

Working with Mirror Volumes

A *mirror volume* is a read-only physical copy of another volume, the *source volume*. You can use mirror volumes in the same cluster (local mirroring) to provide local load balancing by using mirror volumes to serve read requests for the most frequently accessed data in the cluster. You can also mirror volumes on a separate cluster (remote mirroring) for backup and disaster readiness purposes.

For information about "promoting" mirror volumes to read-write mode, see [Using Promotable Mirrors](#).

Once you've created a mirror volume, keeping your mirror synchronized with its source volume is fast. Because mirror operations are based on a **snapshot** of the source volume, your source volume remains available for read and write operations for the entire duration of the process.

i Auditing on remote mirror volumes is not enabled even if it is enabled on source volumes. The `mapcli volume audit` command must be run to enable auditing on a remote mirror volume. Auditing for particular directories, files, and MapR-DB tables in a remote mirror volume is enabled automatically if auditing is enabled for them in the source volume. For details about auditing, see [Auditing of Cluster Administration and Operations on Directories, Files, and Tables](#).

Mirroring Overview

Creating a mirror volume is similar to creating a normal read/write volume. However, when you create a mirror volume, you must specify a source volume that the mirror retrieves content from. This retrieval is called the *mirroring operation*. Like a normal volume, a mirror volume has a configurable replication factor. Only one copy of the data is transmitted from the source volume to the mirror volume; the source and mirror volumes handle their own replication independently.

The MapR system creates a temporary **snapshot** of the source volume at the start of a mirroring operation. The mirroring process reads content from the snapshot into the mirror volume. The source volume remains available for read and write operations during the mirroring process. If the mirroring operation is **schedule-based**, the snapshot expires according to the value of the schedule's **Retain For** parameter. Snapshots created during manual mirroring persist until they are deleted manually.

The mirroring process transmits only the differences between the source volume and the mirror. The initial mirroring operation copies the entire source volume, but subsequent mirroring operations can be extremely fast. The mirroring operation never consumes all available network bandwidth, and throttles back when other processes need more network bandwidth. The server sending mirror data continuously monitors the total round-trip time between the data transmission and arrival, and uses this information to restrict itself to 70% of the available bandwidth (continuously calculated). If the network or servers anywhere along the entire path need more bandwidth, the sending server throttles back automatically. If more bandwidth opens up, the sender automatically increases how fast it sends data. Mirror throttling can be disabled so that all available bandwidth is devoted to mirror operations. See [Disabling Mirror Throttling](#) for details.

During the copy process, the mirror is a fully-consistent image of the source volume. Mirrors are atomically updated at the mirror destination. The mirror does not change until all bits are transferred, at which point all the new files, directories, blocks, etc., are atomically moved into their new positions in the mirror-volume. The previous mirror is left behind as a snapshot, which can be accessed from the `.snapshot` directory. These old snapshots can be deleted on a [schedule](#).

Mirroring is extremely resilient. In the case of a *network partition*, where some or all of the machines that host the source volume cannot communicate with the machines that host the mirror volume, the mirroring operation periodically retries the connection. Once the network is restored, the mirroring operation resumes.

When the root volume on a cluster is mirrored, the source root volume contains a writable volume link, `.rw` that points to the read/write copies of all local volumes. In that case, the mount path `/` refers to one of the root volume's mirrors, and is read-only. The mount path `/.rw` refers to the source volume, and is read/write.

A mount path that consists entirely of mirrored volumes refers to a mirrored copy of the specified volume. When a mount path contains volumes that are not mirrored, the path refers to the target volume directly. In cases where a path refers to a mirrored copy, the `.rw` link is useful for navigating to the read/write source volume. The table below provides examples.

Example Volume Topology with Mirrors

For the four volumes `/`, `a`, `b`, and `c`, the following table indicates the volumes referred to by example mount paths for particular combinations of mirrored and not mirrored volumes in the path:

<code>/</code>	<code>a</code>	<code>b</code>	<code>c</code>	This Path	Refers To This Volume...	Which is...
Mirrored	Mirrored	Mirrored	Mirrored	<code>/a/b/c</code>	Mirror of <code>c</code>	Read-only
Mirrored	Mirrored	Mirrored	Mirrored	<code>/.rw/a/b/c</code>	<code>c</code> directly	Read/Write
Mirrored	Mirrored	<i>Not Mirrored</i>	Mirrored	<code>/a/b/c</code>	<code>c</code> directly	Read/Write
Mirrored	Mirrored	<i>Not Mirrored</i>	Mirrored	<code>/a</code>	Mirror of <code>a</code>	Read-only

Not Mirrored	Mirrored	Mirrored	Mirrored	/a/b/c	c directly	Read/Write
--------------	----------	----------	----------	--------	------------	------------

Setting a Mirror Schedule

You can automate mirror synchronization by setting a [schedule](#). You can also use the `volume mirror start` command to synchronize data manually.

Completion time for a mirroring operation is affected by available network bandwidth and the amount of data to transmit.

For best performance, set the mirroring schedule according to the anticipated rate of data changes and the available bandwidth for mirroring.

Mirror Cascades

In a cascade, one mirror synchronizes to the source volume, and each successive mirror uses a previous mirror as its source. Mirror cascades are useful for propagating data over a distance, then re-propagating the data locally instead of transferring the same data remotely again for each copy of the mirror. In the example below, the `<` character indicates a mirror's source:

```
/ < mirror1 < mirror2 < mirror3
```

A mirror cascade makes more efficient use of your cluster's network bandwidth, but synchronization can be slower to propagate through the chain. For cases where synchronization of mirrors is a higher priority than network bandwidth optimization, make each mirror read directly from the source volume:

```
mirror1 > < mirror2
      /
mirror3 > < mirror4
```

You can create or break a mirror cascade made from existing mirror volumes by changing the source volume of each mirror in the [Volume Properties](#) dialog.

Other Mirror Operations

For more information on mirror volume operations, see the following sections:

- You can set the [topology](#) of a mirror volume to determine the placement of the data.
- You can change a mirror's source volume by changing the source volume in the [Volume Properties](#) dialog.
- To create a new mirror volume refer to [Creating a Volume](#) (requires an Enterprise Edition license and `cv` permission)
- To modify a mirror (including changing its source), see [Modifying a Volume](#)
- To remove a mirror, see [Removing a Volume or Mirror](#)

Local Mirroring

A *local mirror volume* is a mirror volume whose source is on the same cluster. Local mirror volumes are useful for load balancing or for providing a read-only copy of a data set.

You can locate your local mirror volumes in specific servers or on racks with particularly high bandwidth, mounted in a public directory separate from the source volume.

The most frequently accessed volumes in a cluster are likely to be the root volume and its immediate children. In order to load-balance read operations on these volumes, mirror the root volume (typically `mapr.cluster.root`, which is mounted at `/`). By mirroring these volumes, read requests can be served from the mirrors, distributing load across the nodes. Less-frequently accessed volumes that are lower in the hierarchy do not need mirror volumes. Since the mount paths for those volumes are not mirrored throughout, those volumes are writable.

To create a local mirror using the MapR Control System:

1. Log on to the [MapR Control System](#).
2. In the navigation pane, select **MapR-FS > Volumes**.
3. Click the **New Volume** button.

4. In the [New Volume](#) dialog, specify the following values:
 - Select **Local Mirror Volume**.
 - Enter a name for the mirror volume in the **Mirror Name** field. If the mirror is on the same cluster as the source volume, the source and mirror volumes must have different names.
 - Enter the source volume name (not mount point) in the **Source Volume Name** field.
 - To automate mirroring, select a [schedule](#) corresponding to critical data, important data, normal data, or a user-defined schedule from the **Mirror Schedule** dropdown menu.

To create a local mirror using the `volume create` command:

1. Connect via ssh to a node on the cluster where you want to create the mirror.
2. Use the `volume create` command to create the mirror volume. Specify the `source` volume name, provide a name for the mirror volume, and specify a `type` of 1. Example:

```
maprcli volume create -name volume-a -source volume-a -type 1 -schedule 2
```

Remote Mirroring

A *remote mirror volume* is a mirror volume with a source in another cluster. You can use remote mirrors for offsite backup, for data transfer to remote facilities, and for load and latency balancing for large websites. By mirroring the cluster's root volume and all other volumes in the cluster, you can create an entire mirrored cluster that keeps in sync with the source cluster.

Backup mirrors for disaster recovery can be located on physical media outside the cluster or in a remote cluster. In the event of a disaster affecting the source cluster, you can check the time of last successful synchronization to determine how current the backup is (see [Mirror Status](#) below).

Creating Remote Mirrors

Creating remote mirrors is similar to creating local mirrors, except that the mirror volume resides in a different cluster from the source volume. To properly identify the source volume, the source cluster name must also be specified when the mirror volume is created. In addition, you must edit the `mapr-clusters.conf` file so that each cluster can resolve the nodes in the other cluster.

To create a mirror on a remote cluster, you must have the same UID for the `MAPR_USER` (the cluster owner) for both the primary cluster (where the source volume resides) and the remote clusters (where the mirror volumes reside; also known as the destination clusters). You also need to have these volume permissions:

- `dump` permission on the source volumes
- `restore` permission on the mirror volumes at the destination clusters

When a mirror volume is created on a remote cluster (according to the entries in the `mapr-clusters.conf` file), the CLDB checks that the local volume exists in the local cluster. If both clusters are not set up and running, the remote mirror volume cannot be created.

To summarize:

- Each cluster must be already set up and running
- Each cluster must have a unique name
- Every node in each cluster must be able to resolve all nodes in remote clusters, either through DNS or entries in `/etc/hosts`
- The UID for the `MAPR_USER` (cluster owner) must be the same for the source and destination clusters.
- Volume permission is set to `dump` on the source volumes
- Volume permission is set to `restore` on the mirror volumes

Editing the `mapr-clusters.conf` File

To mirror volumes between clusters, start by editing the `mapr-clusters.conf` file on the source volume's cluster and create an entry for each additional cluster that hosts a mirror of the volume. The entry must list the cluster's name, followed by a space-separated list of hostnames and ports for the cluster's CLDB nodes. In addition, use the `secure` parameter to specify whether the clusters are secure or unsecure.

1. On each cluster, make a note of the cluster name and CLDB nodes (the first line in `mapr-clusters.conf`)
2. On each webserver and CLDB node, add the remote cluster's CLDB nodes to `/opt/mapr/conf/mapr-clusters.conf`, using the following format:

```
clustername1 <CLDB> <CLDB> <CLDB>
[ clustername2 <CLDB> <CLDB> <CLDB> ]
[ ... ]
```

For example, suppose you have a cluster, `devcluster1`, with two CLDB nodes, `devclbdb1-1` and `devclbdb1-2`. Now you want to add a second cluster called `devcluster2` with CLDB nodes `devclbdb2-1` and `devclbdb2-2`. Edit the `mapr-clusters.conf` file and add the line for `devcluster2` as shown:

```
devcluster1 devclbdb1-1:7222 devclbdb1-2:7222
devcluster2 devclbdb2-1:7222 devclbdb2-2:7222
```



You must include the port number in the CLDB hostname notation.

3. Set `secure=true` if both clusters are secure. Set `secure=false` if both clusters are not secure.



Mirroring only works between two secure clusters or between two unsecure clusters. Mirroring does not work when one cluster is secure and the other is unsecure.

4. If you set `secure=true`, you must generate a cross-cluster ticket before proceeding. See [Mirror Volumes and Secure Clusters](#), which must be configured before you initiate a remote mirror.
5. On each cluster, restart the `mapr-webserver` service on all nodes where it is running.

To create a remote mirror using the MapR Control System:

1. Log on to the [MapR Control System](#).
2. Check the **Cluster Name** (near the MapR logo). If you are not connected to the cluster on which you want to create a mirror:
 - Click the **[+]** next to the Cluster Name.
 - In the Available Clusters dialog, click the name of the cluster where you want to create a mirror.
 - In the Launching Web Interface dialog, click that cluster again to connect.
3. In the navigation pane, select **MapR-FS > Volumes**.
4. Click the **New Volume** button.
5. In the **New Volume** dialog, specify the following values:
 - Select **Remote Mirror Volume**.
 - Enter a name for the mirror volume in the **Volume Name** field. If the mirror is on the same cluster as the source volume, the source and mirror volumes must have different names.
 - Enter the source volume name (not mount point) in the **Source Volume** field.
 - Enter the source cluster name in the **Source Cluster** field.
 - To automate mirroring, select a **schedule** from the **Mirror Update Schedule** dropdown menu.

To create a remote mirror using the `volume create` command:

1. Connect to a node on the cluster where you wish to create the mirror.
2. Use the `volume create` command to create the mirror volume. Specify the source volume and cluster in the format `<volume>@<cluster>`, provide a name for the mirror volume, and specify a `type` of 1. Example:

```
maprcli volume create -name volume-a -source volume-a@cluster-1 -type 1 -schedule
2
```

Moving Large Amounts of Data to a Remote Cluster

You can use the `volume dump create` command to create volume copies for transport on physical media. The `volume dump create` command creates backup files containing the volumes, which can be reconstituted into mirrors at the remote cluster with the `volume dump restore` command. Associate these mirrors with their source volumes with the `volume modify` command to re-establish synchronization.

Another way to transfer large amounts of data to a remote cluster is to create a small cluster locally and mirror to that local cluster. Then move that cluster to a remote location and enlarge it by adding more nodes.

To set up cross-mirroring between clusters

You can *cross-mirror* between clusters, mirroring some volumes from cluster A to cluster B and other volumes from cluster B to cluster A. To set up cross-mirroring, create entries in `mapr-clusters.conf` as follows:

- Entries in `mapr-clusters.conf` on cluster A nodes:
 - First line contains cluster name and CLDB nodes of cluster A (the local cluster)
 - Second line contains cluster name and CLDB nodes of cluster B (the remote cluster)
- Entries in `mapr-clusters.conf` on cluster B nodes:
 - First line contains cluster name and CLDB nodes of cluster B (the local cluster)
 - Second line contains cluster name and CLDB nodes of cluster A (the remote cluster)

For example, the `mapr-clusters.conf` file for cluster A with three CLDB nodes (`nodeA`, `nodeB`, and `nodeC`) would look like this:

```
clusterA nodeA:7222 nodeB:7222 nodeC:7222
clusterB nodeD:7222
```

The `mapr-clusters.conf` file for cluster B with one CLDB node (`nodeD`) would look like this:

```
clusterB nodeD:7222
clusterA nodeA:7222 nodeB:7222 nodeC:7222
```

By creating additional entries in the `mapr-clusters.conf` file, you can mirror from one cluster to several others.

Mirror Status

You can see a list of all mirror volumes and their current status on the [Mirror Volumes](#) view (in the MapR Control System, select **MapR-FS** then **Mirror Volumes**) or using the `volume list` command. You can see additional information about mirror volumes on the CLDB status page (in the MapR Control System, select **CLDB**), which shows the status and last successful synchronization of all mirrors, as well as the container locations for all volumes. You can also find container locations using the [hadoop mfs](#) commands.

Disabling Mirror Throttling

By default, mirror throttling is enabled, which means that the server that sends mirror data restricts itself to 30% (by default) of the available bandwidth. Mirror throttling is based on the number of outstanding requests on the networks and outstanding I/O requests on disk. It can be tuned using the parameters `mfs.disk.iothrottle.count`, `mfs.disk.resynciothrottle.factor`, and `mfs.network.resynciothrottle.factor` in `mfs.conf` file. When other processes need more network bandwidth, the server throttles back to slow down the rate of data transfer.

By disabling throttling, the mirror operation completes faster. To disable mirror throttling from the command line, run the `volume modify` command on a *source* volume and set the `-mirrorthrottle` option to `false`, as shown in this example:

```
Cluster1> maprcli volume modify -name volA -mirrorthrottle false
```

This command disables throttling for all mirror volumes whose source is `volA`. Note that the `-mirrorthrottle` option only applies to volumes that have mirrors.

Starting a Mirror

When a mirror *starts*, all the data in the source volume is copied into the mirror volume. Starting a mirror volume requires that the mirror volume exist and be associated with a source. After you start a mirror, synchronize it with the source volume regularly to keep the mirror current. You can start a mirror using the `volume mirror start` command, or use the following procedure to start mirroring using the MapR Control System.

To start mirroring using the MapR Control System:

1. In the Navigation pane, expand the **MapR-FS** group and click the [Volumes](#) view.
2. Select the checkbox beside the name of each volume you wish to mirror.

3. Click the **Start Mirroring** button.

Stopping a Mirror

Stopping a mirror halts any replication or synchronization process currently in progress. Stopping a mirror does not delete or remove the mirror volume. Stop a mirror with the `volume mirror stop` command, or use the following procedure to stop mirroring using the MapR Control System.

To stop mirroring using the MapR Control System:

1. In the Navigation pane, expand the **MapR-FS** group and click the **Volumes** view.
2. Select the checkbox beside the name of each volume you wish to stop mirroring.
3. Click the **Stop Mirroring** button.

Pushing Changes to Mirrors

To *push* a mirror means to start pushing data from the source volume to all its local mirrors. You can push source volume changes out to all mirrors using the `volume mirror push` command, which returns after the data has been pushed.

Using Volume Links with Mirrors

When you mirror a volume, read requests to the source volume can be served by any of its mirrors on the same cluster via a `volume link` of type `mirror`. A volume link is similar to a normal volume mount point, except that you can specify whether it points to the source volume or its mirrors.

- To write to (and read from) the source volume, mount the source volume normally. As long as the source volume is mounted below a non-mirrored volume, you can read and write to the volume normally via its direct mount path. You can also use a volume link of type `writable` to write directly to the source volume regardless of its mount point.
- To read from the mirrors, use the `volume link create` command to make a volume link (of type `mirror`) to the source volume. Any read requests from the volume link are distributed among the volume's mirrors. Since the volume link provides access to the mirror volumes, you do not need to mount the mirror volumes.