MySQL Replication: What's New In MySQL 5.7 and MySQL 8

Luís Soares
Software Development Director
MySQL Replication
Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes 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 upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Program Agenda
Program Agenda

1. Introduction
2. Use Cases
3. Enhancements in MySQL 8 (and 5.7)
4. Enhancements in lab.mysql.com
5. Roadmap
6. Conclusion
Introduction
Boom

Web Explodes
Today...

– Technology mesh.
– All things distributed.
– Large amounts of data to handle, transform, store.
– Offline periods are horribly expensive, simply unaffordable.
– Go green requires dynamic and adaptative behavior.
– Much more data to store – e.g. social media, “Look at all of my pictures!”; Monitoring – Keeping logs for N years! ; IoT – and much more.
– Moving, transforming and processing data quicker than anyone else means having an edge over competitors.
– It is a zoo. Distributed coordination and monitoring is key.
Database Replication

Replication
“The process of generating and reproducing multiple copies of data at one or more sites.”
MySQL Database Replication: Overview

```
INSERT ... Server B
binary log
relay log
Server A
binary log

App
INSERT ...

Comm. Framework

Receiver
Meta-data
Update

Applicer
Meta-data
Update
```

```
MySQL Database Replication: Overview

- **App**: INSERT ...
- **Server A**: INSERT ...
- **Server B**: INSERT ...
- **Binary Log**
  - **Persistent log buffer**
- **Relay Log**
  - **Persistent log buffer**
  - **Threaded applier**
  - **Receiver Meta-data Update**
  - **Applier Meta-data Update**
- **Communication Framework**
  - Capture statements or data changes.
  - Send, Receive, ACK, NACK, Heartbeating, ...

---

Tuesday, 24th April 2018, Santa Clara, CA, USA  
Percona Live 2018
MySQL Database Replication: Some Notes

Binary Log

• Logical replication log recording master changes (binary log).
• Row or statement based format (may be intermixed).
• Each transaction is split into groups of events.
• Control events: Rotate, Format Description, Gtid, and more.
MySQL Database Replication: Some Notes

Coordination Between Servers

Since 3.23
asynchronous (native)

Since 5.5
semi-synchronous (plugin)

Since 5.7.17
And in MySQL 8 as of 8.0.1
group replication (plugin)
Use cases
Clustering Made Practical

R eplicate
A utomate
I ntegrate
S cale
E nhance
Replicate

Group Replication

• For highly available infrastructures where:
  – the number of servers has to grow or shrink dynamically;
  – with as little pain as possible.
Automate

Group Replication

• Single-primary mode
  – Automatic PRIMARY/SECONDARY role assignment
  – Automatic new PRIMARY election on PRIMARY failures
  – Automatic setup of read/write modes on PRIMARY and SECONDARIES
  – Automatic global consistent view of which server is the PRIMARY
Integrate
Binary Log

- Logical replication log
  - Extract, transform and load.
  - MySQL fits nicely with other technologies.
Scale

Asynchronous Replication

- Replicate between clusters
  - For disaster recovery

- Read Replicas
  - For read-scale out. Deploy asynchronous read replicas connected to the cluster
Enhance

InnoDB Cluster

• **InnoDB Cluster Integrated Solution**
  – Group Replication for high availability.
  – Asynchronous Replication for Read Scale-out.
  – One-stop shell to deploy and manage the cluster.
  – Seamlessly and automatically route the workload to the proper database server in the cluster.
  – Hide failures from the application.
Enhancements in MySQL 8 (and 5.7)

3.1 Binary Log Metadata

3.2 Operations

3.3 Monitoring

3.4 Performance

3.5 Other
New Metadata in the Binary Log
Easy to extract, transform and load into other systems.

- **New Metadata**
  - Easy to decode what is in the binary log.
  - Further facilitates connecting MySQL to other systems using the binary log stream.
  - Capturing data changes through the binary log is simplified.
  - Also more stats showing where the data is/was at a certain point in time.
Enhancements in MySQL 8 (and 5.7)

3.1 Binary Log Metadata
3.2 Operations
3.3 Monitoring
3.4 Performance
3.5 Other
Multi-Source Replication Filters
Replicate, Filter, Aggregate, Query

users
bio
groups

only users

not users
everything

users
bio
groups
posts
comments

users
posts
comments

A
B
C
Preventing Updates On Replicas that Leave the Cluster
Automatic protection against involuntarily tainting of offline replicas

A joins a single primary cluster and it allows update operations prior to joining.

A becomes a secondary and is automatically set to read only.

A leaves the group and remains read only until the DBA reverts.

Backported to 5.7.20
Primary Election Weights

Choose next primary by assigning election weights to the candidates.

- **B** is the primary.
- **B** is not in the group anymore.
- **C** has higher weight than **A**.
- **C** is elected the new primary.
Online and Automatic Cluster Performance Management

New options to fine tune the cluster automatic flow control in 8.0.

- **B** is struggling to keep up.
- **A** and **C** throttle themselves automatically.
- **B** has a chance to catch up.
- **B** is not struggling any more. System adjusted itself.
Enhancements in MySQL 8 (and 5.7)

3.1 Binary Log Metadata
3.2 Operations
3.3 Monitoring
3.4 Performance
3.6 Other
Monitor Lag With Microsecond Precision
Through the entire asynchronous topology

How much time does my data take to reach D coming from A?
Monitor Lag With Microsecond Precision

From the immediate master

How much time does my data originated in A takes to flow from B to C?
Monitor Lag with Microsecond Precision
For each stage of the replication applier process

- Per Stage Timestamps
  - User can monitor how much time it takes for a specific transaction to traverse the pipeline.
Global Group Stats Available on Every Server
Version, Role and more

• Query one Replica, Get status of all
  – Every replica reports group-wide information about roles and versions of the members of the group.
  – Also available at any replica are group-wide status.
Enhancements in MySQL 8 (and 5.7)

3.1 Binary Log Metadata
3.2 Operations
3.3 Monitoring
3.4 Performance
3.5 Other
Highly Efficient Replication Applier
Write set parallelization

Applier Throughput: Sysbench Update Index

- COMMIT_ORDER
- WRITESET
- WRITESET_SESSION

Backported to 5.7.22
Highly Efficient Replication Applier

Write set parallelization

- WRITESET dependency tracking allows applying a single threaded workload in parallel.
  - Delivers the best throughput of the three dependency trackers, at any concurrency level.

- WRITESET_SESSION in addition to writesets tracks sessions dependencies as well. Two transactions executed on the same session are always scheduled in execution order on replica servers.

- Fast Group Replication recovery – time to catch up.
Fast Group Replication Recovery
Replica quickly online by using WRITESET

Group Replication Recovery Time: Sysbench RW (durable settings)

- Sysbench RW at 33% capacity (workload: 4K TPS on 64 threads)
- Sysbench RW at 66% capacity (workload: 8K TPS on 64 threads)

Group Replication Recovery Time: Sysbench Update Index (durable settings)

- Sysbench RW at 33% capacity (workload: 9K TPS on 64 threads)
- Sysbench RW at 66% capacity (workload: 18K TPS on 64 threads)

Backported to 5.7.22
High Cluster Throughput

More transactions per second while sustaining zero lag on any replica

Asynchronous Replication Sustained Throughput
(Sysbench Update Index, durable settings)

- MySQL 5.7
- MySQL 8.0.3

Asynchronous Replication Sustained Throughput
(Sysbench Update Index, non-durable settings)

- MySQL 5.7
- MySQL 8.0.3
High Cluster Throughput

More transactions per second while sustaining zero lag on any replica

– At lower thread count, the throughput of the system doubles in MySQL 8.0 compared to MySQL 5.7 on durable settings.

– At lower thread count, the throughput of the system more than doubles in MySQL 8.0 compared to MySQL 5.7 on non-durable settings.
Efficient Replication of JSON Documents

Replicate only changed fields of documents (Partial JSON Updates)

- Numbers are from a specially designed benchmark:
  - tables have 10 JSON fields,
  - each transaction modifies around 10% of the data
Efficient Replication of JSON Documents

Replicate only fields of the document that changed (Partial JSON Updates)

Throughput on the Master:
Partial JSON vs Complete JSON

Throughput on the Slave:
Partial JSON vs Complete JSON
Enhancements in MySQL 8 (and 5.7)

3.1 Binary Log Metadata
3.2 Operations
3.3 Monitoring
3.4 Performance
3.5 Other
Changes to defaults in MySQL 8

High performance replication enabled out-of-the-box

• Binary log is on by default.
• Logging of slave updates is on by default.
• Replication metadata is stored in InnoDB tables by default instead of files.
• Row-based applier uses hash scans to find rows instead of table scans.
• Transaction write-set extraction is on by default.
• Binary log expiration is set to 30 days by default.
• Server-id is set to 1 by default instead of 0.
Other MySQL 8 Replication Enhancements

• **Monitoring:** Monitor replication even when disk full
• **Monitoring:** Current query being applied, even for row-based replication
• **Monitoring:** Replication filters statistics in performance schema
• **Monitoring:** Group Replication threads instrumented and shown in performance schema
• **Monitoring:** Group Replication conditional variables and mutexes instrumented and shown in performance schema
• **Recoverability:** Recover DDL and binary log together after a crash
Other MySQL 8 Replication Enhancements

• **Operations:** Restore global transaction identifiers metadata on a non-empty server

• **Operations:** Specify binary log file number after RESET MASTER

• **Operations:** Specify when binary log files are automatically purged (with second precision)

• **Operations:** SAVEPOINT support when write sets are being extracted

• **Operations:** P_S table for consistent log positions (replacing potentially expensive FLUSH TABLE WITH READ LOCKS)

• **Operations:** Support hostnames in Group Replication whitelist
Other MySQL 8 Replication Enhancements

- **Troubleshooting:** Dynamic and high performance debugging of group replication inter-node messaging
Enhancements in Labs

4.1 Operations

4.2 Monitoring
Enhancements in Labs

4.1 Operations

4.2 Monitoring
Automatic Abort Replicas that Leave the Group

Automatically Shutdown When Replica Leaves the Group Involuntarily

- **A** is a secondary and **was automatically set to read only**.

- **A** leaves the group involuntarily (failed to apply changes, network partitioned, etc).

- **A** shuts itself down, thus becoming effectively unavailable for both reads and write operations.

```
@@group_replication_exit_state_action={ READ_ONLY | ABORT_SERVER }
```
Enhancements in Labs

4.1 Operations

4.2 Monitoring
Group Replication Message Cache Memory Usage

- GCS/XCom’s Paxos message cache is instrumented.
- GCS/XCom’s Paxos message cache memory usage is exposed in the performance schema.

```
-- This is a session open on ServerA and the user is reading stats on GCS_Xcom message cache.
ServerA> select * from memory_summary_global_by_event_name where event_name like "%GCS_XCom%"

*************************** 1. row ***************************
EVENT_NAME: memory/group_rpl/GCS_XCom::xcom_cache
COUNT_ALLOC: 28890317
COUNT_FREE: 28840318
SUM_NUMBER_OF_BYTES_ALLOC: 24499151783
SUM_NUMBER_OF_BYTES_FREE: 24470424555
LOW_COUNT_USED: 0
CURRENT_COUNT_USED: 49999
HIGH_COUNT_USED: 50000
LOW_NUMBER_OF_BYTES_USED: 0
CURRENT_NUMBER_OF_BYTES_USED: 28727228
HIGH_NUMBER_OF_BYTES_USED: 135676530
1 row in set (0.01 sec)
```
Roadmap
The Road to MySQL 8 Group Replication and InnoDB Clusters

MySQL 5.6.10
5.6 is GA

GR 0.2.0 labs
Hello world!

MySQL 5.7.9
5.7 is GA
lifecycle interfaces
_P_S tables for GR
server side changes

MySQL 5.7.17
GR is GA

MySQL 8.0.0 DMR
MySQL 8.0.0.2 DMR
More replication enhancements

MySQL 8.0.1 DMR
MySQL 8.0.1.2 DMR
MySQL 8.0.2 DMR
MySQL 8.0.2.3 RC1
bug fixes
partial json updates
monitoring enhancements

MySQL 8.0.3 RC1
bug fixes
monitoring enhancements

MySQL 8.0.4 RC2
bug fixes
monitoring enhancements

MySQL 8.0.5 RC3
bug fixes
monitoring enhancements

MySQL 8.0.6 RC4
bug fixes
monitoring enhancements

MySQL 8.0.7 RC5
bug fixes
monitoring enhancements

MySQL 8.0.8 RC6
bug fixes
monitoring enhancements

MySQL 8.0.9 RC7
bug fixes
monitoring enhancements

MySQL 8.0.10 RC8
bug fixes
monitoring enhancements

MySQL 8.0.11 GA
bug fixes
MySQL InnoDB Cluster: The End Goal
Conclusion
Conclusion

MySQL 8.0.11 GA is out:

- **Performance/efficiency improvements**
  - Smaller recovery times, faster applier, reduced storage footprint.

- **Replication instrumentation**
  - Enhanced lag monitoring, more introspection into Group Replication stats and threads

- **Powerful Dev-Ops**
  - Enhanced multi-source replication filters, controlled primary election and fine tuning of flow control.
  - Improved crash-recovery, new defaults, more flexible GTID handling.
Where to go from here?

• Packages

• Documentation

• Blogs from the Engineers (news, technical information, and much more)
  – http://mysqlhighavailability.com