

# MySQL Database Cluster

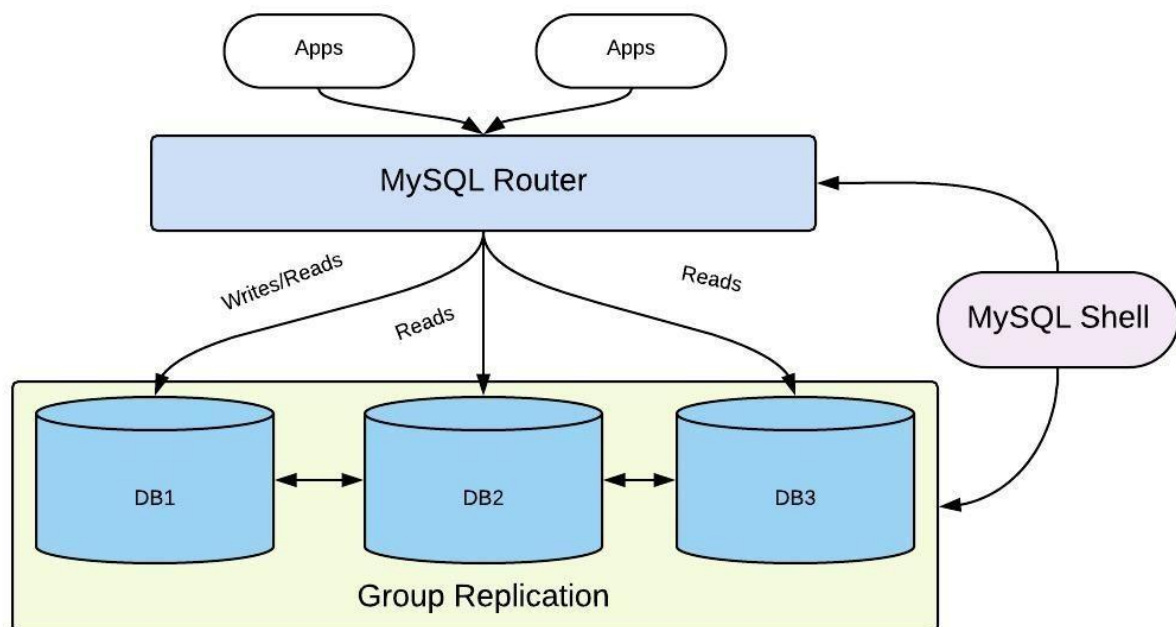
[Reza Asadi](#)

DevOps Engineer

MySQL Cluster is a distributed database management system that provides high availability, fault tolerance, scalability and improved performance. In other words, it is a powerful tool that allows businesses to manage large amounts of data and ensure data stability and durability. In this article, we will examine the performance of this practical tool and examine its numerous advantages.

This powerful tool is designed to provide a reliable and scalable database solution for applications. This process is built on software architecture principles, where each node in the cluster operates independently. This architecture ensures that the failure of a single node does not affect the availability and performance of the entire system.

One of its key features is the ability to automatically distribute data across multiple nodes. This distribution is achieved through a technique called sharding, where data is divided into smaller pieces and stored on different nodes. This ensures that data is evenly distributed and can be accessed quickly and efficiently.



Another important aspect of MySQL Cluster is its support for in-memory databases. By storing the entire dataset in memory, MySQL Cluster can provide extremely fast read and write performance. Additionally, it provides built-in support for real-time analytics, making it an ideal choice for applications that require rapid data processing.

## Availability and fault tolerance in MySQL cluster

MySQL cluster provides high availability and fault tolerance by replicating data across multiple nodes. Each node in the cluster maintains a copy of the data, ensuring that it is always available even if a node fails. Additionally, MySQL Cluster supports automatic failover, where a node or node fails and is automatically replaced by a standby node. This capability minimizes downtime and ensures continuous operation.

## Scalability and performance improvement with MySQL Cluster

MySQL Cluster is highly scalable and can handle large amounts of data and high transaction rates. As data volume increases, additional nodes can be added to the cluster to distribute the load and improve performance. MySQL clustering also provides horizontal scalability where the cluster can be spread across multiple data centers and enable geographically distributed applications.

In addition, the MySQL cluster provides various performance solutions such as parallel query execution and distributed joins. These optimizations enable faster data processing and improved response time, making it an ideal choice for applications that require real-time analytics and complex queries.

We have decided to cluster the MySQL database using Galera

If MariaDB is installed on your operating system, first you must run this command on your operating system and install MySQL.

```
apt install mariadb-server -y
```

After that, to the extent you want to do the cluster, put the following file in the given address and configure the IP.

```
vim /etc/mysql/conf.d/galera.cnf
```

Here we want to perform cluster operation for three systems, so place the three files in the given addresses and set your IPs based on the mentioned items.

First, stop your MySQL with the following command

```
systemctl stop mysqld.service
```

And then copy the following files to the mentioned address

```
[mysqld]
pid-file      = /var/run/mysqld/mysqld.pid
socket        = /var/run/mysqld/mysqld.sock
datadir       = /var/lib/mysql
log-error     = /var/log/mysql/error.log
```

```
binlog_format=ROW
default-storage-engine=innodb
innodb_autoinc_lock_mode=2
bind-address=0.0.0.0
```

```
# Galera Provider Configuration
wsrep_on=ON
wsrep_provider=/usr/lib/galera/libgalera_smm.so

# Galera Cluster Configuration
wsrep_cluster_name="galera"
wsrep_cluster_address="gcomm://192.168.1.1,192.168.1.2,192.168.1.3"

# Galera Synchronization Configuration
wsrep_sst_method=rsync

# Galera Node Configuration
wsrep_node_address="192.168.1.1"
wsrep_node_name="node-1"
```

### for system two

```
[mysqld]
pid-file           = /var/run/mysqld/mysqld.pid
socket             = /var/run/mysqld/mysqld.sock
datadir            = /var/lib/mysql
log-error          = /var/log/mysql/error.log

binlog_format=ROW
default-storage-engine=innodb
innodb_autoinc_lock_mode=2
bind-address=0.0.0.0

# Galera Provider Configuration
wsrep_on=ON
wsrep_provider=/usr/lib/galera/libgalera_smm.so

# Galera Cluster Configuration
wsrep_cluster_name="galera"
wsrep_cluster_address="gcomm://192.168.1.1,192.168.1.2,192.168.1.3"

# Galera Synchronization Configuration
wsrep_sst_method=rsync

# Galera Node Configuration
wsrep_node_address="192.168.1.2"
wsrep_node_name="node-2"
```

### and for system three

```
[mysqld]
pid-file           = /var/run/mysqld/mysqld.pid
socket             = /var/run/mysqld/mysqld.sock
datadir            = /var/lib/mysql
log-error          = /var/log/mysql/error.log

binlog_format=ROW
default-storage-engine=innodb
innodb_autoinc_lock_mode=2
bind-address=0.0.0.0

# Galera Provider Configuration
wsrep_on=ON
wsrep_provider=/usr/lib/galera/libgalera_smm.so
```

```
# Galera Cluster Configuration
wsrep_cluster_name="galera"
wsrep_cluster_address="gcomm://192.168.1.1,192.168.1.2,192.168.1.3"

# Galera Synchronization Configuration
wsrep_sst_method=rsync

# Galera Node Configuration
wsrep_node_address="192.168.1.3"
wsrep_node_name="node-3"
```

And to run the MySQL service, click the following button for each of them to run MySQL

galera\_new\_cluster

On each cluster, first go into your database and run the following commands in order to see the status of your cluster.

```
mysql -uroot -p
use mysql;
select * from wsrep_cluster_members;
```

The output will surprise you

```
MariaDB [mysql]> select * from wsrep_cluster_members;
```

node_uuid	cluster_uuid	node_name	node_incoming_address
13c342bb-4b4b-11ef-91cd-9e6587f6d3df	9f5e3c05-4b4a-11ef-940a-1b26fd421f27	node-2	AUTO
5628d6eb-4b4b-11ef-9969-527746c59e6c	9f5e3c05-4b4a-11ef-940a-1b26fd421f27	node-1	AUTO
7d05d4f5-4b4b-11ef-ae49-377ce6358f8e	9f5e3c05-4b4a-11ef-940a-1b26fd421f27	node-3	AUTO

```
3 rows in set (0.002 sec)
```

## Conclusion

the step-by-step guide on clustering MySQL using Galera showcases a practical approach to implementing a clustered environment, emphasizing the importance of proper configuration and management to achieve optimal results. By following these guidelines, businesses can leverage MySQL Cluster to maintain data integrity, improve performance, and ensure the reliability of their database systems.