Storage engines troubleshooting The Introduction

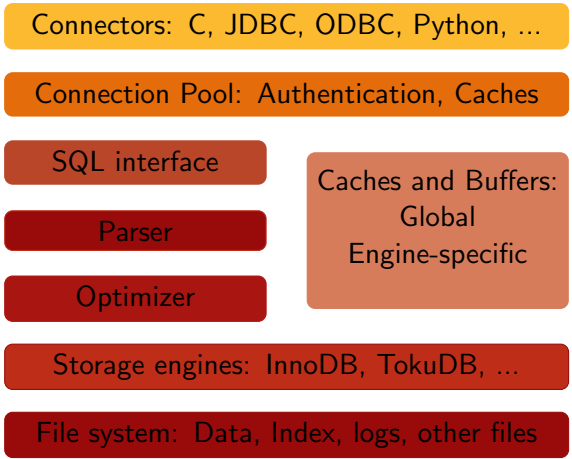
Sveta Smirnova
Principal Technical Services Engineer

July, 14, 2016

Table of Contents

Introduction

MySQL architecture



- Base
  - Installation layout
  - Log files
- Connectors
  - Clients, APIs
- Optimizer
- Cache and buffers
- Storage engines
- Management

Storage engines

- Own data
- Own index format
- Own locking model
- Own diagnostic
- Own log files
- CHECK TABLE

Data

Each engine stores data differently

- InnoDB
  - Writes to redo log first
  - Then flushes to tablespace files
  - We cannot just copy tablespace files after table content was modified: new data can be in redo logs only
- TokuDB
- MyISAM, CSV, ...
- Blackhole
- Federated

Each engine stores data differently

- InnoDB
- TokuDB
  - Writes to recovery log first
  - Each transaction has its own rollback log
  - Fractal Trees are stored in separate block files
    - Table data and indexes are stored in multiple files
- MyISAM, CSV, ...
- Blackhole
- Federated

Each engine stores data differently

- InnoDB
- TokuDB
- MyISAM, CSV, ...
  - Writes go to the data file directly
  - Easier to make binary backups
  - No rollbacks
- Blackhole
- Federated

Each engine stores data differently

- InnoDB
- TokuDB
- MyISAM, CSV, ...
- Blackhole
  - All writes ignored
- Federated

Each engine stores data differently

- InnoDB
- TokuDB
- MyISAM, CSV, ...
- Blackhole
- Federated
  - All reads and writes go to remote server
  - Affected by both connection and underlying storage engine

What does it mean for IO troubleshooting?

- When studying how IO works you need to study what the engine is doing

Does this mean data is independent?

Not!

Example: backup

- Binary backup works differently for different engines
  - XtraBackup/MEB 100 % non-blocking for InnoDB
    - Still has to lock tables to copy *.frm files and other engines
  - TokuDB Hot Backup simply copies files which do not use TokuDB storage engine
    - Can corrupt InnoDB backup
  - mysqlhotcopy backups MyISAM and other tables which use ”simple” engines
    - Not suitable for transactional engines
- Logical backup is not the panacea

Example: backup

- Binary backup works differently for different engines
- Logical backup is not the panacea
  - Different methods for data protection for ”simple” and transactional engines
    - mysqldump ... –lock-all-tables
    - mysqldump ... –single-transaction
  - Mind network and underlying storage engine for Federated!

Example: table statistics

- Used by Optimizer when it creates query plan
- Stored by engine
- Engine can decide
  - If
  - When
  - How
  - it will collect statistics

Correlation with other parts of the server

Locks

- Metadata locks used to protect table definition
- Table locks can be set
  - Implicitly
  - LOCK TABLES
- Engine-specific locks know nothing about server-level locks
- Non-recoverable ”deadlocks” can happen

Locks and Federated

```
mysql> create table t1(f1 int) engine = federated connection='mysql://root@127.0.0.1:13000/test/t1';
Query OK, 0 rows affected (0,05 sec)

source> begin;
Query OK, 0 rows affected (0,00 sec)

source> update t1 set f1=f1*2;
Query OK, 3 rows affected (0,02 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> update t1 set f1=f1*3;
-- waits
```

Locks and Federated

```
source> show processlist;
+----+------+-----+---------+------+----------+-----------------------------------------------+
| Id | User | ... | Command | Time | State    | Info                                          |
+----+------+-----+---------+------+----------+-----------------------------------------------+
|  3 | root | ... | Query   |   30 | updating | UPDATE `t1` SET `f1` = 3 WHERE `f1` = 1 LIMIT 1 |
|  4 | root | ... | Sleep   |   44 |          | NULL                                          |
|  5 | root | ... | Query   |    0 | starting | show processlist                              |
+----+------+-----+---------+------+----------+-----------------------------------------------+
3 rows in set (0,00 sec)

mysql-monitor> show processlist;
+----+------+-----+---------+------+----------+----------------------+
| Id | User | ... | Command | Time | State    | Info                 |
+----+------+-----+---------+------+----------+----------------------+
|  2 | root | ... | Query   |   12 | updating | update t1 set f1=f1*3 |
|  3 | root | ... | Query   |    0 | starting | show processlist     |
+----+------+-----+---------+------+----------+----------------------+
2 rows in set (0,00 sec)
```

Locks and Federated

```
mysql> create table t1(f1 int) engine = federated connection='mysql://root@127.0.0.1:13000/test/t1';
Query OK, 0 rows affected (0,05 sec)

source> begin;
Query OK, 0 rows affected (0,00 sec)

source> update t1 set f1=f1*2;
Query OK, 3 rows affected (0,02 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> update t1 set f1=f1*3;
ERROR 1159 (08S01): Got timeout reading communication packets
```

Locks and Federated: MDL forever

```
mysql> optimize table t1;
-- waits

mysql-monitor> show processlist;
+----+------+-----------------+------+---------+------+-------------+------------------+
| Id | User | Host            | db   | Command | Time | State       | Info             |
+----+------+-----------------+------+---------+------+-------------+------------------+
|  2 | root | localhost:43268 | test | Query   |    4 | System lock | optimize table t1 |
|  4 | root | localhost:43430 | NULL | Query   |    0 | starting    | show processlist  |
+----+------+-----------------+------+---------+------+-------------+------------------+
2 rows in set (0,00 sec)

source> show processlist;
+----+------+-----+---------+------+---------------------------------+--------------------+
| Id | User | ... | Command | Time | State                           | Info               |
+----+------+-----+---------+------+---------------------------------+--------------------+
|  4 | root | ... | Sleep   | 1071 |                                 | NULL               |
|  6 | root | ... | Query   |   11 | Waiting for table metadata lock | OPTIMIZE TABLE `t1` |
...
```

Locks and Federated: MDL forever

```
mysql> optimize table t1;
+---------+----------+----------+-------------------------------------------------+
| Table   | Op       | Msg_type | Msg_text                                        |
+---------+----------+----------+-------------------------------------------------+
| test.t1 | optimize | Error    | Got timeout reading communication packets       |
| test.t1 | optimize | error    | Unknown - internal error 10000 during operation |
+---------+----------+----------+-------------------------------------------------+
2 rows in set (30,43 sec)

source> show processlist;
+----+------+-----+---------+------+---------------------------------+---------------------+
| Id | User | ... | Command | Time | State                           | Info                |
+----+------+-----+---------+------+---------------------------------+---------------------+
|  4 | root | ... | Sleep   | 1104 |                                 | NULL                |
|  6 | root | ... | Query   |   44 | Waiting for table metadata lock | OPTIMIZE TABLE ‘t1‘ |
...
```

Locks and Federated: MDL forever more

```
source> select * from t1;
-- hangs

mysql> update t1 set f1=f1*3;
ERROR 1159 (08S01): Got timeout reading communication packets

source> show processlist;
+----+------+-----+---------+------+---------------------------------+-------------------------------+
| Id | User | ... | Command | Time | State                           | Info                          |
+----+------+-----+---------+------+---------------------------------+-------------------------------+
|  4 | root | ... | Sleep   | 2343 |                                 | NULL                          |
|  6 | root | ... | Query   | 1283 | Waiting for table metadata lock | OPTIMIZE TABLE `t1`           |
| 10 | root | ... | Query   | 1205 | Waiting for table metadata lock | select * from t1              |
| 11 | root | ... | Query   |    0 | starting                        | show processlist              |
| 12 | root | ... | Query   | 1181 | Waiting for table metadata lock | SELECT `f1` FROM `t1` WHERE 1=0 |
| 13 | root | ... | Query   | 1151 | Waiting for table metadata lock | SELECT `f1` FROM `t1` WHERE 1=0 |
| 14 | root | ... | Query   | 1121 | Waiting for table metadata lock | SELECT `f1` FROM `t1` WHERE 1=0 |
+----+------+-----+---------+------+---------------------------------+-------------------------------+
7 rows in set (0,00 sec)
```

Different level locks and InnoDB

```
trx> create table t1(f1 int) engine=innodb;
Query OK, 0 rows affected (0,37 sec)
trx> create table t2(f1 int) engine=innodb;
Query OK, 0 rows affected (0,42 sec)
trx> insert into t1 values(2),(4),(6);
Query OK, 3 rows affected (0,09 sec)
Records: 3  Duplicates: 0  Warnings: 0
trx> insert into t2 (2),(4),(6),(2),(4);
Query OK, 5 rows affected (0,14 sec)
Records: 5  Duplicates: 0  Warnings: 0

trx> select @@tx_isolation;
+-----------------+
| @@tx_isolation  |
+-----------------+
| REPEATABLE-READ |
+-----------------+
1 row in set (0,00 sec)
```

Different level locks and InnoDB

```
trx> begin;
Query OK, 0 rows affected (0,00 sec)
trx> insert into t2 select * from t1;
Query OK, 3 rows affected (0,00 sec)
Records: 3  Duplicates: 0  Warnings: 0
trx> select * from t2;
+------+
| f1   |
+------+
|    2 |
|    4 |
|    6 |
|    2 |
|    4 |
|    2 |
|    4 |
|    6 |
+------+
8 rows in set (0,00 sec)
```

Different level locks and InnoDB

```
lock table> lock table t2 write;
-- waits MDL

alter> alter table t1 engine=innodb;
-- waits MDL

trx> update t1 set f1 = f1/2;
ERROR 1213 (40001): Deadlock found when trying to get lock; try restarting transaction

-- Transaction rolled back!
Query OK, 0 rows affected (21,12 sec)
Records: 0  Duplicates: 0  Warnings: 0
alter>
```

Different level locks and InnoDB

```
-- MDL released!
Query OK, 0 rows affected (37,27 sec)
lock table> delete from t2 limit 5;
Query OK, 5 rows affected (0,08 sec)

lock table> unlock tables;
Query OK, 0 rows affected (0,00 sec)

-- We can modify table now
```

Different level locks and TokuDB

```
trx> create table t1(f1 int) engine=tokudb;
Query OK, 0 rows affected (0,37 sec)
trx> create table t2(f1 int) engine=tokudb;
Query OK, 0 rows affected (0,42 sec)
trx> insert into t1 values(2),(4),(6);
Query OK, 3 rows affected (0,09 sec)
Records: 3  Duplicates: 0  Warnings: 0
trx> insert into t2 (2),(4),(6),(2),(4);
Query OK, 5 rows affected (0,14 sec)
Records: 5  Duplicates: 0  Warnings: 0

trx> select @@tx_isolation;
+-----------------+
| @@tx_isolation  |
+-----------------+
| REPEATABLE-READ |
+-----------------+
1 row in set (0,00 sec)
```

Different level locks and TokuDB

```
trx> begin;
Query OK, 0 rows affected (0,00 sec)
trx> insert into t2 select * from t1;
Query OK, 3 rows affected (0,00 sec)
Records: 3  Duplicates: 0  Warnings: 0
trx> select * from t2;
+------+
| f1   |
+------+
|    2 |
|    4 |
|    6 |
|    2 |
|    4 |
|    2 |
|    4 |
|    6 |
+------+
8 rows in set (0,00 sec)
```

Different level locks and TokuDB

```
lock table> lock table t2 write;
-- waits MDL

alter> alter table t1 engine=innodb;
ERROR 1205 (HY000): Lock wait timeout exceeded; try restarting transaction
alter> alter table t1 engine=tokudb;
ERROR 1205 (HY000): Lock wait timeout exceeded; try restarting transaction
-- Transaction, started earlier, has higher priority
alter> alter table t1 add f2 int;
-- waits MDL

mysql-monitor> show processlist;
+----+------+-----+---------+------+---------------------------------+--------------------------+
| Id | User | ... | Command | Time | State                           | Info                     |
+----+------+-----+---------+------+---------------------------------+--------------------------+
| 66 | root | ... | Query   |  153 | Waiting for table metadata lock | lock table t2 write      |
| 67 | root | ... | Query   |   26 | Waiting for table metadata lock | alter table t1 add f2 int |
| 68 | root | ... | Sleep   |  161 |                                 | NULL                     |
...
```

Replication

- Requires synchronization of the binary log and table content
  - More IO
- SBR does not allow certain statements and transaction isolation levels

Replication

- Requires synchronization of the binary log and table content
- SBR does not allow certain statements and transaction isolation levels

```
mysql> set transaction isolation level read committed;
Query OK, 0 rows affected (0,00 sec)

mysql> update t1 set f1=f1*2;
ERROR 1665 (HY000): Cannot execute statement: impossible to write to binary log since
BINLOG_FORMAT = STATEMENT and at least one table uses a storage engine limited to
row-based logging. InnoDB is limited to row-logging when transaction isolation level
is READ COMMITTED or READ UNCOMMITTED.
```

Replication

- Requires synchronization of the binary log and table content
- SBR does not allow certain statements and transaction isolation levels
- RBR require data
  - Blackhole does not store data
    - INSERTs still logged
    - UPDATEs and DELETEs not
    - Trick with intermediary slave which used Blackhole was broken in certain versions
- Engine can implement specific replication features

Replication

- Requires synchronization of the binary log and table content
- SBR does not allow certain statements and transaction isolation levels
- RBR require data
- Engine can implement specific replication features
  - Read free replication in TokuDB

Diagnostic

- Storage engine must implement diagnostic

```
mysql> check table t1\G
*************************** 1. row ***************************
   Table: test.t1
      Op: check
Msg_type: note
Msg_text: The storage engine for the table doesn't support check
1 row in set (0,00 sec)

mysql> show engine myisam status;
Empty set (0,00 sec)
```

Diagnostic

- Storage engine must implement diagnostic
- Storage engine can implement diagnostic differently
  - InnoDB

```
mysql> SHOW ENGINE INNODB STATUS\G
*************************** 1. row ***************************
  Type: InnoDB
  Name:
Status:
=====================================
2016-07-13 00:23:55 0x7f3c613be700 INNODB MONITOR OUTPUT
=====================================
Per second averages calculated from the last 3 seconds
-----------------
BACKGROUND THREAD
-----------------
srv_master_thread loops: 4 srv_active, 0 srv_shutdown, 463 srv_idle
...
```

Diagnostic

- Storage engine must implement diagnostic
- Storage engine can implement diagnostic differently
  - TokuDB

```
mysql> SHOW ENGINE TOKUDB STATUS;
+--------+------------------------------+------------------------------------------------+
| Type   | Name                         | Status                                         |
+--------+------------------------------+------------------------------------------------+
| TokuDB | disk free space              | more than 10 percent of total file system space |
| TokuDB | time of environment creation | Sat Jun 18 01:19:59 2016                       |
| TokuDB | time of engine startup       | Mon Jul  4 12:02:18 2016                       |
| TokuDB | time now                     | Wed Jul 13 00:25:33 2016                       |
| TokuDB | db opens                     | 30                                             |
| TokuDB | db closes                    | 5                                              |
| TokuDB | num open dbs now             | 25                                             |
| TokuDB | max open dbs                 | 25                                             |
...
```

Diagnostic

- Storage engine must implement diagnostic
- Storage engine can implement diagnostic differently
  - Performance Schema

```
mysql> SHOW ENGINE PERFORMANCE_SCHEMA STATUS;
+--------------------+---------------------------------+----------+
| Type               | Name                            | Status   |
+--------------------+---------------------------------+----------+
| performance_schema | events_waits_current.size       | 176      |
| performance_schema | events_waits_current.count      | 1536     |
| performance_schema | events_waits_history.size       | 176      |
| performance_schema | events_waits_history.count      | 2560     |
| performance_schema | events_waits_history.memory     | 450560   |
| performance_schema | events_waits_history_long.size  | 176      |
| performance_schema | events_waits_history_long.count | 10000    |
...
| performance_schema | performance_schema.memory       | 89409272 |
+--------------------+---------------------------------+----------+
```

Diagnostic

- Storage engine must implement diagnostic
- Information Schema
  - This is engine job to have tables in Information Schema

```
mysql> select table_name from information_schema.tables where
table_schema='information_schema' and 1 in (select table_name like concat('%',
upper(engine), '%') from information_schema.engines);
+----------------------------+
| table_name                 |
+----------------------------+
| INNODB_LOCKS               |
| INNODB_TRX                 |
...
| INNODB_SYS_TABLESTATS      |
+----------------------------+
30 rows in set (0,00 sec)
```

Diagnostic

- Storage engine must implement diagnostic
- Information Schema
- Performance Schema
  - Engine must implement instrumentation in its own code

```
mysql> select name from ‘performance_schema‘.‘setup_instruments‘ join
‘information_schema‘.‘engines‘ where name like concat(’%/’, engine,’/%’);
+-------------------------------------------------------------------------------+
| name                                                                          |
+-------------------------------------------------------------------------------+
| wait/synch/mutex/myisam/MI_SORT_INFO::mutex                                   |
...
| wait/synch/mutex/csv/TINA_SHARE::mutex                                        |
| wait/synch/mutex/innodb/commit_cond_mutex                                     |
...
```

Diagnostic

- Storage engine must implement diagnostic
- Information Schema
- Performance Schema
  - Engine must implement instrumentation in its own code
  - TokuDB currently has no instrumentation

Options

- Storage engine can have its own variables
- Non-specific variables still can affect behavior
  - binlog_format
  - tx_isolation
  - unique_checks
    - Up to engine to raise duplicate key error or not

Options and tools

How to find engine-specific options?

- Usually start with engine_name_
- MyISAM has few which do not follow this format

How to find engine-specific options?

- MyISAM has few which do not follow this format
  - bulk_insert_buffer_size
    - Cache for bulk INSERTs (INSERT ... SELECT, INSERT ... VALUES (...), (...), ..., and LOAD DATA INFILE)
  - concurrent_insert
    - Allow concurrent INSERT and SELECT statements?
  - delay_key_write
    - Keys can be corrupted!
  - ft_*
    - FULLTEXT index options

How to find engine-specific options?

- MyISAM has few which do not follow this format
  - key_buffer_size
    - Shared buffer for MyISAM keys
  - query_cache_wlock_invalidate
    - If LOCK TABLE ... WRITE removes table from the query cache
  - skip_external_locking
    - If MySQL uses system locking for MyISAM tables
    - Can cause corruption if enabled

How to find engine-specific tools?

- Usually contain engine_name
  - myisamchk
    - Table maintenance utility
  - myisam_ftdump
    - Dumps content of the FULLTEXT indexes
    - InnoDB uses tables in Information Schema
  - myisamlog
    - Displays MyISAM log content
  - myisampack
    - Creates compressed read-only tables
- Sometimes only part of the engine_name

How to find engine-specific tools?

- Usually contain engine_name
- Sometimes only part of the engine_name
  - innochecksum
    - Offline checksum utility
  - ps_tokudb_admin
    - Installs/uninstalls TokuDB and TokuBackup plugins
  - tokuftdump
    - Interactive utility to see what messages and/or switch between FTs
  - tokuft_logprint
    - Dumps TokuDB log file to STDOUT

Summary

Summary

- Engine is responsible for data
- Can affect other server parts
- Can be affected by other server parts
- Study engine tools to troubleshoot effectively

More information

- MySQL Server Option and Variable Reference
- Alternative Storage Engines
- The InnoDB Storage Engine
- Percona TokuDB
- RocksDB SE for MySQL: MySQLOnRocksDB

Place for your questions

???

Thank you!

http://www.slideshare.net/SvetaSmirnova

https://twitter.com/svetsmirnova