MariaDB 10.6
What's New?

Vicențiu Ciorbaru
Software Developer Team Lead
@ MariaDB Foundation
whoami

- Vicențiu Ciorbaru
- MariaDB Foundation, Software Developer Team Lead
- MariaDB developer since 2013-
- Implemented Roles, Window Functions and others
Oracle Syntax Functionality

- **MDEV-20025** ADD_MONTHS(date, N)
  Syntactic sugar for: DATE_ADD(date, interval N months)

- **MDEV-20017** TO_CHAR(expr, fmt)
  - Only works for date / time / timestamp
  - Supports
    - YYYY/YYY/YY/RRRR/RR/
    - MM/MON/MONTH/
    - DD/DY/
    - HH/HH12/HH24/
    - MI/SS

- **MDEV-24285** SYS_GUID()
  Syntactic sugar for UUID()
Oracle Syntax Functionality

- **MDEV-24089** ROWNUM
  - `select rownum(), results.* from results;`
    ```
    +----------+-------+-------+
    | rownum() | name  | score |
    +----------+-------+-------+
    |        1 | Alice |    10 |
    |        2 |  Bill |     9 |
    |        3 |  Judy |     8 |
    |        4 |  Jane |     8 |
    +----------+-------+-------+
    ```
  - Behaves similarly to `row_number()` over ()
  - Can be used directly in WHERE clause. Results not always deterministic.
    ```sql
    select rownum(), results.* from results where rownum() > 2;
    Empty set (0.001 sec)
    ```
  - LIMIT optimization applies in practical cases.
    ```sql
    SELECT ... WHERE rownum() < 5
    ```
SQL Syntax Additions

- SELECT ... OFFSET ... FETCH (MDEV-23908)

OFFSET start { ROW | ROWS }
[FETCH { FIRST | NEXT } [ count ] { ROW | ROWS } { ONLY | WITH TIES }]

- Alternative to LIMIT clause if used with ONLY
- WITH TIES has special meaning
- OFFSET is now a reserved keyword. Columns / Table names will require back-quoting when using this name.
OFFSET start { ROW | ROWS }
[FETCH { FIRST | NEXT } [ count ] { ROW | ROWS } { ONLY | WITH TIES }]

1. WITH TIES requires ORDER BY
2. It will return up-to <count> rows plus final "ties" according to ORDER BY

```
SELECT name, score
FROM results
ORDER BY score desc
FETCH FIRST 3 ROWS WITH TIES
```

<table>
<thead>
<tr>
<th>name</th>
<th>score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>10</td>
</tr>
<tr>
<td>Bill</td>
<td>9</td>
</tr>
<tr>
<td>Judy</td>
<td>8</td>
</tr>
<tr>
<td>Jane</td>
<td>8</td>
</tr>
</tbody>
</table>

● GROUP BY & ORDER BY optimizations work where possible
SQL Syntax Additions

MDEV-13115 SELECT ... SKIP LOCKED

- Allows one to select rows that were not locked (only works with InnoDB)

# Connection 1
start transaction;
select id, name, score from scores
where id = 1 for update;

+----+-------+-------+
| id | name  | score |
+----+-------+-------+
|  1 | Alice |  10   |
+----+-------+-------+

# Connection 2
start transaction;
select * from scores
for update skip locked;

+----+------+-------+
| id | name | score |
+----+------+-------+
|  2 | Bill |   9   |
|  3 | Judy |   8   |
|  4 | Jane |   8   |
+----+------+-------+
**MDEV-7317 Ignorable Indexes**

- Allows one to maintain an index, but force all non-insert queries to ignore it.

```sql
CREATE TABLE t1 (id INT PRIMARY KEY, b INT, KEY k1(b));
ALTER TABLE t1 ALTER INDEX k1 ignored;
```

```
EXPLAIN SELECT * FROM t1 ORDER BY b;
+----+------------+-------+------+---------------+------+---------+------+------+----------------+
<table>
<thead>
<tr>
<th>id</th>
<th>select_type</th>
<th>table</th>
<th>type</th>
<th>possible_keys</th>
<th>key</th>
<th>key_len</th>
<th>ref</th>
<th>rows</th>
<th>Extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SIMPLE</td>
<td>t1</td>
<td>ALL</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
<td>3</td>
<td>Using filesort</td>
</tr>
<tr>
<td>----</td>
<td>-------------</td>
<td>-------</td>
<td>------</td>
<td>---------------</td>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>------</td>
<td>----------------</td>
</tr>
</tbody>
</table>
```

```sql
ALTER TABLE t1 ALTER INDEX k1 not ignored;
```

```
EXPLAIN SELECT * FROM t1 ORDER BY b;
+----+------------+-------+------+---------------+------+---------+------+------+-------------+
<table>
<thead>
<tr>
<th>id</th>
<th>select_type</th>
<th>table</th>
<th>type</th>
<th>possible_keys</th>
<th>key</th>
<th>key_len</th>
<th>ref</th>
<th>rows</th>
<th>Extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SIMPLE</td>
<td>t1</td>
<td>ALL</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
<td>NULL</td>
<td>3</td>
<td>Using index</td>
</tr>
<tr>
<td>----</td>
<td>-------------</td>
<td>-------</td>
<td>------</td>
<td>---------------</td>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
```
SQL Syntax Additions

**MDEV-17399 JSON_TABLE**
- SQL Standard Feature
- Allows one to "normalize" JSON data. (Turn JSON into table rows)
- Complex syntax supporting multiple layers of nesting.
- Dedicated talk.

```sql
select * from JSON_TABLE('[1, 2, 3]', '[$*]' columns(id int PATH '$')) t;
```

<table>
<thead>
<tr>
<th>id</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
</tbody>
</table>

+-----+
| id  |
+-----+
Atomic DDL

Meta Task:
MDEV-17567 Atomic DDL

MDEV-24576 Atomic CREATE TABLE
MDEV-25180 Atomic ALTER TABLE
(InnoDB & RocksDB)
MDEV-23844 Atomic DROP TABLE
MDEV-23842 Atomic RENAME TABLE

MDEV-24746 Atomic CREATE TRIGGER
MDEV-24395 Atomic DROP TRIGGER

MDEV-24607 Atomic CREATE VIEW
MDEV-24408 Atomic DROP DATABASE

- Great deal of refactoring was done
- Code instrumentation to force crash at each point during the DDL operation
- Testing of crash recovery after each point.
- Additional benefits: CREATE OR REPLACE is now atomic as well.
SystemD Socket Activation

- MariaDB 10.6 supports socket activation from systemd

- SystemD allows starting processes based on incoming connections on dedicated sockets
  
  ```
  systemctl start mariadb.socket
  ```

- This allows MariaDB to be started on-demand
  - Useful in a shared-hosting environment for fast startup.
  - Different sockets for different servers on the same machine.

- Work is being done to possibly shut down server when idle and socket activated.
Sys Schema

- Sys Schema is now bundled with MariaDB 10.6
- Collection of views / procedures / functions useful for monitoring MariaDB
- Requires `performance_schema=ON`
- Views defined use `mariadb.sys@localhost` as definer
- Port of MySQL's version, but missing certain MySQL only functionality
InnoDB

- Performance improvements when inserting into an **empty table**
  "InnoDB Bulk Insert" [MDEV-515](https://mariadb.org) & [MDEV-24818](https://mariadb.org)
  - If SET unique_checks=0, foreign_key_checks=0;
  - First insertion into a table is done via table-level locking

- [MDEV-21452](https://mariadb.org) & [MDEV-25404](https://mariadb.org) - Removing home grown mutexes to improve performance
  - Side effect: Removed lock_wait timeout wakeup thread.

- [MDEV-24738](https://mariadb.org) - Improve InnoDB's deadlock detector.
  - A new configuration variable innodb_deadlock_report (OFF, BASIC, FULL)

- [MDEV-24883](https://mariadb.org) Support io_uring (liburing) for better scalability with very fast IO hardware.

- [MDEV-25180](https://mariadb.org) Refactoring to support atomic alter table
Thank you!

Contact details:

vicentiu@mariadb.org

About:

https://mariadb.org/vicentiu