

22.9.3 ODBC-Treiber für das DBMS MySQL

22.9.3.1 Installation

Für das DBMS MySQL gibt es zwei Wege, um die speziellen ODBC-Treiber zu installieren.

22.9.3.1.1 1. Weg

Als spezieller ODBC-Treiber für das DBMS MySQL kann auch der ODBC-Treiber 'odbc-mariadb' vom DBMS MariaDB eingesetzt werden. Diesen Treiber installieren Sie entweder über die Anwendungsverwaltung oder geben in einem Terminal folgende Zeile ein:

```
$ sudo apt-get install odbc-mariadb
```

22.9.3.1.2 2. Weg

Die benötigten speziellen ODBC-Treiber für das DBMS MySQL können auch von der Projektseite von MySQL auf <https://dev.mysql.com/downloads/connector/odbc/> heruntergeladen werden. Achten Sie auf die korrekte Auswahl des Betriebssystems und die Version von MySQL im DEB-Paket:



Abbildung 22.9.3.1.1: Auswahl-Dialog für das Betriebssystem und die MySQL-Version

- Zuerst entpacken Sie das Archiv – wie zum Beispiel `mysql-connector-odbc_8.0.24-1ubuntu20.04_amd64.deb`.
- Dann entpacken Sie das Archiv `data.tar.xz` im Archiv.
- Abschließend kopieren Sie unter Root-Rechten die zwei Treiberdateien `libmyodbc8a.so` und `libmyodbc8w.so`, die sich im Verzeichnis `.../usr/lib/x86_64-linux-gnu/odbc` befinden, in das Verzeichnis `/usr/lib/x86_64-linux-gnu/odbc`.
- Fazit:
In der Datei `libmyodbc8a.so` befindet sich der ANSI-Treiber und in der Datei `libmyodbc8w.so` der Unicode-Treiber.

22.9.3.2 Konfiguration

Die Konfiguration der speziellen ODBC-Treiber für MySQL erfolgt in der systemweit geltenden Konfigurationsdatei `/etc/odbcinst.ini`. Ergänzen Sie mit Root-Rechten diese Datei, indem Sie am Ende die folgenden Zeilen für die drei ODBC-Treiber einfügen:

```
[MySQL ANSI]
Description=MySQL ODBC Driver (ANSI version)
Driver=libmyodbc8a.so
UsageCount=1

[MySQL Unicode]
Description=MySQL ODBC Driver (Unicode version)
Driver=libmyodbc8w.so
UsageCount=1

[MySQL MDB]
Description=MariaDB ODBC Driver
Driver=libmaodbc.so
UsageCount=1
```

Die Namen der ODBC-Treiber stehen in den eckigen Klammern, die bei der Konfiguration des Datenquellenamens (`DataSourceName`) benötigt werden.

22.9.3.3 Konfiguration eines Datenquellenamens oder DataSourceName (DSN)

Ergänzen Sie die lokale Konfigurationsdatei `~/odbc.ini` mit den folgenden zwei DSN-Abschnitten. Im ersten Abschnitt wird der spezielle Treiber 'MySQL Unicode' eingesetzt und der Standard-Port 3306 für MySQL:

```
[dsn_my_mysql]
Description      = MySQL DB `mysql` (Unicode)
Driver           = MySQL Unicode
Servername       = localhost
Port             = 3306
User             = root
Password         = ampw
Database         = mysql
ReadOnly         = No
```

Auf den ODBC-Treiber ``MySQL MDB`` aus der Datei `/etc/odbcinst.ini` greifen Sie mit dieser Konfiguration zu:

```
[dsn_my_mdb_mysql]
Description      = MySQL DB `mysql` (MariaDB)
Driver           = MySQL MDB
Servername       = localhost
Port             = 3306
User             = root
Password         = ampw
Database         = mysql
ReadOnly         = No
```

Für den ANSI-Treiber wird kein Datenquellenname konfiguriert.

Für die Inspektion der ODBC-Treiber können Sie das Konsolen-Programm `odbcinst` einsetzen. Mit den Optionen `-q` und `-d` werden die Namen aller (speziellen) ODBC-Treiber aufgelistet, die sich bisher bei `unixODBC` registriert haben:

```
$ odbcinst -d -q
[PostgreSQL ANSI]
[PostgreSQL Unicode]
[SQLite]
[SQLite3]
[MySQL ANSI]
[MySQL Unicode]
[MySQL MDB]
```

Das Kommando `odbcinst -q -s` dagegen listet alle ODBC-Datenquellenamen (DSN) auf, die Sie in der Konfigurationsdatei `~/odbc.ini` definiert haben:

```
$ odbcinst -q -s
[dsn_pg_postgres]
[dsn_sl3_contacts]
[dsn_my_mysql]
[dsn_my_mdb_mysql]
```

22.9.3.4 Test der ODBC-Konfiguration – MySQL

Für das DBMS MySQL werden wegen der ODBC-Treiber aus unterschiedlichen Paketen zwei Tests ausgeführt. Für die beiden Tests wurde in der Konfigurationsdatei `~/odbc.ini` die (System-)Datenbank ``mysql`` eingetragen, die aber nicht genutzt wird. Statt dessen wird die Version von MySQL abgefragt. Die Kenntnis der Version ist hier zweitrangig, weil es nur um den Test ging, sich erfolgreich über das obdc-fähige Client-Programm `isql` mit einer MySQL-Datenbank zu verbinden.

22.9.3.4.1 Test 1

```
$ isql -v dsn_my_mysql root ampw
+-----+
| Connected! |
+-----+
```

```
| sql-statement  
| help [tablename]  
| quit  
+-----+  
SQL> SELECT version();  
+-----+  
| version()  
+-----+  
| 8.0.23-0ubuntu0.20.04.1  
+-----+  
SQLRowCount returns 1  
1 rows fetched  
SQL> quit  
$
```

22.9.3.4.2 Test 2

```
$ isql -v dsn_my_mdb_mysql root ampw  
+-----+  
| Connected!  
| sql-statement  
| help [tablename]  
| quit  
+-----+  
SQL> SELECT version();  
+-----+  
| version()  
+-----+  
| 8.0.23-0ubuntu0.20.04.1  
+-----+  
SQLRowCount returns 1  
1 rows fetched  
SQL> quit  
$
```

Fazit: Beide Tests verliefen erfolgreich!