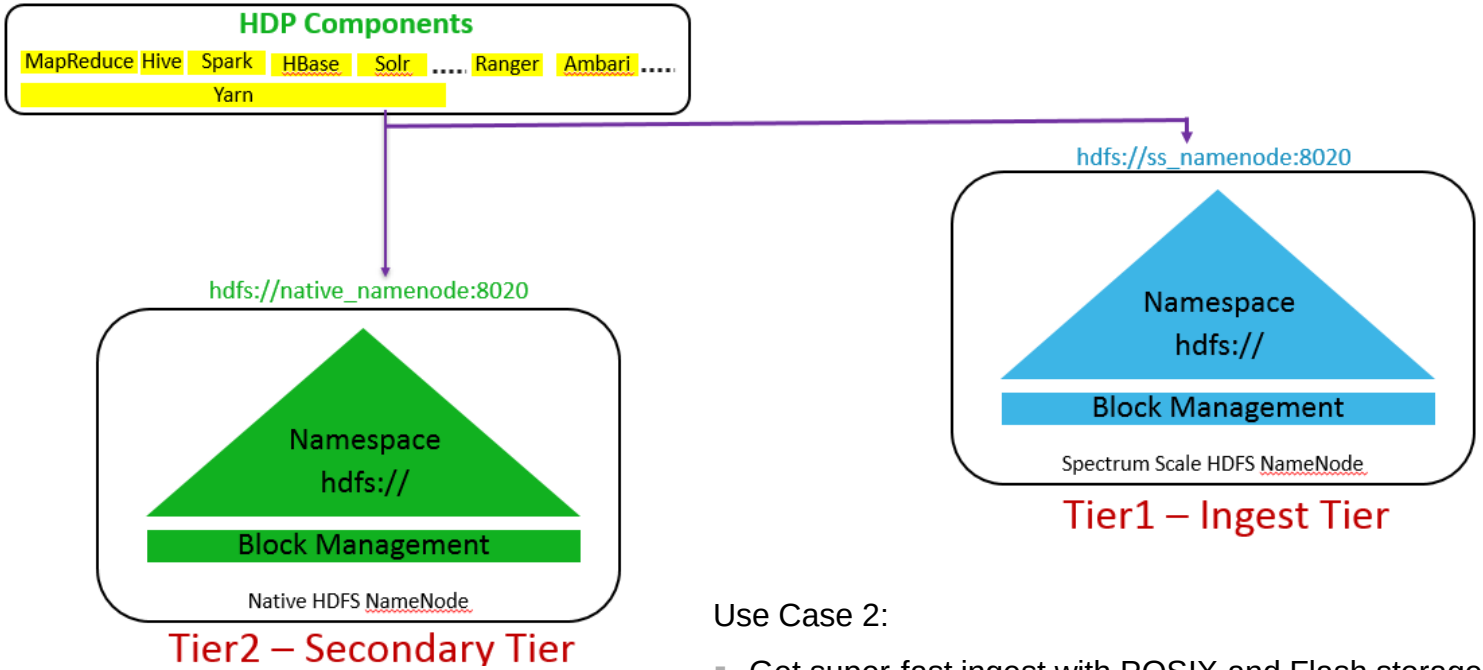
Hadoop storage tiering with IBM Spectrum Scale (1/3)



Use Case 1:

- Enterprises that already have a standard HDP cluster with native HDFS can now add ESS as a storage tier in the same HDP cluster.
- This will help enterprises manage cluster sprawl by adding ESS-based shared storage to their existing HDP clusters.

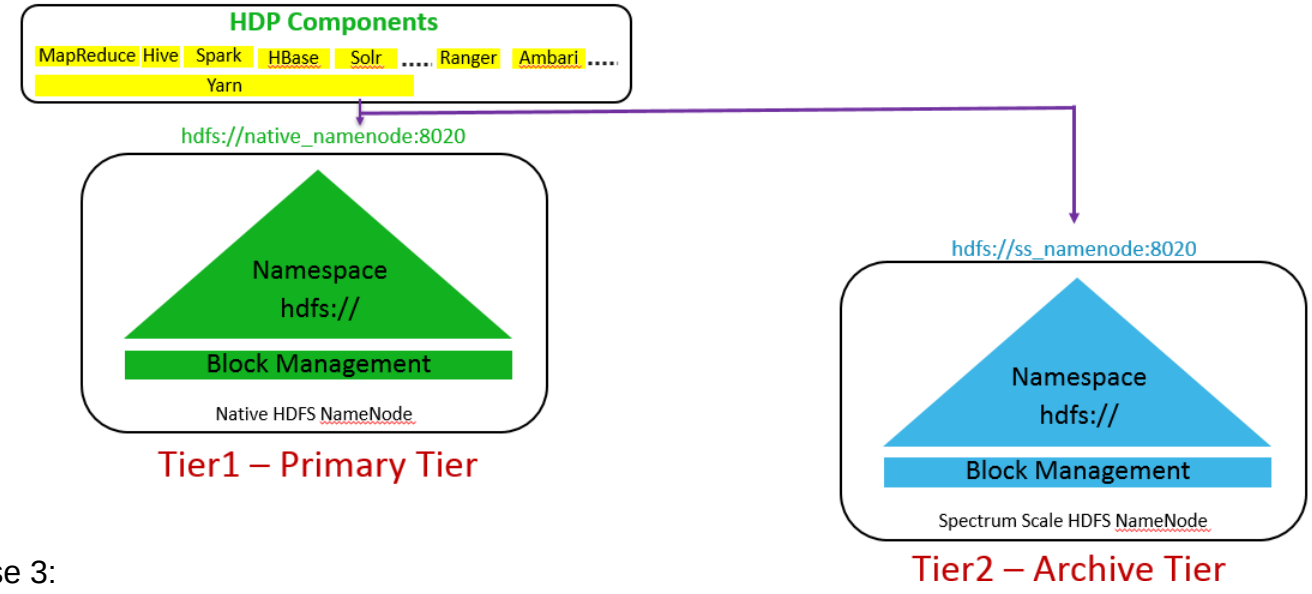
Hadoop storage tiering with IBM Spectrum Scale (2/3)



Use Case 2:

- Get super-fast ingest with POSIX and Flash storage
- Run in-place analytics directly on tier1 storage
- Move cold data to native HDFS tier for archiving

Hadoop storage tiering with IBM Spectrum Scale (3/3)



Use Case 3:

- Move cold data to ESS shared storage
- Grow storage independent of compute
- Eliminate 3-way replication

Terasort Performance HDFS vs IBM ESS

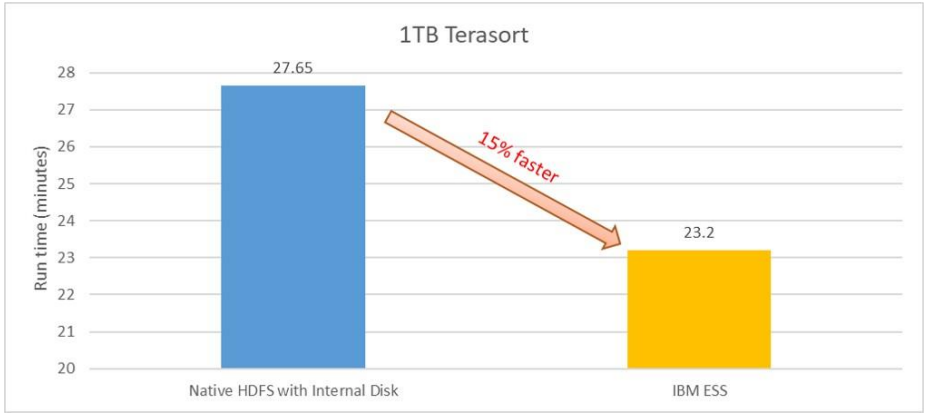
1 x NameNode
Power 822
SMT=8
160 processors
512GB memory
RHEL 7.5
HDP 3.0
Ambari 2.7.1

6 x DataNode
Power 922
SMT=2
158 processors
512GB memory
4 x 8TB HDD drive
2 x 3TB NvME drive for shuffle
RHEL 7.5
HDP 3.0
Ambari 2.7.1

ESS GL4s
ESS 5.3.1.1
Spectrum Scale 5.0.1.2
4TB disk drive
120Gb LACP bond ethernet
1MB 3way mirror metadata
16MB 8+2P data

1 x NameNode
Spectrum Scale 5.0.2
Power 822
SMT=8
160 processors
512GB memory
RHEL 7.5
HDP 3.0
Ambari 2.7.1

6 x DataNode
Spectrum Scale 5.0.2
Power 922
SMT=2
158 processors
512GB memory
4 x 8TB HDD drive
2 x 3TB NvME drive for shuffle
RHEL 7.5
HDP 3.0
Ambari 2.7.1



ESS based shared storage setup performed 15% faster compared with native HDFS with internal disks.

Blog: Hadoop Performance for disaggregated compute and storage configurations based on IBM Spectrum Scale Storage

Key difference between HDFS Transparency and native HDFS

Native HDFS feature	Status in Transparency
encryption	Supported
Storage type	Not supported; take Spectrum Scale storage pool and policy for alternative.
Native HDFS caching(interface "hdfs cacheadmin")	Not supported; Spectrum Scale has its own caching mechanism
NFS Gateway	Not supported; Spectrum Scale protocol is better in scaling and performance
The interface "hadoop dfs xx" The interface "hdfs dfs xxx"	supported
Other interface from "hdfs"(such as "hdfs namenode xxx")	Not supported; transparency doesn't need these interface because they are specific for native HDFS
disctcp over snapshot(--diff option in distcp)	Not supported; Transparency supports other options in distcp except for the option --diff

Refer chapter 10 in HDFS Transparency Guide

Thank You.

IBM Storage & SDI