

What's New With MySQL and MariaDB



<http://bit.ly/newMySQL2014>

Sheeri Cabral

Senior DB Admin/Architect

Mozilla

@sheeri

What's coming



- In MySQL 5.7
- What's here
 - In MariaDB 10

MySQL 5.7



- Option prefixing
 - --group vs. --group-concat-max-len
 - No prefixing
 - With new options, new ambiguity

Cleanup: deprecation



- Deprecated variables removed
 - innodb_use_sys_malloc (5.6)
 - innodb_additional_mem_pool_size (5.6)
- Metadata locking changed
 - metadata_locks_cache size
 - metadata_locks_hash_instances

Cleanup: deprecation



- log_warnings
 - log_error_verbosity
- **EXPLAIN EXTENDED**
 - Recognized, but always enabled anyway
 - Same with **EXPLAIN PARTITIONS**

Cleanup: unused



- `timed_mutexes`
 - does nothing
- `innodb_mirrored_log_groups`
 - only supported value is 1
- `mysql_upgrade` unused options
 - `--basedir`
 - `--datadir`

Cleanup: unused



- SQL modes:
 - ERROR_FOR_DIVISION_BY_ZERO (5.6)
 - NO_ZERO_DATE, NO_ZERO_IN_DATE
 - Part of STRICT_ALL_TABLES,
STRICT_TRANS_TABLES

Cleanup



- ALTER TABLE ... IGNORE
 - To not error on duplicate key issues
- storage_engine
 - use default_storage_engine
- --thread-concurrency
 - Only used on Solaris 8

Cleanup



- DELAYED deprecated (5.6)
 - INSERT/REPLACE DELAYED new warning:
“INSERT DELAYED is no longer supported. The statement was converted to INSERT.”
 - mysqlbinlog no longer writes DELAYED
 - Performance Schema cols removed
 - mysqldump –delayed-insert option

Simplification



- InnoDB table monitor
 - CREATE TABLE mysql.innodb_monitor (a INT) ENGINE=InnoDB;
 - SET GLOBAL innodb_status_output=ON
- InnoDB lock monitor
 - CREATE TABLE mysql.innodb_lock_monitor (a INT) ENGINE=InnoDB;
 - SET GLOBAL innodb_status_output_locks=ON

Simplification



- **SHOW ENGINE INNODB MUTEX**
 - Removed
 - view on Performance Schema instead
- **INFORMATION_SCHEMA.PROFILING**
 - use Performance Schema
- **Table and Tablespace monitor**
 - Changing to INFORMATION_SCHEMA tables

Clarification



- `ENCODE()`, `DECODE()`
 - `AES_ENCRYPT()`, `AES_DECRYPT()`

Long Overdue



- Warnings for using old_passwords
- Auto password expiration
- RPM deployments:
 - No anonymous users
 - No test database
 - Only root@localhost
 - random password
 - \$HOME/.mysql_secret

New Stuff



- Test suite now uses InnoDB as default
- gb18030 character set (China Nat'l Std)
- Ctrl-C cancels current query
 - But does not exit if no query

```
mysql> Ctrl-C -- exit!
```

```
Aborted
```

InnoDB FULLTEXT Search



- Introduced in 5.6
- Parser plugin support in 5.7

InnoDB Spatial Data



- In 5.6, BLOB
- In 5.7 BLOB internal
 - DATA_GEOMETRY data type

InnoDB Temporary Tables



- Separate tablespace for non-compressed
 - `innodb_temp_data_file_path`

InnoDB Temporary Tables



- Metadata no longer in InnoDB system tables
- Reports, metadata on all InnoDB temp tables
 - User-created
 - System
- New table, `INNODB_TEMP_TABLE_INFO`
 - Active tables
- Does not exist until you `SELECT` from it

Performance Enhancements



- Flushing dirty buffer pool pages
- `innodb_page_cleaners`
- Default of 1 – 5.6 behavior
- Pre 5.6 no separate thread for flushing
- **CREATE/DROP/TRUNCATE/ALTER**
performance enhancements

Binlog Dump Thread Locking



- Prior: lock held when reading an event
- Now: Lock only held when reading end of last binlog event
- simultaneous binlog reads and writes
- Multiple binlog reads

mysqlbinlog --rewrite-db



- `--rewrite-db='old->new'`
- Can use multiple times
- `--rewrite-db='old->new' --rewrite-db='old2->new2'`
- Row-based binlog
- With `--database`, rewrite done first
- `--rewrite-db='old->new' --database=new`

Partitioned Tables



- Index Condition Pushdown works
 - Introduced in 5.6 for other tables
- Transportable tablespaces
 - Can copy partitions between instances
 - Easier backups
- Can use HANDLER – low level
 - Directly manipulate tables
 - OPEN/CLOSE/READ

Logs



- mysql CLI --syslog option
 - Introduced in 5.1 for mysqld_safe
- New type of undo logs
 - For temporary tables
 - Regular
 - compressed



Multiple TRIGGERs!

New stuff



- **EXPLAIN FOR CONNECTION id**
 - Introduced in 5.1 for `mysqld_safe`
 - **INSERT, UPDATE, DELETE, SELECT, REPLACE**
 - Like MariaDB's **SHOW EXPLAIN**

Online ALTER TABLE



- RENAME INDEX (all engines)
- InnoDB
- OPTIMIZE TABLE
- No-op table rebuild (ENGINE=InnoDB)
- Increasing VARCHAR size
 - ALTER TABLE t1 ALGORITHM=INPLACE,
CHANGE COLUMN c1 c1 VARCHAR(255)
 - But not decreasing

Fusion-io NVM filesystem



- Non-volatile memory filesystem on Linux
- Provides atomic write capability
- InnoDB doublewrite buffer redundant
- Buffer auto-disabled on NVM

CHANGE MASTER Without STOP SLAVE



- Pre 5.7: ERROR 1198 (HY000): This operation cannot be performed with a running slave; run STOP SLAVE first
- If SQL and IO threads are stopped
 - MASTER_AUTO_POSITION
- If SQL thread is stopped
 - RELAY_LOG_FILE, RELAY_LOG_POS, MASTER_DELAY
- If I/O thread is stopped

MySQL and MariaDB



- MariaDB 5.5 fork of MySQL 5.5
- MariaDB 10 continues the fork
 - Adds in some MySQL 5.6 features

MariaDB 10 features



- GTIDs
 - Crash-safe slave
 - GTID position stored in InnoDB tables
- Multi-source replication
 - Code by a community member, not employee

Multi-source Replication



- CHANGE MASTER 'A' TO ...
- CHANGE MASTER 'B' TO...
- START ALL SLAVES
- SHOW ALL SLAVES STATUS
- STOP SLAVE 'B';
- SHOW SLAVE 'B' STATUS;
- SET
DEFAULT_MASTER_CONNECTION='B';

Parallel Slave



- MySQL's version is per-db
- MariaDB's is parallel even within tables
- Auto-detects independent transactions
- Preserves commit order
- Adapts to the master's load

Storage Engines



- TokuDB
- OQGraph v3
- CONNECT
- CassandraSE
- Sequence
- Spider

TokuDB



- Billions of rows, Tb of data
- 10-20x faster INSERTS
- Up to 90% compression
- No fragmentation
- Online ALTER TABLE
 - Even for column and index adds

CassandraSE



- Use MySQL to access Cassandra
- Use familiar SQL
- JOIN Cassandra and MySQL tables
 - Or CONNECT tables
- Flexible Schema
 - Uses dynamic columns



Dynamic Columns

- BLOB data type – ‘stuff’
- ...SET stuff=COLUMN_CREATE(‘key1’ , ‘value1’ , ‘key2’ , ‘value2’)...
- COLUMN_GET(stuff, ‘key1’ as CHAR)
- COLUMN_EXISTS(stuff, ‘key’)
- Can nest with user variables

CONNECT Storage Engine



- Federated and CSV functionality
- CSV, BIN, INI, XML, DBF, FMT
- MERGE functionality (TBL)
- ODBC, MySQL
- CATALOG
- Pivot/unpivot table
- Filesystem (DIR)
- Lists of objects

Sequence Storage Engine



- Read_only
- Virtual
 - Not even a .frm file!
- Supports indexes
- Supports XA transactions
- Exists only while you use it
- `SELECT * FROM seq_x_to_y`

Sequence Storage Engine



- Odd #'s from 5 backwards to 1
- Table seq_x_to_y_step_z
- e.g. `SELECT * FROM seq_5_to_1_step_2`
- Find all prime numbers < 50

Sequence Storage Engine



- Find all prime numbers < 50
- A prime # x means only x and 1 are divisors
- Every other y , $x \% y$ leaves a remainder
- Only need to check the square root and lower

Sequence Storage Engine



- Find all prime numbers < 50

```
SELECT seq FROM seq_2_to_50 s1
```

```
WHERE 0 NOT IN
```

```
(SELECT s1.seq % s2.seq FROM seq_2_to_50 s2
```

```
WHERE s2.seq < sqrt(s1.seq));
```

Sequence Storage Engine



- Can I name my table seq_1_to_100?
 - If a temp table, yes
 - Otherwise, not if sequence SE is installed
 - Can ALTER TABLE to change SE
 - Then can manipulate/drop

Spider Storage Engine



- Sharding/partitioning
- Transaction/XA support
- HA Cluster
- Supports HandlerSocket

HandlerSocket



- NoSQL plugin (since 5.3)
 - InnoDB
 - XtraDB
 - Spider
- Simple CRUD protocol, not SQL

Optimizer enhancements



<https://mariadb.com/kb/en/optimizer-feature-comparison-matrix/>

Optimizer enhancements



- Engine independent statistics
- `ANALYZE TABLE tbl PERSISTENT`
- Stored in mysql tables
 - `column_stats`
 - `index_stats`
 - `table_stats`
- Cardinality, avg frequency, etc.

Optimizer enhancements



- Extended InnoDB/XtraDB hidden primary key
- EXPLAIN “more detailed and consistent” than MySQL 5.6
- Block Hash Join algorithm
 - Usually faster than batch nested loop (BNL)
- Set memory limits on all join buffers
- Tests earlier for NULL values
- Apply outer table ON conditions earlier

Optimizer enhancements



- Set memory limits on all join buffers
- Tests earlier for NULL values
- Apply outer table ON conditions earlier



Multi-range read optimizations

- Fantastic graphics in knowledgebase

<https://mariadb.com/kb/en/multi-range-read-optimization>

- Records on disk not in index order
- MRR sorts the rows found in disk order
 - In a row buffer
 - Then it goes to disk
 - Less seeking = huge win for magnetic disks
- Slower for small data ranges, small LIMITs



Multi-range read optimizations

- Fantastic graphics in knowledgebase

<https://mariadb.com/kb/en/multi-range-read-optimization>

- Records on disk not in index order
- MRR sorts the rows found in disk order
 - In a row buffer
 - Then it goes to disk
 - Less seeking = huge win for magnetic disks



Multi-range read optimizations

- Slower for small data ranges
- Slower for small LIMITs
- If used, EXPLAIN Extra field has “Rowid-ordered scan”
- Have to turn it on
- `optimizer_switch='mrr=ON'`



Block Hash Join

- As opposed to batch nested loop (BNL)
- Join buffer in hash format
- Records from one table put in join buffer
- Compared to records from 2nd table
- Matches are faster than looping through everything
- Only for equi-joins

Subquery enhancements



- Subquery cache
- Fast EXPLAIN on subqueries
- Exists-to-in optimization
 - Does not handle
 - Aggregate functions
 - GROUP BY, HAVING

MariaDB vs. MySQL



- MariaDB seems to be innovating more
 - “power user” type features
- MySQL has lots of cleanup
- Both have great and desirable features
- Both share with each other

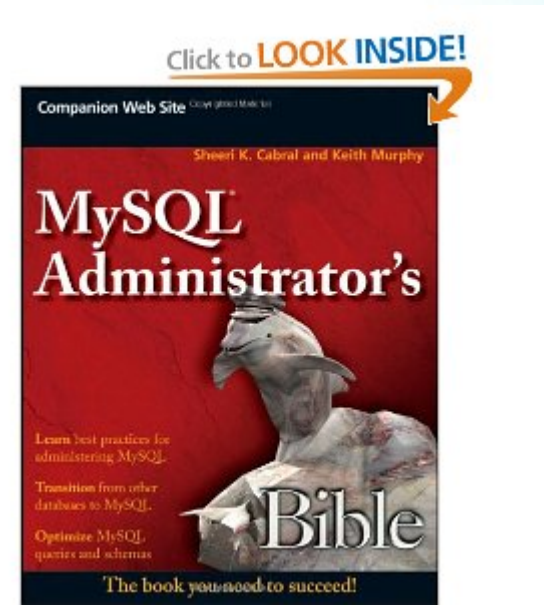
More Resources



www.mysqlmarinate.com

scabral@mozilla.com

OurSQL podcast (oursql.com)



slides: <http://bit.ly/newMySQL2014>

MySQL Administrator's Bible

youtube.com/tcation

<http://planet.mysql.com>