Spectrum Scale Expert Talks

Episode 2:

Best Practices for building a stretched-cluster

IBM Spectrum Scale

Show notes:
www.spectrumscaleug.org/experttalks

Join our conversation:
www.spectrumscaleug.org/join

About the user group

- Independent, work with IBM to develop events
- Not a replacement for PMR!
- Email and Slack community
- www.spectrumscaleug.org/join

IBMCHAMPION

#SSUG

Agenda

- What is a Spectrum Scale stretched cluster?
- Components of a stretched cluster
  - Quorum concerns with stretched clusters
  - File system replication with failure groups
  - File system descriptor quorum
- Bringing this all together
- Some final considerations

Introduction

… or, what is a stretched cluster and why would I want one?

Spectrum Scale business resiliency through replicated redundancy

	Active/Passive	Active/Active
Application layer	• Individual applications may have their own means of doing asynchronous replication. • Aspera, `rsync` – These need to run regularly (maybe use `cron`)	Individual applications may have their own means of doing synchronous replication.
File system layer	Spectrum Scale AFM-DR	Spectrum Scale failure groups and replication – “stretched cluster”
Block layer	• Block synchronous replication, configured for active/passive • Block point-in-time copy	Block synchronous replication, configured for active/active

What is a Spectrum Scale "stretched cluster"?

This is a *single* Spectrum Scale cluster is configured using nodes and storage from two data centers.

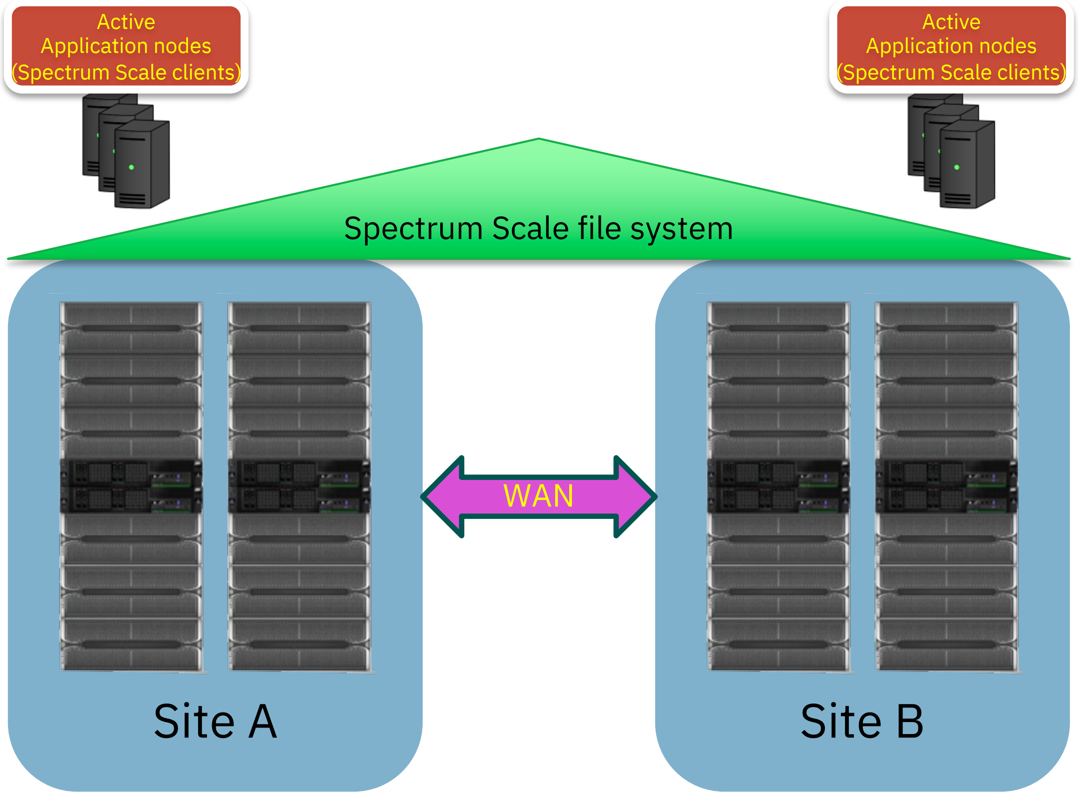
- In other words, it is "stretched" between two sites, connected by a WAN.

File systems in such a cluster are available to systems at both sites and may be actively used concurrently by both sites.

File systems may judiciously use "failure group" replication to ensure both sites have a current instance of all the data.

Careful design can ensure one site remains active, even if the other site (or link) fails.

The result is an **active/active highly available** synchronously replicated Spectrum Scale file system.



How are stretched file systems being used today?

Financial services

- Major hedge fund for financial HPC storage (30-40 miles between sites)

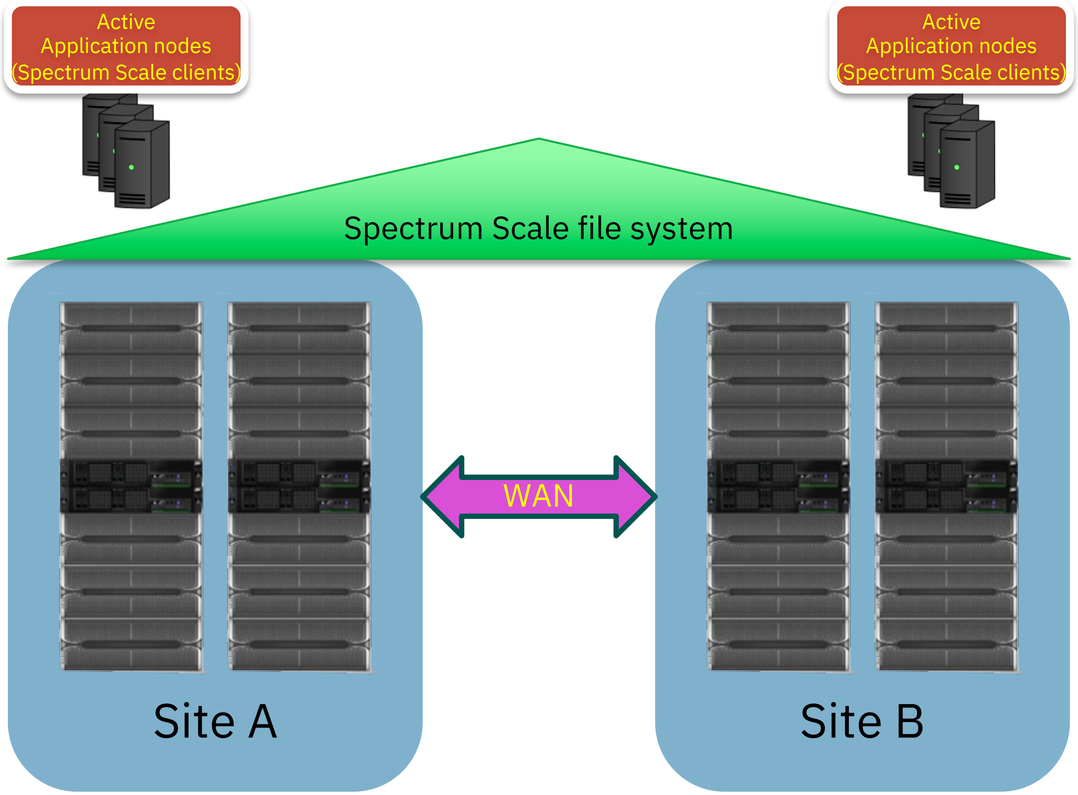
Automotive

- Major manufacturer using a stretched cluster for HPC storage (about 40 miles)

Life sciences

- Major hospitals using stretched clusters for critical patient documents.

How far? About 50-100 miles, but it depends very much on the WAN link and the workload…



Stretched clusters

… and why node quorum matters

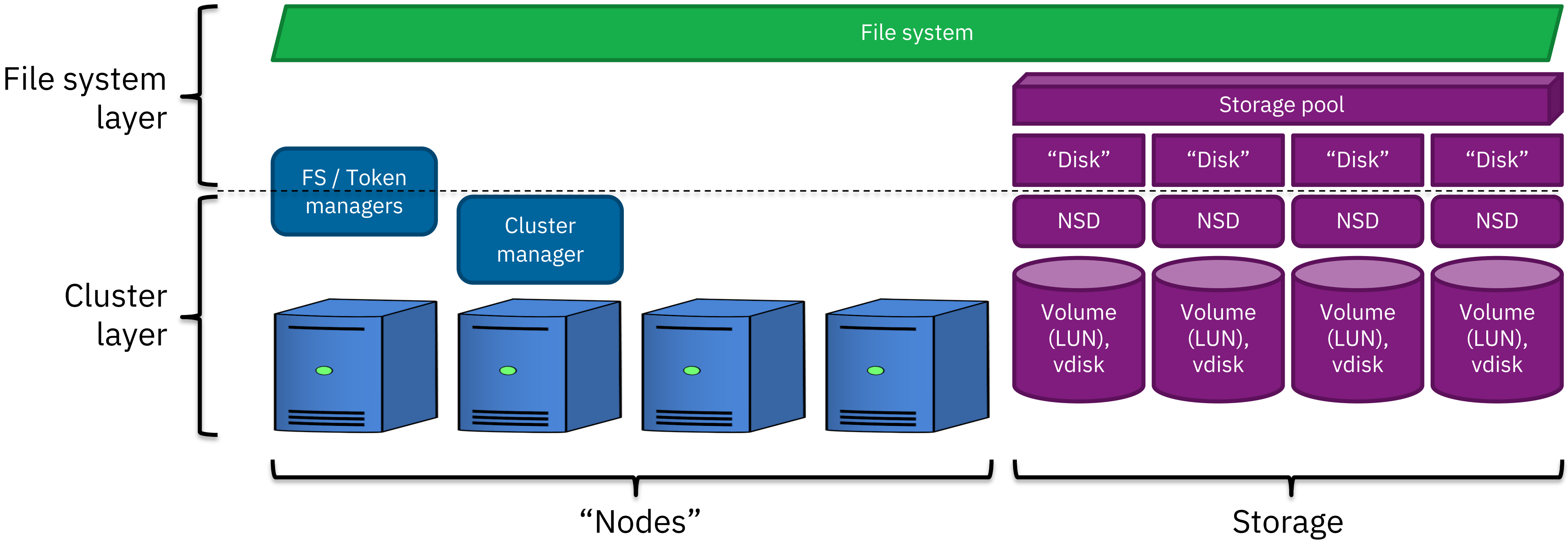
Cluster and file system layers in Spectrum Scale

Layers of Spectrum Scale

The **cluster layer** is the physical resources (systems, volumes) and logical abstractions (nodes, NSDs).

The **file system layer** creates the namespaces and manages their associated storage abstractions.

Stretched clusters enable stretched file systems.

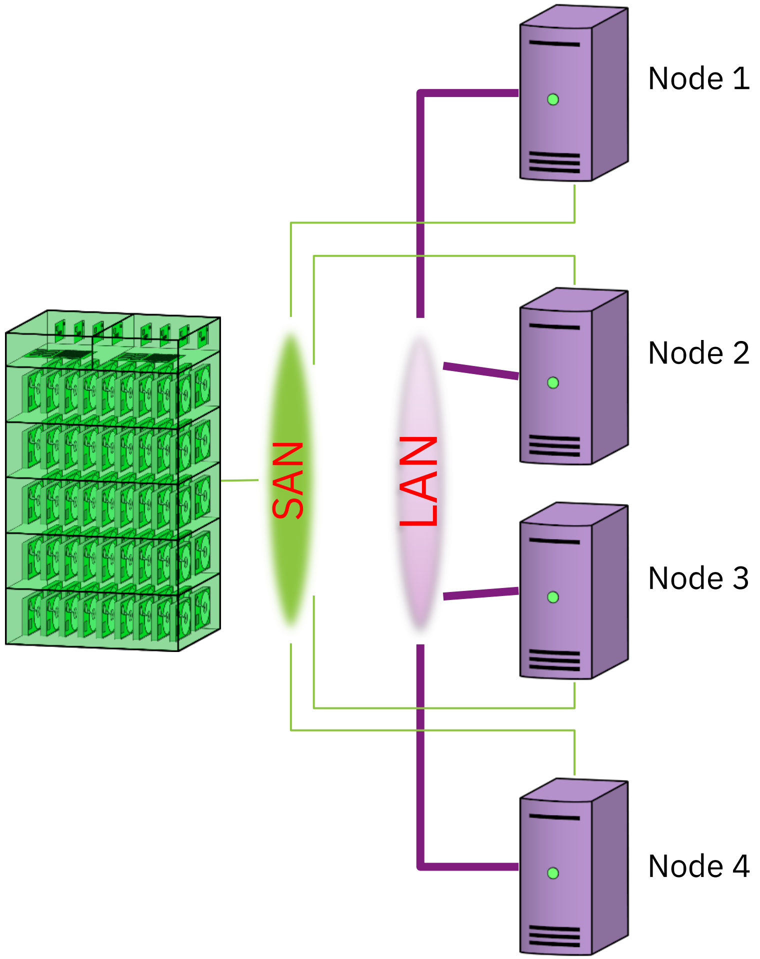


Spectrum Scale clusters

A Spectrum Scale **cluster** is a group of Spectrum Scale systems, or **nodes**, configured into a single administrative grouping:

- All nodes have a common view of the data.
- The nodes are tightly coupled, trusting each others' authentication of users.
- A cluster can have several Spectrum Scale file systems.
- A cluster may share a file system with an authenticated remote cluster.

Importantly, the cluster ensures all nodes in the cluster have a *consistent* view of the Spectrum Scale file systems, even when more than one node is actively accessing a file system (or even the same file).



Cluster manager and the active cluster

Many nodes may be **configured** to be part of the cluster.

The **active cluster** is the set of nodes currently communicating with each other and sharing resources.

A **cluster manager** keeps track of which nodes are currently part of the active cluster, controls access to managed cluster resources, and maintains the configuration of the cluster.

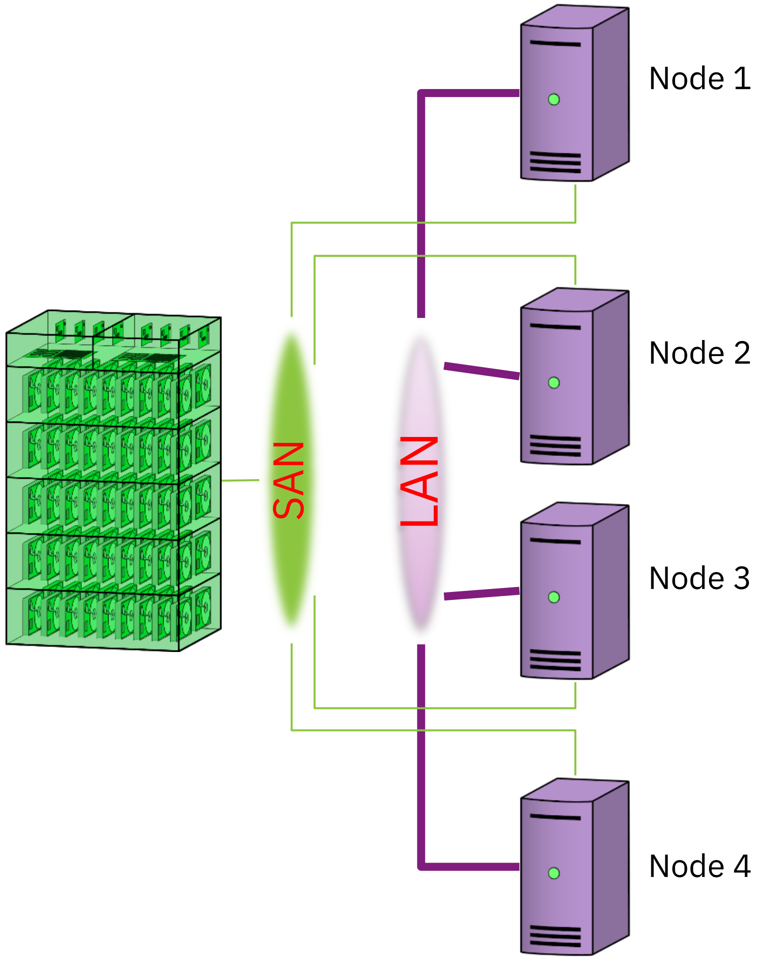
To avoid the cluster manager from being a single point of failure, this is a **role** that may run on one of several cluster nodes.

Because failures happen, the cluster manager may need to **expel** “dead” nodes and **fence** them from resources like shared disks:

- Perhaps a node has failed
- Perhaps a node’s network has failed

Generally a **disk lease** is used to grant a node access to the disk volumes (NSDs) managed by the cluster.

Disk lease renewal also functions as a **heartbeat**, so the cluster manager knows which nodes are part of the active cluster.



Node quorum

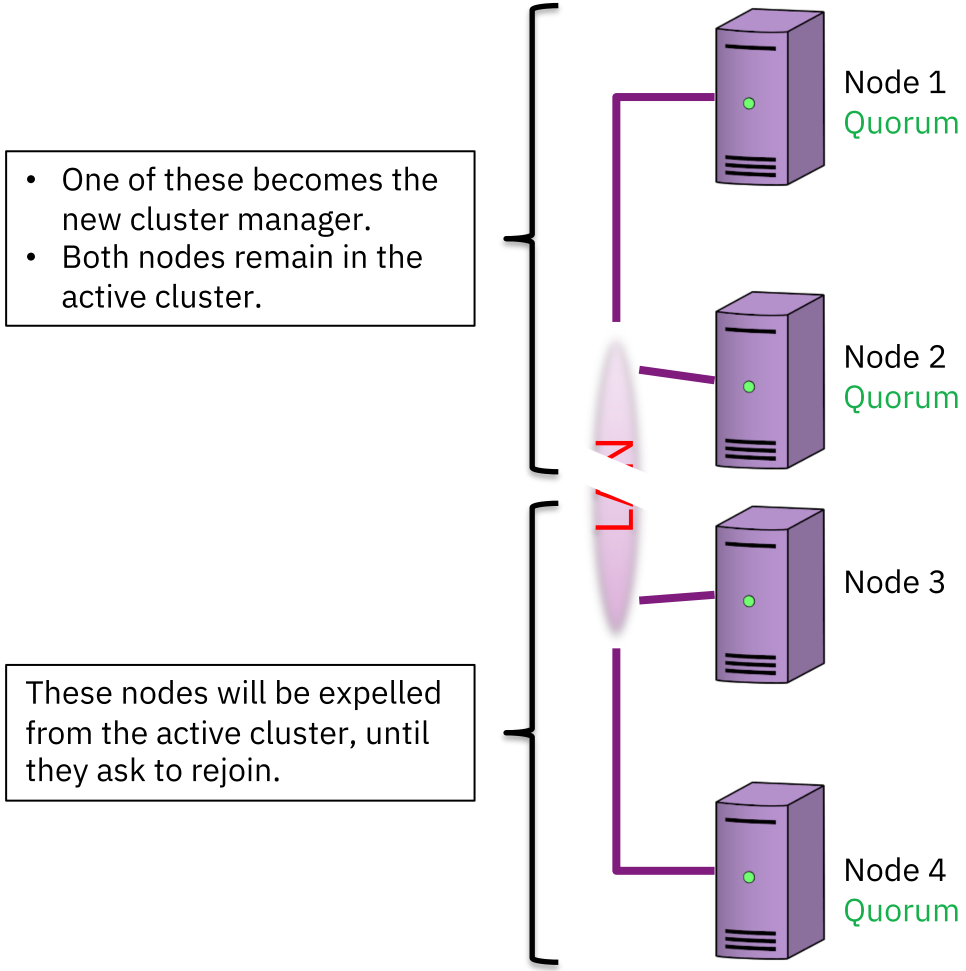
Several nodes are designated as **quorum nodes**, which elect the cluster manager from amongst themselves.

With **node quorum**, a simple majority of the quorum nodes must be active and communicating to choose a cluster manager.

- Generally choose a small odd number of quorum nodes, like 3, 5 or possibly 7. More quorum nodes lengthens recovery time – there is no benefit to choosing more than 7.

To keep the file system consistent and prevent data loss, there must never be more than one cluster manager!

- Such a condition would be called "split-brain", also known as disaster.



- One of these becomes the new cluster manager.
- Both nodes remain in the active cluster.

These nodes will be expelled from the active cluster, until they ask to rejoin.

IBM Spectrum Scale

Checking and changing quorum status

To see which nodes in the cluster are quorum nodes:

```
mmlscluster
```

To determine the current quorum state, use:

```
mmgetstate -aLs
```

To designate a node as a quorum node:

```
mmchnode --quorum -N NODENAME
```

To designate a node as no longer being a quorum node:

```
mmchnode --nonquorum -N NODENAME
```

```
# mmgetstate -aLs

 Node number  Node name       Quorum  Nodes up  Total nodes  GPFS state   Remarks
-------------------------------------------------------------------------------------
       1      a-scale01          2        3          7       active       quorum node
       2      b-scale01          2        3          7       active       quorum node
       3      tiebreak           2        3          7       active       quorum node
       4      a-scale02          2        3          7       active
       5      a-scale03          2        3          7       active
       6      b-scale02          2        3          7       active
       7      b-scale03          2        3          7       active

 Summary information
---------------------
Number of nodes defined in the cluster:            7
Number of local nodes active in the cluster:       7
Number of remote nodes joined in this cluster:     0
Number of quorum nodes defined in the cluster:     3
Number of quorum nodes active in the cluster:      3
Quorum = 2, Quorum achieved
```

Checking cluster manager status

To determine which node is currently the cluster manager (omit -c to also show file system managers):

```
mmlsmgr -c
```

To move the cluster manager function to a particular quorum node:

```
mmchmgr -c NODENAME
```

```
# mmlsmgr
file system      manager node
---------------- ------------------
fs1              10.0.200.11 (a-scale01)

Cluster manager node: 10.0.200.11 (a-scale01)

# mmchmgr -c b-scale01
Appointing node 10.0.200.21 (b-scale01) as cluster manager
Node 10.0.200.21 (b-scale01) has taken over as cluster manager


# mmlsmgr -c
Cluster manager node: 10.0.200.21 (b-scale01)
```

Clusters can be… stretched

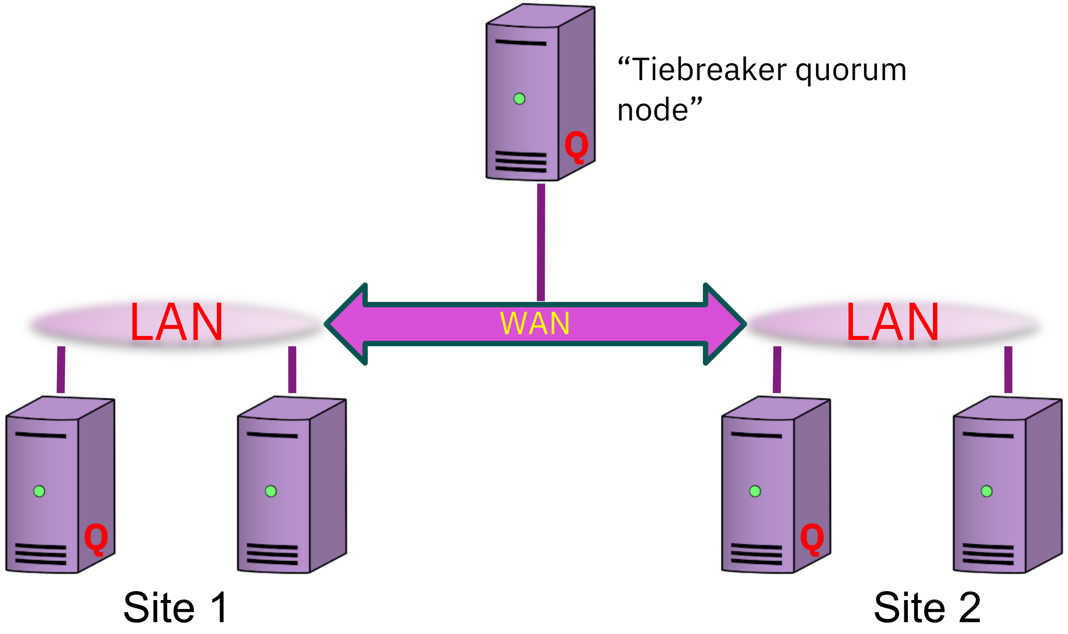
This is still a single Spectrum Scale cluster, but in two parts or "sites", separated or "stretched" with a WAN link.

- If one site fails, the other site should still be available.
- If the WAN link between the sites fails, one site should still be available – we accept that the other site will fail.

Each site has the same number of quorum nodes – but to choose a cluster manager, we need more than half the quorum nodes.

This requirement is met using an additional quorum node that is not part of either site.

- This is typically called the "tiebreaker quorum node".
- If the tiebreaker quorum node is down, the two sites can form quorum without it.



Stretched file systems, failure groups, and replication

… because redundancy and repetition can be good things

Replication and Failure Groups

A **storage *pool*** is a class of storage device. Every NSD (disk volume) is assigned to a pool when it is added to a file system.

A **failure group** indicates the failure domain of an NSD (often linked to the location). Every NSD is assigned to a failure group when it is added to a file system.

For every file system, there is a **metadata replication factor** and **default data replication factor** –these may be 1, 2, or 3 (but no higher than the maximums set for the file system).

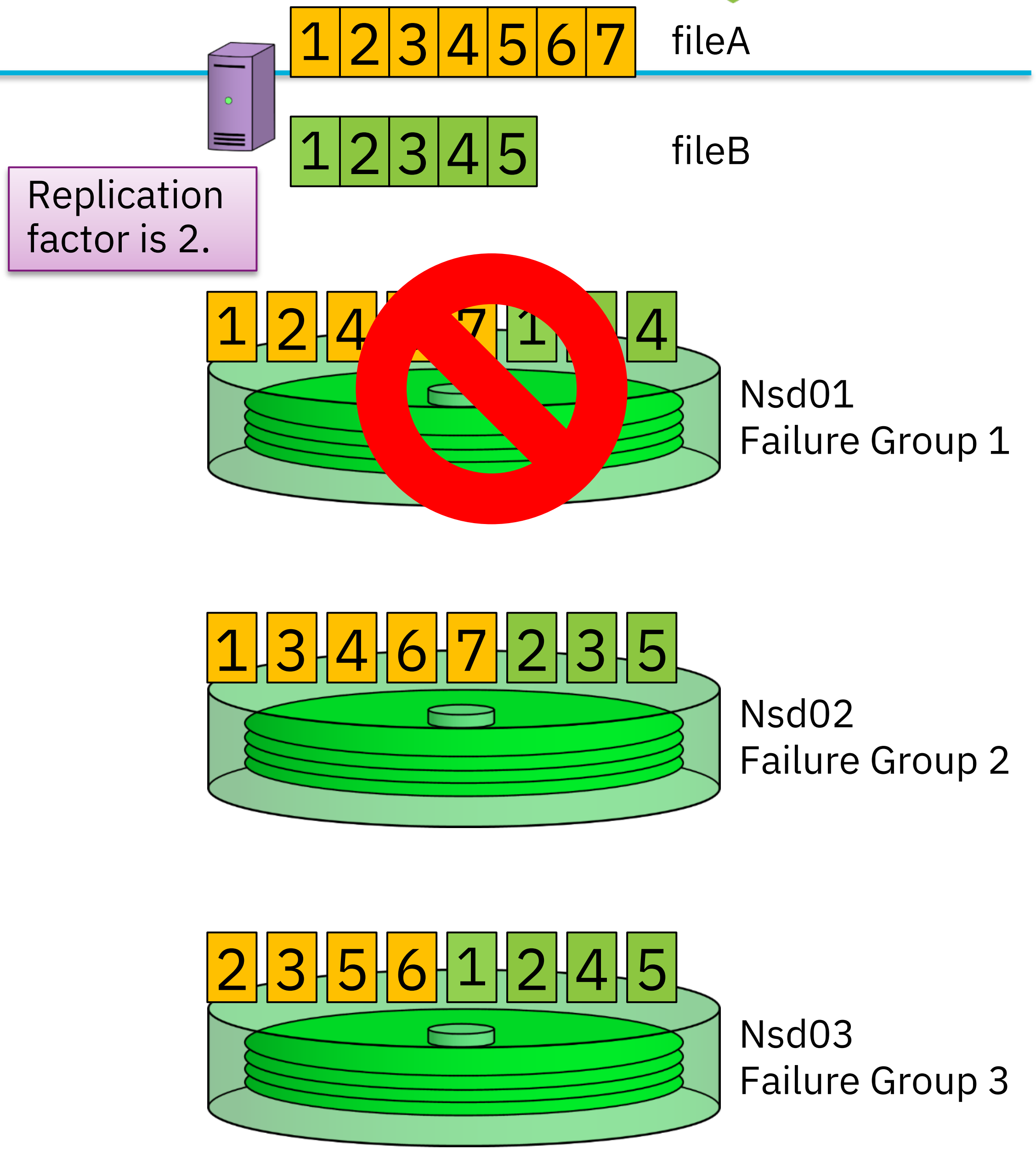
Every file has a storage pool and a replication factor, `r`, associated with it:

- Generally `r` will be the default data replication factor, but it may be adjusted for an individual file (up to the maximum data replication factor for the file system)
- Every block of every file has `r` instances (“replicas”), each in the same pool, but different failure groups.
- This is not disk mirroring, but the effect is similar.

Every block of a file is in the same storage pool.

Failure groups are useful to separate fault tolerant regions (storage server, rack, row, *data centers*, etc.)

Judicious use of replication enables updating file system components (NSD servers, disk firmware, etc.) while the file system remains active.



Creating failure groups

When adding disks to a file system, be sure to include a `failureGroup` clause in the stanza.

This example adds in a couple of disks to both failure groups 1 and 2.

```
mmcrnsd -F fs1-new.stanza
mmadddisk fs1 -F fs1-new.stanza
```

The file system also must be configured to support replication – the maximum data replication (-R) and metadata replication (-M) can not be changed later.

```
mmcrfs fs1 -F fs1.stanza -m 2 -M 3 -r 2 -R 3 \
  -Q yes -A yes -i 4k -S relatime --filesetdf -k all \
  -T /scale/fs1
```

```
%nsd:
    nsd=d1
    device=/dev/dm-2
    servers=scale01,scale02
    failureGroup=1

%nsd:
    nsd=d2
    device=/dev/dm-4
    servers=scale02,scale01
    failureGroup=2

%nsd:
    nsd=d3
    device=/dev/dm-6
    servers=scale01,scale02
    failureGroup=1

%nsd:
    nsd=d4
    device=/dev/dm-7
    servers=scale02,scale01
    failureGroup=2
```

Stretched clusters enable… stretched file systems

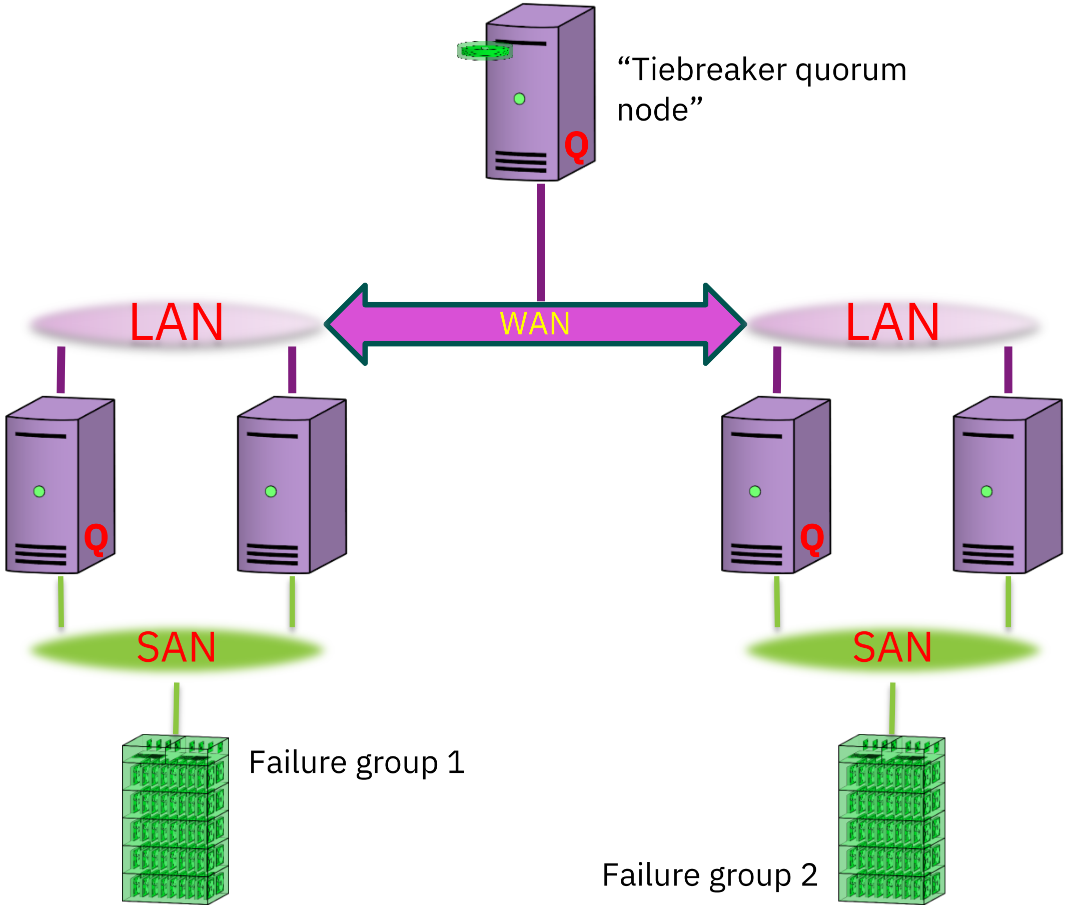
Because the cluster is stretched, all file systems are visible from nodes in both sites.

A **stretched file system** uses replication and failure groups to ensure all metadata and data has a replica at both sites.

- If a site should fail, the other site will be able to continue to read and write to the local failure group.
- When the failed site returns, the recovered failure group will be updated to reflect changes it missed.

A stretched cluster may have both stretched and "unstretched" file systems.

- A "unstretched" file system – one whose storage is only at one site – will become unavailable if that site goes down.



Synchronous writes

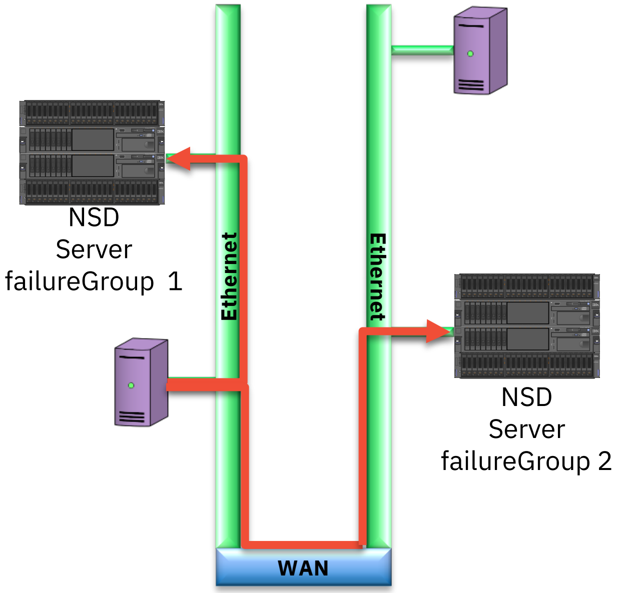
Writes are **synchronous** – all replicas are written in parallel. (OS buffering helps mitigate the WAN performance penalty.)

Updating files also require a log record (metadata) written before the data is written – this is also synchronous.

When a site fails, its disks are placed in the stopped state. However, as files are written, storage is still allocated on the stopped disks.

Other conditions may affect replication – the **`-K`** replication strictness flag affects these cases. Leave this at the default of **`whenpossible`** (see appendix for more details).

Missed updates are checked and corrected when failed disks are brought back online.



Which replica is read?

You can avoid the WAN latency penalty when *reading* data by setting `readReplicaPolicy`.

The `readReplicaPolicy` controls how Scale chooses a replica when a node reads a file block:

`DEFAULT` – *Any* replica may be used to satisfy the request.

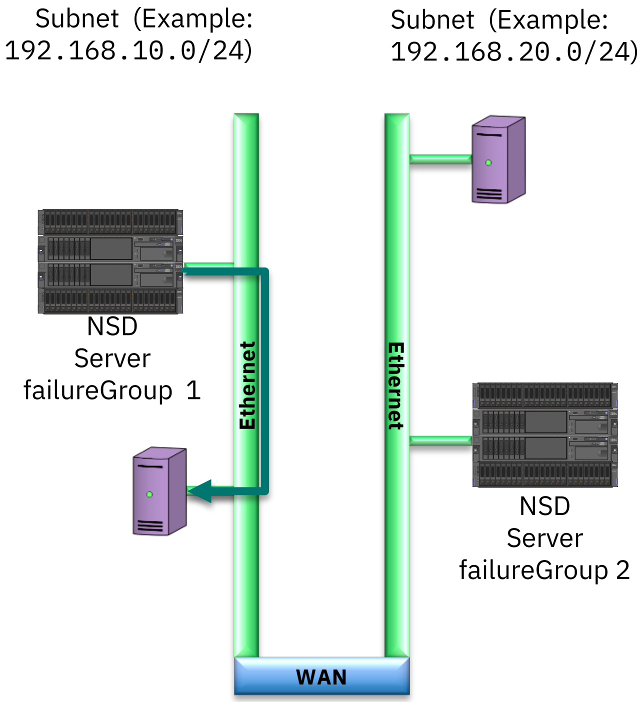
`local` – If a replica can be obtained through a direct connection (SAN), use that one first. If a replica is on an NSD server on the same layer 3 network (subnet) as the requesting node, use that replica. Finally, any replica may be used.

`fastest` – Based on disk statistics, use the “fastest” disk.

Consider aligning failure group strategy to networks, to facilitate using the `local` setting.

The `fastest` setting is best used when intersite latency is large relative to disk seek times.

- `fastestPolicyMinDiffPercent`
- `fastestPolicyNumReadSamples`
- `fastestPolicyMaxValidPeriod`
- `fastestPolicyCmpThreshold`



Subnet (Example: 192.168.10.0/24)
Subnet (Example: 192.168.20.0/24)
NSD Server failureGroup 1
Ethernet
Ethernet
NSD Server failureGroup 2
WAN

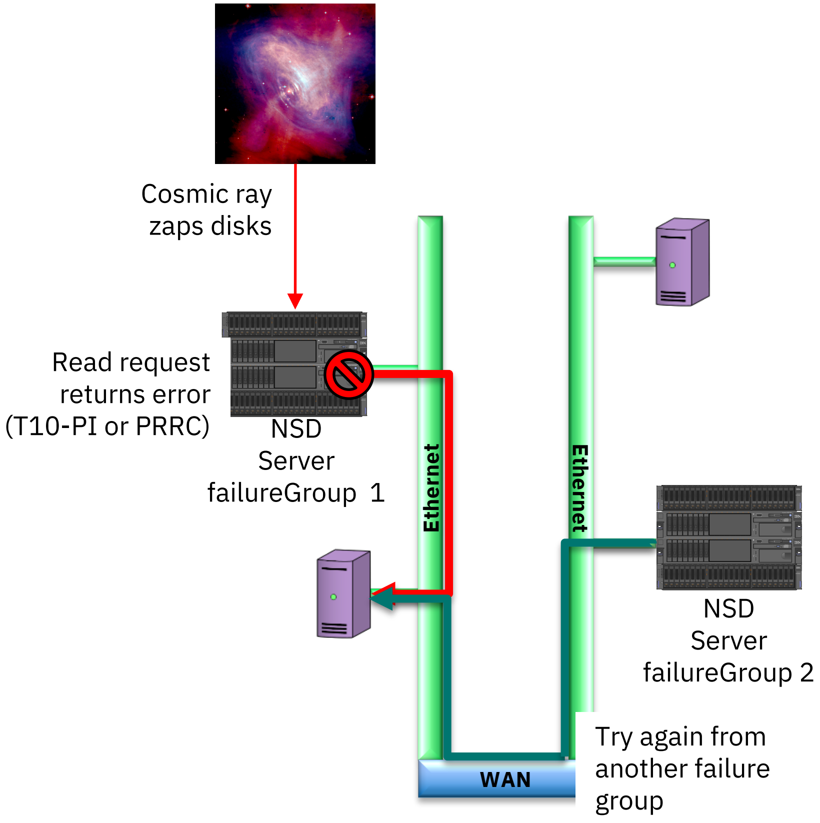
Limitations of failure groups

Metadata blocks are written with checksums, so if a replica has a URE, another replica is used.

Data blocks, unless written to Spectrum Scale RAID disks (e.g, an ESS), have no checksums, so UREs can corrupt data.

- Defend against this by using RAID6 and enabling T10-PI or parity read checks; or use an ESS.
- Note that `nsdCksumTraditional=yes` will enable data block checksum on network transfers, *not on the disk storage itself*.

Do not replicate to thin-provisioned storage without using appropriate `thinDiskType` NSD specifications. If an underlying volume unexpectedly fills, log recovery may also fail, leaving the file system offline.


Cosmic ray zaps disks
Read request returns error (T10-PI or PRRC)
NSD Server failureGroup 1
Ethernet
Ethernet
NSD Server failureGroup 2
WAN
Try again from another failure group

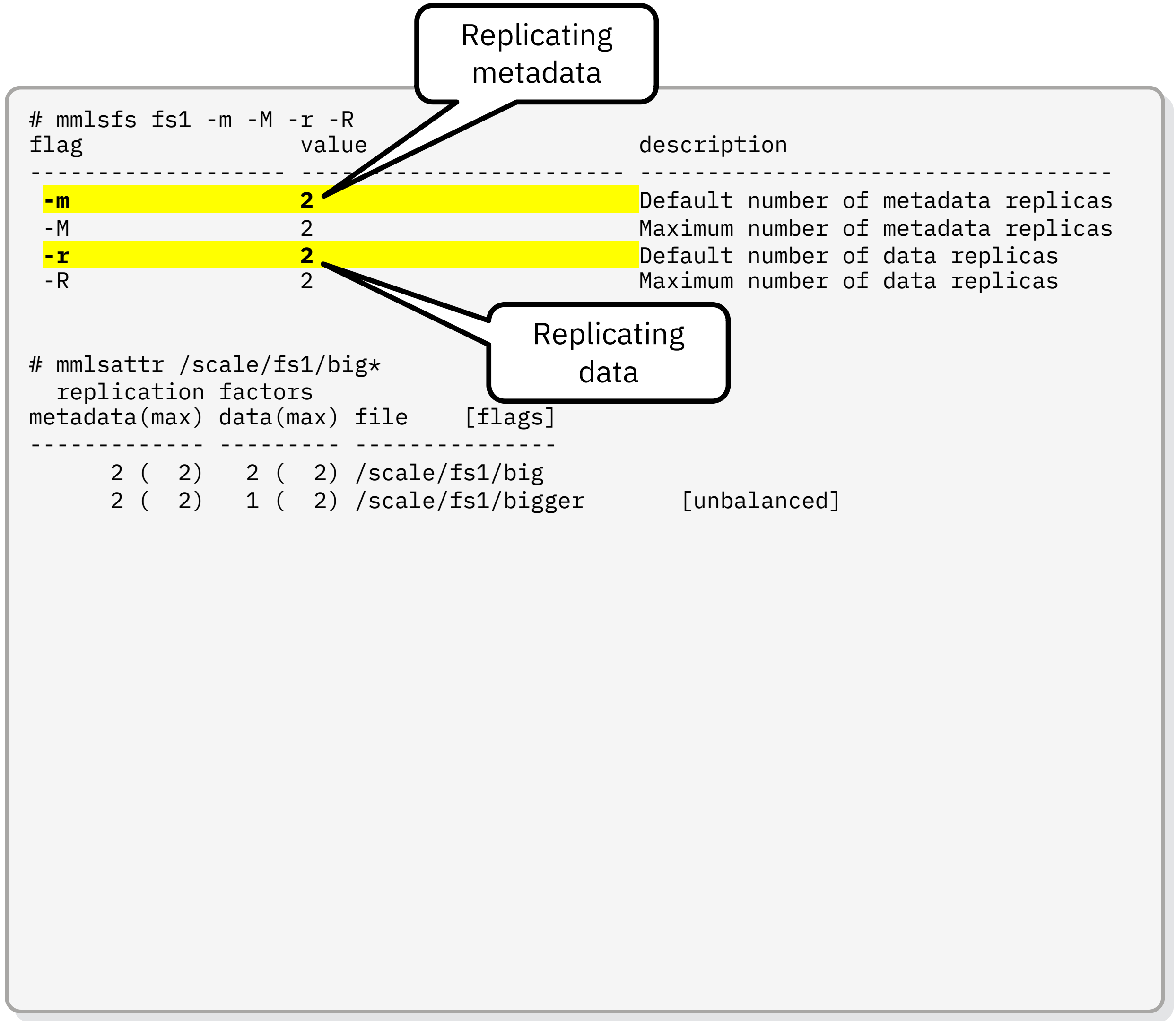
Checking the replication status of file systems and files

Use `mmlsfs` to check on the replication capabilities of the file system.

Use `mmlsattr` to check the replication status of a file.

Use `mmchfs` to change the default replication of a file system (as permitted by the “maximum” settings), followed by `mmrestripefs`.

The policy engine is able to change replication status of a file, but it can not determine which failure groups may be used. (The `mmchattr` command can also change the replication factor of a file.)



```
# mmlsfs fs1 -m -M -r -R
flag                value                    description
------------------- ------------------------ -----------------------------------
 -m                 2                        Default number of metadata replicas
 -M                 2                        Maximum number of metadata replicas
 -r                 2                        Default number of data replicas
 -R                 2                        Maximum number of data replicas


# mmlsattr /scale/fs1/big*
  replication factors
metadata(max) data(max) file     [flags]
------------- --------- ---------------
      2 (  2)   2 (  2) /scale/fs1/big
      2 (  2)   1 (  2) /scale/fs1/bigger       [unbalanced]
```

Replicating metadata

Replicating data

Recovering from failed disks

To start disks after failure and correct any missed updates, use:

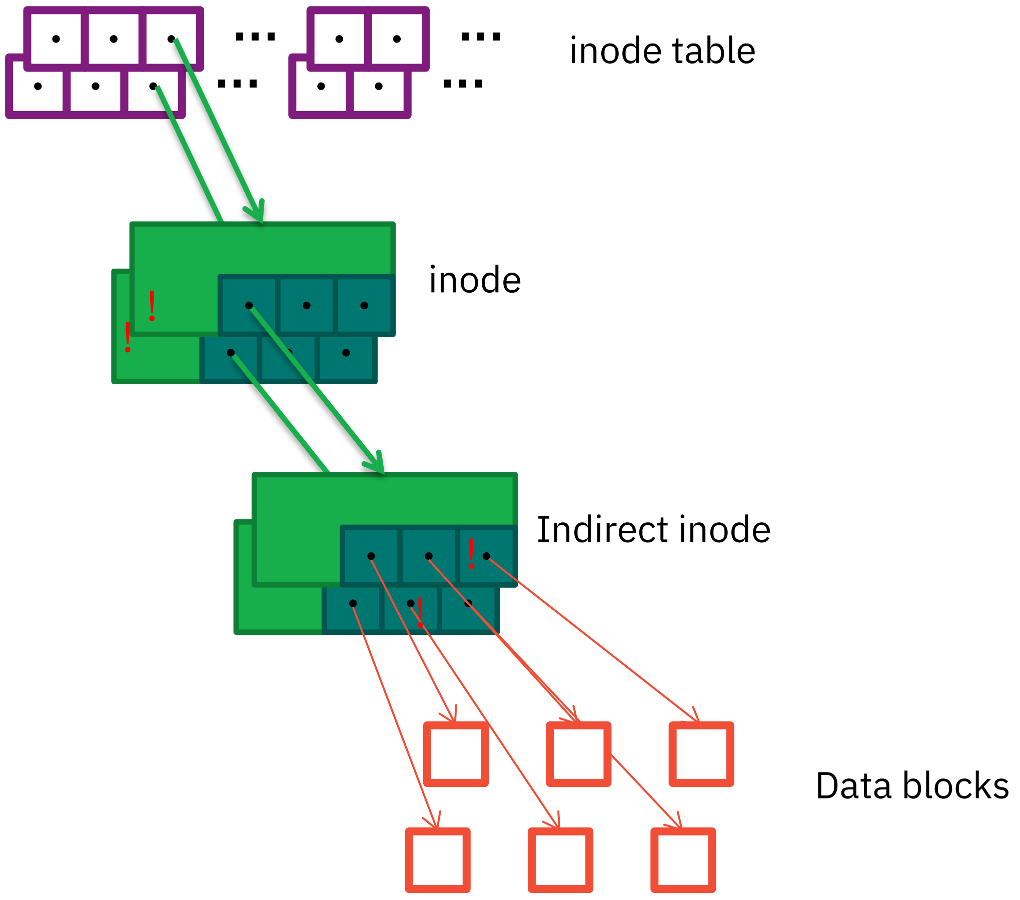
```
mmchdisk FSNAME start -a
```

Recovery uses the “PIT” mechanism to split the work over multiple nodes.

- By default, this work is done using **all** nodes in the cluster.
- The **`defaultHelperNodes`** configuration setting limit this work to a subset of nodes.

Inodes are scanned to determine which have the “missed update” flag set.

- These represent files with blocks needing updates.
- The data block pointers are scanned to determine which have missed updates.
- The updated block is copied into the missed block, and the “missed update” flags are cleared.



File system descriptor quorum

…an annoying detail

… or, why two failure groups are never enough!

A third failure group for File System Descriptor quorum

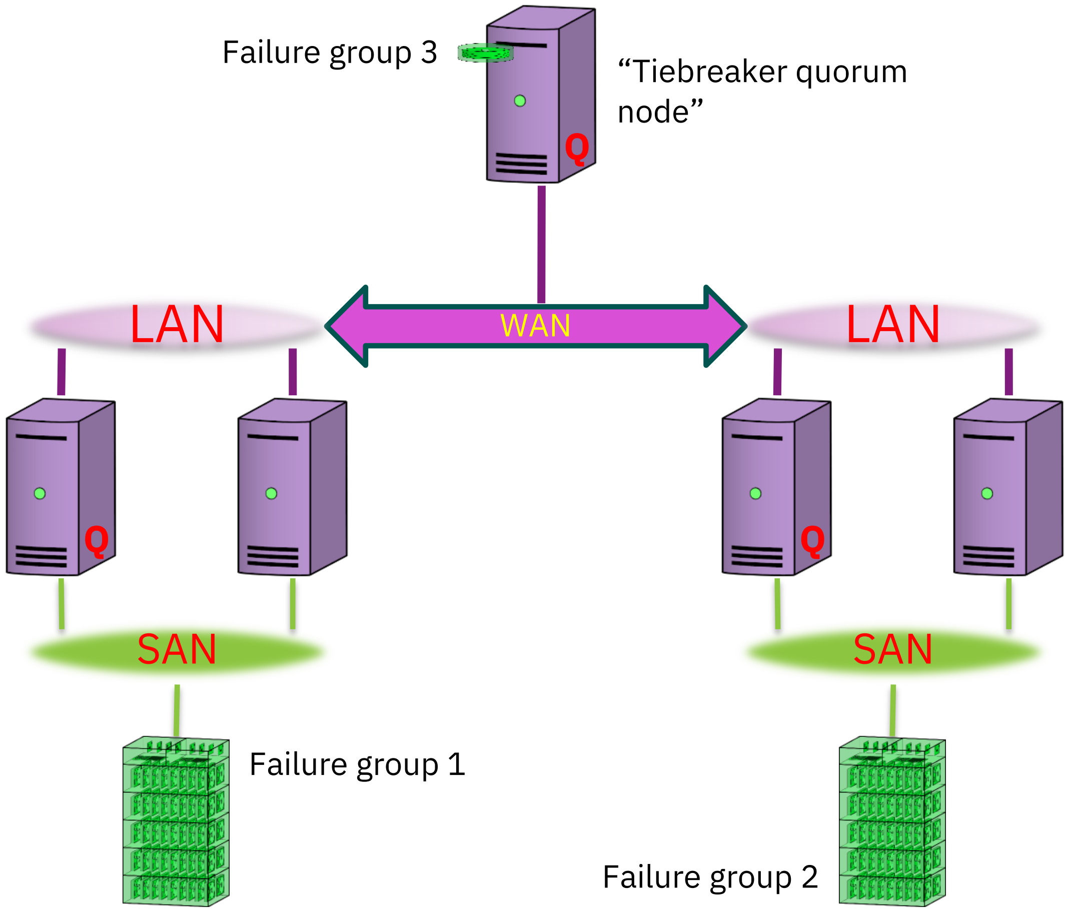
The file system descriptor is a data structure describing attributes and disks in the file system.

Only a very few disks of a file system have current file system descriptors (see appendix), but a majority must be available to mount the file system.

When there are at least 3 disks, then **3 active replicas are maintained, each in a different failure group if there are at least 3 failure groups.**

Each of our main failure groups will receive a copy of the active file system descriptor.

- Put a third failure group on the tiebreaker quorum node, so if we have node quorum, we also have file system descriptor quorum.



Setting up descriptor-only disks

A disk may be offered as a candidate for holding only a file system descriptor by indicating its usage as `descOnly` in the stanza file used to add it to the file system.

`descOnly` only offers the disk as a candidate for an active file system descriptor.

- It alone is not sufficient to guarantee an active file descriptor is written there!
- In other words, it repels data and metadata, but doesn’t attract an active file system descriptor.
- Other rules, like being the only candidate disk in a third or fifth failure group, can force the issue.

A `descOnly` disk can be small (128 MiB is enough).

These are completely different than “tiebreaker quorum disks” – do not configure them as such!

```
%nsd:
    nsd=fs1desc3
    device=/dev/sdb
    servers=tiebreak
    usage=descOnly
    failureGroup=3
```

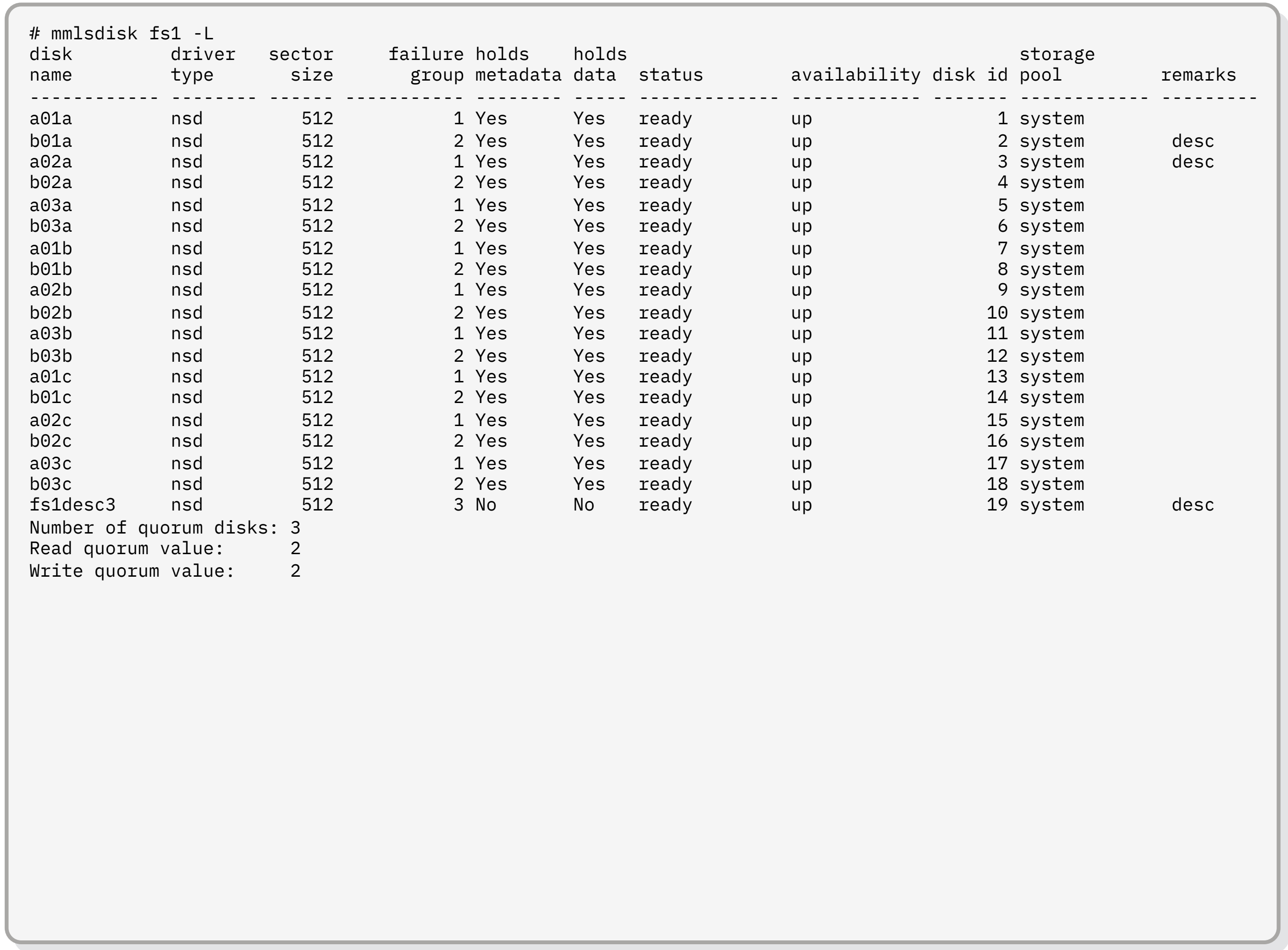
IBM Spectrum Scale

Operations on file system descriptors

To determine which disks have active file system descriptors:

```
mmlsdisk FSNAME -L
```

To move the file system descriptor off a disk, first suspend it (at which point another candidate is chosen), then resume it.



```
# mmlsdisk fs1 -L
disk         driver   sector     failure holds    holds                                    storage
name         type       size       group metadata data  status        availability disk id pool         remarks
------------ -------- ------ ----------- -------- ----- ------------- ------------ ------- ------------ ---------
a01a         nsd         512           1 Yes      Yes   ready         up                 1 system
b01a         nsd         512           2 Yes      Yes   ready         up                 2 system       desc
a02a         nsd         512           1 Yes      Yes   ready         up                 3 system       desc
b02a         nsd         512           2 Yes      Yes   ready         up                 4 system
a03a         nsd         512           1 Yes      Yes   ready         up                 5 system
b03a         nsd         512           2 Yes      Yes   ready         up                 6 system
a01b         nsd         512           1 Yes      Yes   ready         up                 7 system
b01b         nsd         512           2 Yes      Yes   ready         up                 8 system
a02b         nsd         512           1 Yes      Yes   ready         up                 9 system
b02b         nsd         512           2 Yes      Yes   ready         up                10 system
a03b         nsd         512           1 Yes      Yes   ready         up                11 system
b03b         nsd         512           2 Yes      Yes   ready         up                12 system
a01c         nsd         512           1 Yes      Yes   ready         up                13 system
b01c         nsd         512           2 Yes      Yes   ready         up                14 system
a02c         nsd         512           1 Yes      Yes   ready         up                15 system
b02c         nsd         512           2 Yes      Yes   ready         up                16 system
a03c         nsd         512           1 Yes      Yes   ready         up                17 system
b03c         nsd         512           2 Yes      Yes   ready         up                18 system
fs1desc3     nsd         512           3 No       No    ready         up                19 system       desc
Number of quorum disks: 3
Read quorum value:      2
Write quorum value:     2
```

a Spectrum Scale stretched cluster

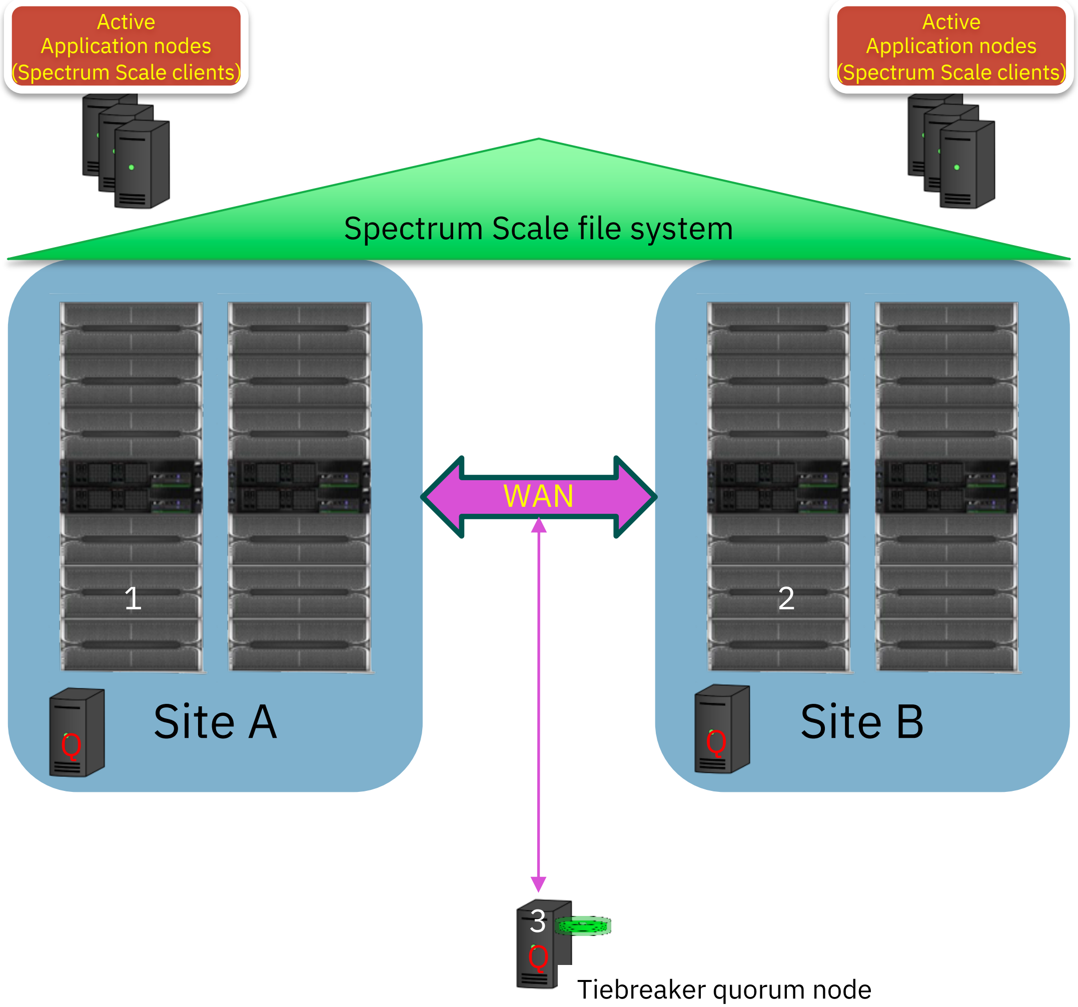
… bringing this all together

Recipe: What goes into a stretched cluster? [Critical]

For a stretched cluster, we need:

1. All disks in each of two sites are assigned to that site's failure group – and total capacity of each failure group should be the same. The file system must use 2-way replication and `whenpossible` replication strictness.
2. Each site has 1 or possibly 2 quorum nodes (must be the same at both sites).
3. Reliable high-bandwidth, low-latency (ideally less than about 10ms) WAN link between both sites, as well as to the tiebreaker quorum node.
4. A tiebreaker quorum node (ideally outside either data center) is part of the cluster, as either the third or fifth quorum node.

- *For each stretched file system in the cluster*, the tiebreaker quorum node needs a small disk (or even a partition), about 128MiB, joined to the file system as a third failure group.
- Keep the cluster manager function off the tiebreaker quorum node.

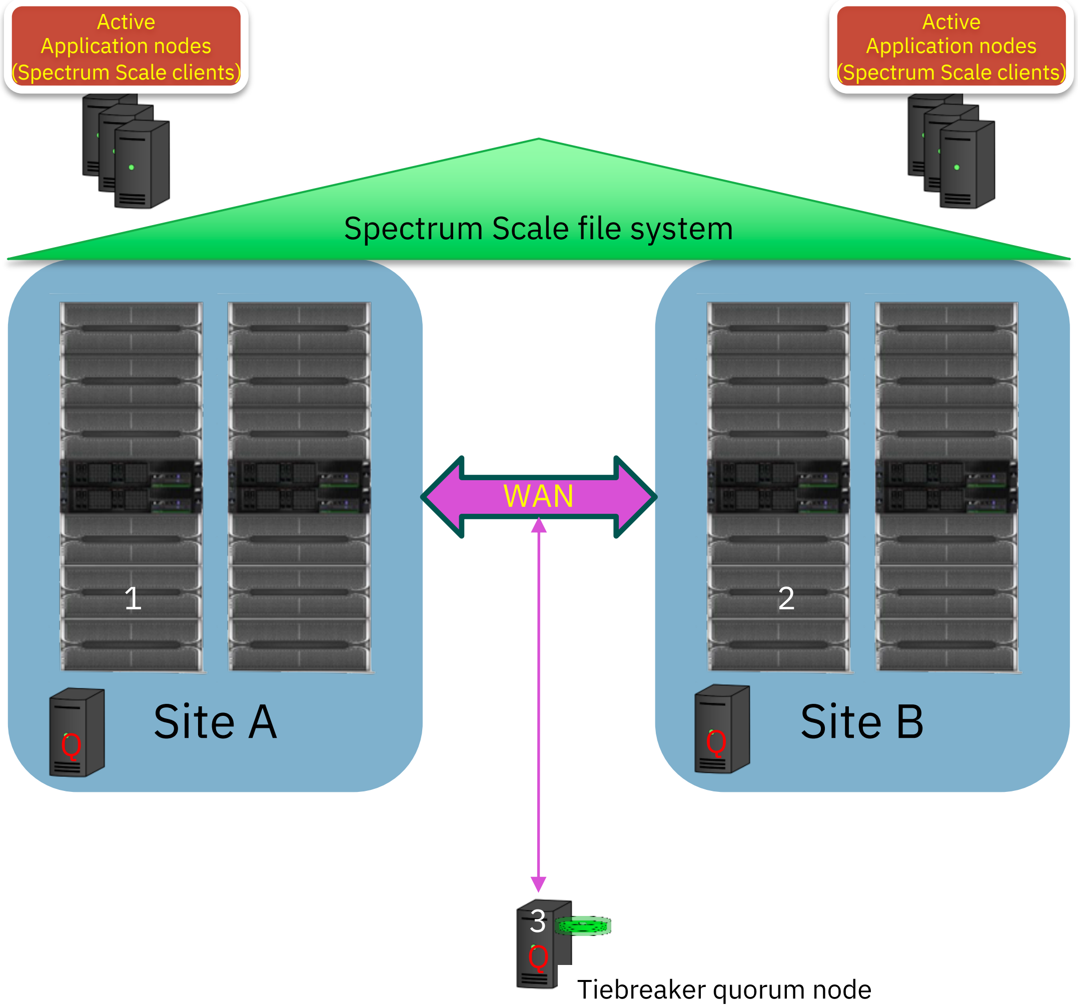


Tiebreaker quorum node with a `descOnly` disk *for each stretched file system*

Recipe: What goes into a stretched cluster? [Best practice]

Some additional best practices:

- Design the network so each site is different layer 3 subnet, allowing the use of `readReplicaPolicy=local`.
- Do not assign the `manager` role to the tiebreaker quorum node – it typically is not well-enough connected to the sites to be a suitable token or file system manager node.
- Learn how to use node classes, and create node classes to help manage site-specific node roles:
  - Aquorum, Bquorum
  - Aces, Bces
  - Ansd, Bnsd
- Choose a set of nodes that will be enlisted for PIT workers and define `defaultHelperNodes`.



Tiebreaker quorum node with a `descOnly` disk *for each stretched file system*

IBM Spectrum Scale

Stretched cluster with Elastic Storage System (ESS)

Typically we want each site with an ESS to have its own management system (EMS).

A word of caution: We are using the word “cluster” in two different ways.

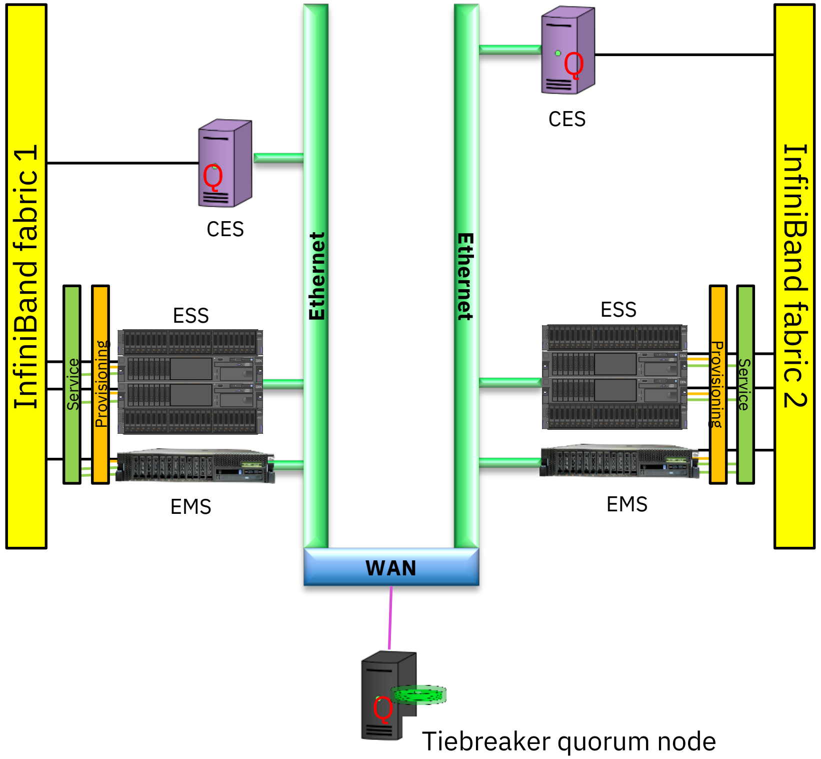
- Set up *separate* **xCAT clusters** at each site.
- Configure everything as a *single* **Spectrum Scale cluster** (before establishing recovery groups).

Federate performance monitoring, so the GUI shows performance of entire Spectrum Scale cluster.

Ideally keep quorum function off ionodes (perhaps placing it on CES nodes).

Use Ethernet for the daemon network, to span the WAN. (It’s possible to bridge IPoIB traffic but don’t expect performance.)

Each site can have local InfiniBand fabrics (make sure `verbsPorts` specifications puts each into a separate fabric).



Tiebreaker quorum node with a `descOnly` disk for each file system

After a failure…

Site B nodes are down, but we still have enough quorum nodes to be quorate.

`mmlsdisk` shows that site B (failure group 2) disks are marked down.

From site A nodes, the file system remains fully functional – applications can both read and write data.

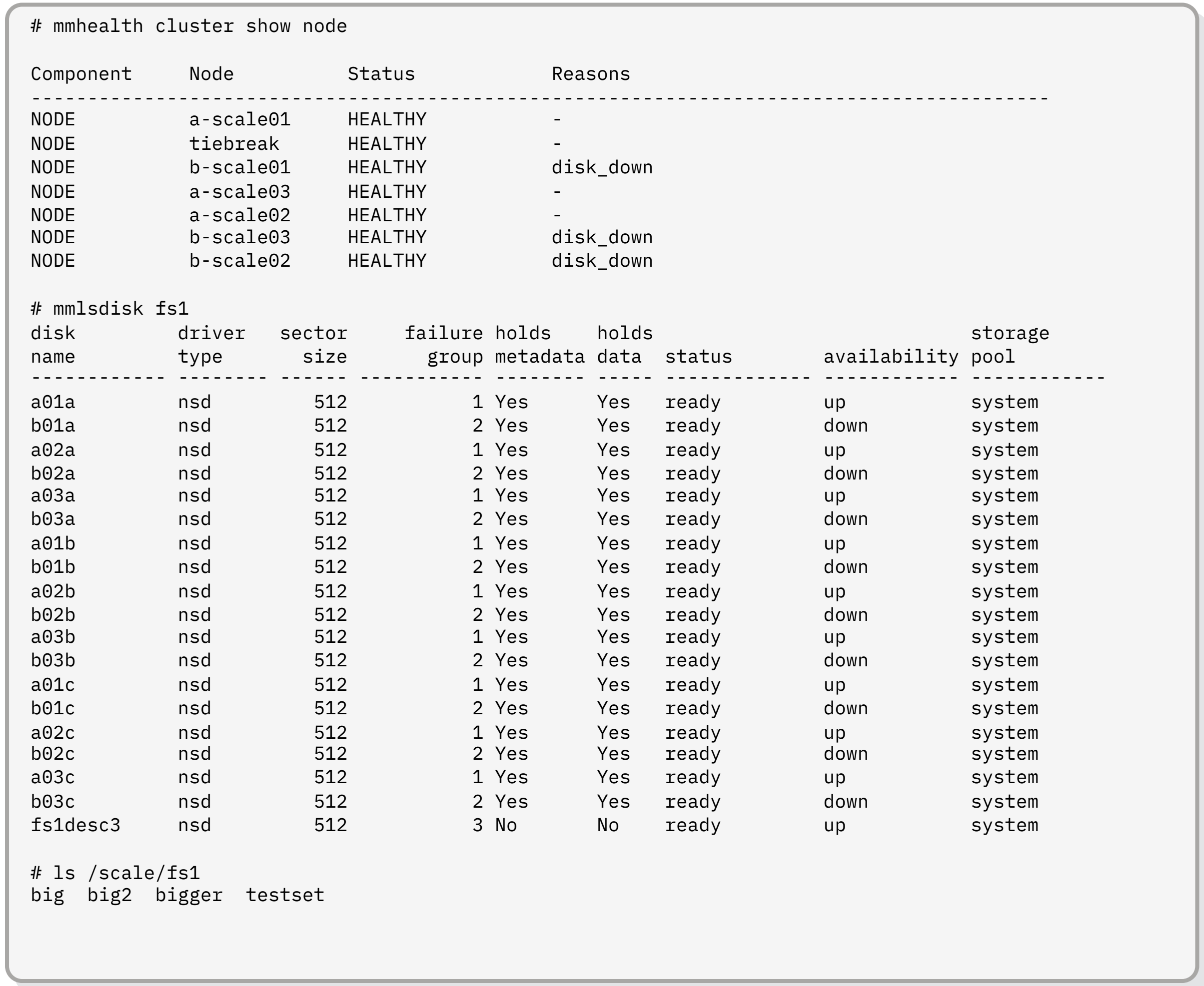
```
# mmgetstate -aL

 Node number  Node name       Quorum  Nodes up  Total nodes  GPFS state   Remarks
------------------------------------------------------------------------------------
       1      a-scale01          2        2          7       active       quorum node
       2      b-scale01          0        0          7       unknown      quorum node
       3      tiebreak           2        2          7       active       quorum node
       4      a-scale02          2        2          7       active
       5      a-scale03          2        2          7       active
       6      b-scale02          0        0          7       unknown
       7      b-scale03          0        0          7       unknown


# mmlsdisk fs1
disk         driver   sector     failure holds    holds                              storage
name         type       size       group metadata data  status        availability pool
------------ -------- ------ ----------- -------- ----- ------------- ------------ ------------
a01a         nsd         512           1 Yes      Yes   ready         up           system
b01a         nsd         512           2 Yes      Yes   ready         down         system
a02a         nsd         512           1 Yes      Yes   ready         up           system
b02a         nsd         512           2 Yes      Yes   ready         down         system
a03a         nsd         512           1 Yes      Yes   ready         up           system
b03a         nsd         512           2 Yes      Yes   ready         down         system
a01b         nsd         512           1 Yes      Yes   ready         up           system
b01b         nsd         512           2 Yes      Yes   ready         down         system
a02b         nsd         512           1 Yes      Yes   ready         up           system
b02b         nsd         512           2 Yes      Yes   ready         down         system
a03b         nsd         512           1 Yes      Yes   ready         up           system
b03b         nsd         512           2 Yes      Yes   ready         down         system
a01c         nsd         512           1 Yes      Yes   ready         up           system
b01c         nsd         512           2 Yes      Yes   ready         down         system
a02c         nsd         512           1 Yes      Yes   ready         up           system
b02c         nsd         512           2 Yes      Yes   ready         down         system
a03c         nsd         512           1 Yes      Yes   ready         up           system
b03c         nsd         512           2 Yes      Yes   ready         down         system
fs1desc3     nsd         512           3 No       No    ready         up           system


# ls /scale/fs1
big  big2  bigger  testset
```

Failback

After site B nodes are back up, we find nodes are healthy other than disks are down.

Disks will remain down until explicitly started.

Meanwhile, even on the site B nodes, the replicated file system is fully accessible.



```
# mmhealth cluster show node

Component     Node          Status            Reasons
------------------------------------------------------------------------------------------
NODE          a-scale01     HEALTHY           -
NODE          tiebreak      HEALTHY           -
NODE          b-scale01     HEALTHY           disk_down
NODE          a-scale03     HEALTHY           -
NODE          a-scale02     HEALTHY           -
NODE          b-scale03     HEALTHY           disk_down
NODE          b-scale02     HEALTHY           disk_down

# mmlsdisk fs1
disk         driver   sector     failure holds    holds                            storage
name         type       size       group metadata data  status        availability pool
------------ -------- ------ ----------- -------- ----- ------------- ------------ ------------
a01a         nsd         512           1 Yes      Yes   ready         up           system
b01a         nsd         512           2 Yes      Yes   ready         down         system
a02a         nsd         512           1 Yes      Yes   ready         up           system
b02a         nsd         512           2 Yes      Yes   ready         down         system
a03a         nsd         512           1 Yes      Yes   ready         up           system
b03a         nsd         512           2 Yes      Yes   ready         down         system
a01b         nsd         512           1 Yes      Yes   ready         up           system
b01b         nsd         512           2 Yes      Yes   ready         down         system
a02b         nsd         512           1 Yes      Yes   ready         up           system
b02b         nsd         512           2 Yes      Yes   ready         down         system
a03b         nsd         512           1 Yes      Yes   ready         up           system
b03b         nsd         512           2 Yes      Yes   ready         down         system
a01c         nsd         512           1 Yes      Yes   ready         up           system
b01c         nsd         512           2 Yes      Yes   ready         down         system
a02c         nsd         512           1 Yes      Yes   ready         up           system
b02c         nsd         512           2 Yes      Yes   ready         down         system
a03c         nsd         512           1 Yes      Yes   ready         up           system
b03c         nsd         512           2 Yes      Yes   ready         down         system
fs1desc3     nsd         512           3 No       No    ready         up           system

# ls /scale/fs1
big  big2  bigger  testset
```

Recovery

Start all disks at once:

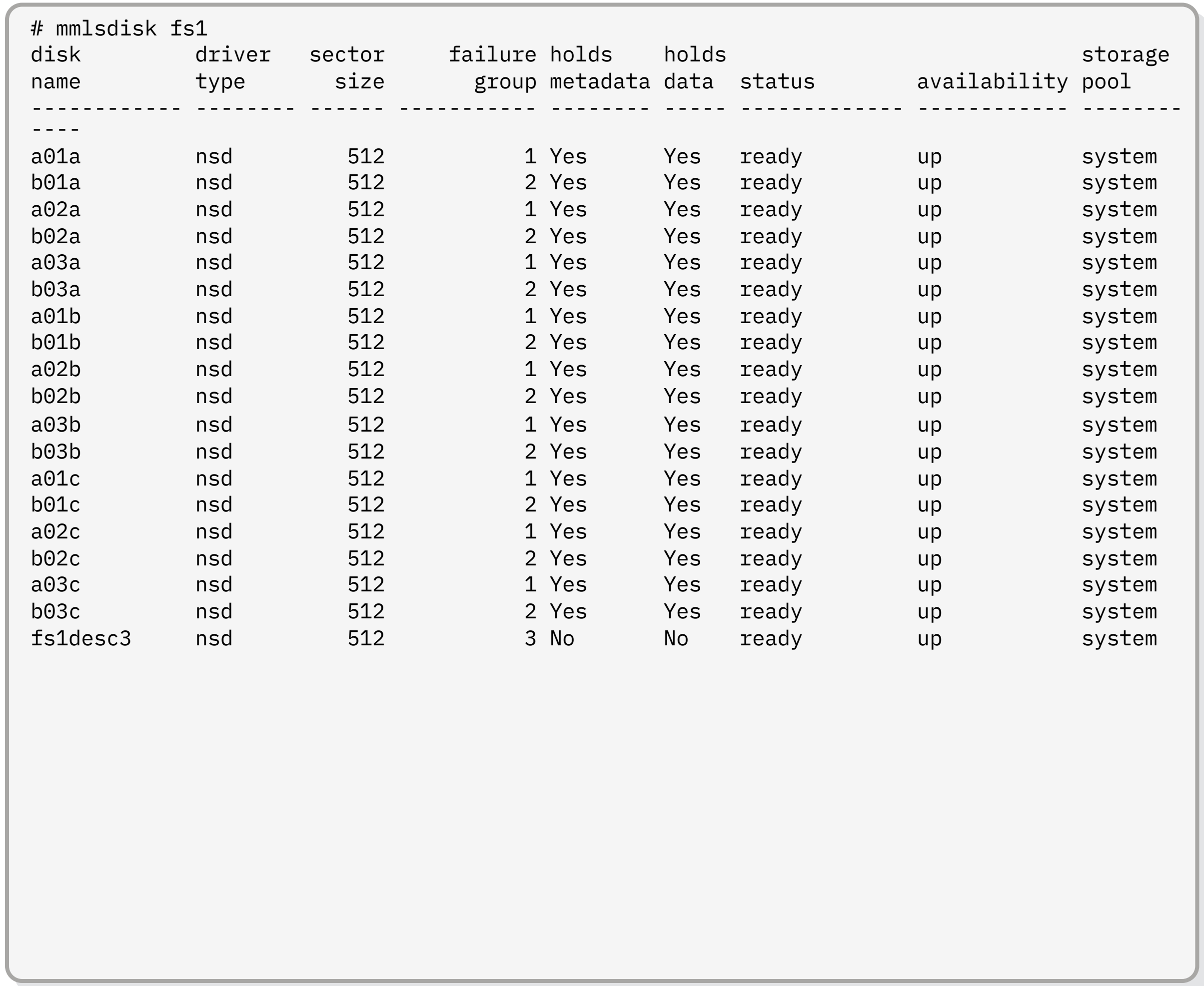
```
mmchdisk fs1 start -a
```

Note that there is no need to restripe the file system after disks are started – the `mmchdisk` command will update the disks as it brings them online.

```
# mmchdisk fs1 start -a
mmnsddiscover:  Attempting to rediscover the disks.  This may take a while ...
mmnsddiscover:  Finished.
tiebreak:  Rediscovered nsd server access to fs1desc3.
a-scale01:  Rediscovered nsd server access to a01b.
a-scale03:  Rediscovered nsd server access to a03a.
a-scale03:  Rediscovered nsd server access to a03b.
a-scale02:  Rediscovered nsd server access to a02a.
b-scale03:  Rediscovered nsd server access to b03a.
a-scale02:  Rediscovered nsd server access to a02b.
b-scale02:  Rediscovered nsd server access to b02a.
b-scale02:  Rediscovered nsd server access to b02b.
b-scale01:  Rediscovered nsd server access to b01a.
b-scale01:  Rediscovered nsd server access to b01b.
a-scale01:  Rediscovered nsd server access to a01c.
a-scale01:  Rediscovered nsd server access to a01a.
a-scale03:  Rediscovered nsd server access to a03c.
a-scale02:  Rediscovered nsd server access to a02c.
b-scale03:  Rediscovered nsd server access to b03b.
b-scale03:  Rediscovered nsd server access to b03c.
b-scale02:  Rediscovered nsd server access to b02c.
b-scale01:  Rediscovered nsd server access to b01c.
Scanning file system metadata, phase 1 ...
 100 % complete on Fri May  8 12:52:52 2020
Scan completed successfully.
Scanning file system metadata, phase 2 ...
 100 % complete on Fri May  8 12:52:52 2020
Scan completed successfully.
Scanning file system metadata, phase 3 ...
Scan completed successfully.
Scanning file system metadata, phase 4 ...
 100 % complete on Fri May  8 12:52:52 2020
Scan completed successfully.
Scanning file system metadata, phase 5 ...
 100 % complete on Fri May  8 12:52:53 2020
Scan completed successfully.
Scanning user file metadata ...
 100.00 % complete on Fri May  8 12:52:53 2020  (     93184 inodes with total       3717 MB data
processed)
Scan completed successfully.
```

After recovery

All disks are now marked as available.



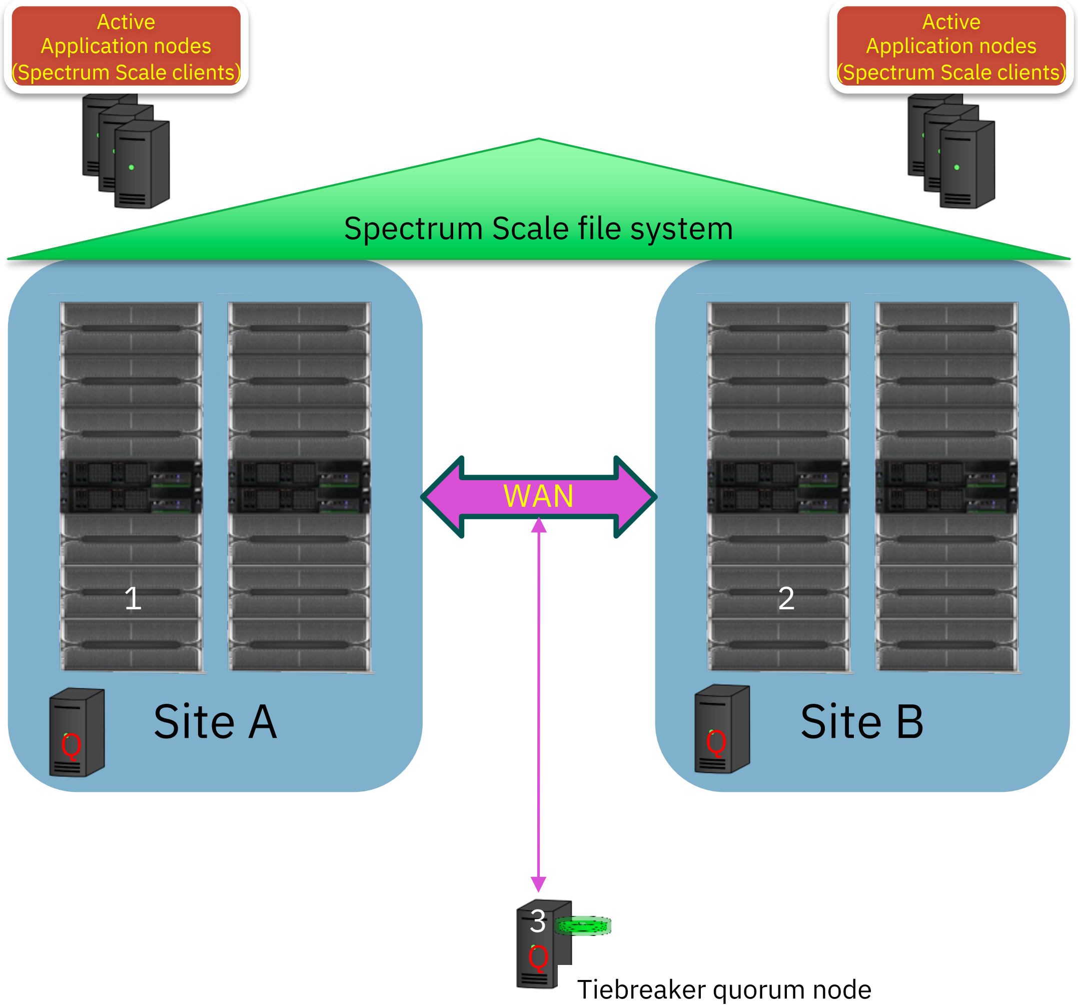
```
# mmlsdisk fs1
disk         driver   sector     failure holds    holds                                   storage
name         type       size       group metadata data  status        availability pool
------------ -------- ------ ----------- -------- ----- ------------- ------------ --------
----
a01a         nsd         512           1 Yes      Yes   ready         up           system
b01a         nsd         512           2 Yes      Yes   ready         up           system
a02a         nsd         512           1 Yes      Yes   ready         up           system
b02a         nsd         512           2 Yes      Yes   ready         up           system
a03a         nsd         512           1 Yes      Yes   ready         up           system
b03a         nsd         512           2 Yes      Yes   ready         up           system
a01b         nsd         512           1 Yes      Yes   ready         up           system
b01b         nsd         512           2 Yes      Yes   ready         up           system
a02b         nsd         512           1 Yes      Yes   ready         up           system
b02b         nsd         512           2 Yes      Yes   ready         up           system
a03b         nsd         512           1 Yes      Yes   ready         up           system
b03b         nsd         512           2 Yes      Yes   ready         up           system
a01c         nsd         512           1 Yes      Yes   ready         up           system
b01c         nsd         512           2 Yes      Yes   ready         up           system
a02c         nsd         512           1 Yes      Yes   ready         up           system
b02c         nsd         512           2 Yes      Yes   ready         up           system
a03c         nsd         512           1 Yes      Yes   ready         up           system
b03c         nsd         512           2 Yes      Yes   ready         up           system
fs1desc3     nsd         512           3 No       No    ready         up           system
```

Some limitations and mitigations

A stretched file system provides active/active high availability, but it is still using a single Spectrum Scale cluster with file systems stretched across sites.

- Thus the entire cluster, or any stretched file system, is subject to misconfiguration or a software error destroying everything.
- Use callbacks to save configuration state after each change – see, for example:

  `/usr/lpp/mmfs/samples/mmsdrbackup.sample`
- Use data protection for the entire cluster.



Tiebreaker quorum node with a `descOnly` disk for each file system

Options for the tiebreaker quorum node

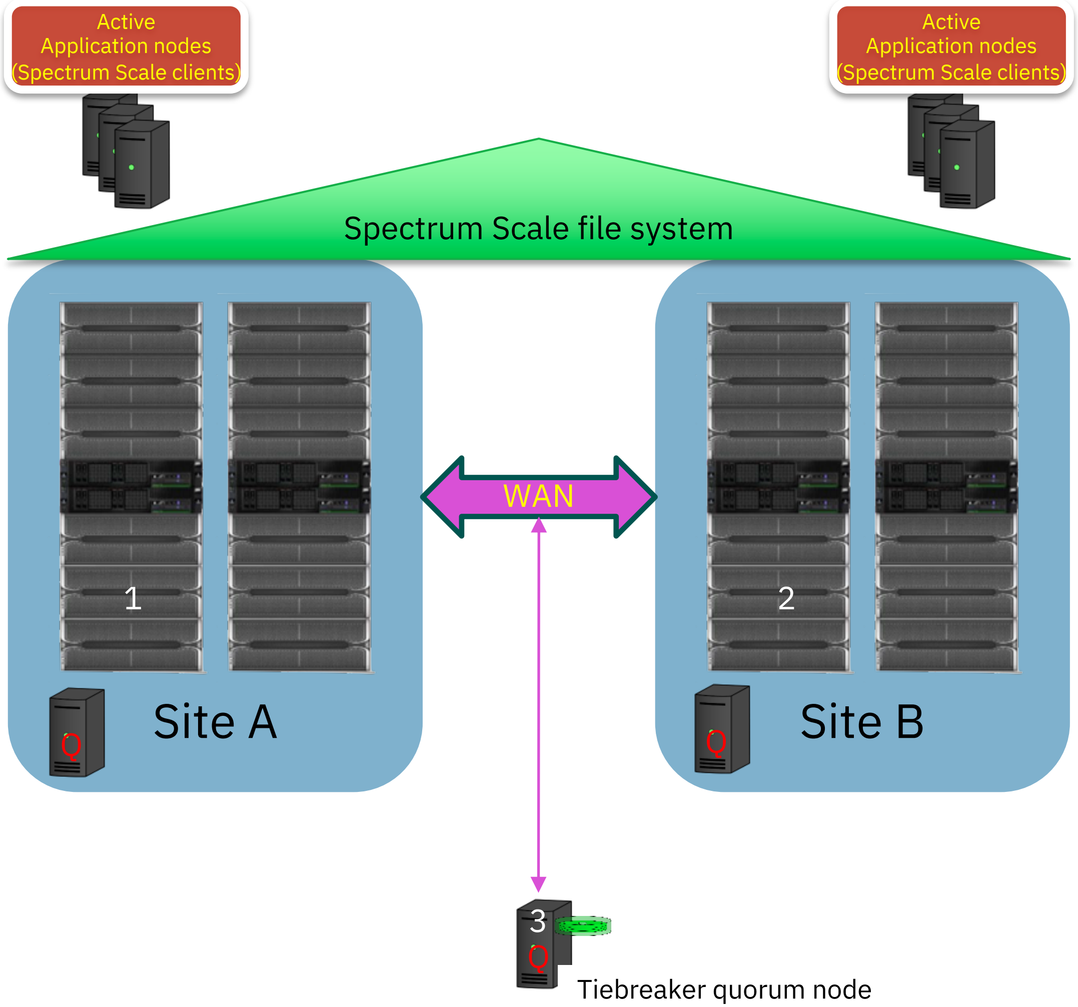
Ideally use an actual third site.

- Perhaps it can go into an off-premises cloud VM, such as in a public cloud?
- Be sure if either storage site fails, the other will still have access to the tiebreaker node.

Sometimes it is unavoidable to have the tiebreaker node in one of your main sites.

- Put it into the site you "favor" – the **primary** site – this is the site that should stay up if the WAN link fails.
- If possible, isolate the tiebreaker node from common failures that may affect the rest of the primary site – put it on different power circuits, network switches, etc.
- The tiebreaker node can be a virtual machine.
- Keep in mind that if the favored site fails, the other site will fail – but you can get Spectrum Scale back up quickly.
- Consider preparing a standby tiebreaker node at the **secondary** site.

Remember you need a *little* storage for file system descriptor NSDs.



Tiebreaker quorum node with a `descOnly` disk for each file system

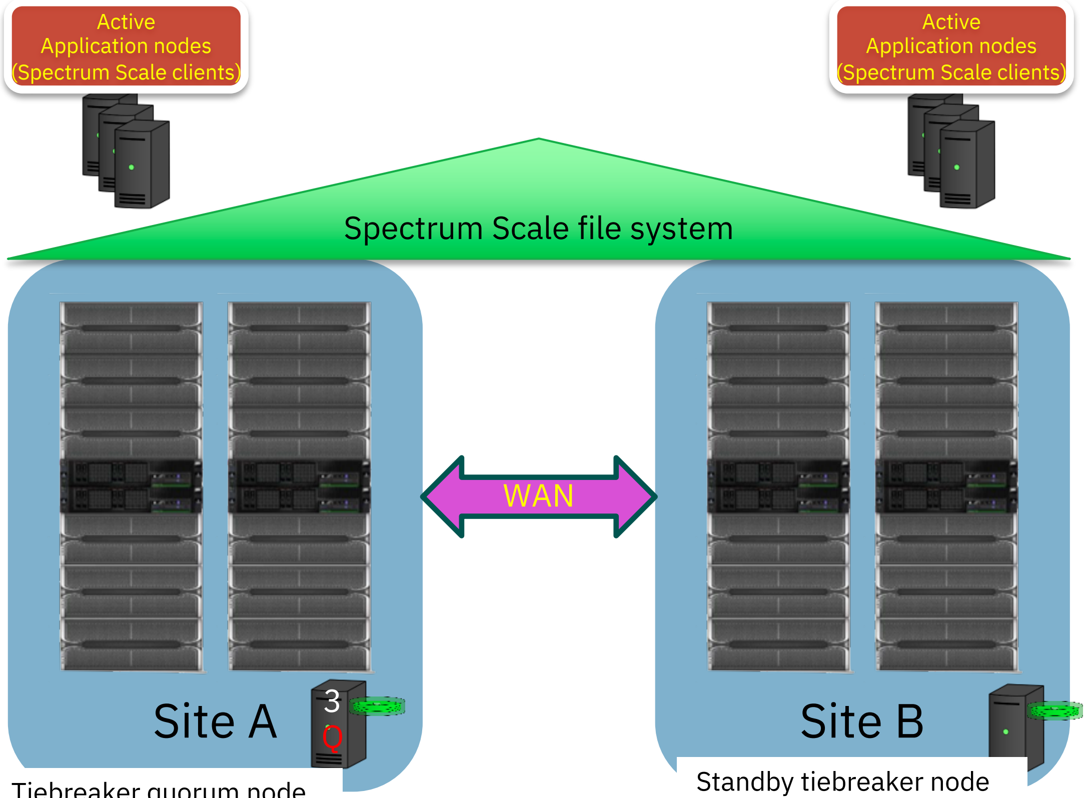
Standby tiebreaker node?

When it is unavoidable for the tiebreaker quorum node to reside in the primary site…

- In the event of an unplanned failure of the primary site goes down, the secondary site will also be down. This will be a **quorum emergency**.
- A standby quorum node in the secondary site can facilitate the recovery process.
- For a planned outage of the primary site, the quorum and file system descriptor functions can be migrated, before the outage, from the normal tiebreaker node to the standby tiebreaker node.

The standby quorum node should be actively part of the Spectrum Scale cluster. Standby file system descriptor disks should already be NSDs.

- **Do not make this a quorum node until needed.**
- **Do not add the NSDs to the file systems until needed.**



Tiebreaker quorum node with a `descOnly` NSD for each file system

Standby tiebreaker node with an unused NSD for each file system

Final considerations

… and the last word

The intersite network

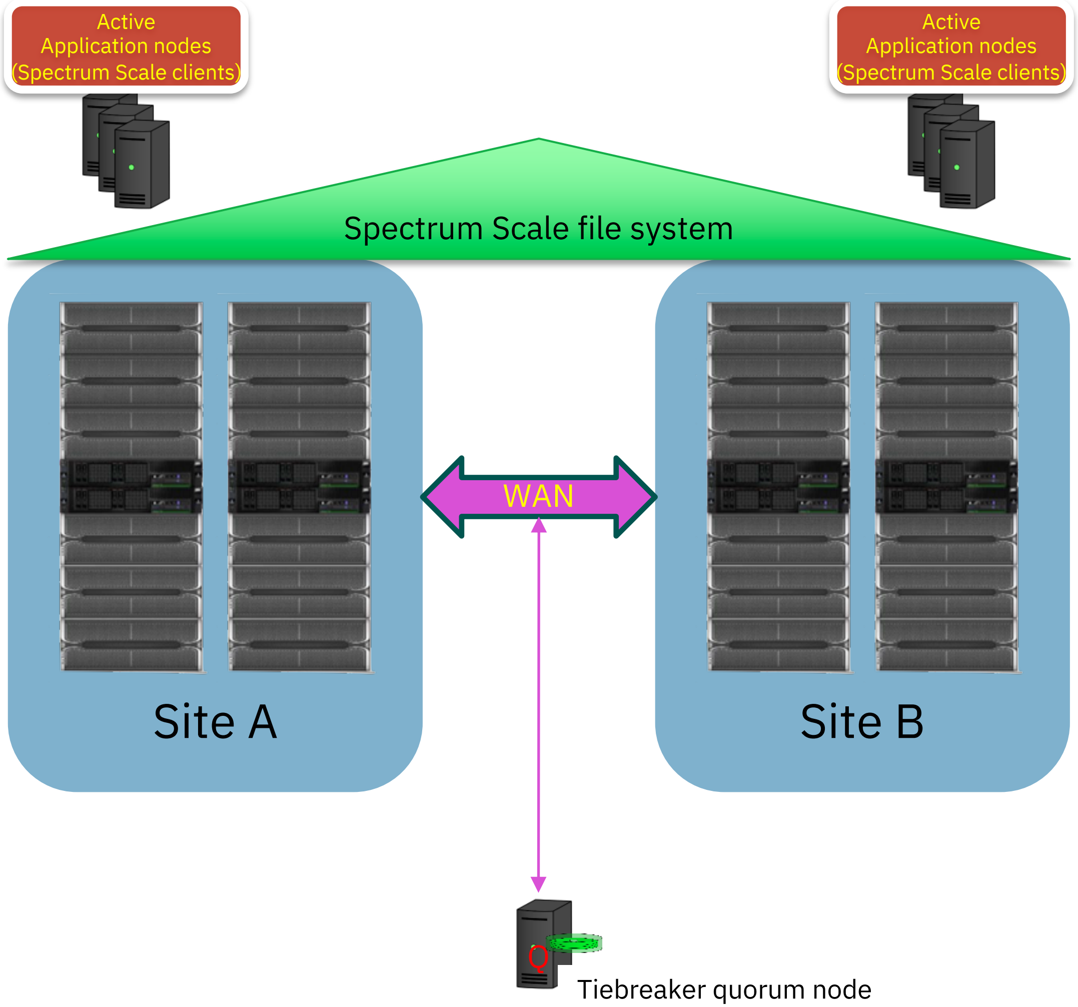
The WAN link needs sufficient bandwidth for the expected workload.

- Pay attention to congestion from other users of this link.
- Pay attention to bonding (and its misuse).

Latency needs to be minimized, but what is tolerable depends on workload.

Latency is a function of distance, as well as other factors. Up to 300km is regularly tested.

Generally I recommend the daemon network be Ethernet (you can still use an InfiniBand fabric for local traffic)



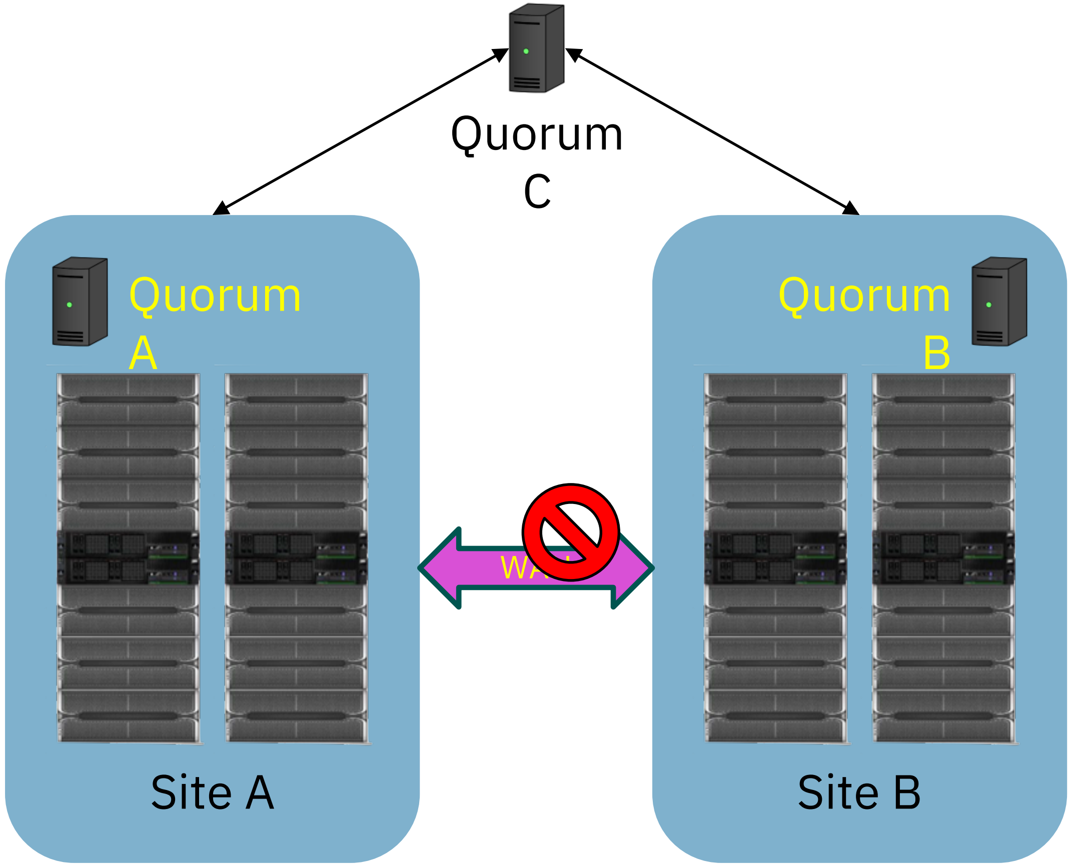
Tiebreaker quorum node with a `descOnly` disk for each file system

Where does the cluster manager run?

Make sure the cluster manager runs on a quorum node at one of the sites, not at a third-site tiebreaker quorum node.

- If the WAN link between sites fails, the site with the cluster manager will remain up.
  - In particular, if one site is "primary", run it there.
  - Callbacks could be used to automate keeping the cluster manager off the tiebreaker node.
  - We want to avoid a situation where the intersite link fails, but the tiebreaker is still visible to both sites.
    - If it were the cluster manager, it would try to keep both sites active, but nodes will be requesting expels when they find they can't communicate cross-site.

This probably limits us to 2-way stretched clusters, even though 3-way replication is supported.



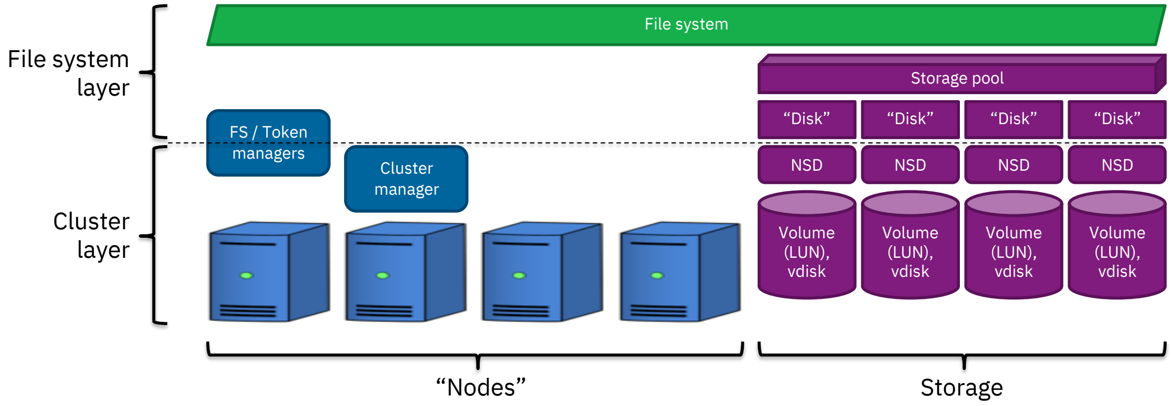
Token management

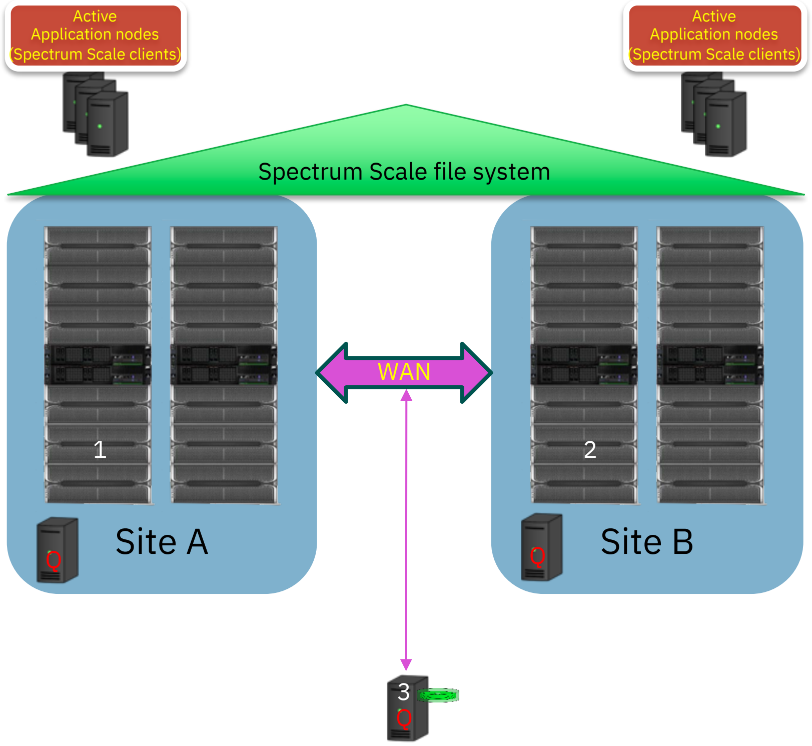
While token management is a file system layer concept, the token managers are common cluster resources.

- Nodes will use token managers on both sides of the WAN link – workloads with bursts of token activity will see latency.
- Increasing `maxStatCache` can reduce token traffic after cache is warm.

If there are no token managers, the file system manager will be on a quorum node and be the only token manager for that file system.

- If one site is actually the primary site for all file systems, put all the token managers there.
- If a single node is sufficient to manage token traffic, it may be possible to have no manager nodes, and perhaps use callbacks to ensure an appropriate quorum node is chosen for each file system's file system manager.




Active Application nodes (Spectrum Scale clients)
Active Application nodes (Spectrum Scale clients)
Spectrum Scale file system
WAN
1
2
Site A
Site B
3

Interactions with some other Spectrum Scale features

Protocol nodes

- If the two sites are in different address spaces, then CES floating addresses can not float between sites.
- High latency between sites will impact performance of SMB cluster file locking.

Transparent Cloud Tiering

- Be sure to place TCT gateway nodes at both sites.

AFM home

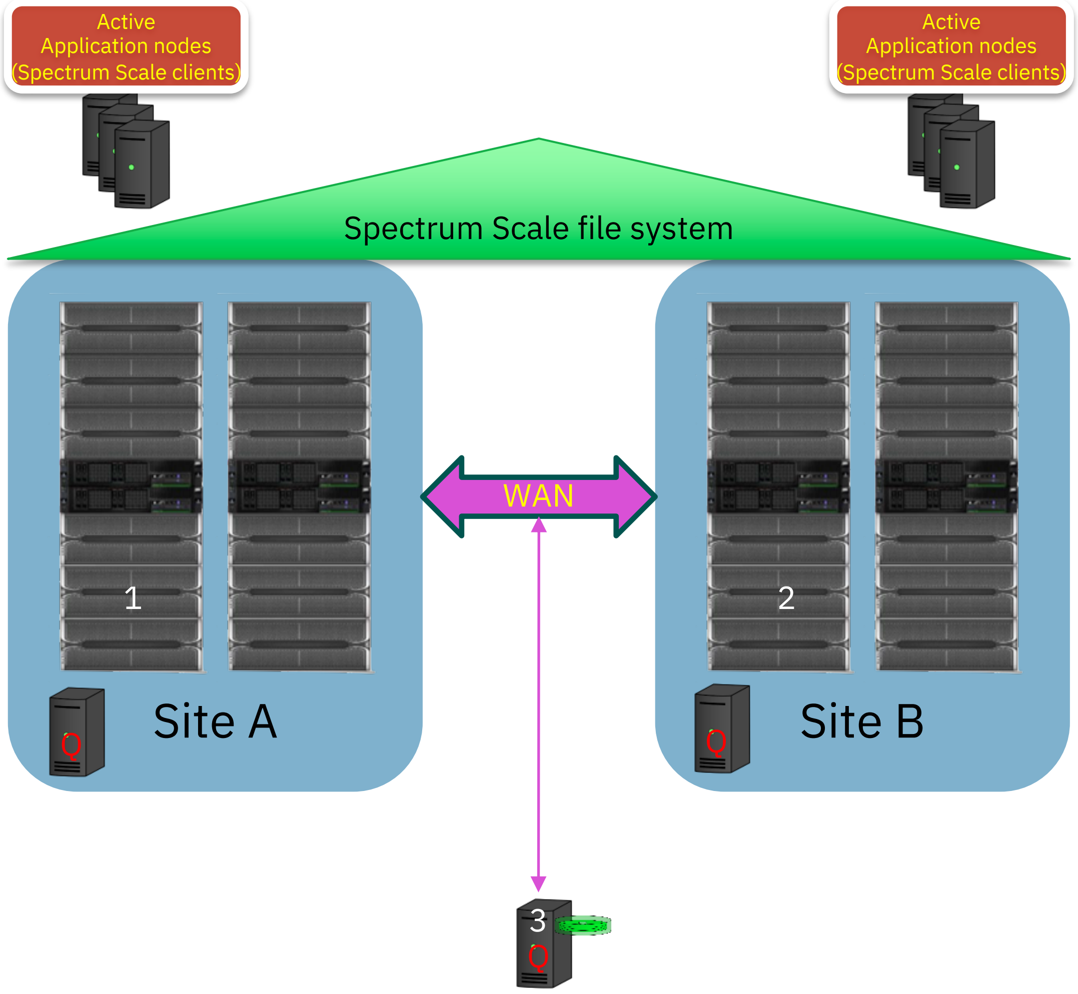
- Typically requires protocol nodes, and for high availability, both sites will need these protocol nodes. Make sure AFM mappings at the cache site can use either site’s protocol nodes. If the home CES addresses don’t cross sites, the cache may disconnect if a site goes down. (Here be dragons. Leverage callbacks at cache site.)

AFM cache

- All nodes communicate with AFM gateway nodes, so write latency is magnified.

Conclusions: Spectrum Scale stretched cluster architecture

Experience shows the Spectrum Scale stretched cluster architecture is a solid architecture for business resiliency.

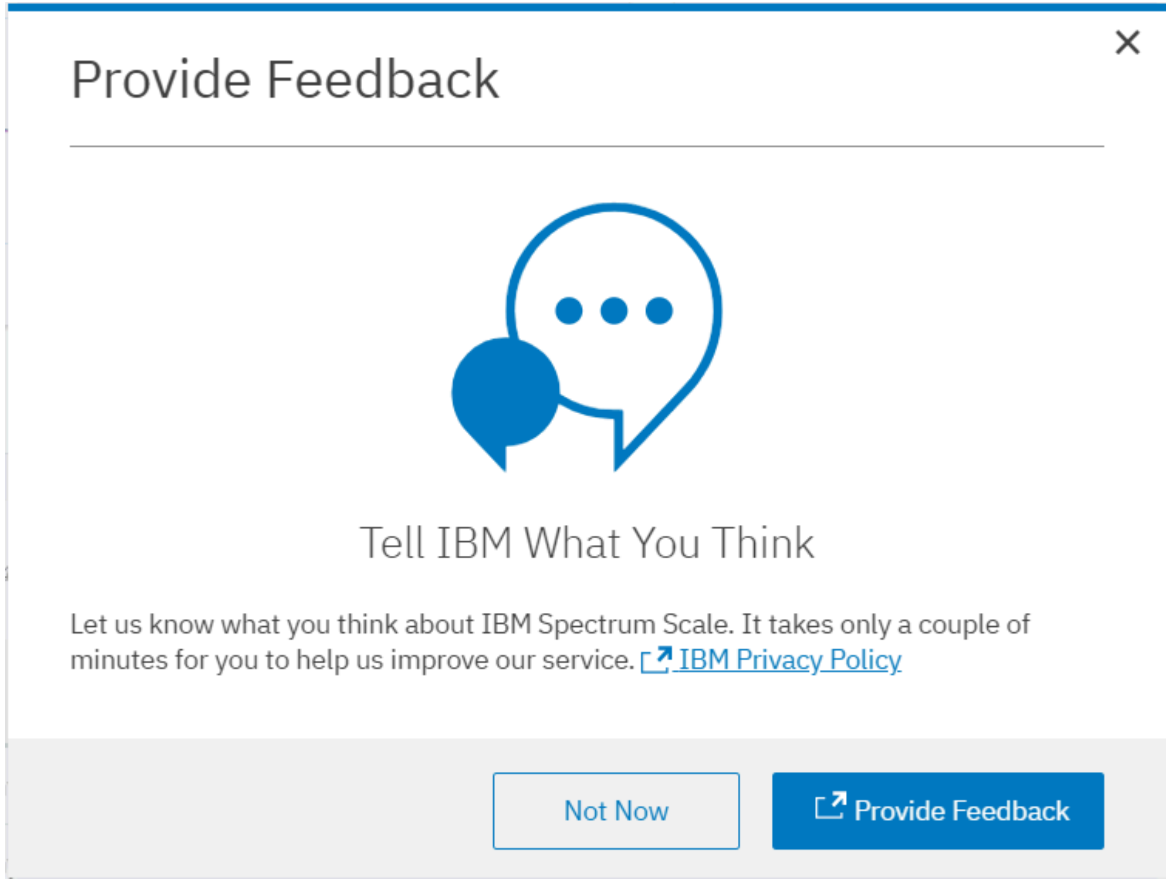


Thank you!

IBM Spectrum Scale

Please help us to improve Spectrum Scale with your feedback

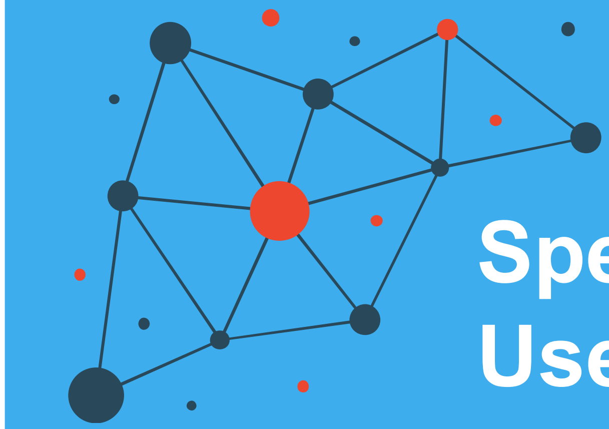
- If you get a survey in email or a popup from the GUI, please respond
- We read every single reply



Provide Feedback

Tell IBM What You Think

Let us know what you think about IBM Spectrum Scale. It takes only a couple of minutes for you to help us improve our service. IBM Privacy Policy



Spectrum Scale User Group

The Spectrum Scale (GPFS) User Group is free to join and open to all using, interested in using or integrating IBM Spectrum Scale.

The format of the group is as a web community with events held during the year, hosted by our members or by IBM.

See our web page for upcoming events and presentations of past events. Join our conversation via mail and Slack.

www.spectrumscaleug.org

Appendix
Tuning quorum

… and what makes a good quorum node?

Tuning disk leasing

`failureDetectionTime` – How many seconds it takes to detect a node is down (default is 35 seconds, same duration as a disk lease).

`leaseRecoveryWait` – When a node fails, wait until known leases expire, then wait this many seconds, before starting recovery. Default is 35 seconds. Intent is to give “in-flight” I/O time to get through controllers on to disks.

`usePersistentReserve` – enables SCSI persistent reserve for disk fencing. *Please check documentation for guidance.*

`minMissedPingTimeout`, **`maxMissedPingTimeout`** – Sets the range in which calculated “missed ping timeout” (MPT) may fall (default between 3 and 60).

- Default MPT is `leaseRecoveryWait-5`.
- After a lease expires, the cluster manager will ping a node, waiting MPT seconds for a response. After that, the node is expelled.

`totalPingTimeout` – Nodes responding to ICMP pings but not sending heartbeats will be declared dead after this timeout (default 120 seconds).

Considerations in choosing quorum nodes

Choose the smallest odd number of nodes that meet necessary redundancy constraints.

- For a stretched cluster, this would typically be 3 or maybe 5 quorum nodes.
  - Each site would have the same number (1 or 2) quorum nodes.
  - A tiebreaker quorum node is also needed, typically a third site, or somehow independent of either site failing.
- Only use tiebreaker disk quorum for small clusters, where all quorum nodes directly access the tiebreaker disks.

Choose reliable nodes!

Network usage is light, but critical.

Quorum function needs local file system `/var/mmfs` to be responsive (not blocked by other system I/O to same disk or controller).

- **An ESS ionode is a poor choice for a quorum node**, since its internal NVMe is on the same disk controller as the local OS disk.

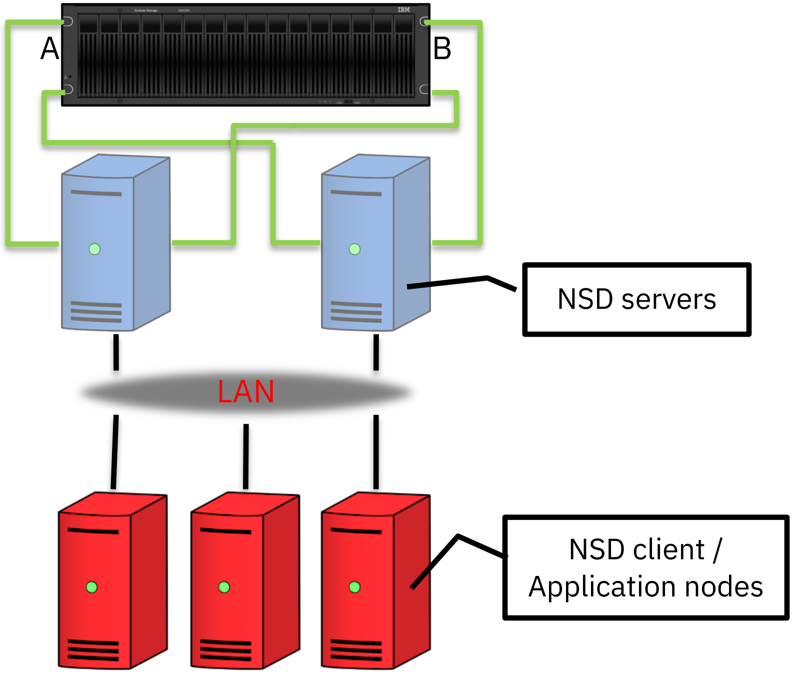
Try to isolate quorum nodes from common failure modes:

  - Put them in separate racks, on separate circuits, etc.

Components of a resilient cluster

A *resilient cluster* uses multiple techniques to survive local failures:

- Multiple links between disks and NSD servers (or application nodes), using multiple controllers and multiple HBAs – multipath is handled by the OS.
- Multiple NSD servers (or application nodes) connected to each disk.
- If this is not an ESS:
  - Use RAID6 volumes.
  - Enable T10-PI or parity read checks – Spectrum Scale replication will not protect you from UREs (except with metadata).
- Plan a robust quorum configuration, using tiebreaker disks if needed.
- Consider the use of multiple *failure groups* to protect against larger failures.
  - Particularly ensure that a single failure won't break file system descriptor quorum.



NSD servers

NSD client / Application nodes

Appendix
File system replication strictness

Synchronous writes and strictness

When all failure groups are on-line, the replication proceeds as expected.

When a site experiences a failure, its disks are placed in the the **stopped** state. Writes will still allocate space in the stopped failure group, and the disks will receive updates when started.

Other conditions may prevent normal allocation of replicas in a failure group. The `-K` replication enforcement parameter controls what happens in these cases:

`no` – If at least one replica can be *allocated*, the `write` completes successfully.

`whenpossible` – If enough failure groups are online, all replicas must be allocated to report success. If not enough are available, do not enforce replication. *This is the default and is generally the correct setting for a stretched cluster.*

`always` – All required replicas must be allocated; else the `write` fails with `ENOSPC`.

	`no`	`whenpossible`	`always`
All disks in a failure group are stopped	Allocates in all failure groups	Allocates in all failure group	Allocates in all failure groups
All disks in a failure group are suspended	Only allocates in unsuspended failure group.[3]	Only allocates in unsuspended failure group.[2,3]	Fails with out of space.[2]
One failure group runs out of space	Allocates in failure groups with space.[1,3]	Fails with out of space.	Fails with out of space.

In all cases, replication factor of files is set.

[1] Failure is silent

[2] Warning when you place disks in this state

[3] Recovery requires `mmrestripefs -r`

Appendix
File system descriptor quorum

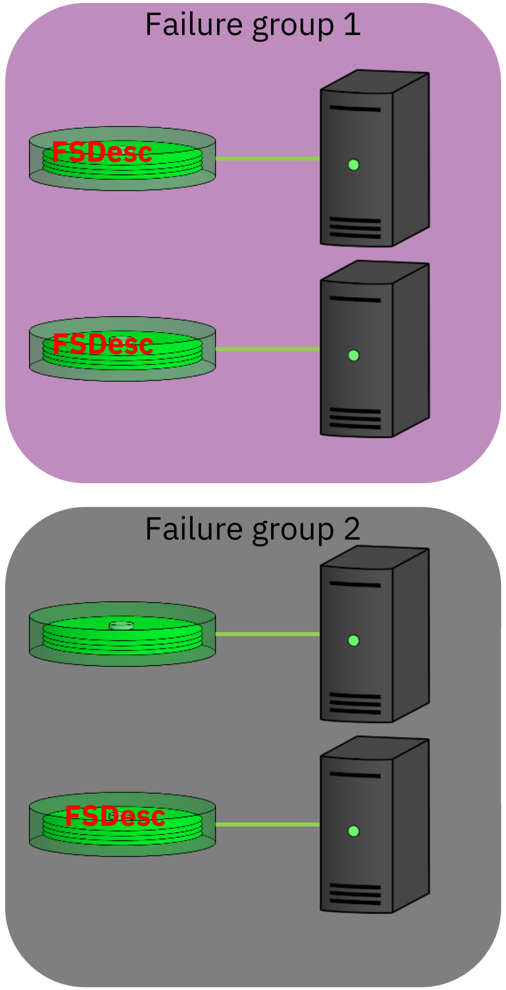
… or, why two failure groups are never enough!

File system descriptor quorum

The file system descriptor is a data structure describing attributes and disks in the file system.

- Every disk in a file system has a replica of the file system descriptor, starting at sector 8.
- However, when there are more than 3 disks in a file system, only several replicas are guaranteed to be active and up to date.
  - If there are at least 5 failure groups, then 5 active replicas are maintained, each in a different failure group.
  - Otherwise, if there are at least 3 disks, then **3 active replicas are maintained, each in a different failure group if there are at least 3 failure groups.**
  - Otherwise maintain an active replica on each disk (there are only 1 or 2 disks).
  - Note that storage pools are ignored when choosing disks.

For each file system, a majority of its active file system descriptor replicas must be available for that file system to be mounted – this is **file system descriptor quorum**.



Even with replication, this file system will not mount if only failure group 2 is active!

Setting up descriptor-only disks

A disk may be offered as a candidate for holding only a file system descriptor by indicating its usage as `descOnly` in the stanza file used to add it to the file system.

`descOnly` only offers the disk as a candidate for an active file system descriptor.

- It alone is not sufficient to guarantee an active file descriptor is written there!
- In other words, it repels data and metadata, but doesn’t attract an active file system descriptor.
- Other rules, like being the only candidate disk in a third or fifth failure group, can force the issue.

A `descOnly` disk can be small (128 MiB is enough).

These are completely different than “tiebreaker quorum disks”!

```
%nsd:
    nsd=fs1desc3
    device=/dev/sdb
    servers=tiebreak
    usage=descOnly
    failureGroup=3
```

Operations on file system descriptors

To determine which disks have active file system descriptors:

```
mmlsdisk FSNAME -L
```

To move the file system descriptor off a disk, first suspend it (at which point another candidate is chosen), then resume it.

```
# mmlsdisk fs1 -L
disk         driver   sector     failure holds    holds                                    storage
name         type       size       group metadata data  status        availability disk id pool         remarks
------------ -------- ------ ----------- -------- ----- ------------- ------------ ------- ------------ ---------
a01a         nsd         512           1 Yes      Yes   ready         up                 1 system
b01a         nsd         512           2 Yes      Yes   ready         up                 2 system        desc
a02a         nsd         512           1 Yes      Yes   ready         up                 3 system        desc
b02a         nsd         512           2 Yes      Yes   ready         up                 4 system
a03a         nsd         512           1 Yes      Yes   ready         up                 5 system
b03a         nsd         512           2 Yes      Yes   ready         up                 6 system
a01b         nsd         512           1 Yes      Yes   ready         up                 7 system
b01b         nsd         512           2 Yes      Yes   ready         up                 8 system
a02b         nsd         512           1 Yes      Yes   ready         up                 9 system
b02b         nsd         512           2 Yes      Yes   ready         up                10 system
a03b         nsd         512           1 Yes      Yes   ready         up                11 system
b03b         nsd         512           2 Yes      Yes   ready         up                12 system
a01c         nsd         512           1 Yes      Yes   ready         up                13 system
b01c         nsd         512           2 Yes      Yes   ready         up                14 system
a02c         nsd         512           1 Yes      Yes   ready         up                15 system
b02c         nsd         512           2 Yes      Yes   ready         up                16 system
a03c         nsd         512           1 Yes      Yes   ready         up                17 system
b03c         nsd         512           2 Yes      Yes   ready         up                18 system
fs1desc3     nsd         512           3 No       No    ready         up                19 system        desc
Number of quorum disks: 3
Read quorum value:      2
Write quorum value:     2
```

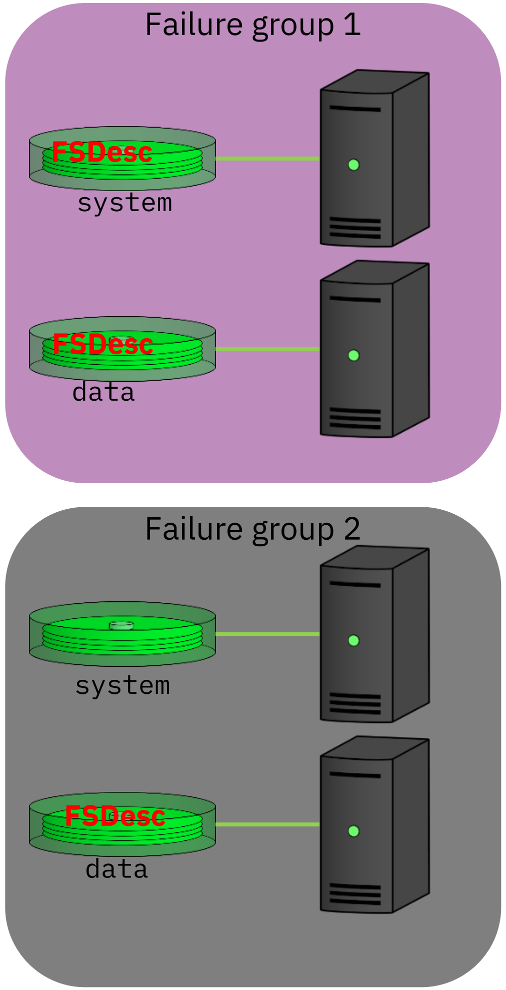
File system descriptors and pools

Even though file system descriptors are metadata, they are *not* bound to the `system` pool.

Disks receiving active file system descriptors are chosen based solely on failure group, irrespective of the pool.

- **If two disks are in the same failure domain, assign them to the same failure group, even if they are in different storage pools.**
- If you ignore this rule, you may find failure descriptors are not where you expected them to be, generally when your stretched cluster didn't stay available during a site failure!

There is no way to force the active file system descriptors to be maintained on a specific class of storage.



Appendix
Dealing with “quorum emergencies”

… because if anything can go wrong, it will!

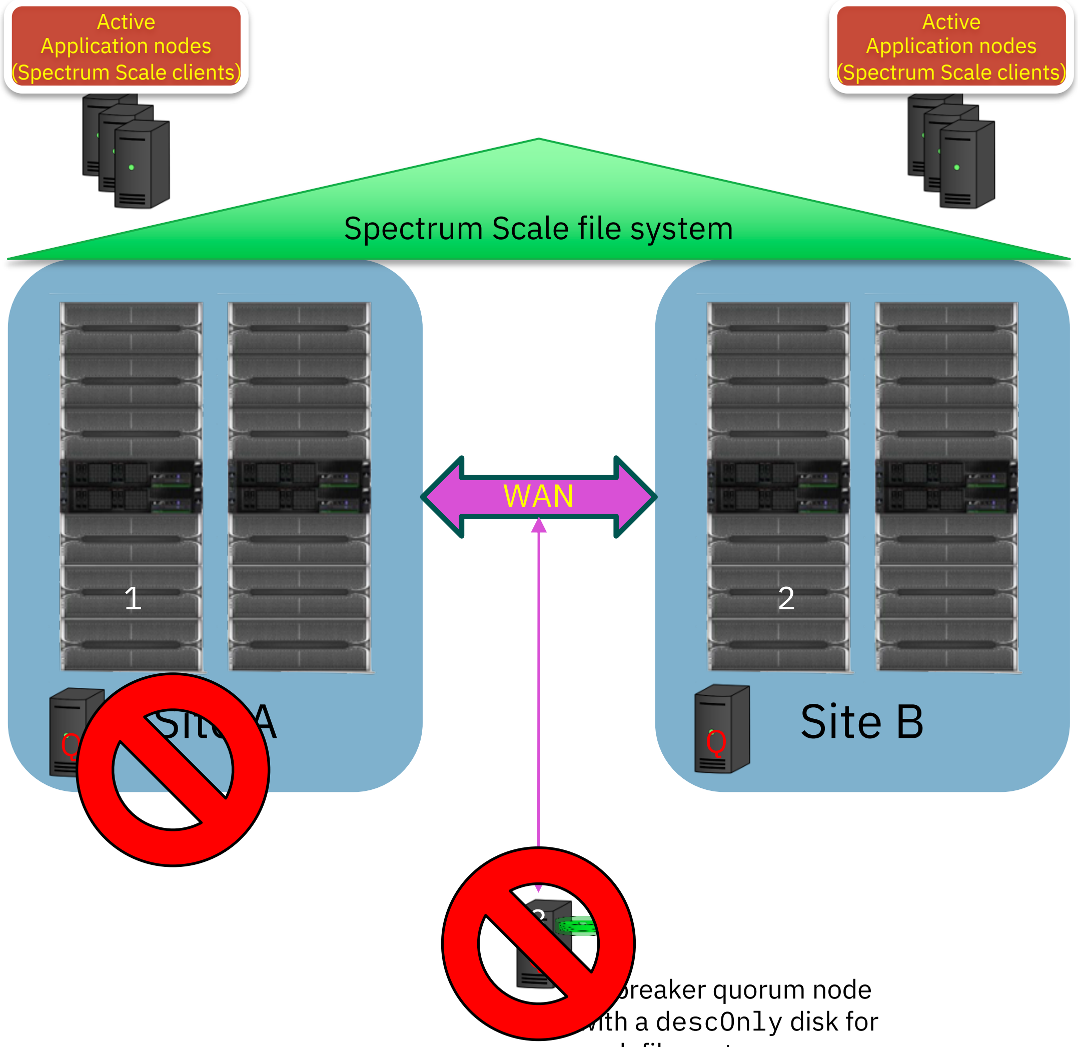
What is a quorum emergency?

Sometimes we lose both a site and the tiebreaker quorum node, or too many file system descriptor disks.

- It isn’t unusual to have the tiebreaker node in one of the sites themselves and hope for the best!
- It isn’t uncommon for someone to forget the file system descriptor disk for a file system (like the CES shared root).

To get back into production, we must:

- Restore cluster quorum
- Restore file system descriptor quorum for all file systems we need online.



…reaker quorum node with a `descOnly` disk for each file system

Emergency recovery (CCR configuration)

Shutdown Spectrum Scale on SURVIVORS:

```
    mmshutdown -N SURVIVORS
```

Remove quorum function from DOWNQUORUM nodes (comma separated):

```
    mmchnode --nonquorum --force --N DOWNQUORUM
```

(You will be prompted to confirm this operation!)

For each file system FSNAME, migrate active file system descriptors from the failure groups that are down (DOWNFG1, DOWNFG2):

```
    mmfsctl FSNAME exclude -G DOWNFG1
    mmfsctl FSNAME exclude -G DOWNFG2
```

Start up Spectrum Scale on surviving nodes.

Failback from a quorum emergency

This will require another outage of the cluster.

First restart Spectrum Scale on the nodes that survived, without mounting the file systems (`-A no`). Do not bring up Spectrum Scale on the nodes that had failed.

Restore the failed tiebreaker site as a quorum node, then bring up Spectrum Scale on it.

Restore the quorum function to the failed quorum nodes, then bring back up all the failed nodes.

Create a file of all the NSDs attached to the nodes that had failed. Use it to again enable them to be file system descriptor candidates.