#### ORACLE

# **Observing and Operating Group Replication**

A continuation from last year's session! New network metrics and more!

**Luís Soares** Senior Software Development Director MySQL, Oracle February, 2, 2024

### **Safe Harbor Statement**

The following is intended to outline our general product direction. It is intended for information purpose only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied up in making purchasing decisions. The development, release and timing of any features or functionality described for Oracle's product remains at the sole discretion of Oracle.

#### Who am I?



Luís Soares MySQL Replication Team Lead Oracle

- Born and raised in Portugal
- Sports: Football, Basket, Karate, Running, Biking
- Physics, Astronomy
- Fault-Tolerance, High Availability, Computers
- Read, Travel, Being with People
- Long time MySQLer

# Agenda

- Introduction
- Observe
- Automate
- Operate
- MySQL Heatwave Service
- Conclusion

| lotro du ctiono  |
|--|
|  |
| IIIIUUUUUUU  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| 5 Convright © 2024: Oracle and/or its affiliates 02 February 2024  |
| 5 Copyright © 2024, Oracle and/or its affiliates 02, February 2024 |
| 5 Copyright © 2024, Oracle and/or its affiliates 02, February 2024 |

# **MySQL** Heatwave Service

Observe, Automate, Operate

#### MySQL Replication powers key features like, but not limited to:

- High Availability
- Inbound Replication
- Outbound Replication
- Managed Read Replicas
- Point-in-time Recovery

#### Simple, intuitive, one-click operations:

- Create DB Systems
- Create Read Replica
- Create Inbound Channel

# Requires a solid, stable and scale aware framework behind it.

#### The need to run, monitor and operate:

- At scale
- Exposed to heterogeneous workloads
- Coping with network bursts or packet delays
- Dealing with the "world" splitting
- Through maintenance



| Create a read replica for the DB system dbsystem  | Create channel   |   |                    |
|---|--|---|--------------------|
| me  | Create a channel for the DB system   | i dbsystem  |                    |
| tysopreadreptics20230130171946  | Name: dbsystem<br>OCID: Show Cody  |   |                    |
| re a description  | 復日 Hide channel filter options   |   |                    |
| Note that it was a second of the second of 20 system spaces dates questions to<br>active point of a second of 20 system spaces dates questions to<br>extern states of the question. | y work, we when and the provided and the | In terms that are typically filtered card in registration. Poils a filter tampfalte to institut your source.<br>Bern manualy. Some constructions of them night cause unsequentiated musicit, if you want to add your own them, make some you check the <u>MARSA documentation</u><br>B System | ×                  |
|   |  |   | Add another filter |
|   | Standalone   | High Availability   |                    |
|   | Single-instance MySQL DB   | Run 3-node MySQL DB System  |                    |



# **MySQL Everywhere**

Observe, Automate, Operate

- Not only on the the MySQL Heatwave Service
  - MySQL is deployed everywhere with similar requirements
- The toolset makes it remarkably easier
  - InnoDB Cluster (HA, resilient, fault-tolerant)
  - InnoDB ReplicaSet (Asynchronous)
  - InnoDB ClusterSet (Across clusters, Across regions)
- There is still, the need to run, monitor and operate:
  - Possibly at scale
  - Exposed to (potentially bounded but still) heterogeneous workloads
  - Coping with unstable network
  - Dealing with the multiple versions of the world
  - While doing maintenance

# **MySQL Everywhere**

Observe, Automate, Operate

#### Observe

- Instrument, emit
- Learn, understand, diagnose
- Trends and historical data
- Robots first, then Humans

#### Automate

- Balance
- Predict
- Self-heal
- Stabilize

#### Operate

- Plan
- Troubleshoot
- Press buttons, turn nods, flip switches

# A continuation from last year...

Yes, this is a continuation from last year's session.

You can find the slides from last year <u>here</u>.

Servers exchange messages:

- Data (transactions)
- Control Messages (the rest)

XCom (Paxos) for distributed coordination.

- Roles: Leader, Acceptor, Learner
- Terms/Slots/Rounds/Ballots
- Phase 1 (Leader Election)
- Phase 2 (Acceptance)
- Phase 3 (Learn/Dissemination)
- Optimizations:
  - Multi-Paxos (accept phase on stable leader)
  - Mencius (streams, empty proposals/no-ops)

Proposing

- Empty proposals may exist to make progress
  - All servers propose (multi-leader, default)
    - Nothing to propose  $\rightarrow$  empty proposal
  - One server proposes (single leader)

Agreeing

- Fast: accept/learn
  - Stable and reliable network; stable servers; fast servers; non-thrashing hosts
- Slower, extra phase: prepare, accept/learn
  - Unstable network; packet loss; jitter; high latency; packet bursts; unstable servers;

There is in an excellent session, delivered at this event too sometime ago, on this topic alone:

"Group Replication: A Journey to the Group Communication Core" (slides)

And there is a great blog post about it too:

"The king is dead, long live the king": Our Paxos-based consensus"

New MySQL Status Variables

#### Data Messages

- Sent counter
- Sent bytes
- Sent round-trip time

#### **Control Messages**

- Sent counter
- Sent bytes
- Sent round-trip time

#### **Transactions Commited Everywhere**

- Garbage collections count
- Garbage collections time

#### **Consensus / Paxos**

- Total rounds Time
- Total rounds counter
- Bytes sent
- Bytes received
- 3-phase (slower) rounds count
- Empty proposals (no-ops) rounds count
- Last successful round timestamp

#### **Consistency Protocol Messages**

- AFTER sync, wait time and count
- BEFORE sync, wait time and count
- AFTER termination, wait time and count



Servers can "disappear" ... and maybe even also comeback before a reconfiguration is triggered!

- Suspected as failed, Detected as failed, Indeed failed (and therefore evicted).
- Suspicions can have a cost, since suspected servers may need to propose no-ops.

| <pre>mysql&gt; select MEMBER_ID,MEMBER_HOST,MEM</pre> | ABER_STATE,MEMBER_ROLE from performance_schema.replic   | cation_group_members; /* on server 3 */ |
|---|---|---|
| MEMBER_ID   | I MEMBER_HOST I MEMBER_STATE I MEMBER_ROLE I  |   |
| 00server1<br>  00server2<br>  00server3               | 192.168.0.1   ONLINE   PRIMARY  <br>  192.168.0.2   UNREACHABLE   SECONDARY  <br>  192.168.0.3   UNLINE   SECONDARY |   |
| 3 rows in set (0.00 sec)                              | -++   |   |
|   |   | IIn + 0 MVSOI & 0 X                     |

Servers can "disappear" ... and maybe even also comeback before a reconfiguration is triggered!

- Suspected as failed, Detected as failed, Indeed failed (and therefore evicted).
- Suspicions can have a cost, since suspected servers may need to propose no-ops.

| <pre>mysql&gt; select MEMBER_ID,MEMBER_HOST,MEME </pre>   | ER_STATE,MEMBE   | R_ROLE from per                      | rformance_schem                           | a.replication_grou | up_members; / | /* on server 3 */ |
|---|--|--------------------------------------|---|--------------------|---------------|-------------------|
| I MEMBER_ID   | MEMBER_HOST  | MEMBER_STATE                         | I MEMBER_ROLE I                           |                    |               |                   |
| 00server1  <br>  00server2  <br>  00server3   | 192.168.0.1  <br>192.168.0.2  <br>192.168.0.3                  | ONLINE<br>UNREACHABLE<br>UNLINE      | PRIMARY  <br>  SECONDARY  <br>  SECONDARY |                    |               |                   |
| 3 rows in set (0.00 sec)  |  |                                      | +   |                    |               |                   |
|   |  |                                      |   |                    | Up to         | MySQL 8.0.X       |
| <pre>mysql&gt; SELECT * FROM performance_schema ************************************</pre>                            | <pre>.replication_c ************************************</pre> | group_communicat<br>*****            | tion_informatio                           | n∖G /* on server 3 | 3 */          |                   |
| WRITE_CONCURRENCY: 1 PROTOCOL VERSION: 8  | .0<br>0 27   |                                      |   |                    |               |                   |
| WRITE_CONSENSUS_LEADERS_PREFERRED: @<br>WRITE_CONSENSUS_LEADERS_ACTUAL: @<br>WRITE_CONSENSUS_SINGLE_LEADER_CAPABLE: @ | 0server1,00s<br>00server1,00s                                  | server2,00…serve<br>server2,00…serve | er3<br>er3                                |                    |               |                   |
| MEMBER_FAILURE_SUSPICIONS_COUNT: {  | "00server1":0  | ),"00server2":4                      | 4,"00…server3":(                          | 0}                 |               |                   |
| I row in set (0.00 sec).  |  |                                      |   |                    |               |                   |

Servers can "disappear" ... and maybe even also comeback before a reconfiguration is triggered!

- Suspected as failed, Detected as failed, Indeed failed (and therefore evicted).
- Suspicions can have a cost, since suspected servers may need to propose no-ops.

| <pre>mysql&gt; select MEMBER_ID,MEMBER_HOST,MEMB</pre>                                     | BER_STATE,MEMBI   | ER_ROLE from pe   | rformance_sche                        | ma.replica      | tion_group_members; /* on server 3 */  |
|--|---|---|---------------------------------------|-----------------|--|
| MEMBER_ID  | MEMBER_HOST   | I MEMBER_STATE  | I MEMBER_ROLE                         | +<br>           |  |
| 00server1<br>  00server2<br>  00server3  | 192.168.0.1<br>192.168.0.2<br>192.168.0.3               | I ONLINE<br>  UNREACHABLE<br>  UNLINE                             | PRIMARY<br>  SECONDARY<br>  SECONDARY | +<br> <br> <br> |  |
| 3 rows in set (0.00 sec)   |   | +   | +                                     | +               |  |
| <pre>mysql&gt; SELECT * FROM performance_schema ************************************</pre> | a.replication_(<br>************************************ | group_communica<br>********<br>server2,00…serv<br>server2,00…serv | tion_informati<br>er3<br>er3          | on\G /* or      | "Hmm What is happening<br>here? Something is not quite<br>right between server3 and<br>server2. Server2 has (transient)<br>troubles speaking to server3! |
| MEMBER_FAILURE_SUSPICIONS_COUNT: ·<br>1 row in set (0.00 sec).                             | ["00…server1":(   | 0,"00…server2":   | 4,"00…server3"                        | :0}             | Wonder if this is causing 3-<br>phase paxos rounds? "  |

Servers can "disappear" ... and maybe even also comeback before a reconfiguration is triggered!

- Suspected as failed, Detected as failed, Indeed failed (and therefore evicted).
- Suspicions can have a cost, since suspected servers may need to propose no-ops.

| mysql> select * from performan                              | ce_schema.global_s | status where variable_name like 'gr_ex | tended%'; /* on server 3 */   |
|---|--------------------|--|---|
| VARIABLE_NAME   | VARIABLE_VALUE     | +<br>                                  |   |
| Gr_extended_consensus_count<br>+<br>1 row in set (0.00 sec) | 1 95 I             | +<br> <br>+                            | "Indeed, I see the counter growing an<br>therefore seems this server3 is taking<br>over and proposing no-ops on behalf<br>of server2. |

I should fix the network connectivity and perhaps consider deploying single leader mode... Could make sense in this scenario to avoid such transient connectivity issues."

d

| Automat                           | e                 |                   |  |
|-----------------------------------|-------------------|-------------------|--|
|                                   |                   |                   |  |
| 19 Copyright © 2024, Oracle and/c | or its affiliates | 02, February 2024 |  |

#### **Replication Applier Automatically Adds Primary Keys**



#### **Replication Applier Automatically Adds Primary Keys**



# **Replication Applier Automatically Adds Primary Keys**

In the MySQL Heatwave Service

- Create an inbound replication channel
- Configure how to handle tables without primary key
- Replicate

| Generate primary key<br>(GENERATE_IMPLICIT_PRIMARY_KEY)<br>Allow replicating a CREATE TABLE or ALTER TABLE transac-<br>tion with no primary keys and automatically generate a new<br>primary key when adding data to such tables. |
|---|
|   |

# **No System Transactions On Cluster Reconfiguration**

Further automation for membership changes – view change transactions go away



# **No System Transactions On Cluster Reconfiguration**

Further automation for membership changes – view change transactions go away



# **No System Transactions On Cluster Reconfiguration**

Further automation for membership changes – view change transactions go away



| ODCIDIC   |
|---|
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
|   |
| 27 Copyright © 2024, Oracle and/or its affiliates 02, February 2024 |
| 27 Copyright © 2024, Oracle and/or its affiliates 02, February 2024 |

## **Control Assignment of Automatic Primary Key Per Channel**



### **Control Assignment of Automatic Primary Key Per Channel**



# **New Syntax**

New batch of terminology changes

- MySQL continues to improve on the terminology front too.
- Operations may be affected if relying on old and deprecated syntax.
- Note the new syntax.

mysql> PURGE BINARY LOGS; -- replaces PURGE MASTER LOGS

```
mysql> SHOW BINARY LOGS;
-- replaces SHOW MASTER LOGS
```

```
mysql> RESET BINARY LOGS AND GTIDS;
-- replaces RESET MASTER
```

```
mysql> [CREATE | ALTER] EVENT ... DISABLE ON REPLICA;
-- replaces [CREATE | ALTER] EVENT ... DISABLE ON SLAVE
```

# MySQL Heatwave Service

33 Copyright © 2024, Oracle and/or its affiliates

 $\bigcirc$ 

# Database

DB System

#### One-click DB System creation



|   | Q US East (Ashbur   | m) 🗸 🔎 🖉 🗘 🦉   |
|---|---|--|
| Create DB Systen  | n   |  |
| Name  | •   |  |
| mysql-ocw   |   |  |
| The user-friendly name for the DB System  | n. It does not have to be unique.   |  |
| Description Optional  |   |  |
| User-provided data about the DB System  | n.  |  |
| User-provided data about the DB System  | High Availability   | HeatWave   |
| User-provided data about the DB System Standalone Single-instance DB System                                   | High Availability<br>Run a DB system with 3 MySQL<br>instances providing automatic<br>failover and zero data loss         | HeatWave<br>DB System that allows you to en-<br>able HeatWave for accelerated<br>query processing, suitable for<br>running both OLTP and OLAP<br>workloads |
| User-provided data about the DB System Standalone Single-instance DB System Create Administrator Username (i) | h.  High Availability Run a DB system with 3 MySQL instances providing automatic failover and zero data loss  credentials | HeatWave<br>DB System that allows you to en-<br>able HeatWave for accelerated<br>query processing, suitable for<br>running both OLTP and OLAP<br>workloads |

# **Backups**

Manual or Automatic

- Retention Period
- When to Backup
- Full or Incremental
- Point-in-Time Recovery (only non-HA DB Systems)

| Edit Backup Plan   |                        |
|--|------------------------|
| Enable automatic backups<br>Enables automatic backups. You must also specify a retention period, and select  | a backup window.       |
| Backup retention period <i>Optional</i><br>The retention period defines how long to store the backups, in days. (i)  |                        |
| 7  | \$                     |
| Enable point in time restore (i)   |                        |
| Enables you to restore from a DB system at a point in time.  |                        |
| <ul><li>Enables you to restore from a DB system at a point in time.</li><li>Select backup window</li></ul>   |                        |
| <ul> <li>Enables you to restore from a DB system at a point in time.</li> <li>Select backup window</li> <li>The backup window start time defines the start of the time period during which y up.</li> </ul>                            | our DB system is backe |
| <ul> <li>Enables you to restore from a DB system at a point in time.</li> <li>Select backup window</li> <li>The backup window start time defines the start of the time period during which y up.</li> <li>Window start time</li> </ul> | our DB system is backe |

#### High Availability RTO and RPO

- Single click High Availability
- Automatic Failover
- Planned Switchover
- Increase Uptime
- Reduce Downtime during a failure event (RTO: Minutes)
- Zero Data Loss during a failure event (RPO: Zero)

| Create MySQL DB System             |  |  |  |
|------------------------------------|--|--|--|
| Standalone                         | High Availability  |  |  |
| Single-instance MySQL DB<br>System | Run 3-node MySQL DB System<br>providing automatic failover and<br>zero data loss |  |  |
|                                    |  |  |  |



# **Inbound and Outbound Replication**

Hybrid Deployments and Migrations

Hybrid deployments

- On-premise and multi-cloud
- OCI as your main site
- OCI as your Disaster Recovery site
- OCI for capacity bursting
- HeatWave for Analytics

#### Live Migrations

Minimize downtime

**Cross-region replication** 

• DB System to DB System



# **Managed Read Replicas**

Grow and Shrink Read Capacity Seamlessly

High performance by scaling your reads.

- A single click creates a Read Replica
  - Provision
  - Launch
  - Setup Replication
  - Monitor and Manage
- Read Replicas are associated with a DB System
  - RO endpoints in the DB System
  - Up to 18 max per DB System
  - Requires a shape of 4 OCPUs or larger
  - CLI, SDK and Terraform support

| (i) Create a  | a read replica for the DB system dbsystem   |
|---|---|
|   |   |
| Name  |   |
| mysqlreadrepli  | ca20230130171946  |
| Description Opti  | onal  |
| Write a descrip   | tion  |
|   |   |
| Se Hide advand  | bed options   |
|   |   |
|   |   |
| Deletion pla  | an Tags   |
|   | an Tags   |
| Deletion pla  | n Tags Tags Tags Tags Tags Tags Tags Tags   |
| Deletion pla  | an Tags Tected read replica and its associated DB system against delete operations. By default, read replicas and DB systems are not delete protected. If you want to delete either the read replica or its associated the option.  |
| Deletion pla<br>Delete pro<br>Protects the<br>system, des | an Tags  tected read replica and its associated DB system against delete operations. By default, read replicas and DB systems are not delete protected. If you want to delete either the read replica or its associated the option. |
| Deletion pla  | n Tags  |
| Deletion pla<br>Delete pro<br>Protects the<br>system, des | n Tags  |
| Deletion pla  | n Tags  |
| Deletion pla  | n Tags  |
| Deletion pla<br>Delete pro<br>Protects the<br>system, des | n Tags  |
| Deletion pla<br>Delete pro<br>Protects the<br>system, des | n Tags  |
| Deletion pla<br>Delete pro<br>Protects the<br>system, des | n Tags betected read replica and its associated DB system against delete operations. By default, read replicas and DB systems are not delete protected. If you want to delete either the read replica or its associated the option. |
| Deletion pla<br>Delete pro<br>Protects the<br>system, des | n Tags  |

# The MySQL Heatwave Service

What?

- The most up to date MySQL, delivered by the MySQL team
- OLTP and OLAP in one database
- The technology you know, feels right at home for a traditional MySQL user
- Straightforward and simple for a new adopters
- Integrated with other Oracle services
- Powerful and secure infrastructure The Oracle Cloud Infrastructure
- Observed, Automated and Operated by MySQL Experts

| Conclusion  |                   |  |
|---|-------------------|--|
|   |                   |  |
| 40 Copyright © 2024, Oracle and/or its affiliates | 02, February 2024 |  |

## Conclusion

- Monitor, Observe
  - MySQL Replication enhancements makes tuning, troubleshooting and root cause analysis easier
- Automate, Automate
  - MySQL Replication has built-in automation that makes it is a great foundation for advanced system architectures
- Control and Operate
  - Overriding automation may be necessary during a maintenance event.
  - Emergency stopping a procedure is sometimes required.

#### **Replication in MySQL 8.x continues to improve usability and operability.**

The MySQL Heatwave Service runs and operates the database for you, on the Oracle Cloud Infrastructure. Go and give it a try if you are not using it already!

