



OPENWORKS

BE UNSTOPPABLE



OPENWORKS

**MARIADB
MAXSCALE**

BROCK WILSON, CUSTOMER ENGINEERING, MARIADB

WHAT IS MARIADB MAXSCALE?

“MariaDB MaxScale is an advanced database proxy for MariaDB Enterprise Server, providing it with enterprise high availability, scalability, security and integration services while at the same time abstracting away the underlying database infrastructure to simplify application development and database administration.”

- Database proxy
- Automated failover
- Read/Write split routing
- Read connection router
- Binlog Router
- CDC(Kafka) router
- Traffic control

SKYSQL USES MARIADB MAXSCALE

“SkySQL is a second generation cloud database service that automates deployment, management and scaling so you can focus on strategic initiatives that move your business forward.”

- Database as a service (DBaaS)
- SkySQL clusters use MariaDB MaxScale
 - This allows for transparent interaction with the database layer
- <https://mariadb.com/resources/blog/mariadb-skysql-a-second-generation-cloud-database-for-modern-applications/>

MARIADB MAXSCALE

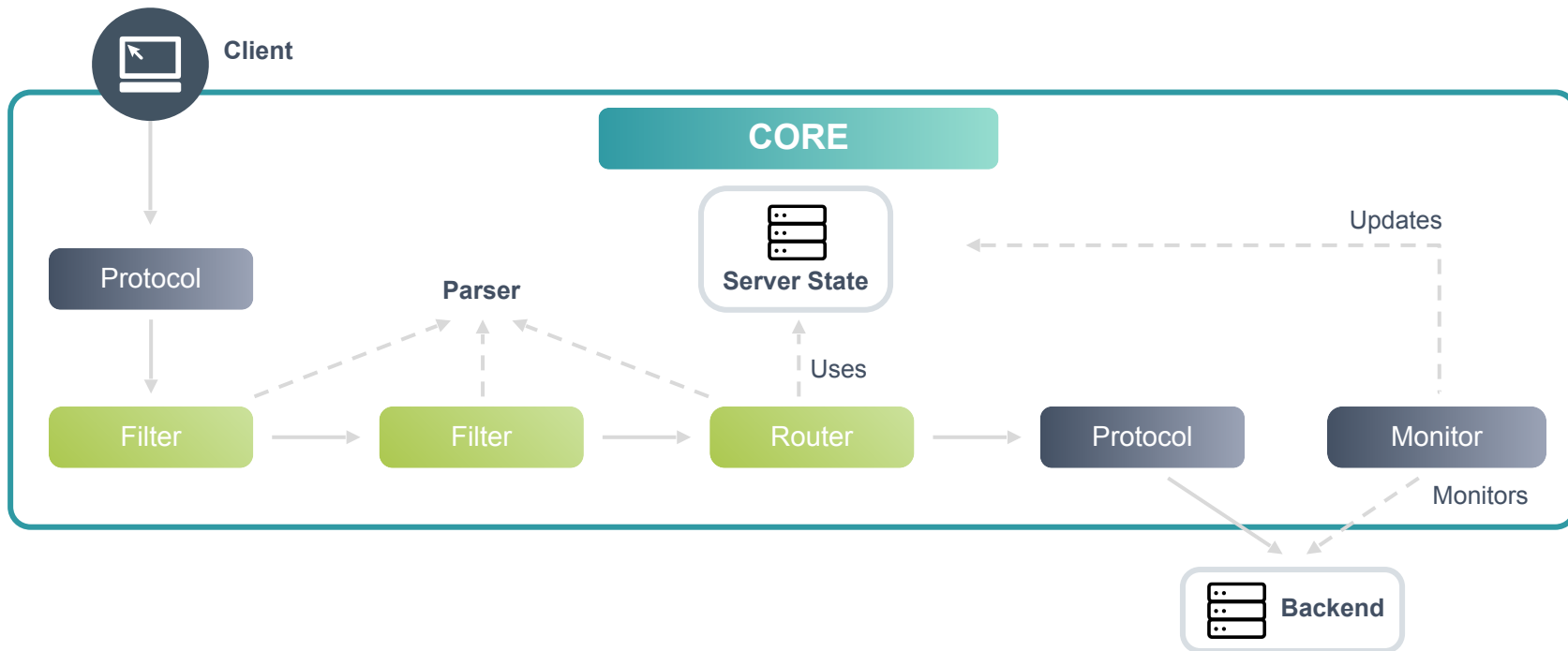
Features

- Insulates client applications from the complexities of the backend database cluster
- Understands the database environment
- Is aware of the state of the database components
- Understands the data that flows through it
- Security improvements to your backend database topology
- GUI, REST API, and command-line tools for operations

Routes requests based on a combo of:

- Defined algorithms
- Component state
- Request contents
- Session state

MAXSCALE INTERNALS



SUPPORTED OS AND DATABASE VERSIONS



OS

RHEL 7 & 8
CentOS 7 & 8
Ubuntu 16.04, 18.04 & 20.04
Debian 9 & 10
SLES 12 & 15



Database

MariaDB - all versions
MySQL 5.5* +



Cluster

MariaDB Cluster
MariaDB/MySQL Primary/Replica
MariaDB ColumnStore
Amazon RDS

* Automatic features such as failover/switchover/automatic rejoin do not work with MySQL as MariaDB has a different GTID implementation that is required by MaxScale. Also, the caching_sha1_password authentication mechanism is also not supported so MySQL 8.0 will need to be configured to use a different default authentication plugin.

WHAT'S NEW FOR MAXSCALE 23.02

- <https://mariadb.com/docs/xpand/release-notes/mariadb-maxscale-23-02/23-02-1/>
- Rebuild Server with MariaDB Monitor
- Dynamic configuration changes for cache filters
- Semi-Synchronous Replication for Binlog Router
- MaxGUI Enhancements
- Dynamic Reload of SSL/TLS Certificates

MAXSCALE CONFIGURATION

Configuration file is `/etc/maxscale.cnf`

Configuration file has sections for

- Servers
- Services
- Listeners
- Monitors
- Filters
- Routers

MaxScale is very configurable. We won't be able to cover all of it today!

SERVERS

- A server represents an individual database server to which a client can be connected via MaxScale
- Backend database servers can be members of multiple services
- Servers as well as monitors, listeners, and more can also be created using dynamic commands.

Backend server configuration example

```
[server1]
type=server
address=127.0.0.1
port=3006
```

- Add a section heading for each server
- Set type to server
- Set address to the IP address of the server
- Set the TCP port used to connect to the server
- Usernames and passwords are defined elsewhere

MAXSCALE DATABASE USER

MaxScale needs a database user account on the MariaDB server in order to verify requests and perform database management tasks

This database user's host needs to be set to the MaxScale server's address

```
CREATE USER 'max_user'@'maxscalehost' IDENTIFIED BY
'mariadb';

GRANT BINLOG ADMIN, READ_ONLY ADMIN, RELOAD,
REPLICATION CLIENT, REPLICATION MASTER ADMIN,
REPLICATION SLAVE ADMIN, REPLICATION CLIENT,
REPLICATION SLAVE, SHOW DATABASES ON *.* TO
'max_user'@'maxscalehost';

GRANT SELECT ON mysql.db TO 'max_user'@'maxscalehost';
GRANT SELECT ON mysql.user TO
'max_user'@'maxscalehost';
GRANT SELECT ON mysql.roles_mapping TO
'max_user'@'maxscalehost';
GRANT SELECT ON mysql.tables_priv TO
'max_user'@'maxscalehost';
GRANT SELECT ON mysql.columns_priv TO
'max_user'@'maxscalehost';
GRANT SELECT ON mysql.proxies_priv TO
'max_user'@'maxscalehost';
```

SAMPLE CONFIGURATION

Configuration file is `/etc/maxscale.cnf`

```
[server1]
type=server
address=127.0.0.1
port=3306
protocol=MariaDBBackend
```

```
[MariaDB-Monitor]
type=monitor
module=mariadbmon
servers=server1
user=myuser
password=mypwd
monitor_interval=2000
...
```

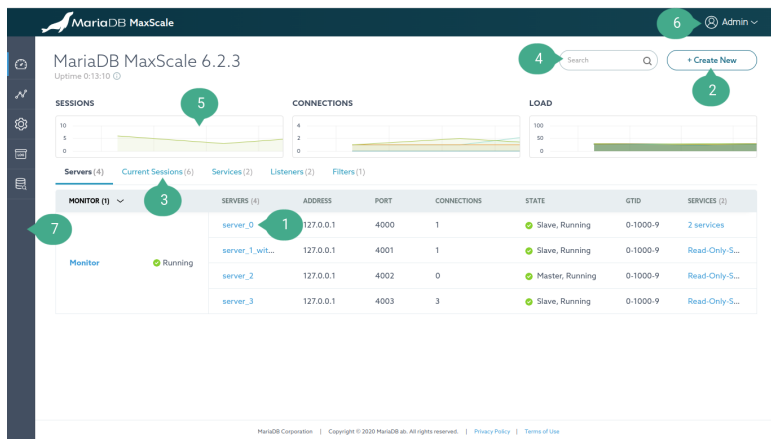
```
...
[Read-Write-Service]
type=service
router=readwritesplit
servers=server1
user=myuser
password=mypwd

[Read-Write-Listener]
type=listener
service=Read-Write-Service
protocol=MariaDBClient
port=4006
```

MAXSCALE GUI CONFIGURATION

```
[maxscale]
...
admin_enabled          = 1
admin_auth             = 1
admin_host             = 0.0.0.0
admin_port             = 8989
admin_gui              = 1
admin_ssl_key          = /var/lib/maxscale/server-key.pem
admin_ssl_cert         = /var/lib/maxscale/server-cert.pem
admin_ssl_ca_cert     = /var/lib/maxscale/ca-cert.pem
```

MAXGUI DASHBOARD



1. [MaxScale object](#). i.e. Service, Server, Monitor, Filter, and Listener (Clicking on it will navigate to its detail page)
2. Create a new MaxScale object.
3. Dashboard Tab Navigation.
4. Search Input. This can be used as a quick way to search for a keyword in tables.
5. Dashboard graphs. Refresh interval is 10 seconds.
 - SESSIONS graph illustrates the total number of current sessions.
 - CONNECTIONS graph shows servers current connections.
 - LOAD graph shows the last second load of thread.
6. Logout of the app.
7. Sidebar navigation menu. Access to the following pages: Dashboard, Visualization, Settings, Logs Archive, Query Editor

MAXSCALE CLI (MAXCTRL)

- list
- show
- set
- clear
- enable
- disable
- create
- destroy
- link
- unlink
- start
- stop
- alter
- rotate
- call
- cluster
- api

Example for "list"

Usage: list <command>

Commands:

servers	List servers
services	List services
listeners <service>	List listeners of a service
monitors	List monitors
sessions	List sessions
filters	List filters
modules	List loaded modules
users	List created network users
commands	List module commands

MAXCTRL COMMANDS

```
list servers|services|listeners|monitors|sessions|filters|modules|users|commands
show server|service|monitor|session|filter|module|maxscale|logging|commands
set server <server> <state>
clear server <server> <state>
enable log-priority|account
disable log-priority|account
create server|monitor|listener|user
destroy server|monitor|listener|user
link service|monitor
unlink service|monitor
start service|monitor|maxscale
stop service|monitor|maxscale
alter server|monitor|service|logging|maxscale
rotate logs
call command
cluster diff|sync
api get
```


SERVER DETAILS

Use “show server” to view the details of a server

```
# maxctrl show server some_server
```

Server	server1
Address	server1.example.com
Port	3306
State	Master, Running
Last Event	server_up
Triggered At	Fri, 14 Dec 2018 22:18:47 GMT
Services	Router1
Monitors	Monitor
Master ID	-1
Node ID	1723144195
Slave Server IDs	

Statistics	{ "connections": 2, "total_connections": 1233790, "persistent_connections": 0, "Active_operations": 0, "routed_packets": 67648583, "adaptive_avg_select_time": "0ns" }
Parameters	{ "address": "server1.accessorysupply.com", "port": 3306, "monitoruser": null, "monitorpw": null, "persistpoolmax": 0, "persistmaxtime": 0, "proxy_protocol": false, "ssl": "false", "ssl_cert": null, "ssl_key": null, "ssl_ca_cert": null, "ssl_version": "MAX", "ssl_cert_verify_depth": 9, "ssl_verify_peer_certificate": true, "disk_space_threshold": null, "type": "server" }

LISTING SERVICES

Use `list services` to get a list of services

```
# maxctrl list services
```

Service	Router	Connections	Total Connections	Servers
Router1	readconnroute	2	1223365	server1
Router2	readconnroute	1	150	server2

```
maxctrl stop service Router1
```

```
maxctrl start service Router1
```

Use `stop service` to stop new clients access to service (i.e., Stop Listening)

LIST SESSIONS

Use `list sessions` to see a list of current sessions.

```
# maxctrl list sessions
```

Id	User	Host	Connected	Idle	Service
1212419	monty	::ffff:172.31.43.89	Wed Dec 19 03:30:01 2018	306.8	Router1
1213567	austin	::ffff:172.31.43.89	Wed Dec 19 03:35:05 2018	3.4	Router1
1213488	ben	::ffff:172.31.43.89	Wed Dec 19 03:34:47 2018	19.7	Router1
1213584	app_user	::ffff:172.31.43.89	Wed Dec 19 03:35:08 2018	0	Router1

ROUTERS

- A router is a MaxScale module that routes inbound queries to one or more of the backend servers
- Queries can be routed by connection or statement
- Connection routing is where complete inbound connections are sent one-to-one with a connection on a backend server
- Statement routing is when different statements on an inbound connection may be sent to different backend servers.
- Some routers include:
 - Binlogrouter
 - ReadConnRoute
 - ReadWritesplit
 - SchemaRouter
 - KafkaCDC

READCONNROUTE ROUTER

Provides load balancing
in a cluster; only option is
`router_options`

```
[Read-Service]
type=service
router=readconnroute
servers=slave1,slave2,slave3
router_options=slave
```

Options for `router_options` Parameter of
ReadConnRoute Module

<code>master</code>	Used to Name a Server in Cluster as Primary
<code>slave</code>	Used to Name Servers as Replicas
<code>synced</code>	Galera Cluster Node with is in Synchronized State
<code>running</code>	Server which is Running and to which MaxScale is able to Connect

READWRITESPLIT ROUTER

Divides read-only queries among multiple nodes while sending write queries to a single node (primary)

```
[RW-Split-Router]
type=service
router=readwritesplit
servers=server1,server2
server3
max_slave_connections=100
disable_sescmd_history=1
slave_selection_criteria=LEAS
T_CURRENT_OPERATIONS
```

Router Parameters for <code>readwritesplit</code> Module	
<code>max_slave_connections</code>	Maximum number of replicas allowed simultaneously for a router session
<code>max_slave_replication_lag</code>	Number of seconds a replica may be behind a primary before removed from routing
<code>use_sql_variables_in</code>	Where queries which read session variable be routed – master or all
<code>slave_selection_criteria</code>	Basis of how router chooses replicas and balances loads
<code>max_sescmd_history</code>	Controls the size of the SQL command history - consisting mostly of SET and USE commands - kept in memory that is re-executed whenever a reconnection occurs
<code>prune_sescmd_history</code>	Prunes the session command history when it exceeds the value configured in <code>max_sescmd_history</code> , limiting the per-session memory use while still allowing safe reconnections
<code>master_accept_reads</code>	Send primary read queries – good for small clusters
<code>strict_multi_stmt</code>	When a client executes a multi-statement query, subsequent queries will be routed to the primary to assure a consistent session state
<code>master_failure_mode</code>	Reaction to primary failing – <code>fail_instantly</code> , <code>fail_on_write</code> , <code>error_on_write</code>

READWRITESPLIT ROUTER READING NODE SELECTION

The `slave_selection_criteria` parameter controls how the router chooses a reading node for a query and how it balances traffic load between all reading nodes

Options for <code>slave_selection_criteria</code> option	
<code>LEAST_GLOBAL_CONNECTIONS</code>	Replica with Least Number of Connections to MaxScale in Cluster
<code>LEAST_BEHIND_MASTER</code>	Replica with Least Replication Lag
<code>LEAST_CURRENT_OPERATIONS</code> (default)	Replica with Least Number of Active Operations
<code>ADAPTIVE_ROUTING</code>	Based on server average response times

READWRITESPLIT ROUTER CAUSAL READS

Aims at resolving the problem with reads that come immediately after a write, which may return inconsistent data when replication is asynchronous

```
[RW-Split-Router]
...
causal_reads = local | global | fast
causal_reads_timeout = 10
lazy_connect = true | false
```

Two main modes of operation

- Global mode uses latest known GTID and waits for a replica to replicate
- Fast mode routes queries to the primary if no up-to-date replica is found

FILTERS



May modify, block or log
a request as it passes through
MaxScale

May be built up into chains
May duplicate requests



TYPES OF FILTERS

Binlog	Consistent Critical Read (CCR)
Cache	Comment
Database Firewall	Hint
Insert Stream	Lua
Masking	Maxrows
Named Server	Query Log All
Regex	Tee
Throttle	Top N

DATABASE HIGH AVAILABILITY

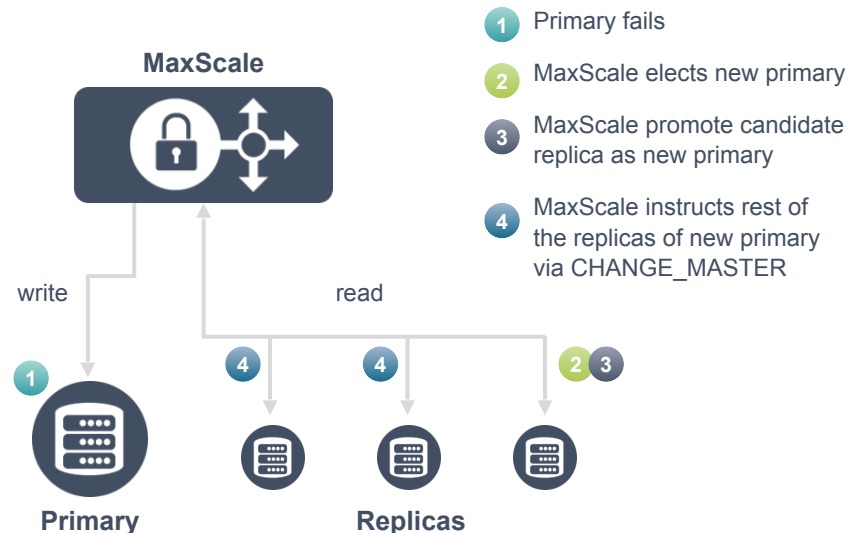
Minimize database service downtime for clients and promote a self-healing cluster

MariaDB Primary-Replica Clusters

- Automatic Failover: election and promotion of replica by MaxScale
- Options to continue routing read queries to replica during failover
- Failover and switchover interface

MariaDB Galera Clusters

- Galera handles node failure
- MaxScale continues routing both read and write queries to nodes in service, while one of the nodes is down



FAILOVER CONFIGURATION

```
[MariaDB-Monitor]
type                = monitor
module              = mariadbmon
servers             = server1,server2,server3
...
backend_connect_timeout = 2
backend_write_timeout   = 2
backend_read_timeout    = 2
backend_connect_attempts = 1
master_conditions      = connected_slave,running_slave
auto_failover          = 1
auto_rejoin            = 1
failcount              = 2
switchover_timeout     = 20
failover_timeout       = 20
```

AUTOMATIC FAILOVER OPTIONS

- Connection Migration: MaxScale will migrate the connection to the new primary
- Session Restore: MaxScale will restore the session to its previous state on the new primary
- Transaction Replay: MaxScale will replay the transaction on the new primary

NEXT STEPS

Check out these sources to learn more about MariaDB

- **OpenWorks sessions to watch**
 - Everything You Ever Wanted to Know About Replication But Were Afraid to Ask
 - What's New in MariaDB Enterprise Server
 - MariaDB Shell: An Upgraded Admin Tool
- **Videos**
 - <https://www.youtube.com/watch?v=snLntgYlevE>
 - Learn how MariaDB MaxScale works with multiple MariaDB instances for HA
- **Reading:**
 - <https://go.mariadb.com/high-availability-guide-MariaDB-whitepaper.html>
 - High Availability guide



THANK YOU



OPENWORKS

BE UNSTOPPABLE