

ZABBIX

SUMMIT
2024

Using Zabbix to monitor MariaDB MaxScale and Galera Cluster



Anders Karlsson

Principal Sales Engineer

Agenda

- Introduction and few words on MariaDB
- What is MariaDB Galera Cluster? And why?
- MariaDB MaxScale – The MariaDB Database Proxy
- High Availability Zabbix repository
- Configuring Zabbix for monitoring MariaDB MaxScale
- Demo
- Conclusions
- Questions and Answers

Introduction



About Anders Karlsson

- I'm **Principal Sales Engineer** at MariaDB
- I'm based in **Ystad** in the very southern part of Sweden
- I have worked with **relational databases** for 40+ years
- Started at **Oracle** in the mid-1980s
- I have since also worked for Informix, MySQL, TimesTen, MariaDB etc.
- I have also had **several different jobs**, beside Sales Engineering, such as Support Engineer, Consultant, Database Architect and more
- When not working I enjoy life with **my twins** and **family**
- My special interests include, but are not limited to **old computers** and **ancient technology** in general



MariaDB Products – Products

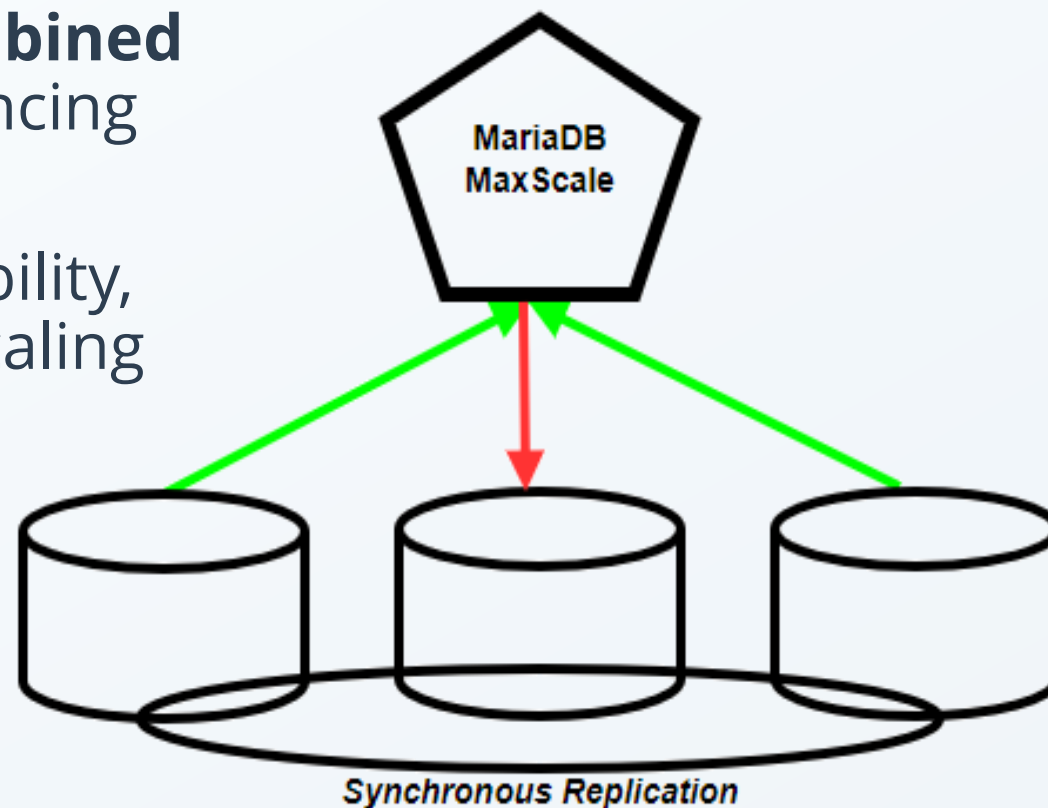
- ▶ **MariaDB Enterprise Server** – An open source database server, based on MariaDB server but with full **long-term support** and **enterprise feature enhancements** such as High Availability, enhanced auditing and **long term support**
- ▶ **MariaDB MaxScale** - A transparent database proxy that works with all the different database servers, providing high availability, filtering, security and scalability
- ▶ **MariaDB Connectors** – Drivers for different languages to connect to MariaDB Server. Includes C, C++, JDBC, ODBC and others
- ▶ **MariaDB Tools** – Tools for managing MariaDB databases

MariaDB Galera Cluster



MariaDB Galera Cluster - Overview

- ▶ **MariaDB Galera Cluster** is a **synchronously** replicated cluster solution
- ▶ MariaDB Galera Cluster is typically **combined with MariaDB MaxScale** for load balancing and fail over
- ▶ The focus for this cluster is High Availability, read-scaling is possible but not write-scaling



MariaDB Galera Cluster - Summary

+ - Pros

- ▶ Using well-known InnoDB engine
- ▶ Read scalability
- ▶ Any data can be queried on any node

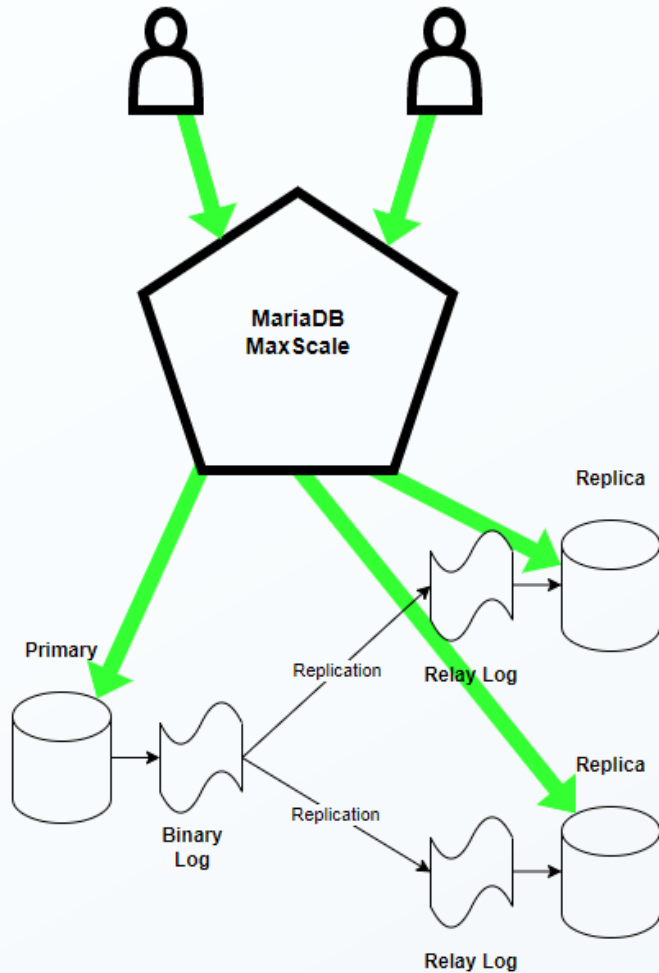
- - Cons

- ▶ All data is on all nodes
- ▶ All data must be replicated to all nodes
- ▶ Limited write scalability
- ▶ Very large databases may be difficult to manage

MariaDB MaxScale



MariaDB MaxScale - Overview

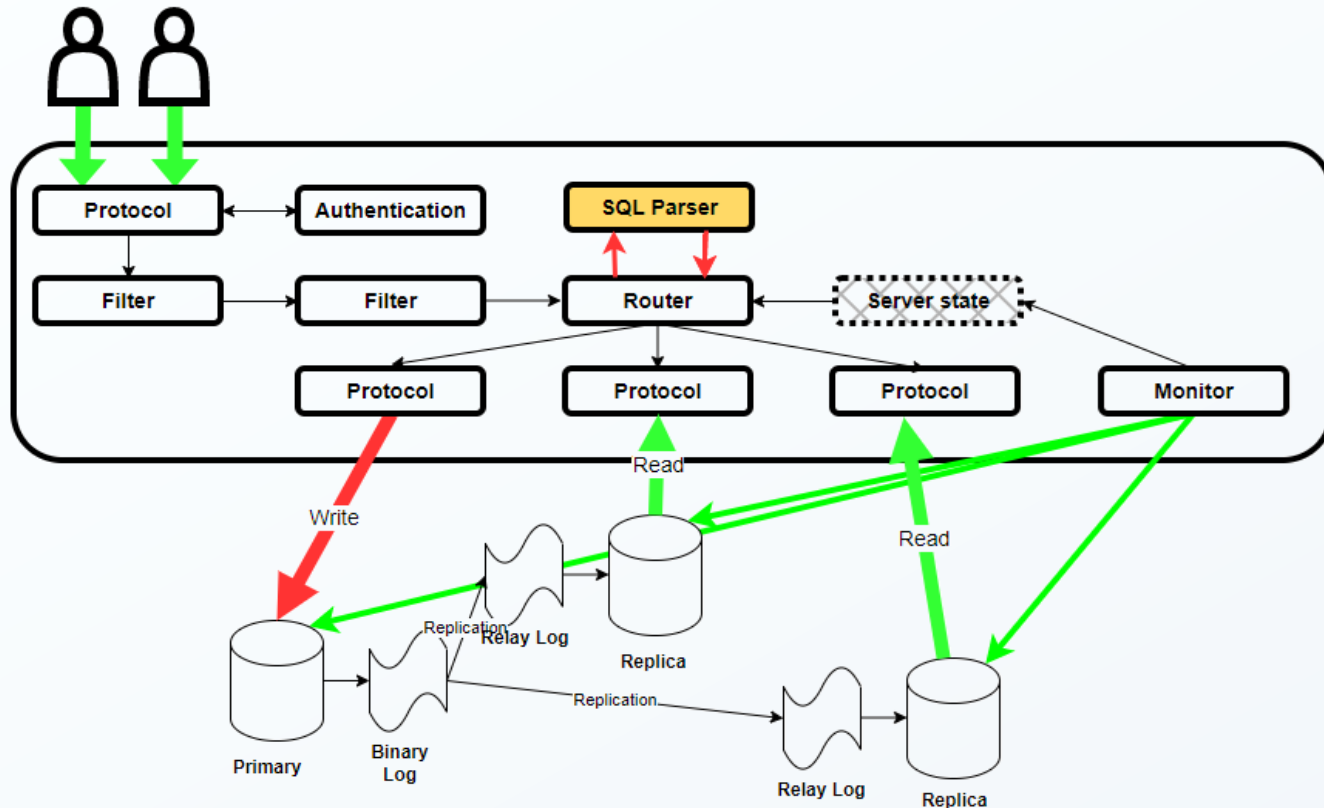


- ▶ MariaDB MaxScale is a **Transparent Database Proxy** placed **in between** the application and the Database Server
- ▶ MariaDB MaxScale is mostly used as a **database router** for clustered database servers to provide High Availability
- ▶ MariaDB MaxScale does have other uses too
- ▶ Data Masking, Streaming, Logging and more
- ▶ MariaDB MaxScale, written in C and C++, is based around **a small lightweight core** and range of plugins

MariaDB MaxScale – High Availability

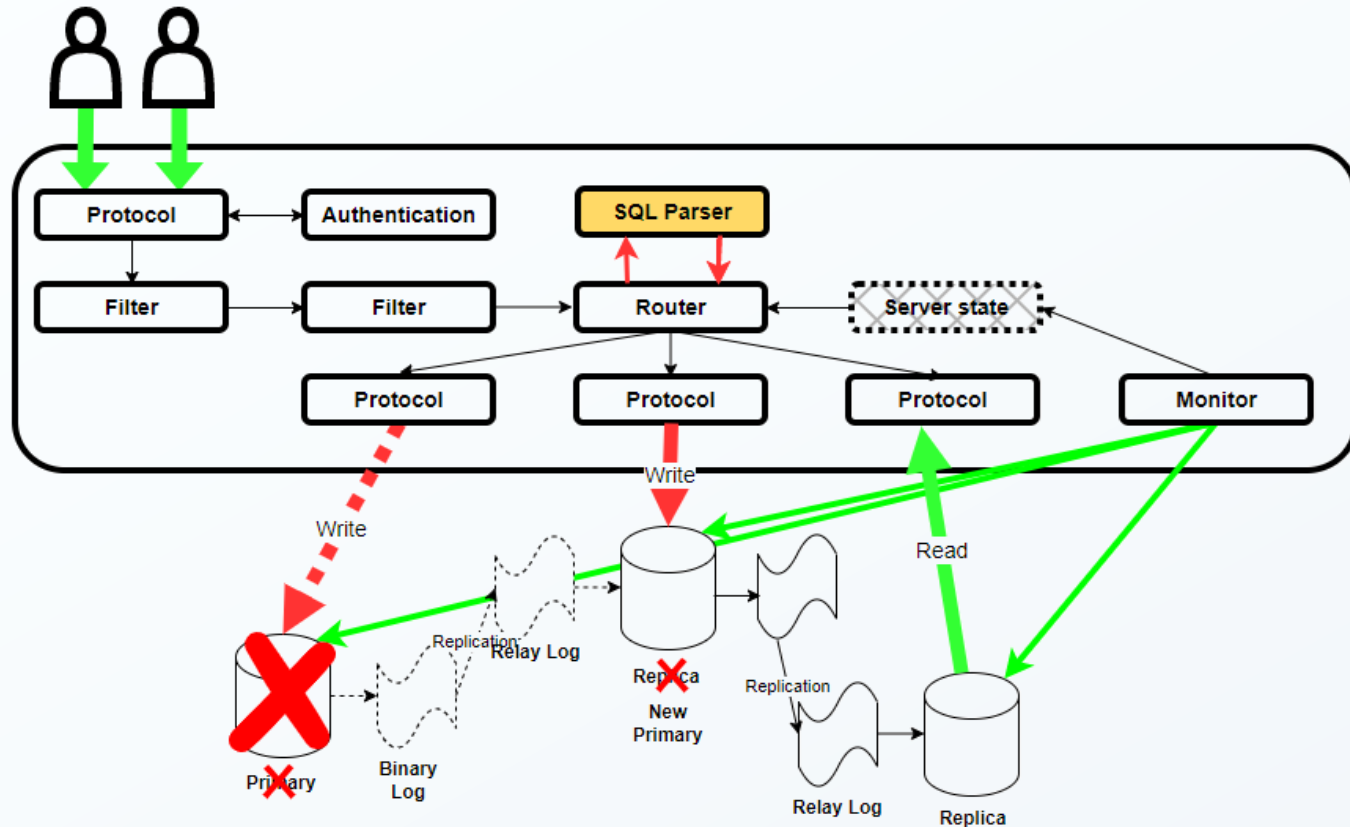
- ▶ MariaDB MaxScale is used to handle High Availability **failover** and **read scalability** with MariaDB database clusters
- ▶ But MariaDB MaxScale is also **High Available in and of itself**
- ▶ MariaDB MaxScale can **synchronize the configuration** between multiple instances
- ▶ MariaDB MaxScale comes with a **GUI**, a **CLI** and an **API**
- ▶ MariaDB MaxScale can be reconfigured without being restarted
 - Nodes can be **enabled** or **disabled**
 - **Routing criteria** can be changed
 - Nodes can be **drained** or taken into **maintenance mode**
 - And more

MariaDB MaxScale – Read scaling



- ▶ The **Monitor** plugin determines which server is the primary and updates **Server State**
- ▶ The **Router** plugin determines if a query is a read or write query using the **SQL Parser**
- ▶ The **Router** module sends write queries to the primary and read queries are **distributed** across the replicas

MariaDB MaxScale – Failover



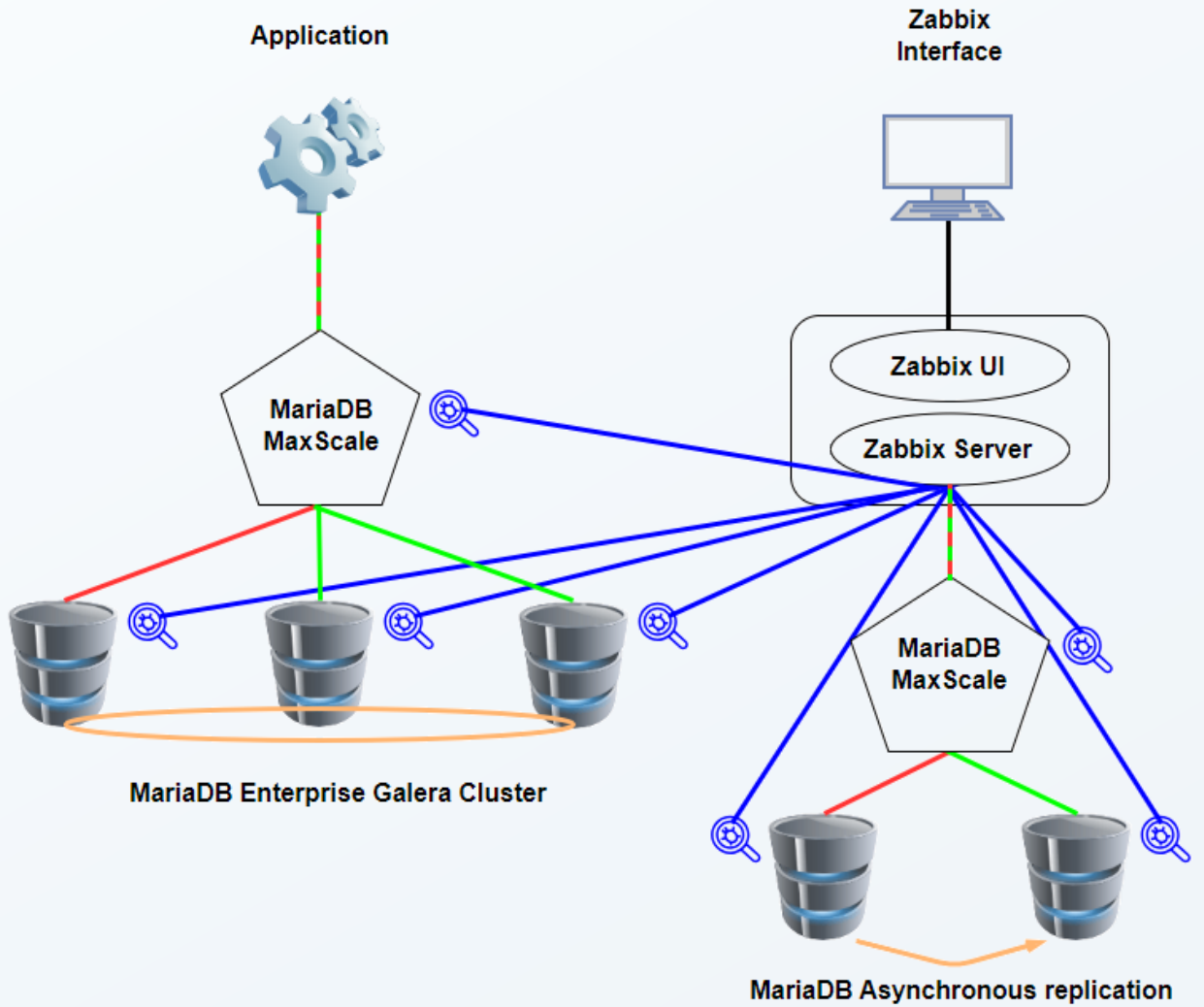
- ▶ The **Monitor** plugin detects a **down Primary**
- ▶ The **Monitor** updates the status of the primary
- ▶ The **Monitor promotes** a Replica to a **New Primary**
- ▶ The **Monitor redirects** other replicas to replicate from the New Primary
- ▶ The **Router** redirects traffic as appropriate

Zabbix High Availability repository



MariaDB MaxScale – Failover

- ▶ The monitored application is using a **MariaDB Enterprise Galera Cluster**
- ▶ **MariaDB MaxScale** is used for load balancing and fail over
- ▶ Zabbix uses a MariaDB Server with **Asynchronous replication** accessed through **MariaDB MaxScale**
- ▶ **MariaDB Servers** are monitored by Zabbix through Agent 2
- ▶ **MariaDB MaxScale** is monitored by HTTP



Zabbix with MariaDB Galera Cluster and MariaDB MaxScale



Galera Cluster – What to monitor

- ▶ Monitoring a database cluster is **different** from monitoring an **individual database server**
- ▶ For one thing, the state of the cluster is somewhat the **sum of the state** of all servers in it
- ▶ Secondly, there are things to monitor that deal with the **interaction between the servers**
- ▶ These are mostly not necessary or relevant to view unless in a non-cluster environment
- ▶ Third, when using a **proxy** or a **load balancer**, the state of this also needs monitoring

Monitoring MariaDB MaxScale

- ▶ MariaDB MaxScale provides a standard **REST/JSON API** that **Zabbix** understands
- ▶ A Zabbix template for MariaDB MaxScale is available at: <https://github.com/nickpyrgio/maxscale-vip>
- ▶ There are other options, but this one works through the **REST API**
- ▶ We will soon look at some **custom monitoring** for MaxScale
- ▶ The nature of MaxScale REST API allows **discovery** of cluster data

Set up MaxScale for Zabbix HTTP

```
[maxscale]
threads=auto
admin_host=192.168.142.240
admin_port=8989
admin_auth=1
admin_enabled=1
admin_secure_gui=false
```

- ▶ Set up admin host to the host where Zabbix Agent runs
- ▶ Set **secure GUI to false** for this demo
- ▶ MariaDB MaxScale can be monitored remotely and by default uses **TLS/SSL**

Zabbix for MaxScale using HTTP

Template ? ×

Template Tags **Macros 4** Value mapping 1

Template macros Inherited and template macros

Macro	Value		Description	
{MAXSCALE.HOST}	192.168.142.240	T ▾	MaxScale host	Remove
{MAXSCALE.PORT}	8989	T ▾	MaxScale port	Remove
{MAXSCALE.PWD}	mariadb	T ▾	MaxScale password	Remove
{MAXSCALE.USER}	admin	T ▾	MaxScale username	Remove

[Add](#)

[Update](#) [Clone](#) [Delete](#) [Delete and clear](#) [Cancel](#)

Zabbix for MaxScale using HTTP

The screenshot shows the Zabbix Item configuration window for an HTTP agent. The configuration is as follows:

- Name:** Get servers
- Type:** HTTP agent
- Key:** maxscale.get_servers
- Type of information:** Text
- URL:** {\$MAXSCALE.HOST}:8989/v1/servers
- Query fields:** A table with one entry: name = value.
- Request type:** GET
- Request body type:** Raw data
- Request body:** (Empty text area)
- Headers:** A table with one entry: name = value.
- Required status codes:** 200
- Follow redirects:**
- Retrieve mode:** Body
- Convert to JSON:**

Buttons at the bottom: Update, Clone, Test, Delete, Cancel.

MaxScale / Galera Macros

Template ? ×

Template Tags **Macros 1** Value mapping

Template macros Inherited and template macros

Macro	Value	Description
<input data-bbox="183 554 665 586" type="text" value="{MAXSCALE.CLUSTER.MONITOR.NAME}"/>	<input data-bbox="685 554 1192 586" type="text" value="monitorabc"/>	<input data-bbox="1284 554 1862 586" type="text" value="Name of the monitor for the Galera cluster"/> <a data-bbox="1870 554 1964 586" href="#">Remove

[Add](#)

MaxScale / Galera Discovery

Discovery rule Preprocessing 1 **LLD macros 1** Filters Overrides

LLD macros

LLD macro	JSONPath	
<input data-bbox="766 429 1345 482" type="text" value="{#MAXSCALE.CLUSTER.MONITOR.NAME}"/>	<input data-bbox="1360 429 2061 482" type="text" value="\$..id"/>	Remove

[Add](#)

MaxScale / Galera Discovery

Item prototype

Item prototype Tags Preprocessing

Parent items MariaDB MaxScale

* Name Cluster {#MAXSCALE.CLUSTER.MONITOR.NAME} current reads

Type Calculated

* Key maxscale.cluster.current.reads[{#MAXSCALE.CLUSTER.MONITOR.NAME}]

Type of information Numeric (unsigned)

* Formula

```
last(/maxscale.cluster.reads[{#MAXSCALE.CLUSTER.MONITOR.NAME}]) - last(/maxscale.cluster.reads[{#MAXSCALE.CLUSTER.MONITOR.NAME}], #2)
```

Units

* Update interval 10s

Custom intervals

Type	Interval	Period	Action
Flexible Scheduling	50s	1-7,00:00-24:00	Remove

Add

* History Do not store Store up to 31d

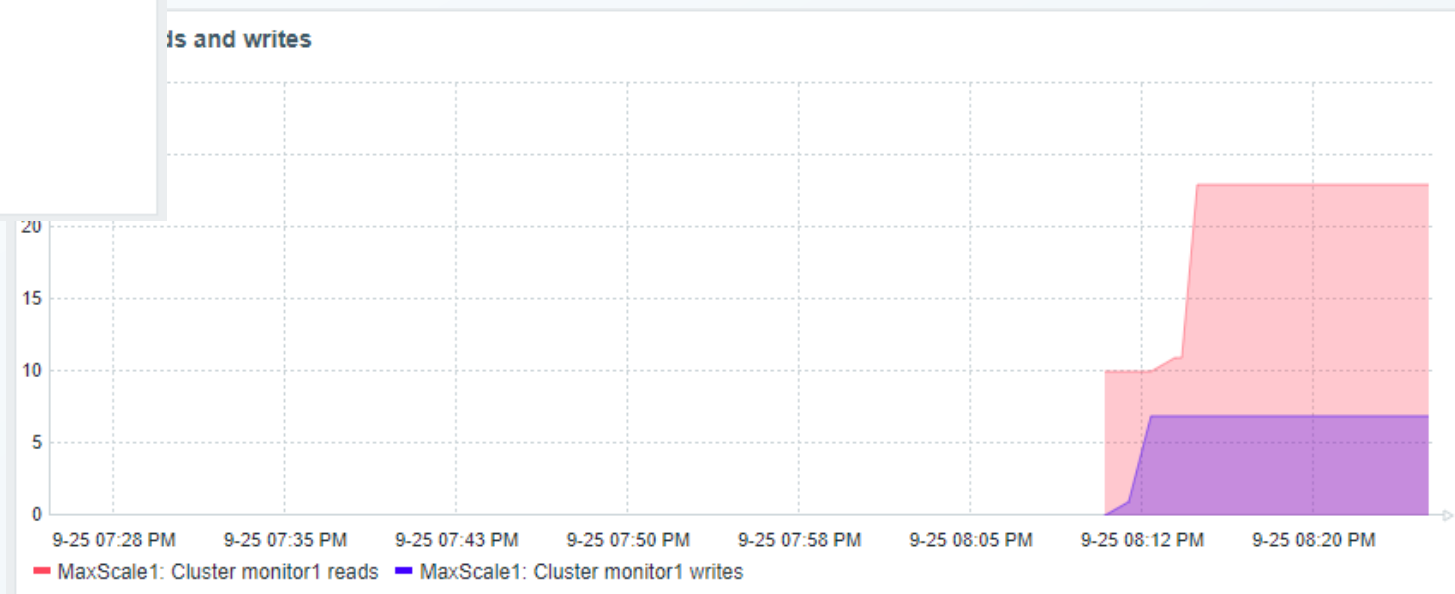
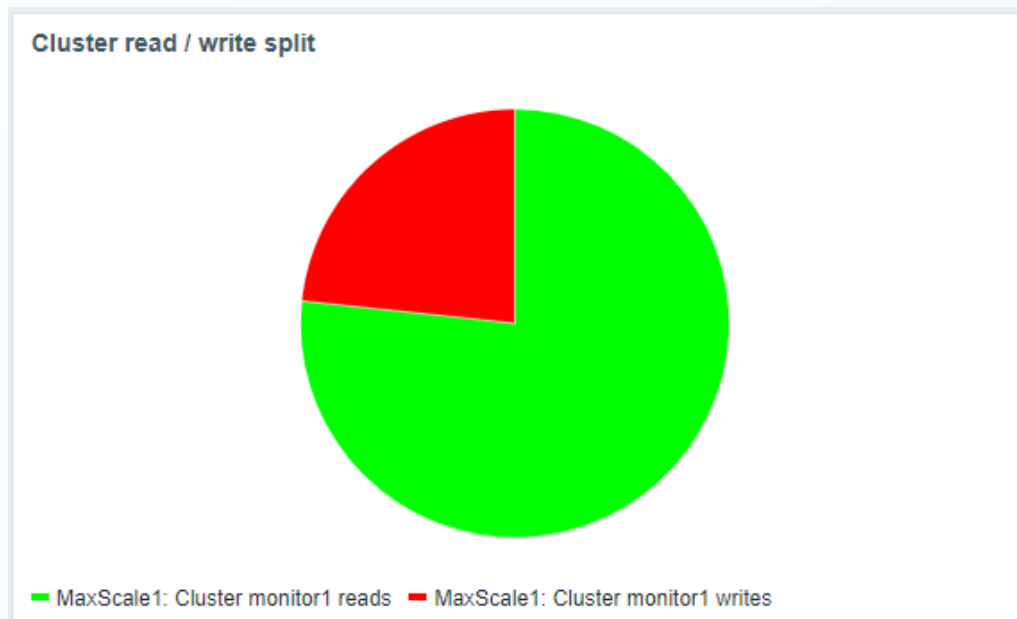
* Trends Do not store Store up to 365d

Value mapping Select

Description

Update Clone Test Delete Cancel

Graph samples



Demo



Conclusions



Conclusions

- ▶ **MariaDB Galera Cluster** is a powerful High Availability solution with read scale-out and distributed cluster capabilities
- ▶ **MariaDB MaxScale** is a transparent database proxy supporting failover and load balancing
- ▶ **MariaDB MaxScale** can be monitored using **REST / JSON**
- ▶ **Zabbix** works well with both MariaDB Galera Cluster and MariaDB MaxScale
- ▶ Using the REST / JSON interface, **Zabbix** can use MariaDB MaxScale for discovery
- ▶ **Zabbix** with MariaDB MaxScale also supports cluster wide monitoring
- ▶ Zabbix can use a **High Available MariaDB Cluster** as repository

ZABBIX
SUMMIT
2024



Anders Karlsson

Principal Sales Engineer