

ORACLE

# Group Replication: Best Practices for Network Deploy

---

**Aníbal Pinto**

MySQL, Group Replication Plugin

07 February 2021



# Agenda

---

- 1 Distributed system resource usage
- 2 Why split network usage
- 3 Configuration
- 4 Improve without split network

# Distributed system and usage of shared resources

---

- Group Replication is a distributed system

# Distributed system and usage of shared resources

---

- Group Replication is a distributed system
- All system components use resources

# Distributed system and usage of shared resources

---

- Group Replication is a distributed system
- All system components use resources
- Separation of resources usage improves system

# Distributed system and usage of shared resources

---

- Group Replication is a distributed system
- All system components use resources
- Separation of resources usage improves system
- We will focus on Group Replication network usage

# Network usage



- Group Replication
  - Group Communication System (GCS)
  - Point-in-time recovery on new member join
- Application usage

# Why split network traffic



## Splitting network traffic

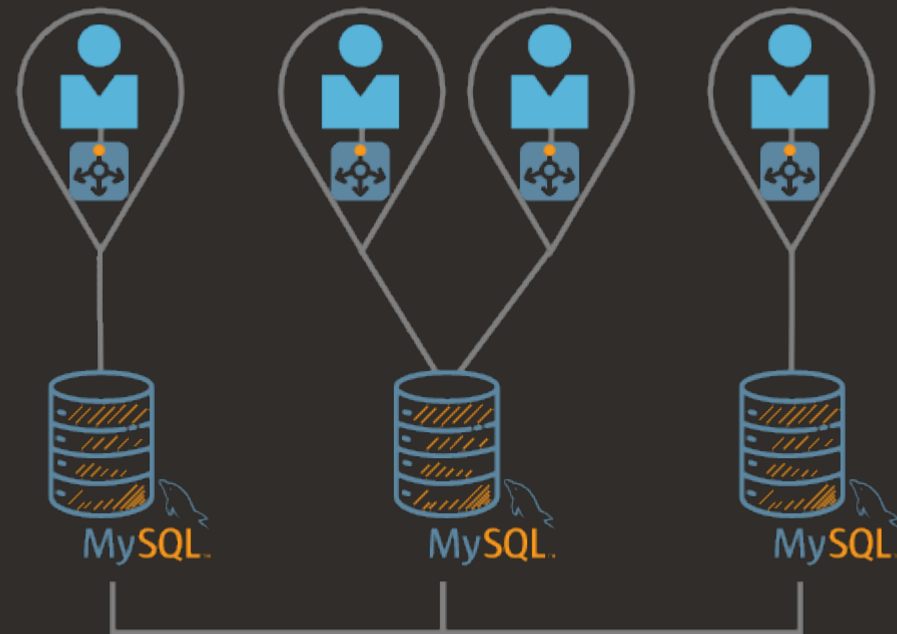
- Internal network has lower latency
- Better performance from Group Replication
- Faster recovery on joining members

## If not possible to split

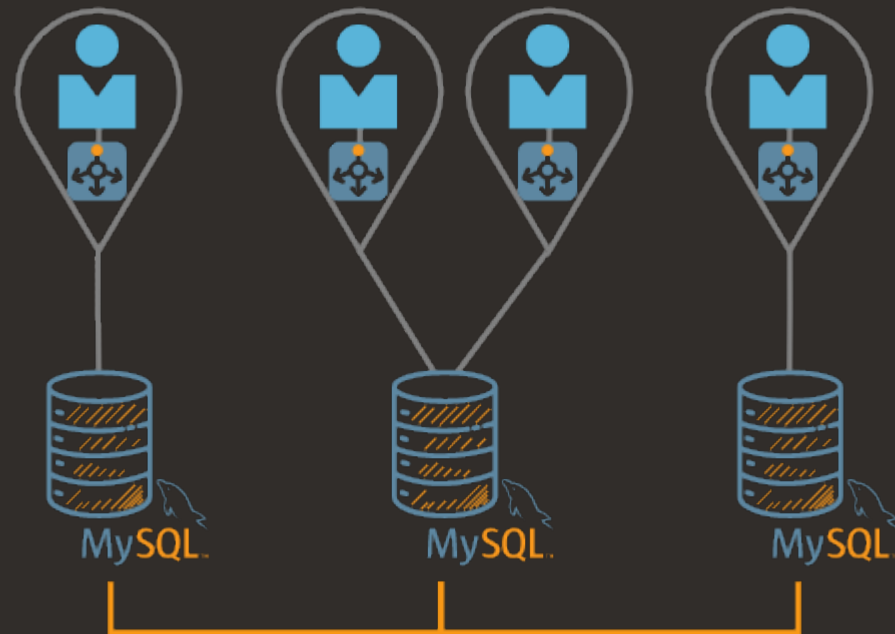
- Quality of Service (QoS) rules to privilege internal replication traffic



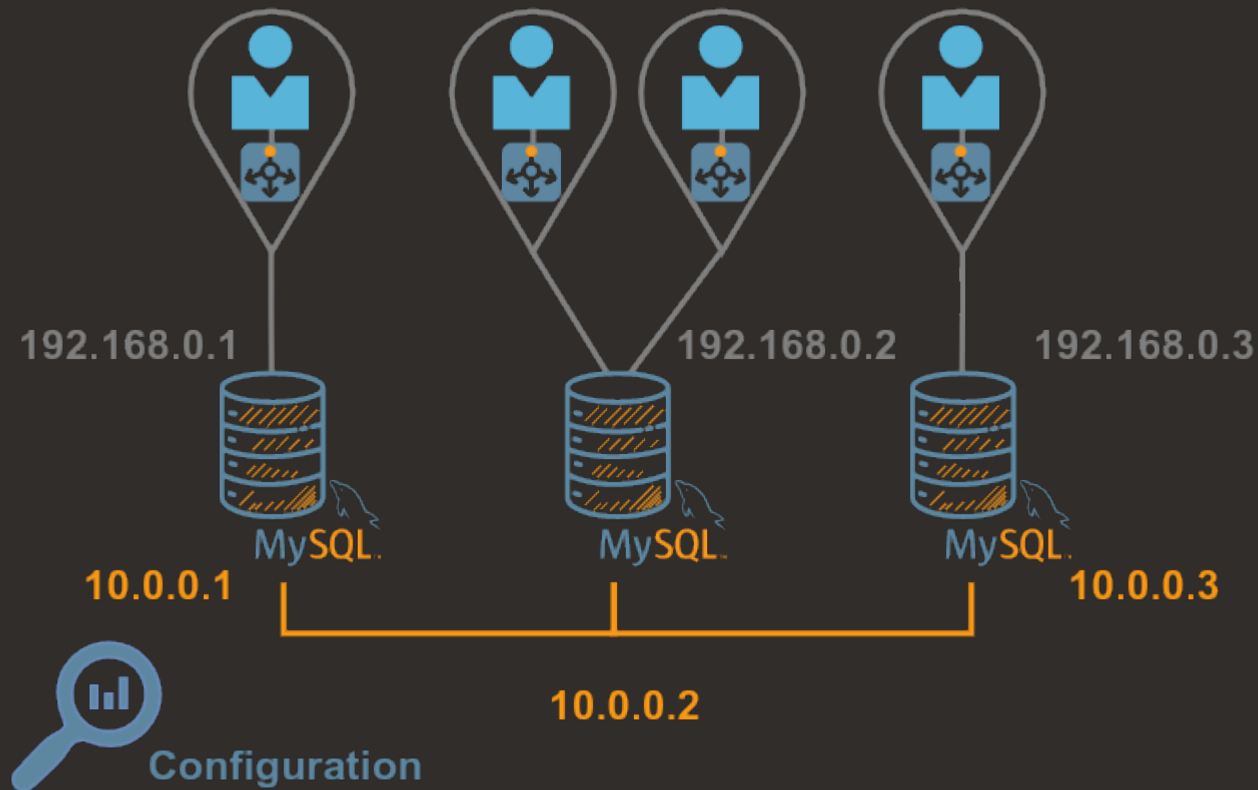
# Basic Setup



# Group Replication using internal network



# Group Replication configuration



# Group Replication configuration



## Group Communication System

- `group_replication_local_address`
- `group_replication_group_seeds`

# Group Replication configuration



## Group Communication System

- `group_replication_local_address`
- `group_replication_group_seeds`

## Recovery

- `group_replication_advertise_recovery_endpoints`

# Group Replication configuration

## Group Communication System

- SET GLOBAL group\_replication\_local\_address = “10.0.0.1:33061”
- SET GLOBAL group\_replication\_group\_seeds = “10.0.0.1:33061,10.0.0.2:33061,10.0.0.3:33061”

## Recovery

- SET GLOBAL group\_replication\_advertise\_recovery\_endpoints = “10.0.0.1:3306”
  - bind\_address must contain both network addresses or wildcard

# Group Replication configuration

## Group Communication System

- SET GLOBAL group\_replication\_local\_address = "10.0.0.1:33061"
- SET GLOBAL group\_replication\_group\_seeds = "10.0.0.1:33061,10.0.0.2:33061,10.0.0.3:33061"

Recovery - we can completely split network using admin\_address and admin\_port

- group\_replication\_advertise\_recovery\_endpoints must use admin\_port
  - SET GLOBAL group\_replication\_advertise\_recovery\_endpoints = "10.0.0.1:33062"
- Implies admin\_address and admin\_port must be set
  - admin\_address= 10.0.0.1
  - admin\_port= 33062
- Privilege SERVICE\_CONNECTION\_ADMIN must be granted to the recovery user
  - GRANT SERVICE\_CONNECTION\_ADMIN ON \*.\* TO "user";

# Internal network not feasible



Some features that can help

- Group Replication member expel timeout
- Auto-rejoin

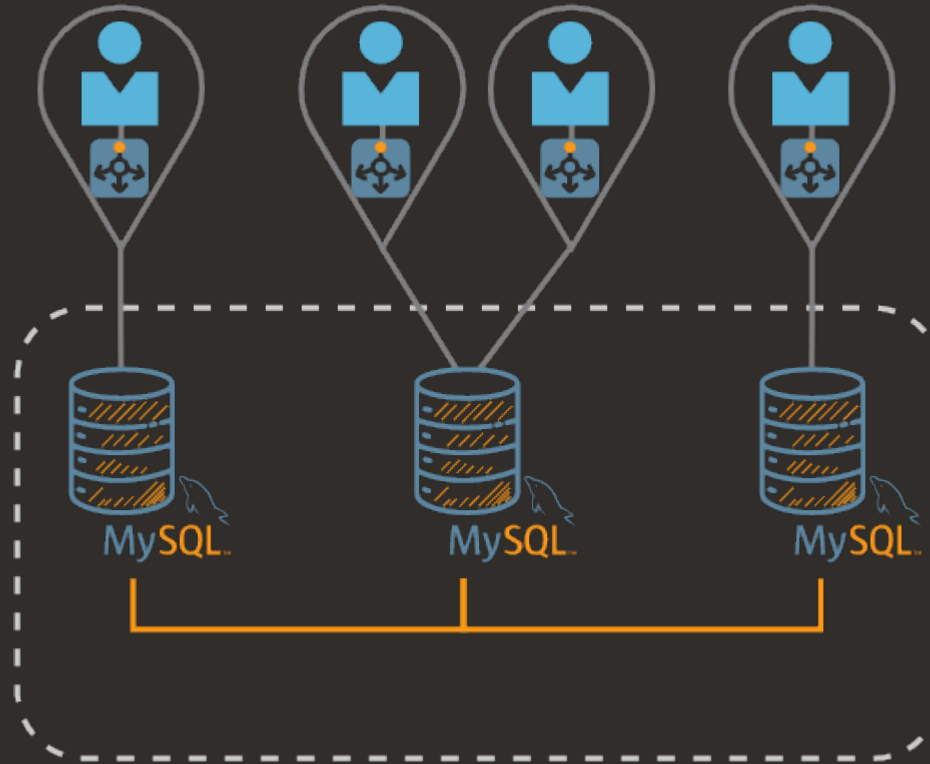


# Member expel timeout

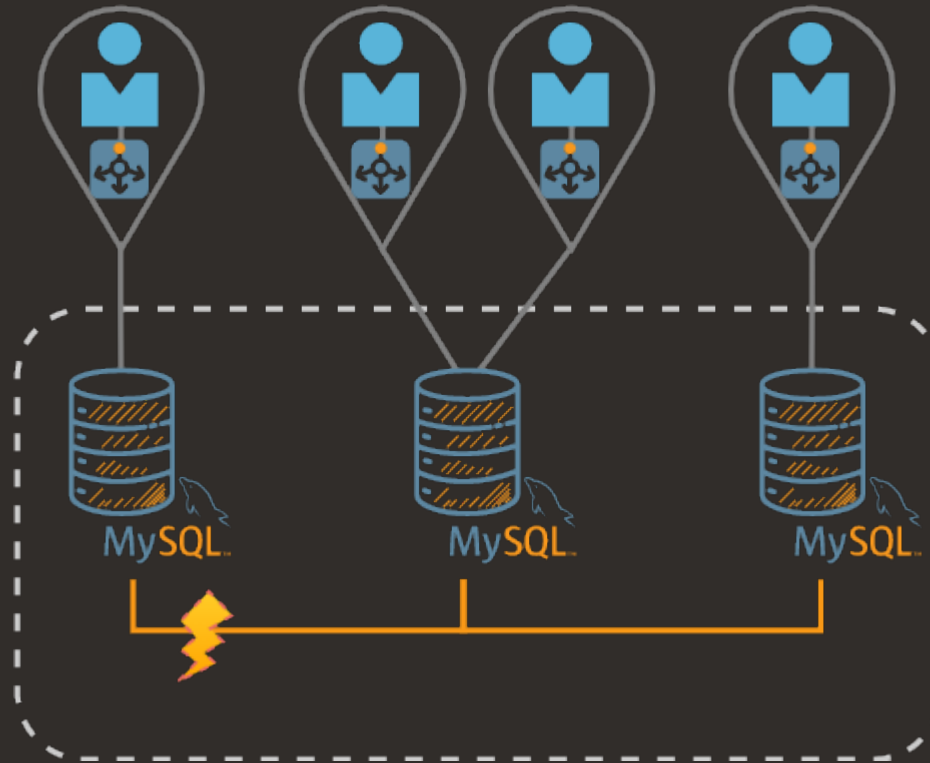


- Nodes will wait 5 second without contact from member to start expel process
- During expel timeout, node will be UNREACHABLE
- After expel timeout, node will be removed from group
- SET GLOBAL group\_replication\_member\_expel\_timeout=<seconds>

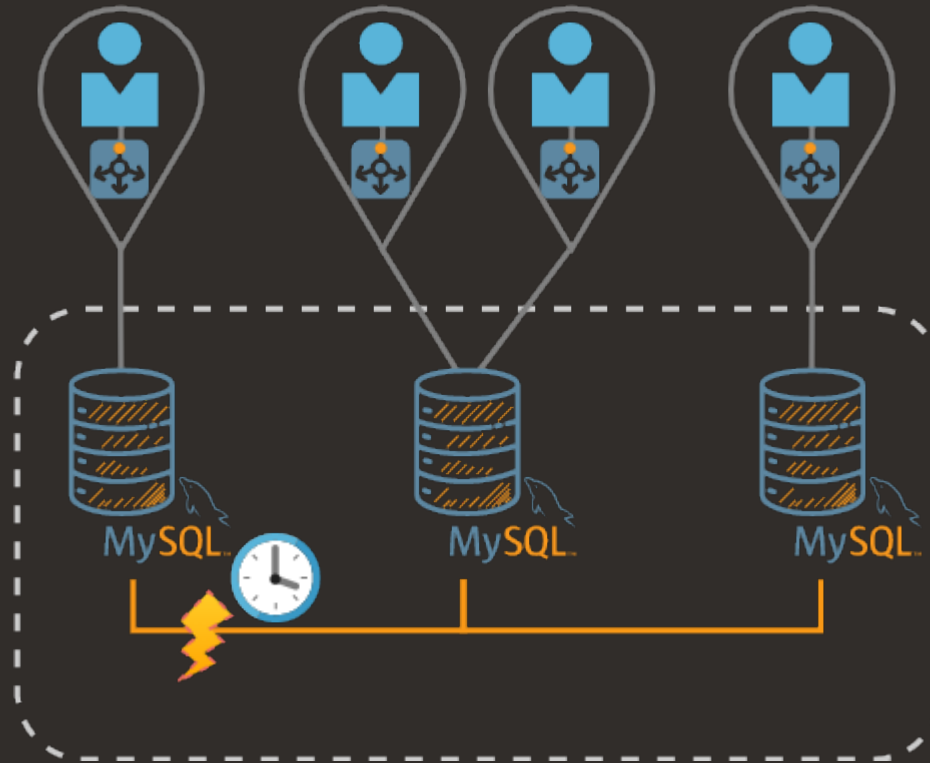
# Member expel timeout



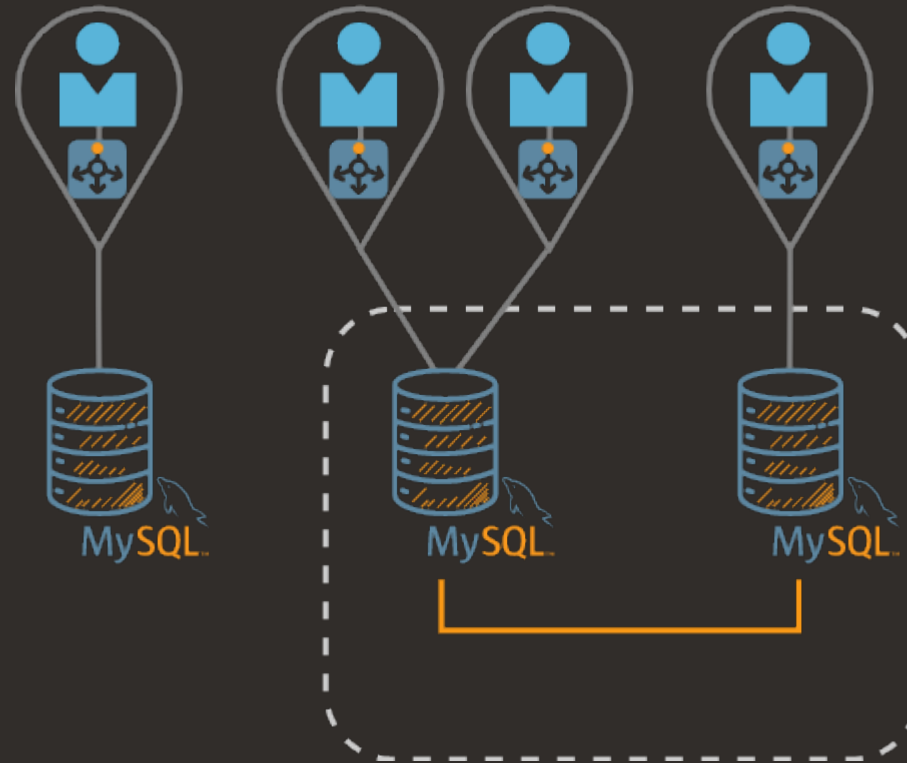
# Member expel timeout



# Member expel timeout



# Member expel timeout

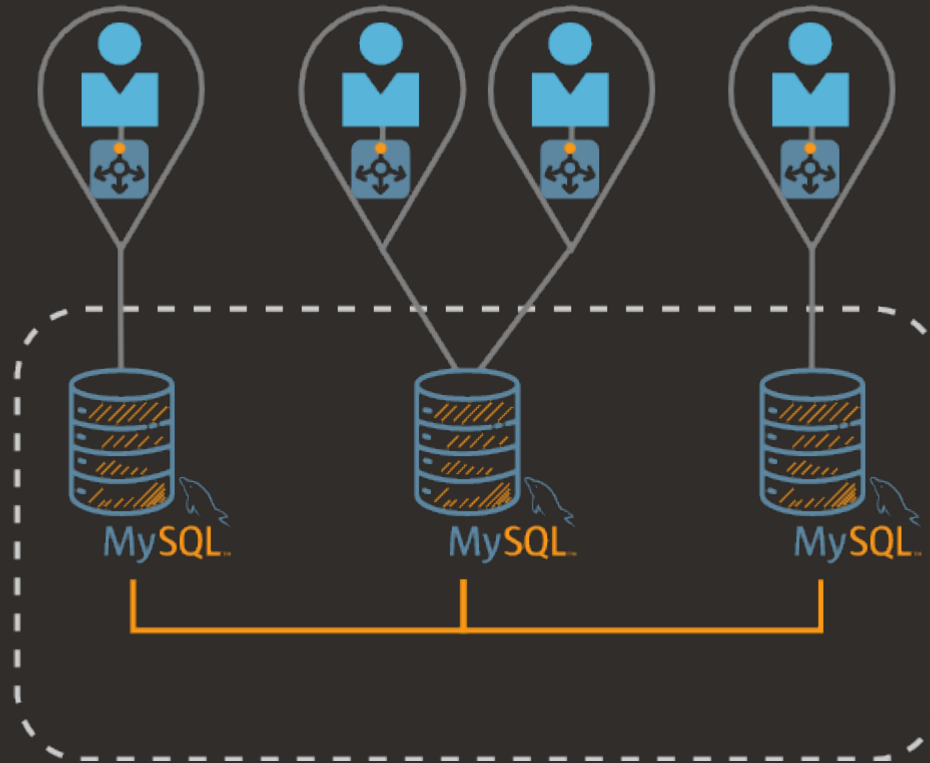


# Auto-rejoin

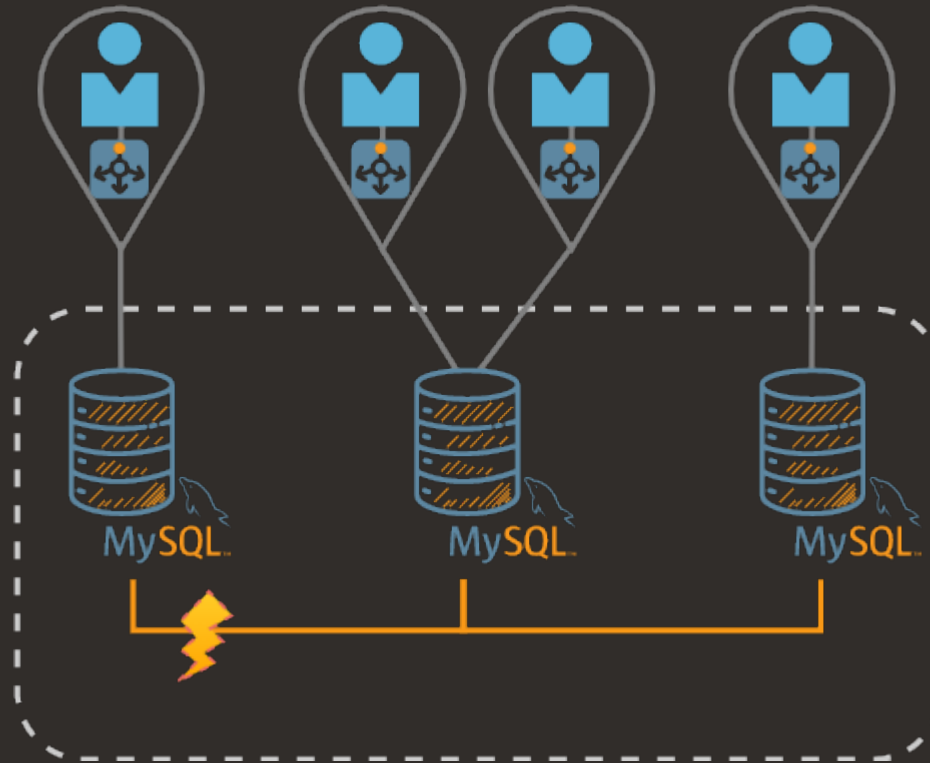


- Node is expelled if not detected for some time by other members
- Due to a network split or high network latency
- On quick network reestablishment the node can reconnect to group
- SET GLOBAL group\_replication\_autorejoin\_tries = <number>

# Auto-rejoin

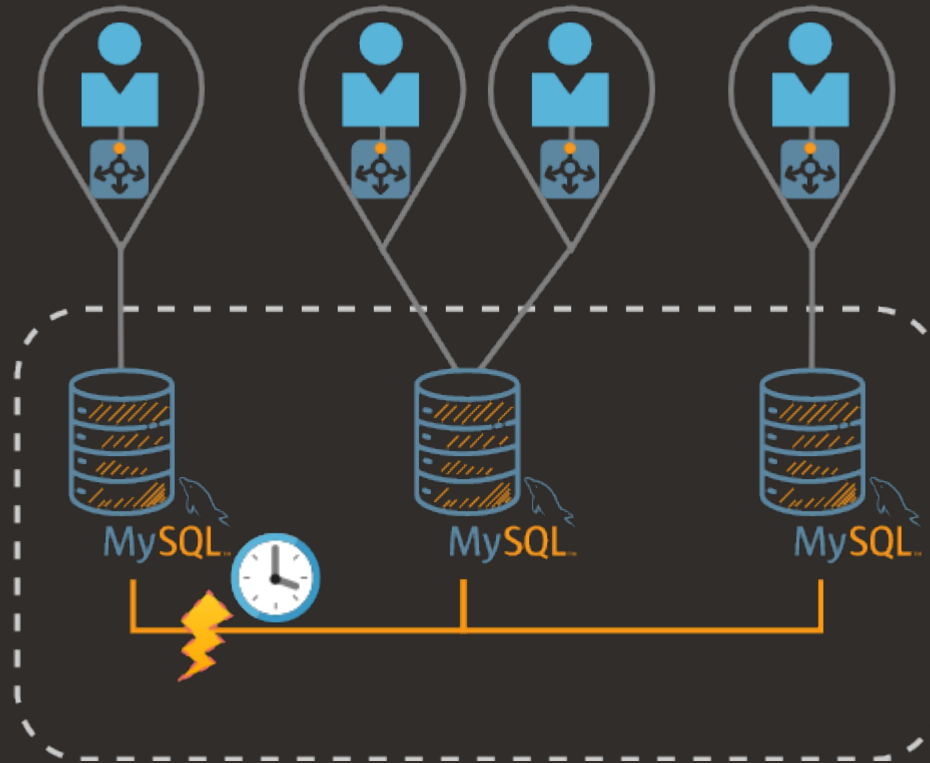


# Auto-rejoin

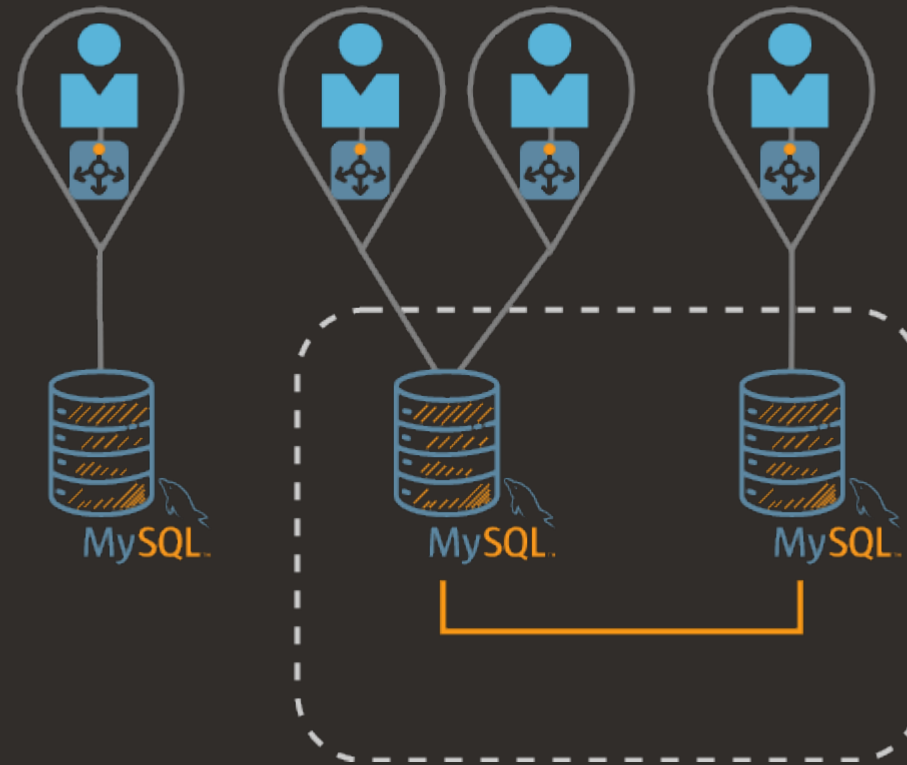




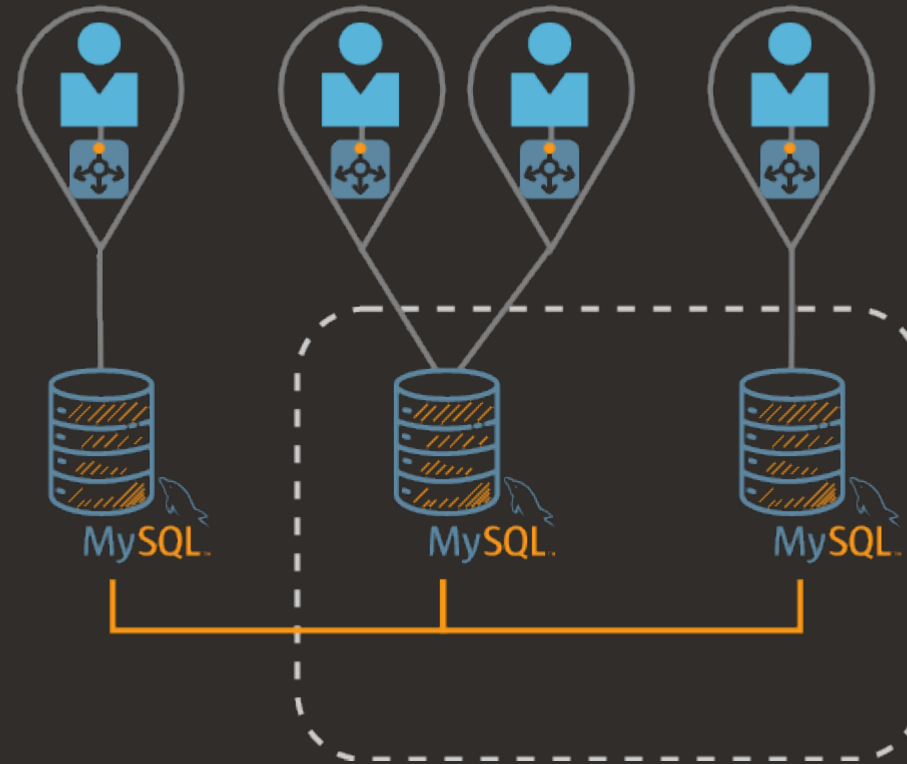
# Auto-rejoin



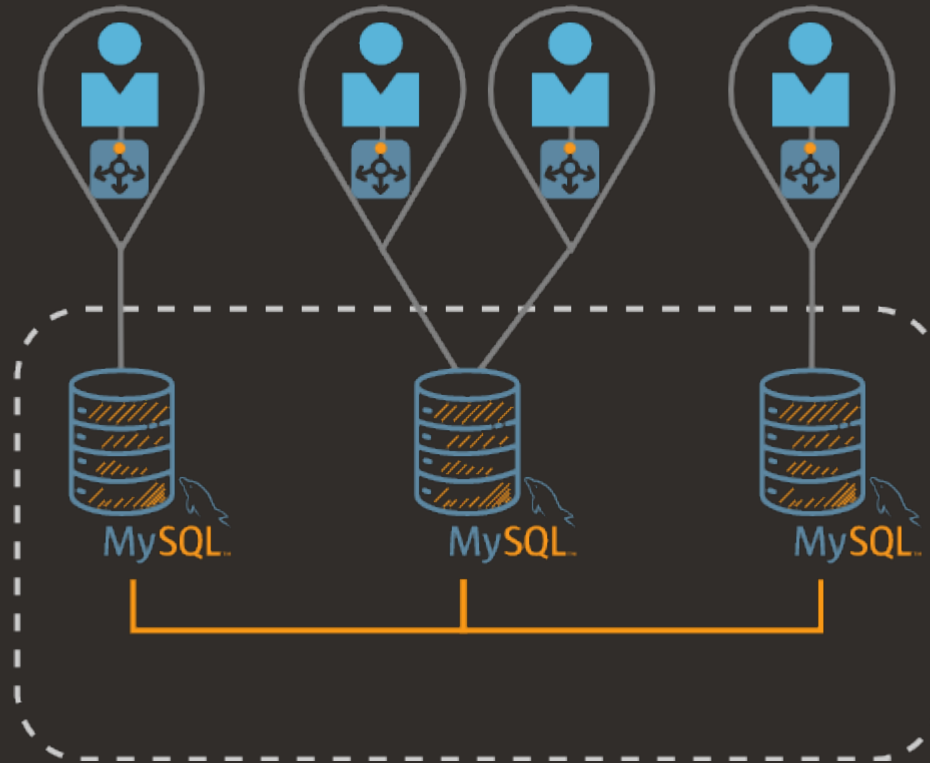
# Auto-rejoin



# Auto-rejoin



# Auto-rejoin



# Thank You

---

<https://mysqlhighavailability.com>





# ORACLE