



w: [www.mariadb.com](http://www.mariadb.com)

e: [info@mariadb.com](mailto:info@mariadb.com)

## **High Availability (HA) Solutions and Roadmap**

Anders Karlsson, Sales Engineer, MariaDB

Last Update: 10.1.2014



## Table of Content

1	Introduction .....	3
2	MariaDB High Availability.....	3
2.1	High Availability components.....	3
2.1.1	Data redundancy.....	3
2.1.2	HA Monitoring, Management and Failover.....	4
3	Roadmap .....	5
3.1	MariaDB .....	5
3.2	MariaDB Galera Cluster.....	5
3.3	MaxScale .....	5
3.3.1	MaxScale core modules.....	5
3.4	MariaDB Enterprise.....	6



## 1 Introduction

MariaDB provides MySQL and MariaDB software and services to the market. One of the hot topics for MariaDB and MySQL environments right now is High Availability – the MariaDB Team has a long experience with this and has several solutions for different needs.

## 2 MariaDB High Availability

The MariaDB Team has a long experience in High Availability and is developing and supporting several components in the area. This section outlines the High Availability options supported by Team MariaDB.

### 2.1 High Availability components

For Database high availability there are three basic components that are necessary:

- Data redundancy - This is to ensure that data present on several servers so that no data is lost when a server is down and to ensure that one can fail over to another server in that case.
- Monitoring and management - Used to control and keep track of services, to manage them and to take appropriate action in the case of a failure.
- Failover mechanism - This is provide routing of traffic to a server that is up and running, and to facilitate failover to another server transparently to the application.

#### 2.1.1 Data redundancy

There are several means of achieving data redundancy with MariaDB / MySQL, and this has developed over the years. Recently some interesting new technologies has entered the arena.

##### 2.1.1.1 Shared disk / disk replication

This is a classic means of achieving data redundancy, but it is more complex than meets the eye. Another issues with this kind of solution is that they are costly. Among the options here is using a SAN or a Linux based disk replication technology such as *DRBD*.

MariaDB offers support for this, but it is no longer the recommended solution in most cases.

##### 2.1.1.2 MySQL Replication

This is software only replication that is built in to MySQL. The advantage is that this is very easy to set up and has low overhead. The issue is mainly that this technology, although great for Scale-out, and this was the initial purpose for this feature in MySQL, does not really work well out of the box for High Availability.

MariaDB supports users using MySQL Replication, and this a very common technology with our customers. It is used less often for High Availability than used to be the case, but for Scale-out it remains very popular and useful. When used with High Availability it is often combined with [MHA](#).

##### 2.1.1.3 Galera

Galera is a synchronous replication software that extends MariaDB/MySQL and is completely separate from classic MySQL Replication. Using Galera a Cluster is created that allows adding and removal of nodes with a live system, where data is always consistent across nodes and where the status of the



w: [www.mariadb.com](http://www.mariadb.com)

e: [info@mariadb.com](mailto:info@mariadb.com)

individual nodes is maintained globally. Also, this is a Master-Master setup that allows for much easier and faster failover.

Galera is supported by MariaDB as a component included with *MariaDB Galera Cluster*, as being supported by [MDBE](#) and [MaxScale](#) as well as a separate product. The MariaDB Team has a long standing relationship with the developers of Galera, *Codership*.

### **2.1.2 HA Monitoring, Management and Failover**

This is an area with a lot more different technologies and which are in many cases generic, i.e. support failover and Load Balancing for any type of server.

#### **2.1.2.1 Hardware Load Balancer**

In some cases a hardware load balancer is in place, such as *BigIP* from *F5 Networks*. This works well with MySQL/MariaDB and in particular with disk based failover solutions and with Galera.

Using a Hardware Load Balancer solution is supported by MariaDB, but the configuration of the Load Balancer and other things specific to this are supported by the provider of the Load Balancer.

#### **2.1.2.2 Linux HA Proxy and similar software solutions**

In the Linux world, *HA Proxy* is a very common software based load balancer and failover mechanism. For Windows, *Windows Server Failover Clustering* is a somewhat equivalent technology.

Using these technologies in a HA Solution is supported by MariaDB, but as in the case of a Hardware Load Balancer, MariaDB doesn't specifically support this component. In the case of *HA Proxy*, MariaDB does have internal competence to set this up and to aid in configuring it.

#### **2.1.2.3 MHA**

*MHA* (MySQL High Availability) is a technology for using MySQL Replication for High Availability. Using MySQL Replication for this has some added challenges in that this replication is Asynchronous, and also in that the way it is set up, failback is sometimes difficult. *MHA* is technology that handles all this. *MHA* is a simple but effective HA solution, applicable in some cases.

MariaDB is the maintainer of *MHA* and fully supports it.

#### **2.1.2.4 Driver / Application based failover**

In some cases, in particular in smaller configurations, the database driver or even the application can support failover. In this case, managing and monitoring the HA solution is handled in some other means.

MariaDB supports using the standard MySQL drivers (also called "connectors") for failover where this is applicable.

#### **2.1.2.5 MaxScale**

*MaxScale* is a server technology with a very wide range of applications. *MaxScale*, among other things, monitors the state of the servers and performs Load Balancing and Failover and also has monitoring interfaces. *MaxScale* also has an API that means it can easily be integrated with infrastructure components and technologies.

MariaDB is the developer of the *MaxScale* server and using *MaxScale* is fully supported by MariaDB.



### 3 Roadmap

MariaDB is about to release several new products that support the use and management of *MariaDB* and *MySQL*. Some of these products has already been released, other are about to be released shortly. All of these products are highly extensible, so once the core products and some initial modules has been released, new modules will be developed on a regular basis, as well as features being added to existing modules and to the server core.

#### 3.1 MariaDB

*MariaDB* is the core database server based on *MySQL* and developed by a set of core, original *MySQL* developers. MariaDB is a fully *MySQL* compatible database but having several unique features.

#### 3.2 MariaDB Galera Cluster

This is the current top-of-the-line offering in terms of High Availability and performance. It consists of MariaDB complemented with [Galera](#) High Availability and a set of supporting programs and services to create a highly available MariaDB environment.

#### 3.3 MaxScale

*MaxScale* is a generic server core, aimed at sitting in between the application and the database server. MaxScale is intended to provide a multitude of services in this role, such as database sharding, failover, load balancing, monitoring and auditing. All these services will be implemented as modules (plugins) using an open API so that third party developers or customers can develop and maintain custom modules.

MaxScale is basically a kind of “exchange” for requests coming in to it through a client protocol and then being authenticated and filtered to reach the database server.

As for the *High Availability* features that are core to MaxScale is online configuration and reconfiguration, which means that the system can be reconfigured when it is live. Another core feature in the Monitor plugin that allows a monitor to determine the current state of the servers and take appropriate action should a server go down for example.

Of the many modules supported, the *router* are probably the most interesting from a functionality point of view. It is this module that determines how a request from a client is forwarded. Connecting to MaxScale from an application is no different from connecting to a MariaDB or *MySQL* server, so existing applications and drivers continues to work as usual.

##### 3.3.1 MaxScale core modules

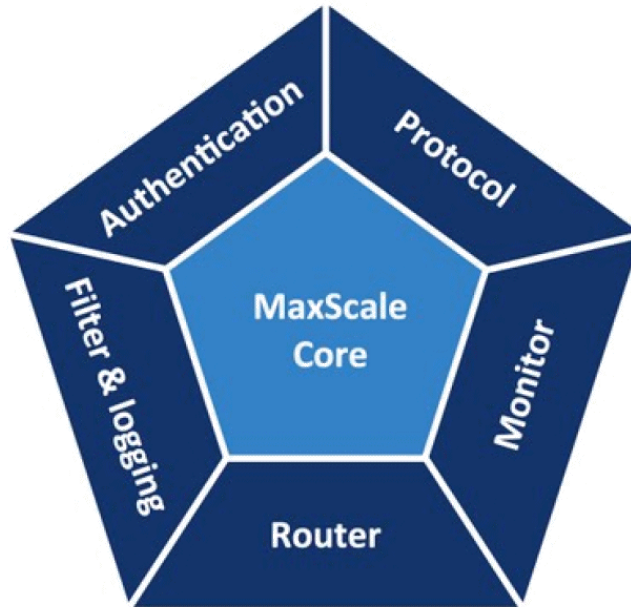
The initial release of MaxScale comes with some core modules:

- *MySQL Monitoring* - This is a module for monitoring the state of the *MySQL* servers in the attached Cluster of servers.
- *MySQL Client* - This is a protocol module for *MySQL* Clients.
- *MySQL Server* - This is the protocol module to connect to *MySQL* Servers.
- *Readconroute* - This is a straightforward load balancing module for *MySQL*.
- *Readwritesplit* - This module is a more advanced load balancing script that directs write to one set of servers and reads to another.



w: [www.mariadb.com](http://www.mariadb.com)

e: [info@mariadb.com](mailto:info@mariadb.com)



### 3.4 MariaDB Enterprise

*MariaDB Enterprise* (or MDBE) builds on MariaDB Galera Cluster but adds a management, monitoring and provisioning GUI. MDBE is currently in Beta. During this year, MDBE and MaxScale will be integrated, so that MDBE provides a GUI for a Cluster managed by MaxScale.