

Cluster Maintenance

🕒 10 minute read

Intro

This post teaches you how to monitor the cluster’s health, manage cluster resources, and perform ongoing maintenance and troubleshooting to keep the cluster in top working order.


Monitor and Manage the Cluster

Monitor the Cluster


Monitoring Tools

Tools


The Cluster can be monitored using MapR tools like MCS, CLI and external tools like Ganglia and Nagios.




MCS



CLI and REST interface



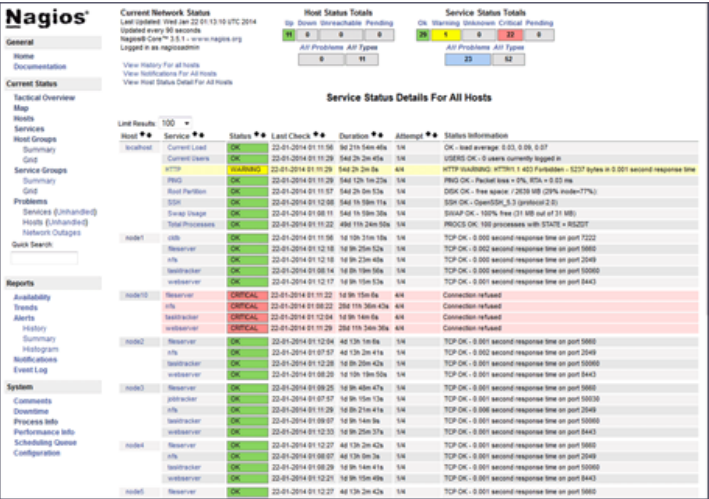
Nagios



Ganglia

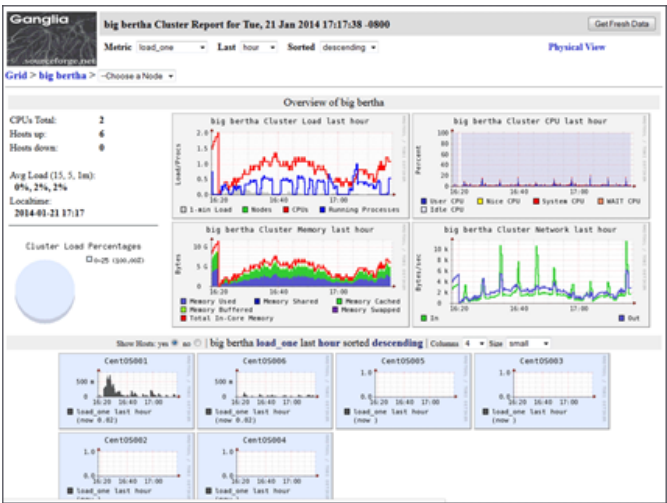
Nagios (<https://www.nagios.org/>)

- Nagios can monitor network and host resources
- Configuration script generator
- Conduit for alarms to go through standard escalation
- Integrates other sources of alarms aside from MapR builtin alarms.



Ganglia (<http://ganglia.sourceforge.net>)

- Monitors Clusters and grids
- Scalable, distributed system monitoring tool
- Unlike Nagios, Ganglia is not used to raise alarms but to report cluster health.
- Show cluster status in a given time period
- Install on CLDB nodes to gather metrics from CLDB like cpu, memory, active server nodes and volumes created on the node and overall cluster.



Maintenance Tasks

1. Overview
 - Regular maintenance
 - Replace disk(s)
 - Maintain nodes
 - Upgrade software
 - Clean out log files
 - Design changes
 - Make cluster HA
 - Expand cluster
 - Balance services

2. Sample Schedule Shown here is a possible Monitoring and Maintenance Schedule :

Daily	Weekly	Monthly	Quarterly
Check alarms	Check logs	Clean up logs	Audit storage use
Check cluster health	Verify snapshots	Re-evaluate node and volume topology	Upgrade software if needed
Add/remove users due to personnel changes	Remove old snapshots	MFS balancing	Check cluster capacity
Check for required hardware maintenance	Verify performance	Verify performance	

Configure and Respond to Alarms

MapR Alarms

About Alarms

- Alarms are raised to notify users of a problem
- When an alarm is raised alerts are posted to the MCS
 - Can also be sent via email
- Configure custom email addresses for specific alerts to accountable entities.

Alarm Classifications

Alarms can be classified into :

1. Cluster Alarms :
- Cluster Space
 - Licensing Issues
 - CLDB problems.

2. Volume Alarms :
- Specific to a Volume.
 - Snapshots and Mirroring failures
 - Under replicated and Unavailable data
 - Volume topology problem.

3. Node Alarms :
- Specific to a Node
 - Disk failures
 - Time-skew problems
 - Failed Services

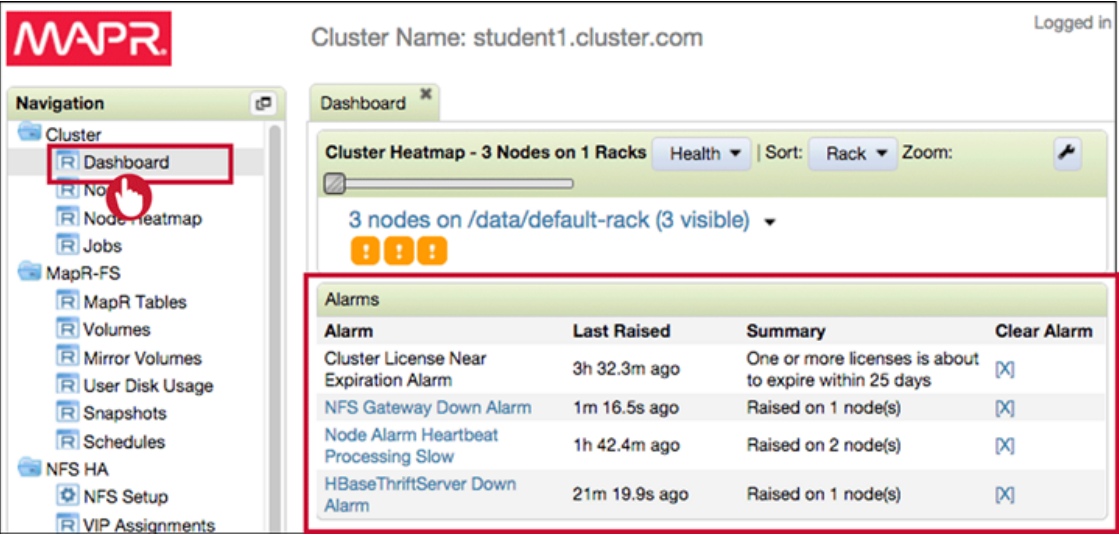
\$ maprccli alarm names



View Alarms

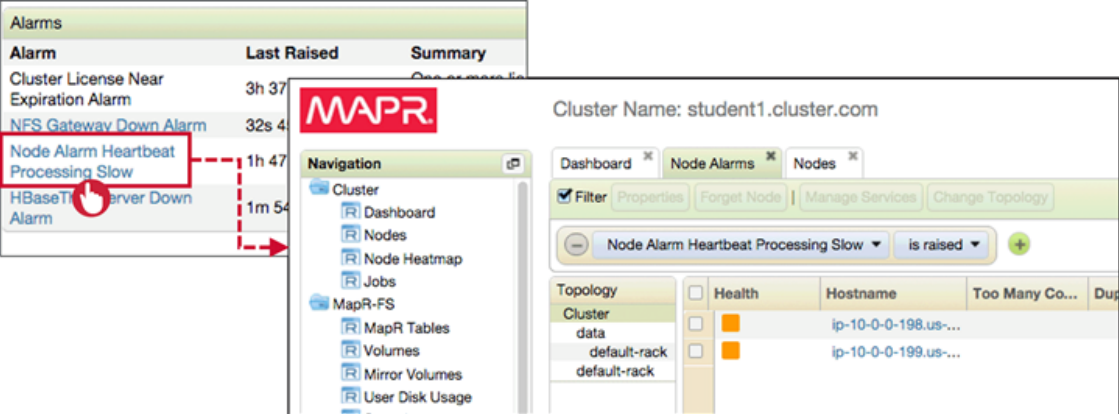
Alarms can be viewed either in Dashboard or MapR CLI. To see all alarms:

\$ maprccli alarm list



View Alarm Details

Vie alarm details in Dashboard and investigate the problems , restart services or nodes affected.



Configure Alerts

Overview

Before alarms are set configure SMTP and user email addresses.

1. Set up SMTP

Configure SMTP

*Provider

Gmail

*SMTP Server

smtp.gmail.com

☒ This server requires an encrypted connection (SSL)

*SMTP Port

587

*Sender's Full Name

John Smith

*Sender's Email Address

Enter a username and password if your SMTP server requires authentication

Sender's User Name

Sender's SMTP Password

Test SMTP Connection

OK

Cancel

2. Configure email addresses of cluster users

Configure Email Addresses

Use company domain: user@my_company.com

Use LDAP:

LDAP Server:

LDAP Port:

LDAP bind domain:

LDAP bind domain password:

LDAP base domain:

UID attribute:

Mail attribute:

OK

Cancel

3. Set up custom alerts

Alerts

Alarm Name	Standard Notification	Additional Email Addresses
Cluster Upgrade In Progress	<input checked="" type="checkbox"/>	
Cluster Unassigned Virtual IPs	<input checked="" type="checkbox"/>	
Cluster License Near Expiration	<input checked="" type="checkbox"/>	
Cluster License Expired	<input checked="" type="checkbox"/>	
Cluster Almost Full	<input checked="" type="checkbox"/>	
Cluster Full	<input checked="" type="checkbox"/>	
Node Count Exceeds License Maximum	<input checked="" type="checkbox"/>	
New Cluster Features Disabled	<input checked="" type="checkbox"/>	
Volume Snapshot Failure(s)	<input checked="" type="checkbox"/>	
Volume Mirror Failure(s)	<input checked="" type="checkbox"/>	
Volume Low Data Replication	<input checked="" type="checkbox"/>	
Volume Data Unavailable	<input checked="" type="checkbox"/>	

OK

Cancel

MCS and CLI

```
$ maprcli alarm config
```

Alarms

Node Alarms

Volume Alarms

User/Group Alarms

Alerts

System Settings

Email Addresses

Permissions

Quota Defaults

Alerts

Alarm Name	Standard Notification	Additional Email Addresses
Cluster Upgrade In Progress	<input checked="" type="checkbox"/>	
Cluster Unassigned Virtual IPs	<input checked="" type="checkbox"/>	
Cluster License Near Expiration	<input checked="" type="checkbox"/>	
Cluster License Expired	<input checked="" type="checkbox"/>	
Cluster Almost Full	<input checked="" type="checkbox"/>	
Cluster Full	<input checked="" type="checkbox"/>	
Node Count Exceeds License	<input checked="" type="checkbox"/>	

Standard alerts send email to the entity involved like Volume Quota Alarms to Volume Accountable Entity. Additional address can be defined for alternative notifications to other target users.

Test Alerts

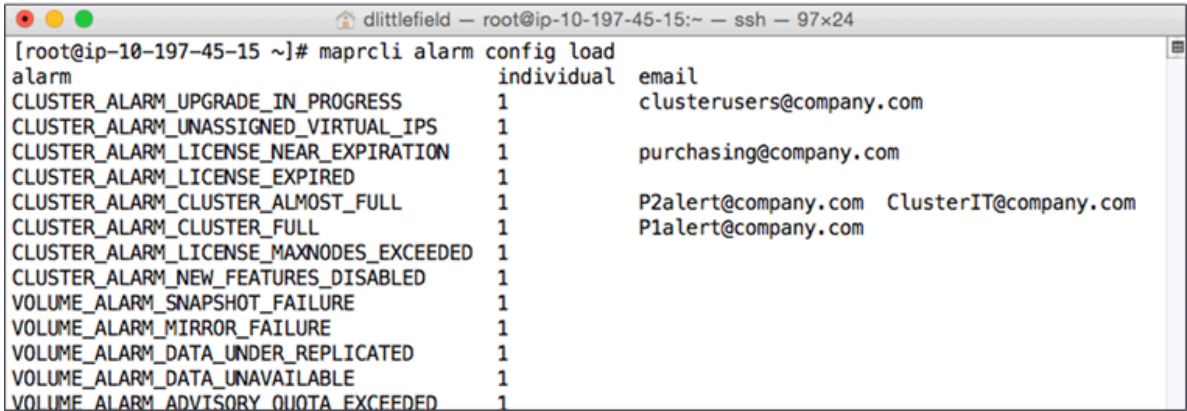
It's a good idea to test alarms because Email address and SMTP are not configured when alarms were defined.

- Raises the specified alarm
- Description appears in maprcli alarm list output and MCS

```
maprcli alarm raise
  -alarm <name>
  [-cluster <cluster>]
  [-entity <cluster/entity/host/node/volume>]
  [-description "<reason>"]
```

Verify Configuration

```
$ maprcli alarm config load
```



Respond to Alarms

Determine Urgency

- Alarms differ in urgency
 - Some resolve on their own
 - Some are early warnings
- Determine:
 - Underlying cause
 - What action is needed
 - How soon action should be taken
- Many alarms are self-explanatory
- Refer to [documentation](http://doc.mapr.com/display/MapR/Alarms+Reference) (http://doc.mapr.com/display/MapR/Alarms+Reference) for details



Example: Normal Operation

CLUSTER_ALARM_UPGRADE_IN_PROGRESS

- Normal operation
- Alarm will clear when upgrade is done



Example: Schedule Action

ALARM_ADVISORYQUOTA_EXCEEDED

- Urgency depends on:
 - Rate of data growth
 - Differential between advisory and hard quotas
 - Which volume or AE



Example: Degraded Performance

NODE_ALARM_SERVICE_*_DOWN

- May impact a single node, or entire cluster
- Criticality varies by service



Example: Urgent Situation

NODE_ALARM_TIME_SKEW

- Alarm raised for time skew
 - Maximum differential: 20 seconds
 - Recommended differential: <500 ms
- System time critical for ZooKeeper functionality
 - Use NTP



Balance Cluster Resources

As the cluster keeps growing and changing it is important to balance location and replication of data. There are two balancers available :

- Disk balancer
 - Redistributes data
- Role balancer
 - Changes container replication role

Balancing starts 30 minutes after CLDB startup, by default

- change with `cldb.balancer.startup.interval.sec`

Disk and Role Balancers

Disk Balancer

- Moves containers between storage pools
- Ensures disk usage on nodes is similar

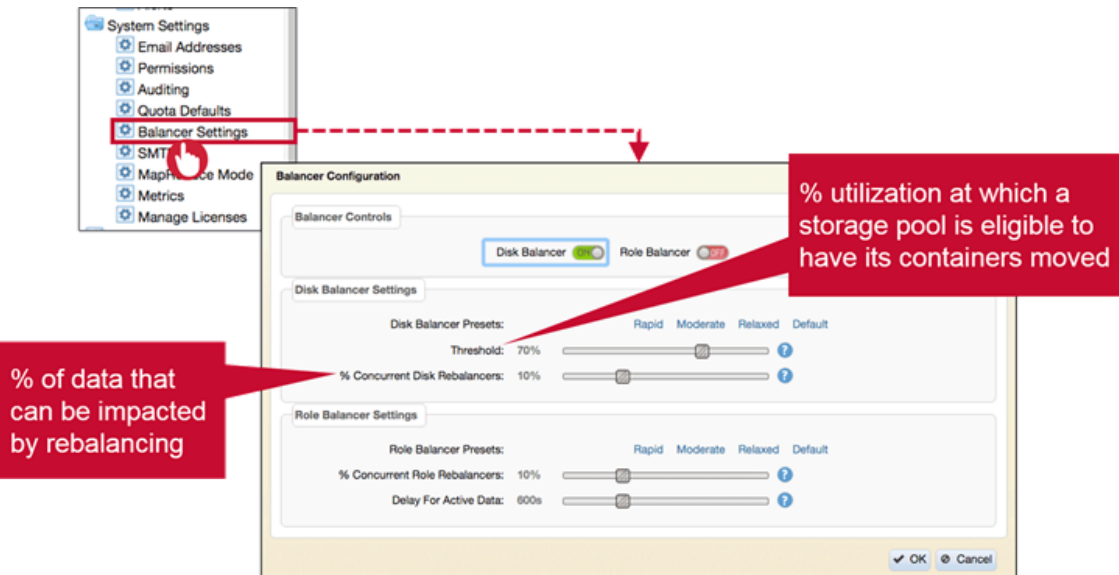
Disk balancer is particularly useful when new nodes or rack is added to matured cluster.



Disk Balancer Details

There are two primary disk balancer settings :

1. Threshold defines a percentage of utilization at which a storage pool is eligible for rebalancing. Can be set between 10%-99%
2. Concurrent disk rebalancers define percentage of data can be actively rebalanced at a point of time. This prevents rebalancing from disrupting normal cluster operations. Can be set from 2%-30%



Role Balancer

- Changes replication roles of containers to ensure all nodes have equal share of master containers.
- Distributes master containers among nodes
- Evenly distributing replication roles balances network bandwidth and also spreads load for writes across cluster.
- Write operation on MapRFS and Read on MapRDB are performed on master container. So role balancing also distributes table reads.



Role Balancer Details

There are two primary role balancer settings

System Settings

Email Addresses

Permissions

Auditing

Quota Defaults

Balancer Settings

Storage

MapReduce Mode

Metrics

Manage Licenses

Balancer Configuration

Balancer Controls

Disk Balancer ☐ Role Balancer ☒

Disk Balancer Settings

Disk Balancer Presets: Rapid Moderate Relaxed Default

Threshold: 70%

% Concurrent Disk Rebalancers: 10%

Role Balancer Settings

Role Balancer Presets: Rapid Moderate Relaxed Default

% Concurrent Role Rebalancers: 10%

Delay For Active Data: 600s

OK

Cancel

% of data that can be impacted by rebalancing

Delay before active data sets can be rebalanced

Enabling Balancers Through the CLI

Balancing: CLI

- Turn on:

```
maprcli config save -values {"<parameter>" : "<value>"}
```

- Adjust:

```
maprcli config save -values { "cldb.balancer.disk.paused" : "0"}
maprcli config save -values { "cldb.balancer.role.paused" : "0"}
```

- Check values:

```
maprcli config load -json | grep balancer
```

Balancer Parameters

List of some of the disk and role balancing parameters

Parameter	Description
cldb.balancer.disk.paused	Set to 0 to turn on, or 1 to pause disk balancing (default = paused)
cldb.balancer.disk.threshold.percentage	% of a storage pool's space that must be used before it is eligible for balancing
cldb.balancer.disk.max.switches.in.nodes.percentage	Maximum % of total cluster data that can be impacted by disk balancing
cldb.balancer.role.paused	Set to 0 to turn on, or 1 to pause role balancing (default = paused)
cldb.balancer.role.max.switches.in.nodes.percentage	Maximum % of total cluster data that can be impacted by role balancing
cldb.balancer.role.sleep.interval.sec	Seconds data must be idle before it can be role balanced
cldb.balancer.startup.interval.sec	Number of seconds CLDB must be active before balancing begins

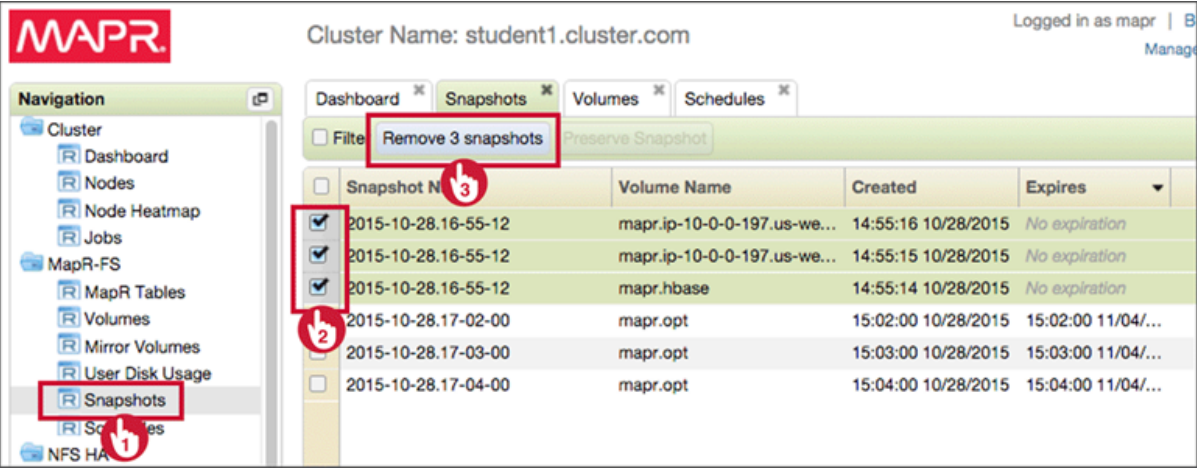
Manage Logs and Snapshots

Manage Snapshots

- Scheduled snapshots expire automatically
- Manual snapshots must be deleted
 - Check regularly (at least monthly)
- Use schedules if possible



Remove Snapshots



```
maprcli volume snapshot list -columns snapshotid -filter [expirytime=="0"]
maprcli volume snapshot remove -snapshots <ID1, ID2, ID3 ...>
```

Manage Log Files

- 1. Overview
 - Some managed automatically
 - Job logs
 - Data audit logs (if enabled)
 - Others must be cleaned out manually
 - maprcli audit logs
 - Cluster audit logs (if enabled)
- 2. Retention Times
 - Retain longer if needed for troubleshooting
 - example: job logs with local logging deleted after 3 hours by default
 - Shorten retention time if logs get too large

Add and Remove Services

Removing Services

The following is the procedure for removing a service from a node.



Stop
the service



Uninstall
the package



Run
configure.sh



Restart
Warden on node

Adding services

Adding services is even easier than removing as follows :



Install
package for service



Run
`configure.sh`



Run
`disksetup`
for Fileserver



Restart
Warden on node

Disk and Node Maintenance

Replace a Failed Disk

Disk Failure

Replacing failed disks is a normal part of cluster maintenance. Factors impacting disk failure rate:

1. Number of Disks
2. Disk Reliability
3. Data Center Environment

What Happens When a Disk Fails?



Disk Failure Information

Look at the Disk Failure Report and perform recommended recovery methods.

1. Disk Failure Report

```
##### Disk Failure Report #####

Disk      : /dev/sdd
Vendor    : [vendor]
Model Number : [model]
Serial Number : [serial]
Firmware Revision : [firmware]
Size      : [total]
Failure Reason : I/O error
Time of Failure : Fri Jan 31 12:48:00 GMT 2014
Resolution :

Please refer to MapR's online documentation at http://www.mapr.com/doc on how to handle disk failures.
In summary, run the following steps:
a. If this appears to be a software failure, go to step b.
   Otherwise, physically remove the disk /dev/sdd.
   Optionally, replace it with a new disk.
b. Run the command "maprccli disk remove -host 127.0.0.1 -disks /dev/sdd" to remove /dev/sdd from MapR-FS.
c. In addition to /dev/sdd, the above command removes all the disks that belong to the same storage pool,
   from MapR-FS. Note down the names of all removed disks.
d. Add all the above removed disks (exclude /dev/sdd) and the new disk to MapR-FS by running the command:
   "maprccli disk add -host 127.0.0.1 -disks <comma separated list of disks>"
   For example, If /dev/sdx is the new replaced disk, and /dev/sdy, /dev/sdz were removed in step c), the
   command would be:
       "maprccli disk add -host 127.0.0.1 -disks /dev/sdx,/dev/sdy,/dev/sdz"
   If there is no new disk, the command would just be:
       "maprccli disk add -host 127.0.0.1 -disks /dev/sdy,/dev/sdz"
```

2. Recovery Methods

Failure Reason	Recommended Course of Action
I/O time out	Increase the value of the <code>mfs.io.disk.timeout</code> parameter in the <code>/opt/mapr/conf/mfs.com</code> file.
No such device	Check if the disk has been renamed. If so, re-add the disk by running the disk add command.
I/O error	Test the drive.
CRC error	Run <code>fsck -n <sp> -d</code> , then bring it back online.
Slow disk	Test the drive.
GUID of disk mismatches with the one in <code>\$INSTALL_DIR/conf/disktab</code> .	After a node restart, the OS can reassign the drive labels (for example, <code>/sda</code>). Edit the <code>disktab</code> file to repair the problem.
Unknown error	Contact MapR Support

3. Alarms Raised

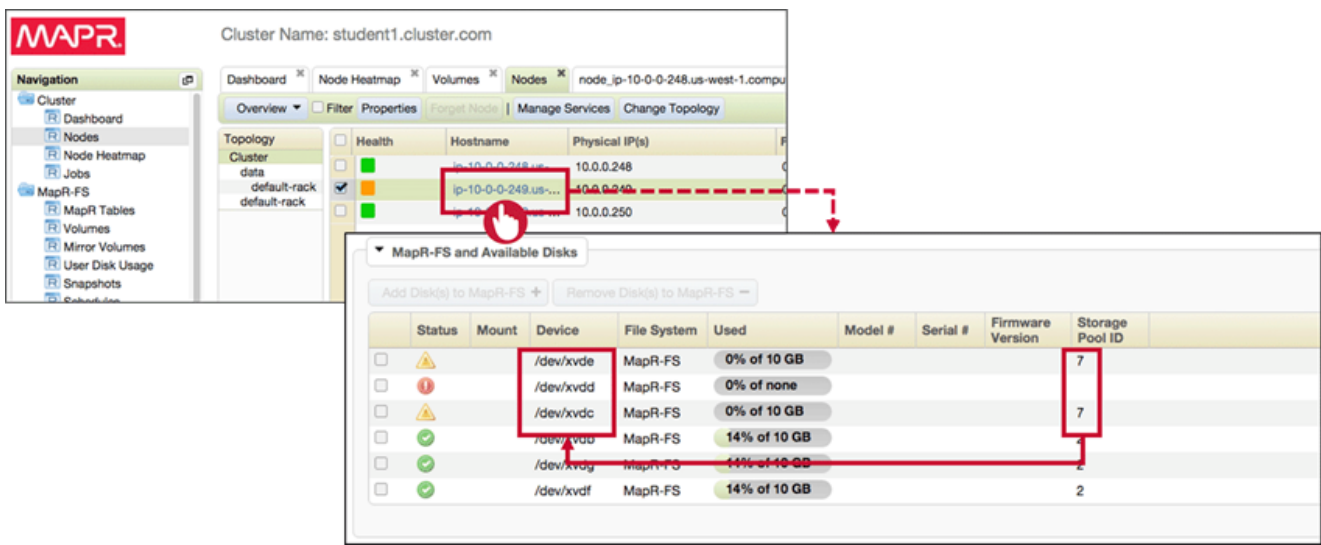
- VOLUME_ALARM_DATA_UNDER_REPLICATED
 - data under available disk is replicated but doesn’t meet minimum replication requirements.
 - automatically re-replicated from another copy and alarm cleared
- VOLUME_ALARM_DATA_UNAVAILABLE
 - If the only copy of data was on failed disk or all disks with that data failed.
 - run `/opt/mapr/server/fsck` to attempt to recover
 - Can bring storagepool online if `fsck` clears disk issues.
 - risk of data loss

Caution : Risk of data loss when running `fsck -r`

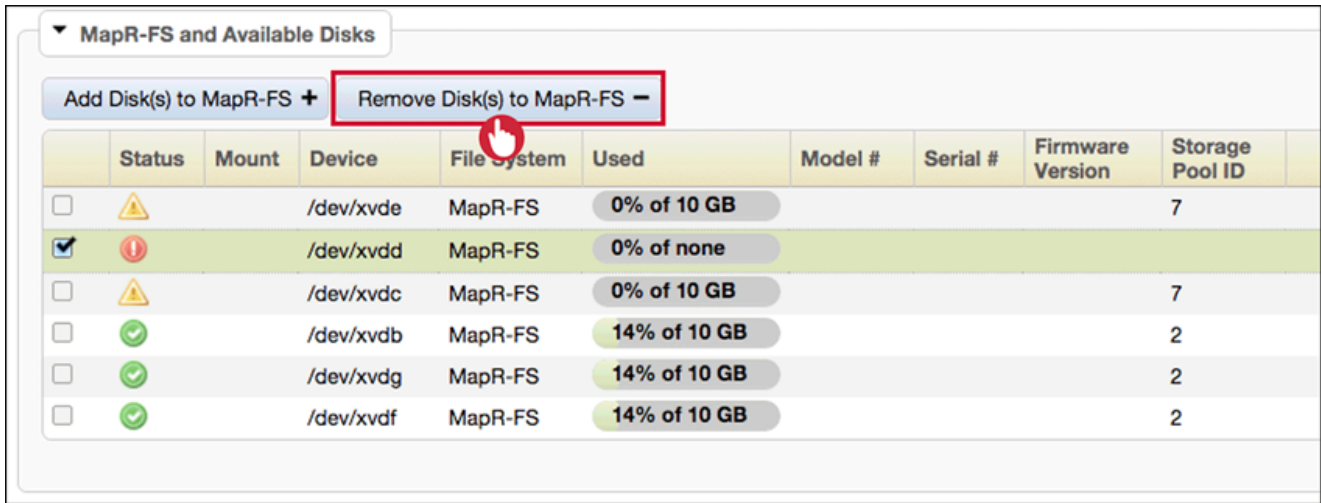
Replace a Failed Disk: MCS

To replace a failed disk in MCS navigate as follows :

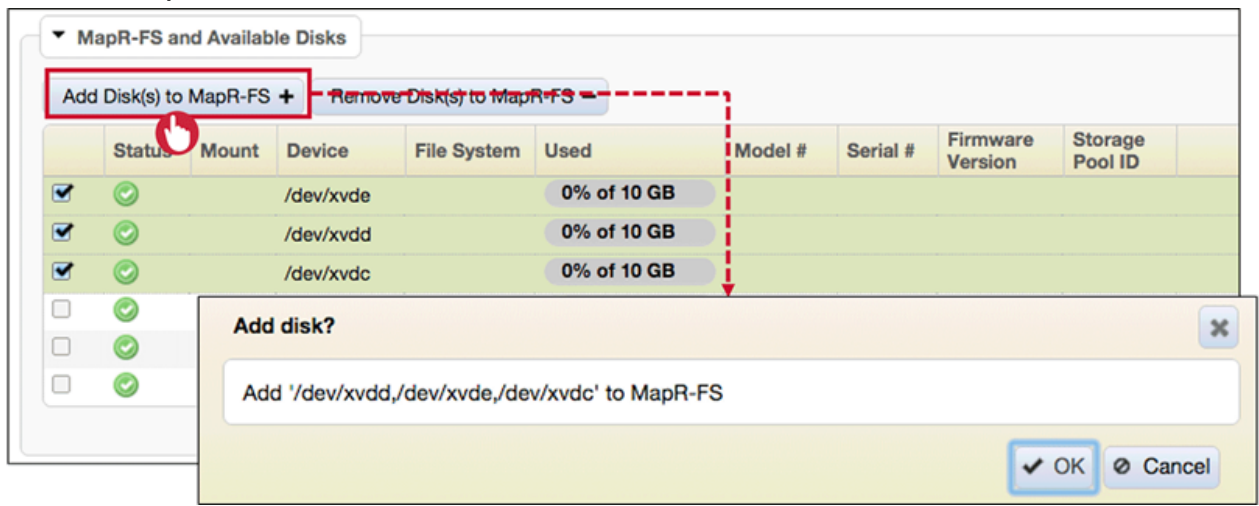
1. Locate the Failed Disk



2. Remove the disk from MapR-FS and physically replace failed disk

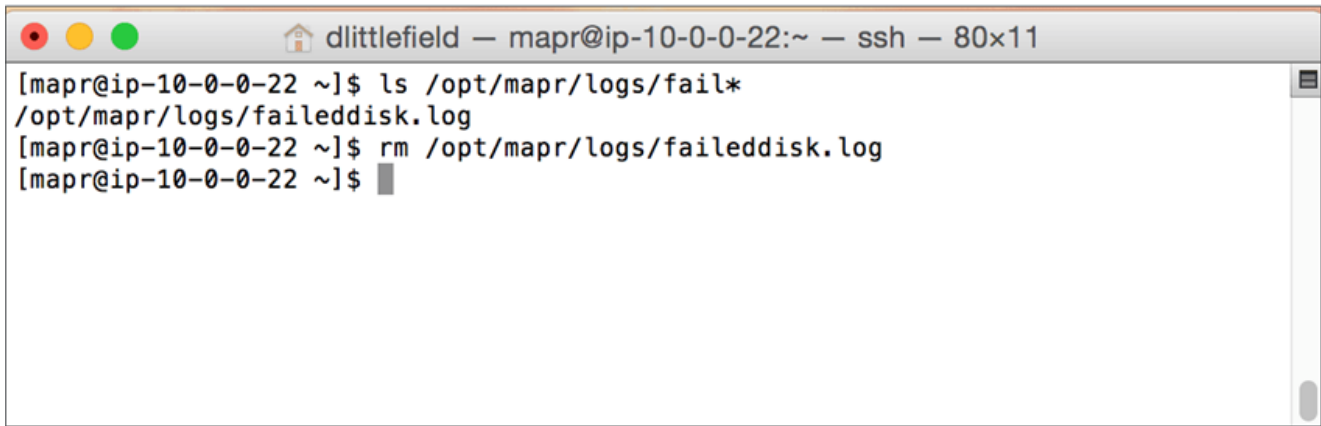


3. Add New/Repaired Disks

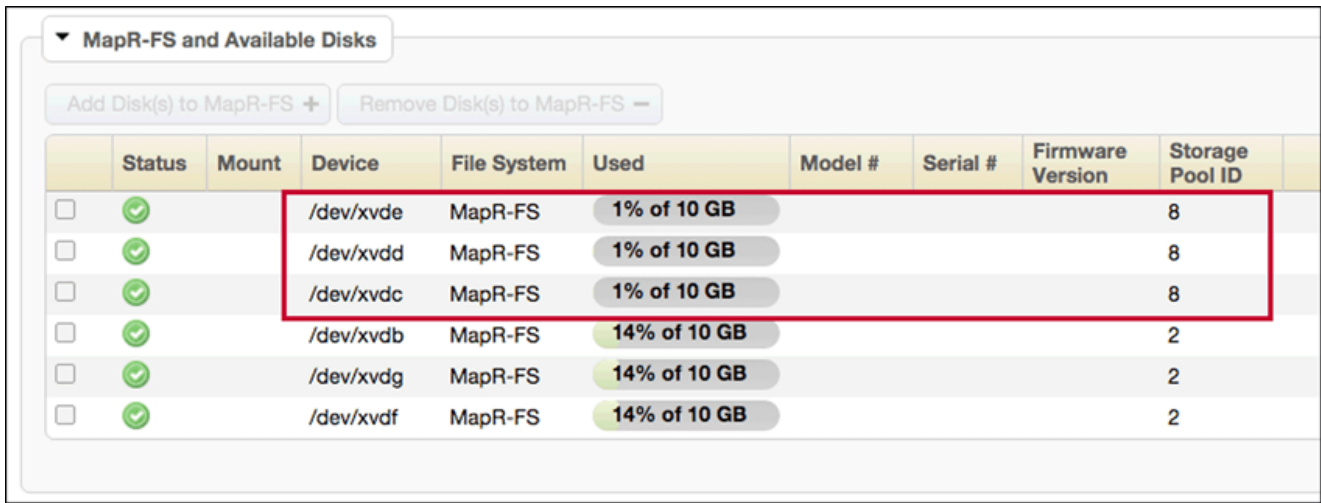


4. Remove the failed disk report when replacement is complete.

```
$ rm /opt/mapr/logs/faileddisk.log
```



5. Verify New Storage Pool generated for new disks.



Replace a Failed Disk: CLI

To replace a failed disk in CLI perform following commands :

1. Run `$ maprccli disk remove`

```
dlittlefield — mapr@ip-10-0-0-76:~ — ssh — 99x24

[mapr@ip-10-0-0-76 ~]$ maprccli disk remove -host `hostname -f`
-disks /dev/xvdb -force 1
```

2. Note drives removed; physically replace failed drive

```
dlittlefield — mapr@ip-10-0-0-76:~ — ssh — 99x24

[mapr@ip-10-0-0-76 ~]$ maprccli disk remove -host `hostname -f`
-disks /dev/xvdb -force 1
message      host
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdb
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdc
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdd
```

3. Remove failed disk report log

```
dlittlefield — mapr@ip-10-0-0-76:~ — ssh — 99x24

[mapr@ip-10-0-0-76 ~]$ maprccli disk remove -host `hostname -f`
-disks /dev/xvdb -force 1
message      host
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdb
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdc
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdd
[mapr@ip-10-0-0-76 ~]$ rm /opt/mapr/logs/faileddisk.log
```

4. Run `$ maprccli disk add` with list of disk being replaced.

```
dlittlefield — mapr@ip-10-0-0-76:~ — ssh — 99x24

[mapr@ip-10-0-0-76 ~]$ maprccli disk remove -host `hostname -f`
-disks /dev/xvdb -force 1
message      host
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdb
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdc
removed.     ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdd
[mapr@ip-10-0-0-76 ~]$ rm /opt/mapr/logs/faileddisk.log
[mapr@ip-10-0-0-76 ~]$ maprccli disk add -host `hostname -f` -disks
/dev/xvdb,/dev/xvdc,/dev/xvdd
message      host
added.       ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdb
added.       ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdc
added.       ip-10-0-0-76.us-west-1.compute.internal  /dev/xvdd
```

Add and Remove Disks

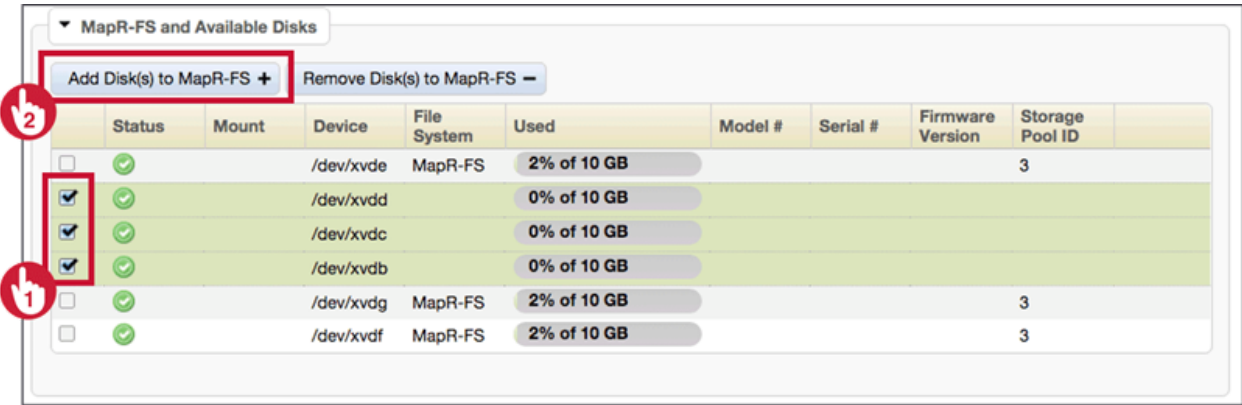
Best Practices

- Add disks in groups to form a storage pool
 - If you add a single disk a storage pool of single disk is created.
 - If you add multiple disks MapR will automatically generated storagepool(s) from them.
- Add disks with homogenous sizes and speeds
 - Optimize performance
 - Maximize capacity
- If you add heterogenous disk the performace and capacity is limited by slowest and lowest of the pool(randomly grouped).
- Add disks in groups by size (or performance like SSD vs HDD); this ensures that only disks of the same size go into a storage pool

Procedures: Adding and Removing Disks

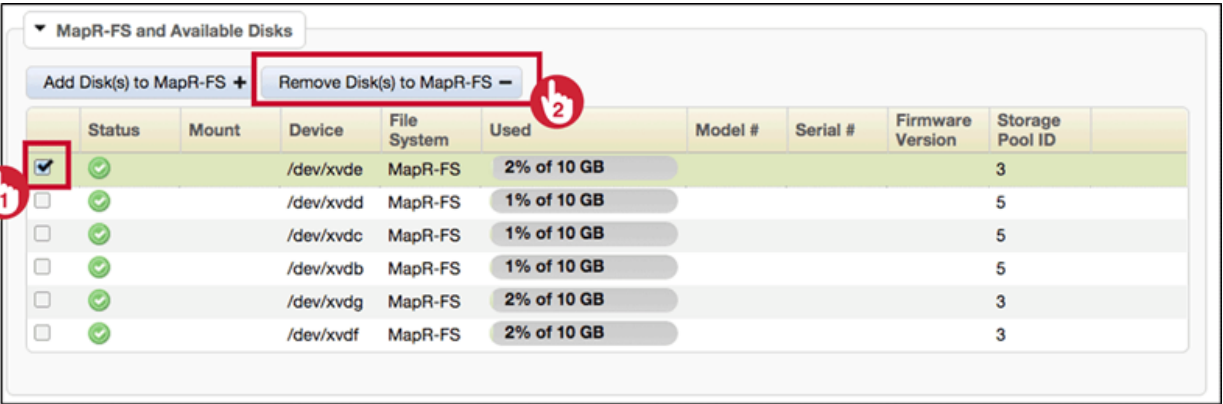
Add Disks

After physically installing the new disks we can add them to MapRFS using MCS or CLI as follows :



Remove Disks

- Removing disk removes the entire storage pool(all the other disks in the pool too)
- Removed disks no longer contain usable data
- Make sure data exists elsewhere (storagepool replicated). GUI will warn if the data is not replicated elsewhere.

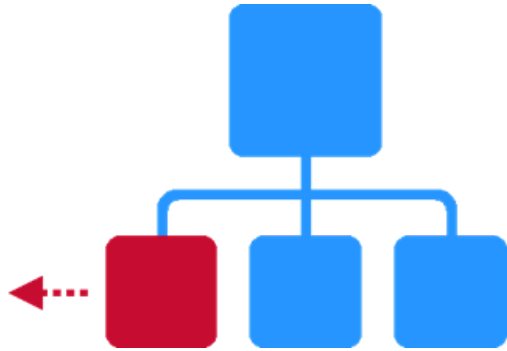


Perform Node Maintenance

Decommissioning vs. Maintenance

1. Decommissioning : Permanently remove from cluster

- Failed services
- Disks unrecoverable
- corrupted filesystems or OS



2. Maintenance : Plan to perform an action but keep the node as part of the cluster

- Upgrade the OS
- Add or replace disk drives
- Diagnose problems



Decommission a Node

Decommission a Node

1. Move node to the `/decommissioned` topology

- Doesn't need to be named that but move the node to a topology outside of `/data`
- Wait for replication to finish
- Verify volume information removed

2. Stop Warden and ZooKeeper

3. Run `maprcli remove node`

4. Remove installed packages

Clean Up

1. Remove `/opt/mapr` directory

2. Remove MapR core files in `/opt/cores`

3. If removed node has CLDB or ZooKeeper, run `configure.sh` on all other nodes to maintain HA

- `configure.sh -C` if CLDB node changed
- `configure.sh -Z` if ZooKeeper node changed

Perform Maintenance on a Node

Maintenance Overview

- No need to decommission
- Run `maprcli node maintenance`
- Replication starts after specified `-timeoutminutes`
- If multiple nodes, perform sequentially
- Don't offline more than half of the ZooKeepers
- Don't offline all CLDB or ResourceManager nodes

Perform Maintenance

1. Issue node maintenance command:

```
$ maprccli node maintenance -nodes <node> -timeoutminutes <minutes>
```

2. Stop Warden

3. Stop ZooKeeper (if installed on node)

4. Perform maintenance and reboot

Add Nodes

Expand a Cluster

When a new node is added to existing cluster you need to consider

- Node Topology
- How Management services are dispersed throughout the cluster?
- How data populates?

Node Topology

- Rack is a typical failure domain as some outages can take down entire rack.
- Rack can support 15+ nodes, you should consider spreading nodes evenly across three racks.
- By default three replicas of each data container, having three racks allow three sub-topology and containers can be replicated across three racks.
- Use fewer nodes per rack to distribute cluster nodes

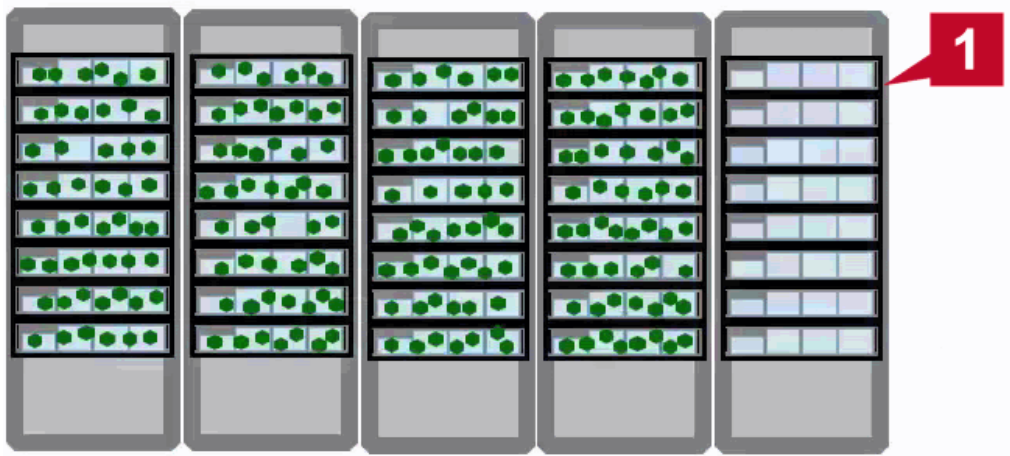
Management Services

- If you have multiple management services on a node, it's recommended to move them to the newly added node
- Recommended to put NFS service on all the nodes.



Data Considerations

1. New data written randomly, weighted toward new nodes so gradually data is distributed.
2. Alternatively distribute data more quickly with disk balancer after a new node is added.



Add Nodes

Here is the overview of steps performed when adding a new node. Refer MapR [documentation](http://doc.mapr.com/display/MapR/Adding+Nodes+to+a+Cluster) (<http://doc.mapr.com/display/MapR/Adding+Nodes+to+a+Cluster>) for more details

Prepare

1. Set default topology to /newnodes
 - Optional
2. Verify node specifications
3. Create MapR user
4. Add MapR repository
- Install and Configure**
5. Add MapR package key
6. Install MapR service packages
7. Run configure.sh

```
$  configure.sh -C <cldb nodes> -Z <zk nodes> -N <cluster name> -M7
```

8. Start Warden on new node
- Finish Up**

9. Format and add disks

10. Set node topology

11. Mount cluster

- Optional
- Only if NFS installed

Troubleshooting

Troubleshooting Different Problem Types

How Do You Know There’s a Problem ?

The three common give-aways are :

- Alarm raised
- Performance issues
- Jobs failing

Resources

Resources

There are several resources to consult during troubleshooting.

- Log files
- MapR documentation
- MapR Answers
- Apache forums and websites
- MapR Support (with Converged Enterprise Edition)



Log File Locations

This table lists some of common log types and their default locations. Some are found in local filesystem and while others in MapRFS

Log Type	Default Location
MapR-specific logs	/opt/mapr/logs
ZooKeeper logs	/opt/mapr/zookeeper/zookeeper-<version>/logs
Ecosystem component logs	/opt/mapr/<component>
YARN job logs	/opt/mapr/hadoop/hadoop-2*/logs
YARN job logs (log aggregation enabled)	/mapr/<cluster name>/tmp/logs

Log File Retention

Many logs expire automatically, so its possible for long running jobs logs are deleted before completion. You can change the retention times.



YARN

Set:

```
yarn.nodemanager.log.retain-seconds
```

in

```
yarn-site.xml
```

/opt/mapr/logs

To retain indefinitely set:

```
log.retention.exceptions
```

in

```
warden.conf
```

Troubleshooting Scenarios

Service Not Running

- Typically corresponds to a raised alarm
- Warden will attempt start or restart 3 times
- Each service has its own logs in `/opt/mapr/logs`
 - Don't forget to check Warden logs

Service Problems

1. Open two terminal windows
2. Window 1:

```
$ service mapr-warden stop
```

3. Window 2:

```
$ tail -f /opt/mapr/logs/warden.log
```

4. Window 1:

```
$ service mapr-warden start
```

Cluster Startup Issues

1. ZooKeepers up?

```
$ service mapr-zookeeper qstatus
```

2. Review `warden.log` and `cldb.log`

3. Get `maprccli node cldbmaster` to succeed

4. Troubleshoot individual services

Data Not Available

- Is affected volume node-specific?
 - `/var/mapr/local/<node name>`
 - Is node offline?
- Failed disks and nodes
- Permission problems

Jobs Not Running

- Start with the YARN job logs
 - command line
 - ResourceManager page (`<IP address>:19888`)
- Permissions problems
- Resource problems
- Recommended to run Yarn Log Aggregator to retain logs for long time centrally in MapRFS.

Degraded Performance

- Check `/opt/mapr/logs/mfs.log-3`
 - Failures communicating with the CLDB nodes or other servers?
- Check node resources
 - CPU, memory, swap space, disk space, network
- Isolate management services

Use Support Utilities

Tools

Core Files

- Copies of contents of memories when certain anamolies are detected.
- Located in `/opt/cores`
 - Name includes service involved
- An alarm is raised when a core file is created
- No immediate action required
 - Service will automatically restart
- Forward to MapR Support when possible

fsck

- Find/fix file system inconsistencies
 - `/opt/mapr/server/fsck`
- Repair storage pools after disk failure
- Verification or repair
 - Use `-r` for repair

Caution: Using `fsck -r` can result in loss of data. Contact support first!

gfsck

- Find/fix cluster/volume/snapshot inconsistencies
 - `/opt/mapr/bin/gfsck`
- Scans and repairs a cluster, volume, or snapshot
- Typical process
 - Take storage pool offline
 - Run `fscck` on storage pool
 - Bring storage pool back online
 - Run `gfsck` on affected cluster/volumes/snapshots

mapr-support-dump.sh

- Collects information/logs from a single node

```
$ /opt/mapr/support/tools/mapr-support-dump.sh
```


- Output at `/opt/mapr/support/dump`
 - Creates a `.tar` file

mapr-support-collect.sh

- Collects information/logs from entire cluster

```
$ /opt/mapr/support/tools/mapr-support-collect.sh
```

- Output at `/opt/mapr/support/collect`
 - Uses SSH and SCP
 - Creates a `.tar` file

 **Tags:**

Administration

Hadoop

MapR

 **Categories:**

MapR

 **Updated:** April 11, 2018