MySQL 9.2 Release Notes

Abstract

This document contains release notes for the changes in MySQL 9.2. For information about changes in a different version of MySQL, see the release notes for that version.

For additional MySQL 9.2 documentation, see the MySQL 9.2 Reference Manual, which includes an overview of features added in MySQL 9.2 (What Is New in MySQL 9.2), and discussion of upgrade issues that you may encounter while upgrading.

MySQL platform support evolves over time; please refer to https://www.mysql.com/support/supportedplatforms/database.html for the latest updates.

Updates to these notes occur as new product features are added, so that everybody can follow the development process. If a recent version is listed here that you cannot find on the download page (https://dev.mysql.com/downloads/), the version has not yet been released.

The documentation included in source and binary distributions may not be fully up to date with respect to release note entries because integration of the documentation occurs at release build time. For the most up-to-date release notes, please refer to the online documentation instead.

For legal information, see the Legal Notices.

For help with using MySQL, please visit the MySQL Forums, where you can discuss your issues with other MySQL users.

Document generated on: 2025-08-12 (revision: 30436)

Table of Contents

Preface and Legal Notices

This document contains release notes for the changes in MySQL 9.2.

Legal Notices

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Changes in MySQL 9.2.0 (2025-01-21, Innovation Release)

- Bugs Fixed

Account Management Notes

- The database cache was not flushed properly following execution of `DROP USER`. (Bug #37132323)
- This release adds the `CREATE_SPATIAL_REFERENCE_SYSTEM` privilege, which allows the user to execute any of the statements `CREATE SPATIAL REFERENCE SYSTEM`, `CREATE OR REPLACE SPATIAL REFERENCE SYSTEM`, and `DROP SPATIAL REFERENCE SYSTEM`. Use of the `SUPER` privilege for this purpose should now be considered deprecated.

  For more information, see Privileges Provided by MySQL. (Bug #37046126, WL #16547)
- Failed password validation was not always handled correctly. (Bug #37041439)

C API Notes

- Process memory usage grew when the `libmysqlclient` API user tried to cache and reuse a prepared statement, preparing it once and then calling either of `mysql_stmt_bind_param()` or `mysql_stmt_bind_named_param()` followed by `mysql_stmt_execute()`, repeatedly without calling the matching `mysql_stmt_close()`, or calling it in the distant future (on application exit, for example).

  We solve this by introducing a separate `MEM_ROOT` object for storing the bind parameters array, which object can be cleared (deallocating the memory) on each call to `mysql_stmt_bind_param()` or `mysql_stmt_bind_named_param()`. (Bug #37202066)
- The Doxygen comments for the `OK_Packet` did not cover all possible cases; this has been updated so that it now accurately reflects the data actually sent by the server. (Bug #35630063)

  References: See also: Bug #35358417, Bug #36374232.

Character Set Support

- Zero rows resulted (where one row was expected) when selecting from a view created with its connection and client character set to `latin1`, where a query on the view used `utf8` as its connection and client character set, the view contained literal values with non-ASCII characters, and the query performed a condition pushdown into a `UNION` of query blocks of the view.

  This problem was related to a previous issue which fixed an error for a similar problem: The problem in that case was properly considering the character set of the view definition when pushing down conditions contained in the view to inner query blocks, but the fix implemented at that time did not take into account the possibility that the view might contain non-ASCII characters.

  This meant that the condition to be pushed down was written to a text string with the wrong character set. We fix this oversight by ensuring that the string is created with the correct character set. (Bug #37111452)

  References: See also: Bug #36246859.

Compilation Notes

- **macOS:** Use `OBJECT` library in place of `routing_guidelines` when compiling with Xcode. (Bug #37350937)
- **macOS:** Removed obsolete CMake code from MacOS builds. (Bug #37258036)

- **macOS:** It is now possible to build MySQL using the Homebrew version of Clang. (Bug #37256912)
- **macOS:** Removed warnings of the form `ld: warning: ignoring duplicate libraries` and warnings specific to `xcodebuild`. (Bug #37065301)
- **Microsoft Windows:** Windows builds are now configured with `-DWITH_ASAN`. (Bug #37309813)
- **Microsoft Windows:** Disabled windows compiler warnings C26445 and C26821 in `cmake/msvc_cppcheck.cmake`. Both of these relate to MSVC substitution of `gsl::span` for `std::span`, which is not used for MySQL. (Bug #37158156)
- **Microsoft Windows:** Removed compiler warnings issued when building MySQL with Visual Studio 2022. (Bug #113870, Bug #36256477)
- **Microsoft Windows:** The CMake option `-DWITH_SASL` is not supported for building MySQL on Windows. Attempting to use this option when doing so now causes the build to terminate with an error. (Bug #113558, Bug #36155972)
- **Solaris:** The minimum required version of GCC to build MySQL on Solaris has been raised to 11.4. See Source Installation Prerequisites, for more information. (Bug #37256600)
- Starting with CMake 3.26, CMake writes the file `CMakeFiles/CMakeConfigureLog.yaml`, which supersedes `CMakeError.log`. References to `CMakeError.log` have therefore been removed. (Bug #37305289)
- Implemented the standards-compliant `my_char_traits<unsigned char>` for use as a drop-in replacement for `std::char_traits<unsigned char>`, which was deprecated in Clang 18 and removed in Clang 19. (Bug #37273525, Bug #37785339)
- The files `stream_cipher.h` and `stream_cipher.cc`, used by `mysys/`, were located in `sql/` but did not depend on any other server code in this directory; these files have now been moved into `mysys/`. (Bug #37257736)
- Removed or fixed issues in a number of copyright header files. (Bug #37238155)
- Removed a **`maybe-uninitialized`** error found in `sql/item.cc` when building MySQL with GCC 14. (Bug #37157201)
- Disabled several `clang-tidy` checks which generated unnecessary warnings when compiling MySQL. (Bug #37129808)
- The CMake code for building `component_mle` assumed that GraalVM binaries were available, either under `/usr/global/share`, or in a location provided on the command line. Now, when neither of these conditions hold, it is possible to download GraalVM from a remote server. (Bug #37121798)
- The version of `libedit` used to compile MySQL was upgraded to 20240808-3.1. (Bug #37101293)
- The server could not be built on Ubuntu 22.04 using Clang 13. (Bug #37075154)
- Removed an error in `mysql_prepare_create_table()` (in the file `sql/sql_table.cc`) found when compiling MySQL with XCode 16. (Bug #37068527)
- Some plugins failed to load when MySQL was built with `-FTLS-MODEL=initial-exec`. Fixed by removing this compiler flag, and falling back to the default (`global-dynamic`) instead. (Bug #37017999)
- MySQL could not be compiled using Clang 19. (Bug #37014761)

- `#include <chrono>` was missing from `plugin/group_replication/libmysqlgcs/src/bindings/xcom/xcom/task.cc` even though `std::chrono::duration_cast()` was referenced in this file. (Bug #116779, Bug #37329617)
- The server could not be built on Fedora 40 (and possibly other Linux platforms) using cmake 3.11, due to an issue with TIRPC. (Bug #116164, Bug #37080195)

Component Notes

- **Group Replication:** This release includes the Group Replication resource Manager component for MySQL Enterprise Edition, which monitors applier channel lag, recovery channel lag, and system resource usage on each secondary, and ejects any member whose lag or memory usage exceeds a user-configurable limit. Expelled servers may attempt to rejoin the group, provided that `group_replication_autorejoin_tries` is not set to `0`.

  The operator can determine upper bounds for applier channel lag, recovery channel lag, and memory usage by setting the system variables `group_replication_resource_manager.applier_channel_lag`, `group_replication_resource_manager.recovery_channel_lag`, and `group_replication_resource_manager.memory_used_limit`, respectively.

  The Group Replication Resource Manager component checks the status of each secondary every 5 seconds, and updates a number of status variables which it provides, showing current channel lag and memory usage, as well as how many times a given threshold has been hit, and if the secondary has ever been expelled from the group, when this last occurred.

  See Group Replication Resource Manager Component, for more information. (WL #14940)

- For the MLE component, the `mysql_option.option_usage` table's `USAGE_DATA` column showed the `used` value as a string rather than as a boolean value.

  For more information, see Option Tracker Tables. (Bug #37122749)

- The entries present in the `mysql_option.option_usage` table depended on whether the Option Tracker was installed before or after the Audit Log.

  See Option Tracker Component, and MySQL Enterprise Audit, for more information. (Bug #37037438)

- Installing the Option Tracker component in one session while executing `UNINSTALL COMPONENT` in a different session caused the server to hang.

  See Option Tracker Component, for more information. (Bug #36991399)

- Errors in configuration files for keyring components were not properly logged. (Bug #36982002)

- `INSTALL COMPONENT` issued concurrently with a `SET PERSIST` which used a subquery could sometimes lead to an unplanned exit of the server. (Bug #36559078)

  References: See also: Bug #35647759.

- This release adds a Connection Control component (`component_connection_control`) intended to replace the Connection Control plugins (see Connection Control Plugins). A single component takes the place of both plugins, which are now deprecated and subject to removal in a future version of MySQL.

  One component replaces both of the deprecated Connection Control plugins; installation of the component requires a single `INSTALL COMPONENT` statement. See Connection Control Component Installation.

The system and status variables associated with the plugin are also deprecated; see Connection Control Component Configuration for information about the variables supported by the Connection Control component in place of the plugin variables.

In addition, the Information Schema `CONNECTION_CONTROL_FAILED_LOGIN_ATTEMPTS` table used by the plugins is now deprecated; the component keeps counts of failed connection attempts in a Performance Schema table `connection_control_failed_login_attempts`.

See The Connection Control Component, for more information. (WL #16573)

Configuration Notes

- **Microsoft Windows:** On Windows, MySQL Configurator no longer executes the deprecated `FLUSH PRIVILEGES` statement. (Bug #37170330)

Deprecation and Removal Notes

- The `FLUSH PRIVILEGES` statement is now deprecated, and causes a warning when issued. You should expect this statement to be removed in a future MySQL release.

  The following constructs are also deprecated, and now cause a warning when used:

  - The `FLUSH_PRIVILEGES` privilege, and granting of this privilege
  - `mysqladmin flush-privileges`
  - `mysqladmin reload`.

  The features listed here do not cause any warnings but should be considered deprecated:

  - Flushing of privileges by `SIGHUP`
  - Flushing of the `caching_sha2` cache by `FLUSH PRIVILEGES`
  - FLushing of privileges by `mysqladmin refresh`

  For more information, see FLUSH Statement, mysqladmin — A MySQL Server Administration Program, and Privileges Provided by MySQL. (WL #16567)

- The Version Tokens plugin is now deprecated, and subject to removal in a future MySQL release. The following related features are also now deprecated, and raise deprecation warnings whenever they are invoked:

  - The functions `version_tokens_delete()`, `version_tokens_edit()`, `version_tokens_lock_exclusive()`, `version_tokens_lock_shared()`, `version_tokens_set()`, `version_tokens_show()`, and `version_tokens_unlock()`
  - The `VERSION_TOKEN_ADMIN` privilege
  - The `version_tokens_session` and `version_tokens_session_number` server system variables

  Attempting to install the `version_tokens` plugin—or to start the server when this plugin is installed—also causes a deprecation warning to be issued.

  For more information, see Version Tokens. (WL #16571)

Doxygen Notes

- The payload length for `COM_QUERY` packets was calculated incorrectly in one of the MySQL packet examples. The payload length is now computed accurately, based on the actual length of the query.

  Our thanks to Konno Satoshi for the contribution. (Bug #116339, Bug #37161043)

Firewall Notes

- In some cases, after performing an upgrade, stored procedures relating to MySQL Enterprise Firewall were not processed correctly. (Bug #36084822)

Installation Notes

- When upgrading from MySQL 5.7 to a later MySQL release series, the system-created `mysql.sys` and `mysql.session` accounts are now modified to use the `caching_sha2_password` authentication plugin instead of the `mysql_native_password` plugin, which is deprecated in MySQL 8.0, and removed in MySQL 9.0. (Bug #36608636)

JavaScript Programs

- The MySQL `ENUM` and `SET` types are now supported for arguments of JavaScript stored routines. For more detailed information, including rules for conversion between these MySQL types and JavaScript types, see Conversion to and from MySQL ENUM and SET. (WL #16599)

- JavaScript programs supported by the MLE Component, available as part of MySQL Enterprise Edition, now supports access from JavaScript routines to user-defined functions, procedures, and variables. MySQL stored functions and procedures can be accessed using the `Schema` methods `getFunction()` and `getProcedure()`which return `Function` objects which can be used with arguments to invoke the routines. `OUT` and `INOUT` parameters of a stored procedure use placeholders created with `mysql.arg()`. For additional information and examples, see Stored Routine API.

  The MLE component also now provides a JavaScript MySQL transaction API which performs the actions of most MySQL transactional SQL statements, such as `START TRANSACTION`, `COMMIT`, `ROLLBACK`, and `SET AUTOCOMMIT`. Support for savepoints is also included. Support for an `SqlError` object is included. For more information, see JavaScript Transaction API, and SqlError Object.

  MySQL user variables can be accessed directly as properties of the JavaScript global `Session` object. For example, a user variable named `myvar` can be read or set as `session.myvar`. See Accessing Session Variables from JavaScript, for more information and examples.

  This release also adds support for direct access to the MySQL builtin functions `RAND()`, `SLEEP()`, `UUID()`, and `IS_UUID()`, as, respectively, `rand()`, `sleep()`, `uuid()`, and `isUUID()`. See MySQL Functions.

  For further information about JavaScript programs and the MLE Component, see JavaScript Stored Programs, and Multilingual Engine Component (MLE). (WL #16585)

- The MLE component, available as part of MySQL Enterprise Edition, now supports reusable JavaScript libraries containing functions which can be called from other JavaScript stored programs. JavaScript libraries can be managed using the `CREATE LIBRARY` and `DROP LIBRARY` SQL statements added in this release; they can be included in other stored JavaScript programs with the `USING` clause added to `CREATE FUNCTION` and `CREATE PROCEDURE`; `USING` supports a list of one or more library names.

  `CREATE LIBRARY` creates a new JavaScript library in a given database. Library code is parsed and checked for validity at creation time, and is rejected if it contains errors. `DROP LIBRARY` drops

a JavaScript library. Library functions can be referred to in other JavaScript stored programs using *`library_name.function_name`* notation. Libraries can be aliased with `USING` when including them in JavaScript programs.

You can obtain the code from a library using the `SHOW CREATE LIBRARY` statement, also new in this release. Two new related Information Schema tables have also been added: The `LIBRARIES` table provides information about JavaScript libraries, and the `ROUTINE_LIBRARIES` provides information about stored routines which include JavaScript libraries.

Counts of library DDL and `SHOW CREATE LIBRARY` statements which have been issued on the server are available as the status variables `Com_create_library`, `Com_drop_library`, and `Com_show_create_library`.

For more information and examples, see Using JavaScript Libraries, as well as Multilingual Engine Component (MLE). (WL #16359, WL #16360, WL #16362, WL #16555)

MySQL Enterprise Notes

- **Replication:** The `rnd_pos()` function of the handler interface for replication applier metrics tables was implemented wrongly; in cases where this function was used, the tables appeared to be missing one row.

  For more information, see Replication Applier Metrics Component. (Bug #37076428, Bug #37132660)

- **Replication:** For most wait operations, we note the time the wait started and the time it ends, and add their difference to the total waiting time, but this was not the case when tracking time spent waiting for the commit order, where we used a function that woke up every second and caused the wait time to be incremented by 1. This 1-second precision was too low for most practical purposes and caused unnecessary complication of the code.

  To fix these issues, we now track waiting time in the same way for commit order waits as we do for other waits.

  For more information, see Replication Applier Metrics Component. (Bug #37053708)

- **Replication:** The MySQL Option Tracker component can now provide usage information about the binary log, group replication, and use of the server as a replica whenever these features are enabled.

  For more information, see Option Tracker Supported Components. (WL #16529)

Optimizer Notes

- **JSON:** Added lookup references to iterator-based `EXPLAIN FORMAT=JSON` for index lookups. The `lookup_references` field in the JSON v2 `EXPLAIN` format corresponds to the `ref` field in the JSON v1 `EXPLAIN` format. (Bug #37126176)

- Some hash joins spent an unreasonably high time in `pack_rows::RequestRowId()`, even though they did not use row IDs.

  Fixed by skipping the loop over `TableCollection::tables()` in `RequestRowId()` in the case where there are no tables from which to request row IDs. (Bug #37243461)

- `EXPLAIN FORMAT=TREE` now shows the clustered primary key scan for RowID-Ordered Retrieval (ROR) intersection plans. (Bug #37199800)

- Pushing down a condition which had an aggregate function in a `WHERE` clause caused the aggregate function to be evaluated when it should not have been. (Bug #36421735)

Performance Schema Notes

- If a user other than root ran `START REPLICA`, `PERFORMANCE_SCHEMA.PROCESSLIST` assigned that user's name to the newly spawned foreground replication threads instead of `system user`.

  As of this release, `system user` is assigned to all foreground system threads. (Bug #37357562)

- `PERFORMANCE_SCHEMA.PROCESSLIST` filtered foreground threads which did not have a user name. As of this release, `system_user` is assigned to foreground threads without a user name. (Bug #37357562)

- Under certain circumstances, a metadata lock can be upgraded or downgraded to a different `LOCK_TYPE`. This change was not reflected in the `PERFORMANCE_SCHEMA.METADATA_LOCKS` table.

  Our thanks to George Ma and the Alibaba team for the contribution. (Bug #116625, Bug #37271768)

Pluggable Authentication

- The following issues relating to the `AUTHENTICATION_POLICY_ADMIN` privilege have now been resolved:

  - For a user having the privilege, it was not possible to create a user for whom a default authentication plugin was omitted, because the defaults specified in the global value of `authentication_policy` were included at creation time whether any were specified or not.

    We fix this by ignoring the global authentication policy when the user performing account creation has the `AUTHENTICATION_POLICY_ADMIN` privilege.

  - For a user not having the privilege, an attempt to create a user without specifying any authentication factors was rejected with an error, because mandatory factors in the default global authentication policy were not included when they were not specified.

    We fix this by including any mandatory factors from the global authentication policy when the user performing the account creation does not have the `AUTHENTICATION_POLICY_ADMIN` privilege

  For more information, see the description of the `authentication_policy` system variable, as well as Privileges Provided by MySQL. (Bug #37027739)

- The `authentication_openid_connect` plugin is now registered with the Option Tracker component.

  For more information, see Option Tracker Component, and OpenID Connect Pluggable Authentication. (Bug #116045, Bug #37041216)

SQL Syntax Notes

- **Important Change:** The `BINLOG` keyword can no longer be used as an unquoted label name in MySQL stored programs. Prior to upgrading to this release, you should update any affected applications accordingly.

  For more information, see Keywords and Reserved Words. (Bug #22574003)

Functionality Added or Changed

- **InnoDB:** During recovery, IBUF merges were disabled for all recovery batches except for the last batch, but is now disabled for all batches. This change prevents the addition of new redo log entries during recovery, as they led to deadlock and recovery performance degradation with reads and writes from the additional IBUF pages. (Bug #33134355, WL #15372)

- **Microsoft Windows:** On Windows, MySQL Configurator has added a command-line interface for configuring a MySQL Server installation. See MySQL Configurator CLI, for more information.

  A known limitation of this release: the new CLI only supports the `configure` action. (WL #16564)

- Binary packages that include `curl` rather than linking to the system `curl` library have been upgraded to use `curl` 8.11.1. (Bug #37389565)

- Refactored code to take advantage of OpenSSL 3.x functionality to improve performance for cryptographic operations, such as those performed by the MySQL `MD5()` and `SHA2()` functions. (Bug #116939, Bug #37185170, WL #16504)

- Added a source code static analysis tool. For usage details, see `scripts/static_analysis.md` in the source code or execute `python3 ./scripts/static_analysis.py --help` for additional help. Use this to check single commits or the entire source code repository. (WL #16329)

- Converted the caching SHA-2 authentication plugin to use the event component API instead of the audit log plugin API. This also removes the `sha2_cache_cleaner` audit plugin. (WL #16572)

Bugs Fixed

- **Incompatible Change:** Corruption occurred in a spatial index when an update of a geometry with a minimal change in the minimum bounding rectangle (MBR) was followed by a delete operation.

  When upgrading to this release, it is recommended that you drop any spatial indexes beforehand, then re-create them after the upgrade is complete. Alternatively, you can drop and re-create such indexes immediately following the upgrade, but before making use of any of the tables in which they occur. You should also be aware that downgrading to any previous version reintroduces the original problem described previously.

  For more information, see Creating Spatial Indexes. (Bug #36452528)

- **InnoDB:** Improved the simulated asynchronous I/O (AIO) handler performance for high volume situations. (Bug #37366607)

- **InnoDB:** Improved asynchronous I/O (AIO) page cleaner thread management performance. (Bug #37359213)

- **InnoDB:** Concurrently truncating a table while querying the Performance Schema sometimes cause MySQL to halt unexpectedly. (Bug #37271715)

- **InnoDB:** It was possible for an `ALTER TABLE` operation using the `INPLACE` algorithm on a table containing both a spatial index and an auto-increment column to cause corruption or, in debug builds, to trigger a debug assert. This was due to the auto-increment column value being overwritten in the old records of the spatial index while the new record was prepared. (Bug #37189985)

- **InnoDB:** Certain IO buffer serializations triggered an assertion in debug builds that caused the system to hang. (Bug #37139618)

- **InnoDB:** Improved `InnoDB` start up time. (Bug #36880863)

  References: This issue is a regression of: Bug #36808732.

- **InnoDB:** An assertion failure was raised when creating a `FULLTEXT` index on a table with an `FTS_DOC_ID` value greater than 4294967295. (Bug #36879147)

  References: See also: Bug #37387224.

- **InnoDB:** Improved `can_older_trx_be_still_active()` so that it no longer uses `trx_sys->rw_trx_list`. (Bug #36729529)
- **InnoDB:** Dropping a primary key, and then adding a new `AUTO_INCREMENT` column as a primary key in descending order using the `INPLACE` algorithm failed.

  Our thanks to Shaohua Wang and the team at Alibaba for the contribution. (Bug #36658450)
- **InnoDB:** Removed a memory leak from the `mysqladmin` client. (Bug #36537941)
- **InnoDB:** Extending a user tablespace produces file extension redo log records (`MLOG_FILE_EXTEND`), but they were not produced when extending the system tablespace. (Bug #36511673)
- **InnoDB:** On Windows, fixed a doublewrite buffer regression that slowed file access, and refactored `FILE_FLAG_OVERLAPPED` flag usage for opening files. (Bug #36259487)
- **InnoDB:** A `DELETE` operation on a table with a self referential foreign key and full-text index could have triggered an assertion. (Bug #36234681)
- **InnoDB:** Removed code that handled obsolete redo log formats. (Bug #35020216)
- **InnoDB:** Common prefix compression for redo log inserts (`MLOG_REC_INSERT`) was disabled but is now enabled when the versions match. (Bug #34946626)

  References: This issue is a regression of: Bug #13899.
- **InnoDB:** Virtual column information for a row containing an externally stored `BLOB` was not always logged during an `UPDATE` operation, which sometimes resulted in an `Index PRIMARY is corrupted` error. (Bug #34574604)
- **InnoDB:** `ON DELETE CASCADE` with generated columns containing secondary indexes sometimes failed, due to virtual column templates not being initialized before deletion.

  Our thanks to Rahul Malik for the contribution. (Bug #33691659)
- **InnoDB:** An update operation attempted to update a virtual column while building an update node for a child table, but should not have done so since foreign key constraints cannot reference virtual columns. (Bug #33327093)
- **InnoDB:** If binary logging was disabled, it was possible for full-text search queries to be incorrect after a server crash.

  Our thanks to Yin Peng and the Tencent team for the contribution. (Bug #116212, Bug #37095383)
- **InnoDB:** Removed a duplicate declaration of the `dd_set_autoinc()` method. (Bug #116175, Bug #37089340)
- **InnoDB:** It was possible for `ALTER TABLE`, which rebuilds `InnoDB` tables using the `INPLACE` algorithm, to be rejected with a duplicate key error due to a non-duplicate record being inserted concurrently while the rebuild was paused to release a page latch.

  Our thanks to Dmitry Lenev and the team at Percona for contributing to this fix. (Bug #115511, Bug #36808088)
- **InnoDB:** The check enforcing the rule that `ALGORITHM=INSTANT` cannot be used on a column referenced by a foreign key constraint from another table did not inspect the last field of said constraint. (Bug #115457, Bug #36796094)

- **InnoDB:** CPU usage statistics did not account for a processor count over 128, which could degrade performance on these larger systems. (Bug #115399, Bug #36765223)

- **InnoDB:** Executing `ALTER TABLE` with `ADD COLUMN` or `DROP COLUMN` against an empty table now uses the `INPLACE` algorithm by default instead of `INSTANT`. This change means the row version is no longer incremented for these simple operations. (Bug #113051, Bug #36004394)

- **InnoDB:** An `ALTER TABLE` operation that rebuilt an `InnoDB` table using the `INPLACE` algorithm potentially led to losing a row of data if a purge occurred concurrently on the altered table that contained a clustered or spatial index.

  Our thanks to Dmitry Lenev and the team at Percona for contributing to this fix. (Bug #110706, Bug #113812, Bug #115608, Bug #116764, Bug #35303494, Bug #36261480, Bug #36846567, Bug #37318367)

- **InnoDB:** Queries with a descending primary key and the `index_merge` optimization sometimes yielded incorrect results such as missing rows. (Bug #106207, Bug #33767814)

- **Replication:** In an InnoDB ClusterSet setup, when `autocommit` was set to `OFF` on all nodes in the cluster, a controlled switchover using MySQL Shell was rejected with Error 1105 (`Unknown error`).

  To fix this, we now force a new transaction in `channel_change_source_connection_auto_failover()` whenever `autocommit=OFF` to prevent table access deadlocks when an info repository transaction is executed after changing `SOURCE_CONNECTION_AUTO_FAILOVER`. (Bug #37173907)

- **Replication:** While large transactions were being received and applied, and a request to stop the replication channel was made using `STOP REPLICA`, MySQL did not do so properly, and subsequently did not process any channel commands. In addition, the server shutdown process did not complete gracefully, and required either the MySQL process to be killed or the host system to be restarted. (Bug #115966, Bug #37008345)

- **Replication:** The log message written when a replica reconnects to the source (when, for example, it is stopped and restarted by issuing `STOP REPLICA` followed by `START REPLICA`) `While initializing dump thread for replica with UUID` *`uuid`*`, found a zombie dump thread with the same UUID. Source is killing the zombie dump thread(`*`thread_id`*`)` has been improved to `Upon reconnection with the replica, while initializing the dump thread for UUID` *`uuid`*`, an existing dump thread with the same UUID was detected. The source is terminating the previous dump thread (`*`thread_id`*`), which is normal and expected`. (Bug #84358, Bug #25330090)

- **Group Replication:** Removed a potential race condition between the internal functions `cs::apply::Commit_order_queue::front()` and `cs::apply::Commit_order_queue::remove()`. (Bug #37223451)

  References: See also: Bug #35206392.

- **Group Replication:** When the primary node unexpectedly left the group and quickly attempted to rejoin, the member which had been elected to remove other, faulty members tried to expel or remove the faulty node but could not do so due to lack of a majority. When the expelled or removed node was the primary, this left the cluster without a primary, resulting in an unusable state. (Bug #36991859)

  References: See also: Bug #37181867.

- **Group Replication:** In some cases, adding a new secondary caused existing secondaries to lag, leading to a deadlock which persisted with no more writes possible until the primary was restarted.

  This deadlock occurred between the ticket manager, which ensures that transactions are committed on the correct side of a view change (before or after the view change), and the commit order manager on the inbound replication channel, which ensures that transactions are committed in the same order in which they are received, when these two managers required different orders. This meant that, in some cases, adding a new secondary caused the group primary to be unable to do writes.

  We solve this issue by ignoring the commit order manager ordering and enforcing the ticket manager ordering for non-conflicting transactions when such a deadlock occurs. A consequence of this is that `replica_preserve_commit_order` may not be strictly honored near a `View_change_log_event`. In other words, `replica_preserve_commit_order` no longer provides a strict guarantee on an inbound channel on a Group Replication primary. `replica_preserve_commit_order` still ensures that transactions are ordered correctly, with the only exception being non-conflicting transactions around view change log events. (Bug #35206392)

  References: See also: Bug #37223451.

- **Group Replication:** Improved garbage collection in Group Replication by eliminating non-essential calls to `is_subset_not_equals()`. (Bug #110673, Bug #35286974)

- **Group Replication:** Removing a group member from a group in which all members were running the same version of MySQL, upgrading it to a later version (from a later release series), and then requesting it to rejoin the group caused the upgraded group member to hang in the recovering state.

- Improved an existing fix made in MySQL 9.1.0 for a condition which allowed queries with unknown columns in scope. (Bug #37341055)

  References: This issue is a regression of: Bug #35058815.

- Calling `mle_set_session_state()` with a type argument that was not a JSON string caused the component to exit. (Bug #37334566)

- A subquery which was marked for materialization during resolution was not optimized when an impossible condition was detected else where in the query. This created problems when optimizer was trying to detect subqueries that needed to be materialized. Since the subquery was not optimized, trying to access the query block's join pointer resulted in a server exit.

  We solve this by checking to see whether the subquery was optimized before finalizing the materialization transform for it. (Bug #37321762, Bug #37847157)

- Certain nested `SELECT` statements produced `ER_DUP_KEY` on a temporary table. (Bug #37309915)

  References: This issue is a regression of: Bug #115597, Bug #36846564.

- A query having a subquery which did not use any tables and which had a nonzero `OFFSET` clause did not return the correct result. For example, `SELECT (SELECT 1 LIMIT 1 OFFSET 10)` returned `1` instead of the expected `NULL`. (Bug #37293822)

- The errors `ER_DD_UPDATE_DATADIR_FLAG_FAIL`, `ER_IB_MSG_FIL_STATE_MOVED_PREV_OR_HAS_DATADIR`, `ER_RPL_KILL_OLD_DUMP_THREAD_ENCOUNTERED`, and `ER_RPL_MTA_ALLOW_COMMIT_OUT_OF_ORDER` were originally defined in MySQL 8.0, but were subsequently assigned different error code numbers (but with the same names) in MySQL 8.4. The numbers assigned in MySQL 8.0 now apply to MySQL 8.0 only; in the MySQL 8.4 and later release series, only the numbers assigned in MySQL 8.4 are used. (Bug #37284176)

- Added missing checks for `NULL` input arguments in `mle::validate()` and `validate_session_options()`. (Bug #37267887)
- For hash joins, time and resources were spent in unnecessary construction and destruction of `HashJoinChunk` objects that were never used, even in cases where a hash join could not be performed completely in memory and subsequently spilled to disk. (Bug #37235166)
- Adding an event to the `sys` schema during an upgrade led to an unplanned shutdown of the server. (Bug #37162611)
- In `sql/item_cmpfunc.cc`, `Item_bool_func2::resolve_type()` made an unchecked call to `Item_bool_func::resolve_type()`; the call to `Item_bool_func::resolve_type()` ignored its return value, and execution continued even in case of an error. (Bug #37143289)
- When evaluating an expression such as *`value`* `BETWEEN 'a' AND 'b'`, the server now checks the string literals and ensures that they use the same character set and collation as the *`value`* to be compared with them. (Bug #37086818)
- Removed an unnecessary assertion in `sql/item_func.cc`. (Bug #37083848)

  References: See also: Bug #29467577.
- Removed a memory leak found in `option_tracker/udfs.cc`. (Bug #37075241)
- AppArmor denied access to `/proc/$pid/task/$thread_id/mem`, a file required to generate a stack trace. (Bug #37063288)

  References: See also: Bug #37387034.
- Added a missing check for an empty pointer to `WalkAndReplace()`, in `sql/sql_resolver.cc`. (Bug #36987582)

  References: This issue is a regression of: Bug #112557, Bug #35855294.
- MySQL allowed outer references in window `PARTITION BY` and `ORDER BY` expressions, which was not in accordance with the SQL standard. (Bug #36921175, Bug #37847161)
- A filter condition in a subquery was sometimes ignored when a query used the `index_subquery` join type for subquery execution, and the subquery table used materialization in the execution plan. The derived table access path replaced the filter condition, resulting a final plan without the filter layer. To fix this, in such cases, we now add the derived table access path along with the filter access path instead of replacing the latter. (Bug #36918913)
- Transformations of some scalar subqueries to derived tables were not always performed correctly. (Bug #36902116)
- Some `UNION` operations similar to `a UNION b UNION c ...` consumed excessive memory. To help keep this from happening, we now flatten equal set operations at the parsing level, before contextualization occurs, which should result in reduced resource usage by such operations. (Bug #36652610)
- Improved the internal function `my_print_help()`. (Bug #36615714)

  References: See also: Bug #37387224.
- Removed incorrect code from `Acl_cache`. (Bug #36608160)
- A subquery containing an aggregate function `WITH ROLLUP` which was part of a row value comparator was not always processed correctly. (Bug #36593235)

References: See also: Bug #37387180. This issue is a regression of: Bug #30969045, Bug #30921780, Bug #26227613, Bug #29134467, Bug #30967158.

- It was possible for errors raised when persisting variables not to be reported correctly. (Bug #36574732)
- Some subqueries using `WITH ROLLUP` were not always processed correctly. (Bug #36421704)
- `MyISAM` block length calculations were not always performed correctly. (Bug #36347992)
- Removed a potential race condition. (Bug #35981769)

  References: See also: Bug #36608160.
- The `mysqldump --column-statistics` option is now disabled by default. (Bug #35209008)
- Fixed an issue relating to FTS and concurrent DDL or DML. (Bug #34633727)
- `DROP VIEW` *`name`* was rejected with `ER_BAD_TABLE_ERROR` if there existed a table with the same *`name`*. (Bug #33200087)
- Setting `explain_json_format_version` to 2 now has the following effects on the output of `EXPLAIN FORMAT=JSON`:

  - The output includes the JSON schema version in *`major`*.*`minor`* format; this is always displayed as `2.0` in MySQL 9.2.
  - Only query attributes are stored directly in the top-level object; in version 1 output, this object contains both query attributes and iterator attributes.

  The output of `EXPLAIN FORMAT=JSON` when `explain_json_format_version = 1` remains completely unchanged in this release. (Bug #116915, Bug #37372130, Bug #35239659)
- The `mysqlslap` utility did not disable SSL when the `--ssl-mode=disable` option was passed in. (Bug #116844, Bug #37353662)
- When taking the `UNION` of a `YEAR` column and a `BOOL` (`TINYINT`) column, the data type of the result was `TINYINT`, due to a flaw in the internal `field_types_merge_rules` array. We fix this by adjusting the result data type in such cases to `SMALLINT`. (Bug #116415, Bug #37192491)
- Incorrect results were returned by some queries that used hash antijoins when the hash table did not fit in the join buffer and spilled to disk. (The query triggering the issue actually specified `LEFT JOIN`, but this was transformed internally from a left outer join to an antijoin.)

  The problem was that some rows in the probe table were skipped when writing the probe rows to chunk files, the skipped rows being those that had `NULL` in part of the join key. Such rows can be skipped for inner joins and semijoins, as they are known to have no match in the build table, but for outer joins and antijoins, rows in the probe table which have no matching row in the build table should be part of the join result, so they must be included in the chunk files.

  We already preserved these rows in the chunk files for outer joins. This fix extends the logic used for that purpose so that it also applies for antijoins. (Bug #116334, Bug #37161583)
- A query containing a common table expression produced different results on the second and subsequent executions in some cases when used in a prepared statement or stored procedure. (Bug #116140, Bug #37074489)

- When the server `sql_mode` included `ANSI_QUOTES`, `SHOW GRANTS` quoted stored function and stored procedure names in backticks (`` ` ``), while using double quotes (`"`) with the names of other objects. Now double quotes are always used in such cases. (Bug #115953, Bug #37003502)

- Window functions having `BIT` values as arguments did not return `BIT`. (Bug #115597, Bug #36846564)

- The `mysqlslap` utility became unresponsive if `--auto-generate-sql-execute-number` and `--concurrency` were specified without `--auto-generate-sql` or `--query`. (Bug #113215, Bug #36048754)

- In MySQL 8.0 and later, queries of the form `SELECT DISTINCT ... FROM t1 WHERE NOT IN(SELECT ...)` were transformed into an antijoin if possible, causing the optimizer not to choose a group skip scan for table t1 whereas it would have been chosen in MySQL 5.7. This resulted in a performance degradation for such queries. Group skip scan is not chosen, since the query is now no longer a single-table query following the antijoin transformation, and this access method is enabled only for single table queries. The same behaviour can be seen for queries which are transformed into semijoins as well. In such cases, group skip scan access method can still be used if the access method is used only for duplicate removal (that is, with `DISTINCT` or `GROUP BY`, but without aggregate functions).

  To fix this, we enable group skip scan when there is only one table in the original query, irrespective of the number of semijoin tables present after internal transformations as long as the query contains no aggregate functions. (Bug #112362, Bug #35842412)

- The `mysql` client did not allow using '#' or '--' inside an optimizer hint comment.

  Our thanks to Kaiwang Chen for the contribution. (Bug #98521, Bug #30875669)

- The `mysqldump --compact` option now disables `--tz-utc`.

  Previously, with `--tz-utc` enabled (the default), `--compact` executed `SET TIME_ZONE='+00:00'` before reading data but did not write this statement to the dump file. This caused an inconsistency, since data was extracted using UTC but readers of the dump did not know that the data used this time zone. (Bug #58491, Bug #11765514)

- When comparing internal representations of `VECTOR` columns, their length properties were not considered.