New features in MySQL Replication

Lars Thalmann, Development Manager, Replication & Backup
Mats Kindahl, Lead Developer, Replication & Plugins
MySQL User Conference 2010
Topics

• Replication features in **MySQL 5.1**
  *Generally Available release Nov 2008*

• New replication features in **MySQL 5.5.2**
  *Development release Feb 2010*

• New replication features in **MySQL 5.5.3**
  *Development release Apr 2010*

• Ongoing projects
Replication features in MySQL 5.1

1. Row-based replication
   - New event types, binlog formats
   - Automatic engine capability control of logging formats

2. MySQL Cluster replication
   - Incident events, binlog injection, replication state tables

3. Safe slave skipping
   - Skips complete transactions

4. Safe replication of unsafe (non-deterministic) statements
   - In MIXED format, use row-based replication for unsafe statements
   - In STATEMENT format, warn about unsafe statements

5. Replication of event definitions
   - Replicated as slave-side-disabled

Deprecated configuration options
Over 250 replication bugs closed at initial release time
New replication features in MySQL 5.5.2

1. **Semisynchronous replication**
   Improved resilience by having master wait for slave to persist events

2. **Slave fsync tuning**
   Fine tune fsyncs so that corruption is less likely on slave crashes

3. **Automatic relay log recovery**
   Let the slave to recover from corrupted relay logs

4. **Replication Heartbeat**
   Avoid spurious relay log rotation when the master is idle.
   Have a more precise failure detection mechanism

5. **Show relay logs**
   View relay log contents by issuing 'SHOW RELAYLOG EVENTS'.

6. **Per server replication filtering**
   Instruct slave to discard events from a master with a specific server id.
1. Semisynchronous Replication

Originally developed by Mark Callaghan and Wei Li, Google Modularized, tested, and bug fixed by Zhenxing He, MySQL

Available as two separate loadable components for the master and the slave
1. Semisynchronous Replication

- MySQL replication is asynchronous
  Transactions may be lost if master cannot recover from crash. When master fails, it may have transactions not replicated to slave

- Semi-synchronous replication ensure redundancy
  Each transaction is safely transmitted to slave before the commit returns

- Fully synchronous replication would be slower
  It would affect performance more significantly. Semi-sync is a compromise
1. Semisynchronous Replication

Google did:

- At least one slave receives the events before commit returns
- The master waits for the acknowledgement using a time-out
- Automatically disables semi-sync on time-out, and then enables semi-sync when slave catch up

MySQL did:

- Merged to MySQL 5.5, removed dependency on InnoDB
- Defined interfaces and implemented as plugins
- Testing and bug fixing
1. Semisynchronous Replication

- **On master**
  - INSTALL PLUGIN 'rpl_semi_sync_master' SONAME 'semisync_master.so';
  - SET rpl_semi_sync_master_enabled=1;
  - SET rpl_semi_sync_master_timeout=1000; (1s, default 10ms)

- **On slave**
  - INSTALL PLUGIN 'rpl_semi_sync_slave' SONAME 'semisync_slave.so';
  - SET rpl_semi_sync_slave_enabled=1;
  - START SLAVE;
1. Semisynchronous Replication

Checking the state

• On master
  – Rpl_semi_sync_master_status
  – Rpl_semi_sync_master_clients
  – Rpl_semi_sync_master_yes_tx
  – Rpl_semi_sync_master_no_tx

• On Slave
  – Rpl_semi_sync_slave_status
Three new variables: `sync_relay_log_info`, `sync_master_info`, `sync_relay_log`

- **`sync_relay_log_info > 0`**
  Synchronize relay-log.info file to disk after that many transactions

- **`sync_relay_log_info = 1`**
  Generally the best choice

- **`sync_relay_log_info = 0`**
  Default. Does not force any synchronization to disk. Server relies on operating system to flush the relay-log.info file
2. Slave fsync tuning

- **sync_master_info = 0**
  Default. Recommended in most situations.

- **sync_master_info > 0**
  Slave synchronize master info after that many transactions.

- **sync_relay_log = 0**
  Default. OS responsible for syncing relay log to disk

- **sync_relay_log = 1**
  The *safest choice*. In the event of a crash you lose at most one statement or transaction from the relay log. Also *slowest choice*. (Unless the disk has a battery-backed cache, which makes synchronization very fast).
3. Automatic Relay Log Recovery

\[ \text{relay\_log\_recovery} = 1 \]

On restart, slave discards all unprocessed relay logs (and retrieves them from master).

This can be used after a slave crash to ensure that potentially corrupted relay logs are not processed.

The default value is 0 (disabled).
4. Replication Heartbeat

- Automatic checking of connection status
- No more relay log rotates when the master is idle
- Detection of master/slave disconnect configurable in millisecs

CHANGE MASTER SET master_heartbeat_period= val;
SHOW STATUS like 'slave_heartbeat_period'
SHOW STATUS like 'slave_received_heartbeats'
5. SHOW RELAYLOG EVENTS

master> use test;
master> CREATE TABLE t1 (a int);
master> INSERT INTO t1 VALUES (1), (2), (3);

slave> START SLAVE;
slave> SHOW RELAYLOG EVENTS IN 'slave-relay-bin.000002';

<table>
<thead>
<tr>
<th>Log_name</th>
<th>Pos</th>
<th>Event_type</th>
<th>Server_id</th>
<th>End_log_pos</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>slave-relay-bin.000002</td>
<td>4</td>
<td>Format_desc</td>
<td>2</td>
<td>107</td>
<td>Server ver: 5.1.40-debug-log, Binlog ver: 4</td>
</tr>
<tr>
<td>slave-relay-bin.000002</td>
<td>107</td>
<td>Rotate</td>
<td>1</td>
<td>0</td>
<td>master-bin.000001;pos=4</td>
</tr>
<tr>
<td>slave-relay-bin.000002</td>
<td>151</td>
<td>Format_desc</td>
<td>1</td>
<td>107</td>
<td>Server ver: 5.1.40-debug-log, Binlog ver: 4</td>
</tr>
<tr>
<td>slave-relay-bin.000002</td>
<td>254</td>
<td>Query</td>
<td>1</td>
<td>193</td>
<td>use <code>test</code>; CREATE TABLE t1 (a int)</td>
</tr>
<tr>
<td>slave-relay-bin.000002</td>
<td>340</td>
<td>Query</td>
<td>1</td>
<td>291</td>
<td>use <code>test</code>; INSERT INTO t1 VALUES (1), (2), (3)</td>
</tr>
</tbody>
</table>
6. Per server replication filtering

Circular replication

The originating server acts as the terminator of its own events:

When an event from A reaches A again, it is removed.
If server A is removed from the circle, then there can be events from A circulating indefinitely.
6. Per server replication filtering

Circular replication

If server A is removed from the circle, server B can be set to terminate A's events in the new circle

Server B> CHANGE MASTER TO MASTER_HOST=C ... IGNORE_SERVER_IDS=(A)
New replication features in MySQL 5.5.3

7. Precise Slave Type Conversions
   Use different types on master and slave and get automatic type promotion and demotion when using RBR

8. Individual Log Flushing
   Selectively flush server logs when using 'FLUSH LOGS'

9. Safe logging of mixed transactions
   Replicate transactions containing both InnoDB and MyISAM changes
7. Precise Slave Type Conversions

- MySQL 5.1 statement-based: Conversions supported
  - Statements executed on slave without checks
    
    master> CREATE TABLE foo (a INT);
    slave> CREATE TABLE foo (a TINYINT);
    master> INSERT INTO foo VALUES (1);
    slave> <<<success>>>

- MySQL 5.1 row-based: Conversions not supported
  - Column types must be the same
    
    master> CREATE TABLE foo (a INT);
    slave> CREATE TABLE foo (a TINYINT);
    master> INSERT INTO foo VALUES (1);
    slave> <<<error>>>
7. Precise Slave Type Conversions

New variable SLAVE_TYPE_CONVERSIONS

Variable is a “set”. Slave must be restarted for variable to have effect
Default value is empty set (types must be identical on master and slave)

**ALL_NON_LOSSY** – enable conversions to types with larger domain

**ALL_LOSSY** – enable conversions to types with smaller domain

e.g. INT ==> TINY, but *not* TINY ==> INT. Lossy conversions are implemented by either truncation or rounding depending on the type

Conversions within Integer, Decimal, String, Binary, BIT, ENUM, and SET domains are supported.
Conversions are not done between domains
7. Precise Slave Type Conversions

- The column types on master and slave determine the rules for conversion. The values are not considered.
- Example, SLAVE_TYPE_CONVERSIONS = 'ALL_LOSSY'):
  master> CREATE TABLE foo (a INT);
  slave> CREATE TABLE foo (a TINYINT);
  master> INSERT INTO foo VALUES (1);
  slave> <<<success>>> 
- Example, SLAVE_TYPE_CONVERSIONS = "":
  master> CREATE TABLE foo (a INT);
  slave> CREATE TABLE foo (a TINYINT);
  master> INSERT INTO foo VALUES (1);
  slave> <<<error>>>
8. Individual log flushing

Flush of individual logs:

```
FLUSH <log_type> LOGS;
```

Examples:

```
FLUSH ERROR LOGS, RELAY LOGS;
FLUSH BINARY LOGS, ENGINE LOGS, SLOW LOGS;
```

Log types supported:

- **SLOW** - close & reopen the slow query log file.
- **ERROR** - close & reopen the error log file.
- **BINARY** - close & reopen the binary log files.
- **ENGINE** - close & reopen any flushable logs for installed storage engines
- **GENERAL** - close & reopen the general query log file
- **RELAY** - close & reopen the relay log files
MySQL 5.1 Replication Architecture

Mixing InnoDB and MyISAM changes in the same transaction can cause slave to be inconsistent with the master

- MyISAM changes are visible immediately when statement ends
- Transactional cache is flushed at commit time
9. Safe logging of mixed transactions

MySQL 5.5 Replication Architecture

Mixing InnoDB and MyISAM changes in the same transaction is correctly serialized in the binary log for MIXED and ROW formats:

- MyISAM changes written to the binary log as soon as the statement ends
- Each thread has 2 caches
9. Safe logging of mixed transactions

- For MIXED and ROW formats, mixing MyISAM and InnoDB is always correctly serialized in binary log.
- For STATEMENT format, an option “binlog-direct-non-transactional-updates” can be used to ensure that MyISAM statements are always immediately written to the binary log. It is not generally recommended to use this option.

- Other consequences of this work
  - Improved performance by reducing number of binlog mutexes
  - Binlog updates always wrapped by BEGIN-COMMIT/ROLLBACK
  - Closed 13 bugs
Future

1. **Server UUIDs**
   Unique server ids making it easier to analyze replication topologies

2. **Remote backup of binary logs using mysqlbinlog tool**
   Retrieve the binary log from master

3. **Options for writing full or partial row images in RBR**
   Optimize for performance, disk size, or functionality

4. **Modular replication architecture**
   Use different replication tools to replicate to/from a MySQL server

5. **Replication checksums**
   Detect transmission or disk corruptions
Future

6. **Transactional replication information**
   Automatically recover from a slave crash

7. **Informational events**
   Original statement for RBR, User and IP of statement executor, engine-dependent information

8. **Scriptable replication**
   Write your own plugin (e.g. replication filtering on data or statement type, extraction of data, pre-heating of caches)

9. **Information schema for replication state**
   Query the state instead of using show commands

10. **Preallocated binlog files**
    Improve performance by not having to append to files
Future

11. DTrace replication probes
   Internal performance monitoring and debugging

12. Multi-threaded slave for higher performance
   Reduce possibility for slave to lag behind master

13. High resolution replication delay measurement
   IO and SQL delay separately measured in milliseconds

14. Improved built-in filtering of MIXED mode
   Always filter on actually modified database
BUG#46364 MyISAM transbuffer problems (NTM problem)

BUG#49119 Master crashes when executing 'REVOKE ... ON {PROCEDURE|FUNCTION} FROM

BUG#51021 current_stmt_binlog_row_based not removed in next-mr

BUG#49259 Slave I/O thread could not register on master

BUG#40611 MySQL cannot make a binary log after sequential number beyond unsigned

BUG#49557 "semisync" plugin test fails on Windows

BUG#45785 LIMIT in function does not cause RBL if binlog_format=MIXED

BUG#50192 Strange effect in replication test, trigger, auto_increment

BUG#37148 Most callers of mysql_bin_log.write ignore the return result

BUG#51194 relay_log_space being reported as gigantic

BUG#51663 Unable to connect to remote mysql server for replication using ssl after

BUG#20413 sql_slave_skip_counter is not shown in show variables

BUG#37187 `INFORMATION_SCHEMA`.`GLOBAL_VARIABLES`: inconsistent status

BUG#49873 Replication thread exiting (Error 1364) - field doesn't have default v

BUG#51621 DROP VIEW on a nonexistent view on a master causes replication to brea

BUG#47418 RBR fails, failure with mixup of base/temporary/view TABLE DDL

BUG#48091 valgrind errors when slave has double not null and master has double n

BUG#42757 Redundant use of LOCK_log in MYSQL_BIN_LOG::write(Log_event*)

BUG#50577 ERROR 1594 : Relay log read failure: Could not parse relay log event e

BUG#50862 mysql_upgrade breaks replication

BUG#46827 rpl_circular_for_4_hosts failed on PB-2

BUG#35319 Almost certain crash(es) of mysqld with the "rpl*" tests from mysql-te

BUG#43579 mysql_upgrade tries to alter log tables on replicated database

BUG#50061 Test "rpl_loaddata_symlink" uses a Unix-specific "run-slave" script

BUG#50474 rpl_slave_load_remove_tmpfile failed on windows debug enabled binary

BUG#50060 "semi_sync" plugin crashes the server on Ubuntu

BUG#49740 rpl.rpl_temporary fails in PB2 in mysql-trunk-merge

BUG#49012 rpl.rpl_locktrans_myisam fails on pushbuild, on sles10-ia64-a

BUG#48135 Test rpl.rpl_get_master_version_and_clock 'row' failed on Windows

BUG#34600 Rolled-back punch transactions not replicated correctly

BUG#42928 binlog_format setting prevents server from start if binary logging is

BUG#45673 Semisynch reports correct operation even if no slave is connected

BUG#37221 SET AUTOCOMMIT=1 does not commit binary log

BUG#45852 Semisynch: Last_IO_Error: Fatal error: Failed to run 'after_queue_even

BUG#12190 CHANGE MASTER has differ path requiremts on MASTER_LOG_FILE and RELAY_

BUG#42244 Segmentation fault at rpl_handler.h

BUG#45973 rpl_semi_sync fails in valgrind platform with warnings

BUG#40278 Replication failure on RBR + MyISAM + SAVEPOINTs

BUG#47016 rpl_do_grant fails on PB-2 with a failing connect

BUG#47287 RBR: replication diff on basic case with txn- and non-txn tables in a

BUG#49747 1511: 3:24:10 Replication thread exiting (Error 1364)

BUG#39012 rpl.rpl_locktrans_myisam fails on pushbuild, 5.1.9-

BUG#46572 DROP VIEW is not working correctly on Solaris

BUG#43784 assert or broken replication with geometrycollection and decimal colum

BUG#47678 Changes to n-tables that happen early in a trans, are only flushed upo

BUG#43784 assert or broken replication with geometrycollection and decimal colum

BUG#47678 Changes to n-tables that happen early in a trans, are only flushed upo

BUG#49186 Concurrent insert/select causes slave abort if select is under read co

BUG#49388 Concurrent insert/select causes slave abort if select is under read co

BUG#45846 rpldb00012 failed on PB2

BUG#42208 rollback failures in master log while running on slave

BUG#43578 rpl truncate Myisam timeouts sporadically on pushbuild

BUG#41226 mysqlbinlog -d errors when binary is not written

BUG#48573 swithing to RBR in mixed mode replication for queries with SYSDATE()

BUG#47678 Changes to n-tables that happen early in a trans, are only flushed upo

BUG#44770 RBR: replication diff on basic case with ten- and non-tion tables in a

BUG#49016 Replication thread exiting (Error 1364)

BUG#49016 Replication thread exiting (Error 1364)

BUG#46572 DROP TEMPORARY table IF EXISTS does not have a consistent behavior in

BUG#48509 Bug in master Master Replication

BUG#45989 Slave cannot replay relay logs again

BUG#48509 Bug in master Master Replication

BUG#42947 impossible to do point-in-time recovery from older binary

BUG#47323 mysqlbinlog --verbose displays bad output when events contain subset o

BUG#49124 No LAST_ERR for max allowed packet errors

BUG#44778 assert or broken replication with geometrycollection and decimal colum

BUG#47678 Changes to n-tables that happen early in a trans, are only flushed upo

BUG#49186 Concurrent insert/select causes slave abort if select is under read co

BUG#49388 Concurrent insert/select causes slave abort if select is under read co

BUG#45846 rpldb00012 failed on PB2

BUG#42208 rollback failures in master log while running on slave

BUG#43578 rpl truncate Myisam timeouts sporadically on pushbuild

BUG#41226 mysqlbinlog -d errors when binary is not written

BUG#48573 swithing to RBR in mixed mode replication for queries with SYSDATE()

BUG#47678 Changes to n-tables that happen early in a trans, are only flushed upo

BUG#44770 RBR: replication diff on basic case with ten- and non-tion tables in a

BUG#49016 Replication thread exiting (Error 1364)

BUG#49016 Replication thread exiting (Error 1364)

BUG#46572 DROP TEMPORARY table IF EXISTS does not have a consistent behavior in

BUG#48509 Bug in master Master Replication

BUG#45989 Slave cannot replay relay logs again

BUG#48509 Bug in master Master Replication

BUG#42947 impossible to do point-in-time recovery from older binary

BUG#47323 mysqlbinlog --verbose displays bad output when events contain subset o

BUG#49124 No LAST_ERR for max allowed packet errors

BUG#44778 assert or broken replication with geometrycollection and decimal colum

BUG#47678 Changes to n-tables that happen early in a trans, are only flushed upo

BUG#44770 RBR: replication diff on basic case with ten- and non-tion tables in a

BUG#49016 Replication thread exiting (Error 1364)

BUG#49016 Replication thread exiting (Error 1364)
BUG#38994 Diagnostics_area::set_error_status Assertion `! is_set() || can_overwrite`

BUG#46191 InnoDB Crashed when exec query from relay-bin.

BUG#42941 `--database` paramater to mysqlbinlog fails with RBR

BUG#46215 binlog of insert, update statements are displayed as a string can not

BUG#45774 Selecting the data, ordered by the field, containing the same value in

BUG#46862 Auto-closing of temporary tables broken by replicate-rewrite-db

BUG#45150 LOST_EVENTS not documented

BUG#45575 slave status reporting incorrect master_exec_pos

BUG#38074 rpl_row_mysqlbinlog at line 235: could not sync with master

BUG#46209 binlog_tmp_table fails sporadically: missing rows in result

BUG#47566 rpl.rpl_semi_sync fails sporadically

BUG#45694 Deadlock in replicated statement is not retried

BUG#28976 Mixing trans and non-trans tables in one transaction results in incorr

BUG#40930 rpl.rpl_extraCol_falcon fails doing STOP SLAVE

BUG#46149 rpl_deadlock fails sporadically in pb2

BUG#45238 rpl_slave_skip, rpl_change_master failed (lost connection) for STOP SL

BUG#46030 rpl_truncate_3innodb causes server crash on windows

BUG#46014 rpl_stm_reset_slave crashes the server sporadically in pb2

BUG#42311 load data infile replication 4.1 to 6.0 fails

BUG#37223 PURGE MASTER LOGS command hangs in cluster replication

BUG#47160 rpl_binlog_max_cache_size fails sporadically on PB2

BUG#21869 Rpl_recovery_rank: Remove variable without function

BUG#48277 binlog reports error on inserts

BUG#48154 Replication Insert into select from

BUG#47425 replication breaks with Illegal mix of collations

BUG#47854 Replication not working when using database.table in statements

BUG#47103 RBR slave crash in table_def::type when modifying a merge table

BUG#44981 Vagrind warning when CHANGE MASTER executed

BUG#34654 RESET SLAVE and START SLAVE does not clear Last_IO_Errno

BUG#44312 deadlock between IO thread and SLAVE START

BUG#38716 slave crashed after 'stop slave' during concurrent stop/start/reset sl

BUG#44179 reset slave crashes in my_error when reset_logs returns non-zero

BUG#38205 Row-based Replication (RBR) causes inconsistencies: HA_ERR_FOUND_DUPP

BUG#0319 if while a non-transactional slave is replicating a transaction, possi

BUG#44415 Server crashes when start slave is issued

BUG#44287 mysqlbinlog output garbage if incident-events have a unknown id

BUG#40672 'Writing one row to the row-based binary log failed' and max_binlog_ca

BUG#43794 Slave SQL thread aborts with RBR on non-transactional tables

BUG#40654 'rpl_extraColmaster_myisam, rpl_extraColmaster_innodb'

BUG#42415 UPDATE with LIMIT clause unsafe for statement format even when ORDER B

BUG#45014 log-slow-slave-statements does not work in 5.1.34

BUG#33238 mysqlbinlog cannot read binlog with >2000 LOAD DATA INFILE, and is slo

BUG#45784 INSERT DELAYED in SP does not cause RBL if binlog_format=MIXED

BUG#45824 Unsafe stmt in proc doesn't generate warning if binlog_format=STATEMENT

BUG#45673 select slave statements when non-transactional table gets lock wa

BUG#444270 RESET SLAVE does not reset Last_IO_Error or Last_IO_Error

BUG#44426 Incident events are silent in mysqlbinlog output

BUG#40934 Semisynchronous replication: Change initial state after installation B

BUG#41097 result files for rpl_extraColmaster_myisam, rpl_extraColmaster_innodb

BUG#42199 rpl_ndb_mixed_tables test fails in 6.0-bugteam

BUG#45511 rpl.rpl_binlog_corruption fails with warning messages in Valgrind

BUG#44581 Slave stops when transaction with non-transactional table gets lock wa

BUG#43929 binlog corruption when max_binlog_cache_size is exceeded

BUG#45492 lpsl_up failures sporadically in check testcases

BUG#45511 rpl.rpl_binlog corruption fails with warning messages in Valgrind

BUG#42199 rpl.rpl_extraColMyisam test fails in 6.0-bugbase

BUG#42749 infinite loop writing to row based binlog - processlist show "freeing

BUG#43954 log-slow-slave-statements does not work in in 5.1.34

BUG#44442 Incident events are silent in mysqlbinlog output

BUG#44307 Test "rpl000082" fails in 'start/stop slave' on NEW5 and IAM4 (sic)

BUG#45783 sorry, bad bug report

BUG#42861 Assigning invalid directories to --slave-load-tmpdir crashes the slave

BUG#43929 Maria replication failure in row-based/mixed-mode replication

BUG#44389 rpl_row_mysqlbinlog fails on windows due to operating system sp

BUG#43076 rpl.rpl_idempotency fails sporadically on pushbuild

BUG#42861 if while a non-transactional slave is replicating a transaction, possi
Yes... it is more robust than ever before

Fixed

• 397 replication bugs closed since last user conference
  • 77 with attached customer support issues
• 75-80% of our development time is spent on bug fixing
  • Aiming for the highest quality of the core code base
  • For the benefit of community and customers

• 143 open non-feature replication bugs
  • 24 with attached customer support issues
Related Conference Sessions and References

• Mon 8:30am-12am Replication Tutorial
• Tue 2pm-3pm New Replication Features
• Wed 10:50-11:50am Mysteries of the Binary Log
• Thu 2:00-2:45pm Intro to using MySQL in Clouds
• Thu 2:50-3:35pm Replication Tricks & Tips

• Website: dev.mysql.com/replication

Dr. Lars Thalmann
Development Manager, Replication & Backup Technology
lars@mysql.com

Dr. Mats Kindahl
Lead Developer, Replication Technology
mats@mysql.com
The presentation 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.