

QGIS Application - Bug report #20123

Import to MS SQL failed due to international characters

2018-10-16 04:37 PM - Martin Dobias

Status: Open	
Priority: Normal	
Assignee:	
Category: Data Provider/MSSQL	
Affected QGIS version: 3.3(master)	Regression?: No
Operating System: linux	Easy fix?: No
Pull Request or Patch Applied: No	Resolution:
Crashes QGIS or corrupts data: No	Copied to github as #: 27945

Description

When I load the attached layer to QGIS (dbf encoding is UTF-8), I am unable to import the data to MS SQL through the browser dock. It says the export failed with this output:

Failed to import some layers!

Feature write errors:

Creation error for features from #0 to #0. Provider errors was:

```
[FreeTDS][SQL Server]Incorrect syntax near ','. [FreeTDS][SQL Server]Error converting characters into server's character set.
```

Some character(s) could not be converted QODBC3: Unable to execute statement

Creation error for features from #0 to #0. Provider errors was:

```
[FreeTDS][SQL Server]Incorrect syntax near ','. [FreeTDS][SQL Server]Error converting characters into server's character set.
```

Some character(s) could not be converted QODBC3: Unable to execute statement

Only 0 of 8 features written.

History

#1 - 2018-11-04 12:28 AM - Martin Dobias

TODO: test whether this is unix only problem or affects also windows

There seems to be an issue with how bound values with unicode values are handled somewhere in the stack (QODBC driver / unixodbc / FreeTDS).

Pro-Tip: with FreeTDS it is possible to add "DumpFile=/tmp/freetds.log" to the connection string to get a very detailed log if anyone wants to dig deeper.

This works:

```
q = QSqlQuery(db)
print(q.prepare("insert into [kraje] values (null, N'ěžřůš',pythonX'")
print(q.exec_())
```

This does not work:

```
q = QSqlQuery(db)
print(q.prepare("insert into [kraje] values (null, ?,pythonY'")
q.addBindValue(u"ěžřůš")
print(q.exec_())
```

Interestingly enough, the snippet that does not work still adds a row, just the unicode string value is missing.

Using N'...' instead of binding of values of strings fixes the problem but it feels like a hack - pasting here for completeness:

```
--- a/src/providers/mssql/qgsmssqlprovider.cpp
+++ b/src/providers/mssql/qgsmssqlprovider.cpp
@@ -911,7 +911,10 @@ bool QgsMssqlProvider::addFeatures( QgsFeatureList &flist, Flags flags )
     first = false;

     statement += QStringLiteral( "[%1]" ).arg( fld.name() );
-   values += QStringLiteral( "?" );
+   if ( fld.type() == QVariant::String )
+     values += QStringLiteral( "N'%1'" ).arg( attrs[i].toString() );
+   else
+     values += QStringLiteral( "?" );
  }

  // append geometry column name
@@ -1009,7 +1012,7 @@ bool QgsMssqlProvider::addFeatures( QgsFeatureList &flist, Flags flags )
  else if ( type == QVariant::String )
  {
    // binding a TEXT value
-   query.addBindValue( attrs.at( i ).toString() );
+   //query.addBindValue( attrs.at( i ).toString() );
  }
  else if ( type == QVariant::Time )
  {
```

#2 - 2018-11-04 12:45 AM - Nyal Dawson

I'd try without freetds and try with the official mssql odbc drivers instead, just to narrow down the issue

Files

mssql-fail-kraje.zip	1.28 MB	2018-10-16	Martin Dobias
----------------------	---------	------------	---------------