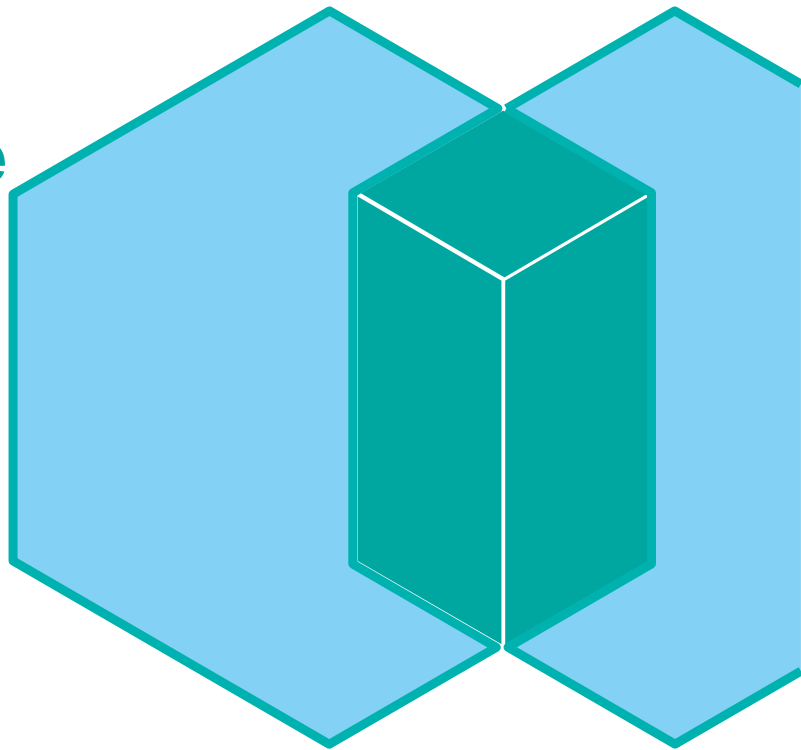# Deep-dive on Spectrum Scale Reliability, Availability and Serviceability improvements

Mathias Dietz
Spectrum Scale RAS Architect

IBM Research and Development
in Kelsterbach/Frankfurt, Germany
mdietz@de.ibm.com

NETWORK RESILIENCY

Network Resiliency

Typical Problem:

Random nodes get expelled from the cluster

- Network Communication between some cluster nodes is blocked by a firewall
- Network is instable/high packet loss

Spectrum Scale requires:

- Any-to-any node communication in the cluster !
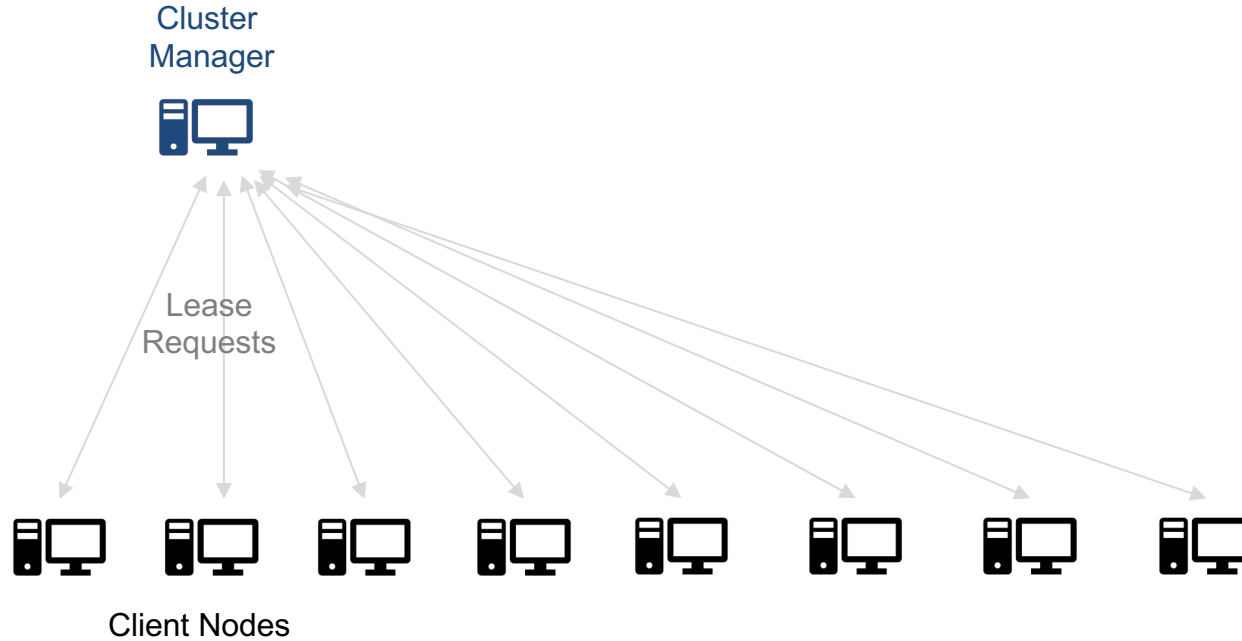- Stable network connectivity without packet loss
- **But why ?**

Spectrum Scale Network Communication

Cluster
Manager

Client Nodes

Spectrum Scale Network Communication

Cluster Manager

Lease Requests

Client Nodes

Spectrum Scale Network Communication

Spectrum Scale Network Communication

Spectrum Scale Network Communication

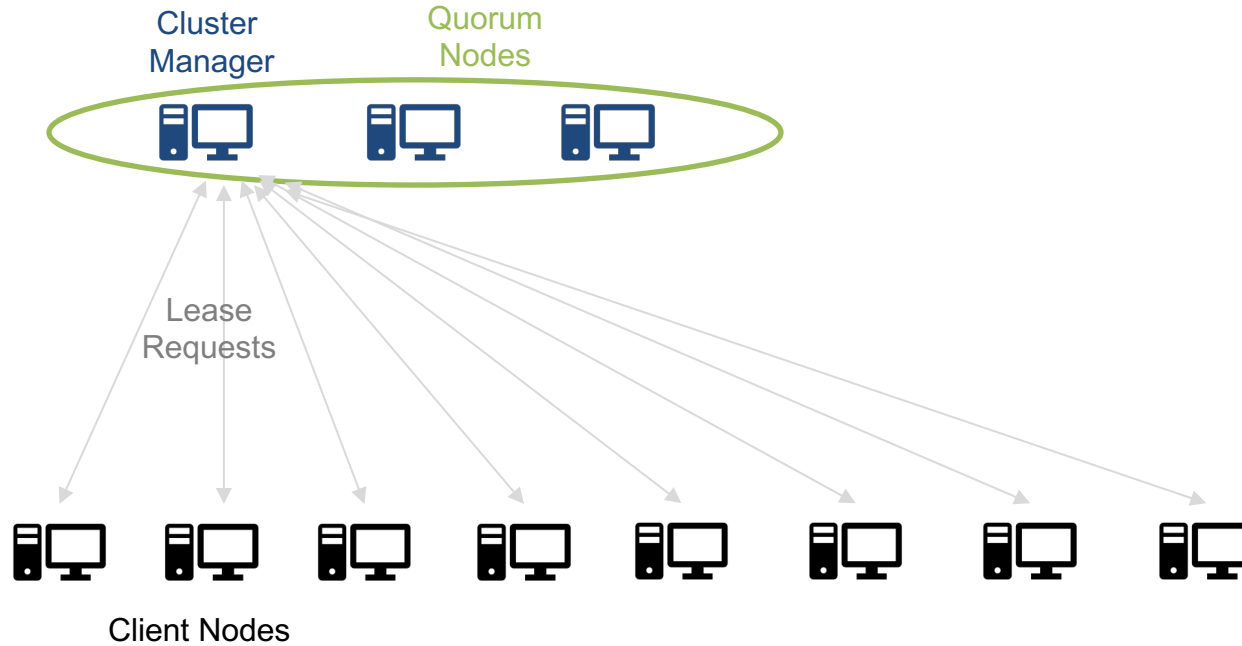# Spectrum Scale Network Communication

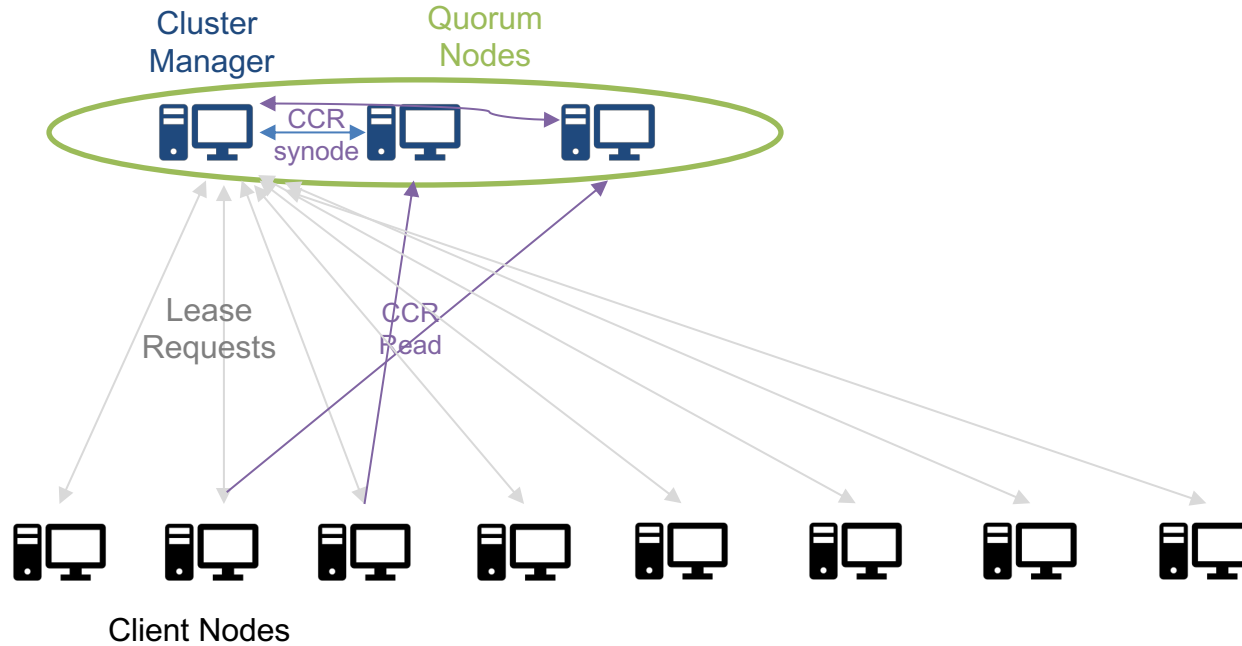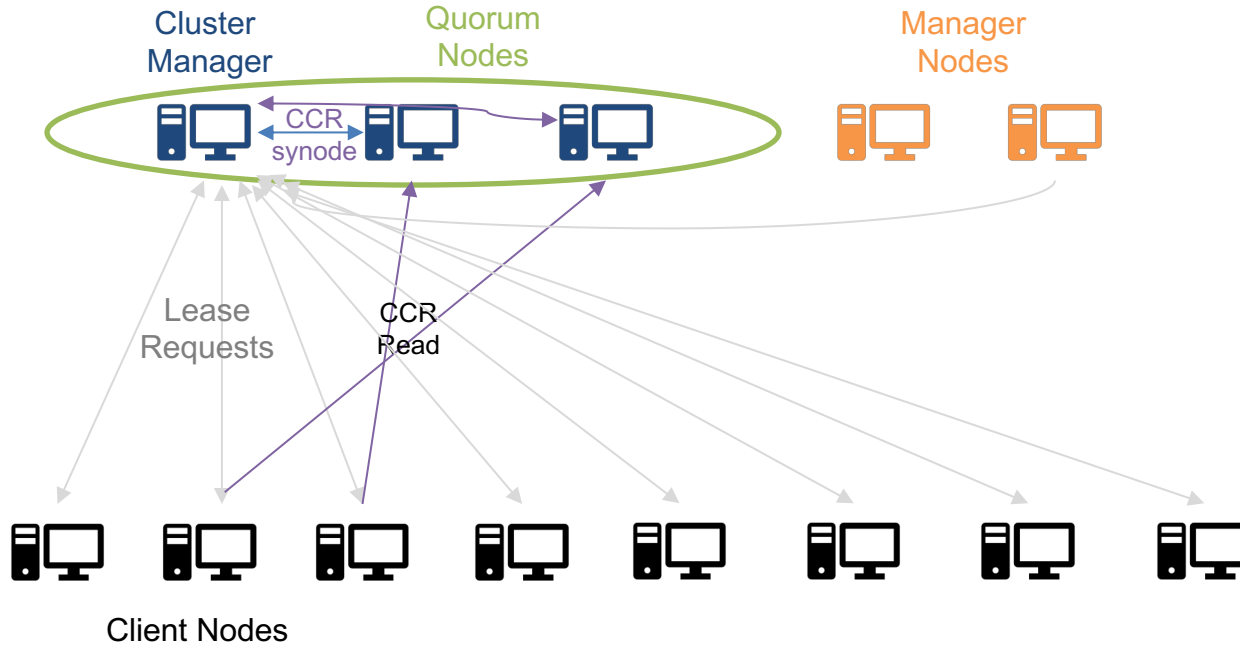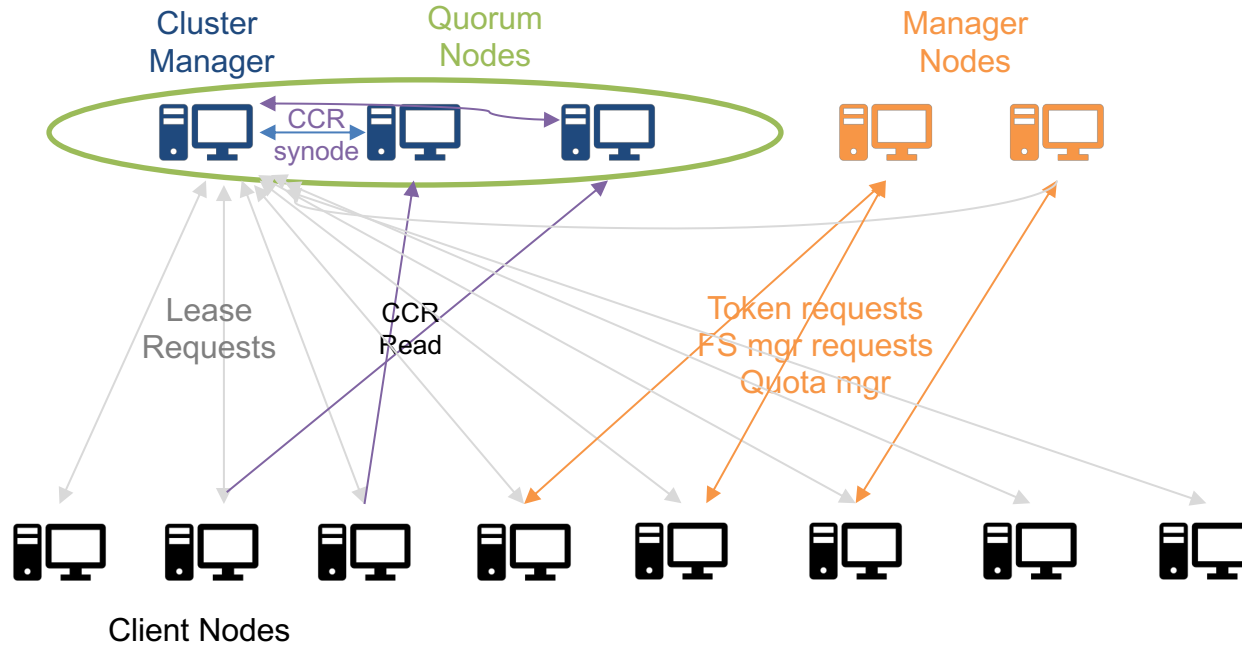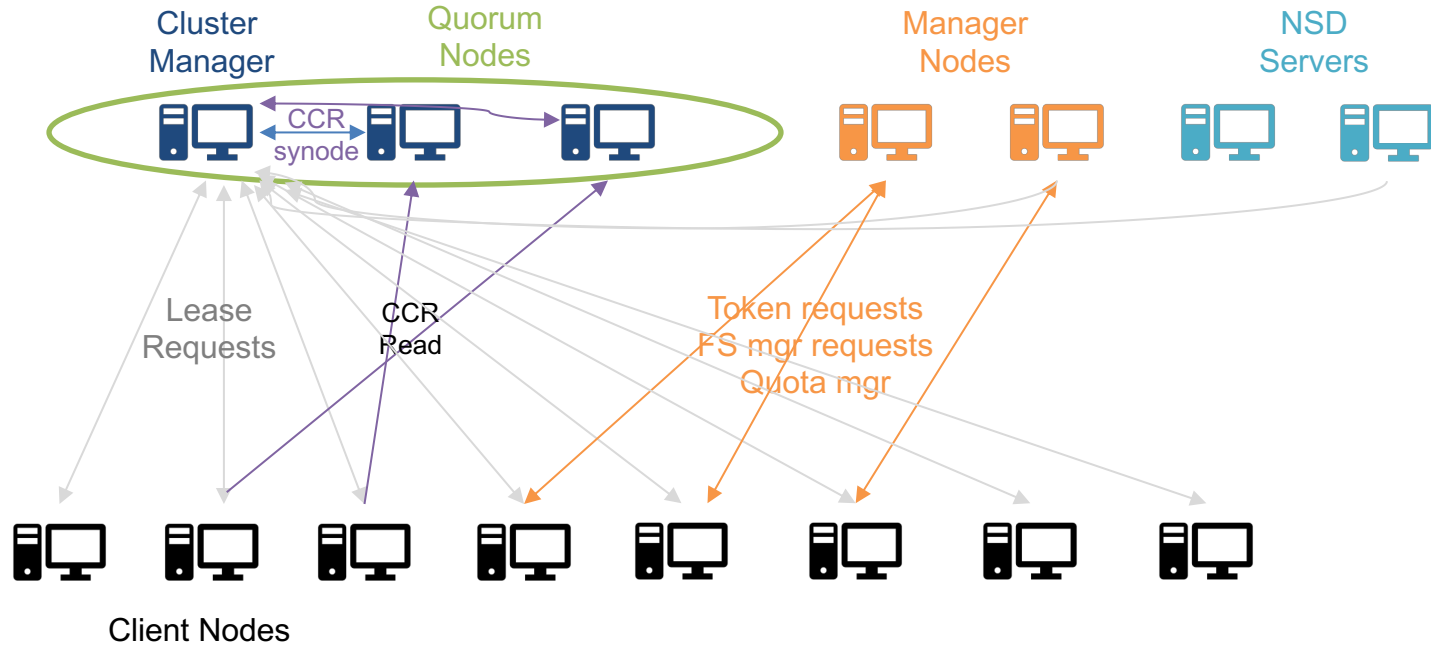# Spectrum Scale Network Communication

Spectrum Scale Network Communication
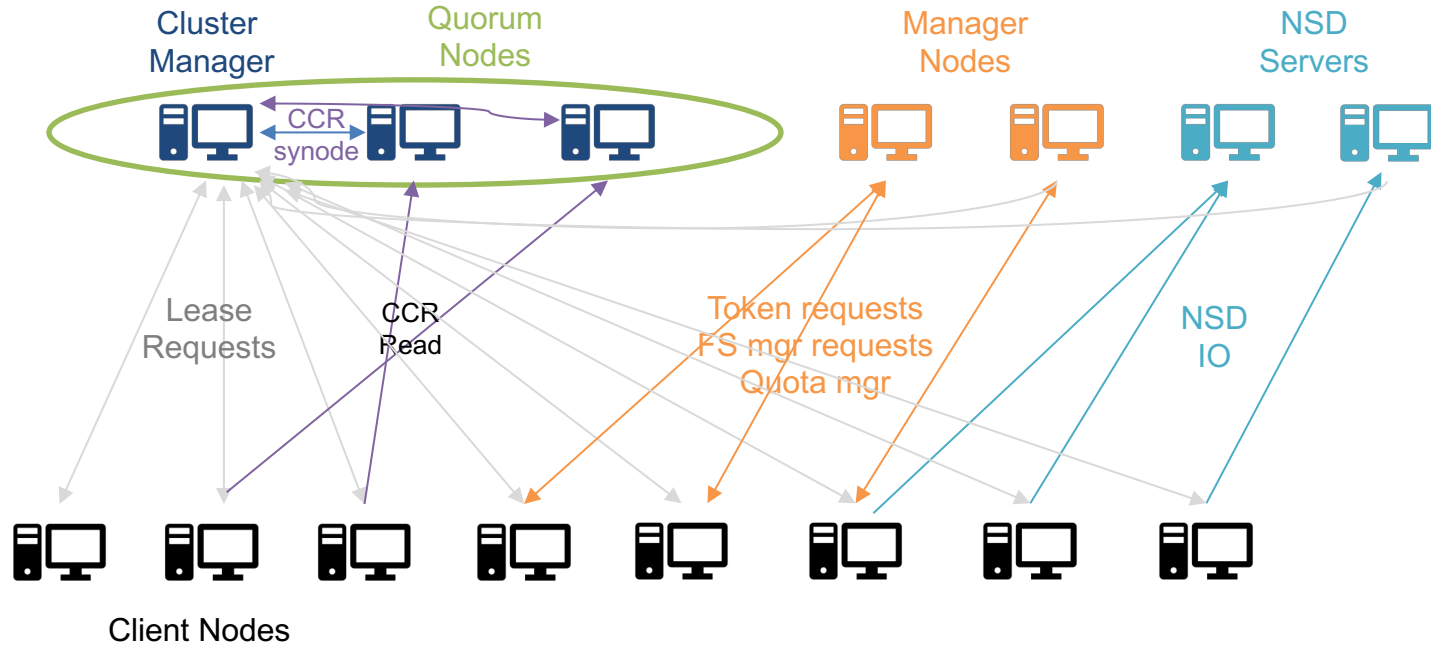
Spectrum Scale Network Communication
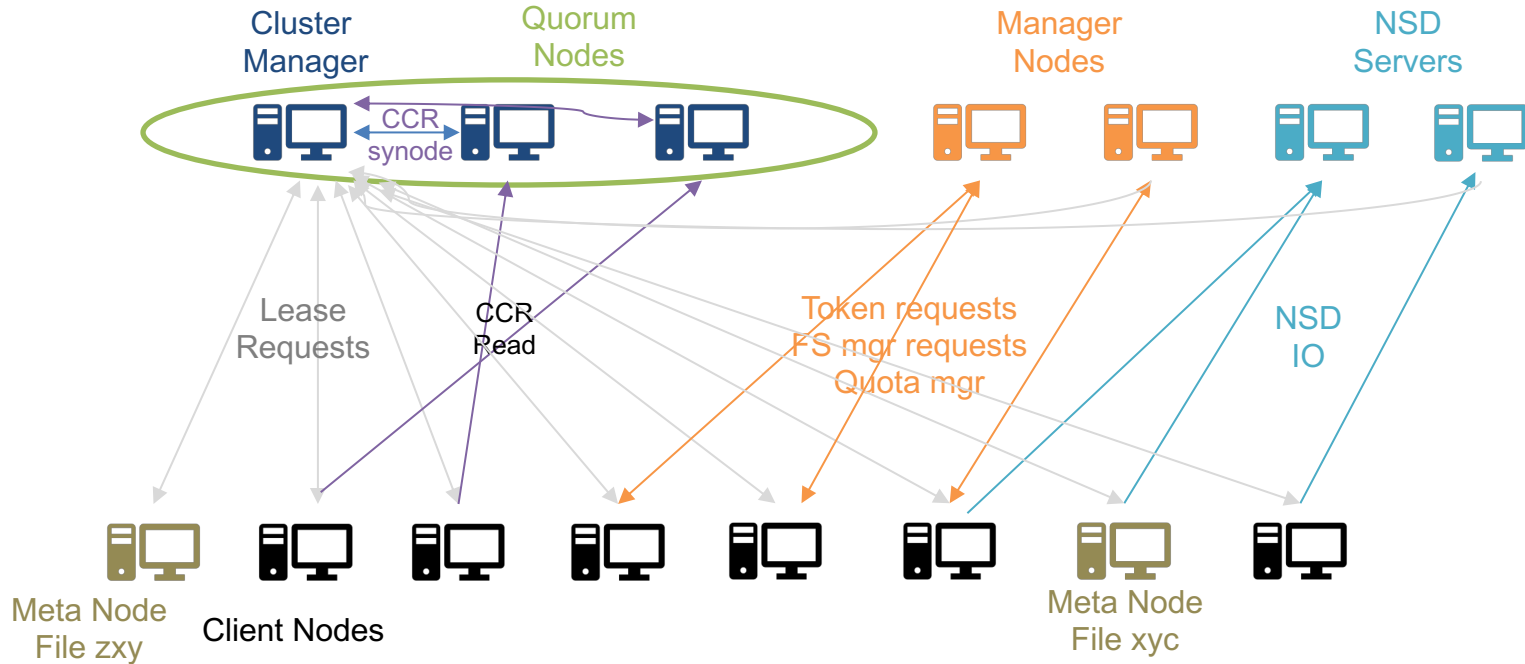
Spectrum Scale Network Communication
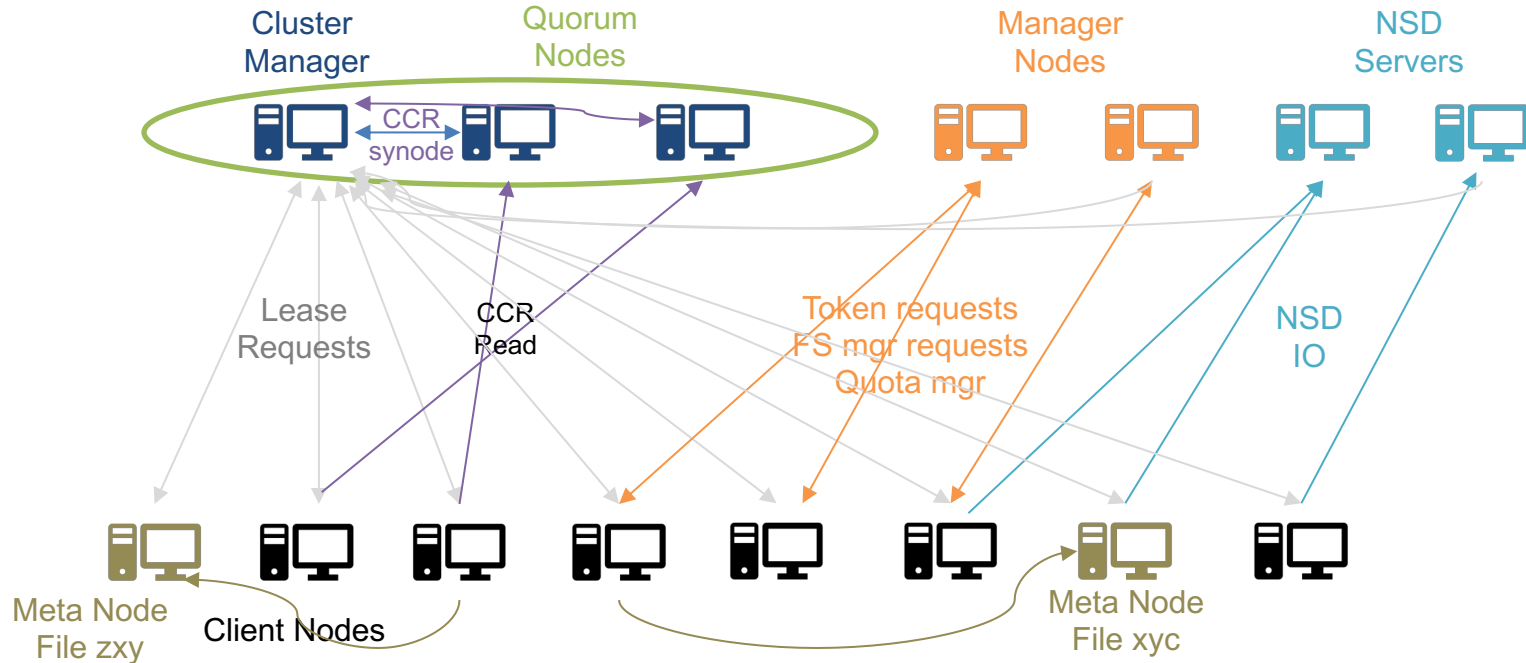
Spectrum Scale Network Communication
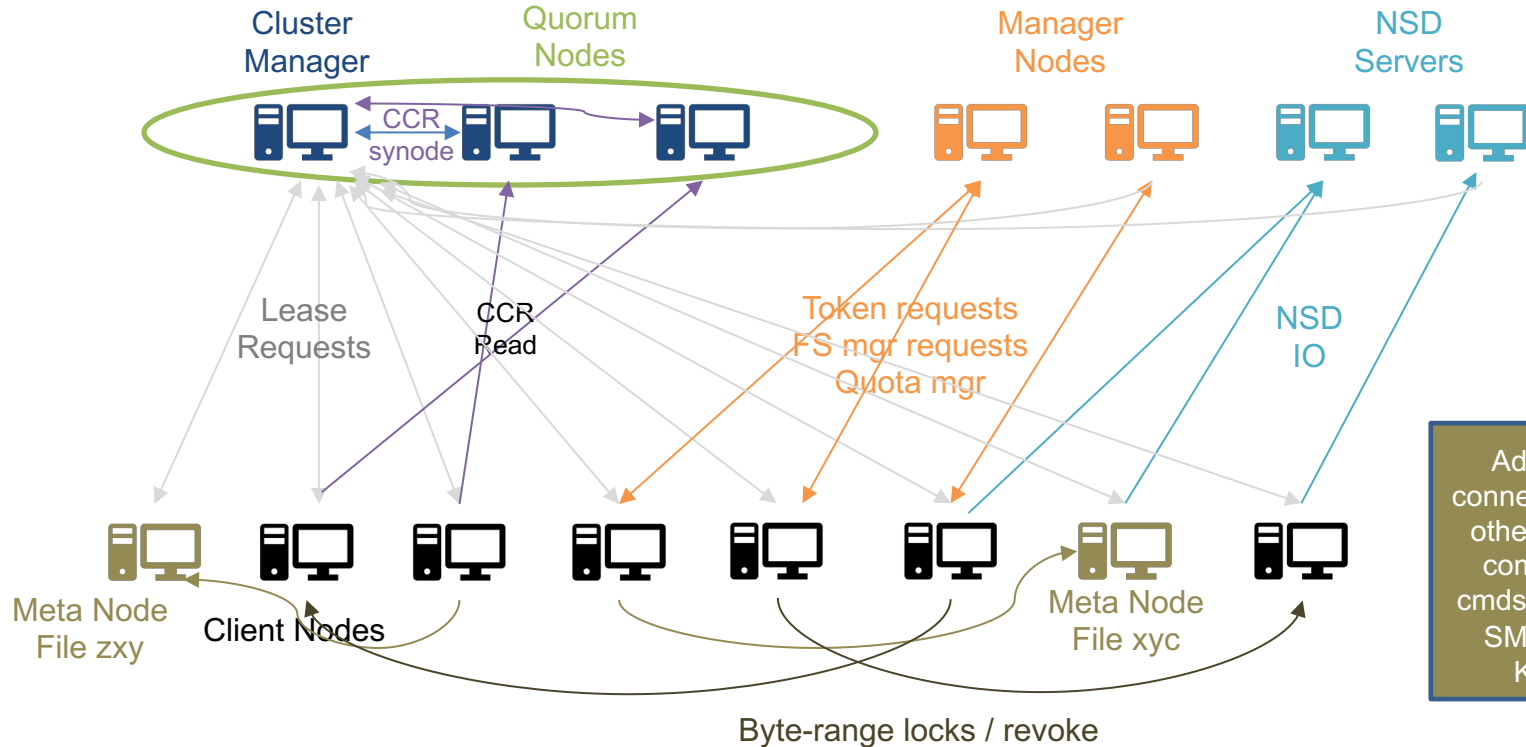
Blocked network communication between two nodes = expel

- If node cannot talk to Cluster Manager (CM), then node will be expelled
- If node A cannot talk to node B but both can talk to the Cluster Manager, then the CM will decide which node to expel (A or B):
  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
  6. Otherwise, expel whoever joined the cluster more recently.
  - A custom callout script can be configured to customize the behavior

Node expels are often caused by instable network infrastructure or invalid firewall configurations !

Use mmnetverify to test network connectivity

mmnetverify can test any-to-any node connectivity
- Supports many different operations (connectivity, port, protocol, data, etc.)
- -v = additional verbose output

```
#> mmnetverify connectivity
  md-12 checking communication with node md-13.
  Operation resolution: Success.
  Operation ping: Success.
  Operation shell: Success.
  Operation copy: Success.
  ….
#>  mmnetverify port –N all
  md-12 checking communication with node md-13.
  Operation daemon-port: Success.
  Operation sdrserv-port: Success.
  Operation tsccmd-port: Success.
  md-13 checking communication with node md-12
  ….
```

*tsccmd-port: picks one port out of the tscCmdPortRange (used by admin cmds)

Spectrum Scale Network Communication - Multicluster
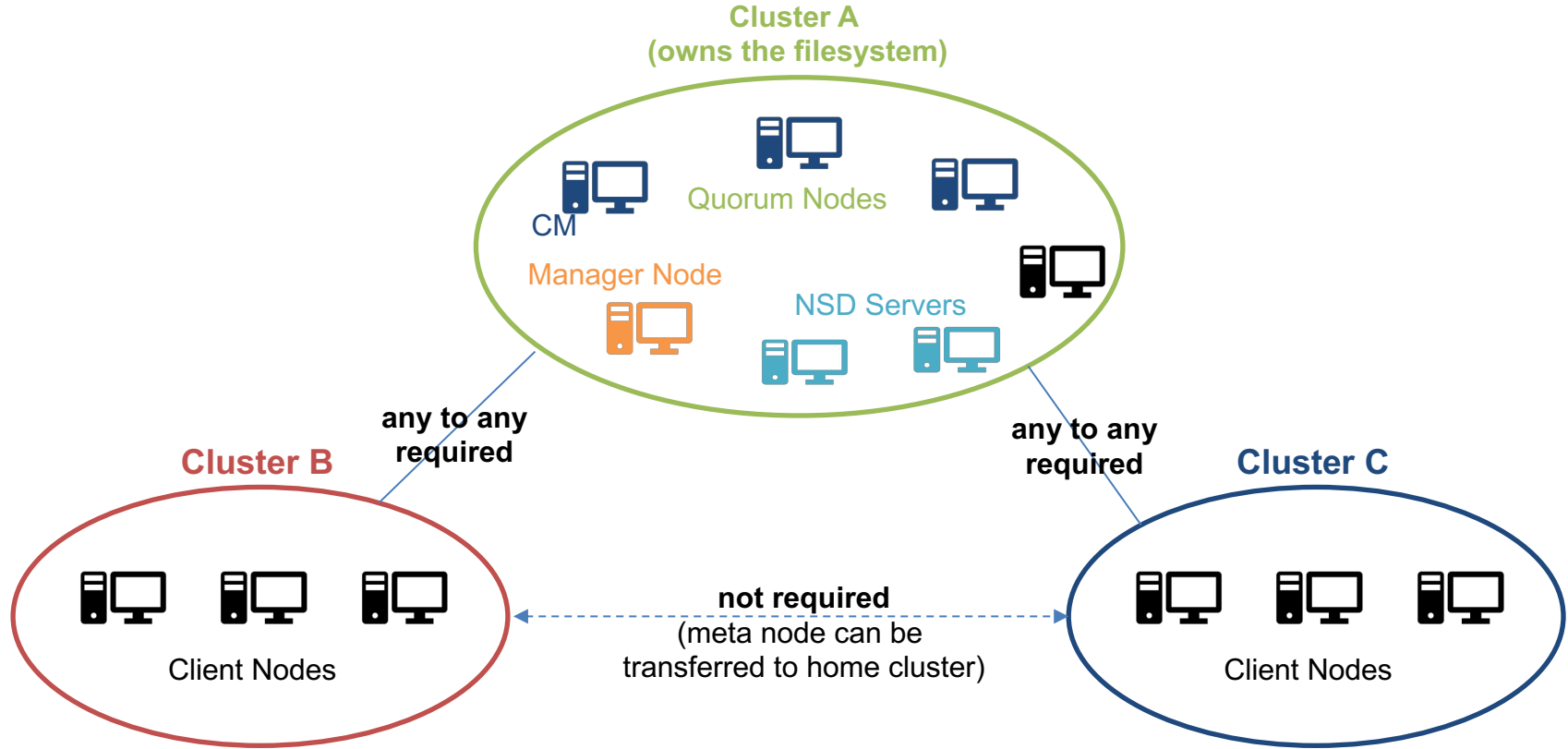
mmnetverify multi-cluster check

- mmnetverify can test remote-cluster connectivity
  - --cluster <remote cluster name> <contact_node>,…

```
#> mmnetverify remote-cluster -v
md-11 checking cluster communications.
  Checking that remote cluster nodes are reachable.
    Obtaining remote cluster configuration.
    Checking connectivity with node-22.novalocal of cluster gpfs-cluster-2.novalocal.
    Pinging node-22.novalocal of cluster gpfs-cluster-2.novalocal (10.0.100.28).
    Ping time for node-22.novalocal of cluster gpfs-cluster-2.novalocal (10.0.100.28): 0.000853 seconds.
    Node node-22.novalocal of cluster gpfs-cluster-2.novalocal is not active.
    Checking connectivity with node-21.novalocal of cluster gpfs-cluster-2.novalocal.
    Pinging node-21.novalocal of cluster gpfs-cluster-2.novalocal (10.0.100.27).
    Ping time for node-21.novalocal of cluster gpfs-cluster-2.novalocal (10.0.100.27): 0.000363 seconds.
    Node node-21.novalocal of cluster gpfs-cluster-2.novalocal is not active.
  Operation remote-cluster: Success.
```

- If ssh is not allowed to remote cluster contact nodes, start mmnetverify daemon
  - *mmnetverify --remote-cluster-daemon [--remote-cluster-port portNumber]*

CLUSTER HEALTH STATE

Cluster Health State

Problem: Where to start with checking the cluster health ?

Need a single entry point to show the health state of the entire cluster and the ability to drill down into the details.

Recommended approach:

1) **Run „*mmhealth cluster show*" to get a consolidated cluster state view**

```
#> mmhealth cluster show
Component          Total     Failed    Degraded   Healthy    Other
----------------------------------------------------------------------------------------------
FILESYSTEM          2          0          1          1          0
CESIP               1          0          1          0          0
FILEAUDITLOG        4          3          0          0          1
```

Cluster Health State

2) Drill down into cluster level details using *"mmhealth cluster show <component>"*

- <component> can also be „node" to see the status of each node
- Add –v option to see sub-components of the node

```
#> mmhealth cluster show node -v
Component        Node          Status        Reasons
---------------------------------------------------------------------------------------------
NODE         md-11.novalocal  TIPS          auditc_service_failed, unmounted_fs_check, gpfs_maxstatcache_low
 PERFMON                      HEALTHY       -
 GUI                         HEALTHY       -
 GPFS                        TIPS          gpfs_maxstatcache_low
 FILEAUDITLOG                FAILED        auditc_service_failed
 FILESYSTEM                  DEGRADED      unmounted_fs_check
 NETWORK                     HEALTHY       -
NODE         md-12.novalocal  HEALTHY       -
 GPFS                        HEALTHY       -
 PERFMON                     HEALTHY       -
 …
```

* Permanently hide Tips using „*mmhealth event hide <eventname>*"

Cluster Health State

3) Check details on node level running „*mmhealth node show -v*" on the node

- Use –N <nodename> to run command on a particular node
- Add –v option to see sub-components and entities

```
#> # mmhealth node show
Node name:      md-12.novalocal
Node status:    TIPS
Status Change:  1 min. ago


Component      Status       Status Change    Reasons
-------------------------------------------------------------------------------------------
GPFS           TIPS         1 min. ago       gpfs_maxstatcache_low, gpfs_pagepool_small
NETWORK        HEALTHY      1 min. ago       -
FILESYSTEM     DEGRADED     1 min. ago       unmounted_fs_check(gpfs0)
CES            TIPS         1 min. ago       ip_not_hostable
CESIP          DEGRADED     1 min. ago       ces_ips_unassigned
FILEAUDIT      FAILED       1 min. ago       auditc_service_failed(gpfs0)
PERFMON        HEALTHY      1 min. ago       -
```

Cluster Health State

3) Check details of failed component by running „*mmhealth node show <component>*"

- Use –N <nodename> to run command on a particular node
- Add –v option to see sub-components and entities

```
#> mmhealth node show gpfs

Node name:    md-12.novalocal

Component     Status       Status Change     Reasons
------------------------------------------------------------------------------------
GPFS          TIPS         27 min. ago       gpfs_maxstatcache_low, gpfs_pagepool_small


Event                        Parameter     Severity   Active Since    Event Message
------------------------------------------------------------------------------------------------
gpfs_maxstatcache_low        GPFS          TIP        27 min. ago     The GPFS maxStatCache is lower than the maxFilesToCache setting.
gpfs_pagepool_small          GPFS          TIP        27 min. ago     The GPFS pagepool is smaller than or equal to 1G.
```
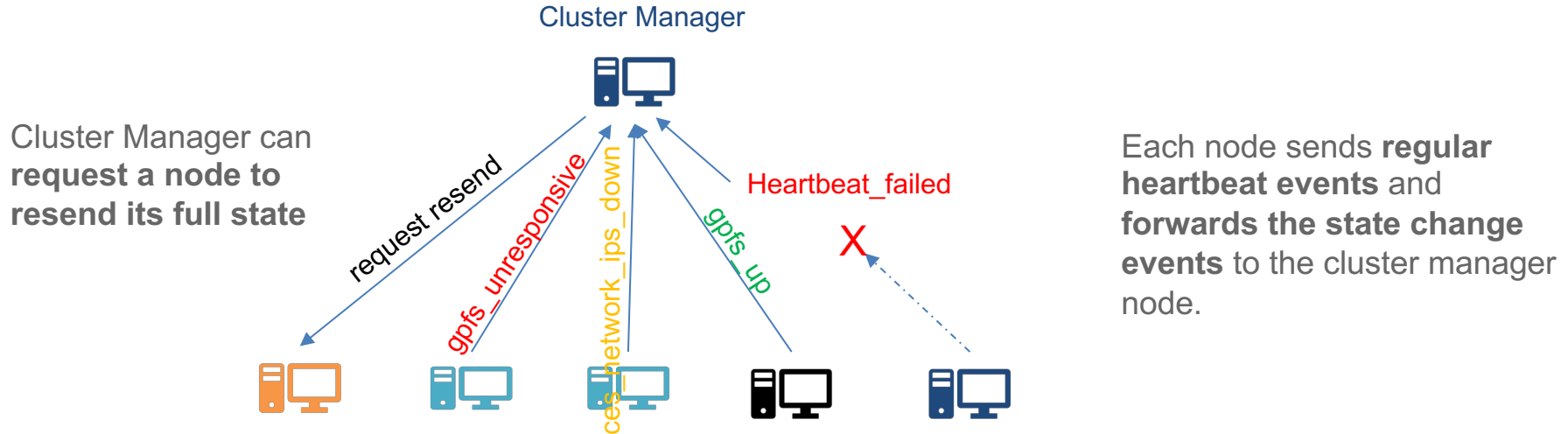
Cluster Health State inconsistencies

Problem:

Sometimes the cluster state shows different results than the node state.

E.g. SMB state FAILED on cluster level but already HEALTHY on node level.

To answer this we need to explain how the cluster state consolidation works.

Cluster Health State

A **consolidated cluster state view** is build on the cluster manager

Cluster Manager

Cluster Manager can **request a node to resend its full state**

request resend

gpfs_unresponsive

ces_network_ips_down

gpfs_up

Heartbeat_failed

X

Each node sends **regular heartbeat events** and **forwards the state change events** to the cluster manager node.

Health monitoring runs on any GPFS node (Linux, AIX), monitors Spectrum Scale components and checks for common failure conditions (e.g. local fs full)

Cluster Health State

Summary:

- Event forwarding to cluster state manager can be delayed or interrupted when network issues happen, but should cleanup itself and is **"eventually consistent"**
- If in doubt, local node state (mmhealth node show) is always right
- When the cluster state manager detects a inconsistency or when a cluster manager failover happens, it will **request the nodes to resend the full state.**
- In case of permanent network issues or node outage, **heartbeat_missing event** will indicate that the node is not sending any event for the last 30min.
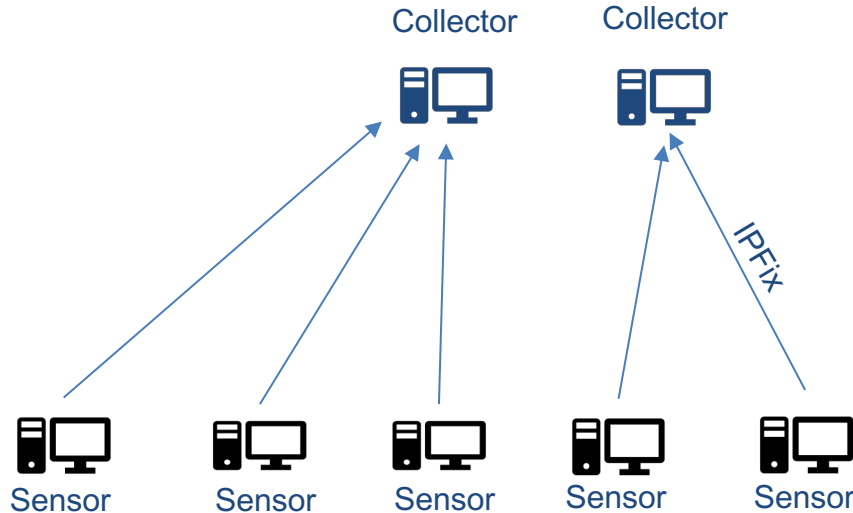
- A manual resync can be triggered:

```
#>  mmhealth node show --resync
Resend events to cluster state manager
```

PERFORMANCE MONITORING
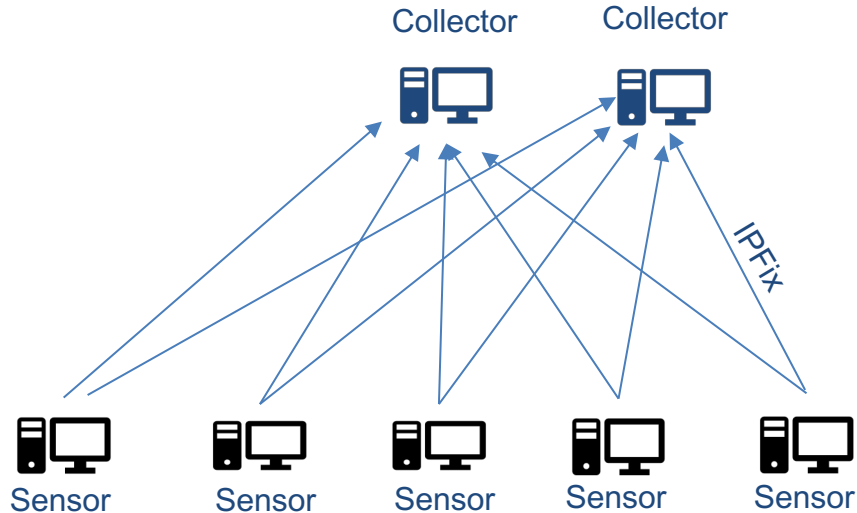
Performance and Usage Monitoring

Collectors keep a in-memory time series database with
performance data of the individual nodes

Sensors gather the performance metrics from Spectrum Scale components, Linux, etc. and send
it through the IPFix protocol to collectors

Performance and Usage Monitoring

Collectors keep a in-memory time series database with
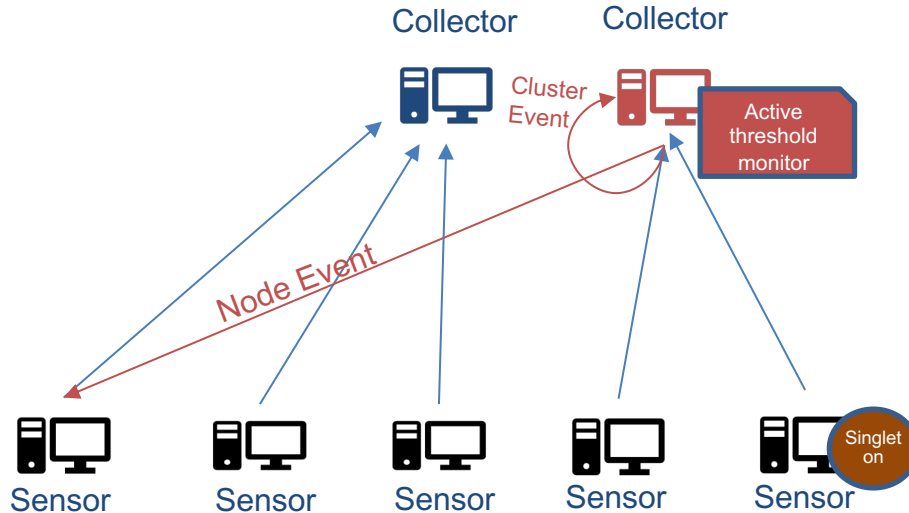performance data of the individual nodes

Collector   Collector

Setting „**Collector Redundancy**" to 2
will ensure that the performance data
is mirrored to two collectors

IPFix

Sensor   Sensor   Sensor   Sensor   Sensor

Sensors gather the performance metrics from Spectrum Scale components, Linux, etc. and send
it through the IPFix protocol to collectors

Performance and Usage Thresholds

Custom thresholds can be set on any performance and usage metric using mmhealth cmd or GUI.

Collector   Collector

Cluster Event

Active threshold monitor

Node Event

Sensor   Sensor   Sensor   Sensor   Sensor

Singlet on

An „active thresholds collector" is choosen out of the collector nodes.

„*mmhealth cluster show threshold –v*" will show which node it is.

Node specific threshold events are forwarded to the corresponding node and shown there in mmhealth. Cluster wide threshold events will be shown on the active threshold monitor node.

PROTOCOL SERVICEABILITY

CES Clusters with multiple subnets / VLANs

CES supports hosting of multiple subnets or VLANs.
Inhomogeneous network connectivity can mask problems

Example:
Node A,B,C can host 172.10.x.x ip range
Node C,D can host 192.168.x.x ip range

Without CES groups defined :
- CES will try to assign any IP to any node (fails if node has no connecetivity)
- CES can not know if a node is supposed to have connectivity or not
  - e.g. Failure of one network link on node C might remain undiscovered

Recommendation: Always define CES Network groups if multiple subnets/VLANs are used.

CES Cluster

A B C D

172.10.0.100/24 192.168.0.10/24
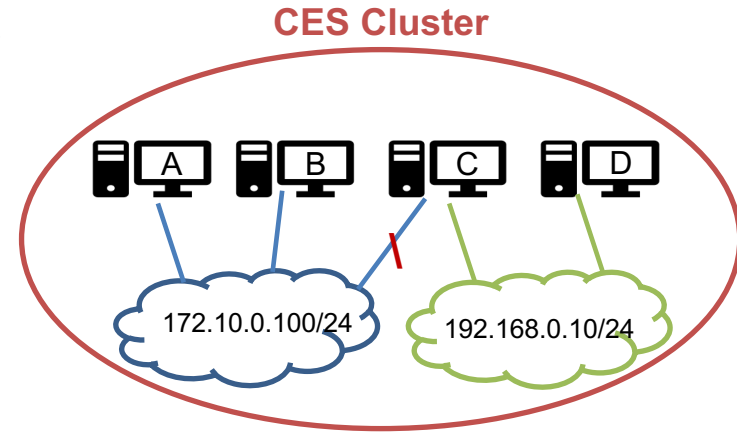
CES Clusters with multiple subnets / VLANs

Health monitoring will detect if a particular CES IP cannot be hosted <u>anywhere</u> in the cluster. *mmces address list –full-list* will report nodes which cannot host a particular CES IP

```
#> mmhealth cluster show cesip
Component     Node           Status         Reasons
------------------------------------------------------------------------------
CESIP         md-12.novalocal    DEGRADED      ces_ips_unassigned


#> mmces address list --full-list
cesAddress      cesNode         attributes  cesGroup   preferredNode  unhostableNodes
--------------- --------------- ----------- ---------- -------------- ----------------
172.0.0.12        unassigned      none        none       none           md-12.novalocal,md-13.novalocal,md-14.novalocal
192.168.0.150  md-14.novalocal  none        none       none           none
192.168.0.151  md-13.novalocal  none        none       none           none
192.168.0.152  md-14.novalocal  none        none       none           none
192.168.0.153  md-14.novalocal  none        none       none           md-12.novalocal
```

NFS Ganesha conflicts with kNFS lockd

Problems:

- Doing a local nfsv3 mount on a CES node will start the kNFS lockd. This will prevent Ganesha lock manager to work properly.
- A running kNFS server conflicts with Ganesha NFS (mountd)

mmhealth will detect those conditions by looking at rpcinfo port registration:

```
#> mmhealth node show nfs -v
Component      Status      Status Change          Reasons
-------------------------------------------------------------------------------
NFS            DEGRADED     2019-03-12 16:19:09     rpc_nlockmgr_inv_user, rpc_mountd_inv_user
```

Maintenance of CES filesystems (ces suspend/resume)

Problem: Filesystems used by CES could not be easily unmounted to do maintenance (e.g. fsck)

CES suspend function has been extended:

   1) to optionally stop all running CES services (closing file handles on the fs)

   2) CES services will not be started after a reboot if node is suspended

```
#> mmces node suspend –stop
NFS: service succesfully stopped.
SMB: service succesfully stopped.
CTDB: service succesfully stopped.
Node md-12 now in suspended state.

-> Now do your maintenance operation  (e.g. mmfsck, reboot,..)

#> mmces node resume --start
CTDB: service succesfully started.
SMB: service succesfully started.
NFS: service succesfully started.
Node md-12 no longer in suspended state.
```

CES IP Failover

Problem: Sometimes hard to understand the reason for a CES IP failover.

Improvement: Improved failure notification for CES IP failover events.

> **#> mmhealth node eventlog | grep move_cesips**
>
> 2019-03-12 15:15:35.438213 CET move_cesips_info INFO A move request for ip addresses was executed. **Reason: …**

Possible reasons are:

Move_address_due_to_balanced_load_policy, node_affinity_policy, even_coverage_policy.

enableIP_ip_not_assigned

enableIP_try_to_enable

disableIP_link_down

disableIP_interface_not_active

Remove_all_IPs_from_starting_node

Remove_all_IPs_from_dead_node

mmces node suspend / resume

Release_address_due_to_network_shutdown_detected_by_mmhealth.

Release_address_due_to_network_shutdown_detected_by_ces.

QUESTIONS ?

THANK YOU

Disclaimer

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.