MySQL Shell 9.7 Release Notes

Abstract

This document contains release notes for the changes in MySQL Shell 9.7.

For additional MySQL Shell documentation, see http://dev.mysql.com/.

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: 2026-04-21 (revision: 31081)

Table of Contents

Preface and Legal Notices

This document contains release notes for the changes in MySQL Shell 9.7.

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 Shell 9.7.0 (2026-04-21)

Note

These release notes were created with the assistance of MySQL HeatWave GenAI.

- AdminAPI Added or Changed Functionality
- AdminAPI Bugs Fixed
- Utilities Added or Changed Functionality
- Bugs Fixed

AdminAPI Added or Changed Functionality

- As of this release, `cluster.reset_recovery_accounts_password()` is deprecated. It is replaced by the `reset_replication_accounts_password()` function.

  `reset_recovery_accounts_password()` enables you to reset, or recreate, the credentials for the internal recovery and replication accounts and is now supported for Cluster, ClusterSet, and ReplicaSet deployments.

  See Resetting Internal Replication Account Credentials. (Bug #37977363, WL #16904)

AdminAPI Bugs Fixed

- In previous versions, AdminAPI created internal accounts without explicitly specifying the authentication plugin; the server's default authentication plugin was used. This default has changed between versions and plugins such as `mysql_native_password` have been removed entirely. As a result, some AdminAPI-managed topologies could contain accounts created with outdated or obsolete plugins, causing authentication failures when upgrading server instances.

As of this release, AdminAPI creates internal replication accounts with the `caching_sha2_password` authentication plugin, if the target instance is running MySQL 8.0.4, or higher. Also, `reset_replication_accounts_password()` can be used on existing topologies with the `recreate` option to recreate internal accounts with the `caching_sha2_password` authentication plugin. (Bug #39051958)

- If a target instance was configured to enforce a password policy stricter than the one used by AdminAPI, errors similar to the following were returned and the operation failed:

```
RuntimeError: Dba.create_cluster: Unable to generate a password that complies with active
MySQL server password policies for account accountName
```

The AdminAPI generated passwords using a fixed length of 32 characters. If `validate_password.length` was set to a value larger than 32, the error was returned.

As of this release, for MySQL 8.0.18 or higher, internal accounts are created using `IDENTIFIED BY RANDOM PASSWORD`, ensuring the generated password complies with the active password policy. For older versions, the existing password generation method is used, but the value of `validate_password.length` is retrieved, thereby ensuring the generated password conforms to the target's policy. (Bug #38728092)

- If a Read Replica, configured with the default `'replicationSources: PRIMARY'`, failed over to another source in a multi-primary cluster, `status()` incorrectly returned the warning `current sources don't match the configured ones`. This occurred because the check assumed only one PRIMARY was valid and flagged any other as a mismatch.

As of this release, all PRIMARY replicationSources are treated as valid across all primaries in multi-primary Clusters. (Bug #38598757)

Utilities Added or Changed Functionality

- MySQL Shell's copy and dump utilities now report accounts using the `mysql_native_password` authentication plugin as errors if ocimds is enabled for target versions 8.4 or higher.

A new compatibility option is added to the dump utilities, which is enabled by default for the copy utilities if the authentication plugin is enabled, `target_has_mysql_native_password`. This option makes it possible to dump or copy accounts from MySQL HeatWave Service DB Systems on which the `mysql_native_password` authentication plugin was enabled after they were created. (Bug #38852692)

Bugs Fixed

- Fixed an issue relating to the REST client. (Bug #39116948)
- MySQL Shell did not correctly handle JSON documents with large amounts of nesting. (Bug #39116933, Bug #39116954)

Changes in MySQL Shell 9.6.0 (2026-01-20)

Note

These release notes were created with the assistance of MySQL HeatWave GenAI.

AdminAPI Added or Changed Functionality

- In previous releases, the periodic updates to the `last_check_in` field of the `mysql_innodb_cluster_metadata.routers` table generated very large binary logs. This was caused by MySQL Router storing full configuration details in the `attributes` field of the `mysql_innodb_cluster_metadata.routers` table and the entire row being copied to the binary log every time `last_check_in` was updated.

  As of this release, a new version of the Metadata Schema is added, v2.4.0. In this version, `last_check_in` is moved to a new table, `mysql_innodb_cluster_metadata.router_stats`.

  If the schema version is less than 2.4.0, MySQL Router carries on using `mysql_innodb_cluster_metadata.routers.last_check_in`.
  If the schema version is 2.4.0, or higher, MySQL Router uses `mysql_innodb_cluster_metadata.router_stats.last_check_in`.

  

  Important

  You must manually upgrade your Metadata Schema with `dba.upgrade_metadata()`.

  The following commands now grant `INSERT` and `UPDATE ON` to MySQL Router accounts:

  - `setup_router_account()`: grants `INSERT` and `UPDATE ON` to new accounts.
  - `dba.upgrade_metadata()`: grants `INSERT` and `UPDATE ON` to existing accounts.

  (Bug #37926324, WL #17027, WL #17028)

Utilities Added or Changed Functionality

- The Binary Log Dumping and Loading utilities now use the new GTID library introduced in MySQL Server 9.6. (Bug #38667140, WL #16076)
- When running a dump with the `ocimds` option enabled, if a table is not using the InnoDB engine, an additional check is executed to verify if it is possible to change the table's engine to InnoDB. If this check fails, an error is reported, which requires a manual fix.

  As a result, the `ocimds:true` option and the `force_innodb` compatibility option now require the `CREATE TEMPORARY TABLES` privilege. (Bug #38650807)
- The Upgrade Checker utility's queries have been optimized for performance. (Bug #38628445)
- When loading a dump with the `'handleGrantErrors': 'ignore'` option set, if a GRANT statement with multiple roles/privileges fails, the load utility will apply the GRANTs one by one, and inform about the success or failure of each. (Bug #38624926)

- The dump utilities can now check for the deprecated authentication plugin, authentication_fido. (Bug #38550124)
- When dumping with the consistent option enabled, if consistency cannot be achieved due to missing privileges and cannot be verified, an additional message is displayed explaining how to create a consistent dump. (Bug #38452568)
- The following MySQL object filtering options are added to the Upgrade Checker utility:
  - `includeSchemas` and `excludeSchemas`
  - `includeTables` and `excludeTables`
  - `includeRoutines` and `excludeRoutines`
  - `includeTriggers` and `excludeTriggers`
  - `includeEvents` and `excludeEvents`
  - `includeUsers` and `excludeUsers`

  See Upgrade Checker Utility. (WL #17172)

Utilities Bugs Fixed

- If run against an instance containing a table with a spatial index, the Upgrade Checker utility's output contained an unescaped `target_version` placeholder. (Bug #38748787)
- If run in JavaScript mode, with --json=raw, MySQL Shell could close unexpectedly while running a dump or load operation. Errors similar to the following were returned:

  ```
  malloc(): unaligned tcache chunk detected
  ```

  (Bug #38645196)
- When resuming a load, if some of the tables were bulk loaded in a previous run, the summary printed at the end was incorrect. The summary listed the total number of bulk load-compatible tables in the dump, not the number of tables which were loaded. (Bug #38644211)
- Gathering the list of views from an instance without the `information_schema.view_table_usage` table was slow when the number of views was high. (Bug #38629024)
- Upgrade Checker utility did not report an unsupported upgrade if the output format was set to JSON. (Bug #38571667)
- When resuming a load with the `'dropExistingObjects'` option set to true, if the user executing the load lacked the `ALLOW_NONEXISTENT_DEFINER` privilege, the operation could fail. Errors were returned similar to the following:

  ```
  While dropping the account 'root'@'localhost': MySQL Error 4006
  (HY000): Operation DROP USER failed for 'root'@'localhost' as it is
  referenced as a definer account in a view.: DROP USER IF EXISTS
  'root'@'localhost'
  ```

  Previously, when resuming the load, user accounts were always dropped, unlike other DDL objects whose status is tracked in the progress file, regardless of whether they had been created in the previous run.

As of this release, creation of user accounts is logged in the progress file. If the load is resumed, accounts that were previously created are now dropped, if `'dropExistingObjects'` is enabled, and are not created again. (Bug #38566495)

- Dumping a MariaDB table that had an explicitly specified collation and contained a string column without a collation clause failed with the following error:

```
 Failed to replace default collation of table `sakila`.`ptable` from
'utf8mb4_uca1400_ai_ci' to 'utf8mb4_0900_ai_ci'.
```

  When the dump utility attempted to replace unsupported collations with their closest counterparts, it expected to find the collation specified twice, once for the column and once for the table. As only one collation clause was present, it reported an error when the second occurrence was missing. (Bug #38560511)

- If a 5.7 instance contained orphaned objects, the Upgrade Checker utility's syntax check failed with an unknown database error.

  As of this release, the orphaned objects check is moved to the start of the process and any orphaned object discovered is automatically excluded from further checks. (Bug #38555376)

- When loading a dump into a server with the global system variable `sql_generate_invisible_primary_key` enabled, if the account lacked privileges to disable this variable at the session level, the operation failed with an Access denied error during session setup.

  As of this release, an error is displayed explaining the issue. (Bug #38520826)

- When loading a MySQL 5.7 dump, if the target is 8.0, or when the `ignoreVersion` option is enabled, the loader now automatically removes all SQL modes which are not available in 8.0, or higher. The following modes are now removed:

  - `POSTGRESQL`
  - `ORACLE`
  - `MSSQL`
  - `DB2`
  - `MAXDB`
  - `NO_KEY_OPTIONS`
  - `NO_TABLE_OPTIONS`
  - `NO_FIELD_OPTIONS`
  - `MYSQL323`
  - `MYSQL40`
  - `NO_AUTO_CREATE_USER`

  (Bug #38499424)

- MySQL Shell would not dump a table completely if it had a composite key, with an ENUM column which was not the first in the key definition, and the key values were not listed alphabetically. (Bug #38475999)

- Under certain circumstances, the Upgrade Checker utility reported obsolete SQL modes as NOTICE instead of ERROR. For example, NO_AUTO_CREATE_USER.

  As of this release, the obsoleteSqlModeFlags check behaves in the following way:

  - NOTICE: if the flag is not persisted in the configuration file, which implies the value will be fixed during the upgrade.
  - WARNING: if no configuration file is provided to the utility, and it is not possible to determine if the issue represents a problem or not.
  - ERROR: if the configuration file is provided and the obsolete flag is defined there.

  (Bug #38424168)

Functionality Added or Changed

- Option Tracker now supports MySQL Shell.

  See Option Tracker Component. (WL #16659, WL #17000)

Bugs Fixed

- It was not possible to run commands containing Not Equal operators over X Protocol. Errors similar to the following were returned:

  ```
  ValueError: Insufficient number of values for placeholders in query
  ```

  Additionally, queries such as the following returned the error regardless of the protocol used:

  ```
  session.run_sql('select 1 != ?', [2])
  ```

  (Bug #38661681)

Changes in MySQL Shell 9.5.0 (2025-10-21)

Note

These release notes were created with the assistance of MySQL HeatWave GenAI.

Utilities Added or Changed Functionality

- Upgrade check names are now grouped by categories: `schema`, `routine`, `accounts`, and `config`. (Bug #38274934)
- MySQL Shell dump utilities now support the CSV dialect, conforming to RFC4180, `'csv-rfc-unix'`.
  - `fieldsTerminatedBy: ,`

  - `fieldsEnclosedBy: "`
  - `fieldsOptionallyEnclosed: true`
  - `fieldsEscapedBy: not set`
  - `linesTerminatedBy: <LF>`

  (Bug #38245434)

- Added the `convert_invalid_accounts` compatibility option to the dump utilities. This option converts accounts with unsupported authentication plugins to `caching_sha2_password` and locks the account. (Bug #38237729)

Utilities Bugs Fixed

- MySQL Shell could close unexpectedly when running the upgrade checker utility on a MySQL Server 5.7. The error occurred on the `invalidEngineForeignKey` upgrade check if a table name contained non-letter characters such as hyphens. (Bug #38442101)
- When resuming an interrupted load, duplicate entries could be present in the progress file, affecting byte skipping in subsequent resumes. (Bug #38408207)
- When using anonymous arguments with the `-- util copyTables` command, the error message did not clearly indicate the correct usage of the options. Running the command without options now displays a clear syntax message to help avoid ambiguity. (Bug #38377905)
- When dumping with `ocimds: true`, schemas and tables with `ENCRYPTION` disabled were updated and reported as modified, creating logging noise, even though MySQL HeatWave Service disallows enabling the statement and does not generate errors if the `CREATE` statement explicitly disables it.

  `CREATE SCHEMA` or `TABLE` statements with disabled `ENCRYPTION` option are no longer modified or reported.

  Also, roles which already exist in MySQL HeatWave Service are no longer reported as warnings. (Bug #38264847)
- Using the `loadDump()` utility with the `dropExistingObjects` option enabled resulted in an error during data loading if a loading schema did not exist. Errors were returned similar to the following:

  ```
  ERROR: Error executing SQL statement for schema `schemaName`:
  Unknown database 'schemaName'
  ```

  (Bug #38249362)
- If a table had both a unique and a non-unique key on the same referenced column, the upgrade checker utility's foreign key check could raise a false positive error. (Bug #38194922)
- Dumping from a database other than MySQL, with binary logging enabled, could fail when `SHOW MASTER STATUS` was executed, due to the absence of an expected fifth column, `Executed_Gtid_Set`. Errors similar to the following were returned:

  ```
  get_string(4): index out of bounds (ArgumentError)
  ```

  As of this release, MySQL Shell accesses this column only if it is available. (Bug #38144597)

- The copyInstance() utility failed when copying from server version 9.3.0 with MySQL Shell 9.3.1. Error similar to the following was returned:

```
Requested MySQL version '9.3.1' is newer than the maximum version
'9.3.0' supported by this version of MySQL Shell (ArgumentError)
```

  (Bug #38107377)

- Loading a dump using the collation `utf8mb4_uca1400_ai_ci` to MySQL failed with `Unknown collation: 'utf8mb4_uca1400_ai_ci'`. `utf8mb4_uca1400_ai_ci` is not available in MySQL.

  As of this release, the dump utilities detect unsupported collations and replace them with the nearest compatible collation available in MySQL. A warning is displayed for each collation replaced. (Bug #38089433)

- Loading a dump with the schema option and checksum verification enabled failed with a logic error. Errors were returned similar to the following:

```
Unable to find schema test whose existence is being checked
(LogicError)
```

  (Bug #38034277)

- If loading a dump failed, due to duplicate database objects, a second attempt would succeed even though it should have failed for the same reason. This was caused by the presence of a progress file, which the second attempt saw as evidence the load was valid.

  As of this release, the progress file is not created until just before loading begins. (Bug #37326937)

Functionality Added or Changed

- Shell plugins now support standard shell print formatting, enabling consistent output for information such as errors or warnings through the `shell.print()` function. (Bug #38397677)

Bugs Fixed

- It is no longer possible to override the language used in `\source` commands unless they are run interactively. (Bug #38182270)

- It was not possible to establish a connection to MySQL 5.6 or older. The connection required session tracker support, which is not available on these servers. Errors were returned similar to the following:

```
Session tracker not supported by server
```

  (Bug #38144635)

- Boolean command line options were not handled consistently. (Bug #38062258)

- Tab-completion did not complete filenames for the `\source` command. (Bug #37345664)

- The `--show-warnings` option returned an error that it required an argument:

```
option --show-warnings requires an argument
```

  (Bug #34878119)

Changes in MySQL Shell 9.4.0 (2025-07-22)

Note

These release notes were created with the assistance of MySQL HeatWave GenAI.

Deprecation and Removal Notes

- The following, deprecated in previous releases, were removed in this release:
  - `-mc`
  - `-ma`
  - `-mx`
  - `--dba=enableXProtocol`
  - `--dba-log-sql`
  - `shell.options.outputFormat`
  - `.routingOptions()`
  - `addInstances` and `removeInstances` were removed from *`Cluster`*`.rescan()`

  (WL #16922)

AdminAPI Added or Changed Functionality

- `dba.deploy_sandbox_instance()` is extended with the option `mysqldPath`. This option enables you to specify the path to the `mysqld` binary, or MySQL installation root, to use as the seed instance for your sandbox deployment. (Bug #25423030)

AdminAPI Bugs Fixed

- It was not possible to import a Routing Guideline with `import_routing_guideline()` if the guideline's name was already used by an existing guideline on the target topology. Using the `force` option overwrote the existing guideline with the imported guideline.

  As of this release, the `rename` option is added to `import_routing_guideline()`, enabling you to define an alternate name for the imported guideline, which is used in the event of a naming clash. (Bug #37750932)

- *RoutingGuideline*.show({"router": *routerId*}) failed if the guideline was defined on a ClusterSet. (Bug #37739359)

Utilities Added or Changed Functionality

- Partitioned tables without a primary key are now reported as errors when running a dump with the create_invisible_pks compatibility option set, as all columns used in the partitioning expression must be part of every unique key. Errors are returned similar to the following:

```
ERROR: Table `s`.`t` does not have a
  Primary Key, this cannot be fixed automatically because the
  table is partitioned (this issue needs to be fixed manually)
```

  (Bug #37892879)

- The Dump and Load utilities' ociAuth option now supports instance_obo_user. (Bug #37781203)

- It is now possible to disable collection of host-specific diagnostics information, with the collectDiagnostics utility's hostInfo option.

  hostInfo option defaults to true. To disable, set to false. (Bug #36391259)

Utilities Bugs Fixed

- util.dumpInstance returned the following error when run against MySQL HeatWave DB Systems:

```
ERROR: User 'admin'@'%' is granted restricted privilege: OPTION_TRACKER_OBSERVER
(fix this with 'strip_restricted_grants' compatibility option)
```

  The list of privileges has been updated to include OPTION_TRACKER_OBSERVER. (Bug #37958876)

- It was not possible to dump data from a MySQL HeatWave DB System running MySQL 8.4, if the Hypergraph Optimizer was enabled. Errors were returned similar to the following:

```
MySQL Error 3999 (42000): The hypergraph optimizer does
not yet support 'EXPLAIN with TRADITIONAL format'
```

  As of this release, queries which chunk or fetch data from MySQL HeatWave-enabled tables now use an optimizer hint which disables offloading to MySQL HeatWave. This also improves dump performance. (Bug #37904121)

- The load dump utility failed if the target MySQL instance did not have Performance Schema enabled. An error similar to the following was returned:

```
ERROR: [Worker006]: Error opening connection to MySQL: MySQL Error 1683 (HY000):
'ps_current_thread_id': The Performance Schema is not enabled.
```

  As of this release, if the call to ps_current_thread_id() fails for any reason, it is logged, but does not cause the load to stop. (Bug #37867455)

  References: See also: Bug #36197620.

- A fix introduced in 9.1.0 enabled the dump utilities to compare the server version to the MySQL Shell version and return an error if the server's minor version was greater than MySQL Shell's.

  As of this release, an error is returned only if the server has a higher major version. If the minor version is greater, a warning is logged. (Bug #37866205)

References: See also: Bug #36701854.

- MySQL Shell logging has been improved for `LOAD DATA` warnings generated by util.importTable and util.loadDump. Previously, `LOAD DATA` warnings were printed to the terminal and were difficult to locate in the log. For example:

```
Warning: schema@table@123.tsv.zst error 1062: Duplicate entry
'1234567' for key 'table.PRIMARY'
```

  These messages have been improved, making them easier to find in the log. For example:

```
Warning: An error has been reported while loading data into
`schema`.`table` from 'schema@table@123.tsv.zst' file, error 1062:
Duplicate entry '1234567' for key 'table.PRIMARY'
```

  (Bug #37800574)

- The MySQL REST Service-specific account, `ocirest`, is automatically excluded when dumping with `ocmids:true`, or loading into MySQL HeatWave DB Systems. (Bug #37792183)

- If util.copyInstance or util.loadDump connected using an account without the `ALLOW_NONEXISTENT_DEFINER` privilege, and tried to copy a non-view DDL object with a DEFINER clause set to an account included in the dump, an error was returned similar to the following:

```
Error processing schema `<schema>`: Access denied; you need
(at least one of) the SUPER or ALLOW_NONEXISTENT_DEFINER privilege(s)
for this operation
```

  As of this release, the accounts are created before any other DDL objects and grants are applied before view placeholders are replaced with views. (Bug #37669785)

- The Upgrade Checker utility returned a false positive for partial keys in schemas referencing full keys in other schemas. (Bug #37651453)

- `util.copySchemas()` could hang if a dump was loaded or copied using an account which lacked the `SELECT` privilege on the Performance Schema. The operation hung when indexes were rebuilt. Errors were returned similar to the following:

```
MySQL Error 1142 (42000): SELECT command denied to user for table 'events_stages_current'
```

  As of this release, error handling is added for this issue. (Bug #37593239)

Functionality Added or Changed

- MySQL Shell session objects now support storing client data, providing an alternative to using global dictionaries tied to session IDs, and improving management of session-specific data for plugins written in Python or JS. Several other improvements were made to session objects as part of this enhancement.

  The following changes were made:

  - `setClientData()` and `getClientData()` methods added to the classic and X session objects.
  - Added a `trackSystemVariable()` method to the `ClassicSession` object to allow tracking `SQL_MODE`.
  - Added a `sqlMode` property to the `ClassicSession` object.

  (Bug #37710803)

- Shell now supports the storage of generic secrets. You can use the following new functions to manage your secrets:
  - `shell.storeSecret(key, value)`: Stores a secret with the given key.
  - `shell.readSecret(key)`: Reads a secret with the given key.
  - `shell.deleteSecret(key)`: Deletes a secret with the given key.
  - `shell.deleteAllSecrets()`: Deletes all secrets
  - `shell.listSecrets()`: Lists keys of all secrets

  

  Note

  Secrets stored with this API are not accessible to the credential methods listed here: Working with Credentials, nor can this API cannot access credentials managed by that API.

  See Generic Secret Storage for more information. (WL #16958)

Bugs Fixed

- Custom SQL Handlers did not correctly handle statements preceded by a full line comment. A syntax error was returned, similar to the following:

  ```
  mysqlsh.DBError: MySQL Error (1064): You have an error in your SQL syntax;
  check the manual that corresponds to your MySQL server version for the right
  syntax to use near 'yourStatement' at line 2
  ```

  (Bug #37979837)

- MySQL Shell now supports the `--local-infile` command-line parameter, similarly to the `mysql` client.

  See mysqlsh — The MySQL Shell for more information. (Bug #37960814)

- Kerberos authentication is now supported on macOS. The `authentication_kerberos_client` plugin is now bundled on macOS installations of MySQL Shell. (Bug #37777584)

- MySQL Shell could crash when dumping a table with a functional, or virtual, index.

  The following error was returned:

  ```
  Segmentation fault (core dumped)
  ```

  (Bug #37770454)

- Using placeholders in SQL queries, such as `CREATE REST ... OPTIONS { "cache_ttl": ? }` with parameters, could return a parse error. (Bug #37196079)

- MySQL Shell read all profiles in an AWS config file, including those which were not properly prefixed with `profile`. Error messages were returned similar to the following:

  ```
  Could not obtain credentials to assume role using profile 'default';:
  Partial AWS credentials found in config file
  (/path/.aws/config, profile:
    nameOfProfile),
  ```

```
missing the value of 'aws_secret_access_key' (RuntimeError);.
```

As of this release, MySQL Shell correctly handles AWS configuration profiles by requiring the `profile` prefix, ensuring that only valid profiles are read. (Bug #36916939)

- The `runSql()` method in the X session did not support `!` placeholders. Errors were returned similar to the following:

```
        MySQL Error (5015): Session.run_sql: Too many arguments
```

(Bug #34715428)

- Connections to invalid SSH URIs were not properly closed, resulting in multiple defunct sshd processes. As of this release, tunnel handling is improved and unused tunnels are closed automatically. (Bug #33564687)

- It was not possible to start MySQL Shell on ARM platforms if `PAGE_SIZE` was set to 64K. An error similar to the following was displayed:

```
mysqlsh: error while loading shared libraries:
libssh.so.4 ELF load command aligment not page-aligned
```

(Bug #118021, Bug #37854467)

Changes in MySQL Shell 9.3.0 (2025-04-15)

AdminAPI Bugs Fixed

- Removing a Replica Cluster from a ClusterSet using *`ClusterSet`*`.remove_cluster()` with `dissolve:false` failed with a query-related error similar to the following if the ClusterSet contained at least one Routing Guideline:

```
ERROR: Error removing Replica Cluster: MySQL Error 1451: Failed to execute query on
Metadata server serverName:portNumber:
Cannot delete or update a parent row: a foreign key constraint fails
.....
NOTE: Reverting changes...
Segmentation fault (core dumped)
```

As of this release, all Routing Guidelines affected are reassigned to the target Cluster instead of the ClusterSet. If an active guideline is present, it is disabled. A warning is printed for each reassignment, indicating the guideline may no longer be suitable for a standalone Cluster. (Bug #37480018)

- The AdminAPI required all members of an InnoDB Cluster to be ONLINE before allowing a switchover. Group Replication only requires that the group has quorum.

As of this release, the precondition checks in *`Cluster`*`.set_primary_instance()` require quorum only, instead of requiring that all members are ONLINE.

Note

`Cluster``.switch_to_multi_primary_mode()` and *`Cluster`*`.switch_to_single_primary_mode()` remain unchanged because `auto_increment_*` values must be updated on all cluster members, requiring all members are ONLINE.

(Bug #37399749)

Utilities Added or Changed Functionality

- If a prefix PAR does not end in a slash (/), an error similar to the following is returned:

```
Invalid PAR, expected:
https://namespace.objectstorage.region.oci.customer-oci.com/p/secret/n/namespace/b/bucket/o/[prefix/
```

  As of this release, if the slash is not present, the dump and load utilities add it automatically. (Bug #37629756)

- It is now possible to define the Oracle Cloud Infrastructure Object Storage connection details using the oci-cli environment variables.

  For information on these variables, see CLI Environment Variables.

  - `OCI_CLI_CONFIG_FILE`
  - `OCI_CLI_PROFILE`
  - `OCI_CLI_AUTH`
  - Using API Key-based authentication:
    - `OCI_CLI_USER`
    - `OCI_CLI_REGION`
    - `OCI_CLI_FINGERPRINT`
    - `OCI_CLI_KEY_CONTENT`
    - `OCI_CLI_KEY_FILE`
    - `OCI_CLI_PASSPHRASE`
    - `OCI_CLI_TENANCY`
  - Session Token-based authentication:
    - `OCI_CLI_REGION`
    - `OCI_CLI_KEY_FILE`
    - `OCI_CLI_PASSPHRASE`

  - `OCI_CLI_TENANCY`
  - `OCI_CLI_SECURITY_TOKEN_FILE`

  (Bug #37592840)

- The `mysql_option_tracker_persister` role is now excluded from the dump when `ocimds: true` and is excluded when loading a dump into a MySQL HeatWave DB System. (Bug #37457569)
- The Upgrade Checker now reports potential corruption of spatial indices in tables. It is possible, when updating a geometry field, to corrupt the spatial index, resulting in rows becoming invisible or even cause a crash when deleted and replaced with new records. The Upgrade Checker now warns about possible corruption on every table containing a spatial index.

  This check reports only if one of the following conditions is met:

  - Server version >= 8.0.3
  - Server ersion < 8.0.41 (on 8.0 series only)
  - Server version < 8.4.4 (on 8.4 series only)
  - Server version < 9.2.0 (on 9 series only)
  - Target version >= 8.0.41 (on 8.0 series only)
  - Target version >= 8.4.4 (on 8.4 series only)
  - Target version >= 9.2.0 (on 9 series only)

  (Bug #37360136)

- A timeout option is added to the Upgrade Checker utility, `checkTimeout: n`. The timeout value (seconds) applies to each check performed by the utility. If the timeout is reached while a check is being performed, the check is cancelled and the utility starts the next check and resets the timeout.

  See Running the Utility. (Bug #111769, Bug #35608739, WL #16757)

- MySQL Shell dump, load, and copy utilties now support libraries. The following were added:

  - `libraries`: (boolean) enables or disables the inclusion of libraries. Default is true.

    Added to `dumpInstance()`, `dumpSchemas()`, `copyInstance()`, and `copySchemas()`.

  - `includeLibraries`: (array of strings) List of libraries to include.

    Added to `dumpInstance()`, `dumpSchemas()`, `copyInstance()`, `copySchemas()`, and `loadDump()`.

  - `excludeLibraries`: (array of strings) List of libraries to exclude.

    Added to `dumpInstance()`, `dumpSchemas()`, `copyInstance()`, `copySchemas()`, and `loadDump()`.

  See MySQL Shell Utilities.

  See Using JavaScript Libraries for information on library support. (WL #16731)

Utilities Bugs Fixed

- Dump and load operations failed randomly due to a bug in the bundled Curl package.

  Curl is upgraded to 8.12.1 in this release. (Bug #37576066)

- The Upgrade Checker did not display a warning for the deprecated plugins `rpl_semi_sync_master` and `rpl_semi_sync_slave`. (Bug #37516017)

- MySQL Shell was updated for compatibility with the privilege changes made in MySQL HeatWave Service.

  The following privileges were added to MySQL HeatWave Service:

  - `FLUSH_PRIVILEGES`
  - `TRANSACTION_GTID_TAG`

  (Bug #37513534)

- The Upgrade Checker failed to check privileges if `SQL_MODE` contained `NO_BACKSLASH_ESCAPES`.

  The following error was returned:

  ```
  error: Unable to check permissions: You have an error in your SQL syntax; check the manual that
  corresponds to your MySQL server version for the right syntax to use near
  'root\'@\'localhost\'' LIMIT 1' at line 1
  Util.checkForServerUpgrade:
  The upgrade check needs to be performed by user with PROCESS, and SELECT privileges. (RuntimeError)
  ```

  (Bug #37484875)

- It is no longer necessary to set `ocimds:true` for a dump from a MySQL HeatWave Service DB System which will be imported into another MySQL HeatWave Service DB System. (Bug #37419320)

- Dump and load utilities did not retry transfers to HTTP storage if an error occurred. They also did not report HTTP errors when reading from HTTP storage. This also affected PARs.

  As of this release, transfers are retried and HTTP errors are reported. (Bug #37408073)

- `util.importTable()`, importing from a single file, could close unexpectedly if one of the background threads encountered an error. An error similar to the following was displayed:

  ```
          libc++abi: terminating
          stack trace
  ```

  (Bug #37408046)

Functionality Added or Changed

- The connection option `plugin-authentication-webauthn-device` is added in this release. This option enables you to specify the libfido2 device to use. Default is 0 (first device). (WL #16770)

Bugs Fixed

- Under certain circumstances MySQL Shell could crash during auto-completion on options such as `shell.options.history.auto`. (Bug #37528585)

- The SQL code completion for several statements has been corrected to remove the extra `CREATE_SYMBOL` from the grammar. This issue resulted in incorrect suggestions when using SQL code completion. (Bug #37393439)
- MySQL Shell did not honor Ctrl+c interrupts when running stored procedures, such as `sys.heatwave_load`. The procedure continued to run despite multiple Ctrl+C attempts, and displayed a `"Result printing interrupted"` message after completion. This issue was specific to stored procedures. (Bug #37345487)
- If the query_helper was used with trigger filters, MySQL Shell closed unexpectedly. (Bug #37207914)

Changes in MySQL Shell 9.2.0 (2025-01-21)

AdminAPI Added or Changed Functionality

- This release introduces Routing Guidelines, a flexible and unified configuration interface enabling users to customize routing behavior. Routing Guidelines are defined as a JSON document, stored in the metadata schema.

  A Routing Guideline document classifies all destinations in the topology and enables you to define strict rules on what type of client session is connected to what member of the topology.

  A new class, `RoutingGuideline` is added to the AdminAPI. See the `AdminAPI` section of the MySQL Shell JavaScript API or the MySQL Shell Python API.

  The following methods were added to the Cluster, ClusterSet, and ReplicaSet classes:

  - `create_routing_guideline(name[, json[, options]])`
  - `get_routing_guideline([name])`
  - `routing_guidelines()`
  - `remove_routing_guideline(name)`
  - `import_routing_guideline(file, [options])`

  The following methods were updated:

  - `set_routing_option()`
  - `routing_options()`
  - `router_options()`
  - `list_routers()`

See Routing Guidelines. (WL #14022, WL #14023)

AdminAPI Bugs Fixed

- `replicationLag` of *`ReplicaSet`*`.status()` returned `null` if replication was idle.

  As of this release, the following changes were made:

  - `replicationLag` is set to `null` if the replication connection, or SQL thread, is not running.
  - `replicationLag` is set to `applier_queue_applied` when the last queued transaction matches the last applied transaction, or the applying transaction count is 0 (zero).

  (Bug #35914505)

- It was not possible to remove a member from a ReplicaSet using *`ReplicaSet`*`.removeInstance()` with the `force` option enabled, if the member was in an ERROR state. (Bug #35282392)
- The ReplicaSet metadata schema table, `INSTANCES`, was not populated properly when upgrading the metadata schema to version 2.2.0. As a result, MySQL Router did not recognize the topology and rejected connections to it. (Bug #116231, Bug #37101286)

Utilities Added or Changed Functionality

- As of this release, MySQL Shell no longer supports Instance Metadata Service (IMDS) v1. (Bug #37380810)
- The schema `MYSQL_AUTOPILOT` is excluded by dump and load operations with `ocimds:true`. (Bug #37278169)
- As of this release, the upgrade checker utility writes all compatibility issues and fixes to the log file, instead of only writing to the console. (Bug #37154456)
- The `util.checkForServerUpgrade` check, `routineSyntax` is renamed to `syntax` and now uses the SQL version of the target server for the syntax checks.

  See Utility Checks. (Bug #36589651, Bug #35640118, WL #16236)

- It is now possible to dump and load binary logs using the following new utilities:

  - `util.dumpBinlogs (outputUrl, {options})`: Dumps binary logs generated since a specific point in time to the given local or remote directory.
  - `util.loadBinlogs (outputUrl, {options})`: Loads binary log dumps created by MySQL Shell from a local or remote directory.

  See MySQL Shell Utilities. (WL #15977)

Utilities Bugs Fixed

- If `convertBsonTypes` was enabled, the JSON import utility failed when importing negative BSON values.

  An error similar to the following was returned:

  ```
  ValueError: Unexpected data, expected to find an integer string processing extended JSON for $number
  ```

(Bug #37243264)

- Amazon RDS made the following changes in 8.0.36:

  - It is no longer possible to grant `BACKUP_ADMIN`.
  - It is no longer possible to lock MySQL system tables.

  As a result, consistent dumps from Amazon RDS are not possible.

  As of this release, if locking the MySQL system tables fails with an Access Denied error, a warning is printed and the dump continues.

  Additionally, for non-Amazon RDS instances, if the user has the `BACKUP_ADMIN` privilege and `LOCK INSTANCE FOR BACKUP` succeeds, the MySQL system tables are not locked. (Bug #37226153)

- Under certain circumstances, such as a large amount of data chunking, the `maxRate` dump option did not properly limit the throughput due to a gap between the start of the dump and the start of the data dump. As of this release, `maxRate` is used only when data is being dumped. (Bug #37216767)

- MySQL Shell could hang when running a dump with `consistent: true` under an account which lacked privileges to execute `FLUSH TABLES WITH READ LOCK`.

  As of this release, query events are checked only if they contain data, and `GRANT` and `REVOKE` statements are flagged as unsafe. (Bug #37158908)

- The upgrade checker utility returned a false positive for foreign keys defined on tables in the Cluster metadata schema. (Bug #36975599)

- The `importTable` utility generated the following error if run over an X Protocol connection:

  ```
  Util.importTable: A classic protocol session is required to perform
  this operation. (RuntimeError)
  ```

  As of this release, similarly to the other dump, export, load, and copy utilities, `importTable` creates a classic connection for the operation. (Bug #34582616)

Functionality Added or Changed

- As of this release, the Google V8 JavaScript engine is replaced by Oracle GraalVM. (Bug #34370637)
- As of this release, the default value of `history.autoSave` is changed to `true`, enabling the command history by default. (Bug #31746532)

Bugs Fixed

- In MySQL Shell 8.0.40, RPM installation failed on Oracle Linux 8 due to a dependency on Python 3.9. As of this release, MySQL Shell bundles Python 3.13. (Bug #37479400)
- If a Windows account name contained non-ASCII characters, MySQL Shell could not load Python plugins because the file path was not correctly encoded. "No such file or directory" error was returned. (Bug #37105233)
- Under certain circumstances, a Ctrl+c key combination could lead to a deadlock. (Bug #35998250)

Changes in MySQL Shell 9.1.0 (2024-10-15)

AdminAPI Added or Changed Functionality

- AdminAPI now performs version compatibility checks in operations which create or modify replication channels.

  See Replication Compatibility Checks. (WL #16408)

AdminAPI Bugs Fixed

- Running a `rescan()` operation on a Replica Cluster, could result in a warning that `group_replication_view_change_uuid` is required, but not configured. This occurred in a mixed version ClusterSet, where the primary Cluster is on version 8.3.0, or higher, and the replicas were on versions lower than 8.3.0. The check for `group_replication_view_change_uuid` was incorrectly performed on the primary Cluster instead of the target Replica Cluster. (Bug #36657936)

- If clone-based recovery failed while adding an instance, the instance and Cluster could be left in an inconsistent state, returning multiple errors, and unrecoverable by `rescan()`. It was not possible to remove the instance with `remove_instance()`, either.

  `add_instance()` did not properly handle clone-related errors and did not stop Group Replication if errors were detected.

  As of this release, `add_instance()` handles clone-related errors, reverts the state of the target instance and stops Group Replication on that instance. (Bug #36657628)

- Running AdminAPI operations against MySQL 5.7 instances resulted in an error, due to an attempt to run a query on a non-existent Performance Schema table.

  As of this release, error handling is included to account for such issues. (Bug #36652642)

- It was possible to define duplicate instance definitions in the `replicationSources` list of the *`Cluster`*`.add_replica_instance()` operation. The operation failed with an error.

  As of this release, the `replicationSources` list is validated for such duplicates in all operations which use it. The operation fails with an informative error. (Bug #36614218)

- AdminAPI commands which add or rejoin instances to Clusters using Clone did not check the existing members for compatibility. If no compatible members were available, Group Replication fell back to incremental recovery which can fail if Clone was the only supported provisioning method. An unexpected and unhelpful error was returned.

  As of this release, the following commands have been updated to check compatibility and provide appropriate feedback:

  - *`ReplicaSet`*`.add_instance()`

- *ReplicaSet*.rejoin_instance()
- *Cluster*.add_replica_instance()
- *Cluster*.rejoin_instance()
- *ClusterSet*.create_replica_cluster()

For a donor to be considered compatible for a recipient:

- They must be running on the same operating system.
- They must be running on the same platform.
- The versions must be compatible:
  - Both must be version 8.0.17, or higher.
  - If both are version 8.0.37, or higher, only their major and minor versions need match. For example, 8.4.0 and 8.4.3.
  - For versions 8.0.17, or higher, and lower than 8.0.37, the major, minor, and patch numbers must match.

(Bug #36054619, Bug #36682741)

- The prerequisite checks for *Cluster*.add_replica_instance() did not check the minor and patch version number of the metadata schema. Read Replicas require a minimum metadata schema version of 2.2.0.

  As of this release, *Cluster*.add_replica_instance() checks the metadata schema version and displays an error if the metadata schema version is lower than 2.2.0.

  The prerequisite checks for create_cluster() and create_replica_set() are also updated. (Bug #35595811)

Utilities Added or Changed Functionality

- As of this release, the MYSQL_OPTION schema is excluded by the dump utilities when ocimds: true and is automatically excluded when loading a dump into a MySQL HeatWave DB System. (Bug #37023079)
- As of this release, MySQL Shell supports the following AWS credential providers:
  - Assume Role Provider
  - Amazon Elastic Container Service (Amazon ECS) credentials
  - Amazon Instance Metadata Service (Amazon IMDS) credentials

  See Cloud Service Configuration. (Bug #37000238, Bug #36957811, Bug #36957782, Bug #36947132, WL #15885)
- The compatibility option, unescape_wildcard_grants, is added in this release. When enabled, this strips escape characters in grants on schemas, replacing escaped _ and \% wildcards in schema names with _ and % wildcard characters. When the partial_revokes system variable is enabled, the

`\` character is treated as a literal, which could lead to unexpected results. It is strongly recommended to check each such grant before enabling this option. (Bug #36524862)

- A `dropExistingObjects` option was added to the `copyInstance` and `loadDump` utilities. This option enables you to drop existing user accounts and objects in the target before creating them from the dump or copied instance.

  

  Note

  Schemas are not dropped.

  See Options for Load Control and Options for Copy Control. (Bug #36050341, Bug #36561962)

- The index creation step of a load operation now includes percentage completion information. (Bug #35495220)

- The following utilities now support multipart uploads for prefix pre-authenticated requests:

  - `util.dumpInstance(outputUrl[, options])`
  - `util.dumpSchemas(schemas, outputUrl[, options])`
  - `util.dumpTables(schema, tables, outputUrl[, options])`

  In previous releases, the data was written to a local temporary folder before uploading to Object Storage. As of this release, the data is written directly to Object Storage. (WL #16003)

Utilities Bugs Fixed

- The upgrade checker utility generated a false positive if `secure_file_priv` contained a valid directory. (Bug #36846572)

- Under certain circumstances, using zstd compression, the dump utilities could generate corrupted data files. (Bug #36836188)

- Attempting to run the dump or copy utilities from MySQL Shell 8.0.x against a more recent version of the server, such as 8.4.0, could result in a syntax error.

  Many breaking changes have been made to MySQL syntax and configuration between 8.0.37 and 8.4.x, and higher, such as replacing `SHOW MASTER STATUS` with `SHOW BINARY LOG STATUS`, for example. There were also many removals. See the release notes for those server versions for more information.

  As of this release, the dump and copy utilities raise an error when such incompatibilities are detected and recommend the appropriate MySQL Shell upgrade.

  

  Important

  It is always recommended to use the latest version of MySQL Shell.

  (Bug #36701854)

- Running any of the diagnostics utilities against an instance with binary logging disabled, and using an X Protocol connection, resulted in an error.

  As of this release, the diagnostics utilities use classic connections, only. If the user connects to an instance using X Protocol, and runs any of the diagnostics utilities, they automatically establish a classic connection to the instance. (Bug #36613129)

- Mismatched `lower_case_table_names` values could cause dump loading to fail. As of this release, the `lower_case_table_names` value of the source is added to the `@.json` file and is checked by the load utility for mismatch with the target's value. A warning is displayed if the values do not match. (Bug #36509026)
- Dump process failed if an empty table was encountered and `compression` was set to `none`. (Bug #34891382)
- The upgrade checker utility returned an error claiming that it was not possible to upgrade from MySQL 8.0.x to MySQL 8.4.x. This upgrade path is possible.

  See Upgrade Paths for information. (Bug #115798, Bug #36930714)
- Running the diagnostic utility `util.debug.collectDiagnostics` with the parameter `schemaStats:true` returned MySQL Error 1242 if a table had more than one column of type BLOB. (Bug #115033, Bug #36658194)
- The upgrade checker utility reported Zero Date errors for views using the `NOW()` and `SYSDATE()` functions.

  As of this release, views are excluded from the Zero Date check. (Bug #114347, Bug #36403042)

Functionality Added or Changed

- The MySQL Shell log format has been updated to include the process (PID) and thread ID (TID). (Bug #114343, Bug #36402272, Bug #24678752)
- MySQL Shell now supports OpenID Connect. The `authentication_openid_connect` client plugin is bundled with your MySQL Shell installation.

  See Using OpenID Connect.

  OpenID Connect authentication is supported by MySQL Enterprise Edition, only. (WL #16470)

Bugs Fixed

- MySQL Shell failed to start if installed by MSI on Microsoft Windows 11 with Visual Studio Redistributable version 14.3x or lower. On Windows platforms, MySQL Shell requires Visual Studio Redistributable version 14.4x or higher.

  See Microsoft Visual C++ Redistributable latest supported downloads. (Bug #37049411)
- The MySQL Shell MSI progress dialogs displayed numbered placeholders instead of the installation values. (Bug #37033676)
- When running MySQL Shell over SSH, if a command was entered on the command line, but not executed, closing the SSH session could result in the command being executed without user input. (Bug #36861912)
- MySQL Shell closed unexpectedly if autocomplete (tab) was used for a `\command` followed by a space. (Bug #36858205)
- MySQL Shell set the environment variable `PYTHONHOME` on all platforms even if Python was not bundled. As of this release, `PYTHONHOME` is no longer set and the bundled Python executable is moved to another directory on Windows platforms, which corresponds to the structure used by a standalone Python installation. (Bug #36836320)
- The Python library `hashlib` could not be imported on MacOS platforms. (Bug #36803237)

- MySQL Shell did not register interactive connections as interactive. As a result, the system variable `interactive_timeout` was not honored.

  As of this release, interactive connections are treated as such. You can also define a connection as interactive with the new MySQL Shell connection option, `client-interactive`. (Bug #36339280)

- MySQL Shell error codes were not returned for commands run in CLI mode. (Bug #36156164)

- The URI parser threw an exception if the URI contained an unescaped @ character. (Bug #36105235)

- Corrected a typo in the return value for `ClassicResult.getStatementId`.

  Thanks to Mustafa Uzun for the contribution. (Bug #112718, Bug #35912863)

Changes in MySQL Shell 9.0.1 (2024-07-23)

Bugs Fixed

- It was not possible to start MySQL Shell on Oracle Linux 8 running on ARM platforms if `PAGE_SIZE` was set to 64K. An error similar to the following was displayed:

  ```
  mysqlsh: error while loading shared libraries:
  libantlr4-runtime.so.4.10.1: ELF load command alignment
  not page-aligned
  ```

  (Bug #36792750)

Changes in MySQL Shell 9.0.0 (2024-07-01)

Important

This release is no longer available for download. It was removed due to a dependency on a version of the server which is also no longer available for download. See MySQL 9.0.0 Release Notes. Please upgrade to MySQL Shell 9.0.1 instead.

- AdminAPI Added or Changed Functionality
- AdminAPI Bugs Fixed
- Utilities Added or Changed Functionality
- Utilities Bugs Fixed
- Functionality Added or Changed
- Bugs Fixed

AdminAPI Added or Changed Functionality

- InnoDB ReplicaSet locking has been updated and simplified; no longer requiring multiple per-instance locks.

  See InnoDB ReplicaSet Locking. (Bug #35913824)

- It is now possible to dissolve a ClusterSet, partially or completely, using the following methods:

  - `clusterSet.dissolve()`: dissolves the ClusterSet completely by stopping replication between each replica and primary Cluster and removing any metadata related to the ClusterSet. Each cluster of the ClusterSet is also dissolved.
  - The `dissolve` option of `clusterSet.removeCluster()`: removes the specified Cluster from the ClusterSet, enabling it to function as a standalone Cluster.

  See Dissolving a ClusterSet. (Bug #108373, Bug #34564242, WL #15378)

- It is now possible to execute SQL on all reachable members of a Cluster, ClusterSet, or ReplicaSet, using `.execute()`.

  See Executing SQL on Topologies. (WL #15820)

AdminAPI Bugs Fixed

- MySQL Shell closed unexpectedly when calling certain AdminAPI functions on EL7 platforms. (Bug #36651010)
- `dba.reboot_cluster_from_complete_outage()` disabled `super_read_only` on the primary member of an INVALIDATED Cluster. As a result, clients continued to perform updates and introduce errant transactions.

  As of this release, `dba.reboot_cluster_from_complete_outage()` enables `super_read_only` on the primary member and disables the Group Replication action `mysql_disable_super_read_only_if_primary`. (Bug #36562916)
- If an attempt to create a Replica Cluster failed due to a timeout and the revert also failed due to a timeout, the Replica Cluster could be left in an inconsistent state; ONLINE, but not associated with the ClusterSet's metadata. This specific issue was caused by low values for `wait_timeout` and `interactive_timeout`.

  The following changes were made:

  - `wait_timeout` is checked and, if set to a value lower than the default of 8 hours, is set to 8 hours.
  - *`Cluster`*`.rescan()` is extended with a new option, `repairMetadata` which can be enabled to resolve inconsistencies in the Cluster's metadata.
  - *`Cluster`*`.dissolve()` can now be used on Clusters in this inconsistent state.

  (Bug #36495756)
- The documentation for Rescanning a Cluster did not make clear that while `group_replication_transaction_size_limit` is set to the maximum value in Replica Clusters, the original value is stored in the metadata schema and is restored by *`Cluster`*`.rescan()` in the event of a switchover or failover. This overwrites any user-defined value set on the Replica Cluster.

  The documentation is updated with this information. (Bug #36494958)
- Recent enhancements enabled MySQL Router to expose its configuration and defaults in the metadata schema. However, the `remove_router_metadata` function was not updated to properly clean this data.

  Also, the `router_options()` function filtered the common options from the default options list. (Bug #36276246)

- `reboot_cluster_from_complete_outage()` failed for a Replica Cluster if Group Replication was stopped and the replication channel reset. This failure was caused by the absence of the recovery user.

  As of this release, `reboot_cluster_from_complete_outage()` always recreates the recovery user. (Bug #35181181)

Utilities Added or Changed Functionality

- The following check was added to the Upgrade Checker utility:

  - `foreignKeyReferences`: Checks for foreign keys referencing non-unique and partial indexes.

  (Bug #36553868)

- The following Upgrade Checker utility sysvar checks have been combined into a single check, `sysvars`:

  - `sysVarsNewDefaults`
  - `sysvarAllowedValues`
  - `removedSysVars`

  See Utility Checks. (WL #16262)

- The following utilities now support Instance and Resource Principal authentication on Oracle Cloud Infrastructure:

  - `util.dumpInstance()`
  - `util.dumpSchemas()`
  - `util.dumpTables()`
  - `util.loadDump()`
  - `util.exportTable()`
  - `util.importTable()`

  Each utility is extended with a new option, `ociAuth`, which enables you to specify one of the following Oracle Cloud Infrastructure authentication types:

  - `api_key`
  - `instance_principal`
  - `resource_principal`
  - `security_token`

  See MySQL Shell Utilities. (WL #15884)

Utilities Bugs Fixed

- The formatting of the report returned by the Upgrade Checker utility's `invalidPrivilege` check is improved in this release. Instead of returning a message for each user, the users are grouped with the message. (Bug #36613895)

- As of this release, all failed connections to the supported object storage platforms are retried three times, with a 1 second delay between retries.

  If a failure occurs 10 minutes after the connection was created, the delay is changed to an exponential back-off strategy:

  - First delay: 3-6 seconds
  - Second delay: 18-36 seconds
  - Third delay: 40-80 seconds

  (Bug #36597063, Bug #36256053)

  References: See also: Bug #35396788.

- `restrict_fk_on_non_standard_key`, introduced in MySQL 8.4.0, prohibits creation of foreign keys which reference only part of a composite key when enabled. This system variable is enabled by default on MySQL HeatWave Service and resulted in errors loading dumps which contained such keys when detected by the Upgrade Checker utility's `foreignKeyReferences` check.

  As of this release, a new compatibility option is added the dump utilities, `force_non_standard_keys`. This disables checks for non-standard foreign keys, and cause the loader to set the session value of `restrict_fk_on_non_standard_key` to OFF. (Bug #36553849)

- Primary keys defined on an `ENUM` column were reported as missing for dumps with `ocimds:true`. This was caused by a fix in an earlier version which instructed the dump utility to ignore primary keys or unique indexes which contain one or more `ENUM` columns when selecting an index for chunking.

  As of this release, information about the index selected for chunking and whether the table has a primary key is separated. (Bug #36493316)

  References: See also: Bug #35180061.

- As of this release, the Schema Dump utility uses the `JSON` output format for its generated `EXPLAIN` plans. This was done for compatibility reasons. (Bug #36470302)

- It was not possible to run the Upgrade Checker utility against an MySQL HeatWave Service DB System. An error was returned relating to missing `RELOAD` privileges. `RELOAD` is not granted to MySQL HeatWave Service users.

  `RELOAD` is not required by the Upgrade Checker when run against MySQL HeatWave Service DB Systems, it is only required against MySQL 5.7.x. As such, the requirement is removed in this release. (Bug #36361159)

- The Upgrade Checker utility did not validate the value of the `configPath` parameter.

  As of this release, the value of `configPath` is validated before running the upgrade checks. (Bug #36332625)

- The Upgrade Checker utility behaved inconsistently in the absence of certain privileges. Sometimes returning an error and sometimes attempting to run its checks. (Bug #36332031)

- The Upgrade Checker utility's `sysvarAllowedValues` did not take into account empty strings as valid values for certain variables, such as `ssl_cipher`, resulting in false negative errors in the report. (Bug #36298612)

- It was not possible to use the Copy utilities with certain MySQL-compatible databases. SQL syntax errors were returned. (Bug #36297963)
- `util.collect_diagnostics()` failed with an `AttributeError` when run against an InnoDB Cluster on which the Group Replication plugin was uninstalled from one or more members. The utility attempted to retrieve values for Group Replication system variables which did not exist because the plugin was uninstalled.

  Thanks to Ioannis Androulidakis for the contribution. (Bug #114707, Bug #36589677)
- Fixed an issue with non-ASCII character handling in the Upgrade Checker utility's `schemaInconsistency` check.

  Thanks to Daniel Lenski and Amazon for the contribution. (Bug #114127, Bug #36340714)

Functionality Added or Changed

- The `mysql_native_password` client plugin is now bundled with MySQL Shell. (Bug #36610756)
- MySQL Shell supports the VECTOR type introduced in MySQL 9.0.

  (Bug #36252107, Bug #36253120)
- In previous versions, if neither port nor protocol were specified in the connection string, MySQL Shell automatically used the X Protocol. As of this release:
  - If neither port nor protocol are specified, the default is Classic Protocol.
  - If port is specified but protocol is not, the default is Classic Protocol.

  In either case, if Classic Protocol fails, X Protocol is attempted instead. (WL #16317)
- The `shell` global object is extended with `register_Sql_Handler`, a function which enables you to register a custom SQL handler. This handler can execute pre-processing steps for the given SQL statements and extend the supported SQL statements with non-standard SQL.

  See Custom SQL Handler. (WL #15949)

Bugs Fixed

- MySQL Shell closed unexpectedly if a native Python object was passed to a Python plugin function. (Bug #36502096)
- MySQL Shell did not prompt for a password if `-p` was specified on the command line without an argument. (Bug #36433418)
- Under certain circumstances, a password prompt was not returned although no password was provided on the command line or defined in a configuration file. (Bug #36422502, Bug #36422492)
- `--no-password` did not work if a password was defined in the server's configuration file or if it was provided earlier in the command line. (Bug #36422408)
- Auto-complete was not available in the `\sql` command. (Bug #36420389)
- If `logSql` was set to `ERROR`, MySQL Shell logged the SQL without filtering for unsafe statements. As of this release, the pattern defined in `logSql.ignorePatternUnsafe` is used to filter unsafe SQL from the log.

Also, the pattern which triggered the filter is logged. (Bug #36014067)

- Special characters, such as tab or newline, were not supported in utility calls from the command line. For example, in the following command, `\t` was not properly handled:

```
> mysqlsh root@localhost -- util import-table sample_us.tsv --schema=test --table=samples --fieldsTermi
```

(Bug #34887426)

- Upgrading MySQL Shell 8.0.35, or higher, on Windows platforms, resulted in multiple installations instead of overwriting the existing installation. (Bug #113732, Bug #36259270)