

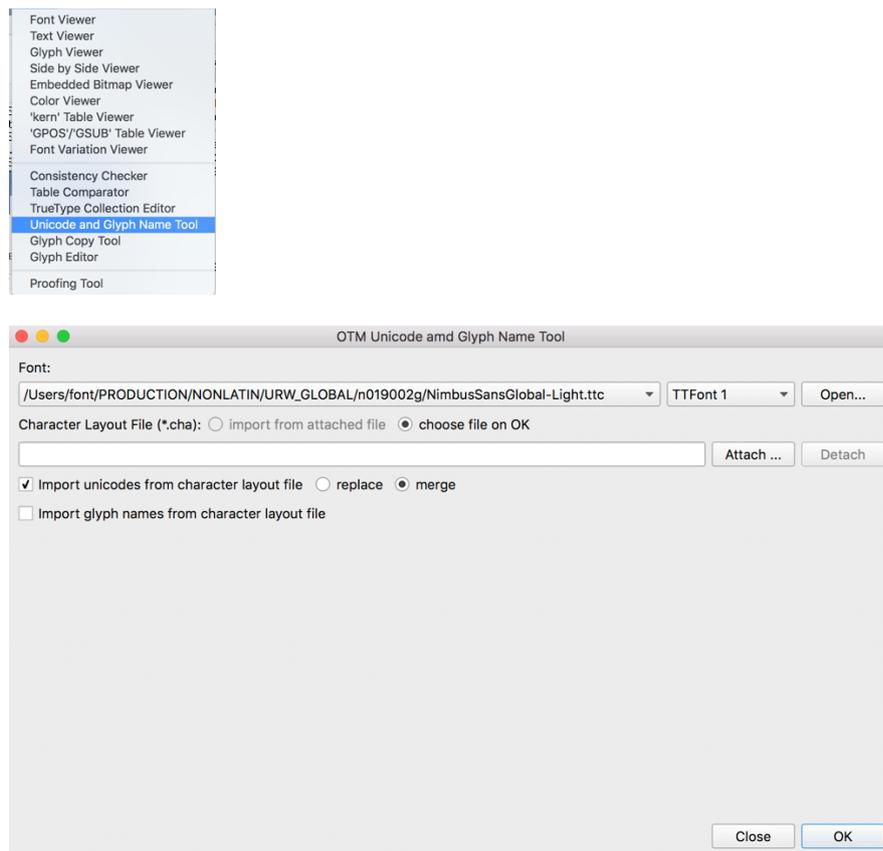
## OTM 8.0 Unicode and GlyphName Tool

The mechanism in TTF/OTF/TTC/OTC fonts to map Unicodes or other encoding schemes to glyph ID's is realized with the 'cmap' table. There are different formats available for the 'cmap' subtables: today the most important are the ones that assign Unicode code points. Most common subtables are the one for Microsoft Unicodes BMP (platform ID 3, encoding ID 1, format 4), and the one for the full Unicode range, which has platform ID 3, encoding ID 10, format 12.

There are often requests to add a specific glyph at a specific Unicode code point, or to change the Unicode mapping of some glyphs in an existing font. This can be done in OTM glyph by glyph, but changing, for example, a complete font family this way can be *quite* time consuming.

Recently the introduction of local glyph variants in the OpenType specification has introduced a new demand for local versions of a font. For example, in the Cyrillic script there are some glyphs that have a different form in Russia and in Bulgaria. With OpenType Layout Features one can switch between the two local variants easily, but some applications cannot handle these features. Furthermore, some users might prefer fonts which have the local forms as a default. Since the font contains both variants, this requires only the change of the Unicode cmap.

In order to enable easy altering and editing of the 'cmap' table we have introduced a new tool:



The tool only supports the modification of Unicode-based 'cmap' subtables, not other encodings, such as JIS or GB or KSC.

The 'import glyph names [...]' option allows to overwrite glyph names either in the 'post' table in case of TTC/TTF fonts, or in the CFF glyph list for OTF/OTC fonts.

The tool allows to import Unicode code points, as well as glyph names from a text file. The text file must have the structure of a urw Character Layout file (.cha): this is a csv- file with some specific information about the columns. The easiest way to create such a file, is simply exporting it from the font with the 'Export..' option in the 'File' menu of OTM.

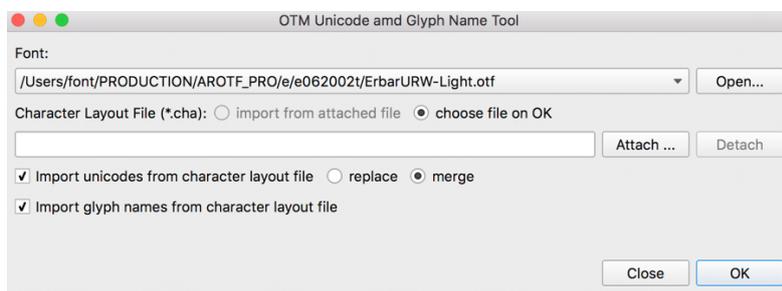
Here is an example:

```
Version 002.000
Starttable
GlyInd;URWNum;UNINum;Width;PSName
0;418;;250;.notdef
1;1;x0020;250;space
2;2;x0021;306;exclam
3;3;x0022;354;quotedbl
4;4;x0023;503;numbersign
5;5;x0024;480;dollar
.....
277;277;x017f;201;uni017F
.....
411;411;;516;amacron.salt
412;412;;427;umacron.salt
413;413;;427;uogonek.salt
414;414;;588;Adieresis.salt1
415;415;;608;Adieresis.salt2
416;416;;808;Odieresis.salt
417;417;;581;Udieresis.salt
Endtable
```

If you want to change the name of a glyph, for example 'UNI017F' to 'longs', you can simply edit the exported .cha file:

```
.....
277;277;x017f;201;longs
.....
```

and click on the 'ok' button. You will be then asked via a standard file dialogue to select your text file.



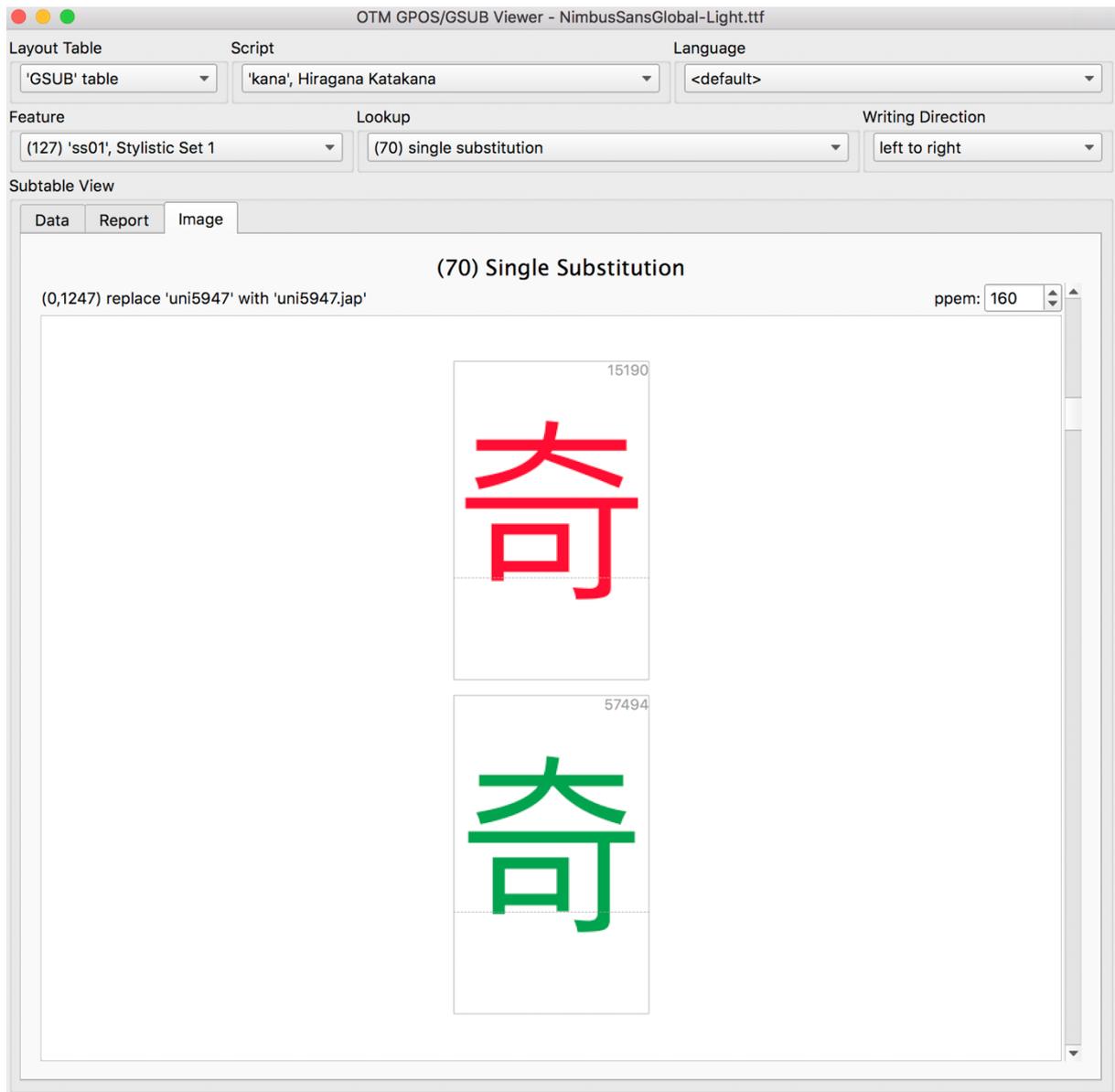
There are two options:

- If you choose 'replace' the cmap will be reconstructed from the text file. All entries in the existing cmap will be lost.
- If you choose 'merge' only the entries in the text file will be modified or added: the existing cmap will still have all other non-changed entries.

**Example:** creating a TTC file with local font versions:

The second option 'merge' makes it easy to create local font variants. The following demonstrates how this works with a global font.

We start with a font that has about 65400 glyphs. The font contains simplified Chinese, traditional Chinese, and Japanese glyphs –many of them are sharing the same Unicode code points. The default setting is that the standard glyphs are having the simplified Chinese forms, whereas the traditional Chinese and the Japanese forms are implemented as stylistic sets and locale variants as well, and which can be accessed via OpenType Layout Features :



In the .cha file we use the convention to name the stylistic alternates with their Unicode code point, followed by the abbreviation for the local variant, for example ' uni5947. jap' . These glyphs will also be located in the PUA at U+E5947.

We can now easily generate a file, which maps the Japanese variants to the correct Unicode values. This will look as follows:

Version 002.000

Starttable

GlyInd;URWNum;UNINum;Width;PSName

56245;56245;x301c;2048;uni301C.jap

56246;56246;x301f;2048;uni301F.jap

56249;56249;x3402;2048;uni3402.jap

56250;56250;x3406;2048;uni3406.jap

56251;56251;x342c;2048;uni342C.jap

56252;56252;x342e;2048;uni342E.jap

56253;56253;x346a;2048;uni346A.jap

56254;56254;x34bc;2048;uni34BC.jap

56255;56255;x34c1;2048;uni34C1.jap

56256;56256;x3563;2048;uni3563.jap

.....

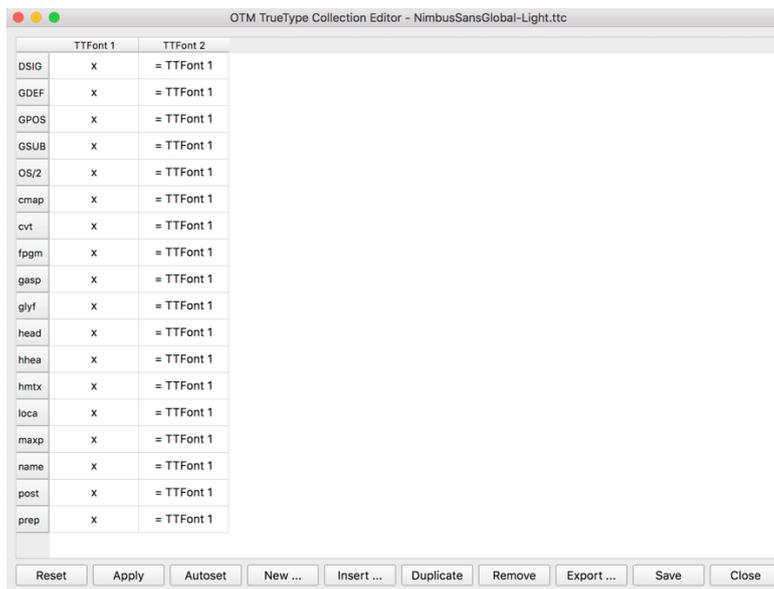
57494;57494;x5947;2048;uni5947.jap

.....

Endtable

We can generate a TTC file now with the following steps:

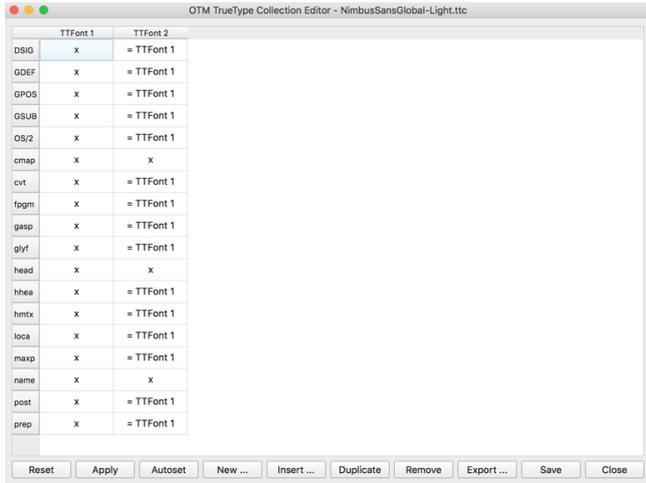
1. Create an empty TTC file NimbusSansGlobal-Light.ttc with the 'OTM TrueType Collection Editor'.
2. Import the NimbusSansGlobal-Light.ttf font.
3. Duplicate this TTF.



4. Import the text file with the reassigned Unicode code points and change the cmap of font 2.
5. Edit the name table of font 2.

Now we have created a TTC file with two fonts, sharing the 'glyf' table and being different only in the Unicode mapping and the naming. This allows now to access the Japanese glyphs in the second font as standard without invoking any OpenType Layout Features.

TTX File with different cmap (encoding) and different 'name' table:



Here is the default glyph for Unicode u+5947 in font 1 and font 2 in the TTX file:



Naming table for the two fonts:

Table Lengths	'head' entries	'hhea' entries	'vhea' entries	'OS/2' entries	'maxp' entries	'post' header	'CFF' top dictionary
[p,e];[id]	NimbusSansGlobal-Light.ttc#0		NimbusSansGlobal-Light.ttc#1		Description		
0 [1;0;0;0]	Copyright 2018 by URW Type Foundry GmbH	Copyright 2018 by URW Type Foundry GmbH	Copyright 2018 by URW Type Foundry GmbH	Macintosh; Roman; English; Copyright notice			
1 [1;0;0;1]	NimbusSansGlobal	NimbusSansGlobalJP	NimbusSansGlobalJP	Macintosh; Roman; English; Font Family name			
2 [1;0;0;2]	Light	Light	Light	Macintosh; Roman; English; Font Subfamily name			
3 [1;0;0;3]	URW:NimbusSansGlobal-Light:2018	URW:NimbusSansGlobalJP-Light:2018	URW:NimbusSansGlobalJP-Light:2018	Macintosh; Roman; English; Unique font identifier			
4 [1;0;0;4]	NimbusSansGlobal Light	NimbusSansGlobalJP Light	NimbusSansGlobalJP Light	Macintosh; Roman; English; Full font name			
5 [1;0;0;5]	Version 8.50	Version 8.50	Version 8.50	Macintosh; Roman; English; Version string			
6 [1;0;0;6]	NimbusSansGlobal-Light	NimbusSansGlobalJP-Light	NimbusSansGlobalJP-Light	Macintosh; Roman; English; Postscript name			
7 [1;0;0;16]	NimbusSansGlobal	NimbusSansGlobalJP	NimbusSansGlobalJP	Macintosh; Roman; English; Preferred Family			
8 [1;0;0;17]	Light	Light	Light	Macintosh; Roman; English; Preferred Subfamily			
9 [3;1;1033;0]	Copyright 2018 by URW Type Foundry GmbH	Copyright 2018 by URW Type Foundry GmbH	Copyright 2018 by URW Type Foundry GmbH	Microsoft; Unicode BMP only; English, United States; Copyright notice			
10 [3;1;1033;1]	NimbusSansGlobal Light	NimbusSansGlobalJP Light	NimbusSansGlobalJP Light	Microsoft; Unicode BMP only; English, United States; Font Family name			
11 [3;1;1033;2]	Regular	Regular	Regular	Microsoft; Unicode BMP only; English, United States; Font Subfamily name			
12 [3;1;1033;3]	URW:NimbusSansGlobal-Light:2018	URW:NimbusSansGlobalJP-Light:2018	URW:NimbusSansGlobalJP-Light:2018	Microsoft; Unicode BMP only; English, United States; Unique font identifier			
13 [3;1;1033;4]	NimbusSansGlobal Light	NimbusSansGlobalJP Light	NimbusSansGlobalJP Light	Microsoft; Unicode BMP only; English, United States; Full font name			
14 [3;1;1033;5]	Version 8.50	Version 8.50	Version 8.50	Microsoft; Unicode BMP only; English, United States; Version string			
15 [3;1;1033;6]	NimbusSansGlobal-Light	NimbusSansGlobalJP-Light	NimbusSansGlobalJP-Light	Microsoft; Unicode BMP only; English, United States; Postscript name			
16 [3;1;1033;16]	NimbusSansGlobal	NimbusSansGlobal	NimbusSansGlobal	Microsoft; Unicode BMP only; English, United States; Preferred Family			
17 [3;1;1033;17]	Light	Light	Light	Microsoft; Unicode BMP only; English, United States; Preferred Subfamily			