

HP OpenSource プループリント
MySQL 5.0
日本語設定ガイド

ver. 1.0



はじめに

本書は、オープンソースのデータベースである MySQL Server 5.0 で日本語データ処理を行う場合に必要な情報と設定方法について説明します。

本書の範囲

本書では、MySQL Server 5.0 で日本語処理を行うための情報や MySQL Server 側及び MySQL Server を利用するクライアント側の設定方法を説明します。MySQL Server のインストール方法やクライアントのインストール方法等はマニュアルまたはその他資料をご参照ください。

なお、本書で使用されている「文字コード」という言葉は MySQL での「Character Set」に相当し、厳密な「文字コード」の定義と異なります。本書では「文字コード」を MySQL の「Character Set」の意味と同様に、「文字集合」と「文字エンコーディング」が統合された意味で使用しています。

本書の構成

目次

第 1 章 MySQL Server での日本語処理概要

MySQL Server 5.0 で行われる日本語処理の概要を説明します。

第 2 章 MySQL Server 5.0 での日本語設定方法

MySQL Server 5.0 側の日本語処理に関する設定方法について説明します。

第 3 章 クライアントの日本語設定方法

MySQL Server 5.0 を利用するクライアント側での日本語処理に関する設定方法について説明します。

付録

Appendix1: 日本語設定・確認方法一覧

Appendix2: 日本語処理における注意事項

Appendix3: MySQL Server 設定ファイル

目次

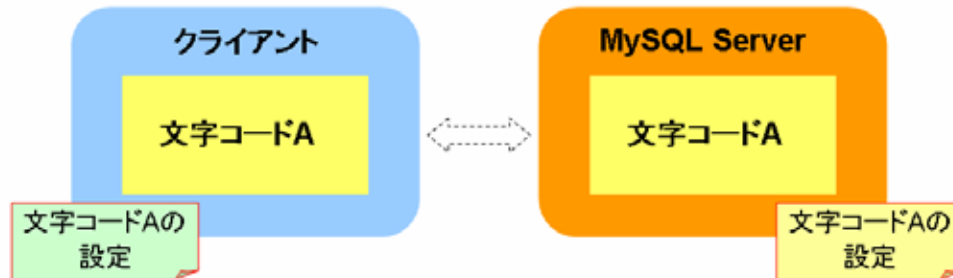
はじめに	2
本書の範囲	2
本書の構成	2
目次	3
1 MySQL Server での日本語処理概要	4
1.1 MySQL Server 側の文字コード設定	4
1.2 MySQL Server でサポートされている日本語関係の文字コード	5
2 MySQL Server 側の日本語設定方法	5
2.1 MySQL Server(mysql) の文字コード	5
2.1.1 設定方法	5
2.1.2 確認方法	6
2.2 データベースの文字コード	6
2.2.1 設定方法	6
2.2.2 確認方法	6
2.3 テーブルの文字コード設定	7
2.3.1 設定方法	7
2.3.2 確認方法	7
2.4 カラムの文字コード設定	7
2.4.1 設定方法	7
2.4.2 確認方法	7
3 クライアントの日本語設定方法	8
3.1 mysql コマンドラインツール	8
3.1.1 設定方法	8
3.1.2 確認方法	8
3.2 Javaアプリケーション	9
3.2.1 設定方法	9
3.2.2 確認方法	9
Appendix 1: 日本語設定・確認方法一覧	10
MySQL Server 側	10
クライアント側	10
Appendix 2: 日本語処理における注意事項	10
日本語のデータベース名とテーブル名	10
Windows Vista 対応について	10
Javaアプリケーションでの EUC_JP_Solaris の使用	11
Appendix 3: MySQL Server 設定ファイル	11
Windows の場合	11
Unix系の場合	12

MySQL Server 5.0 では、MySQL Server 側とクライアント側で正しく設定を行うことで日本語を処理することができます。

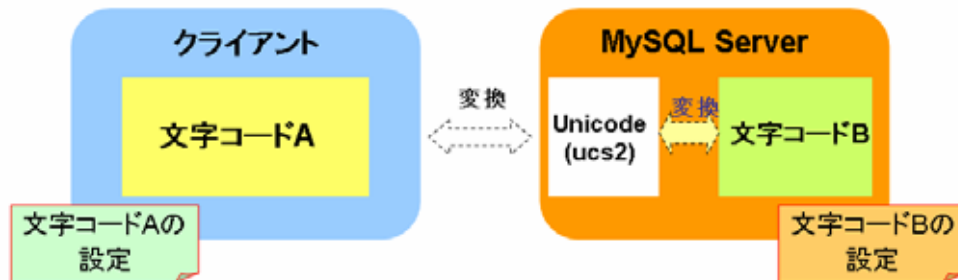
1 MySQL Server での日本語処理概要

MySQL では、MySQL Server 側と MySQL Server にアクセスするクライアント側の両方で文字コードを設定します。

MySQL Server 側とクライアント側で同一の文字コードを使用する場合は、以下の様に同一の文字コードを設定します。



しかし、MySQL Server 側とクライアント側の文字コード設定が異なる場合には、以下の様にそれぞれ使用する文字コードを設定します。この場合 MySQL Server 側で自動的に文字コードの変換が行われますが、異なる文字コード間で直接変換が行われるのではなく、Unicode(ucs2)を介して変換が行われます。

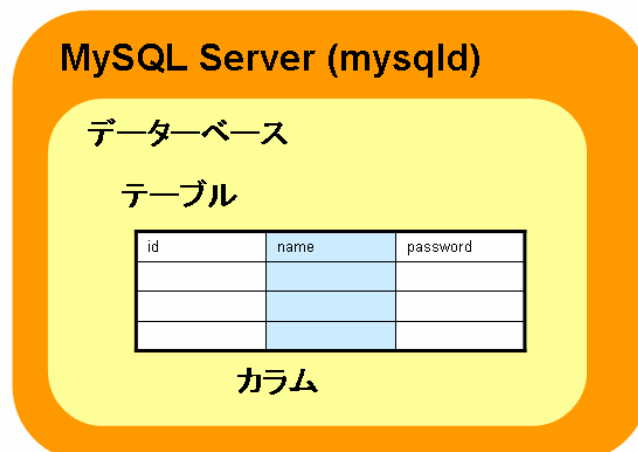


1.1 MySQL Server 側の文字コード設定

MySQL Server 側ではいくつか場所に文字コードを設定することができます。

文字コードが設定できる場所は以下となります。

- 文字コードが設定できる対象
- MySQL Server (mysqld)
 - データベース
 - テーブル
 - カラム



これらの文字コードの設定方法については、「2. MySQL Server 側の日本語設定方法」をご参照ください。

また、それぞれの設定の優先順位は次の通りです。

優先度	対象
高	カラム
	テーブル
	データベース
低	MySQL Server

1.2 MySQL Server でサポートされている日本語関係の文字コード

MySQL Server 5.0 でサポートされている日本語が扱える文字コードのうち、使用が推奨される文字コードは次の通りです。

MySQL での文字コード	説明
cp932	文字エンコーディング方法は Shift-JIS。 文字集合は Windows で使用されている CP932 の文字集合(JIS X201、JIS X208、NEC 特殊文字、NEC 選定 IBM 拡張文字、IBM 拡張文字)。
eucjpms	文字エンコーディング方法は EUC-JP。 文字集合は cp932 と同等。
utf8	文字エンコーディング方法は RFC3629 で定義されている UTF-8。ただし 3 バイトまでの UTF-8 に対応。 文字集合は Unicode の BMP に対応。

注意：MySQL では“sjis”、“eucjp” という文字コードもサポートされていますが、どちらも Windows で使用されている NEC 特殊文字などの文字集合に対応していない等の制限があるため、特に理由の無い限り使用しないほうが良いでしょう。通常は“sjis”の代わりに“cp932”、“eucjp”の代わりに“eucjpms”を使用してください。

2 MySQL Server 側の日本語設定方法

「1.1 MySQL Server 側で文字コードを設定できる箇所」の記述の通り、MySQL Server では複数の箇所に文字コードを設定することが可能です。以下にそれぞれの設定方法を確認方法を説明します。

2.1 MySQL Server(mysql) の文字コード

2.1.1 設定方法

MySQL Server(mysql) の文字コードを設定するには、MySQL の設定ファイル(my.cnf もしくは my.ini)の[mysql]セクションで、**character_set_server** システム変数を設定します。

```
[mysql]
character_set_server = cp932
```

参考：MySQL の設定ファイルについては「Appendix 3: MySQL Server の設定ファイル」をご参照ください。

もしくは、MySQL Server 起動時に **-character-set-server** オプションでも設定することができます。

```
# ./bin/mysql --character-set-server=eucjpms --user=mysql &
```

注意 : my.cnf ファイルで character_set_server を設定し、同時に MySQL Server 起動時に -character-set-server オプションを設定した場合には、起動オプションの -character-set-server の設定が優先されます。

2.1.2 確認方法

MySQL Server(mysql)の文字コードは、**character_set_server** システム変数を参照することで確認できます。

```
mysql> SHOW VARIABLES LIKE 'character_set_server';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| character_set_server | cp932 |
+-----+-----+
1 row in set (0.00 sec)
```

2.2 データベースの文字コード

2.2.1 設定方法

データベースの文字コードを設定するには、データベースの作成時に実行する **CREATE DATABASE** 文に **CHARACTER SET** オプションを指定することで設定することができます。

```
mysql> CREATE DATABASE db_cp932 CHARACTER SET cp932;
```

データベース作成時に CHARACTER SET オプションを指定しなかった場合には、MySQL Server(mysql) の文字コードがデータベース文字コードになります。

2.2.2 確認方法

特定のデータベースの文字コードは、**information_schema** データベースの **SCHEMATA** テーブルから確認することができます。

```
mysql> use information_schema
mysql> SELECT SCHEMA_NAME, DEFAULT_CHARACTER_SET_NAME FROM SCHEMATA WHERE
SCHEMA_NAME = 'db_cp932';
+-----+-----+
| SCHEMA_NAME | DEFAULT_CHARACTER_SET_NAME |
+-----+-----+
| db_cp932    | cp932                      |
+-----+-----+
1 row in set (0.01 sec)
```

また、デフォルトのデータベースの文字コードは、**character_set_database** システム変数を参照することで確認できます。

```
mysql> SHOW VARIABLES LIKE 'character_set_database';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| character_set_database | cp932 |
+-----+-----+
1 row in set (0.00 sec)
```

2.3 テーブルの文字コード設定

2.3.1 設定方法

テーブルの文字コードを設定するには、テーブルの作成時に実行する **CREATE TABLE** 文に **CHARACTER SET** オプションを指定することで設定することができます。

```
mysql> CREATE TABLE tb1_cp932 (id int, name varchar(100)) CHARACTER SET cp932;
```

テーブル作成時に **CHARACTER SET** オプションを指定しなかった場合には、データベースの文字コードがテーブルの文字コードになります。

2.3.2 確認方法

テーブルの文字コードは、**SHOW CREATE TABLE** 文で確認することができます。

```
mysql> SHOW CREATE TABLE tb1_cp932\G
***** 1. row *****
Table: tb1_cp932
Create Table: CREATE TABLE `tb1_cp932` (
  `id` int(11) default NULL,
  `name` varchar(100) default NULL
) ENGINE=MyISAM DEFAULT CHARSET=cp932
1 row in set (0.00 sec)
```

2.4 カラムの文字コード設定

2.4.1 設定方法

カラムの文字コードを設定するには、テーブルの作成時に実行する **CREATE TABLE** 文中のカラムを定義する場所に **CHARACTER SET** オプションを指定することで設定することができます。

```
mysql> CREATE TABLE tb2_cp932 (id int, name varchar(100) CHARACTER SET cp932);
```

カラムを定義する場所に **CHARACTER SET** オプションを指定しなかった場合には、カラムの文字コードはデータベースの文字コードになります。

2.4.2 確認方法

特定のカラムの文字コードは、**information_schema** データベースの **COLUMNS** テーブルから確認することができます。

```
mysql> SELECT COLUMN_NAME, CHARACTER_SET_NAME FROM COLUMNS WHERE TABLE_NAME =
'tb2_cp932';
+-----+-----+
| COLUMN_NAME | CHARACTER_SET_NAME |
+-----+-----+
| id          | NULL                |
| name       | cp932               |
+-----+-----+
2 rows in set (0.01 sec)
```

3 クライアントの日本語設定方法

MySQL Server 5.0 では、MySQL Server にアクセスするクライアント側にも文字コードを正しく設定する必要があります。以下、代表的なクライアントである mysql コマンドラインツールと、Java アプリケーションでの設定方法と確認方法を説明します。

3.1 mysql コマンドラインツール

3.1.1 設定方法

mysql コマンドラインツール(以下 mysql)で文字コードを設定するには、MySQL の設定ファイル(my.cnf もしくは my.ini) [mysql] セクションに **default-character-set** を設定する方法をお勧めします。この設定を行っておくことで、mysql を起動する際に文字コード用の特別なオプションを指定しなくて済むようになります。

```
$ cat /etc/my.cnf
...
[mysql]
default-character-set = cp932
...
$ ./bin/mysql -u root
```

参考 : MySQL の設定ファイルについては「Appendix 3: MySQL Server の設定ファイル」をご参照ください。

上記のような my.cnf ファイルでの設定を行わない場合には、mysql 起動時に **-default-character-set=<文字コード>** オプションを指定します。

```
$ ./bin/mysql -u root --default-character-set=cp932
```

注意(1) : 同時に my.cnf での default-character-set と mysql に -default-character-set オプションを指定した場合には、mysql の -default-character-set オプションが優先されます。

注意(2) : mysql の文字コード設定を行わない場合、mysql の文字コードは latin1 となり、日本語処理が正しく行われません。mysql で日本語処理を行う場合は必ず上記のいずれかの設定を行ってください。

3.1.2 確認方法

mysql コマンドラインツールの文字コードは、**SHOW VARIABLES LIKE 'char%'**; コマンドの実行結果から、以下のシステム変数を参照することで確認できます。

- **character_set_client**
- **character_set_connection**
- **character_set_results**

```
mysql> SHOW VARIABLES LIKE 'char%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| character_set_client | cp932 |
| character_set_connection | cp932 |
| character_set_database | eucjpms |
| character_set_filesystem | binary |
| character_set_results | cp932 |
| character_set_server | eucjpms |
| character_set_system | utf8 |
+-----+-----+
```



```
| character_sets_dir | /mysql/share/mysql/charsets/ |
+-----+-----+
```

3.2 Javaアプリケーション

Java アプリケーションから MySQL Server に接続するには、MySQL 社が提供している MySQL Server 用 JDBC ドライバである Connector/J を利用します。

3.2.1 設定方法

Java アプリケーションの文字コードを設定するには、MySQL Server への接続時の JDBC URL に次のプロパティと値を設定します。

characterEncoding= <Java の文字コード名>

同プロパティで設定された Java の文字コードは、Connector/J によって、次の表にしたがって MySQL の文字コードに変換されます。

MySQL の文字コード名	Java の文字コード名
cp932	Windows-31J もしくは MS932
eucjms	EUC_JP_Solaris
utf8	UTF-8

メモ: MySQL の文字コード sjis と ujis の使用は推奨されませんが、sjis に対応する Java の文字コード名は SJIS となり、ujis に対応する Java の文字コード名は EUC_JP となります。

以下は Java アプリケーションから文字コード Windows-31J を設定し、MySQL Server への接続を得るソースコード例です。

```
Class.forName("com.mysql.jdbc.Driver");
Connection conn =
DriverManager.getConnection(
"jdbc:mysql://localhost:3306/test" +
"?user=dbuser&password=dbpass&characterEncoding=Windows-31J");
```

3.2.2 確認方法

Java アプリケーションの文字コードの確認は、アプリケーションから例えば以下のように **SHOW VARIABLES LIKE 'char%'** を発行することで確認できます。

```
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery("SHOW VARIABLES LIKE 'char%'");
while (rs.next()) {
String key = rs.getString(1); // char_* のシステム変数名
String val = rs.getString(2); // システム変数の値
System.out.println(key + " : " + val);
}
stmt.close();
conn.close();
```

Appendix 1: 日本語設定・確認方法一覧

MySQL Server 側

対象	設定方法	確認方法
MySQL Server (mysqld)	(1) my.cnf ファイル中の [mysqld] セクションで character_set_server = <文字コード> (2) MySQL Server 起動時の -character-set-server=<文字コード> オプション	SHOW VARIABLES LIKE 'character_set_server' 文
データベース	CREATE DATABASE 文の CHARACTER SET オプション	SHOW VARIABLES LIKE 'character_set_database' 文
テーブル	CREATE TABLE 文の CHARACTER SET オプション	SHOW CREATE TABLE 文
カラム	CREATE TABLE 文で定義するカラムへの CHARACTER SET オプション	information_schema データベースの COLUMNS テーブル中の CHARACTER_SET_NAME カラムを検索

クライアント側

クライアント	設定方法	確認方法
mysql コマンドラインツール	(1) -default-character-set=<文字コード> オプション (2) my.cnf ファイル中の [mysql] セクションで default-character-set = <文字コード>	SHOW VARIABLES LIKE 'char%' 文で以下のシステム変数を確認 - character_set_client - character_set_connection - character_set_result
Java アプリケーション	JDBC URL に characterEncoding=<Java の文字コード> プロパティを設定	Java アプリケーションから SHOW VARIABLES LIKE 'char%' 文などを実行

Appendix 2: 日本語処理における注意事項

日本語のデータベース名とテーブル名

MySQL Server ではデータベースを作成するとデータベース名と同じ名前のディレクトリ(フォルダ)がファイルシステム上に作成され、同様にテーブルを作成するとテーブルと同じ名前のファイルが作成されます。MySQL Server 5.0 ではデータベース名とテーブル名に対応するディレクトリ(フォルダ)名とファイル名はどちらも UTF-8 で作成されます。

日本語名のデータベースやテーブルを作成すると、MySQL Server が稼動する OS(オペレーティング・システム)によっては、期待通りのデータベース(ディレクトリ or フォルダ)名やテーブル(ファイル)名が作成されない場合や、他の OS へのポータビリティに関する問題が生じる場合があります。

したがって、MySQL Server 5.0 では日本語名のデータベースやテーブルの作成/使用は行わない方がよいでしょう。

なお、この動作は MySQL Server 5.1 で改善される予定です。

Windows Vista 対応について

Windows Vista では日本語文字集合として JIS X 0213:2004(JIS2004)が採用されており、同文字集合には UTF-8 の 4 バイトで表現される文字が含まれます。

そのため MySQL Server のクライアントが Windows Vista の場合には、クライアントから UTF-8 の 4 バイト文字が MySQL Server に入力される可能性があります。

ところが、MySQL Server 5.0 では UTF-8 の 3 バイト文字までしか対応しておらず、また、UTF-8 の 4 バイト文字が入力された場合には、入力された行の UTF-8 の 4 バイト文字以降の文字はすべてデータ - ベースに格納されません。

したがって、MySQL Server 5.0 では基本的に Windows Vista での日本語文字集合 JIS X 0213:2004(JIS2004)には対応していません。

なお、MySQL Server や各種 Connector の同文字集合への対応は、MySQL 6.0 の予定となっています。

ワークアラウンド

MySQL Server 5.0 で UTF-8 の 4 バイト文字を格納したい場合には、格納するカラムをバイナリ型 (VARBINARY/BLOB) と定義することで擬似的に格納することができます。

ただし、この場合バイナリ型は 1 バイト=1 文字として計算されてしまいますので、そのようなカラムは CHAR_LENGTH()関数が正しい値を返さないなど、文字列としての正しい処理が行えませんので注意が必要となります。

Javaアプリケーションでの EUC_JP_Solaris の使用

MySQL Server 文字コード eucjms と Java の文字コード EUC_JP_Solaris には完全な互換性はありません。そのため MySQL Server 側で eucjms を使用したい場合には、Java 文字コード EUC_JP_Solaris ではなく、Windows-31J もしくは UTF-8 に設定してほうがよいでしょう。

その場合は、JDBC URL に次の設定を行ってください。

```
characterEncoding=Windows-31J&characterSetResults=Windows-31J
```

もしくは、

```
characterEncoding=UTF-8&characterSetResults=UTF-8
```

これにより、MySQL Server の文字コード eucjms と Java 側の文字コードの変換は、MySQL Server 側で Unicode(ucs2)を介して正しく行われることになります。

Appendix 3: MySQL Server 設定ファイル

MySQL Server の設定は、複数の設定ファイル(my.cnf や my.ini)から読み込むことができます。それらの設定ファイル名や設定ファイルの場所、MySQL Server が読み込む設定ファイルの順番は、MySQL Server が稼動するプラットフォームによって異なります。

Windows の場合

Windows 上の MySQL Server は、以下の表にリストされている設定ファイルを、リストの上から順に読み込みます。同じ設定が複数の設定ファイルに記述されている場合は、最後に読み込まれた設定が有効になります。

設定ファイル
<i>WINDIR</i> \my.ini
C:\my.ini
<i>INSTALLDIR</i> \my.ini
-defaults-extra-file=<任意の設定ファイル>

- **WINDIR** は Windows の echo %WINDIR% コマンドで表示されるフォルダです。
- **INSTALLDIR** は MySQL Server をインストールしたフォルダです。

Unix系の場合

Unix 系のプラットフォーム上の MySQL Server は、以下の設定ファイルを表の上にリストされている順に読み込みます。同じ設定が複数の設定ファイルに記述されている場合は、最後に読み込まれた設定が有効になります。

設定ファイル
/etc/my.cnf
<i>\$MYSQL_HOME</i> /my.cnf
--defaults-extra-file=<任意の設定ファイル>
~/my.cnf

- ***\$MYSQL_HOME*** はユーザーが設定できる環境変数です。***\$MYSQL_HOME*** 環境変数が設定されておらず、`mysqld_safe` コマンドを使用して MySQL Server を起動する場合には、***\$MYSQL_HOME*** 環境変数は以下のログで設定されます。
 - MySQL をインストールしたディレクトリを **BASEDIR** に設定し、データディレクトリを **DATADIR** に設定します。
 - **BASEDIR** に `my.cnf` が存在せず、**DATADIR** に `my.cnf` が存在する場合にのみ、***\$MYSQL_HOME*** は **DATADIR** に設定されます。それ以外の場合は、***\$MYSQL_HOME*** は **BASEDIR** に設定されます。

お問い合わせはカスタマー インフォメーションセンターへ
03-5304-6660 月～金9:00～19:00 土10:00～18:00(日、祝祭日、年末年始および5/1を除く)
Linux/オープンソース製品に関する情報は <http://www.hp.com/jp/linux/>

記載されている会社名および商品名は、各社の商標または登録商標です。
記載事項は2007年4月現在のものです。
本書に記載された内容は、予告なく変更されることがあります。
HP製品とサービスに対する保証は、それらに付属する保証書に記載された事項に限られます。
ここに記載した内容は一切追加の保証を意味するものではありません。
本書中の技術的あるいは編集上の誤り、省略に対して、いかなる責任も負いかねますのでご了承ください。
© Copyright 2007 Hewlett-Packard Development Company, L.P.

日本ヒューレット・パッカード株式会社
〒102-0076 東京都千代田区五番町7

