MySQL Connector/J 6.0 Release Notes

Abstract

This document contains release notes for the changes in each release of MySQL Connector/J.

For additional Connector/J documentation, see MySQL Connector/J 6.0 Developer Guide.

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 (http://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 either the MySQL Forums or MySQL Mailing Lists, where you can discuss your issues with other MySQL users.

For additional documentation on MySQL products, including translations of the documentation into other languages, and downloadable versions in variety of formats, including HTML and PDF formats, see the MySQL Documentation Library.

Document generated on: 2016-08-17 (revision: 9605)

Table of Contents

Preface and Legal Notices

This document contains release notes for the changes in each release of MySQL Connector/J.

Legal Notices

Changes in MySQL Connector/J 6.0

Changes in MySQL Connector/J 6.0.4 (Not yet released, Milestone 3)

Version 6.0.4 has no changelog entries, or they have not yet been published because the product version has not yet been released.

Changes in MySQL Connector/J 6.0.3 (2016-06-24, Milestone 2)

Version 6.0.3 Milestone 2 is the second development release of the 6.0 branch of MySQL Connector/J, providing an insight into upcoming features. It is suitable for use with MySQL server versions 5.5, 5.6, and 5.7. It supports the Java Database Connectivity (JDBC) 4.2 API.

Functionality Added or Changed

- For MySQL server 5.7.5 and later, the EOF packet in the MySQL server/client protocol has been deprecated and replaced by the OK packet. The change is now supported by Connector/J. (Bug #23212347)
- SSL is now supported for connections to a MySQL server using the X Protocol. (Bug #21532788)
- The following objects are no longer extensions of the `Extension` interface, but get their own implementations now, which reduces their dependencies on other objects:
  - `BalanceStrategy`
  - `ProfilerEventHandler`
  - `AuthenticationPlugin`

Bugs Fixed

- An excessive amount of memory was used when the connection properties `enablePacketDebug` and `traceProtocol` were both set to "`true`." (Bug #23535571)
- Connector/J hung, returned a **`NullPointerException`**, or returned an incorrect exception when using result sets with the properties `TYPE_FORWARD_ONLY` and `CONCUR_UPDATABLE`. It was due to an inaccurate check for the cursor for the result set. This fix makes sure Connector/J checks accurately on whether a cursor has been requested, both when executing a statement and fetching its results. (Bug #23201930)
- With some Tomcat web applications, when Connector/J connects to the server with `useLegacyDatetimeCode=false` without setting `serverTimeZone`, a NullPointerException was returned. This was because the timezone property file for Connector/J was loaded by the bootstrap class loader, which did not know the location of the property file and thus failed to load it. This fix avoids the problem by making Connector/J use the same class loader for both the property file and the Connector/J classes. (Bug #23197026, Bug #81214)
- When using server-side prepared statements with `profileSQL=true` and `useInformationSchema=true`, an *`SQLException`* (" `ResultSet is from UPDATE. No Data`") occurred when the client tried to advance to the next row in the result set. This was due to a failure of an internal query for metadata, which is now prevented by this fix. (Bug #23188498)
- The download package for Connector/J 6.0.2 Milestone 1 contained the Developer Guide for the wrong version of Connector/J. (Bug #23111273, Bug #81089)
- A **`NullPointerException`** occurred in `com.mysql.cj.mysqlx.io.AsyncMessageReader` due to a race condition when there were more than 2000 concurrent connections taking place. (Bug #23044312)
- `getTimestamp()` returned wrong value for the fractional part of a `TIME` or `DATETIME` field. (Bug #22932078)

- `LoadBalanceConnectionGroupManager.removeHost()` was not removing hosts as expected. This fix tries to ensure that host removals will be successful under different situations. (Bug #22848249)

  References: See also: Bug #22678872.

- For a load-balanced connection, an *`ArrayIndexOutOfBoundsException`* was thrown when `ConnectionGroupManager.removeHost()` was called. It was due to an error in `LoadBalancedConnectionProxy.removeHost()`, which has now been fixed. (Bug #22730682)

- A Fabric connection threw a *`NullPointerException`* when all hosts have been removed from its host list, or when the internal load-balanced connection became null due to inconsistency of the replication connection. This fix adds to Connector/J the abilities to handle those situations. Also, a new connection property, `loadBalanceHostRemovalGracePeriod`, has been introduced, which sets the grace period for waiting to remove the currently active host from a load-balanced connection. See the entry for the new property in Driver/Datasource Class Names, URL Syntax and Configuration Properties for Connector/J for details. (Bug #22678872)

  References: See also: Bug #22848249.

- After a failed call of a stored procedure, any subsequent X DevAPI call resulted in a hang. (Bug #22038729)

- At every connection, Connector/J got the `sql_mode` variable from the server and tried to parse it as a number; because sql_mode is not a number (except for very old versions of MySQL), an *`NumberFormatException`* was always thrown and then caught by the code. This fix refactored the code to avoid the unnecessary throwing and catching of the error. (Bug #21181466, Bug #77171)

- The exception message in `CallableStatement()` for incorrect output parameter registration gave little detail and the wrong error code. (Bug #18068303, Bug #71131)

- On very fast servers with other third-party components accessing the data, a `ConcurrentModificationException` was sometimes thrown. This fix prevents the exception by adding a synchronization to `ConnectionImpl.closeAllOpenStatements()`, and by refactoring part of the code inside the method. (Bug #16736619, Bug #59462)

- When working with MySQL server 5.5 and 5.6, because Connector/J did not check the version number of the server it was connected to, errors and failures occurred when there was an attempt to use certain version-dependent features (for example, using fractional seconds for servers earlier than 5.6.4). With this fix, the proper exceptions are thrown in those cases.

Changes in MySQL Connector/J 6.0.2 (2016-04-11, Milestone 1)

Version 6.0.2 Milestone 1 is the first development release of the 6.0 branch of MySQL Connector/J, providing an insight to upcoming features. Although some of these are still under development, this release includes the following new features and changes (in comparison to the current Connector/J 5.1 production release).

The major features of Connector/J 6.0 include:

- Supports MySQL 5.5, 5.6, and 5.7.

- Supports the JDBC 4.2 specification.

- Supports the Java 8 platform (use Connector/J 5.1 for Java 7 or earlier).

- Supports the new X DevAPI. The X DevAPI enables application developers to write code that combines the strengths of the relational and document models using a modern, NoSQL-like syntax that does not assume previous experience writing traditional SQL. To learn more about how to write applications using the X DevAPI see the X DevAPI User Guide. For more information about how the X DevAPI is implemented in Connector/J, see MySQL Connector/J X DevAPI Reference.

Please note that the X DevAPI requires at least MySQL Server version 5.7.12 or higher with the X Plugin enabled. For general documentation about how to get started using MySQL as a document store, see Using MySQL as a Document Store.

For other significant changes from Connector/J 5.1 to 6.0, see Changes in Connection Properties and Changes in the Connector/J API.

Connector/J 6.0.2 includes all the bug fixes that have been incorporated into Connector/J 5.1.38, plus the fixes described below.

Bugs Fixed

- `FabricMySQLDataSource.getConnection()` threw a null pointer exception when a master failover took place. (Bug #22598938)
- The OSGi manifest file in the Connector/J JAR file did not expose the MySQL Fabric packages, so the Fabric-related classes could not be resolved even though they were present in the JAR file. (Bug #22385172)