IBM Storage & SDI

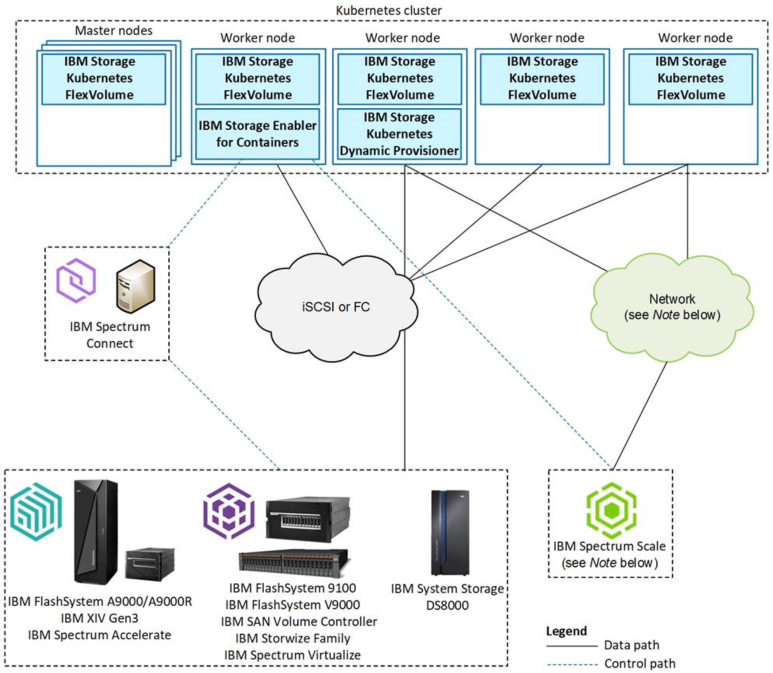
# What’s New in Spectrum Scale and ESS

Wei Gong

Spectrum Scale Development and Client Adoption

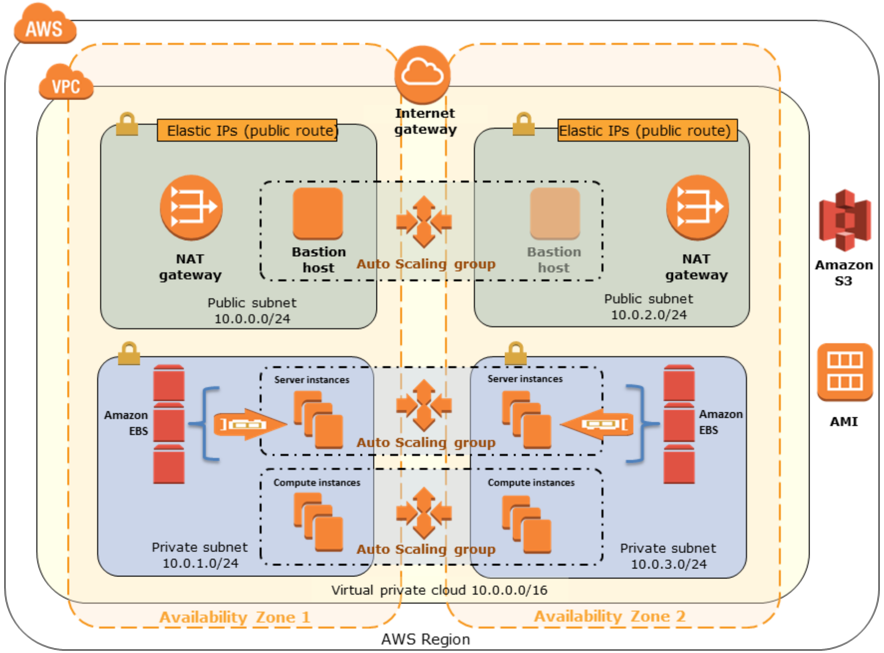
# New workload

# IBM Spectrum Scale with IBM Storage Enabler for Container


Kubernetes cluster
Master nodes
IBM Storage Kubernetes FlexVolume
Worker node
IBM Storage Kubernetes FlexVolume
IBM Storage Enabler for Containers
Worker node
IBM Storage Kubernetes FlexVolume
IBM Storage Kubernetes Dynamic Provisioner
Worker node
IBM Storage Kubernetes FlexVolume
Worker node
IBM Storage Kubernetes FlexVolume
IBM Spectrum Connect
iSCSI or FC
Network (see *Note* below)
IBM FlashSystem A9000/A9000R
IBM XIV Gen3
IBM Spectrum Accelerate
IBM FlashSystem 9100
IBM FlashSystem V9000
IBM SAN Volume Controller
IBM Storwize Family
IBM Spectrum Virtualize
IBM System Storage DS8000
IBM Spectrum Scale (see *Note* below)
Legend
Data path
Control path

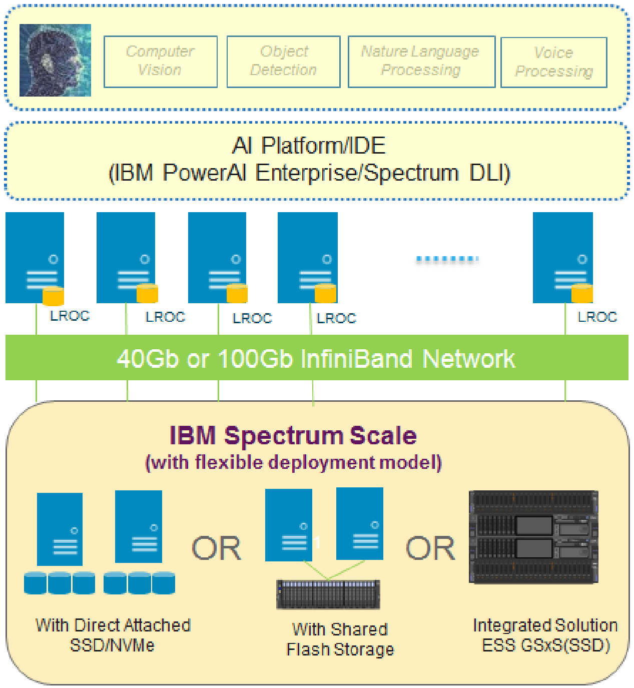
- IBM Storage Enabler for Containers allows IBM storage systems to be used as persistent volumes for stateful applications running in Kubernetes clusters.
- Based on open source project Ubiquity
- Create new or use exist fileset to export storage service to container
- Leverage Spectrum Scale Rest API and Quota features
- Supports kubernetes and IBM Cloud Private (ICP)
- Support RHEL 7 on x86/ppc64le/System Z, Ubuntu on x86 and SLES12 on System Z

# IBM Spectrum Scale on AWS

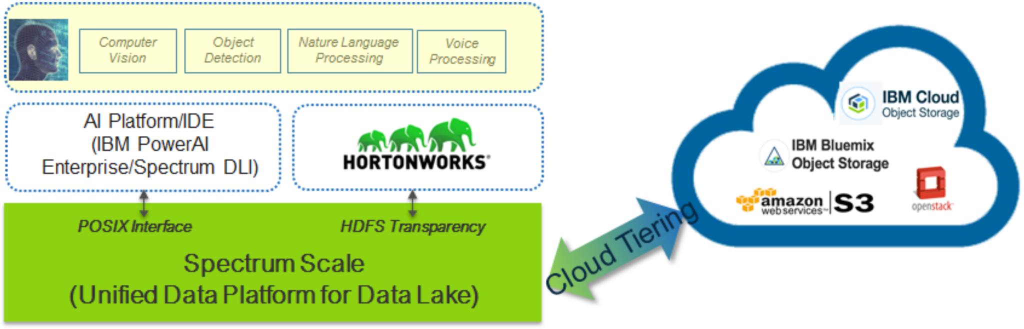


- IBM Spectrum Scale on AWS automates the deployment of IBM Spectrum Scale on AWS for users who require highly available access to a shared name space across multiple instances with good performance, without requiring an in-depth knowledge of IBM Spectrum Scale.
- Deploy IBM Spectrum Scale on AWS in 20-45 minutes (depending on the number and types of instances you're using)
- Use AWS EC2 instance as Spectrum Scale nodes (NSD and clients) and EBS as NSD
- Support user interaction to set parameters for the cluster and file system
- Knowledge Center

# IBM Spectrum Storage for AI


Computer Vision
Object Detection
Nature Language Processing
Voice Processing
AI Platform/IDE
(IBM PowerAI Enterprise/Spectrum DLI)
LROC
LROC
LROC
LROC
LROC
40Gb or 100Gb InfiniBand Network
IBM Spectrum Scale
(with flexible deployment model)
OR
OR
With Direct Attached SSD/NVMe
With Shared Flash Storage
Integrated Solution ESS GSxS(SSD)

IBM Spectrum Scale Reference Architecture for AI



Reference Architecture for large Data Lake

- IBM Spectrum Scale can fully meet the performance and scalability requirements for AI workloads
- Coupled with continuous innovation including performance enhancements such as LROC and RDMA.
- IBM Spectrum Scale also provides advanced features that help to better fit it into the bigger infrastructure picture, include integration with Hadoop environments to support in-place analytics, transparent cloud tiering for big data etc.

Blog: Your Optimal Choice of AI Storage for Today and Tomorrow

# Performance

# File System Core Performance Enhancement

- maxActiveIallocSegs attribute improves the performance of deletes and unlinks
  - Change default from 1 to 8, valid range is 1-64.
  - Used to improve file deletion performance in scenarios like following:
    - Nodes have created a large number of files in separated directories, each node creating files in its own directories.
    - Processes or threads on multiple nodes are concurrently delete files in those directories.

- maxStatCache
  - Make it take effect again on Linux which can provide significant performance enhancement for ‘ls -l’ like command, mdtest and so on
  - “mmcachectl” command can show if inode in in stat cache (in pagepool)
  - Set it to 4 x maxFilesToCache

- FSCK runtime estimate
  - --estimate-only option

# Operation and Monitoring

# autoBuildGPL configuration option

- autoBuildGPL configures a cluster to rebuild the GPL automatically whenever a new level of the Linux kernel is installed or whenever a new level of IBM Spectrum Scale is installed.
- Before starting GPFS, if the kernel module is missing, automatically call mmbuildgpl to build the GPL if autoBuildGPL parameter is configured.
- This parameter does not apply to the AIX® and Windows environments.

*mmchconfig autoBuildGPL={no|yes|quiet|verbose|quiet-verbose|verbose-quiet}*

*Where:*

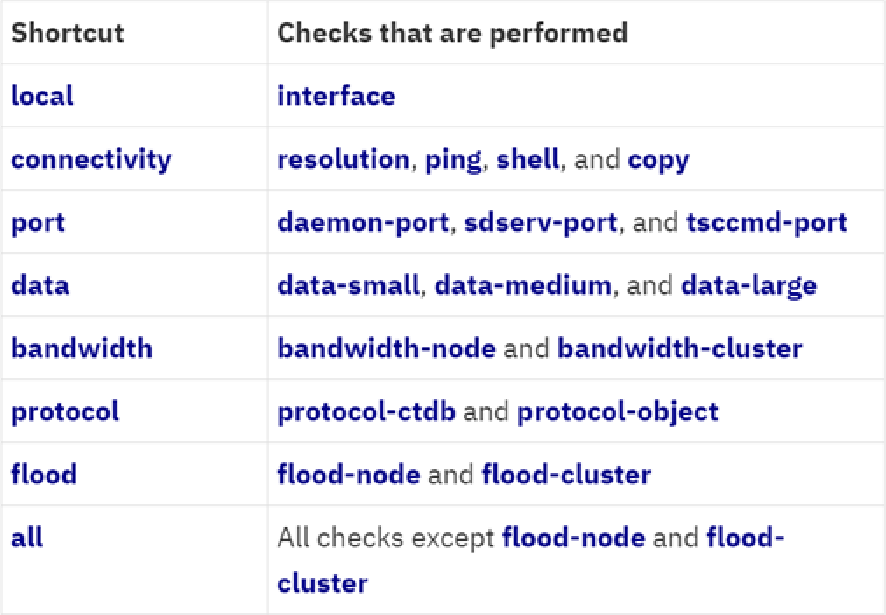
- *no This is the default. No action will be taken if no kernel module is found*
- *yes mmbuildgpl will be called to build the GPL if the kernel module is missing*
- *quiet Same as yes. The mmbuildgpl command will be called with --quite option.*
- *verbose Same as yes. The mmbuildgpl command will be called with -v option.*
- *quiet-verbose or verbose-quiet*
- *Both --quite and -v will be passed to mmbuildgpl*

# mmnetverify

- Support remote cluster
  - Test each cluster which is listed in the mmsdrfs file
  - Can test more with the **--cluster** command line parameter
- Can run before and after cluster is created
  - --configuration-file option

```
node Node [AdminName]
rshPath Path
rcpPath Path
tscTcpPort Port
mmsdrservPort Port
tscCmdPortRange Min-Max
subnets Addr[,Addr...]
cluster Name Node[,Node...]
```

Network check activities:



| Shortcut | Checks that are performed |
|---|---|
| **local** | **interface** |
| **connectivity** | **resolution**, **ping**, **shell**, and **copy** |
| **port** | **daemon-port**, **sdserv-port**, and **tsccmd-port** |
| **data** | **data-small**, **data-medium**, and **data-large** |
| **bandwidth** | **bandwidth-node** and **bandwidth-cluster** |
| **protocol** | **protocol-ctdb** and **protocol-object** |
| **flood** | **flood-node** and **flood-cluster** |
| **all** | All checks except **flood-node** and **flood-cluster** |

# mmcachectl command

- The mmcachectl command displays information about files and directories in the local page pool cache
- Can display information for a single file, for the files in a fileset, or for all the files in a file system
- Doesn’t support LROC so far

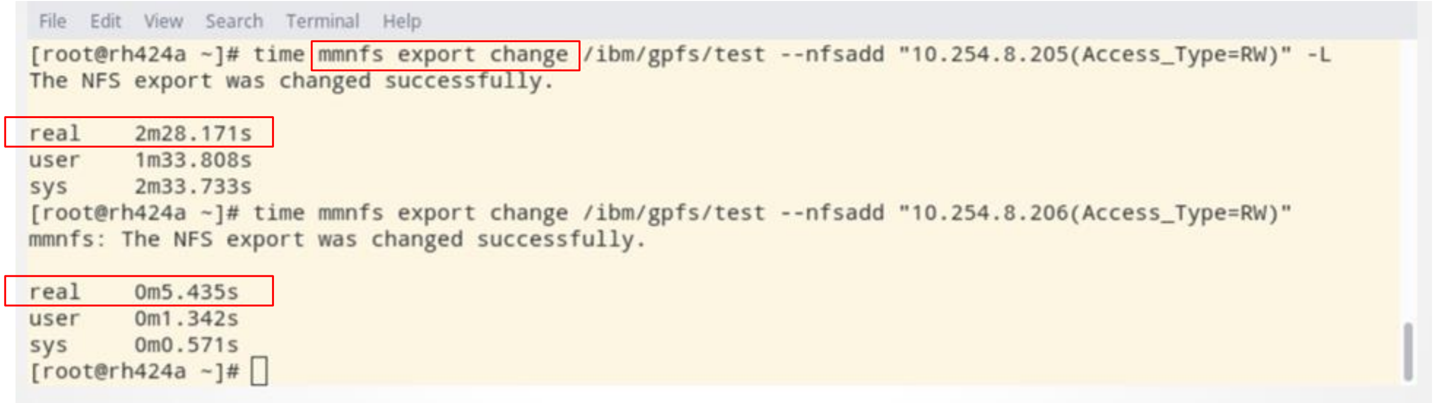
```
[root@icp1 ~]# mmcachectl show --show-filename
FSname       Fileset  Inode    SnapID   FileType     NumOpen    NumDirect  Size         Cached         Cached          FileName
             ID                                      Instances  IO         (Total)      (InPagePool)   (InFileCache)
--------------------------------------------------------------------------------------------------------------------------------
gpfs1        0        89089    0        file         0          0          0            0              F               /gpfs1/FooFile
gpfs1        0        3        0        directory    0          0          262144       262144         F               /gpfs1/
gpfs1        0        89088    0        file         0          0          0            0              F               /gpfs1/testfile
gpfs1        0        4        0        special      1          0          4194304      4194304        F               -

File count: 4
[root@icp1 ~]#
```

# Maintenance mode

- The maintenance mode is designed to enable a maintenance window to a Spectrum Scale file system. Used when some maintenance actions need to be taken to the NSD disks or server nodes, including the backend storage systems
- Goals
  - Disable file system mounts while maintenance is occurring or it is already on
  - No write I/O activities (except the write to turn off the maintenance mode)
  - No disk would be marked down
  - A per file system basis
  - File system health check operations are allowed.
- When to use file system maintenance mode
  - Take maintenance to NSD disks in server host side or backend storage
  - Take maintenance to NSD server nodes, like shutting down the server nodes.
  - Shutdown the whole Spectrum Scale cluster

# NFS Protocols

- Re-write mmnfs command to make it run faster



```
[root@rh424a ~]# time mmnfs export change /ibm/gpfs/test --nfsadd "10.254.8.205(Access_Type=RW)" -L
The NFS export was changed successfully.

real    2m28.171s
user    1m33.808s
sys     2m33.733s
[root@rh424a ~]# time mmnfs export change /ibm/gpfs/test --nfsadd "10.254.8.206(Access_Type=RW)"
mmnfs: The NFS export was changed successfully.

real    0m5.435s
user    0m1.342s
sys     0m0.571s
[root@rh424a ~]#
```

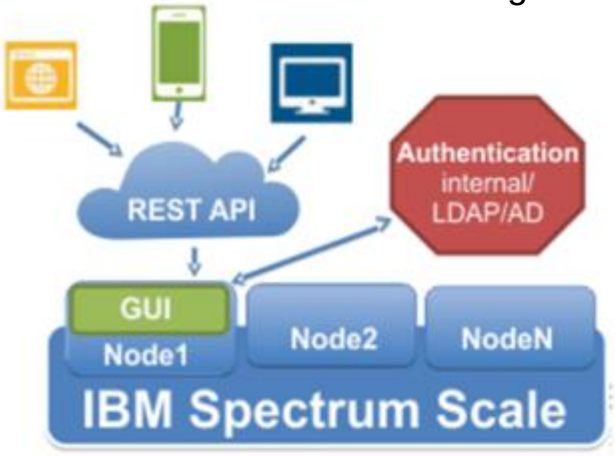
- Pseudo path allows to hide the actual path on the gpfs cluster
  - mnnfs export command --pseudo option
  - Only support CLI so far. No GUI integration yet

- Ganesha_stat interface
  - Enable to display statistics in GPFS layer (FSAL)

# GUI and Rest API

## GUI optimization

- Ability to enable and disable File Audit Logging
- Reduce CPU and Memory on GUI node
- Reduce call to mmhealth
- Help with ESS EMS and GUI node for large cluster

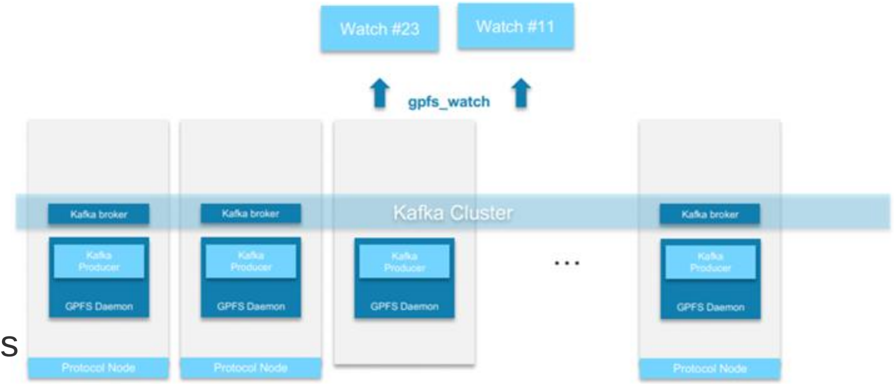


## Rest API extra endpoints

| URL | Operation | Description |
|---|---|---|
| /filesystems/{filesystemname}/audit | PUT | Enable/Disable FAL (mmaudit) |
| /smb/shares/{shareName}/acl | DELETE, GET, PUT | SMB share ACL management |

# Security

# Watch Folder



- Flexible API that allows programmatic actions to be taken based on filesystem events
  - Run against folders, independent fileset
- Modeled after Linux inotify, but works with clustered filesystems, and supports recursive watches for filesets and inode spaces
- Primary components
  - GPFS API (gpfs_watch.h) – Refer /usr/lpp/mmfs/samples/util/tswf.C as example
  - Mmwatch command - provides information of all watches running within cluster
- A watch folder application uses the API to run as an executable C program on node within GPFS cluster
  - Utilizes message queue to receive events from multiple nodes and consume from the node running the program
  - Events come in from all eligible nodes within cluster and from accessing clusters

25 watches per file system

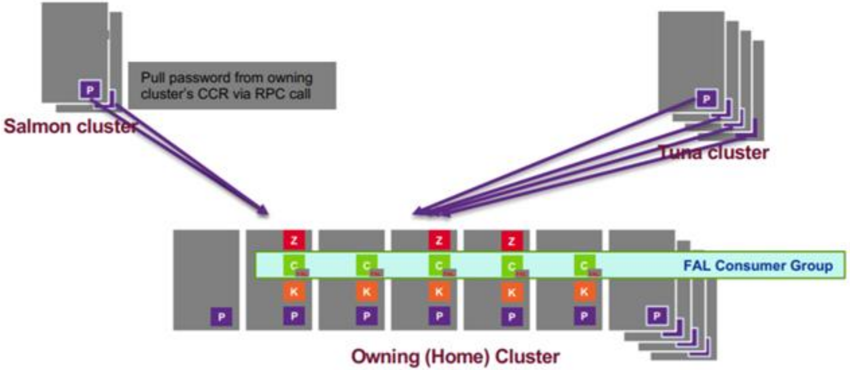
100 watches per cluster

# File Audit Log

- Have option to choose a subset of events to monitor
  - ACLCHANGE,CLOSE,CREATE,DESTROY,GPFSATTRCHANGE,OPEN,RENAME,RMDIR,UNLINK,XATTRCHANGE
  - Ability to update events being monitored without disabling and re-enabling file audit logging

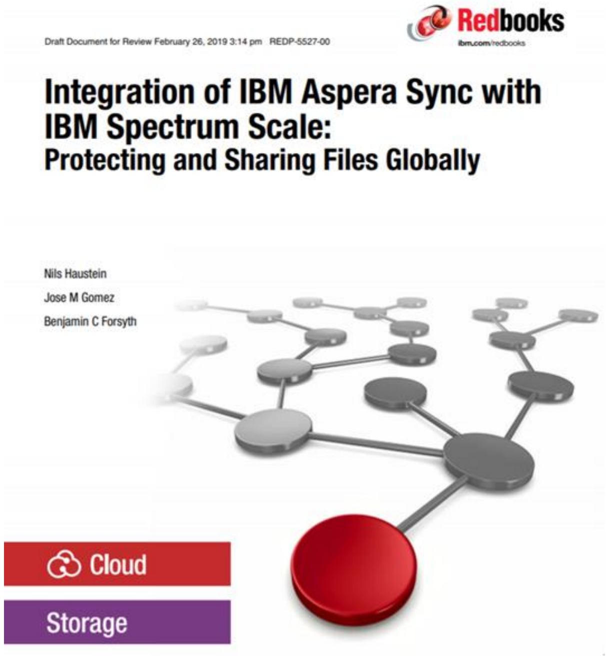
- Require 40 GB local disk space per file system being audit per broker node
  - Improved parallelization by doubling the number of partitions and maximum segment size
  - Can be overridden using the “--degraded” option that only requires 10 GB of local disk space per filesystem being audited per broker node. Save on space by reducing number of partitions

- Support for remote mounted filesystem, such as in a ESS cluster
  - Producers on remote clusters can send events to file system owning cluster



# Data Management

# IBM Aspera Sync with IBM Spectrum Scale



Draft Document for Review February 26, 2019 3:14 pm REDP-5527-00

Integration of IBM Aspera Sync with IBM Spectrum Scale:
Protecting and Sharing Files Globally

Nils Haustein
Jose M Gomez
Benjamin C Forsyth

- Use Case:
  - File share
  - File migration
  - Disaster Recovery
- Aspera sync 3.9 and higher incorporates support for IBM Spectrum Scale extended attributes
- Can leverage Spectrum Scale policy engine for fast scan and parallel migration

# Big Data and Analytics

# Hortonworks + Cloudera

# What are we working on for coming release?

ibm.com/storage

# Thank You.

IBM Storage & SDI