NoSQL + SQL = MySQL 8.0

Tomas Ulin
VP Engineering, MySQL
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.
MySQL Powers the Web
MySQL Innovation: 5.7 -> 8.0

MySQL 5.7
- 3x Better Performance
- Replication Enhancements
- Optimizer Cost Model
- JSON Support
- Improved Security
- Sys & Performance Schema
- GIS

MySQL InnoDB Cluster
- MySQL Group Replication
- MySQL Router
- MySQL Shell

MySQL 8.0
- NoSQL Document Store
- JSON
- CTEs
- Window Functions
- Data Dictionary
- InnoDB
- Replication
- Roles
- Unicode
- GIS

2 Years in Development
400+ Worklogs
5000+ Bugs Fixed
500 New Tests
MySQL 8.0: Document Store

NoSQL + SQL

• Document oriented data storage for MySQL
  – Full JSON document support through SQL and new X DevAPI NoSQL interface

• Schema-less and schema based data in the same technology stack
  – Use COLLECTIONs of documents & relational TABLEs together

• Rapid Prototyping & Simple CRUD APIs
  – Modern APIs using “method chaining” and asynchronous execution (e.g. promises, callbacks, etc.)

• Connectors for many different languages and frameworks
  – Node.JS, Java, NET, C++/C, PHP, Python
MySQL 8.0: Document Store Architecture

Applications
- MySQL Connectors (Node.js, Python, Java, .Net)

Scripting
- MySQL Shell

SQL API
- CRUD and SQL APIs
- Std Protocol
- X Protocol

SQL
- Relational Tables

NoSQL
- JSON Collections

 août "_id": "232342",
  "firstname": "John",
  "lastname": "Doe",
  "city": "anytown"}

City | Name  | Pop | Area |
--- | --- | --- | --- |
   | Anytown | 2003 | 33   |
   | Othertown | 23000 | 400  |
MySQL 8.0: Document Store
Designed for modern Developers

• Full Node.js integration – Support for “Promises”
• Autocompletion support in IDEs – Due to method chaining support
• Intuitive Documentation & Tutorials – Example:

**COLLECTION.add Function**
MySQL 8.0: Shell
Get started in minutes

• Rapid prototyping capabilities
  – Using JavaScript and Python

• Full SQL and X DevAPI support
  – With built in auto-completion

• InnoDB Cluster support
  – Setup your HA solution within minutes

• DevOps Tool
  – Designed for DevOps operations
MySQL 8.0: JSON_TABLE Function

• JSON is **NOT** limited to CRUD, can also be used in complex queries
• JSON_TABLE converts JSON into SQL table
• Leverage existing SQL framework for aggregation

• Convert JSON column into a table with 2 columns:

```sql
SELECT people.* FROM t1, JSON_TABLE(json_col, '$.people[*]' COLUMNS (name VARCHAR(40) PATH '$.name', address VARCHAR(100) PATH '$.address')) people WHERE people.address LIKE '%San Francisco%';
```
MySQL 8.0: JSON_TABLE – Nested Arrays

[ 
  { "father":"John", "mother":"Mary", "marriage_date":"2003-12-05", "children": [ 
    { "name":"Eric", "age":12 },
    { "name":"Beth", "age":10 } ] },
  { "father":"Paul", "mother":"Laura", "children": [ 
    { "name":"Sarah", "age":9},
    { "name":"Noah", "age":3},
    { "name":"Peter", "age":1} ] }
]

<table>
<thead>
<tr>
<th>id</th>
<th>father</th>
<th>married</th>
<th>child_id</th>
<th>child</th>
<th>age</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>John</td>
<td>1</td>
<td>1</td>
<td>Eric</td>
<td>12</td>
</tr>
<tr>
<td>1</td>
<td>John</td>
<td>1</td>
<td>2</td>
<td>Beth</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>Paul</td>
<td>0</td>
<td>1</td>
<td>Sarah</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>Paul</td>
<td>0</td>
<td>2</td>
<td>Noah</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Paul</td>
<td>0</td>
<td>3</td>
<td>Peter</td>
<td>1</td>
</tr>
</tbody>
</table>
# MySQL 8.0: JSON Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Function</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>JSON_ARRAY_APPEND()</td>
<td>JSON_MERGE<a href="">_PRESERVE</a></td>
<td>JSON_PRETTY()</td>
</tr>
<tr>
<td>JSON_ARRAY_INSERT()</td>
<td>JSON_OBJECT()</td>
<td>JSON_STORAGE_SIZE()</td>
</tr>
<tr>
<td>JSON_ARRAY()</td>
<td>JSON_QUOTE()</td>
<td>JSON_STORAGE_FREE()</td>
</tr>
<tr>
<td>JSON_CONTAINS_PATH()</td>
<td>JSON_REMOVE()</td>
<td>JSON_ARRAYAGG()</td>
</tr>
<tr>
<td>JSON_CONTAINS()</td>
<td>JSON_REPLACE()</td>
<td>JSON_OBJECTAGG()</td>
</tr>
<tr>
<td>JSON_DEPTH()</td>
<td>JSON_SEARCH()</td>
<td>JSON_MERGE_PATCH()</td>
</tr>
<tr>
<td>JSON_EXTRACT()</td>
<td>JSON_SET()</td>
<td>JSON_TABLE()</td>
</tr>
<tr>
<td>JSON_INSERT()</td>
<td>JSON_TYPE()</td>
<td></td>
</tr>
<tr>
<td>JSON_KEYS()</td>
<td>JSON_UNQUOTE()</td>
<td></td>
</tr>
<tr>
<td>JSON_LENGTH()</td>
<td>JSON_VALID()</td>
<td></td>
</tr>
</tbody>
</table>

---

*Oracle* © 2018, Oracle and/or its affiliates. All rights reserved.
MySQL 8.0: Efficient Replication of JSON Documents

Replicate only changed fields of documents (Partial JSON Updates)

- Numbers are from a specially designed benchmark:
  - tables have 10 JSON fields,
  - each transaction modifies around 10% of the data
MySQL 8.0: Transactional Data Dictionary

Atomic and crash-safe DDL
MySQL 8.0: CTEs & Window Functions

• Common Table Expression (WITH clause)
  – Non-recursive
  – Recursive
  – Used for hierarchy traversal

• Window functions
  – Aggregation, ranking, analytics
  – Used for analytics and reporting
MySQL 8.0: Better Handling of Hot Row Contention

SELECT * FROM tickets
WHERE id IN (1,2,3,4)
AND order_id IS NULL
FOR UPDATE
NOWAIT;

Error immediately if a row is already locked

SELECT * FROM tickets
WHERE id IN (1,2,3,4)
AND order_id IS NULL
FOR UPDATE
SKIP LOCKED;

Non deterministically skip over locked rows
MySQL 8.0: Invisible Indexes

• Indexes are “hidden” to the MySQL Optimizer
  – Not the same as “disabled indexes”
  – Contents are fully up to date and maintained by DML

• Two use cases:
  – Soft Delete (*Recycle Bin*)
  – Staged Rollout
MySQL 8.0: SQL Roles

- Flexible, Properly Architected Roles
- Create and Drop Roles, Grant to Roles
- Grant Roles to Roles, Grant Roles to Users
- Limit Hosts that can use roles, Define Default Roles
- And even visualize Roles with SQL function ROLES_GRAPHML()
MySQL 8.0: Performance Schema

SELECT * FROM sys.session
1000 active sessions

Time in Seconds (Lower is better)
MySQL 8.0: INFORMATION_SCHEMA
100 schemas times 50 tables (5000 tables)

- Count All Schemas
- Schema aggregate size stats
- All Dynamic Table Info
- All Static Table Info
- Auto Increments Near Limit
- Count All Columns
- Count All Indexes

Time in Seconds (Lower is better)

MySQL 8.0
MySQL 5.7

30x Faster
MySQL 8.0: UTF-8 Encoding

• MySQL 8.0 defaults to utf8mb4
• Emoji characters used as input
• Latest Unicode 9.0 Support
• New collations based on DUCET, accent and case sensitive collations
• Chinese, Japanese, Korean
MySQL 8.0: GIS

• Simple
  – Completely built in and ready to use
  – No extra configuration or Installation

• Powerful
  – Full Geography Support
    • Projected – Flat/Across 2 dimensions
    • Geographic – Spheroid
  – Details -
    • 5107 predefined SRSs from the EPSG Dataset 9.2, 4628 projected, 479 geographic

• Developed by experts, now part of Boost - open source
MySQL 8.0: Persist Configuration

• Persist GLOBAL Server Variables
  – `SET PERSIST max_connections = 500;`

• Examples Include:
  – Offline_mode
  – Read_Only

• Requires no filesystem access

• Captures timestamp and change user
MySQL 8.0: InnoDB Improvements

• Document Store/JSON
  – More efficient BLOBs, up to 4x improvement on update
  – Better performance for both small and large BLOBs

• New Scalable WAL
  – Lock free design

• Atomic DDL

• Persistent Auto Increment

• Contention Aware Transaction Scheduling (CATS)
  – Contribution from University of Michigan

• Removed IO bottlenecks
  – More scalable design
MySQL 8.0: Alter Table - Instant Add Column

• Contribution from Tencent
  – Only a metadata change
  – No copying of data
  – Smaller final data size
  – Forward compatibility with old data file

• ALTER TABLE ... ADD COLUMN c, ALGORITHM = INSTANT

• Supports DYNAMIC/COMPACT/REDUNDANT row formats
MySQL 8.0: Replication Improvements

• WRITESET based parallel replication applier
  – JF @ booking.com experiences 10x better performance

• Efficient Replication of JSON Documents
  – Much smaller replication storage and network footprint for JSON fields

• Per Channel replication filters in Multi-source
  – Able to deploy different filters for different channels

• Crash-recovery of DDLs and Replication
  – Resiliency (Replication, Metadata and Data recovered together after a crash)

• Performance schema enhancements
  – Better assessment of slave lag consolidated view over a cluster
MySQL 8.0: Group Replication Improvements

• Specify next-primary in the event of the current Primary failure
  – User can define the next server to win the primary election

• Automatically prevent updates on replicas that leave the cluster
  – Automatically fence servers that drop out of the group

• Online and automated Cluster Performance Management Fine Tuning
  – Flow control has more configuration parameters
“High Availability becomes a core first class feature of MySQL!”
MySQL 8.0: SysBench IO Bound Read Only (Point Selects)

2x Faster than MySQL 5.7

Queries per Second

Users

MySQL 8.0

MySQL 5.7

OS: Oracle Linux 7.4
CPU: 48 cores-HT Intel Skylake 2.7Ghz
(2 CPU sockets, Intel(R) Xeon(R) Platinum 8168 CPU)
RAM: 256GB
Storage: x2 Intel Optane flash devices
(Intel (R) Optane (TM) SSD P4800X Series)
MySQL 8.0: SysBench Read Write (update nokey)

2x Faster than MySQL 5.7

OS: Oracle Linux 7.4
CPU: 48 cores-HT Intel Skylake 2.7Ghz
(2CPU sockets, Intel(R) Xeon(R) Platinum 8168 CPU)
RAM: 256GB
Storage: x2 Intel Optane flash devices
(Intel(R) Optane(TM) SSD P4800X Series)
All these features plus...

- Source code now documented with Doxygen
- Plugin Infrastructure!
- Expanded GIS Support
- Expanded Query Hints Support
- Improved Scan Query Performance
- Improved BLOB Storage
- Improved Memcached Interface
- Cost Model Improvements
- Scalability Improvements
- Atomicity in Privileges
- Parser Refactoring
- Improvements to Temporary Tables
- C++11 and Toolchain Improvements

- GTID_PURGED always settable
- Persistent Auto Increment
- Native InnoDB Data dictionary
- Improved Information Schema performance
- SQL Grouping Function
- Optimizer Trace detailed sort statistics
- Descending Indexes
- Smaller Package Downloads
- JSON Aggregate, Pretty print functions
- JSON performance improvements
- Expanded Query Hints
- Improved usability of cost constant configuration

- Group Replication in 8.0
- Transaction Save Point support in Group Replication
- Improved Replication Applier Lag Statistics in Performance Schema
- Per Multi-source Channel Replication Filters
- Atomic DDL extended to the Binary Log
- Performance Improvements on the Replication Applier
- Parallel Replication Applier Policy
- Binary Log Management Enhancements
- Additional Metadata Into the Binary Log
MySQL 8.0: Upgrade Checker

• Quick and Easy MySQL Shell Utility
  – JavaScript
  – Python

• Identifies Issues Based on Severity
  – No Issues
  – Potential Errors
  – Errors that must be fixed before Upgrading

• Recommends Fixes
  – Schema, Configuration
  – Data on Server, etc.

```bash
$js util.checkForServerUpgrade("root@localhost:3306")
MySQL server at localhost:3306 will now be checked for compatibility issues.
Upgrading to MySQL 8.0...

- issues found:

  age of db objects with names conflicting with reserved keywords in 8.0.

  age of utf8mb3 charset:

    The following objects use the utf8mb3 character set. It is recommended
    to use utf8mb4 instead, for improved Unicode support.

    e_schema.city.name - column's default character set: utf8
    e_schema.city.country_code - column's default character set: utf8

- age of use ZEROFILL/display length type attributes:

  The following table columns specify a ZEROFILL/display length attribute
  be aware that they will be ignored in MySQL 8.0

  big_table.ORDINAL_POSITION - bigint(21) unsigned
```
MySQL 8.0 – Innovating and Evolving

- DocStore
  - SQL & NoSQL
- InnoDB Cluster
  - HA made easy
- SQL
- JSON
- GIS
- Replication
- Reliability
- Observability
- Manageability
- Security
- Performance
- Scalability
Learn More from the Oracle MySQL Engineering Team

• MySQL 8.0: Key Features (Geir Høydalsvik)
• MySQL 8.0: Document Store (Mike Zinner)
• MySQL 8.0: Optimizer (Manyi Lu)
• MySQL 8.0: InnoDB (Sunny Bains)
• MySQL 8.0: Replication (Luis Soares)
• MySQL 8.0: Performance & Benchmarks (Dimitri Kravtchuk)
• MySQL 8.0: How to Save Bandwidth (Georgi Kodinov)
• MySQL InnoDB Cluster in a Nutshell (Frederic Descamps)
• Running JavaScript Stored Programs Inside MySQL (Øystein Grøvlen)