

(URW)⁺⁺

Dutch Type Library

OTM 2.4 Addendum



's-Hertogenbosch/Hamburg

2011



Typography means more than bringing order to the passing on of information; it means elevating to the sublime the mould in which the process of passing on is cast.

Frank E. Blokland

Limited user rights

You may never use DTL font production software to edit data of which you do not own the rights, including but not limited to intellectual property rights, copyright and trademark rights, unless the rightful claimant has given his written and signed consent.

Export...

The export of the OpenType layout tables from a font opened in OTM has been much improved. As described on page 10 of the DTL OTMaster manual, dumping of the binary layout tables involves interpretation. Also on the same page of the manual one can read that the OTM 2.3 user is '[...] advised to read and possibly correct exported feature files, especially if a font's layout behavior is complex: OTMaster's AFDKO-syntax dump is not perfect yet'.

In OTM 2.4 the interpretation of the OpenType layout tables has been meliorated and editing is not required anymore; the exported features can be read-in without any alterations. And although the way the exported features are listed could be different (due to the inevitable interpretation) from the features-description that was used for generating the font originally, the functionality should be identical.

```
feature onum {
  sub @FIG_TAB_LINING by @FIG_FIT_OLDSTYLE;
} onum;
```

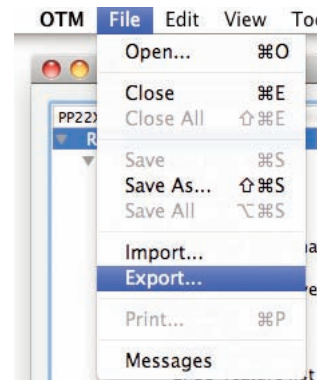
Feature description as used for generating the original font data (the glyph classes are defined in the file also, of course).

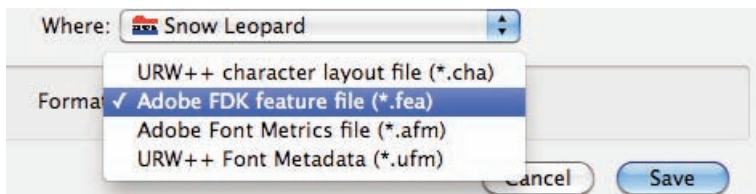
```
*****
# Feature "onum": Oldstyle Figures
# used in script: latn
# from a lookup of type GSUB_LOOKUP_TYPE_SINGLE
*****
lookup GSUB_LOOKUP_00005
{
  lookupflag 0;
  sub \zero by \zerooldstyle;
  sub \one by \oneoldstyle;
  sub \two by \twooldstyle;
  sub \three by \threeoldstyle;
  sub \four by \fouroldstyle;
  sub \five by \fiveoldstyle;
  sub \six by \sixoldstyle;
  sub \seven by \sevenoldstyle;
  sub \eight by \eightoldstyle;
  sub \nine by \nineoldstyle;
} GSUB_LOOKUP_00005;
```

The same feature information as exported by OTM 2.4.

To export the features choose **File > Export...** and subsequently the 'Select file to export' dialog will show up, in which the 'Adobe FDK feature file (*.fea)' option can be selected (besides, of course, the target directory).

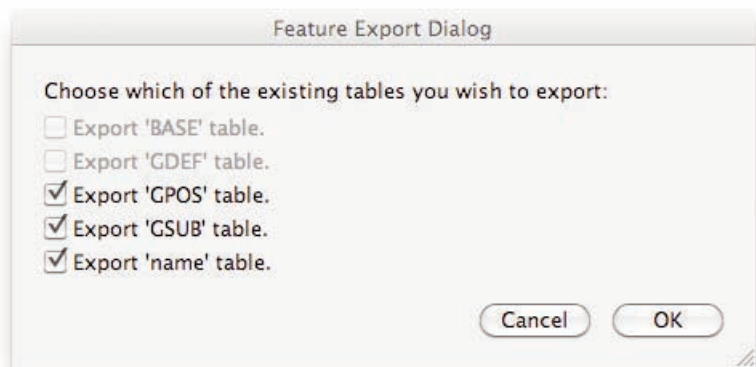
*Choose the **Export...** function for exporting OpenType Layout features.*





The 'Select file to export' dialog.

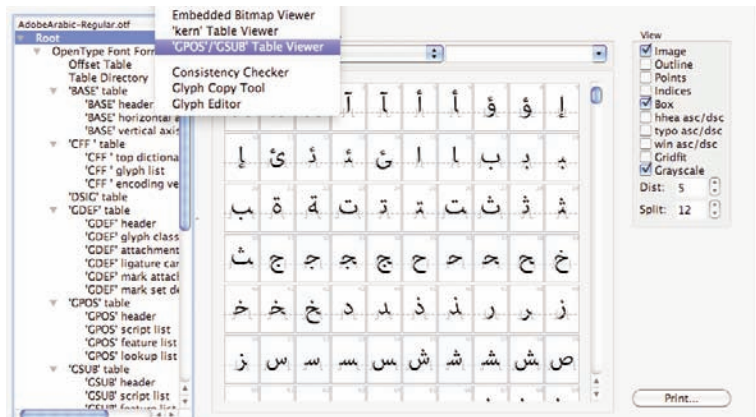
Next the 'Feature Export Dialog' will show up asking which of the tables present in the font have to be exported.



The 'Feature Export Dialog'.

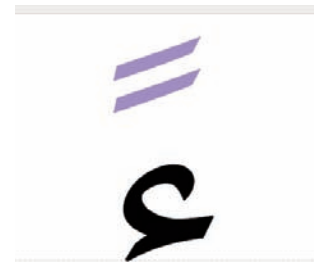
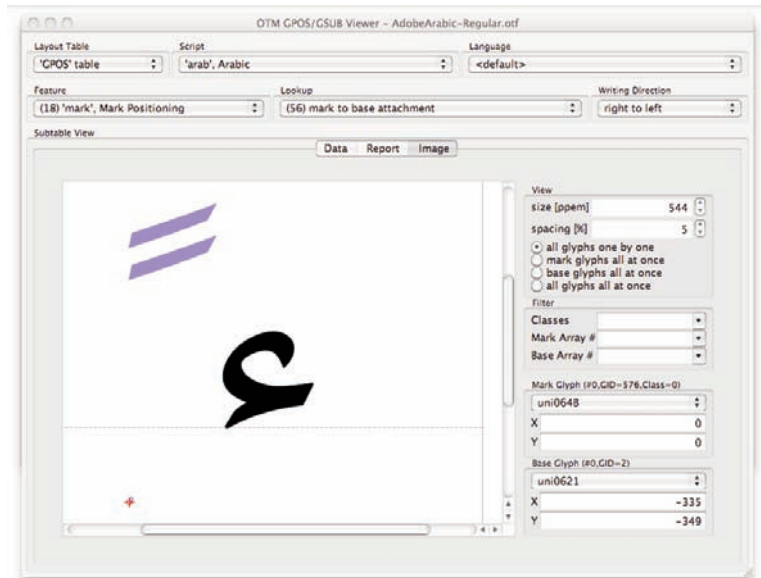
The exported features files can be modified using a text editor. Altered features files can be imported via **File > Import...** function.

Any OpenType layout features file with the AFDKO syntax can be imported, even if the listed features are not covered by the font's character set. The modified Hatch OpenType ('HOT') tool will remove the non-covered features during font generation. Of course, to make all applicable features work, the character naming convention should be identical in the font and in the OpenType layout features file.



The OpenType layout features can be checked and (partly) edited via the **Tools > 'GPOS'/'GSUB' Table Viewer**.

One can use the **Import...** features functionality for instance for the positioning of marks in Arabic fonts. A standardized features file containing the necessary information can be imported and subsequently the mark attachment points can be edited interactively or numerically via the **Tools > 'GPOS'/'GSUB' Table Viewer** for the correct positioning of the (mark to) marks.



Mark positioning can be done interactively or numerically.

Note on the export of URW++ Font Metadata (UFM)

The UFM file contains the font metadata information, i.e. info on font naming and vertical metrics, but also info on matters such as weight, style and embedding. This text file is used in a FontMaster or IKARUS based workflow together with a glyph database (BE [Bezier] or IK [IKARUS] format), kerning file (BE or kern.fea), PAR file (binary file for the storage of some additional information) and an OpenType layout features file.

Currently the vertical metrics are recalculated in the exported UFM file to an em of 15000 units, which is the default for the IKARUS format. The naming information stored in the exported UFM file is relatively limited and the **name** records in the **name** table can contain (many) more entries. This limitation can simply be circumvented by first exporting the OpenType layout features from the font as described above and subsequently copying the **name** table information in the UFM file. The only thing that subsequently has to be changed in the copied **name** table information is the 'nameid' entry, with which each line starts; this has to be changed into 'TTName'. Any resulting duplicate naming entry in the original UFM file will be superseded by the info in the 'TTName' string.

In the 'post' header, the 'formatType' version number can be changed.

Type	Name	Value	Comment
Fixed	formatType	0x00020000	0x00010000 for version 1.0.
Fixed	italicAngle	0.000	Italic angle in counter-clockwise degrees from the vertical.
FWORD	underlinePosition	-293	Distance top of underline from baseline (negative = below baseline).
FWORD	underlineThickness	41	Suggested value for the underline thickness.
ULONG	isFixedPitch	0	Set to 0 if the font is proportionally spaced.
ULONG	minMemType42	0	Minimum memory usage when downloaded.
ULONG	maxMemType42	0	Maximum memory usage when downloaded.
ULONG	minMemType1	0	Minimum memory usage when downloaded as a Type 1.
ULONG	maxMemType1	0	Maximum memory usage when downloaded as a Type 1.

'post' table

The **post** table contains information to enable printing of TrueType fonts on PostScript-printers. In the **post** table the *formatType* entry in the **post** header can be changed in OTM 2.4 from format 2 to format 3 and vice versa.

Changing from format 2 to 3 will remove the **post** names entries.

Doing the opposite will generate a **post** table set of glyph names. Also format 1 and 4 can be entered here, but this will mess up things, while this function will not re-encode the font in question in any manner.

Type	Name	Value	Comment
USHORT	version	0	Version number (0 = without, 1 = with ClearType).
USHORT	numRanges	3	Number of records to follow.

In the 'gasp' header, the table version number (0 or 1) can be changed.

'gasp' table

In the **gasp** table > **gasp** header the *version* entry can be changed between 0 and 1. As can be read in the OpenType specs, the **gasp** table '[...]' contains information that describes the preferred rasterization techniques for the typeface when it is rendered on grayscale-capable devices'. In other words, the info in the **gasp** table controls the rendering of type on the screen.

There are two table versions, namely '0' and '1'. Version number 'zero' is the oldest one and controls 'standard' grid-fitting in combination with anti-aliasing. Version 'one' enables optimized settings for the ClearType rasterizer, i.e. it makes tuning for subpixel rendering possible (symmetric smoothing in combination with symmetric grid fitting).

OTF Consistency Checker

The OTF Consistency Checker has a new entry, namely **Language**. This function can be used to check the coverage by a font of scripts and languages.



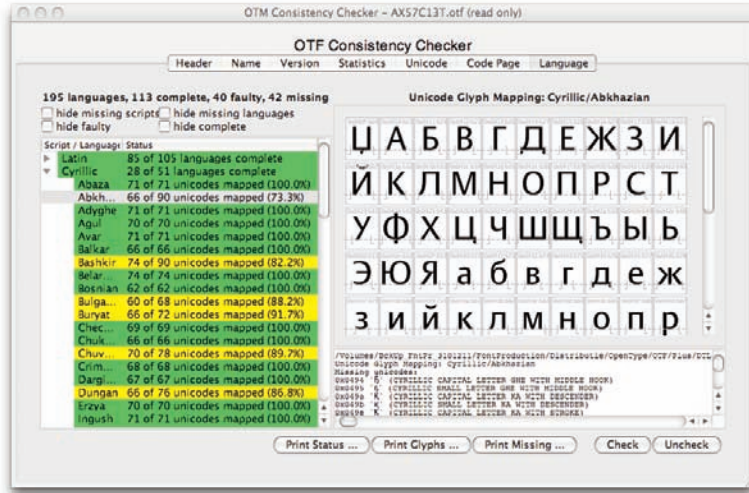
The **Language** entry of the OTF Consistency Checker.

To start the function, click on the **Check** button. On the left half of the dialog the supported scripts will be listed. Selecting a certain script will result in a listing of (partly) supported languages. Selecting a language of which not all characters are covered, will result in a listing of the ones that are missing.

There are four check boxes which control the forenamed listing:

- **hide missing scripts**, which result in a listing of only the scripts covered by the font.
- **hide faulty**, which will show only the scripts within a script that are covered for 100 percent. Languages that are completely covered are indicated by a green ribbon. Languages that are ‘faulty’ are indicated by a yellow ribbon. Languages that are only partly covered are indicated by a red ribbon, which has to be interpreted as ‘very faulty’ and these cannot be hidden with this checkbox.
- **hide missing languages**, which results in the showing of the languages within a script which are (partly – depending on the use of the other check boxes) covered.
- **hide complete**, which does the opposite of the **hide faulty** check box.

A listing of supported languages by a font is presented per script.



Selecting the **Print Status...** button will send the current information (depending on the check box settings) shown on the 'Script|Language | Status' part of the dialog to the selected printer.

Selecting the **Print Glyphs...** button will send an overview of all the characters in the font (including info on the Unicode code points) to the selected printer.

Selecting the **Print Missing...** button will send an overview of all the characters missing in the font for a certain language to the selected printer.

Unicode Glyph Mapping: Cyrillic/Abkhazian
Missing unicodes:
0x0494 'Ҕ' (CYRILLIC CAPITAL LETTER GHE WITH MIDDLE HOOK)
0x0495 'ҕ' (CYRILLIC SMALL LETTER GHE WITH MIDDLE HOOK)
0x049a 'Ґ' (CYRILLIC CAPITAL LETTER KA WITH DESCENDER)
0x049b 'қ' (CYRILLIC SMALL LETTER KA WITH DESCENDER)
0x049e 'Җ' (CYRILLIC CAPITAL LETTER KA WITH STROKE)
0x049f 'җ' (CYRILLIC SMALL LETTER KA WITH STROKE)
0x04a6 'Җ' (CYRILLIC CAPITAL LETTER PE WITH MIDDLE HOOK)
0x04a7 'җ' (CYRILLIC SMALL LETTER PE WITH MIDDLE HOOK)
0x04a8 'Ҙ' (CYRILLIC CAPITAL LETTER ABKHASIAN HA)

An overview of missing glyphs can be sent to a printer.

The language-functionality in the OTF Consistency Checker was partly based on several sources from the internet, including Michael Everson's site *The alphabets of Europe* (<http://www.evertype.com/alphabets>), www.geonames.de, www.omniglot.com, and, of course, *Wikipedia*.

OTM 2.4 Addendum

Edition 2.4/2011; supports DTL OTMaster version 2.3
 © 2011 Dutch Type Library and URW++ Design & Development
Text & Design FEB
Typeset in DTL Argo, DTL Haarlemmer and DTL Haarlemmer Sans

Dutch Type Library

Zwaenestede 49
 5221 KC 's-Hertogenbosch
 The Netherlands
phone + 31 (0)73 614 95 36
fax + 31 (0)73 613 98 23
e-mail info@dutchtypelibrary.com
website www.dtl.nl and www.fonttools.org

URW++ Design & Development GmbH

Poppenbütteler Bogen 36
 22399 Hamburg
 Germany
phone + 49 (0)40 60 60 52 28
fax + 49 (0)40 60 60 51 11
e-mail info@urwpp.de
website www.urwpp.de

The third party product names used in the DTL OTMaster manual are for identification purposes only. All trademarks and registered trademarks are the property of their respective owners. The following trademarks may or may not be marked in this manual:

OpenType is either a registered trademark or trademark of Microsoft Corporation in the United States of America and/or other countries.

PostScript is a registered trademark of Adobe Systems Incorporated.

TrueType is a trademark of Apple Computer, Incorporated.

Adobe is a registered trademark of Adobe Systems Incorporated.

Apple and Macintosh are registered trademarks of Apple Computer, Incorporated.

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States of America and/or other countries.

Other company, product, and service names occasionally or incidentally mentioned in this manual may be trademarks or service marks of others.