

MySQL 5.5.25 リリースノート（日本語翻訳）

注意

- ・ 退化バグによってディスク使用量が過剰になる可能性があるため（詳細については、Bug #65745 を参照）、MySQL 5.5.25 は MySQL 5.5.25a によって無効化される。現在の 5.5.25 を使用しているユーザは、5.5.25a にアップグレードすること。5.5.25 へのアップグレードを検討中のユーザは、かわりに 5.5.25a にアップグレードすること。

機能の追加と変更

- ・ **重要な変更:** レプリケーション: [SHOW BINARY LOGS](#) ステートメント（および等価の [SHOW MASTER LOGS](#)）は、[REPLICATION CLIENT](#) 権限を持つユーザが実行できるようになる（以前は、どちらの形のステートメントを使用する際にも [SUPER](#) 権限が必要だった）。
- ・ [--safe-mode](#) サーバオプションは非推奨となり、MySQL 5.6 では削除される。

修正されたバグ

- ・ パフォーマンス: *InnoDB*: [適応フラッシュ](#)に関連するアルゴリズムが改善された。この修正により、[圧縮](#)が使用されデータセットが[バッファプール](#)より大きい場合のフラッシュ処理の率が高くなり、プールが解放される（Bug #13990648、Bug #65061）。
- ・ *InnoDB*: [REPEATABLE READ](#) アイソレーションレベルを使用するトランザクションで、*InnoDB* テーブルに対して [UPDATE](#) または [DELETE](#) ステートメントを使用すると、他のトランザクションによって最近コミットされた行が見落とされる可能性があった [14.3.9.2 項「Consistent Nonlocking Reads」](#) で説明されているように、[REPEATABLE READ](#) トランザクションにおける DML ステートメントは、他のトラン

ザクシオンによってコミットされた行に、たとえクエリでその行が返されなかった場合でも適用される (Bug #14007649、Bug #65111)。

- **InnoDB**: [Innodb buffer pool pages flushed](#) ステータス変数が、本来の値の 2 倍に誤って設定されていた。この変数の値は、[Innodb_pages_written](#) の値を超えてはならない (Bug #14000361、Bug #65030)。
- **InnoDB**: 自身を参照するカラムの[外部キー](#)を作成しようとしたときのエラー処理とメッセージが改善された。以前のメッセージでは、問題がない場合でも[データディクショナリ](#)に問題があると報告されていた (Bug #12902967)。
- **InnoDB**: **InnoDB** テーブルが大きいと、2 時間というタイムアウト値のために [CHECK TABLE](#) ステートメントが失敗する可能性があった。一般的なストレージデバイスの場合、この問題はおよそ 200 または 350GB (I/O 速度によって異なる) を超えるテーブルで発生した。この修正で、チェックするテーブルに対して実行されるロックが緩和され、タイムアウトは起きにくくなる。また **InnoDB** が [CHECK TABLE QUICK](#) 構文を認識するので、タイムアウトの可能性が完全に回避される (Bug #11758510、Bug #50723)。
- **レプリケーション**: [SHOW BINLOG EVENTS](#) の複数インスタンスを同時実行すると、理論的には MySQL Server がクラッシュする可能性があった (Bug #13979418)。
- **レプリケーション**: ステートメントベースのレプリケーションと、レプリケーションフィルタリングのサーバオプションを使用する際に、[AUTO_INCREMENT](#)、[LAST_INSERT_ID\(\)](#)、[RAND\(\)](#)、またはユーザ変数を使用するステートメントは、スレーブ上の誤ったコンテキストで適用される可能性があった ([16.2.3 項「How Servers Evaluate Replication Filtering Rules」](#) を参照) (Bug #11761686、Bug #54201)。

参考: Bug #11754117、Bug #45670、Bug #11746146、Bug #23894 も参照。

- **レプリケーション**: 複合プライマリーキーを持つテーブルで、そこに含まれる [AUTO_INCREMENT](#) カラムが複合キーの最初のカラムでない場合にそのテーブルに [INSERT](#) すると、ステートメントベースのバイナリログまたはレプリケーションに対して安全ではない。現在このようなステートメントは安全でないとマークされ、[STATEMENT](#) バイナリログフォーマットを使用するとエラーになる。詳細について

は、[16.1.2.3 項「Determination of Safe and Unsafe Statements in Binary Logging」](#)と、[16.4.1.1 項「Replication and AUTO_INCREMENT」](#)を参照。

注意

[InnoDB](#) ストレージエンジンを使用するテーブルは、この問題の影響を受けない。[InnoDB](#) では、[AUTO_INCREMENT](#) カラムを含む複合キーの作成が許可されず、このカラムがキーの最初のカラムになることはないためである。

(Bug #11754117、Bug #45670)

参考: Bug #11761686、Bug #54201、Bug #11746146、Bug #23894 も参照。

- [SHOW TABLES](#) は、必要な情報があらかじめディスクキャッシュに存在しない場合、極端に遅くなった (Bug #60961、Bug #12427262) 。
-

※本翻訳は、理解のための便宜的な訳文として、オラクルが著作権等を保有する英語原文を NRI の責任において翻訳したものであり、変更情報の正本は英語文です。また、翻訳に誤訳等があったとしても、オラクルには一切の責任はありません。