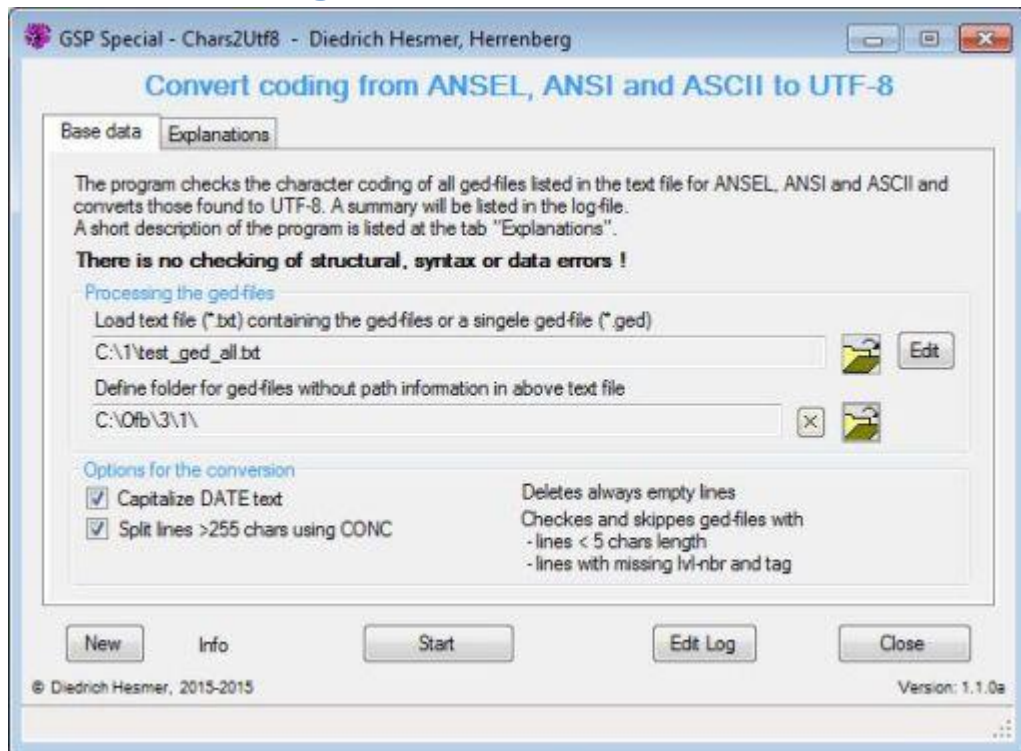




Conversion to UTF-8 coding



Functionality

UTF-8 is the preferred coding for ged-files. This coding allows an unproblematic representation of international characters. Several programs don't support this coding at their export, other programs require UTF-8 coded ged-files during import. Beside UTF-8 the GEDCOM standard draft 5.5.1 allows the use of ASCII, ANSEL and UNICODE coding. The ANSI coding is not allowed according to the standard but still widely used.

This program checks the actual coding of ged-files and converts files with ANSEL, ANSI and ASCII into UTF-8 coding. In one run a single file or a group of files can be converted.

In addition to the conversion, following will be done:

- Empty lines will be deleted.
- Check the existence of the HEADER record and the actual CHARS coding.
- Check short lines and the existence of the level number and the tag in every line.
- Optional long lines > 255 characters can be split and the text following the DATE tag can be converted to upper case.

The converted file will be stored with the filename of the original file and an attached "_u" (xxx.ged will become xxx_u.ged). The original file remains unchanged. The new file will be stored in the folder of the original file. In addition a log-file will be created which can be opened by any text editor. This file contains:

- Statistics of the conversion process.
- Errors found with filename and line number, if possible.



Description

After starting the program, above main screen will show up.

- It is mandatory to define the ged-file(s) to be converted. Two alternatives are available:
 - Defined by a text file containing line by line every ged-file to be checked and converted in the format:
 - Full_path\filename.ged, where full_path starts with the drive letter
 - Filename.ged - in this case the full_path to these files have to be defined at the main screen

This file must have ".txt" as extension. See at the screen the "Load text file ..." textbox and all the ged-files listed in the text file will be converted.

Example of a text file with 4 ged-files:

```
test1.GED
1\die_af\test2.GED
C:\OFB\1\paf\cook\Wilk.ged
C:\OFB\1\die_af\sour\af_dh_td.ged
```

Line 1 + 2 do not have a full path.

- Define directly a single ged-file

This file must have ".ged" as extension and has also to be loaded by the "Load text file ..." textbox and only this ged-file will be converted.

- Optional a folder can be defined to create the full path for files without full path. Only one path can be defined. This path will be put in front of those lines, which do not have a colon (:) in position 2 (line 1 + 2 have no colon in position 2).

If the path "C:\OFB\3\1\" will be defined as in above screen, the 4 lines of above example will be modified to:

```
C:\OFB\3\1\test1.GED
C:\OFB\3\1\1\die_af\test2.GED
C:\OFB\1\paf\cook\Wilk.ged
C:\OFB\1\die_af\sour\af_dh_td.ged
```

- During conversion of every ged-file will be checked for fails described above. Two further options for correcting fails are also available:
 - Text following the DATE tag may be capitalized.
 - Lines with a length > 255 characters may be split using the CONCatenate tag.



Buttons

- [Symbols with an open folder] Load the file resp. the folder.
- [Edit] Edit the ged-file by the standard text editor.
- [X] Clears the text box with the folder name.
- [?] Opens the description.
- [New] Clears all definitions.
- [Info] Shows the folder where the ini-file is stored (contains all settings at the last "Close").
- [Start] Starts the conversion.
- [Edit Log] Opens the log-file by the text editor.
- [Close] Writes the ini-file with all settings and closes the program. Closing the program by the little "x" at the upper right corner of the screen will not write the ini-file.)

Extract of a Log-file:

```
Created at 11.02.2015 21:38:49 - Chars2Utf8 version: 0.0.2
Base: "c:\1\test_ged_all.txt" from 10.02.2015 19:43:27
Following files not found or not ged-files or ends with "_u.ged":
C:\Ofb\3\1\bk\test_f\test2_u.ged
C:\Ofb\3\1\die_af\af-hes-paf_u.ged
C:\Ofb\3\1\die_af\af_dh_test-vn_u.ged
C:\Ofb\3\1\die_af\sour\af_dh_neu_sor_u.ged
...
C:\Ofb\3\1\paf\sour\willy50_u.ged
574 ged-files found in text file
11.02.2015 21:38:49: Start processing
Following files not converted due to errors:
Line nbr. 9390 "1 " to short -> file "C:\Ofb\3\1\ages\richt\hd-ofb9-2014.ged"
Line nbr. 261 "Bad Pyrmon..." digit + tag missing -> file "C:\Ofb\3\1\bk\spiel\2014 (2).ged"
Line nbr. 315 "1 -DET VOG..." digit + tag missing -> file "C:\Ofb\3\1\easy_tree\wink\1-4test-vf.ged"
Line nbr. 22 "3_ZUS" digit + tag missing -> file "C:\Ofb\3\1\gen_p\marzi\miesen.ged"
Unknown CharSet "" in HEAD -> file "C:\Ofb\3\1\gramps\paktig\hesmer_test.ged"
Line nbr. 58 "2 07 MAR 1..." digit + tag missing -> file "C:\Ofb\3\1\paf\karl\ble83.ged"
Line nbr. 366 "... " to short -> file "C:\Ofb\3\1\turck\kb_meinerz.ged"
"0 HEAD" line missing -> file "C:\Ofb\3\1\webtree\utischill\uti_familytree_test-reduzierer_red.ged"

11.02.2015 21:45:37: Conversion completed
253 files skipped - already UTF-8
271 ANSI files converted
46 ANSEL files converted
2 ASCII files converted
46.341.153 Total lines read
46.397.016 Total lines written
```

The upper part lists files not found or no ged-files or already ending with _u.ged. These will not be converted. The middle part lists faulty files with respect to missing HEADER, CHARSET unknown or not supported, line length or missing level number and tag. These will be skipped. The lower part lists a summary of the conversion process.

Note:

During reading the data, there is **no** checking of correct structures and GEDCOM syntax, valid data and consistency of the data of the ged-file. This should be assured prior the conversion by the other GSP programs.

Versions

(N = New, C = corrected, M = modified)

- 1.0.0 - 20.02.2015 - N: Initial publication
- 1.0.1 - 25.02.2015 - N: German UI
- 1.1.0 - 13.03.2015 - M: File name _u
- 1.1.1 - 25.02.2017 - M: File open standardized
- 1.1.2 - 01.07.2017 - M: License function
- 1.1.3 - 10.05.2018 - N: "?" Show Docu