



FontCreator 9.1

© 1997 - 2016 by High-Logic B.V. All rights reserved.

FontCreator Manual

© 1997 - 2016 by High-Logic B.V. All rights reserved.

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

FontCreator is a trademark of High-Logic.

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

Apple, the Apple Logo and Macintosh are registered trademarks and TrueType is a trademark of Apple Computer, Inc. registered in the United States and other countries.

Adobe and PostScript are trademarks of Adobe Systems Incorporated which may be registered in certain jurisdictions.

All other trademarks and registered trademarks are the sole property of their respective owners. The Unicode Character Database is provided as is by Unicode, Inc.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

February 2016

Table of Contents

Part I Getting Started

1 Welcome to FontCreator 9.1	2
2 What's New in FontCreator 9.1	3
3 Technical Support	3
4 Registration	4
5 Credits	4

Part II Quick Start Tutorial

1 Your First Font -- a Brief Tutorial	7
2 Edit a Glyph - Character 3	7
3 Edit a Glyph - Character A	8
4 Edit a Glyph - Character B	9
5 Test and Install the Font	10

Part III About Fonts

1 TrueType	14
2 OpenType	14
3 Color Extension	14
4 Web Open Font Format (WOFF)	15
5 Font Copyright	15

Part IV Opening, Creating and Saving Fonts

1 Working with Font Projects	17
2 Open a Font	17
3 Create a Font Project	19
4 Save a Project	20
5 Export a Font	20
6 Close a Font	22

Part V Editing Fonts

1 Editing Options	24
Undo Command	24
Redo Command	24
Repeat Your Last Action	24
Finding a Glyph	24
Tags	25
2 Editing Fonts	25
Using the Font Overview	25

Used By	26
Insert Characters	27
Insert Glyphs	29
Copy and Paste Glyphs	30
Adding a New Character - the EURO Sign	31
Font Name	32
Font Type	32
Font Embedding	32
Monospaced versus Proportional	33
Unicode versus Symbol	33
Recommended Glyphs	34
3 Editing Glyphs	37
Introduction	37
Empty Glyphs	38
Simple Glyphs	38
Introduction.....	38
Contours.....	41
Points	43
Join and Split Contours.....	44
Freedraw.....	45
Composite Glyphs	46
Introduction.....	46
Add Glyph Member.....	48
Glyph Member Properties.....	48
Complete Composites.....	50
Glyph Metrics	50
Color Glyphs	51
Introduction.....	51
Add Glyph Member.....	52
Palettes and Colors.....	53
4 Format	54
Properties	54
Identification.....	54
Extended.....	56
Legal	59
Custom	61
General	62
Overview	62
Font Header Flags.....	65
Metrics	67
Ranges	70
Overview	70
Unicode Character Range.....	72
Code Page Character Range.....	72
Characteristics.....	74
Smoothing.....	75
Overview	75
Insert Range	76
Font Information	77
Calculated Fields.....	77
Unsupported Tables.....	79
OpenType Layout Features	80
Introduction.....	80

Supported types of substitution and positioning.....	81
OpenType Designer.....	83
OpenType Designer.....	83
Class Manager.....	87
Subtable Manager.....	88
Designer Settings.....	89
Autokern Settings.....	90
Substitutions	91
Single Adjustment.....	91
Pair Adjustment.....	92
Marks	94
Cursive Attachment.....	95
Anchor Manager.....	96
Chained Context.....	97
Feature Parameters.....	97
Script Editor.....	100
OpenType Layout Feature Code Editor.....	100
Script Syntax.....	101
Basics	101
Comments	101
Feature	101
Class	102
Language	102
Lookup	103
Lookupflags	104
Subtable	104
Script	105
Sub	105
Pos	106
Examples and Help.....	109
Glyph substitutions.....	109
Automatic OpenType Layout Features.....	109
5 Tools	109
Glyph Transformer	109
AutoMetrics	111
Metrics	111
Glyph Selection.....	113
AutoKern	114
Setup	114
Kern	116
Import Images	116
Import Raster Image.....	116
Import Vector Image.....	118
Sorting Glyphs	121
Generate Glyph Names	122
TrueType Collection	122
External Tools	123
6 Printing	123
Print Font	123
Print Glyph	124
7 Font Validation	124
Setup	124
Results	127

8 Testing and Installing Fonts	127
MainType	127
International Keyboard	127
Test Your Font	128
Test TrueType/OpenType.....	128
Test Font - Edit Text Samples.....	129
Test Web font.....	129
Installing Fonts	130
Character Map	131

Part VI Toolbars

1 Overview	134
2 User Notes	134
3 Transform	135
4 Glyph Properties	135
Glyph Properties	135
Select Character	137
5 Validation	138
6 Preview	141
7 Background Image	142
8 Comparison	143
9 Samples	143
10 Color Glyph Members	145
11 Palette	145
12 Anchors	146

Part VII Customizing FontCreator

1 Options	150
General	150
Overview	150
Glyph	151
Preview	152
Naming	153
Validation	154
Advanced	155
2 Grid Options	157
3 Metrics Options	158
4 Guidelines	159
Options	159
Guideline	160
5 Keyboard Shortcuts	161
6 FontCreator Data Files	165

Index

167

Part



1 Getting Started

1.1 Welcome to FontCreator 9.1

Introduction

FontCreator allows you to create and edit TrueType, OpenType and Web Fonts.

The editor lets you easily select and modify the entire character set. Features include the ability to convert images to outlines, thus enabling you to create fonts with your own signature, logo and handwriting.

The intuitive interface makes FontCreator the perfect tool for both new and experienced users. The advanced validation features make the design process easy and help you to avoid common mistakes.

The OpenType Designer allows you to easily add OpenType Layout Features to your fonts.

Key features

- Create and edit TrueType and OpenType fonts
- Opening and exporting of Web fonts
- OpenType Layout Features
- Redesign existing characters
- Add missing characters
- Convert vector and raster based images (e.g. a signature, logo or handwriting) to character outlines
- Edit or regenerate font names
- Fix character mappings
- Generate, modify and clean up kerning pairs
- Correct fonts that display incorrectly
- Add or correct composite glyphs
- Transform individual glyphs or an entire font (e.g. to make a bold version)
- Split TrueType Collection or Extract TrueType Fonts from TrueType Collection
- Preview fonts before installing
- Install fonts in Windows

Make sure you have the latest version of FontCreator:

<http://www.high-logic.com/> 

1.2 What's New in FontCreator 9.1

New features in this version of FontCreator include:

- OpenType Designer which allows you to visually edit OpenType Layout Features including, but not limited to:
 - OpenType class based kerning
 - Mark to Base, Mark to Mark, Mark to Ligature, and Curative attachments
 - Substitutions like ligatures and alternates
 - Feature parameters for character variants, optical size, and stylistic sets
 - Anchor Manager
- Import of Visual OpenType Layout (VOLT) projects
- Unicode 8 support
- New settings added to the web font test page
- Copy glyph outline as vector data to clipboard
- Optimised font export
- Improved optical metrics
- Enhanced auto kerning

A complete list of current and previous changes can be found online:

<http://www.high-logic.com/fontcreator/history.html> 

1.3 Technical Support

Printed User Manual

If you prefer reading printed manuals, a PDF version of the entire help file is available through the main menu (**Help** -> **User Manual**).

Online User Manual

The user manual is also available online:

<http://www.high-logic.com/fontcreator/manual9/index.html> 

Forum

The forum is available to you for support and information about managing and designing fonts. This forum has become a place where all users of FontCreator can share their knowledge. Membership of the forum is free. There's a good chance other people have asked the same questions as you, so you may be able to find the answers you need. Feedback and suggestions are also welcome in the forum.

<http://forum.high-logic.com/> 

Frequently Asked Questions

View the most frequently asked questions (and answers) about FontCreator here:

<http://www.high-logic.com/fontcreator/faq.html> ↗

1.4 Registration

Evaluation

During the trial period, FontCreator runs in **Professional Edition** mode and most features are enabled. After using FontCreator for a 30-day trial period, you must register and pay for it, or remove it from your system. Registering makes your copy legal and helps support our efforts to develop innovative products to best serve your needs. Thank you for your support of FontCreator!

Why Register?

Registration allows you to continue using FontCreator 9.1 and entitles you to the following benefits:

- All future versions of FontCreator 9.x.
- Direct support from the developers.

Three Editions

FontCreator is available in three editions: a home edition for personal use which can't be used for commercial use, a standard edition that will suit most font designers, and a professional edition to get the most out of your fonts. To view the differences between the FontCreator editions see the comparison chart:

<http://www.high-logic.com/font-editor/fontcreator/comparison.html> ↗

To Register

The quickest and most convenient way to register is online via credit card. Other supported payment methods like checks and cash are also supported. For more details take a look at our registration page:

<http://www.high-logic.com/register.html> ↗

1.5 Credits

FontCreator contains several unmodified libraries that are available under the MPL.

- VirtualShellUtilities from <http://www.mustangpeak.net/>;
- Virtual Treeview from <http://soft-gems.net/>;
- ADOM from <http://philo.de/xml/>
- SynEdit from <http://synedit.sourceforge.net>

FontCreator also uses an unmodified version of TTFAutohint which is available under the FTL. The FreeType Project (www.freetype.org).

Part



2 Quick Start Tutorial

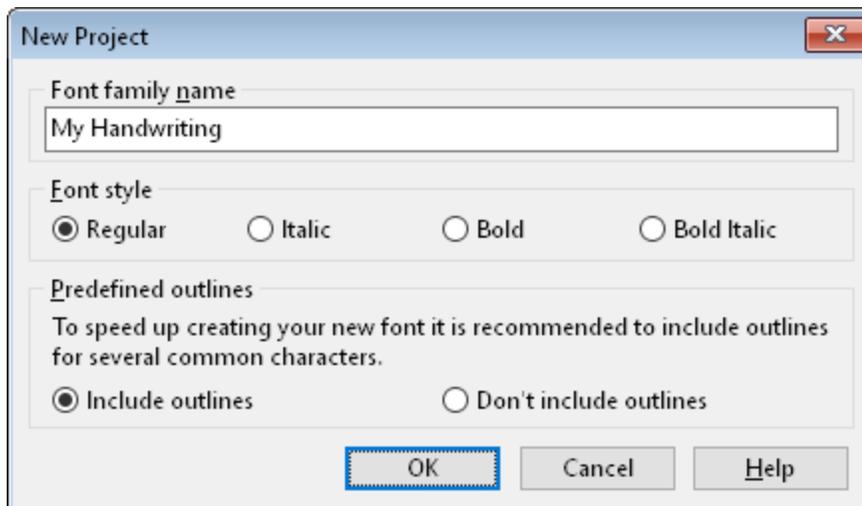
2.1 Your First Font -- a Brief Tutorial

The quickest way to introduce yourself to FontCreator is to make a font. To make it personal we will describe how to make a font of your own handwriting.

On the **File** menu, click **New Project** to start a new font project and create a font.

It is important to give our handwriting font a personal name, otherwise we will not be able to identify (and install) the font properly. The font name is not the same as the filename (for example **timesbd.ttf** has **Time New Roman Bold** as the font name, where Times New Roman is the font family name and Bold is the font style).

To give our font a name we type "My Handwriting" in the Font family name field and press the **OK** button.

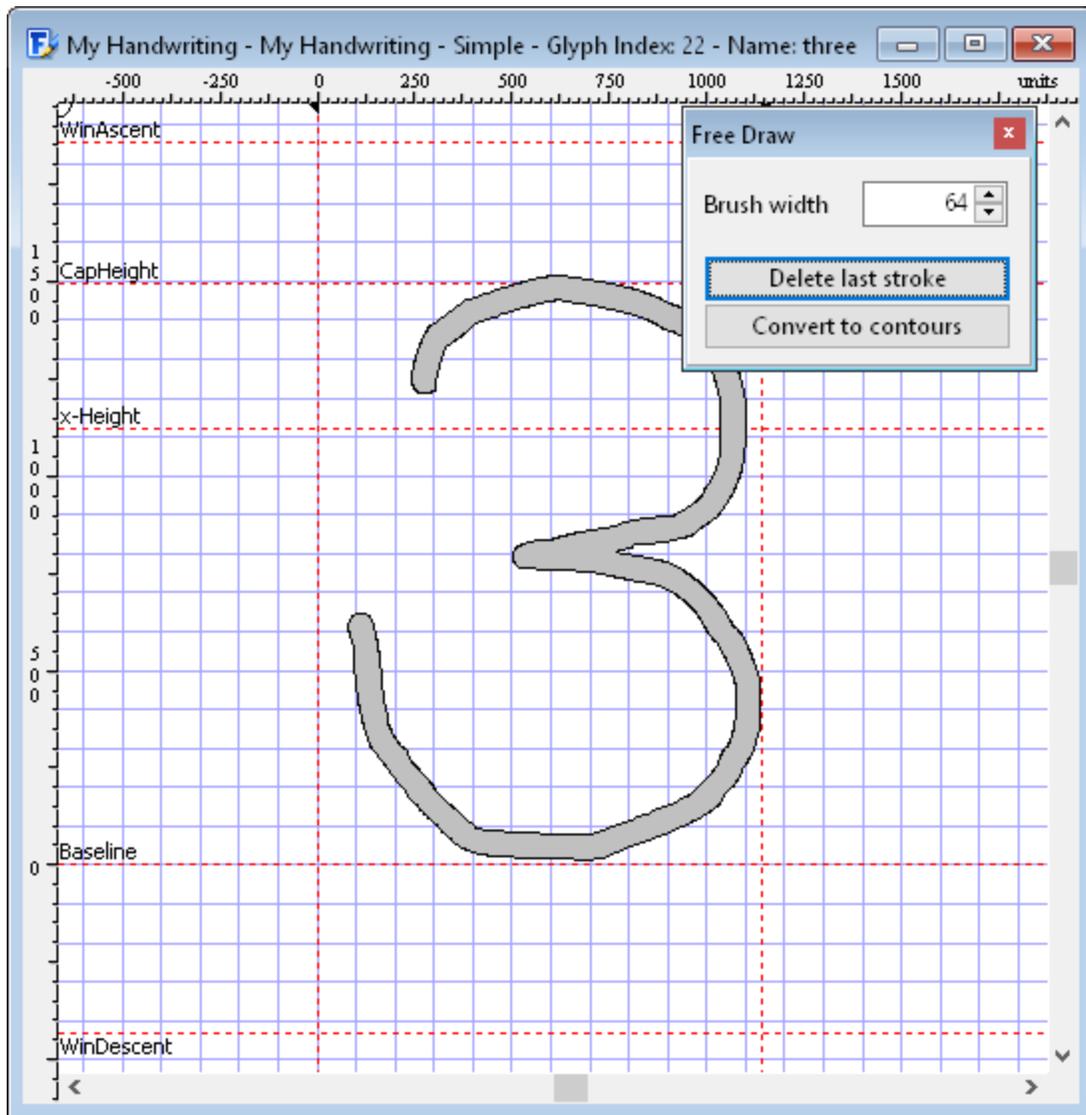


Now you will see the **Font Overview** window. All cells have a caption and to give you a visual guide, most cells contain a sample character shown as light grey outlines. Sample characters don't really exist in the font, you have to add glyph data yourself. In this tutorial we will add this data for glyphs mapped to characters 3, A and B.

2.2 Edit a Glyph - Character 3

From the **Font Overview** window we double-click on the cell with the caption "3". Now we see the corresponding **Glyph Edit** window. Choose **Freedraw Contours** in the **Insert** menu. During this operation a left mouse-click draws and a right-click erases .

Press the **Convert to contours** button to convert the FreeDraw image to contours.



The contours should stay within the visible area, this area is defined by the WinAscent and WinDescent reference lines and the left and right bearings (vertical dashed lines). Make sure the glyph is within this area, parts outside this area will most likely be invisible. The left and right bearings can be changed by dragging them to their desired position. The glyph should be positioned between these lines, otherwise characters will overlap each other.

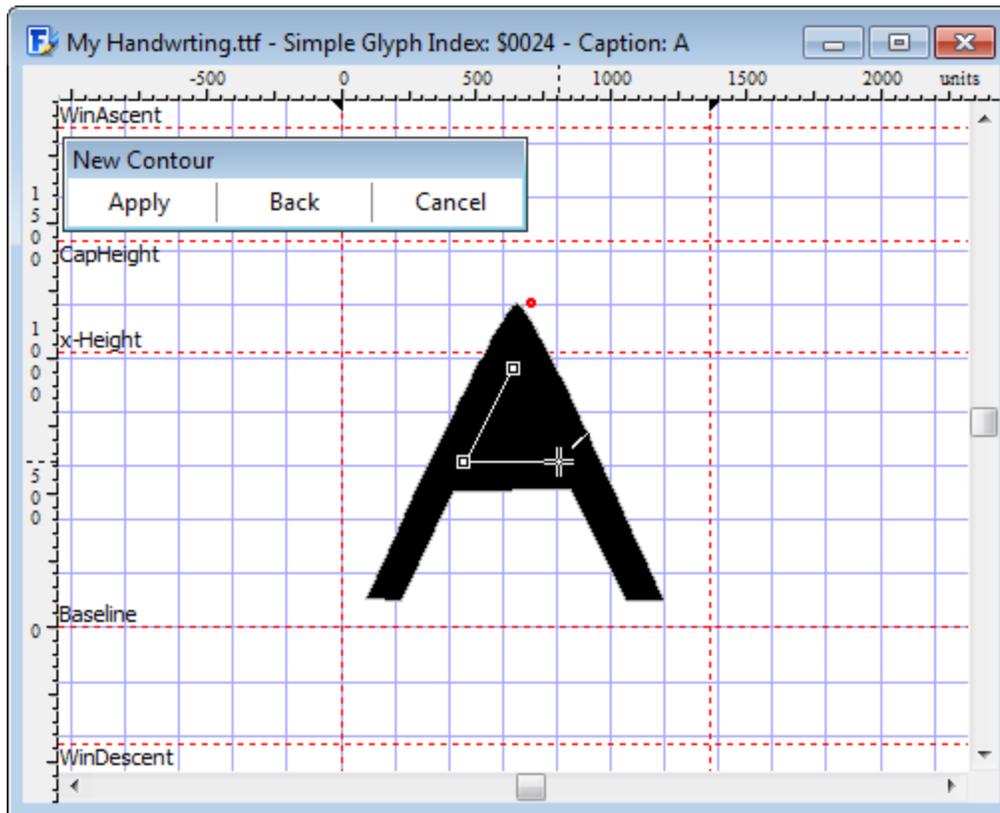
We close the **Glyph Edit** window to return to the **Font Overview** window.

2.3 Edit a Glyph - Character A

From the **Font Overview** window we double-click on the cell with the caption "A". Now we see the corresponding **Glyph Edit** window. Choose **Contour** in the

Insert menu. During this operation a left mouse-click creates a point onto the contour and a right mouse-click creates a point off the contour (used to create a Bézier curve).

Press the **Apply** button (or click on the first point) to finish the contour. We create two contours to complete the glyph.



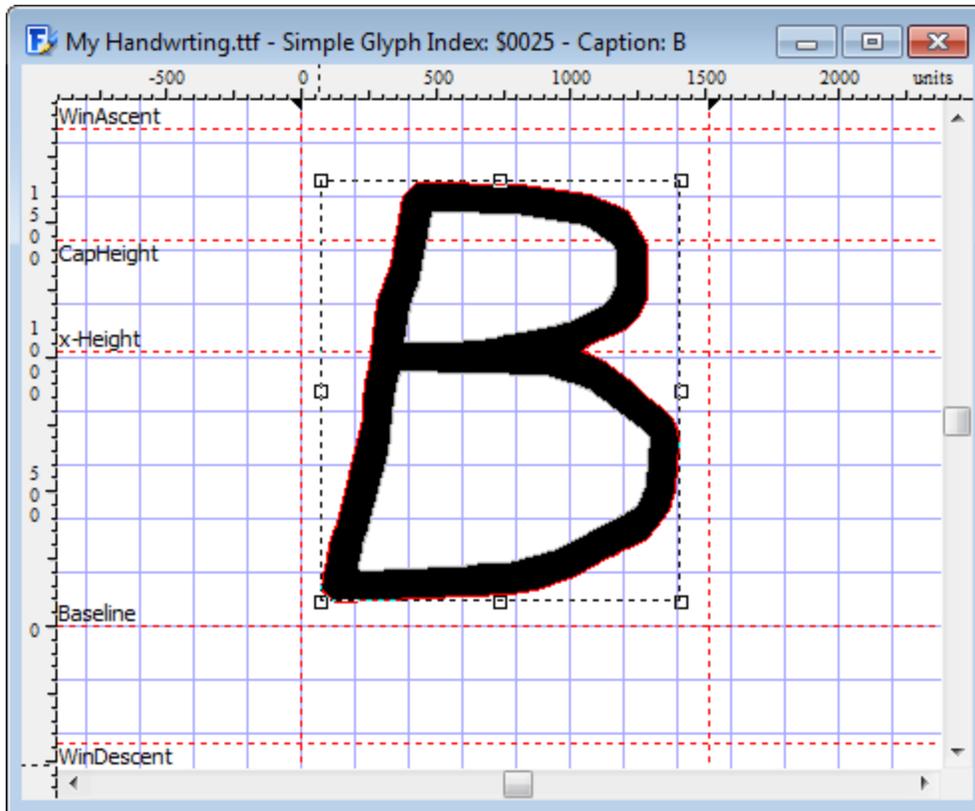
Contours that need to be filled black must have a clockwise direction. If we want to make a white area inside an existing contour we must make the direction of the new contour counter clockwise.

Finally make sure the contours are within the visible area (as described in the previous section); if necessary adjust the left and right bearings.

We close the **Glyph Edit** window to return to the **Font Overview** window.

2.4 Edit a Glyph - Character B

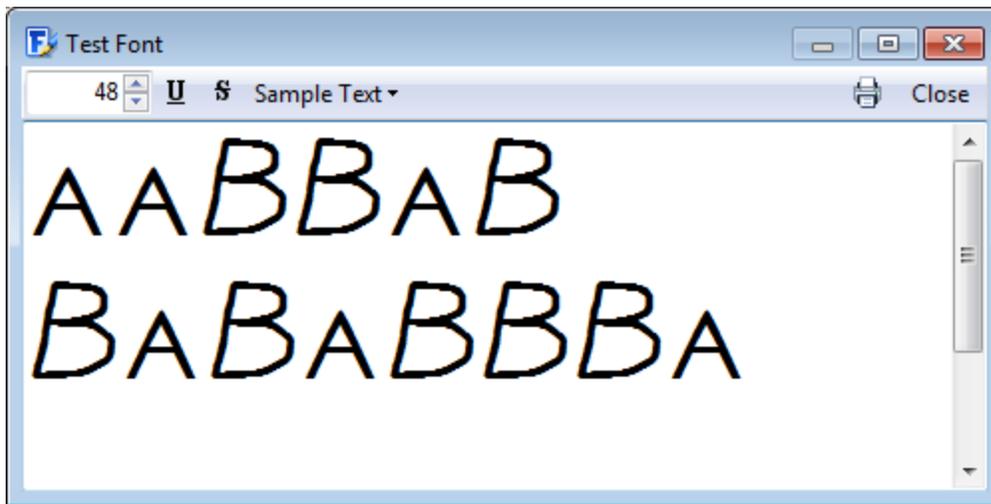
To personalize character "B" we want to import an image of our own handwritten "B". This image should not be too small or too large, we recommend an image dimension of 300x300 pixels. Bitmap, GIF, PNG and JPG images can be created with a paint program and if you have a scanner you could also import an image of your character "B" into the appropriate software.



From the **Font Overview** window we double-click on the cell with the caption "B". Choose **Import Image** in the **Tools** menu. Press the **Load** button to select the image you want to use and press the **Generate** button. Now you will see your image in the **Glyph Edit** window. Finally make sure the contours are within the visible area (as described in the previous section); if necessary adjust the left and right bearings.

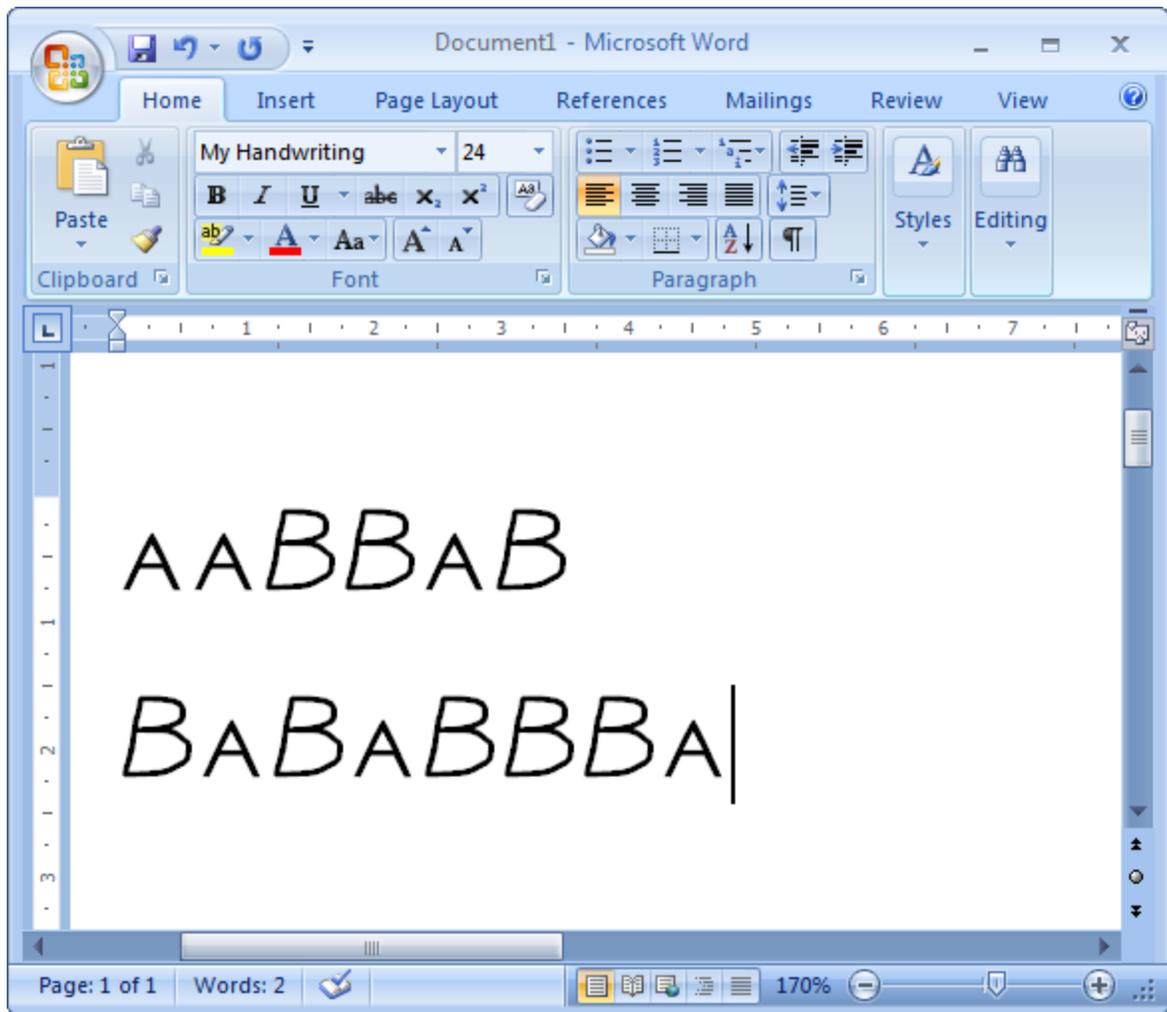
2.5 Test and Install the Font

Now we have finished the first three characters we want to see the result. We test the new font by selecting **Test** in the **Font** menu. Type upper case "A" and "B" and number "3" in the text field to see your creation.



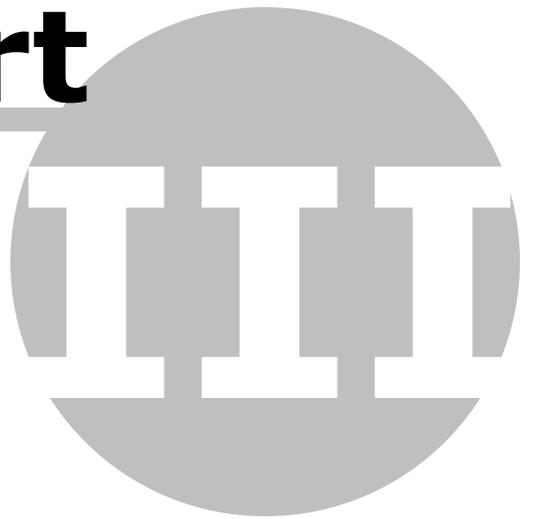
To make your font available to other applications select **Install** in the **Font** menu. The **Font Installation** wizard will guide you through the installation process.

At the end of the installation process you will be informed that the font is installed successfully. Almost all applications (like Word) immediately allow you to use the font, but sometimes require a restart before they recognize the newly installed font.



When you are satisfied with the font, you need to export and install it.

Part



3 About Fonts

3.1 TrueType

TrueType is a scalable font technology designed by Apple Computer, and has been superseded by the OpenType format. However people still refer to TrueType fonts, while in fact 99.9% of all fonts that come with Windows are OpenType fonts.

See also: <http://forum.high-logic.com/viewtopic.php?f=5&t=1619>

A TrueType font file contains data, in table format, that comprises an outline font. Rasterizers use combinations of data from the tables contained in the font to render the glyph outlines.

3.2 OpenType

The OpenType font format is an extension of the TrueType font format, allowing support for PostScript font data. Technically there are two OpenType Font flavors; TrueType based and PostScript based. OpenType was developed jointly by Microsoft and Adobe to produce a hybrid between Type 1 and TrueType fonts, with additional features that work on Macintosh and Windows computers. OpenType fonts can include OpenType layout features, which allow font designers to design better international and high-end typographic fonts.

Some descriptions (of various fields in this document) are copied from the specification available online at:

<http://www.microsoft.com/typography/otspec/> 

3.3 Color Extension

As revealed at the Microsoft Build Developer Conference in June 2013, Windows 8.1 comes with a revolutionary extension to the OpenType font standard, which introduces multi-color fonts. The technology which is both simple and powerful uses multi-layer glyphs which are in essence scalable outlines that are rendered and processed like any other character, except each layer has its own color.

High-Logic is convinced this new technology will open new colorful doors. That is why FontCreator was the first font editor to support the new multi-color fonts extension.

The beauty of the color extension, is that the fonts will continue to work like any other font on devices and systems that don't support the extension yet. That is why it is strongly recommended to always include the base outlines for each glyph (used as fallback in case color fonts are not supported, or not wanted in a particular situation).

3.4 Web Open Font Format (WOFF)

The Web Open Font Format (WOFF) is a font format based on the same principles as OpenType and TrueType fonts but has been optimized for use in web pages. WOFF is supported across all recent major browsers. Fonts that are used in webpages are also called "web fonts".

FontCreator supports both importing and exporting WOFF fonts.

More information about Webfonts and how to use them:

<http://www.webfonts.com> 

More information about the Web Open Font Format:

<http://www.w3.org/TR/WOFF/> 

3.5 Font Copyright

Unless you know otherwise, you should assume all fonts to be copyrighted works that are someone's property and treat them as you would any other software. Fonts are software products in their own right, and are protected by international copyright law as well as individual license agreements. Even redistributing so-called 'freeware' or 'public domain' fonts is problematic. If you have created a font yourself (without using anything from other fonts), it is your property.

The use of any commercial font is governed by the terms of its manufacturer's End User License Agreement (EULA). Several major font vendors specifically allow altering a font, as long as the altered font is only used on machines for which you have licensed the original font. If you have questions about what can or can't be done with a font, you should contact that font's manufacturer.

The Copyright Notice field in the **Legal** tab on the **Font Properties** dialog may direct you to the copyright holder, but be aware that this field may be blank, or may have been altered. Also the License Agreement and the License Agreement Link fields might have important information.

Part



4 Opening, Creating and Saving Fonts

4.1 Working with Font Projects

FontCreator uses its own file format to store the font data, several other settings and configuration parameters:

- Font data
- Font properties
- Export settings
- Guideline, Grid and Metrics settings
- OpenType Layout Features
- User notes

This will make sure that regardless of the font format you export, all information about the font will remain available.

Opening existing project files

There are several ways to open existing project files:

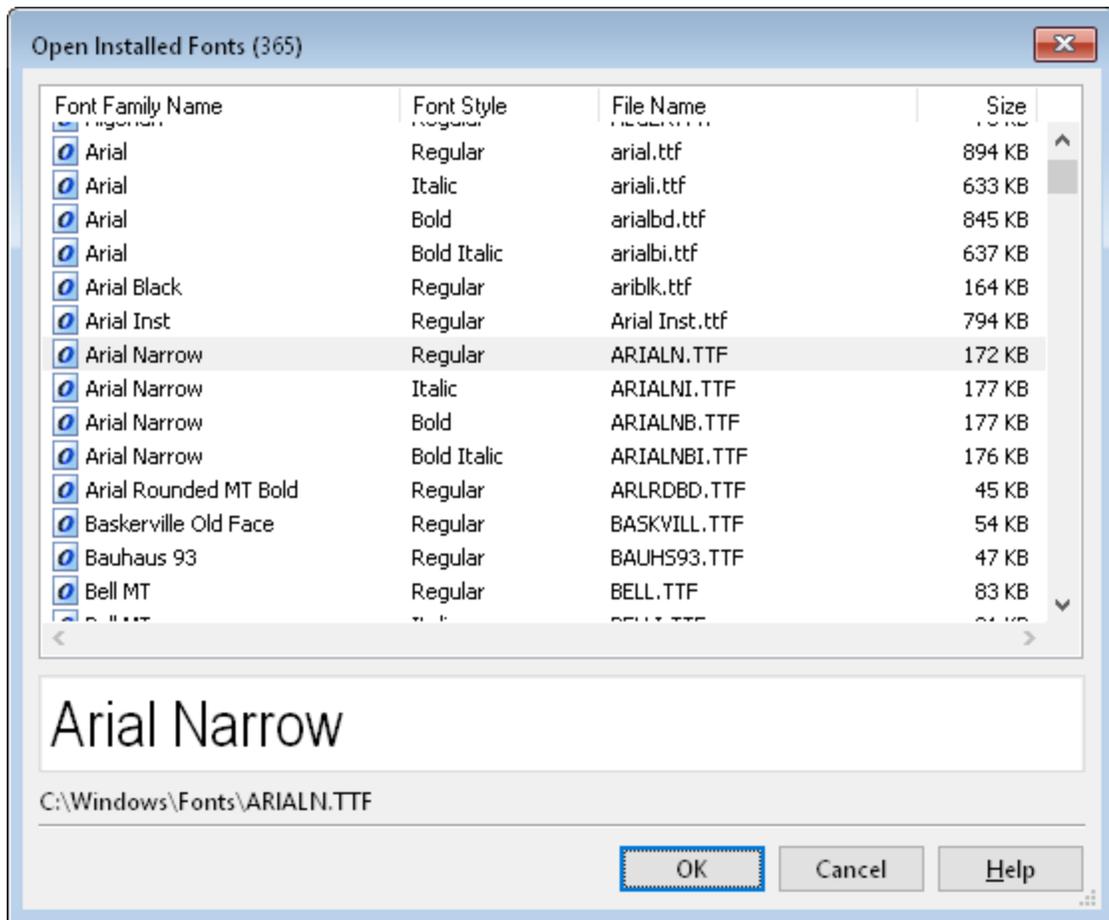
- Select Open project from the File menu
- Select a recent project from the Reopen submenu in the File menu
- Select a recent project in the Windows Taskbar Jumplist (only available on Windows 7 and later)
- Double-click a project file in the Windows Explorer
- Drag a project file from the Windows Explorer onto FontCreator

4.2 Open a Font

There are several ways to open a font file.

Open fonts already installed on your system

To open an installed font file select **Open** from the **File** menu and choose **Installed Font** option.



Open any font that is available

From the **File** menu, select **Open** and choose **Font File** option to open a font through Windows default open dialog box.

Warning: If you want to open fonts that are located in Windows fonts folder you should use the Installed Fonts command (or drag and drop), because this folder is marked (by Windows) to be a special folder and therefore behaves differently (e.g. it shows font names instead of file names).

Drag and drop a font file

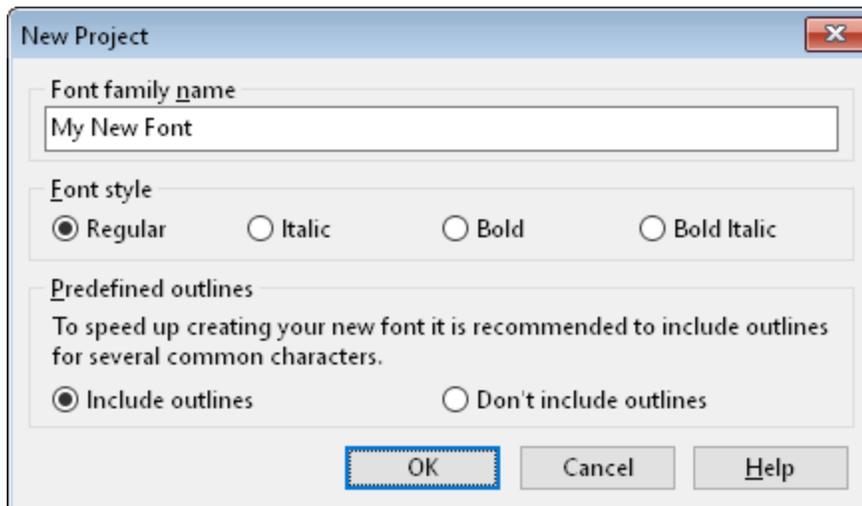
Another way to open a font, is to drag a font file from Windows Explorer and drop it onto FontCreator.

Reopen a font file

To open a font that you've used recently, choose **Reopen** from the **File** menu to display the names of the last ten used fonts. Click on the font you want to use.

4.3 Create a Font Project

On the **File** menu, click **New Project** to create a new font.



Font family name

In Windows, the Font family name is displayed in the font menu. The **Font family name** will appear as the font name when you select a font in a word-processing program, etc.

Font style

The font will be identified through its **Font family name** and the **Font style**. To create a full font family, you will need to create four fonts, each with a different **Font style**, but with the same **Font family name**.

Predefined outlines

Selecting "Include outlines" will add outlines for several common characters. This will speed up creating your new font. You can replace or modify these glyphs later. The outlines can be used royalty free in your own fonts.

Note: On the **Personalize** tab, accessed by **Tools -> Options**, there are default naming values, that will be added to the new font.

See also:

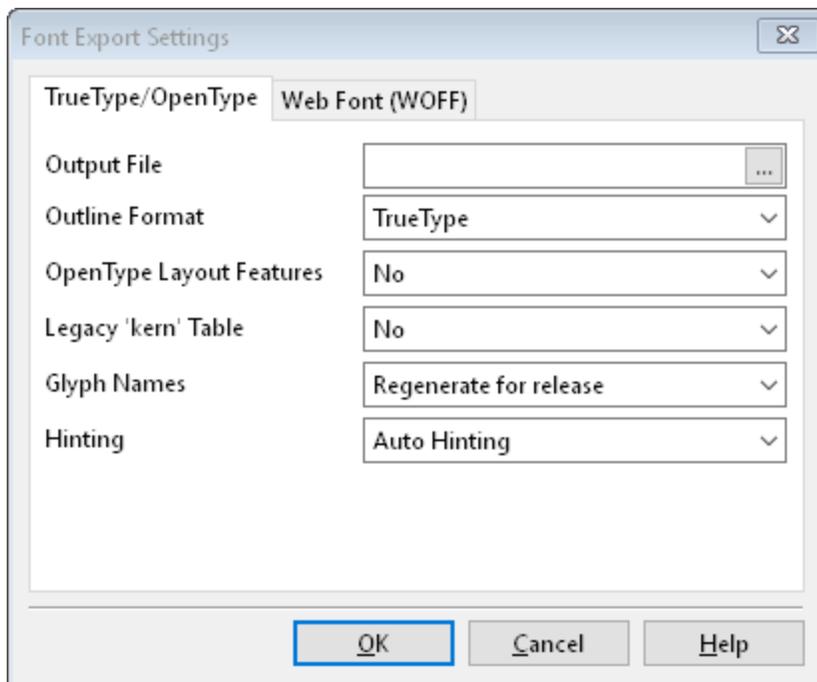
[Default Naming Values](#)
[Insert Characters](#)

4.4 Save a Project

To save an active project select **Save Project** in the **File** menu. If you want to save the active project with a different name, or in a different location, select **Save Project As** in the **File** menu, choose a name and location and click **Save**.

4.5 Export a Font

To export an active project to a TTF/OTF or WOFF font select one of the export formats in the **Export Font** submenu in the **File** menu. The first time you export your font you will be asked to choose an output location. If you want to export the active project with a different file name, or in a different location, select **Export Font As** in the **File** menu.



When you open an existing font, the export settings will be set in a way that they match the original font properties as much as possible.

Output File

Location where the file will be exported. If this field is left blank, FontCreator will show a save dialog the first time you export the font.

Warning: It is not recommended to export your font directly into the Windows fonts folder.

Warning: Files locked by Windows can't be saved. This happens when the Operating System keeps the font in memory. You could try to uninstall (delete) the font through the Windows fonts folder.

Outline Format

TrueType - The font is exported with TrueType based outlines (quadratic Bézier curves)

CFF (Postscript) - The font is exported with CFF based outlines (cubic Bézier curves)

OpenType Layout Features

No - No OpenType Layout Features are embedded in the font.

Yes - OpenType Layout Features are embedded in the font.

Note: OpenType Layout Features are ignored for Symbol Fonts.

Glyph Names

It is recommended to include Glyph names in TrueType/OpenType fonts but they can be omitted in WOFF fonts to reduce the file size.

Note: Glyph names are always included in CFF based fonts

Legacy 'kern' Table

The kern table is mostly useful for fonts to stay backwards compatible with older word processing software, but we recommend to no longer include it. Instead use OpenType Layout Features to include kerning.

No - No legacy kern table is added to the font

Yes - A legacy kern table is added to the font. The kern table will be build from the first pair adjustment lookup located in the Latin script, default language, kern feature. The lookup must only contain values for XAdvance for the first glyph.

Note: The legacy 'kern' table is never included in CFF based fonts.

Hinting

Hinting information will improve readability on screen for smaller font sizes.

No Hinting - No hinting is added to the font

Autohinting - Hinting information is automatically generated

Keep Original - Stores hinting information that was originally available in the font.

Note: Hinting is not available for CFF outlines

XML Metadata (Web Font only)

If enabled, a small xml file is embedded in the font that allows web browsers to identify the font without having to load the entire font.

Legacy TrueType Fonts

Sometimes it's preferred to export a font in the old TrueType format for older software. Set the export settings to the following settings:

Outline Format: TrueType

OpenType Layout Features: No

Glyph Names: Yes

Legacy 'kern' Table: Yes

Hinting: No Hinting

4.6 Close a Font

To close the current font or project select **Close** in the **File** menu. To close all active fonts and/or projects at once select **Close All** in the **File** menu.

Part



5 Editing Fonts

5.1 Editing Options

5.1.1 Undo Command

The Undo command from the **Edit** menu reverses the last action made to the active font. So using Undo returns the font to its state prior to the most recent operation.

Note: The **Undo** button on the toolbar has a small arrow which allows you to pull down a menu and select multiple actions to be undone.

5.1.2 Redo Command

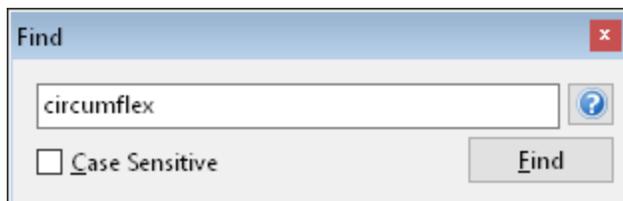
The **Redo** command from the **Edit** menu re-applies the actions or commands on which you have used the Undo command. FontCreator supports Multiple Redo, which is particularly useful if you have performed the undo command more than you had intended. If this occurs, and you want to re-apply them, either choose the **Redo** command as many times as necessary or use the drop arrow on the **Redo** button located on the **Standard** toolbar.

5.1.3 Repeat Your Last Action

When you select **Repeat** from the **Edit** menu the program will repeat the last action you have done. When the **Repeat** command can't be selected, you can't repeat the last action.

5.1.4 Finding a Glyph

You can search for glyphs and or characters by their glyph name and mappings. Select **Find** on the **Edit** menu to open the **Find** window or use the keyboard shortcut CTRL+F.



The input field accepts several kinds of keywords:

- Single character : Entering a single character will find the entered character if it exists in the font. (Example: "a") Note that this type of search is case-sensitive regardless of the case-sensitive checkbox!
- Part of glyph name

- Start with: "A*" will return Agrave Aacute etc.
- Ends with: "*grave" will return Agrave Ugrave etc.
- "#353" will show that exact Glyph ID
- Unicode ranges \$0032-\$0046, \$0012 etc (can be entered in decimal or hexadecimal)

5.1.5 Tags

Tags allow you to mark glyphs so they appear with a small colored bar in the Font Overviews. To tag a glyph, right click on one or more selected glyphs and select one of the tags from the **Tag** submenu, use the keyboard shortcut keys (CTRL + 1-5) or drag and drop them from the font overview window onto one of the tag categories. You can view all glyphs that have a tag by selecting the **Tagged** category from the categories panel or select one of the tag subcategories to view all glyphs that have a specific tag assigned to them.

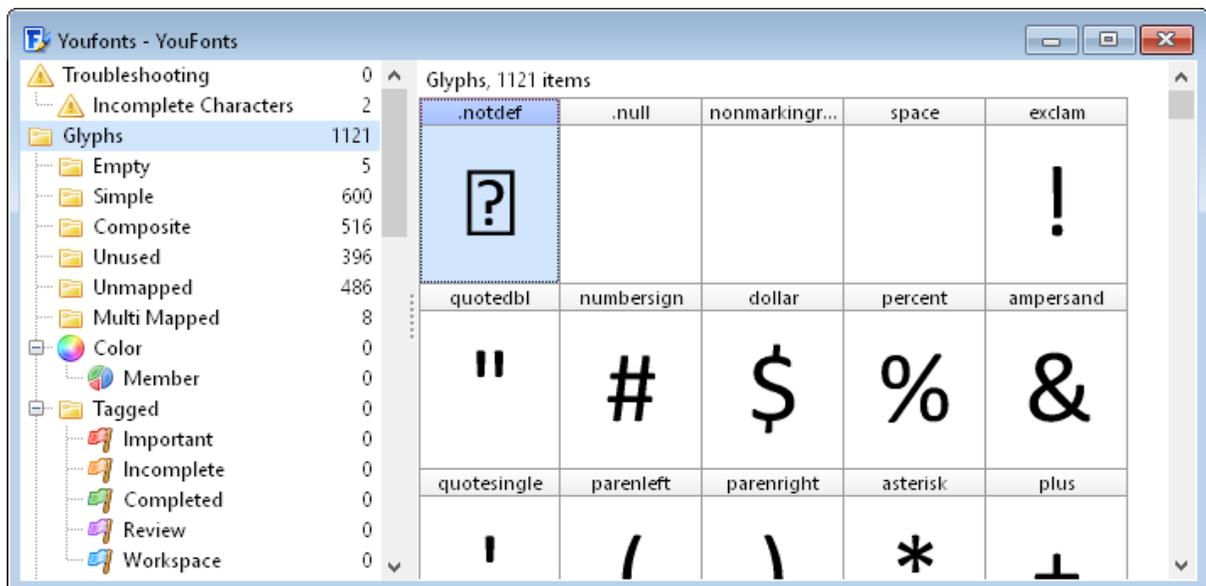
You can select glyphs with a specific tag through **Select Tagged** command from the **Edit** menu.

Note: Each glyph can only have one tag.

5.2 Editing Fonts

5.2.1 Using the Font Overview

In the **Font Overview** window there is a list of categories and a grid where all cells have a caption and a part that shows the glyph.



The categories panel is a convenient way to quickly show a subset of characters or glyphs and allows you to display them in several groupings and orderings. To change the current grouping and ordering use the **Overview Toolbar** on the main toolbar.

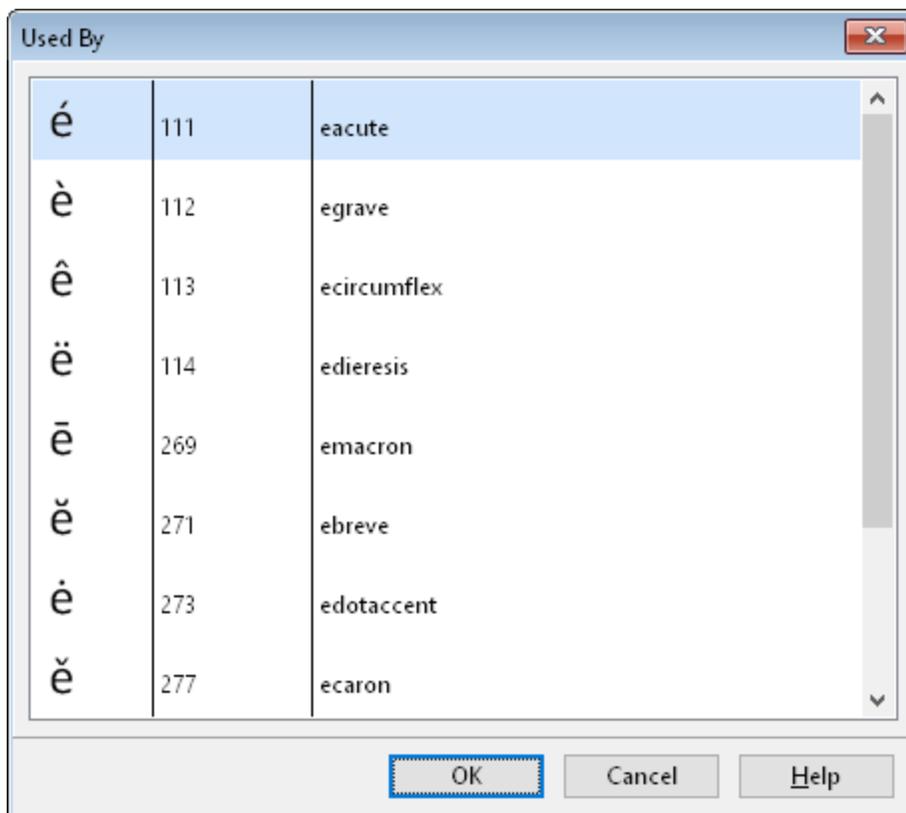
When creating Unicode fonts, the categories panel shows a convenient list of all Unicode blocks that contain at least one mapped glyph. When you add new characters through the **Add Characters** dialog or change character mappings the Unicode categories are automatically updated. If you are creating a symbol font, a symbol category will be shown instead.

Each cell has a caption that is used to display the glyph name, Unicode name, the codepoints or the glyph index. To select the kind of caption, use the **Overview Toolbar** or right-click in the **Font Overview** window and select a specific caption. Setting the caption type to **Automatic** will automatically change the captions to the value of the current selected grouping method. This means that when you group by Advance Width, the cell captions display the **Advance Width** value.

You can also switch between decimal and hexadecimal values for the codepoints and glyph indexes in the **Display Format** submenu of the **View** menu.

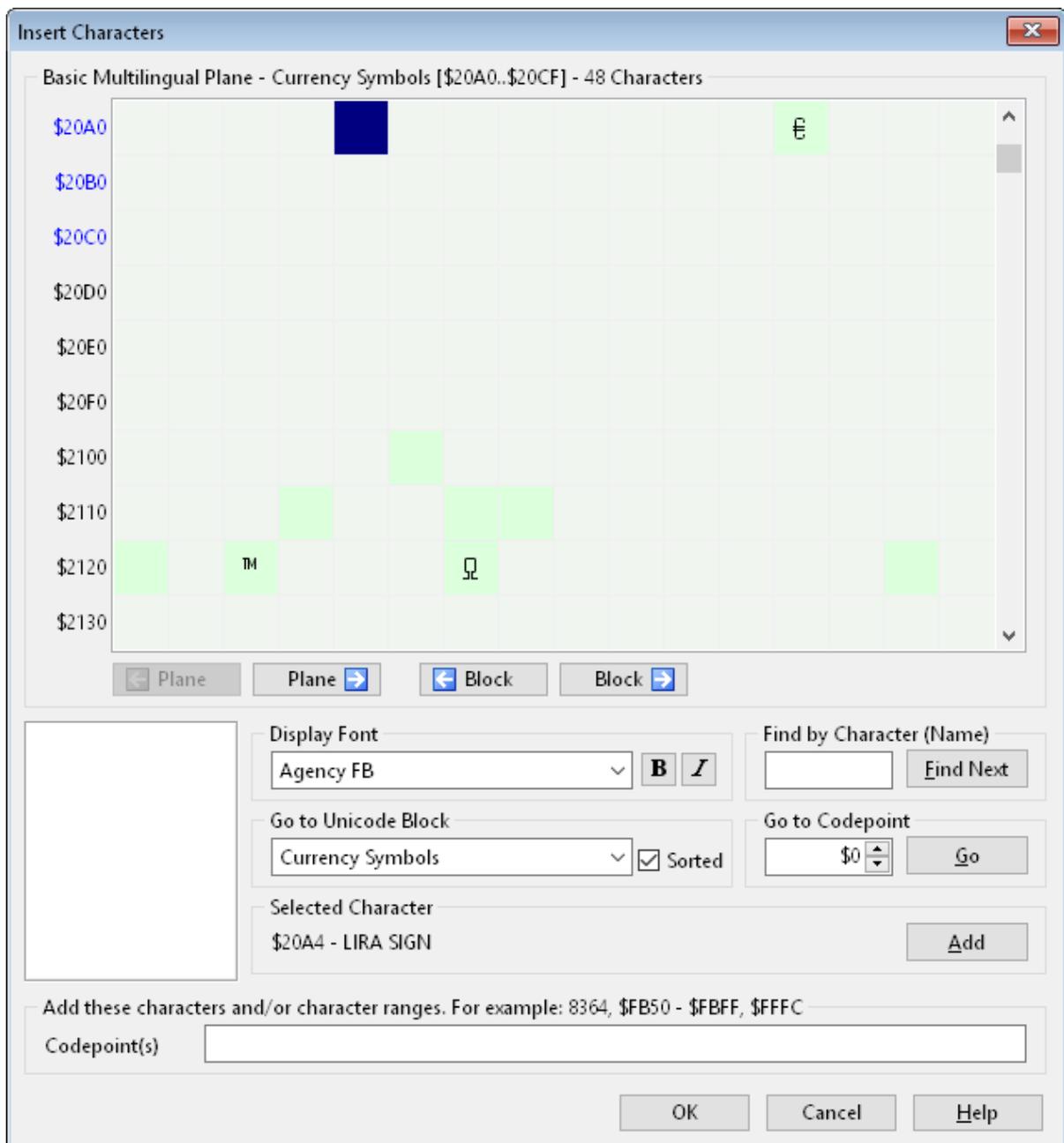
5.2.2 Used By

The Used By window (available by right-clicking a glyph in the **Font Overview** window or **Glyph Edit** window and selecting the **Used By** menu item) is used to display an overview of all glyphs that use the selected glyph. To jump directly to one of the displayed glyphs double-click it in the window or select it and click the **OK** button.



5.2.3 Insert Characters

Select **Characters** in the **Insert** menu to add glyphs with their character mappings and glyph names to the font. This option is available when the **Font Overview** window is active.



Select a Unicode block from the combo box, "Go to Unicode Block". Uncheck the "Sorted" field to sort the blocks in numerical code-point order instead of alphabetical order. If necessary, choose another installed font which includes the characters to be added. Characters can be added anyway, but it helps if the glyphs can be previewed.

Use the next and previous block or next and previous plane buttons to scroll through the font. The code point of the character to add can be found by entering the decimal value in the "Go to Code Point" field, or by entering the

hexadecimal value preceded by a dollar sign. For example, enter 8364 or \$20AC to find the Euro Sign (€).

Either a character or (part of) the Unicode name of the character can be used in the field, "Find by Character (Name)". For example, type "€" or "euro" to find the "EURO SIGN". When enter a character it will find an exact match, but with part of the Unicode name you can find another match if you press the Find Next button again.

Click on a character to select it and preview it in the glyph preview at bottom left. Double-click the character, or click the Add button, to add its code-point to the list of selected characters at the bottom of the dialog. Keep adding individual characters by double-clicking, or hold down the Shift key and double-click to add a range of characters. The codepoints will be displayed in Hexadecimal or Decimal notation depending on the setting in **View -> Display**. You can also type codepoints into the characters to add field, separated by commas (or hyphens to add a range of characters), or cut and paste a predefined list of characters from a text file. For example, pasting 256-383 then clicking OK would add the entire Latin Extended-A character set.

Click OK to dismiss the dialog and add the characters to the current font. If the glyphs exist in the overview sample font, and if "Show samples in empty glyphs" is on, grey outlines of the new characters will be displayed in the font overview.

Characters or entire character sets can also be added using [Transform Scripts](#). See the topic: Glyph Transformer.

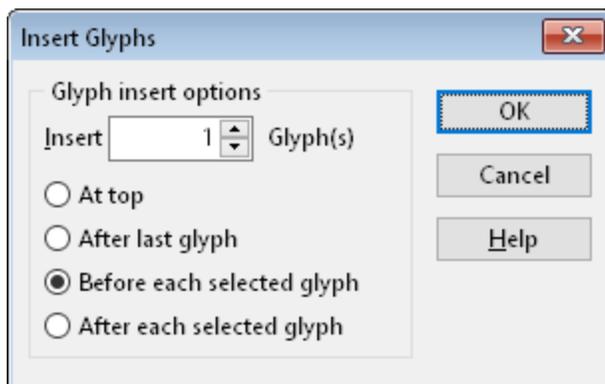
Note: The number of glyphs that may be included in one font is limited to 65535.

See also:

[Glyph Transformer](#)

5.2.4 Insert Glyphs

Select **Glyphs** in the **Insert** menu to add glyphs to the font. This option is available when the **Font Overview** window is active.

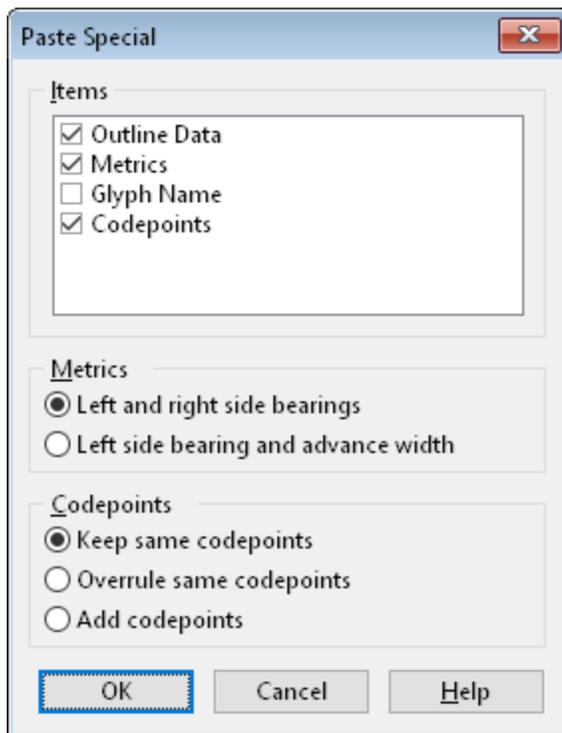


Note: The number of glyphs that may be included in one font is limited to 65535.

5.2.5 Copy and Paste Glyphs

The **Paste Special** command is used to specify what parts of a group of glyphs (already copied to the clipboard) should be pasted.

You can select any number of glyphs in the **Font Overview** window by clicking on them while holding down the **Ctrl** key. You can perform several operations on the selected glyphs. If you paste the selection to another font the glyphs will be pasted to the selected glyphs. If you want the pasted glyphs to be placed in another position you can select the same number of glyphs you copied prior to the paste action, or select just one glyph.



Outline Data will copy the outlines.

Metrics are the Left Side Bearing and Advance Width of each glyph

Glyph Name will paste all glyph names

Codepoints will paste the mappings. Keep same codepoints adds new mappings to the glyphs. Mappings that already exist will be reassigned to the pasted glyph(s) when the **Overrule same codepoints** option is selected. You can also choose to add mappings. **Add codepoints** adds all mappings. When mappings

are already available, they will be reassigned to the pasted glyph(s).

5.2.6 Adding a New Character - the EURO Sign

Sometimes a font is missing one or more characters. This will explain how to add the EURO sign to an existing font.

There are two ways to add a character to a font.

1a. **The easiest and recommended way is to simply add a character at the end of the glyph list**

- Select **Characters** from the **Insert** menu. Enter "€" or "EURO SIGN" into the **Find by Character (Name)** and press the **Find Next** button. Now select the **Add** button and press the **OK button**.

1b. **Alternatively you can add a glyph at the end of the glyph list and manually set the glyph name and codepoint**

- Select **Glyphs** from the **Insert** menu and insert one glyph after the last glyph.
- Open the **Glyph Properties** tool window (View -> Glyph Properties or shortcut F3) and enter "Euro" as Glyph Name.
- Click the **Generate** button  next to the codepoints to automatically fill in the proper code-point.
- Click the **Apply** button to assign the changes

2. **Edit the glyph**

Double-click the Euro glyph to open the **Glyph Edit** window. There are several ways to add contours to the glyph:

- Freedraw contours (Insert -> Freedraw Contours...)
- Import an image (Tools -> Import Image)
- Add new contours (right-click and select "New Contour...")
- Paste contours from other glyphs
- Drag outlines from the Samples toolbar
- Copy and paste from other software like Adobe Illustrator

3. **Adjust the right and left bearings**

Around the glyph there are four lines that represent the Bearings. These are shown by default but you can hide them through the **Show Bearings** button on the **Drawing** toolbar. The left and right bearings can be changed by dragging them to their desired position.

You could also adjust the bearings from the Glyph Properties tool window.

4. **Test the font.**

Select **Test TTF/OTF** or **Test WOFF** in the **Font** menu.

Add a Euro sign in the text area with Alt-0128 or use [MainType](#) utility to copy and paste the EURO character into the text area.

5.2.7 **Font Name**

Be careful not to confuse the font name with the file name. Windows uses the file name to install a font, while the font name is used to identify the font.

To change the font name select **Properties** from the **Font** menu.

5.2.8 **Font Type**

If you want to make a bold font out of a regular font, you also need to make all your glyph outlines bold. This can be easily accomplished through the transform features, using the Medium to Bold script. There are several other scripts that can make a font thin, italic, etc.

If the font already contains bold outlines, but is recognized as a regular font, then you need to change a few settings on the **Font Properties** dialog:

On the **Identification** tab:

- Font Subfamily
- Weight
- Bold checkbox
- Italic checkbox
- Italic Angle

On the Characteristics tab

- PANOSE Weight

5.2.9 **Font Embedding**

Select **Properties** from the **Font** menu and select the **Legal** tab page. Here you

can alter the **Font Embedding Licensing Rights**.

Note: Embedding symbol fonts may not be possible in Word.

Note: To embed a font, the Glyph Names must be exported for some applications (like Adobe Acrobat).

5.2.10 Monospaced versus Proportional

A monospaced font is a font where all characters have the same width. These fonts are often used to emulate typewriter output for reports, tabular work and technical documentation.

In a proportional font the width of each character, including the space character, varies with the shape of the character. Proportional fonts are easier to read and are preferred for publishing applications.

From proportional to monospaced

To change a proportional spaced font into a monospaced font, follow these steps:

- Select the **AutoMetrics** command (**Tools** menu) to force the advance width to be the same for all glyphs including .notdef and combining marks. The only exception is (format) control characters (e.g. .null) which are allowed to be zero-width.
- The advance width of combining marks may be collapsed through OpenType glyph positioning (e.g. using an OpenType single adjustment lookup in the mark feature).
- If necessary, change the outlines of glyphs that are too wide.
- In the **Properties** dialog (**Font** menu) on the **Characteristics** tab set **Family Kind** to 2 (Latin Text) and **Proportion** to 9 (Monospaced).

5.2.11 Unicode versus Symbol

Important: Symbol fonts are legacy; we recommend designing a Unicode font instead. If there are codepoints assigned for your characters, then use them, otherwise use the private use area.

Symbol character sets have a special meaning: all of the characters in the Unicode range 0xF000 - 0xFFFF (inclusive) will be used to enumerate the symbol character set. All glyphs in this range are mapped to the range 0x0000 - 0x00FF.

Symbol fonts do not form words, so line breaks can occur after any character code. A spell checker should not check symbol font-formatted material.

You can convert your font between Unicode and Symbol though the **Convert**

Font menu item in the **Tools** menu.

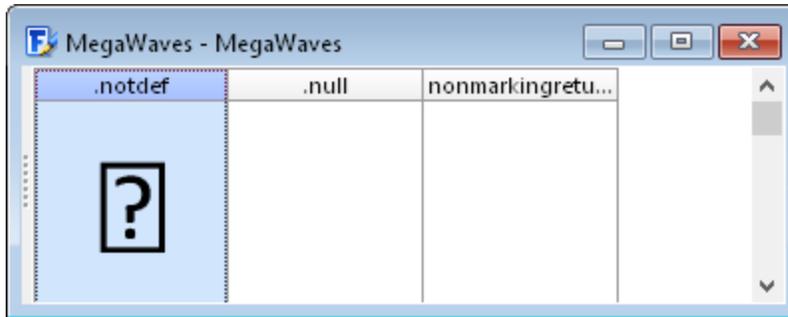
Note: only the first 224 characters of symbol fonts will be accessible: a space and up to 223 printing characters.

5.2.12 Recommended Glyphs

In addition to script and language specific punctuation and native numbers, the following glyphs are highly recommended for inclusion in fonts.

First four glyphs

TrueType outline fonts should have the following four glyphs at the beginning of a font. These were listed in Apple's original TrueType specification. These glyphs are recommended to allow for the same version of the font to work on both Windows and Macintosh.



Glyph 0 is the .notdef (missing character) glyph.

Glyph 1 is the .null glyph; it has no contours and zero advance width.

Glyph 2 is the nonmarkingreturn glyph; it has no contours and positive advance width.

Glyph 3 is the space (and no-break space) glyph; it has no contours and positive advance width.

Index	Glyph Name	Unicode
0	.notdef	
1	.null	
2	nonmarkingreturn	
3	space	\$0020

Glyph 2 and 3 should have the same advance width.

The .notdef glyph is very important for providing the user feedback that a glyph is not found in the font. This glyph should not be left without an outline as the user will only see what looks like a space if a glyph is missing and will not be aware of the active font's limitation.

It is recommended that the shape of the .notdef glyph be either an empty rectangle, a rectangle with a question mark inside of it, or a rectangle with an "X". Creative shapes, like swirls or other symbols, may not be recognized by users as indicating that a glyph is missing from the font and is not being displayed at that location.

General punctuation and 'Latin' numbers

Glyph Name	Descriptive Name	Sample	Unicode
space	space		\$0020
exclam	exclamation mark	!	\$0021
quotedbl	quotation mark	"	\$0022
numbersign	number sign	#	\$0023
dollar	dollar sign	\$	\$0024
percent	percentsign	%	\$0025
ampersand	ampersand	&	\$0026
quotesingle	apostrophe	'	\$0027
parenleft	left parenthesis	(\$0028
parenright	right parenthesis)	\$0029
asterisk	asterisk	*	\$002A
plus	plus sign	+	\$002B
comma	comma	,	\$002C
hyphen	hyphen-minus	-	\$002D
period	period	.	\$002E
slash	slash	/	\$002F
zero	digit zero	0	\$0030

one	digit one	1	\$0031
two	digit two	2	\$0032
three	digit three	3	\$0033
four	digit four	4	\$0034
five	digit five	5	\$0035
six	digit six	6	\$0036
seven	digit seven	7	\$0037
eight	digit eight	8	\$0038
nine	digit nine	9	\$0039
colon	colon	:	\$003A
semicolon	semicolon	;	\$003B
less	less-than sign	<	\$003C

Quotes and Ellipsis

Smart quotes (also known as curly quotes) are fancy characters which make text look better compared to the straight apostrophe (') and straight quote or inches character ("). Microsoft Word also automatically changes 3 periods to an ellipsis.

If your font does not support these characters, you can either turn the "Smart Quotes" and "Ellipsis" features off in the application (In Microsoft Word it's under Tools -> AutoCorrect) or make sure these glyphs and their mappings are available in the font.

Glyph Name	Descriptive Name	Sample	Unicode
quoteleft	left single quotation mark	`	\$2018
quoteright	right single quotation mark	'	\$2019
quotedbleft	left double quotation mark	“	\$201C
quotedblright	right double quotation mark	”	\$201D
ellipsis	horizontal ellipsis	...	\$2026

Other suggested glyphs

Glyph Name	Descriptive Name	Sample	Unicode
nbspace *	no-break space		\$00A0
currency	currency sign	₣	\$00A4
section	section sign	§	\$00A7
logicalnot	not sign	¬	\$00AC
degree	degree sign	°	\$00B0
paragraph	paragraph sign	¶	\$00B6
periodcentered	middle dot	•	\$00B7
endash	en dash	–	\$2013
emdash	em dash	—	\$2014
bullet	bullet	•	\$2022
euro	euro currency symbol	€	\$20AC
bulletoperator **	bullet operator	•	\$2219

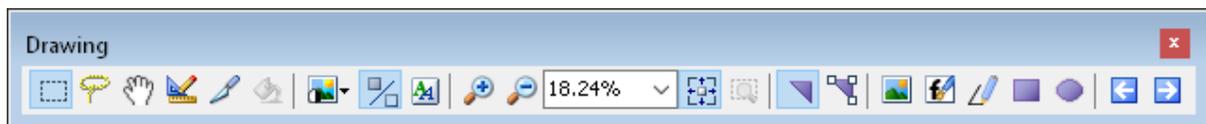
* nbspace is usually mapped to the space glyph

** bulletoperator is usually mapped to the periodcentered glyph

5.3 Editing Glyphs

5.3.1 Introduction

From the **Font Overview** window you can double-click a glyph to open a **Glyph Edit** window and edit the selected glyph. You can also select a glyph, right-click it and select **Edit**.



You can "zoom in" to get a close-up view of your glyph or "zoom out" to see more of the page at a reduced size. You can use the edit field located on the **Drawing** toolbar to change the zoom percentage. The **Zoom to Selected** button will be enabled as soon as contours or points are selected. Pressing this

button will zoom into the current selection. When you press the **Fit to Window** button, the glyph will be shown with the largest zoom factor that also shows the ascender and descender lines.

In a TrueType font, glyph shapes are described by their outlines. There are three glyph outline types:

Empty glyphs

Simple glyphs

Composite glyphs

Composite glyphs are simply a combination of two or more other glyphs - usually, but not limited to, a base character and one or more diacritical marks that are placed above and/ or below the base character.

To select a composite member click on it. If you want to modify specific member data double-click a composite member to open the **Composite Glyph Properties** window.

Other parts of the manual have a comprehensive explanation of the three glyph types.

5.3.2 Empty Glyphs

Empty glyphs (like the space glyph) don't have outlines but they do have an advance width. In a **Glyph Edit** window you can change an empty glyph into a simple glyph by adding contours. You can also change an empty glyph into a composite glyph by adding a **Composite Glyph Member**.

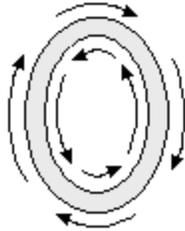
5.3.3 Simple Glyphs

5.3.3.1 Introduction

A simple glyph consists of a series of contours. Contours are composed of straight lines and curves. Curves are defined by a series of points that describe second order Bézier-splines. The TrueType Bézier-spline format uses two types of points to define curves, those that are on the curve and those that are off the curve. Any combination of off and on curve points is acceptable when defining a curve. Straight lines are defined by two consecutive on curve points. There is always an imaginary on curve point between two off curve points.

In order to distinguish between contour and point related operations, you can choose to work in contour or point mode. At any time you can change between

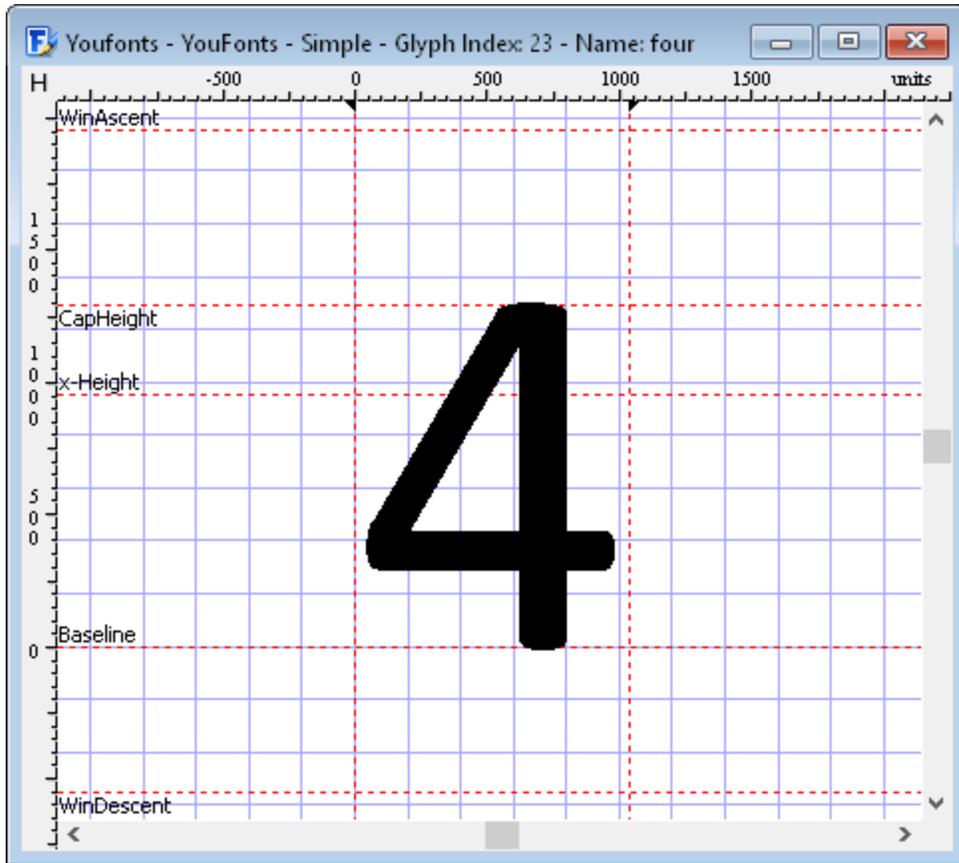
Contour and Point mode; select the appropriate **Mode** from the **View** menu, press one of the triangle buttons on the **Drawing** toolbar or double-click anywhere in the glyph edit window.



Contours that need to be filled black must have a **clockwise direction**. If we want to make a white area inside an existing contour we must make the direction of the new contour **counter clockwise**. Contour direction is determined by seeing in which direction the point index values increase or decrease. Contour direction is from the smaller point index to the larger. The general rule is that the contour direction should be such that "black is on the right". Using the glyph "O" as an example, the outer contour should travel clockwise, and the inner contour counter clockwise.

Click the **Correct contour directions** button on the **Validation** toolbar to correct the direction of all misoriented contours in a simple glyph or select **Direction** from the **Edit** menu to change the direction of the selected contour(s).

The character "4" is represented by a glyph with two contours. One contour you see as the black area and the white area within this glyph is the other contour.



From the **Drawing** toolbar you can change the way that you modify the glyph. In the **Glyph Edit** window, you can change between **Contour** and **Point** mode by double clicking inside the edit area, select the **Mode** from the **View** menu or use the appropriate button on the **Drawing** toolbar. The main difference between **Contour mode** and **Point mode** is that in **Contour mode** all operations are related to the contours while in **Point mode** you can change parts of the contours (e.g. move, add and delete points).

Holding down the Shift key while dragging points or contours restricts movement so the selection moves only in the x or y direction.

In the **Glyph Edit** window (in **Point mode**) the rectangles represent on curve points and the circles off curve points.

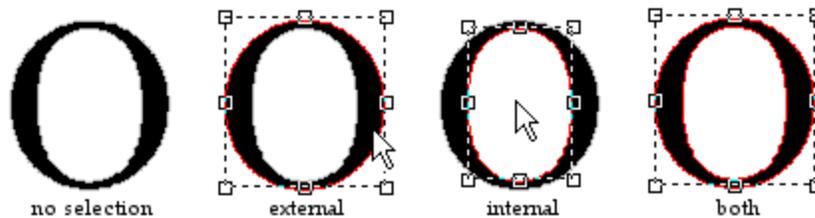
Tip You can nudge the selected contour(s) or point(s) up, down, left, or right by pressing the arrow keys. By holding down the Ctrl key and pressing the arrow keys finer nudging is available. By holding down the Shift key and pressing the arrow keys coarser nudging is available.

5.3.3.2 Contours

You can perform several operations on contours while in **Contour mode**.

Select contours

To select an external contour of a black area of a glyph click within the black area. To select an internal contour of a black area click within the internal white area.



To select more than one contour, press and hold down the Shift key while selecting contours. Another way is to click on the workspace where no contours are and, while holding down the left mouse button, drag a rectangle around all contours you want to select simultaneously.

Use the Ctrl-A shortcut or select **Select All** from the **Edit** menu to select all contours. Holding down the Shift key and clicking a contour already selected will remove that contour from the current selection.

Resize selected contours

When you select one or more contours, a box with "resizing handles" shows up around the selected contour(s). Click and drag one of those resizing handles to change the size of the selected contour(s). By default the selected contour(s) remain proportional to the original size, as you resize it. Press the Shift key while you drag one of the resizing handles located at the corners to resize freely.

Move selected contours

To constrain a selection so it moves only horizontally or vertically, press Shift as you drag the selection. Press Alt to ignore the snap to grid and snap to guidelines

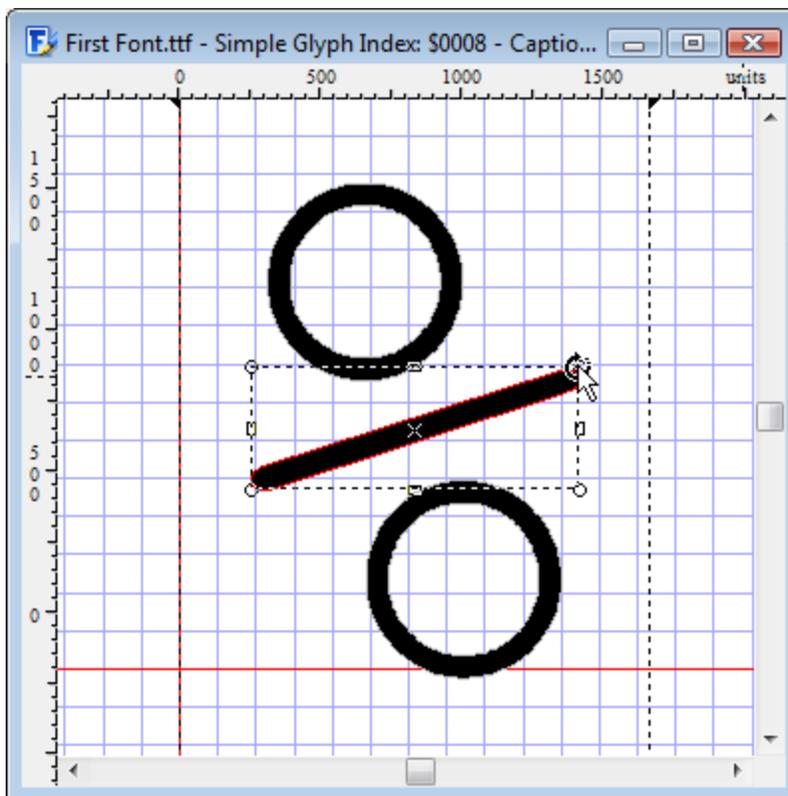
features.

Duplicate selected contours

To copy selected contours to a new position, hold down the Ctrl key as you drag the selection. Hold the Alt key to ignore the snap to grid and snap to guidelines features. Hold down the Shift key to move only in vertical or horizontal direction. Any combination of these keys is allowed.

Rotate and skew selected contours

You can rotate by first selecting a contour (or more) and then select it again (don't double-click). The little rectangles on the corners will change into circles. These can be used to rotate the selected contour(s). The other four rectangles are used to skew the selection. To change the origin for the rotation you can move the little cross in the middle of your selection. To rotate by 15° steps, hold down the SHIFT key while rotating the selected contours. If SHIFT is held down on the skew operation, then it will skew by 1° steps.



There are several ways to add outlines (contours) to a glyph:

- Using the Freedraw Contours tool
- Select the **Add rectangle** or **Add ellipse** button on the **Drawing** toolbar and place it on the Glyph Edit window.

- Draw a new contour
- First select **Contour** from the **Insert** menu. A left mouse click creates a point on the contour and a right mouse click creates a point off the contour for a Bézier curve. Creating points while holding down the Ctrl key generates straight lines (horizontal or vertical). After you have created your contour you press the Apply button.
- **Samples** toolbar:
- Drag and drop a glyph from the **Samples** toolbar into a **Glyph Edit** window.
- Copy contours from other glyphs, even from other fonts and paste them. (These operations carried out in their respective **Font Overview** windows)
- Import a stored vector or bitmap file of an image of a glyph into a **Glyph Edit** window using **Import** in the **Tools** menu.
- Paste a Clipboard image of a glyph into the **Glyph Edit** window. The Clipboard image could be created in another application which has graphics editing capabilities e.g. a cropping function. This operation is performed using the **Paste** option in the **Edit** menu.

5.3.3.3 Points

In **Point mode** the rectangles represent on curve points and the circles off curve points.

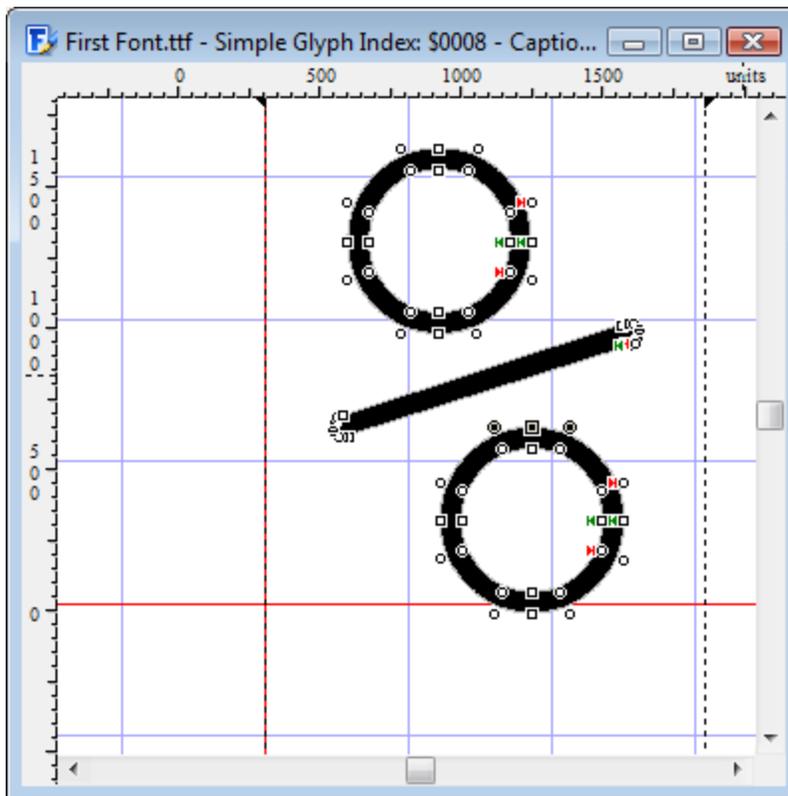
Select points

To select a single point click on it. To select more than one point hold down the Shift key while clicking on several points or another way is to hold down the left mouse button and drag a rectangle around the points you want to select simultaneously, whether a few or all points in the glyph. Use the Ctrl-A shortcut or select Select All from the Edit menu to select all points. Hold down the Shift key and select points you want to add or remove from the current selection.

Move selected points

To constrain a selection so it moves only horizontally or vertically, press Shift as you drag the selection. Press Alt to ignore the snap to grid and snap to guidelines features.

There are several operations related to points (move, add, delete, change points to on or off curve, etc.). Select one or more points and then right-click one of them to open a sub-menu with even more point related features.



Every contour has a start point with a green mark and an end point with a red mark. To change the start point, right-click a point and select **First Point**.

Smooth Curves around On Curve Points

It is not always easy to keep an outline smooth around on curve points. Fortunately there is a command that will do just that. Right-click one or more on curve points, which are surrounded by off curve points, and select **Smooth Curves around On Curve Points**. If no points are selected it will apply to all points.

If an on-curve point lies almost on the line between its neighbour off curve points, it will be moved onto the line. If the 3 points are all almost horizontal or vertical, all 3 points will be aligned.

All points affected will be selected after the operation.

5.3.3.4 Join and Split Contours

Both Contour mode and Point mode have ways to combine and split contours. Usually Union, Intersection and Exclusion, available in Contour mode, and Knife, available in both modes, are recommended. If these features don't give expected results the two remaining features (Join Contours and Split Contour) available in

Points mode might help.

Union, Intersection and Exclusion in Contour mode

Use the **Union** feature to merge several overlapping contours.

Use the **Intersection** feature to keep all overlapping parts.

Use the **Exclusion** feature to remove all overlapping parts.

Knife in both modes

Use the **Knife** feature to split contours. Hold down the SHIFT key while using the Knife feature to force the cut line by 15° steps.

Join Contours and Split Contour in Point mode

When you want to combine two contours you have to select one point on each contour. Next right-click one of these points and click "Join Contours" on the shortcut menu.

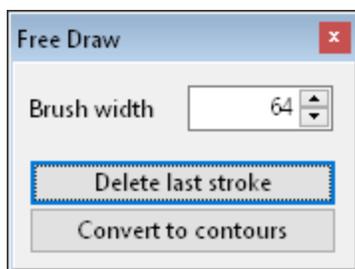
To split a contour into two contours, select two points (these points should not be neighbors) on the same contour and then right-click one of these points and click "Split Contour" on the shortcut menu.

Note: Make sure the two combining contours have the same direction.

Note: **Union, Intersection, Exclusion and Knife** are not available in the Home Edition of FontCreator.

5.3.3.5 Freedraw

The **Freedraw** tool can be used to manually draw lines that can be converted into contours. To enter **Freedraw** mode select **Freedraw Contours** from the **Insert** menu, or select the **Freedraw** tool from the **Drawing Toolbar**.



When the **Freedraw** tool is active, the Freedraw toolwindow will be visible. On this toolwindow you can set the Brush width, delete the last stroke or convert the drawn lines into contours.

To draw straight horizontal or vertical lines you can hold down the SHIFT key while drawing.

To erase part of the drawing, you can use the right mouse button as an eraser tool.

When you're satisfied, click on "Convert to contours" to add the contour to the currently displayed glyph (contours cannot be added to composite glyphs). To cancel, just close the Free Draw dialogue with the close button.

The new contour will not be joined to existing contours, even if it was drawn crossing them.

5.3.4 Composite Glyphs

5.3.4.1 Introduction

Composite glyphs are made out of "soft links" to one or more other glyphs - usually, but not limited to, a base character and one or more diacritical marks that are placed above and/or below the base character.

Create composite glyphs

You can create a composite glyph when you are editing an empty glyph in a **Glyph Edit** window. To add a composite glyph member select **Glyph** from the **Insert** menu. To add one or more glyphs as composite glyph members first copy glyphs from the **Font Overview** window and then paste them into a **Glyph Edit** window that contains an empty glyph or a composite glyph.

Move selected glyph members

You can change the position of each of the used glyphs within the **Glyph Edit** window.

To constrain a selection so it moves only horizontally or vertically, press Shift as you drag the selection. Press Alt to ignore the snap to grid and snap to guidelines features.

Transformations

Besides moving glyph members, it is possible to perform operations like scale and rotation by double-clicking on the glyph member or right-clicking and select **Glyph Member Properties** .

Duplicate selected glyph members

To copy selected composite glyph members to a new position, hold down the Ctrl key as you drag the selection. Hold the Alt key to ignore the snap to grid and snap to guidelines features. Hold down the Shift key to move only in vertical or horizontal direction. Any combination of these keys is allowed.

Convert to simple glyph

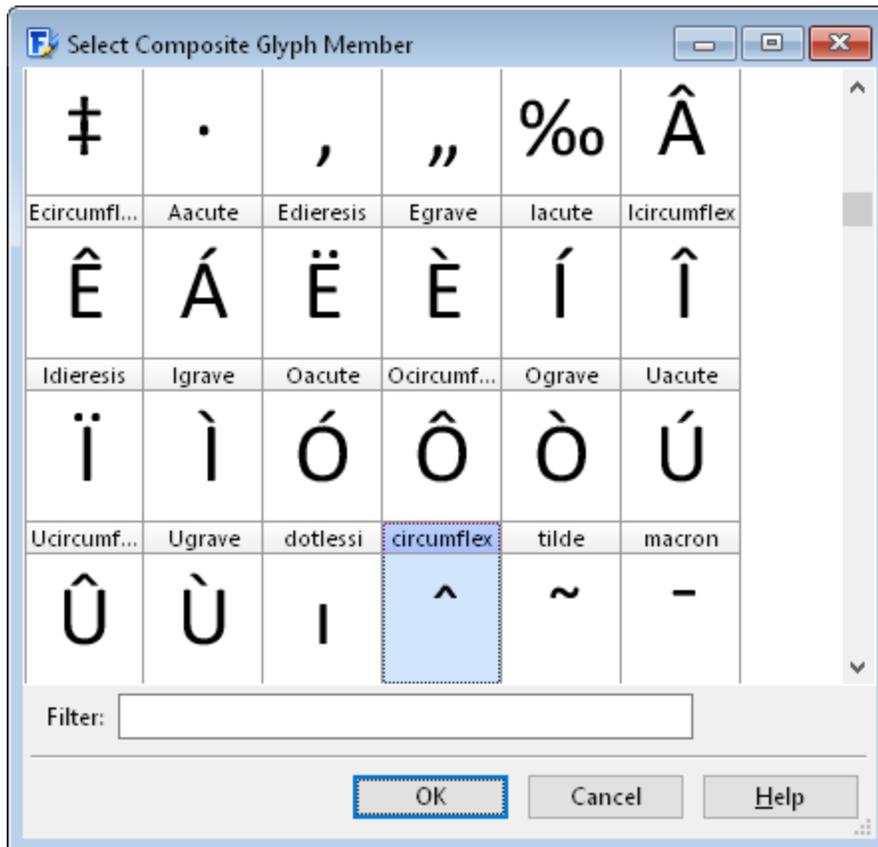
To convert a composite glyph to a simple glyph select the glyph in the **Font Overview** window or in the **Glyph Edit** window and select **Make Simple** from the **Edit** menu.

Join composite glyph members

If composite glyph members intersect such as with C cedilla, **Get Union of Contours** on the **Glyph** toolbar will convert the composite glyph to a simple glyph and join intersecting contours into one contour.

5.3.4.2 Add Glyph Member

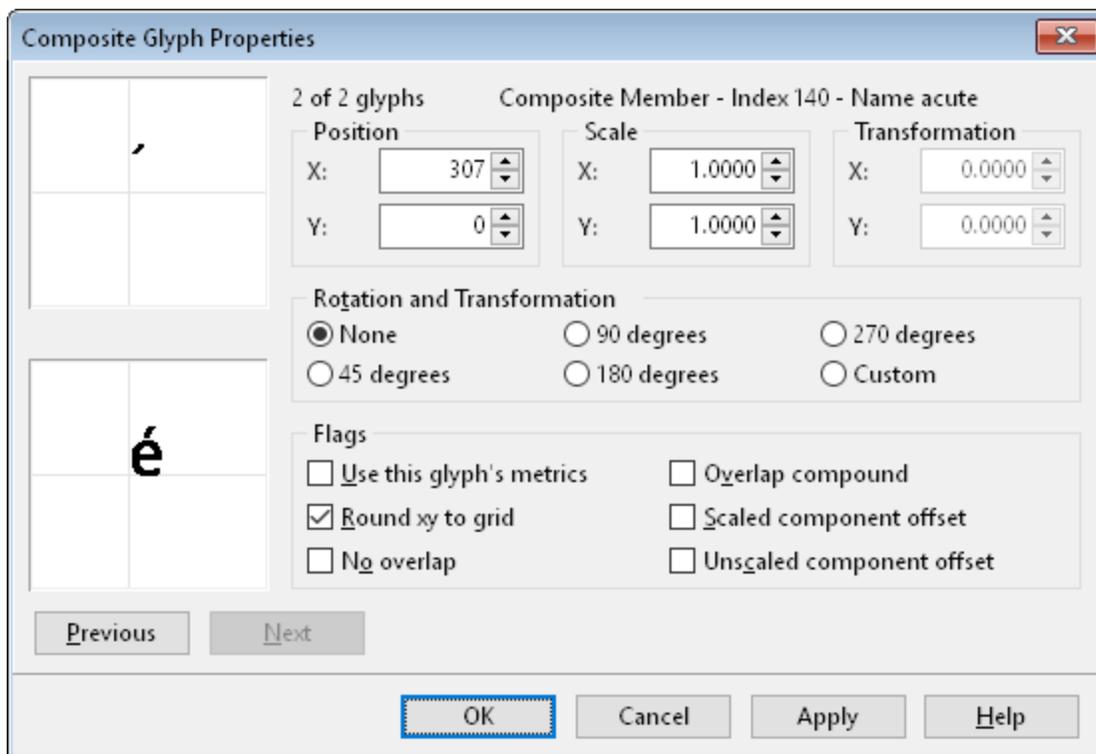
You can add a composite glyph member to an empty glyph, which will then become a composite glyph. You can also add more glyph members to existing composite glyphs. If you wish to edit an empty glyph or modify a composite glyph you can select **Glyph** from the **Insert** menu.



In the **Select Composite Glyph Member** window, select the glyph you want to add to the composite glyph and press the **OK** button.

5.3.4.3 Glyph Member Properties

When you have opened a composite glyph in the **Glyph Edit** window, you can modify the properties of each composite glyph member. Either double-click a Glyph Member or right-click it and then select **Glyph Member Properties**.



In the **Composite Glyph Properties** window you can modify the position, scale, rotation, transformation and special flags of the selected member. Use the **Previous** and **Next** buttons to walk through the members.

The Flags consist of a set of fields:

Use this glyph's metrics

If set, this forces the advance width and left side bearing (and right side bearing) for the composite to be equal to those from this original glyph. This works for hinted and unhinted characters. Only one member can have this option selected. Without this flag, the actual left side bearing and right side bearing of the composite glyph will be used.

Round xy to grid

Round the x and y offsets.

No overlap

This bit is reserved/obsolete. Do not set it.

Overlap compound

Used by Apple in GX fonts.

Scaled component offset

Composite designed to have the component offset scaled (designed for Apple rasterizer).

Unscaled component offset

Composite designed not to have the component offset scaled (designed for the Microsoft TrueType rasterizer).

Note that the behavior of the **Use this glyphs metrics** operation is undefined for rotated composite components.

5.3.4.4 Complete Composites

Use the **Complete Composites** feature to add composite glyph members to your glyphs. To use this powerful feature, select a glyph, or a range of glyphs, right-click and select **Complete Composites**. The selected glyphs will be composed using data in CompositeData.xml, which is read when FontCreator first uses the feature. This feature works with over two thousand glyphs that are defined in this file. To get the most out of this feature follow these guidelines:

- The selected glyph(s) must be mapped with a Windows Unicode mapping.
- If the selected glyph(s) are empty or simple they will be replaced wherever composites are defined. **Do not include simple glyphs in your selection if you don't wish to replace them with composites.**
- If the selected glyph(s) are already composites they will be replaced only if the composite glyph members are different, not if their positions or scale factors are different.
- All composite members must be present in the font. If any composite members are mapped, but still empty, the composite glyph will be completed, though obviously missing the contours that have not been defined yet.

Info: Read the documentation about the content of the file CompositeData.xml and an explanation about how to modify and add glyphs. This document is available from our website:

<http://www.high-logic.com/font-editor/fontcreator/tutorials.html> 

Note: **Complete Composites** is not available in the Home Edition of FontCreator.

5.3.5 Glyph Metrics

Several horizontal and vertical lines, also known as glyph metrics, will help you

with your font design. You can activate them through the [Metrics Options](#) window.

To change the metrics, click **Properties** on the **Font** menu and then click the **Metrics** tab.

In the Glyph Edit window there are two vertical lines (normally the glyph outline lies in between these lines) that represent the left and right side bearings. These are shown by default but you can hide them through the Show Metrics button on the Drawing toolbar. The left and right side bearings can be changed by dragging them to their desired position. You can also adjust the bearings through the **Glyph Properties** tool window. White space should be evenly distributed between the left and right side bearings of glyphs except when font is specifically designed with ligatures for joining characters in a cursive script font.

Nonspacing combining marks should have a zero advance width. They are used in mark to base, mark to ligature, and mark to mark lookups in the OpenType Designer.

Note: Lining numbers (i.e. the digits 0 - 9) should all have the same advance width.

Tip: In the **Glyph Edit** window you can step through with the **Back** and **Forward** blue arrows in the **Drawing** toolbar to make adjustments.

See also:

[Metrics Options](#)

5.3.6 Color Glyphs

5.3.6.1 Introduction

Each glyph can have its own color information assigned to it. If a host application does not support the color font extension, the regular outlines will be shown. There are a couple of similarities with making a composite glyph, but there are also several differences.

Create a single color glyph

The easiest way to create a color version of a glyph is to select the "Colorize" option from the right-click menu when editing a glyph. This will automatically add the current glyph and assign a single color to it.

Create a multi-color glyph

To make a multi-color glyph, you need to add other glyphs and then define a color for each of these color members. Unlike composite glyph members, you can't change the size or position of these color members. You will need to make sure the individual glyphs are all designed to fit into the color glyph. One other important requirement is the fact that the metrics of the base glyph and the color members must be the same. So the offsets (usually set to 0) and the advance widths have to match.

Remove Color

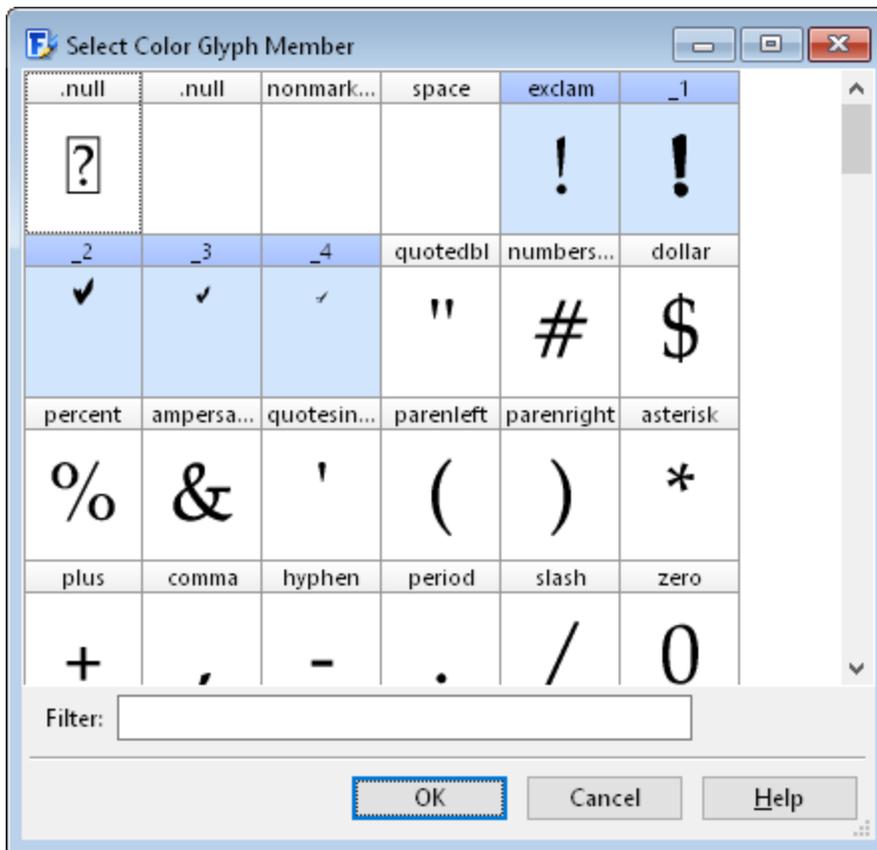
To remove all color information from a glyph, do right-click it in the glyph overview window, and select "Decolorize".

Switching between color and normal mode

When editing a glyph, click the "Color Mode" button  or select **Color Mode** from the **View** menu.

5.3.6.2 Add Glyph Member

To add one or more Color Glyph members, make sure you are in **Color Mode** and right-click in the Glyph Edit window and select **Add** or click the Add icon .



In the **Select Color Glyph Member** window select the glyphs you want to add to the color glyph and press the **OK** button.

You can change the order of the color glyph members via the up and down buttons on the [Color Glyph Member](#) toolbar, the **Order** options in the right-click menu or via the **Alignment** toolbar. To show and/or hide toolbars, select **Toolbars** on the **View** menu.

5.3.6.3 Palettes and Colors

A font can contain one or more palettes that allow a host application to quickly change the color scheme. By using palettes you can make different color schemes that allow your font to use different colors with, for example a different background color.

The palette colors are global, so if you change a color which is assigned to several color members, then all those members will use the updated color. If you want a unique color for a specific color glyph member, then add a color to the palette and use that instead.

When in **Color Mode** you can assign colors to specific members, by selecting the **Paint Bucket** tool  from the **Drawing Toolbar** and click on one of the color glyph members in the Glyph Edit window. You can change the current color by clicking on one of the palette entries in the **Palette Toolbar**.

See also:

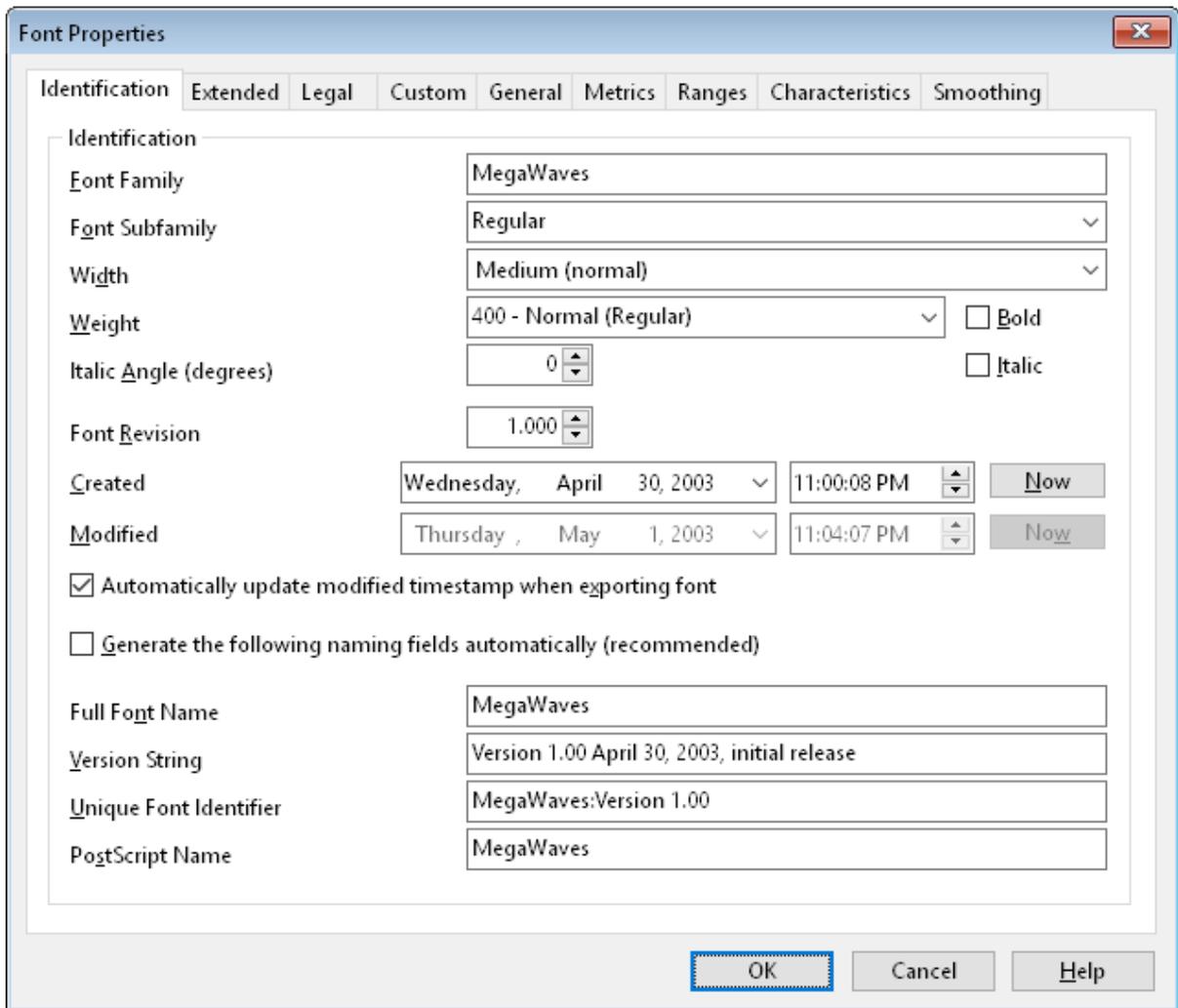
[Palette](#)

5.4 Format

5.4.1 Properties

5.4.1.1 Identification

Global and horizontal layout information about the font is found on the **Identification** page (in the **Font Properties** window). On the **Font** menu, click **Properties**, and then click the **Identification** tab.



The screenshot shows the 'Font Properties' dialog box with the 'Identification' tab selected. The dialog has a title bar with a close button (X) and a tabbed interface with the following tabs: Identification, Extended, Legal, Custom, General, Metrics, Ranges, Characteristics, and Smoothing. The 'Identification' tab is active and contains the following fields and controls:

- Font Family:** MegaWaves
- Font Subfamily:** Regular (dropdown)
- Width:** Medium (normal) (dropdown)
- Weight:** 400 - Normal (Regular) (dropdown) with checkboxes for **Bold** and **Italic**.
- Italic Angle (degrees):** 0 (spin box)
- Font Revision:** 1.000 (spin box)
- Created:** Wednesday, April 30, 2003 (dropdown) 11:00:08 PM (dropdown) **Now** (button)
- Modified:** Thursday, May 1, 2003 (dropdown) 11:04:07 PM (dropdown) **Now** (button)
- Automatically update modified timestamp when exporting font**
- Generate the following naming fields automatically (recommended)**
- Full Font Name:** MegaWaves
- Version String:** Version 1.00 April 30, 2003, initial release
- Unique Font Identifier:** MegaWaves:Version 1.00
- PostScript Name:** MegaWaves

At the bottom of the dialog are three buttons: **OK**, **Cancel**, and **Help**.

In general it is strongly recommended to only use letters, digits and the space character for the naming fields. For example the Mac refuses fonts which contain a quote in the name, so **Jack's font** will fail to work on a Mac. For CJK fonts, you can use custom naming fields to provide names for Chinese, Japanese, Korean, etc.

Font Family

The name the user sees. Maximum length is 31 characters.

Font Subfamily

The name of the style.

Width

Indicates a relative change from the normal aspect ratio (width to height ratio) as specified by a font designer for the glyphs in a font.

Note: Although every character in a font may have a different numeric aspect ratio, each character in a font of normal width has a relative aspect ratio of one. When a new type style is created of a different width class (either by a font designer or by some automated means) the relative aspect ratio of the characters in the new font is some percentage greater or less than those same characters in the normal font -- it is this difference that this parameter specifies.

Weight

Indicates the visual weight (degree of blackness or thickness of strokes) of the characters in the font.

Note: Windows automatically adds fake bold to any font with weight set to either Thin or Extra-light (Ultra-light), so better avoid these weights.

Italic Angle

Italic angle in degrees from the vertical. Zero for upright text, positive for text that leans to the right (forward).

Font revision

Set by the font manufacturer.

Note: For historical reasons, the **Font revision version** is not used by Windows to determine the version of a font. Instead, Windows evaluates the version string from the **Version String** field.

Created

The date and time the font was created, press the **Now** button to set these fields to the

current date and time.

Modified

The date and time the font was last modified, press the **Now** button to set these fields to the current date and time.

Automatically update modified timestamp when exporting font

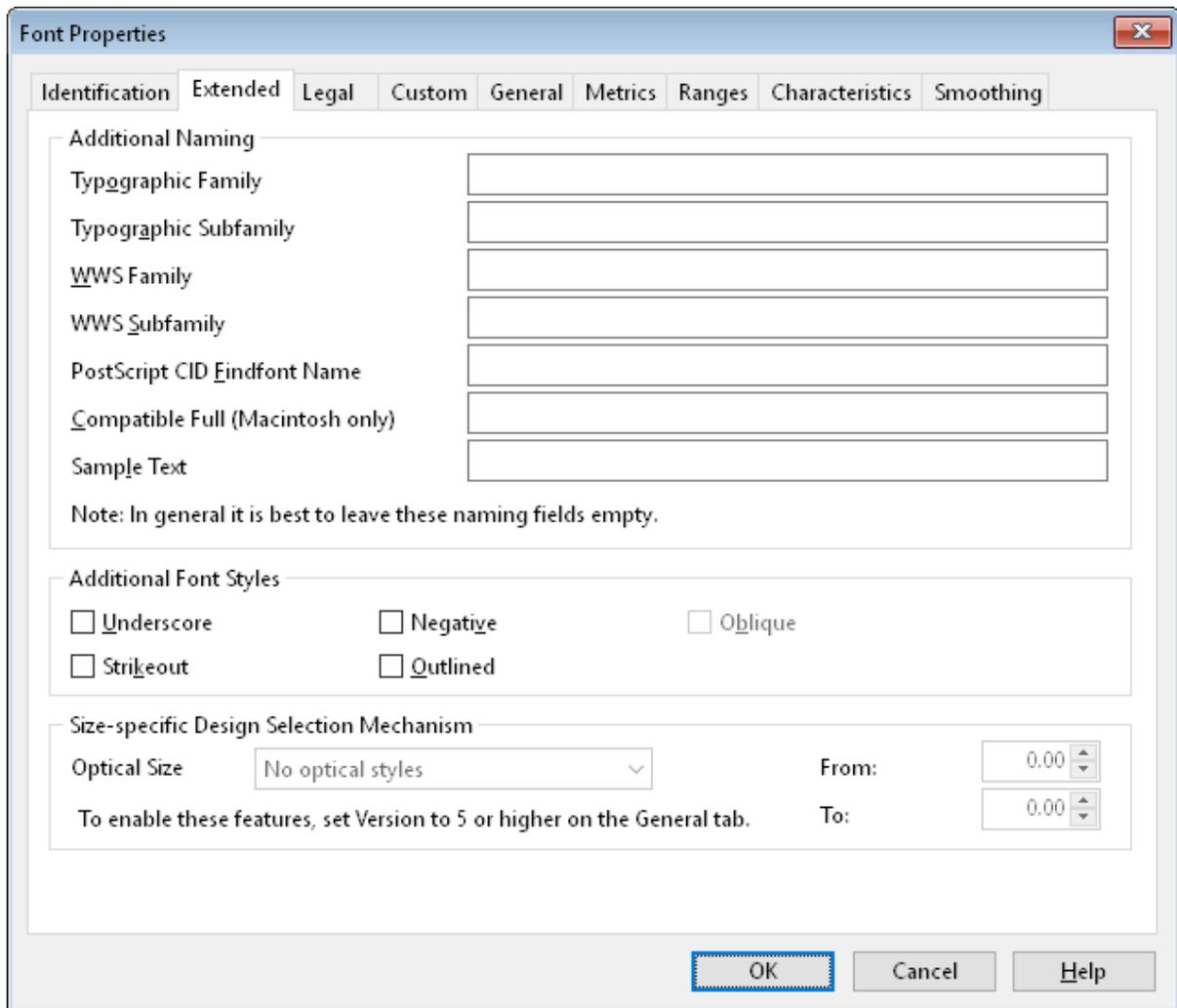
When checked, the timestamp will be automatically set to the current date and time when you export a font.

Generate the following naming fields automatically (recommended)

When checked, the **Full Font Name**, **Version String**, **Unique Font Identifier** and **PostScript Name** fields are automatically updated

5.4.1.2 Extended

The **Extended** page is used to alter the **Additional Naming** and **Additional Font Styles** fields. On the **Font** menu, click **Properties**, and then click the **Extended** tab.



Typographic Family & Typographic Subfamily (Windows only)

In Windows, the Family name is displayed in the font menu; the Subfamily name is presented as the Style name. For historical reasons, font families have contained a maximum of four styles, but font designers may group more than four fonts to a single family. The Typographic Family and Typographic Subfamily IDs allow font designers to include the preferred family/subfamily groupings. These IDs are only present if they are different from fields Font Family name and Font Subfamily name.

WWS Family

The WWS Family name *

WWS Subfamily

The WWS Subfamily name *

PostScript CID Findfont Name

Its presence in a font means that the Postscript name field in the Naming window holds a PostScript font name that is meant to be used with the "composefont" invocation in order to invoke the font in a PostScript interpreter.

This field must be restricted to the printable ASCII subset, codes 33 through 126, except for the 10 characters: '[' , ']' , '(' , ')' , '{' , '}' , '<' , '>' , '/' , '%' .

Compatible Full (Macintosh only)

On the Macintosh, the menu name is constructed using the FOND resource. This usually matches the Full Name. If you want the name of the font to appear differently than the Full Name, you can insert the Compatible Full Name in this field.

Sample Text

This can be the font name, or any other text that the designer thinks is the best sample to display the font in.

Additional Font Styles

These are rarely used styles that indicate the font is primarily a decorative or special purpose font.

Underscore: Characters are underscored

Strikeout: Characters are overstruck

Negative: Characters have their foreground and background reversed

Outlined: Outline (hollow) characters, otherwise they are solid

Oblique: Font contains oblique characters (available if Version is set to 4 or higher on the General tab)

Size-specific Design Selection Mechanism

These fields are used for fonts with multiple optical styles. When working across multiple optical fonts, there should be no intentional gaps or overlaps in the ranges.

From: This value (in points) is the lower value of the size range for which this font has been designed. The value is inclusive—meaning that the font was designed to work best at this point size through, but not including, the point size indicated **To**. When used with other optical fonts that set **From** and **To**, it would be expected that another font has this same value as this entry in the **To** field, unless this font is designed for the lowest size range.

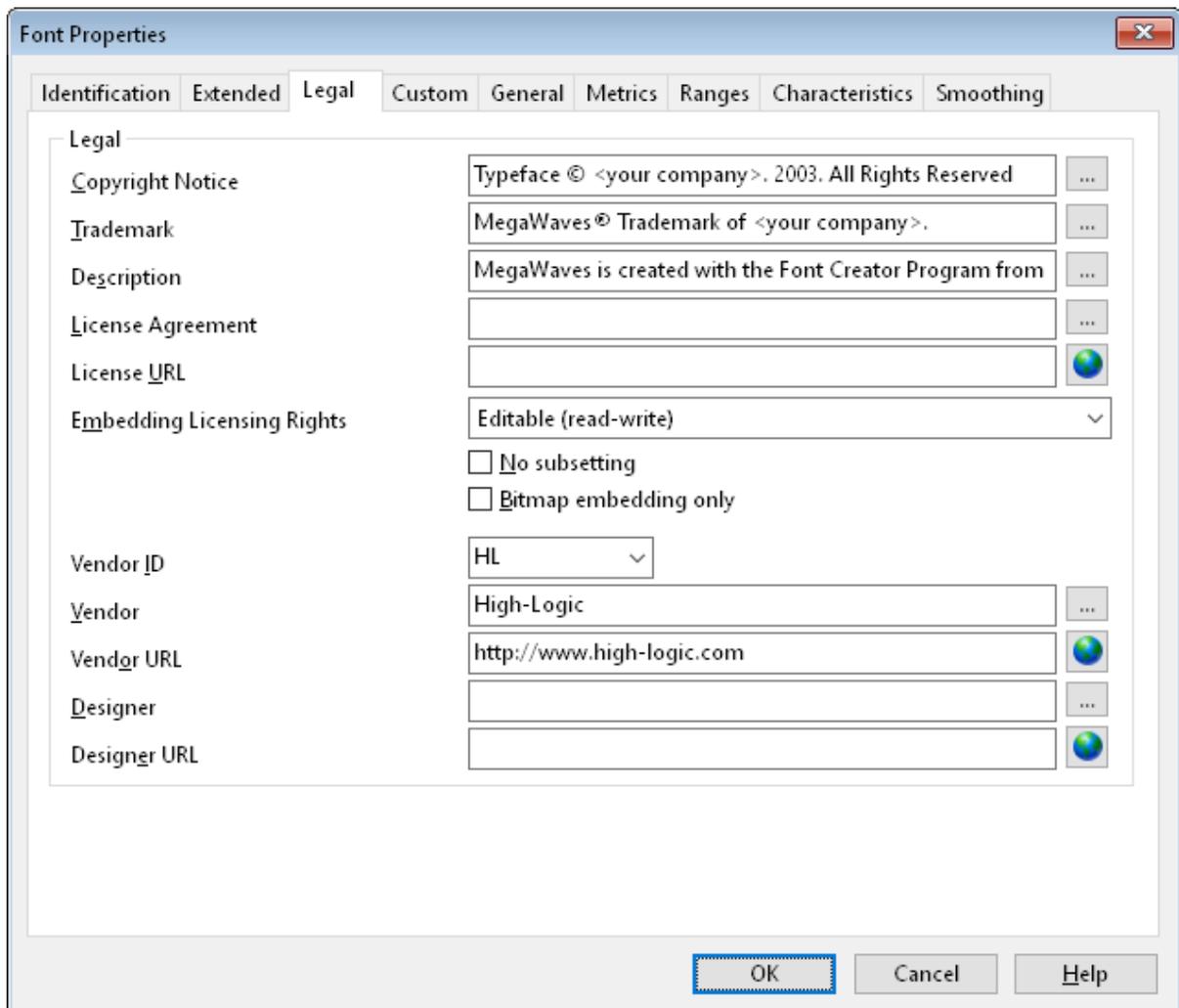
To: This value (in points) is the upper value of the size range for which this font

has been designed. The value is exclusive—meaning that the font was designed to work best below this point size down to the **From** threshold. When used with other optical fonts that set **From** and **To**, it would be expected that another font has this same value as this entry in the **From** field, unless this font is designed for the highest size range.

*) For more information about WWS please visit our forums at <http://forum.high-logic.com> or visit this website: <http://blogs.msdn.com/b/text/archive/2007/04/23/wpf-font-selection-model.aspx>

5.4.1.3 Legal

The **Legal** page is used to enter the Copyright, Trademark, Licensing, Vendor, and Designer information. On the **Font** menu, click **Properties**, and then click the **Legal** tab.



Some Naming fields can be modified within the **Edit Naming Field** window accessed by pressing the [...] button.

Embeddable fonts may be stored in a document. When a document with embedded fonts is opened on a system that does not have the font installed (the remote system), the embedded font may be loaded for temporary (and in some cases, permanent) use on that system by an embedding-aware application. Embedding licensing rights are granted by the vendor of the font.

Note: If multiple embedding flags are set, the least restrictive license granted takes precedence. For example, **Restricted License embedding** and **Editable embedding** are set, **Editable embedding** takes precedence over **Restricted License embedding** and the font may be embedded with editable rights. For compatibility purposes, most vendors granting Editable embedding rights are also setting the **Preview & Print** flag. This will permit an application that only supports **Preview & Print** embedding to detect that font embedding is allowed.

Installable (no embedding restrictions)

Fonts that have none of the flags set are installable embedding fonts. Fonts with this setting indicate that they may be embedded and permanently installed on the remote system by an application. The user of the remote system acquires the identical rights, obligations and licenses for that font as the original purchaser of the font, and is subject to the same end-user license agreement, copyright, design patent, and/or trademark as was the original purchaser.

Editable (read-write)

Fonts with this flag set indicate that they may be embedded in documents, but must only be installed temporarily on the remote system. In contrast to Preview & Print fonts, documents containing Editable fonts may be opened "read-write;" editing is permitted, and changes may be saved.

Preview & Print (read-only)

Fonts with this flag set indicate that they may be embedded within documents but must only be installed temporarily on the remote system. Any document which includes a Preview & Print embedded font must be opened "read-only;" the application must not allow the user to edit the document; it can only be viewed and/or printed.

Restricted (embedding is not allowed)

Fonts that have this flag set must not be modified, embedded or exchanged in any manner without first obtaining permission of the legal owner. Caution: note that for Restricted License embedding to take effect, it must be the only level of embedding selected (as noted in the previous paragraph).

No subsetting

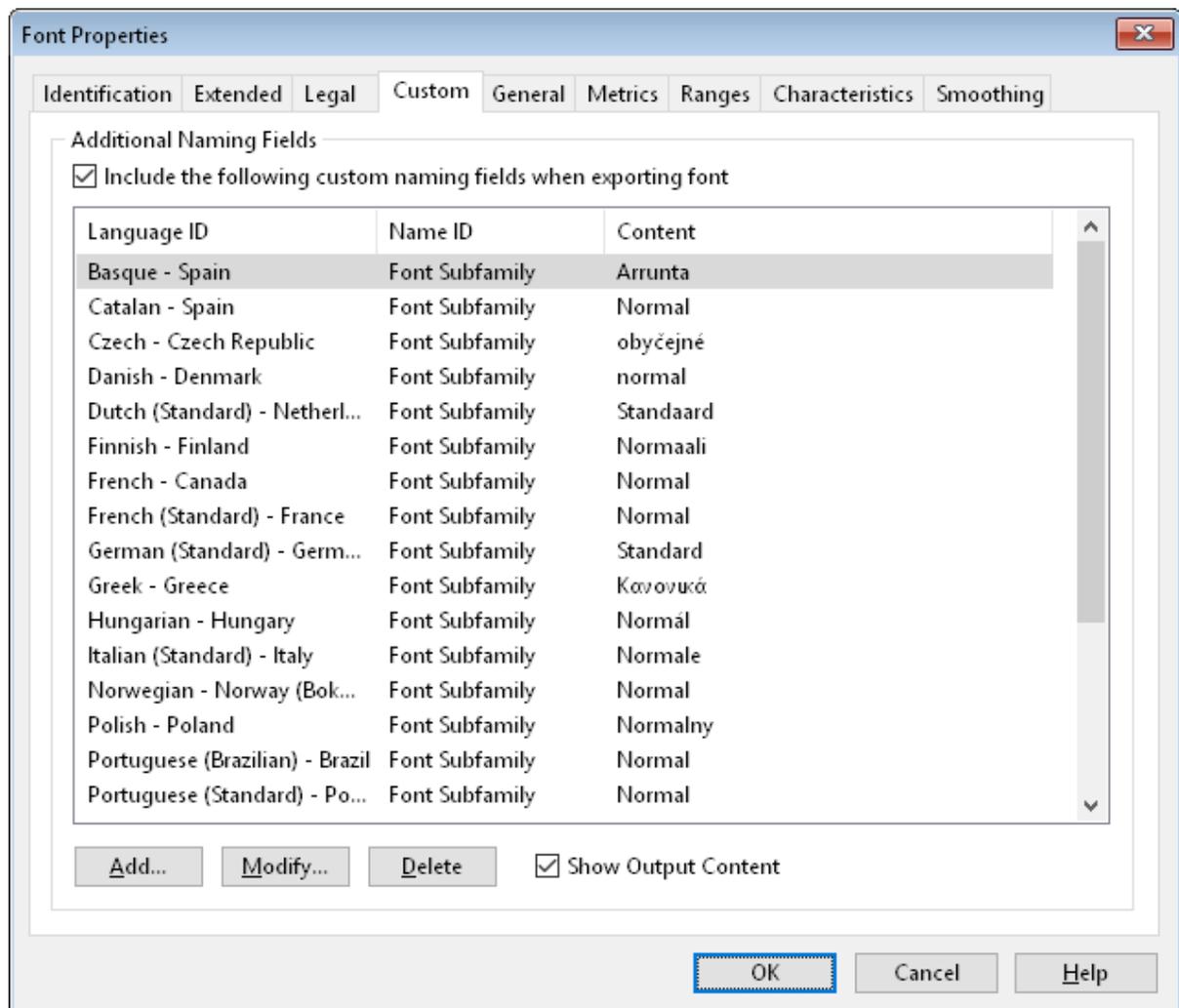
When this flag is set, the font may not be subsetted prior to embedding.

Bitmap embedding only

When this flag is set, only bitmaps contained in the font may be embedded. No outline data may be embedded.

5.4.1.4 Custom

The **Custom** page is used to add **Additional Naming Fields**. On the **Font** menu, click **Properties**, and then click the **Custom** tab.

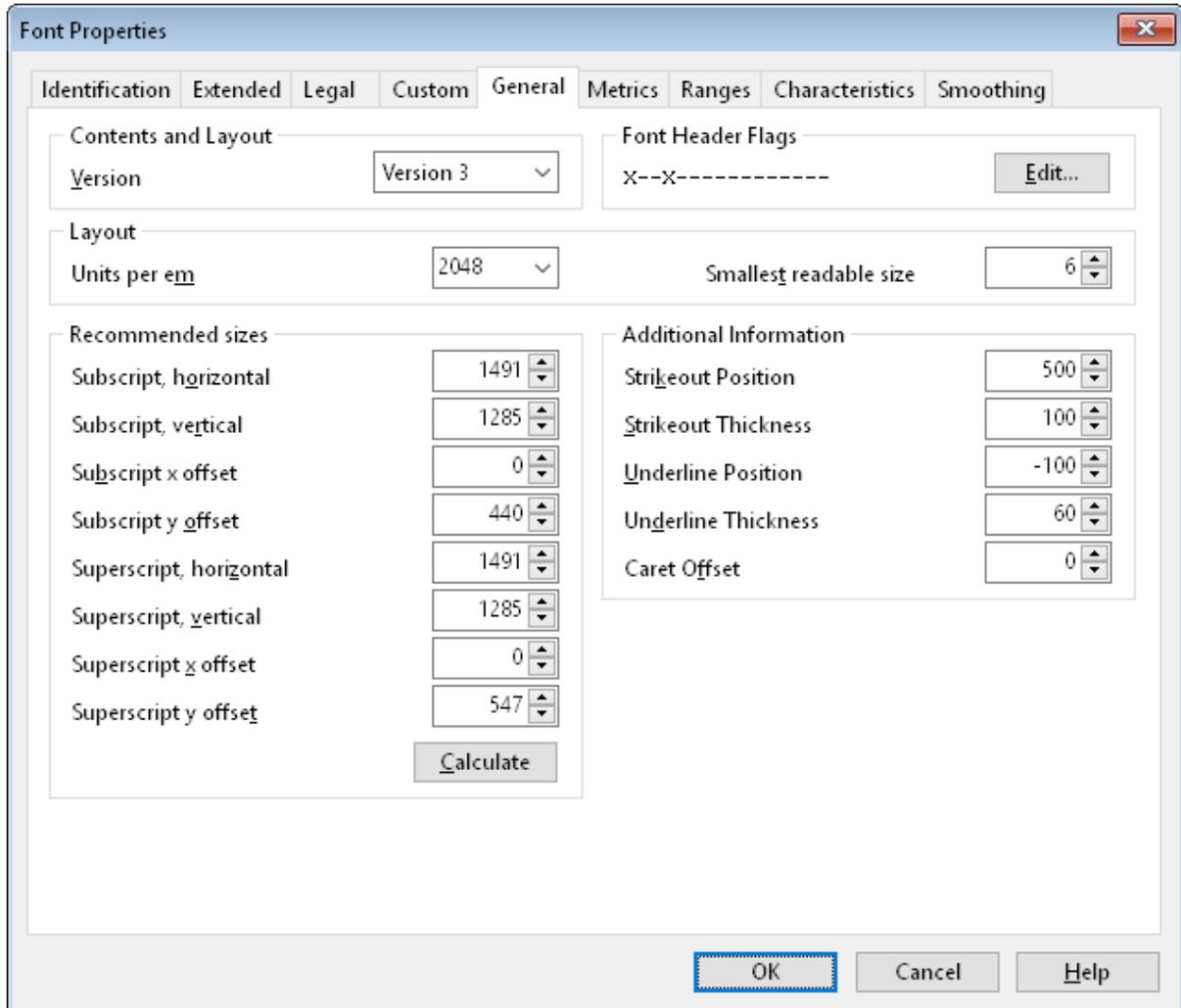


Custom naming fields are mostly used to add language specific Font Subfamily entries, but are also used for CJK fonts to specify Chinese, Japanese, or Korean naming fields.

5.4.1.5 General

5.4.1.5.1 Overview

This page consists of a set of metrics. On the **Format** menu, click **Settings**, and then click the **General** tab.



Contents and Layout Version

It can take years before operating systems and software support changes to the OpenType specification. For compatibility reasons it is possible to choose between these versions:

Version 3: Released October 2002.

Version 4: Released May 2008. Added **Oblique** as additional font style, **use typo metrics**, and **WWS support**

Version 5: Released March 2015. Added **Size-specific Design Selection**

Mechanism

Font Header Flags

This field is used to specify the global information about the font. Press the Edit button to modify this field through the [Font Header Flags window](#).

Units Per Em

Valid range is from 16 to 16384. This value should be a power of 2 for fonts that have TrueType outlines. Nowadays 2048 units per em value is the best value for all fonts of any size including large Latin or non-Latin script fonts.

For OpenType fonts with CFF based outlines it is recommend to set units per em to 1000.

This value is used to convert values in the pixel coordinate system by multiplying them by a scale. This scale is:

$$\text{PointSize} * \text{resolution} / (72 \text{ points per inch} * \text{units_per_em})$$

Smallest read. size

Smallest readable size in pixels. Most fonts have this value set to 9.

Recommended sizes:

Note: Word processors that support OpenType features will probably use the Subscript (subs) and Superscript (sups) features, if available in the font.

Subscript, horizontal

The recommended horizontal size in font design units for subscripts for this font.

Subscript, vertical

The recommended vertical size in font design units for subscripts for this font.

Subscript x offset

The recommended horizontal offset in font design units for subscripts for this font.

Subscript y offset

The recommended vertical offset in font design units from the baseline for subscripts for this font.

Superscript, horizontal

The recommended horizontal size in font design units for superscripts for this font.

Superscript, vertical

The recommended vertical size in font design units for superscripts for this font.

Superscript x offset

The recommended horizontal offset in font design units for superscripts for this font.

Superscript y offset

The recommended vertical offset in font design units from the baseline for superscripts for this font.

Calculate

It is up to word-processing software to use these values for superscript and subscript effects. However, be aware that not all software uses these values uniformly, so when the default superscripts and subscripts look correct in one application, they might look wrong in another application. The calculate button calculates the values so that they look correct in Microsoft Word 2010.

Strikeout size

Width of the strikeout stroke in font design units. This field should normally be the width of the em dash for the current font. If the size is one, the strikeout line will be the line represented by the strikeout position field. If the value is two, the strikeout line will be the line represented by the strikeout position and the line immediately above the strikeout position. For a Roman font with a 2048 em square, 102 is suggested.

Strikeout position

The position of the strikeout stroke relative to the baseline in font design units. The value of strikeout position should not interfere with the recognition of standard characters, and therefore should not line up with crossbars in the font. For a Roman font with a 2048 em square, 460 is suggested.

Underline Position

Suggested values for the underline position (negative values indicate below baseline).

Underline Thickness

Suggested values for the underline thickness.

Note: Word processing applications decide whether they use these `underlinePosition` and `underlineThickness` values or use their own defaults.

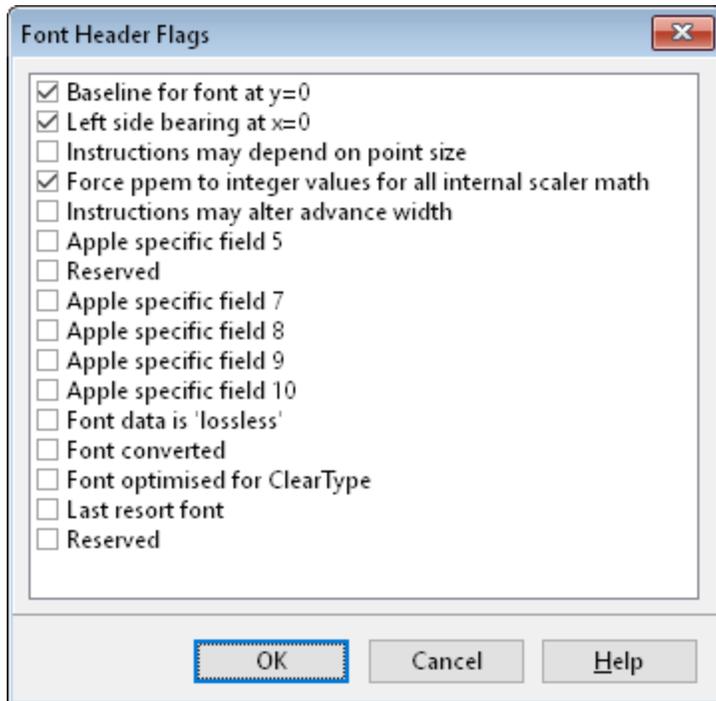
Warning: Some Word processors may crash when this value is set to zero.

Caret Offset

The amount by which a slanted highlight on a glyph needs to be shifted to produce the best appearance. Set to 0 for non-slanted fonts.

5.4.1.5.2 Font Header Flags

The Font Header Flags give global information about the font.



Baseline for font at y=0

Set this field when the baseline for all glyphs is at $y = 0$ (that is, the x-axis).

Left side bearing point at x=0

Set this field when the x-position of the leftmost black bit is assumed to be the left side bearing. This field can be ignored, as FontCreator will ensure this field is correct when a font is exported.

Instructions may depend on point size

Set this flag when there are instructions that depend on point size. Instructions may use point size explicitly in place of pixels per em. This means that scaling a 12 point screen font to obtain the equivalent printer font may not produce the identical result as requesting a 12 point printer font.

Force ppeM to integer values for all internal scaler math

When this field is selected integer scaling will be used instead of fractional scaling. Fractional ppeM sizes may be used if this field is not selected.

Instructions may alter advance width

When this field is set it allows the font to alter device dependent widths (the advance widths might not scale linearly).

Apple specific field 5

This field should be set in fonts that are intended to be laid out vertically, and in which the glyphs have been drawn such that an x-coordinate of 0 corresponds to the desired vertical baseline.

Reserved

Reserved, do not check this field

Apple specific field 7

This field should be set if the font requires layout for correct linguistic rendering (e.g. Arabic fonts).

Apple specific field 8

This field should be set for a GX font which has one or more metamorphosis effects designated as happening by default.

Apple specific field 9

This field should be set if the font contains any strong right-to-left glyphs.

Apple specific field 10

This field should be set if the font contains Indic-style rearrangement effects.

Font data is 'lossless'

Set this flag when font data is 'lossless', as a result of having been compressed and decompressed with the Agfa MicroType Express engine.

Font converted

Set this flag when the font is converted (produce compatible metrics).

Font optimised for ClearType

Set this field if the font is optimised for ClearType

Last resort font

If set, indicates that the glyphs are simply generic symbolic representations of the assigned code point ranges and don't truly represent support for those code points. If unset, indicates that the glyphs represent proper support for those code points.

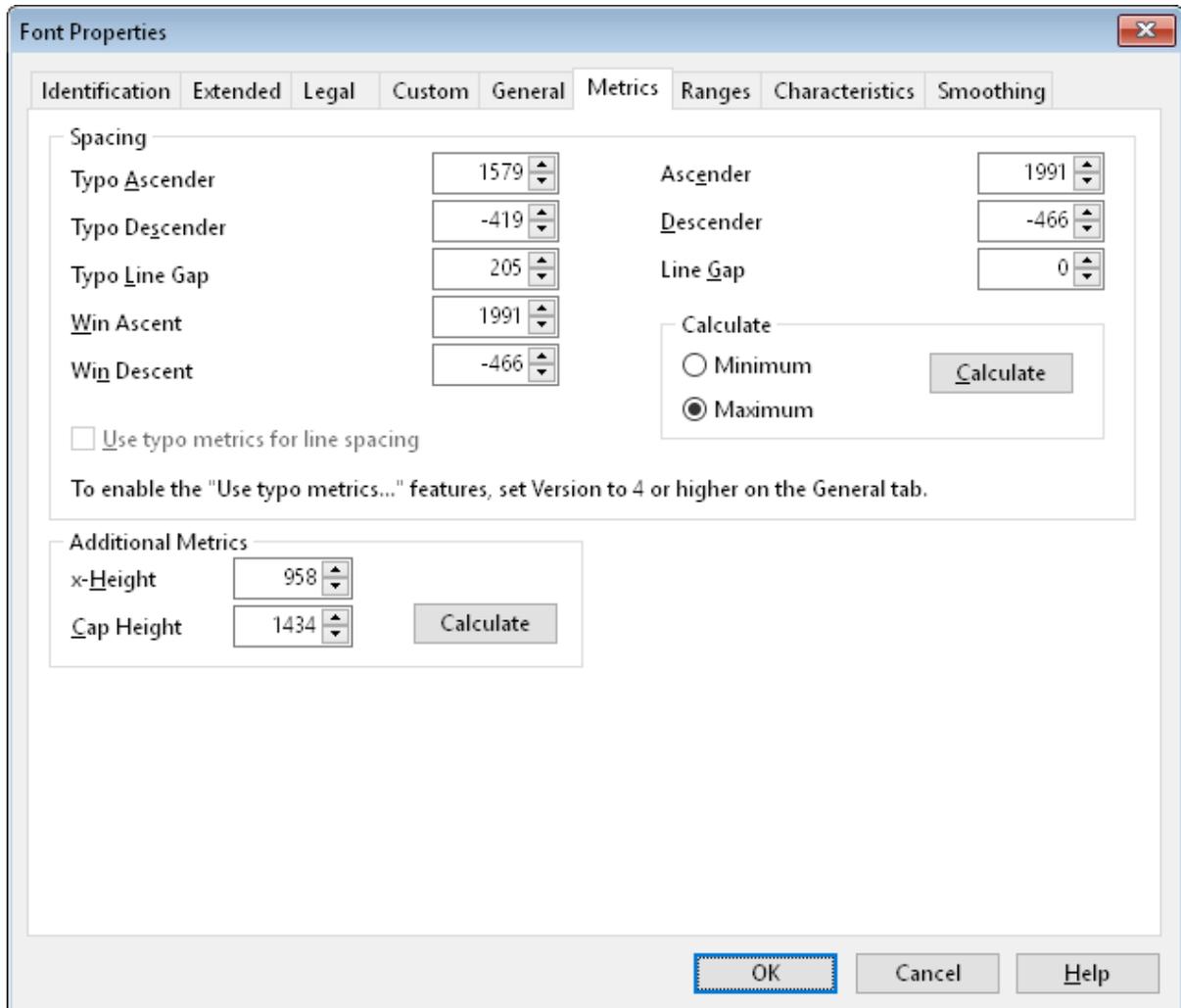
Reserved

Reserved, do not check this field

Note: The Apple specific fields should be set according to Apple's specification. However, they are not implemented in OpenType.

5.4.1.6 Metrics

This page consists of a set of spacing fields and font design flags. On the Format menu, click **Settings**, and then click the **Metrics** tab.



Typo Ascender

The typographic ascender for this font. Remember that this is not the same as the **Ascender** value in the **Metrics** tab, which Apple defines in a far different manner.

The suggested usage for **Typo Ascender** is that it be used in conjunction with **units per em** to compute a typographically correct default line spacing. The goal is to free applications from Macintosh or Windows-specific metrics which are constrained by backward compatibility requirements. These new metrics, when combined with the character design widths, will allow applications to lay out

documents in a typographically correct and portable fashion.

For CJK (Chinese, Japanese, and Korean) fonts that are intended to be used for vertical writing (in addition to horizontal writing), the required value for **Typo Ascender** is that which describes the top of the design space (also known as em-square). For example, if the design space of the font extends from coordinates 0,-120 to 1000,880 (that is, a 1000x1000 box set 120 design units below the Latin baseline), then the value of **Typo Ascender** must be set to 880. Failing to adhere to these requirements will result in incorrect vertical layout.

Typo Descender

The typographic descender for this font. Remember that this is not the same as the **Descender** value in the **Metrics** tab, which Apple defines in a far different manner.

The suggested usage for **Typo Descender** is that it be used in conjunction with units per em to compute a typographically correct default line spacing. The goal is to free applications from Macintosh or Windows-specific metrics, which are constrained by backward compatibility requirements. These new metrics, when combined with the character design widths, will allow applications to lay out documents in a typographically correct and portable fashion.

For CJK (Chinese, Japanese, and Korean) fonts that are intended to be used for vertical writing (in addition to horizontal writing), the required value for **Typo Descender** is that which describes the bottom of the design space (aka, em-square). For example, if the design space of the font extends from coordinates 0,-120 to 1000,880 (that is, a 1000x1000 box set 120 design units below the Latin baseline), then the value of **Typo Descender** must be set to -120. Failing to adhere to these requirements will result in incorrect vertical layout.

Typo Line Gap

The typographic line gap for this font. Remember that this is not the same as the **Line Gap** value, which Apple defines in a far different manner.

The suggested usage for **Typo Line Gap** is that it be used in conjunction with **units per em** to compute a typographically correct default line spacing. Typical values average 7-10% of units per em.

Win Ascent

The ascender metric for Windows. This, too, is distinct from Apple's Ascender value and from the **Typo Ascender** value. **Win Ascent** is computed as the yMax for all characters in the Windows ANSI character set. Win Ascent is used to compute the Windows font height and default line spacing. For **Symbol fonts**, it is the same as yMax.

Win Descent

The descender metric for Windows. This, too, is distinct from Apple's Descender value and from the **Typo Descender** value. **Win Descent** is computed as the -yMin for all characters in the Windows ANSI character set. **Win Descent** is used to compute the Windows font height and default line spacing. For **Symbol fonts**, it is the same as -yMin.

Use typo metrics for line spacing

If set, it is strongly recommended to use **Typo Ascender - Typo Descender + Typo Line Gap** as a value for default line spacing for this font.

Note: this field can only be used with Contents and Layout version 4 or higher. That version field is available on the General tab.

Ascender (Macintosh-specific)

Typographic ascent

Descender (Macintosh-specific)

Typographic descent

Line Gap (Macintosh-specific)

Typographic line gap. Negative Line Gap values are treated as zero.

Tip: To automatically calculate ascender and descender values press the Calculate button.

Additional Metrics

x-Height

This metric specifies the distance between the baseline and the approximate height of non-ascending lowercase letters measured in Funits. This value would normally be specified by a type designer but in situations where that is not possible, for example when a legacy font is being converted, the value may be set equal to the top of the unscaled and unhinted glyph bounding box of the glyph encoded at U+0078 (LATIN SMALL LETTER X). If no glyph is encoded in this position the field should be set to 0.

This metric, if specified, can be used in font substitution: the xHeight value of one font can be scaled to approximate the apparent size of another.

CapHeight

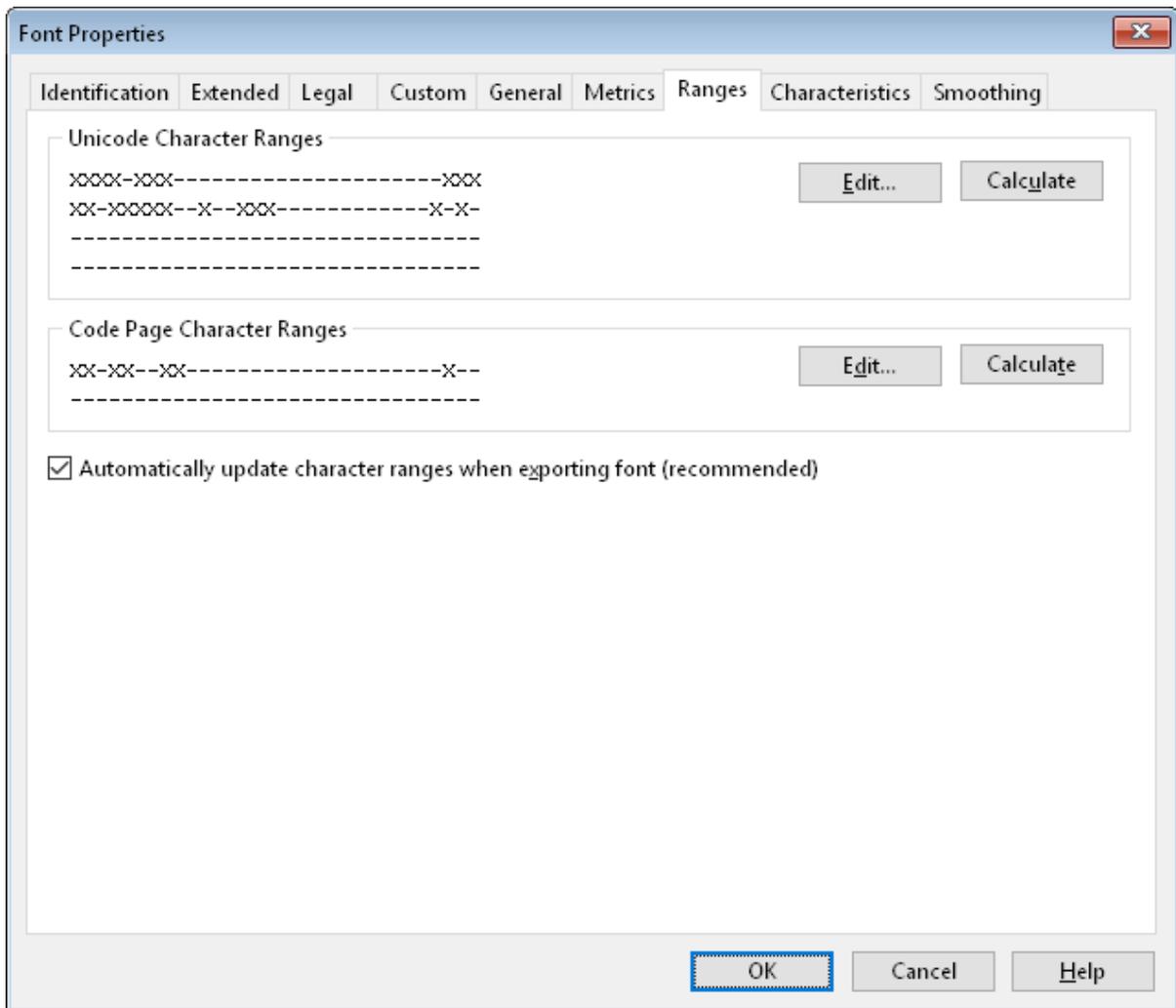
This metric specifies the distance between the baseline and the approximate height of uppercase letters measured in Funits. This value would normally be specified by a type designer but in situations where that is not possible, for example when a legacy font is being converted, the value may be set equal to the top of the unscaled and unhinted glyph bounding box of the glyph encoded at U+0048 (LATIN CAPITAL LETTER H). If no glyph is encoded in this position the field should be set to 0.

This metric, if specified, can be used in systems that specify type size by capital height measured in millimeters. It can also be used as an alignment metric; the top of a drop capital, for instance, can be aligned to the CapHeight metric of the first line of text.

5.4.1.7 Ranges

5.4.1.7.1 Overview

The **Ranges** page on the **Font Properties** window has several fields related to character ranges and additional metrics. On the **Font** menu, click **Properties**, and then click the **Ranges** tab.



Unicode Character Ranges

This field is used to specify the Unicode blocks or ranges encompassed by the font file in the **mappings** for the **Windows platform**. The Ranges depend on the selected Content and Layout version. Press the **Edit** button to modify this field through the [Unicode Character Range window](#) or press the **Calculate** button to generate the value.

Code Page Character Ranges

This field is used to specify the code pages encompassed by the font file in the **mappings** for the **Windows platform**. Press the **Edit** button to modify this field through the [Code Page Character Range window](#) or press the **Calculate** button to generate the value.

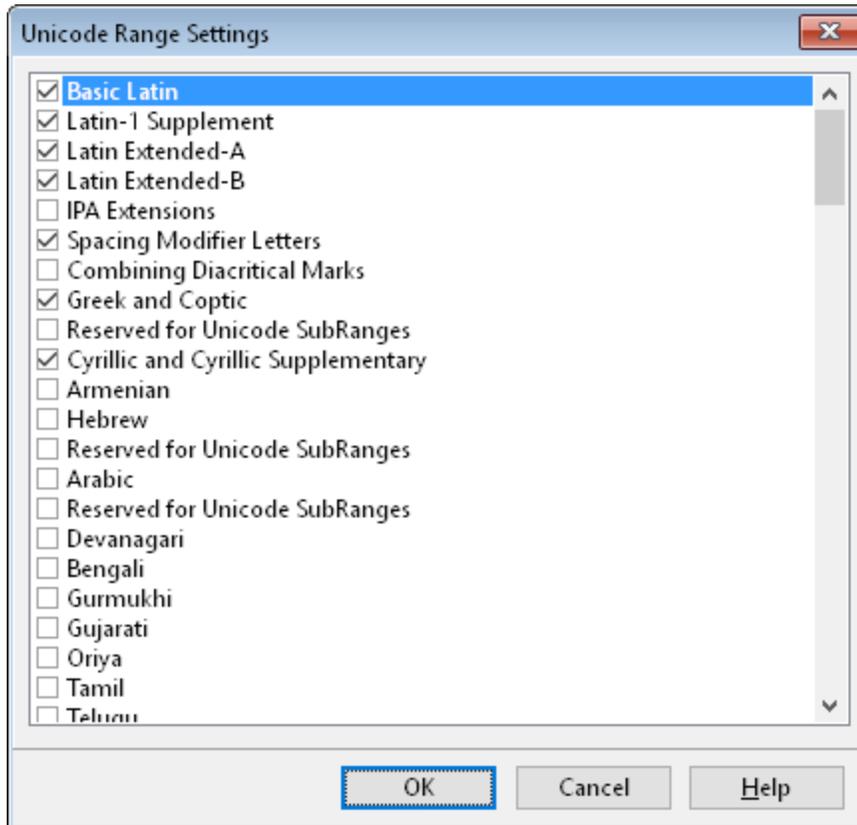
Automatically update character ranges when exporting font (recommended)

When enabled the character ranges will be updated when the font is exported. This will make sure that the ranges are always correct.

Note: The specific content of the ranges depend on the Contents and Layout version as available on the General tab. You don't have to worry about these settings if the Automatically update is enabled.

5.4.1.7.2 Unicode Character Range

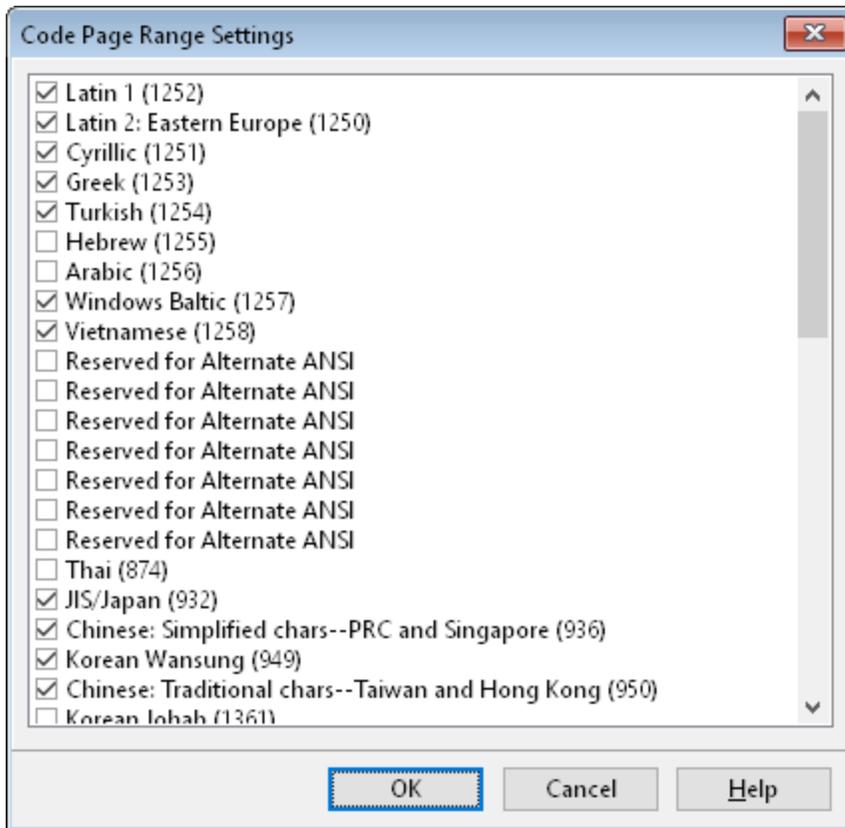
This field is used to specify the Unicode blocks or ranges encompassed by the font file in the **mappings** for the **Windows platform**. If a Unicode range is selected it is considered functional. The determination of "functional" is left up to the font designer, although character set selection should attempt to be functional by ranges if at all possible.



5.4.1.7.3 Code Page Character Range

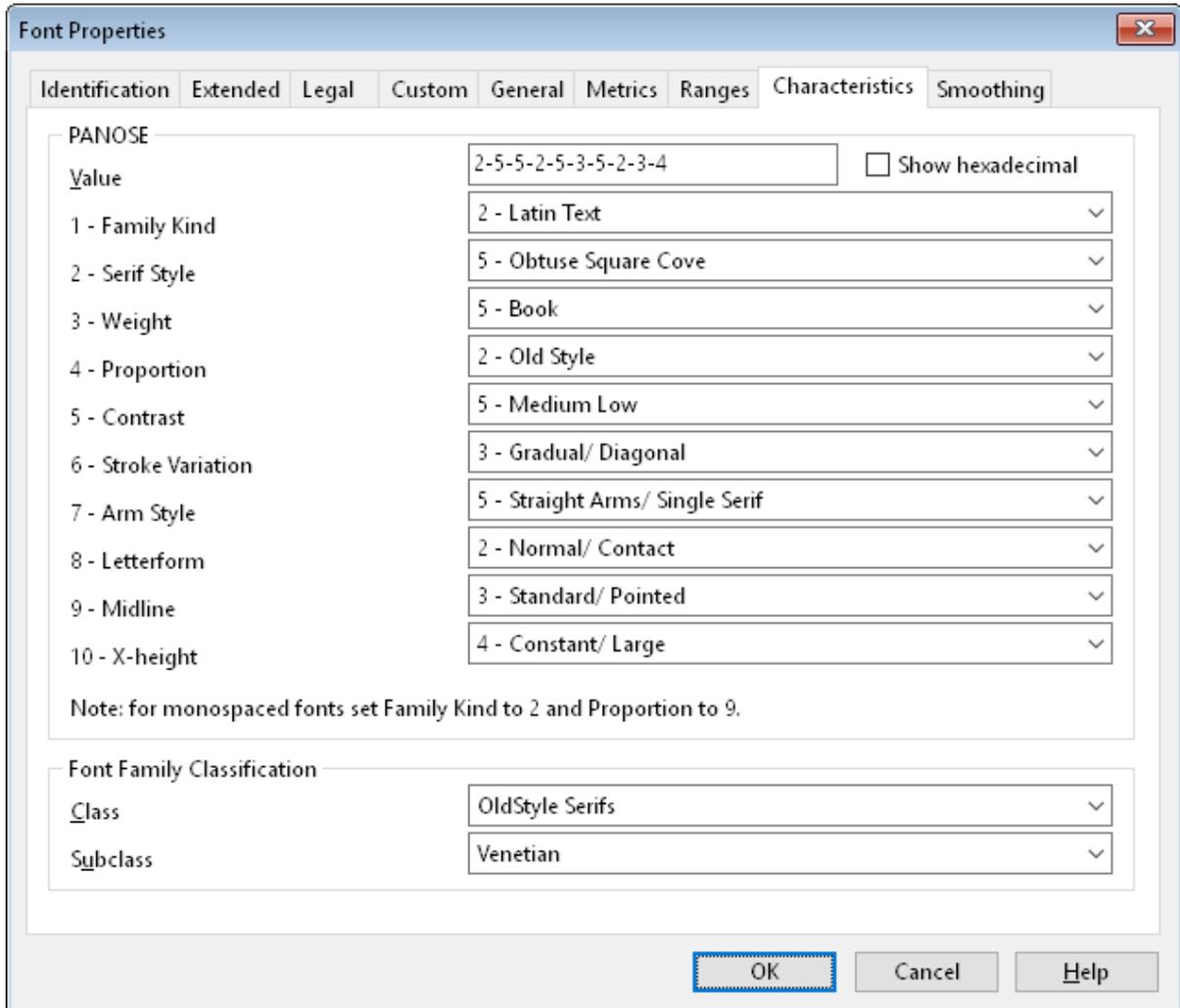
This field is used to specify the code pages encompassed by the font file in the **mappings** for the **Windows platform**. If the Windows platform is **Windows Symbol**, then the Symbol Character Set should be selected.

If a code page is selected then the code page is considered functional. The determination of "functional" is left up to the font designer, although character set selection should attempt to be functional by code pages if at all possible.



Note: Symbol character sets have a special meaning. If the Symbol Character Set is selected, and the font file contains a Windows Symbol platform, then all of the characters in the Unicode range 0xF000 - 0xFFFF (inclusive) will be used to enumerate the symbol character set. If this code page is not selected, any characters present in that range will not be enumerated as a symbol character set.

5.4.1.8 Characteristics



PANOSE

These fields are used to describe the visual characteristics of a given typeface. These characteristics are then used to associate the font with other fonts of similar appearance having different names. The PANOSE evaluation document details the specifications for assigning PANOSE numbers.

<http://www.panose.com/> 

Font Family Classification

These fields are a classification of the font-family design.

The font class and font subclass are registered values assigned by IBM to each font family. These fields are intended for use in selecting an alternate font when the requested font is not available. The font class is the most general and the font subclass is the most specific. More information about this field is available

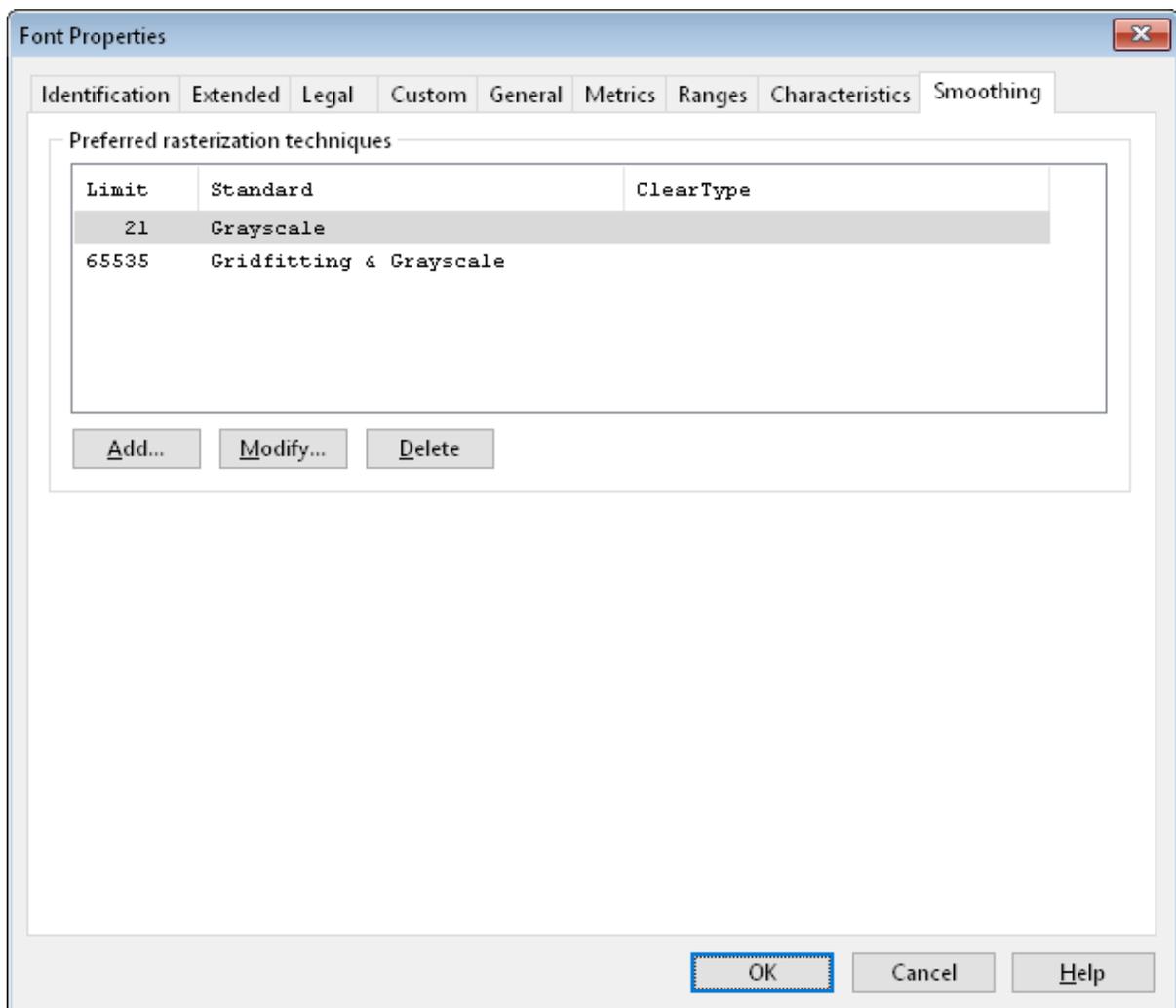
online:

<http://www.microsoft.com/typography/otspec/ibmfc.htm> 

5.4.1.9 Smoothing

5.4.1.9.1 Overview

The **Smoothing** page contains information which describes the preferred rasterization techniques for the typeface when it is rendered on grayscale-capable devices. It also has some use for monochrome devices, which may use the table to turn off hinting at very large or small sizes, to improve performance. On the **Font** menu, click **Properties**, and then click the **Smoothing** tab.



If there are no ranges defined in a typeface, the rasterizer may apply default rules to decide how to render the glyphs on grayscale devices. The rasterizer will use the ClearType related values, if ClearType is enabled.

Note: Grayscale rendering and smoothing was invented for screen output and will not be used by printer drivers.

At very small sizes, the best appearance on grayscale devices can usually be achieved by rendering the glyphs in grayscale without gridfitting. At intermediate sizes, gridfitting (also known as hinting) and monochrome rendering will usually produce the best appearance. At large sizes, the combination of gridfitting and grayscale rendering will typically produce the best appearance.

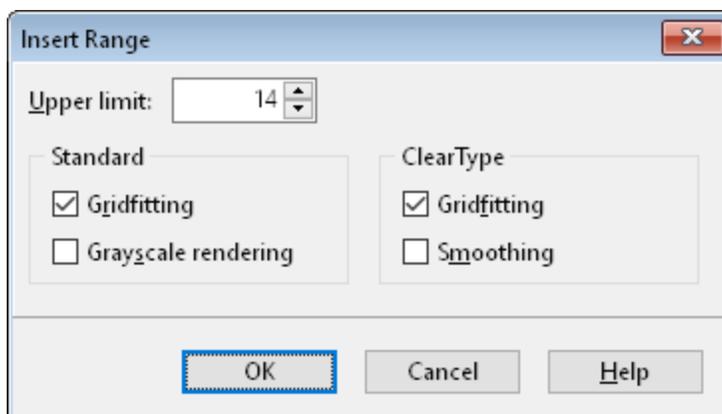
To add a new range, press the **Add** button. To remove a range, select it and press the **Delete** button. To remove all ranges, right-click the list view and select Delete All.

Note: OpenType fonts with CFF based outlines use a different technique, so the smoothing settings will only be included with OpenType fonts with TrueType based outlines.

Note: When Auto Hinting is selected on exporting the font, the setting will be replaced by one single entry to always use grayscale rendering, for all sizes, with gridfitting for standard hinting, and gridfitting and smoothing for ClearType.

5.4.1.9.2 Insert Range

In the **Insert Range** window set the **Upper limit** and optionally check the **Standard** and **ClearType** fields and press the **OK** button.



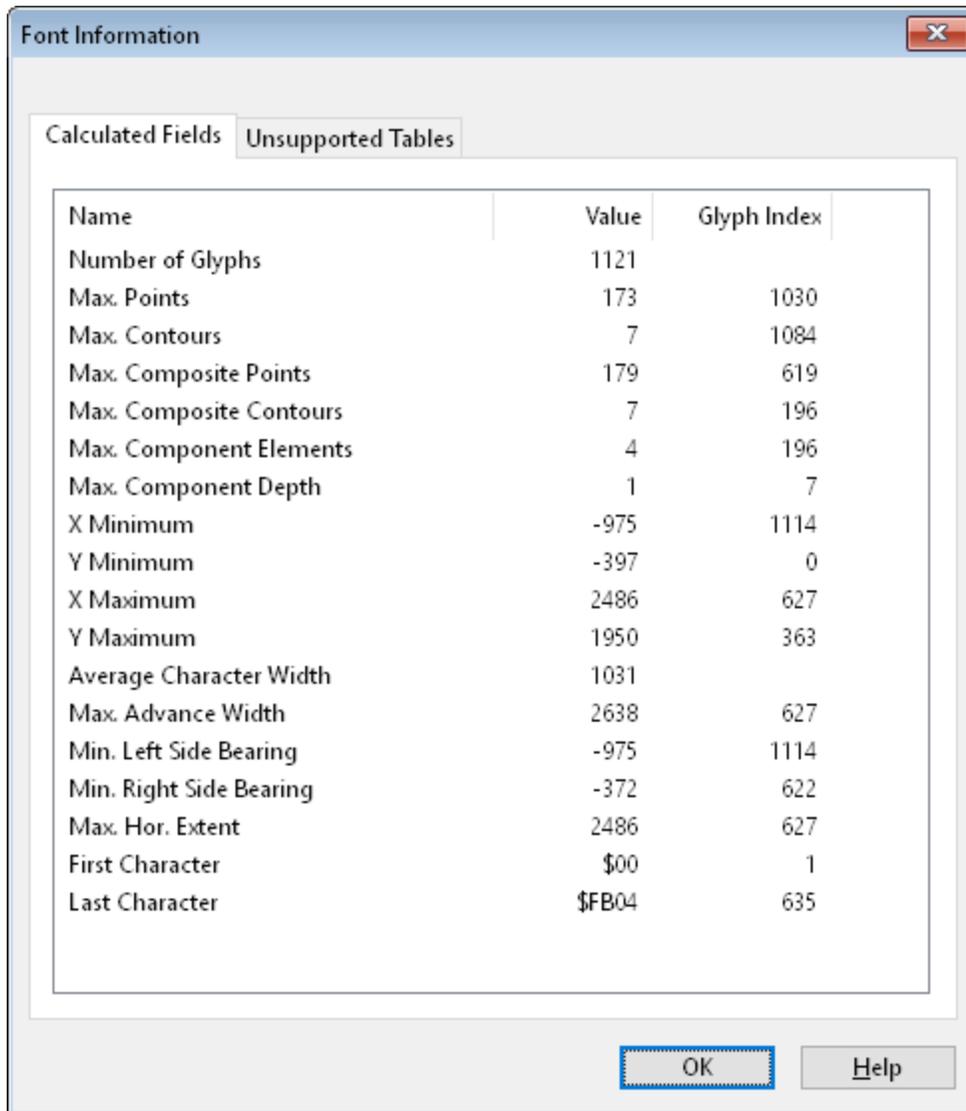
See also:

[Smoothing - Overview](#)

5.4.2 Font Information

5.4.2.1 Calculated Fields

Information about the font is found on the Font Information dialog (from the Font menu).



These fields can't be modified directly as they are calculated and depend on other font related data, but you can double-click a row to jump to the specific glyph.

Number of Glyphs

The number of glyphs in the font.

Max. Points

Maximum points in a non-composite glyph.

Max. Contours

Maximum contours in a non-composite glyph.

Max. Composite Points

Maximum points in a composite glyph.

Max. Composite Contours

Maximum contours in a composite glyph.

Max. Component Elements

Maximum number of components referenced at "top level" for any composite glyph.

Max. Component Depth

Maximum levels of recursion; 1 for simple components.

X Minimum for all glyph bounding boxes**Y Minimum for all glyph bounding boxes****X Maximum for all glyph bounding boxes****Y Maximum for all glyph bounding boxes**

The bounding box values computed using only glyphs that have contours.

Average Character Width

Average width of all characters

Max. Advance Width

Maximum advance width value

Min. Left Side Bearing

Minimum left side bearing value

Min. Right Side Bearing

Minimum right side bearing value

Max. Hor. Extent

Maximum extent value

First Character

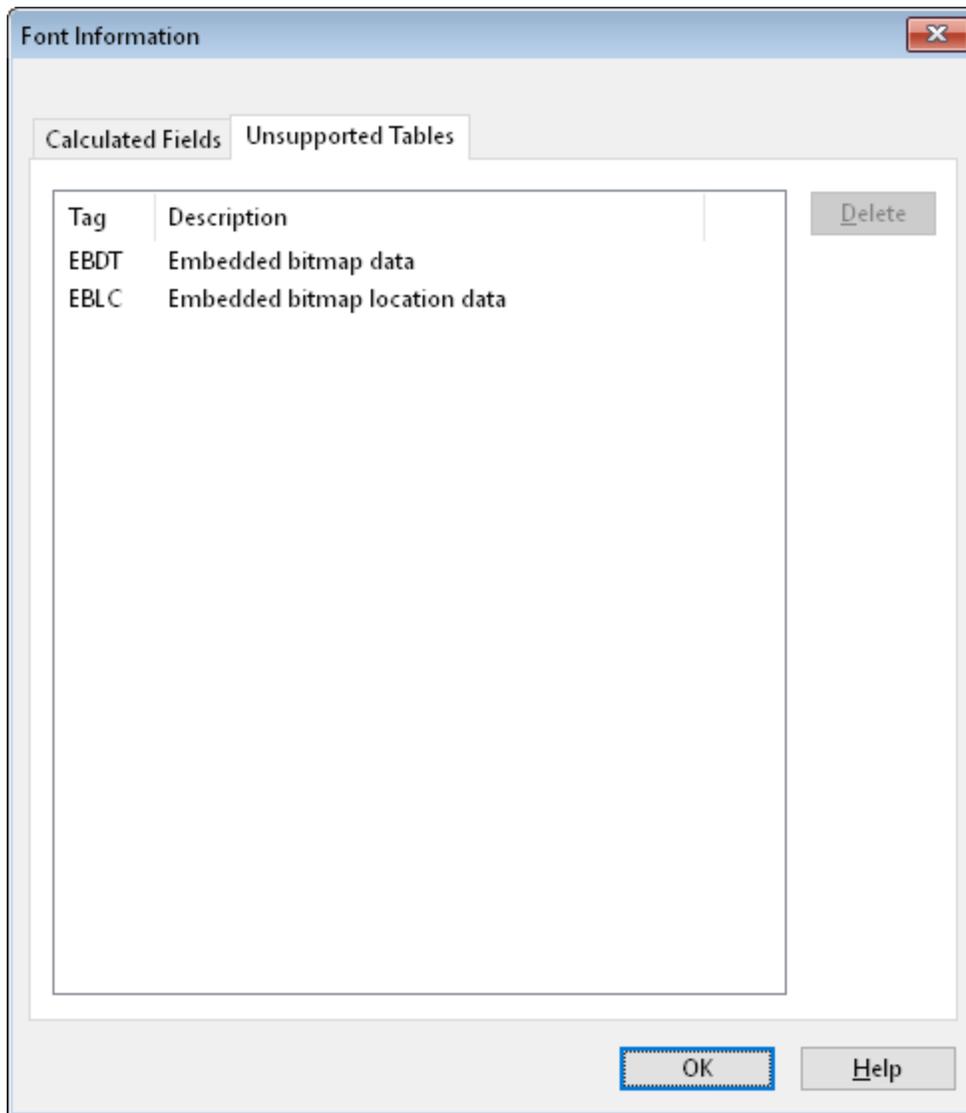
The minimum Unicode index (character code) in this font, according to the Windows Unicode BMP (UCS-2) or Windows Symbol mapping table. For most fonts supporting Win-ANSI or other character sets, this value would be 0x0020.

Last Character

The maximum Unicode index (character code) in this font, according to the Windows Unicode BMP (UCS-2) or Windows Symbol mapping table. This value depends on which character sets the font supports.

5.4.2.2 Unsupported Tables

Over the years the TrueType and OpenType font specifications have been updated with new tables and numerous tables have become obsolete. FontCreator supports all common font tables.



EBDT and EBLC can be considered legacy tables. font designers might decide to keep them for compatibility reasons.

If you want to delete an unsupported table, select the table on the **Unsupported Tables** page and press the **Delete** button.

5.4.3 OpenType Layout Features

5.4.3.1 Introduction

OpenType Layout Features can be used to extend the functionality of fonts and create advanced typographic features and add additional capabilities such as ligatures, alternate glyphs, two-dimensional glyph positioning, specific functionality for different scripts and languages and much more.

When a font is opened, FontCreator will process all supported features in the font and split them up into two separate parts. The first part, the glyph substitutions,

will be decompiled into a script that can be edited using the built-in script editor. The second part, the glyph positioning, will be processed to make them visually editable in the OpenType Designer.

If the font you're opening contains **Microsoft VOLT** project data for OpenType Layout Features, you will be asked if you want to convert this into a FontCreator project. If you choose not to use Microsoft VOLT project data, the binary OpenType Layout Features (GPOS, GSUB, and GDEF tables) will be used instead. You can also import Microsoft VOLT projects later through the [OpenType Designer](#).

Note: Microsoft VOLT project conversion and importing is only available in the **Professional** edition of FontCreator and some features are not [supported](#).

5.4.3.2 Supported types of substitution and positioning

The OpenType Layout Feature specification describes eight types of substitution lookups in the glyph substitution table (GSUB). The following table shows those currently supported by FontCreator:

#	Type	Script	Designer	Description
1	Single	Yes	Yes	Substitute a single glyph by another single glyph (a -> b)
2	Multiple	Yes	Yes	Substitute a single glyph by other multiple glyphs (a -> xyz)
3	Alternate	Yes	Yes	Substitute a single glyph by one of multiple alternates (a -> x or y or z)
4	Ligature	Yes	Yes	Substitute multiple glyphs by a single ligature (f f i -> ffi)
5	Context	Yes	Yes	Substitute one or more glyphs in context
6	Chaining Context	Yes	Yes	Substitute context specific glyphs (3rd -> 3 rd)
7	Extension Substitution	Auto*	Auto*	
8	Reverse chaining context	Yes	Yes	Applied in reverse order, replace single glyph in chaining context

The OpenType Layout Feature specification describes nine types of positioning lookups in the glyph positioning table (GPOS). The following table shows those currently supported by FontCreator:

#	Type	Script	Design	Description
			er	
1	Single adjustment	Yes	Yes	Change the position of a single glyph (sub/superscript)
2	Pair adjustment	Yes	Yes	Mostly used to define kerning pairs
3	Cursive attachment	Yes	Yes	
4	Mark to base attachment	Yes	Yes	Attach a combining mark such as a diacritic to a base glyph
5	Mark to ligature attachment	Yes	Yes	Attach a combining mark to a ligature
6	Mark to mark attachment	Yes	Yes	Attach a combining mark to another mark
7	Context Positioning	Yes	Yes	Position one or more glyphs in context
8	Chained Context Positioning	Yes	Yes	Position one or more glyphs in chained context
9	Extension Substitution	Auto*	Auto*	

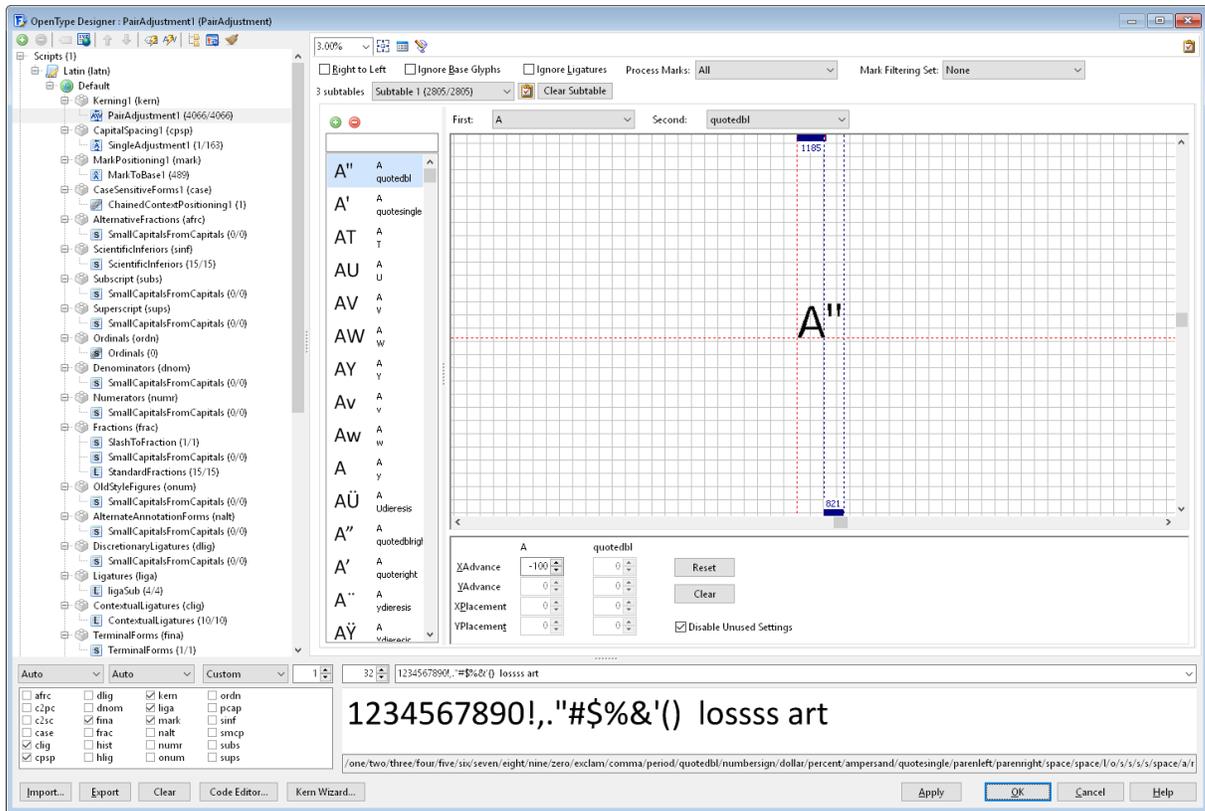
* Extension substitution is a special kind of lookup that is only used for fonts with a lot of features. FontCreator will automatically include such lookup if needed.

In addition FontCreator also supports all [feature parameters](#) currently defined.

So basically FontCreator supports all OpenType Layout Features. For an up-to-date list of things currently not fully supported, please visit our forums at <http://forum.high-logic.com/viewtopic.php?f=4&t=5098>

5.4.3.3 OpenType Designer

5.4.3.3.1 OpenType Designer



The OpenType Designer provides you with an easy to use visual way to edit glyph positionings. In the left pane you see how the features and lookups are organized in your font, and depending on the lookup type you can edit the properties in the right pane.

The splitter on the dialogue can be dragged to adjust the width of the panes. Double-click the thumb bar (|) to dock the script pane when it is not needed or to undock it.

The left toolbar allows you to add, remove, move, rename and change scripts, languages, features and lookups. Most of these functions are also available in the right-click menu when you have selected an item.



Add Add a new script, language, feature or lookup. The available options depend on which item is selected in the tree. If, for example, you select a feature, the current script and language will already be pre-determined on the add dialog.

Delete Deletes a script, language, feature and/or lookup. A popup window will ask you to confirm and select how you want to delete the

selected item.

- Rename** Rename the selected item. Please note that each script, language, feature or lookup name has to be unique.
- Change tag** Allows you to change the tag (type) of scripts, languages and features. Please note that each tag may only exist once on the same level of the tree.
- Move up** *) Move a lookup or feature up
- Move down** *) Move a lookup or feature down
- Autoname** This will provide all your scripts, languages, features and lookups with consistent and if possible meaningful names
- Autokern** This will automatically create kerning pairs for pair adjustment lookups. See also [Autokern](#).
- Autogroup** Only affects Single and Pair Adjustment lookups: Will group pairs and single adjustments together based on their left and right side bearing, width and current adjustment value. This will greatly reduce the number of visible kerning pairs and allow you to quickly change adjustment values of multiple glyphs at once.
- Break classes** Only affects Single and Pair Adjustment lookups: Will break all classes into separate kerning pairs or single adjustments.
- Cleanup** Permanently deletes all unused lookups and features, you can use this to quickly clean up your lookups and features.

*) The order in which lookups are defined is also the order in which they are processed by host applications. This button allows you to move a lookup down so they are processed earlier or later. As soon as the host application has found a match in one of the lookups it will stop processing that specific lookup. This can be useful to create an "override" for kerning pairs for example. Consider a large kern lookup which contains several class-based pairs. If one or two pairs need extra modification, you can add another lookup to the feature and add the "exceptions" before the actual kern table. This way the host application will find the "exception" first and will skip the "wrong" value from the large lookup.

The right-click menu on the left pane also provides some additional lookup specific actions:

- Export** Only available for Single and Pair Adjustment lookups: Allows you to export only a single lookup table (including all used classes) and import it into another font.
- Break** Break kerning classes into separate kerning pairs.

classes

Autogroup Perform auto grouping on the selected lookup table only.

Import

The import function allows you to import an OpenType Layout Feature Definition script. If the script contains only a lookup table, the lookup table will be added to the existing lookups. If the script contains script and language declarations, all existing items will be deleted and overwritten. The **Professional** Edition of FontCreator 9.1 also allows you to import **Microsoft VOLT** project (*.vtp) files.

Export

The export function allows the export of all scripts, languages, features and lookup definitions, including those that are not in use. Class definitions will also be exported. If you want to export only a single lookup table, you can right-click on one of the lookup tables, and select "Export Lookup". Please note that exporting a single lookup is only supported for Single and Pair adjustments.

Clear

Remove **all** scripts, languages, features, lookups, classes, and anchors.

Lookupflags

Right to Left From a technical point of view, this is only important for Cursive Attachment lookups. When checked the last glyph in a given sequence to which the cursive attachment lookup is applied, will be positioned on the baseline.

For all other lookups it is used within FontCreator to indicate that the lookup is used for right to left writing. For kerning pairs this means that the first and second glyph will be visually swapped.

Ignore Base Glyphs If checked, the processing application will skip over the base glyphs*

Ignore Ligatures If checked, the processing application will skip over ligatures*

Process Marks Defines which marks* should be processed

Mark Defines which marks* should be filtered

Filtering Set

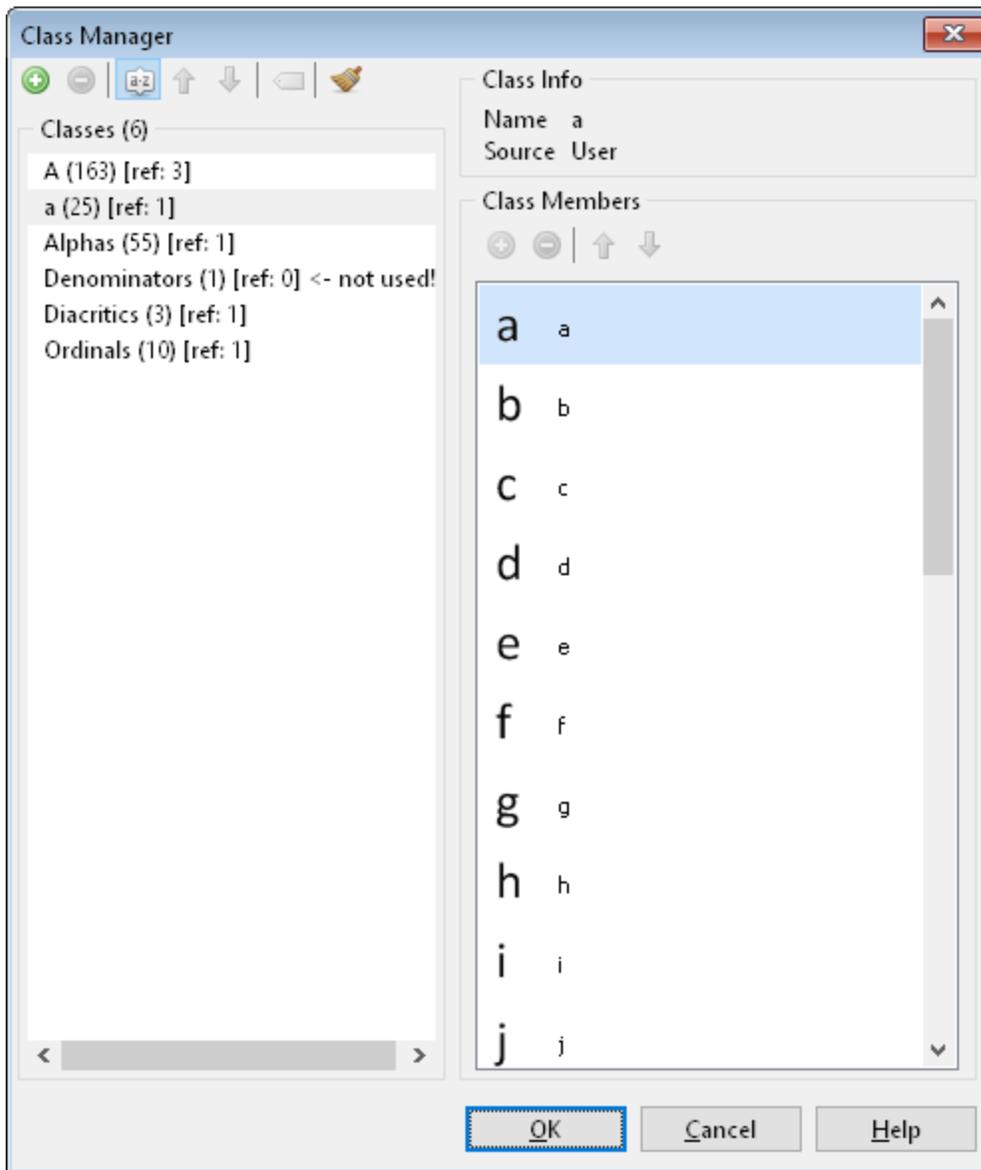
***) Base, Ligature, and Mark are OpenType Types which can be defined through the [Glyph Properties toolwindow](#).**

Clear subtable will delete all entries of the currently selected subtable, to clear an entire lookup table (including all subtables) use the right-click menu in the left pane.

Settings

The settings button will open the [Designer Settings](#)

5.4.3.3.2 Class Manager

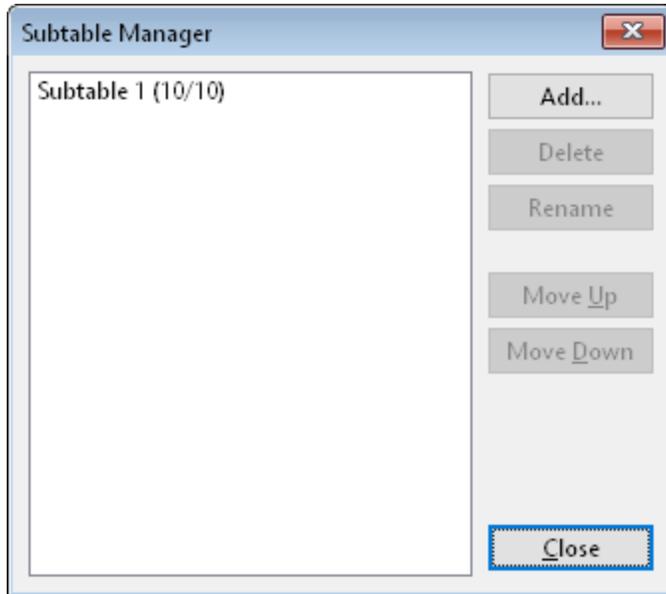


The class manager allows you to manage the classes used in the OpenType Designer. Here you can add, delete and rename classes and add or remove the glyphs in the class. Please note that when you delete a class, add or remove any of its members, the lookups associated with the class will also be changed. This means for example that when you add or remove a glyph to/from a class, the associated kerning pairs will also be added or removed automatically.

The cleanup button in the toolbar will delete all classes that are not used by any of the lookups.

Note: In general the order in which items appear in a class is not important for glyph positioning, however for glyph substitutions it is.

5.4.3.3.3 Subtable Manager

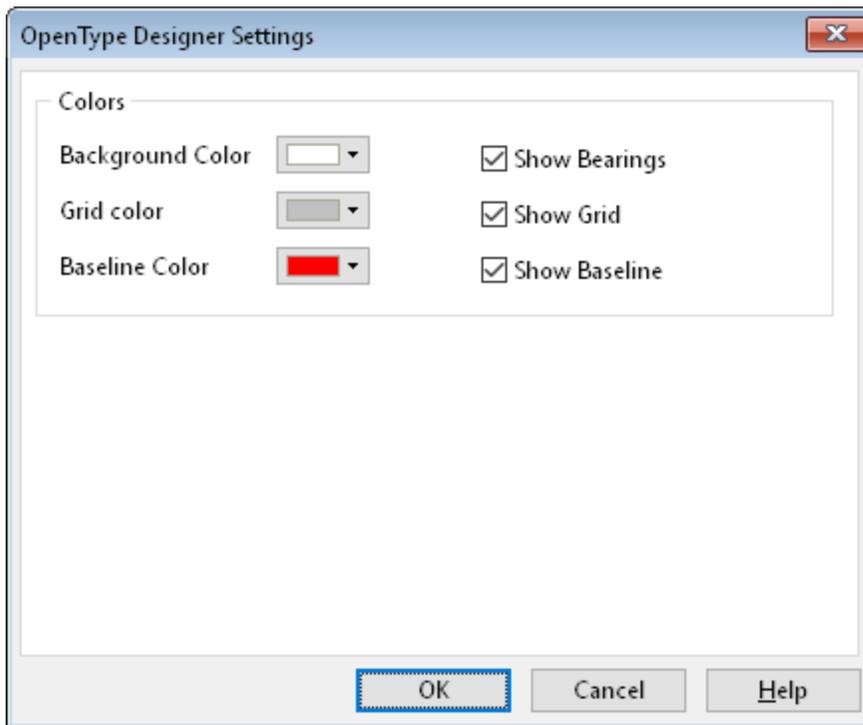


When a lookup contains a lot of entries, it is recommended (and sometimes even mandatory) to create multiple subtables to keep things organized. The Subtable Manager allows you to add, remove, rename and move subtables.

Just as with normal lookups, the order in which subtables appear controls how they are processed.

Note: There are technical limitations to how many items can be stored in each subtable. When a font is exported, FontCreator will break up subtables into several separate subtables if necessary. We do however recommend splitting larger lookups into several subtables yourself to keep your items organized.

5.4.3.3.4 Designer Settings



The OpenType Designer Settings allow you to modify the look and feel of the designer.

Background Color

Color of the background

Grid color

Color of the grid

Baseline Color

Color of the baseline

Show Bearings

Enables/Disables the drawing of the bearings

Show Grid

Enables/Disables the grid

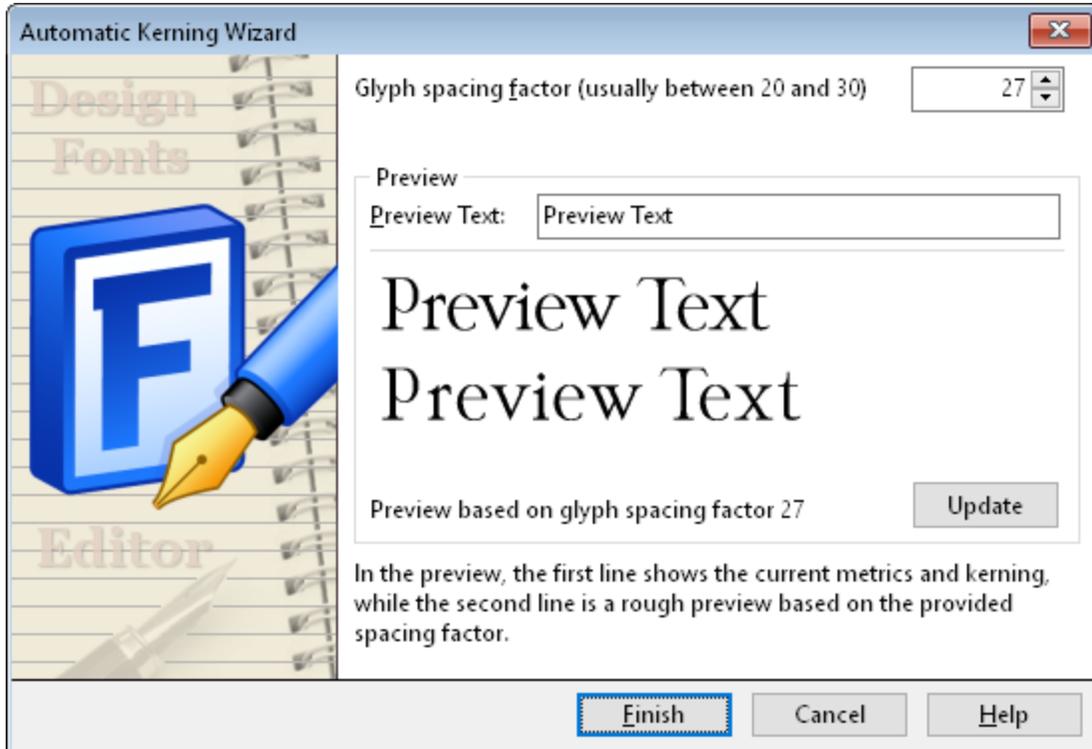
Show Baseline

Enables/Disables the drawing of the baseline

5.4.3.3.5 Autokern Settings

With **Autokerning**, you can generate new kern values for the existing kerning pairs. You can start this dialog by selecting a Pair Adjustment lookup in the OpenType Designer dialog, right-click it, and then select Autokern..

You can use the Preview area at the bottom of the OpenType Designer dialog to test the kerning pairs.



Glyph spacing factor allows you to define the distance between glyphs. The larger the factor the more space between glyphs, thus the larger the left and right side bearings.

Preview is where you can define a preview text sample which will be shown in the preview area.

The **Next** button takes you to the next where you can [set additional options](#).

Note: The **Automatic Kerning** wizard is not available in the Home Edition of FontCreator.

See also:

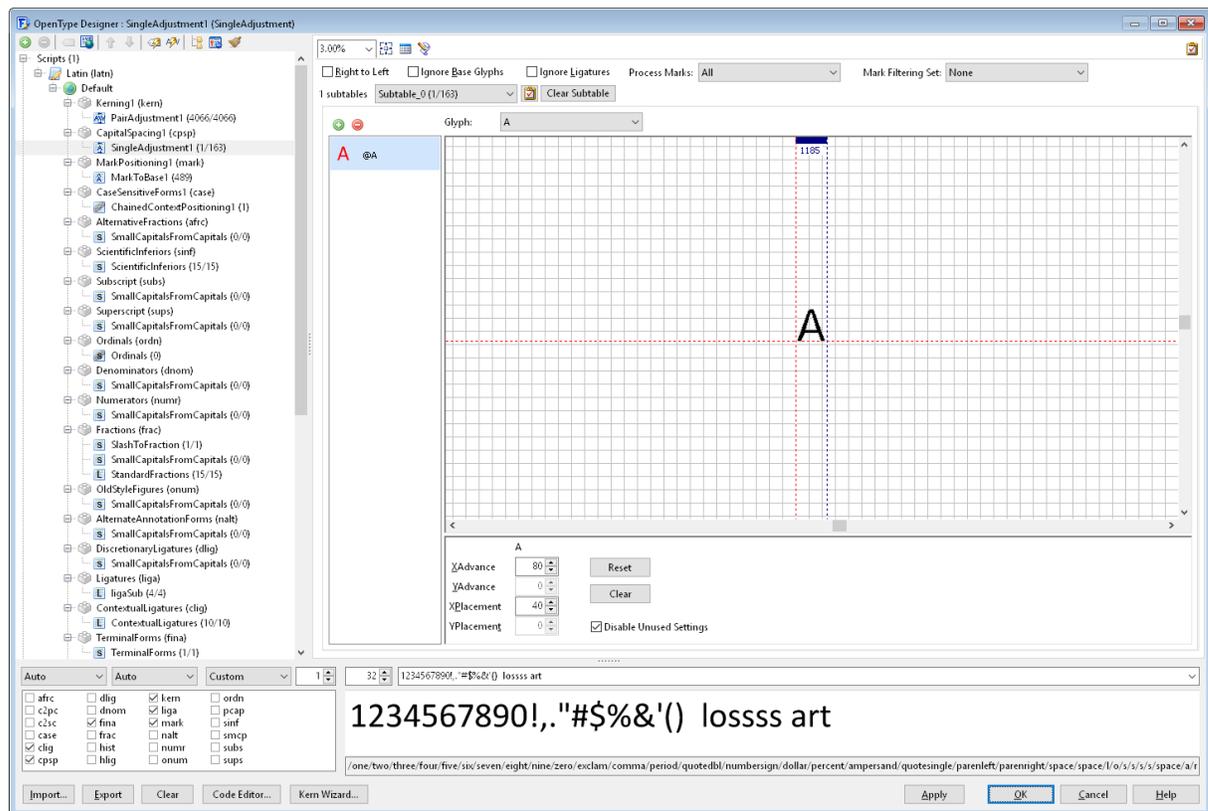
[Autokern new pair adjustment lookup](#)

5.4.3.3.6 Substitutions

Substitution lookups allow you to supply glyph substitutes. There are several types of glyph substitution. The most common one is the ligature substitution, but which ones you want to use depends on what you want to achieve.

Within the OpenType Designer you can add specific substitution lookups to your features.

5.4.3.3.7 Single Adjustment



Single Adjustment is commonly used to create a Capital Spacing, Superscript or Subscript feature.

On the right pane you can change the properties for each glyph separately, or when you create a class of glyphs, modify the properties of each glyph in the class simultaneously.

When you select a class, the droplist on the top of the right pane allows you to select a single member from the class, please note however that when you edit

a value, it will be for all glyphs in the class at once.

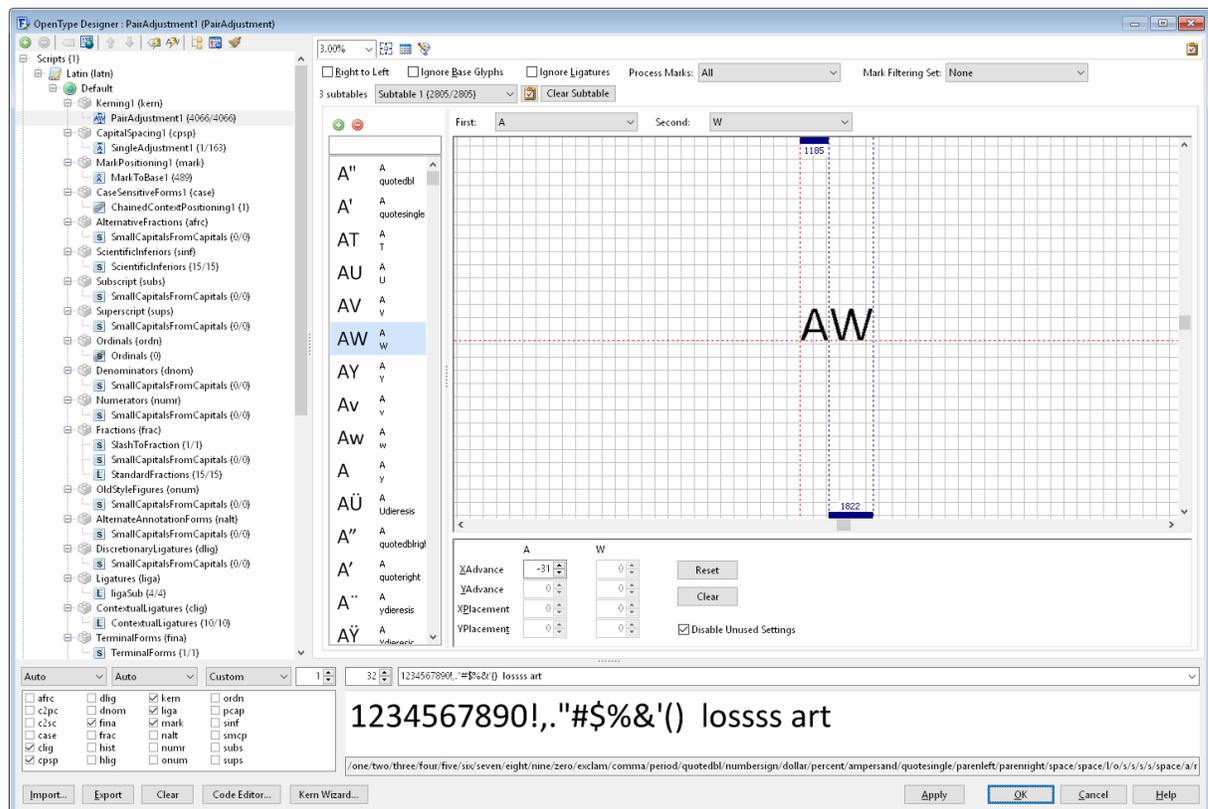
To modify the values, you can either type in the values manually, or use your mouse and keyboard to drag the glyph into the right position.

- The left and right mouse buttons allow you to change the XAdvance
- While holding down the shift button, you can change the XPlacement and YPlacement.

Note: For a description of the XAdvance, YAdvance, XPlacement and YPlacement fields, please see the [Pos Syntax](#)

Add...	Add a new Single Adjustment
Delete	Delete the selected Single Adjustment
Reset	Reset the values to their previous values
Clear	Set all fields to zero

5.4.3.3.8 Pair Adjustment



Pair adjustments are most commonly used to create kerning pairs.

Note: Even though you can set values for the second glyph, it is recommended to keep those values all to zero. If you do set any of those values to non-zero,

the text layout engine will skip the next possible pair for adjustment, which is usually not what a font designer intends, or expect to happen.

On the right pane you can change the properties for each glyph separately, or when you create a class of glyphs, modify the properties of each glyph in the class simultaneously.

When you select a pair containing a class, the droplist on the top of the right pane allows you to select a single glyph from the class, please note however that when you edit a value, it will be for all glyphs in the class at once.

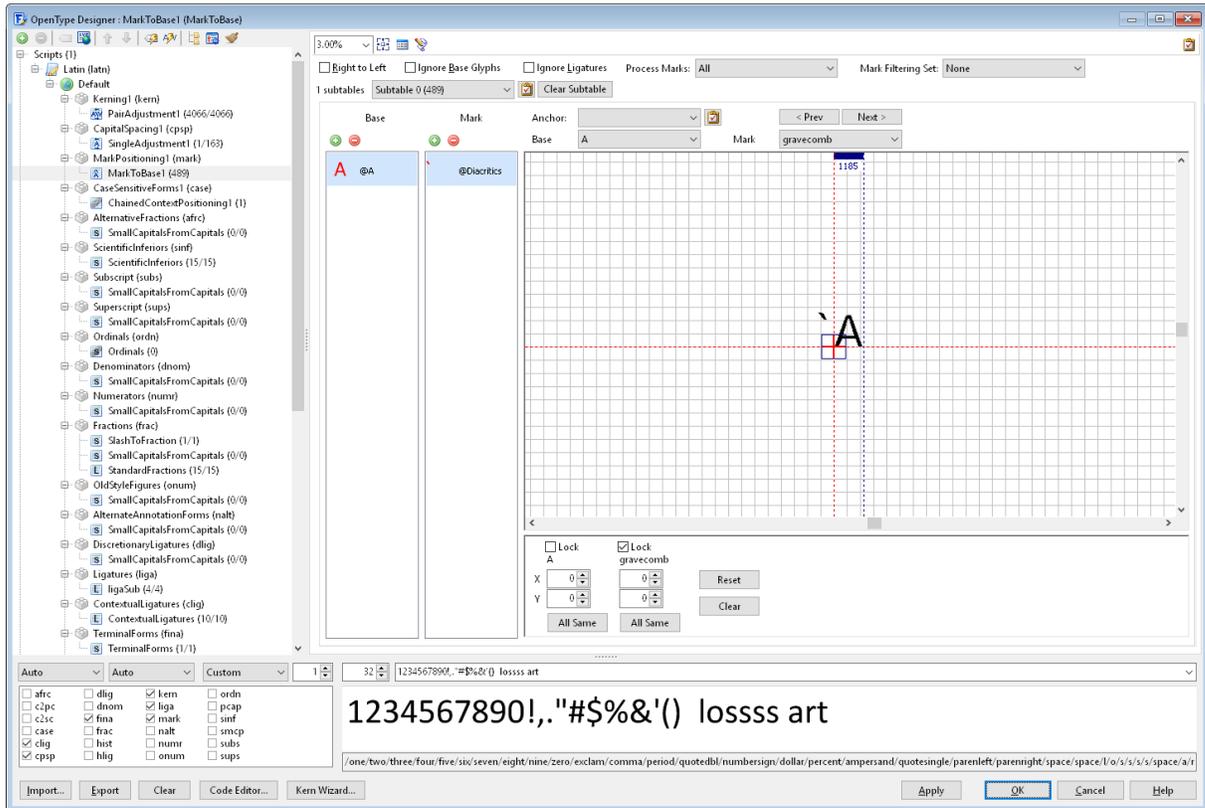
To modify the values, you can either type in the values manually, or use your mouse and keyboard to drag the glyph into the right position.

- The left and right mouse buttons allow you to change the XAdvance of the glyph you click on
- While holding down the shift button, you can change the XPlacement and YPlacement of the glyph you click on.

Note: For a description of the XAdvance, YAdvance, XPlacement and YPlacement fields, please see the [Pos Syntax](#)

Add...	Add a new Pair Adjustment
Delete	Delete the selected Pair Adjustment
Reset	Reset the values to their previous values
Clear	Set all fields to zero

5.4.3.3.9 Marks



Mark to Base, Mark to Ligature, and Mark to Mark are commonly used to add diacritic marks to base and ligature glyphs. Mark to Base and Mark to Mark are identical in usage, the only difference is that for mark to mark only mark glyphs may be used.

With Mark to Base, Mark to Ligature, Mark to Mark, glyphs are connected to each other via so called anchors. Anchors control the behaviour how the mark glyph interacts with the base glyph.

Each base glyph has an anchor point that is defined with the left X and Y value

and is indicated by the anchor icon:  Modifying these values will change the location of where each of the defined mark glyph will be drawn.

Each mark glyph also has an anchor point that is defined with the right X and Y value. Modifying these values will change the location of where the mark glyph will be drawn in relation to the base glyph anchor.

In short:

Modify left X,Y: Change position of all marks in relation to the base glyph

Modify right X,Y: Change position of mark in relation to all base glyphs

When used correctly, anchors are a very powerful tool to quickly change the position of several base/mark glyph combinations without having to modify all of your pairs. For example: if you create an anchor for uppercase glyphs and an anchor for lowercase glyphs, you could change the location of the mark of all lowercase glyphs at once, without modifying the uppercase glyphs.

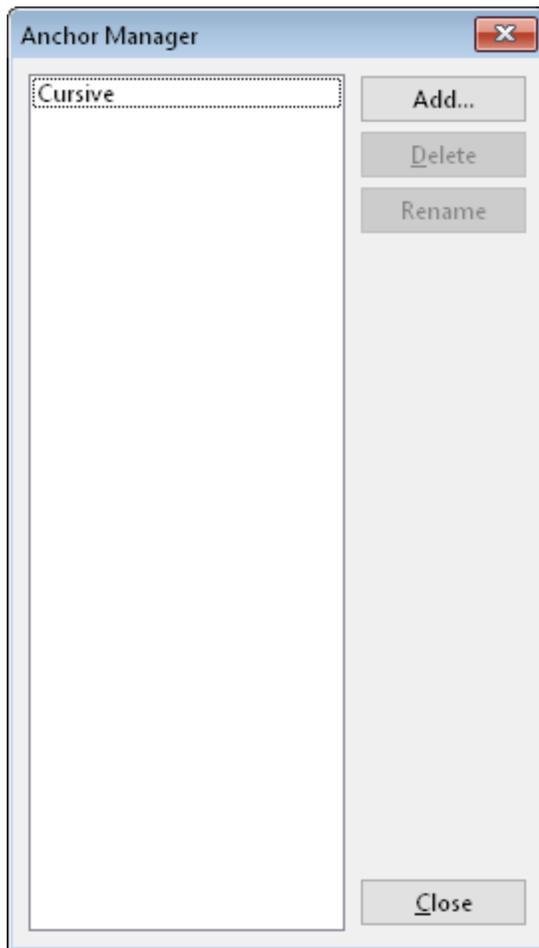
< Prev	Moves to the next Base/Mark pair
Next >	Moves to the previous Base/Mark pair
Reset	Reset the values to their previous values
Clear	Set all fields to zero
All Same	This will set the same anchor value for all glyphs in the currently selected glyph class

Combining marks and signs that appear in text not in conjunction with a valid consonant base are considered invalid. In Windows, Uniscribe displays these marks using the fallback rendering mechanism, on a dotted circle. For the fallback mechanism to work properly, a font should contain a glyph for the dotted circle (U+25CC). In case this glyph is missing from the font, the invalid signs will be displayed on the missing glyph shape (.notdef).

5.4.3.3.10 Cursive Attachment

A Cursive Attachment lookup is mostly used to describe cursive scripts and other glyphs that are connected with the special **Cursive** anchor class, which allows glyphs to connect through entry and exit anchors.

5.4.3.3.11 Anchor Manager

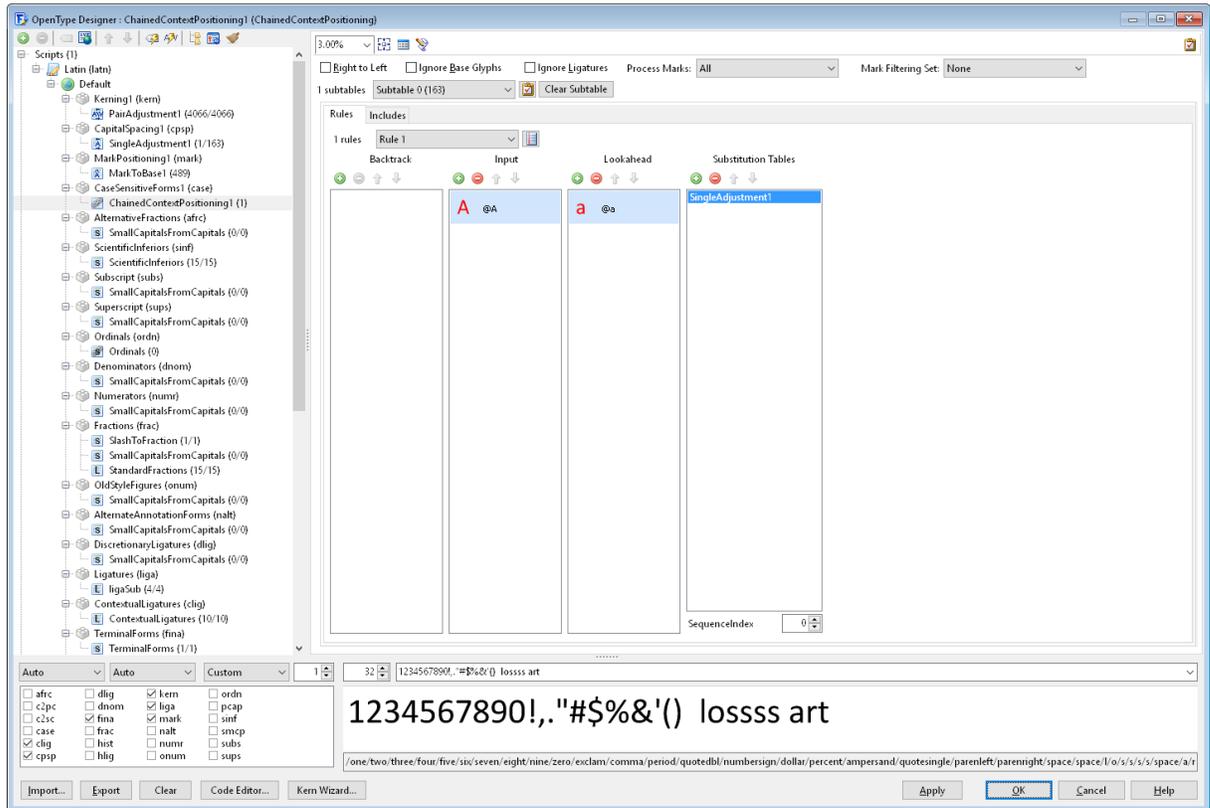


You can use the Anchor manager to add, remove and rename anchor classes.

Cursive is a special anchor class which is solely used with Cursive Attachment lookups.

You can add anchors to glyphs through the Anchors toolbar, Glyph Edit window, and the OpenType Designer dialog.

5.4.3.3.12 Chained Context



Chained Context positioning is most commonly used to change the position of certain glyphs in a specified sequence of glyphs.

5.4.3.3.13 Feature Parameters

There are several features that support additional parameters:

- Character Variants (cv01 - cv99)

- Optical size (size)
- Stylistic Sets (ss01 - ss20)

FontCreator supports all these feature parameters. To get access to those parameters, select or add the specific feature within the OpenType Designer dialog.

Character Variant

Feature Parameters - Character Variant - Descriptive Names and Codepoints

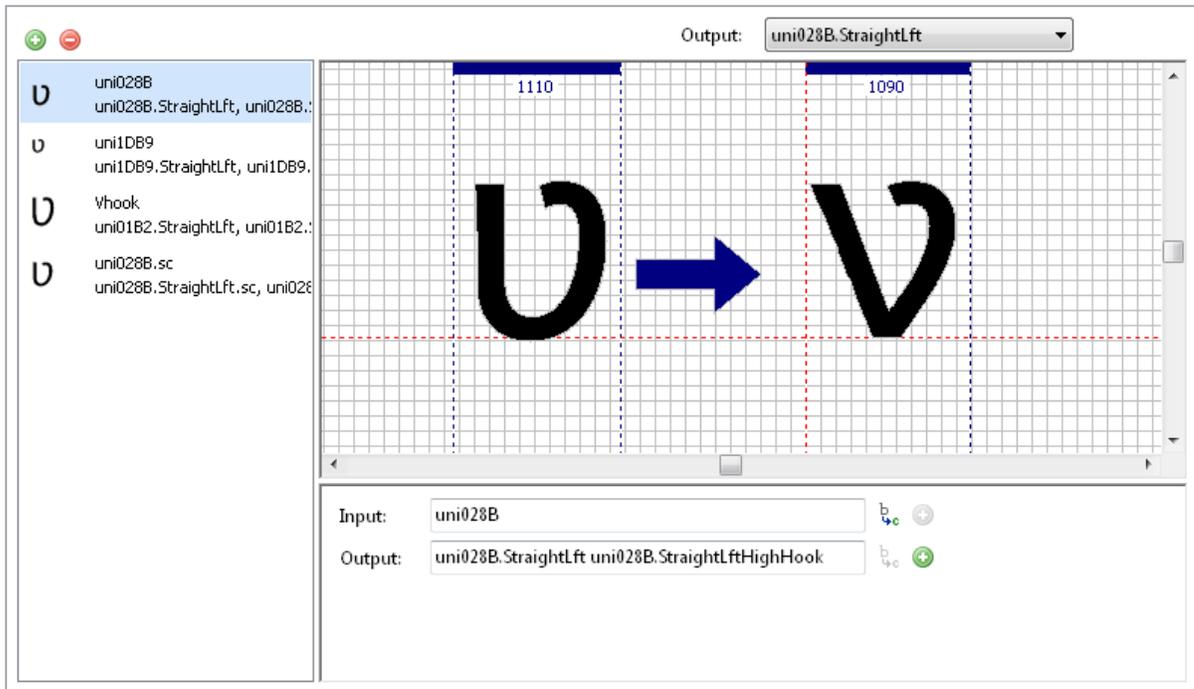
Language ID	Label	Tooltip	Sample Text
English - United States	V-hook alternates	V-hook alts	Uu*

Variant	Language ID	Content
Variant 1	English - United States	Straight with low hook
Variant 2	English - United States	Straight with high hook

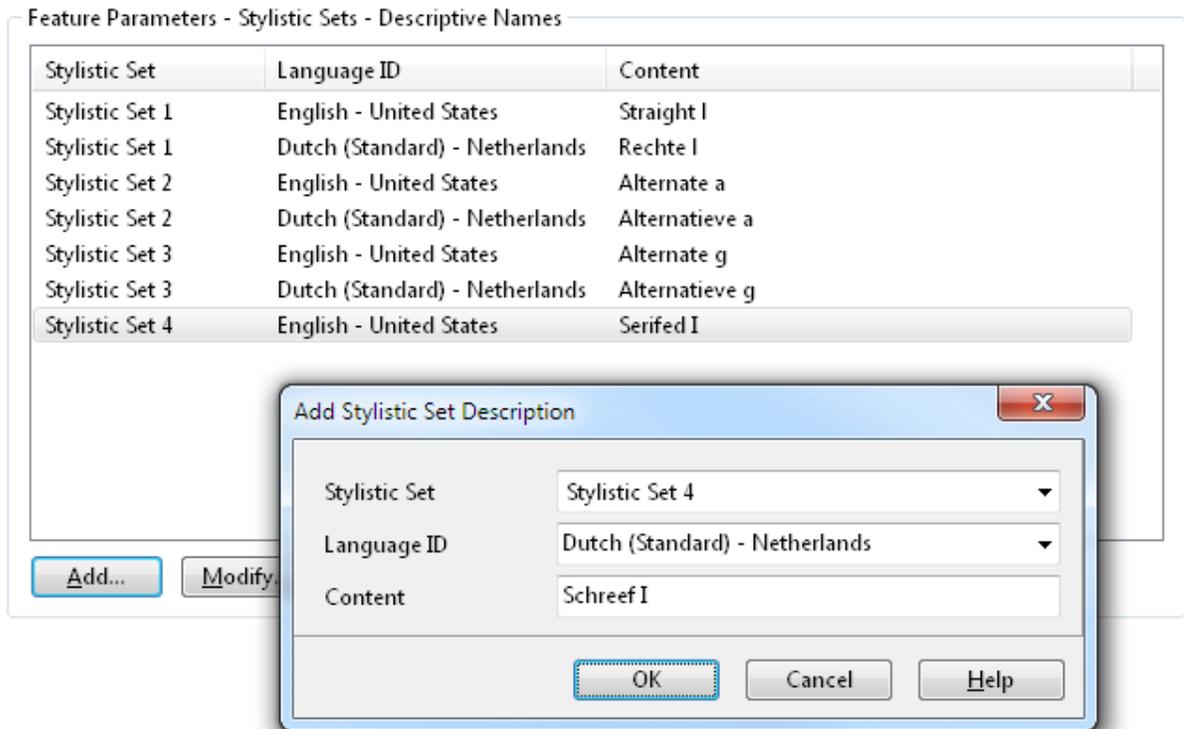
Character codepoints for which this feature provides glyph variants:

\$028B, \$1DB9

The above Character Variant feature contains the following Alternates lookup:



Stylistic Sets



As shown above, all Stylistic Sets feature parameters are grouped together.

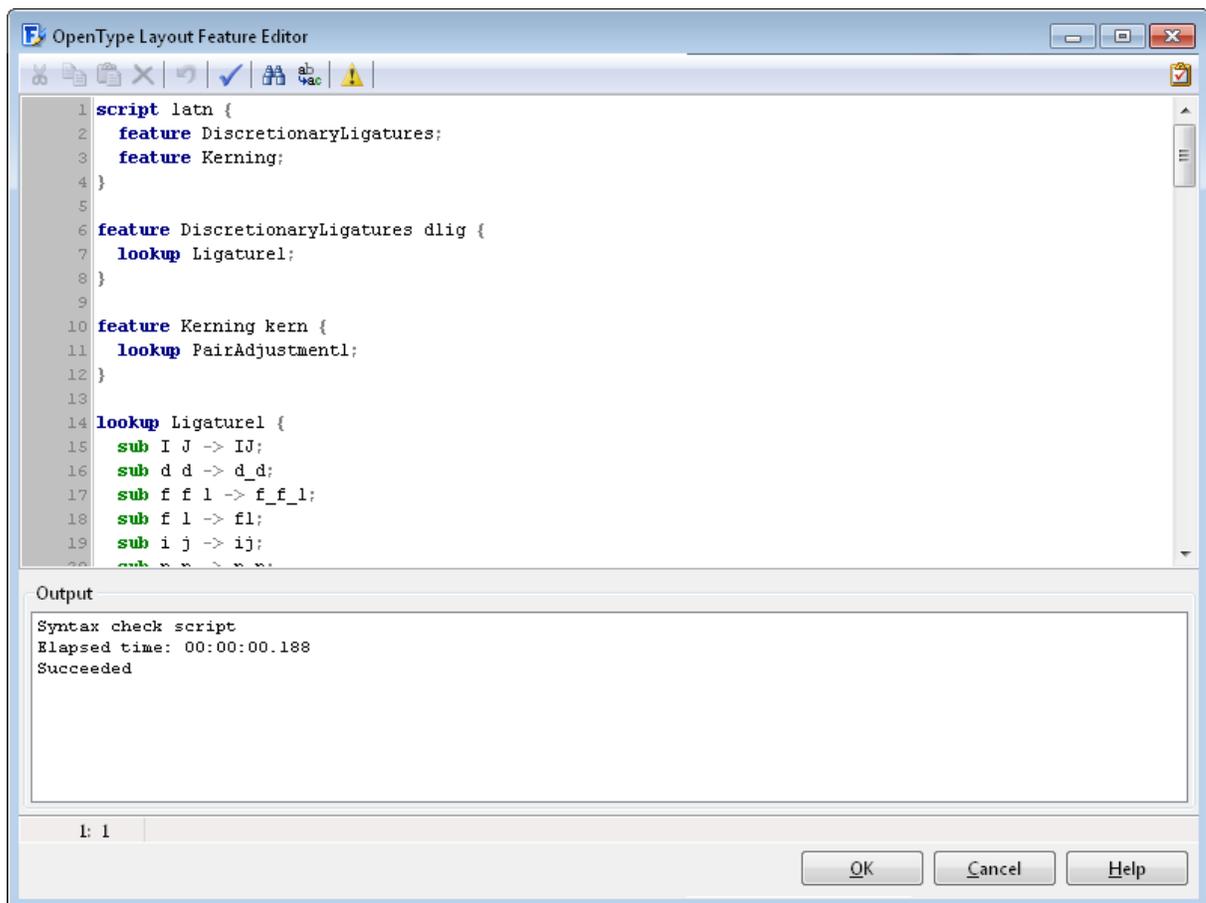
Note: At the time of writing this manual, only a few applications make use of one or more of these parameters, but since more and more professional fonts do contain them, it is just a matter of time before we see advanced word processing software and desktop publishing applications to support these feature parameters.

5.4.3.4 Script Editor

5.4.3.4.1 OpenType Layout Feature Code Editor

The **OpenType Layout Feature Code Editor** allows you to take full control over all the supported OpenType Layout Features in your font. Because adding features can be very time-consuming the editor uses a scripting language that is very easy to use and understand.

The OpenType editor supports a script-based syntax to define OpenType Layout Features for both Glyph Substitutions (GSUB) and Glyph Positioning (GPOS).



To test if the syntax is correct and all **Glyph** names are valid, select **Compile** in the **Actions** menu, use the shortcut key F9, or click the button on the toolbar. If there are any syntax errors or glyph names that could not be resolved, the **Output** window will list them. You can double-click on the error to quickly jump

to the line where the error occurred.

If you want the compiler to ignore unknown glyphs, and allow empty classes and lookups, then click the warning icon in the toolbar.

When you click the **OK** button, the code will be compiled and on success will replace all your existing scripts, features, lookups, classes, and anchors.

5.4.3.4.2 Script Syntax

5.4.3.4.2.1 Basics

A script consist of several "blocks" containing declarations for scripts, features, lookups and classes. A block is started by the keyword and optional name and/or tag following a left curly bracket "{". A block is closed by a right curly bracket "}".

It is not possible to nest these blocks, with the exception of the language block which has to be nested into a script block. Names of blocks must be unique across the script and they are case-sensitive. It is possible to reference a block before it is declared. A compiler error will be generated when a feature is referenced that does not exist, and when a feature is declared but not referenced.

Basic layout of a script:

```
script <tag> {
  <feature references>
}

feature <name> <tag> {
  <lookup references>
}

lookup <name> {
  [featureflags <flags>]
  <substitution declarations>
}
```

5.4.3.4.2.2 Comments

To add comments to your script, simply add a # (number sign) in front of it:

```
# This is comment
```

In general comments are not useful, as they will be lost as soon as the code is compiled into binary OpenType layout features.

5.4.3.4.2.3 Feature

The feature keyword is used in two ways: To reference a feature and to declare a feature. A feature is referenced from a script block to indicate the specified feature is available in that script.

Referencing a feature:

```
script <tag> {  
    feature MyLigatures; # referencing feature "MyLigatures"  
}
```

Declaring a feature:

```
feature <name> <tag> {  
    <feature declaration>  
}
```

Where:

- **name** is a name you can create yourself. You may only use a-z, A-Z, 0-9 and "_" in your name. Names are case-sensitive, and may only be declared once.
- **tag** is one of the feature tags defined on the Microsoft list of feature tags. A full list of available tags can be found at <http://www.microsoft.com/typography/otspec/featurelist.htm> 

5.4.3.4.2.4 Class

The class keyword is used to declare a class of glyphs to be used for Chained Context Substitution and several GPOS lookup types.

```
class @<class name> [ <glyphs> ];
```

Where:

- **name** is a name you can create yourself. You may only use a-z, A-Z, 0-9 and "_" in your name and it must start with an @ (at) sign. Names are case-sensitive, and may only be declared once.

5.4.3.4.2.5 Language

The language keyword is used to include a feature into a specific language. All features not assigned to a specific language will be added to the default language. The language keyword can only be used inside a script block. Assigning a feature to a specific language:

```
script <scripttag> {  
    language <tag> {  
        <features> # only available in this language  
    }  
    <features> # available in the default language  
}
```

Where:

- **tag** is one of the language tags defined on the Microsoft list of language tags (deprecated languages are not supported). A full list of supported languages can be found at <http://www.microsoft.com/typography/otspec/languagetags.htm> 

Don't forget to include features to the default language, as the default language will be used as fallback in case the language of text in a document isn't available in the script. Windows always uses the default language, although that might change in the future.

Note: There are fonts that have unknown language tags like "TUR " and "IPA ", these are either deprecated or common mistakes. The following table lists several of these tags and their proper replacements

Wrong tag	Correct tag	Description
DHV	DIV	Dhivehi
IPA	IPPH	Phonetic transcription
TUR	TRK	Turkish
CHN	ZHS or ZHH	Chinese Simplified or Chinese, Hong Kong SAR

5.4.3.4.2.6 Lookup

The lookup keyword is used in two ways: To reference a lookup and to declare a lookup. A lookup is referenced from a feature block to indicate the specified lookup should be used for that feature. Lookups can also be referenced from Chained context lookups

Referencing a lookup from a feature:

```
feature MyLigatures liga {
  lookup <name>;
}
```

Referencing a lookup from a chained context lookup:

```
lookup MyChainedContextLookup {
  context (@BackTrackClasses) @InputClasses (@LookAheadClasses);
  sub 0 <lookup name>;
}
```

Declaring a lookup:

```
lookup <name> <tag> {
```

```
<lookup declarations>
}
```

Note: It is possible to set optional lookup flags (properties) via the **lookupflags** keyword.

5.4.3.4.2.7 Lookupflags

The **lookupflags** keyword is used to modify several flags (properties) of a **lookup**. The current supported flags are:

RightToLeft	This bit relates only to the correct processing of the cursive attachment lookup type (GPOS lookup type 3). When this bit is set, the last glyph in a given sequence to which the cursive attachment lookup is applied, will be positioned on the baseline.
IgnoreBaseGlyphs	Skips over base glyphs
IgnoreLigatures	Skips over ligatures
IgnoreMarks	Skips over combining marks
UseMarkFilteringSet	Indicates that the lookup table structure is followed by a MarkFilteringSet field. The layout engine skips over all mark glyphs not in the mark filtering set indicated.

Applying lookupflags to a lookup:

```
lookup MyLookupTable {
  lookupflags <flags>;
}
```

When you want to apply more than one flag to a lookup, simply separate them by spaces.

Note: Lookup flag names are case-sensitive.

5.4.3.4.2.8 Subtable

Sometimes lookup tables are very large and it's better to break them up into several smaller tables: for this the keyword subtable can be used. Subtables are most commonly used in kerning tables that contain a lot of kerning pairs. We recommend a subtable for every 16,000 kerning pairs.

Declaring a subtable is only possible from within a normal lookup table:

```
lookup MyKerningLookup {
  subtable [name] {
    <lookup declarations>
  }
}
```

When declaring multiple subtables, the first subtable is not required to obey the

subtable syntax, but any consecutive subtable is. This means that:

```
lookup MyKerningLookup {
  <lookup declarations>

  subtable [name] {
    <lookup declarations>
  }
}
```

is valid, but

```
lookup MyKerningLookup {
  subtable [name] {
    <lookup declarations>
  }
  <lookup declarations>
}
```

is not.

The subtable name is optional, but we recommend defining one for clarity.

Note: The first declared subtable will define the lookup type and all consecutive subtables will have to be of the same type.

5.4.3.4.2.9 Script

The script keyword is used to declare a block of features for a specific script. When the features are to be used for all scripts, the script "dflt" (default) should be used.

```
script <tag> {
  feature <featurename>;
}
```

- **tag** is one of the script tags defined on the Microsoft list of script tags. The proposed script tags on that list are also supported. A full list of script tags can be found at: <http://www.microsoft.com/typography/otspec/scripttags.htm> 

5.4.3.4.2.10 Sub

The **sub** keyword declares a substitute. As explained in the supported substitution types section there are several substitution types. Substitutions can only be declared in lookups. It is not possible to declare a substitution directly in a feature. Each lookup can only have one type of substitution; this means that if you want to use several substitution types in a single feature, multiple lookups have to be declared.

Declaring Single (Type 1) substitutions

```
lookup MyLookupTable {
  sub A -> a.smcp;
}
```

Declaring Multiple (Type 2) substitutions

```
lookup MyLookupTable {
  sub ffi -> f f i;
}
```

Declaring Alternate (Type 3) substitutions

```
lookup MyLookupTable {
  sub asterisk -> [asterisk asteriskmath uni2051 uni2042 uni203B uni273B];
}
```

Declaring Ligature (Type 4) substitutions

```
lookup MyLookupTable {
  sub f f i -> ffi;
}
```

Declaring Chained Context (Type 6) substitutions

```
lookup MyLookupTable {
  context (@<backtrackclasses>) @<inputclasses (@<lookaheadclasses>);
  sub 0 <substitution table>;
}
```

IMPORTANT: The order in which substitute declarations appear is also the way they are processed by applications supporting OpenType Layout Features. This means that in the case of ligatures:

```
lookup MyLookupTable {
  sub f i -> fi;
  sub f f i -> ffi;
}
```

is not the same as:

```
lookup MyLookupTable {
  sub f f i -> ffi;
  sub f i -> fi;
}
```

and the latter declaration will have the proper result. Why? When the sequence "f i" is encountered it will be replaced by the fi character and will no longer match the f f i sequence. In the latter example "f f i" is matched before "f i" and the result is as expected.

5.4.3.4.2.11 Pos

The **pos** keyword declares a glyph positioning. As explained in the supported glyph positioning types there are several positioning types. Positioning

declarations can only be declared in lookups. It is not possible to declare a positioning directly in a feature. Each lookup can only have one type of positioning; this means that if you want to use several positioning types in a single feature, multiple lookups have to be declared.

For most positioning declarations several coordinates can be defined that determine the glyph's positioning changes. Each declaration consists of one to four coordinates.

The format of a coordinate is (including brackets):

<XAdvance YAdvance XPlacement YPlacement>

The XAdvance must always be specified, the YAdvance, XPlacement and YPlacement are optional.

XAdvance	horizontal adjustment for advance
YAdvance	vertical adjustment for advance
XPlacement	horizontal adjustment for placement
YPlacement	vertical adjustment for placement

Declaring Single adjustments (Type 1) positioning

Single positioning is used to alter the position of a single glyph or glyph class and can be used to easily create subscript and superscript like features using the same glyphs.

```
lookup MyLookupTable {
  pos A <-20 [0 0 0]>;
}
```

For classes the syntax is very similar:

```
class @class1 [A B C D E F G H I J K L M N O P Q R S T U V W X Y Z]

lookup MyLookupTable {
  pos @class1 <-20>
}
```

Declaring Pair adjustments (Type 2) positioning

Pair positioning is used to alter the position of 2 glyphs or glyph classes and is mostly used to define kerning pairs.

```
lookup MyLookupTable {
  pos A B <-10 [0 0 0]> [<0 [0]>];
}
```

For classes the syntax is very similar:

```
class @class1 [A B C D E F G H I J K L M N O P Q R S T U V W X Y Z]
class @class2 [a b c d e f g h i j k l m n o p q r s t u v w x y z]

lookup MyLookupTable {
  pos @class1 @class2 <-20>
}
```

This will create 676 kerning pairs with a value of -20 with only one line of code.

It is also possible to combine a class with a single glyph:

```
class @class1 [A B C D E F G H I J K L M N O P Q R S T U V W X Y Z]

lookup MyLookupTable {
  pos @class1 hyphen <-10>
}
```

This will create 26 kerning pairs with a value of -10 with only one line of code

Declaring Mark to Base (Type 4) positioning

Mark to base positioning is used to attach mark glyphs (diacritics) to base glyphs using anchors.

```
lookup MyLookupTable {
  mark acute 0 0;
  mark ring 0 0;
  pos A mark -10 30;
  pos B mark -10 20;

  mark esp 0 0;
  pos A mark -10 20;
}
```

Declaring Mark to Mark (Type 6) positioning

Mark to mark positioning is used to attach mark glyphs to other mark glyphs.

The syntax is exactly the same for Mark to Mark positioning, but only mark glyphs are used.

Declaring Chained Context (Type 8) positioning

This syntax is identical to the [GSUB syntax](#). The only difference is that the substitution tables are GPOS lookups instead of GSUB lookups.

5.4.3.4.2.12 Examples and Help

For example scripts and help with creating custom scripts, please visit our forums at:

<http://forum.high-logic.com/> 

5.4.3.5 Glyph substitutions

5.4.3.5.1 Automatic OpenType Layout Features

FontCreator can automatically add a lot of common ligatures and other OpenType Layout Features to your font. The only requirement is that your glyph names are properly set. For a full list of features and their required glyph names please see the following web page:

<http://www.high-logic.com/fontcreator/otlf/features.xml> 

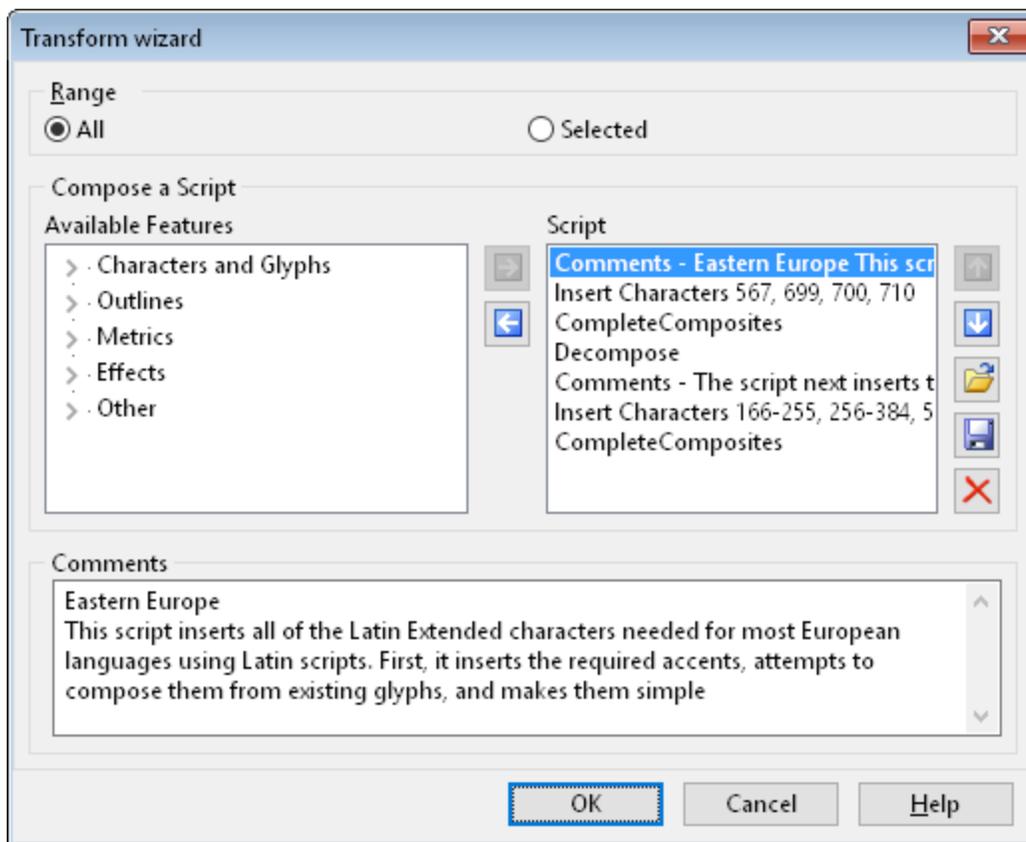
These common features can only be added if the font has no OpenType features yet (or if you explicitly remove those features by clicking the Clear button in the OpenType Designer dialog). To add these common features, open the OpenType Designer, and from there click the **Code Editor** button. FontCreator will then show a confirmation dialog.



5.5 Tools

5.5.1 Glyph Transformer

The **Glyph Transformer** wizard can be selected from the **Tools** menu. Glyph transformations are scripts for changing the shape and size of glyphs, and for automating other repetitive tasks.



FontCreator comes with more than 40 powerful scripts. There are scripts that allow you to change a font into an italic or bold version. Other scripts extend a font's range by adding characters for Greek Extended, Eastern Europe, Vietnamese, Ligatures, Small Capitals, and more. Each script contains descriptive comments and advice on how to use it.

These scripts can be modified to suit your needs, and you can compose custom scripts by adding commands from the list of available features on the left. Save them to use again later using the save icon, and load a saved script using the folder icon. Press the **OK** button to execute the currently loaded script on the current glyph in the glyph edit window, on the selected glyphs, or on the entire font.

Tip: Save the font and copy the selected glyphs before using the Transform wizard, as this operation is not undoable.

Info: If you really want to get the most out of this feature, we encourage you to read the document **Using Glyph Transformations**, available from our website:

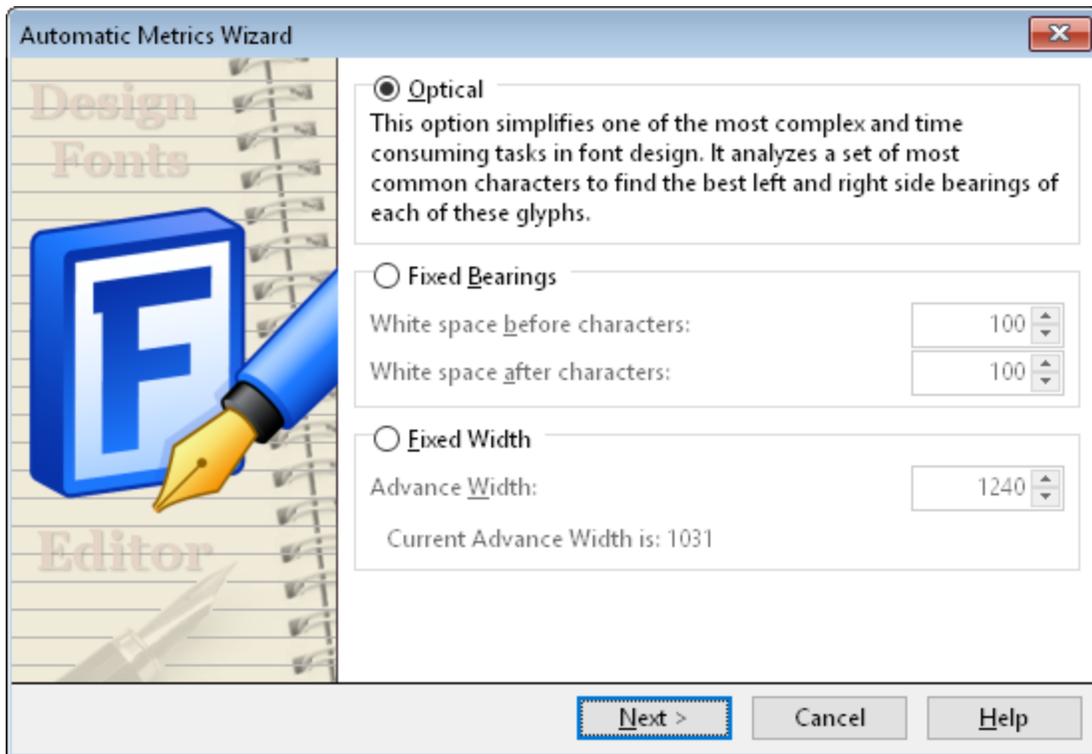
<http://www.high-logic.com/font-editor/fontcreator/tutorials.html> 

Note: The **Transform** wizard is not available in the Home Edition of FontCreator.

5.5.2 AutoMetrics

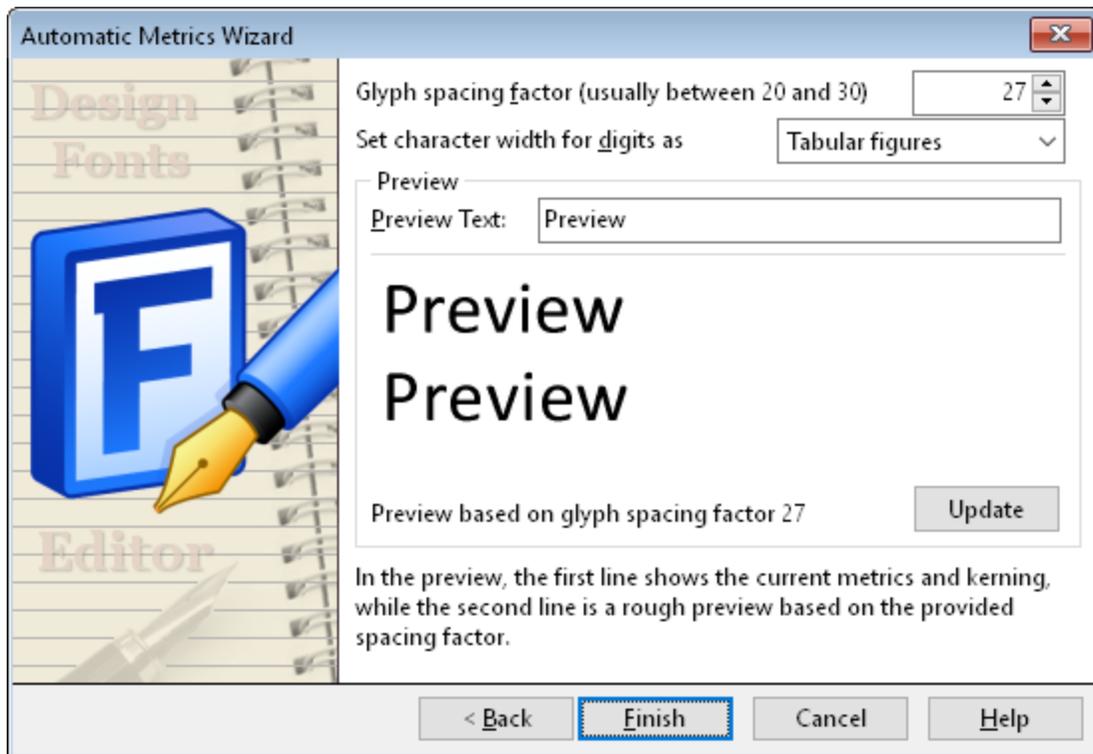
5.5.2.1 Metrics

With the **Automatic Metrics** wizard (select **AutoMetrics** from the **Tools** menu), you can generate the bearings for a selection of glyphs.



Optical

In Optical mode all Latin characters are analyzed to find the best optical space before and after each character. Please note that this process can take from several minutes up to several hours.



Glyph spacing factor allows you to define the distance between glyphs. The larger the factor the more space between glyphs, thus the larger the left and right side bearings.

Character width for digits allows you to choose how to calculate the advance width for digits:

- **Tabular:** It is common digits all have the same advance width, as it allows numerals to align vertically in tables and financial statements.
- **Proportional:** All digits have their own advance width.

Preview is where you can define a preview text sample which will be shown in the preview area.

This feature is not available in the Home and Standard editions of FontCreator.

Note: Add specific OpenType Layout Features if you want to include both Tabular and Proportional digits.

Fixed Bearings

In Fixed Bearings mode the white spaces before and after the selected glyphs will be modified. You can select the glyphs in step 2.

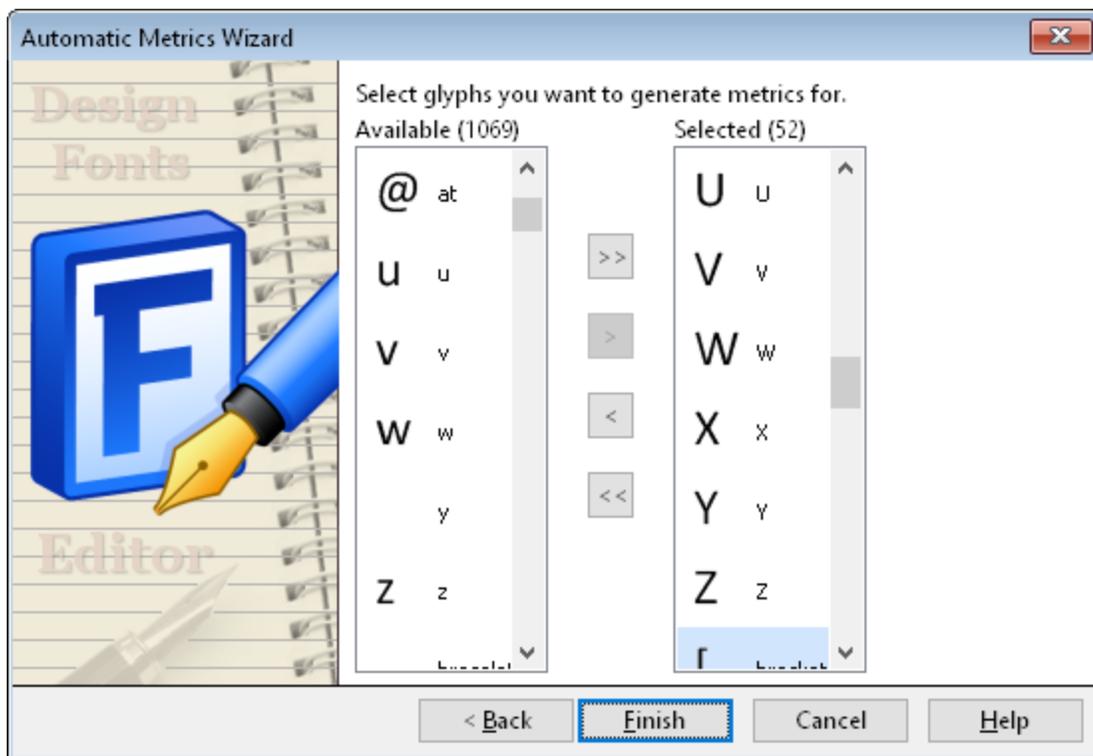
Fixed Width

In Fixed Width mode the advance width will be modified so all selected glyphs have the exact same width. This is especially useful for converting a proportional font into a monospaced font. You can select the glyphs in step 2.

Note: **Fixed** might also be useful for the digits (characters 0 to 9), as it is common they all have the same advance width.

5.5.2.2 Glyph Selection

In the left hand panel you see the glyphs which are present in your font. The right hand panel will contain the glyphs which you select for modification of bearings.



Use the buttons located between the two panels of glyphs to move glyphs:

- The top button will transfer **all glyphs** from the left hand panel to the right hand panel. This is useful if many glyphs need modification. The unwanted glyphs can be selected and returned to the left panel.
- The second button transfers only **selected glyphs** from the left panel to the right panel for modification.
- The third button returns only **selected glyphs** from the right panel to the left panel. These glyph bearings, perhaps, require no modification.
- The bottom button will transfer **all glyphs** from the right panel to the left

panel.

Press the **Finish** button to return to the **Font Overview** window to check the result.

5.5.3 AutoKern

5.5.3.1 Setup

With the **Automatic Kerning** wizard, you can generate kerning pairs for all Latin characters. The pairs will be added to a new pair positioning lookup which is added to the kern feature which is available for Latin script with default language. You can start this wizard by selecting AutoKern from the Tools menu, alternatively you can open this wizard from the OpenType Designer dialog.

Kerning is the reducing/increasing of the space allocated between two glyphs to make them fit more comfortably. Sometimes you want the bearings to be different in special situations. When you want to change the distance between two characters you could use kerning pairs. For example the A and the V could be closer (AV) together than TV.

It is important to first ensure the left and right side bearings are all set correctly for the individual glyphs. This can be done manually through the [Glyph Properties toolwindow](#), or automatically through the [Optical Metrics feature](#).

Not all Operating Systems and applications support kerning. If they don't support kerning they simply ignore the kerning pairs. Many sophisticated word processors, most desktop publishing (DTP) applications, and all modern web browsers have kerning support.

You can use the Preview area at the bottom of the OpenType Designer dialog to test the kerning pairs. You can also install the font and use an application that supports kerning.

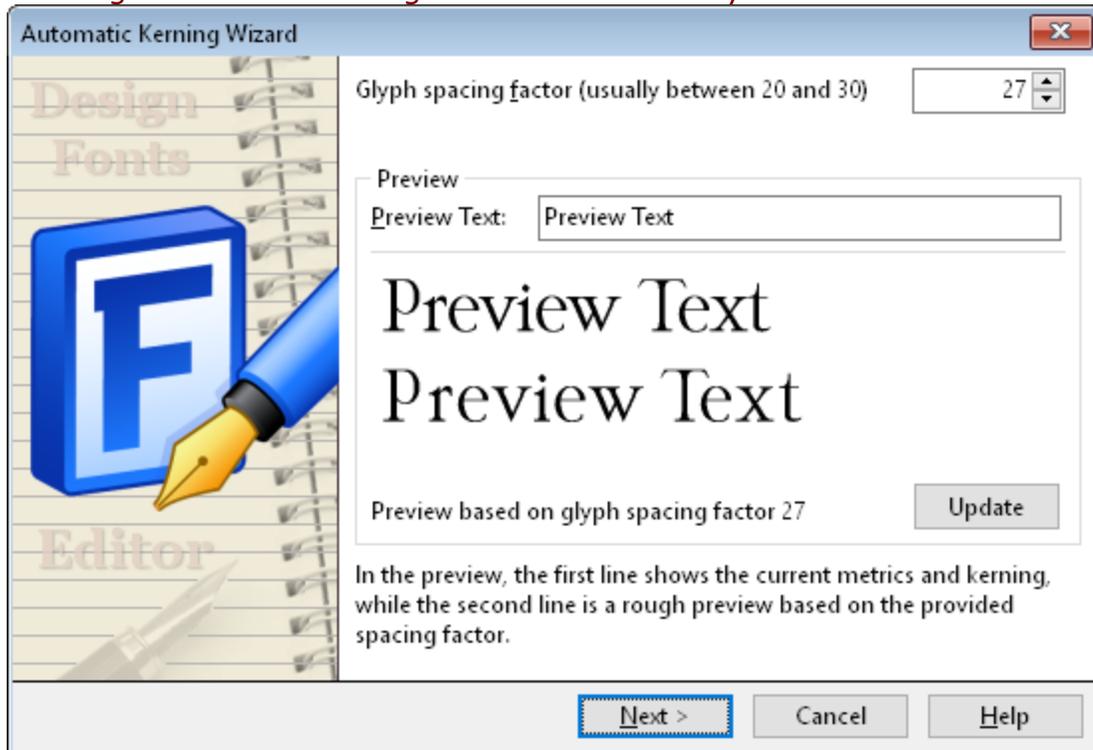
Tip: The [Test Font window](#), which temporarily installs the font, also supports OpenType kerning. The name of the temporarily installed font looks like **FontName 012345**. So while the **Test Font** window is open, you will be able to use the font in any application (e.g. Word). However, when you open the **Test Font** window again you'll have to change the font's name in the application, because the temporary font name identification number always changes.

In Microsoft Office Word select **Font** from the **Format** menu and select the **Character Spacing** tab. There you can turn on kerning in Word by checking the

Kerning for fonts field.

Note: Kerning in Symbol fonts won't be used in Microsoft Word.

Warning: Automatic Kerning can't be used with symbol fonts.



Glyph spacing factor allows you to define the distance between glyphs. The larger the factor the more space between glyphs, thus the larger the left and right side bearings.

Preview is where you can define a preview text sample which will be shown in the preview area.

The **Next** button takes you to the next where you can [set additional options](#).

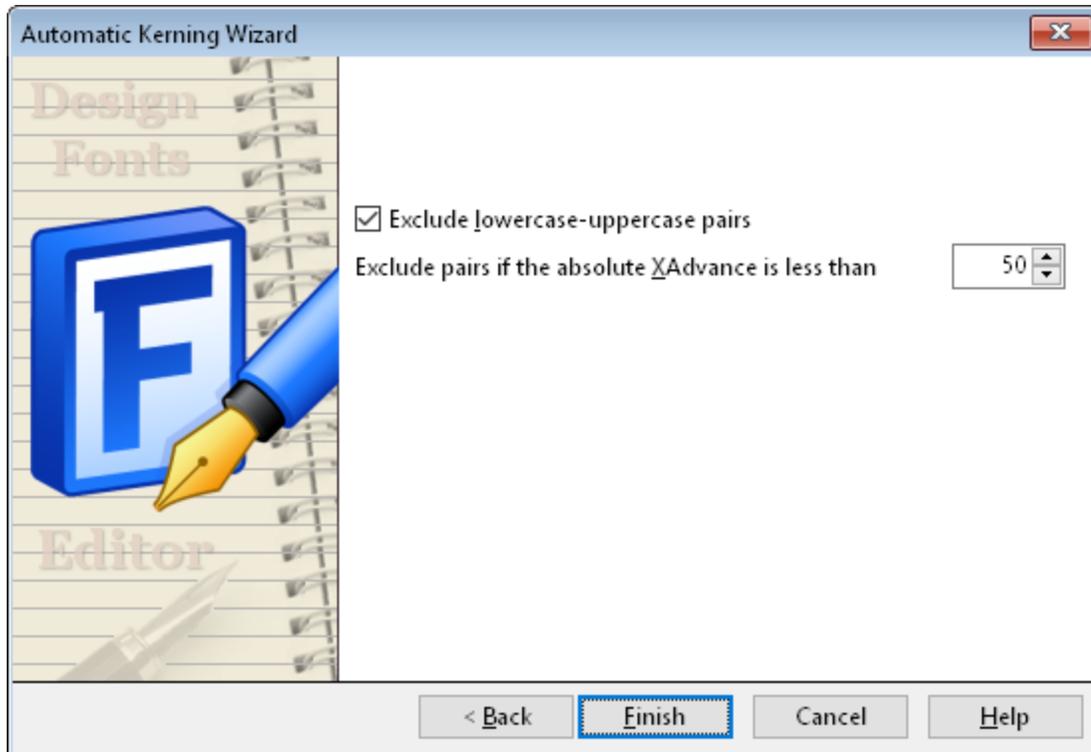
Note: The **Automatic Kerning** wizard is not available in the Home Edition of FontCreator.

See also:

[Autokern existing pair adjustment lookup](#)

5.5.3.2 Kern

In this step, enter the values for white space between glyphs and the minimum absolute kerning value.



Select **Exclude lowercase-uppercase pairs** if you don't want to generate pairs for LC-UC combinations (for example aB and mE).

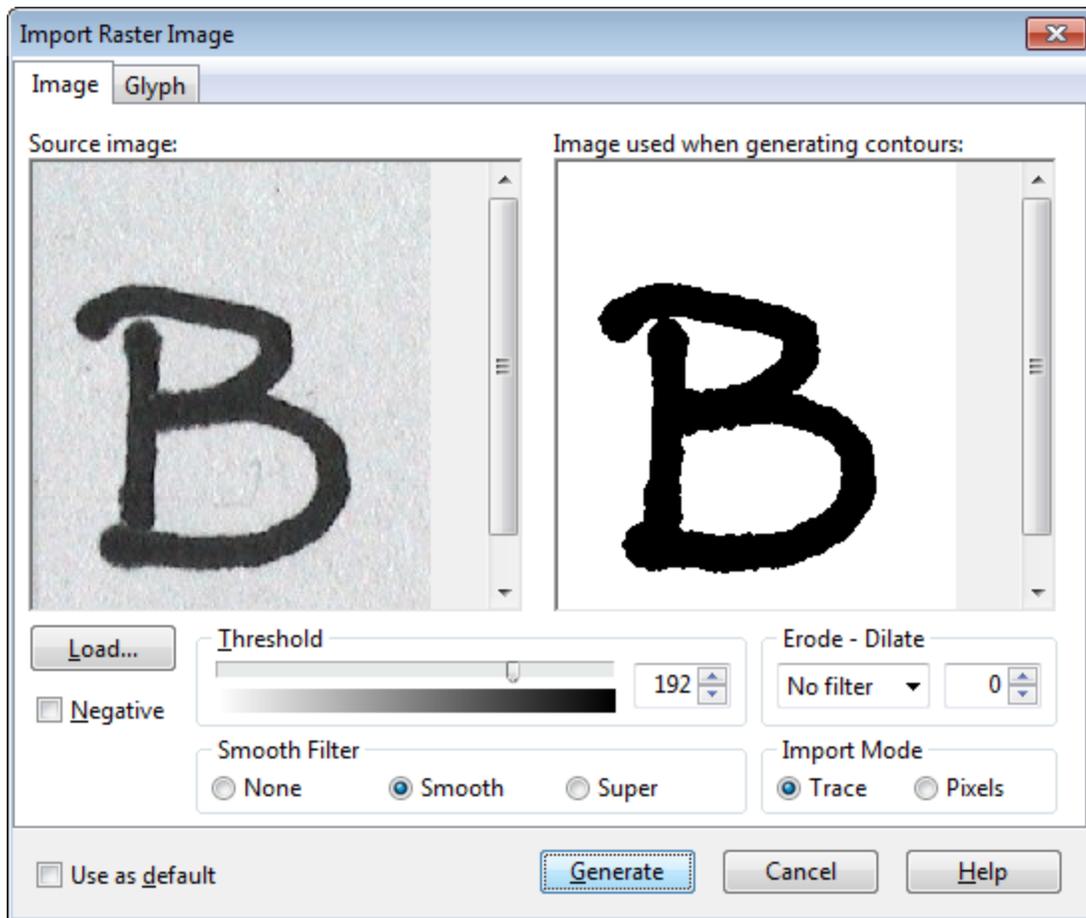
You can exclude pairs if the kern value is below a certain threshold value.

Click the **Finish** button to start generating kerning pairs, otherwise click the **Back** or **Cancel** button.

5.5.4 Import Images

5.5.4.1 Import Raster Image

Import image can be selected from the **Tools** menu when you have activated a **Glyph Edit** window or it can be selected from the context menu after right-clicking in the window.



When you click on the **Load** button you get an open dialog box where you can open an image file (recommended image dimension between 100x100 and 500x500 pixels). The image will be displayed on the left and there will be a bitmap that is going to be used for the conversion on the right. There are some filters and other operations you can apply to the source image before starting the conversion.

The **Threshold** level is used to convert a color image into a black and white image. The **Threshold** level is the lightness value above which colors are turned black. All colors with lightness values above the level are turned into black. At a Threshold level of 1, all colors except white change to black.

Check the **Negative image** field to invert the image.

Dilation causes objects to grow in size and **erosion** causes objects to shrink.

The amount that they grow or shrink depends upon the value specified on the right of the selection box.

Use the Smooth Filter to smooth the image. This will usually reduce the number of generated points.

When the **Import Mode** is set to **Trace**, the image will be converted with curves. This is the recommended setting. In case you don't want curves (especially useful for bar code fonts and pixel fonts), set the **Import Mode** to **Pixels**.

Check the **Default** field to save the current settings as the default for each new import. These settings are also used when pasting an image from the clipboard. Choosing to press the **Cancel** button will retain these default settings.

Use the settings on the **Glyph** tab to position and resize the generated contours.

When you press the **Generate** button the conversion will start.

Tip: You can also paste an image from the clipboard or drag and drop image(s) from explorer into the **Font Overview** and **Glyph Edit** windows.

Note: You can't import images into composite glyphs.

5.5.4.2 Import Vector Image

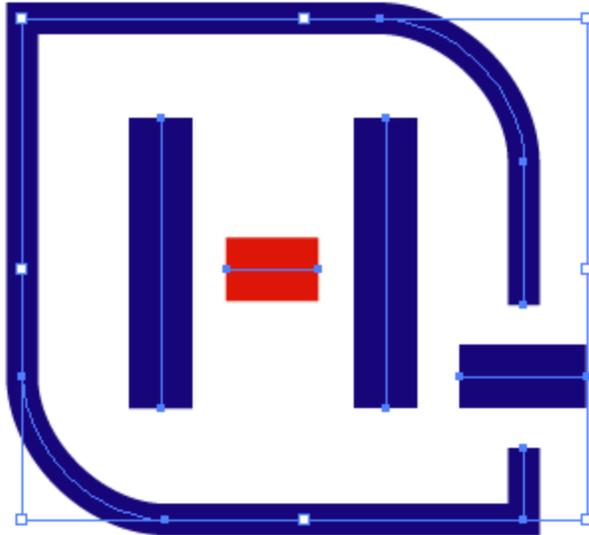
Import Image can be selected from the **Tools** menu when you have activated a **Glyph Edit** window or it can be selected from the context menu after right-clicking in the window. When you open a vector based image file, the file will be instantly imported.

In vector based image editing software you can use all kinds of objects to create your images. Such objects can be paths, lines, shapes, text, etc. That's not it, as you can also apply specific strokes to each of these objects. These strokes control thickness, how segments join, and the appearance of both ends of an open path. There are numerous other capabilities like fill objects, gradients, etc.

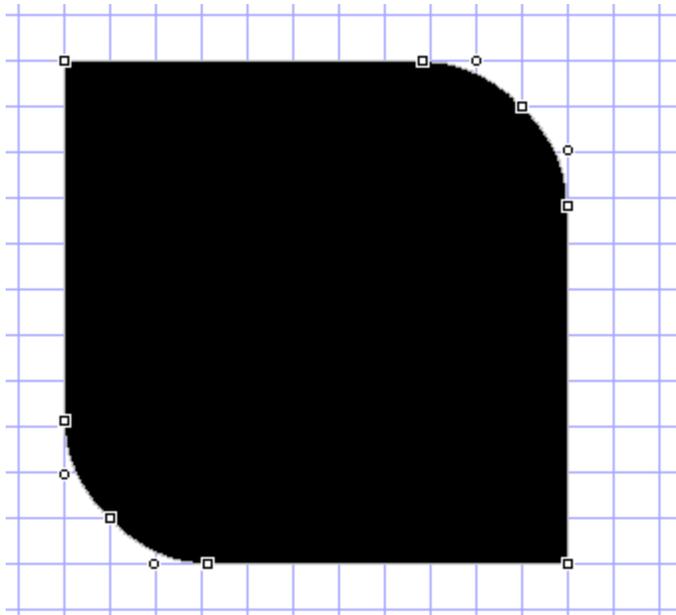
FontCreator can only extract the bare paths, thus completely ignores the strokes, fills, etc. And since glyphs can only consist of closed contours, all open paths are automatically closed.

Example

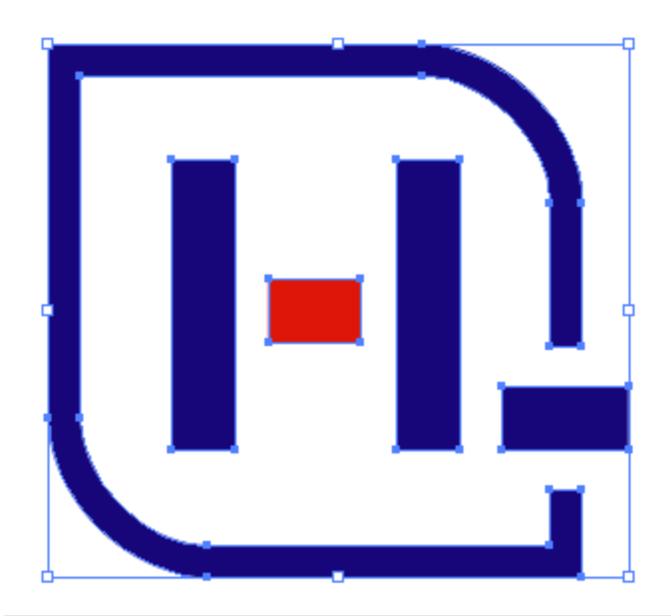
Here is a logo created with Adobe Illustrator. It contains four lines and a rectangle with two rounded corners and a gap at the lower right. All of these objects contain strokes to give them thickness.



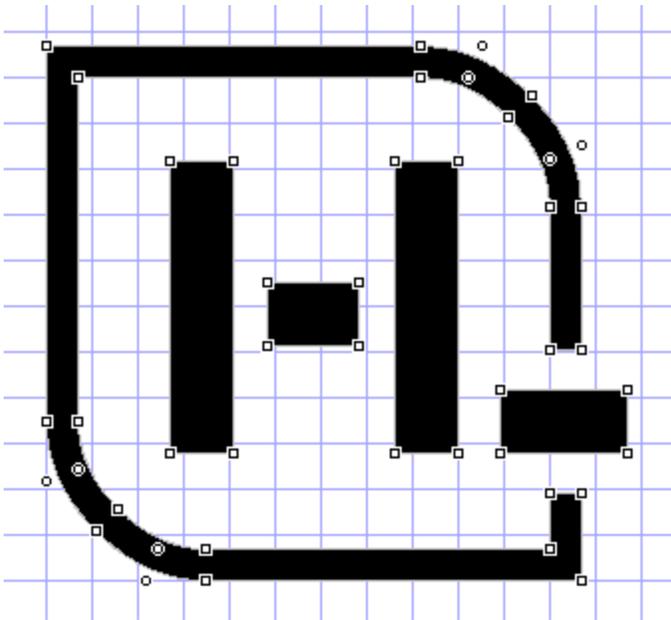
When imported into FontCreator, the result is not as intended as the strokes are not imported, thus the thickness of the objects is stripped off. The lines are all discarded as contours with only two points are useless. The rectangle is imported but the gap is no longer there.



One easy step ensures our vector image can be safely imported into FontCreator. In Adobe Illustrator, select all objects, then from the main menu select Object -> Path -> Outline Stroke. The result is shown here:



When we import this version of the vector image the result is a perfect fit!



5 contours with 44 points is the final result.

Tip: You can also paste an image from the clipboard or drag and drop image(s) from explorer into the **Font Overview** and **Glyph Edit** windows.

Note: You can't import images into composite glyphs.

5.5.5 Sorting Glyphs

To change the order of the glyphs within a font select one of the options from the Sort Glyphs submenu in the Tools menu.

No matter which option you select, these glyphs (if available) will always come first in the new order.

- .notdef
- .null
- .nonmarkingreturn

Unicode Codepoints

Glyphs will be sorted by these rules:

- Glyphs are sorted by their Windows mappings
- Then all remaining glyphs are sorted by their glyph names

The Unicode code points are retrieved from the Windows Character to Glyph Index Mappings.

Glyph names

Glyphs will be sorted by these rules:

- The glyphs are sorted by their Glyph names
- Then all remaining glyphs are sorted by their Windows mappings

Glyph type (empty, simple, composite)

Glyphs will be sorted by these rules:

- The glyphs are sorted by their glyph type
- Then all remaining glyphs are sorted by their Windows mappings

Alphanumeric

Glyphs will be sorted by these rules:

- The glyphs are sorted by their mapped character. This features uses default compare functions as used in Windows.
- Then all remaining glyphs are sorted by their Windows mappings

5.5.6 Generate Glyph Names

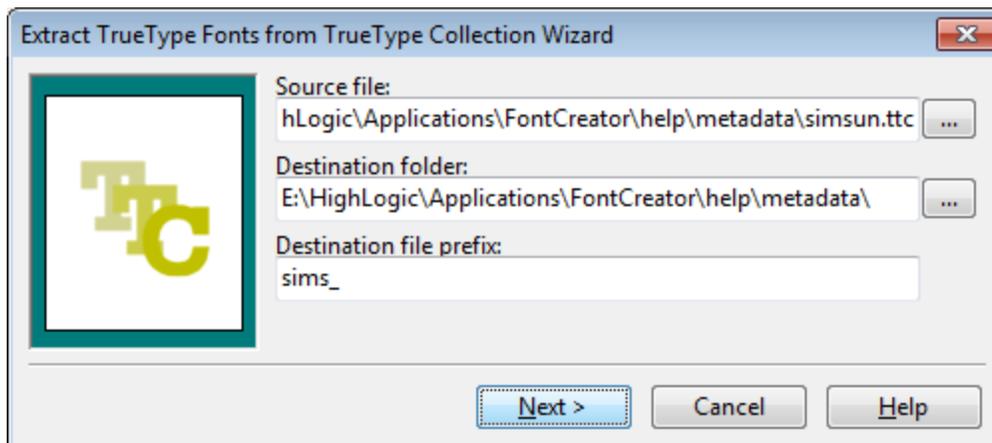
Generate Glyph Names will generate glyph names for all glyphs with a known code-point. It will also try to set names for glyphs that are used in the specified OpenType Layout Features.

Note: You can manually add your own glyph names by editing the "glyphnames.dat" file in the user data folder.

See [FontCreator data files](#) for more information.

5.5.7 TrueType Collection

A TrueType Collection file is one or more fonts (TrueType or OpenType) combined into one file. The **Extract TrueType Fonts from TrueType Collection Wizard**, available from the **Tools** menu, can extract those fonts.



Source file

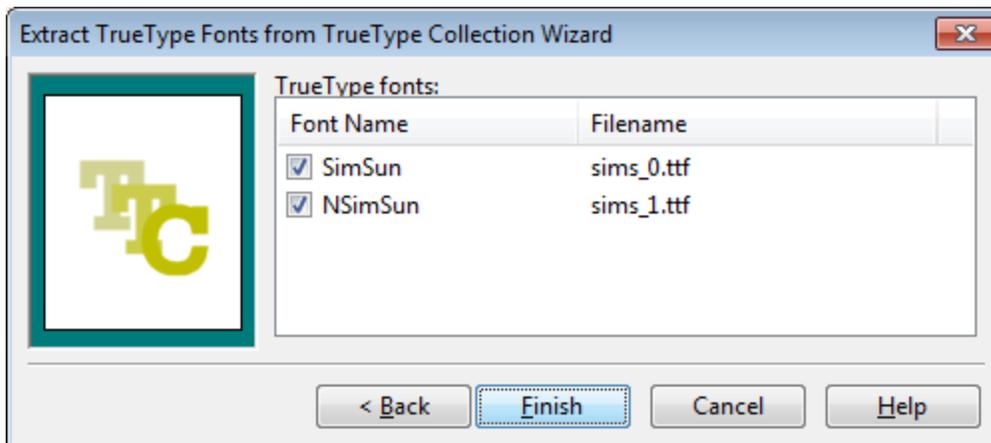
Select the TrueType Collection file.

Destination folder

The extracted font files will be saved in this folder.

Destination file prefix

The extracted files start with this prefix.



Here you will see all fonts available in the TrueType Collection. Select the fonts you want to extract and press the **Finish** button.

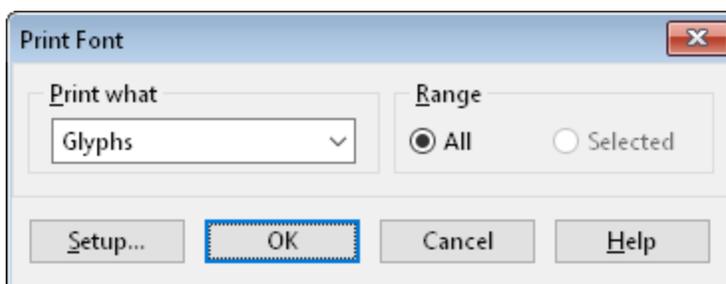
5.5.8 External Tools

To quickly access MainType, Windows Fonts folder and Character Map select **Launch External** from the **Tools** menu and select the tool you want to open.

5.6 Printing

5.6.1 Print Font

This option (select **Print** in the **File** menu) is available when the **Font Overview** window is active. If a **Glyph Edit** window is active you will be able to [print a glyph](#).



You can choose what kind of font information you want to print:

- Glyphs
- Properties

Print Glyphs

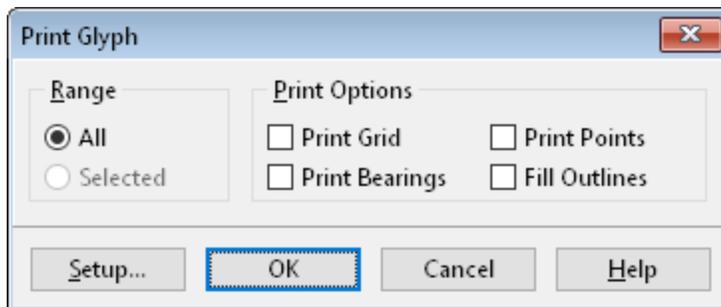
Print all or the selected glyphs.

Print Properties

Print all fields from the [Font Properties window](#).

5.6.2 Print Glyph

Select **Print** in the **File** menu to print a single glyph. This option is available when a **Glyph Edit** window is active. If the **Font Overview** window is active you will be able to [print the font](#).

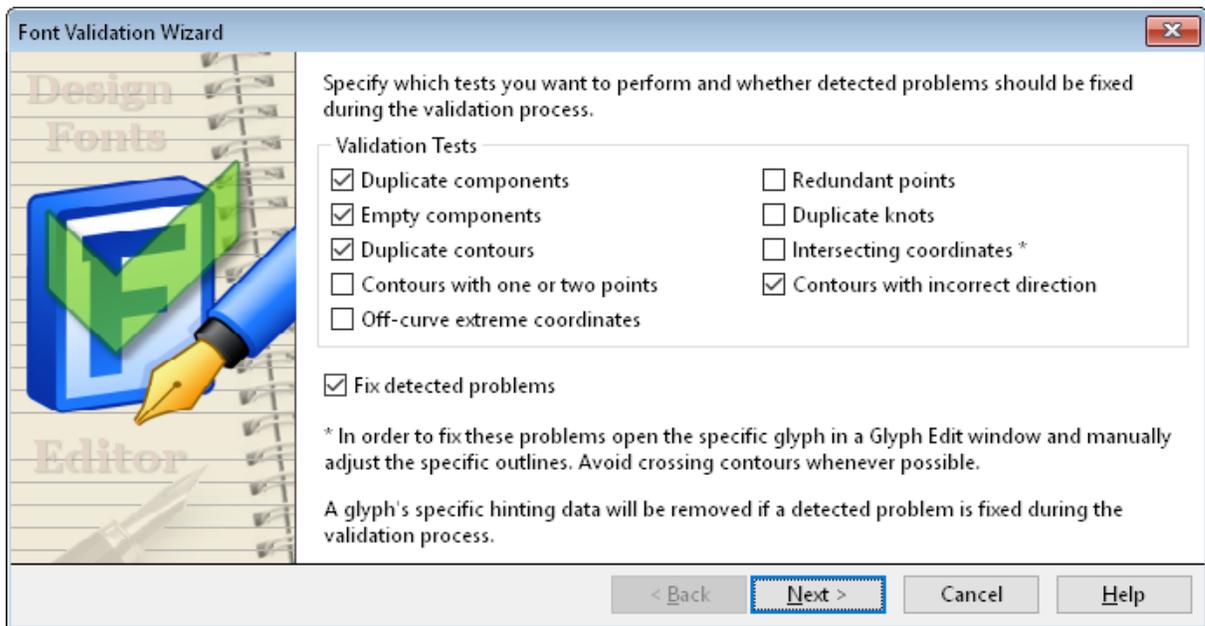


The print options allow you to print the grid, points and bearings, and you can choose to fill the outlines.

5.7 Font Validation

5.7.1 Setup

Designing glyphs can be very complicated. The **Font Validation** wizard, available from the **Font** menu, identifies common potential problems and if possible points you to the specific item (e.g. glyphs, contours and coordinates). It validates all glyphs and optionally fixes detected problems.



The specific validation tests are explained here:

Duplicate components

This problem will be reported when validating composite glyphs with two or more identical glyph members.

Empty components

This problem will be reported when validating composite glyphs with empty glyph members.

Duplicate contours

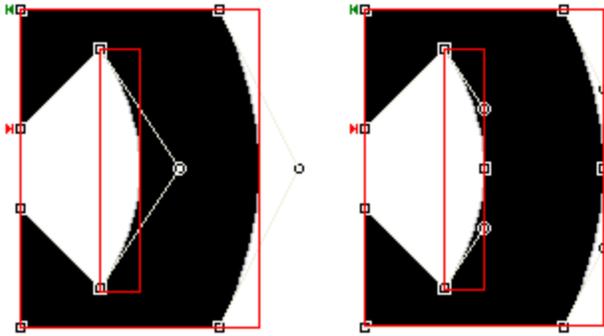
This problem will be reported when validating simple glyphs with two or more identical contours.

Contours with one or two points

This problem will be reported when validating simple glyphs with contours with one or two points.

Off-curve extreme coordinates

This test checks whether all off-curve points are inside the global bounding box (and optionally their local bounding box). You can customize this feature through the Options window.



The left image shows two red rectangles. The large one is the global bounding box and the smaller rectangle is a local bounding box between an on-off-on curve sequence. Both off-curve points lie outside their bounding box. With local detection enabled, the right image shows the result of clicking the **Add on-curve extremes** button on the **Validation** toolbar.

Redundant points

This problem will be reported when validating simple glyphs with contours with redundant points.

Duplicate knots

This problem will be reported when validating simple glyphs with contours with two adjacent points that have the same coordinates but one is on-curve and the other is off-curve.

Intersecting coordinates

This problem will be reported when validating composite glyphs with intersecting glyph members and when validating simple glyphs with (self-)intersecting contours. Avoid crossing contours whenever possible.

Note: Older PostScript Level 2 drivers do not support overlapping contours.

Contours with unknown direction

This problem will be reported when validating simple glyphs with contours that have an unknown direction.

Contours with incorrect direction

This problem will be reported when validating simple glyphs with contours that have an incorrect direction (Contours that need to be filled black must have a clockwise direction. If we want to make a white area inside an existing contour we must make the direction of the new contour counterclockwise.).

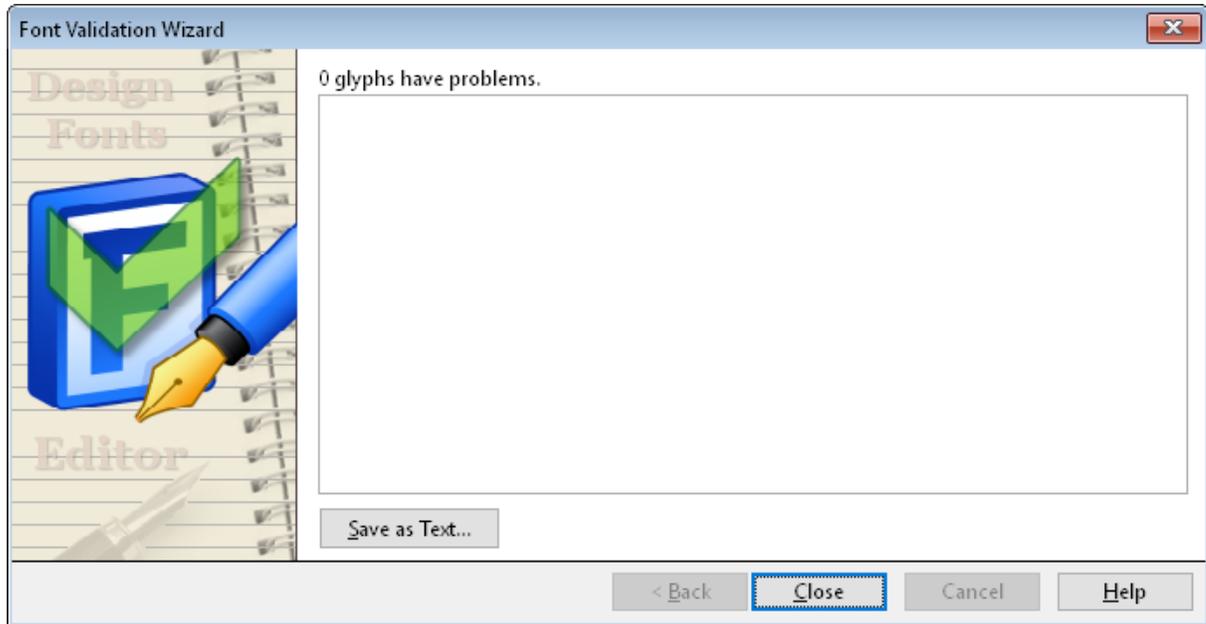
Note: This test will not be performed when Duplicate contours, Intersecting

coordinates or Contours with unknown direction have been reported.

Note: The Validation features are not available in the Home Edition of FontCreator.

5.7.2 Results

After the validation process the (remaining) problems are shown for all glyphs. Optionally the report can be saved.



Note: The **Validation** features are not available in the Home Edition of FontCreator.

5.8 Testing and Installing Fonts

5.8.1 MainType

MainType is a powerful font manager for Windows that helps you maintain your fonts. Although its main purpose is installing and uninstalling fonts, it also allows you to insert special characters into documents and the **Test** window.

You can download MainType from here:

<http://www.high-logic.com/> 

5.8.2 International Keyboard

Instead of memorizing a long list of Alt number combinations, or using the Character Map, you could change your language and layout from English-US to

United States-International (or any other available language on your system).

Here's how:

- Single-click the Start menu, mouse over Settings, and then single-click the Control Panel.
- Double-click the Keyboard icon and then click the Language tab
- Click Add.
- Under "Keyboard layout," place a checkmark next to United States-International.
- Click OK.
- Click Apply. You may be asked to insert your Windows system disk to finish loading the process.

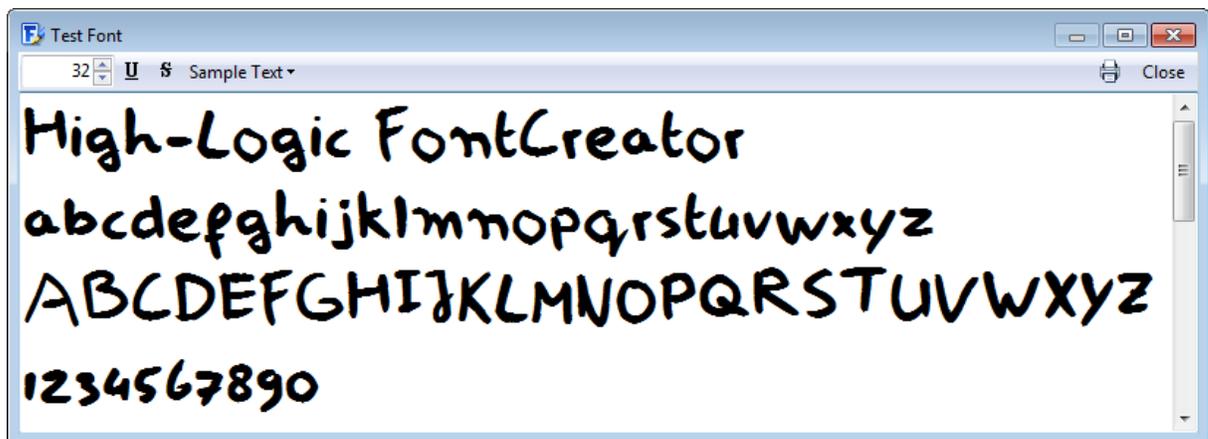
You can type in other languages without knowing the Alt codes for a non-English alphabet. For example, type ~ followed by N to get Ñ. A tilde followed by any letter will place the appropriate accent on the letter you choose.

5.8.3 Test Your Font

5.8.3.1 Test TrueType/OpenType

Test TrueType/OpenType font

If you want to know how your font is going to look you can test it any time during the development process. Choose **Test TTF/OTF...** from the **Font** menu or use the shortcut **F5**

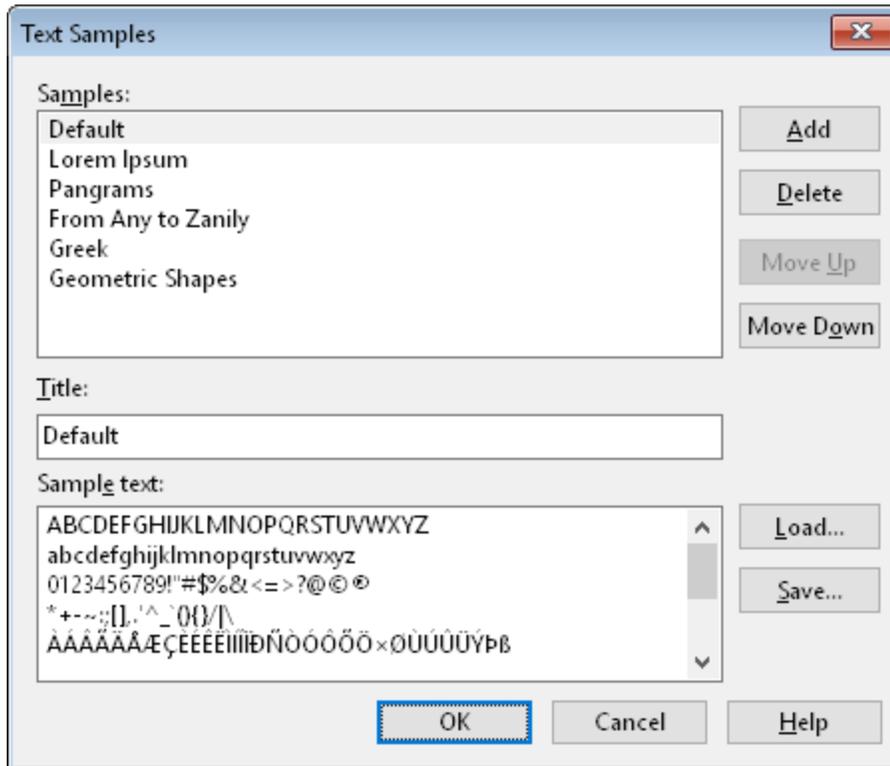


You can enter your own text into the **Test** window. FontCreator will remember this text so you can always test your fonts with your preferred text.

If you want to test how your font will look when it's printed you press the **Print** button.

5.8.3.2 Test Font - Edit Text Samples

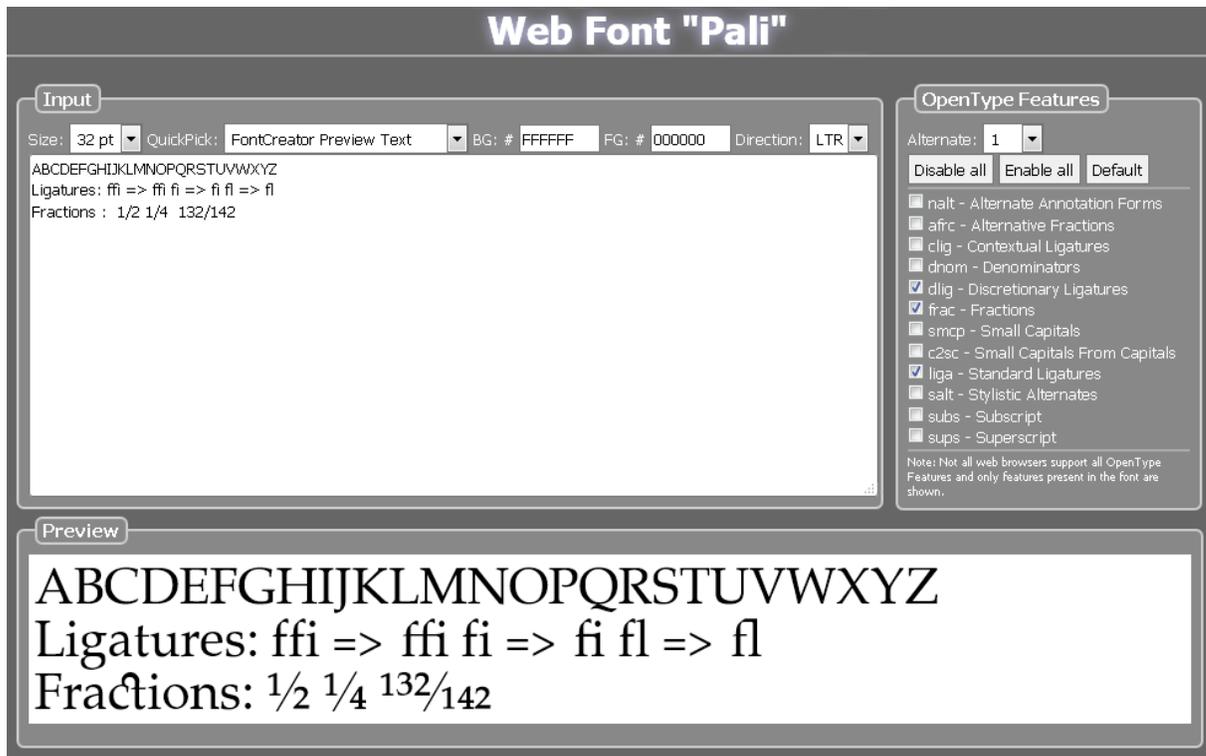
You can edit the text samples used on the test TrueType/OpenType dialog by right-clicking on the test area and select **Edit Text Samples**.



Here you can add, delete and change the order of the text-samples.

5.8.3.3 Test Web font

You can also test your font and **OpenType Layout Features** as a web font on a locally generated web page. Choose **Test WOFF** from the **Font** menu or use the shortcut **CTRL+F5**



If OpenType Layout Features are present, you can toggle them using the checkboxes on the right.

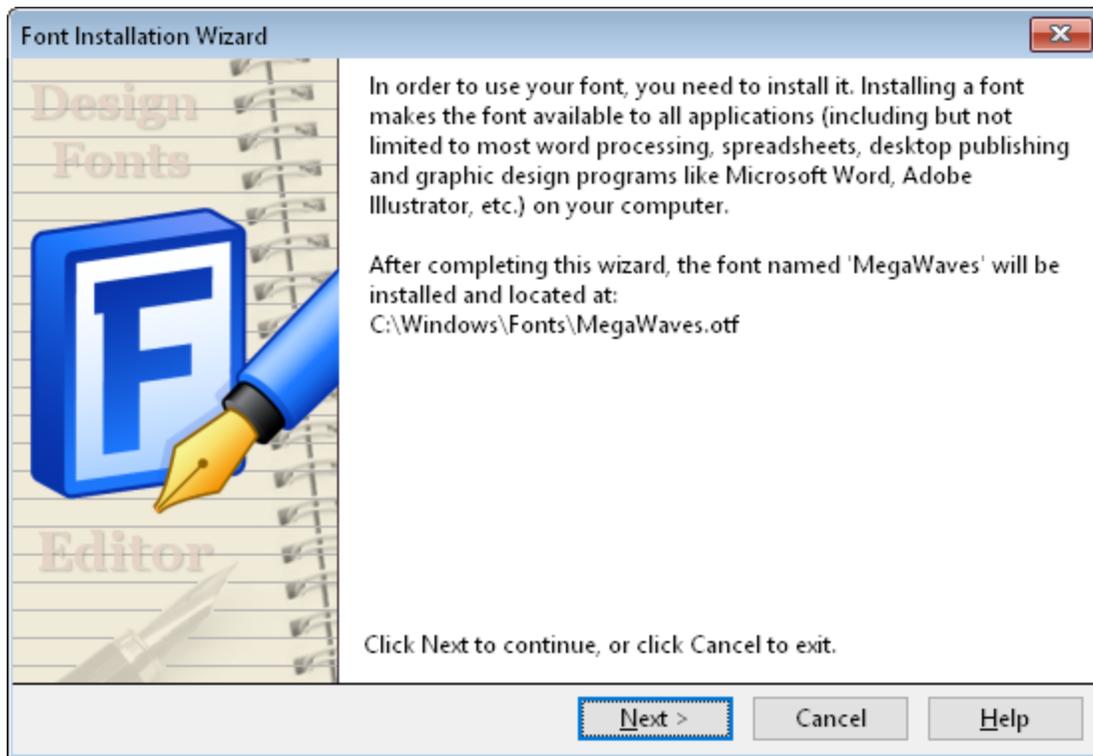
5.8.4 Installing Fonts

Although it is possible to install a font through Windows fonts folder, FontCreator has its own **Font Installation** wizard. To make your font available to other applications select **Install** in the **Font** menu.

The **Font Installation** wizard will guide you through the installation process. At the end of the installation process you will be informed that the font is installed successfully. Now you will be able to select the font in any program that supports TrueType fonts.

Note: To install a modified font, the font must be exported to file.

Note: It is not recommended to export font files directly into Windows Fonts folder.



Note: If you are reinstalling the font, it is recommended you delete the font BEFORE installing the new version.

Note: Don't just drop the font into the Windows Fonts folder!

5.8.5 Character Map

With **Windows Character Map** you can insert special characters into documents and the **Test** window.

From your Start menu mouse over Programs, Accessories, and System Tools, and then click on Character Map.

In the Character Map select the appropriate font. Click the letter or symbol you need and click the Select button. Add more characters if needed and finally click the Copy button. Now go to your document and select Paste from the Edit menu or use the shortcut Ctrl-V.

Note: To type special characters (like the copyright sign) of the font in your word processor or page layout program, hold down the Alt key, and then, by using the numeric keypad, type 0 (zero) followed by the corresponding decimal character code. Make sure NUM LOCK is on.

Another way to input Unicode characters is the hexadecimal entry method that works with WordPad and Microsoft Word. Probably more applications will follow soon. Basically you type a character's hexadecimal code (in ASCII), making corrections if needed, and then type Alt+X. This will replace the hexadecimal code by the corresponding Unicode character. The same key combination ALT-X can be used to reveal a character's code. If the hexadecimal code is preceded by one or more hexadecimal digits, you need to "select" the code so that the preceding hexadecimal characters aren't included in the code. The code can range up to the value 0x10FFFF, which is the highest character in the 17 planes of Unicode.

Part



6 Toolbars

6.1 Overview

The toolbars give you one-click access to many of the commands on the menus. Some menu items have toolbar icons next to them so that you can quickly associate the command with that icon.



FontCreator's toolbars can be shown or hidden as needed. By default the **Standard, Drawing, Grid, Glyph, Font Overview** and the **Align or Distribute** toolbars are docked below the menu bar. Choose **Toolbars** from the **View** menu to select toolbars to display or hide, or right-click on any toolbar to get the same submenu.

The window tab bar cannot be moved or undocked. You can however close it through the **Toolbars** item from the **View** menu.

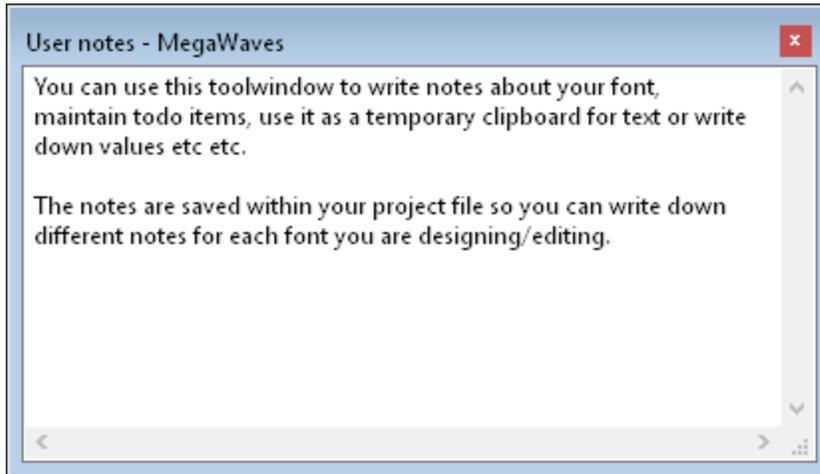
Tips: Toolbars can be rearranged by dragging, docked on the left, right, top, or bottom of the FontCreator window, or made floating. In the **Glyph Edit** Window, to quickly close a floating toolbar, click the **Close** button on that toolbar, or use its shortcut key.

To undock a docked toolbar, double-click its grab handle where the four-arrow cursor is displayed. To dock it again, double-click the floating toolbar's title bar. The **Transform, Validation, Preview, Background Image** and **Comparison** toolbars cannot be docked, but they can be toggled on/off with shortcut keys F6, F7, F8, F9, and F11 respectively.

6.2 User Notes

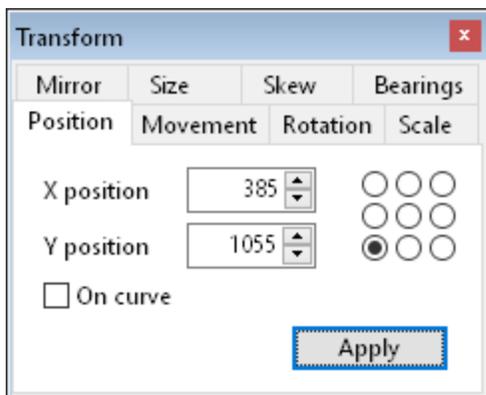
The **User Notes** toolbar can be used to write down some text that will be saved

with your project. The User notes will not be exported to your fonts.



6.3 Transform

The **Transform** toolbar contains powerful tools for editing simple glyphs. Contours can be precisely repositioned, resized, moved, skewed, scaled, rotated, or mirrored. Nodes can be repositioned or moved by precise increments. The glyph's bearings can also be set.

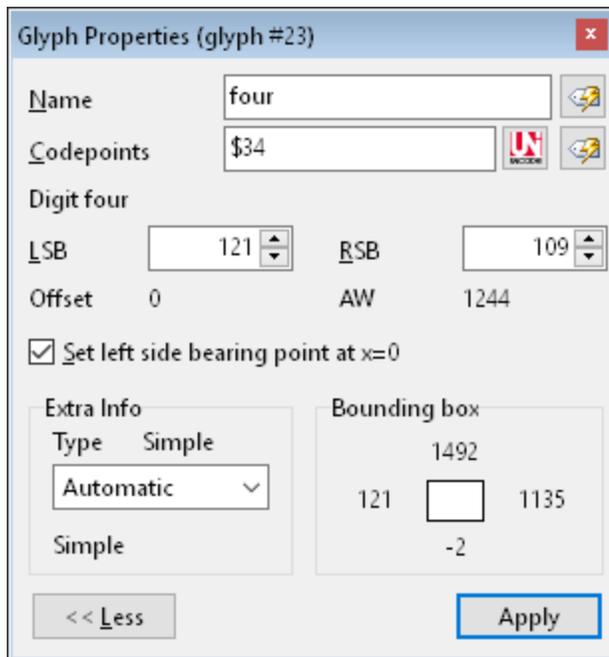


The Transform toolbar can be toggled on and off using the **F6** shortcut key, but is only available in the **Glyph Edit** Window.

6.4 Glyph Properties

6.4.1 Glyph Properties

The **Glyph Properties** toolbar provides a quick way to view and edit some of the most common glyph properties.



Glyph Name

The Glyph Name of the glyph. Press the Generate Name button  to let FontCreator fill in the field.

Codepoints

The codepoints assigned to this glyph. Press the **Select Unicode Character** button  to select a character from the Unicode Character list. Press the

Generate Codepoint button  to let FontCreator fill in the field automatically. It is possible to enter multiple code points by separating them by commas, but it's common to have only one character mapped to each glyph.

Unicode name

The Unicode name of the first code-point

Left Side Bearing (LSB)

The (horizontal) start position of the glyph.

Right Side Bearing (RSB)

The (horizontal) end position of the glyph.

Offset

The horizontal offset between $x=0$ and the **Left Side Bearing**.

Advance Width (AW)

The width of the glyph and its surrounding space.

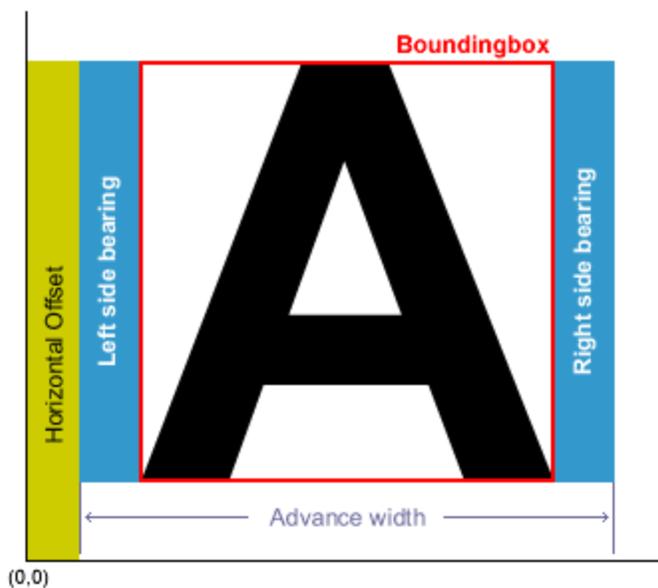
More/Less

This button allows you to collapse or expand the toolbar, showing only the most common information.

Extra Info:

Type: Shows the glyph outline type which can be empty, simple or composite.

OpenType Type Combo box: Shows the Glyph OpenType Type, which is important for the [OpenType Layout Features](#). It can be set to Automatic (so FontCreator determines the actual type), Unassigned, Simple, Ligature, Mark, or Component. With Ligatures an additional edit box allows you to specify the number of components. This defines the number of anchor values used with [MarkToLigature lookups](#).

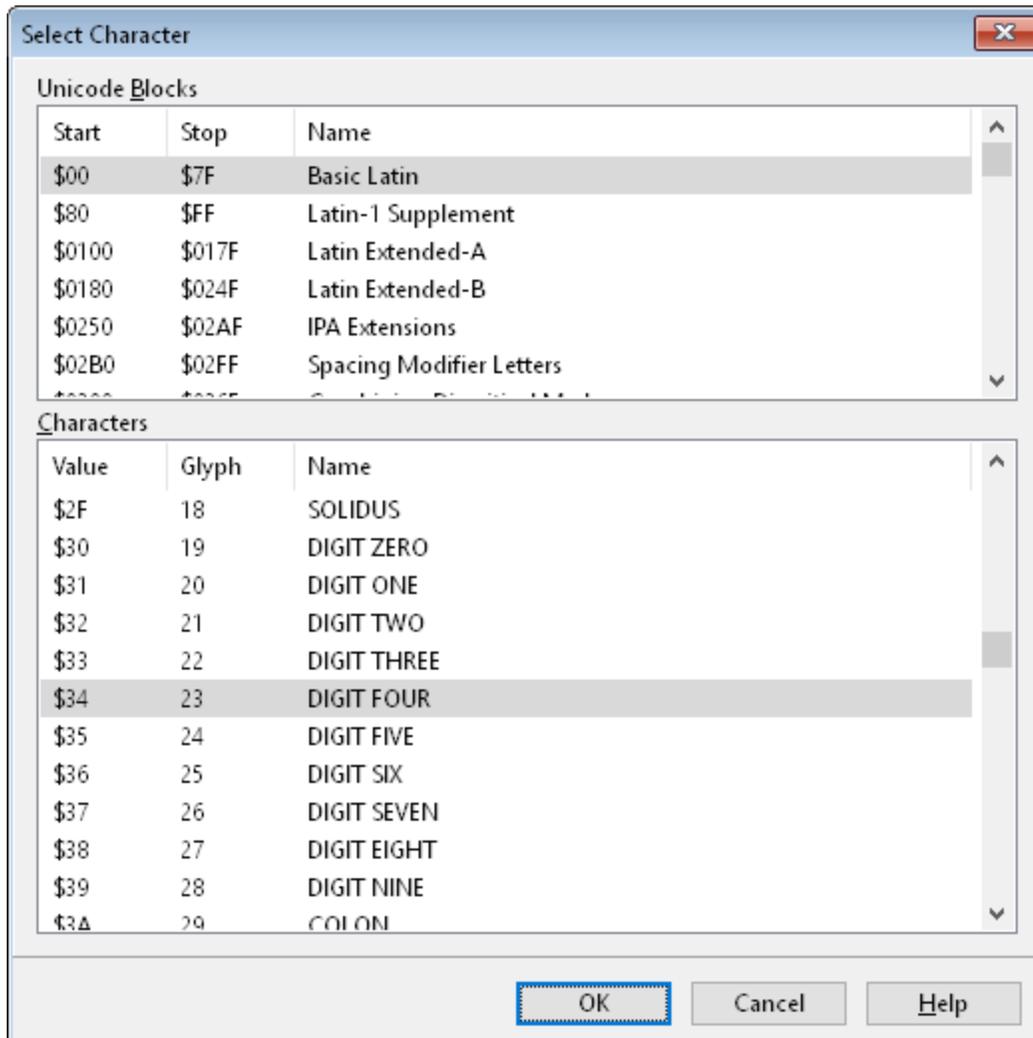


Tip: You could also adjust the **Left Side Bearing** and **Advance Width** in the Glyph Edit window, by changing the vertical [bearings](#).

6.4.2 Select Character

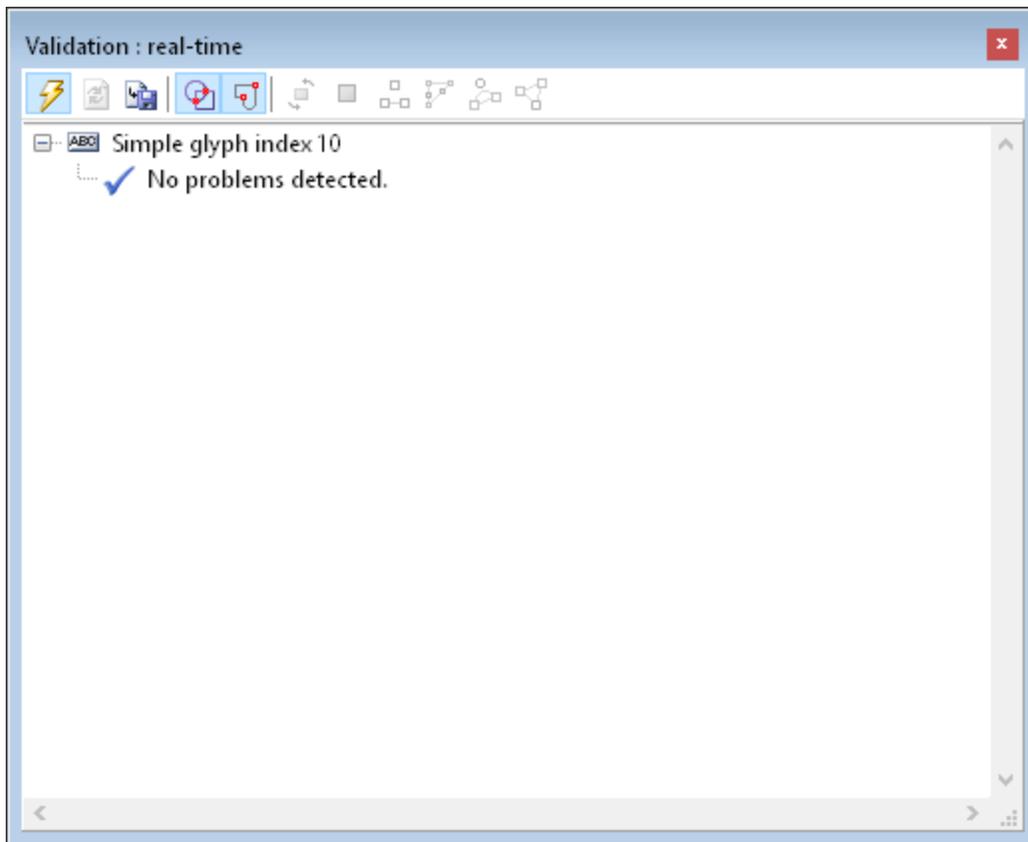
The **Select Character** dialog provides a quick way to change a character mapping. This dialog is available through the **Glyph Properties** tool window by

clicking the **Select Character** button .



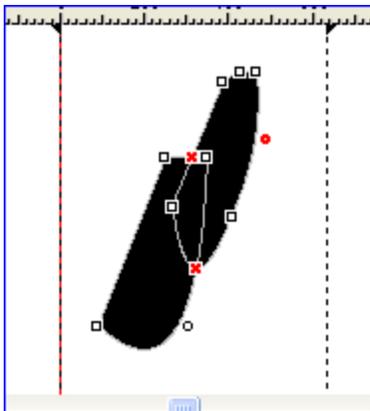
6.5 Validation

Use the **Validation** toolbar to locate and solve common glyph problems. It can be toggled on and off using the **F7** shortcut key (or **Show glyph validation report** button on the **Glyph** toolbar), but is only available in the **Glyph Edit** window.



Note: Because the validation can be very time-consuming, real-time glyph problem detection will be disabled for too complex glyphs. The limits for real-time glyph validation are set on the **Validation** tab of the **Options** dialog.

Red marks in the **Glyph Edit** window will show the position of the located problems.



Not every problem should be classified as an error, it is the designer's decision to correct or ignore potential problems. Double-click on a reported problem to focus the problematic point or contour. There are several buttons on the **Validation** toolbar that will perform actions that will automatically solve some specific problems.

Enable real-time glyph problem detection

Use this button to enable and disable real-time glyph problem detection.

Refresh

When real-time validation is not enabled, use this button to revalidate the glyph.

Save report

Use this button to save the report to file.

Show intersecting components and contours

Intersections will be shown when this button is enabled and is down. Red crosses in the **Glyph Edit** window will show the position of the intersections.

Show warning points

Warning points will be shown when this button is enabled and is down. Red bullets in the **Glyph Edit** window will show the position of the located problems.

Correct contour directions

Use this button to correct the direction of all misoriented contours in a simple glyph. This button is only enabled when contour direction problems are detected.

Note: This test will not be performed when **Duplicate contours** or **Intersecting coordinates** have been reported.

Remove duplicate components and contours

This button will remove duplicate components from a composite glyph and will remove duplicate contours from a simple glyph.

Remove empty components and contours with one and two points

This button will remove empty components from a composite glyph and will remove contours with one and two points from a simple glyph.

Remove redundant points

Press this button to remove all redundant points.

Note: This feature won't remove duplicate knots as this would affect the outline.

Add on-curve extremes

Press this button to add on-curve extremes. This feature will add global or local extremes, as customized through the **Options** window.

Note: As this feature generates new points, this could lead to new redundant points.

Note: The **Validation** features are not available in the Home Edition of FontCreator.

See also:

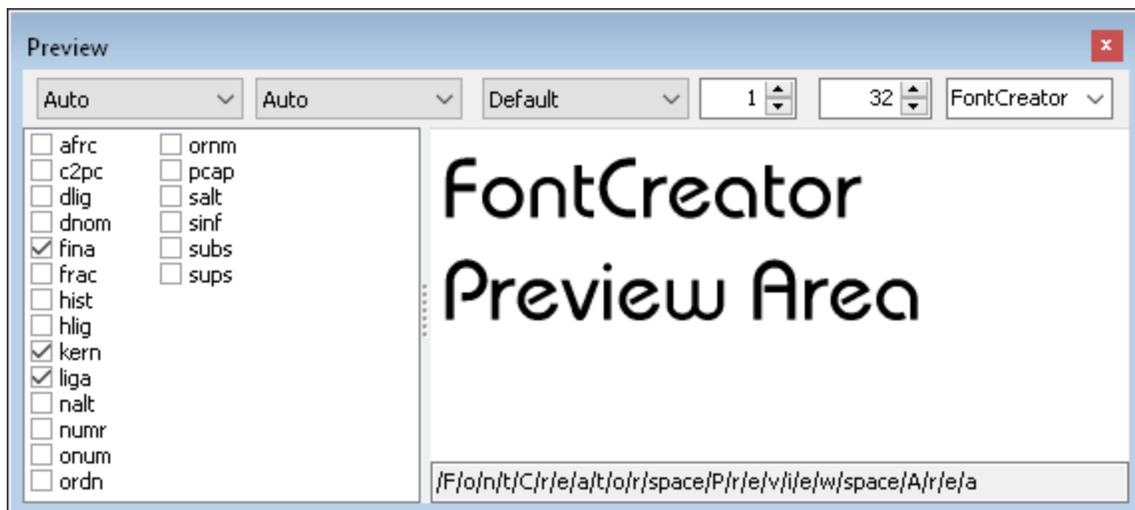
[Font Validation](#)

6.6 Preview

While editing a font, you can preview the results with the Preview toolbar. You can choose a standard text sample from the drop-down list or enter your own text. It can be toggled on and off using the **F8** shortcut key and is available in both the **Font Overview** window and the **Glyph Edit** window.

Selected glyphs in the overview window can be displayed in the **Preview** toolbar by pressing the **"P"** shortcut key. You can also use **"Shift P"** to add the selected glyphs to the current text.

The preview is not a full blown OpenType text shaping engine, but is still extremely useful while testing your font, metrics, kerning and other OpenType layout features it contains. Use the [Test TrueType/OpenType](#) to see how your font behaves in Windows, and use [Test Web font](#) to test it in your default Web browser.



The drop down lists available at the upper left and the first edit control after these lists are all related to OpenType Layout Features as defined in the OpenType Designer dialog.

The edit control can be used to select which alternates to show.

To test your kerning, select the **kern feature** in the list of features. You can select multiple features so you can see how they work together.

On the upper right side there are two fields which allow you to set the size of the text and the text itself. To force a line-break in the preview area, enter "/" newline" in the preview text.

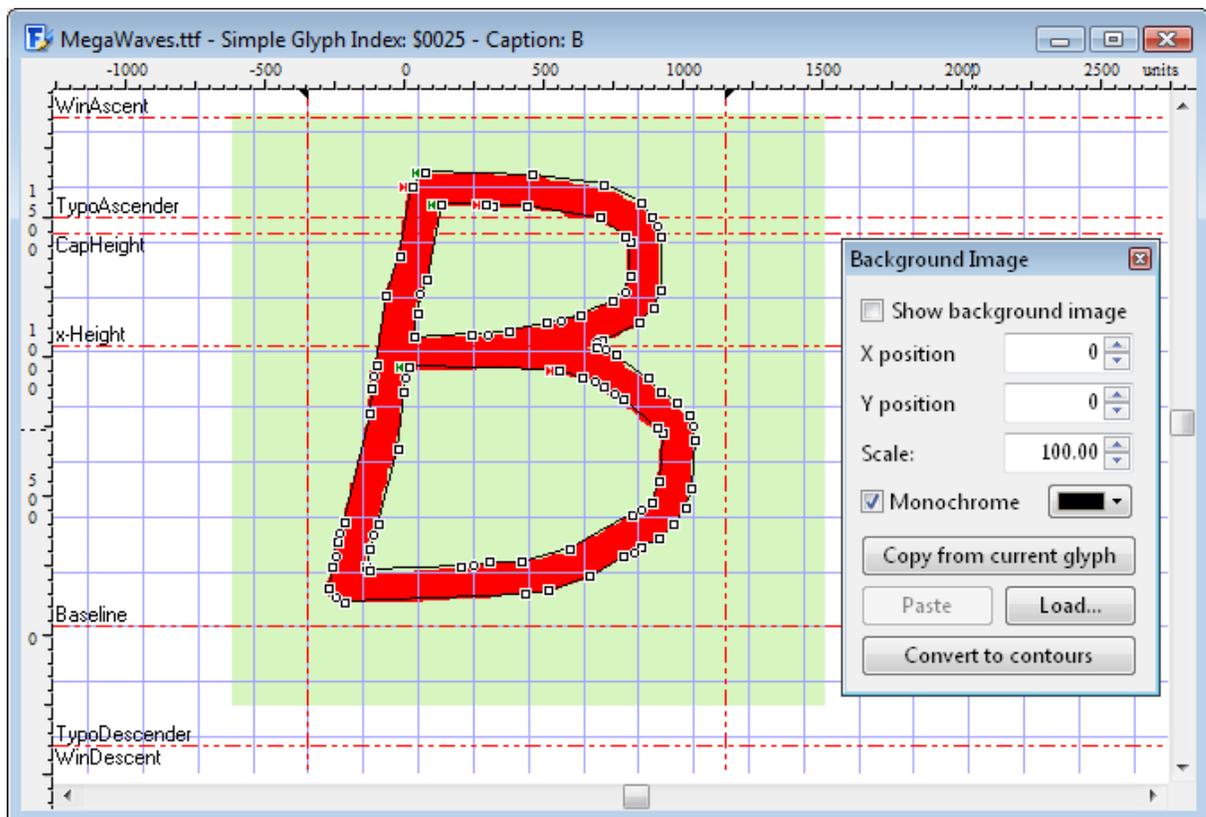
The **Preview** toolbar can be resized to show more text.

If you wish to edit a glyph's outline or adjust its metrics, click the specific glyph in the preview area to jump to it in the Font Overview or Glyph Edit window.

Note: If the Glyph Display Mode (available from the View menu) is set to Auto or Color, then the foreground and background color come from the Palette toolbar.

6.7 Background Image

You can add a background image on a **Glyph Edit** window through the **Background** toolbar. It can help you with your glyph design. It can be toggled on and off using the **F9** shortcut key, but is available only in the **Glyph Edit** window.

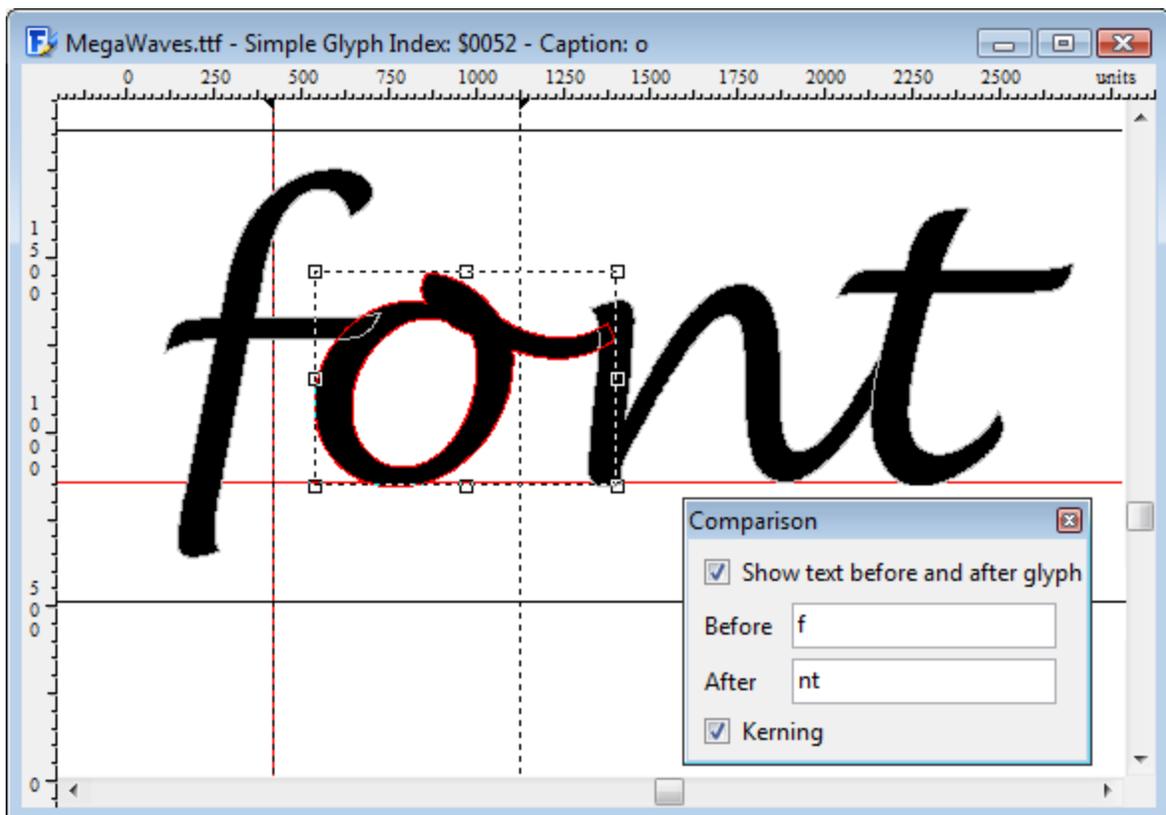


Click the **Copy from current glyph** button to place a copy of the current outline as background image. To add a background image paste an image from the clipboard, or click the **Load** button and select the image you want to use. Use the position and scale fields to move and scale the image. Check the **Monochrome** box, to show the background image in one color.

The **convert to contours** button converts the raster based image into vector based contour which will become part of the existing glyph outline.

6.8 Comparison

You can show glyphs (with kerning) before and after the current glyph in a **Glyph Edit** window through the **Comparison** toolbar. It can help you to position diacritics in composite glyphs or to accurately align flowing scripts.



Since the comparison also shows the active OpenType features as set in the Preview toolbar, it is also a powerful way to test and adjust glyphs in a specific context.

6.9 Samples

The **Samples** toolbar provides a powerful way of adding some contours you

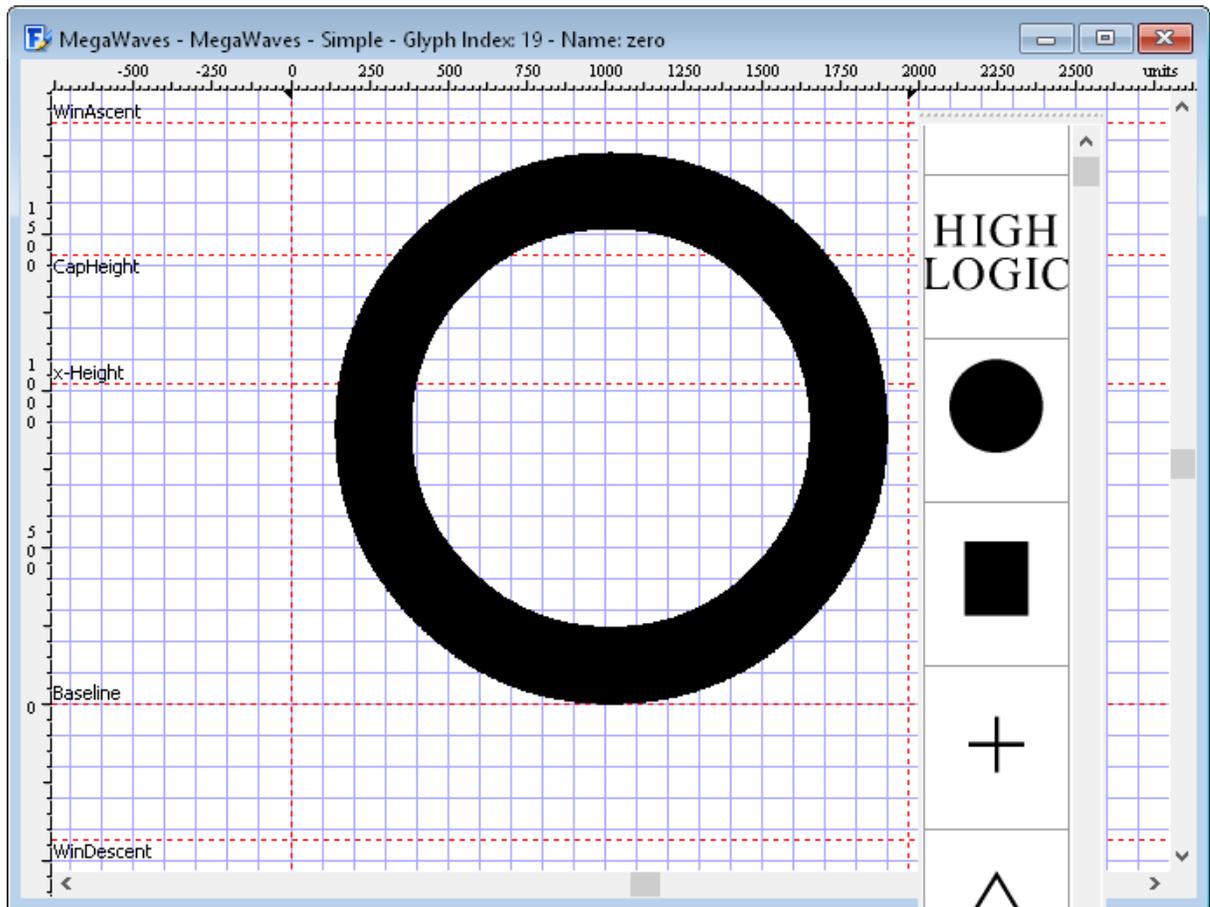
may frequently want to use in fonts or in glyphs.

You can use any of these glyphs in your own fonts. The outlines are royalty free. Credits go to:

Bhikkhu Pesala - <http://www.softerviews.org/Fonts.html>

With the **Samples** toolbar you can drag and drop a sample glyph to a cell on the **Font Overview** or **Glyph Edit** window. The toolbar can be docked or floating, and can be toggled on and off using the **F12** shortcut key.

Note: When you drop a sample onto a composite glyph on the Font Overview window it will be converted to a Simple Glyph first.

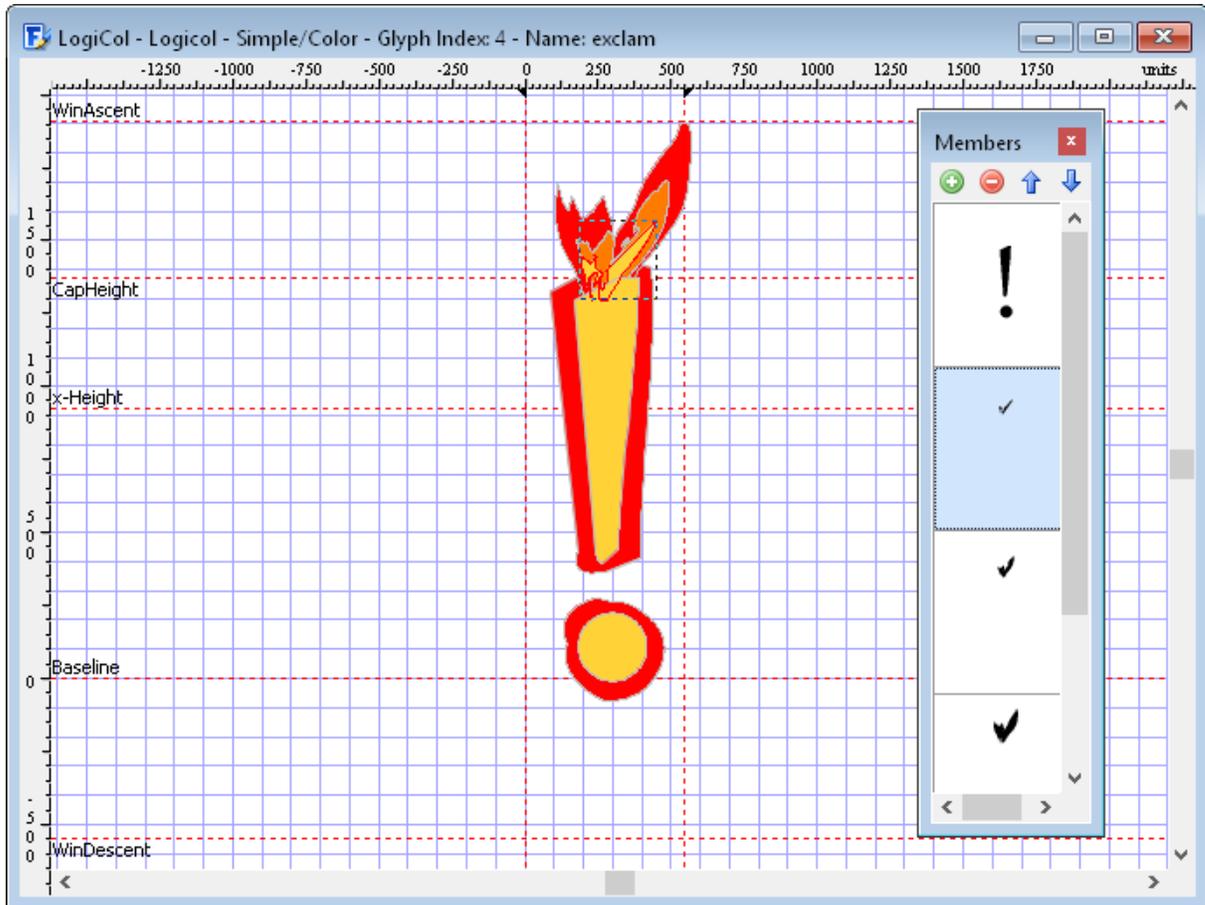


You can change the samples by making a special samples font, which includes your own samples. From the **Tools** menu you go to **Options** and go to the **Samples** page. Here you can select the font filename to be used in the **Samples**

toolbar.

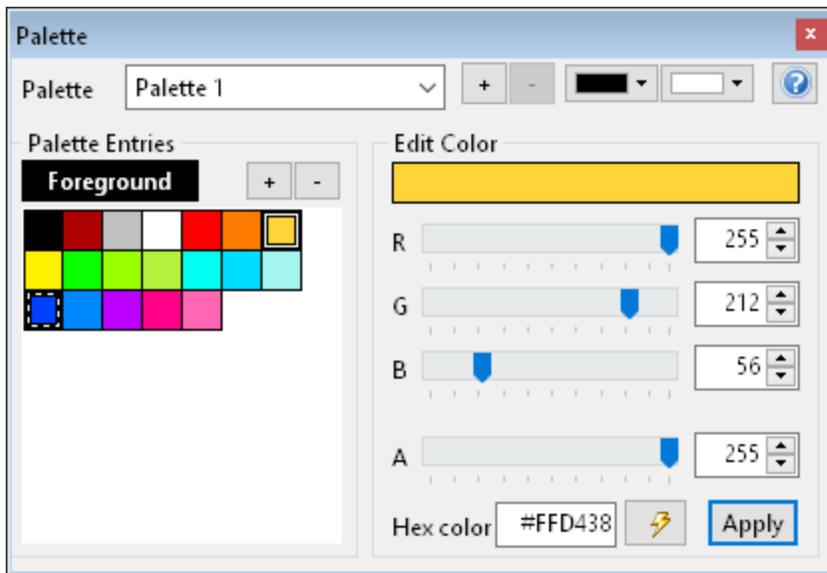
6.10 Color Glyph Members

The Color Glyph Members toolbar gives you a quick overview of the glyphs used in a color glyph and allows you to add and remove glyphs. When glyph members overlap then the order in which they appear becomes important. You can change the glyph member order by using the **Move Up** and **Move Down** buttons.



6.11 Palette

The Palette toolbar allows you to edit the palettes and the colors used in your font.



The palette drop down list shows the active palette. You can add and/or remove palettes by clicking the + and - buttons. Note that there must be at least one palette.

The foreground and background color selectors allow you to set the foreground and background colors that will be used in FontCreator's Glyph Edit windows, Preview toolbar and Font Overview. Note that each palette has its own fore- and background colors.

The Palette entries list the current available colors for the active palette. You can modify a color by selecting the color and using the sliders, edit fields or the color chooser. Use the **Apply** button to process the changes. You can use the + and - buttons to add or remove colors. Note that only unused colors can be removed.

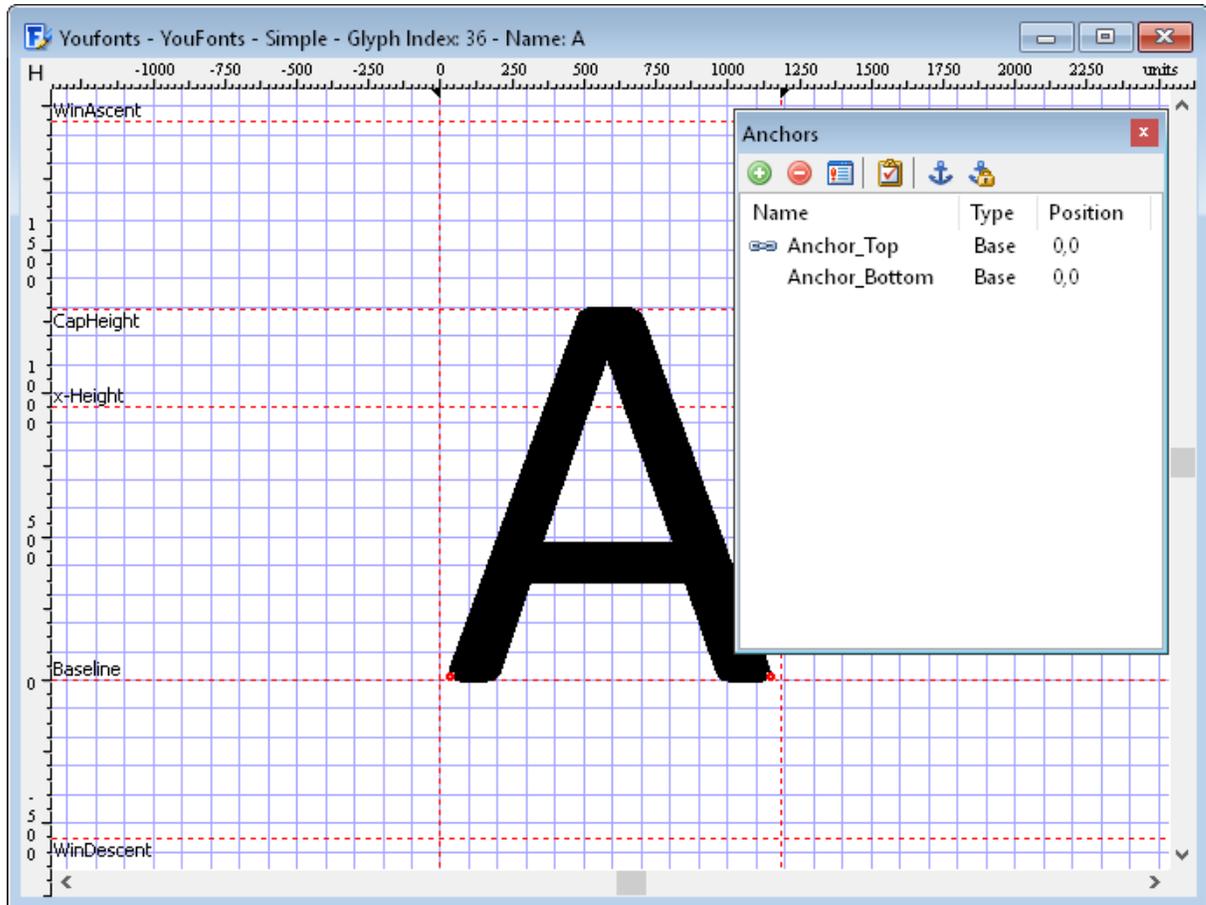
The **Foreground** palette color is a special color that depends on the font color that the user or host application has set as active font color.

6.12 Anchors

The Anchors toolbar gives you a quick overview of the anchors defined for a glyph. Anchors are used in the following OpenType lookups:

- Mark to Base
- Mark to Mark
- Mark To Ligature
- Cursive Attachments

With the first three lookups the anchors define where glyphs are stacked onto each other, and with cursive attachments the special Entry and Exit anchors are used to place two glyphs next to each other.



When an anchor is in use by a lookup it cannot be deleted, and a linked icon (☞) will appear in front of it. To edit an anchor you can drag it in the glyph edit window, click the "Edit" button on the toolwindow toolbar, or double-click on the entry in the list.

To add an anchor at a specific position, right-click inside the Glyph Edit window, and select Add Anchor.

Add Anchor

Add a new Anchor

Delete Anchor

Deletes the selected anchor (only available when anchor is not in use)

Edit Anchor

Edit selected anchor position

Anchor Manager

Add/Delete/Modify Anchor classes

Part

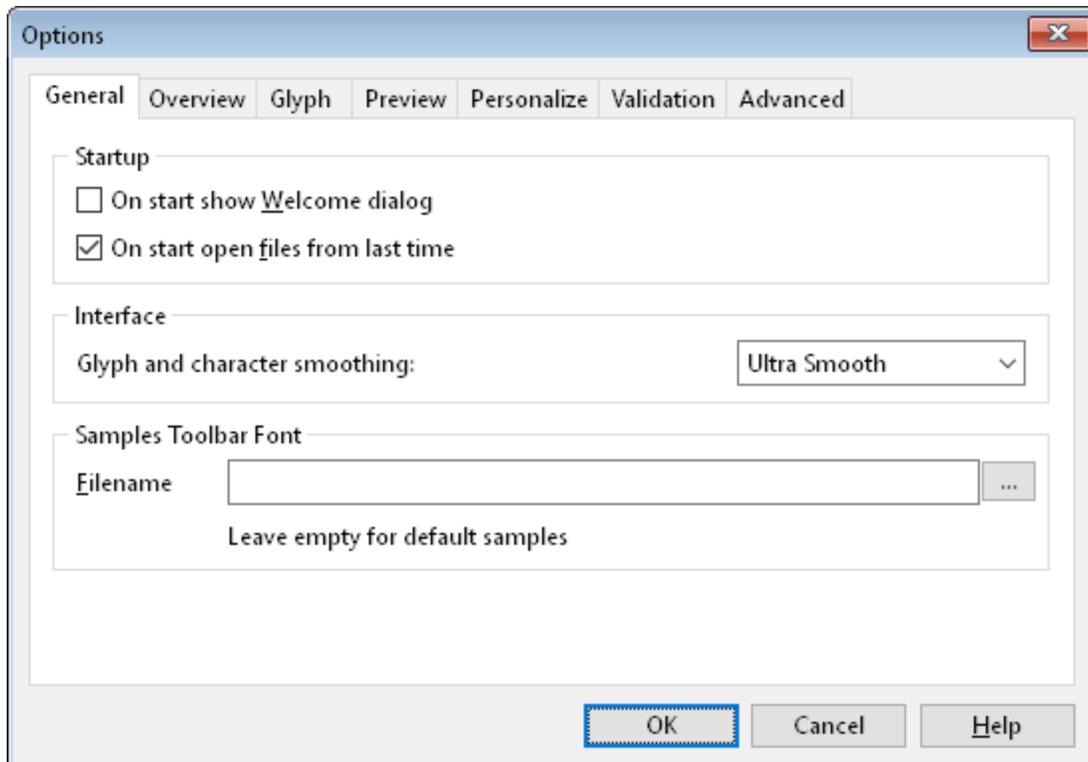
VII

7 Customizing FontCreator

7.1 Options

7.1.1 General

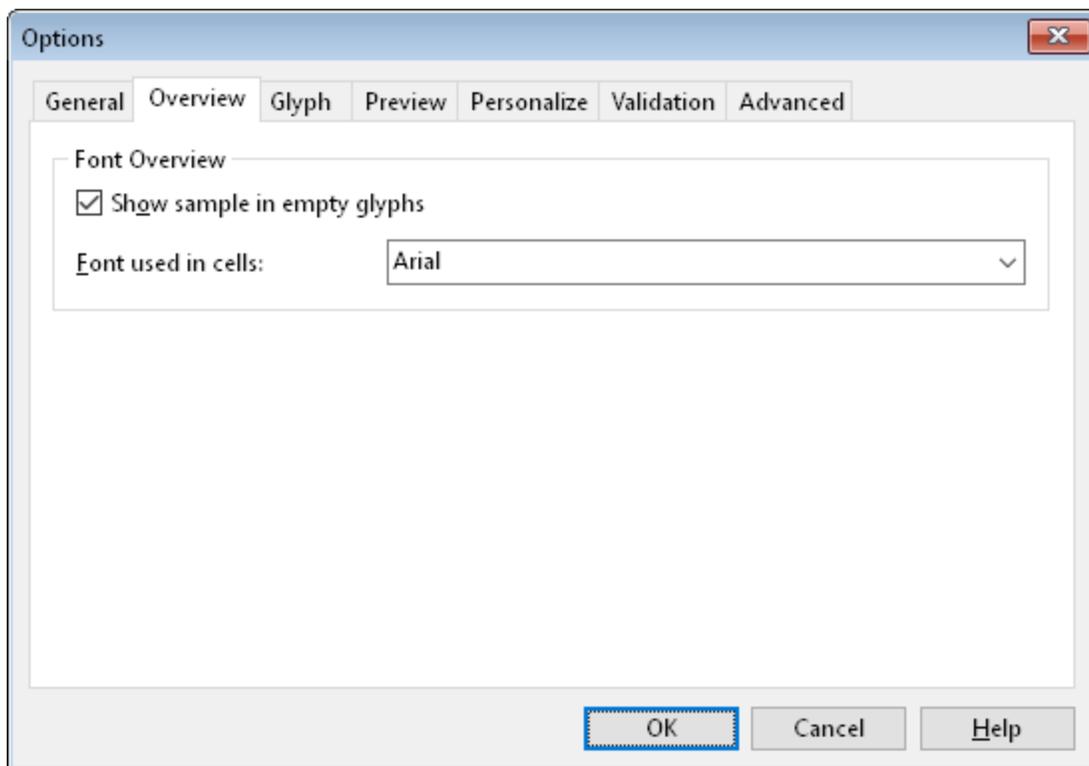
You can customize the way fonts are loaded, saved and shown with the **Options** window (Select **Options** from the **Tools** menu).



Use the fields on the **General** page when you want to change the welcome dialog, the interface settings or the **Sample Toolbar** font.

7.1.2 Overview

On the **Tools** menu, click **Options**, and then click the **Overview** tab. Here you can adjust the **Font Overview** settings.

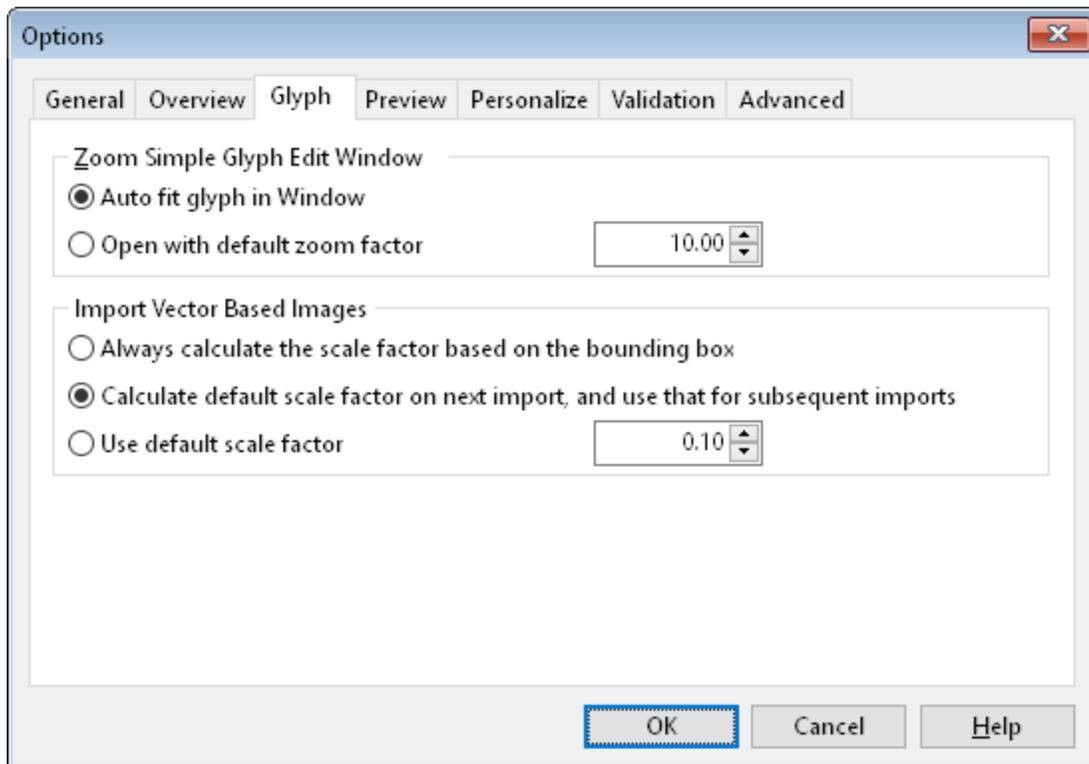


Show sample in empty glyphs

Will show a grey sample glyph for empty glyphs in the font overview. You can change the font to use by selecting one from the installed fonts list.

7.1.3 Glyph

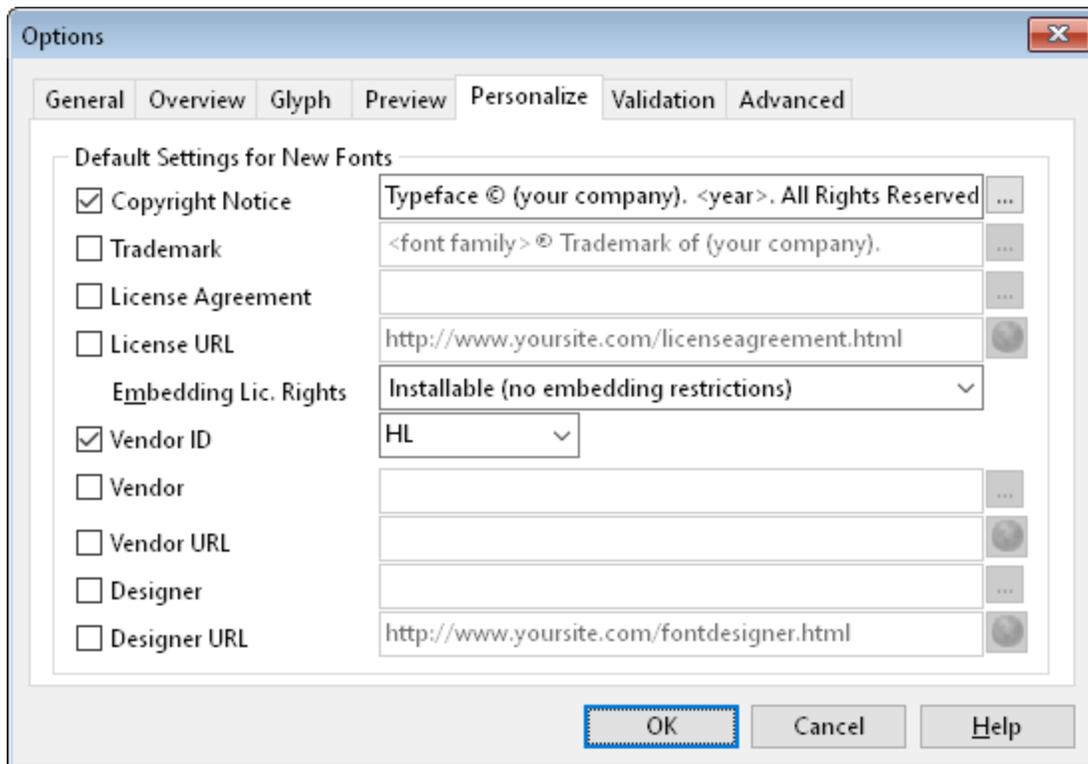
On the **Tools** menu, click **Options**, and then click the **Glyph** tab.



Here you can set the way a **Glyph Edit** window will show the glyph and how Vector Based Images are imported.

7.1.4 Preview

On the **Tools** menu, click **Options**, and then click the Preview tab.

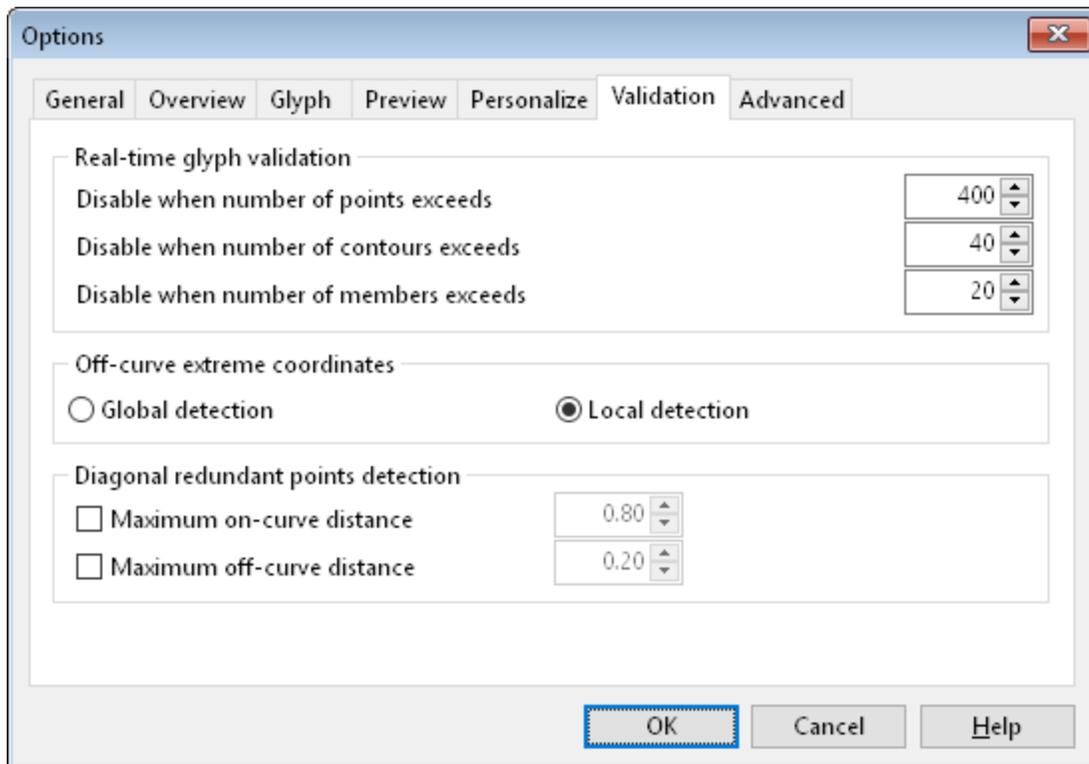


Default Naming for New Fonts

These fields are used as initial naming fields when a new font is created. <year> will be replaced by the current year and will be replaced by the actual font family name.

7.1.6 Validation

On the **Tools** menu, click **Options**, and then click the **Validation** tab.



The Real-time glyph validation settings are used to determine whether the real-time glyph problem detection should be disabled as it would be too time consuming. These settings are only used by the real-time glyph problem detection. When disabled, use the **Show glyph problem report** to open the **Glyph Problem Report window**.

Off-curve extreme coordinates detection can be performed globally or locally.

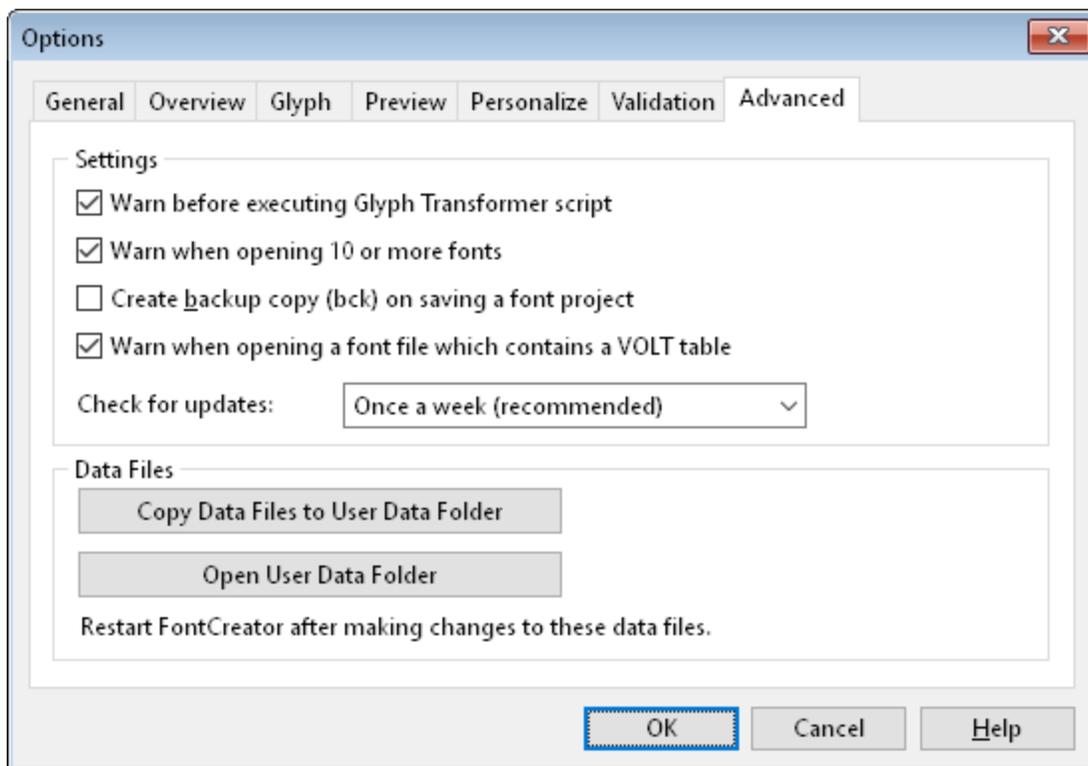
Note: The Validation features are not available in the Home Edition of FontCreator.

See also:

[Font Validation](#)

7.1.7 Advanced

On the **Tools** menu, click **Options**, and then click the **Advanced** tab.



Warn before executing Glyph Transformer script

This indicates and determines if a warning will be issued when the Transform wizard is finalized. If this is not checked, no warning will be given.

Warn when opening 10 or more fonts

This indicates and determines if a warning will be issued when opening 10 or more fonts. Opening a lot of fonts at the same time will require a lot of system resources.

Create backup copy (bck) on saving a font project

Copies the previous version of a font project as a backup copy every time you save a font project. Each new backup copy replaces the previous backup copy. FontCreator saves the backup copy (with a file name extension .bck) in the same folder as the original.

Check for updates

Set the interval that FontCreator uses to check for updates. An interval of Once a week will ensure you are always working with the latest version of FontCreator.

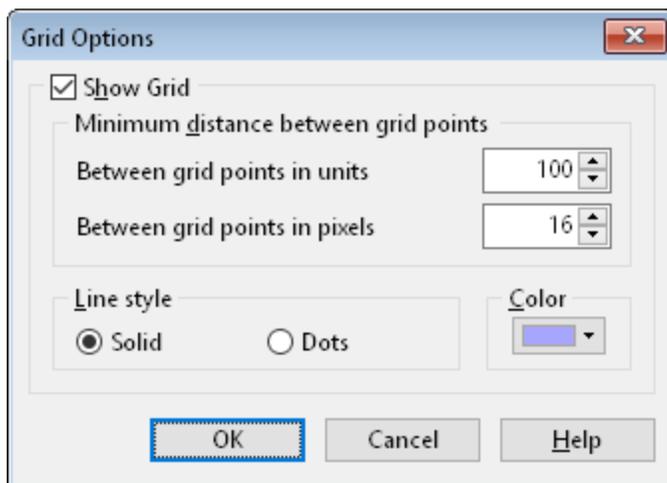
Data Files

FontCreator uses several data files for advanced settings and customizations.

Normally these files are stored in a system folder where they cannot be changed. If you want to edit these data files they must first be copied to your user data folder. First click "**Copy Data Files to User Data Folder**" and then "**Open User Data Folder**" to start editing the files. See [FontCreator data files](#) for more information.

7.2 Grid Options

You can adjust the way the grid in the **Glyph Edit** window is drawn in the **Grid Options** window (select **Grid Options** from the **Tools** menu).



Here you can change the minimum distance, color and style.

The **Snap to Grid** function automatically places selected glyphs, contours and points along the grid. When you release a selection, FontCreator moves it until the edges are aligned with the nearest grid lines.

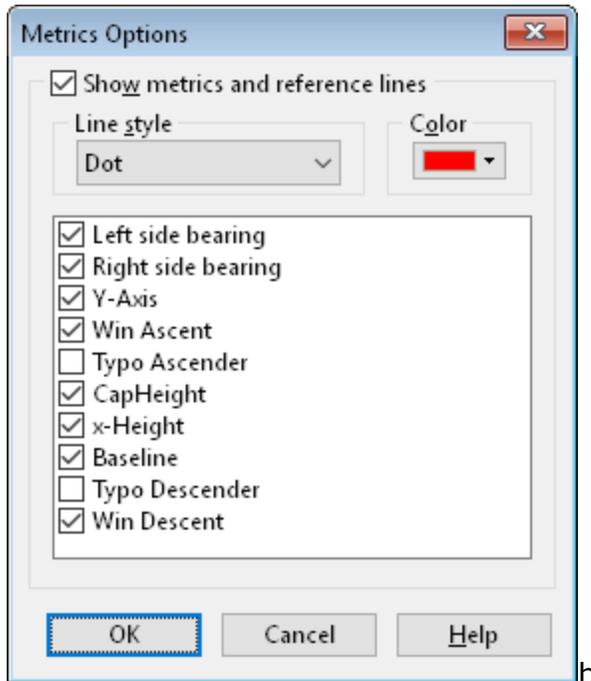
You must choose the Show Grid command before you can use the Snap to Grid function. To activate the **Snap to Grid** function, click the **Snap to Grid** button on the toolbar or choose **Snap to Grid** from the **View** menu.

When the Snap to Grid function is active, its button on the menu and toolbar appears pressed in. Choose the command or click the button again to turn off the function.

Note: Grid options are project specific. This means you can have different grid settings for each of your projects.

7.3 Metrics Options

To open the **Metrics Options** window click **Metrics Options** on the **Tools** menu. Here you can specify what metrics and reference lines should be shown in the Glyph Edit window.



Left side bearing	The space on the left side of the glyph that is part of the advance width
Right side bearing	The space on the right side of the glyph that is part of the advance width
Y-Axis	The vertical line where $x=0$.
Win Ascent	The ascender metric for Windows
Typo Ascender	The typographic ascender for the font
CapHeight	This metric specifies the distance between the baseline and the approximate height of uppercase letters
x-Height	This metric specifies the distance between the baseline and the approximate height of non-ascending lowercase letters
Baseline	The imaginary line upon which the letters of the font rest. This is always the X-Axis.
Typo Descender	The typographic ascender for the font
Win Descent	The descender metric for Windows

You can change the position of several of these font metrics through the Metrics tab on the Font Settings dialog. From the main menu select the Format menu, click Settings, and then click the Metrics tab.

Note: Metrics options are project specific. This means you can have different metrics settings for each of your projects.

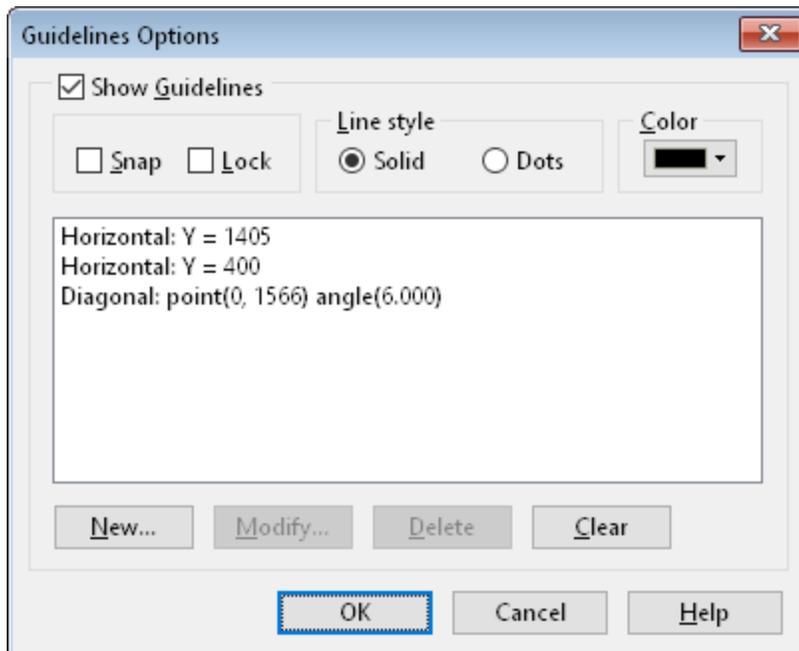
See also:

[Glyph Metrics](#)

7.4 Guidelines

7.4.1 Options

To open the **Guidelines Options** window click **Guidelines Options** on the **Tools** menu. The **Guidelines Options** window is also activated when you double-click the rulers in the **Glyph Edit** window.



Here you can show and hide the guidelines, and change the guidelines style and color.

The **Snap to Guidelines** function automatically places selected glyphs, contours and points along the guidelines. When you release a selection, the FontCreator moves it until the edges are aligned with the nearest **horizontal** or **vertical** guidelines. It is also possible to snap points to **diagonal** guidelines. You must choose the Show Guidelines command before you can use the Snap to

Guidelines function. To activate the **Snap to Guidelines** function, select the Snap check box, click the Snap to Guidelines button on the toolbar or choose Snap to Guidelines from the **View** menu. Select the **Lock** check box, or select the **Lock Guidelines** button on the toolbar, to prevent accidental movement of the guidelines.

Press the **New** button to define a new guideline. Press the **Modify** button to modify an existing guideline. To delete an existing guideline select it from the list view and click the **Delete** button. Use the **Clear** button to remove all guidelines.

If you want to add a horizontal or vertical guideline you can drag one from the top or left-hand ruler in the **Glyph Edit** window. Hold down the left mouse button and release it after you have moved the pointer to the desired position. To remove a guideline, simply drag it back to the ruler.

In order to rotate a guideline, press and hold down the Shift key before moving the guideline. Rotating a horizontal or vertical guideline will change the guideline into a diagonal guideline.

Select one or more contours (or one or more composite glyph members) and right-click and then select **Add Bounding Guidelines** to add two horizontal and two vertical guidelines that correspond to the selection bounding box.

To copy a guideline to a new position, hold down the Ctrl key as you drag the guideline.

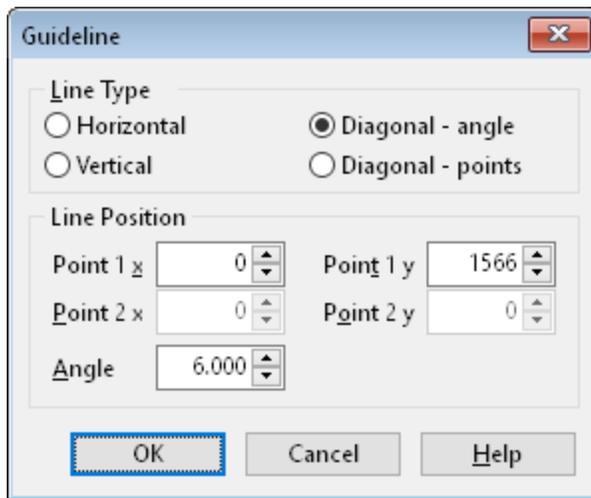
Tip: Select one point in a **Glyph Edit** window and press the **G** key on your keyboard to add a horizontal and vertical guideline that goes through the selected point.

Tip: Select two points in a **Glyph Edit** window and press the **G** key on your keyboard to add a guideline that goes through the selected points.

Note: User defined guidelines are project specific. This means you can have different user defined guidelines for each of your projects.

7.4.2 Guideline

Use the **Guideline** window to create or modify guidelines.



Line Type

Choose between horizontal and vertical lines and two diagonal line types.

Line Position

Both horizontal and vertical lines only need one value.

A diagonal line needs either one point with an angle or two points.

Tip: The **Guideline** window is activated when you double-click a guideline in the **Glyph Edit** window.

7.5 Keyboard Shortcuts

The following keyboard shortcuts can be used to quickly accomplish frequent tasks:

General Shortcuts

Press	To
Ctrl+Tab (or Ctrl+F6)	Next Window
Ctrl+Shift+Tab	Previous Window
F1	Help
F2	Show/Hide User notes Toolbar
Shift+F2	Show/Hide Palette Toolbar

F3	Glyph Properties Toolbar
Shift+F3	Show/Hide Color Glyph Member Toolbar
F4	Show/Hide Statusbar
Ctrl+F4	Close Window
F5	Test Font as TTF/OTF
Ctrl+F5	Test Font as WOFF
F6	Show/Hide Transform Toolbar
Ctrl-F6 (or Ctrl+Tab)	Next Window
F7	Show/Hide Glyph Validation Toolbar *
F8	Show/Hide Preview Toolbar
Ctrl-F8	Show OpenType Designer/ Edit Lookup in Code Editor
F9	Show/Hide Background Image Toolbar
F10	Activates the Main Menu
Shift-F10	Shows the context menu
Ctrl-F10	Launch MainType
F11	Show/Hide Comparison Toolbar
Ctrl-F11	Launch Windows Font Folder
F12	Show/Hide Samples Toolbar
Ctrl-F12	Launch Windows Character Map
Delete	Delete selection
Ctrl-A	Select All
Ctrl-C	Copy
Ctrl-E	Paste Special
Ctrl-F	Displays the Find dialog box
Ctrl-H	Toggle Fill Outlines
Ctrl-I	Open Installed Fonts
Ctrl-J	Display values as Hexadecimal
Ctrl-K	Display values as Decimal

Ctrl-N	New Project
Ctrl-O	Opens Font or Font Project
Ctrl-P	Print Glyph, Font, or Font Properties
Ctrl-R	Repeat last action
Ctrl-S	Save Project
Ctrl-T	Invert Selection
Ctrl-U	Clear Selection
Ctrl-V	Paste
Ctrl-X	Cut
Ctrl-W	Close current window
Ctrl-Y	Redo
Ctrl-Z	Undo
Ctrl-Shift-E	Export font as TrueType/OpenType font
Ctrl-Shift-W	Export font as Web Open Font Format (WOFF) font
Ctrl-Shift-A	Export font in all formats

Keys for working in the Font Overview window

Press	To
Enter	Open Glyph Edit window
Backspace	Clear selected glyphs
P	The first 256 characters mapped to the selected glyphs will be placed into the preview toolbar
Ctrl + Plus Sign	Increase the cell size
Ctrl + Minus Sign	Decrease the cell size
Ctrl + Shift + Plus Sign	Increase character size
Ctrl + Shift + Minus Sign	Decrease character size
Ctrl + 0	Reset character and cell size to their defaults
Ctrl + L	Reset category panel to default width

Keys for working in the Glyph Edit window

Press	To
Plus Sign or Ctrl+Plus Sign	Increase the zoom factor
Minus Sign or Ctrl+Minus Sign	Decrease the zoom factor
Ctrl + 0	Reset zoom factor to default
Alt+Left Arrow	Go to previous glyph
Alt+Right Arrow	Go to next glyph
N	Change selected points to on curve
F	Change selected points to off curve
G	Add guideline through two selected nodes, or add orthogonal guidelines at a single selected node
A	Add points after selected points
O	Toggle Fill Outline
H	While pressed down, hides everything except the glyph
Q	Select previous contour/point index or composite glyph member
W	Select next contour/point index or composite glyph member
P	Toggle between point and contour mode
C	Toggle between standard and color mode
K	Knife (not in color mode)
M	Measure
B	Paint bucket (only in color mode)
/	Fit to Window
Space+Left mouse button	Pan around the Glyph Edit window
Double-click	While editing an empty or simple glyph, this will switch contour/point mode While editing a composite glyph, this will open the

	composite glyph member properties window
Double-click on rulers	Edit guidelines
Double-click on guideline	Adjust guideline
Shift-click on guideline	Rotate guideline
Ctrl+Drag guideline	Duplicate guideline
Click on selected contour	show/hide rotate and skew handles
Shift+Mouse	Constrain movement, scaling, skew, and rotation
Ctrl+Mouse	Duplicate selected contour

You can reset all toolbar positions and sizes by pressing the Ctrl key while starting FontCreator.

* Not available in the Home Edition of FontCreator.

7.6 FontCreator Data Files

FontCreator uses several extra data files to read settings and other program specific information. These should not be edited directly, but copied to the windows user data folder. This can be easily done through the **Settings** dialog on the **Advanced** tab.

Unicode/Blocks.txt	Standard Unicode database file containing the Unicode blocks
Unicode/UnicodeData.txt	Standard Unicode database file containing all the Unicode glyph information
glyphlist.dat	The Adobe glyph list
tags.txt	Controls the names of the five tags that can be assigned to each glyph
preview.txt	Contains the standard preview texts for the Preview toolbar
glyphnames.dat	Contains the default glyph names when opening existing fonts and when you generate glyph names on the Glyph Properties dialog

Note: FontCreator only reads these files at startup. This means that after you

have changed any of the files, you must restart FontCreator for the changes to take effect.

Index

- A -

Advance Width 111, 135
 Advanced 155
 Anchors 94, 95, 96, 146
 Appearance 75
 Ascent 67
 AutoKern 114
 Autokerning 90
 AutoMetrics 113

- B -

Background Image 142
 Backup 155
 Baseline 50, 158
 Bearings 50
 Bitmap to Outline 116

- C -

CapHeight 50, 158
 Caption 25
 Character 31
 Character Map 123, 131
 Character Ranges 70
 Characteristics 74
 Characters 27
 Insert 27
 CharMap 123, 131
 CJK Fonts 61
 Class Manager 87
 Classes 87
 ClearType 75
 Close Font 22
 Code Page Character Ranges 72
 Color 14
 Glyphs 51
 Commands 134
 Comparison 143
 Complete Composites 50
 Composite Glyph 46
 Composites 50

Contour 37, 38, 41
 Direction 38
 Insert 8
 Mode 38
 Convert 116, 118
 Composite Glyph 46
 Copy Glyph 30
 Copyright 15, 153
 Copyright Notice 59
 Create Font 19
 credits 4
 Curve 38
 Custom naming fields 61
 Customize 150

- D -

Descent 67
 Designer 59, 153
 Direction 8, 138
 Duplicate 138

- E -

Ellipsis 34
 Embedding 32
 Empty Glyph 38
 Empty Glyphs 150
 Euro Sign 31
 Exclusion 44
 Export
 Settings 20
 External Tools 123
 Extreme 138

- F -

Family 54
 Features 2
 File Name 17, 32
 Find 24
 Flags 65
 Font
 Information 77
 Table 79
 Font Header Flags 62, 65
 Font matching 74

Font Name 32
 Font overview 25, 150
 Font Style 56
 Font Type 32
 FontCreator
 buying 4
 editions 4
 getting help 3
 manual 3
 Fonts
 Color 14
 Fonts folder 123
 Freedraw 7, 45
 Full Font Name 54

- G -

Gasp 75
 Generate Glyph Names 122
 Glyph 37
 Composite 37
 Empty 37
 Insert 29
 Mappings 135
 Properties 135
 Simple 37
 Type 25, 37
 Glyph Member 48, 52
 Glyph Name 135
 Glyph Transformer 109
 Grayscale 75
 Grayscale Rendering 76
 Grid Options 157
 Gridfitting 76
 Guideline 160
 Guidelines Options 159

- H -

Handwriting 7
 Hinting 79

- I -

Image to Outline 116, 118
 Import 151
 Import Image 116, 118

Install 10
 Installing Fonts 130
 International 127
 Intersection 44, 138
 Italic Angle 54

- J -

Join Contours 44

- K -

Kerning
 Automatic 116
 Kerning Pairs 92
 Keyboard 127
 Keystroke 131
 Knife 44

- L -

Language 61, 127
 Layout 127
 Left Side Bearing 50, 111, 135, 158
 Legal 59
 License Agreement 59, 153

- M -

Mailing List 2
 Manual
 PDF 3
 Mappings 137
 Marks 94
 Menu 134
 Metrics 48, 50
 Automatic 113
 Metrics Options 158
 Monospaced 33

- N -

Naming 153
 New Features 3

- O -

Open Font 17
 OpenType 14
 OpenType Designer 83
 Autokerning 90
 Settings 89
 OpenType Features
 Cursive Attachment 95
 Mark to Base 94
 Mark to Ligature 94
 Mark to Mark 94
 OpenType Layout Features 80, 100
 Alternates 91
 Anchor Manager 96
 Basics 101, 104
 Chained Context 97
 Comments 101
 Examples 109
 Feature 101
 Group 102
 Language 102
 Ligatures 91
 Lookup 103
 LookupFlags 104
 Multiple substitution 91
 Pair Adjustment 92
 Pos 106
 Script 105
 Single Adjustment 91
 Single substitution 91
 Sub 105
 Substitutions 91
 Supported substitutions 81

- P -

Palette 51
 Colors 145
 Panose 74
 Paste Glyph 30
 Personalize 153
 Point 43, 53
 Mode 38
 PostScript Name 54
 Preview 141, 145

Print
 Font 123
 Glyph 124
 Project
 Close 22
 File 17
 Open 17
 Save 20
 Proportional 33

- Q -

Quotation Mark 34
 Quotes 34

- R -

Raster Image 116
 Redo 24
 Redundant 138
 Register 4
 Repeat 24
 Revision 54
 Right Side Bearing 50, 135, 158
 Rotate 41

- S -

Samples 143, 150
 Samples font 150
 Save Font 20
 Scanner 9
 Search 24
 Select 37
 Shortcut 161
 Simple Glyph 38
 Smoothing 75, 76
 Sort 121
 Split Contours 44
 Strikeout 62
 SubFamily 54
 Subscript 62
 SubTable Manager 88
 SubTables 88
 Support 2
 Symbol 33, 72

- T -

Table 79
 Ordering 79
Tags 25
Template 19
Test 10
Test Fonts 128, 129
TextSamples
 Add 129
 Delete 129
 Edit 129
Toolbars 134
Trademark 59, 153
Transform 109, 135
TrueType 14
TrueType Collection 122
TTC 122
Typo Ascender 50
Typo Descender 50

- U -

Underline 62
Undo 24
Unicode 33
Unicode Character Ranges 72
Union 44
Unique Font Identifier 54
Units Per Em 62
Updates 155
Used Glyph 26
User Notes 134

- V -

Validation 124, 138, 154
 Results 127
Vector 151
Vector Image 118
Vector to Outline 118
Vendor 59
Version 54
VOLT 80, 83, 100

- W -

Warning Points 138
Warnings 155
Web OpenType Font Format 15
Weight 54
Width 54
Win Ascent 50, 158
Win Descent 50, 158
Wizard
 AutoMetrics 113
WOFF 15
WWS 56

- X -

X-Height 50, 158

- Z -

Zoom 37, 151