

MySQL 5.1.62 リリースノート（日本語翻訳）

機能の追加と変更

- バージョン 5.1.24 以前の MySQL に存在した `utf8_general_ci` と `ucs2_general_ci` の動作を維持するため、新しい `utf8_general_mysql500_ci` と `ucs2_general_mysql500_ci` の照合が追加された。Bug #27877 では元の照合のエラーが修正されたが、ドイツ語におけるラテン小文字のエスツェット '`ß`' が含まれるカラムについて非互換性が生じた（修正の結果、この文字を比較すると以前は等しいと評価されなかった文字が、等しいと評価されるようになった）。5.1.24 以前のバージョンから MySQL 5.1.24 以降にアップグレードした後に現れる兆候として、[CHECK TABLE](#) を実行すると次のエラーが生成される。

- ```
Table upgrade required.
Please do "REPAIR TABLE `t`" or dump/reload to fix it!
```

この問題を [REPAIR TABLE](#) で修正することはできなかった。新しい照合を使用すると、MySQL 5.1.24 以前で作成された古いテーブルを現在のバージョンの MySQL にアップグレードすることが可能。

テーブルファイルを移動しないバイナリアップグレード後に、対象のテーブルを変換するには、新しい照合を使用するようにテーブルを変更する。たとえばテーブル `t1` に、問題のある `utf8` カラムが 1 つ以上あると仮定した場合、このテーブルをテーブルレベルで変換するには、次のようなステートメントを使用する。

```
ALTER TABLE t1
CONVERT TO CHARACTER SET utf8 COLLATE utf8_general_mysql500_ci;
```

カラムごとの基準で変更を適用する場合には、次のようなステートメントを使用する（`COLLATE` 句を除き、元の指定どおりにカラム定義を繰り返すこと）。

```
ALTER TABLE t1
MODIFY c1 CHAR(N) CHARACTER SET utf8 COLLATE
utf8_general_mysql500_ci;
```

ダンプと再ロードの手順でテーブルをアップグレードするには、[mysqldump](#) を使用してテーブルをダンプし、新しい照合が使用されるようにダンプファイルで [CREATE TABLE](#) ステートメントを変更してから、テーブルを再ロードする。

適切な変更を行うと、[CHECK TABLE](#) でエラーが報告されなくなる。

詳細については、[2.13.3 項「Checking Whether Tables or Indexes Must Be Rebuilt」](#) と [2.13.4 項「Rebuilding or Repairing Tables or Indexes」](#) を参照 (Bug #43593、Bug #11752408)。

- yaSSL は 1.7.2 から 2.2.0 にアップグレードされた。

### 修正されたバグ

- **セキュリティ修正:** Bug #63775 が修正された。
- **矛盾を含む可能性がある変更:** 以前の変更 (MySQL 5.1.59 と 5.5.16) により、General Availability ステータスのシリーズ (MySQL 5.1 と 5.5) における日付処理の動作が変更されたことが判明した。この変更は元に戻された。

この変更は、引数として [DATE \(\)](#) 関数を渡されたとき関数の一部がより厳格になるというもので、日にちの部分がゼロの不完全な日付が拒否されていた。

[CONVERT TZ \(\)](#)、[DATE ADD \(\)](#)、[DATE SUB \(\)](#)、[DAYOFYEAR \(\)](#)、[LAST DAY \(\)](#)、[TIMESTAMPDIFF \(\)](#)、[TO DAYS \(\)](#)、[TO SECONDS \(\)](#)、[WEEK \(\)](#)、[WEEKDAY \(\)](#)、[WEEKOFYEAR \(\)](#)、[YEARWEEK \(\)](#) の各関数が影響を受けた。これ以前の動作が回復された (Bug #13458237)。

- **重要な変更:** **InnoDB:** **UPDATE** 操作が原因で行のサイズが大きくなると、その行に関する情報が引き続き **InnoDB** ページサイズの制約内に収まるように、他の (更新されていない) カラムがオフページストレージに移動される可能性があった。新しく割り当てられるオフページデータへのポインタは、ページが割り当てられ書き込まれるまで設定されないため、カラムをページ外へ移動中にシステムがクラッシュすると、データが失われる可能性があった。この問題は、Barracuda ファイル形式で [ROW\\_FORMAT=DYNAMIC](#) または [ROW\\_FORMAT=COMPRESSED](#) を使用するテーブル

において、特に `innodb file per table` 設定が有効な場合に頻繁に発生した。ページ割り当て操作は、`.ibd` テーブルスペースファイルを拡張する際に頻繁に発生するためである。この問題は、InnoDB のバージョン、ファイル形式、行形式がどのような組み合わせでも発生する可能性があった。

これに関連する問題として、`UPDATE` 操作や、削除されてマークされたレコードを再利用する `INSERT` 操作の際に、アイソレーションレベルにかかわらず、影響されるカラムのデータが他のトランザクションで無効になる場合があった。

カラムデータを元のページから移動してポインタで置き換える操作の順序が修正され、カラムデータが転送される正確なタイミングでクラッシュが発生しても、クラッシュリカバリ時に転送は再実行されなくなる。

MySQL 5.1 では、この修正は InnoDB プラグインに適用されるが、組み込みの InnoDB ストレージエンジンには適用されない (Bug #13721257、Bug #12612184、Bug #12704861)。

- *InnoDB*: ゼロ長の値 (') を含むカラムにインデックスを作成する際、デバッグビルドでのみ、誤った表明が発生する可能性があった (Bug #13654923)。
- *InnoDB*: `ALTER TABLE ... ADD COLUMN` などの DDL 操作が停止し、最終的に `fil_rename_tablespace` についての `Error 1005: Can't create table` メッセージでタイムアウトする可能性があった (Bug #13636122、Bug #62100、Bug #63553)。
- *InnoDB*: C プリプロセッサシンボルとマクロ `HAVE_purify`、`UNIV_INIT_MEM_TO_ZERO`、`UNIV_SET_MEM_TO_ZERO` への参照が、InnoDB ソースコードから削除された。これらは Valgrind 用に用意されたデバッグビルドでのみ使用されたもので、`UNIV_MEM_INVALID()` マクロの呼び出しに置き換えられている (Bug #13418934)。
- *InnoDB*: InnoDB テーブルに対する DDL 操作によって MySQL サーバがビジー状態になり、次の表明エラーで停止する可能性があった。

```
InnoDB: Failing assertion: trx->error_state == DB_SUCCESS
```

このエラーは、1023 個の UNDO スロットがすべて同時トランザクションによって使用されている際に DDL 操作を実行すると発生した。InnoDB の UNDO スロットの数を増やしたことにより同時トランザクションの数（UNDO スロットに対応する）が 1K から 128K に増えたため、MySQL 5.5 および 5.6 ではこのエラーが発生しにくくなった（Bug #12739098、Bug #62401）。

- **InnoDB:** 1024 個の同時 InnoDB トランザクションが実行され、innodb file per table 設定が有効な場合、InnoDB テーブルに対する CREATE TABLE 操作が失敗する可能性があった。CREATE TABLE が失敗した後の .ibd ファイルが放置されていたため、ロードの削除後にもテーブルを作成できなかった。

この修正で、不適切な .ibd ファイルを削除するエラー処理が追加される。InnoDB の UNDO スロットの数を増やしたことにより同時トランザクションの数が 1K から 128K に増えたため、MySQL 5.5 および 5.6 ではこのエラーが発生しにくくなった（Bug #12400341）。

- **InnoDB:** InnoDB パーティションドテーブルを Linux システムから Windows システムにコピーすると、次のエラーが発生する可能性があった。

- 101115 14:19:53 [ERROR] Table .¥test¥d has no primary key in InnoDB data dictionary, but has one in MySQL!

Linux から Windows に InnoDB テーブルをコピーする際の解決策としては通常、Linux で lower\_case\_table\_names オプションを有効にしてテーブルを作成する。パーティションドテーブルのファイル名に **#P#** を追加した場合は、この解決方法が適用されなかった（Bug #11765438、Bug #58406）。

- **InnoDB:** \$TMPDIR 変数のパスが / 文字で終わっている場合、InnoDB ストレージエンジンを使用するテンポラリテーブルでサーバ起動時にエラーが発生する可能性があった。エラーログは以下ようになる。

- 120202 19:21:26 InnoDB: Operating system error number 2 in a file operation.
- InnoDB: The error means the system cannot find the path specified.

- InnoDB: If you are installing InnoDB, remember that you must create
- InnoDB: directories yourself, InnoDB does not create them.
- 120202 19:21:26 InnoDB: Error: trying to open a table, but could not
- InnoDB: open the tablespace file './t/#sql7750\_1\_0.ibd'!
- InnoDB: Have you moved InnoDB .ibd files around without using the
- InnoDB: commands DISCARD TABLESPACE and IMPORT TABLESPACE?
- InnoDB: It is also possible that this is a temporary table #sql...,  
InnoDB: and MySQL removed the .ibd file for this.

この問題の回避策としては、類似のテンポラリテーブルを再作成し、その `.frm` ファイルを、エラーメッセージで示された名前の下にある `tmpdir` にコピー (`#sql1123.frm` など) して、`tmpdir` を末尾のスラッシュなしで通常の値 (`/var/tmp` など) に設定して `mysqld` を再起動していた。起動時に、MySQL は `.frm` ファイルを認識し、孤立したテンポラリテーブルに対して `DROP TABLE` を発行する (Bug #11754376、Bug #45976)。

- `BETWEEN` 句で参照されている `CHAR` カラムに対するインデックスを使用するクエリが、無効な結果を返す可能性があった (Bug #13463488、Bug #63437)。
- 範囲条件を評価する際にオプティマイザが `DECIMAL` 値の変換を実行すると、結果が正しくならない可能性があった (Bug #13453382)。
- `--xml` オプションを指定すると、`mysqldump--routines` がストアドルーチン、トリガ、イベントのダンプに失敗した (Bug #11760384、Bug #52792)。
- `FEDERATED` テーブルを使用するレプリケーションスレーブで、長時間実行する操作がタイムアウトしてエラー1160「Got an error writing communication packets」などが発生する可能性があった。`FEDERATED` テーブルを複製しなくても、この問題は発生した (Bug #11758931、Bug #51196)。

参考: Bug #12896628、Bug #61790 も参照。

- ステートメントを開始しようとして失敗しても、クライアントはステートメントの実行より前にエラーメッセージを受け取る準備ができていないため、クライアントにその問題が報告されない可能性があった。ユーザはクエリを実行できないため、明確なエラーがないまま切断されるだけだった。

この問題が修正され、クライアントはステートメントの開始と同時にエラーに対応できるようになるので、ユーザが切断される前にエラーが報告されるようになる (Bug #1175281、Bug #47032)。

- 行形式が固定幅のテーブルを修復するためにソート修復のメソッドを指定して [myisamchk](#) を使用すると、行ポインタのサイズが小さくなり、結果的に最大データファイルサイズが小さくなる可能性があった (Bug #48848、Bug #11756869)。
- ロックが不適切なため、テーブルに対する修正およびチェック操作と同時に [ARCHIVE](#) テーブルへの挿入を行うとテーブルが破損した (Bug #37280、Bug #11748748)。
- ある状況下では、[SUBSTRING INDEX \(\)](#) の結果の、前の行の内容への依存が正しくなかった (Bug #42404、Bug #11751514)。

※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。