Character Set and Localization Issues: What encoding is used for files saved by SQLyog?

Starting from SQLyog version 6.0 every file stored by SQLyog is UTF8-encoded. That applies to:

- configuration files and log files (.ini, .log , .err)
- SQL-files generated by the various backup and export functionalities
- SJA job files
- SQL scripts saved from the program's SQL editor (file ..save)

The reason for this is that when such files contain a string with localized/special characters, those characters will always display as what there are: Latin, Cyrillic, Arabic, Chinese of whatever characters!

Also in those situations where:

1) Strings contain characters for a language where no non-unicode implementation/no ANSI codepage exist.

2) Strings contains characters from more than one ANSI codepage

.. using ANSI encoding simply would not work. The first situation applies to dozens (or probably hundreds) of languages - of which several are spoken and written by several million people around the world. That is true for a lot of Indian languages for example. And the second example won't need to be very 'exotic' either: Special characters from Western European languages (ex: French, Spanish) cannot be used with special characters from Central European languages (ex: Polish, Czech) in one ANSI codepage. A unicode encoding (utf8 or utf16) is required!

Also in the situation where

3) Strings contains characters from one ANSI codepage and the actual computer uses a LOCALE where another ANSI codepage is used

.. various problems would arise!

When opening such file in SQLyog editor we will check for the encoding by identifying characteristic byte patterns for UTF8 and the native Windows ('little endian' UTF16) encodings. If characteristic patterns for either are not found ANSI will be assumed. So with SQLyog you can open files generated by SQLyog or any other program.

However 'the other way around' you may need to re-encode the file to ANSI if the program where you want to open a file generated by SQLyog does not support UTF8. A simple way to do this is to open the file in a text editor (like Notepad) and 'save as' and explicitly select ANSI as

the encoding to use. That will work smoothly if neither of the conditions 1) or 2) above prevents it.

Most important it is documented that the 'mysql' command line client on Windows only handles ANSI-encoded files. Please refer to http://bugs.mysql.com/bug.php?id=29323

Unique solution ID: #1132
Author: Peter Laursen
Last update: 2007-06-30 07:29