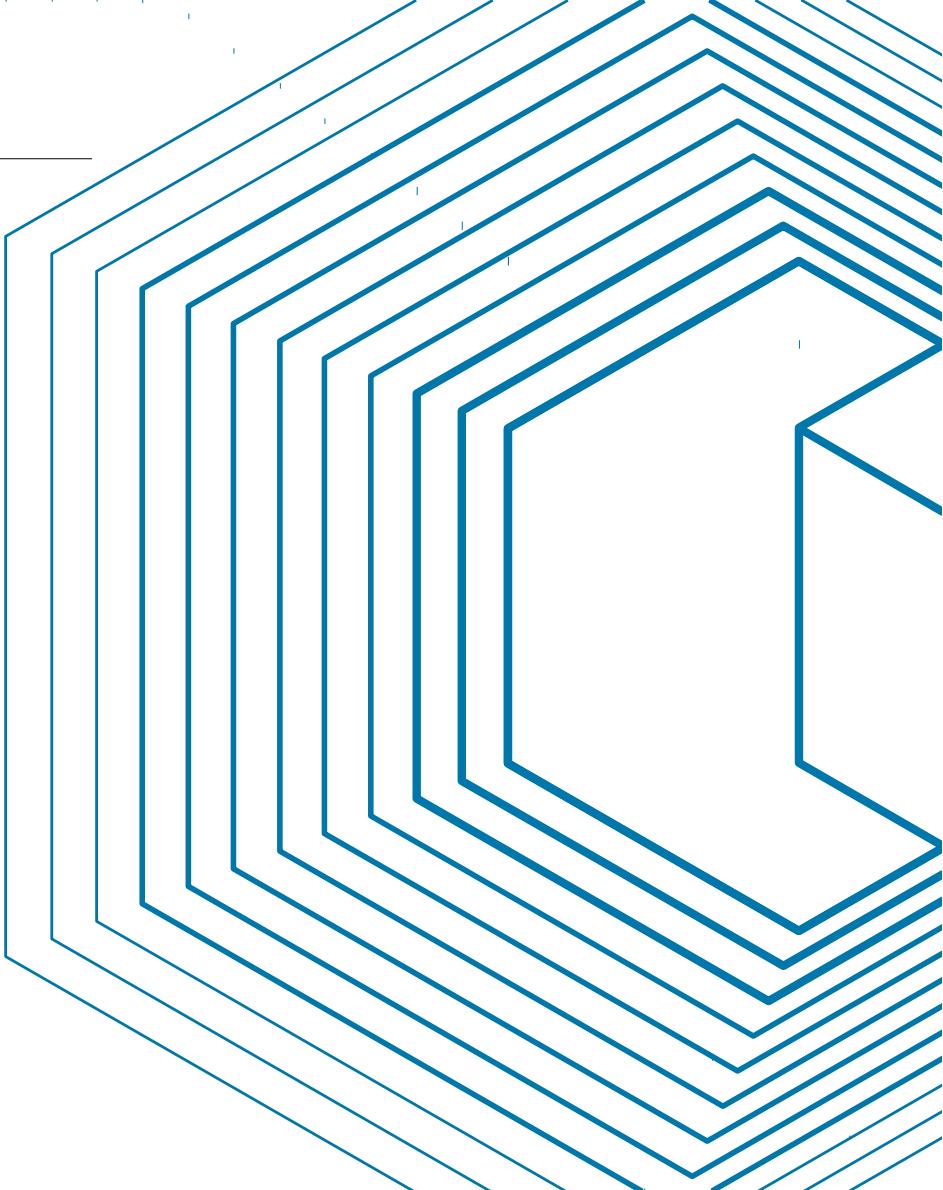


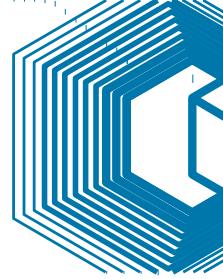
Spectrum Scale

news & nice



olaf.weiser@de.ibm.com
IBM Deutschland
SpectrumScale ESCC Germany

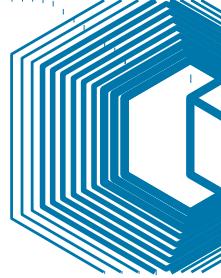
Agenda



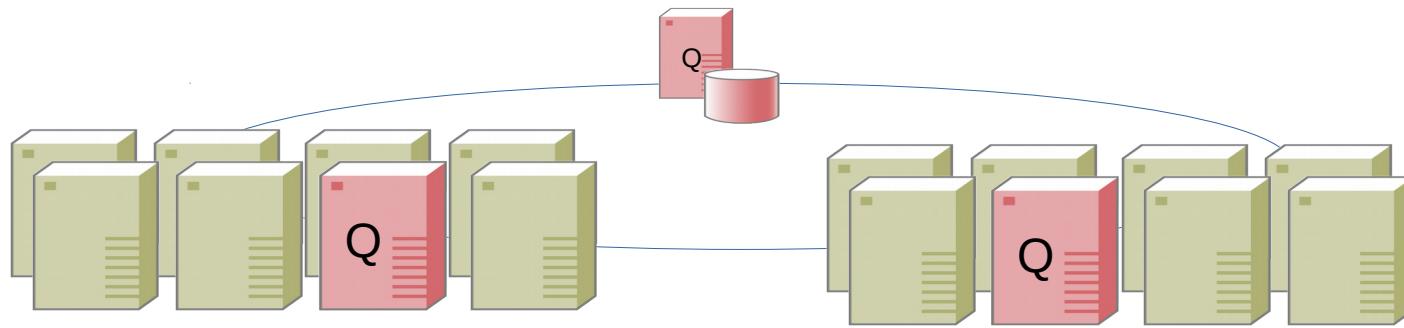
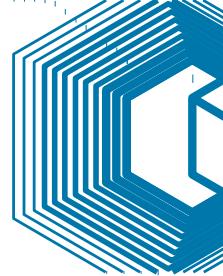
best practice

some news

Agenda

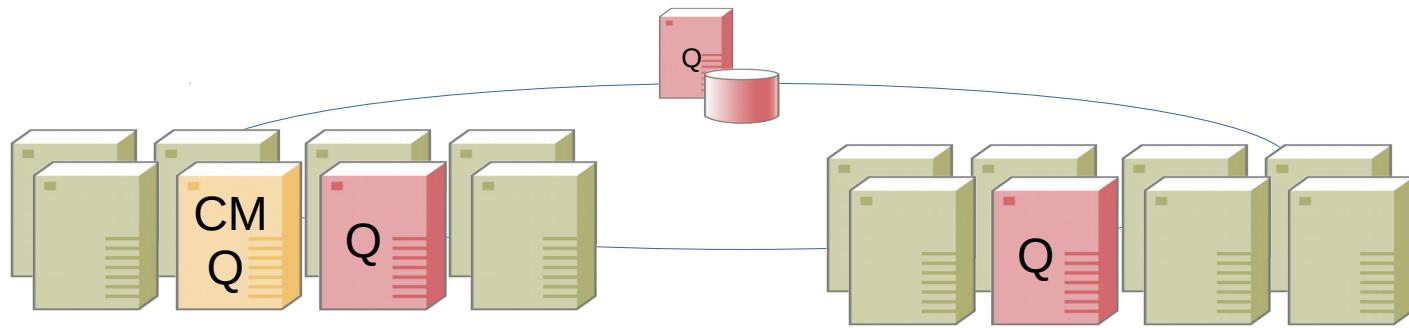
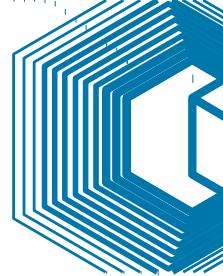


- ▶ Handling node failures
- ▶ CCR restore best practice
- ▶ Auto tuning
- ▶ NFSv4 ACL
- ▶ ESS for SAP HANA workloads some news
- ▶ ubiquity



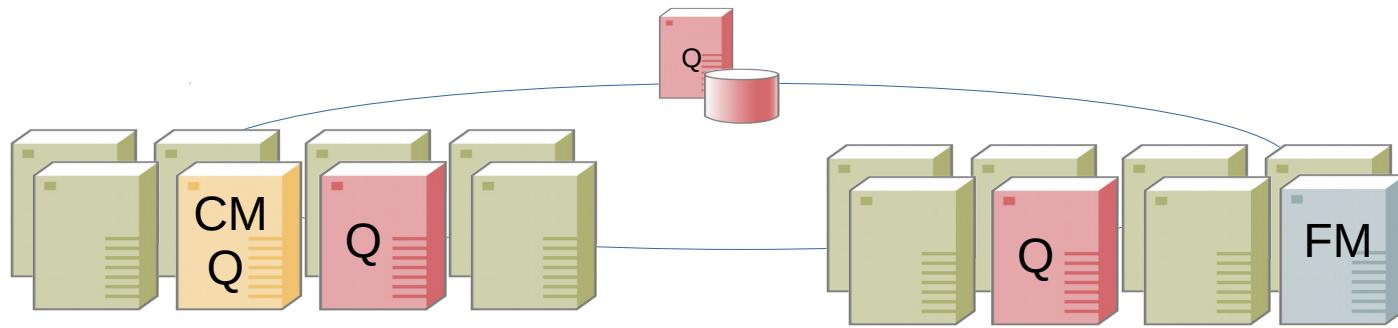
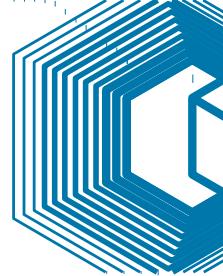
Cluster is up n running, when

- majority of Quorum nodes are up n running
- CCR: configuration changes rely on availability of Q nodes
- a Quorum node is a special node role



Cluster is up n running, when

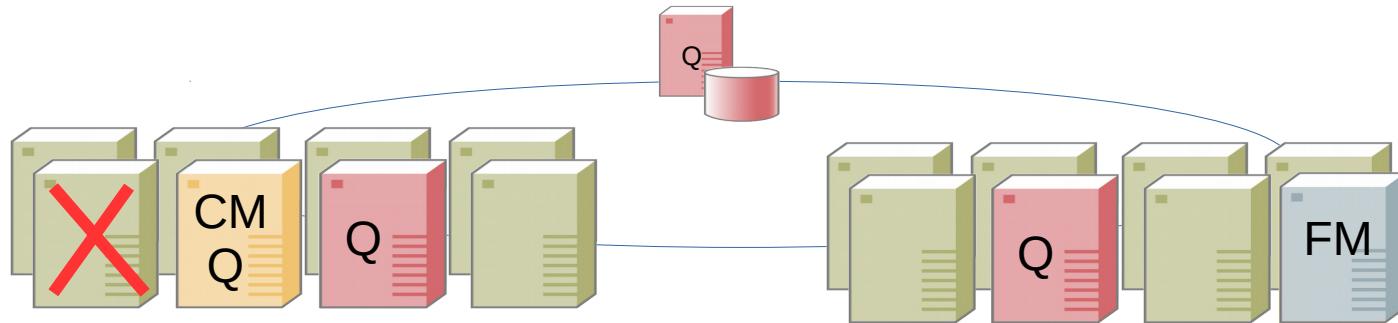
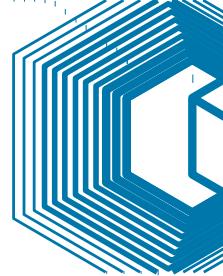
- majority of Quorum nodes are up n running
- CCR: configuration changes rely on availability of Q nodes
- a Quorum node is a special node role
- a cluster manager (CM) is a special node



Cluster is up n running, when

- majority of Quorum nodes are up n running
- CCR: configuration changes rely on availability of Q nodes
- a Quorum node is a special node role
- the cluster manager (CM) is a special node
- a File system manager (FM) is a special node

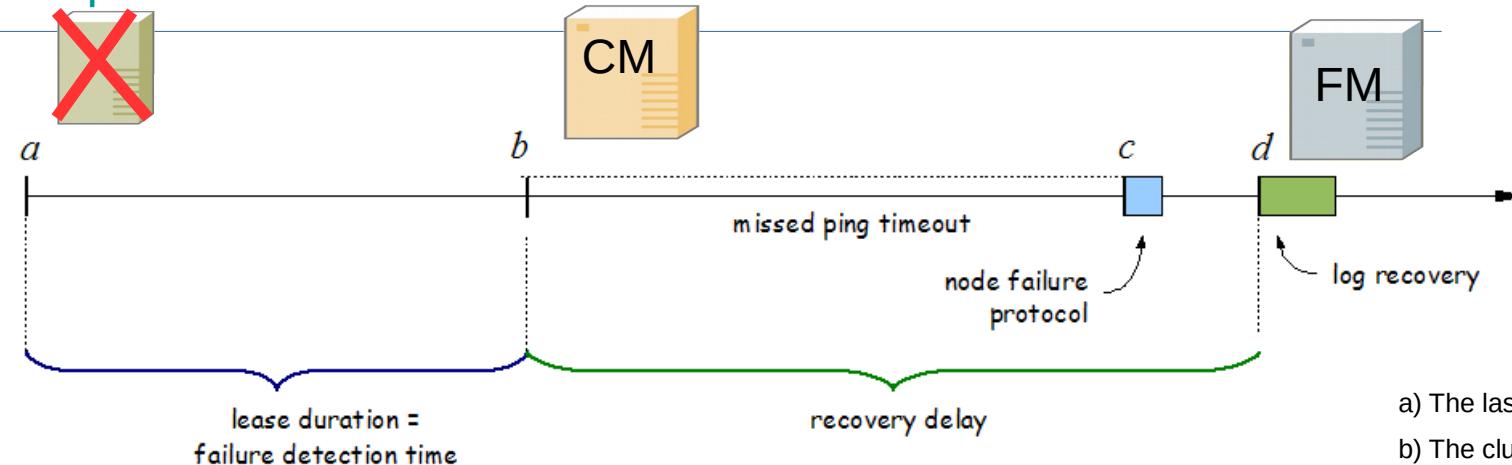
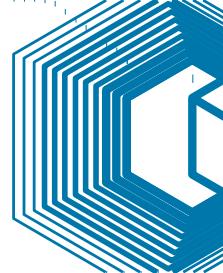
Simple node failure



Cluster is up n running, when

- majority of Quorum nodes are up n running
- CCR: configuration changes rely on availability of Q nodes
- a Quorum node is a special node role
- the cluster manager (CM) is a special node
- a File system manager (FM) is a special node

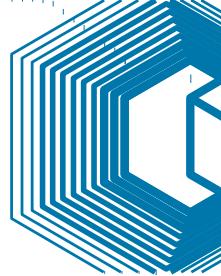
Simple node failure



- a) The last time the failed node renewed its lease
- b) The cluster manager detects that the lease has expired, and starts pinging the node
- c) The cluster manager decides that the node is dead and runs the node failure protocol
- d) The file system manager starts log recovery

- **failureDetectionTime** (default 35 seconds)
- **leaseRecoveryWait** (default 35 seconds)
- leaseDuration is set equal to failureDetectionTime
- missedPingTimeout is set equal to recoveryWait minus a few seconds*

*to allow time for the cluster manager to run the node failure protocol before the recoveryWait runs out.



failureDetectionTime

mmfs.log.latest

```
[root@beer1 beer]# tail -f /var/adm/ras/mmfs.log.latest
[....]
2017-03-01_08:31:28.934+0100: [N] Node 10.0.1.13 (beer3) lease renewal is overdue. Pinging to check if it is alive
```

failureDetectionTime

Definition: The number of seconds it will take the GPFS cluster manager to detect that a node has not renewed its disk lease. If a node does not renew its disk lease in failureDetectionTime seconds, the GPFS cluster manager will start to ping the node to determine if the node has failed.

Default Value: 35 - Minimum and Maximum Value: 10 and 300

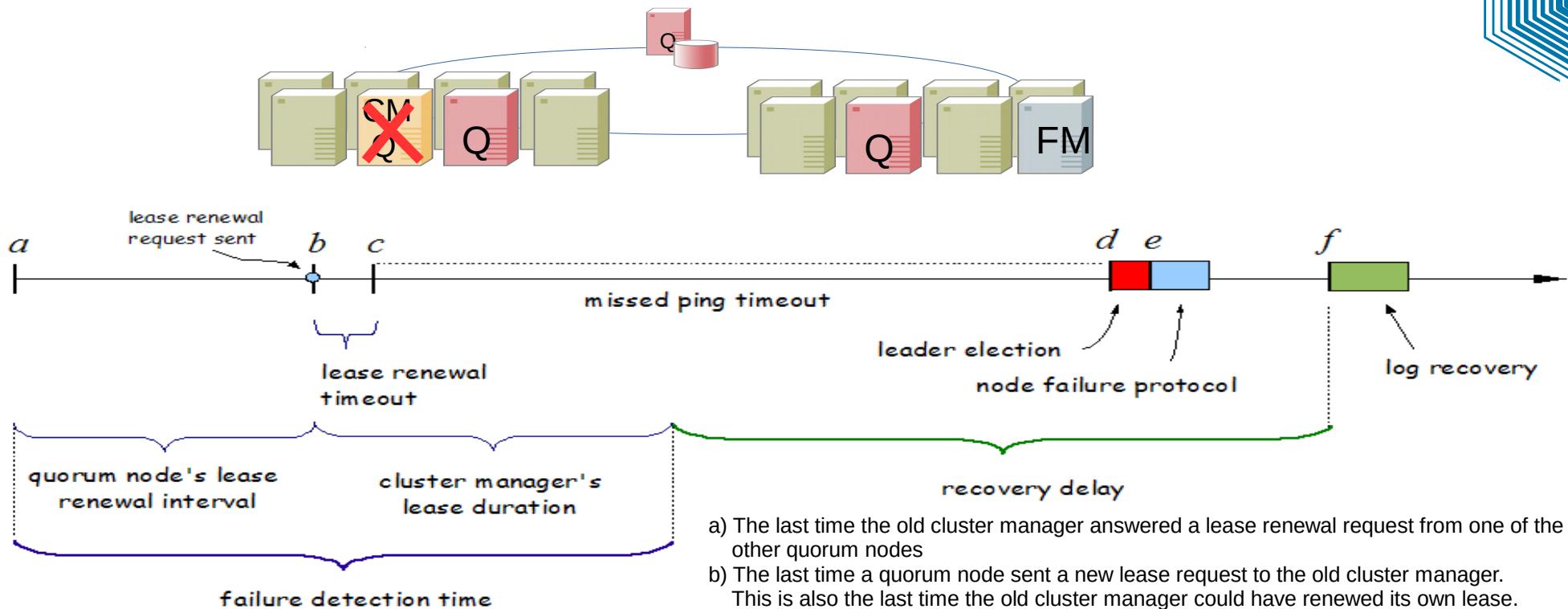
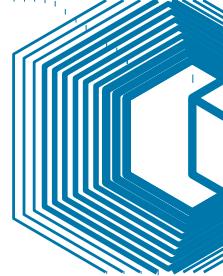
hint:

clusters with a large number of nodes, FDT may be increased to reduce the number of lease renewal messages received by the GPFS cluster manager. Example: 5000 nodes / 35 seconds = 142 lease renewals / second.

From experience, if the value of failureDetectionTime is increased, it is sometimes increased to 60 or 120 seconds.

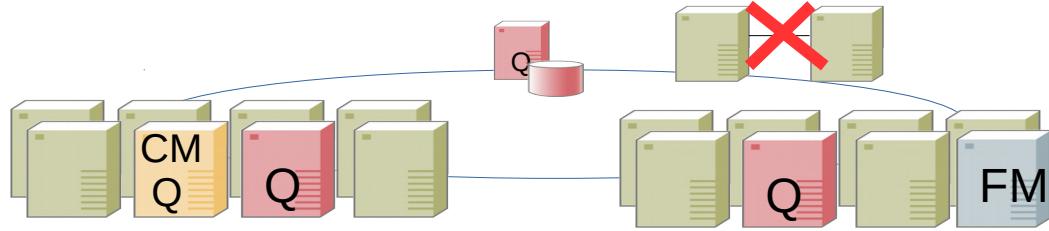
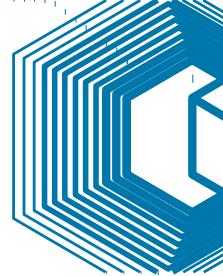
Notes: GPFS must be down on all nodes to change the value of failureDetectionTime

Cluster manager node failure



- The last time the old cluster manager answered a lease renewal request from one of the other quorum nodes
- The last time a quorum node sent a new lease request to the old cluster manager. This is also the last time the old cluster manager could have renewed its own lease.
- A quorum node detects that it is unable to renew its lease and starts pinging the old cluster mgr
- The quorum node decides that the old cluster manager is dead and runs an election to take over as new cluster manager.
- The election completes and the new cluster manager runs the node failure protocol.
- The file system manager starts log recovery

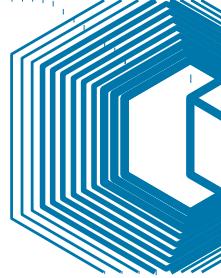
Network issues between a some other nodes



```
/* We have evidence that both nodes are still up. In this case, give
   preference to
   1. quorum nodes over non-quorum nodes
   2. local nodes over remote nodes
   3. manager-capable nodes over non-manager-capable nodes
   4. nodes managing more FSs over nodes managing fewer FSs
   5. NSD server over non-NSD server
   Otherwise, expel whoever joined the cluster more recently.
```

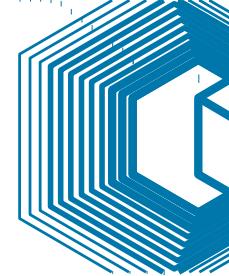
```
After all these criteria are applied, give a chance to the user
script
   to reverse the decision.
*/
```

Agenda



- ▶ Handling node failures
- ▶ **CCR restore**
- ▶ Auto tuning
- ▶ NFSv4 ACL
- ▶ ESS for SAP HANA workloads
- ▶ ubiquity

CCR / SDR restore in case of node failure – manual recover



– (1) – restore the node / reinstall the node

– (2) – check status
/var/mmfs empty

– (3) – install gpfs rpms

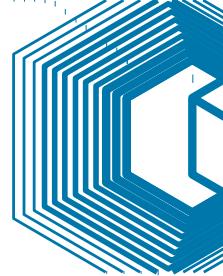
– (4) – mmbuildgpl / check status

```
[root@beer3 ~]# cd /var/mmfs  
-bash: cd: /var/mmfs: No such file or directory
```

```
[root@beer3 4.2.2.0]# rpm -ihv gpfs.base-4.2.2-0.x86_64.rpm gpfs.docs-4.2.2-0.noarch.rpm  
gpfs.ext-4.2.2-0.x86_64.rpm gpfs.gpl-4.2.2-0.noarch.rpm gpfs.gskit-8.0.50-57.x  
86_64.rpm gpfs.license.std-4.2.2-0.x86_64.rpm gpfs.msg.en_US-4.2.2-0.noarch.rpm  
Preparing... # ##### [100%]  
[...]
```

```
[root@beer3 4.2.2.0]# ll /var/mmfs/  
total 0  
drwxr-xr-x. 2 root root 64 Mar 4 10:44 ces  
drwxr-xr-x. 2 root root 6 Mar 4 10:43 etc  
drwxr-xr-x. 4 root root 40 Mar 4 10:43 gen  
drwxr-xr-x. 2 root root 6 Mar 4 10:43 mmbackup  
drwx----- 2 root root 6 Mar 4 10:43 mmpmon  
drwxr-xr-x. 2 root root 73 Mar 4 10:43 mmssysmon  
drwx----- 4 root root 34 Mar 4 10:43 ssl  
drwxr-xr-x. 3 root root 26 Mar 4 10:47 tmp
```

CCR / SDR restore in case of node failure – manual recover



– (5) – status gpfs on the failed node

```
[root@beer3 4.2.2.0]# mmgetstate
mmgetstate: This node does not belong to a GPFS cluster.
mmgetstate: Command failed. Examine previous error messages to determine cause.
[root@beer3 4.2.2.0]#
```

– (6) – Status from healthy node

```
[root@beer1 ~]# mmgetstate -a
beer3: mmremote: determineMode: Missing file /var/mmfs/gen/mmsdrfs.
beer3: mmremote: This node does not belong to a GPFS cluster.
mmdsh: beer3 remote shell process had return code 1.
```

Node number	Node name	GPFS state
1	beer1	active
2	beer2	active
3	beer3	unknown

```
[root@beer1 ~]#
```

CCR / SDR restore in case of node failure – manual recover



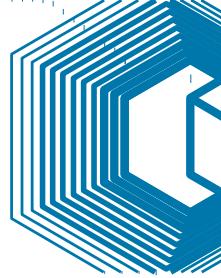
– (7) – sdrrestore on the failed node

```
[root@beer3 ~]# mmsdrrrestore -p beer1 -R /usr/bin/scp
Sat Mar  4 10:56:46 CET 2017: mmsdrrrestore: Processing node beer3
genkeyData1
mmsdrrrestore: Node beer3 successfully restored.
[root@beer3 ~]# mm1snodenode
GPFS nodeset      Node list
-----
          beer1      beer1 beer2 beer3
[root@beer3 ~]#
```

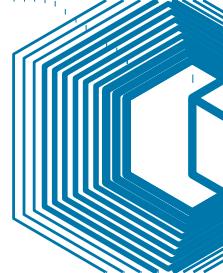
– (8) – check status

```
[root@beer1 ~]# mmgetstate -a
              Node number  Node name      GPFS state
-----
              1            beer1        active
              2            beer2        active
              3            beer3        down
[root@beer1 ~]#
```

Agenda



- ▶ Handling node failures
- ▶ CCR restore
- ▶ **Auto tuning**
- ▶ NFSv4 ACL
- ▶ ESS for SAP HANA workloads
- ▶ ubiquity



SA23-1452-06 Administration and Programming Reference

Attribute name and Description	-i option allowed	-I option allowed	GPFS must be stopped on all	List of NodeNames allowed	Change takes effect
workerThreads Sets an integrated group of variables that tune file system performance.	no	no	no	yes	on restart of the daemon
worker1Threads Sets the maximum number of concurrent file operations	yes (only when adjusting value down)	yes (only when adjusting value down)	no	yes	on restart of the daemon

workerThreads

Controls an integrated group of variables that tune file system performance. Use this variable to tune file systems in environments that are capable of high sequential or random read/write workloads or small-file activity. For new installations of the product, this variable is preferred over **worker1Threads** and **prefetchThreads**.

Important: If you set **workerThreads** to a non-default value, do not set **worker1Threads**.

The default value is 48. The valid range is 1-8192. However, the maximum value of **workerThreads** plus **prefetchThreads** plus **nsdMaxWorkerThreads** is 8192. The **-N** flag is valid with this variable.

Automatic tuning - workerThreads



```
[root@beer1 ~]# mmfsadm dump version | head -3  
Dump level: verbose  
Build branch "4.2.2.0".
```

```
[root@beer1 ~]# mmlsconfig  
Configuration data for cluster beer1:
```

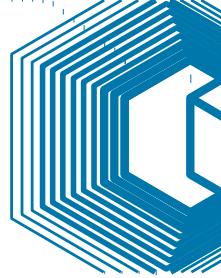
```
-----  
clusterName beer1
```

```
[...]
```

```
workerThreads 96
```

workerThreads

```
[root@beer1 ~]# mmfsadm dump config | grep "^\."  
. flushedDataTarget 32  
. flushedInodeTarget 32  
. logBufferCount 3  
. logWrapThreads 12  
. maxAllocRegionsPerNode 4  
. maxBackgroundDeletionThreads 4  
. maxBufferCleaners 24  
. maxFileCleaners 24  
. maxGeneralThreads 512  
. maxInodeDeallocPrefetch 8  
. parallelWorkerThreads 16  
. prefetchThreads 72  
. sync1WorkerThreads 24  
. sync2WorkerThreads 24  
. syncBackgroundThreads 24  
. syncWorkerThreads 24  
. worker3Threads 8  
[root@beer1 ~]#
```



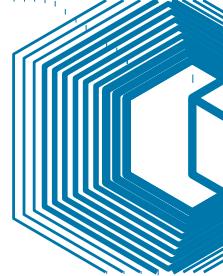
Auto tuning – client side – ignorePrefetchLUNCount / pagepool

```
[root@beer1 gpfs]# mmfsadm dump config | grep -e ignorePrefetchLUNCount  
ignorePrefetchLUNCount 0  
[root@beer1 gpfs]#
```

Best practice:

- set when using GNR based NSDs
- set when using large LUNs from powerful storage back ends

```
[root@beer1 gpfs]# mmfsadm dump config | grep -i prefetchPct -w -e pagepool  
prefetchPct 20  
pagepool .....
```



... if using ESS.... everything is preconfigured... ...

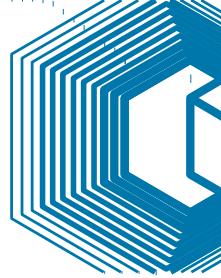


if your backend is not an ESS ...

```
[root@beer1 gpfs]# mmfsadm dump config | grep -i pagepool  
nsdBufSpace (% of PagePool) 30  
nsdRAIDBufferPoolSizePct (% of PagePool) 50
```

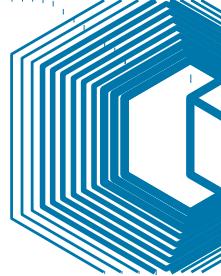
```
[root@beer1 gpfs]# mmfsadm dump config | grep -i -e worker -e smallthread | grep -i nsd[M,S]  
nsdMaxWorkerThreads 512  
nsdMinWorkerThreads 16  
nsdSmallThreadRatio 0  
[root@beer1 gpfs]#
```

Agenda



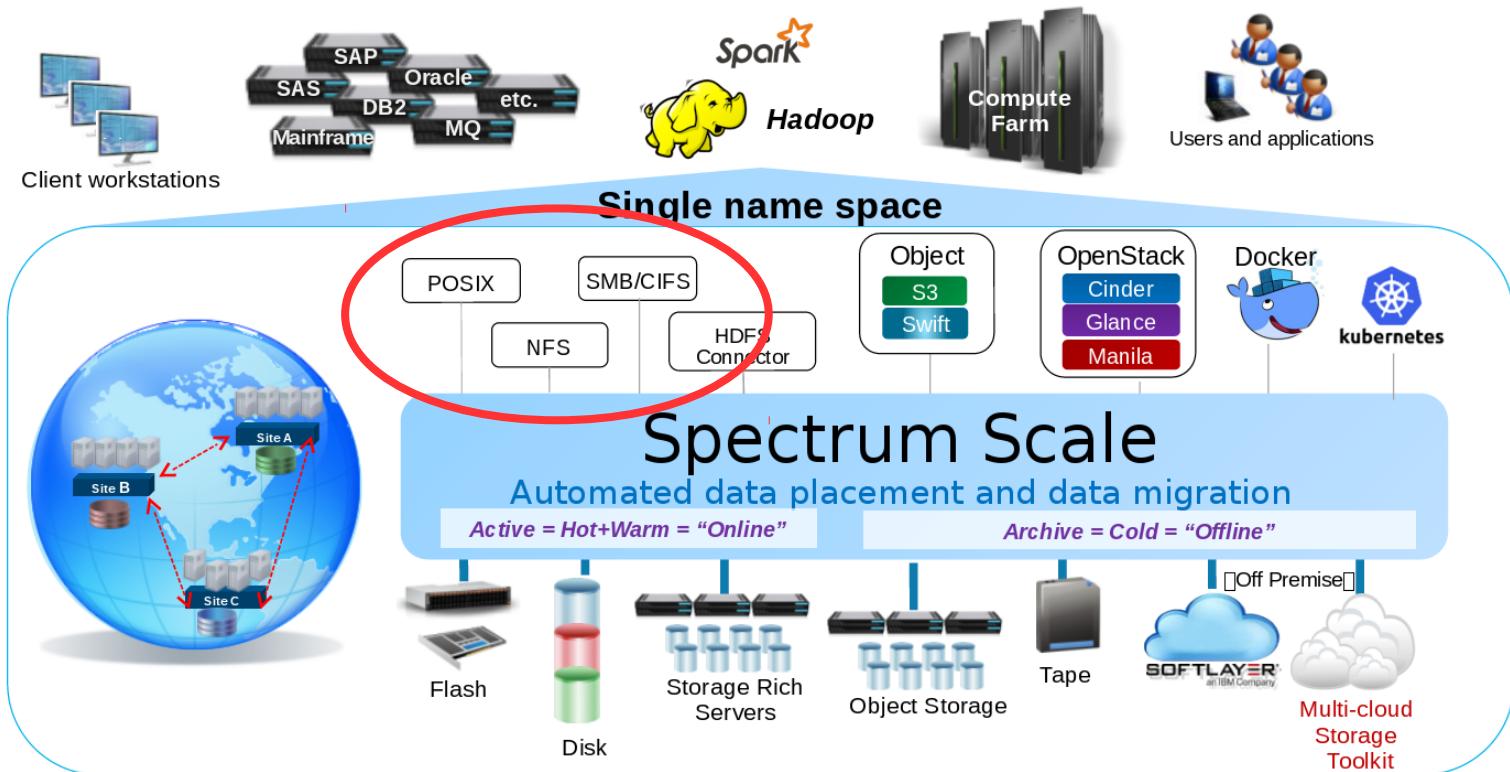
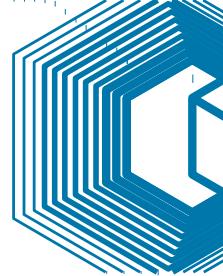
- ▶ Handling node failures
- ▶ CCR restore
- ▶ Auto tuning
- ▶ **NFSv4 ACL**
- ▶ ESS for SAP HANA workloads
- ▶ ubiquity

Spectrum Scale - NFSv4 ACLs

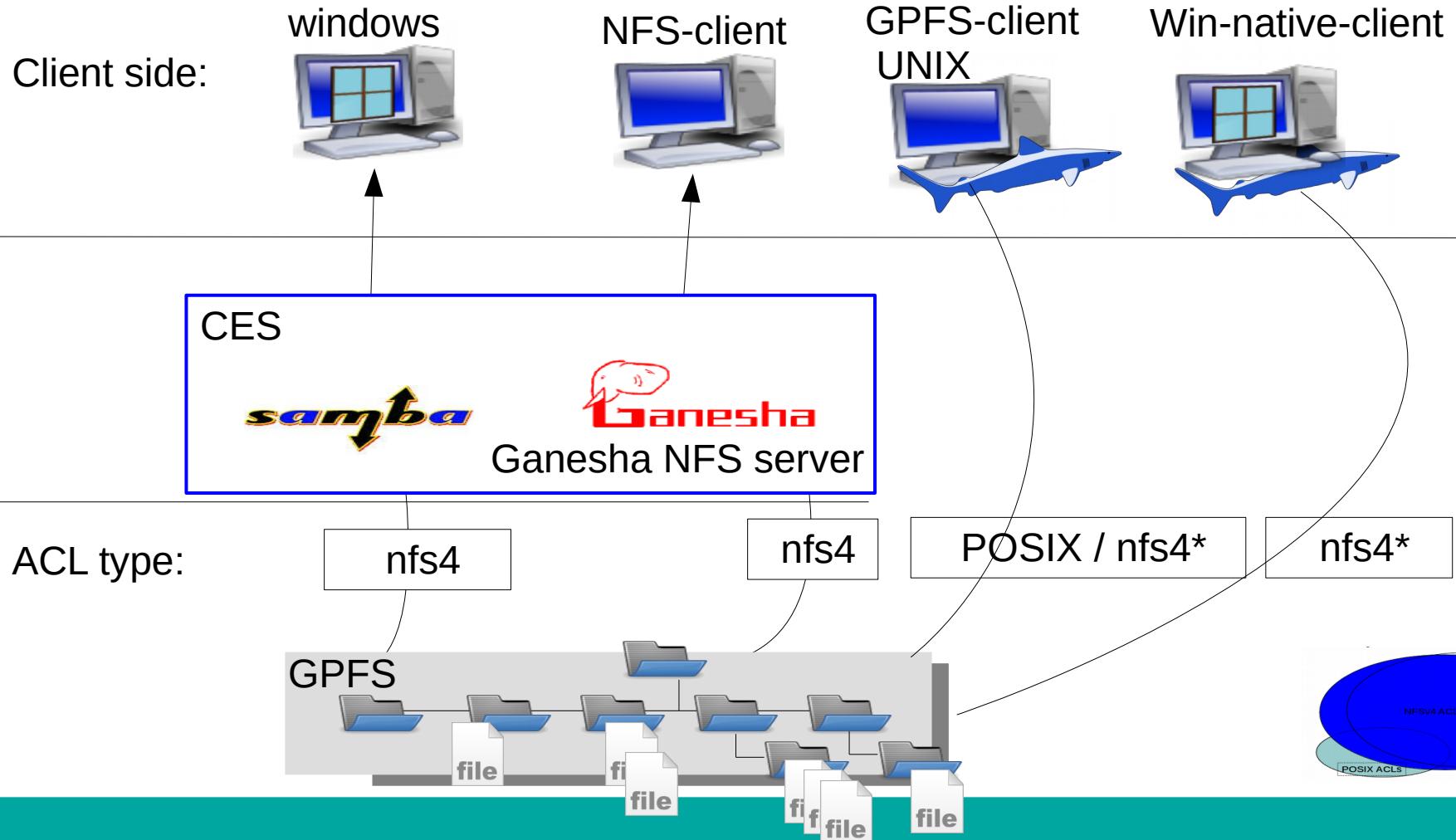


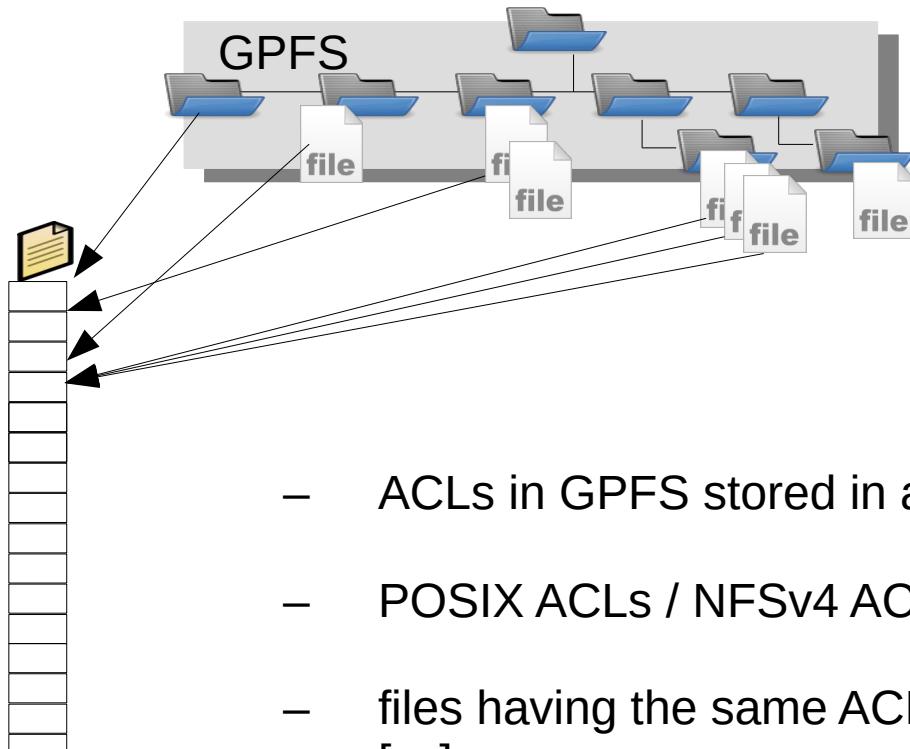
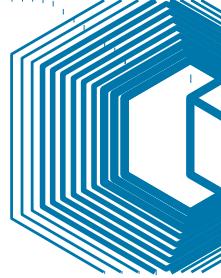
- Finer-grained control of user access for files and directories
 - better NFS security
 - improved interoperability with CIFS
 - removal of the NFS limitation of 16 groups per user
- defined in RFC3530
<http://www.ietf.org/rfc/rfc3530.txt>



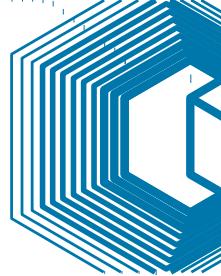


SpectrumScale – Windows and Unix client access





- ACLs in GPFS stored in a hidden file
 - POSIX ACLs / NFSv4 ACL format supported in parallel (mmlsfs -k)
 - files having the same ACL , have the same hash value
- [...]
- ```
extendedAcl 50
```
- [...]



## NFSv4 ACL – understanding special names

- NFS V4 provides for a set of special names that are not associated with a specific local UID or GID.
- representing/similar translated Unix ModeBits

- special:owner@
- special:group@
- special:everyone@

```
[root@tlinc04 fs1]# mmgetacl file1
```

```
#NFSv4 ACL
```

```
#owner:root
```

```
#group:root
```

```
special:owner@:rw-c:allow
```

```
(X)READ/LIST (X)WRITE/CREATE (X)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAME
(-)DELETE (-)DELETE_CHILD (X)CHOWN (-)EXEC/SEARCH (X)WRITE_ACL (X)WRITE_ATTR (X)WRITE_NAMED
```

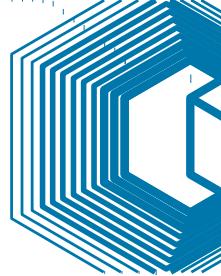
```
special:group@:r---:allow
```

```
(X)READ/LIST (-)WRITE/CREATE (-)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(-)DELETE (-)DELETE_CHILD (-)CHOWN (-)EXEC/SEARCH (-)WRITE_ACL (-)WRITE_ATTR (-)WRITE_NAMED
```

```
special:everyone@:----:allow
```

```
(-)READ/LIST (-)WRITE/CREATE (-)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(-)DELETE (-)DELETE_CHILD (-)CHOWN (-)EXEC/SEARCH (-)WRITE_ACL (-)WRITE_ATTR (-)WRITE_NAMED
```

```
[root@tlinc04 fs1]# ls -l file1
-rw-r----- 1 root root 6 Sep 9 20:34 file1
```



## NFSv4 ACL – regular entry

```
[root@beer1 fs1]# mmgetacl /x/beer/fs1/subdir1
```

```
#NFSv4 ACL
#owner:root
#group:root
special:owner@:rw-c:allow:FileInherit:DirInherit
(X)READ/LIST (X)WRITE/CREATE (X)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(-)DELETE (-)DELETE_CHILD (X)CHOWN (-)EXEC/SEARCH (X)WRITE_ACL (X)WRITE_ATTR (X)WRITE_NAMED

special:group@:r--:allow:FileInherit:DirInherit
(X)READ/LIST (-)WRITE/CREATE (-)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(-)DELETE (-)DELETE_CHILD (-)CHOWN (-)EXEC/SEARCH (-)WRITE_ACL (-)WRITE_ATTR (-)WRITE_NAMED

special:everyone@:r--:allow:FileInherit:DirInherit
(X)READ/LIST (-)WRITE/CREATE (-)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(-)DELETE (-)DELETE_CHILD (-)CHOWN (-)EXEC/SEARCH (-)WRITE_ACL (-)WRITE_ATTR (-)WRITE_NAMED
```

```
user:laff:rwx:allow:FileInherit:DirInherit
```

```
(X)READ/LIST (X)WRITE/CREATE (X)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(X)DELETE (X)DELETE_CHILD (X)CHOWN (X)EXEC/SEARCH (X)WRITE_ACL (X)WRITE_ATTR (X)WRITE_NAMED
```

```
[root@beer1 fs1]# touch /x/beer/fs1/subdir1/file.laff.from.root
```

```
[root@beer1 fs1]# ls -l /x/beer/fs1/subdir1/file.laff.from.root
```

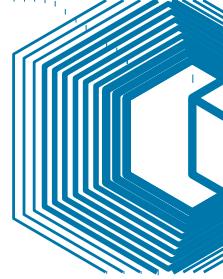
```
-rw-r--r--. 1 root root 0 Mar 7 15:47 /x/beer/fs1/subdir1/file.laff.from.root
```

```
[root@beer1 fs1]# su - laff -c "echo \" hallo \" >> /x/beer/fs1/subdir1/file.laff.from.root "
```

```
[root@beer1 fs1]# cat /x/beer/fs1/subdir1/file.laff.from.root
```

```
hallo
```

# Spectrum Scale – handling ACLs ( 1 / 3 )



(1) by default chmod overwrites NFSv4 ACLs

```
[root@beer1 fs1]# chmod g+w subdir1
```

```
[root@beer1 fs1]# mmgetacl /x/beer/fs1/subdir1
```

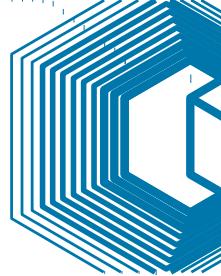
```
#NFSv4 ACL
#owner:root
#group:root
special:owner@:rw-c:allow
(X)READ/LIST (X)WRITE/CREATE (X)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(-)DELETE (X)DELETE_CHILD (X)CHOWN (-)EXEC/SEARCH (X)WRITE_ACL (X)WRITE_ATTR (X)WRITE_NAMED

special:group@:rw--:allow
(X)READ/LIST (X)WRITE/CREATE (X)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(-)DELETE (X)DELETE_CHILD (-)CHOWN (-)EXEC/SEARCH (-)WRITE_ACL (-)WRITE_ATTR (-)WRITE_NAMED

special:everyone@:r---:allow
(X)READ/LIST (-)WRITE/CREATE (-)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAMED
(-)DELETE (-)DELETE_CHILD (-)CHOWN (-)EXEC/SEARCH (-)WRITE_ACL (-)WRITE_ATTR (-)WRITE_NAMED
```

```
[root@beer1 fs1]#
```

```
[root@beer1 fs1]# ll /x/beer/fs1
total 0
drw-rw-r--. 4 root root 4096 Mar 7 15:47 subdir1
[root@beer1 fs1]# █
```



(2) old way: ( older releases ... )

mmlsconfig

[...]

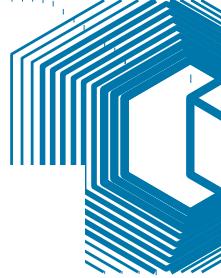
**AllowDeleteAclOnChmod 1**

[...]

→ enables

obsolete

accept / reject chmod on files with NFSv4



## Spectrum Scale – handling ACLs ( 3 /3 )

- since current R 4.x
- supports fileset level permission change

```
--allow-permission-change PermissionChangeMode
 Specifies the new permission change mode. This mode
 controls how chmod and ACL operations are handled on
 objects in the fileset. Valid modes are as follows:

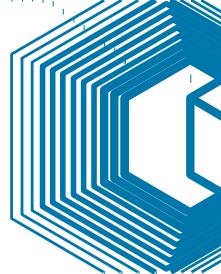
 chmodOnly
 Specifies that only the UNIX change mode
 operation (chmod) is allowed to change access
 permissions (ACL commands and API will not be
 accepted).

 setAclOnly
 Specifies that permissions can be changed using
 ACL commands and API only (chmod will not be
 accepted).

 chmodAndSetAcl
 Specifies that chmod and ACL operations are
 permitted. If the chmod command (or setattr
 file operation) is issued, the result depends
 on the type of ACL that was previously
 controlling access to the object:
 * If the object had a Posix ACL, it will be
 modified accordingly.
 * If the object had an NFSv4 ACL, it will be
 replaced by the given UNIX mode bits.

 Note: This is the default setting when a
 fileset is created.

 chmodAndUpdateAcl
 Specifies that chmod and ACL operations are
 permitted. If chmod is issued, the ACL will be
 updated by privileges derived from UNIX mode
 bits.
```



## Spectrum Scale – NFSv4 example: allow permission change

```
[root@beer1 fs1]# mmputacl -i /tmp/acl subdir1
```

```
[root@beer1 fs1]# mmgetacl subdir1
```

```
[...]
```

```
user:laff:rwx:allow:FileInherit:DirInherit
```

```
(X)READ/LIST (X)WRITE/CREATE (X)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)READ_NAME
(X)DELETE (X)DELETE_CHILD (X)CHOWN (X)EXEC/SEARCH (X)WRITE_ACL (X)WRITE_ATTR (X)WRITE_NAME[
```

```
[root@beer1 fs1]# mmchfileset beer fs1 --allow-permission-change chmodAndUpdateAcl
```

```
Fileset fs1 changed.
```

```
[root@beer1 fs1]#
```

```
[root@beer1 fs1]# chmod g+w subdir1
```

```
:root@beer1 fs1]# ls -l
:total 0
:rw-rw-r--. 2 root root 6 Mar 8 00:07 subdir1
root@beer1 fs1]# █
```

```
[root@beer1 fs1]# mmgetacl subdir1
```

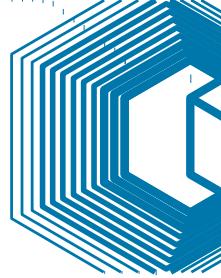
```
[...]
```

```
user:laff:rwx:allow:FileInherit:DirInherit
```

```
(X)READ/LIST (X)WRITE/CREATE (X)APPEND/MKDIR (X)SYNCHRONIZE (X)READ_ACL (X)READ_ATTR (X)RE
(X)DELETE (X)DELETE_CHILD (X)CHOWN (X)EXEC/SEARCH (X)WRITE_ACL (X)WRITE_ATTR (X)WRITE
```

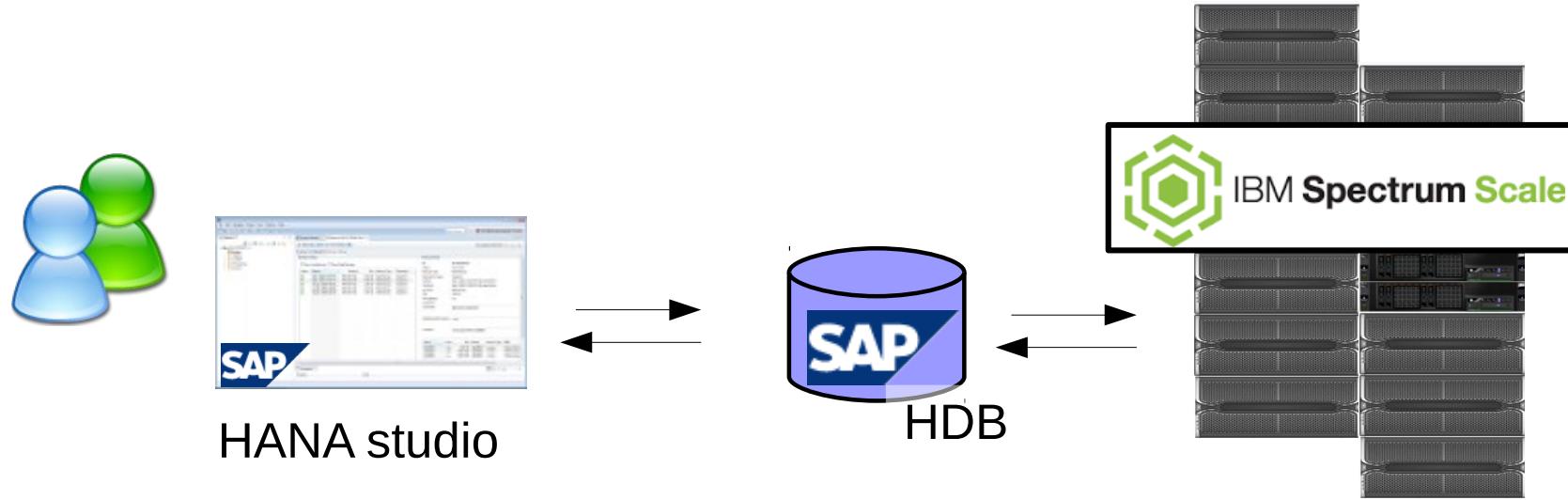
# Agenda

---



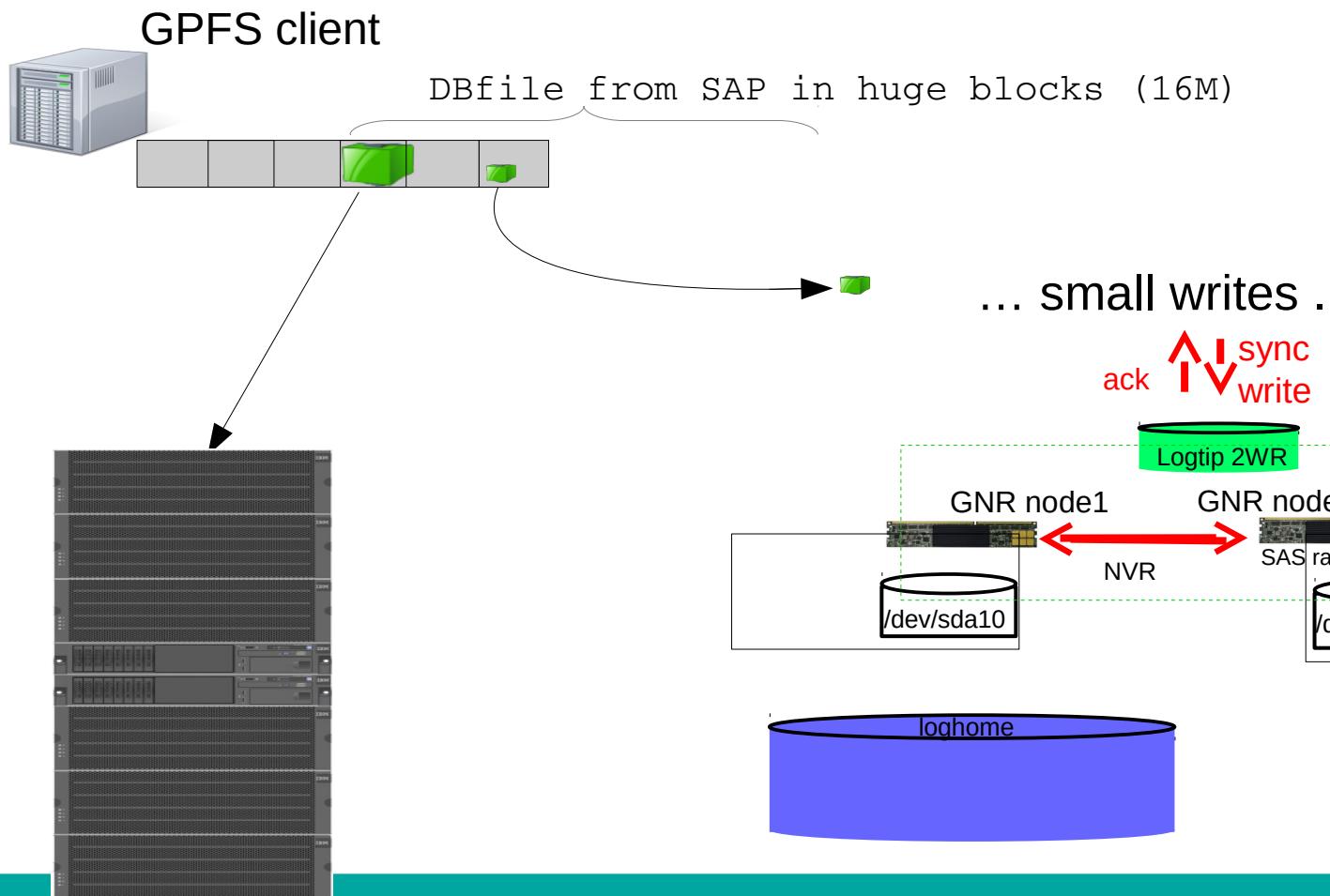
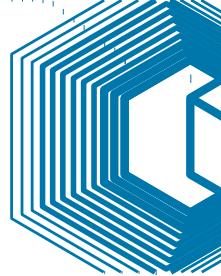
- ▶ Handling node failures
- ▶ CCR restore
- ▶ Auto tuning
- ▶ NFSv4 ACL
- ▶ **ESS for SAP HANA workloads**
- ▶ ubiquity

# SpectrumScale & ESS for SAP HANA

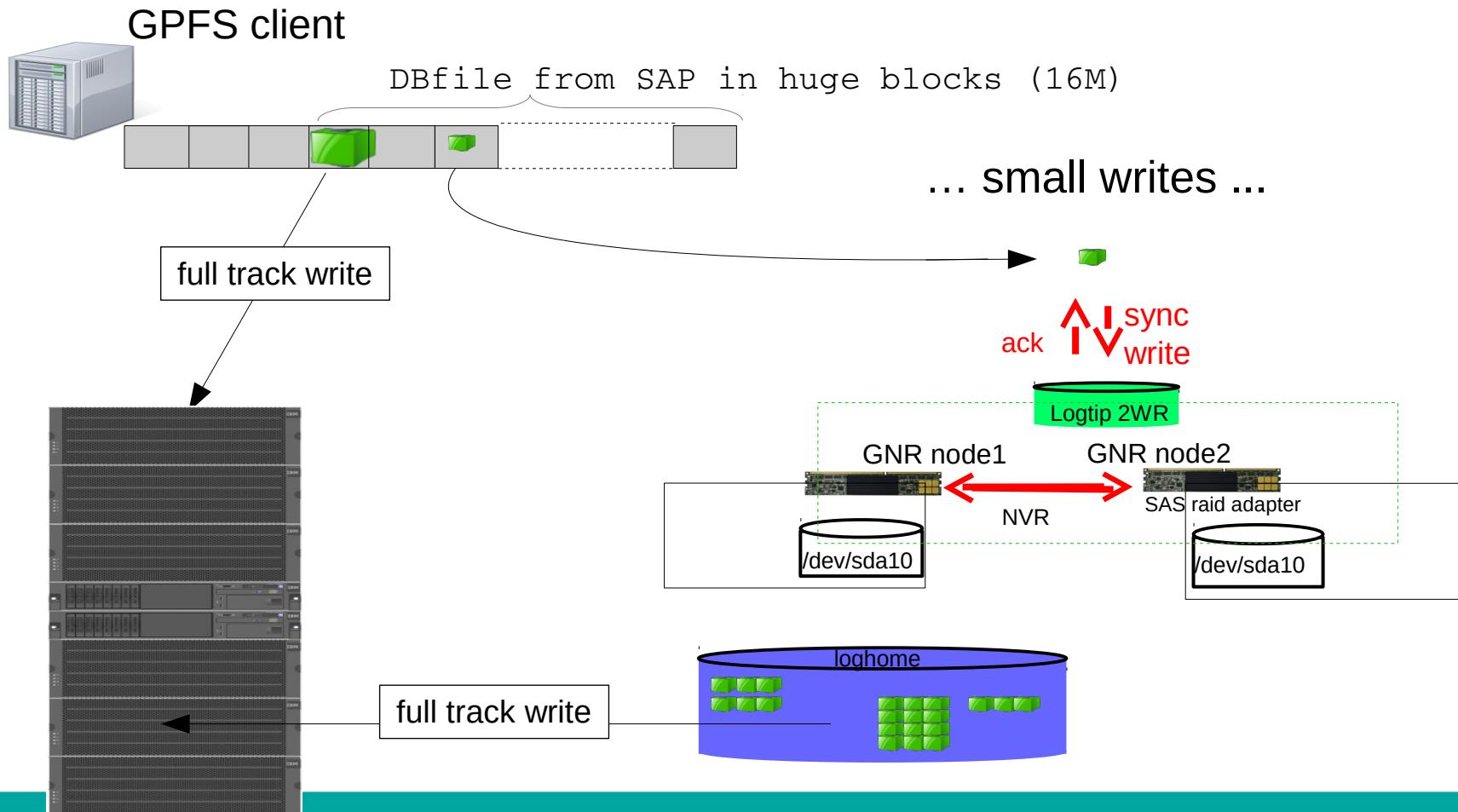


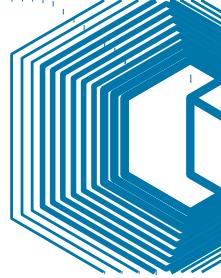
- 1.) all benefits from SpectrumScale (replication, filesets , management, provisioning)
- 2.) e2e integration SpectrumScale snapshots in HANA studio
- 3.) HANA DB workload – DIO intensive write workload

# SpectrumScale & ESS for SAP HANA



# SpectrumScale & ESS for SAP HANA





```
mmchconfig disableDIO=yes,aioSyncDelay=10 -N hananode
```

## dioReentryThreshold

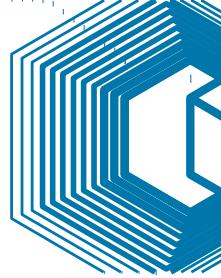
- performance optimization, when writing a new file sequentially with DIO
- once a block is allocated, next set of writes can be executed as DIO until end of block
- drop out of DIO into buffered to allocate next block generates a non trivial overhead
- better performance, when staying in buffered mode
- **dioReentryThreshold=*n*** means... wait until *n* blocks worth of I/Os  
(that could have been executed as DIO before we actually switch back to DIO mode)

## disableDIO

- DIO is always just a hint
- acc. POSIX , DIO versus O\_SYNC in addition of O\_DIRECT
- if set, GPFS will execute all DIO requests as buffered I/O
- this parameter does not cheat anything

# Agenda

---



- ▶ Handling node failures
- ▶ CCR restore
- ▶ Auto tuning
- ▶ NFSv4 ACL
- ▶ ESS for SAP HANA workloads
- ▶ **ubiquity**

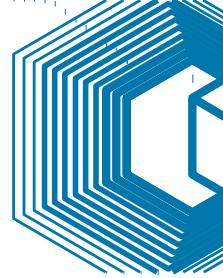


# ***Ubiquity Container Volume Service***

***Unified File Service Across  
CloudFoundry, Docker, Kubernetes, and More!***

Sandeep Gopisetty, Heiko Ludwig, Nagapramod  
Mandagere, Mohamed Mohamed, Robert Engel  
Cloud Systems Research – Almaden

Dean Hildebrand, Amit Warke  
Cloud Storage Software Research – Almaden



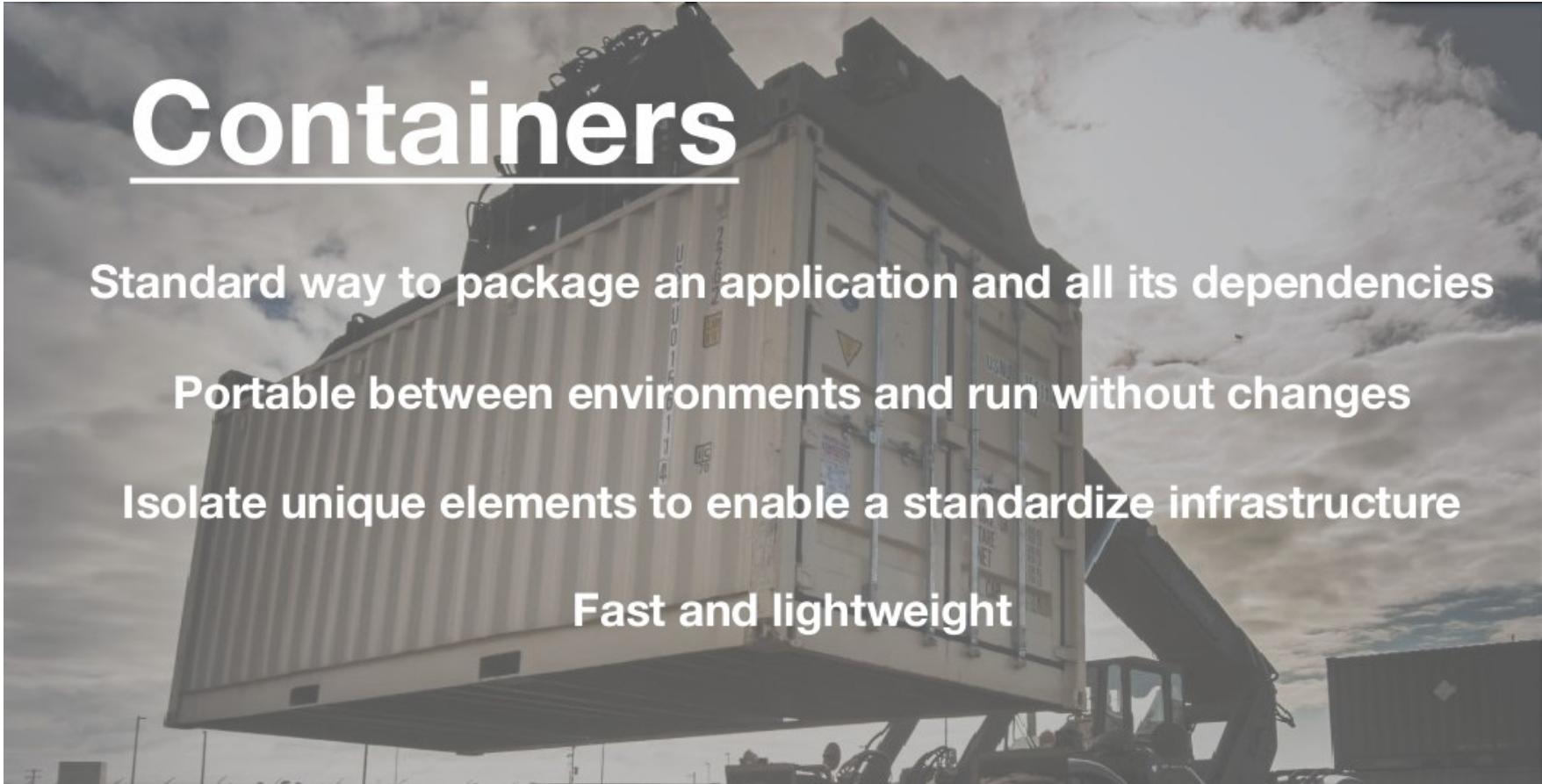
# Containers

**Standard way to package an application and all its dependencies**

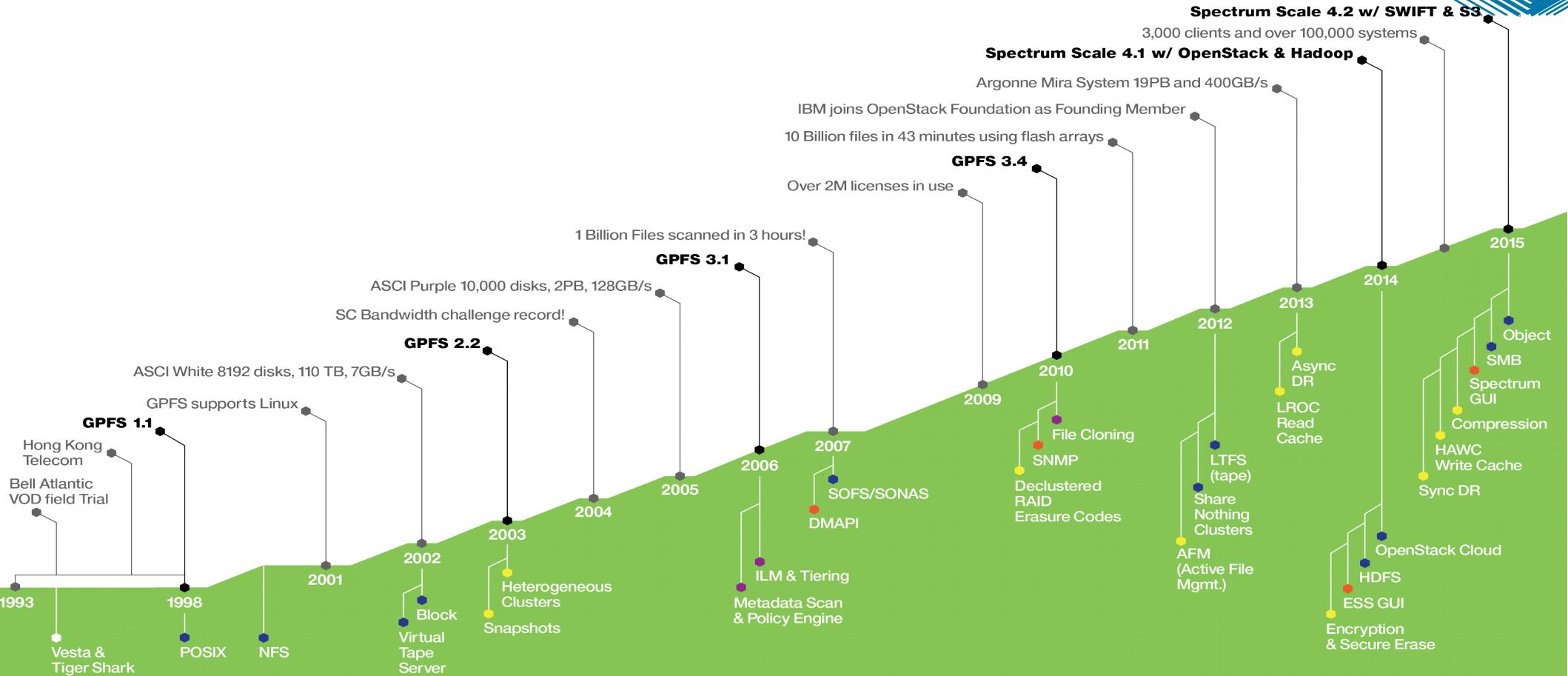
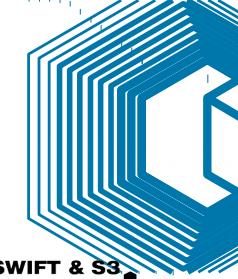
**Portable between environments and run without changes**

**Isolate unique elements to enable a standardize infrastructure**

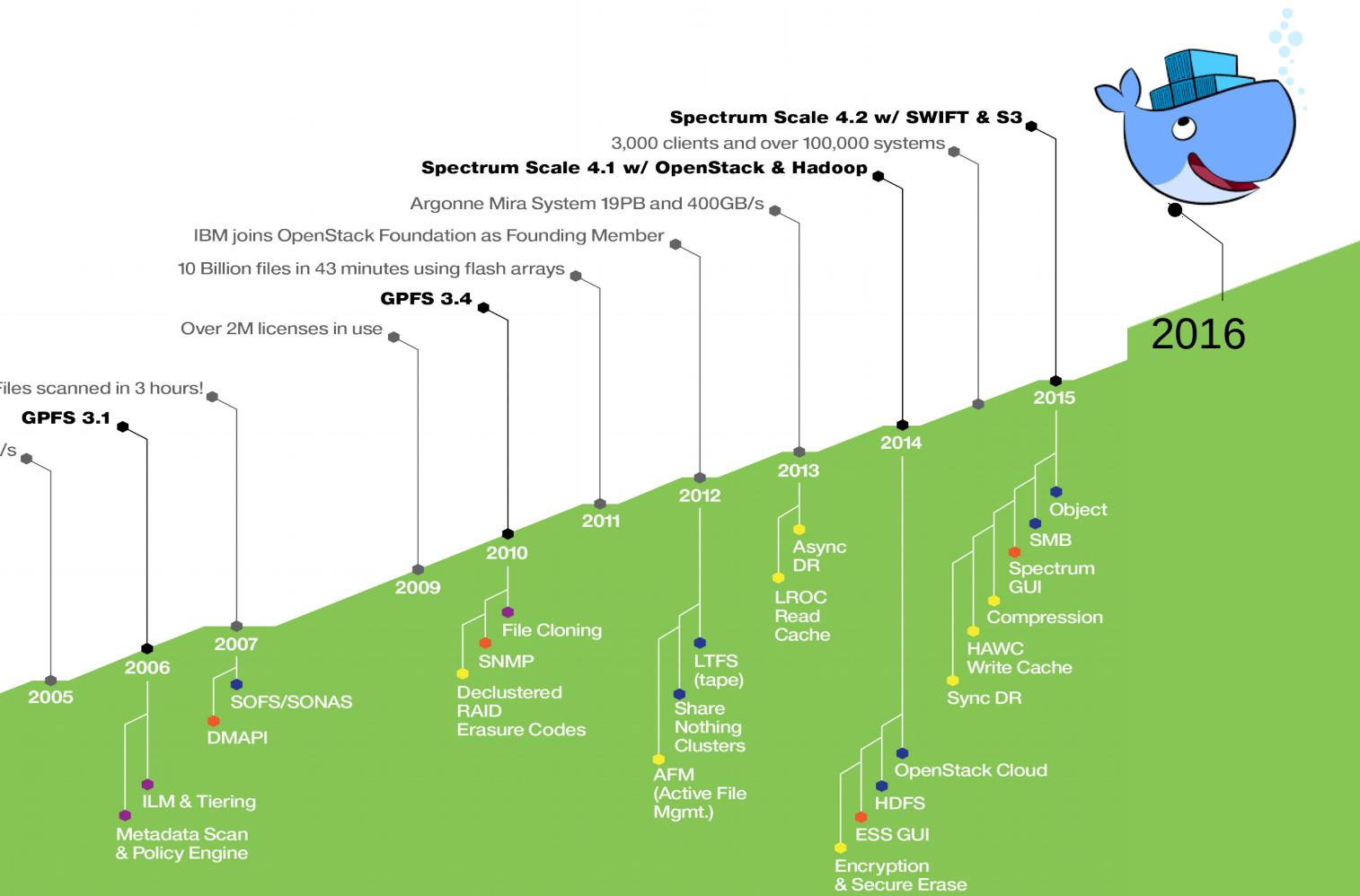
**Fast and lightweight**



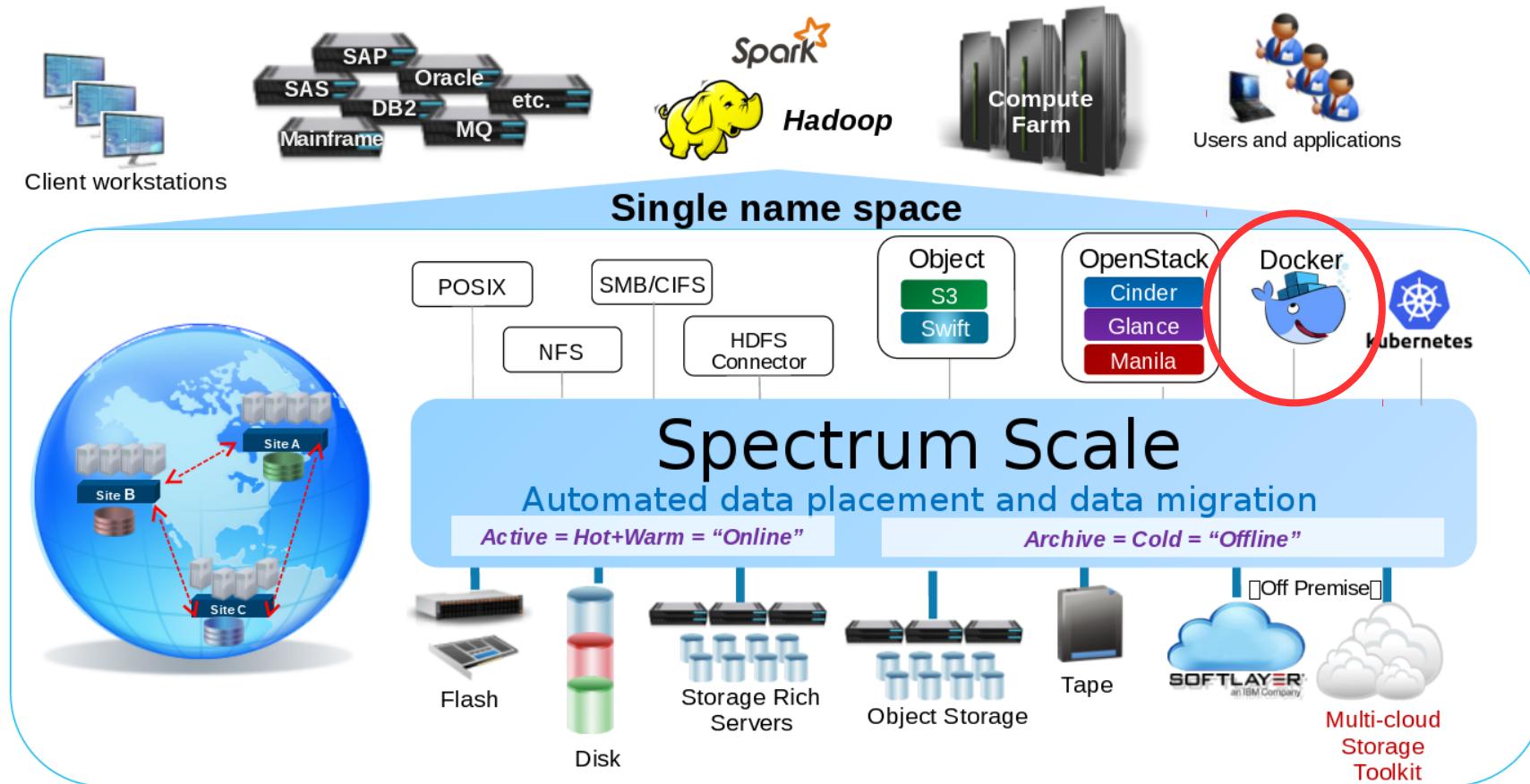
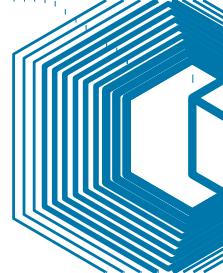
# SpectrumScale and Ubiquity – added in 2016 / 2017

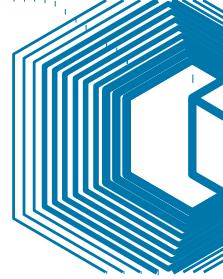


SpectrumScale and Ubiquity – added in 2016 / 2017



# SpectrumScale and Ubiquity – added in 2016 / 2017





- Decoupled from SpectrumScale release
- Published / available on github
- Planned: open source

The screenshot shows the GitHub organization page for 'almaden-containers'. The page features a large teal 'H' icon, the organization's name 'almaden-containers', and three repository cards: 'ubiquity-docker-plugin', 'ubiquity', and 'ubiquity-k8s'. Each card includes a brief description, a star rating, and a 'Go' button. A sidebar on the right lists organization members.

https://github.ibm.com/almaden-containers

This organization Search Pull requests Issues Gist

almaden-containers

Repositories People 0

Filters Find a repository...

**ubiquity-docker-plugin**  
Ubiquity plugin for docker/swarm  
Updated 4 days ago

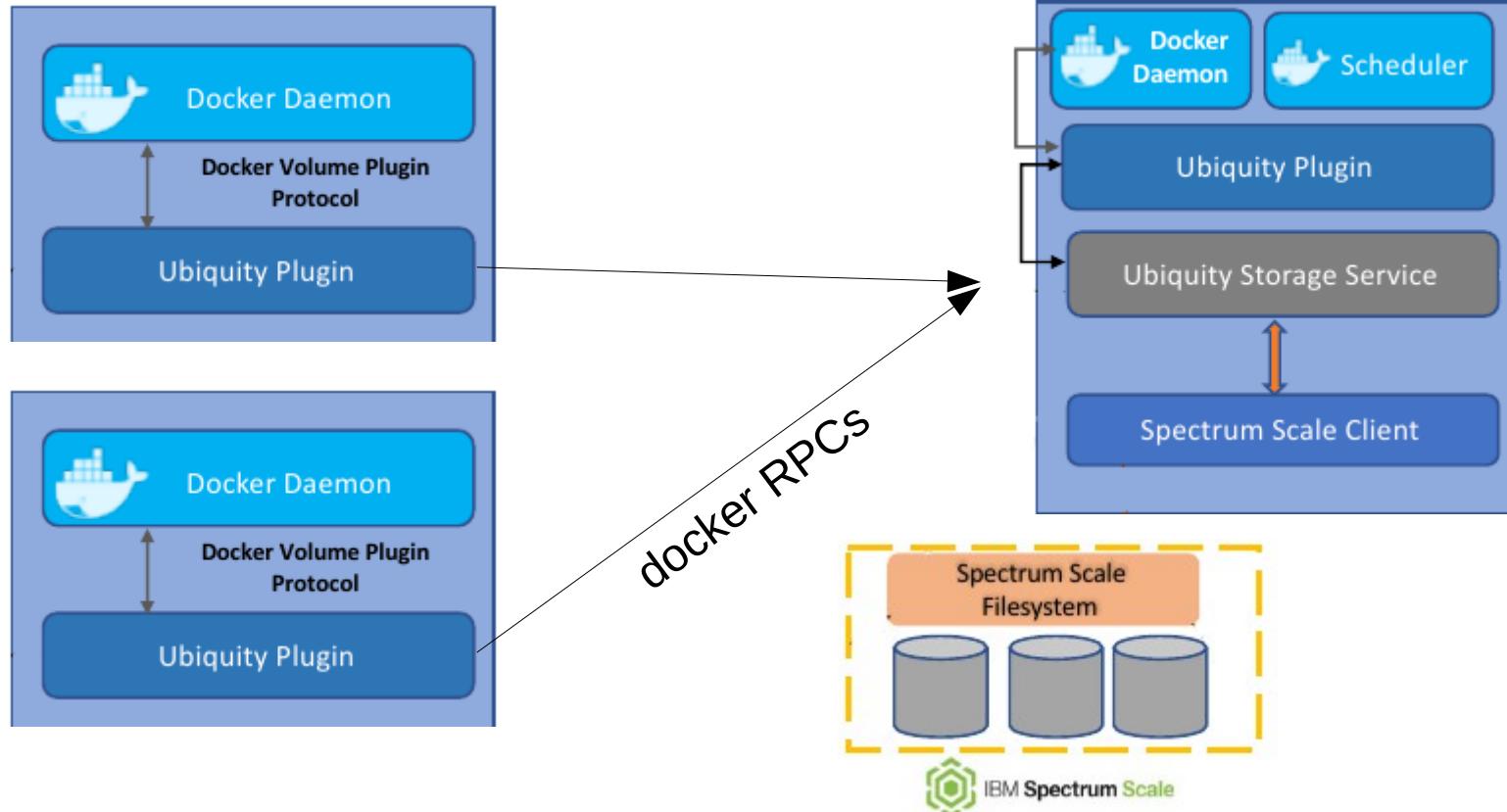
**ubiquity**  
Storage Service for container runtimes  
Updated 4 days ago

**ubiquity-k8s**  
Updated 5 days ago

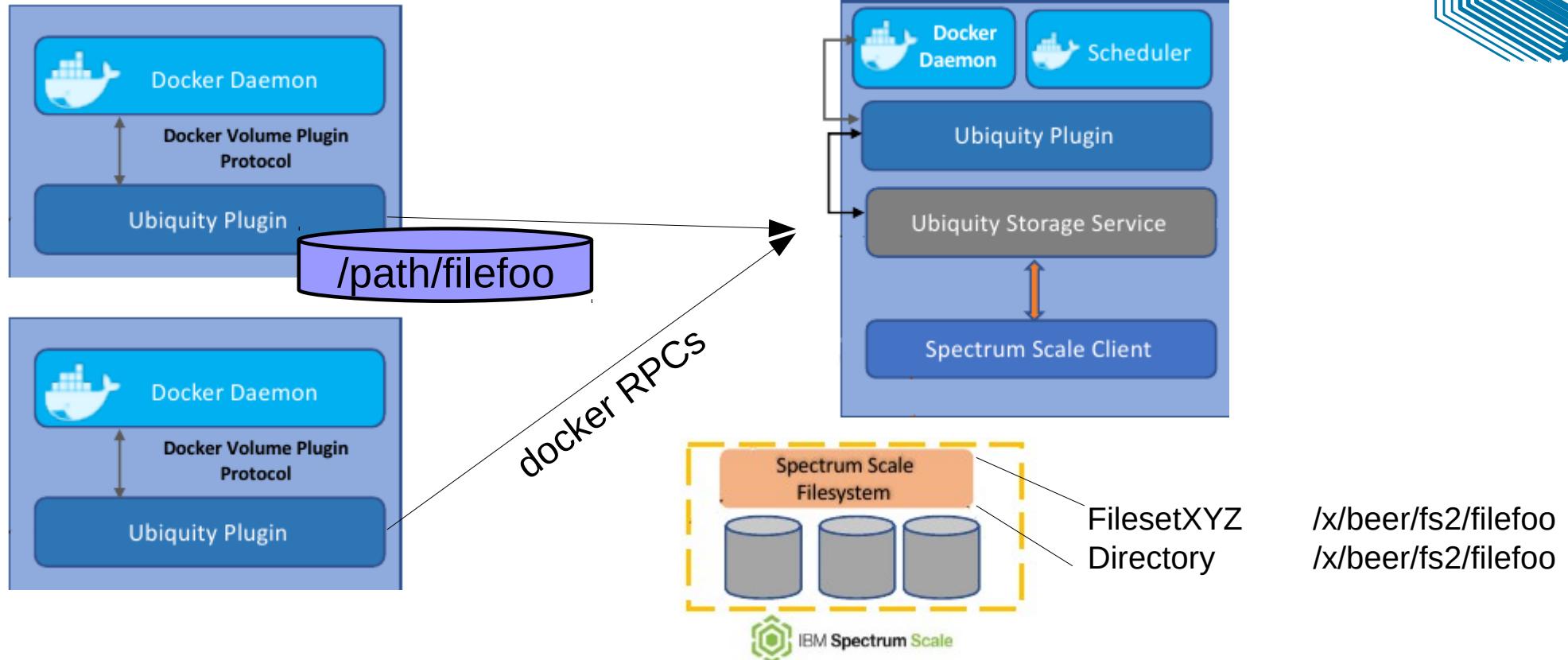
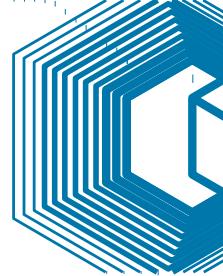
People

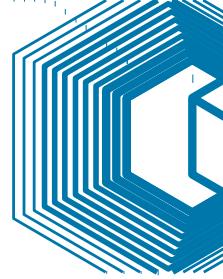
This organization members. You see who's a part of it

# Docker and SpectrumScale integration



# Docker and SpectrumScale integration





Ubiquity Storage Service

- creating fileset volumes
- light weight volumes

## File set volumes

```
docker volume create -d spectrum-scale --name demo1 --opt filesystem=beer
```

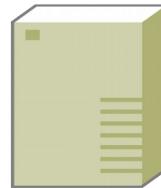
## Light wight volumes

```
docker volume create -d spectrum-scale --name demo5 --opt type=lightweight
--opt fileset=LtWtVolFileset --opt filesystem=beer
```

# Docker - example



## SpectrumScale's view



```
[root@beer1 system]# mmfsfileset beer
Filesets in file system 'beer':
Name Status Path
root Linked /beer
demo1 Linked /beer/demo1
demo2 Linked /beer/demo2
demo3 Linked /beer/demo3
demo6 Linked /beer/demo6
demo8 Linked /beer/demo8
```

## container's (dockers) view

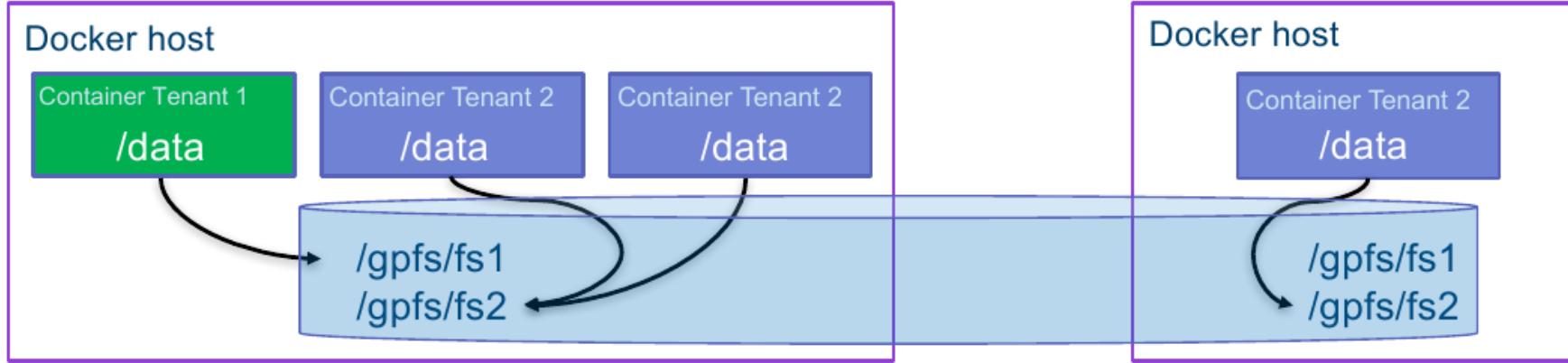
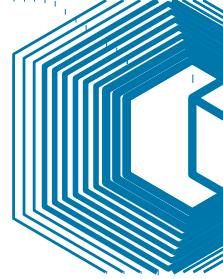
```
[root@beer1 system]# docker volume ls
DRIVER VOLUME NAME
spectrum-scale demo1
spectrum-scale demo3
spectrum-scale demo8
```

```
[root@beer1 tmp]# docker run -t -i --volume-driver spectrum-scale
--volume demo1:/container/inside/beer --entrypoint /bin/sh alpine
```

```
/ # df -h | grep container
beer 10.0G 478.0M 9.5G 5% /container/inside/beer
/ #
```

```
/ # ls -l /container/inside/beer
total 0
-rw-r--r-- 1 root root 0 Feb 23 12:25 bernd_das_brot
-rw-r--r-- 1 root root 12 Feb 22 09:04 create_new_file_mittwoch
-rw-r--r-- 1 root root 0 Feb 20 20:13 created_afterwards
-rw-r--r-- 1 root root 0 Feb 18 10:35 created_inside
-rw-r--r-- 1 root root 29 Feb 18 10:36 created_outside
-rw-r--r-- 1 root root 29 Feb 18 10:36 created_outside
-rw-r--r-- 1 root root 14 Feb 21 15:17 neues_file_outside
-rw-r--r-- 1 root root 0 Feb 21 15:35 test
/ #
```

```
[root@beer1 system]# ll /beer/demo1
total 0
-rw-r--r-- 1 root root 0 Feb 23 13:25 bernd_das_brot
-rw-r--r-- 1 root root 0 Feb 20 21:13 created_afterwards
-rw-r--r-- 1 root root 0 Feb 18 11:35 created_inside
-rw-r--r-- 1 root root 29 Feb 18 11:36 created_outside
-rw-r--r-- 1 root root 12 Feb 22 10:04 create_new_file_mittwoch
-rw-r--r-- 1 root root 14 Feb 21 16:17 neues_file_outside
-rw-r--r-- 1 root root 0 Feb 21 16:35 test
[root@beer1 system]#
```



## Details

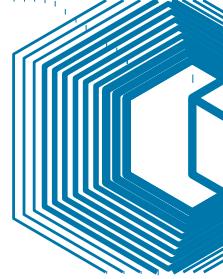
- Single or separate userid namespaces between containers and hosts
- Data sharing across containers and hosts
- All POSIX commands supported from container

## A Few Benefits

- Multi-tenant access
  - Container can only access its volumes
  - Allow root access in container without allowing root access to file system
  - ACLs can add an extra level of security
- Native client performance

# Agenda

---



best practice

some news



olaf.weiser@de.ibm.com

IBM Deutschland

SpectrumScale Support Specialist