mysql driver manual

A libdbi driver providing connectivity to MySQL databases using libmysqlclient.

Mark Tobenkin

mmt@users.sourceforge.net

mysql driver manual: A libdbi driver providing connectivity to MySQL databases using libmysqlclient.

by Mark Tobenkin

Revision History

Revision 0.4	2005-07-15
Revision 0.3	2003-4-12
Revision 0.2	2003-2-4
Revision 0.1	2002-11-5

Table of Contents

Preface

libdbi (http://libdbi.sourceforge.net) is a database abstraction layer written in C. It implements a framework that can utilize separate driver libraries for specific database servers. The libdbi-drivers (http://libdbi-drivers.sourceforge.net) project provides the drivers necessary to talk to the supported database servers.

This manual provides information about the MySQL driver. The manual is intended for programmers who write applications linked against libdbi and who want their applications to work with the MySQL driver.

Questions and comments about the MySQL driver should be sent to the libdbi-drivers-devel (mailto:libdbi-drivers-devel@lists.sourceforge.net) mailing list. Visit the libdbi-drivers-devel list page (http://lists.sourceforge.net/lists/listinfo/libdbi-drivers-devel) to subscribe and for further information. Questions and comments about the libdbi library should be sent to the appropriate libdbi mailing list.

The MySQL driver is maintained by Mark Tobenkin (mailto:mmt@users.sourceforge.net).

Chapter 1. Introduction

MySQL (http://www.mysql.org), as of the time of this document's writing, is one of the fastest growing database systems worldwide. With approximately 4 million installations, and 30,000 daily downloads, the server has found a niche in the website development community. Quick deployment and ease of use are among it's assets.

Chapter 2. Installation

This chapter describes the prerequisites and the procedures to build and install the mysql driver from the sources.

2.1. Prerequisites

The following packages have to be installed on your system:

libdbi

> This library provides the framework of the database abstraction layer which can utilize the MySQL driver to perform database operations. The download page as well as the mailing lists with bug reports and patches are accessible at sourceforge.net/projects/libdbi (http://sourceforge.net/projects/libdbi). The current version of the MySQL driver requires libdbi version 0.8.x.

libmysqlclient

> This library implements the client API for MySQL programs. Find the most recent release at www.mysql.org (http://www.mysql.org). The current version of the mysql driver was tested with MySQL version 4.1.10a, and it should work with other 4.x releases.

2.2. Build and install the mysql driver

First you have to unpack the libdbi-drivers archive in a suitable directory. Unpacking will create a new subdirectory `libdbi-drivers-X.Y` where "X.Y" denotes the version:

```
$ tar -xzf libdbi-drivers-X.Y.tar.gz
```

The libdbi-drivers project consists of several drivers that use a common build system. Therefore you *must* tell configure explicitly that you want to build the mysql driver (you can list as many drivers as you want to build):

```
$ cd libdbi-drivers
```

```
$ ./configure --with-mysql
```

Run **./configure --help** to find out about additional options.

Then build the driver with the command:

```
$ make
```

> **Note:** Please note that you may have to invoke **gmake**, the GNU version of make, on some systems.

Then install the driver with the command (you'll need root permissions to do this):

```
$ make install
```

To test the operation of the newly installed driver, use the command:

```
$ make check
```

This command creates and runs a test program that performs a few basic input and output tests. The program will ask for an administrator username and password. This user can be any user who has database creation priveleges. Be sure to provide the name of a database which does not already exist (avoid ’test’ which is used on many systems).

> **Note:** If for some reason you need to re-create the autoconf/automake-related files, try running **./autogen.sh**. I’ve found out that the current stable autoconf/automake/libtool versions (as found in FreeBSD 4.7 and Debian 3.0) do not cooperate well, so I found it necessary to run the older autoconf 2.13. If necessary, edit `autogen.sh` so that it will catch the older autoconf version on your system.

Chapter 3. Driver options

Your application has to initialize libdbi drivers by setting some driver options with the `dbi_conn_set_option()` and the `dbi_conn_set_option_numeric()` library functions. The mysql driver supports the following options:

host

The hostname or IP address of the MySQL database server. Use an empty string or "localhost" to connect to a MySQL server running on the local machine.

port (numeric)

The port used to remotely connect to the MySQL database server over TCP. Use "0" (zero) to accept the default socket.

username

The name of the user accessing the database.

password

The password of the user accessing the database.

dbname

The name of the database used for the initial connection. If your program must be able to connect to a fresh MySQL installation, use the system database `mysql`.

encoding

The IANA name of a character encoding which is to be used as the connection encoding. Input and output data will be silently converted from and to this character encoding, respectively. The list of available character encodings depends on your local MySQL installation. If you set this option to "auto", the connection encoding will be the same as the default encoding of the database.

mysql_include_trailing_null (numeric)

This item will tell the driver whether or not to include trailing null values ('\0') at the end of binary strings. This applies to the types BLOB, MEDIUMBLOB, LARGEBLOB etc. A numeric value of 0 will tell the driver to leave off the NULL value. A value of 1 will tell the driver to include the trailing NULL character.

mysql_unix_socket

The filename of the Unix socket used to connect to a MySQL database server running on the local machine. Provide an empty string to use the default socket.

mysql_compression (numeric)

A value larger than zero causes the client/server communication to be compressed. Set this to zero to use no data compression.

Chapter 4. Peculiarities you should know about

This chapter lists known peculiarities of the mysql driver. This includes MySQL features that differ from what you know from the other database servers supported by libdbi, and it includes features and misfeatures introduced by the mysql driver. It is the intention of the driver author to reduce the number of misfeatures in future releases if possible.

4.1. MySQL (mis)features

- To allow for row seeking, results are loaded into memory. This is very inefficient and may provide a bottleneck for large applications.
- DATETIME, TIMESTAMP, DATE and TIME are all treated as the DBI type DATETIME. This is currently a string, but will change in later releases.
- The DECIMAL type is treated as a string, as it has arbitrary precision.