Upcoming Changes in MySQL 5.7
Morgan Tocker, MySQL Community Manager
http://www.tocker.ca/
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.
Almost 4 Years of MySQL Innovation

MySQL 5.6
MySQL 5.5
MySQL 5.7
MySQL Enterprise
MySQL Migration Wizard
MySQL Workbench 6.0
MySQL Workbench 5.2 & 6.0
MySQL Cluster 7.3
MySQL Cluster 7.2
MySQL Cluster 7.1
MySQL Appliances for Hadoop
MySQL Utilities
Windows installer & Tools
Oracle Certifications
MySQL Enterprise Monitor 2.3 & 3.0
MySQL Enterprise Backup
MySQL Enterprise Security
MySQL Enterprise Scalability
MySQL Enterprise HA
MySQL Enterprise Audit
MySQL 5.7

- Absolutely amazing things are coming!
  - Performance Improvements with InnoDB.
  - Many more…
- New features won’t be the focus of this talk.
Subject of this talk

- Oracle is committed to MySQL’s long term success.
- Prepared to do the heavy lifting required to improve the overall server architecture.
- This talk is about:

  Sharing our ideas.
  Providing you an opportunity for feedback.
  Trying to make the transition as smooth as possible.

Photo Credit: [http://www.flickr.com/photos/magnusvk](http://www.flickr.com/photos/magnusvk)
MySQL 5.7 (cont.)

- Breakage is *painful*.
- It is important to:
  - be as careful as possible
  - provide as much notice as possible.
  - communicate *why* change is required.
Ways to coerce users in the right direction

1. Change the default to use desired behaviour.
2. Issue warning about feature deprecation.
3. Remove feature.

Situation dependent.
MySQL 5.7 employs all 3 techniques.
“Deprecation is a status applied to a computer software feature, characteristic, or practice indicating it should be avoided, typically because it is being superseded.”

Deprecation

• Will understandably upset users.
• However, it is not the same as removal.
• Good stewardship to define an “officially supported method” when >1 exist.
The Main Event

- These are the potential changes we have asked for feedback on. All appear at: http://www.tocker.ca/categories/community/

- Very important! Being listed here does not guarantee any change will be made.
Changes to Replication Defaults

• MySQL 5.6 improved durable replication performance considerably (group commit fix)
• Proposal is to make replication durable by default starting from 5.7:
  • sync_binlog = 1
  • master-info-repository = TABLE
  • relay-log-info-repository = TABLE
### SHOW ENGINE INNODB MUTEX

```sql
mysql> show engine innodb mutex;
+---------+-----------------+-------------------+
<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>InnoDB</td>
<td>log/log0log.c:775</td>
<td>os_waits=26</td>
</tr>
<tr>
<td>InnoDB</td>
<td>log/log0log.c:771</td>
<td>os_waits=1</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=3219</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=6990</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=4619</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=5627</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=7873</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=4466</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=16929</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=19305</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=16301962</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=11649</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=950471</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=6545</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=4262</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=5642</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=7878</td>
</tr>
<tr>
<td>InnoDB</td>
<td>buf/buf0buf.c:1208</td>
<td>os_waits=387166</td>
</tr>
<tr>
<td>InnoDB</td>
<td>fil/fil0fil.c:1559</td>
<td>os_waits=1265</td>
</tr>
<tr>
<td>InnoDB</td>
<td>srv/srv0srv.c:987</td>
<td>os_waits=460452</td>
</tr>
<tr>
<td>InnoDB</td>
<td>combined buf/buf0buf.c:900</td>
<td>os_waits=38503</td>
</tr>
<tr>
<td>InnoDB</td>
<td>log/log0log.c:832</td>
<td>os_waits=184</td>
</tr>
<tr>
<td>InnoDB</td>
<td>combined buf/buf0buf.c:901</td>
<td>os_waits=77</td>
</tr>
</tbody>
</table>
+---------+-----------------+-------------------+
23 rows in set (0.56 sec)
```
SHOW ENGINE INNODB MUTEX (cont.)

- Overlaps with Performance Schema.
- **Proposal in MySQL 5.7:**
  - To be deprecated.
  - We want to be able to ‘pick a favourite’ in order to keep user experience consistent.
InnoDB Monitor tables

• Existed in a time when information_schema did not. Usage was:

CREATE TABLE innodb_monitor (a INT) ENGINE=INNODB;

/* view to the error log */

DROP TABLE innodb_monitor;
• **5.7 Proposal:**
  - `innodb_monitor`. To be replaced with `SET GLOBAL innodb_monitor=ON|OFF`.
  - `innodb_lock_monitor`. To be replaced with `SET GLOBAL innodb_lock_monitor=ON|OFF`.
  - `innodb_tablespace_monitor`. To be removed. `information_schema` will become the recommended alternative.
  - `innodb_table_monitor`. To be removed. `information_schema` will become the recommended alternative.
  - `innodb_mem_validate`. To be removed. This depends on `UNIV_MEM_DEBUG`, which is not normally enabled even in debug builds.
ALTER IGNORE TABLE

- Useful for adding PRIMARY/UNIQUE keys on tables with duplicate keys.
  - Will silently drop rows for you!
- Has strange semantics for replication and ALTER TABLE.
ALTER IGNORE TABLE (cont.)

• Proposal went ahead already:
  • Deprecated in 5.6.17.
  • Removed in 5.7 DMR4
Simplify SQL_MODE options

- Proposal is to reduce the number of mode options available.
- Aim is to increase usage of SQL_MODE options, encourage more users to run MySQL in a stricter way.
Simplify SQL_MODE options (cont.)

- Current recommendation:
  
  sql-mode="STRICT_TRANS_TABLES, ERROR_FOR_DIVISION_BY_ZERO, NO_AUTO_CREATE_USER, NO_AUTO_VALUE_ON_ZERO, NO_ENGINE_SUBSTITUTION, NO_ZERO_DATE, NO_ZERO_IN_DATE, ONLY_FULL_GROUP_BY"
5.7 Proposal:

- Remove the options `ERROR_FOR_DIVISION_BY_ZERO, NO_ZERO_DATE` and `NO_ZERO_IN_DATE`.
- These behaviours be enabled by `STRICT_TRANS_TABLES` or `STRICT_ALL_TABLES`.
- Improve error reporting when a behaviour is influenced by an SQL mode.
Enable Only Full Group By SQL_MODE

mysql> SELECT id, invoice_id, description 
FROM invoice_line_items GROUP BY invoice_id;

+-----+-----------------+------------------+
| id  | invoice_id      | description      |
+-----+-----------------+------------------+
| 1   | 1               | New socks        |
| 3   | 2               | Shoes            |
| 5   | 3               | Tie              |
+-----+-----------------+------------------+

3 rows in set (0.00 sec)
Only Full Group By (cont.)

- Proposal is to by default product an error from MySQL 5.7.
- Will still be configurable via SQL MODE only_full_group_by.

```sql
mysql> SELECT id, invoice_id, description
    FROM invoice_line_items GROUP BY invoice_id;
ERROR 1055 (42000): 'test.invoice_line_items.id' isn't in GROUP BY
```
EXPLAIN PARTITIONS and EXPLAIN EXTENDED deprecation

mysql> EXPLAIN SELECT * FROM table_a\G
*************************** 1. row ***************************
  id: 1
  select_type: PRIMARY
  table: table_a
  partitions: NULL
  type: ALL
  possible_keys: NULL
  key: NULL
  key_len: NULL
  ref: NULL
  rows: 3
  filtered: 100.00
  Extra: NULL
2 rows in set, 1 warning (0.00 sec)
EXPLAIN PARTITIONS and EXPLAIN EXTENDED deprecation

• Motivated by Internal Code Cleanup
• Two optional flags that only add value - no reason other than automated tooling format preferences not to enable them.
EXPLAIN PARTITIONS and EXPLAIN EXTENDED deprecation (cont.)

• Proposal:
  • Enables two flags by default, and deprecates the use of the extended syntax.
  • Went ahead in 5.7 DMR3.
Deprecation of NULL synonym \N

mysql> SELECT NULL is \N;
+--------------+
| NULL is \N   |
+--------------+
| 1            |
+--------------+
1 row in set (0.00 sec)

mysql> INSERT INTO tablea VALUES (3, \N);
Query OK, 1 row affected (0.00 sec)

mysql> SELECT * FROM tablea WHERE b IS \N;
+------+
| a    |
| b    |
+------+
| 3    |
| NULL |
+------+
1 row in set (0.00 sec)
Deprecation of NULL synonym \N (cont.)

- Proposal:
  - For removal in either 5.7 or 5.8.
  - Still seeking feedback.
Query Cache

- Defaults to DISABLED in 5.6.
- We have “ideas but not plans” for future improvements.
Federated Storage Engine

• Users encouraged to use Multi-source replication over Federated Storage Engine.
• We are seeking feedback for use cases for Federated.
MyISAM Merge

• Similar to Partitioning but MyISAM Only.
  - Does not offer partition pruning.
• Major limitation was partition exchange - introduced in 5.6.
• Will blog seeking remaining use cases for MERGE storage engine very soon.
Last Comment

- All of these items have been blogged about here:
  http://www.tocker.ca/categories/community/